PENGONTROL RUMAH JAMUR JARAK JAUH DI LENGKAPI DENGAN PERINGATAN SMS BERBASIS MIKROKONTROLLER RENESAS R8C/TINY

SKRIPSI

Diajukan Untuk Memenuhi Salah Satu Syarat Memperoleh Gelar Sarjana Teknik
Pada Jurusan Teknik Elektro S-1 Konsentrasi Elektronika

Oleh :

BAMBANG WISWANTO
02.17.104

KONSENTRASI ELEKTRONIKA
JURUSAN TEKNIK ELEKTRO S-1
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2007

[illegible]

[illegible]

[illegible]

LEMBAR PERSETUJUAN

PENGONTROL RUMAH JAMUR JARAK JAUH DI LENGKAPI DENGAN PERINGATAN SMS BERBASIS MIKROKONTROLLER RENESAS R8C/TINY

SKRIPSI

Diajukan Untuk Memenuhi Salah Satu Syarat Memperoleh Gelar Sarjana Teknik
Pada Jurusan Teknik Elektro S-1 Konsentrasi Elektronika

Oleh :

BAMBANG WISWANTO

02.17.104

Mengetahui
Ketua Jurusan Teknik Elektro S-1

Ir. R. Yudi Limpraptono, MT
NIP. 1039500274

Dosen Pembimbing

Joseph Dedy Irawan, ST, MT
NIP. 132315178

KONSENTRASI ELEKTRONIKA

JURUSAN TEKNIK ELEKTRO S-1

FAKULTAS TEKNOLOGI INDUSTRI

INSTITUT TEKNOLOGI NASIONAL MALANG

2007

INSTITUT TEKNOLOGI NASIONAL
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO S-1
Jl. Karanglo KM. 2 Malang

BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI

Nama	: Bambang Wiswanto
NIM	: 02.17.104
Jurusan	: Teknik Elektro S-1
Konsentrasi	: Teknik Elektronika
Masa Bimbingan	: 12 Januari 2007 – 12 Juli 2007
Judul Skripsi	: Pengontrol Rumah Jamur Jarak Jauh Di Lengkapi Dengan Peringatan SMS Berbasis Mikrokontroller Renesas R8C/Tiny

Dipertahankan di hadapan Tim Penguji Skripsi Jenjang Strata Satu (S-1) pada :

Hari	: Jum'at
Tanggal	: 23 Maret 2007
Nilai	: 83 (A)

PANITIA UJIAN SKRIPSI

KETUA	SEKRETARIS
(Ir. Mochtar Asroni, MSME)	(Ir. F. Yudi Limpraptono, MT)
NIP.Y. 1018100036	NIP.Y. 1039500274

ANGGOTA PENGUJI

PENGUJI I	PENGUJI II
(Sotyohadi, ST, MSc)	(M. Ashar, ST, MT)
NIP.	NIP.

ABSTRAKSI

PENGONTROL RUMAH JAMUR JARAK JAUH DI LENGKAI DENGAN PERINGATAN SMS BERBASIS MIKROKONTROLLER RENESAS R8C/TINY

(Bambang Wiswanto, 02.17.104, Teknik Elektro S-1/Elektronika)
(Dosen Pembimbing : Joseph Dedy Irawan, ST, MT)

Kata Kunci *: Renesas R8C/Tiny, RS 485, RS 232, CPU, HP Siemens, Relay 12V,LM 35, HS15P, BD400, LM358, Blower, Hairdryer.*

Sistem telekontrol secara konvensinal terdiri dari sensor, pemroses data, komunikasi serial, PC, HP Siemens, penghasil kelembaban, dan pemanas sebagai pengatur sikulasi udara. Sensor berfungsi sebagai pendeteksi udara dalam rumah jamur. Kemudian di inputkan ke MCU *(Mikrokontroller unit)*,MCU mengirimkan ke PC *(Personal Computer)* melalui RS485 dan RS232.PC akan memproses semua inputan dari MCU dan akan ditampilkan table/grafik.

Dengan kemampuan komunikasi serial (RS485) agar dapat dikontrol dengan jangkauan maksimal 1,2 Km, dengan fasilitas SMS *(Short Message Service)* dapat digunakan untuk pemberitahuan/peringatan dlam keadaan ekstrim. Aplikasi mikrokontroller Renesas R8C/Tiny pada telekontrol digunakan sebagai kendali system.

KATA PENGANTAR

Alhamdulillah, puji syukur kehadirat-Mu Ya Allah yang telah memberikan rahmat dan hidayah-Nya, sehingga penyusun dapat menyelesaikan skripsi ini dengan judul "Pengontrol Rumah Jamur Jarak Jauh Di Lengkapi Dengan Peringatan SMS Berbasis Mikrokontroller Renesas R8C/Tiny". Skripsi ini merupakan persyaratan kelulusan Studi di Jurusan Teknik Elektro S-1 Konsentrasi Teknik Elektronika ITN Malang dan untuk mencapai gelar Sarjana Teknik.

Keberhasilan peyelesaian laporan skripsi ini tidak lepas dari dukungan dan bantuan berbagai pihak. Untuk itu penyusun menyampaikan terima kasih kepada :

1. Bapak Ir. F. Yudi Limpraptono, MT selaku Ketua Jurusan Teknik Elektro S-1.
2. Bapak Joseph Dedy Irawan, ST, MT selaku Dosen Pembimbing .
3. Ayah dan Ibu serta saudara-saudara kami yang telah memberikan do'a restu, dorongan, semangat dan biaya.
4. Rekan-rekan Instruktur di Laboratorium Elektronika.
5. Teman-teman ELKA-4 yang tak henti-hentinya memberikan semangat dan motivasi dalam penyelesaian penyusunan skripsi ini.
6. Semua yang telah membantu dalam penyelesaian penyusunan skripsi ini.

Penyusun telah berusaha semaksimal mungkin dan meyadari sepenuhnya akan keterbatasan pengetahuan dalam menyelasaikan laporan ini. Untuk itu penyusun mengharapkan saran dan kritik yang membangun dari pembaca demi kesempurnaan laporan ini.

Harapan penyusun semoga laporan skripsi ini memberikan manfaat bagi perkembangan ilmu pengetahuan dan pembaca.

Malang, Maret 2007

Penyusun

DAFTAR ISI

DAFTAR TABEL

DAFTAR GAMBAR

BAB 1

PENDAHULUAN

1.1. Latar Belakang

Pengaplikasian teknologi dimanfaatkan untuk menciptakan alat-alat yang dapat berguna bagi kehidupan manusia, misalnya dalam bidang pertanian seperti budidaya jamur. Kita tahu bahwa jamur merupakan sumber protein nabati yang diperlukan tubuh. Jamur sendiri membutuhkan lingkungan yang mendukung agar bisa tumbuh dengan baik terutama masalah suhu / temperatur dan kelembaban. Saat ini kebanyakan para petani jamur masih menggunakan sistem manual untuk menjaga suhu / temperatur dan kelembaban udara. Artinya para petani harus setiap saat berada di rumah jamurnya untuk menjaga agar temperatur dan kelembaban udara di rumah jamur tersebut tetap berada pada kondisi yang diperlukan jamur untuk tumbuh dengan baik. Apabila terdapat beberapa rumah jamur, maka petani jamur harus berjalan ke masing-masing rumah jamur untuk mengukur dan menjaga temperatur dan kelembaban agar tetap berada pada kondisi yang diinginkan. Dimana hal ini harus dilakukan secara periodik untuk menjaga kualitas pertumbuhan jamur, dikarenakan temperatur dan kelembaban sangat mempengaruhi kelangsungan hidup jamur. Hal ini akan sangat melelahkan para petani dan menyebabkan kurang efektif dan efisien.

Untuk mengatasi hal tersebut diperlukan teknologi yang mampu menyediakan sarana komunikasi jarak jauh yang tepat dan efisien. Dengan menggunakan sistem telekontrol para petani jamur tidak perlu berjalan ke rumah

jamur untuk mengetahui berapa tingkat temperatur dan kelembaban pada masing-masing rumah jamur dan berupaya melakukan aktivitas untuk menyesuaikan tingkat kelembabannya. Para petani cukup mengawasi melalui sebuah komputer yang terdapat dalam rumah petani dan aktivitas penyesuaian temperatur dan kelembaban dapat dilakukan secara otomatis melalui PC (Personal Computer) untuk menggerakkan mekaniknya. Selain itu jika terjadi kondisi ekstrim (kelembaban dan suhu tidak pada range yang telah ditentukan) dan petani tersebut sedang tidak berada di tempat, secara otomatis PC akan mengirimkan pesan ke handphone melalui SMS (Short Message Service).

1.2 Rumusan Masalah

Dengan memperhatikan uraian latar belakang di atas, maka rumusan masalah akan ditekankan pada :

1. Bagaimana merancang dan membuat alat kontrol temperatur dan kelembaban dengan menggunakan pengontrol mikrokontroller.
2. Bagaimana merancang dan membuat perangkat lunak pada PC untuk memonitor dan memberi perintah alat kontrol temperatur dan kelembaban.
3. Bagaimana merancang dan membuat protokol komunikasi data antara PC dan mikrokontroler.
4. Bagaimana merancang dan membuat protokol komunikasi data antara PC dan HP (Handphone).

1.3 Tujuan

Tujuan pembuatan alat simulasi Telekontrol Rumah Jamur ini adalah membuat kemudahan pemantauan temperatur dan kelembaban pada rumah jamur yg dapat di kontrol secara otomatis.

1.4 Batasan Masalah

Untuk menghindari meluasnya masalah, maka dalam pembahasan akan lebih ditekankan pada :

1. Pengiriman pesan ke HP pemilik jamur hanya berupa pemberitahuan kondisi apabila keadaan temperatur dan kelembaban berada di luar range yang ditentukan.
2. Tidak membahas jaringan GSM (Global System Mobile).
3. Tidak membahas tentang sistem internal dari telepon seluler.
4. Hanya membahas spesifikasi jamur secara umum.
5. Hanya mengontrol suhu dan kelembaban.
6. Tidak membahas catu daya.

1.5 Metodologi Penulisan

Beberapa tahapan yang digunakan dalam penyelesaian laporan akhir ini adalah sebagai berikut :

1. Studi Literatur

   Studi literatur ini bertujuan untuk mencari landasan teori mengenai sistem telekontrol.

2. Perancangan dan Pembuatan Program

   Perancangan dan pembuatan program ini membahas tentang bagaimana cara mengirim data yang terdapat pada sensor ke komputer dan dapat dibaca secara tepat serta mengontrol motor untuk membuat penyesuaian temperatur dan kelembaban.

3. Pengujian dan Analisa

   Dilakukan untuk mengetahui apakah sistem telah berfungsi dengan optimal, jika masih terdapat kesalahan dapat diketahui penyebabnya serta dilakukan perbaikan sehingga program dan sistem dapat berfungsi optimal.

1.6 Sistematika Pembahasan

Sistematika pembahasan Laporan Akhir Telekontrol Rumah Jamur Berbasis MCU Yang Dapat Diakses Melalui HP, perinciannya sebagai berikut :

1) BAB I PENDAHULUAN

Bab ini membahas tentang hal-hal yang menjadi latar belakang, tujuan, rumusan masalah, dan metodologi serta sistematika pembahasan.

2) BAB II LANDASAN TEORI

Bab ini menguraikan teori-teori yang mendasari dan mendukung dalam perencanaan serta pembuatan hardware dan software seperti pengetahuan dasar MCU R8C, temperature dan kelembaban jamur, RS 232, RS 485, Borland Delphi.

3) BAB III PERENCANAAN DAN PEMBUATAN ALAT

Bab ini membahas tentang perencanaan serta pembuatan hardware dan software.

4) BAB IV PENGUKURAN DAN PENGUJIAN ALAT

Bab ini membahas tentang pengujian hardware dan software serta pembahasan mengenai hasil pengujian tersebut per blok sistem.

5) BAB V PENUTUP

Bab ini membahas tentang kesimpulan dan saran terhadap laporan akhir untuk pengembangan menjadi sistem yang lebih sempurna.

BAB II

LANDASAN TEORI

2.1. Jamur

Jamur dalam bahasa daerah (Sunda) dikenal dengan sebutan " supa " atau dalam bahasa Inggris disebut " mushroom " termasuk golongan fungi atau cendawan. Sedangkan menurut ahli mikrologi, jamur ialah fungi yang mempunyai bentuk tubuh buah seperti payung (Dr. Ir. Meity Sinaga, 1990 : 1).

Kehidupan jamur berawal dari spora (basidiospora) yang kemudian akan berkecambah membentuk hifa yang berupa benang-benang halus. Hifa ini akan tumbuh ke seluruh bagian media tumbuh. Kemudian dari kumpulan hifa atau miselium akan berbentuk gumpalan kecil seperti simpul benang yang menandakan bahwa tubuh buah jamur mulai terbentuk. Simpul itu berbentuk bundar atau lonjong dan dikenal dengan stadia kepala jarum (*pin head*). Simpul ini akan membesar dan diberi istilah stadia kancing kecil atau *small botton.* Selanjutnya stadia kancing kecil terus membesar mencapai stadia kancing (*button*) dan stadia telur (*egg*). Pada stadia ini, tangkai dan tudung yang tadinya tertutup selubung universal mulai membesar. Selubung tercabik, kemudian diikuti stadia perpanjangan (*elongation*). Cawan (*volva*) pada stadia ini terpisah dengan tudung (*pileus*) karena perpanjangan tangkai (*stalk*). Stadia yang terakhir adalah stadia dewasa tubuh buah (Dr. Ir. Meity Sinaga, 1990 : 1).

Kandungan protein jamur lebih tinggi dibandingkan dengan kandungan protein pada tumbuh-tumbuhan secara umum. Walaupun tidak setinggi protein

hewani, ikan atau telur, tapi hampir sebanding dengan protein susu, jagung atau kacang-kacangan dan lebih tinggi dari protein sayur-sayuran dan buah-buahan.

Berikut ini tabel mengenai perbandingan nilai gizi beberapa jenis jamur yang edible dengan bahan makanan lain dalam berat segar :

Tabel 2.1 Nilai gizi beberapa jenis jamur dibandingkan dengan bahan makanan lain *)

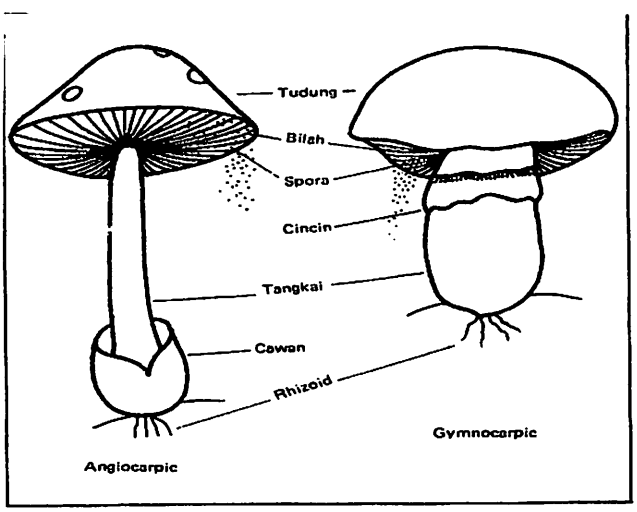
No	Jenis makanan	Protein (%)	Lemak (%)	Karbohidrat (%)
1.	*Agaricus sp.*	4,8	0,2	3,5
2.	*Boletus edulis*	5,4	0,4	5,2
3.	*Loctarius deliosus*	3,0	0,8	3,0
4.	*Cantarellus cibarius*	2,6	0,4	3,8
5.	Bayam	2,2	0,3	1,7
6.	Kentang	2,0	0,1	20,9
7.	Kubis	1,5	0,1	4,2
8.	Daging sapi	21,0	5,5	0,5
9.	*Volvariella volvaceae* (jamur merang)	1,8	0,3	12 - 48

*** Meity Sinaga, Jamur Merang dan Budidayanya.Jakarta, 1990 hlm. 8**

2.1.1. Daur Hidup Jamur

Sebagai organisme yang tidak berklorofil, jamur tidak dapat melakukan proses fotosintesis seperti tumbuh-tumbuhan. Dengan demikian jamur tidak dapat menggunakan langsung energi matahari. Jamur mendapat makanan dalam bentuk jadi seperti glukosa, protein dan senyawa pati. Bahan makanan ini akan diurai dengan bantuan enzim yang diproduksi oleh hifa menjadi senyawa yang dapat diserap dan digunakan untuk tumbuh dan berkembang. Semua jamur bersifat saprofit yaitu organisme yang hidup dari senyawa organik yang telah mati (Dr. Ir. Meity Sinaga, 1990 : 3-4).

Secara alami jamur dapat tumbuh dalam musim tertentu dalam satu tahun. Hal ini terjadi karena ketergantungan hidupnya akan temperatur dan kelembaban tertentu. Umumnya musim yang tepat untuk berburu jamur di alam bebas adalah pada musim hujan untuk daerah tropika dan musim gugur untuk daerah sub tropika.



GAMBAR 2.1 Bagian-Bagian Jamur

(Sumber : Meity Sinaga, Jamur Merang dan Budidayanya)

Bagian-bagian tubuh jamur terdiri dari :

1. Tundung sebagai pelindung tubuh jamur.
2. Bilah merupakan bagian pinggir dari jamur.
3. Spora yang jika ditanam akan tumbuh menjadi jamur baru.
4. Tangkai merupakan bagian penopang jamur.
5. Cawan sebagai tempat tumbuhnya tangkai.
6. Rizoid yang merupakan akar dari jamur.

2.1.2. Temperatur Dan Kelembaban Jamur

Menurut kemampuan hidup pada temperatur tertentu, jamur terbagi dalam tiga golongan yaitu :

1. Psikrofilik adalah jamur yang tumbuh pada kisaran temperatur antara 0 – 30° C, dengan temperature optimum 15° C
2. Mesofilik adalah jamur yang tumbuh pada kisaran temperature 25 – 37° C dengan temperature optimum sekitar 30° C.
3. Termofilik adalah jamur yang membutuhkan temperature optimum 55° C.

Untuk daerah Asia Tenggara umumnya jamur yang tumbuh adalah jenis mesofilik. Selain temperatur, kelembaban merupakan faktor yang paling berpengaruh dalam pertumbuhan jamur. Umumnya jamur akan tumbuh dengan baik pada keadaan udara yang lembab. Hal ini erat hubungannya dengan kebutuhan jamur akan air, baik dalam bentuk air ataupun uap air. Adapun kelembaban yang diperlukan jamur berkisar antara 60 – 90 %, sedangkan suhu

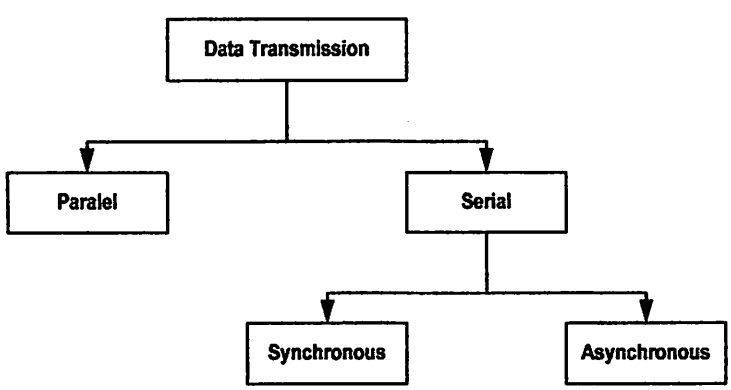
yang diperlukan jamur untuk hidup dengan baik adalah sekitar 25°C – 29°C dengan kondisi terbaik untuk hidup pada nilai 27°C.

Pengaturan suhu dan kelembaban tersebut di dalam ruangan dapat dilakukan dengan menyemprotkan air bersih ke dalam ruangan. Pengaturan kondisi lingkungan sangat penting bagi pertumbuhan tubuh buah. Apabila suhu terlalu tinggi, sedangkan kelembaban terlalu rendah maka primordia (bakal jamur) akan kering dan mati (Cahyana YA, Muchrodji, M. Bakrun , 1997 : 9).

Di samping suhu dan kelembaban, faktor cahaya dan sirkulasi udara perlu diperhatikan dalam budidaya jamur. Sirkulasi udara harus cukup, tidak terlalu besar tetapi tidak juga terlalu kecil. Intensitas cahaya yang diperlukan pada saat pertumbuhan sekitar 10 %.

2.2. Metode Transmisi Data Digital

Pengiriman data biner melalui suatu saluran dapat dilakukan dengan dua mode yaitu mode pentransmisian parallel dan mode pentransmisian serial.

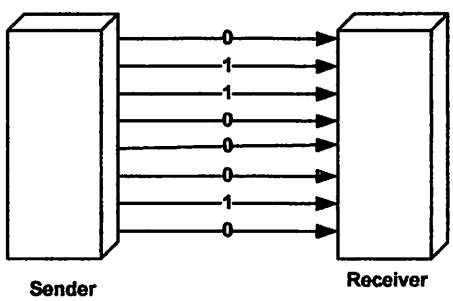


Gambar 2.2 Metode Transmisi Data Digital

(Sumber : William. L. Schwebwer, 1998)

2.2.1. Transmisi Data Parallel

Transmisi data paralel adalah transmisi sejumlah n bit data yang dapat dikirimkan dan diterima dalam waktu yang bersamaan dengan menggunakan n saluran. Jadi setiap bit data mempunyai 1 saluran. Keuntungan dari transmisi paralel yaitu pengiriman data menjadi lebih cepat, sedangkan kerugiannya yaitu biaya yang lebih banyak karena membutuhkan jumlah saluran yang lebih banyak. Karena permasalahan biaya menyebabkan metode ini hanya digunakan untuk komunikasi jarak dekat.



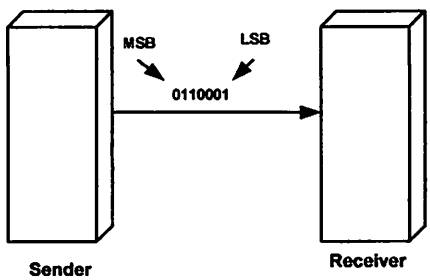
Gambar 2.3 Transmisi Data Paralel

(Sumber : William.L.Schwebwer, 1988)

2.2.2. Transmisi Data Serial

Transmisi data serial adalah transmisi data yang pengiriman dan penerimaan datanya berurutan tiap bitnya. Jadi kita hanya membutuhkan satu saluran untuk mengirimkan data antar dua perangkat komunikasi. Keuntungan transmisi serial yaitu biaya lebih murah karena hanya membutuhkan satu saluran saja sehingga banyak digunakan untuk komunikasi jarak jauh. Sedangkan

kerugiannya adalah kecepatan pengiriman data lebih rendah dibandingkan dengan transmisi paralel.



Gambar 2.4 Transmisi Data Serial

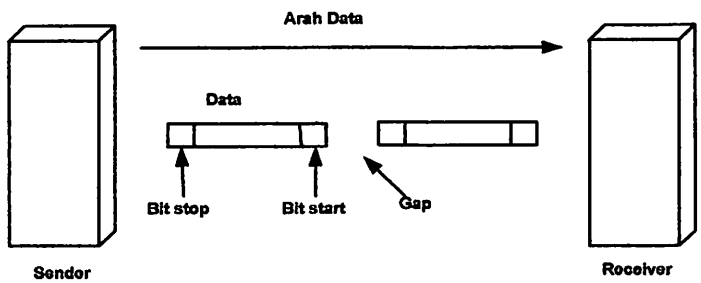
(Sumber : William.L.Schwebwer, 1988)

Pengiriman data akan dimulai dari LSB (*Least Significant Bit*) dan diakhiri dengan MSB (*Most Significant Bit*). Setiap karakter yang dikirimkan, disusun sesuai dengan suatu urutan dengan bit tertentu. Berdasarkan formatnya, salah satu jenis komunikasi serial yaitu :

2.2.2.1. Transmisi Data Serial Asinkron

Pada pengiriman data asinkron (tidak sinkron), setiap karakter dikirimkan sebagai satu kesatuan (*entity*) bebas yang berarti bahwa waktu antara pengiriman bit terakhir dari sebuah karakter dan bit pertama dari sebuah karakter berikutnya tidak tetap. Pengiriman data asinkron lebih sederhana daripada pengiriman sinkron karena hanya isyarat data saja yang dikirimkan. Detak penerima dibangkitkan secara lokal di dalam penerima dan tetap dijaga agar sesuai dengan detak pengirim yang menggunakan bit awal (start bit) dan bit akhir (stop bit) yang dikirimkan setiap karakter dan data yang satu dengan data selanjutnya

dipisahkan oleh gap. Penyesuaian detak pengiriman dan penerima terjadi karakter per karakter.



Gambar 2.5 Transmisi Data Serial Asinkron

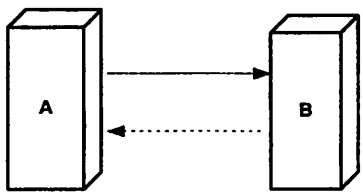
(Sumber : William.L.Schwebwer, 1988)

Gambar di atas menunjukkan sinkronisasi awal-akhir, bit awal dan akhir tidak membawa informasi, tetapi hanya menunjukkan awal dan akhir setiap karakter. Dari gambar dapat dilihat bahwa bit kedelapan disebut bit paritas, diikutsertakan dalam bentuk gelombang tersebut. Bit ini akan dipasang pada 1 atau 0 untuk meyakinkan cacah bit pada setiap karakter adalah genap untuk paritas genap, atau ganjil untuk paritas ganjil.

Berdasarkan arah komunikasinya, pengiriman data serial dibedakan menjadi :

- **Half Duplex**

Merupakan sistem komunikasi yang mengirimkan data dalam satu arah. Sistem ini tidak dapat mengirimkan data secara bersamaan sehingga perlu saling menunggu secara bergantian untuk berkomunikasi. Pada gambar di bawah menunjukkan komunikasi half duplex.

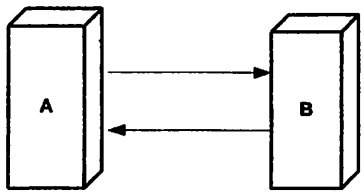


Gambar 2.6 Komunikasi Half Duplex

(Sumber : DC Green, Komunikasi Data, 1998 : 27)

- **Full Duplex**

Merupakan sistem komunikasi yang mengirimkan data dalam dua arah. Sistem ini dapat mengirimkan data secara bersamaan sehingga tidak perlu saling menunggu secara bergantian untuk komunikasi. Pada gambar di bawah menunjukkan komunikasi full duplex.



Gambar 2.7 Komunikasi Full Duplex

(Sumber : DC Green, Komunikasi Data, 1998 : 28)

2.3. Unit Komunikasi

2.3.1. Komunikasi Serial RS 232

RS 232 merupakan salah satu jenis interface dalam proses transfer data antar komputer dalam bentuk serial transfer. RS 232 merupakan kependekan dari **Recommended Standart Number 232**. RS 232 dibuat untuk interface antara peralatan terminal data dengan peralatan komunikasi data, dengan menggunakan

data biner serial sebagai data yang ditransmisikan. Serial interface RS 232 memberi ketentuan logic level sebagai berikut :

- Logic 1 disebut " mark " terletak antara -3 Volt hingga -15 Volt.
- Logic 0 disebut " space " terletak antara +3 Volt hingga +15 Volt.

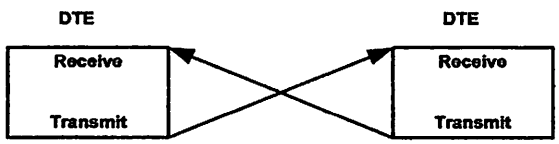
Daerah tegangan antara -3 Volt hingga +3 Volt adalah invaled level, yaitu daerah yang tidak memiliki logic sehingga daerah tersebut harus dihindari. Suatu saluran data RS 232 yang memberi keadaan ini berarti ada kesalahan. Demikian pula saluran daerah lebih negatif dari -15 volt dan daerah lebih positif dari +15 Volt. Pada saat pengiriman data, sebelum mengirim data bit 0 harus diawali dengan start bit terlebih dahulu kemudian baru mengirim bit 0. Setelah mengirim bit 7 masih harus diakhiri dengan stop bit.

2.3.1.1. Dasar-Dasar Serial Interface

Dasar-dasar serial interface telah diuraikan mengenai fungsi dari interface RS 232C. Dalam pembahasan berikut ini, akan dijelaskan bagaimana dasar-dasar interface tersebut. Proses transfer secara serial menggunakan alat RS 232C yang dibuat oleh *Electronic Industry Assosiation* (EIA) antara terminalnya, biasanya menggunakan DTE (*Data Terminal Equipment*) untuk masing-masing terminal. Kadang diperlukan seperangkat peralatan untuk kebutuhan komunikasi yang lebih kompleks, misalnya dengan memanfaatkan modem. Perangkat tersebut sering disebut dengan DCE (*Data Communication Equipment*).

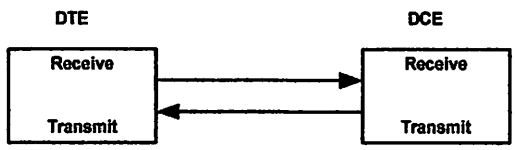
Data yang ditransfer dari suatu terminal akan diterima oleh terminal lainnya, dan demikian pula sebaliknya melalui seperangkat peralatan di atas.

Gambar di bawah menjelaskan konsep transfer antara DTE dengan DTE dan DTE dengan DCE. Jenis data yang akan ditransfer adalah dalam bentuk biner (bit per bit transfer) dengan satuan baut untuk kecepatan transfernya (bit per detik). Dalam proses transfer ini harus terdapat suatu peralatan yang berfungsi sebagai *hand shake* (jabat tangan) yaitu sebagai pemantau status yang diterima atau ada untuk memberikan respon yng sesuai. Dalam merancang perangkat lunak komunikasi serial, hand shake disempurnakan dengan menambahkan karakter pengendali dalam deretan atau jumlah bit yang ditransfer, biasanya disebut start bit dan stop bit.



Gambar 2.8 Blok Diagram Transfer Data DTE dengan DTE

(Sumber : Malvino)



Gambar 2.9 Blok Diagram Transfer Data DTE dengan DCE

(Sumber : Malvino)

Secara sederhana dapat dijelaskan bagaimana konsep interface antara DTE dengan DCE yang dilakukan berulang-ulang sampai semua data selesai ditransfer, adalah sebagai berikut :

- Ketika DTE ingin mengirim data, sebuah protokol yaitu RTS dikirimkan untuk memberitahu DCE.
- Pada saat itu masukan RTS pada DCE menjadi aktif.
- Jika DCE mampu menerima balasan data, maka ia akan membalasnya dengan mengirim CTS.
- Begitu DTE menerima balasan, masukan CTS-nya diaktifkan.
- Pengiriman data dilakukan melalui TxD.
- Penerimaan data dilakukan melalui RxD.

2.3.1.2. Pin- Pin pada EIA RS 232 dan Kegunaannya

Secara praktis untuk kebutuhan transfer data cukup 9 pin yang digunakan. Adapun masing-masing adalah sebagai berikut :

a) Pin 1 Protective Ground

Pin ini berguna untuk menghindarikejutan karakteristik listrik karena kegagalan suatu daya. Dalam kasus bagaimanapun, pada standart RS 232 pin 1 bukan merupakan suatu keharusan.

b) Pin 2 Transmitted Data (TxD / TD)

Berguna sebagai jalur pengiriman data dari DTE ke DCE.

c) Pin 3 Received Data (RxD / RD)

Berguna sebagai jalur penerimaan data dari DCE ke DTE.

d) Pin 4 Request to Send (RTS)

Berguna untuk memberitahu DCE bahwa DTE akan mengirim data. RTS merupakan sebuah protokol perangkat keras yang mendahului pengiriman data dari DTE ke DCE.

e) Pin 5 Clear to Send (CTS)

Berguna untuk memberitahu DTE bahwa DCE siap untuk menerima data. CTS merupakan sebuah protokol perangkat keras yang mendahului penerimaan data dari DTE ke DCE.

f) Pin 6 Data Set Ready (DSR)

Berguna untuk memberitahu DTE bahwa DCE aktif dan siap untuk bekerja.

g) Pin 7 Signal Ground

Berguna sebagai referensi semua tegangan interface.

h) Pin 8 Data Carrier Detect (DCD)

Berguna pada DTE untuk tidak memperbolehkan penerimaan data.

i) Pin 9 Data Terminal Ready (DTR)

Berguna untuk memberitahu DCE bahwa DTE aktif dan siap untuk bekerja.

Tabel 2.2 Konfigurasi Pin DB 9

Pin No	Name	Notes / Description
1	DCD	Data Carrier Detect
2	RD	Receive Data
3	TD	Transmit Data
4	DTR	Data Terminal Ready

5	**SGND**	**Ground**
6	**DSR**	**Data Set Ready**
7	**RTS**	**Request To Send**
8	**CTS**	**Clear To Send**
9	**RI**	**Ring Indicator**

(Sumber : Malvino)

2.3.1.3. Protokol Komunikasi pada RS 232

Beberapa protokol dalam interface RS 232 adalah :

- Start Bit

  Merupakan sebuah bit dengan logika '0', dimana bit ini yang menandakan bahwa akan ada karakter atau data yang mengikutinya. Bit ini langsung diberikan tanpa harus mensetnya terlebih dahulu.

- Data Bit

  Merupakan bit yang mewakili dari karakter yang diikutinya. Data bit ini dapat diset antara 5 sampai 8 bit.

- Parity Bit

  Merupakan bit yang digunakan sebagai error checking pada receiver, apabila terjadi kesalahan maka receiver akan menset error flag (*parity error*) pada spesial register. Parity bit ini menghitung jumlah data yang berlogika '1' pada data bit. Perhitungan jumlah data bit tersebut tergantung dari jenis parity yang diset. Untuk parity '*event*' jumlah data bit yang berlogika '1' ditambah dengan parity bit akan menghasilkan jumlah yang ganjil. Sedangkan untuk parity '*mark*' merupakan parity bit

yang selalu berlogika '1' demikian pula pada space parity bit selalu berlogika '0' dan parity '*none*' merupakan parity bit yang diabaikan.

- Stop Bit

  Merupakan bit yang menandakan akhir dari suatu paket data (biasanya 1 byte data). Seperti pada start bit, bit langsung diberikan pada serial device. Stop bit ini dapat diset panjangnya menjadi satu bit, satu setengah atau dua bit.

- Baud Rate

  Sebenarnya baud rate berarti pergantian kondisi tiap detik, tetapi karena hanya ada dua kondisi pada komunikasi serial (yaitu logic '1' dan '0') maka dapat juga digunakan untuk menunjukkan kecepatan dari transmisi bit (bit persecond).

2.3.2. Komunikasi Serial RS 485

Standart RS 485 diterapkan oleh *Electronic Industry Association* dan *Telecommunication Industry Association* pada tahun 1983. Standart RS 485 hanya membicarakan karakteristik sinyal dalam transmisi data secara *Balanced Digital Multipoint System.* Jadi jauh lebih sederhana dibandingkan dengan standart RS 232 yang mencakup ketentuan tentang karakteristik sinyal, macam-macam sinyal dan konektor yang dipakai, serta konfigurasi sinyal pada kaki-kaki di konektor dan juga penentuan tata cara pertukaran informasi antara komputer dan alat-alat pelengkapnya.

RS 485 bisa dipakai untuk saluran sejauh 4000 feet dan kecepatan lebih dari 1 Megabit / detik. Pada RS 485 menggunakan saluran ganda (*Differential* atau *Balanced Transmision*) yang memakai satu pasang kabel untuk mengirim satu sinyal, informasi logika ditafsirkan dari beda tegangan antara 2 utas kabel saluran. Tegangan pada kedua utas kabel saluran selalu berlawanan, saat satu kabel bertegangan tinggi maka kabel yang lain bertegangan rendah, demikian pula sebaliknya. Rangkaian penerima sinyal membandingkan tegangan kedua kabel saluran, level logika pada bagian output ditentukan oleh kabel mana yang lebih positif.

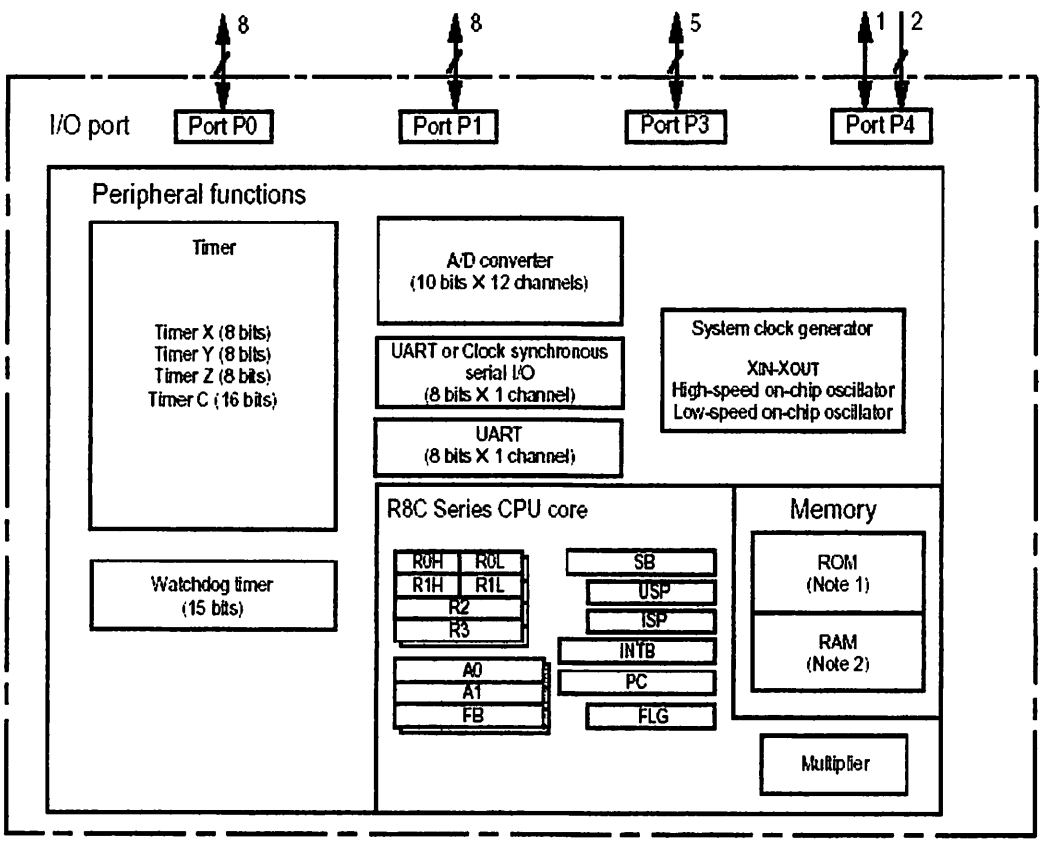
Meskipun demikian, saluran ganda tidak dipakai untuk transmisi yang memerlukan banyak saluran, mengingat RS 485 memakai kabel jauh lebih banyak sehingga mahal. Untuk penghematan kabel, bahkan saluran ganda sering dipakai untuk saluran *half duplex*, yaitu saluran dua arah secara bergantian yang hanya menggunakan satu pasang kabel, bisa dipakai untuk menghubungkan *line generator* dan banyak *line receiver* menjadi satu, sistem ini disebut sebagai *komunikasi multidrop.*

Meskipun balanced data transmission lebih rumit, tapi mempunyai sifat yang sangat kebal terhadap gangguan listrik, sehingga bisa dipakai untuk menyalurkan data lebih jauh dengan kecepatan lebih tinggi.

2.4 Mikrokontroler RENESAS R8C / Tiny

Mikrokontroler RENESAS dibangun menggunakan proses gerbang silicon CMOS dengan kemampuan tinggi menggunakan CPU seri R8C/Tiny dan dikemas

dalam modul plastic dengan jumlah pin sebanyak 32. Mikrokontroler ini beroperasi menggunakan perintah canggih khususnya efisiensi perintah dengan level tinggi. Mikrokontroler ini mempunyai 1 Mbytes kapasitas alamat, yang bisa digunakan untuk mengeksekusi perintah dengan kecepatan tinggi. Data flash ROM sebesar 2 KB x 2 blocks.



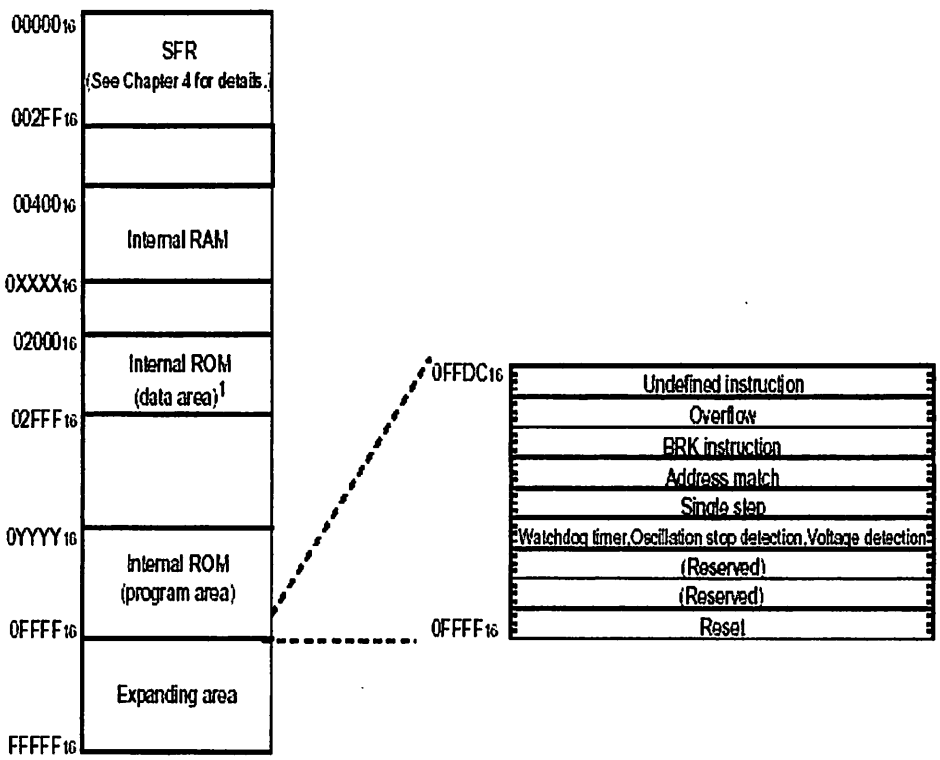
Gambar 2.10 Blok Diagram MCU Renesas

(Sumber : Datasheet Renesas)

Mikrokontroler Renesas R8C/Tiny mempunyai struktur memori yang terdiri atas :

- Space alamat hingga 1 Mbytes dari alamat 00000_{16} sampai $FFFFF_{16}$.

- ROM internal (*program area*) dialokasikan pada alamat terendah dimulai dari alamat $0FFFF_{16}$. Misalnya, 16 Kbyte ROM internal dialokasikan pada alamat yang dimulai dari $0C000_{16}$ sampai $0FFFF_{16}$.
- ROM internal untuk *data area* dialokasikan pada alamat 02000_{16} sampai $02FFF_{16}$.
- Sedangkan RAM internal dialokasikan pada arah alamat yang lebih tinggi dimulai dari alamat 00400_{16}.
- Special function register (SFR) dialokasikan pada alamat mulai dari 00000_{16} sampai $002FF_{16}$. Fungsi register control peripheral dialokasikan disini.



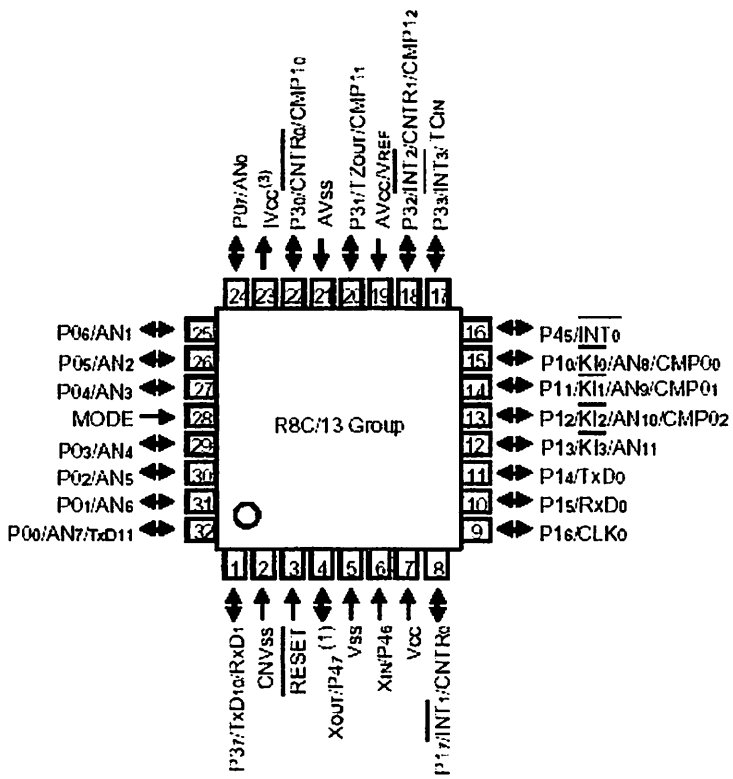
NOTES:

1. The data flash ROM block A (2K bytes) and block B (2K bytes) are shown.
2. Blank spaces are reserved. No access is allowed.

Type name	Internal ROM		Internal RAM	
	Size	Address 0YYYY₁₆	Size	Address 0XXXX₁₆
R5F21134FP, R5F21134DFP	16K bytes	0C000₁₆	1K bytes	007FF₁₆
R5F21133FP, R5F21133DFP	12K bytes	0D000₁₆	768 bytes	006FF₁₆
R5F21132FP, R5F21132DFP	8K bytes	0E000₁₆	512 bytes	005FF₁₆

Gambar 2.11 Memory Map

(Sumber : Datasheet Renesas)



Gambar 2.12 Konfigurasi Pin R8C/Tiny

(Sumber : Datasheet Renesas)

Keterangan fungsi masing-masing pin :

1. V_{CC}

   Digunakan untuk sumber tegangan dengan range nilai antara 2,7 – 5,5 Volt.

2. V_{SS}

   Range tegangan 0 Volt

3. IV_{CC}

   Pin ini digunakan untuk menyetabilkan sumber tegangan internal. Pin ini dihubungkan ke V_{SS} melalui kapasitor 0,1 μF.

4. AV_{CC}, AV_{SS}

   Pin ini merupakan input power supply untuk A/D Converter. Pin ini dihubungkan ke pin V_{CC}, sedangkan AV_{SS} dihubungkan ke V_{SS}. Hubungkan pin AV_{SS} dan AV_{CC} dengan kapasitor.

5. Reset

   Merupakan input reset pada MCU.

6. CNV_{SS}

   Pin ini dihubungkan ke V_{SS} melalui resistor.

7. Mode

   Pin ini dihubungkan ke V_{CC} melalui resistor.

8. X_{IN}, X_{OUT}

   Pin ini disediakan untuk pembangkitan rangkaian I/O pada clock utama. Hubungkan resonator keramik atau osilator kristal antara X_{IN} dan X_{OUT}. Untuk menggunakan clock derived external, masukkan ke pin X_{IN} dan pin X_{OUT} dibiarkan terbuka.

9. $INT_0 - INT_3$

   Merupakan pin input interupt

10. $KI_0 - KI_3$

    Merupakan pin Key Input interrupt.

11. $CNTR_0$ (I/O)

    Merupakan timer pin X I/O

12. $CNTR_0$ (O)

    Merupakan timer pin X output.

13. $CNTR_1$

    Merupakan timer pin Y I/O

14. TZ_{OUT}

    Merupakan timer pin Z output.

15. TC_{IN}

    Merupakan timer pin C input.

16. $CMPO_0 - CMPO_3$ dan $CMPO_{10} - CMPO_{13}$

    Merupakan timer pin C output.

17. CLK_0

    Merupakan transfer clock untuk pin I/O.

18. RxD_0 dan RxD_1

    Pin serial data input.

19. TxD_0, TxD_{10} dan TxD_{11}

    Pin serial data output.

20. V_{REF}

    Referensi pin input tegangan untuk A/D Converter. Hubungkan V_{REF} ke V_{CC}.

21. $AN_0 - AN_{11}$

    Pin input analog untuk A/D Converter.

22. $P0_0 - P0_1$, $P1_0 - P1_7$, $P3_0 - P3_3$, $P3_7$ dan $P4_5$

    Merupakan port 8 bit CMOS I/O. $P1_0 - P1_7$ juga berfungsi sebagai port LED driver.

2.5 SMS (Short Message Service)

2.5.1 Pengkonversian 7 bit (Septet) ke 8 bit (Oktet)

SMS yang dispesifikasikan oleh organisasi ETSI (European Telecommunication Standart Institute) memiliki panjang 160 karakter, bit tiap karakter adalah 7 bit ditunjukkan dalam tabel 7 bit default alphabet.

Dalam pengirimannya, SMS 7 bit (septet) diubah dalam oktet. Dijelaskan dalam contoh berikut : Pesan " hellohello " terdiri atas 10 karakter, tiap karakter diubah dalam 7 bit, pengubahan pesan ini ditunjukkan dalam tabel.

Tabel 2.3 Karakter Dalam Septet Untuk Pesan hellohello

Pesan	7 Bit (Septet)
h	1101000
e	1100101
l	1101100
l	1101100
o	1101111
h	1101000
e	1100101
l	1101100
l	1101100
o	1101111

(Sumber : Trik pemrograman SMS, 2002 : 15)

Septet pertama (h) diubah dalam sebuah oktet dengan menambah satu bit paling kanan dari septet kedua agar jumlah bit menjadi 8. Bit ini dimasukkan pada bagian paling kiri dari 7 bit karakter 'h' sehingga 1 + 1101000 = 11101000 (" E8") menjadi 8 bit. Bit paling kanan dari karakter kedua telah terpakai sehingga karakter kedua memerlukan 2 bit dari karakter ketiga untuk membuat 8 bit. Demikian seterusnya hingga karakter terakhir, sehingga diperoleh format 8 bit untuk tiap karakter yang ditunjukkan dalam tabel.

Tabel 2.4 Hasil Pengubahan 7 bit default ke 8 bit

8 Bit Oktet	Heksadesimal
11101000	E8
00110010	32
10011011	9B
11111101	FD
01000110	46
10010111	97
11011001	D9
11101100	EC
110111	37

(Sumber : trik Pemrograman SMS, 2002 : 15)

Jadi, hellohello akan dikirimkan dalam heksadesimal sebagai berikut : E8329BFD4697DEC37

2.5.2 AT Command

AT Command bertugas mengirim dan menerima data dari dan menuju SMS. AT Command yang penting dalam sistem SMS adalah :

- ***AT+CMGS*** (untuk mengirim SMS)

AT+CMGS = n (untuk mengirim SMS)

n = jumlah pasangan heksa PDU SMS dimulai setelah nomor SMS center (maksimal 140)

Tabel 2.5 Parameter *command syntax* AT+CMGS

Command	Respon
Jika mode PDU(+CMGF=0): +CMGS=<length><CR>PDU yang ada <ctrl Z/ESC>	Sukses:+CMGS:<mr> Gagal:+CMS:<error>
+CMGS=?S	

(Sumber : www.siemens-mobile.com)

- ***AT+CMGL*** (untuk memeriksa SMS)

AT+CMGL = n? (untuk memeriksa SMS) antara lain :

- n = 0 untuk SMS baru di INBOX (pesan belum terbaca)
- n = 1 untuk SMS lama di INBOX (pesan telah terbaca)
- n = 2 untuk SMS Unset di OUTBOX (pesan tidak terkirim)
- n = 3 untuk SMS sent di OUTBOX (pesan terkirim)
- n = 4 untuk semua SMS

Tabel 2.6 Parameter command syntax AT+CMGL

Command	Respon
+CMGL=[<stat>]	Jika mode PDU(+CMGF=0) dan command sukses : +CMGL:<index>,<stat>,<length><CR><LF><pdu>[<CR><LF> +CMGL:<index>,<stat>,<length><CR><LF><pdu>[...]] Jika gagal :+CMS ERROR:<err>
+CMGL=?	+CMGL: (status yang didukung <stat>s)

(Sumber : www.siemens-mobile.com)

- ***AT+CMGR*** (untuk menerima SMS)

Tabel 2.7 Parameter command syntax AT+CMGR

Command	Respon
CMGR<index>	Sukses:CMGR:<stat>,<length><CR><LF><pdu> Gagal:+CMS:<error>
+CMGR=?	OK

(Sumber : www.siemens-mobile.com)

- ***AT+CMGD*** (untuk menghapus SMS)

AT+CMGD = (untuk menghapus SMS)

n = nomor referensi SMS yang ingin dihapus.

Tabel 2.8 Parameter command syntax AT+CMGD

Command	Respon
+CMGD = <index>	+CMS : <error>

+CMGD = ?	

(Sumber : www.siemens-mobile.com)

- ***AT+CMGF*** (untuk SMS format)

Tabel 2.9 Parameter command syntax AT+CMGF

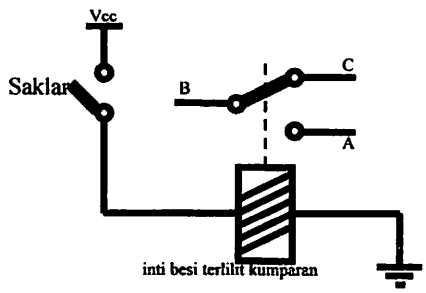
Command	Respon
+CMGF = [<mode>]	
+CMGF =?	+CMGF:<mode>
+CMGF=?	+CMGF: (pilihan yang mendukung <mode>s)

(Sumber : www.siemens-mobile.com)

2.6 Relay

Relay adalah suatu alat untuk membuka dan menutup kontak secara elektrik dengan tujuan menghubungkan fungsi dari rangkaian ke rangkaian lainnya. Relay mempunyai impedansi yang rendah yang nilai impedansi kumparannya berkisar antara beberapa puluh sampai beberapa ratus ohm, sedangkan sumber penggerak masukan umumnya memiliki impedansi yang jauh lebih tinggi. Daya akan ditransfer secara maksimal dari sinyal penggerak ke kumparan relay jika impedansi sumber penggerak dan impedansi kumparan sesuai. Kontak-kontak tersebut ada dua macam, yaitu *Normally Open* dan *Normally Closed.* Normally open adalah relay yang kontaknya terbuka pada saat belum ada arus yang melalui kumparan dan tertutup saat ada arus. Sedangkan

normally closed adalah relay yang kontaknya tertutup sebelum ada arus yang melalui kumparan dan terbuka pada saat ada arus.



Gambar 2.13 Relay

(Sumber : Malvino)

Cara kerja arus adalah sebagai berikut :

Jika ada masukan arus maka pada kumparan akan terdapat induksi magnetik yang nantinya akan menarik pegas kontak untuk merubah posisi awalnya menjadi terhubung ke bagian yang diinginkan. Setelah arus berhenti, maka tidak ada induksi sehingga kontak akan kembali ke posisi semula.

2.7 Borland Delphi

Borland Delphi adalah salah satu bahasa pemrograman terbaik saat ini. Dalam pembuatan sebuah program, Delphi menggunakan sistem yang disebut RAD *(Rapid Application Development).* Sistem ini memanfaatkan bahasa pemrograman *visual* yang membuat seorang programmer lebih mudah mendesain program tampilan (*user interface).* Cara ini sangat bermanfaat untuk membuat program yang bekerja dalam sistem Windows yang memang tampilan layarnya lebih rumit (sekaligus indah dilihat) dibandingkan DOS. Dengan bahasa

pemrograman biasa (*non visual),* waktu seorang programer akan lebih banyak dihabiskan untuk mendesain atau memperindah tampilan program dari pada menulis program utamanya sendiri.

2.7.1 Kelebihan-kelebihan Program Delphi

Delphi menyediakan fasilitas yang luas mulai dari fungsi untuk membuat *form* hingga fungsi untuk menggunakan beberapa format *file* basis data yang populer (*dBASE, Paradox*, dsb).

Fasilitas-fasilitas Delphi meliputi :

a. Komponen yang dapat dipakai ulang dan dapat dikembangkan Delphi mempermudah pembuatan program bagi komponen-komponen *Windows* seperti label, button dan bahkan dialog. Sebagai contoh, dalam *Windows*, dialog untuk menyimpan *file* ditemukan dalam banyak aplikasi. Dialog tersebut telah tersedia dalam Delphi dan dapat langsung digunakan tanpa harus memrogram.
b. Komponen visual dan komponen non-visual sudah tersedia dalam Delphi. Komponen-komponen yang dapat ditemui antara lain: komponen *button*, komponen-komponen *Database*, komponen Menu dan Dialog. Komponen-komponen *Database* memungkinkan untuk menampilkan data tanpa memprogram, cukup dengan menggunakan tombol mouse.
c. Dapat mengakses VBX. Delphi dapat mengakses objek-objek VBX secara langsung. Dalam Delphi VBX dianggap sebagai kumpulan komponen yang dapat digunakan langsung untuk membuat aplikasi.

d. *Template* aplikasi dan *Template Form.* Dalam Delphi telah didefinisikan *template* aplikasi dan *template form* yang dapat dipakai untuk membuat semua aplikasi dengan lebih cepat.

e. Lingkungan pengembangan Delphi. Beberapa contoh fasilitas di dalam Delphi yang dapat diatur sesuai kebutuhan adalah Palet Komponen (*Component Pallete*), *Editor Program* dan *Template Form.*

f. Program yang terkompilasi. Kebanyakan lingkungan pengembangan visual pada Windows menyatakan dapat mengkompilasi program. Namun sebenarnya mereka hanya dapat mengkompilasi sebagian program dan kemudian menggabungkan *interpreter* dan *pcode* dalam sebuah *file*. Dengan cara ini didapatkan eksekusi yang lambat. Di dalam Delphi, program yang dihasilkan benar-benar program yang terkompilasi tanpa *interpreter* dan *pcode* sehingga dapat berjalan lebih cepat. Program Delphi yang kecil dapat diserahkan dalam bentuk sebuah *file* EXE tanpa harus meyertakan *file* DLL.

g. Kemampuan mengakses data dalam bermacam format. Dalam Delphi terdapat *Borland Database Engine* (BDE) yang digunakan untuk mengakses format *file* data yang ada. BDE telah melalui beberapa tahap perkembangan. Sebelumnya BDE dikenal sebagai ODAPI, kemudian IDAPI, dan sekarang menjadi standar untuk akses semua jenis data yang ada saat ini.

2.7.2 Istilah pada Pemrograman Delphi

Pada bahasa pemrograman Delphi terdapat beberapa istilah penting yang perlu diketahui untuk mempermudah dalam menggunakannya. Istilah-istilah tersebut antara lain:

a. ***Event.***

Event adalah aktifitas yang terjadi saat penggunaan aplikasi dan terjadi karena adanya tindakan dan pemakai, seperti klik maouse atau penekanan tombol keyboard. Event juga bisa terjadi karena adanya pengaturan oleh timer atau karena tindakan pemakai sendiri.

b. **File Eksekusi.**

File Eksekusi adalah file program yang telah diterjemahkan ke dalam bahasa mesin. File ini dijalankan dari luar aplikasi pembuatannya, yang biasanya mempunyai ekstensi.EXE.

c. **Fungsi**

Fungsi adalah prosedur yang mengembalikan sebuah nilai.

d. ***Icon.***

Icon adalah gambar yang merepresentasikan aplikasi.

e. **Komponen Visual.**

Komponen Visual merupakan pembentukkan obyek di atas form yang akan menjadi tampilan program, sehingga merupakan ciri dari sebuah bahasa

pemrograman visual. Komponen visual ini kebanyakan berfungsi sebagai input dan output dari program-program yang dibuat.

f. **Obyek.**

Obyek merupakan bagian dari perlengkapan suatu aplikasi yang mengenai spesifikasi properti tersendiri. Sebagai contoh, form (lembar untuk mendesain) adalah sebuah obyek. Demikian juga dengan komponen visual. Obyek ini tidak terbatas pada form dan komponen visual saja, tetapi bisa saja diambil dari apliksi lain misalnya obyek dari MS-Excel atau MS-Word.

g. Properti

Properti adalah bagian yang membangun sebuah obyek. Properi ini sangat menentukan obyek yang sedang dibuat. Misalnya saja mengenai judul, panjang, lebar dan sebagainya.

h. **Prosedur.**

Produser adalah sekumpulan kode rutin yang ditulis dalam satu blok tersendiri. Blok tersebut hanya akan dieksekusi jika judul atau namanya dipanggil.

i. **Proyek.**

Proyek adalah kumpulan dari file yang terorganisir dan membentuk sebuah program.

j. **Unit.**

Unit adalah sekelompok prosedur dan fungsi yang sejenis.

k. ***User Interface.***

User Interface atau tampilan program adalah bagian aplikasi yang kita lihat. User Interface ini berisi komponen visual yang tersedia bagi pemakai aplikasi, seperti tombol window, grafik, suara dan sebagainya.

1. ***Window***

Window merupakan kotak persegi panjang pada layar yang berisi aplikasi atau bagian dari aplikasi. Window mirip dengan selembar kertas yang dapat dibuka dan ditutup, ditutupi dengan kertas lain, ditulisi bagian atasnya dan sebagainya.

2.7.3 Visual Component Library

Pada Delphi dijumpai dua jenis komponen yaitu Visual Component (VC) dan NonVisual Component (NVC). Perbedaan dari kedua komponen tersebut adalah komponen visual akan selalu tampak baik pada mode desain ataupun mide run-time. Sedangkan komponen nonvisual penampakannya hanya pada mode desain dan biasanya hanya berbentuk icon. Baik visual maupun nonvisual, kedua komponen ini sangat penting dalam membangun sebuah program.

Meskipun terdiri dari dua jenis, semua komponen tersebut digabungkan dalam sebuah library dan dikelompokkan berdasarkan kegunaannya. Pada Delphi, Visual Component Library dibagi dalam sembilan kelompok standar yaitu Standard, Additional, Win95, Data Aceces, Data Control, Win3.1, Dialogs, System, Quick Reports dan dua kelompok tambahan yaitu OCX dan Samples.

Berikut ini penjelasan masing-masing komponen yang disediakan oleh Delphi.

2.7.3.1 Bagian Standard

Bagian ini berisikan komponen visual maupun nonvisual yang sering digunakan, yang terdiri dari empat belas komponen yaitu :

MainMenu.

Komponen ini digunakan untuk mendesain dan menciptakan menu bar yang ada pada form (NVC).

PopupMenu.

Komponen ini digunakan untuk mencipta menu popup yang akan keluar saat mengklik kanan mouse (NVC).

Label.

Komponen ini digunakan unuk membuat teks di form atau obyek lain tanpa dapat diubah oleh pemakai program (VC).

Edit.

Komponen ini digunakan sebagai input/output satu baris teks. Pemakai program dapat mengubah teks ini (VC).

2.7.3.2 Bagian Data Acces

Bagian ini semuanya berisi nonvisual untuk berhubungan dan berkomunikasi dengan file database. Pada Delphi versi Desktop, bagian ini terdiri dari delapan komponen. Akan tetapi pada versi Developer dan Clien/Server, bagian ini terdiri dari sembilan komponen :

DataSource

Komponen ini digunakan untuk menghubungkan komponen Table atau Query dengan komponen tempat data akan ditampilkan (NVC).

Table

Komponen ini digunakan untuk menghubungkan tabel pada suatu database dengan program yang dibuat (NVC).

Query

Komponen ini digunakan untuk membuat dan mengeksekusi SQL query pada SQL Server Database atau Database lokal (NVC).

StoredProc

Komponen ini digunakan untu mengeksekusi prosedur yang tersimpan di SQL Server (NVC).

Database

Komponen ini digunakan untuk membuat hubungan program yang dibuat dengan database server (NVC).

Session

Komponen ini digunakan untuk melengkapi pengontrolan secara global terhadap aplikasi yang berhubungan dengan database (NVC).

BatchMove

Komponen ini memungkinkan kita untuk memodifikasi file database server pada program yang dibuat dan kemudian dapat memperbaharui server tersebut (NVC).

UpdateSQL

Komponen ini digunakan untuk membuat pembaharuan terhadap SQL Database (NVC).

2.7.3.3 Bagian Data Control

Bagian ini berisikan komponen visual maupun nonvisual untuk menampilkan dan mengolah data database. Bagian ini terdiri dari duabelas komponen yaitu :

DBGrid

Komponen ini digunakan untuk menampilkan data-data dalam bentuk baris dan kolom.

DBNavigator

Komponen ini digunakan ubtuk membuat pengontrol yang bisa menavigasi database dan mempunyai kemampuan untuk mengubah data tersebut (VC).

DBtext

Komponen ini digunakan untuk menampilkan data suatu field database dengan versi komponen label. dengan demikian pemakai program tidak dapat memodifikasi data suatu field yang ditampilkan (VC).

DBEdit

Komponen ini mirip dengan Dbtext hanya saja versi dari komponen edit, dengan demikian kita bisa memodifikasi data field database yang ditampilkan (VC).

DBMemo

Komponen ini untuk menampilkan data field database dengan versi komponen Memo (VC).

DBImage

Komponen ini digunakan untuk menampilkan data suatu field data base dengan versi komponen Image (VC).

DBListBox

Komponen ini digunakan untuk menampilkan data suatu field database dengan versi komponen ListBox (VC).

DBComboBox

Komponen ini digunakan untuk menampilkan data suatu field database dengan versi komponen ComboBox (VC).

DBCheckBox

Komponen ini digunakan untuk menampilkan data suatu field database dengan versi komponen ComboBox (VC).

DBRadioGroup

Komponen ini digunakan untuk menampilkan data suatu field database dengan versi komponen

DBLookupListBox

Komponen ini digunakan untuk membuat ListBox yang menampilkan isi sebuah dataset (VC).

DBLookupComboBox

Komponen ini untuk membuat ComboBox yang menampilkan isi sebuah dataset (VC).

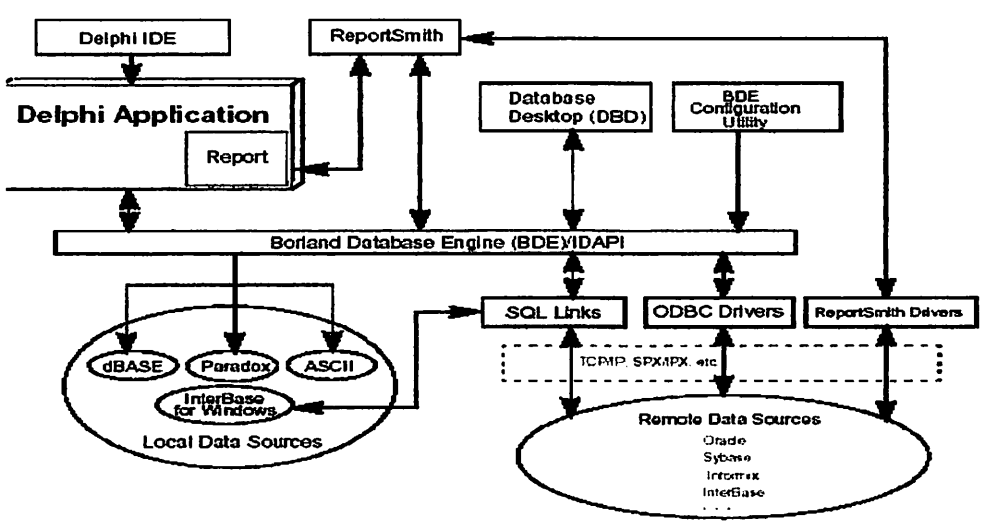
2.7.4 Database

Database dipakai untuk menyimpan data sehingga dapat dimanipulasi dengan mudah. Tanpa database, programmer akan membuat prosedur-prosedur yang rumit untuk memanipulasi data. Delphi mempunyai komponen-komponen

yang dipakai untuk mengakses tabel dan method untuk memanipulasi record. Ada beberapa model database, tetapi yang paling banyak dipakai adalah database relasioanal. Database relasional menyimpan data dalam tabel logika yang terdiri atas baris dan kolom. Kolom dari tabel dinamakan field dan baris dinamakan record. Satu kolom berisi data yang sejenis dan satu record berisi data yang berhubungan.

2.7.4.1 Borland Database Engine

Untuk mengakses database lokal maupun client server, Delphi memerlukan Borland Database Engine(BDE). BDE adalah kumpulan DLL dan perangkat yang mampu mengakses bermacam-macam database. Gambar 2.15 memperlihatkan jika kita ingin mengakses suatu database aplikasi kita harus menggunakan BDE. (Reisdorph Kent).



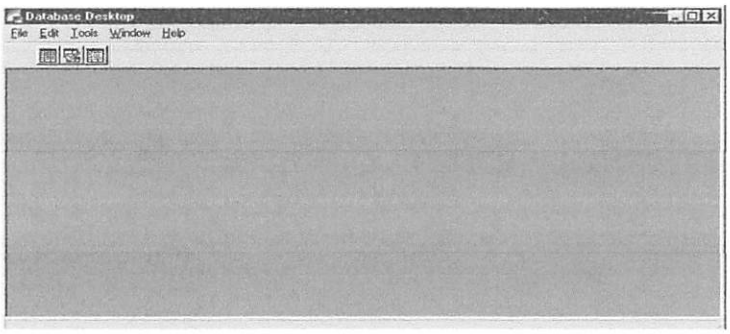
Gambar 2.14 Arsitektur database Delphi

(Anonim, 1995: 2)

Program aplikasi yang menggunakan BDE pada saat desain, harus menyertakan file BDE ini pada saat program didistribusikan. Letak BDE biasanya di C:\Program Files\Common Files\Borland Shared\BDE.

2.7.4.2 Membuat Tabel dengan Database Desktop

"Database Desktop adalah fasilitas yang dimiliki Delphi unutk membantu membuat, menampilkan, mengurutkan dan mencari data yang ada pada table. Database Desktop dapat menangani table Paradox,dBase dan format SQL." (Inge Martina, 2002: 48). Memanggil Database Desktop adalah dengan cara mengikuti urutan **Start | Programs | Borland Delphi 7 | Database Desktop**. Akan muncul tampilan seperti pada Gambar 2.16

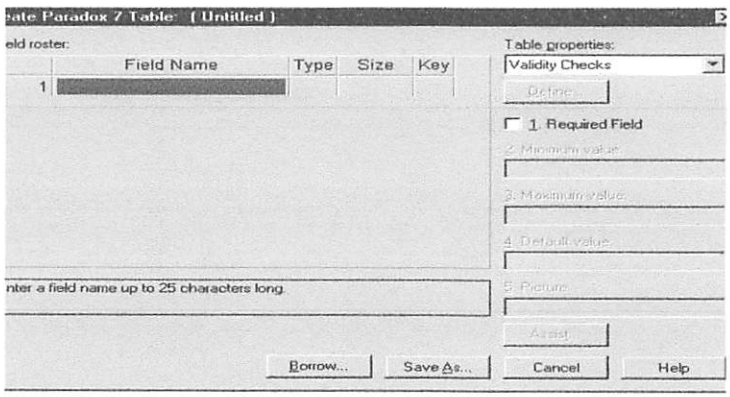


Gambar 2.15 Borland Database Desktop

(Borland Database Desktop V7.0)

Untuk membuat tabel, pilih **File|New|Table**. Kita dapat memilih jenis databasenya, defaulnya **Paradox7**, Klik **OK**. Muncul tampilan seperti pada

Gambar 2.17. *Field Name* adalah nama masing-masing field tabel yang kita buat. *Type* adalah jenis data dari *field*, penjelasan *type* data didaftar pada Tabel 2.11. *Size* adalah ukuran *field.*



Gambar 2.16 Membuat Tabel dengan Database Desktop

(Borland Database Desktop V7.0)

Untuk membuat table, kita harus mengetahui tipe-tipe data yang sesuai dengan jenis tabel yang akan dibuat. Tabel 2.5 mendaftar tipe data tabel paradox.

Keterangan:

* Banyaknya digit setelah titik desimal

* * Panjang Memo dan formatted memo. Disimpan di file tersendiri dengan ext .MB

*** Opsional

Tabel 2.10 Tipe data tabel Paradox

Simbol	Ukuran	Tipe	Penjelasan
A	1 – 255	Alpha	Strings dengan panjang maksimum 255 karakter
N		Number	Bilangan -10^307 sampai 10^308
$		Money	Sebuah nilai dengan format mata uang dan sebuah simbol mata uang
S		Short	Bilangan -32,767 sampai 32,767
I		Long Integer	Bilangan –2147483648 sampai 2147483647
#	0 – 32*	BCD	Data numerik dalam format BCD
D		Date	Berisi data tanggal
T		Time	Berisi data jam
@		Timestamp	Berisi data waktu dan tanggal
M	1 - 240**	Memo	Strings yang terlalu panjang jika simpan di field Alpha
F	0 - 240**	Formatted Memo	Seperti field memo dengan pengecualian text dapat diformat misalnya ukuran dan warna
G	0 - 240***	Graphic	Berisi data berupa gambar
O	0 -	OLE	Berisi berbagai macam data seperti citra,

	240***		suara, documents dan lain-lain
L		Logical	Nilai boolean dengan ukuran 1 bit
+		Autoincrement	Berisi data long integer dengan nilai yang hanya bisa dibaca. Menghapus sebuah record tidak mempengaruhi nilai pada record lain
B	0 - 240***	Binary	Data biner, biasanya untuk menyimpan suara
Y	1 – 255	Bytes	Data byte, biasaya untuk menyimpan data bar code atau strip magnetic

(Sumber: Anonim, 1992-1996)

2.8 Pengolah Sinyal

Piranti-piranti yang terlibat dalam pengolahan sinyal adalah sebagai berikut :

2.8.1 Operasional Amplifier

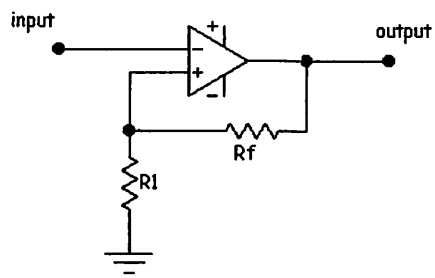
Keluaran suatu rangkaian sebelum memasuki rangkaian selanjutnya jika sinyalnya masih kecil membutuhkan suatu penguat. Perangkat elektronika yang umum digunakan sebagai penguat adalah penguat operasional. Karakteristik op-amp yang pokok adalah :

- Impedansi masuka amat tinggi, sehingga arus masukan dapat diabaikan.
- Penguatan tinggi.
- Impedansi keluaran rendah, sehingga tidak terpengaruh oleh pembebanan.

2.8.1.1 Macam Operasional Amplifier

2.8.1.1.1 Inverting Amplifier

Penguatan tegangan yang dihasilkan adalah sesuai dengan persamaan (1), dimana tegangan keluaran memiliki polaritas yang berlawanan dengan polaritas tegangan masukan.



Gambar 2.17 Rangkaian Inverting Amplifier

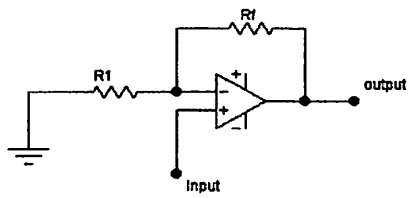
(Sumber : Malvino)

$$A_v = V_{out} / V_{in} = -R_f / R_{in}$$

$$V_{out} = (-R_f / R_{in}).V_{in} \quad .....(1)$$

2.8.1.1.2 Non Inverting Amplifier

Penguatan tegangan yang dihasilkan adalah sesuai dengan persamaan (2), dimana tegangan keluaran memiliki polaritas yang sefasa dengan polaritas tegangan masukan.



Gambar 2.18 Rangkaian Non Inverting

(Sumber : Malvino)

$A_v = V_{out} / V_{in}$

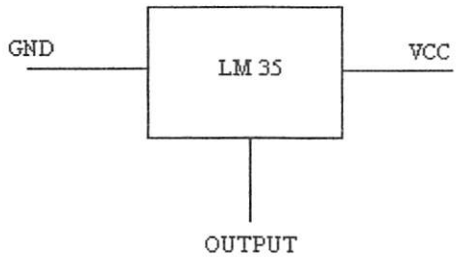
$V_{out} = (1 + (R_f / R_{in})).V_{in}$ .....(2)

2.9 Sensor Temperatur

Sensor suhu yang digunakan adalah LM 35 yang mempunyai output dalam skala derajat Celcius. Pada saat suhu 0°C, output sensor LM 35 mengeluarkan tegangan 0 V. Setiap kenaikan 1°C, output sensor LM 35 naik sebesar 10 mV. Sensor LM 35 membutuhkan power supply sebesar 5 V. Contoh pada suhu 37,5°C, tegangan keluarannya 0,375 V.

Kalau pengindera semacam itu diperlukan, tegangan acuan yang stabil yang perlu dikurangkan dari hasil pembacaan.

Keunggulan lain dari LM 35 adalah konsumsi arus yang kurang dari 60 μA.

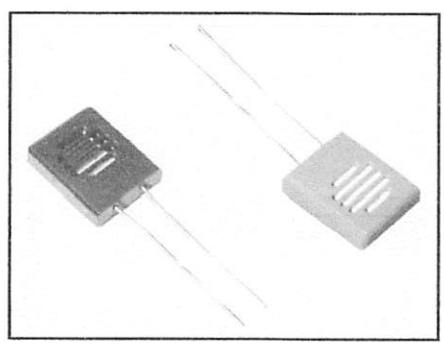


Gambar 2.19 Pin Sensor Suhu LM35

(Sumber : Datasheet LM35)

2.10 Sensor Kelembaban

HS15P merupakan salah satu tipe sensor terbuat dari bahan polimer yang mempunyai kehandalan dalam pengukuran nilai relativitas kelembaban. HS15P dapat mengukur relativitas kelembaban dalam area operasi 20 – 100 %. Selain itu, kemampuan HS15P dalam membaca nilai relativitas kelembaban juga bertahan lama dan murah harganya. HS15P ini bekerja pada temperature 0 sampai 50°C.



Gambar 2.20 Sensor Kelembaban HS15P

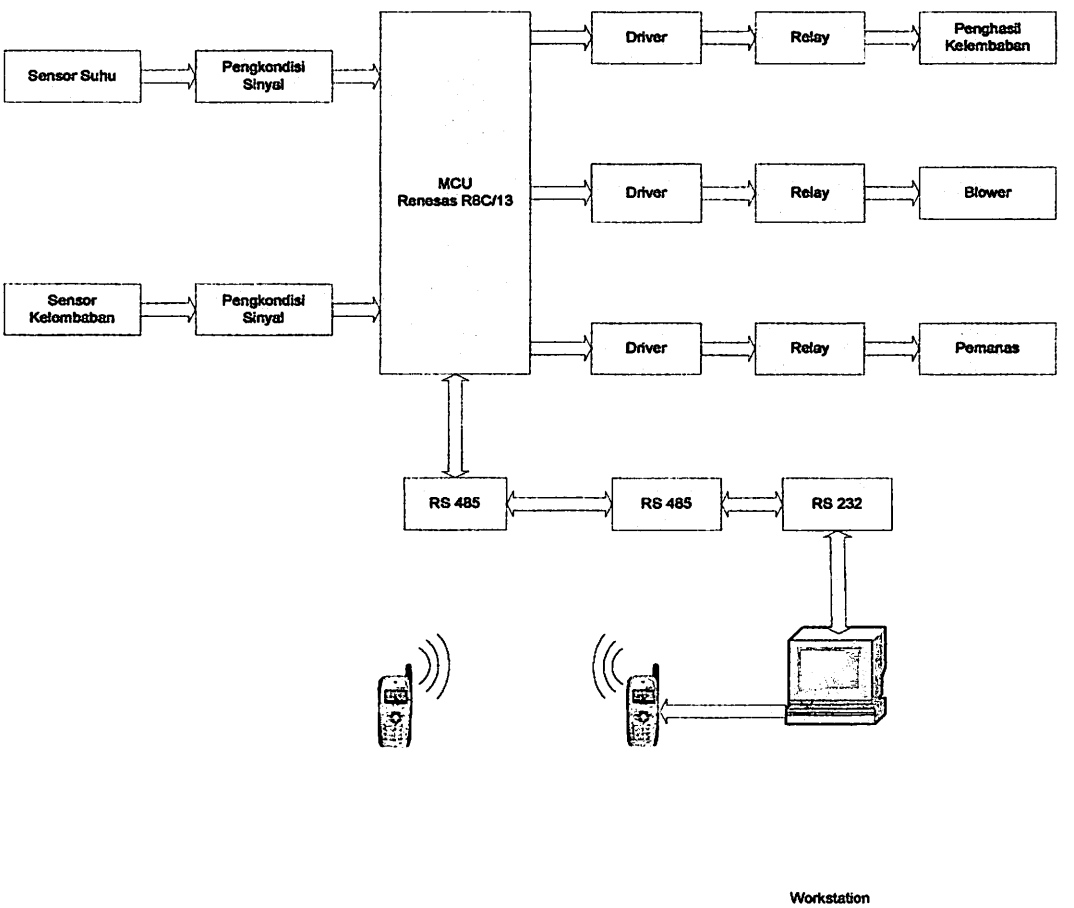
(Sumber : Datasheet HS15P)

Output dari HS15P ini berupa impedansi yang bergantung dari suhu. Misalnya, pada suhu 25°C dan relativitas kelembaban 50 %, output impedansinya sekitar 60 KΩ ± 30 KΩ. Tegangan rata-rata yang dibutuhkan adalah AC 1 Vrms dengan frekuensi rata-rata sekitar 1 KHz. Sedangkan konsumsi daya yang dibutuhkan sekitar 0,3 mW

BAB III

PERENCANAAN DAN PEMBUATAN ALAT

3.1 Blok Diagram



Gambar 3.1 Blok Diagram Sistem

3.2 Perencanaan Sistem

Sensor suhu dan kelembaban diletakkan di rumah jamur dan akan membaca data dalam rumah jamur. Jika suhu dan kelembaban dalam rumah jamur mengalami perubahan, maka dapat menyebabkan perubahan impedansi pada sensor. Sensor pendeteksi kelembaban dan suhu memiliki keluaran berupa besaran analog. Rangkaian pengkondisi sinyal perlu ditambahkan untuk menguatkan sinyal output dari sensor. Agar keluaran sensor dapat dibaca oleh mikrokontroler, maka keluaran dari sensor harus diubah dahulu menjadi besaran digital oleh rangkaian ADC (Analog to Digital Converter) yang sudah terdapat pada MCU RENESAS yang digunakan.

Pada media rumah jamur terdapat beberapa kondisi udara yang bisa mempengaruhi pertumbuhan jamur. Diantaranya kondisi terlalu panas dan lembab, panas dan kering, dingin dan lembab, dingin dan kering, dingin, panas, kering, dan lembab. Kondisi – kondisi di atas harus diatasi agar pertumbuhan jamur tidak terhambat.

Pada saat kondisi di rumah melebihi suhu set point, PC mengirim perintah ke MCU untuk menghidupkan blower dan blower akan mati jika suhu sudah berada pada range. Pada kondisi kurang dari set point, maka PC akan memerintahkan MCU untuk menghidupkan pemanas sehingga suhu di rumah jamur naik dan pemanas mati jika suhu sudah berada pada range yang ditentukan . Sedangkan saat kelembaban rumah jamur melebihi set point kelembaban, maka PC memerintahkan MCU untuk menghidupkan blower. Dan jika kelembaban di

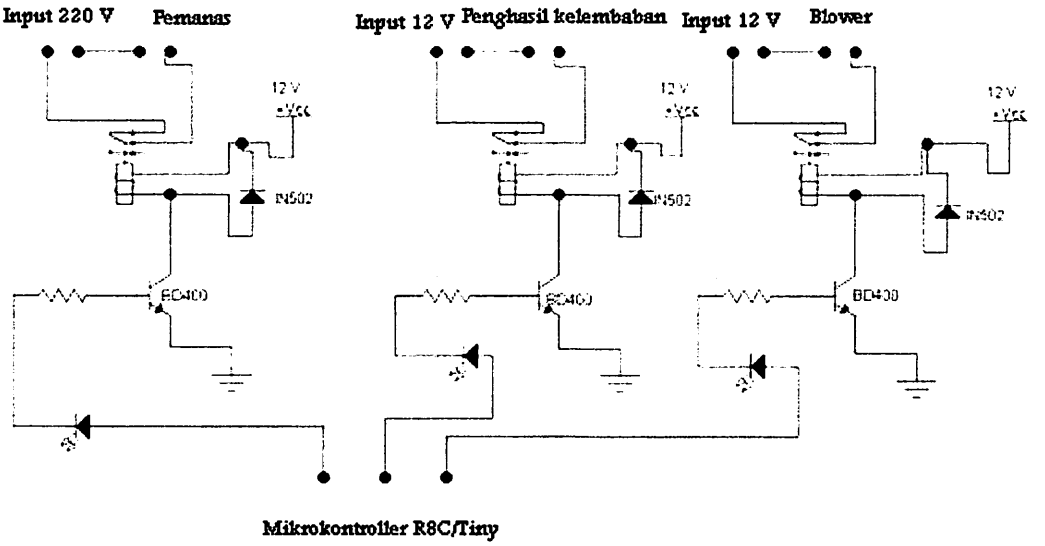
rumah jamur kurang dari set point kelembaban, maka penghasil kelembaban akan hidup dan melembabkan rumah jamur.

Mikrokontroler dihubungkan ke PC sebagai tampilan dan pengolah data dari sistem ini. Data suhu dan kelembaban rumah jamur dikirimkan ke PC dengan menggunakan interface RS 232 yang diperpanjang dengan RS 485. Alasan menggunakan RS 485 sebagai perpanjangan yaitu agar dapat dikontrol dari luar rumah jamur dengan jarak maksimum 1,2 Km. Pada PC data tersebut akan ditampilkan dalam bentuk tabel dan grafik suhu dan kelembaban. Apabila terjadi kondisi diluar range suhu dan kelembaban yang ditentukan, secara otomatis PC akan mengirimkan pemberitahuan melalui SMS ke pemilik rumah jamur.

3.3 Perencanaan Hardware

3.3.1 Perencanaan Rangkaian Driver Relay

Rangkaian ini berfungsi untuk menyambungkan dan memutuskan rangkaian yang ada di luar MCU. Untuk menjalankan relay diperlukan arus yang cukup besar karena port tidak mampu mensupply arus yang dibutuhkan relay. Maka dari itu diperlukan rangkaian driver relay yang menggunakan transistor sebagai penguat arusnya. Rangkaian driver relay tersusun seperti gambar di bawah ini.



Gambar 3.2 Rangkaian Driver Relay

Relay yang digunakan adalah relay 12 V(1A), relay cukup handal digunakan sebagai pengganti switch pengaktif kontaktor daya yang berdaya kecil. Resistansi kumparan relay adalah 156 Ω. Output TTL pada rangkaian control sebesar 2,4 Volt (Vbb) dan Vbe = 0,7 V. Sehingga resistor pada rangkaian switching dapat ditentukan :

Diketahui :

BD400, mempunyai :

Hfe = 100

Vcc = 12 V

Sehingga :

Ic = (Vcc – Vce) / Rc

$= (12 - 0,7) / 156$

$= 72,43$ mA

Ib $= Ic / Hfe$

$= 72,43 / 100$

$= 0,7243$ mA

Rb $= (Vbb - Vbe) / Ib$

$= (2,4 - 0,7) / 0,7243$

$= 2,35$ KΩ

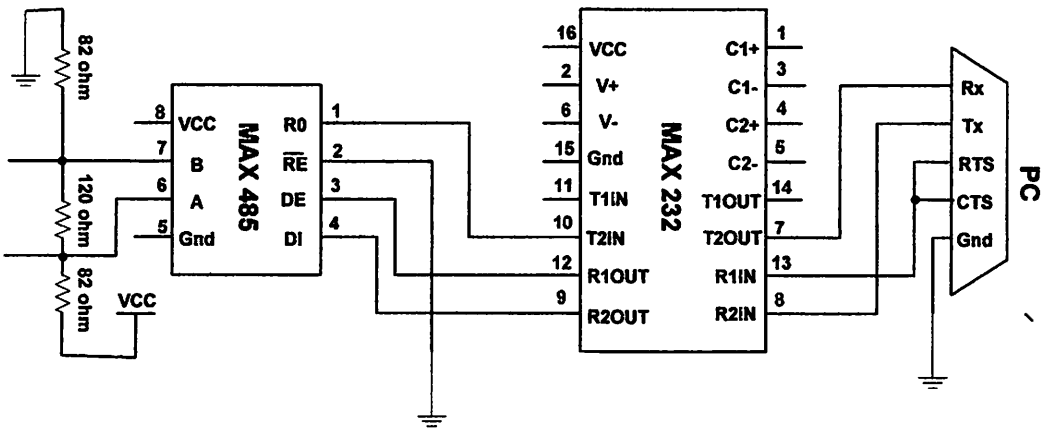
Pada relay juga dipasang sebuah dioda dengan bias reverse. Dioda ini digunakan untuk melindungi transistor dari pulsa back EMF yang dibangkitkan oleh induktansi kumparan relay ketika relay dimatikan.

3.3.2 Perencanaan Rangkaian Interface

Untuk dapat berkomunikasi antara MCU dengan PC yang berjarak sekitar 1,2 Km, perlu disesuaikan sistem signal yang dipakai yaitu dengan menggunakan standart RS 485 yang mempunyai konfigurasi half duplex. Pada perencanaan interface ini terbagi menjadi 2 bagian, yaitu :

- Pada PC

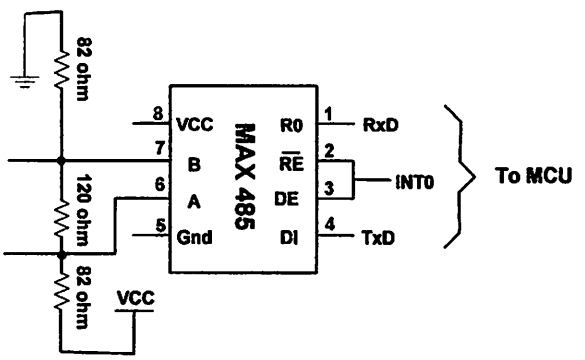
Pada PC (menggunakan COM / RS 232) perlu ditambahkan pengubah sistem sinyal dari RS 485 menjadi RS 232 (COM 1) untuk itu dibutuhkan rangkaian sebagai berikut :



Gambar 3.3 Rangkaian Konverter (RS 485 to RS 232)

- Pada MCU

Untuk komunikasi serial yang dimiliki oleh level TTL, maka dari MCU (hasil komunikasi) yang berupa RS 485 dengan menggunakan chip Max 485. Bentuk gambar sebagai berikut ;



Gambar 3.4 Rangkaian MCU to RS 485

3.3.3 Perencanaan Pengkondisi Sinyal

3.3.3.1 Perencanaan Pengkondisi Sinyal Pada Sensor Suhu

3.3.3.1.1 Perencanaan Rangkaian Sensor Suhu

Suhu ruang yang diinginkan untuk ruangan pembudidayaan jamur adalah sekitar 25 °C - 29 °C. Karena itu sensor yang digunakan harus memiliki daerah operasi yang mencakup suhu yang diinginkan yaitu 25 °C - 29 °C. Salah satu sensor suhu yang memiliki range tersebut diatas adalah adalah sensor suhu LM35.

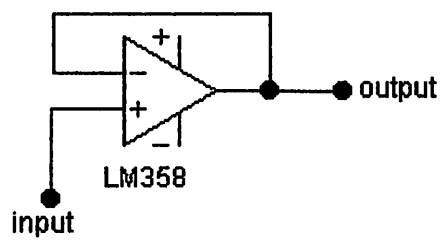
Sensor suhu merupakan tranduser yang berfungsi untuk mendeteksi perubahan suhu menjadi sinyal listrik dalam bentuk tegangan. Sensor suhu yang digunakan adalah IC LM35 keluaran National Semikonduktor dengan pertimbangan antara lain rangkaiannya sederhana, keluarannya linier terhadap suhu, terkalibrasi secara langsung dalam derajat celcius, harga terjangkau dan mudah didapatkan. Rangkaian Sensor suhu LM35 memiliki tegangan operasi yang umum digunakan yaitu 4 hingga 20 volt.

IC LM35 mempunyai impedansi masukan yang tinggi dan impedansi keluaran yang rendah, mempunyai sensitivitas ± 10mV/°C dan jangkauan operasi suhu -55°C - 150°C. Tegangan catu yang digunakan $5V_{dc}$. Tegangan keluaran dari sensor adalah 10 mV/°C. Jangkauan pengaturan suhu yang direncanakan adalah 25°C-29°C (suhu ruang jamur), sehingga keluaran sensor adalah 250mV-290mV . Keluaran dari sensor suhu selanjutnya dihibungkan ke pengkondisi sinyal. Tabel 3.1 menunjukkan hubungan antara suhu dan tegangan keluaran.

3.3.3.1.2 Rangkaian Pengkondisi Sinyal Analog

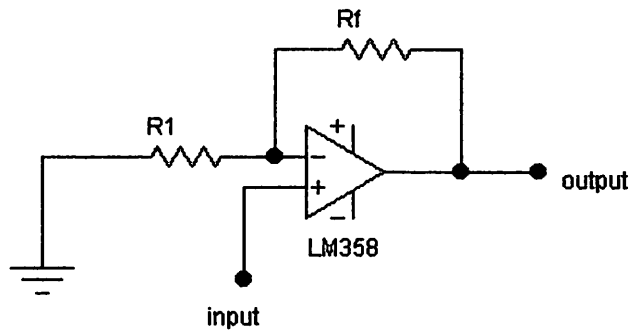
Untuk memperoleh resolusi suhu yang diinginkan dan dapat dibaca oleh ADC, maka keluaran sensor suhu harus dihubungkan dengan rangkaian pengkondisi sinyal analog. Pengkondisi sinyal analog terdiri dari penguat penyangga (*buffer)* dan penguat tak membalik *(non-inverting amplifier).* Rangkaian pengkondisi sinyal analog berupa penguat *buffer* dan penguat *non-inverting* ditunjukkan gambar di bawah.

- Rangkaian Buffer



Gambar 3.5 Rangkaian Buffer

Penguat penyangga atau pengikut tegangan berfungsi untuk merubah tegangan pada impedansi tinggi menjadi tegangan yang sama pada impedansi rendah. Karena sensor suhu IC LM35 menghasilkan tegangan yang berubah-ubah sesuai dengan perubahan suhu, penguat penyangga memastikan agar sinyal ini tidak terpengaruh oleh pembebanan dari penguat berikutnya.



Gambar 3. 6 Penguat Non Inverting

Keluaran dari penguat penyangga dikuatkan lagi oleh penguat *non-inverting*. Besarnya penguatan dari penguat *non-inverting* ini akan ditentukan oleh besarnya resolusi pengukuran suhu yang diinginkan. Jika keluaran tegangan dari sensor adalah 10mV/°C maka besar tegangan ini adalah ½ LSB dari resolusi ADC dengan V_{ref} =5 volt, dimana resolusi 1 LSB adalah:

$$\text{Resolusi} = \frac{5V}{2^8 - 1} = 0{,}0196V = 19{,}6mV \approx 20\ mV$$

Sistem diharapkan agar kenaikan setiap °C setara dengan data 1 bit pada respon ADC, maka keluaran dari sensor suhu LM35 memerlukan penguatan sebesar :

$$A = \frac{resolusi\ ADC\ x 1 bit}{sensitifitas\ sensor}$$

$$= \frac{20x1}{10} = 2x$$

Untuk memperoleh penguatan sesuai dengan yang diinginkan maka nilai R_1 dan R_f dapat dicari dengan menggunakan persamaan sebagai berikut :

$$A = 1 + \frac{R_f}{R_1}$$

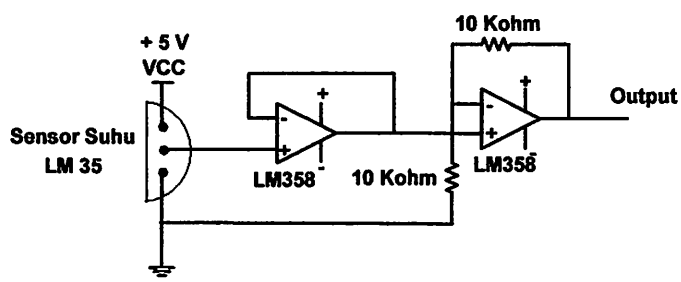
$$2 = 1 + \frac{R_f}{R_1}$$

$$\frac{R_f}{R_1} = 1$$

Jadi nilai $R_f = R_1$

Dipilih nilai R_f = 10 KΩ.

Maka R_1 = 10 K Ω.



Gambar 3.7 Rangkaian Sensor Suhu

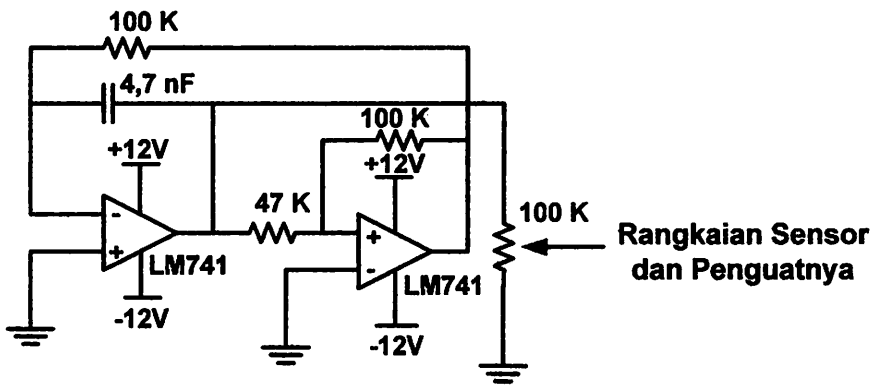
3.3.3.2 Perencanaan Pengkondisi Sinyal Pada Sensor Kelembaban

3.3.3.2.1 Perencanaan Rangkaian Sensor Kelembaban

Dalam perancangan ini, nilai kelembaban yang diinginkan adalah 60% - 90% RH. Sehingga sensor kelembaban yang digunakan harus memiliki area operasi yang mencakup nilai tersebut. Dalam hal ini sensor HS15P memiliki kriteria tersebut dengan area operasi 20% - 100% RH.

3.3.3.2.2 Rangkaian Pembangkit Oscilator

Untuk membangkitkan sensor HS15P ini , diperlukan tegangan AC 1 Vrms dengan frekuensi 1 kHZ. Oleh karena itu, perlu dibuat rangkaian oscilator 1 kHZ yang ditunjukkan pada gambar 3.8 dibawah ini.



Gambar 3.8 : Rangkaian Oscilator 1 kHz

Bila ditentukan R = 47 kΩ dan nR = 100 kΩ, maka nilai n dapat ditentukan :

$R = 47\ k\Omega$

$nR = 100\ k\Omega$

$$n = \frac{nR}{R}$$

$$= \frac{100k}{47k} = \frac{100}{47}$$

$$= 2{,}127$$

Persamaan frekuensi untuk rangkaian oscilator ini adalah [Robert F. Cuoghlin & Frederick F. . Driscoll, 1992 : 125].

$$f = \frac{n}{4R1C}$$

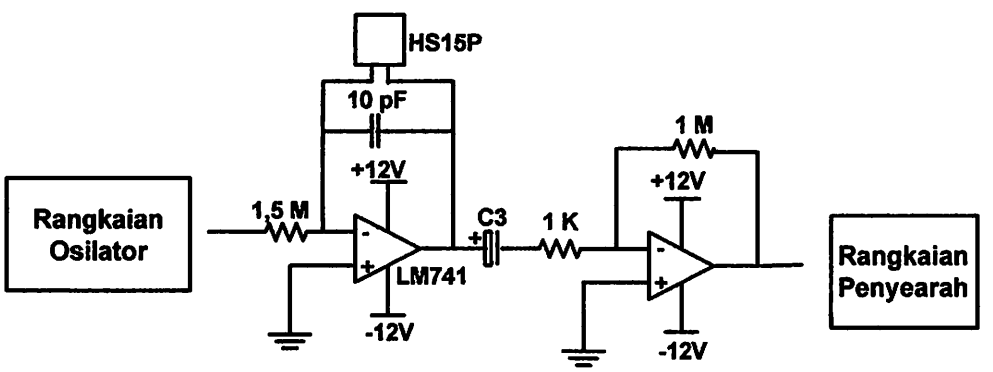
Pada perencanaan ditetapkan nilai

R_1 = 100 kΩ, C = 4,7 nF, R = 47 kΩ, dan nR = 100 kΩ.

Sehingga diperoleh n = 2 dan $f = \frac{2}{4.10^5.4{,}7.10^{-9}} = 1063{,}83\text{Hz}$

3.3.3.2.3 Rangkaian Sensor HS15P dan Penguatnya

Berikutnya adalah merancang rangkaian sensor kelembaban dan penguat tegangannya, seperti pada gambar berikut :



Gambar 3.9 : Rangkaian Sensor Kelembaban HS15P dan Penguatan Pembaliknya

Besarnya penguatan rangkaian Op-Amp sebagai penguat pembalik ditentukan dengan persamaan [Robert F. Cuoghlin & Frederick F. Driscoll, 1992 : 38].

$$A = -\frac{R_f}{R_i}$$

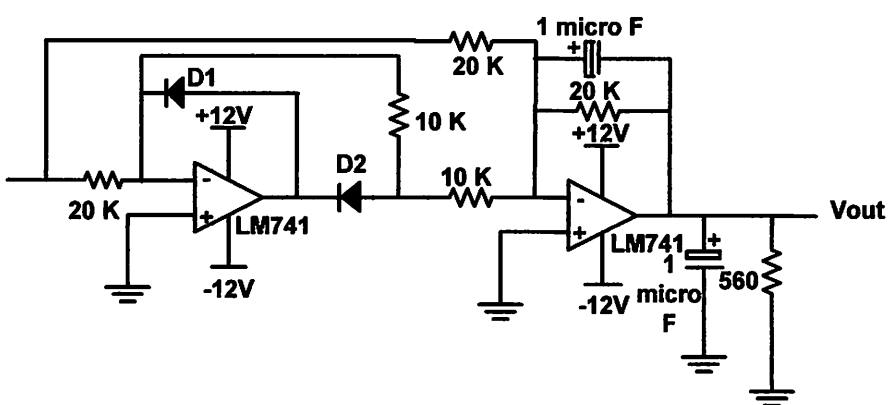
Pada Op-Amp pertama, besar R_f tergantung dari bacaan sensor kelembaban HS15P sedangkan R_1 ditentukan sebesar 1,5 MΩ. Penguatan yang diberikan oleh Op-Amp kedua dengan R_1 = 1 kΩ adalah sebesar :

$$A = -\frac{1M\Omega}{1K\Omega} = 1000$$

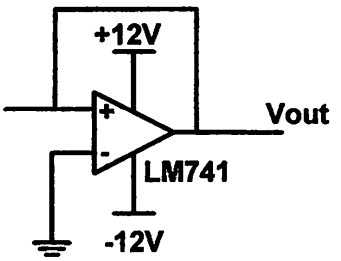
Diantara kedua Op-Amp dipasang kapasitor dengan tujuan sebagai filter untuk menghalangi lewatnya sinyal DC.

3.3.3.2.4 Rangkaian Penyearah

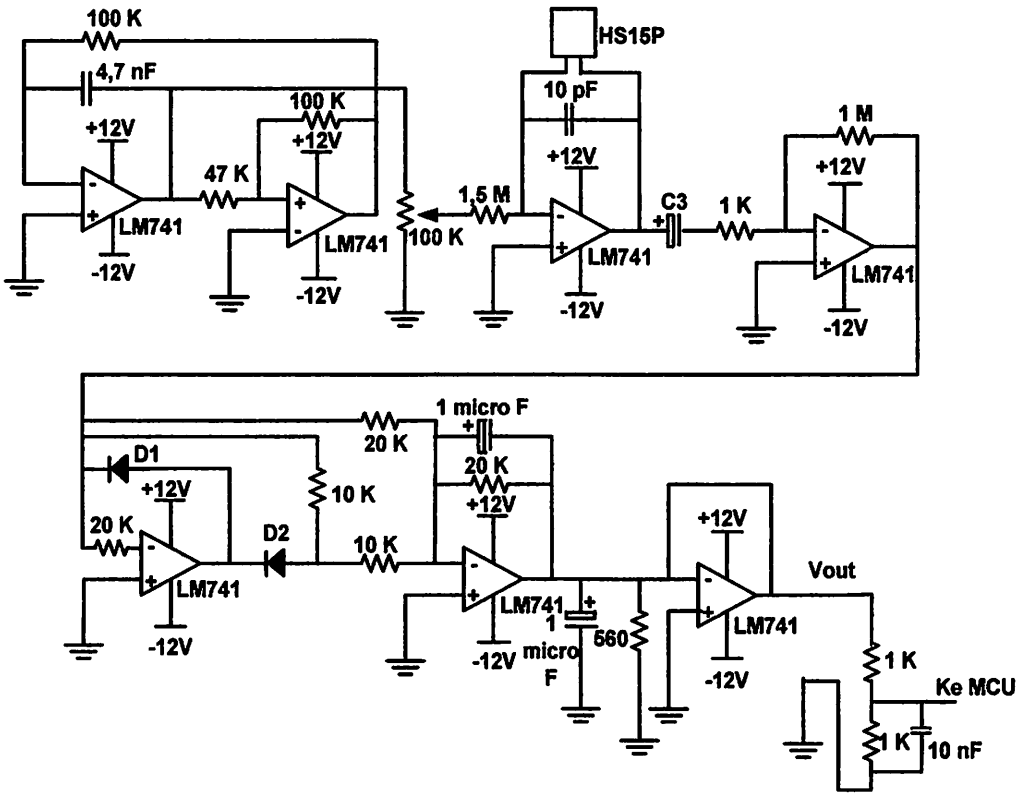
Langkah berikutnya adalah merancang rangkaian penyearah seperti pada gambar . Rangkaian penyearah ini terdiri dari dua Op-Amp dan dua dioda yang kemudian diikuti rangkaian pengikut tegangan. Nilai R adalah sebesar 20 kΩ.



Gambar 3.10 : Rangkaian Penyearah Gelombang Penuh



Gambar 3.11 : Rangkaian Pengikut Tegangan



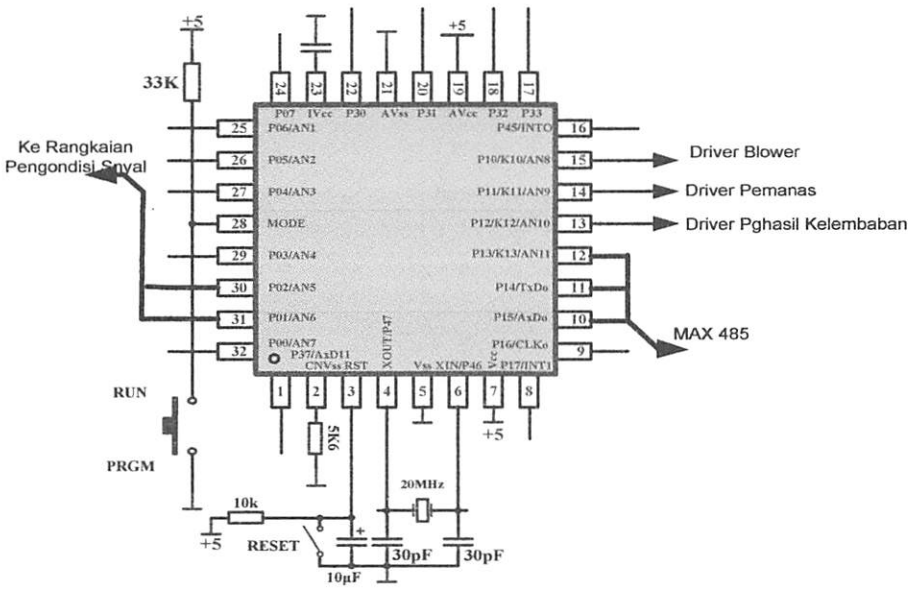
Gambar 3.12 : Rangkaian Sensor Kelembaban Keseluruhan

3.3.5 Perencanaan MCU Renesas R8C/Tiny

MCU merupakan rangkaian slave yang berfungsi sebagai penerima perintah-perintah dari PC. Beberapa alasan mengapa kita menggunakan MCU

Renesas R8C/Tiny adalah rangkaiannya yang praktis karena tersusun dalam satu modul yang sangat mudah kita gunakan, sederhana dan tidak banyak memakan tempat. Selain itu rangkaian ADC nya sudah ada pada jenis Mikrokontroler ini sehingga kita tidak perlu membuat rangkaian ADC baru.

Berikut ini gambar rangkaian MCU Renesas R8c/Tiny beserta hubungannya dengan rangkaian lain :



Gambar 3.13 Rangkaian Koneksi Renesas R8C/Tiny

Fungsi MCU yang digunakan sebagai slow aplikasi MCU yang mengontrol suhu dan kelembaban di rumah jamur adalah sebagai berikut :

- Port 0 memiliki fungsi sebagai berikut :
    - Port 0.1 digunakan sebagai input ADC dari sensor suhu.
    - Port 0.2 digunakan sebagai input ADC dari sensor kelembaban.

- Port 1 memiliki fungsi sebagai berikut :
    - Port 1.0 sebagai pengontrol sirkulasi udara
    - Port 1.1 sebagai pengontrol pemanas
    - Port 1.2 sebagai pengontrol kelembaban
    - Port 1.3 sebagai kontrol
    - Port 1.4 (TxD) untuk pengiriman data ke PC
    - Port 1.5 (RxD) untuk penerimaan data dari PC

3.4 Perencanaan Software

3.4.1 MCU

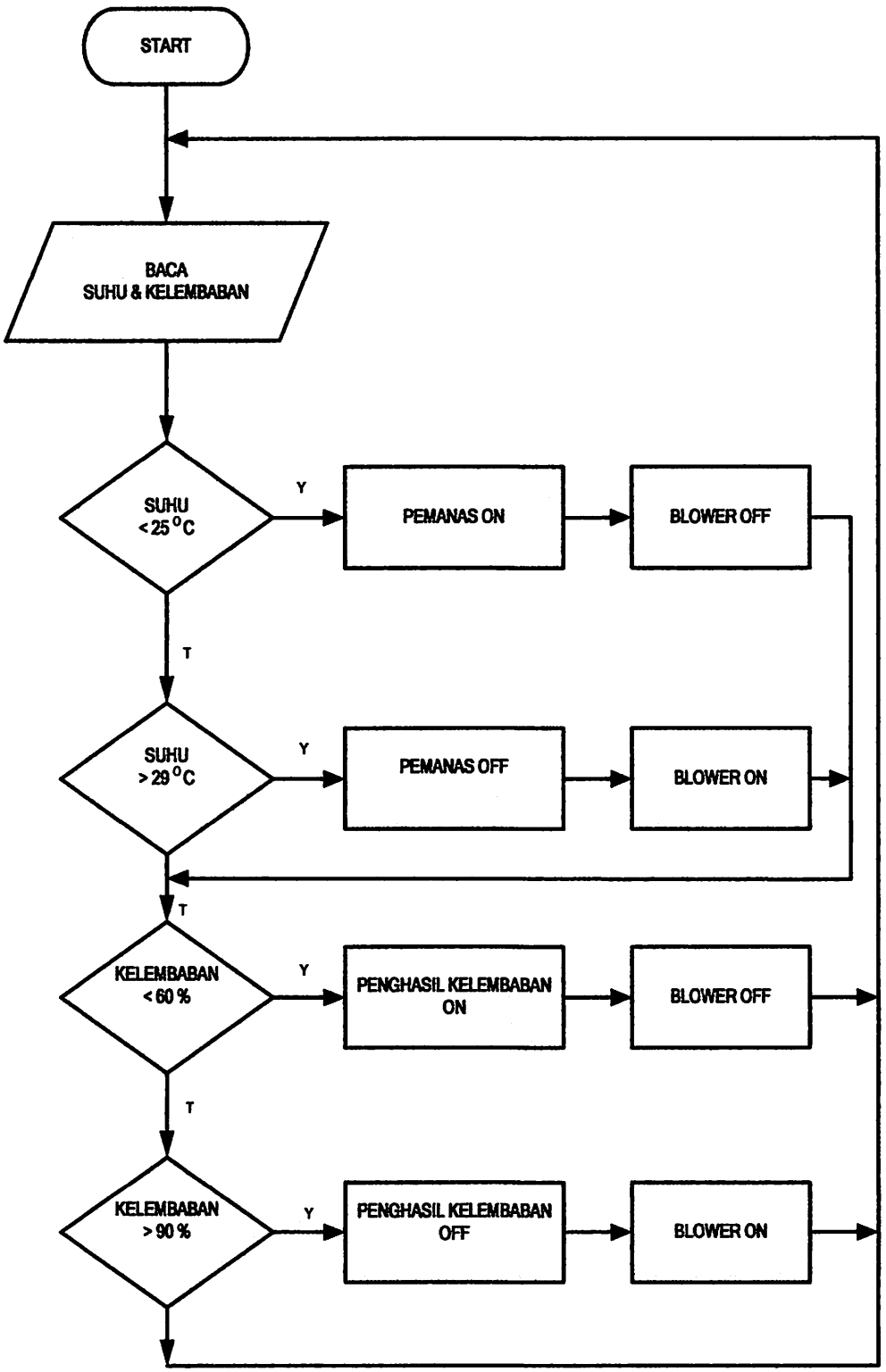
Proses penyimpanan program ke memori I/O di MCU adalah :

- P0 sebagai input dari sensor
    - P0.1 sebagai input data dari sensor suhu
    - P0.2 sebagai input data dari sensor kelembaban
- P1 digunakan sebagai
    - Port 1.0 sebagai pengontrol sirkulasi udara
    - Port 1.1 sebagai pengontrol pemanas
    - Port 1.2 sebagai pengontrol kelembaban
    - Port 1.3 sebagai kontrol
    - Port 1.4 (TxD) untuk pengiriman data ke PC
    - Port 1.5 (RxD) untuk penerimaan data dari PC

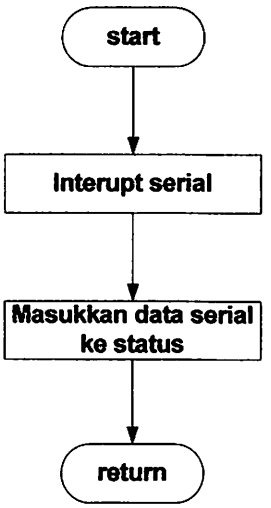
PC memerintahkan MCU untuk mengambil data ADC secara periodik. Setelah MCU mengambil data dari ADC, PC membandingkan data dengan set point yang telah ditentukan pada PC.

Jika suhu di rumah jamur kurang dari set point, maka PC akan memerintahkan untuk menghidupkan heater. Setelah suhu yang sesuai diperoleh, heater dimatikan. Apabila suhu di rumah jamur melebihi suhu setting, maka PC akan memerintahkan MCU untuk menyalakan blower dan mengeluarkan udara panas dan menggantinya dengan udara dari luar.

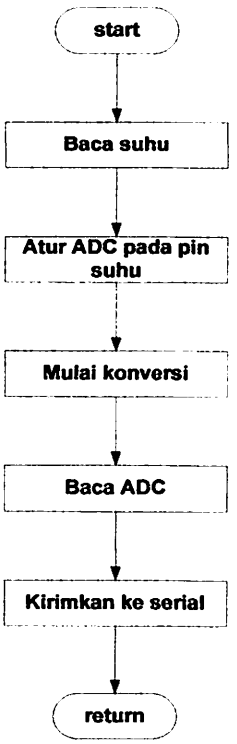
Untuk kelembaban, apabila kelembaban di rumah jamur melebihi set point, maka PC akan memerintahkan MCU untuk menyalakan blower agar udara yang memiliki kelembaban berlebih dapat dikeluarkan. Namun jika kelembaban kurang dari ketentuan, sprayer dinyalakan. Sprayer akan mati bila kelembaban telah sesuai dengan ketentuan.



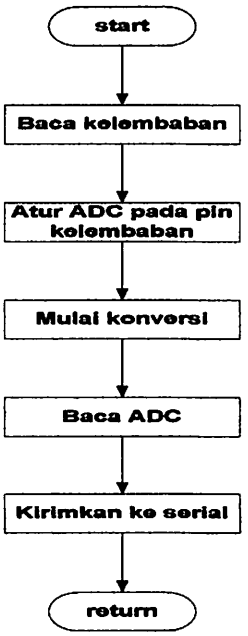
Gambar 3.14 Flow Chart Sistem MCU



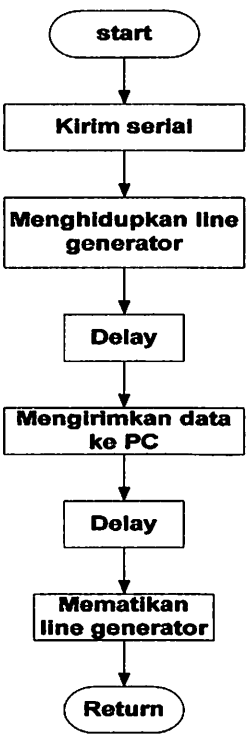
Gambar 3.15 Flow Chart Unit Serial



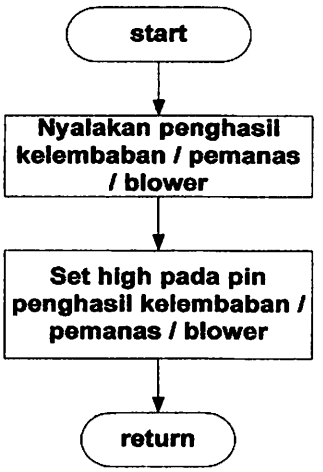
Gambar 3.16 Flow Chart Baca ADC Data Suhu



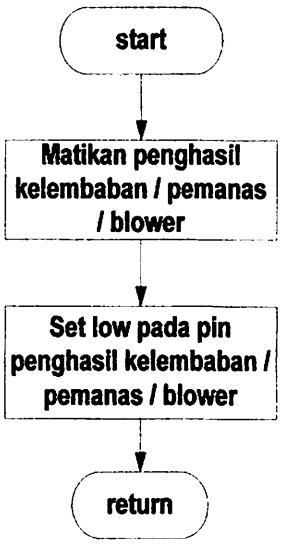
Gambar 3.17 Flow Chart Baca ADC Data Kelembaban



Gambar 3.18 Flow Chart Fungsi Kirim Serial



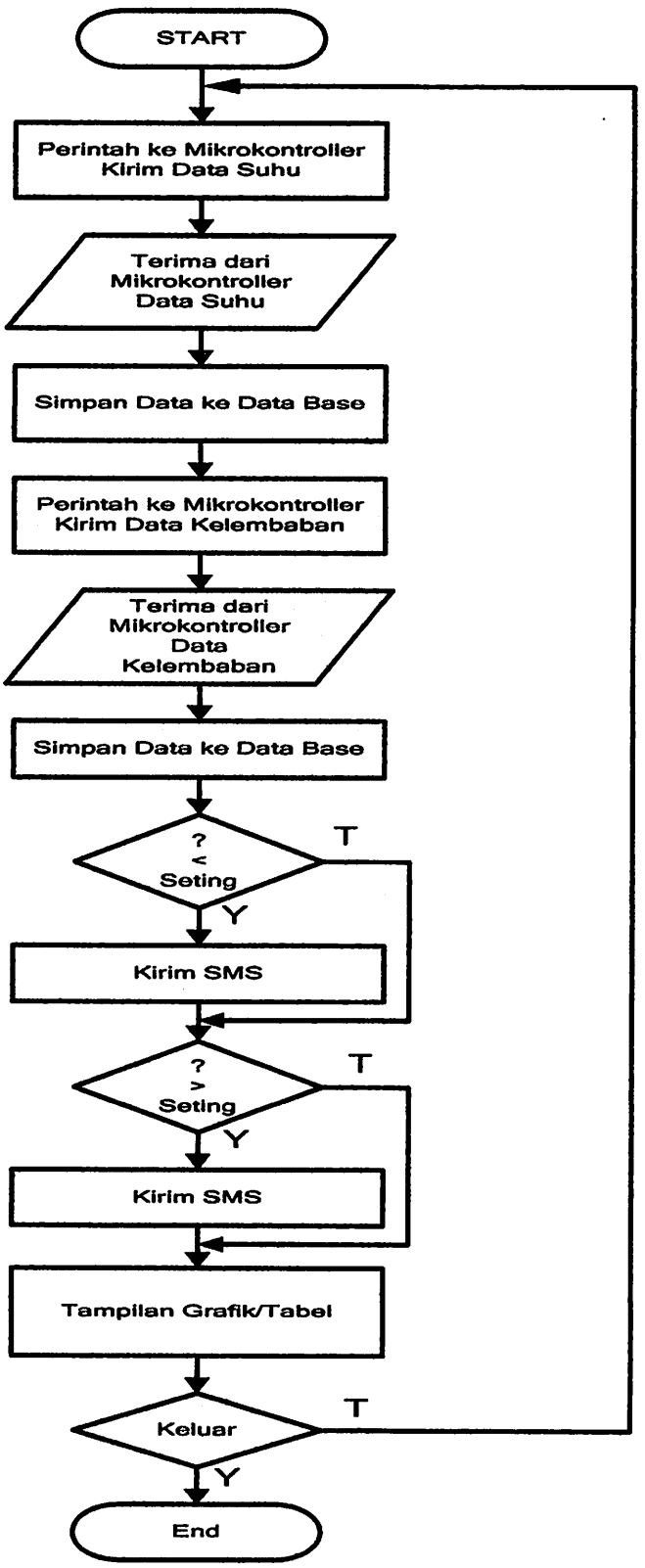
Gambar 3.19 Flow Chart Sistem On Mekanik



Gambar 3.20 Flow Chart Sistem Off Mekanik

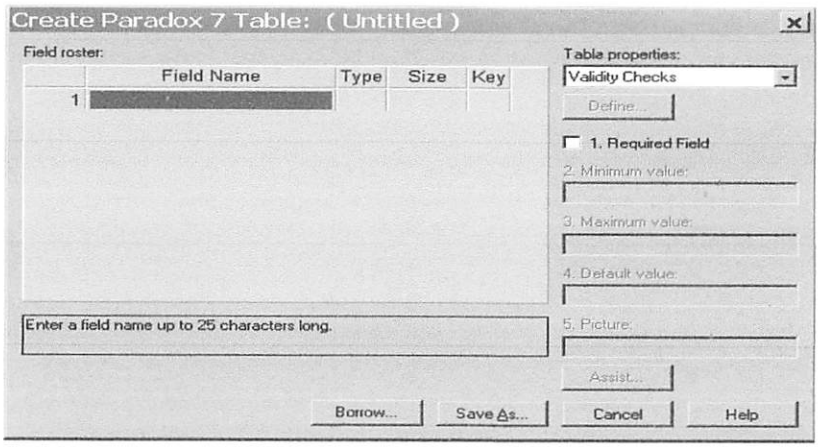
3.4.2 PC (*Personal Computer*)

Perencanaan program tampilan pada PC ini menggunakan Bahasa Pemrograman Borland Delphi. Pada PC dibuat tampilan dalam bentuk tabel dan grafik dari data perubahan suhu dan kelembaban pada rumah jamur.



Gambar 3.21 Flow Chart Tampilan pada PC

Adapun pembuatan tampilan database desktop adalah sebagai berikut :



Gambar 3.22 Tampilan Data Base Desktop

Adapun langkah-langkah pembuatan database dekstop adalah sebagai berikut :

1) Membuat *table database*. Sorot menu utama *windows* **Start**. Kemudian pilih **File** → **New** → **Table** → **OK**
2) Untuk tampilan yang diharapkan adalah No, Tanggal, Waktu dan Suhu atau Kelembaban dengan ketentuan sebagai berikut :

File Name	**Type**
No	+ (auto increment)
Tanggal	D (Date 0)
Waktu	T (Time)
Suhu	N (Number)
Kelembaban	N (Number 0)

3) Menghubungkan database desktop dengan Delphi.

Pada form Delphi, klik icon BDE ambil **table**, Data access → **data source** untuk menghubungkan table dengan grid, Data control → **grid**

4) Membuat grafik

Pada **dB Chart** klik kanan lalu pilih **Edit Chart** → **Add** → **Line** → **OK.**

Sorot menu **Series**, kemudian klik **Data Source** → **Data Set** (data yang ada di table yang sudah dibuat) → **Close**

Langkah-langkah membuat program Delphi :

1. Membuka menu **File/New**, pilih icon **Application** dari **New Items** dan klik.
2. Memakai *project inspector* untuk mengubah beberapa *properties* dari *form* dan komponen-komponennya.
3. Menambahkan komponen ke *form* yang sudah disediakan oleh Delphi, kemudian ulangi langkah 2.
4. Setelah mengubah *properties*, pilih dan klik *tab* **Event**. Pilih event **Onclick**, klik ganda kolom isinya.
5. Kemudian masuk ke *code editor* untuk memasukkan program. Perhatikan nama prosedurnya, lalu mengisikan atau, mengetikkan programnya diantara **begin...end** yang sudah tersedia.
6. Selanjutnya menjalankan *project*. Pilih **Run** atau tekan tombol **F9**.

Pembuatan tampilan pada PC memerlukan beberapa komponen Delphi, yaitu :

1. *Data Source*

   Komponen ini digunakan untuk menghubungkan komponen *Table* atau *Query* dengan komponen tempat data akan ditampilkan (NVC).

2. *Table*

   Komponen ini digunakan untuk menghubungkan tabel pada suatu *database* dengan program yang dibuat (NVC).

3. *Query*

   Komponen ini digunakan untuk membuat dan mengeksekusi SQL *query* pada SQL *Server Database* atau Database lokal (NVC).

4. *Comm Serial*

   Komponen ini digunakan untuk komunikasi serial.

5. *Edit*

   Komponen ini digunakan sebagai input / output satu baris teks. Pemakai program dapat mengubah teks ini (VC).

6. *Button*

   Komponen ini digunakan untuk mmbuat tombol, sehingga pemakai bisa memasukkan perintah pada program aplikasi dengan menekan (klik) tombol ini.

7. *Combo Box*

   Komponen ini digunakan untuk menampilkan data suatu field database dengan versi komponen ComboBox (VC).

8. *DB Grid*

   Komponen ini digunakan untuk menampilkan data-data dalam bentuk baris dan kolom.

9. *DB Chart*

   Komponen ini digunakan untuk menampilkan data-data dalam bentuk grafik.

BAB IV

PENGUKURAN DAN PENGUJIAN ALAT

Pada bab ini membahas tentang pengukuran dan pengujian alat yang dirancang, dimana meliputi perangkat keras (hardware) dan perangkat lunak (software). Untuk mengetahui sistem yang dirancang sesuai dengan fungsi yang diharapkan, dilakukan pengujian terhadap sistem aplikasi tersebut baik secara keseluruhan atau subsistem. Berikut penjelasan mengenai prosedur pengukuran dan dan data hasil pengujian.

4.1 Pengukuran

4.1.1 Rangkaian Sensor dan Pengondisi Sinyal

- Tujuan Pengukuran

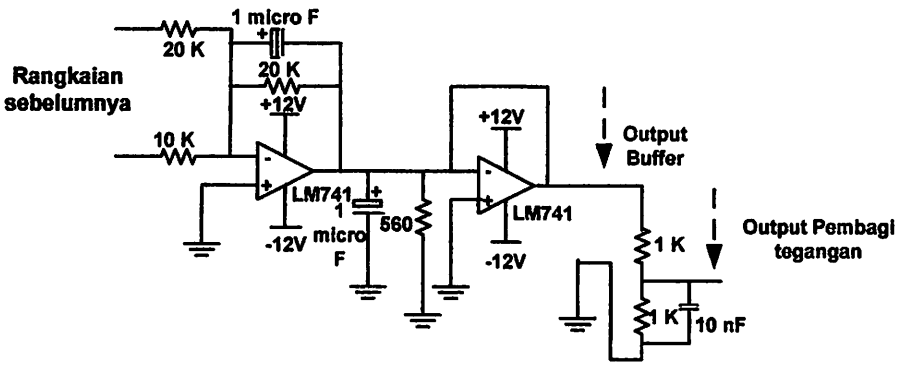
  1. Mengetahui tegangan keluaran sensor suhu dan kelembaban
  2. Mengetahui prinsip kerja rangkaian pengondisi sinyal

- Peralatan yang digunakan

  1. Power Supply
  2. Multimeter analog / digital
  3. Rangkaian Pengondisi sinyal
  4. Rangkaian Sensor suhu
  5. Rangkaian Sensor Kelembaban

4.1.1.1 Pengujian Rangkaian Sensor Kelembaban

- Diagram pengukuran



Gambar 4.1 Diagram Pengujian Rangkaian Pengondisi Sinyal dan Rangkaian Sensor Kelembaban

- Tabel Data Pengukuran

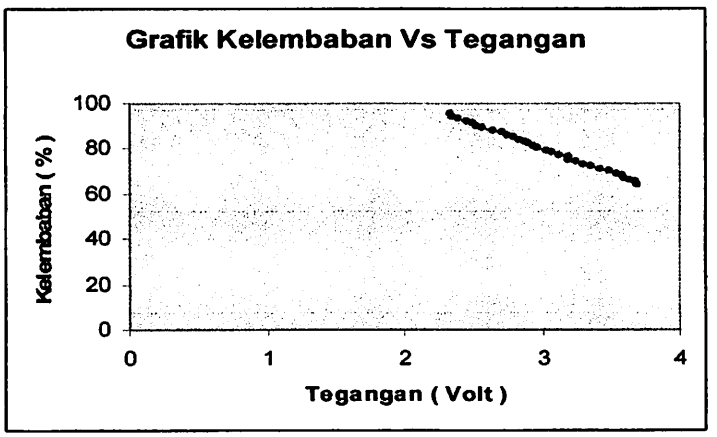
Tabel 4.1 Tabel Pengujian kelembaban

Kelembaban (%)	Pengukuran		Perhitungan
	Output Buffer (V)	Output Pembagi Tegangan (V)	Output Pembagi Tegangan (V)
64	7,35	3,69	3,675
65	7,34	3,68	3,67
66	7,27	3,64	3,635
67	7,19	3,60	3,595

68	7,17	3,58	3,585
69	7,11	3,55	3,555
70	6,99	3,49	3,495
71	6,85	3,42	3,425
72	6,74	3,36	3,37
73	6,62	3,30	3,31
74	6,50	3,25	3,25
75	6,37	3,19	3,185
76	6,41	3,20	3,205
77	6,25	3,13	3,125
78	6,15	3,08	3,075
79	6,07	3,04	3,035
80	5,91	2,96	2,955
81	5,87	2,94	2,935
82	5,83	2,90	2,915
83	5,75	2,87	2,875
84	5,66	2,83	2,83
85	5,56	2,79	2,78
86	5,52	2,75	2,76
87	5,43	2,71	2,715
88	5,29	2,65	2,645
89	5,14	2,57	2,57

90	5,07	2,53	2,535
91	4,99	2,49	2,495
92	4,91	2,45	2,455
93	4,79	2,39	2,395
94	4,70	2,35	2,35
95	4,65	2,33	2,325

Berikut ini adalah grafik antara persen kelembaban dengan tegangan keluaran.



Gambar 4.2 Grafik Kelembaban dan Tegangan

Di bawah ini merupakan output osilator 1 KHz pada rangkaian sensor kelembaban. Pada tampilan osiloskop di bawah, tampak keluaran sinyal dari rangkaian osilator 1 KHz yang digunakan sebagai pencatu dari sensor kelembaban HS15P. Pada saat pengujian, parameter yang diberikan adalah sebagai berikut :

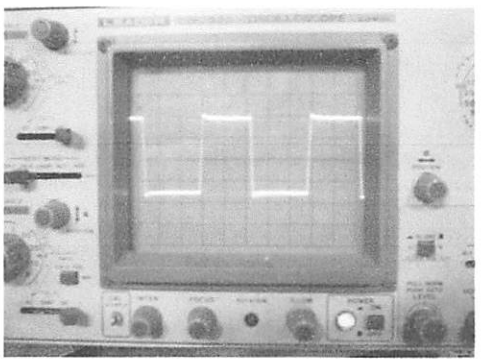
Volt / Div = 5 V

Time / Div =0,2 ms

Sehingga , f = 1 / T

= 1 / (5 x 0,2 ms)

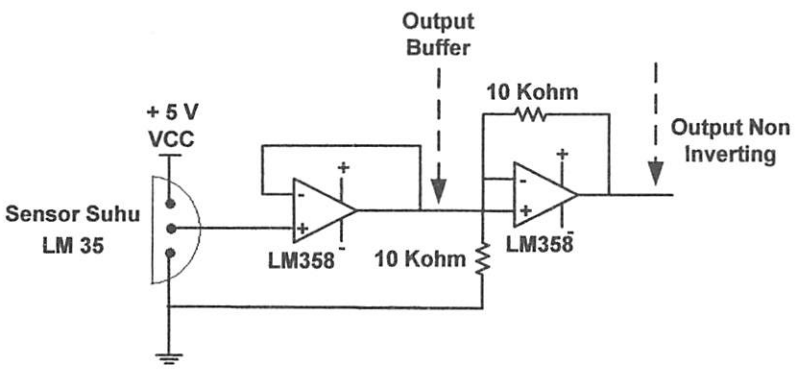
= 1 KHz



Gambar 4.3 Gambar Sinyal Osilator 1 KHz

4.1.1.2 Pengujian Rangkaian Sensor Suhu

- Diagram Pengukuran



Gambar 4.4 Gambar Pengujian Rangkaian Pengondisi Sinyal dan Rangkaian Sensor Suhu

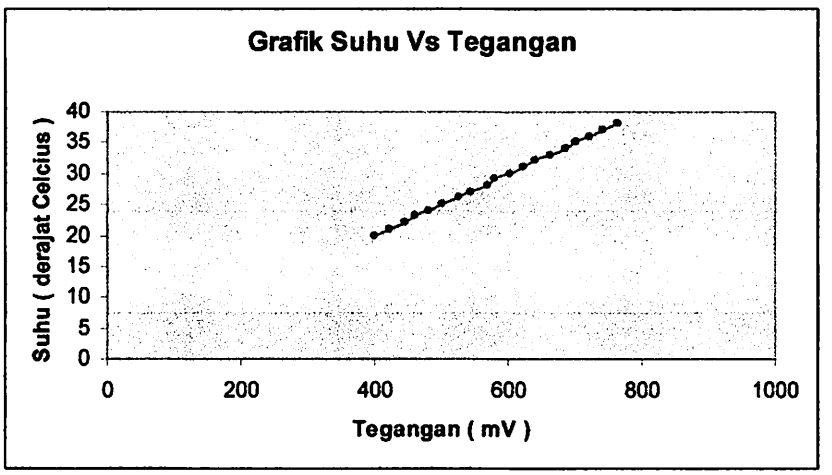
- Tabel Data Pengukuran

Tabel 4.2 Data Pengujian suhu

Suhu (°C)	Pengukuran		Perhitungan
	Vo Buffer (mV)	Vo Non Inverting (mV)	Vo Non Inverting (mV)
20	204,5	401	409
21	213	422,5	426
22	223,5	445	447
23	233	460	466
24	245,5	482	491
25	252	502	504
26	262,5	525,5	525
27	273,5	543,5	547
28	285,5	570,4	571
29	292	580	584
30	303,5	602,3	607
31	312,5	623	625
32	322	640	644
33	332,5	663	665
34	344	685	688

35	352,5	701	705
36	362,5	722,6	725
37	371,5	743	743
38	384	765	768

Berikut ini adalah grafik antara suhu dan tegangan keluaran dari rangkaian sensor suhu.



Gambar 4.5 Grafik Suhu dan Tegangan

- Tabel Data Pengukuran

Tabel 4.3 Data Pengujian Penurunan Nilai Kelembaban Terhadap Waktu

Kelembaban (%)	**Waktu (detik)**
97 – 96	46
96 – 95	17

95 – 94	18
94 – 93	10
93 – 92	24
92 – 91	17
91 – 90	16
90 – 89	12
89 – 88	11
88 – 87	13
87 – 86	11
86 – 85	8
85 – 84	5
84 – 83	8
83 – 82	9
82 – 81	12
81 – 80	7
80 – 79	16
79 – 78	5
78 – 77	6
77 – 76	13
76 – 75	6
75 – 74	7
74 – 73	9

73 – 72	6
72 – 71	5
71 – 70	7
70 – 69	9
69 – 68	5
66 – 67	6
67 – 66	6
66 - 65	23

Tabel 4.4 Pengujian Penurunan Suhu Terhadap Waktu

Suhu (°C)	**Waktu (detik)**
41 – 40	12
40 – 39	26
39 – 38	14
38 – 37	15
37 – 36	11
36 – 35	34
35 – 34	27
34 – 33	29
33 – 32	21

32 – 31	15
31 – 30	45
30 – 29	43
29 – 28	23
28 – 27	49
27 - 26	45

- Analisa Data

Dari pengujian kemampuan blower dalam mengurangi kelembaban dan suhu dalam ruangan, didapatkan hasil kurang dari 1 menit untuk mengurangi kelembaban tiap %-nya dan suhu / °C-nya.

4.1.1.3 Pengujian Penghasil kelembaban

Pengujian ini dilakukan untuk mengetahui seberapa besar kemampuan penghasil kelembaban dalam menambah kelembaban di rumah jamur agar sesuai dengan set point yang telah ditentukan. Alat ukur kelembaban yang digunakan adalah *Thermohygrometer*, yaitu suatu alat ukur yang terdiri dari dua *thermometer* yang salah satu *thermometer* tersebut dibungkus kain basah. Selisih dari pengukuran kedua *thermometer* tersebut merupakan nilai kelembaban relative.

- Tabel Pengukuran

Tabel 4.5 Pengujian Kenaikan Kelembaban Terhadap Waktu

Kelembaban (%)	Waktu (detik)
64 – 65	29
65 – 66	46
66 – 67	18
67 – 68	29
68 – 69	17
69 – 70	17
70 – 71	6
71 – 72	17
72 – 73	18
73 – 74	12
74 – 75	6
75 – 76	17
76 – 77	18
77 – 78	7
78 – 79	8
79 – 80	17
80 – 81	12
81 – 82	17

82 – 83	17
83 – 84	41
84 – 85	11
85 – 86	27
86 – 87	56
87 – 88	46
88 – 89	55
89 – 90	23
90 – 91	12
91 – 92	32
92 – 93	23
93 – 94	30
94 - 95	19

4.1.1.4 Pengujian Pemanas

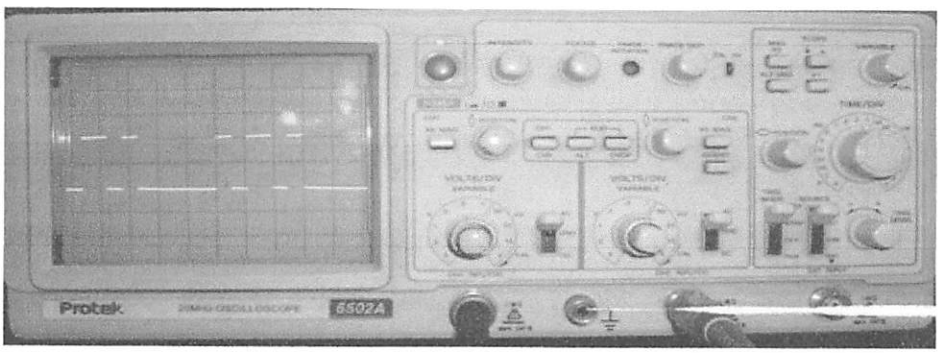
Table 4.6 Pengujian Kenaikan Suhu Terhadap Waktu

Suhu (°C)	Waktu (Detik)
27 – 28	15
28 – 29	10
29 – 30	11
30 – 31	10
31 – 32	9
32 – 33	7
33 – 34	8
34 – 35	10
35 – 36	9
36 – 37	8
37 – 38	5
38 – 39	9
39 – 40	7
40 – 41	5
41 – 42	3
42 - 43	5

4.2 Pengujian software

4.2.1 Pengujian MCU

Pengujian pengiriman data dari MCU ke PC ini dimaksudkan untuk mengetahui apakah data yang berasal dari MCU dapat dikirimkan dengan baik ke PC. Pengetesan dilakukan pada port P1.4 MCU. Sehingga didapat sinyal data sebagai berikut :

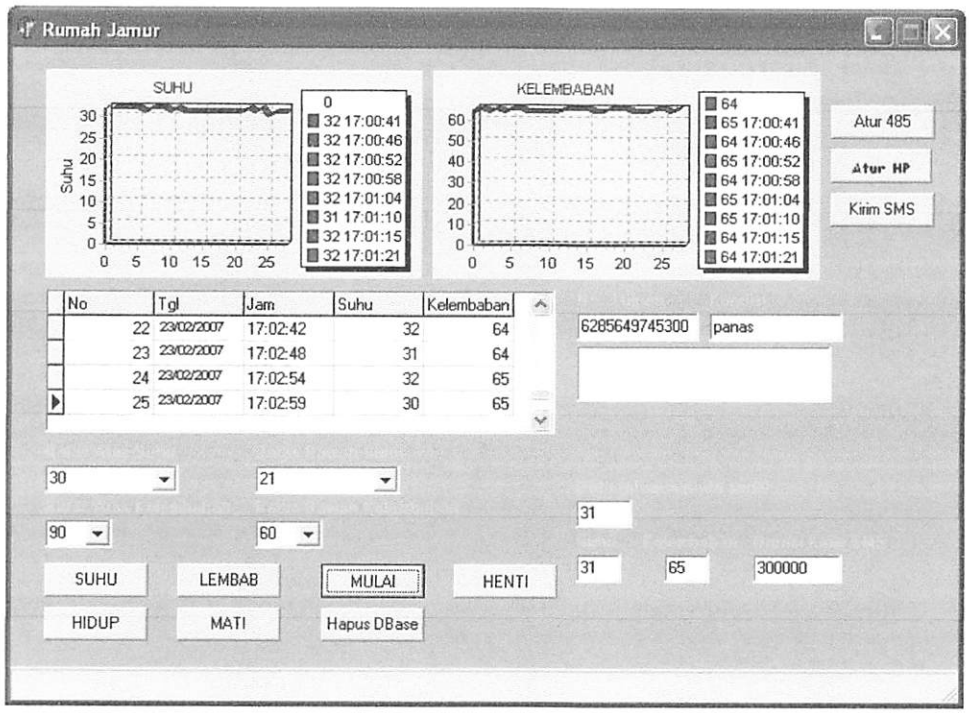


Gambar 4.6 Sinyal pengiriman data dari MCU ke PC

Dari hasil pengujian di atas menunjukkan bahwa pengiriman data yang berasal dari MCU telah dapat dikirimkan ke PC. Pengetesan dilakukan dengan menggunakan osiloskop pada *time/div* = 0,2 ms, *volt/div* = 0,2 volt dan dari hasil pengujian diatas dapat dihitung frekuensi ≈ 862 Hz.

4.2.2 Pengujian PC

Data dari pengukuran suhu dan kelembaban rumah jamur yang ada di lokasi, dikirimkan oleh MCU ke PC dengan tampilan sebagai berikut :

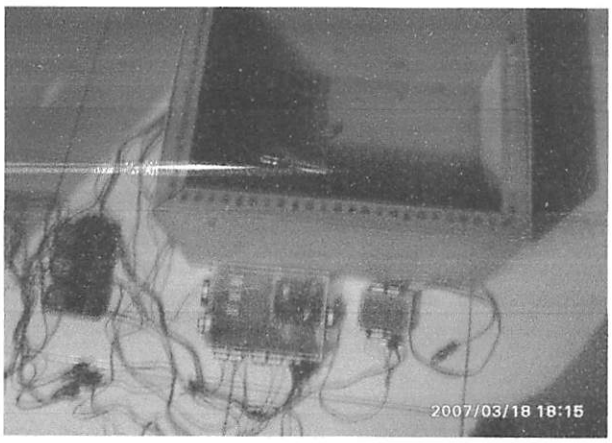


Gambar 4.7 Tampilan Record Data dan Grafik

Dari tampilan di atas, data hasil pengukuran suhu dan kelembaban akan ditampilkan pada tabel *record* data. Pada grafik, terdapat grafik perubahan suhu dan kelembaban. Perubahan atau gerakan pada grafik tergantung pada data yang tersimpan pada record data atau akan mengikuti data pada *record* data. Gambar 4.8 merupakan tampilan hasil pada waktu yang tercatat atau terekam.

4.3 Pengujian Sistem Keseluruhan

Pengujian ini bertujuan untuk mengetahui apakah sistem dapat berjalan dengan baik. Berikut ini merupakan gambar rangkaian sistem secara keseluruhan.



Gambar 4.8 Sistem Mekanik dan Hardware

Gambar 4.9 Contoh Tampilan Pesan Kondisi Ekstrim

BAB V

PENUTUP

5.1 Kesimpulan

Dari hasil perencanaan dan pembahasan laporan akhir ini dapat diperoleh kesimpulan :

1. Penambahan dan pengurangan kadar kelembaban pada sistem ini tidak stabil karena disebabkan oleh lambatnya respon sensor kelembaban terhadap nilai kelembaban.
2. Pengiriman pesan berupa SMS ke pemilik rumah jamur hanya terjadi jika kondisi di rumah berada di luar set point yang telah ditentukan.
3. Dengan adanya sistem ini, pemilik jamur tidak perlu datang langsung ke rumah jamurnya karena sistem ini sudah bekerja secara otomatis.
4. Pada sistem ini, MCU bekerja sebagai control otomatis yang bekerja berdasarkan perintah PC.
5. Untuk pengiriman data dari mikrokontroller ke PC tidak terpengaruh oleh jarak, dengan catatan masih di bawah batas maximum (1,2 Km).

5.2 Saran

Diharapkan alat ini dapat lebih diperluas aplikasi dan pengembangannya di masa yang akan datang, antara lain :

1. Pengontrolan seluruh sistem telekontrol rumah jamur dilakukan oleh MCU
2. Dalam perkembangan selanjutnya tidak hanya memberitahukan adanya kondisi yang terjadi, besar nilai suhu dan kelembaban, tetapi informasi yang dikirim ke HP pemilik berupa grafik suhu dan kelembaban berdasar waktu sehingga lebih praktis.
3. Sistem ini digunakan di rumah jamur yang sebenarnya.

DAFTAR PUSTAKA

- Richard Blocher, Dipl. phys, 2004, *Dasar Elektronika* : ANDI Yogyakarta.
- Wasito S, 2001.'Vademekum Elektronika ', Edisi kedua, PT Gramedia Pystaka Utama, Jakata.
- Dr. Ir. Meity Suradji Sinaga, M.Sc. 2000, 'Jamur Merang dan Budidayanya ', PT Penebar Swadaya , Jakarta.
- Widodo Budiharto, S. Si., Mkom, 2004, 'Interfacing Komputer dan Mikrokontroller ', PT Gramedia, Jakarta.
- Renesas R8C/TINY Data Sheet.
- Komunikasi Serial RS485 Data Sheet.

☆☆☆☆☆

LAMPIRAN

INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNIK INDUSTRI
JURUSAN TEKNIK ELEKTRO

FORMULIR BIMBINGAN SKRIPSI

Nama : BAMBANG WISWANTO

NIM : 0217104

Masa Bimbingan : 12 Januari 2007 – 12 Juli 2007

Judul Skripsi : PENGONTROL RUMAH JAMUR JARAK JAUH DI LENGKAPI DENGAN PERINGATAN SMS BERBASIS MIKROKONTROLLER RENESAS R8C/TINY

No.	Tanggal	Uraian	Paraf Pembimbing
1.	05-02-2007	Pendahuluan	
2.	05-02-2007	Landasan Teori	
3.	07-02-2007	Perencanaan dan pembuatan alat	
4.	08-02-2007	Pengukuran dan pengujian alat	
5.	09-02-2007	Revisi Perhitungan pengujian alat	
6.	10-02-2007	Demo alat	
7.	12-02-2007	Acc Seminar Hasil	
8.	15-3-2007	Acc kompre	
9.			
10.			

Malang,
Dosen Pembimbing

Joseph Dedy Irawan, ST, MT
NIP. 132315178

INSTITUT TEKNOLOGI NASIONAL
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO S-1
Jl. Karanglo KM. 2 Malang

FORMULIR PERBAIKAN SKRIPSI

Dalam pelaksanaan Ujian Skripsi Jenjang Strata Satu (S-1) Jurusan Teknik Elektro Konsentrasi Teknik Elektronika, maka perlu adanya perbaikan skripsi untuk mahasiswa :

Nama	: Bambang Wiswanto
NIM	: 02 17 104
Jurusan	: Teknik Elektro S-1
Konsentrasi	: Teknik Elektronika
Masa Bimbingan	: 12 Januari 2007 s/d 12 Juli 2007
Judul Skripsi	: Pengontrol Rumah Jamur Jarak Jauh Di Dengan Peringatan SMS Berbasis Mikrokontroller Renesas R8C/Tiny

Tanggal	Uraian	Paraf
23 Maret 2007	Blok Diagram?	
	Kesimpulan?	

Diperiksa / Disetujui :

PENGUJI

(M. Ashar, ST, MT)
NIP.

Mengetahui :

Dosen Pembimbing

(Joseph Dedy Irawan, ST, MT)
NIP. 132315178

INSTITUT TEKNOLOGI NASIONAL
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO

Formulir Perbaikan Ujian Skripsi

Dalam pelaksanaan Ujian Skripsi Janjang Strata 1 Jurusan Teknik Elektro Konsentasi T. Energi Listrik / T. Elektronika, maka perlu adanya perbaikan skripsi untuk mahasiswa :

NAMA : BAMBANG WISWANTO
NIM : 02.17.104
Perbaikan meliputi :

- Kesimpulan
- Blok Diagram

Malang,

( )

```
/************************************************************************/
/*                                                                      */
/*  FILE        :tes.c                                                  */
/*  DATE        :Thu, Peb 8, 2007                                       */
/*  DESCRIPTION :Main Program                                           */
/*  CPU TYPE    :Other                                                  */
/*                                                                      */
/*  This file is generated by Renesas Project Generator (Ver.4.0).      */
/*                                                                      */
/************************************************************************/

#include <stdio.h>
#include "sfr_r813.h"
#include "timer.h"
#include "lcdku.h"
#include "serial.h"
//#include "adda.h"
//#include "i2c.h"

unsigned char b,ai;
unsigned int i,a,s;

#define control         p1_3
#define blower  p1_0
#define pemanas         p1_1
#define lembab  p1_2
//adc suhu an6
//adc kelembapan an5

void initdir()
{
pd1=0xff;
pd1_5=0;
}

void initADC(void)                                      //rutin inisialisasi adc
{
adcon0 = 0x01;  //prescaler fAD/2 (invalid), A/D conv disabled, P0 group selected, oneshot

mode, pin AN1 selected
adcon1 = 0x38; //0x38;  //Vref  connected, freq select at fAD valid, 10 bit mode
adcon2 = 0x00;  //0;//without sample and hold
//adst = 1;             //start conversion
}
//dari rx ke tx harus ada delay
unsigned char rx485()
{
  control=0;
 while(ri_u0c1 == 0);
        ri_u0c1 =0;
        b=u0rb;
        u0c1 |= 1;
        return b;
}

void tx485(char kirim)
{
  control=1;
   u0tb=kirim;
        while(txept_u0c0 == 0);
        //txept_u1c0 = 0;
        u0rb = kirim;
        u0c1 |= 1;
        delay(100);
        control=0;
}
void kirims()
{

                a=0;
        for(ai=0;ai<20;ai++)
        {
        adcon0 &= 0xF8 ;
        adcon0 |= 0x06;
        asm("nop");
        adst = 1;
```

```
        while(adst) ;
        a=ad+a;
        }
        s=a/40+1;
        tx485(s);
}
void kiriml()
{
a=0;
        for(ai=0;ai<20;ai++)
        {
        adcon0 &= 0xF8 ;
        adcon0 |= 0x05;
        asm("nop");
        adst = 1;
        while(adst) ;
        a=ad+a;
        }
        s=a/80+1;
        tx485(s);
}
void kirimsk()
{

        a=0;
        for(ai=0;ai<20;ai++)
        {
        adcon0 &= 0xF8 ;
        adcon0 |= 0x06;
        asm("nop");
        adst = 1;
        while(adst) ;
        a=ad+a;
        }
        s=a/40;
        a=48+s/100;tx485(a);
        a=48+((s/10)%10);tx485(a);
        a=48+(s%10);tx485(a);
}
void kirimlk()
{
a=0;
        for(ai=0;ai<20;ai++)
        {
        adcon0 &= 0xF8 ;
        adcon0 |= 0x05;
        asm("nop");
        adst = 1;
        while(adst) ;
        a=ad+a;
        }
        s=a/80;
        a=48+s/100;tx485(a);
        a=48+((s/10)%10);tx485(a);
        a=48+(s%10);tx485(a);
}

#pragma INTERRUPT rx
void rx(void)
{
while(ri_u1c1 == 0);
        ri_u1c1 =0;
        u1c1 |= 1;

        }


void main(void)
{
/* Inisialisasi Variable*/
        //unsigned char data;
        /* Inisialisasi tampila MK */
        asm("FCLR I");  // Interrupt disable
        prcr = 1;               // Protect off
        cm13 = 1;               // X-in X-out = Clock External
        cm15 = 1;               // XCIN-XCOUT drivecapacity select bit : HIGH
```

```
    cm05 = 0;                   // X-in on
    cm16 = 0;                   // Main clock = No division mode
    cm17 = 0;
    cm06 = 0;                   // CM16 and CM17 enable
    asm("nop");
    asm("nop");
    asm("nop");
    asm("nop");
    ocd2 = 0;                   // Main clock change (x-tal)
    prcr = 0;                   // Protect on
    //s1ric=2;s0ric=0;txic=1;
    /* p1 sebagai keluaran dan p0 sebagai masukan */
    //pd1=0xff;
    initdir();
    pd3_7=0;
    asm("FSET I");
    /* Proses memasukkan dan mengeluarkan data */
    asm("FCLR I");
    //initlcd();
    //init_timer_X();
initSerial();
    initADC();
    Send_Text("ReNeSaS Test Serial");
    //kirim_serial('ReNeSaS Test Serial');


    while (1) ////////////////////////////////////////////////////
    {
    a=rx485();
    delay_ms(100);
    kirim_serial(a);
    //delay_ms(1000);
    if(a=='a')kirims();
    else if(a=='b')kiriml();
    else if(a=='c') pemanas=1;
    else if(a=='d') pemanas=0;
    else if(a=='e') lembab=1;
    else if(a=='f') lembab=0;
    else if(a=='g') blower=1;
    else if(a=='h') blower=0;
    else if(a=='i') kirimsk();
    else if(a=='j') kirimlk();
    }
```

```
unit Unit1;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, Text2pdu, StdCtrls, Buttons, CPort, ExtCtrls, ComCtrls, jpeg,
  DB, DBTables, Mask, DBCtrls, FMTBcd, DBXpress, SqlExpr, RpDefine, RpCon,
  RpConDS, DBClient, Provider, Grids, DBGrids, TeEngine, Series, TeeProcs,
  Chart, DbChart;

type
  TForm1 = class(TForm)
    Text2pdu1: TText2pdu;
    ComPort1: TComPort;
    BHP: TSpeedButton;
    StatusBar1: TStatusBar;
    BKirimSMS: TButton;
    BSuhu: TButton;
    BMulai: TBitBtn;
    Bhenti: TBitBtn;
    BLembab: TButton;
    DBChart1: TDBChart;
    Series1: TLineSeries;
    DBChart2: TDBChart;
    Series2: TLineSeries;
    CBSuhuAtas: TComboBox;
    CBSuhuBawah: TComboBox;
    CBLembabAtas: TComboBox;
    CBLembabBawah: TComboBox;
    DBGrid1: TDBGrid;
    Edit1: TEdit;
    Edit2: TEdit;
    Edit3: TEdit;
    Edit4: TEdit;
    Table1: TTable;
    DataSource1: TDataSource;
    Memo1: TMemo;
    ComPort2: TComPort;
    ETlp: TEdit;
    ESmsDikirim: TEdit;
    Button1: TButton;
    Button2: TButton;
    LKet: TLabel;
    Timer1: TTimer;
    BHapus: TButton;
    Timer2: TTimer;
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    Label4: TLabel;
    Label5: TLabel;
    Label6: TLabel;
    Label7: TLabel;
    Label8: TLabel;
```

```
    Label9: TLabel;
    Label10: TLabel;
    Button3: TButton;
    Memo2: TMemo;
    procedure BHPClick(Sender: TObject);
    procedure ComPort1RxChar(Sender: TObject; Count: Integer);
    procedure KirimSms(No:String ;Isi:String );
    procedure BKirimSMSClick(Sender: TObject);
    procedure BSuhuClick(Sender: TObject);
    procedure BLembabClick(Sender: TObject);
    procedure kirimData(data : String);
    procedure ComPort2RxChar(Sender: TObject; Count: Integer);
    procedure BMulaiClick(Sender: TObject);
    procedure Button1Click(Sender: TObject);
    procedure Button2Click(Sender: TObject);
    procedure Timer1Timer(Sender: TObject);
    procedure BhentiClick(Sender: TObject);
    procedure BHapusClick(Sender: TObject);
    procedure FormCreate(Sender: TObject);
    procedure FormClose(Sender: TObject; var Action: TCloseAction);
    procedure Timer2Timer(Sender: TObject);
    procedure Button3Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;
Const CRLF = #13+#10;
      CTRZ = #26;
      SMSCenter = '081100000';
var
  Form1: TForm1;
  pdusms: TPDU;
  dtsms,sms: tsms;
  jawab, jawab1, jawabMK, data, data_sms,sms_del,DataSerial,sms_dikirim: string;
  sdhbaca: boolean;
  pducmgc: string;
  z: integer;
implementation

//uses Database;

{$R *.dfm}

procedure TForm1.KirimSms(No:String ;Isi:String );
var Nonya, Isinya: string;
begin
  StatusBar1.SimpleText:='Mengirim SMS';
  Nonya:='0'+Copy(No,3,length(No)-1);
  Isinya:=Isi;
  pdusms:=Text2pdu1.texttosms('',Nonya,Isinya);
  ComPort1.WriteStr('AT+CMGC='+IntToStr(pdusms.Panjang)+CRLF);
  Sleep(500);
  pducmgc:='00'+Copy(pdusms.PDU,5,length(pdusms.PDU));
  ComPort1.WriteStr(pducmgc+CTRZ);
```

```
end;

procedure TForm1.kirimData(data : String);
begin
  ComPort2.FlowControl.ControlRTS:=rtsDisable;
  sleep(5);
  ComPort2.WriteStr(data);
  sleep(100);
  ComPort2.ClearBuffer(true,true);
  ComPort2.FlowControl.ControlRTS:=rtsEnable;
end;


procedure TForm1.BHPClick(Sender: TObject);
begin
ComPort1.ShowSetupDialog;
end;

procedure TForm1.ComPort1RxChar(Sender: TObject; Count: Integer);
var data:string;
begin
ComPort1.ReadStr(data,count);
Memo1.Text:=Memo1.Text+data;
end;

procedure TForm1.BKirimSMSClick(Sender: TObject);
begin
Memo1.Clear;
KirimSms(ETlp.Text,ESmsDikirim.Text);
end;

procedure TForm1.BSuhuClick(Sender: TObject);
begin
Memo1.Clear;
BMulai.Click;
LKet.Caption:='SUHU';
kirimData('a');
end;

procedure TForm1.BLembabClick(Sender: TObject);
begin
Memo1.Clear;
BMulai.Click;
LKet.Caption:='LEMBAB';
kirimData('b');
end;

procedure TForm1.ComPort2RxChar(Sender: TObject; Count: Integer);
var data_s,S:String;
 jml, data,no, suhu, lembab,faktor:integer;


begin
Timer1.Enabled:=false;
ComPort2.ReadStr(data_s,count);   // membaca data dari serial
```

```
Memo2.Text:=Memo2.Text+data_s;
S := Pchar(data_s);                    // membentuk data serial menjadi type PCHAR (CHAR)

  try                                  // fungsi yang berguna menangani masalah kesalahan pengolahan data atau yang lain
    Edit1.Text:=IntToStr(byte(S[1]));  // menampilkan data yang telah diubah
    Table1.Last;
    if Table1.FieldByName('no').AsString='' then no:=1
    else no:=Table1.FieldByName('no').AsInteger+1;  // ketiga diatas digunakan untuk mengambil nomor

    if LKet.Caption='SUHU' then  // jika data suhu maka melakukan proses di bawah
    begin
    Table1.Append;               // menambah baris baru dalam tabel
    Table1.FieldByName('no').AsInteger:=no;
    Table1.FieldByName('tgl').AsDateTime:=Date;
    Table1.FieldByName('jam').AsDateTime:=Time;
    suhu :=  byte(S[1]);     // round : fungsi pembulatan
    Table1.FieldByName('suhu').AsInteger:=suhu;
    Table1.Post;     // post : menyimpan

    end;

    if LKet.Caption='LEMBAB' then
    begin
    Table1.Edit;        // langsung mengisi baris yang sudah ditunjuk
    faktor:=130-(byte(S[1])-130);
    lembab :=  round((8*(faktor+59.625)/37)+40);
    //lembab := round((16*(faktor+59.625)/37)+8);
    suhu:=Table1.FieldByName('suhu').AsInteger;
    Edit2.Text:=IntToStr(suhu);
    Edit3.Text:=IntToStr(lembab);

    Table1.FieldByName('kelembaban').AsInteger:=lembab;
    Table1.Post;
      if (lembab < StrToInt(CBLembabBawah.Text)) and (suhu < StrToInt(CBSuhuBawah.Text))
then
        begin
        kirimData('c');
        kirimData('c');
        kirimData('c');
        kirimData('e');
        kirimData('h');
        ESmsDikirim.Text:='dingin kering';
        Application.ProcessMessages;
//    BKirimSMS.Click;
        end
      else if (lembab > StrToInt(CBLembabAtas.Text)) and (suhu < StrToInt(CBSuhuBawah.Text
)) then
        begin
        kirimData('c');
        kirimData('c');
        kirimData('c');
        kirimData('f');
        kirimData('g');
        ESmsDikirim.Text:='dingin lembab';
        Application.ProcessMessages;
```

```
//    BKirimSMS.Click;
        end
      else if (lembab < StrToInt(CBLembabBawah.Text)) and (suhu > StrToInt(CBSuhuAtas.Text
)) then
        begin
        kirimData('d');
        kirimData('d');
        kirimData('d');
        kirimData('e');
        kirimData('g');
        ESmsDikirim.Text:='panas kering';
        Application.ProcessMessages;
//    BKirimSMS.Click;
        end
      else if (lembab > StrToInt(CBLembabAtas.Text)) and (suhu > StrToInt(CBSuhuAtas.Text
)) then
        begin
        kirimData('c');
        kirimData('c');
        kirimData('c');
        kirimData('e');
        kirimData('h');
        ESmsDikirim.Text:='dingin kering';
        Application.ProcessMessages;
//    BKirimSMS.Click;
        end
      else if (lembab < StrToInt(CBLembabBawah.Text))then
        begin
        kirimData('d');
        kirimData('d');
        kirimData('d');
        kirimData('e');
        kirimData('h');
        ESmsDikirim.Text:='kering';
        Application.ProcessMessages;
//    BKirimSMS.Click;
        end
      else if (lembab > StrToInt(CBLembabAtas.Text))then
        begin
        kirimData('d');
        kirimData('d');
        kirimData('d');
        kirimData('f');
        kirimData('g');
        ESmsDikirim.Text:='lembab';
        Application.ProcessMessages;
//    BKirimSMS.Click;
        end
      else if (suhu > StrToInt(CBSuhuAtas.Text)) then
        begin
        kirimData('d');
        kirimData('d');
        kirimData('d');
//    sleep(500);
        kirimData('f');
```

```
        kirimData('g');
        ESmsDikirim.Text:='panas';
        Application.ProcessMessages;
/     BKirimSMS.Click;
        end
      else if (suhu < StrToInt(CBSuhuBawah.Text)) then
        begin
        kirimData('c');
        kirimData('c');
        kirimData('c');
/     sleep(500);
        kirimData('f');
        Application.ProcessMessages;
        kirimData('h');
/Button1.Click;
/     ShowMessage('dingin');
        ESmsDikirim.Text:='dingin';
        Application.ProcessMessages;
/     BKirimSMS.Click;
        end
      else
        begin
        kirimData('d');
        kirimData('d');
        kirimData('d');
        kirimData('f');
        kirimData('h');
        ESmsDikirim.Text:='normal';
        end;

    end;
  except
  end;
'imer1.Enabled:=true;
nd;

rocedure TForm1.BMulaiClick(Sender: TObject);
egin
/ComPort1.Open;
/ComPort2.Open;
'imer1.Enabled:=true;
'imer2.Interval:=StrToInt(Edit4.Text);
'imer2.Enabled:=true;
nd;

rocedure TForm1.Button1Click(Sender: TObject);
egin
        kirimData('c');
        kirimData('e');
        kirimData('g');
nd;

rocedure TForm1.Button2Click(Sender: TObject);
egin
        kirimData('d');
```

```
        kirimData('f');
        kirimData('h');
nd;

rocedure TForm1.Timer1Timer(Sender: TObject);
egin
f LKet.Caption='SUHU' then BLembab.Click
lse BSuhu.Click;
nd;

rocedure TForm1.BhentiClick(Sender: TObject);
egin
/ComPort1.Close;
/ComPort2.Close;
imer1.Enabled:=False;
imer2.Enabled:=False;
nd;

rocedure TForm1.BHapusClick(Sender: TObject);
egin
able1.First;
f (Table1.FieldByName('no').AsString<>'') or (Table1.FieldByName('suhu').AsString<>'') or
 (Table1.FieldByName('kelembaban').AsString<>'') then
  begin
    repeat
    Table1.Delete;
    until Table1.Eof;
  end
nd;

rocedure TForm1.FormCreate(Sender: TObject);
egin
omPort1.Open;
omPort2.Open;
nd;

rocedure TForm1.FormClose(Sender: TObject; var Action: TCloseAction);
egin
omPort1.Close;
omPort2.Close;
nd;

rocedure TForm1.Timer2Timer(Sender: TObject);
egin
f ESmsDikirim.Text<>'normal' then BKirimSMS.Click;
nd;

rocedure TForm1.Button3Click(Sender: TObject);
egin
omPort2.ShowSetupDialog;
nd;

nd.
```

09B0111-0110

Everywhere you imagine. RENESAS

6

R8C/13 Group

Hardware Manual

RENESAS 16-BIT SINGLE-CHIP MICROCOMPUTER

M16C FAMILY/R8C/Tiny SERIES

Rev. 1.10
Revision date: Apr 27, 2005

Renesas Technology
www.renesas.com

Keep safety first in your circuit designs!

1. Renesas Technology Corp. puts the maximum effort into making semiconductor products better and more reliable, but there is always the possibility that trouble may occur with them. Trouble with semiconductors may lead to personal injury, fire or property damage. Remember to give due consideration to safety when making your circuit designs, with appropriate measures such as (i) placement of substitutive, auxiliary circuits, (ii) use of non-flammable material or (iii) prevention against any malfunction or mishap.

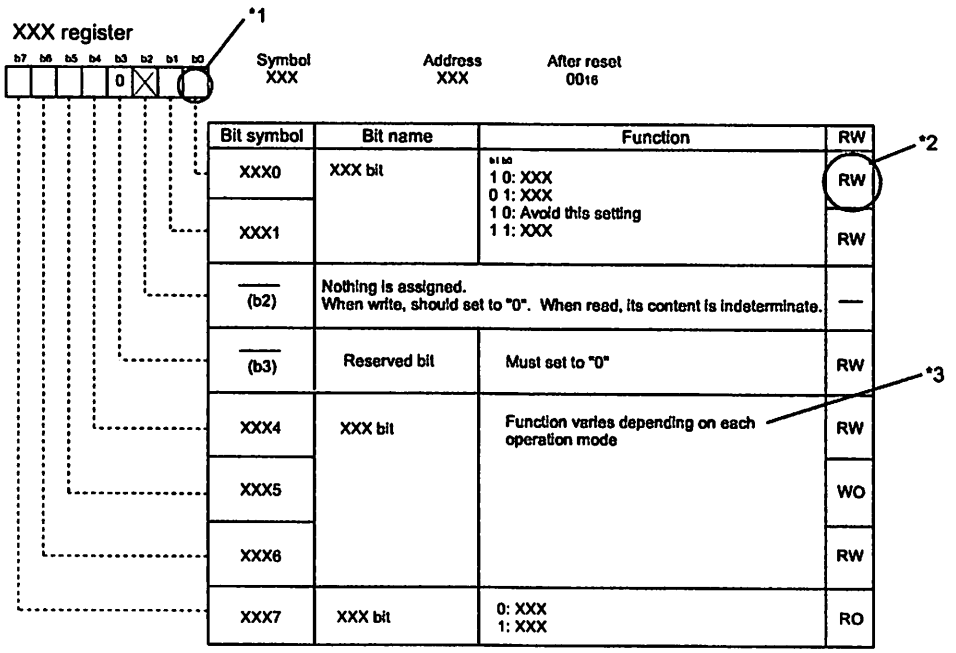
Notes regarding these materials

1. These materials are intended as a reference to assist our customers in the selection of the Renesas Technology Corp. product best suited to the customer's application; they do not convey any license under any intellectual property rights, or any other rights, belonging to Renesas Technology Corp. or a third party.
2. Renesas Technology Corp. assumes no responsibility for any damage, or infringement of any third-party's rights, originating in the use of any product data, diagrams, charts, programs, algorithms, or circuit application examples contained in these materials.
3. All information contained in these materials, including product data, diagrams, charts, programs and algorithms represents information on products at the time of publication of these materials, and are subject to change by Renesas Technology Corp. without notice due to product improvements or other reasons. It is therefore recommended that customers contact Renesas Technology Corp. or an authorized Renesas Technology Corp. product distributor for the latest product information before purchasing a product listed herein.
   The information described here may contain technical inaccuracies or typographical errors. Renesas Technology Corp. assumes no responsibility for any damage, liability, or other loss rising from these inaccuracies or errors.
   Please also pay attention to information published by Renesas Technology Corp. by various means, including the Renesas Technology Corp. Semiconductor home page (http://www.renesas.com).
4. When using any or all of the information contained in these materials, including product data, diagrams, charts, programs, and algorithms, please be sure to evaluate all information as a total system before making a final decision on the applicability of the information and products. Renesas Technology Corp. assumes no responsibility for any damage, liability or other loss resulting from the information contained herein.
5. Renesas Technology Corp. semiconductors are not designed or manufactured for use in a device or system that is used under circumstances in which human life is potentially at stake. Please contact Renesas Technology Corp. or an authorized Renesas Technology Corp. product distributor when considering the use of a product contained herein for any specific purposes, such as apparatus or systems for transportation, vehicular, medical, aerospace, nuclear, or undersea repeater use.
6. The prior written approval of Renesas Technology Corp. is necessary to reprint or reproduce in whole or in part these materials.
7. If these products or technologies are subject to the Japanese export control restrictions, they must be exported under a license from the Japanese government and cannot be imported into a country other than the approved destination.
   Any diversion or reexport contrary to the export control laws and regulations of Japan and/or the country of destination is prohibited.
8. Please contact Renesas Technology Corp. for further details on these materials or the products contained therein.

How to Use This Manual

This hardware manual provides detailed information on features in the R8C/13 Group microcomputer.
Users are expected to have basic knowledge of electric circuits, logical circuits and microcomputer.

Each register diagram contains bit functions with the following symbols and descriptions.



*1

Blank: Set to "0" or "1" according to your intended use
0: Set to "0"
1: Set to "1"
X: Nothing is assigned

*2

RW: Read and write
RO: Read only
WO: Write only
–: Nothing is assigned

*3

Terms to use here are explained as follows.

- Nothing is assigned
  Nothing is assigned to the bit concerned. When write, set to "0" for new function in future plan.
- Reserved bit
  Reserved bit. Set the specified value.
- Avoid this setting
  The operation at having selected is not guaranteed.
- Function varies depending on each operation mode
  Bit function varies depending on peripheral function mode.
  Refer to register diagrams in each mode.

M16C Family Documents

Document	Contents
Short Sheet	Hardware overview
Data Sheet	Hardware overview and electrical characteristics
Hardware Manual	Hardware specifications (pin assignments, memory maps, specifications of peripheral functions, electrical characteristics, timing charts)
Software Manual	Detailed description about instructions and microcomputer performance by each instruction
Application Note	• Application examples of peripheral functions • Sample programs • Introductory description about basic functions in M16C family • Programming method with the assembly and C languages

Table of Contents

SFR Page Reference

Address	Register	Symbol	Page
0000₁₆			
0001₁₆			
0002₁₆			
0003₁₆			
0004₁₆	Processor mode register 0	PM0	45
0005₁₆	Processor mode register 1	PM1	45
0006₁₆	System clock control register 0	CM0	31
0007₁₆	System clock control register 1	CM1	31
0008₁₆	High-speed on-chip oscillator control register 0	HR0	33
0009₁₆	Address match interrupt enable register	AIER	67
000A₁₆	Protect register	PRCR	44
000B₁₆	High-speed on-chip oscillator control register 1	HR1	33
000C₁₆	Oscillation stop detection register	OCD	32
000D₁₆	Watchdog timer reset register	WDTR	69
000E₁₆	Watchdog timer start register	WDTS	69
000F₁₆	Watchdog timer control register	WDC	69
0010₁₆ 0011₁₆ 0012₁₆	Address match interrupt register 0	RMAD0	67
0013₁₆			
0014₁₆ 0015₁₆ 0016₁₆	Address match interrupt register 1	RMAD1	67
0017₁₆			
0018₁₆			
0019₁₆	Voltage detection register 1	VCR1	22
001A₁₆	Voltage detection register 2	VCR2	22
001B₁₆			
001C₁₆			
001D₁₆			
001E₁₆	INT0 input filter select register	INT0F	60
001F₁₆	Voltage detection interrupt register	D4INT	23
0020₁₆			
0021₁₆			
0022₁₆			
0023₁₆			
0024₁₆			
0025₁₆			
0026₁₆			
0027₁₆			
0028₁₆			
0029₁₆			
002A₁₆			
002B₁₆			
002C₁₆			
002D₁₆			
002E₁₆			
002F₁₆			
0030₁₆			
0031₁₆			
0032₁₆			
0033₁₆			
0034₁₆			
0035₁₆			
0036₁₆			
0037₁₆			
0038₁₆			
0039₁₆			
003A₁₆			
003B₁₆			
003C₁₆			
003D₁₆			
003E₁₆			
003F₁₆			

Address	Register	Symbol	Page
0040₁₆			
0041₁₆			
0042₁₆			
0043₁₆			
0044₁₆			
0045₁₆			
0046₁₆			
0047₁₆			
0048₁₆			
0049₁₆			
004A₁₆			
004B₁₆			
004C₁₆			
004D₁₆	Key input interrupt control register	KUPIC	53
004E₁₆	AD conversion interrupt control register	ADIC	53
004F₁₆			
0050₁₆	Compare 1 interrupt control register	CMP1IC	53
0051₁₆	UART0 transmit interrupt control register	S0TIC	53
0052₁₆	UART0 receive interrupt control register	S0RIC	53
0053₁₆	UART1 transmit interrupt control register	S1TIC	53
0054₁₆	UART1 receive interrupt control register	S1RIC	53
0055₁₆	INT2 interrupt control register	INT2IC	53
0056₁₆	Timer X interrupt control register	TXIC	53
0057₁₆	Timer Y interrupt control register	TYIC	53
0058₁₆	Timer Z interrupt control register	TZIC	53
0059₁₆	INT1 interrupt control register	INT1IC	53
005A₁₆	INT3 interrupt control register	INT3IC	53
005B₁₆	Timer C interrupt control register	TCIC	53
005C₁₆	Compare 0 interrupt control register	CMP0IC	53
005D₁₆	INT0 interrupt control register	INT0IC	53
005E₁₆			
005F₁₆			
0060₁₆			
0061₁₆			
0062₁₆			
0063₁₆			
0064₁₆			
0065₁₆			
0066₁₆			
0067₁₆			
0068₁₆			
0069₁₆			
006A₁₆			
006B₁₆			
006C₁₆			
006D₁₆			
006E₁₆			
006F₁₆			
0070₁₆			
0071₁₆			
0072₁₆			
0073₁₆			
0074₁₆			
0075₁₆			
0076₁₆			
0077₁₆			
0078₁₆			
0079₁₆			
007A₁₆			
007B₁₆			
007C₁₆			
007D₁₆			
007E₁₆			
007F₁₆			

Blank columns are all reserved space. No use is allowed.

SFR Page Reference

Address	Register	Symbol	Page
0080₁₆	Timer Y, Z mode register	TYZMR	80/88
0081₁₆	Prescaler Y	PREY	81
0082₁₆	Timer Y secondary	TYSC	81
0083₁₆	Timer Y primary	TYPR	81
0084₁₆	Timer Y, Z waveform output control register	PUM	82/90
0085₁₆	Prescaler Z	PREZ	89
0086₁₆	Timer Z secondary	TZSC	89
0087₁₆	Timer Z primary	TZPR	89
0088₁₆			
0089₁₆			
008A₁₆	Timer Y, Z output control register	TYZOC	81/89
008B₁₆	Timer X mode register	TXMR	71
008C₁₆	Prescaler X	PREX	72
008D₁₆	Timer X register	TX	72
008E₁₆	Timer count source setting register	TCSS	72/82/90
008F₁₆			
0090₁₆ 0091₁₆	Timer C register	TC	103
0092₁₆			
0093₁₆			
0094₁₆			
0095₁₆			
0096₁₆	External input enable register	INTEN	60
0097₁₆			
0098₁₆	Key input enable register	KIEN	65
0099₁₆			
009A₁₆	Timer C control register 0	TCC0	103
009B₁₆	Timer C control register 1	TCC1	104
009C₁₆ 009D₁₆	Capture and compare 0 register	TM0	103
009E₁₆ 009F₁₆	Compare 1 register	TM1	103
00A0₁₆	UART0 transmit/receive mode register	U0MR	112
00A1₁₆	UART0 bit rate register	U0BRG	111
00A2₁₆ 00A3₁₆	UART0 transmit buffer register	U0TB	111
00A4₁₆	UART0 transmit/receive control register 0	U0C0	112
00A5₁₆	UART0 transmit/receive control register 1	U0C1	113
00A6₁₆ 00A7₁₆	UART0 receive buffer register	U0RB	111
00A8₁₆	UART1 transmit/receive mode register	U1MR	112
00A9₁₆	UART1 bit rate register	U1BRG	111
00AA₁₆ 00AB₁₆	UART1 transmit buffer register	U1TB	111
00AC₁₆	UART1 transmit/receive control register 0	U1C0	112
00AD₁₆	UART1 transmit/receive control register 1	U1C1	113
00AE₁₆ 00AF₁₆	UART1 receive buffer register	U1RB	111
00B0₁₆	UART transmit/receive control register 2	UCON	113
00B1₁₆			
00B2₁₆			
00B3₁₆			
00B4₁₆			
00B5₁₆			
00B6₁₆			
00B7₁₆			
00B8₁₆			
00B9₁₆			
00BA₁₆			
00BB₁₆			
00BC₁₆			
00BD₁₆			
00BE₁₆			
00BF₁₆			

Blank columns are all reserved space. No use is allowed.

Address	Register	Symbol	Page
00C0₁₆ 00C1₁₆	AD register	AD	127
00C2₁₆			
00C3₁₆			
00C4₁₆			
00C5₁₆			
00C6₁₆			
00C7₁₆			
00C8₁₆			
00C9₁₆			
00CA₁₆			
00CB₁₆			
00CC₁₆			
00CD₁₆			
00CE₁₆			
00CF₁₆			
00D0₁₆			
00D1₁₆			
00D2₁₆			
00D3₁₆			
00D4₁₆	AD control register 2	ADCON2	127
00D5₁₆			
00D6₁₆	AD control register 0	ADCON0	126
00D7₁₆	AD control register 1	ADCON1	126
00D8₁₆			
00D9₁₆			
00DA₁₆			
00DB₁₆			
00DC₁₆			
00DD₁₆			
00DE₁₆			
00DF₁₆			
00E0₁₆	Port P0 register	P0	141
00E1₁₆	Port P1 register	P1	141
00E2₁₆	Port P0 direction register	PD0	141
00E3₁₆	Port P1 direction register	PD1	141
00E4₁₆			
00E5₁₆	Port P3 register	P3	141
00E6₁₆			
00E7₁₆	Port P3 direction register	PD3	141
00E8₁₆	Port P4 register	P4	141
00E9₁₆			
00EA₁₆	Port P4 direction register	PD4	141
00EB₁₆			
00EC₁₆			
00ED₁₆			
00EE₁₆			
00EF₁₆			
00F0₁₆			
00F1₁₆			
00F2₁₆			
00F3₁₆			
00F4₁₆			
00F5₁₆			
00F6₁₆			
00F7₁₆			
00F8₁₆			
00F9₁₆			
03FA₁₆			
00FB₁₆			
00FC₁₆	Pull-up control register 0	PUR0	142
00FD₁₆	Pull-up control register 1	PUR1	142
00FE₁₆	Port P1 drive capacity control register	DRR	142
00FF₁₆	Timer C output control register	TCOUT	104
≈			≈
01B3₁₆	Flash memory control register 4	FMR4	169
01B4₁₆			
01B5₁₆	Flash memory control register 1	FMR1	169
01B6₁₆			
01B7₁₆	Flash memory control register 0	FMR0	168

0FFFF₁₆	Option function select register	OFS	69

R8C/13 Group

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER

1. Overview

This MCU is built using the high-performance silicon gate CMOS process using a R8C/Tiny Series CPU core and is packaged in a 32-pin plastic molded LQFP. This MCU operates using sophisticated instructions featuring a high level of instruction efficiency. With 1M bytes of address space, it is capable of executing instructions at high speed.
The data flash ROM (2 KB X 2 blocks) is embedded.

1.1 Applications

Electric household appliance, office equipment, housing equipment (sensor, security), general industrial equipment, audio, etc.

1.2 Performance Outline

Table 1.1. lists the performance outline of this MCU.

Table 1.1 Performance outline

Item		Performance
CPU	Number of basic instructions	89 instructions
	Shortest instruction execution time	50 ns (f(XIN) = 20 MHz, VCC = 3.0 to 5.5 V) 100 ns (f(XIN) = 10 MHz, VCC = 2.7 to 5.5 V)
	Operating mode	Single-chip
	Address space	1M bytes
	Memory capacity	See Table 1.2.
Peripheral function	Interrupt	Internal: 11 factors, External: 5 factors, Software: 4 factors, Priority level: 7 levels
	Watchdog timer	15 bits x 1 (with prescaler) Reset start function selectable
	Timer	Timer X: 8 bits x 1 channel, Timer Y: 8 bits x 1 channel, Timer Z: 8 bits x 1 channel (Each timer equipped with 8-bit prescaler) Timer C: 16 bits x 1 channel Circuits of input capture and output compare.
	Serial interface	•1 channel Clock synchronous, UART •1 channel UART
	A/D converter	10-bit A/D converter: 1 circuit, 12 channels
	Clock generation circuit	2 circuits •Main clock generation circuit (Equipped with a built-in feedback resistor) •On-chip oscillator (high-speed, low-speed) On high-speed on-chip oscillator the frequency adjustment function is usable.
	Oscillation stop detection function	Stop detection of main clock oscillation
	Voltage detection circuit	Included
	Power on reset circuit	Included
	Port	Input/Output: 22 (including LED drive port), Input: 2 (LED drive I/O port: 8)
Electrical characteristics	Power supply voltage	VCC = 3.0 to 5.5V (f(XIN) = 20MHz) VCC = 2.7 to 5.5V (f(XIN) = 10MHz)
	Power consumption	Typ.9 mA (VCC = 5.0V, (f(XIN) = 20MHz,High-speed mode) Typ.5 mA (VCC = 3.0V, (f(XIN) = 10MHz,High-speed mode) Typ.35 µA (VCC = 3.0V, Wait mode, Peripheral clock stops) Typ.0.7 µA (VCC = 3.0V, Stop mode)
Flash memory	Program/erase voltage	VCC = 2.7 to 5.5 V
	Number of program/erase	10,000 times (Data area)
		1,000 times (Program area)
Operating ambient temperature		-20 to 85°C -40 to 85°C (D-version)
Package		32-pin plastic mold LQFP

1.3 Block Diagram

Figure 1.1 shows this MCU block diagram.

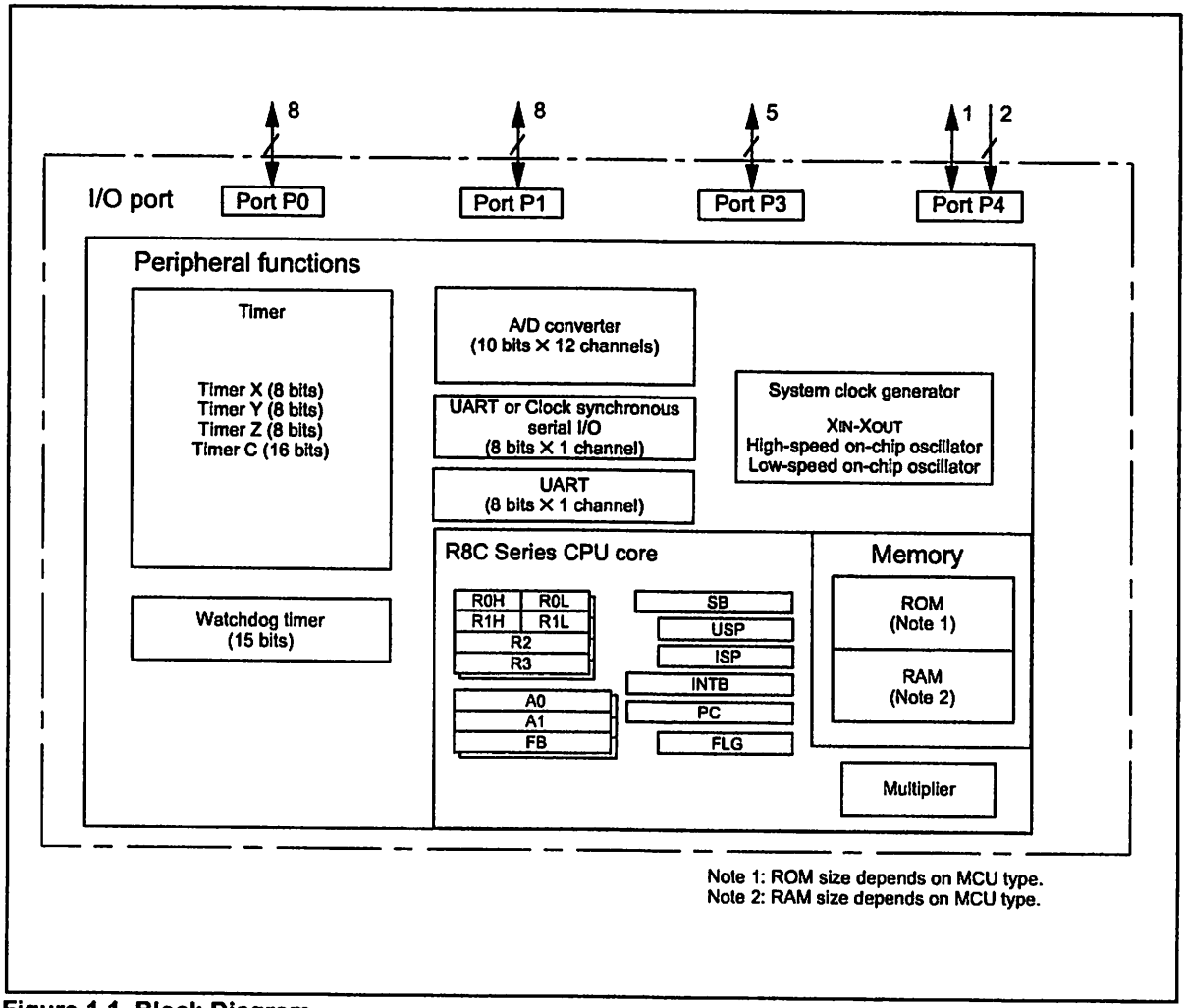


Figure 1.1 Block Diagram

1.4 Product Information

Table 1.2 lists the products.

Table 1.2 Product List

As of April 2005

Type No.	ROM capacity		RAM capacity	Package type	Remarks
	Program area	Data area			
R5F21132FP	8K bytes	2K bytes x 2	512 bytes	PLQP0032GB-A	Flash memory version
R5F21133FP	12K bytes	2K bytes x 2	768 bytes	PLQP0032GB-A	
R5F21134FP	16K bytes	2K bytes x 2	1K bytes	PLQP0032GB-A	
R5F21132DFP	8K bytes	2K bytes x 2	512 bytes	PLQP0032GB-A	D version
R5F21133DFP	12K bytes	2K bytes x 2	768 bytes	PLQP0032GB-A	
R5F21134DFP	16K bytes	2K bytes x 2	1K bytes	PLQP0032GB-A	

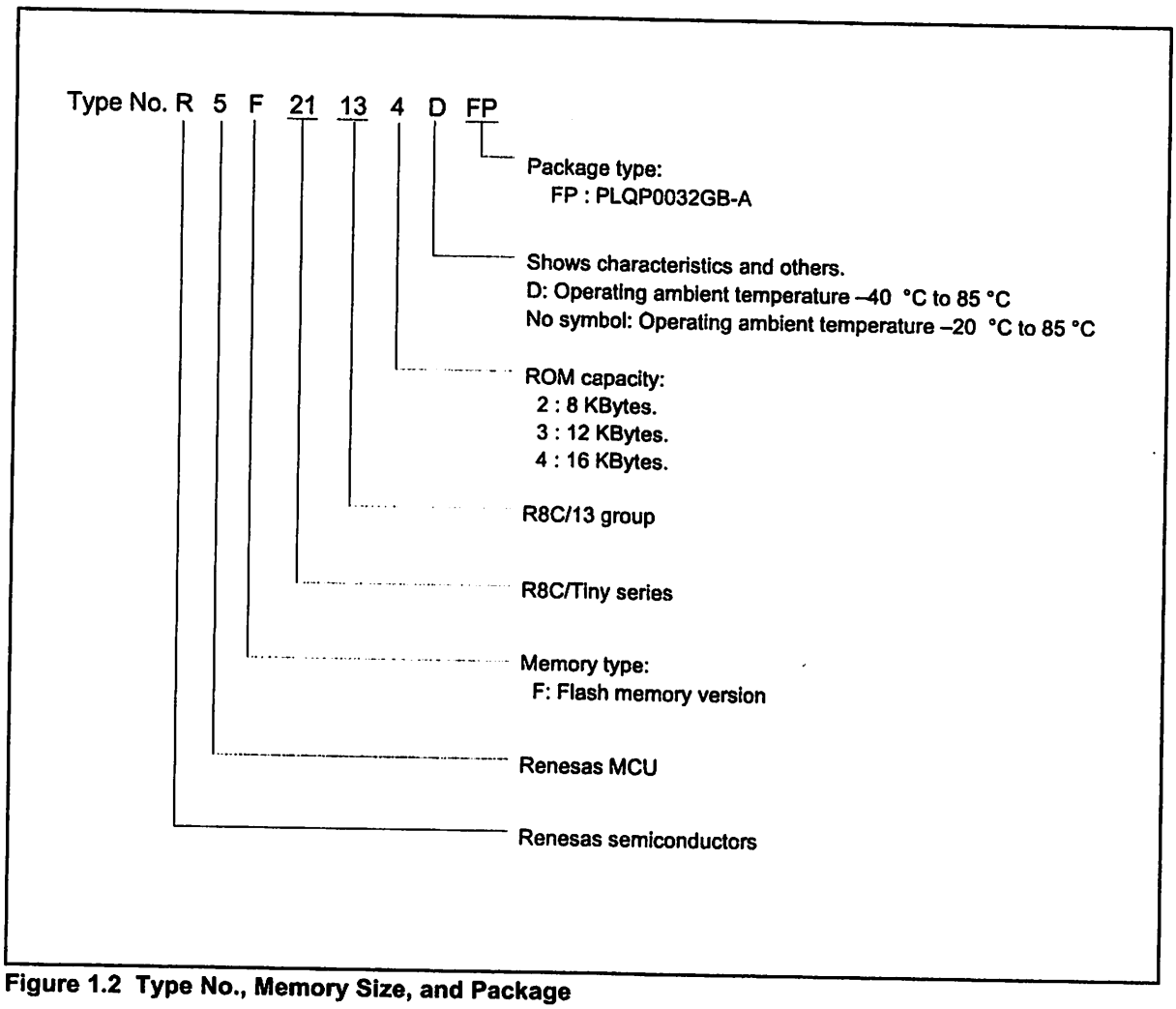


Figure 1.2 Type No., Memory Size, and Package

1.5 Pin Assignments

Figure 1.3 shows the pin configuration (top view).

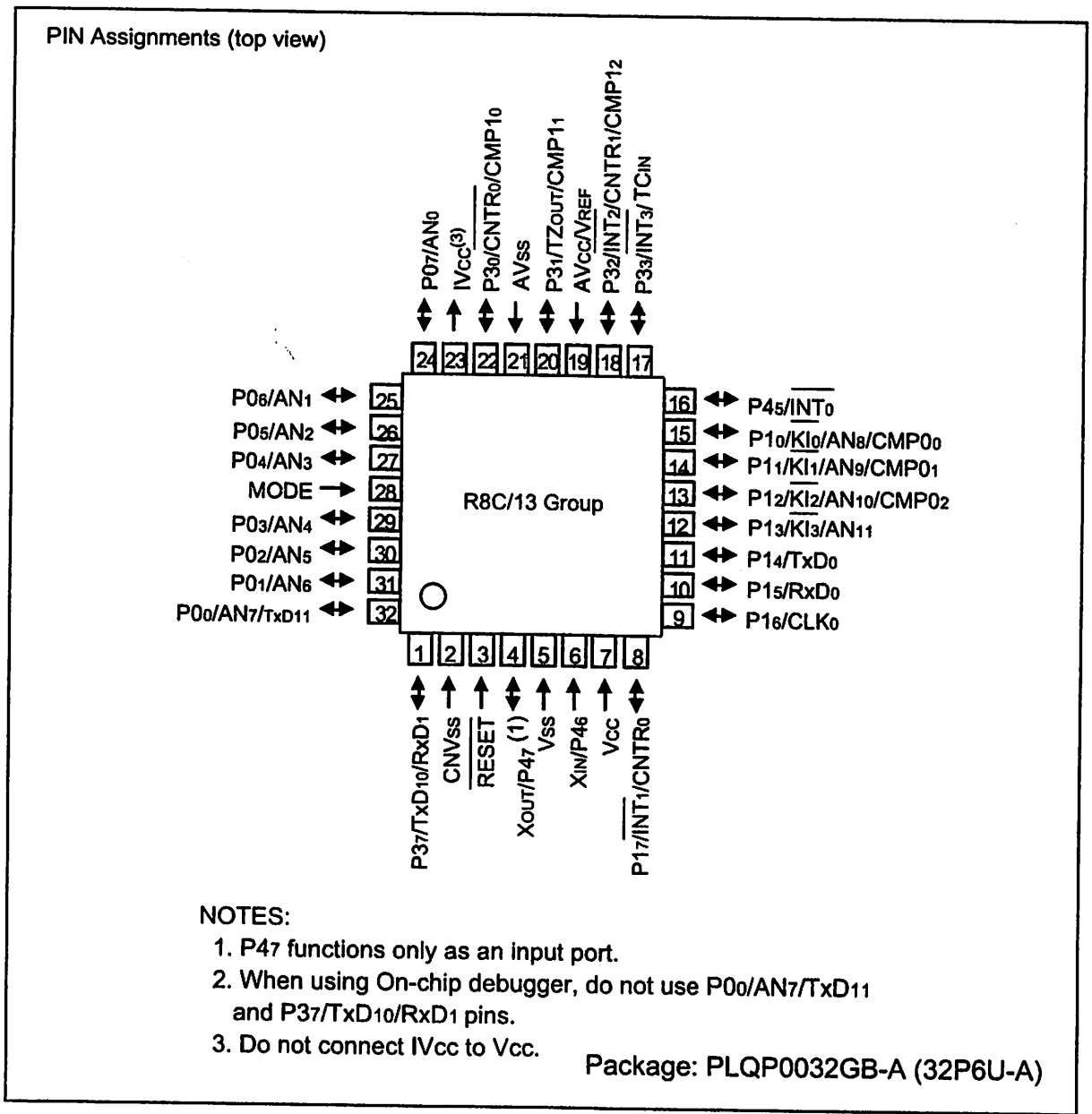


Figure 1.3 Pin Assignments (Top View)

1.6 Pin Description

Table 1.3 shows the pin description

Table 1.3 Pin description

Signal name	Pin name	I/O type	Function
Power supply input	Vcc, Vss	I	Apply 2.7 V to 5.5 V to the Vcc pin. Apply 0 V to the Vss pin.
IVcc	IVcc	O	This pin is to stabilize internal power supply Connect this pin to Vss via a capacitor (0.1 µF) Do not connect to Vcc
Analog power supply input	AVcc, AVss	I	These are power supply input pins for A/D converter. Connect the AVcc pin to Vcc. Connect the AVss pin to Vss. Connect a capacitor between pins AVcc and AVss.
Reset input	$\overline{\text{RESET}}$	I	"L" on this input resets the MCU.
CNVss	CNVss	I	Connect this pin to Vss via a resistor(1)
MODE	MODE	I	Connect this pin to Vcc via a resistor
Main clock input	XIN	I	These pins are provided for the main clock generating circuit I/O. Connect a ceramic resonator or a crystal oscillator between the XIN and XOUT pins. To use an externally derived clock, input it to the XIN pin and leave the XOUT pin open.
Main clock output	XOUT	O	
$\overline{\text{INT}}$ interrupt input	$\overline{\text{INT0}}$ to $\overline{\text{INT3}}$	I	These are $\overline{\text{INT}}$ interrupt input pins.
Key input interrupt input	$\overline{\text{KI0}}$ to $\overline{\text{KI3}}$	I	These are key input interrupt pins.
Timer X	CNTR0	I/O	This is the timer X I/O pin.
	$\overline{\text{CNTR0}}$	O	This is the timer X output pin.
Timer Y	CNTR1	I/O	This is the timer Y I/O pin.
Timer Z	TZOUT	O	This is the timer Z output pin.
Timer C	TCIN	I	This is the timer C input pin.
	CMP00 to CMP03, CMP10 to CMP13	O	These are the timer C output pins.
Serial interface	CLK0	I/O	This is a transfer clock I/O pin.
	RxD0, RxD1	I	These are serial data input pins.
	TxD0, TxD10, TxD11	O	These are serial data output pins.
Reference voltage input	VREF	I	This is a reference voltage input pin for A/D converter. Connect the VREF pin to Vcc.
A/D converter	AN0 to AN11	I	These are analog input pins for A/D converter.
I/O port	P00 to P07, P10 to P17, P30 to P33, P37, P45	I/O	These are 8-bit CMOS I/O ports. Each port has an I/O select direction register, allowing each pin in that port to be directed for input or output individually. Any port set to input can select whether to use a pull-up resistor or not by program. P10 to P17 also function as LED drive ports.
Input port	P46, P47	I	These are input only pins.

NOTES :

1. Refer to "19.8 Noise" for the connecting reference resistor value.

2. Central Processing Unit (CPU)

Figure 2.1 shows the CPU registers. The CPU has 13 registers. Of these, R0, R1, R2, R3, A0, A1 and FB comprise a register bank. There are two register banks.

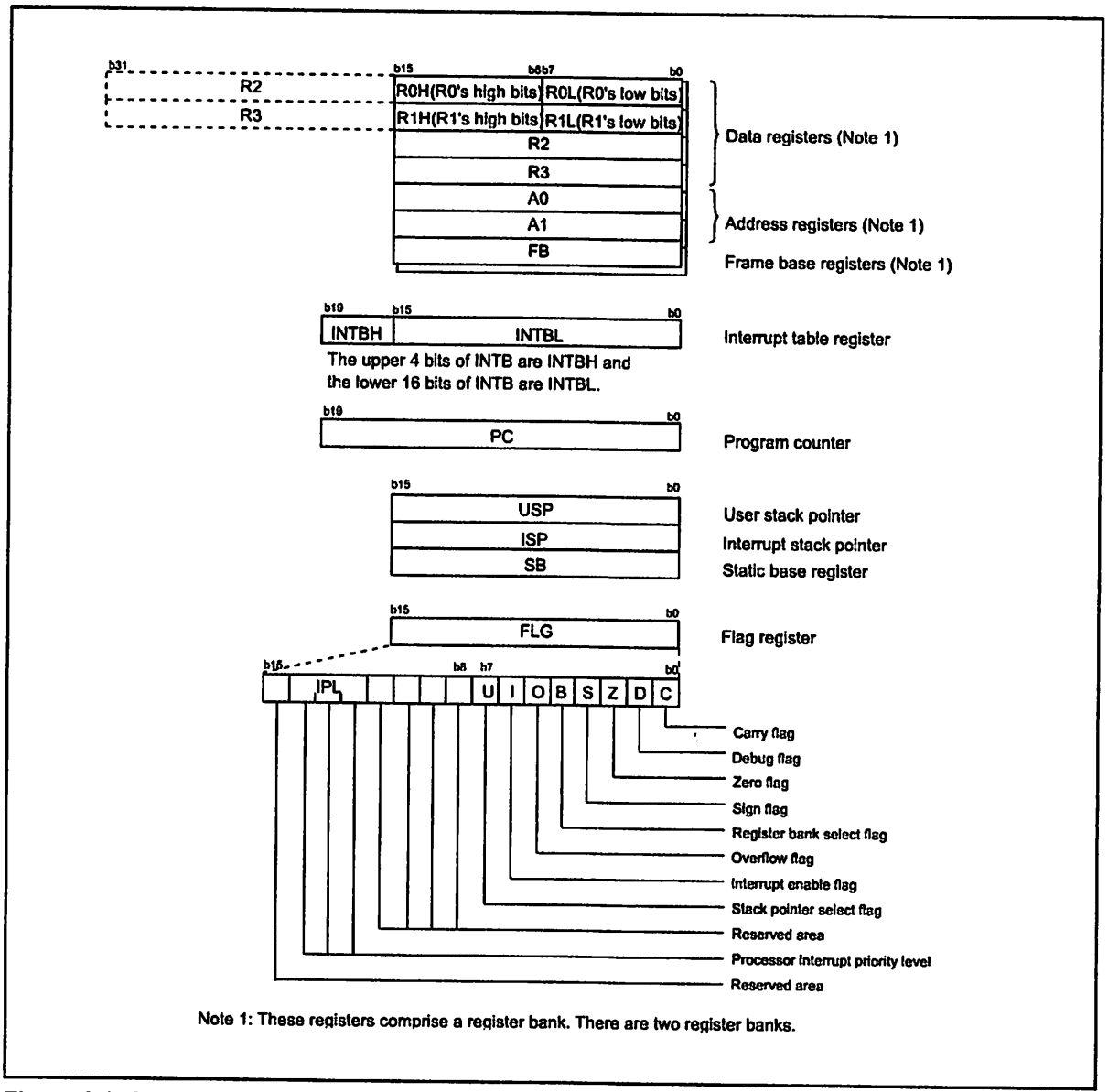


Figure 2.1 Central Processing Unit Register

2.1 Data Registers (R0, R1, R2 and R3)

The R0 register consists of 16 bits, and is used mainly for transfers and arithmetic/logic operations. R1 to R3 are the same as R0.

The R0 register can be separated between high (R0H) and low (R0L) for use as two 8-bit data registers. R1H and R1L are the same as R0H and R0L. Conversely, R2 and R0 can be combined for use as a 32-bit data register (R2R0). R3R1 is the same as R2R0.

2.2 Address Registers (A0 and A1)

The register A0 consists of 16 bits, and is used for address register indirect addressing and address register relative addressing. They also are used for transfers and logic/logic operations. A1 is the same as A0.

In some instructions, registers A1 and A0 can be combined for use as a 32-bit address register (A1A0).

2.3 Frame Base Register (FB)

FB is configured with 16 bits, and is used for FB relative addressing.

2.4 Interrupt Table Register (INTB)

INTB is configured with 20 bits, indicating the start address of an interrupt vector table.

2.5 Program Counter (PC)

PC is configured with 20 bits, indicating the address of an instruction to be executed.

2.6 User Stack Pointer (USP) and Interrupt Stack Pointer (ISP)

Stack pointer (SP) comes in two types: USP and ISP, each configured with 16 bits.

Your desired type of stack pointer (USP or ISP) can be selected by the U flag of FLG.

2.7 Static Base Register (SB)

SB is configured with 16 bits, and is used for SB relative addressing.

2.8 Flag Register (FLG)

FLG consists of 11 bits, indicating the CPU status.

2.8.1 Carry Flag (C Flag)

This flag retains a carry, borrow, or shift-out bit that has occurred in the arithmetic/logic unit.

2.8.2 Debug Flag (D Flag)

The D flag is used exclusively for debugging purpose. During normal use, it must be set to "0".

2.8.3 Zero Flag (Z Flag)

This flag is set to "1" when an arithmetic operation resulted in 0; otherwise, it is "0".

2.8.4 Sign Flag (S Flag)

This flag is set to "1" when an arithmetic operation resulted in a negative value; otherwise, it is "0".

2.8.5 Register Bank Select Flag (B Flag)

Register bank 0 is selected when this flag is "0" ; register bank 1 is selected when this flag is "1".

2.8.6 Overflow Flag (O Flag)

This flag is set to "1" when the operation resulted in an overflow; otherwise, it is "0".

2.8.7 Interrupt Enable Flag (I Flag)

This flag enables a maskable interrupt.

Maskable interrupts are disabled when the I flag is "0", and are enabled when the I flag is "1". The I flag is cleared to "0" when the interrupt request is accepted.

2.8.8 Stack Pointer Select Flag (U Flag)

ISP is selected when the U flag is "0"; USP is selected when the U flag is "1".

The U flag is cleared to "0" when a hardware interrupt request is accepted or an INT instruction for software interrupt Nos. 0 to 31 is executed.

2.8.9 Processor Interrupt Priority Level (IPL)

IPL is configured with three bits, for specification of up to eight processor interrupt priority levels from level 0 to level 7.

If a requested interrupt has priority greater than IPL, the interrupt is enabled.

2.8.10 Reserved Area

When write to this bit, write "0". When read, its content is indeterminate.

3. Memory

Figure 3.1 is a memory map of this MCU. The address space extends the 1M bytes from address 0000016 to FFFFF16.

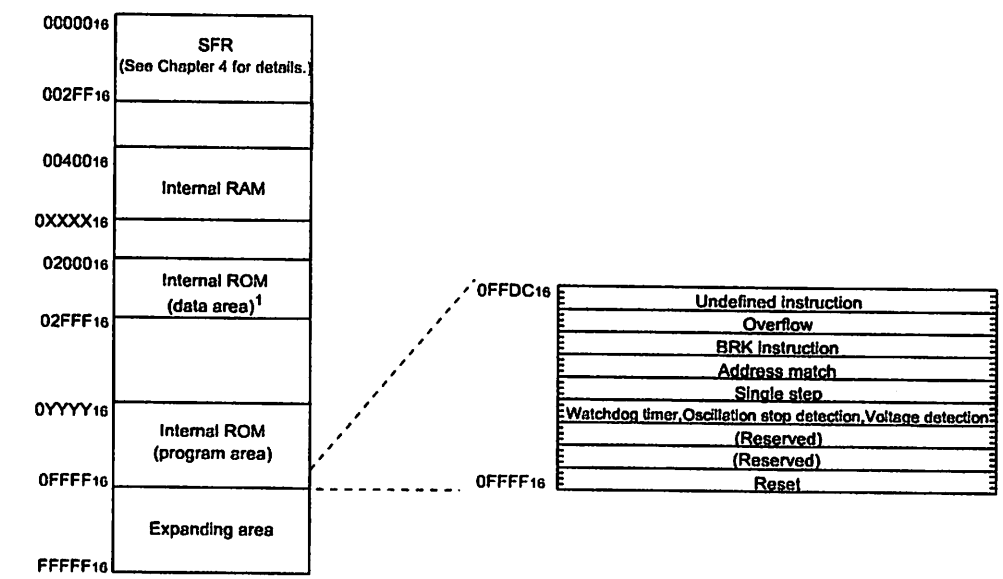
The internal ROM (program area) is allocated in a lower address direction beginning with address 0FFFF16. For example, a 16-Kbyte internal ROM is allocated to the addresses from 0C00016 to 0FFFF16.

The fixed interrupt vector table is allocated to the addresses from 0FFDC16 to 0FFFF16. Therefore, store the start address of each interrupt routine here.

The internal ROM (data area) is allocated to the addresses from 0200016 to 02FFF16.

The internal RAM is allocated in an upper address direction beginning with address 0040016. For example, a 1-Kbyte internal RAM is allocated to the addresses from 0040016 to 007FF16. In addition to storing data, the internal RAM also stores the stack used when calling subroutines and when interrupts are generated.

Special function registers (SFR) are allocated to the addresses from 0000016 to 002FF16. Peripheral function control registers are located here. Of the SFR, any space which has no functions allocated is reserved for future use and cannot be used by users.



NOTES:
1. The data flash ROM block A (2K bytes) and block B (2K bytes) are shown.
2. Blank spaces are reserved. No access is allowed.

Type name	Internal ROM		Internal RAM	
	Size	Address 0YYYY16	Size	Address 0XXXX16
R5F21134FP, R5F21134DFP	16K bytes	0C00016	1K bytes	007FF16
R5F21133FP, R5F21133DFP	12K bytes	0D00016	768 bytes	006FF16
R5F21132FP, R5F21132DFP	8K bytes	0E00016	512 bytes	005FF16

Figure 3.1 Memory Map

4. Special Function Register (SFR)

SFR(Special Function Register) is the control register of peripheral functions. Tables 4.1 to 4.4 list the SFR information

Table 4.1 SFR Information(1)[1]

Address	Register	Symbol	After reset
0000₁₆			
0001₁₆			
0002₁₆			
0003₁₆			
0004₁₆	Processor mode register 0 [1]	PM0	00₁₆
0005₁₆	Processor mode register 1	PM1	00₁₆
0006₁₆	System clock control register 0	CM0	01101000₂
0007₁₆	System clock control register 1	CM1	00100000₂
0008₁₆	High-speed on-chip oscillator control register 0	HR0	00₁₆
0009₁₆	Address match interrupt enable register	AIER	XXXXXX00₂
000A₁₆	Protect register	PRCR	00XXX000₂
000B₁₆	High-speed on-chip oscillator control register 1	HR1	40₁₆
000C₁₆	Oscillation stop detection register	OCD	00000100₂
000D₁₆	Watchdog timer reset register	WDTR	XX₁₆
000E₁₆	Watchdog timer start register	WDTS	XX₁₆
000F₁₆	Watchdog timer control register	WDC	00011111₂
0010₁₆	Address match interrupt register 0	RMAD0	00₁₆
0011₁₆			00₁₆
0012₁₆			X0₁₆
0013₁₆			
0014₁₆	Address match interrupt register 1	RMAD1	00₁₆
0015₁₆			00₁₆
0016₁₆			X0₁₆
0017₁₆			
0018₁₆			
0019₁₆	Voltage detection register 1 [2]	VCR1	00001000₂
001A₁₆	Voltage detection register 2 [2]	VCR2	00₁₆ [3] 10000000₂ [4]
001B₁₆			
001C₁₆			
001D₁₆			
001E₁₆	INT0 input filter select register	INT0F	XXXXX000₂
001F₁₆	Voltage detection interrupt register [2]	D4INT	00₁₆ [3] 01000001₂ [4]
0020₁₆			
0021₁₆			
0022₁₆			
0023₁₆			
0024₁₆			
0025₁₆			
0026₁₆			
0027₁₆			
0028₁₆			
0029₁₆			
002A₁₆			
002B₁₆			
002C₁₆			
002D₁₆			
002E₁₆			
002F₁₆			
0030₁₆			
0031₁₆			
0032₁₆			
0033₁₆			
0034₁₆			
0035₁₆			
0036₁₆			
0037₁₆			
0038₁₆			
0039₁₆			
003A₁₆			
003B₁₆			
003C₁₆			
003D₁₆			
003E₁₆			
003F₁₆			

X : Undefined

NOTES:
1. Blank columns are all reserved space. No access is allowed.
2. Software reset or the watchdog timer reset does not affect this register.
3. Owing to Reset input.
4. In the case of RESET pin = H retaining.

Table 4.2 SFR Information(2)(1)

Address	Register	Symbol	After reset
0040₁₆			
0041₁₆			
0042₁₆			
0043₁₆			
0044₁₆			
0045₁₆			
0046₁₆			
0047₁₆			
0048₁₆			
0049₁₆			
004A₁₆			
004B₁₆			
004C₁₆			
004D₁₆	Key input interrupt control register	KUPIC	XXXXX0002
004E₁₆	AD conversion interrupt control register	ADIC	XXXXX0002
004F₁₆			
0050₁₆	Compare 1 interrupt control register	CMP1IC	XXXXX0002
0051₁₆	UART0 transmit interrupt control register	S0TIC	XXXXX0002
0052₁₆	UART0 receive interrupt control register	S0RIC	XXXXX0002
0053₁₆	UART1 transmit interrupt control register	S1TIC	XXXXX0002
0054₁₆	UART1 receive interrupt control register	S1RIC	XXXXX0002
0055₁₆	INT2 interrupt control register	INT2IC	XXXXX0002
0056₁₆	Timer X interrupt control register	TXIC	XXXXX0002
0057₁₆	Timer Y interrupt control register	TYIC	XXXXX0002
0058₁₆	Timer Z interrupt control register	TZIC	XXXXX0002
0059₁₆	INT1 interrupt control register	INT1IC	XXXXX0002
005A₁₆	INT3 interrupt control register	INT3IC	XXXXX0002
005B₁₆	Timer C interrupt control register	TCIC	XXXXX0002
005C₁₆	Compare 0 interrupt control register	CMP0IC	XXXXX0002
005D₁₆	INT0 interrupt control register	INT0IC	XX00X0002
005E₁₆			
005F₁₆			
0060₁₆			
0061₁₆			
0062₁₆			
0063₁₆			
0064₁₆			
0065₁₆			
0066₁₆			
0067₁₆			
0068₁₆			
0069₁₆			
006A₁₆			
006B₁₆			
006C₁₆			
006D₁₆			
006E₁₆			
006F₁₆			
0070₁₆			
0071₁₆			
0072₁₆			
0073₁₆			
0074₁₆			
0075₁₆			
0076₁₆			
0077₁₆			
0078₁₆			
0079₁₆			
007A₁₆			
007B₁₆			
007C₁₆			
007D₁₆			
007E₁₆			
007F₁₆			

X : Undefined

NOTES:

1. Blank columns are all reserved space. No access is allowed.

Table 4.3 SFR Information(3)(1)

Address	Register	Symbol	After reset
008016	Timer Y, Z mode register	TYZMR	0016
008116	Prescaler Y	PREY	FF16
008216	Timer Y secondary	TYSC	FF16
008316	Timer Y primary	TYPR	FF16
008416	Timer Y, Z waveform output control register	PUM	0016
008516	Prescaler Z	PREZ	FF16
008616	Timer Z secondary	TZSC	FF16
008716	Timer Z primary	TZPR	FF16
008816			
008916			
008A16	Timer Y, Z output control register	TYZOC	0016
008B16	Timer X mode register	TXMR	0016
008C16	Prescaler X	PREX	FF16
008D16	Timer X register	TX	FF16
008E16	Count source set register	TCSS	0016
008F16			
009016 009116	Timer C register	TC	0016 0016
009216			
009316			
009416			
009516			
009616	External input enable register	INTEN	0016
009716			
009816	Key input enable register	KIEN	0016
009916			
009A16	Timer C control register 0	TCC0	0016
009B16	Timer C control register 1	TCC1	0016
009C16 009D16	Capture, compare 0 register	TM0	0016 00162
009E16 009F16	Compare 1 register	TM1	FF16 FF16
00A016	UART0 transmit/receive mode register	U0MR	0016
00A116	UART0 bit rate register	U0BRG	XX16
00A216 00A316	UART0 transmit buffer register	U0TB	XX16 XX16
00A416	UART0 transmit/receive control register 0	U0C0	000010002
00A516	UART0 transmit/receive control register 1	U0C1	000000102
00A616 00A716	UART0 receive buffer register	U0RB	XX16 XX16
00A816	UART1 transmit/receive mode register	U1MR	0016
00A916	UART1 bit rate register	U1BRG	XX16
00AA16 00AB16	UART1 transmit buffer register	U1TB	XX16 XX16
00AC16	UART1 transmit/receive control register 0	U1C0	000010002
00AD16	UART1 transmit/receive control register 1	U1C1	000000102
00AE16 00AF16	UART1 receive buffer register	U1RB	XX16 XX16
00B016	UART transmit/receive control register 2	UCON	0016
00B116			
00B216			
00B316			
00B416			
00B516			
00B616			
00B716			
00B816			
00B916			
00BA16			
00BB16			
00BC16			
00BD16			
00BE16			
00BF16			

X : Undefined

NOTES:

1. Blank columns are all reserved space. No access is allowed.
2. When the output compare mode is selected (the TCC13 bit in the TCC1 register = 1), the value is set to FFFF16.

Table 4.4 SFR Information(4)[1]

Address	Register	Symbol	After reset
00C0₁₆	AD register	AD	XX16
00C1₁₆			XX16
00C2₁₆			
00C3₁₆			
00C4₁₆			
00C5₁₆			
00C6₁₆			
00C7₁₆			
00C8₁₆			
00C9₁₆			
00CA₁₆			
00CB₁₆			
00CC₁₆			
00CD₁₆			
00CE₁₆			
00CF₁₆			
00D0₁₆			
00D1₁₆			
00D2₁₆			
00D3₁₆			
00D4₁₆	AD control register 2	ADCON2	0016
00D5₁₆			
00D6₁₆	AD control register 0	ADCON0	00000XXX2
00D7₁₆	AD control register 1	ADCON1	0016
00D8₁₆			
00D9₁₆			
00DA₁₆			
00DB₁₆			
00DC₁₆			
00DD₁₆			
00DE₁₆			
00DF₁₆			
00E0₁₆	Port P0 register	P0	XX16
00E1₁₆	Port P1 register	P1	XX16
00E2₁₆	Port P0 direction register	PD0	0016
00E3₁₆	Port P1 direction register	PD1	0016
00E4₁₆			
00E5₁₆	Port P3 register	P3	XX16
00E6₁₆			
00E7₁₆	Port P3 direction register	PD3	0016
00E8₁₆	Port P4 register	P4	XX16
00E9₁₆			
00EA₁₆	Port P4 direction register	PD4	0016
00EB₁₆			
00EC₁₆			
00ED₁₆			
00EE₁₆			
00EF₁₆			
00F0₁₆			
00F1₁₆			
00F2₁₆			
00F3₁₆			
00F4₁₆			
00F5₁₆			
00F6₁₆			
00F7₁₆			
00F8₁₆			
00F9₁₆			
03FA₁₆			
00FB₁₆			
00FC₁₆	Pull-up control register 0	PUR0	00XX00002
00FD₁₆	Pull-up control register 1	PUR1	XXXXXX0X2
00FE₁₆	Port P1 drive capacity control register	DRR	0016
00FF₁₆	Timer C output control register	TCOUT	0016
≈			≈
01B3₁₆	Flash memory control register 4	FMR4	010000002
01B4₁₆			
01B5₁₆	Flash memory control register 1	FMR1	1000000X2
01B6₁₆			
01B7₁₆	Flash memory control register 0	FMR0	000000012
0FFF₁₆	Option function select register [2]	OFS	Note 2

X : Undefined

NOTES:

1. The blank areas, 0100₁₆ to 01B2₁₆ and 01B8₁₆ to 02FF₁₆ are reserved and cannot be used by users.
2. The watchdog timer control bit is assigned. Refer to "Figure11.2 OFS, WDC, WDTR and WDTS registers" of Hardware Manual for details

5. Reset

There are three types of resets: a hardware reset, a software reset, and an watchdog timer reset.

5.1 Hardware Reset

There are three kinds of hardware reset: hardware reset 1, hardware reset 2, and power-on reset. After reset, the low-speed on-chip oscillator clock divided by 8 is automatically selected for the CPU.

5.1.1 Hardware Reset 1

A reset is applied using the $\overline{\text{RESET}}$ pin. When an "L" signal is applied to the $\overline{\text{RESET}}$ pin while the power supply voltage is within the recommended operating condition, the pins are initialized (see Table 5.1 "Pin Status When $\overline{\text{RESET}}$ Pin Level is 'L'"). When the input level at the $\overline{\text{RESET}}$ pin is released from "L" to "H", the CPU and SFR are initialized, and the program is executed starting from the address indicated by the reset vector. Figure 5.1 shows the CPU register status after reset and figure 5.2 shows the reset sequence. The internal RAM is not initialized. If the $\overline{\text{RESET}}$ pin is pulled "L" while writing to the internal RAM, the internal RAM becomes indeterminate.
Figures 5.3 to 5.4 show the reset circuit example using the hardware reset 1. Refer to Chapter 4, "Special Function Register (SFR)" for the status of SFR after reset.

- When the power supply is stable

(1) Apply an "L" signal to the $\overline{\text{RESET}}$ pin.
(2) Wait for 500 µs ($1/f_{RING-S} \times 20$).
(3) Apply an "H" signal to the $\overline{\text{RESET}}$ pin.

- Power on

(1) Apply an "L" signal to the $\overline{\text{RESET}}$ pin.
(2) Let the power supply voltage increase until it meets the recommended operating condition.
(3) Wait td(P-R) or more until the internal power supply stabilizes.
(4) Wait for 500 µs ($1/f_{RING-S} \times 20$).
(5) Apply an "H" signal to the $\overline{\text{RESET}}$ pin.

Table 5.1 Pin Status When $\overline{\text{RESET}}$ Pin Level is "L"

Pin name	Pin status
P0	Input port
P1	Input port
$P3_0$ to $P3_3$, $P3_7$	Input port
$P4_5$ to $P4_7$	Input port

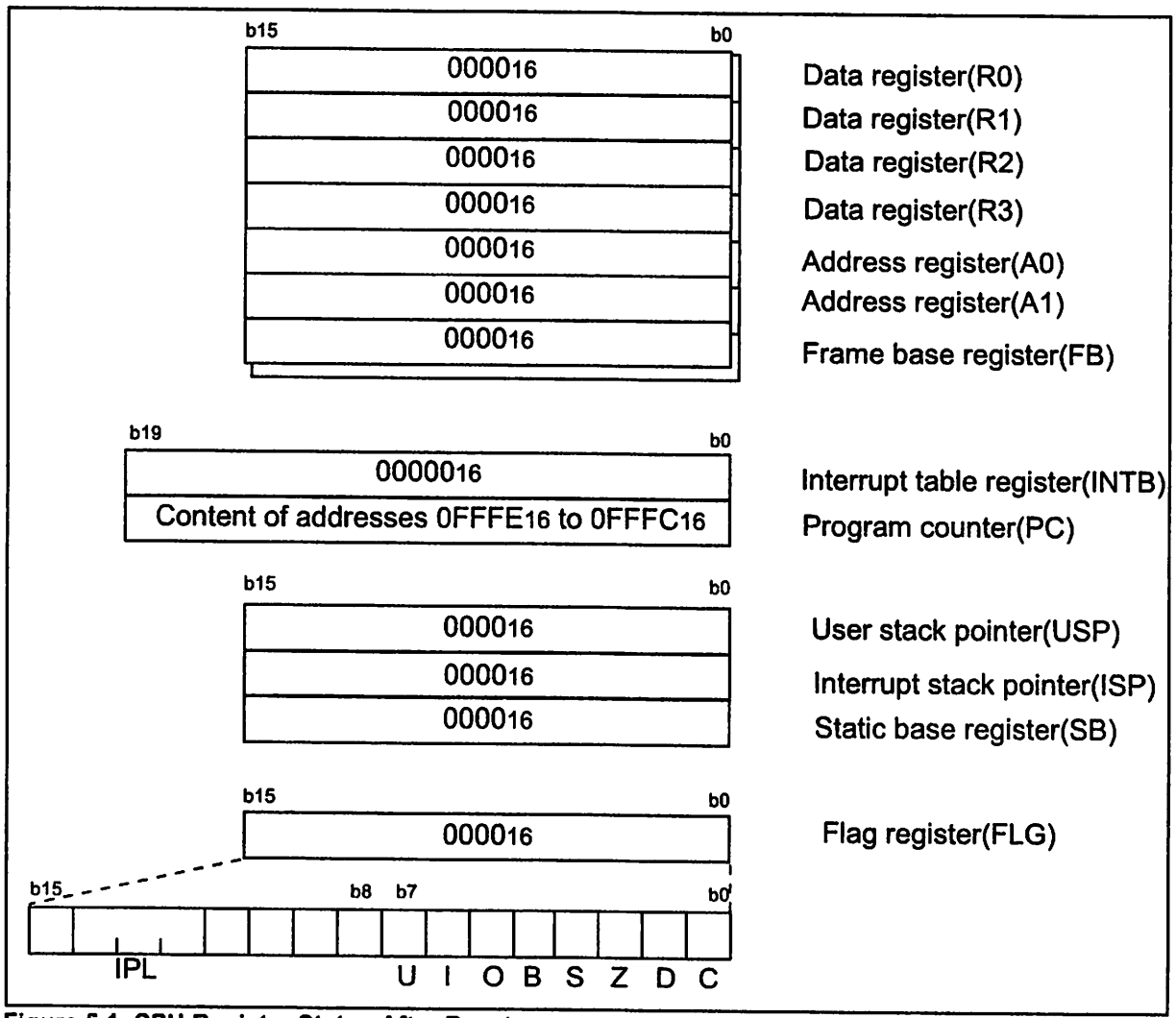


Figure 5.1 CPU Register Status After Reset

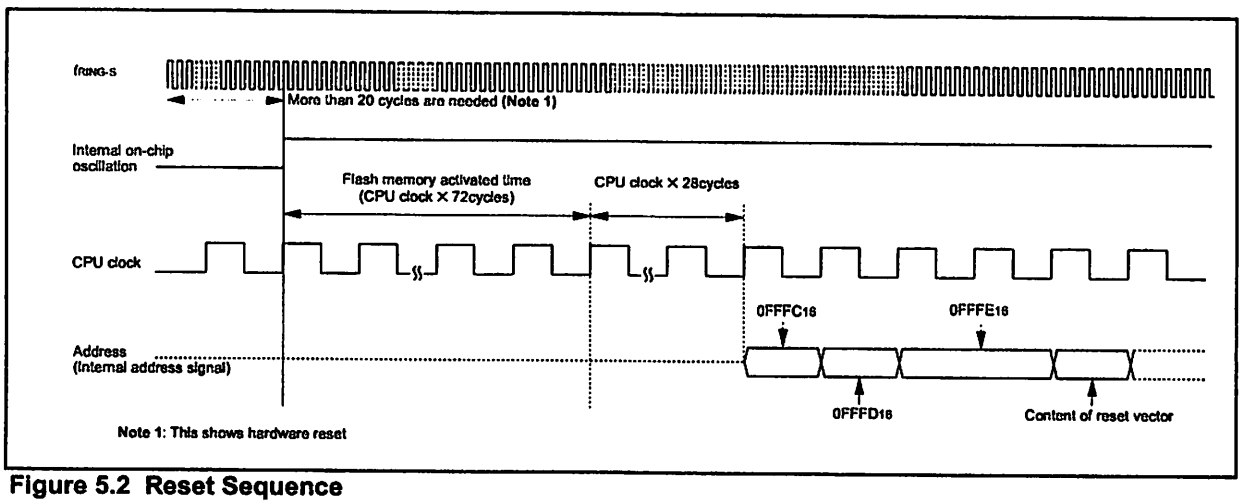


Figure 5.2 Reset Sequence

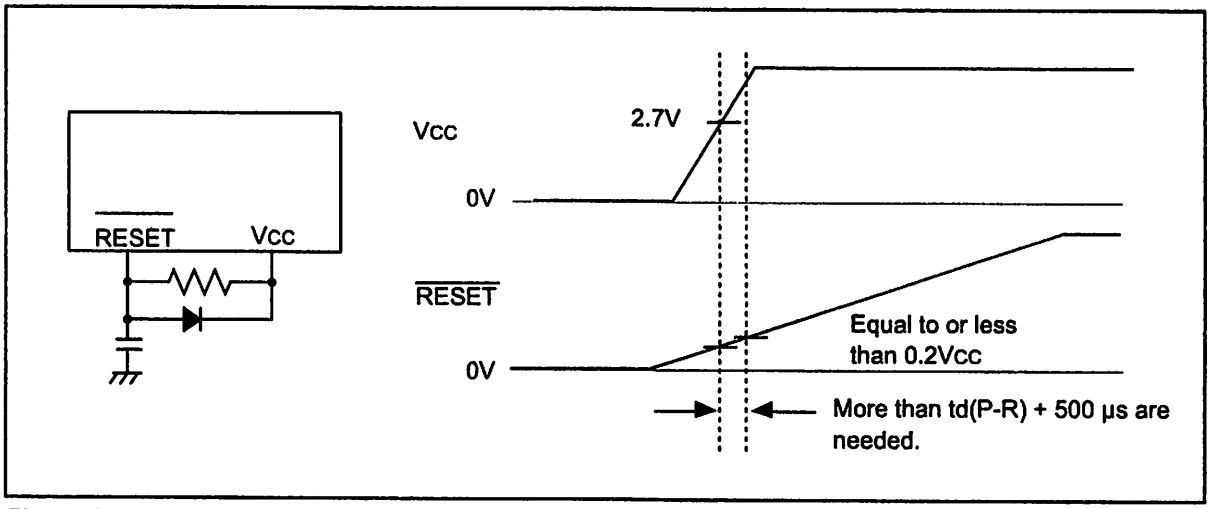


Figure 5.3 Example Reset Circuit Using The Hardware Reset 1

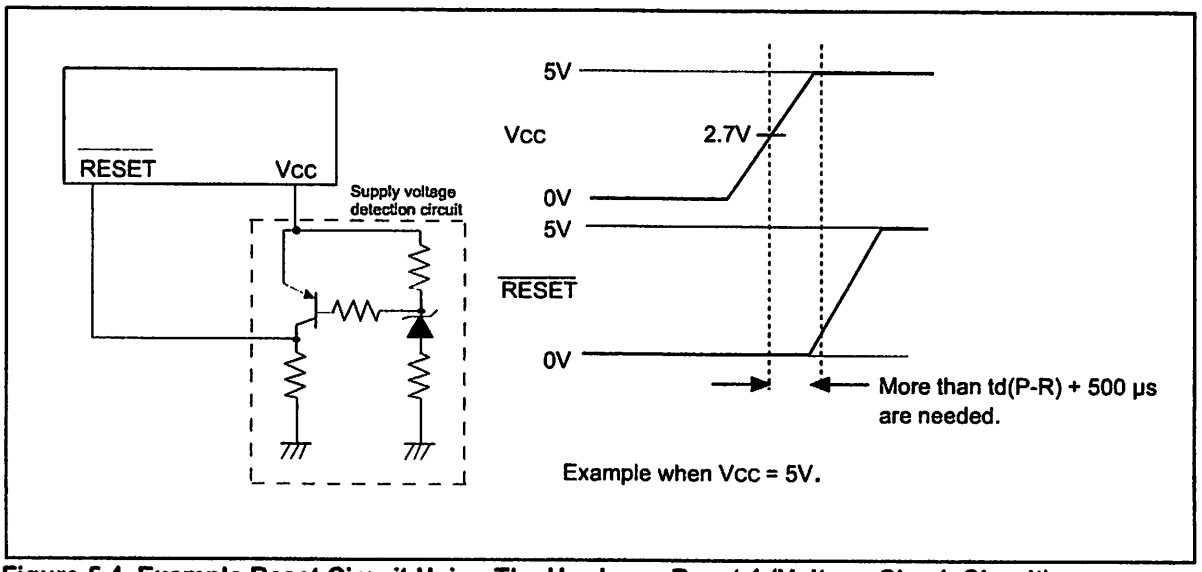


Figure 5.4 Example Reset Circuit Using The Hardware Reset 1 (Voltage Check Circuit)

5.1.2 Hardware Reset 2

This is the reset generated by the voltage detection circuit which is built-in to the microcomputer. The voltage detection circuit monitors the input voltage at Vcc input pin. The microcomputer is reset when the voltage at the Vcc input pin drops below Vdet if all of the following conditions hold true.

- The VC27 bit in the VCR2 register is set to "1" (voltage detection circuit enabled)
- The D40 bit in the D4INT register is set to "1" (voltage detection interrupt enabled)
- The D46 bit in the D4INT register is set to "1" (hardware reset 2 when going through Vdet)

When using a digital filter (D41 bit in the D4INT register is set to "1"), set the CM14 bit in the CM1 register to "0"(low-speed on-chip oscillator oscillates).

Conversely, when the input voltage at the Vcc pin rises to Vdet or more, the pins, CPU, and SFR are initialized and counting the low-speed on-chip oscillator starts. When counting the low-speed on-chip oscillator clock 32 times, the internal reset is exited and the program is executed beginning with the address indicated by the reset vector. The initialized pins and registers and the status thereof are the same as in hardware reset 1. Refer to Section 5.4 "Voltage Detection Circuit."

5.1.3 Power-on Reset Function

The power-on reset is the function which can reset the microcomputer without the external reset circuit. The $\overline{\text{RESET}}$ pin should be connected to the Vcc pin via about 5 kΩ pull-up resistance using the power-on reset function, the function turns to active and the microcomputer has its pins, CPU, and SFR initialized.

When the input voltage at the Vcc pin reaches to the Vdet level, count operation of the low-speed on-chip oscillator clock starts. When the operation counts the low-speed on-chip oscillator clock for 32 times, the internal reset is released. Then the program is executed starting from the address indicated by the reset vector. The initialized pins and registers and the status thereof are the same as in hardware reset 1 excluding the following bits.

- The D40 bit in the D4INT register turns to "1" automatically (voltage detection interrupt enabled)
- The D46 bit in the D4INT register turns to "1" automatically (hardware reset 2 when going through Vdet)

Additionally, the hardware reset 2 turns to active after the power-on reset. This is because the VC27 bit in the VCR2 register is set to "1" (voltage detection circuit enabled) after the power-on reset same as the hardware reset 1, so that hardware reset 2 active conditions are all satisfied including above D40 and D46 bit conditions.

Figure 5.5 shows the power-on reset circuit. Figure 5.6 shows the power-on reset operation.

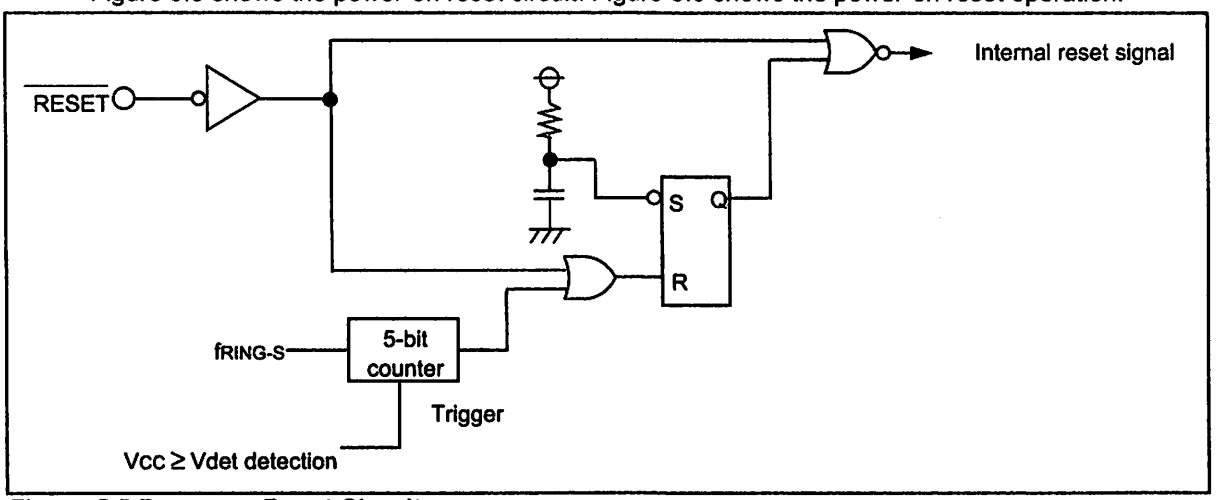
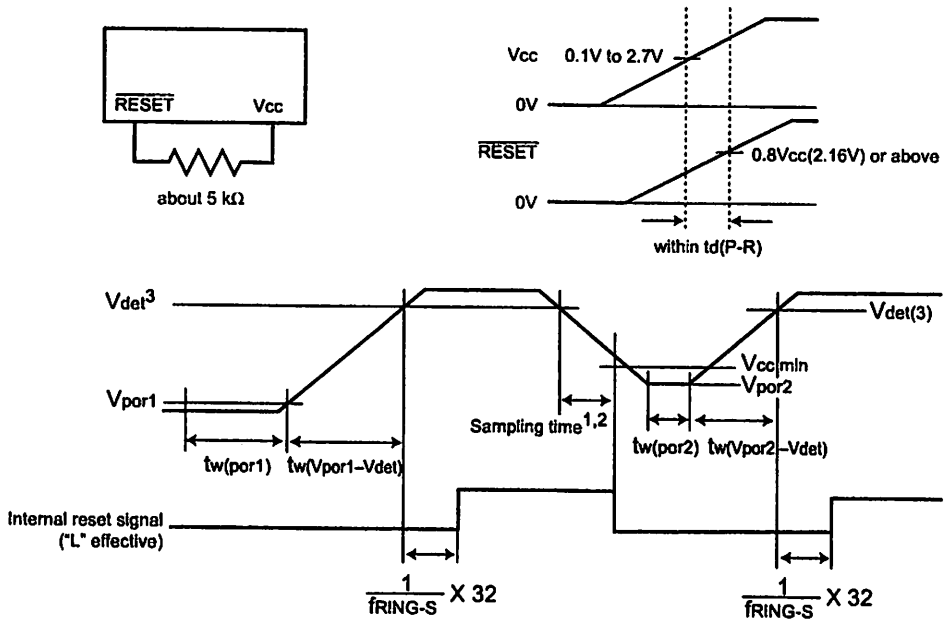


Figure 5.5 Power-on Reset Circuit



NOTES:
1. Hold the voltage of the microcomputer operation voltage range (Vccmin or above) within sampling time.
2. A sampling clock is selectable. Refer to "5.4 Voltage Detection Circuit" for details.
3. Vdet shows the voltage detection level of the voltage detection circuit. Refer to "5.4 Voltage Detection Circuit" for details.
4. Refer to Table 16.7, 16.8 for electrical characteristics.

Figure 5.6 Power-on Reset Operation

5.2 Software Reset

When the PM03 bit in the PM0 register is set to "1" (microcomputer reset), the microcomputer has its pins, CPU, and SFR initialized. Then the program is executed starting from the address indicated by the reset vector.

After reset, the low-speed on-chip oscillator clock divided by 8 is automatically selected for the CPU.

Some SFRs are not initialized by the software reset. Refer to Chapter 4, "SFR."

5.3 Watchdog Timer Reset

Where the PM12 bit in the PM1 register is "1" (reset when watchdog timer underflows), the microcomputer initializes its pins, CPU and SFR if the watchdog timer underflows. Then the program is executed starting from the address indicated by the reset vector.

After reset, the low-speed on-chip oscillator clock divided by 8 is automatically selected for the CPU.

Some SFRs are not initialized by the watchdog timer reset. Refer to Chapter 4, "SFR."

5.4 Voltage Detection Circuit

The voltage detection circuit monitors the input voltage at the Vcc pin with respect to Vdet. The user program can check for voltage detection using the VC13 bit or set up the voltage detection interrupt register to generate a hardware reset 2 or voltage detection interrupt.

Figure 5.7 shows the voltage detection circuit. Figure 5.8 shows VCR1 and VCR2 registers. Figure 5.9 shows the D4INT register. Figure 5.10 shows an operation example of the voltage detection circuit. Figure 5.11 to 5.12 show the operation example of the voltage detection circuit to get out of stop mode.

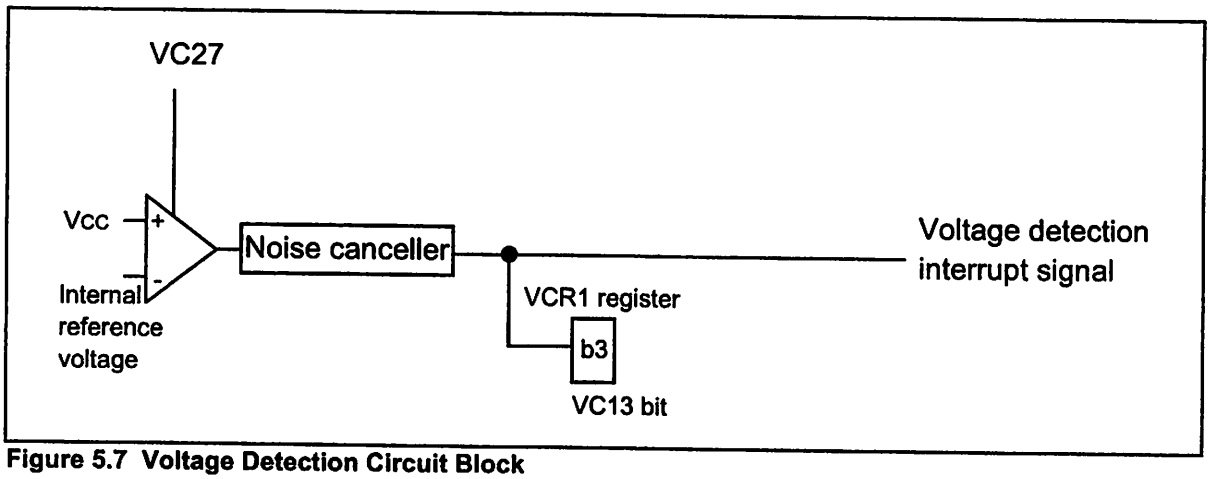


Figure 5.7 Voltage Detection Circuit Block

Voltage detection register 1

b7	b6	b5	b4	b3	b2	b1	b0
0	0	0	0		0	0	0

Symbol: VCR1 Address: 001916 After reset(2): 000010002

Bit symbol	Bit name	Function	RW
(b2-b0)	Reserved bit	Should set to "0"	RW
VC13	Voltage monitor flag1	0:Vcc < Vdet 1:Vcc ≥ Vdet or voltage detection circuit disabled	RO
(b7-b4)	Reserved bit	Should set to "0"	RW

Notes:
1. The VC13 bit is valid when the VC27 bit in the VCR2 register is set to "1" (voltage detection circuit enabled). The VC13 bit is set to "1" (Vcc≥Vdet or voltage detection circuit disabled) when the VC27 bit in the VCR2 register is set to "0" (voltage detection circuit disabled).
2. Software reset or the watchdog timer reset does not affect this register.

Voltage detection register 2(1)

b7	b6	b5	b4	b3	b2	b1	b0
	0	0	0	0	0	0	0

Symbol: VCR2 Address: 001A16 After reset(3): Reset input : 001616 RESET pin = "H" retaining : 100000002

Bit symbol	Bit name	Function	RW
(b6-b0)	Reserved bit	Should set to "0"	RW
VC27	Voltage monitor enable bit(2)	0: Voltage detection circuit disabled 1: Voltage detection circuit enabled	RW

Notes:
1. Set the PRC3 bit in the PRCR register to "1" (write enabled) before writing to this register.
2. Set the VC27 bit to "1" (voltage detect circuit enabled) when hardware reset 2 is used. After the VC27 bit is set to "1", the voltage detection circuit elapses for $t_{d(E-A)}$ before starting operation.
3. Software reset or the watchdog timer reset does not affect this register.

Figure 5.8 VCR1 Register and VCR2 Register

Voltage detection interrupt register[1]

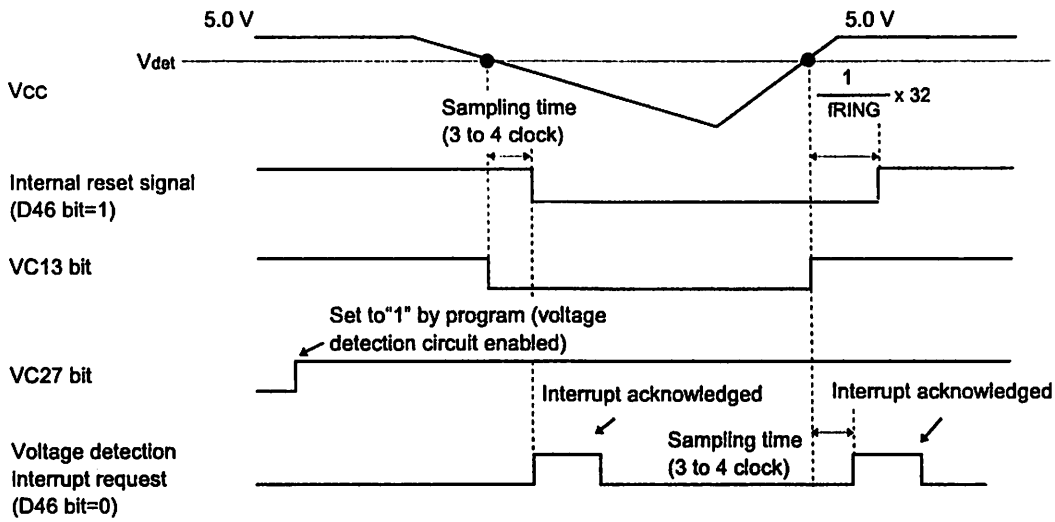
b7	b6	b5	b4	b3	b2	b1	b0

Symbol: D4INT
Address: 001F16
After reset(10): Reset input : 0016; RESET pin = H retaining : 01000001₂

Bit symbol	Bit name	Function	RW
D40	Voltage detection interrupt enable bit[7]	0 : Disable 1 : Enable	RW
D41	Voltage detection digital filter disable mode select bit	0: Digital filter enable mode (digital filter circuit enabled) 1: Digital filter disable mode (digital filter circuit disabled)	RW
D42	Voltage change detection flag[3, 4, 5]	0: Not detected 1: Vdet passing detection	RW
D43	WDT overflow detect flag[3, 4]	0: Not detected (flag clear) 1: Detected	RW
DF0	Sampling clock select bit	b5b4 00 : fRING-S divided by 1 01 : fRING-S divided by 2 10 : fRING-S divided by 4 11 : fRING-S divided by 8	RW
DF1			RW
D46	Voltage monitor mode select bit[6]	0: Voltage detection interrupt request is generated when passing through Vdet 1: Hardware reset 2 when passing through Vdet	RW
D47	Voltage detection condition select bit(11)	Voltage detection interrupt request is generated or hardware reset 2 when Vcc passes Vdet[9] 0: Over Vdet 1: Below Vdet	RW

Notes:
1. Set the PRC3 bit in the PRCR register to "1" (write enable) before writing to this register.
2. If the voltage detection interrupt needs to be used to get out of stop mode again after once used for that purpose, reset the D41 bit by writing a "0" and then a "1".
3. Valid when the VC27 bit in the VCR2 register is set to "1" (voltage detection circuit enabled).
4. If the VC27 bit is set to "0" (voltage detection circuit disabled), the D42 and D43 bits are set to "0" (not detected).
5. This bit is set to "0" by writing a "0" in a program. (writing a "1" has no effect.)
6. Valid when the D40 bit is set to "1" (voltage detection interrupt enabled).
7. The D40 bit is valid when the VC27 bit in the VCR2 register is set to "1" (voltage detection circuit enabled). When setting the D40 bit to "1", the following setting is required.
   (1) Set the VC27 bit "1".
   (2) Wait for td(E-A) until the detecter circuit operates.
   (3) Wait for the sampling time (the sampling clock which is selected in the DF0 bit to DF1 bit times 4 cycles.)
   (4) Set the D40 bit to "1".
   (5) Set the CM14 bit in the CM1 register to "0" (low-speed on-chip oscillator on).
8. Valid when the D41 bit is set to "1" (digital filter disabled mode).
9. The D46 bit can be selected.
10. The software reset or the watchdog timer reset do not affect this register.
11. When the D46 bit is set to "1" (hardware reset 2 when Vdet passes), set the D47 bit to "1" (below Vdet). (Do not set to "0").

Figure 5.9 D4INT Register



The above applies to the following conditions.
- D4INT register D40 bit = 1 (voltage detection interrupt enabled)
- D4INT register D41 bit = 0 (digital filter enabled mode)

Sampling time : 4 cycles of sampling clock selected in DF0 bit to DF1 bit

Figure 5.10 Operation Example of Voltage Detection Circuit

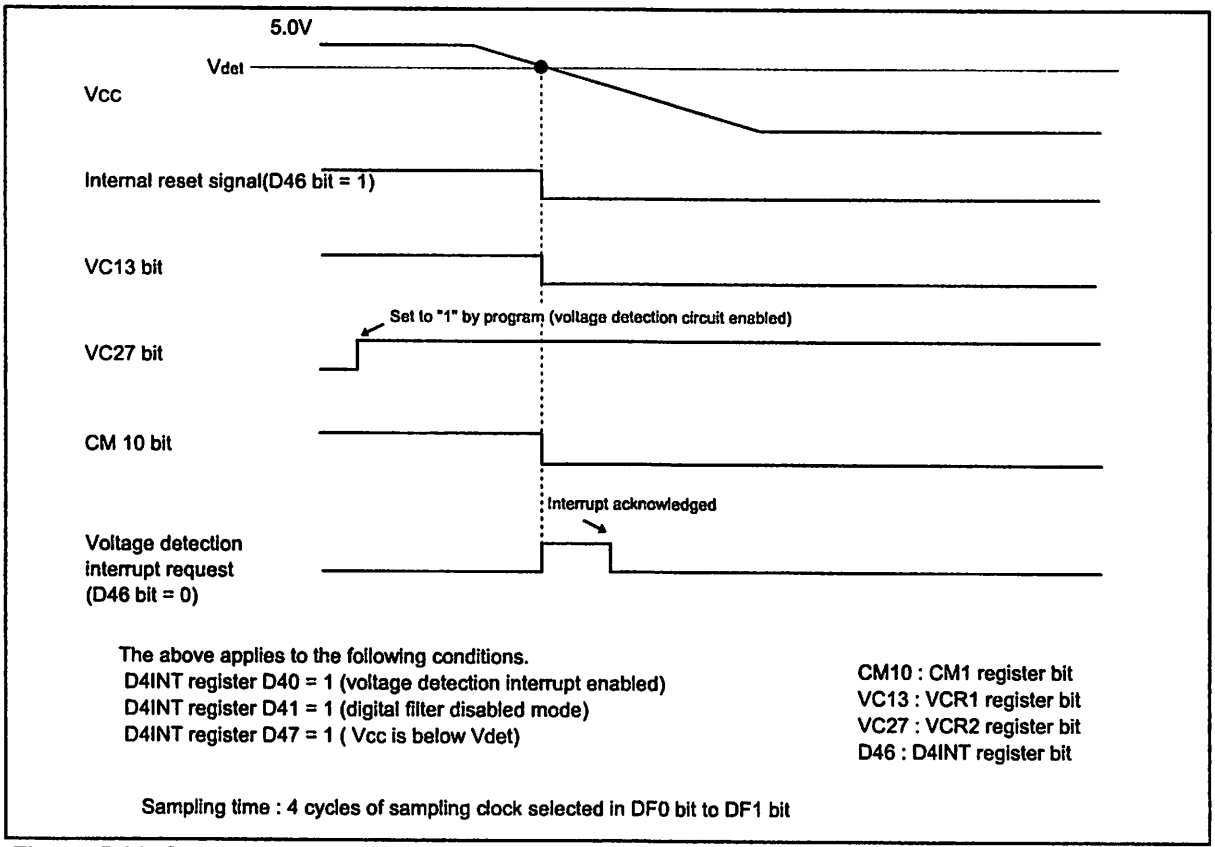


Figure 5.11 Operation Example of Voltage Detection Circuit to get out of Stop mode (1)

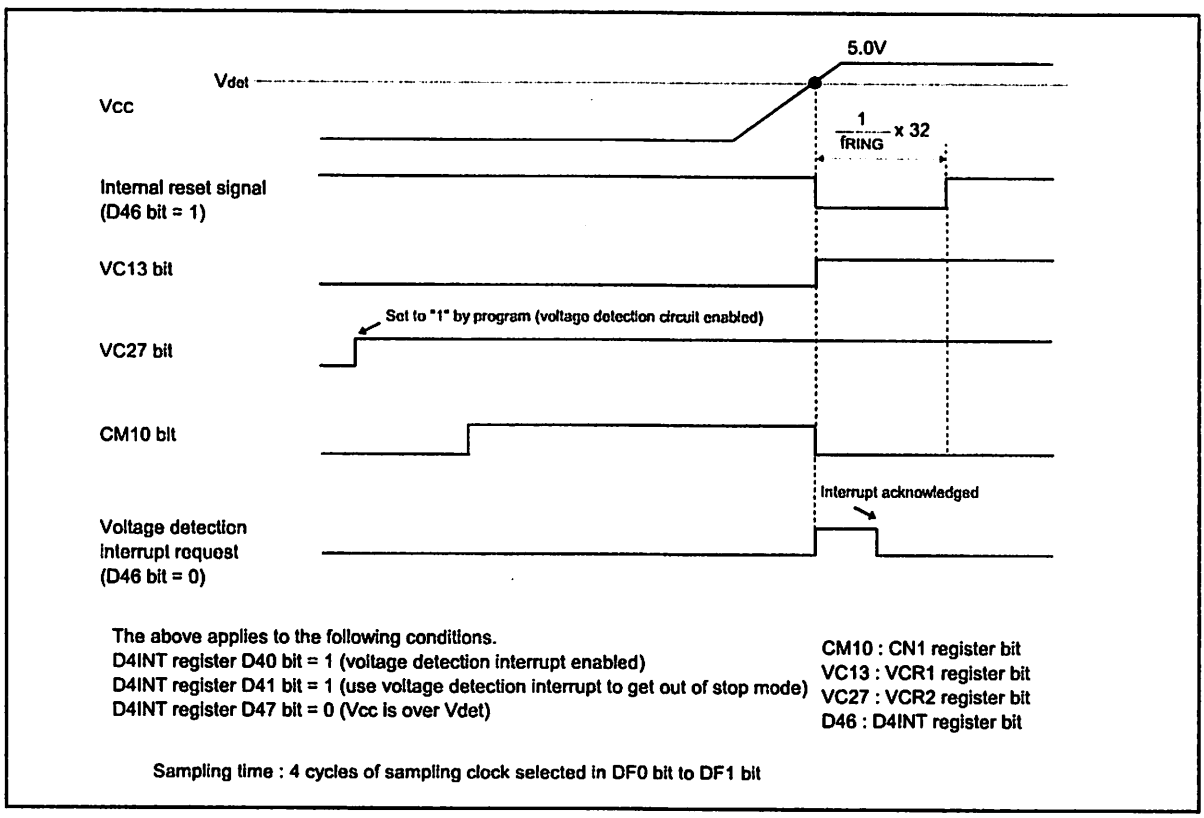


Figure 5.12 Operation Example of Voltage Detection Circuit to get out of Stop mode (2)

5.4.1 Voltage Detection Interrupt

Figure 5.13 shows the block diagram of voltage detection interrupt generation circuit.
Refer to 5.4.2, "Exiting Stop Mode on a Voltage Detection Circuit" for Getting out of stop mode due to the voltage detection interrupt.

A voltage detection interrupt is generated when the input voltage at the Vcc pin rises to Vdet or more or drops below Vdet if all of the following conditions hold true in normal operation mode and wait mode.

- The VC27 bit in the VCR2 register is set to "1" (voltage detection circuit enabled)
- The D40 bit in the D4INT register is set to "1" (voltage detection interrupt enabled)
- The D46 bit in the D4INT register is set "0" (voltage detection interrupt selected)

To use the digital filter (D41 bit in the D4INT register is set to "0"), set the CM14 bit in the CM1 register to "0" (low-speed on-chip oscillator on). Figure 5.14 shows an operation example of voltage detection interrupt generation circuit.
The voltage detection interrupt shares the interrupt vector with the watchdog timer interrupt and oscillation stop detection interrupt.
The D42 bit in the D4INT register becomes "1" when passing through Vdet is detected after the voltage inputted to the Vcc pin is up or down.
A voltage detection interrupt request is generated when the D42 bit changes state from "0" to "1". The D42 bit needs to be set to "0" in a program.
Table 5.2 lists the voltage detection interrupt request generation conditions.
It takes 4 cycles of sampling clock until the D42 bit is set to "1" since the voltage which inputs to Vcc pin passes Vdet.
It is possible to set the sampling clock detecting that the voltage applied to the Vcc pin has passed through Vdet with the DF0 to DF1 bits in the D4INT register.

Table 5.2 Voltage Detection Interrupt Request Generation Conditions

Operation mode	VC27 bit	D40 bit	D41 bit	D42 bit	D46 bit	VC13 bit	CM14 bit
Normal operation mode[1]	1	1	0 or 1	0	0	From 0 to 1[2]	0
						From 1 to 0[2]	
Wait mode	1	1	0 or 1	0	0	From 0 to 1[2]	0
						From 1 to 0[2]	

Notes:
1. The status except the wait mode and stop mode is handled as the normal mode. (Refer to Chapter 6, "Clock Generation Circuit.")
2. Refer to Figure 5.14, "Operation Example of Voltage Detection Interrupt Generation Circuit" for interrupt generation timing.

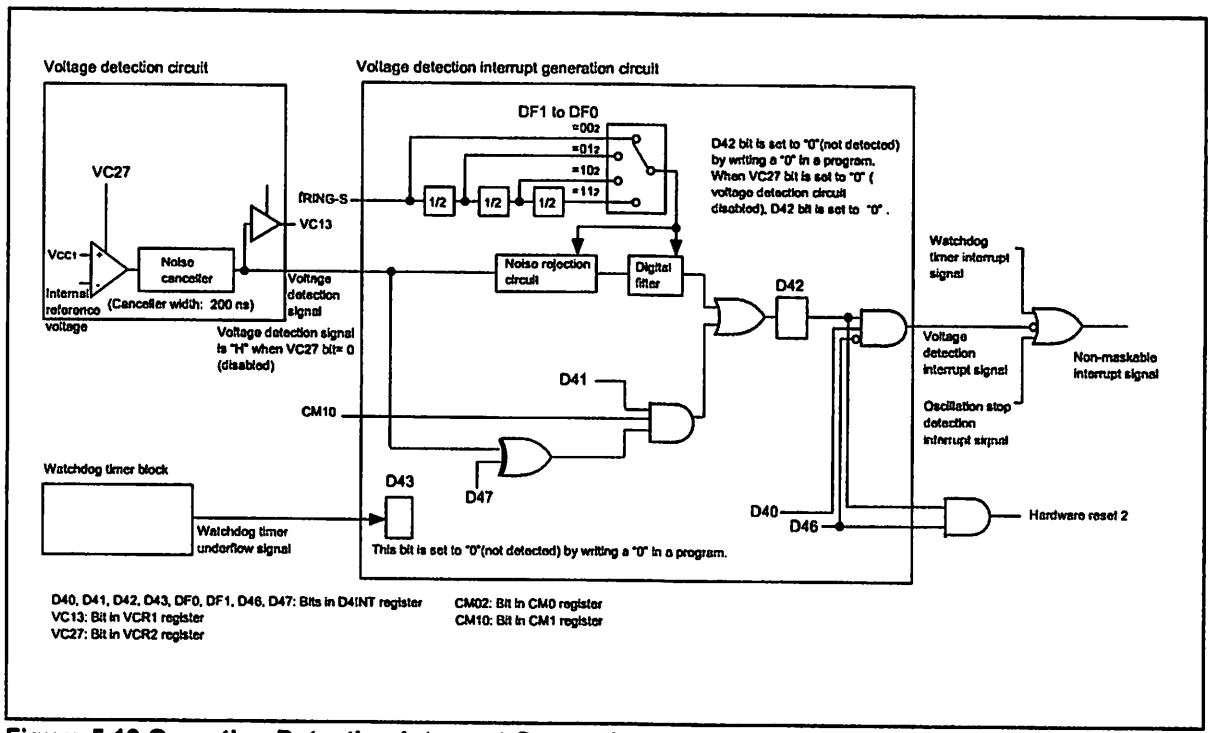


Figure 5.13 Operation Detection Interrupt Generation Block

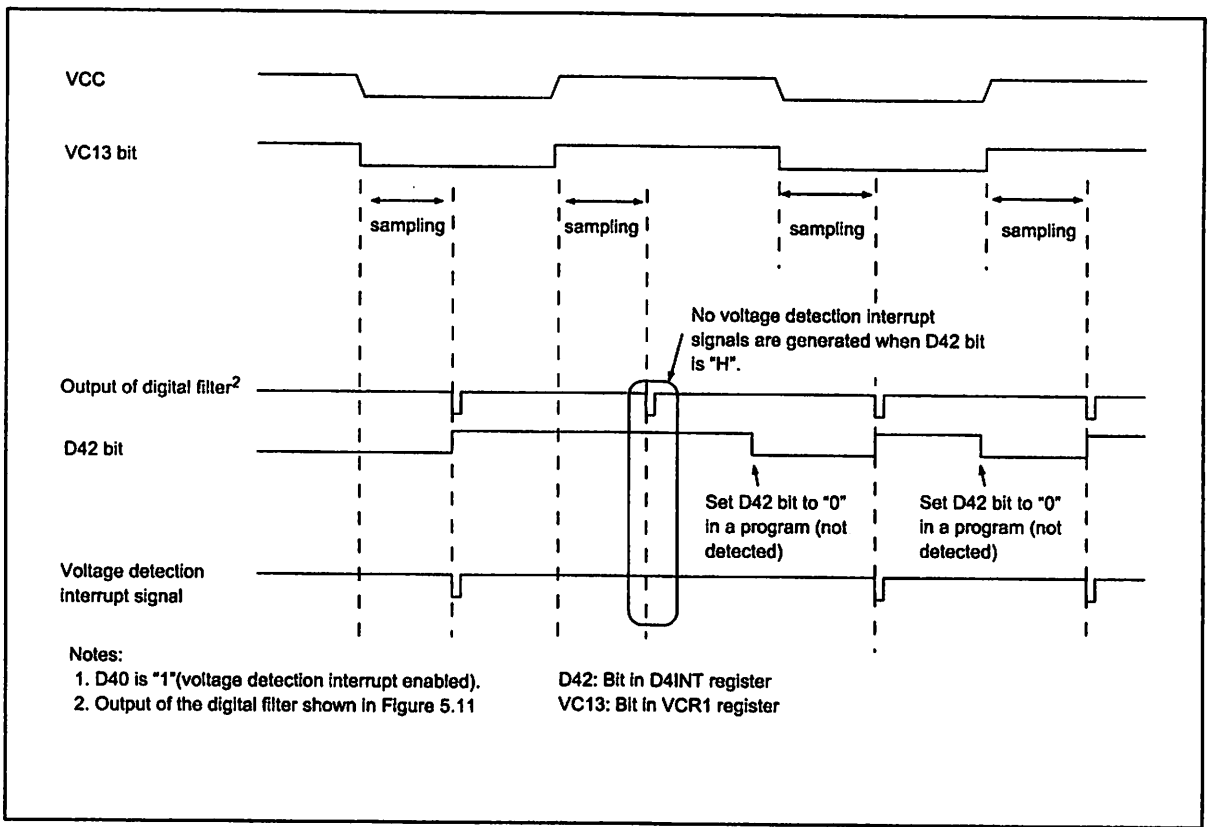


Figure 5.14 Voltage Detection Interrupt Generation Circuit Operation Example

5.4.2 Exiting Stop Mode on a Voltage Detection Interrupt

A voltage detection interrupt is generated when the input voltage at the Vcc pin rises to Vdet or more or drops below Vdet if all of the following conditions hold true in stop mode.

- The VC27 bit in the VCR2 register is set to "1" (voltage detection circuit enabled)
- The D40 bit in the D4INT register is set to "1" (voltage detection interrupt enabled)
- The D41 bit in the D4INT register is set "1" (digital filter disabled mode)
- The D46 bit in the D4INT register is set "0" (voltage detection interrupt selected)

The voltage detection interrupt shares the interrupt vector with the watchdog timer interrupt and oscillation stop detection interrupt.

The D42 bit in the D4INT register becomes "1" when passing through Vdet is detected after the voltage inputted to the Vcc pin is up or down.

A voltage detection interrupt request is generated when the D42 bit changes state from "0" to "1". The D42 bit needs to be set to "0" in a program.

Table 5.3 lists the voltage detection interrupt request generation conditions to get out of stop mode.

Table 5.3 Voltage Detection Interrupt Request Generation Conditions to get out of Stop mode

Operation mode	VC27 bit	D40 bit	D41 bit	D42 bit	D46 bit	D47 bit	VC13 bit
Stop mode	1	1	1	0	0	0 or 1	From 0 to 1
							From 1 to 0

Notes:
1. The status except the wait mode and stop mode is handled as the normal mode. (Refer to Chapter 6, "Clock Generation Circuit.")
2. Refer to Figure 5.14, "Operation Example of Voltage Detection Interrupt Generation Circuit" for interrupt generation timing.

6. Clock Generation Circuit

The clock generation circuit contains two oscillator circuits as follows:

- Main clock oscillation circuit
- On-chip oscillator (oscillation stop detect function)

Table 6.1 lists the clock generation circuit specifications. Figure 6.1 shows the clock generation circuit. Figures 6.2 to 6.4 show the clock-related registers.

Table 6.1 Clock Generation Circuit Specifications

Item	Main clock oscillation circuit	On-chip oscillator	
		High-speed on-chip oscillator	Low-speed on-chip oscillator
Use of clock	• CPU clock source • Peripheral function clock source	• CPU clock source • Peripheral function clock source • CPU and peripheral function clock sources when the main clock stops oscillating	• CPU clock source • Peripheral function clock source • CPU and peripheral function clock sources when the main clock stops oscillating
Clock frequency	0 to 20 MHz	Approx. 8 MHz	Approx. 125 kHz
Usable oscillator	• Ceramic resonator • Crystal oscillator	———	———
Pins to connect oscillator	XIN, XOUT[1]	Note[1]	Note[1]
Oscillation starts and stops	Present	Present	Present
Oscillator status after reset	Stopped	Stopped	Oscillating
Other	Externally derived clock can be input[2]	———	———

Notes:
1. Can be used as P46 and P47 when the on-chip oscillator clock is used for CPU clock while the main clock oscillation circuit is not used.
2. Set the CM05 bit in the CM0 register to "1" (main clock stops) and the CM13 bit in the CM1 register to "1" (XIN-XOUT pin) when the external clock is input.

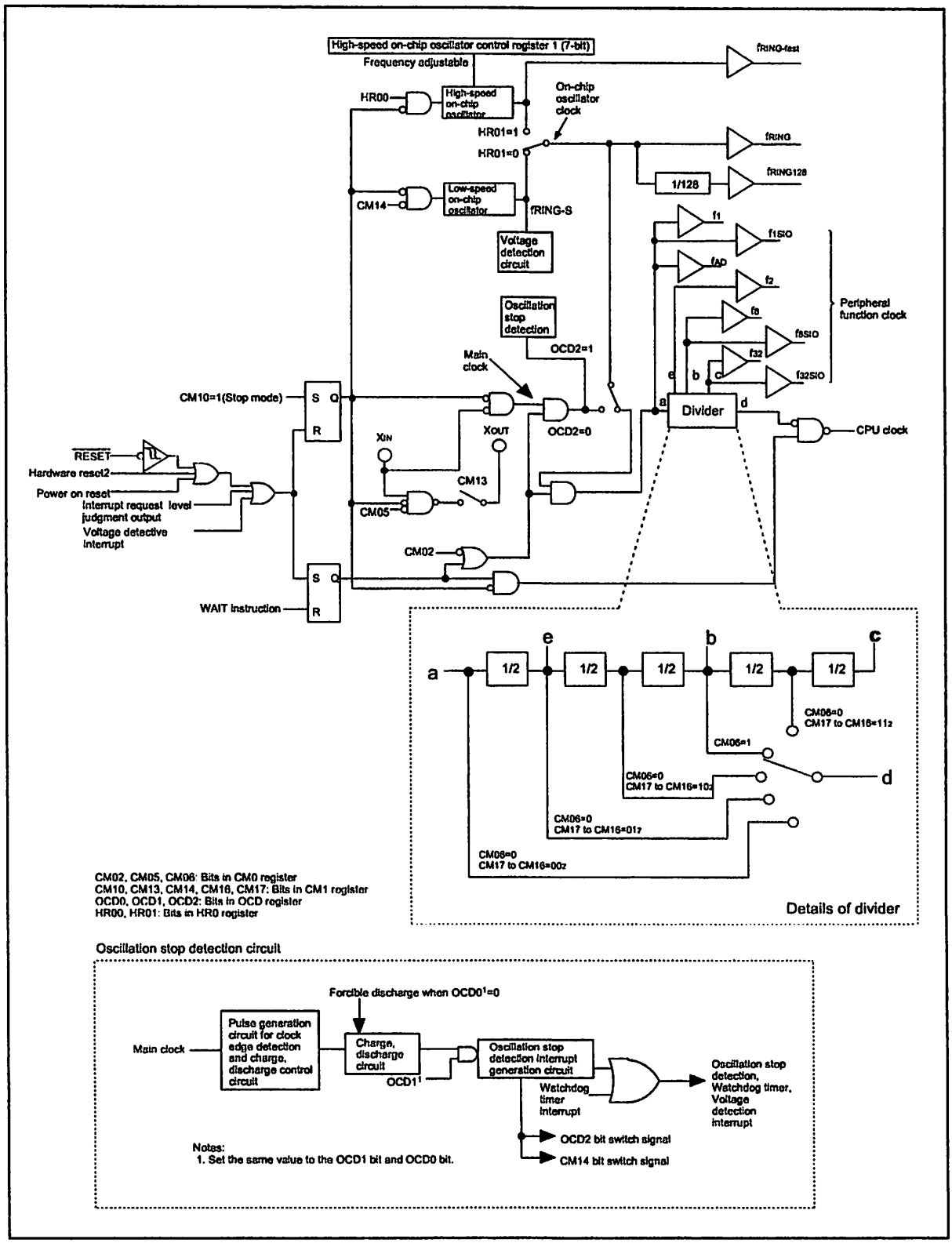


Figure 6.1 Clock Generation Circuit

System clock control register 0[1]

b7	b6	b5	b4	b3	b2	b1	b0
0			0	1		0	0

Symbol: CM0 | Address: 000616 | After reset: 6816

Bit symbol	Bit name	Function	RW
(b1-b0)	Reserved bit	Set to "0"	RW
CM02	WAIT peripheral function clock stop bit	0 : Do not stop peripheral function clock in wait mode 1 : Stop peripheral function clock in wait mode	RW
(b3)	Reserved bit	Set to "1"	RW
(b4)	Reserved bit	Set to "0"	RW
CM05	Main clock (XIN-XOUT)stop bit[2, 4]	0 : On 1 : Off[3]	RW
CM06	CPU clock division select bit 0[5]	0 : CM16 and CM17 valid 1 : Divide-by-8 mode	RW
(b7)	Reserved bit	Set to "0"	RW

Notes
1: Set the PRC0 bit of PRCR register to "1" (write enable) before writing to this register.
2: The CM05 bit is provided to stop the main clock when the on-chip oscillator mode is selected. This bit cannot be used for detection as to whether the main clock stopped or not. To stop the main clock, the following setting is required:
(1) Set the OCD0 and OCD1 bits in the OCD register to "002" (disable oscillation stop detection function).
(2) Set the OCD2 bit to "1" (on-chip oscillator clock selection).
3: Set the CM05 bit to "1" (main clock stops) and the CM13 bit in the CM1 register to "1" (XIN-XOUT pin) when the external clock is input.
4: When the CM05 bit is set to "1" (main clock stop), P46 and P47 can be used as input ports.
5: When entering stop mode from high or middle speed mode, the CM06 bit is set to "1" (divide-by-8 mode).

System clock control register 1[1]

b7	b6	b5	b4	b3	b2	b1	b0
					0	0	

Symbol: CM1 | Address: 000716 | After reset: 2016

Bit symbol	Bit name	Function	RW
CM10	All clock stop control bit[4,7]	0 : Clock on 1 : All clocks off (stop mode)	RW
(b1)	Reserved bit	Set to "0"	RW
(b2)	Reserved bit	Set to "0"	RW
CM13	Port XIN-XOUT switch bit[7]	0 : Input port P46, P47 1 : XIN-XOUT pin	RW
CM14	Low-speed on-chip oscillation stop bit[5,6]	0 : Low-speed on-chip oscillator on 1 : Low-speed on-chip oscillator off	RW
CM15	XIN-XOUT drive capacity select bit[2]	0 : LOW 1 : HIGH	RW
CM16	CPU clock division select bit 1[3]	b7 b6 0 0 : No division mode 0 1 : Division by 2 mode 1 0 : Division by 4 mode 1 1 : Division by 16 mode	RW
CM17			RW

Notes:
1: Write to this register after setting the PRC0 bit of PRCR register to "1" (write enable).
2: When entering stop mode from high or middle speed mode, the CM15 bit is set to "1" (drive capacity high).
3: Effective when the CM06 bit is "0" (CM16 and CM17 bits enable).
4: If the CM10 bit is "1" (stop mode), the internal feedback resistor becomes ineffective.
5: The CM14 bit can be set to "1" (low-speed on-chip oscillator off) if the OCD2 bit=0 (main clock selected). When the OCD2 bit is set to "1" (selecting on-chip oscillator clock), the CM14 bit is set to "0" (low-speed on-chip oscillator on). This bit remains unchanged when "1" is written.
6: When using voltage detection interrupt circuit, CM14 bit is set to "0"
7: When the CM10 bit is set to "1" (stop mode) or the CM05 bit in the CM0 register to "1" (main clock stops) and the CM13 bit is set to "1" (XIN-XOUT pin), the XOUT (P47) pin becomes "H".
When the CM13 bit is set to "0" (input ports, P46, P47), the P47 (XOUT) enters input mode.

Figure 6.2 CM0 Register and CM1 Register

Oscillation stop detection register[1]

b7	b6	b5	b4	b3	b2	b1	b0
0	0	0	0				

Symbol: OCD | Address: 000C16 | After reset: 0416

Bit symbol	Bit name	Function	RW
OCD0	Oscillation stop detection enable bit	b1 b0 0 0: The function is disabled[4] 0 1: Do not set 1 0: Do not set 1 1: The function is enabled[7]	RW
OCD1			
OCD2	System clock select bit[6]	0: Select main clock[7] 1: Select on-chip oscillator clock[2]	RW
OCD3	Clock monitor bit[3, 5]	0: Main clock on 1: Main clock off	RO
(b7-b4)	Reserved bit	Set to "0"	RW

Notes:
1. Set the PRC0 bit in the PRCR register to "1" (write enable) before rewriting this register.
2. The OCD2 bit is set to "1" (selecting on-chip oscillator clock) automatically if a main clock oscillation stop is detected while the OCD1 to OCD0 bits are set to "11₂" (oscillation stop detection function enabled). If the OCD3 bit is set to "1" (main clock stops), the OCD2 bit remains unchanged when trying to write "0" (main clock selected).
3. The OCD3 bit is enabled when the OCD1 to OCD0 bits are set to "11₂" (oscillation stop detection function enabled). Read the OCD3 bit several times with the oscillation stop detection interrupt processing program to determine the main clock state.
4. The OCD1 to OCD0 bits should be set to "00₂" (oscillation stop detection function disabled) before entering stop mode or on-chip oscillator (main clock stops).
5. The OCD3 bit remains set to "0" (main clock on) if the OCD1 to OCD0 bits are set to "00₂".
6. The CM14 bit goes to "0" (low-speed on-chip oscillator on) if the OCD2 bit is set to "1" (on-chip oscillator clock selected).
7. Refer to Figure 6.7 "switching clock source from low-speed on-chip oscillator to main clock" for the switching procedure when the main clock re-oscillates after detecting an oscillation stop.

Figure 6.3 OCD Register

High-speed on-chip oscillator control register 0[3]

b7	b6	b5	b4	b3	b2	b1	b0
0	0	0	0	0	0	0	

Symbol	Address	After reset
HR0	000816	0016

Bit symbol	Bit name	Function	RW
HR00	High-speed on-chip oscillator enable bit	0: High-speed on-chip oscillator off 1: High-speed on-chip oscillator on	RW
HR01	High-speed on-chip oscillator select bit[1]	0: Low-speed on-chip oscillator selected[2] 1: High-speed on-chip oscillator selected	RW
----- (b7-b2)	Reserved bit	Set to "0"	RW

Notes:
1. The HR01 bit should be changed under the following conditions.
   - HR00 = 1 (high-speed on-chip oscillator on)
   - CM1 register CM14 bit = 0 (low-speed on-chip oscillator on)
2. When writing "0"(low-speed on-chip oscillator selected) to the HR01 bit, do not write "0"(high-speed on-chip oscillator stops) to the HR00 bit simultaneously. Set the HR00 bit to "0" after setting the HR01 bit to "0".
3. Set the PRC0 bit in the PRCR register to "1" (write enable) before rewriting this register.

High-speed on-chip oscillator control register 1[1]

b7	b6	b5	b4	b3	b2	b1	b0
0							

Symbol	Address	After reset
HR1	000B16	4016

Function	RW
The frequency of high-speed on-chip oscillator is adjusted with bits 0 to bits 6. Period of high-speed on-chip oscillator = td(HR offset) + (64 × b6 + 32 × b5 + 16 × b4 + 8 × b3 + 4 × b2 + 2 × b1 + b0) × td(HR) Bit 7 should be set to "0".	RW

Note:
1. Set the PRC0 bit in the PRCR register to "1" (write enable) before rewriting this register.

Figure 6.4 HR0 Register and HR1 Register

The following describes the clocks generated by the clock generation circuit.

6.1 Main Clock

This clock is supplied by a main clock oscillation circuit. This clock is used as the clock source for the CPU and peripheral function clocks. The main clock oscillator circuit is configured by connecting a resonator between the XIN and XOUT pins. The main clock oscillator circuit contains a feedback resistor, which is disconnected from the oscillator circuit during stop mode in order to reduce the amount of power consumed in the chip. The main clock oscillator circuit may also be configured by feeding an externally generated clock to the XIN pin. Figure 6.5 shows examples of main clock connection circuit.

During reset and after reset, the main clock is turned off.

The main clock starts oscillating when the CM05 bit in the CM0 register is set to "0" (main clock on) after setting the CM13 bit in the CM1 register to "1" (XIN- XOUT pin).

To use the main clock for the CPU clock, set the OCD2 bit in the OCD register to "0" (selecting main clock) after the main clock becomes oscillating stably.

The power consumption can be reduced by setting the CM05 bit in the CM0 register to "1" (main clock off) if the OCD2 bit is set to "1" (on-chip oscillator clock selected).

Note that if an externally generated clock is fed into the XIN pin, the main clock cannot be turned off by setting the CM05 bit to "1". If necessary, use an external circuit to turn off the clock.

During stop mode, all clocks including the main clock are turned off. Refer to Section 6.4, "Power Control."

Note: Insert a damping resistor if required. The resistance will vary depending on the oscillator and the oscillation drive capacity setting. Use the value recommended by the maker of the oscillator.
When the oscillation drive capacity is set to low, check that oscillation is stable. Also, if the oscillator manufacturer's data sheet specifies that a feedback resistor be added external to the chip, insert a feedback resistor between XIN and XOUT following the instruction.

Figure 6.5 Examples of Main Clock Connection Circuit

6.2 On-Chip Oscillator Clock

This clock is supplied by a on-chip oscillator. There are two kinds of on-chip oscillator: high-speed on-chip oscillator and low-speed on-chip oscillator. These oscillators are selected by the bit HR01 bit in the HR0 register.

6.2.1 Low-Speed On-Chip Oscillator Clock

The clock derived from the low-speed on-chip oscillator is used as the clock source for the CPU clock, peripheral function clock, fRING, fRING128 and fRING-S.

After reset, the on-chip oscillator clock derived from low-speed on-chip oscillator by divided by 8 is selected for the CPU clock.

If the main clock stops oscillating when the OCD1 to OCD0 bits in the OCD register are "112" (oscillation stop detection function enabled), the low-speed on-chip oscillator automatically starts operating, supplying the necessary clock for the microcomputer.

The frequency of the low-speed on-chip oscillator varies depending on the supply voltage and the operation ambient temperature. The application products must be designed with sufficient margin to accommodate the frequency range.

6.2.2 High-Speed On-Chip Oscillator Clock

The clock derived from high-speed on-chip oscillator is used as the clock source for the CPU clock, peripheral function clock, fRING, fRING128, and fRING1-fast.

After reset, the on-chip oscillator clock derived from high-speed on-chip oscillator is halted. The oscillation is started by setting the HR00 bit in the HR0 register to "1" (high-speed on-chip oscillator on). The frequency can be adjusted by the HR1 register.

The relationship between the value of HR1 register and the period of high-speed on-chip oscillator is shown below. It is noted that the difference in delay between the bits should be adjusted by changing each bit. Bit 7 should be set be "0".

Period of high-speed on-chip oscillator = td(HR offset) + (64 × b6 + 32 × b5 + 16 × b4 + 8 × b3 + 4 × b2 + 2 × b1 + b0) × td(HR)

b0 to b6 : Bits in HR1 register

6.3 CPU Clock and Peripheral Function Clock

There are two types of clocks: CPU clock to operate the CPU and peripheral function clock to operate the peripheral functions. Also refer to "Figure 6.1 Clock Generating Circuit".

6.3.1 CPU Clock

This is an operating clock for the CPU and watchdog timer.

The clock source for the CPU clock can be chosen to be the main clock or on-chip oscillator clock.

The selected clock source can be divided by 1 (undivided), 2, 4, 8 or 16 to produce the CPU clock. Use the CM06 bit in the CM0 register and the CM17 to CM16 bits in the CM1 register to select the divide-by-n value.

After reset, the low-speed on-chip oscillator clock divided by 8 provides the CPU clock.

Note that when entering stop mode from high or middle speed mode, the CM06 bit is set to "1" (divide-by-8 mode).

6.3.2 Peripheral Function Clock (f_1, f_2, f_8, f_{32}, f_{AD}, f_{1SIO}, f_{8SIO}, f_{32SIO}, f_{RING}, $f_{RING128}$)

These are operating clocks for the peripheral functions.

Of these, fi (i=1, 2, 8, 32) is derived from the main clock or on-chip oscillator clock by dividing them by i. The clock fi is used for timers X, Y, Z and C.

The clock f_{jSIO} (j=1, 8, 32) is derived from the main clock or on-chip oscillator clock by dividing them by j. The clock f_{jSIO} is used for serial interface.

The f_{AD} clock is produced from the main clock or the on-chip oscillator clock and is used for the A/D converter.

When the WAIT instruction is executed after setting the CM02 bit in the CM0 register to "1" (peripheral function clock turned off during wait mode), the clocks fi, f_{jSIO}, and f_{AD} are turned off.

6.3.3 f_{RING} and $f_{RING128}$

These are operating clocks for the peripheral functions.

The f_{RING} runs at the same frequency as the on-chip oscillator, and can be used as the souce for the timer Y. The $f_{RING128}$ is derived from the f_{RING} by dividing it by 128, and can be used for Timer C.

When the WAIT instruction is executed, the clocks f_{RING} and $f_{RING128}$ are not turned off.

6.3.4 f_{RING}-fast

This is used as the count source for the timer C. The f_{RING}-fast is derived from the high-speed on-chip oscillator and provided by setting the HR00 bit to "1" (high-speed on-chip oscillator on).

When the WAIT instruction is executed, the clock f_{RING}-fast is not turned off.

6.4 Power Control

There are three power control modes. All modes other than wait and stop modes are referred to as normal operation mode.

6.4.1 Normal Operation Mode

Normal operation mode is further classified into four modes.

In normal operation mode, because the CPU clock and the peripheral function clocks both are on, the CPU and the peripheral functions are operating. Power control is exercised by controlling the CPU clock frequency. The higher the CPU clock frequency, the greater the processing capability. The lower the CPU clock frequency, the smaller the power consumption in the chip. If the unnecessary oscillator circuits are turned off, the power consumption is further reduced.

Before the clock sources for the CPU clock can be switched over, the new clock source to which switched must be oscillating stably. If the new clock source is the main clock, allow a sufficient wait time in a program until it becomes oscillating stably.

- **High-speed Mode**

  The main clock divided by 1 (undivided) provides the CPU clock. If the CM14 bit is set to "0" (low-speed on-chip oscillator on) or the HR00 bit in the HR0 register is set to "1" (high-speed on-chip oscillator on), the fRING and fRING128 can be used for timers Y and C. When the HR00 bit is set to "1", fRING-fast can be used for timer C.

- **Medium-speed Mode**

  The main clock divided by 2, 4, 8 or 16 provides the CPU clock. If the CM14 bit is set to "0" (low-speed on-chip oscillator on) or the HR00 bit in the HR0 register is set to "1" (high-speed on-chip oscillator on), the fRING and fRING128 can be used for timers Y and C. When the HR00 bit is set to "1", fRING-fast can be used for timer C.

- **High-speed, Low-speed, On-Chip Oscillator Mode**

  The on-chip oscillator clock divided by 1 (undivided), 2, 4, 8 or 16 provides the CPU clock. The on-chip oscillator clock is also the clock source for the peripheral function clocks. When the HR00 bit is set to "1", fRING-fast can be used for timer C.

Table 6.2 Setting Clock Related Bit and Modes

Modes		OCD register	CM1 register	CM0 register	
		OCD2	CM17, CM16	CM06	CM05
High-speed mode		0	00_2	0	0
Medium-speed mode	divided by 2	0	01_2	0	0
	divided by 4	0	10_2	0	0
	divided by 8	0	—	1	0
	divided by 16	0	11_2	0	0
High-speed, low-speed on-chip oscillator mode[1]	no division	1	00_2	0	0 or 1
	divided by 2	1	01_2	0	0 or 1
	divided by 4	1	10_2	0	0 or 1
	divided by 8	1	—	1	0 or 1
	divided by 16	1	11_2	0	0 or 1

Notes:
1. The low-speed on-chip oscillator is used as the on-chip oscillator clock when the CM1 register CM14 bit=0 (low-speed on-chip oscillator on) and HR0 register HR01 bit=0 (low-speed on-chip oscillator selected).
The high-speed on-chip oscillator is used as the on-chip oscillator clock when the HR0 register HR00 bit=1 (high-speed on-chip oscillator on) and HR01 bit=1 (high-speed on-chip oscillator selected).

6.4.2 Wait Mode

In wait mode, the CPU clock is turned off, so are the CPU and the watchdog timer because both are operated by the CPU clock. Because the main clock and on-chip oscillator clock both are on, the peripheral functions using these clocks keep operating.

- **Peripheral Function Clock Stop Function**

  If the CM02 bit is "1" (peripheral function clocks turned off during wait mode), the f_1, f_2, f_8, f_{32}, f_{1SIO}, f_{8SIO}, f_{32SIO}, and f_{AD} clocks are turned off when in wait mode, with the power consumption reduced that much.

- **Entering Wait Mode**

  The microcomputer is placed into wait mode by executing the WAIT instruction.

- **Pin Status During Wait Mode**

  The status before wait mode is retained.

- **Exiting Wait Mode**

  The microcomputer is moved out of wait mode by a hardware reset or peripheral function interrupt.

  When using a hardware reset to exit wait mode, set the ILVL2 to ILVL0 bits for the peripheral function interrupts to "0002" (interrupts disabled) before executing the WAIT instruction.

  The peripheral function interrupts are affected by the CM02 bit. If CM02 bit is "0" (peripheral function clocks not turned off during wait mode), all peripheral function interrupts can be used to exit wait mode. If CM02 bit is "1" (peripheral function clocks turned off during wait mode), the peripheral functions using the peripheral function clocks stop operating, so that only the peripheral functions clocked by external signals can be used to exit from wait mode.

  Table 6. 3 lists the interrupts to exit wait mode and the usage conditions.

  When using a peripheral function interrupt to exit wait mode, set up the following before executing the WAIT instruction.

  1. In the ILVL2 to ILVL0 bits in the interrupt control register, set the interrupt priority level of the peripheral function interrupt to be used to exit wait mode.
     Also, for all of the peripheral function interrupts not used to exit wait mode, set the ILVL2 to ILVL0 bits to "0002" (interrupt disable).
  2. Set the I flag to "1".
  3. Enable the peripheral function whose interrupt is to be used to exit wait mode.
     In this case, when an interrupt request is generated and the CPU clock is thereby turned on, an interrupt sequence is executed.

  The CPU clock turned on when exiting wait mode by a peripheral function interrupt is the same CPU clock that was on when the WAIT instruction was executed.

Table 6.3 Interrupts to Exit Wait Mode and Usage Conditions

Interrupt	CM02=0	CM02=1
Serial interface interrupt	Can be used when operating with internal or external clock	Can be used when operating with external clock
Key input interrupt	Can be used	Can be used
A/D conversion interrupt	Can be used in one-shot mode	— (Do not use)
Timer X interrupt	Can be used in all modes	Can be used in event counter mode
Timer Y interrupt	Can be used in all modes	Can be used when counting inputs from CNTR1 pin in timer mode
Timer Z interrupt	Can be used in all modes	(Do not use)
Timer C interrupt	Can be used in all modes	(Do not use)
$\overline{INT}$ interrupt	Can be used	Can be used ($\overline{INT0}$ and $\overline{INT3}$ can be used if there is no filter.
Voltage detection interrupt	Can be used	Can be used
Oscillation stop detection interrupt	Can be used	— (Do not use)

6.4.3 Stop Mode

In stop mode, all oscillator circuits are turned off, so are the CPU clock and the peripheral function clocks. Therefore, the CPU and the peripheral functions clocked by these clocks stop operating. The least amount of power is consumed in this mode. If the voltage applied to Vcc pin is VRAM or more, the internal RAM is retained.

However, the peripheral functions clocked by external signals keep operating. The following interrupts can be used to exit stop mode.

- Key interrupt
- $\overline{\text{INT0}}$ to $\overline{\text{INT2}}$ interrupts ($\overline{\text{INT0}}$ can be used only when there is no filter.)
- INT3 interrupt (INT3 can be used when there is no filter and Timer C output compare mode (the TCC13 bit in the TCC1 register is set to "1")
- Timer X interrupt (when counting external pulses in event counter mode)
- Timer Y interrupt (when counting inputs from CNTR1 pin in timer mode)
- Serial interfaces interrupt (when external clock is selected)
- Voltage detection interrupt

- **Entering Stop Mode**

The microcomputer is placed into stop mode by setting the CM10 bit of CM1 register to "1" (all clocks turned off). At the same time, the CM06 bit of CM0 register is set to "1" (divide-by-8 mode) and the CM15 bit of CM10 register is set to "1" (main clock oscillator circuit drive capability high).

Before entering stop mode, set the OCD1 to OCD0 bits to "002" (oscillation stop detection function disable).

- **Pin Status in Stop Mode**

The status before wait mode is retained.

However, the XOUT(P47) pin is held "H" when the CM13 bit in the CM1 register is set to "1" (XIN-XOUT pin). The P47(XOUT) is in input state when the CM13 bit is set to "0" (input port P46, P47).

- **Exiting Stop Mode**

The microcomputer is moved out of stop mode by a hardware reset or peripheral function interrupt.

When using a hardware reset to exit stop mode, set the ILVL2 to ILVL0 bits for the peripheral function interrupts to "0002" (interrupts disabled) before setting the CM10 bit to "1".

When using a peripheral function interrupt to exit stop mode, set up the following before setting the CM10 bit to "1".

1. In the ILVL2 to ILVL0 bits in the interrupt control register, set the interrupt priority level of the peripheral function interrupt to be used to exit stop mode.
   Also, for all of the peripheral function interrupts not used to exit stop mode, set the ILVL2 to ILVL0 bits to "0002".
2. Set the I flag to "1".
3. Enable the peripheral function whose interrupt is to be used to exit stop mode.
   In this case, when an interrupt request is generated and the CPU clock is thereby turned on, an interrupt sequence is executed.

The main clock divided by 8 of the clock which is used right before stop mode is used for the CPU clock when exiting stop mode by a peripheral function interrupt.

Figure 6.6 shows the state transition of power control.

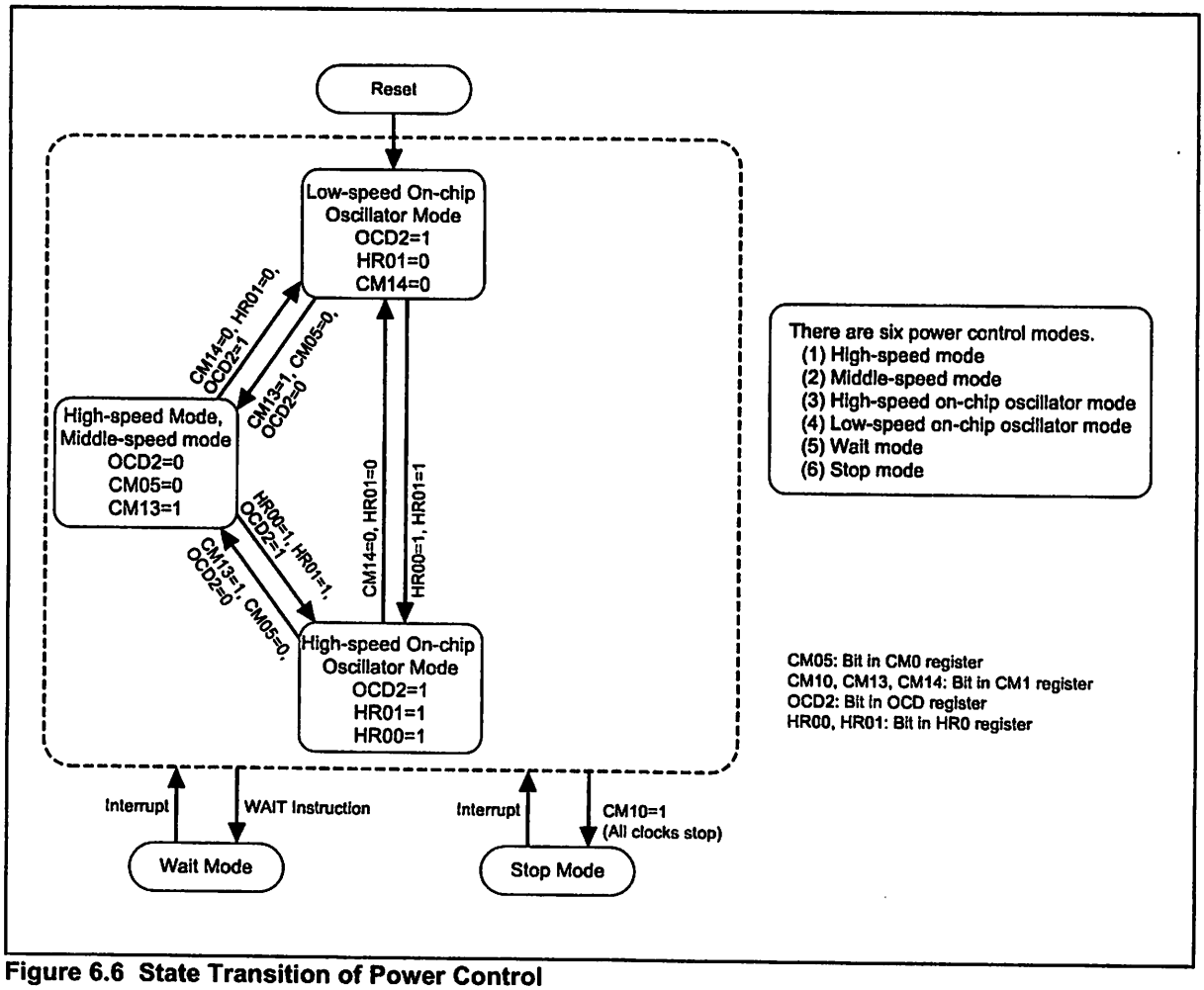


Figure 6.6 State Transition of Power Control

6.5 Oscillation Stop Detection Function

The oscillation stop detection function is such that main clock oscillation circuit stop is detected. The oscillation stop detection function can be enabled and disabled by the OCD1 to OCD0 bits in the OCD register.

Table 6.4 lists the specifications of the oscillation stop detection function.

Where the main clock corresponds to the CPU clock source and the OCD1 to OCD0 bits are "11$_2$" (oscillation stop detection function enabled), the system is placed in the following state if the main clock comes to a halt:

- OCD register OCD2 bit = 1 (selecting on-chip oscillator clock)
- OCD register OCD3 bit = 1 (main clock stopped)
- CM1 register CM14 bit = 0 (low-speed on-chip oscillator oscillating)
- Oscillation stop detection interrupt request occurs

Table 6.4 Oscillation Stop Detection Function Specifications

Item	Specification
Oscillation stop detectable clock and frequency bandwidth	f(X_{IN}) ≥ 2 MHz
Enabling condition for oscillation stop detection function	Set OCD1 to OCD0 bits to "11$_2$" (oscillation stop detection function enabled)
Operation at oscillation stop detection	Oscillation stop detection interrupt occurs

6.5.1 How to Use Oscillation Stop Detection Function

- The oscillation stop detection interrupt shares the vector with the watchdog timer interrupt. If the oscillation stop detection and watchdog timer interrupts both are used, the interrupt factor must be determined. Table 6.5 shows how to determine the interrupt factor with the oscillation stop detection interrupt, watchdog timer interrupt and voltage detection interrupt.
- Where the main clock re-oscillated after oscillation stop, the clock source for the CPU clock and peripheral functions must be switched to the main clock in the program.
  Figure 6.7 shows the procedure for switching the clock source from the low-speed on-chip oscillator to the main clock.
- To enter wait mode while using the oscillation stop detection function, set the CM02 bit to "0" (peripheral function clocks not turned off during wait mode).
- Since the oscillation stop detection function is provided in preparation for main clock stop due to external factors, set the OCD1 to OCD0 bits to "002" (oscillation stop detection function disabled) where the main clock is stopped or oscillated in the program, that is where the stop mode is selected or the CM05 bit is altered.
- When using the low-speed on-chip oscillator clock for the CPU clock and clock sources of peripheral functions after detecting the oscillation stop, set the HR01 bit in the HR0 register to "0" (low-speed on-chip oscillator selected) and the OCD1 to OCD0 bits to "112" (oscillation stop detection function enabled). When using the high-speed on-chip oscillator clock for the CPU clock and clock sources of peripheral functions after detecting the oscillation stop, set the HR01 bit to "1" (high-speed on-chip oscillator selected) and the OCD1 to OCD0 bits to "112" (oscillation stop detection function enabled).

Table 6.5 Interrupt Factor Determination of Oscillation Stop Detection, Watchdog Timer Interrupt or Voltage Detection Interrupt

Generated Interrupt Factor	Bit showing interrupt source
Oscillation stop detection ((a) or (b))	(a) The OCD3 bit in the OCD register = 1
	(b) The OCD1 to OCD0 bits in the OCD register = 11_2 and the OCD2 bit = 1
Watchdog timer	The D43 bit in the D4INT register = 1
Voltage detection	The D42 bit in the D4INT register = 1

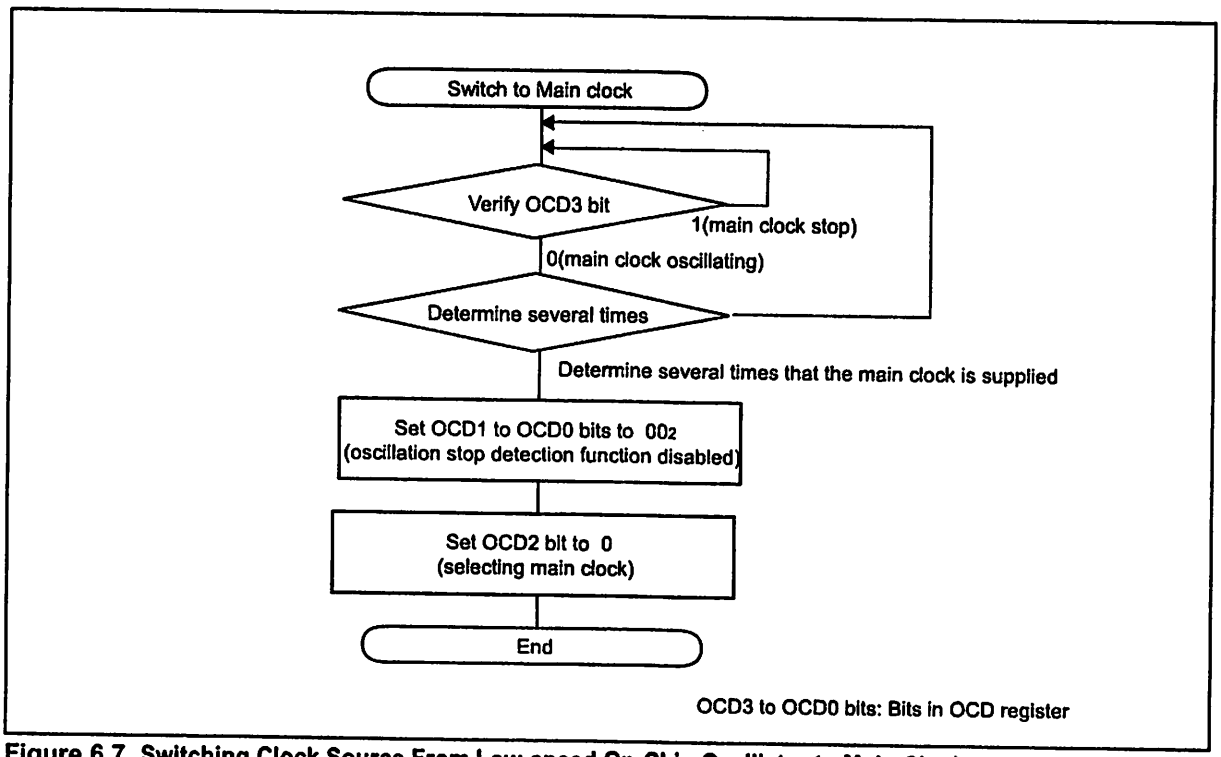


Figure 6.7 Switching Clock Source From Low-speed On-Chip Oscillator to Main Clock

7. Protection

In the event that a program runs out of control, this function protects the important registers so that they will not be rewritten easily. Figure 7.1 shows the PRCR register. The following lists the registers protected by the PRCR register.

- Registers protected by PRC0 bit: CM0, CM1, and OCD, HR0, HR1 registers
- Registers protected by PRC1 bit: PM0 and PM1 registers
- Registers protected by PRC2 bit: PD0 register
- Registers protected by PRC3 bit: VCR2 and D4INT registers

Set the PRC2 bit to "1" (write enabled) and then write to any address, and the PRC2 bit will be set to "0" (write protected). The registers protected by the PRC2 bit should be changed in the next instruction after setting the PRC2 bit to "1". Make sure no interrupts will occur between the instruction in which the PRC2 bit is set to "1" and the next instruction. The PRC0,PRC1 and PRC3 bsits are not automatically set to "0" by writing to any address. They can only be set to "0" in a program.

Protect register

b7	b6	b5	b4	b3	b2	b1	b0
		0	0				

Symbol	Address	After reset
PRCR	000A16	00XXX0002

Bit symbol	Bit name	Function	RW
PRC0	Protect bit 0	Enable write to CM0, CM1, OCD, HR0, HR1 registers 0 : Write protected 1 : Write enabled	RW
PRC1	Protect bit 1	Enable write to PM0, PM1 registers 0 : Write protected 1 : Write enabled	RW
PRC2	Protect bit 2	Enable write to PD0 register 0 : Write protected 1 : Write enabled[1]	RW
PRC3	Protect bit 3	Enable write to VCR2, D4INT registers 0 : Write protected 1 : Write enabled	RW
$\overline{\text{(b5-b4)}}$	Reserved bit	When write, should set to "0"	RW
$\overline{\text{(b7-b6)}}$	Reserved bit	When read, its content is "0".	RO

Notes:
1. The PRC2 bit is set to "0" by writing to any address after setting it to "1". Other bits are not set to "0" by writing to any address, and must therefore be set to "0" in a program.

Figure 7.1 PRCR Register

8. Processor Mode

8.1 Types of Processor Mode

The processor mode is single-chip mode. Table 8.1 shows the features of the processor mode. Figure 8.1 shows the PM0 and PM1 register.

Table 8.1 Features of Processor Mode

Processor mode	Access space	Pins which are assigned I/O ports
Single-chip mode	SFR, internal RAM, internal ROM	All pins are I/O ports or peripheral function I/O pins

Processor mode register 0(1)

b7	b6	b5	b4	b3	b2	b1	b0
X	X	X	X		0	0	0

Symbol: PM0 Address: 000416 After reset: 0016

Bit symbol	Bit name	Function	RW
— (b2-b0)	Reserved bit	Set to "0"	RW
PM03	Software reset bit	Setting this bit to "1" resets the microcomputer. When read, its content is "0".	RW
— (b7-b4)	Nothing is assigned. When write, set to "0". When read, its content is "0".		—

NOTES:
1. Set the PRC1 bit in the PRCR register to "1" (write enable) before writing to this register.

Processor mode register 1(1)

b7	b6	b5	b4	b3	b2	b1	b0
0	X	X	X	X		0	

Symbol: PM1 Address: 000516 After reset: 0016

Bit symbol	Bit name	Function	RW
PM10	Data area access enable bit	0 : Disabled 1 : Enabled	RW
— (b1)	Reserved bit	Set to "0"	RW
PM12	WDT interrupt/reset switch bit(2)	0 : Watchdog timer interrupt 1 : Watchdog timer reset2	RW
— (b6-b3)	Nothing is assigned. When write, set to "0". When read, its content is 0.		—
— (b7)	Reserved bit	Set to "0"	RW

Notes:
1. Set the PRC1 bit in the PRCR register to "1" (write enable) before writing to this register.
2. PM12 bit is set to "1" by writing a "1" in a program. (Writing a "0" has no effect.)

Figure 8.1 PM0 Register and PM1 Register

9. Bus

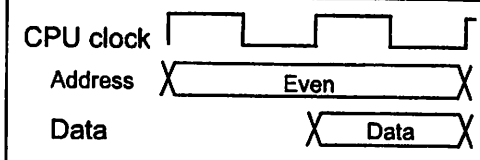
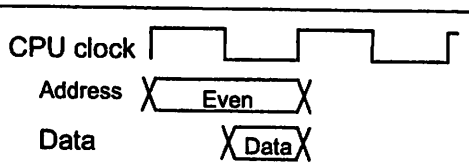
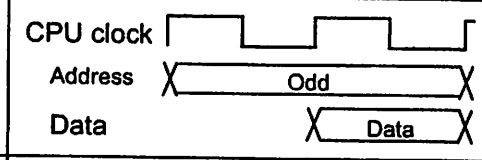
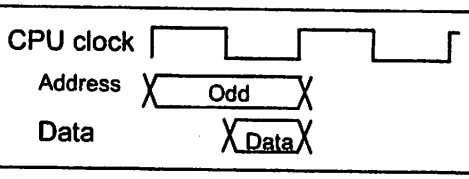
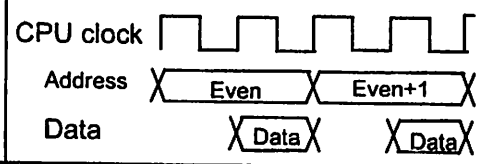
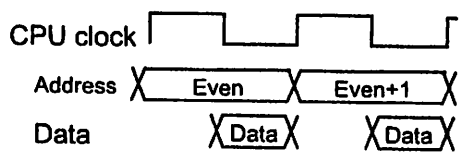
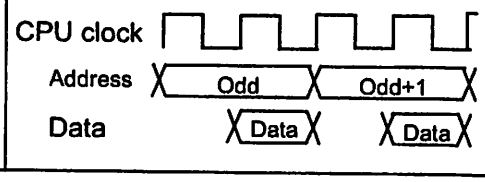
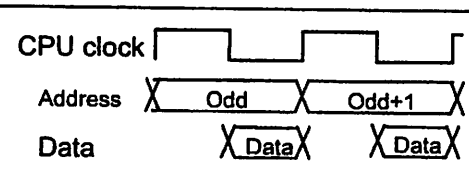
During access, the ROM/RAM and the SFR have different bus cycles. Table 9.1 shows bus cycles for access space.

The ROM/RAM and SFR are connected to the CPU through an 8-bit bus. When accessing in word (16 bits) units, these spaces are accessed twice in 8-bit units. Table 9.2 shows bus cycles in each access space.

Table 9.1 Bus Cycles for Access Space

Access space	Bus cycle
SFR	2 CPU clock cycles
ROM/RAM	1 CPU clock cycles

Table 9.2 Access Unit and Bus Operation

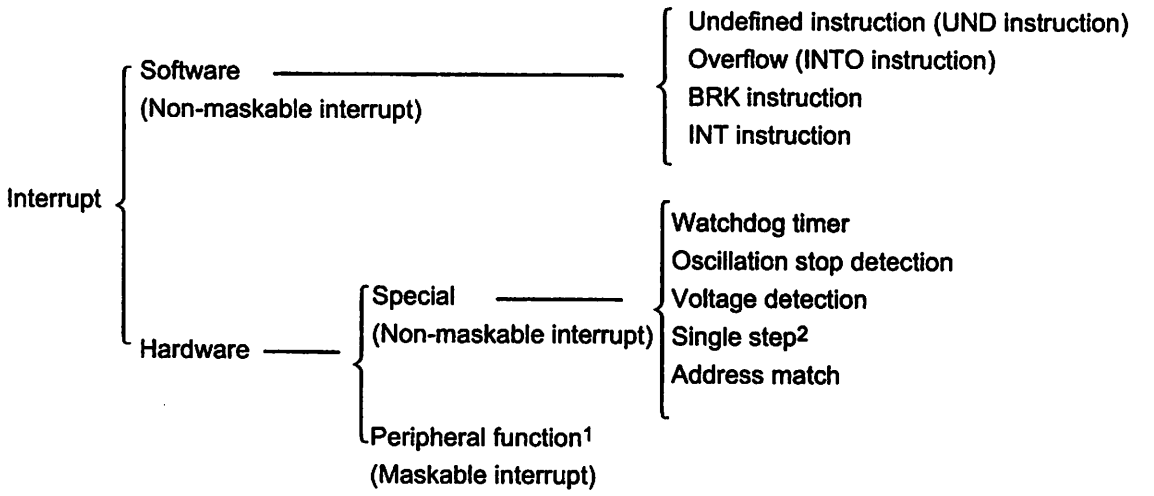
Space	SFR	ROM/RAM
Even address byte access		
Odd address byte access		
Even address word access		
Odd address word access		

10. Interrupt

10.1 Interrupt Overview

10.1.1 Type of Interrupts

Figure 10.1 shows types of interrupts.



Notes:
1. Peripheral function interrupts are generated by the peripheral functions built in the microcomputer system.
2. Avoid using this interrupt because this is a dedicated interrupt for development support tools only.

Figure 10.1 Interrupts

- Maskable Interrupt: An interrupt which can be enabled (disabled) by the interrupt enable flag (I flag) or whose interrupt priority **can be changed** by priority level.
- Non-maskable Interrupt: An interrupt which cannot be enabled (disabled) by the interrupt enable flag (I flag) or whose interrupt priority **cannot be changed** by priority level.

10.1.2 Software Interrupts

A software interrupt occurs when executing certain instructions. Software interrupts are non-maskable interrupts.

- **Undefined Instruction Interrupt**

  An undefined instruction interrupt occurs when executing the UND instruction.
- **Overflow Interrupt**

  An overflow interrupt occurs when executing the INTO instruction with the O flag set to "1" (the operation resulted in an overflow). The following are instructions whose O flag changes by arithmetic:

  ABS, ADC, ADCF, ADD, CMP, DIV, DIVU, DIVX, NEG, RMPA, SBB, SHA, SUB
- **BRK Interrupt**

  A BRK interrupt occurs when executing the BRK instruction.
- **INT Instruction Interrupt**

  An INT instruction interrupt occurs when executing the INT instruction. Software interrupt numbers 0 to 63 can be specified for the INT instruction. Because software interrupt Nos. 4 to 31 are assigned to peripheral function interrupts, the same interrupt routine as for peripheral function interrupts can be executed by executing the INT instruction.

  In software interrupt numbers 0 to 31, the U flag is saved to the stack during instruction execution and is cleared to "0" (ISP selected) before executing an interrupt sequence. The U flag is restored from the stack when returning from the interrupt routine. In software interrupt numbers 32 to 63, the U flag does not change state during instruction execution, and the SP then selected is used.

10.1.3 Hardware Interrupts

Hardware interrupts are classified into two types — special interrupts and peripheral function interrupts.

(1) Special Interrupts

Special interrupts are non-maskable interrupts.

- **Watchdog Timer Interrupt**

  Generated by the watchdog timer. Once a watchdog timer interrupt is generated, be sure to initialize the watchdog timer. For details about the watchdog timer, refer to Chapter 11, "Watchdog Timer."

- **Oscillation Stop Detection Interrupt**

  Generated by the oscillation stop detection function. For details about the oscillation stop detection function, refer to Chapter 6, "Clock Generation Circuit."

- **Voltage Detection Interrupt**

  Generated by the voltage detection circuit. For details about the voltage detection circuit, refer to Section 5.4, "Voltage Detection Circuit."

- **Single-step Interrupt**

  Do not normally use this interrupt because it is provided exclusively for use by development support tools.

- **Address Match Interrupt**

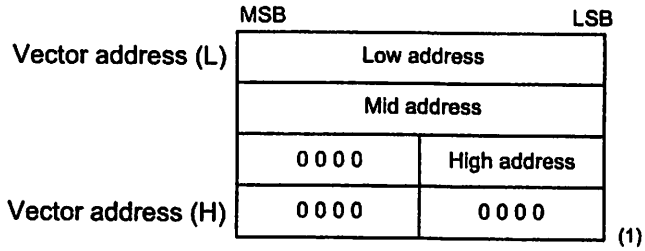
  An address match interrupt is generated immediately before executing the instruction at the address indicated by the RMAD0 to RMAD1 register that corresponds to one of the AIER register's AIER0 or AIER1 bit which is "1" (address match interrupt enabled). For details about the address match interrupt, refer to Section 10.4, "Address Match Interrupt."

(2) Peripheral Function Interrupts

Peripheral function interrupts are maskable interrupts and generated by the microcomputer's internal functions. The interrupt factors for peripheral function interrupts are listed in Table 10.2. "Relocatable Vector Tables". For details about the peripheral functions, refer to the description of each peripheral function in this manual.

10.1.4 Interrupts and Interrupt Vector

One interrupt vector consists of 4 bytes. Set the start address of each interrupt routine in the respective interrupt vectors. When an interrupt request is accepted, the CPU branches to the address set in the corresponding interrupt vector. Figure 10.2 shows the interrupt vector.



Note1: The OFS register is assigned to the 0FFFF16 address. Refer to "Figure11.2 OFS, WDC, WDTR and WDTS registers" for the OFS register details.

Figure 10.2 Interrupt Vector

- **Fixed Vector Tables**

The fixed vector tables are allocated to the addresses from 0FFDC16 to 0FFFF16. Table 10.1 lists the fixed vector tables. In the flash memory version of microcomputer, the vector addresses (H) of fixed vectors are used by the ID code check function. For details, refer to Section 17.3, "Functions to Prevent Flash Memory from Rewriting."

Table 10.1 Fixed Vector Tables

Interrupt factor	Vector addresses Address (L) to address (H)	Remarks	Reference
Undefined instruction	0FFDC16 to 0FFDF16	Interrupt on UND instruction	R8C/Tiny Series software manual
Overflow	0FFE016 to 0FFE316	Interrupt on INTO instruction	
BRK instruction	0FFE416 to 0FFE716	If the contents of address 0FFE716 is FF16, program execution starts from the address shown by the vector in the relocatable vector table.	
Address match	0FFE816 to 0FFEB16		18.1 Address match interrupt
Single step[1]	0FFEC16 to 0FFEF16		
• Watchdog timer • Oscillation stop detection • Voltage detection	0FFF016 to 0FFF316		11. Watchdog timer 6. Clock generation circuit 5.4Voltage detection circuit
(Reserved)	0FFF416 to 0FFF716		
(Reserved)	0FFF816 to 0FFFB16		
Reset	0FFFC16 to 0FFFF16		Reset

Note: Do not normally use this interrupt because it is provided exclusively for use by development support tools.

• **Relocatable Vector Tables**

The 256 bytes beginning with the start address set in the INTB register comprise a reloacatable vector table area. Table 10.2 lists interrupts and vector tables located in the relocatable vector table.

Table 10.2 Relocatable Vector Tables

Interrupt factor	Vector address[1] Address (L) to address (H)	Software interrupt number	Reference
BRK instruction[2]	+0 to +3 (000016 to 000316)	0	R8C/Tiny Series software manual
——— (Reserved)		1 to 12	
Key input	+52 to +55 (003416 to 003716)	13	10.3 Key input interrupt
A/D	+56 to +59 (003816 to 003B16)	14	14 A/D converter
——— (Reserved)		15	
Compare 1	+64 to +67 (004016 to 004316)	16	12.4 Timer C
UART0 transmit	+68 to +71 (004416 to 004716)	17	13. Serial Interface
UART0 receive	+72 to +75 (004816 to 004B16)	18	
UART1 transmit	+76 to +79 (004C16 to 004F16)	19	
UART1 receive	+80 to +83 (005016 to 005316)	20	
$\overline{\text{INT2}}$	+84 to +87 (005416 to 005716)	21	10.2.3 $\overline{\text{INT}}$ interrupt
Timer X	+88 to +91 (005816 to 005B16)	22	12.1 Timer X
Timer Y	+92 to +95 (005C16 to 005F16)	23	12.2 Timer Y
Timer Z	+96 to +99 (006016 to 006316)	24	12.3 Timer Z
$\overline{\text{INT1}}$	+100 to +103 (006416 to 006716)	25	10.2.3 $\overline{\text{INT}}$1 interrupt
$\overline{\text{INT3}}$	+104 to +107 (006816 to 006B16)	26	10.2.4 $\overline{\text{INT3}}$ interrupt
Timer C	+108 to +111 (006C16 to 006F16)	27	12.4 Timer C
Compare 0	+112 to +115 (007016 to 007316)	28	12.4 Timer C
$\overline{\text{INT0}}$	+116 to +119 (007416 to 007716)	29	10.2.1 $\overline{\text{INT0}}$ interrupt
——— (Reserved)		30	
——— (Reserved)		31	
Software interrupt[2]	+128 to +131 (008016 to 008316) to +252 to +255 (00FC16 to 00FF16)	32 to 63	R8C/Tiny Series software manual

Notes:
1. Address relative to address in INTB.
2. These interrupts cannot be disabled using the I flag.

10.1.5 Interrupt Control

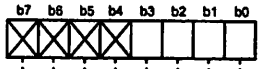
The following describes how to enable/disable the maskable interrupts, and how to set the priority in which order they are accepted. What is explained here does not apply to nonmaskable interrupts.

Use the FLG register's I flag, IPL, and each interrupt control register's ILVL2 to ILVL0 bits to enable/disable the maskable interrupts. Whether an interrupt is requested is indicated by the IR bit in each interrupt control register.

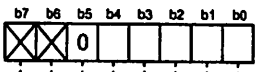
Figure 10.3 shows the interrupt control registers.

Interrupt control register[2]

Symbol	Address	After reset
KUPIC	004D16	XXXXX0002
ADIC	004E16	XXXXX0002
CMP1IC	005016	XXXXX0002
S0TIC, S1TIC	005116, 005316	XXXXX0002
S0RIC, S1RIC	005216, 005416	XXXXX0002
INT2IC	005516	XXXXX0002
TXIC	005616	XXXXX0002
TYIC	005716	XXXXX0002
TZIC	005816	XXXXX0002
INT1IC	005916	XXXXX0002
INT3IC	005A16	XXXXX0002
TCIC	005B16	XXXXX0002
CMP0IC	005C16	XXXXX0002



Bit symbol	Bit name	Function	RW
ILVL0	Interrupt priority level select bit	b2 b1 b0 0 0 0 : Level 0 (interrupt disabled) 0 0 1 : Level 1 0 1 0 : Level 2 0 1 1 : Level 3 1 0 0 : Level 4 1 0 1 : Level 5 1 1 0 : Level 6 1 1 1 : Level 7	RW
ILVL1			RW
ILVL2			RW
IR	Interrupt request bit	0 : Interrupt not requested 1 : Interrupt requested	RW[1]
— (b7-b4)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		—



Symbol	Address	After reset
INT0IC	005D16	XX00X0002

Bit symbol	Bit name	Function	RW
ILVL0	Interrupt priority level select bit	b2 b1 b0 0 0 0 : Level 0 (interrupt disabled) 0 0 1 : Level 1 0 1 0 : Level 2 0 1 1 : Level 3 1 0 0 : Level 4 1 0 1 : Level 5 1 1 0 : Level 6 1 1 1 : Level 7	RW
ILVL1			RW
ILVL2			RW
IR	Interrupt request bit	0: Interrupt not requested 1: Interrupt requested	RW[1]
POL	Polarity select bit[3, 4]	0 : Selects falling edge 1 : Selects rising edge	RW
(b5)	Reserved bit	Must always be set to "0"	RW
— (b7-b6)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		—

Notes:
1. Only "0" can be written to the IR bit. (Do not write "1").
2. To rewrite the interrupt control register, do so at a point that does not generate the interrupt request for that register. Refer to the paragraph 19.2.6 "Changing Interrupt Control Registers".
3. If the INTOPL bit in the INTEN register is set to "1" (both edges), set the POL bit to "0 " (selecting falling edge).
4. The IR bit may be set to "1" (interrupt requested) when the POL bit is rewritten. Refer to the paragraph 19.2.5 "Changing Interrupt Factor".

Figure 10.3 Interrupt Control Registers

• **I Flag**

The I flag enables or disables the maskable interrupt. Setting the I flag to "1" (enabled) enables the maskable interrupt. Setting the I flag to "0" (disabled) disables all maskable interrupts.

• **IR Bit**

The IR bit is set to "1" (interrupt requested) when an interrupt request is generated. Then, when the interrupt request is accepted and the CPU branches to the corresponding interrupt vector, the IR bit is cleared to "0" (= interrupt not requested).
The IR bit can be cleared to "0" in a program. Note that do not write "1" to this bit.

• **ILVL2 to ILVL0 Bits and IPL**

Interrupt priority levels can be set using the ILVL2 to ILVL0 bits.
Table 10.3 shows the settings of interrupt priority levels and Table 10.4 shows the interrupt priority levels enabled by the IPL.

The following are conditions under which an interrupt is accepted:

- I flag = 1
- IR bit = 1
- interrupt priority level > IPL

The I flag, IR bit, ILVL2 to ILVL0 bits and IPL are independent of each other. In no case do they affect one another.

Table 10.3 Settings of Interrupt Priority Levels

ILVL2 to ILVL0 bits	Interrupt priority level	Priority order
000_2	Level 0 (interrupt disabled)	
001_2	Level 1	Lowest
010_2	Level 2	
011_2	Level 3	
100_2	Level 4	
101_2	Level 5	
110_2	Level 6	
111_2	Level 7	Highest

Table 10.4 Interrupt Priority Levels Enabled by IPL

IPL	Enabled interrupt priority levels
000_2	Interrupt levels 1 and above are enabled
001_2	Interrupt levels 2 and above are enabled
010_2	Interrupt levels 3 and above are enabled
011_2	Interrupt levels 4 and above are enabled
100_2	Interrupt levels 5 and above are enabled
101_2	Interrupt levels 6 and above are enabled
110_2	Interrupt levels 7 and above are enabled
111_2	All maskable interrupts are disabled

• **Interrupt Sequence**

An interrupt sequence — what are performed over a period from the instant an interrupt is accepted to the instant the interrupt routine is executed — is described here.

If an interrupt occurs during execution of an instruction, the processor determines its priority when the execution of the instruction is completed, and transfers control to the interrupt sequence from the next cycle. If an interrupt occurs during execution of either the SMOVB, SMOVF, SSTR or RMPA instruction, the processor temporarily suspends the instruction being executed, and transfers control to the interrupt sequence.

The CPU behavior during the interrupt sequence is described below. Figure 10.4 shows time required for executing the interrupt sequence.

(1) The CPU gets interrupt information (interrupt number and interrupt request priority level) by reading the address 0000016. Then it clears the IR bit for the corresponding interrupt to "0" (interrupt not requested).
(2) The FLG register immediately before entering the interrupt sequence is saved to the CPU internal temporary register(Note).
(3) The I, D and U flags in the FLG register become as follows:
The I flag is cleared to "0" (interrupts disabled).
The D flag is cleared to "0" (single-step interrupt disabled).
The U flag is cleared to "0" (ISP selected).
However, the U flag does not change state if an INT instruction for software interrupt numbers 32 to 63 is executed.
(4) The CPU's internal temporary register (Note) is saved to the stack.
(5) The PC is saved to the stack.
(6) The interrupt priority level of the accepted interrupt is set in the IPL.
(7) The start address of the relevant interrupt routine set in the interrupt vector is stored in the PC.

After the interrupt sequence is completed, the processor resumes executing instructions from the start address of the interrupt routine.

Note: This register cannot be used by user.

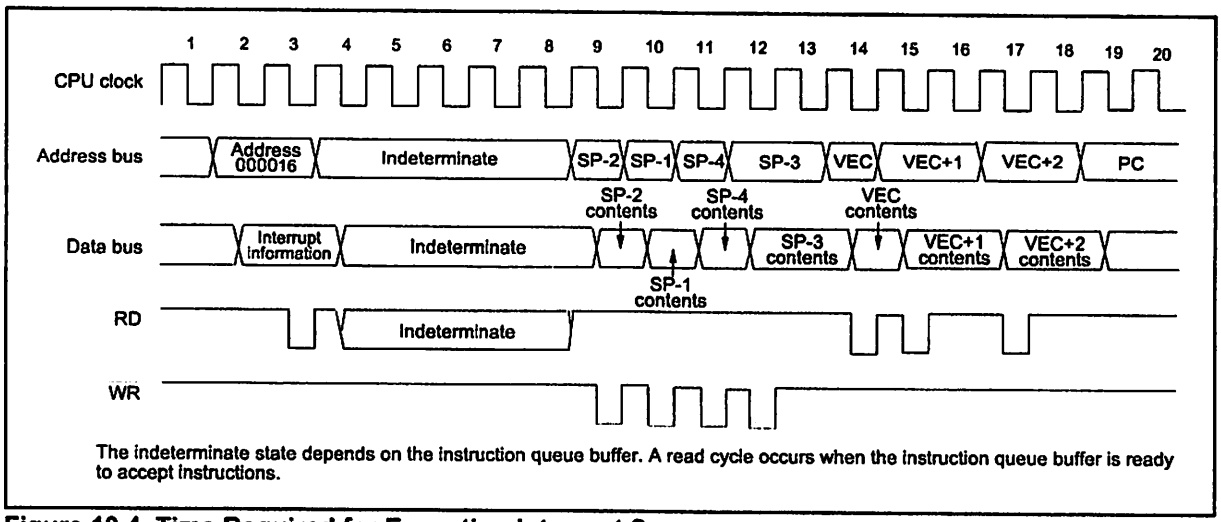
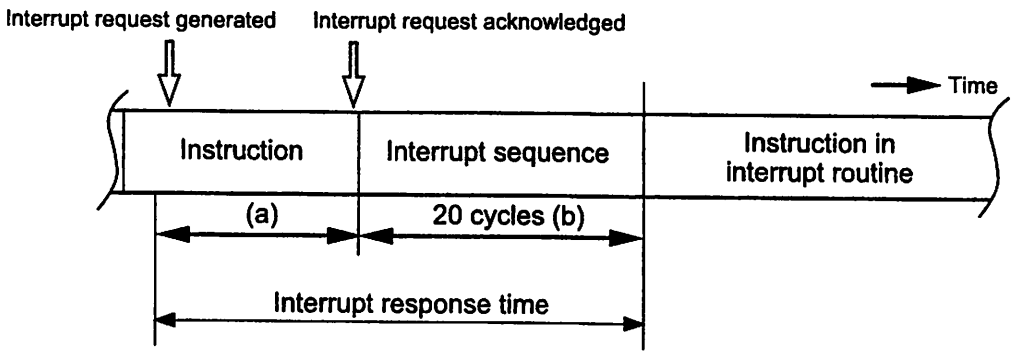


Figure 10.4 Time Required for Executing Interrupt Sequence

• **Interrupt Response Time**

Figure 10.5 shows the interrupt response time. The interrupt response or interrupt acknowledge time denotes a time from when an interrupt request is generated till when the first instruction in the interrupt routine is executed. Specifically, it consists of a time from when an interrupt request is generated till when the instruction then executing is completed (see #a in Figure 10.5) and a time during which the interrupt sequence is executed (20 cycles, see #b in Figure 10.5).



(a) A time from when an interrupt request is generated till when the instruction then executing is completed. The length of this time varies with the instruction being executed. The DIVX instruction requires the longest time, which is equal to 30 cycles (without wait state, the divisor being a register).
(b) 21 cycles for address match and single-step interrupts.

Figure 10.5 Interrupt Response Time

• **Variation of IPL when Interrupt Request is Accepted**

When a maskable interrupt request is accepted, the interrupt priority level of the accepted interrupt is set in the IPL.

When a software interrupt or special interrupt request is accepted, one of the interrupt priority levels listed in Table 10.5 is set in the IPL. Shown in Table 10.5 are the IPL values of software and special interrupts when they are accepted.

Table 10.5 IPL Level That Is Set to IPL When A Software or Special Interrupt Is Accepted

Interrupt factors	Level that is set to IPL
Watchdog timer, oscillation stop detection, voltage detection	7
Software, address match, single-step	Not changed

• **Saving Registers**

In the interrupt sequence, the FLG register and PC are saved to the stack.
At this time, the 4 high-order bits in the PC and the 4 high-order (IPL) and 8 low-order bits in the FLG register, 16 bits in total, are saved to the stack first. Next, the 16 low-order bits in the PC are saved.
Figure 10.6 shows the stack status before and after an interrupt request is accepted.
The other necessary registers must be saved in a program at the beginning of the interrupt routine.
Use the PUSHM instruction, and all registers except SP can be saved with a single instruction.

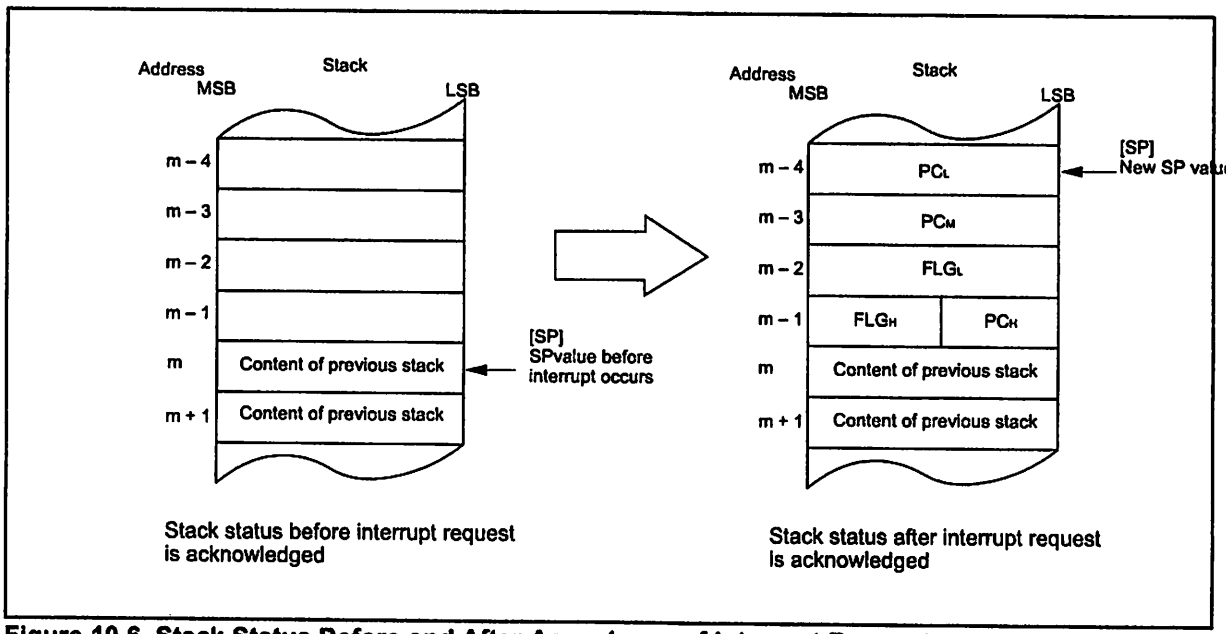


Figure 10.6 Stack Status Before and After Acceptance of Interrupt Request

The registers are saved in four steps, 8 bits at a time. Figure 10.7 shows the operation of the saving registers.

Note: When any INT instruction in software numbers 32 to 63 has been executed, this is the SP indicated by the U flag. Otherwise, it is the ISP.

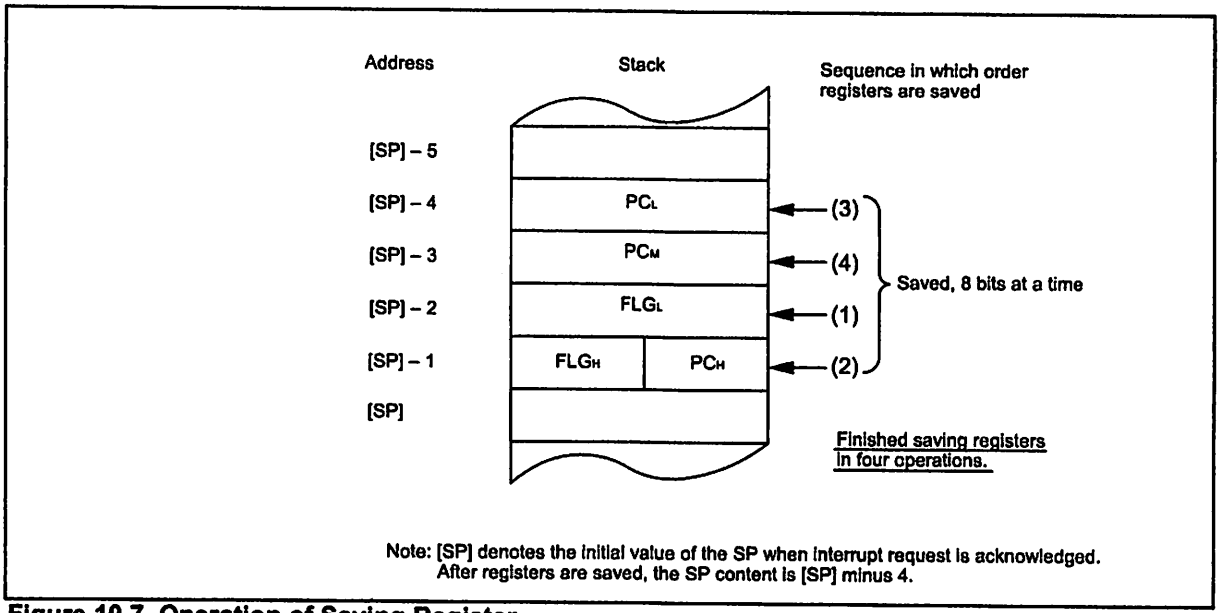


Figure 10.7 Operation of Saving Register

• **Returning from an Interrupt Routine**

The FLG register and PC in the state in which they were immediately before entering the interrupt sequence are restored from the stack by executing the REIT instruction at the end of the interrupt routine. Thereafter the CPU returns to the program which was being executed before accepting the interrupt request.

Return the other registers saved by a program within the interrupt routine using the POPM or similar instruction before executing the REIT instruction.

• **Interrupt Priority**

If two or more interrupt requests are generated while executing one instruction, the interrupt request that has the highest priority is accepted.

For maskable interrupts (peripheral functions), any desired priority level can be selected using the ILVL2 to ILVL0 bits. However, if two or more maskable interrupts have the same priority level, their interrupt priority is resolved by hardware, with the highest priority interrupt accepted.

The watchdog timer and other special interrupts have their priority levels set in hardware. Figure 10.8 shows the Hardware Interrupt Priority.

Software interrupts are not affected by the interrupt priority. If an instruction is executed, control branches invariably to the interrupt routine.

Reset > WDT/Oscillation stop detection/Voltage detection > Peripheral function > Single step > Address match

Figure 10.8 Hardware Interrupt Priority

• **Interrupt Priority Resolution Circuit**

The interrupt priority resolution circuit is used to select the interrupt with the highest priority among those requested.

Figure 10.9 shows the Interrupts Priority Select Circuit.

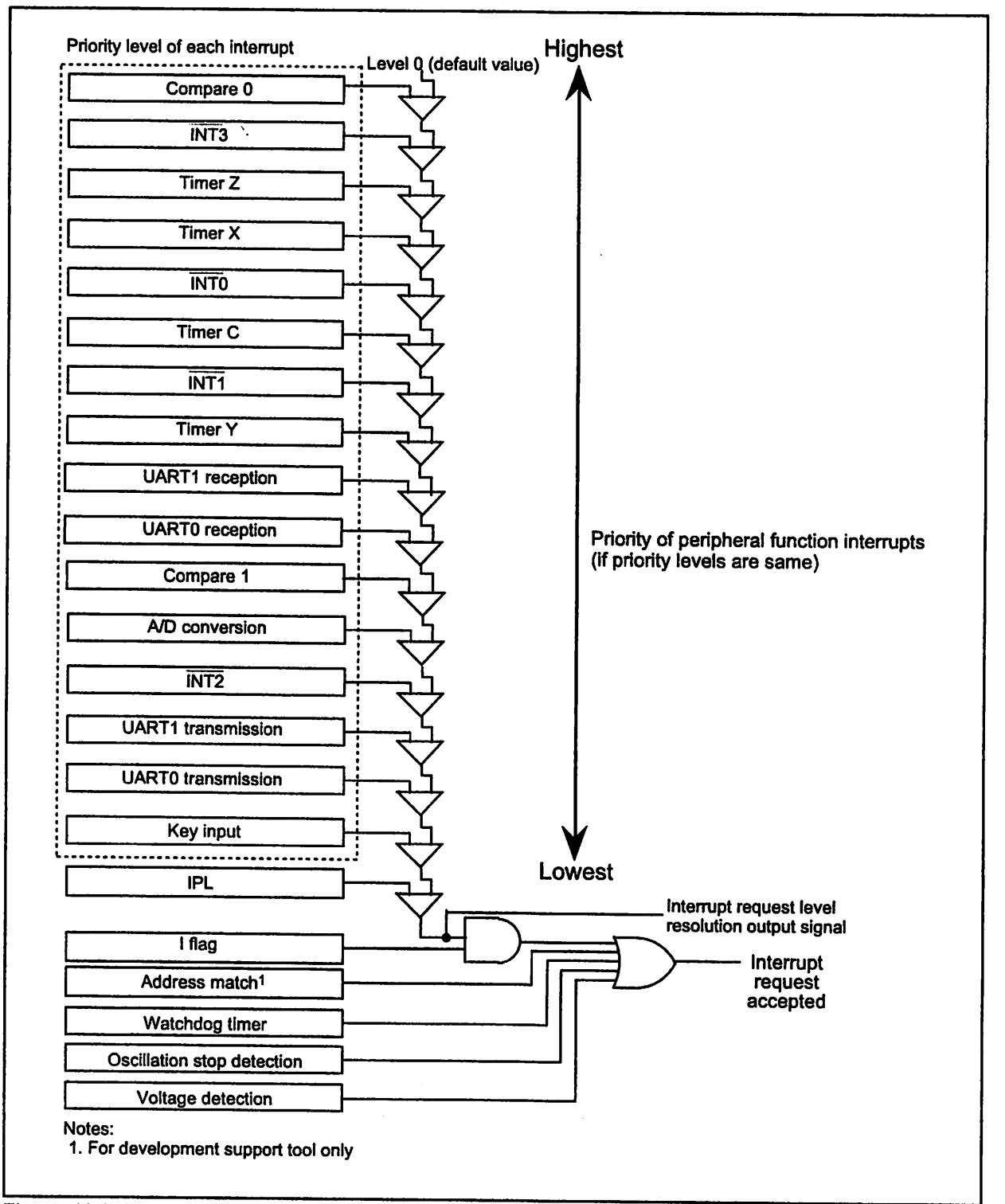


Figure 10.9 Interrupts Priority Select Circuit

10.2 INT Interrupt

10.2.1 INT0 Interrupt

INT0 interrupt is triggered by an INT0 input. When using INT0 interrupts, the INT0EN bit in the INTEN register must be set to "1" (enabling). The edge polarity is selected using the INT0PL bit in the INTEN register and the POL bit in the INT0IC register.

The INT0 pin is shared with the external trigger input pin of Timer Z.

Figure 10.10 shows the INTEN and INT0F registers.

External input enable register

b7	b6	b5	b4	b3	b2	b1	b0
0	0	0	0	0	0		

Symbol	Address	After reset
INTEN	0096_{16}	00_{16}

Bit symbol	Bit name	Function	RW
INT0EN	INT0 input enable bit[1]	0 : Disabled 1 : Enabled	RW
INT0PL	INT0 input polarity select bit[2]	0 : One edge 1 : Both edges	RW
— (b7-b2)	Reserved bit	Set to "0"	RW

Notes:
1. This bit must be set while the INT0STG bit in the PUM register is set to "0" (one-shot trigger disabled).
2. When setting the INT0PL bit to "1" (selecting both edges), the POL bit in the INT0IC must be set to "0" (selecting falling edge).
3. The IR bit in the INT0IC register may be set to "1" (interrupt requested) when the INT0PL bit is rewritten. Refer to the paragraph 19.2.5 "Changing Interrupt Factor" in the Usage Notes Reference Book.

INT0 input filter select register

b7	b6	b5	b4	b3	b2	b1	b0
X	X	X	X	X	0		

Symbol	Address	After reset
INT0F	$001E_{16}$	XXXXX0002

Bit symbol	Bit name	Function	RW
INT0F0	INT0 input filter select bit	b1 b0 0 0 : No filter 0 1 : Filter with f1 sampling 1 0 : Filter with f8 sampling 1 1 : Filter with f32 sampling	RW
INT0F1			RW
— (b2)	Reserved bit	Set to "0"	RW
— (b7-b3)	Nothing is assigned. When write, set to "0". If read, it content is indeterminate.		—

Figure 10.10 INTEN and INT0F Registers

10.2.2 INT0 Input Filter

The INT0 input has a digital filter which can be sampled by one of three sampling clocks. The sampling clock is selected using the INT0F1 to INT0F0 bits in the INT0F register. The IR bit in the INT0IC register is set to "1" (interrupt requested) when the sampled input level matches three times. When the INT0F1 to INT0F0 bits are set to "01$_2$", "10$_2$", or "11$_2$", the P4_5 bit in the P4 register indicates the filtered value.

Figure 10.11 shows the INT0 input filter configuration. Figure 10.12 shows an operation example of INT0 input filter.

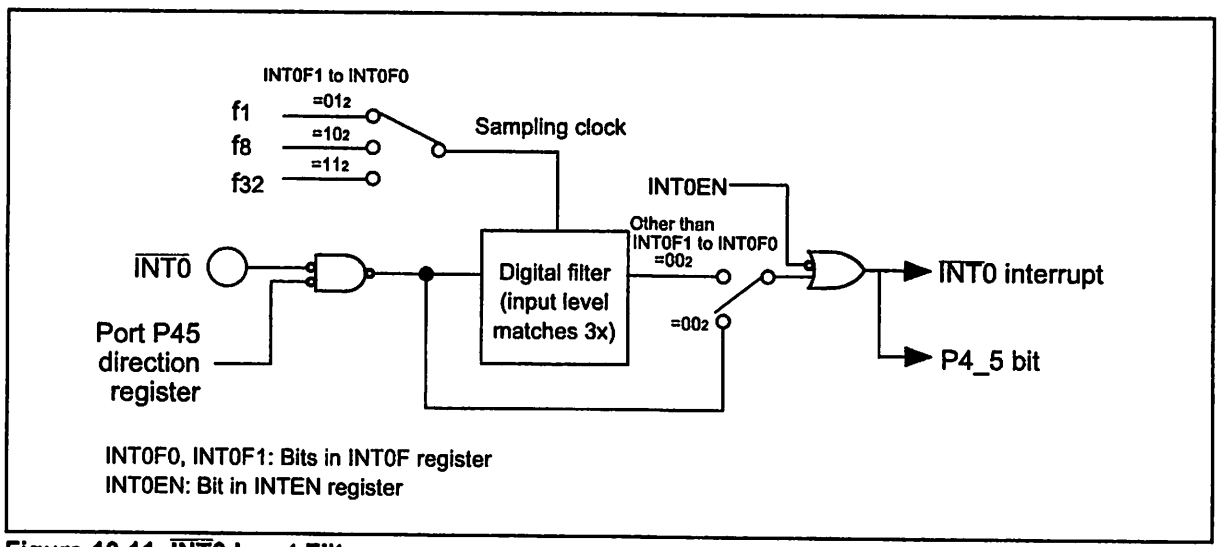


Figure 10.11 INT0 Input Filter

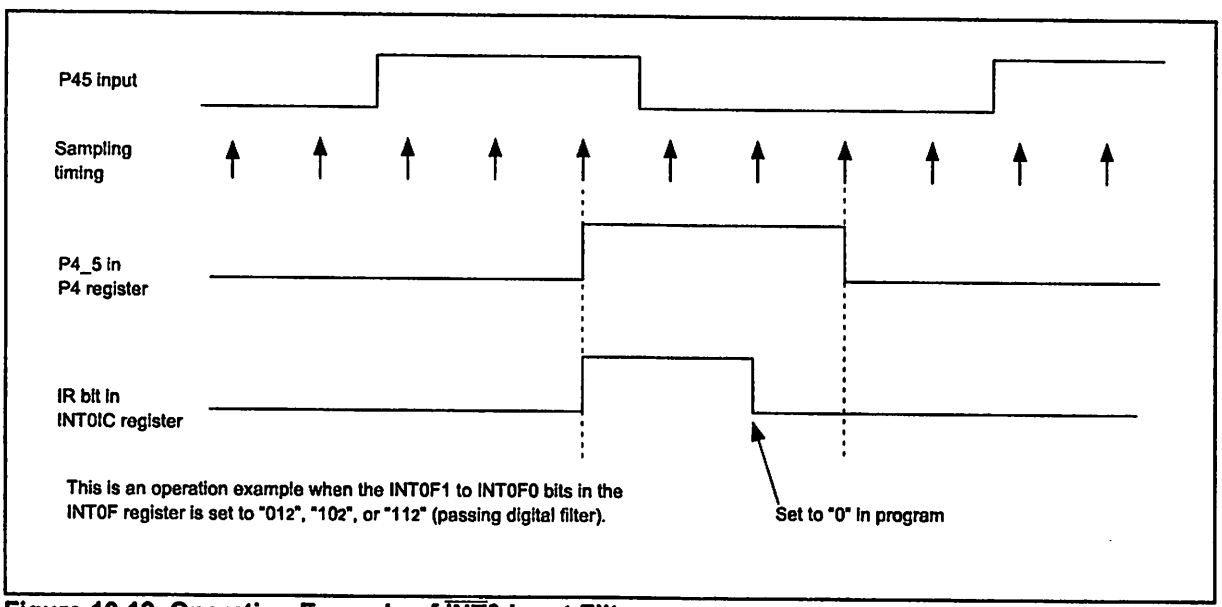


Figure 10.12 Operation Example of INT0 Input Filter

10.2.3 INT1 Interrupt and INT2 Interrupt

INT1 interrupts are triggered by INT1 inputs. The edge polarity can be selected with the R0EDG bit in the TXMR register. The INT1 pin is shared with the CNTR0 pin.

INT2 interrupts are triggered by INT2 inputs. The edge polarity can be selected with the R1EDG bit in the TYZMR register. The INT2 pin shared with the CNTR1 pin.

Figure 10.13 shows the TXMR and TYZMR registers when using INT1 and INT2 interrupts.

Timer X mode register

b7	b6	b5	b4	b3	b2	b1	b0	Symbol	Address	After reset
0	0	0	0			0	0	TXMR	008B16	0016

Bit symbol	Bit name	Function	RW
TXMOD0	Operation mode select bit 0, 1	b1 b0 0 0 : Timer mode or pulse period measurement mode[3]	RW
TXMOD1			RW
R0EDG	INT1/CNTR0 polarity switching bit[1, 2]	0 : Rising edge 1 : Falling edge	RW
TXS	Timer X count start flag	0 : Stops counting 1 : Starts counting	RW
TXOCNT	Set to "0" in timer mode		RW
TXMOD2	Operation mode select bit 2	0 : Other than pulse period measurement mode[3]	RW
TXEDG	Set to "0" in timer mode		RW
TXUND	Set to "0" in timer mode		RW

Notes:
1. The IR bit in the INT1IC may be set to "1" (interrupt requested) when the R0EDG bit is rewritten. Refer to the paragraph 19.2.5 "Changing Interrupt Factor" in the Usage Notes Reference Book.
2. This bit is used to select the polarity of INT1 interrupt in timer mode.
3. When using INT1 interrupts, should select timer mode.

Timer Y, Z mode register

b7	b6	b5	b4	b3	b2	b1	b0	Symbol	Address	After reset
							0	TYZMR	008016	0016

Bit symbol	Bit	Function	RW
TYMOD0	Timer Y operation mode bit	0 : Timer mode[1]	RW
R1EDG	INT2/CNTR1 polarity switching bit[2]	0 : Rising edge 1 : Falling edge	RW
TYWC	Timer Y write control bit	Functions varies depending on the operation mode	RW
TYS	Timer Y count start flag	0 : Stops counting 1 : Starts counting	RW
TZMOD0	Timer Z-related bit		RW
TZMOD1			RW
TZWC			RW
TZS			RW

Notes:
1. When using INT2 interrupts, must set to timer mode.
2. The IR bit in the INT2IC may be set to "1" (interrupt requested) when the R1EDG bit is rewritten. Refer to the paragraph 19.2.5 "Changing Interrupt Factor" in the Usage Notes Reference Book.

Figure 10.13 TXMR Register and TYZMR Register when INT1 and INT2 Interrupt Used

10.2.4 $\overline{\text{INT3}}$ Interrupt

$\overline{\text{INT3}}$ interrupts are triggered by $\overline{\text{INT3}}$ inputs. The TCC07 bit in the TCC0 register should be se to "0" ($\overline{\text{INT3}}$). The $\overline{\text{INT3}}$ input has a digital filter which can be sampled by one of three sampling clocks. The sampling clock is selected using the TCC11 to TCC10 bits in the TCC1 register. The IR bit in the INT3IC register is set to "1" (interrupt requested) when the sampled input level matches three times. The P3_3 bit in the P3 register indicates the previous value before filtering regardless of values set in the TCC11 to TCC10 bits.
The $\overline{\text{INT3}}$ pin is shared with the TC_{IN} pin.

When setting the TCC07 bit to "1" (f_{RING128}), $\overline{\text{INT3}}$ interrupts are triggered by f_{RING128} clock. The IR bit in the INT3IC register is set to "1" (interrupt requested) every f_{RING128} clock cycle or every half f_{RING128} clock cycle.
Figure 10.14 shows the TCC0 and TCC1 registers.

Timer C control register 0

b7	b6	b5	b4	b3	b2	b1	b0	Symbol	Address	After reset
	0	0						TCC0	009A16	0016

Bit symbol	Bit name	Function	RW
TCC00	Timer C control bit	0 : Count stop 1 : Count start	RW
TCC01	Timer C count source select bit[1]	b2 b1 0 0 : f1 0 1 : f8 1 0 : f32 1 1 : fRING-fast	RW
TCC02			RW
TCC03	INT3 interrupt and capture polarity select bit[1, 2]	b4 b3 0 0 : Rising edge 0 1 : Falling edge 1 0 : Both edges 1 1 : Avoid this setting	RW
TCC04			RW
(b6-b5)	Reserved bit	Set to "0"	RW
TCC07	INT3 interrupt/capture input switching bit[1, 2]	0 : INT3 1 : fRING128	RW

Notes:
1. Change this bit when TCC00 bit is set to "0" (count stop).
2. The IR bit in the INT3IC may be set to "1" (interrupt requested) when the TCC03, TCC04, or TCC07 bit is rewritten. Refer to the paragraph 19.2.5 "Changing Interrupt Factir" in the Usage Notes Reference Book.

Timer C control register 1

b7	b6	b5	b4	b3	b2	b1	b0	Symbol	Address	After reset
								TCC1	009B16	0016

Bit symbol	Bit name	Function	RW
TCC10	INT3 input filter select bit[1]	b1 b0 0 0: No filter 0 1: Filter with f1 sampling 1 0: Filter with f8 sampling 1 1: Filter with f32 sampling	RW
TCC11			RW
TCC12	Timer C counter reload select bit[2, 3]	0: No reload (free-run) 1: Set TC register to "000016" at compare 1 match	RO
TCC13	Compare 0/Capture select bit	0: Capture (input capture mode)[2] 1: Compare 0 output (output compare mode)	RW
TCC14	Compare 0 output mode select bit[3]	b5 b4 0 0: CMP output remains unchanged even when compare 0 matched 0 1: CMP output is reversed when compare 0 signal is matched 1 0: CMP output is set to low when compare 0 signal is matched 1 1: CMP output is set to high when compare 0 signal is matched	RW
TCC15			
TCC16	Compare 1 output mode select bit[3]	b7 b6 0 0: CMP output remains unchanged even when compare 1 matched 0 1: CMP output is reversed when compare 1 signal is matched 1 0: CMP output is set to low when compare 1 signal is matched 1 1: CMP output is set to high when compare 1 signal is matched	RW
TCC17			

Notes:
1. Input is recognized only when the same value from INT3 pin is sampled three times in succession.
2. Modify the TCC13 bit when the TCC00 bit in the TCC0 register is set to "0"(count stops)
3. Set the TCC12, TCC14 to TCC17 bits to "0" when the TCC13 bit is set to "0"(input capture mode).

Figure 10.14 TCC0 Register and TCC1 Register

10.3 Key Input Interrupt

A key input interrupt is generated on an input edge of any of the $\overline{KI0}$ to $\overline{KI3}$ pins. Key input interrupts can be used as a key-on wakeup function to exit wait or stop mode. $\overline{KIi}$ input can be enabled or disabled selecting with the KIiEN (i=0 to 3) bit in the KIEN register. The edge polarity can be rising edge or falling edge selecting with the KIiPL bit in the KIEN register. Note, however, that while input on any $\overline{KIi}$ pin which has had the KIiPL bit set to "0" (falling edge) is pulled low, inputs on all other pins of the port are not detected as interrupts. Similarly, while input on any $\overline{KIi}$ pin which has had the KIiPL bit set to "1" (rising edge) is pulled high, inputs on all other pins of the port are not detected as interrupts.

Figure 10.15 shows a block diagram of the key input interrupt.

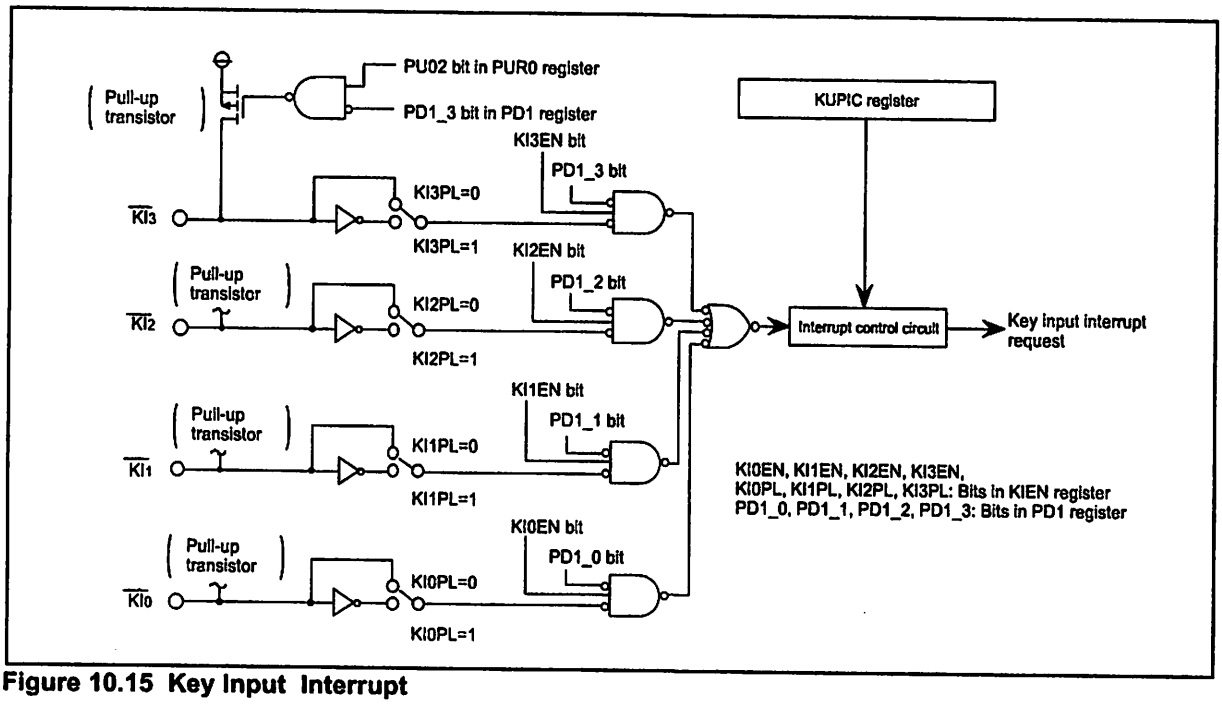


Figure 10.15 Key Input Interrupt

Key input enable register

b7 b6 b5 b4 b3 b2 b1 b0

Symbol: KIEN | Address: 0098₁₆ | After reset: 00₁₆

Bit symbol	Bit name	Function	RW
KI0EN	KI0 input enable bit	0 : Disabled 1 : Enabled	RW
KI0PL	KI0 input polarity select bit	0 : Falling edge 1 : Rising edges	RW
KI1EN	KI1 input enable bit	0 : Disabled 1 : Enabled	RW
KI1PL	KI1 input polarity select bit	0 : Falling edge 1 : Rising edges	RW
KI2EN	KI2 input enable bit	0 : Disabled 1 : Enabled	RW
KI2PL	KI2 input polarity select bit	0 : Falling edge 1 : Rising edges	RW
KI3EN	KI3 input enable bit	0 : Disabled 1 : Enabled	RW
KI3PL	KI3 input polarity select bit	0 : Falling edge 1 : Rising edges	RW

Notes:
1. The IR bit in the KUPIC register may be set to "1" (interrupt requested) when the KIEN register is rewritten. Refer to the paragraph 19.2.5 "Changing Interrupt Factor" in the Usage Notes Reference Book.

Figure 10.16 KIEN Register

10.4 Address Match Interrupt

An address match interrupt is generated immediately before executing the instruction at the address indicated by the RMADi register (i=0, 1). Set the start address of any instruction in the RMADi register. Use the AIER0 and AIER1 bits in the AIER register to enable or disable the interrupt. Note that the address match interrupt is unaffected by the I flag and IPL.

The value of the PC that is saved to the stack when an address match interrupt is acknowledged varies depending on the instruction at the address indicated by the RMAD i register (see the paragraph "register saving" for the value of the PC). Not appropriate return address is pushed on the stack. There are two ways to return from the address match interrupt as follows:

- Change the content of the stack and use a REIT instruction.
- Use an instruction such as POP to restore the stack as it was before an interrupt request was acknowledged. And then use a jump instruction.

Table 10.6 lists the value of the PC that is saved to the stack when an address match interrupt is acknowledged.

Figure 10.17 shows the AIER, and RMAD1 to RMAD0 registers.

Table 10.6 Value of PC Saved to Stack when Address Match Interrupt Acknowledged

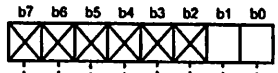
Address indicated by RMADi register (i=0,1)	PC value saved[Note]
• 16-bit operation code instruction • Instruction shown below among 8-bit operation code instructions ADD.B:S #IMM8,dest SUB.B:S #IMM8,dest AND.B:S #IMM8,dest OR.B:S #IMM8,dest MOV.B:S #IMM8,dest STZ.B:S #IMM8,dest STNZ.B:S #IMM8,dest STZX.B:S #IMM81,#IMM82,dest CMP.B:S #IMM8,dest PUSHM src POPM dest JMPS #IMM8 JSRS #IMM8 MOV.B:S #IMM,dest (However, dest = A0 or A1)	Address indicated by RMADi register + 2
• Instructions other than the above	Address indicated by RMADi register + 1

Note: See the paragraph "saving registers" for the PC value saved.

Table 10.7 Relationship Between Address Match Interrupt Sources and Associated Registers

Address match interrupt factors	Address match interrupt enable bit	Address match interrupt register
Address match interrupt 0	AIER0	RMAD0
Address match interrupt 1	AIER1	RMAD1

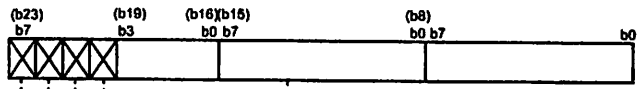
Address match interrupt enable register



Symbol	Address	After reset
AIER	000916	XXXXXX002

Bit symbol	Bit name	Function	RW
AIER0	Address match interrupt 0 enable bit	0 : Interrupt disabled 1 : Interrupt enabled	RW
AIER1	Address match interrupt 1 enable bit	0 : Interrupt disabled 1 : Interrupt enabled	RW
— (b7-b2)	Nothing is assigned. When write, set to "0". When read, their contents are indeterminate.		—

Address match interrupt register i (i = 0, 1)



Symbol	Address	After reset
RMAD0	001216 to 001016	X000001₆
RMAD1	001616 to 001416	X000001₆

Function	Setting range	RW
Address setting register for address match interrupt	000001₆ to FFFFF16	RW
— (b7-b4)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.	—

Figure 10.17 AIER Register and RMAD0 to RMAD1 Registers

11. Watchdog Timer

The watchdog timer is the function of detecting when the program is out of control. Therefore, we recommend using the watchdog timer to improve reliability of a system. Figure 11.1 shows the watchdog timer block diagram.The watchdog timer contains a 15-bit counter which counts down the clock derived by dividing the CPU clock using the prescaler. Whether to generate a watchdog timer interrupt request or apply a watchdog timer reset as an operation to be performed when the watchdog timer underflows after reaching the terminal count can be selected using the PM12 bit in the PM1 register. The PM12 bit can only be set to "1" (reset). Once this bit is set to "1", it cannot be set to "0" (watchdog timer interrupt) in a program. Refer to Section 5.3, "Watchdog Timer Reset" for details.
The divide-by-N value for the prescaler can be chosen to be 16 or 128 with the WDC7 bit in the WDC register. The period of watchdog timer can be calculated as given below. The period of watchdog timer is, however, subject to an error due to the prescaler.

$$\text{Watchdog timer period} = \frac{\text{Prescaler dividing (16 or 128) X Watchdog timer count (32768)}}{\text{CPU clock}}$$

For example, when CPU clock = 16 MHz and the divide-by-N value for the prescaler= 16, the watchdog timer period is approx. 32.8 ms.

Figure 11.2 shows the OFS, the WDC, the WDTR and the WDTS registers. The watchdog timer operation after reset can be selected using the WDTON bit in the option function select register (0FFFF$_{16}$ address).

- When the WDTON bit is "0" (the watchdog timer is started automatically after reset), the watchdog timer and the prescaler both start counting automatically after reset.
- When the WDTON bit is "1" (the watchdog timer is inactive after reset), the watchdog timer and the prescaler both are inactive after reset, so that the watchdog timer is activated to start counting by writing to the WDTS register.

The WDTON bit can not be changed in a program. When setting the WDTON bit, write "0" into bit 0 of 0FFFF$_{16}$ address using a flash writer. The watchdog timer is nitialized by writing to the WDTR register and the counting continues.

In stop mode and wait mode, the watchdog timer and the prescaler are stopped. Counting is resumed from the held value when the modes or state are released.

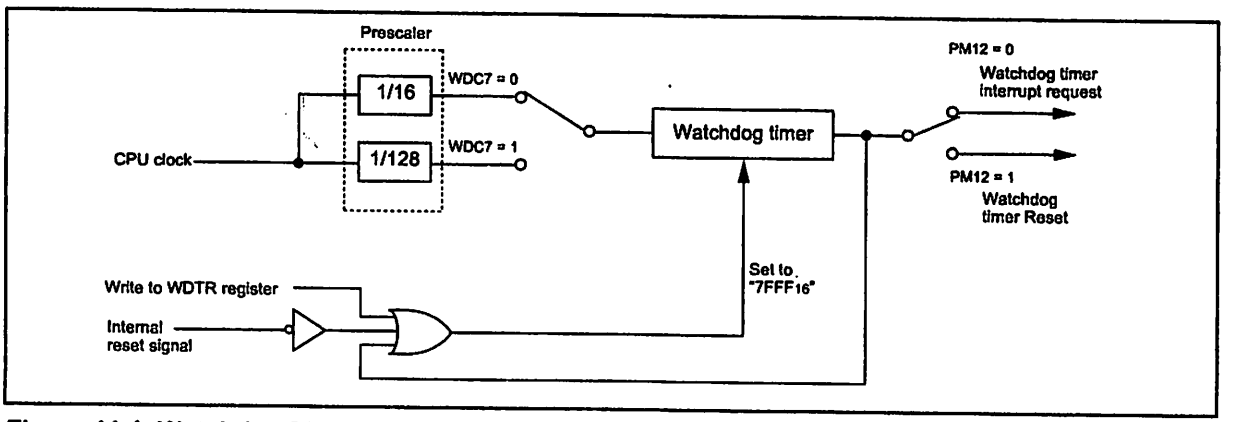


Figure 11.1 Watchdog Timer Block Diagram

Option function select register[1]

b7	b6	b5	b4	b3	b2	b1	b0
1	1	1	1	1	1	1	

Symbol	Address	Before shipment
OFS	0FFFF16	FF16

Bit symbol	Bit name	Function	RW
WDTON	Watchdog timer start select bit[2]	0: The watchdog timer starts automatically after reset 1: The watchdog timer is inactive after reset	RW
(b7 - b1)	Reserved bit	Set to "1"	RW

Note1: The OFS register can not be changed in a program. Set using a flash writer.

Watchdog timer control register

b7	b6	b5	b4	b3	b2	b1	b0
	0	0					

Symbol	Address	After reset
WDC	000F16	000111112

Bit symbol	Bit name	Function	RW
(b4-b0)	High-order bit of watchdog timer		RO
(b5)	Reserved bit	Must set to "0"	RW
(b6)	Reserved bit	Must set to "0"	RW
WDC7	Prescaler select bit	0 : Divided by 16 1 : Divided by 128	RW

Watchdog timer reset register

b7	b0

Symbol	Address	After reset
WDTR	000D16	Indeterminate

Function	RW
The watchdog is initialized after a write instruction to this register. The watchdog timer value is always initialized to "7FFF16" regardless of whatever value is written.	WO

Watchdog timer start register

b7	b0

Symbol	Address	After reset
WDTS	000E16	Indeterminate

Function	RW
The watchdog timer starts counting after a write instruction to this register.	WO

Figure 11.2 OFS, WDC, WDTR and WDTS Registers

14. A/D Converter

The A/D converter consists of one 10-bit successive approximation A/D converter circuit with a capacitive coupling amplifier. The analog inputs share the pins with P00 to P07 and P10 to P13. Therefore, when using these pins, make sure the corresponding port direction bits are set to "0" (input mode).

When not using the A/D converter, set the VCUT bit to "0" (Vref unconnected), so that no current will flow from the VREF pin into the resistor ladder, helping to reduce the power consumption of the chip.

The result of A/D conversion is stored in the AD register.

Table 14.1 shows the performance of the A/D converter. Figure 14.1 shows a block diagram of the A/D converter, and Figures 14.2 and 14.3 show the A/D converter-related registers.

Table 14.1 Performance of A/D converter

Item	Performance
Method of A/D conversion	Successive approximation (capacitive coupling amplifier)
Analog input voltage[1]	0V to Vref
Operating clock ϕ_{AD}[2]	AVcc = 5V f_{AD}, divide-by-2 of f_{AD}, divide-by-4 of f_{AD} AVcc = 3V divide-by-2 of f_{AD}, divide-by-4 of f_{AD}
Resolution	8-bit or 10-bit (selectable)
Integral nonlinearity error	AVcc = Vref = 5V • 8-bit resolution ±2 LSB • 10-bit resolution ±3 LSB AVcc = Vref = 3.3 V • 8-bit resolution ±2 LSB • 10-bit resolution ±5 LSB
Operating modes	One-shot mode and repeat mode[3]
Analog input pins	12 pins (AN0 to AN11)
A/D conversion start condition	ADST bit in ADCON0 register is set to "1" (A/D conversion starts)
Conversion speed per pin	• Without sample and hold function 8-bit resolution: 49 ϕ_{AD} cycles, 10-bit resolution: 59 ϕ_{AD} cycles • With sample and hold function 8-bit resolution: 28 ϕ_{AD} cycles, 10-bit resolution: 33 ϕ_{AD} cycles

Notes:

1. Does not depend on use of sample and hold function.
2. The frequency of ϕ_{AD} must be 10 MHz or less.
   When AVcc is less than 4.2V, ϕ_{AD} must be $f_{AD}/2$ or less by dividing f_{AD}.
   Without sample and hold function, the ϕ_{AD} frequency should be 250 kHz or more.
   With the sample and hold function, the ϕ_{AD} frequency should be 1 MHz or more.
3. In repeat mode, only 8-bit mode can be used.

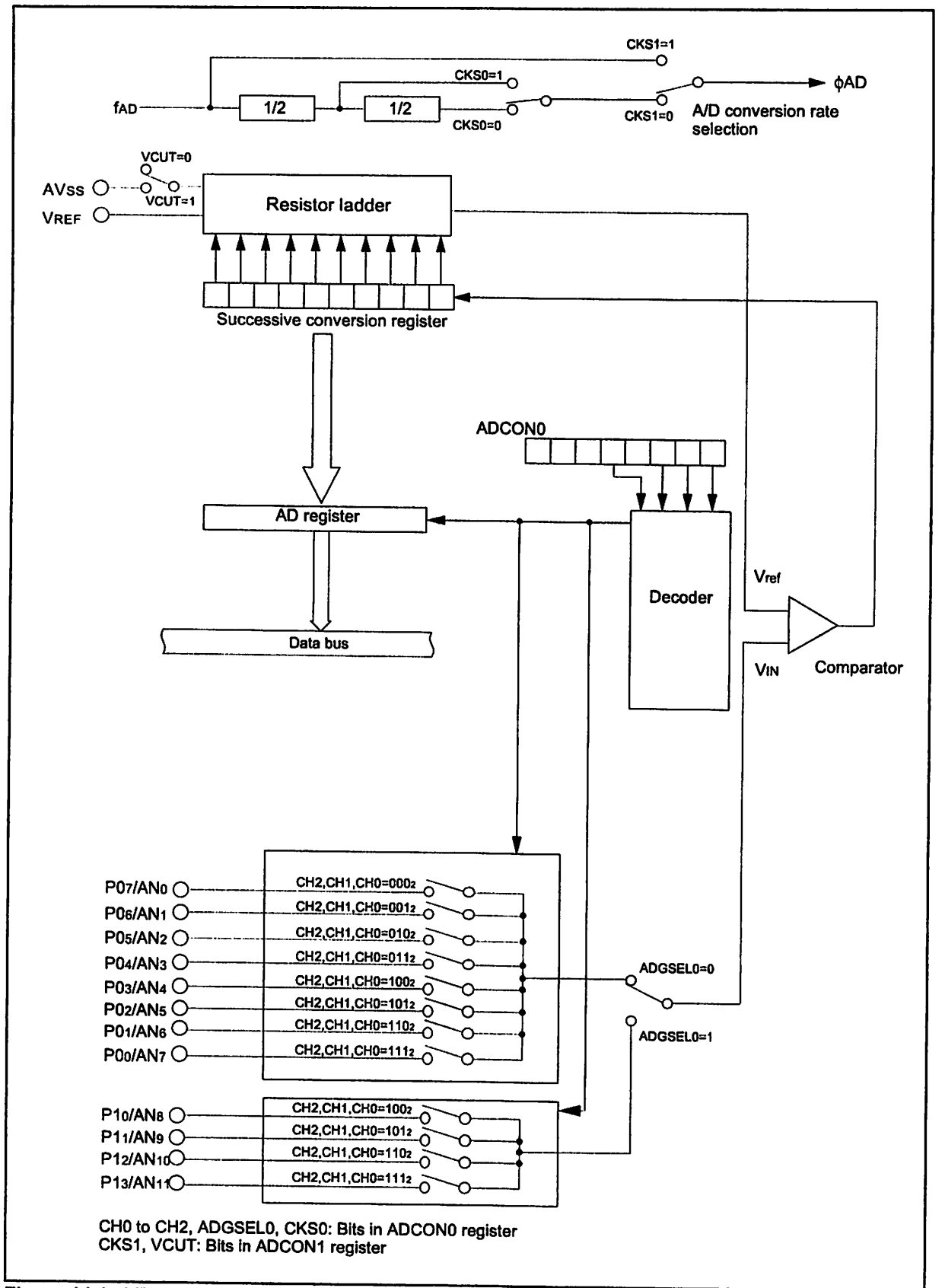


Figure 14.1 A/D Converter Block Diagram

AD control register 0[1]

b7	b6	b5	b4	b3	b2	b1	b0	Symbol	Address	After reset
		0						ADCON0	00D616	00000XXX2

Bit symbol	Bit name	Function	RW
CH0	Analog input pin select bit	See Note 4.	RW
CH1			RW
CH2			RW
MD	AD operation mode select bit[2]	0 : One-shot mode 1 : Repeat mode	RW
ADGSEL0	AD input group select bit[4]	0: Port P0 group selected (AN0 to AN7) 1: Port P1 group selected (AN8 to AN11)	RW
(b5)	Reserved bit	Set to "0"	RW
ADST	A/D conversion start flag	0 : A/D conversion disabled 1 : A/D conversion started	RW
CKS0	Frequency select bit 0[3]	0 : fAD/4 is selected 1 : fAD/2 is selected	RW

Notes:
1. If the ADCON register is rewritten during A/D conversion, the conversion result is indeterminate.
2. When changing A/D operation mode, set analog input pin again.
3. This bit is valid when the CKS1 bit in the ADCON1 register is set to "0".
4. The analog input pin can be selected by a combination of the CH2 to CH0 bits and ADGSEL0 bit as follows:

CH2 to CH0	ADGSEL0=0	ADGSEL0=1
0002	AN0	Avoid these settings
0012	AN1	
0102	AN2	
0112	AN3	
1002	AN4	AN8
1012	AN5	AN9
1102	AN6	AN10
1112	AN7	AN11

AD control register 1[1]

b7	b6	b5	b4	b3	b2	b1	b0	Symbol	Address	After reset
0	0				0	0	0	ADCON1	00D716	0016

Bit symbol	Bit name	Function	RW
(b2-b0)	Reserved bit	Set to "0"	RW
BITS	8/10-bit mode select bit[2]	0 : 8-bit mode 1 : 10-bit mode	RW
CKS1	Frequency select bit 1[3]	0 : CKS0 bit in ADCON0 register is valid 1 : fAD is selected	RW
VCUT	Vref connect bit[4]	0 : Vref not connected 1 : Vref connected	RW
(b6-b7)	Reserved bit	Set to "0"	RW

Notes:
1. If the ADCON1 register is rewritten during A/D conversion, the conversion result is indeterminate.
2. In repeat mode, the BITS bit must be set to "0" (8-bit mode).
3. The φAD frequency must be 10 MHz or less.
4. If the VCUT bit is reset from "0" (Vref unconnected) to "1" (Vref connected), wait for 1 µs or more before starting A/D conversion.

Figure 14.2 ADCON0 Register and ADCON1 Register

AD control register 2[1]

b7	b6	b5	b4	b3	b2	b1	b0
X	X	X	X	0	0	0	

Symbol	Address	After reset
ADCON2	00D4₁₆	00₁₆

Bit symbol	Bit name	Function	RW
SMP	AD conversion method select bit	0 : Without sample and hold 1 : With sample and hold	RW
------ (b3-b1)	Reserved bit	Set to "0"	RW
------ (b7-b4)	Nothing is assigned. When write, write "0". When read, its content is "0".		----

Notes:
1. If the ADCON2 register is rewritten during A/D conversion, the conversion result is indeterminate.

AD register

Symbol	Address	After reset
AD	00C1₁₆-00C0₁₆	Indeterminate

(b15) b7							(b8) b0	b7							b0
X	X	X	X	X	X	X									

Function		RW
When BITS bit in ADCON1 register is set to "1" (10-bit mode)	When BITS bit in ADCON1 register is set to "0" (8-bit mode)	
8 low-order bits of A/D conversion result	A/D conversion result	RO
2 high-order bits of A/D conversion result	When read, its content is indeterminate.	RO
Nothing is assigned. When write, set to "0". When read, its content is "0".		----

Figure 14.3 ADCON2 Register and AD Register

14.1 One-shot Mode

In one-shot mode, the input voltage on one selected pin is A/D converted once. Table 14.2 lists the specifications of one-shot mode. Figure 14.4 shows the ADCON0 and ADCON1 registers in one-shot mode.

Table 14.2 One-shot Mode Specifications

Item	Specification
Function	Input voltage on one pin selected by CH2 to CH0 and ADGSEL0 bit is A/D converted once.
Start condition	Set ADST bit to "1"
Stop condition	• Completion of A/D conversion (ADST bit is set to "0") • Set ADST bit to "0"
Interrupt request generation timing	End of A/D conversion
Input pin	One of AN0 to AN11, as selected
Reading of result of A/D converter	Read AD register

AD control register 0[1]

b7	b6	b5	b4	b3	b2	b1	b0	Symbol	Address	After reset
		0		0					ADCON0	00D616
								00000XXX2		

Bit symbol	Bit name	Function	RW
CH0	Analog input pin select bit[2]	See Note 4	RW
CH1			RW
CH2			RW
MD	AD operation mode select bit[2]	0 : One-shot mode	RW
ADGSEL0	AD input group select bit[4]	0: Port P0 group selected (AN0 to AN7) 1: Port P1 group selected (AN8 to AN11)	RW
(b5)	Reserved bit	Set to "0"	RW
ADST	A/D conversion start flag	0 : A/D conversion disabled 1 : A/D conversion started	RW
CKS0	Frequency select bit 0[3]	0 : fAD/4 is selected 1 : fAD/2 is selected	RW

Notes:
1. If the ADCON0 register is rewritten during A/D conversion, the conversion result is indeterminate.
2. When changing A/D operation mode, set analog input pin again.
3. This bit is valid when the CKS1 bit in the ADCON1 register is set to "0".
4. The analog input pin can be selected by a combination of the CH2 to CH0 bits and ADGSEL0 bit as follows:

CH2 to CH0	ADGSEL0=0	ADGSEL0=1
0002	AN0	Avoid these settings
0012	AN1	
0102	AN2	
0112	AN3	
1002	AN4	AN8
1012	AN5	AN9
1102	AN6	AN10
1112	AN7	AN11

AD control register 1[1]

b7	b6	b5	b4	b3	b2	b1	b0	Symbol	Address	After reset	
0	0	1			0	0	0		ADCON1	00D716	0016

Bit symbol	Bit name	Function	RW
(b2-b0)	Reserved bit	Set to "0"	RW
BITS	8/10-bit mode select bit	0 : 8-bit mode 1 : 10-bit mode	RW
CKS1	Frequency select bit 1[2]	0 : CKS0 bit in ADCON0 register is valid 1 : fAD is selected	RW
VCUT	Vref connect bit[3]	1 : Vref connected	RW
(b6-b7)	Reserved bit	Set to "0"	RW

Notes:
1. If the ADCON1 register is rewritten during A/D conversion, the conversion result is indeterminate.
2. The φAD frequency must be 10MHz or less.
3. If the VCUT bit is reset from "0" (Vref unconnected) to "1" (Vref connected), wait for 1 µs or more before starting A/D conversion.

Figure 14.4 ADCON0 Register and ADCON1 Registers in One-shot Mode

14.2 Repeat Mode

In repeat mode, the input voltage on one selected pin is A/D converted repeatedly. Table 14.3 lists the specifications of repeat mode. Figure 14.5 shows the ADCON0 and ADCON1 registers in repeat mode.

Table 14.3 Repeat Mode Specifications

Item	Specification
Function	Input voltage on one pin selected by CH2 to CH0 and ADGSEL0 bits is A/D converted repeatedly
Start condition	Set ADST bit to "1"
Stop condition	Set ADST bit to "0"
Interrupt request generation timing	None generated
Input pin	One of AN0 to AN11, as selected
Reading of result of A/D converter	Read AD register

AD control register 0[1]

b7	b6	b5	b4	b3	b2	b1	b0
		0		1			

Symbol: ADCON0 — Address: 00D6₁₆ — After reset: 00000XXX₂

Bit symbol	Bit name	Function	RW
CH0	Analog input pin select bit[2]	See Note 4	RW
CH1			RW
CH2			RW
MD	AD operation mode select bit [2]	1 : Repeat mode	RW
ADGSEL0	AD input group select bit[4]	0: Port P0 group selected (AN0 to AN7) 1: Port P1 group selected (AN8 to AN11)	RW
(b5)	Reserved bit	Set to "0"	RW
ADST	A/D conversion start flag	0 : A/D conversion disabled 1 : A/D conversion started	RW
CKS0	Frequency select bit 0[3]	0 : fAD/4 is selected 1 : fAD/2 is selected	RW

Notes:
1. If the ADCON0 register is rewritten during A/D conversion, the conversion result is indeterminate.
2. When changing A/D operation mode, set analog input pin again.
3. This bit is valid when the CKS1 bit in the ADCON1 register is set to "0".
4. The analog input pin can be selected by a combination of the CH2 to CH0 bits and ADGSEL0 bit as follows:

CH2 to CH0	ADGSEL0=0	ADGSEL0=1
000₂	AN0	Avoid these settings
001₂	AN1	
010₂	AN2	
011₂	AN3	
100₂	AN4	AN8
101₂	AN5	AN9
110₂	AN6	AN10
111₂	AN7	AN11

AD control register 1[1]

b7	b6	b5	b4	b3	b2	b1	b0
0	0	1		0	0	0	0

Symbol: ADCON1 — Address: 00D7₁₆ — After reset: 00₁₆

Bit symbol	Bit name	Function	RW
(b2-b0)	Reserved bit	Set to "0"	RW
BITS	8/10-bit mode select bit[2]	0 : 8-bit mode	RW
CKS1	Frequency select bit 1[3]	0 : CKS0 bit in ADCON0 register is valid 1 : fAD is selected	RW
VCUT	Vref connect bit[4]	1 : Vref connected	RW
(b6-b7)	Reserved bit	Set to "0"	RW

Notes:
1. If the ADCON1 register is rewritten during A/D conversion, the conversion result is indeterminate.
2. In repeat mode, the BITS bit must be set to "0" (8-bit mode).
3. The fAD frequency must be 10 MHz or less.
4. If the VCUT bit is reset from "0" (Vref unconnected) to "1" (Vref connected), wait for 1 µs or more before starting A/D conversion.

Figure 14.5 ADCON0 Register and ADCON1 Register in Repeat Mode

14.3 Sample and Hold

If the SMP bit in the ADCON2 register is set to "1" (with sample-and-hold), the conversion speed per pin is increased to 28 ØAD cycles for 8-bit resolution or 33 ØAD cycles for 10-bit resolution. Sample-and-hold is effective in all operation modes. Select whether or not to use the sample-and-hold function before starting A/D conversion.

When performing the A/D conversion, charge the comparator capacitor inside the microcomputer. Figure 14.6 shows the A/D conversion timing diagram.

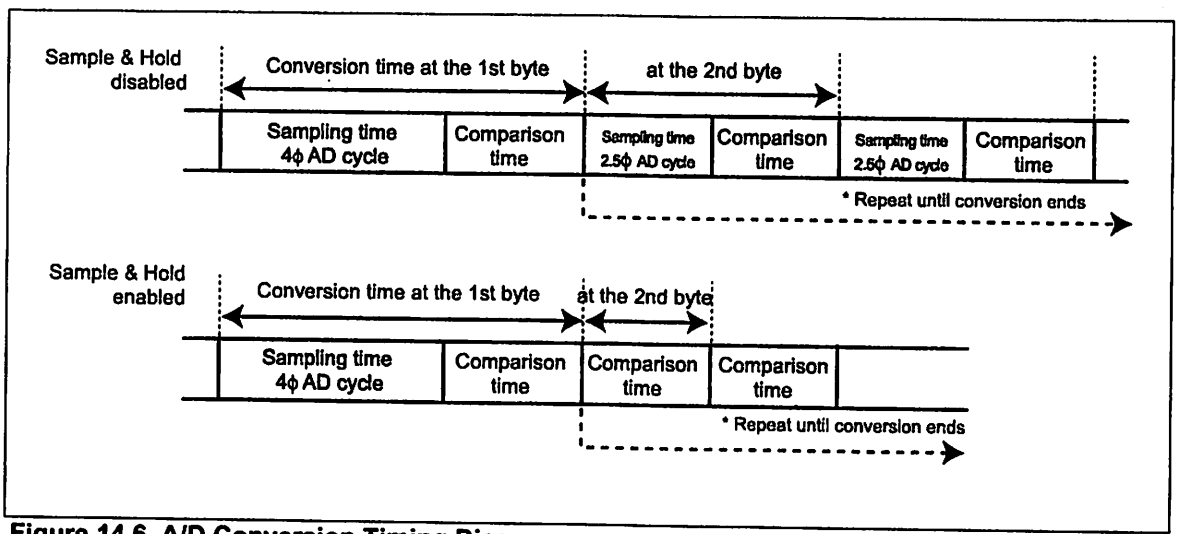


Figure 14.6 A/D Conversion Timing Diagram

14.4 A/D conversion cycles

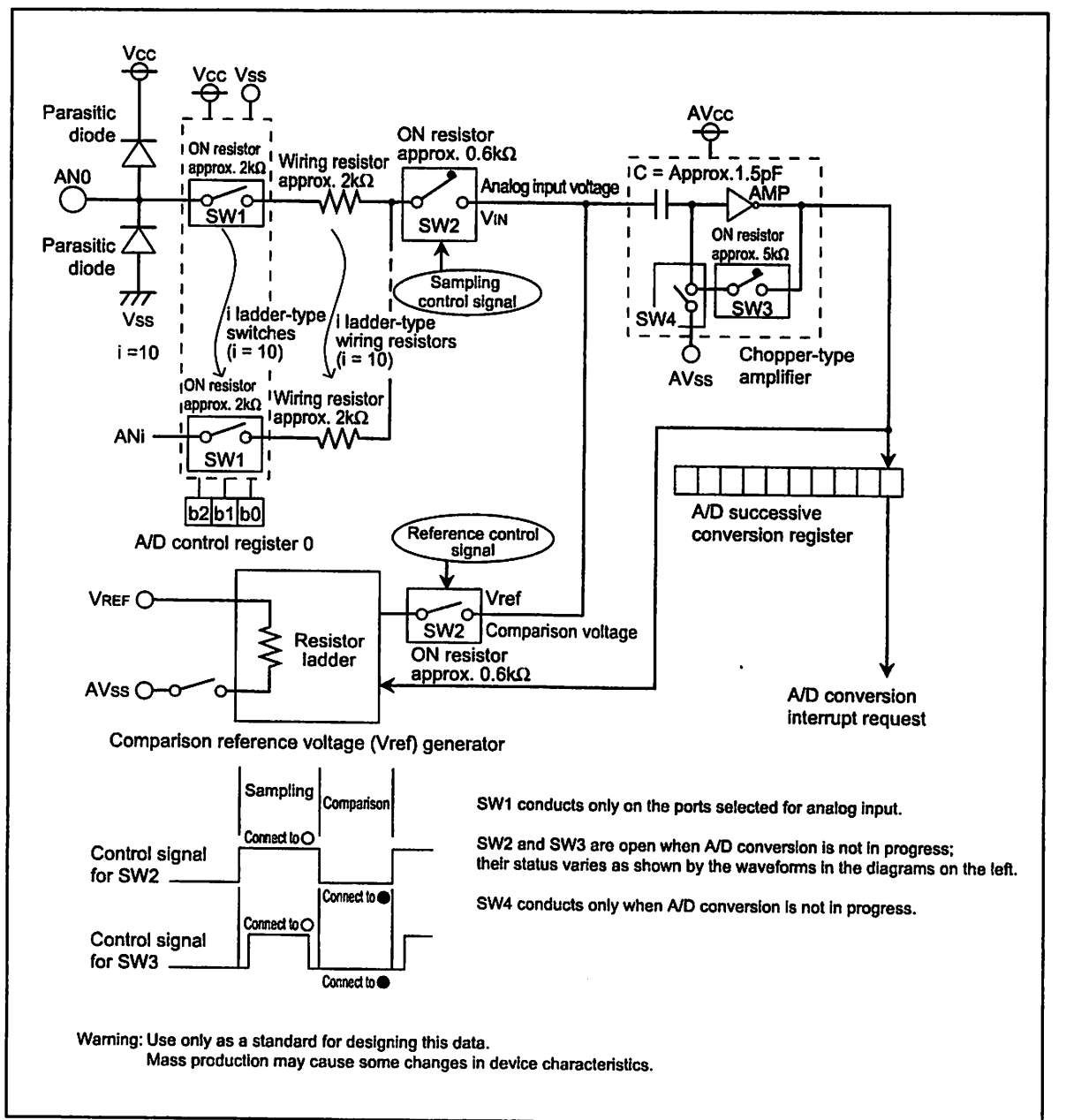
Figure 14.7 shows the A/D conversion cycles.

			Conversion time at the 1st bit		Conversion time at the 2nd bit and the follows		End process
A/D conversion mode		Conversion time	Sampling time	Comparison time	Sampling time	Comparison time	End process
Without sample & hold	8 bits	49 ϕ AD	4 ϕ AD	2.0 ϕ AD	2.5 ϕ AD	2.5 ϕ AD	8.0 ϕ AD
Without sample & hold	10 bits	59 ϕ AD	4 ϕ AD	2.0 ϕ AD	2.5 ϕ AD	2.5 ϕ AD	8.0 ϕ AD
With sample & hold	8 bits	28 ϕ AD	4 ϕ AD	2.5 ϕ AD	0.0 ϕ AD	2.5 ϕ AD	4.0 ϕ AD
With sample & hold	10 bits	33 ϕ AD	4 ϕ AD	2.5 ϕ AD	0.0 ϕ AD	2.5 ϕ AD	4.0 ϕ AD

Figure 14.7 A/D Conversion Cycles

14.5 Internal Equivalent Circuit of Analog Input

Figure 14.8 shows the internal equivalent circuit of analog input.



Warning: Use only as a standard for designing this data.
Mass production may cause some changes in device characteristics.

Figure 14.8 Internal Equivalent Circuit to Analog Input

14.6 Inflow Current Bypass Circuit

Figure 14.9 shows the configuration of the inflow current bypass circuit, figure 14.10 shows the example of an inflow current bypass circuit where Vcc or more is applied.

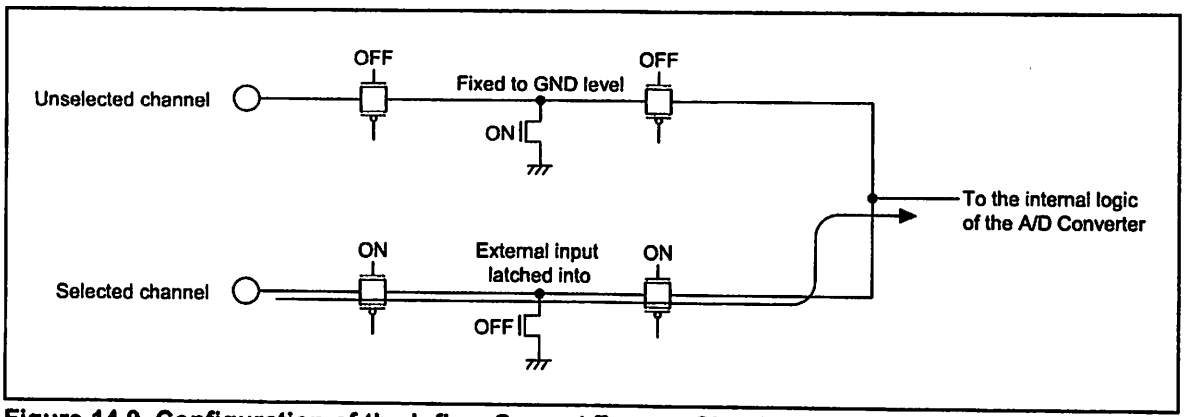


Figure 14.9 Configuration of the Inflow Current Bypass Circuit

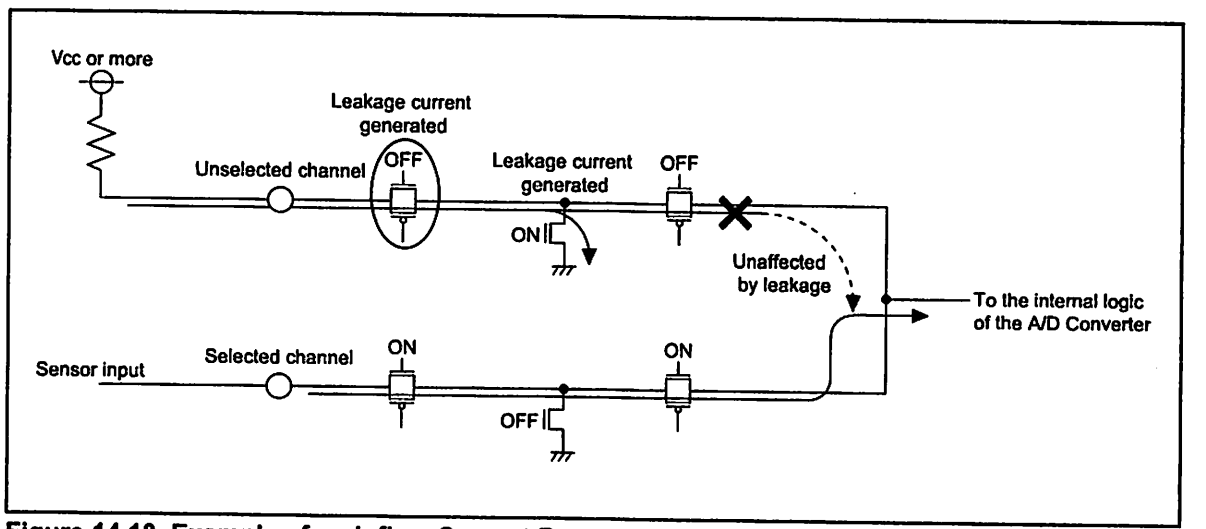


Figure 14.10 Example of an Inflow Current Bypass Circuit where Vcc or More is Applied

15. Programmable I/O Ports

15. 1 Description

The programmable input/output ports (hereafter referred to as "I/O ports") consist of 22 lines P0, P1, P30 to P33, P37, and P45. Each port can be set for input or output every line by using a direction register, and can also be chosen to be or not be pulled high every 4 lines. The port P1 allows the drive capacity of its N-channel output transistor to be set as necessary. The port P1 can be used as LED drive port if the drive capacity is set to "HIGH".

P46 and P47 can be used as an input only port if the main clock oscillation circuit is not used.

Figures 15.1 to 15.5 show the I/O ports. Figure 15.6 shows the I/O pins.

Each pin functions as an I/O port or a peripheral function input/output.

For details on how to set peripheral functions, refer to each functional description in this manual. If any pin is used as a peripheral function input, set the direction bit for that pin to "0" (input mode). Any pin used as an output pin for peripheral functions is directed for output no matter how the corresponding direction bit is set.

15.1.1 Port Pi Direction Register (PDi Register, i = 0, 1, 3, 4)

Figure 15.7 shows the PDi register.

This register selects whether the I/O port is to be used for input or output. The bits in this register correspond one for one to each port.

15.1.2 Port Pi Register (Pi Register, i = 0 to 4)

Figure 15.8 shows the Pi register.

Data I/O to and from external devices are accomplished by reading and writing to the Pi register. The Pi register consists of a port latch to hold the output data and a circuit to read the pin status. For ports set for input mode, the input level of the pin can be read by reading the corresponding Pi register, and data can be written to the port latch by writing to the Pi register.

For ports set for output mode, the port latch can be read by reading the corresponding Pi register, and data can be written to the port latch by writing to the Pi register. The data written to the port latch is output from the pin. The bits in the Pi register correspond one for one to each port.

15.1.3 Pull-up Control Register 0, Pull-up Control Register 1 (PUR0 and PUR1 Registers)

Figure 15.9 shows the PUR0 and PUR1 registers.

The PUR0 and PUR1 register bits can be used to select whether or not to pull the corresponding port high in 4 bit units. The port chosen to be pulled high has a pull-up resistor connected to it when the direction bit is set for input mode.

15.1.4 Port P1 Drive Capacity Control Register (DRR Register)

Figure 15.9 shows the DRR register.

The DRR register is used to control the drive capacity of the port P1 N-channel output transistor. The bits in this register correspond one for one to each port.

16. Electrical Characteristics

Table 16.1 Absolute Maximum Ratings

Symbol	Parameter	Condition	Rated value	Unit
Vcc	Supply voltage	Vcc=AVcc	-0.3 to 6.5	V
AVcc	Analog supply voltage	Vcc=AVcc	-0.3 to 6.5	V
V_I	Input voltage		-0.3 to Vcc+0.3	V
V_O	Output voltage		-0.3 to Vcc+0.3	V
P_d	Power dissipation	Topr=25 °C	300	mW
T_{opr}	Operating ambient temperature		-20 to 85 / -40 to 85 (D version)	°C
T_{stg}	Storage temperature		-65 to 150	°C

Table 16.2 Recommended Operating Conditions

Symbol	Parameter	Conditions	Standard			Unit
			Min.	Typ.	Max.	
Vcc	Supply voltage		2.7		5.5	V
AVcc	Analog supply voltage			Vcc[3]		V
Vss	Supply voltage			0		V
AVss	Analog supply voltage			0		V
V_{IH}	"H" input voltage		0.8Vcc		Vcc	V
V_{IL}	"L" input voltage		0		0.2Vcc	V
$I_{OH\ (sum)}$	"H" peak all output currents: Sum of all pins' IOH (peak)				-60.0	mA
$I_{OH\ (peak)}$	"H" peak output current				-10.0	mA
$I_{OH\ (avg)}$	"H" average output current				-5.0	mA
$I_{OL\ (sum)}$	"L" peak all output currents: Sum of all pins' IOL (peak)				60	mA
$I_{OL\ (peak)}$	"L" peak output current: Except P1₀ to P1₇				10	mA
	P1₀ to P1₇	Drive ability HIGH			30	mA
		Drive ability LOW			10	mA
$I_{OL\ (avg)}$	"L" average output current: Except P1₀ to P1₇				5	mA
	P1₀ to P1₇	Drive ability HIGH			15	mA
		Drive ability LOW			5	mA
f(X_{IN})	Main clock input oscillation frequency	3.0V ≤ Vcc ≤ 5.5V	0		20	MHz
		2.7V ≤ Vcc < 3.0V	0		10	MHz

Note
1: Referenced to Vcc = AVcc = 2.7 to 5.5V at Topr = -20 to 85 °C / -40 to 85 °C unless otherwise specified.
2: The mean output current is the mean value within 100ms.
3: Set Vcc=AVcc

17. Flash Memory Version

17.1 Overview

The flash memory version has two modes—CPU rewrite and standard serial I/O—in which its flash memory can be operated on.

Table 17.1 outlines the performance of flash memory version (see "Table 1.1 Performance" for the items not listed on Table 17.1).

Table 17.1 Flash Memory Version Performance

Item		Specification
Flash memory operating mode		2 modes (CPU rewrite and standard serial I/O)
Erase block		See "Figure 17.1. Flash Memory Block Diagram"
Method for program		In units of byte
Method for erasure		Block erase
Program, erase control method		Program and erase controlled by software command
Protect method		Protect for Block 0 and 1 by FMR02 bit in FMR0 register
		Protect for Block 0 by FMR16 bit and Block 1 by FMR16 bit
Number of commands		5 commands
Number of program and erasure (1)	Block0 and 1 (program area)	1,000 times
	BlockA and B (data area)	10,000 times
ROM code protection		Standard serial I/O mode is supported.

NOTES:

1: Definition of program/erase times
The program/erase times are defined to be per-block erase times. When the program/erase times are n times (n=1,000 or 10,000 times), to erase n times per block is possible. For example, if performing one-byte write to the distinct addresses on the Block A of 2K-byte block 2,048 times and then erasing that block, the number of the program/erase cycles is one time. if rewriting more than 1,000 times, run the program until the vacant areas are all used to reduce the substantial rewrite times and then erase. Avoid rewriting only particular blocks and rewrite to average the program and erase times to each block. Also keep the erase times as inrformation and set up the limit times.

Table 17.2 Flash Memory Rewrite Modes

Flash memory rewrite mode	CPU rewrite mode	Standard serial I/O mode
Function	User ROM area is rewritten by executing software commands from the CPU. EW0 mode: Can be rewritten in any area other than the flash memory EW1 mode: Can be rewritten in the flash memory	User ROM area is rewritten by using a dedicated serial programmer. Standard serial I/O mode 1 : Clock synchronous serial I/O Standard serial I/O mode 2 : UART
Areas which can be rewritten	User ROM area	User ROM area
Operation mode	Single chip mode	Boot mode
ROM programmer	None	Serial programmer

18. On-chip debugger

The microcomputer has functions to execute the on-chip debugger. Refer to "Appendix 2 Connecting examples for serial writer and on-chip debugging emulator". Refer to the respective on-chip debugger manual for the details of the on-chip debugger. Next, here are some explanations for the respective functions. Debugging the user system which uses these functions is not available. When using the on-chip debugger, design the system without using these functions in advance. Additionally, the on-chip debugger uses the address $0C000_{16}$ to $0C7FF_{16}$ of the flash memory, thus avoid using for the user system.

18.1 Address match interrupt

The interrupt request is generated right before the arbitrary address instruction is executed. The debugger break function uses the address match interrupt. Refer to "10.4 Address match interrupt" for the details of the address match interrupt. Also, avoid setting the address match interrupt (the registers of AIER, RMAD0, RMAD1 and the fixed vector tables) with using the user system when using the on-chip debugger.

18.2 Single step interrupt

The interrupt request is generated every time one instruction is executed. The debugger single step function uses the single step interrupt. The other interrupt is not generated when using the single step interrupt. The single step interrupt is only for the developed support tool.

18.3 UART1

The UART1 is used for the communication with the debugger (or the personal computer). Refer to "13. Serial Interface" for the details of UART1. Also, avoid using the UART1 and the functions ($P0_0$/AN_7 and $P3_7$) which share the UART1 pins.

18.4 BRK instruction

The BRK interrupt request is generated. Refer to "10.1 Interrupt overview" and "R8C/Tiny series software manual". Also, avoid using the BRK instruction with using the user system when using the on-chip debugger.

). Usage Notes

19.1 Stop Mode and Wait Mode

19.1.1 Stop Mode

When entering stop mode, set the CM10 bit to "1" (stop mode) after setting the FMR01 bit to "0" (CPU rewrite mode disabled). The instruction queue pre-reads 4 bytes from the instruction which sets the CM10 bit in the CM1 register to "1" (stop mode) and the program stops. Insert at least 4 NOP instructions after inserting the JMP.B instruction immediately after the instruction which sets the CM10 bit to "1".

• Example of entering stop mode

```
Program Example   BCLR    1, FMR0     ; CPU rewrite mode disabled
                  BSET    0, PRCR     ; Protect exited
                  BSET    0, CM1      ; Stop mode
                  JMP.B   LABEL_001
              LABEL_001:
                  NOP
                  NOP
                  NOP
                  NOP
```

19.1.2 Wait Mode

When entering wait mode, execute the WAIT instruction after setting the FMR01 bit to "0" (CPU rewrite mode disabled). The instruction queue pre-reads 4 bytes from the WAIT instruction and the program stops. Insert at least 4 NOP instructions after the WAIT instruction.
Also, the value in the specific internal RAM area may be rewritten when exiting wait mode if writing to the interna RAM area before executing the WAIT instruction and entering wait mode. The area for a maximum of 3 bytes is rewirtten from the following address of the internal RAM in which the writing is performed before the WAIT instruction. If this causes a problem, avoid by inserting the JMP.B instruction between the writing instruction to the internal RAM area and WAIT instruction as shown in the following program example.

• Example to execute WAIT instruction

```
Program Example   MOV.B   #055h,0601h     ; Write to internal RAM area
                  •••
                  JMP.B   LABEL_001
              LABEL_001 :
                  FSET    I               ; Interrupt enabled
                  BCLR    1,FMR0          ; CPU rewrite mode disabled
                  WAIT                    ; Wait mode
                  NOP
                  NOP
                  NOP
                  NOP
```

When accessing any area other than the internal RAM area between the writing instruction to the internal RAM area and execution of the WAIT instruction, this situation will not occur.

19-0122; Rev 5; 2/96

MAXIM

Low-Power, Slew-Rate-Limited RS-485/RS-422 Transceivers

General Description

The MAX481, MAX483, MAX485, MAX487–MAX491, and MAX1487 are low-power transceivers for RS-485 and RS-422 communication. Each part contains one driver and one receiver. The MAX483, MAX487, MAX488, and MAX489 feature reduced slew-rate drivers that minimize EMI and reduce reflections caused by improperly terminated cables, thus allowing error-free data transmission up to 250kbps. The driver slew rates of the MAX481, MAX485, MAX490, MAX491, and MAX1487 are not limited, allowing them to transmit up to 2.5Mbps.

These transceivers draw between 120µA and 500µA of supply current when unloaded or fully loaded with disabled drivers. Additionally, the MAX481, MAX483, and MAX487 have a low-current shutdown mode in which they consume only 0.1µA. All parts operate from a single 5V supply.

Drivers are short-circuit current limited and are protected against excessive power dissipation by thermal shutdown circuitry that places the driver outputs into a high-impedance state. The receiver input has a fail-safe feature that guarantees a logic-high output if the input is open circuit.

The MAX487 and MAX1487 feature quarter-unit-load receiver input impedance, allowing up to 128 MAX487/MAX1487 transceivers on the bus. Full-duplex communications are obtained using the MAX488–MAX491, while the MAX481, MAX483, MAX485, MAX487, and MAX1487 are designed for half-duplex applications.

Applications

Low-Power RS-485 Transceivers

Low-Power RS-422 Transceivers

Level Translators

Transceivers for EMI-Sensitive Applications

Industrial-Control Local Area Networks

Features

- **In µMAX Package: Smallest 8-Pin SO**
- **Slew-Rate Limited for Error-Free Data Transmission (MAX483/487/488/489)**
- **0.1µA Low-Current Shutdown Mode (MAX481/483/487)**
- **Low Quiescent Current:**
  120µA (MAX483/487/488/489)
  230µA (MAX1487)
  300µA (MAX481/485/490/491)
- **-7V to +12V Common-Mode Input Voltage Range**
- **Three-State Outputs**
- **30ns Propagation Delays, 5ns Skew (MAX481/485/490/491/1487)**
- **Full-Duplex and Half-Duplex Versions Available**
- **Operate from a Single 5V Supply**
- **Allows up to 128 Transceivers on the Bus (MAX487/MAX1487)**
- **Current-Limiting and Thermal Shutdown for Driver Overload Protection**

Ordering Information

PART	TEMP. RANGE	PIN-PACKAGE
MAX481CPA	0°C to +70°C	8 Plastic DIP
MAX481CSA	0°C to +70°C	8 SO
MAX481CUA	0°C to +70°C	8 µMAX
MAX481C/D	0°C to +70°C	Dice*

Ordering Information continued at end of data sheet.

** Contact factory for dice specifications.*

Selection Table

PART NUMBER	HALF/FULL DUPLEX	DATA RATE (Mbps)	SLEW-RATE LIMITED	LOW-POWER SHUTDOWN	RECEIVER/ DRIVER ENABLE	QUIESCENT CURRENT (µA)	NUMBER OF TRANSMITTERS ON BUS	PIN COUNT
MAX481	Half	2.5	No	Yes	Yes	300	32	8
MAX483	Half	0.25	Yes	Yes	Yes	120	32	8
MAX485	Half	2.5	No	No	Yes	300	32	8
MAX487	Half	0.25	Yes	Yes	Yes	120	128	8
MAX488	Full	0.25	Yes	No	No	120	32	8
MAX489	Full	0.25	Yes	No	Yes	120	32	14
MAX490	Full	2.5	No	No	No	300	32	8
MAX491	Full	2.5	No	No	Yes	300	32	14
MAX1487	Half	2.5	No	No	Yes	230	128	8

ABSOLUTE MAXIMUM RATINGS

Supply Voltage (V_{CC}) 12V
Control Input Voltage ($\overline{RE}$, DE) ..... -0.5V to (V_{CC} + 0.5V)
Driver Input Voltage (DI) ..... -0.5V to (V_{CC} + 0.5V)
Driver Output Voltage (A, B) ..... -8V to +12.5V
Receiver Input Voltage (A, B) ..... -8V to +12.5V
Receiver Output Voltage (RO) ..... -0.5V to (V_{CC} +0.5V)
Continuous Power Dissipation (T_A = +70°C)
8-Pin Plastic DIP (derate 9.09mW/°C above +70°C) 727mW
14-Pin Plastic DIP (derate 10.00mW/°C above +70°C) ..800mW
8-Pin SO (derate 5.88mW/°C above +70°C) 471mW
14-Pin SO (derate 8.33mW/°C above +70°C) 667mW
8-Pin µMAX (derate 4.1mW/°C above +70°C) 830mW
8-Pin CERDIP (derate 8.00mW/°C above +70°C) 640mW
14-Pin CERDIP (derate 9.09mW/°C above +70°C) 727mW
Operating Temperature Ranges
MAX4_ _C_ _/MAX1487C_ A ..... 0°C to +70°C
MAX4_ _E_ _/MAX1487E_ A ..... -40°C to +85°C
MAX4_ _MJ_/MAX1487MJA ..... -55°C to +125°C
Storage Temperature Range ..... -65°C to +160°C
Lead Temperature (soldering, 10sec) ..... +300°C

Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of the specifications is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

DC ELECTRICAL CHARACTERISTICS

(V_{CC} = 5V ±5%, T_A = T_{MIN} to T_{MAX}, unless otherwise noted.) (Notes 1, 2)

PARAMETER	SYMBOL	CONDITIONS		MIN	TYP	MAX	UNITS
Differential Driver Output (no load)	V_{OD1}					5	V
Differential Driver Output (with load)	V_{OD2}	R = 50Ω (RS-422)		2			V
		R = 27Ω (RS-485), Figure 4		1.5		5	
Change in Magnitude of Driver Differential Output Voltage for Complementary Output States	ΔV_{OD}	R = 27Ω or 50Ω, Figure 4				0.2	V
Driver Common-Mode Output Voltage	V_{OC}	R = 27Ω or 50Ω, Figure 4				3	V
Change in Magnitude of Driver Common-Mode Output Voltage for Complementary Output States	ΔV_{OD}	R = 27Ω or 50Ω, Figure 4				0.2	V
Input High Voltage	V_{IH}	DE, DI, $\overline{RE}$		2.0			V
Input Low Voltage	V_{IL}	DE, DI, $\overline{RE}$				0.8	V
Input Current	I_{IN1}	DE, DI, $\overline{RE}$				±2	µA
Input Current (A, B)	I_{IN2}	DE = 0V; V_{CC} = 0V or 5.25V, all devices except MAX487/MAX1487	V_{IN} = 12V			1.0	mA
			V_{IN} = -7V			-0.8	
		MAX487/MAX1487, DE = 0V, V_{CC} = 0V or 5.25V	V_{IN} = 12V			0.25	mA
			V_{IN} = -7V			-0.2	
Receiver Differential Threshold Voltage	V_{TH}	-7V ≤ V_{CM} ≤ 12V		-0.2		0.2	V
Receiver Input Hysteresis	ΔV_{TH}	V_{CM} = 0V			70		mV
Receiver Output High Voltage	V_{OH}	I_O = -4mA, V_{ID} = 200mV		3.5			V
Receiver Output Low Voltage	V_{OL}	I_O = 4mA, V_{ID} = -200mV				0.4	V
Three-State (high impedance) Output Current at Receiver	I_{OZR}	0.4V ≤ V_O ≤ 2.4V				±1	µA
Receiver Input Resistance	R_{IN}	-7V ≤ V_{CM} ≤ 12V, all devices except MAX487/MAX1487		12			kΩ
		-7V ≤ V_{CM} ≤ 12V, MAX487/MAX1487		48			kΩ

DC ELECTRICAL CHARACTERISTICS (continued)

(V_{CC} = 5V ±5%, T_A = T_{MIN} to T_{MAX}, unless otherwise noted.) (Notes 1, 2)

PARAMETER	SYMBOL	CONDITIONS			MIN	TYP	MAX	UNITS
No-Load Supply Current (Note 3)	I_{CC}	MAX488/MAX489, DE, DI, $\overline{RE}$ = 0V or V_{CC}				120	250	µA
		MAX490/MAX491, DE, DI, $\overline{RE}$ = 0V or V_{CC}				300	500	
		MAX481/MAX485, $\overline{RE}$ = 0V or V_{CC}	DE = V_{CC}			500	900	
			DE = 0V			300	500	
		MAX1487, $\overline{RE}$ = 0V or V_{CC}	DE = V_{CC}			300	500	
			DE = 0V			230	400	
		MAX483/MAX487, $\overline{RE}$ = 0V or V_{CC}	DE = 5V	MAX483		350	650	
				MAX487		250	400	
			DE = 0V			120	250	
Supply Current in Shutdown	I_{SHDN}	MAX481/483/487, DE = 0V, $\overline{RE}$ = V_{CC}				0.1	10	µA
Driver Short-Circuit Current, V_O = High	I_{OSD1}	-7V ≤ V_O ≤12V (Note 4)			35		250	mA
Driver Short-Circuit Current, V_O = Low	I_{OSD2}	-7V ≤ V_O ≤12V (Note 4)			35		250	mA
Receiver Short-Circuit Current	I_{OSR}	0V ≤ V_O ≤ V_{CC}			7		95	mA

SWITCHING CHARACTERISTICS—MAX481/MAX485, MAX490/MAX491, MAX1487

(V_{CC} = 5V ±5%, T_A = T_{MIN} to T_{MAX}, unless otherwise noted.) (Notes 1, 2)

PARAMETER	SYMBOL	CONDITIONS		MIN	TYP	MAX	UNITS
Driver Input to Output	t_{PLH}	Figures 6 and 8, R_{DIFF} = 54Ω, C_{L1} = C_{L2} = 100pF		10	30	60	ns
	t_{PHL}			10	30	60	
Driver Output Skew to Output	t_{SKEW}	Figures 6 and 8, R_{DIFF} = 54Ω, C_{L1} = C_{L2} = 100pF			5	10	ns
Driver Rise or Fall Time	t_R, t_F	Figures 6 and 8, R_{DIFF} = 54Ω, C_{L1} = C_{L2} = 100pF	MAX481, MAX485, MAX1487	3	15	40	ns
			MAX490C/E, MAX491C/E	5	15	25	
			MAX490M, MAX491M	3	15	40	
Driver Enable to Output High	t_{ZH}	Figures 7 and 9, C_L = 100pF, S2 closed			40	70	ns
Driver Enable to Output Low	t_{ZL}	Figures 7 and 9, C_L = 100pF, S1 closed			40	70	ns
Driver Disable Time from Low	t_{LZ}	Figures 7 and 9, C_L = 15pF, S1 closed			40	70	ns
Driver Disable Time from High	t_{HZ}	Figures 7 and 9, C_L = 15pF, S2 closed			40	70	ns
Receiver Input to Output	t_{PLH}, t_{PHL}	Figures 6 and 10, R_{DIFF} = 54Ω, C_{L1} = C_{L2} = 100pF	MAX481, MAX485, MAX1487	20	90	200	ns
			MAX490C/E, MAX491C/E	20	90	150	
			MAX490M, MAX491M	20	90	200	
$\lvert t_{PLH} - t_{PHL} \rvert$ Differential Receiver Skew	t_{SKD}	Figures 6 and 10, R_{DIFF} = 54Ω, C_{L1} = C_{L2} = 100pF			13		ns
Receiver Enable to Output Low	t_{ZL}	Figures 5 and 11, C_{RL} = 15pF, S1 closed			20	50	ns
Receiver Enable to Output High	t_{ZH}	Figures 5 and 11, C_{RL} = 15pF, S2 closed			20	50	ns
Receiver Disable Time from Low	t_{LZ}	Figures 5 and 11, C_{RL} = 15pF, S1 closed			20	50	ns
Receiver Disable Time from High	t_{HZ}	Figures 5 and 11, C_{RL} = 15pF, S2 closed			20	50	ns
Maximum Data Rate	f_{MAX}			2.5			Mbps
Time to Shutdown	t_{SHDN}	MAX481 (Note 5)		50	200	600	ns

SWITCHING CHARACTERISTICS—MAX481/MAX485, MAX490/MAX491, MAX1487 (continued)

(V_{CC} = 5V ±5%, T_A = T_{MIN} to T_{MAX}, unless otherwise noted.) (Notes 1, 2)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Driver Enable from Shutdown to Output High (MAX481)	$t_{ZH(SHDN)}$	Figures 7 and 9, C_L = 100pF, S2 closed		40	100	ns
Driver Enable from Shutdown to Output Low (MAX481)	$t_{ZL(SHDN)}$	Figures 7 and 9, C_L = 100pF, S1 closed		40	100	ns
Receiver Enable from Shutdown to Output High (MAX481)	$t_{ZH(SHDN)}$	Figures 5 and 11, C_L = 15pF, S2 closed, A - B = 2V		300	1000	ns
Receiver Enable from Shutdown to Output Low (MAX481)	$t_{ZL(SHDN)}$	Figures 5 and 11, C_L = 15pF, S1 closed, B - A = 2V		300	1000	ns

SWITCHING CHARACTERISTICS—MAX483, MAX487/MAX488/MAX489

(V_{CC} = 5V ±5%, T_A = T_{MIN} to T_{MAX}, unless otherwise noted.) (Notes 1, 2)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Driver Input to Output	t_{PLH}	Figures 6 and 8, R_{DIFF} = 54Ω, C_{L1} = C_{L2} = 100pF	250	800	2000	ns
	t_{PHL}		250	800	2000	
Driver Output Skew to Output	t_{SKEW}	Figures 6 and 8, R_{DIFF} = 54Ω, C_{L1} = C_{L2} = 100pF		100	800	ns
Driver Rise or Fall Time	t_R, t_F	Figures 6 and 8, R_{DIFF} = 54Ω, C_{L1} = C_{L2} = 100pF	250		2000	ns
Driver Enable to Output High	t_{ZH}	Figures 7 and 9, C_L = 100pF, S2 closed	250		2000	ns
Driver Enable to Output Low	t_{ZL}	Figures 7 and 9, C_L = 100pF, S1 closed	250		2000	ns
Driver Disable Time from Low	t_{LZ}	Figures 7 and 9, C_L = 15pF, S1 closed	300		3000	ns
Driver Disable Time from High	t_{HZ}	Figures 7 and 9, C_L = 15pF, S2 closed	300		3000	ns
Receiver Input to Output	t_{PLH}	Figures 6 and 10, R_{DIFF} = 54Ω, C_{L1} = C_{L2} = 100pF	250		2000	ns
	t_{PHL}		250		2000	
| t_{PLH} - t_{PHL} | Differential Receiver Skew	t_{SKD}	Figures 6 and 10, R_{DIFF} = 54Ω, C_{L1} = C_{L2} = 100pF		100		ns
Receiver Enable to Output Low	t_{ZL}	Figures 5 and 11, C_{RL} = 15pF, S1 closed		20	50	ns
Receiver Enable to Output High	t_{ZH}	Figures 5 and 11, C_{RL} = 15pF, S2 closed		20	50	ns
Receiver Disable Time from Low	t_{LZ}	Figures 5 and 11, C_{RL} = 15pF, S1 closed		20	50	ns
Receiver Disable Time from High	t_{HZ}	Figures 5 and 11, C_{RL} = 15pF, S2 closed		20	50	ns
Maximum Data Rate	f_{MAX}	t_{PLH}, t_{PHL} < 50% of data period	250			kbps
Time to Shutdown	t_{SHDN}	MAX483/MAX487 (Note 5)	50	200	600	ns
Driver Enable from Shutdown to Output High	$t_{ZH(SHDN)}$	MAX483/MAX487, Figures 7 and 9, C_L = 100pF, S2 closed			2000	ns
Driver Enable from Shutdown to Output Low	$t_{ZL(SHDN)}$	MAX483/MAX487, Figures 7 and 9, C_L = 100pF, S1 closed			2000	ns
Receiver Enable from Shutdown to Output High	$t_{ZH(SHDN)}$	MAX483/MAX487, Figures 5 and 11, C_L = 15pF, S2 closed			2500	ns
Receiver Enable from Shutdown to Output Low	$t_{ZL(SHDN)}$	MAX483/MAX487, Figures 5 and 11, C_L = 15pF, S1 closed			2500	ns

NOTES FOR ELECTRICAL/SWITCHING CHARACTERISTICS

Note 1: All currents into device pins are positive; all currents out of device pins are negative. All voltages are referenced to device ground unless otherwise specified.

Note 2: All typical specifications are given for V_{CC} = 5V and T_A = +25°C.

Note 3: Supply current specification is valid for loaded transmitters when DE = 0V.

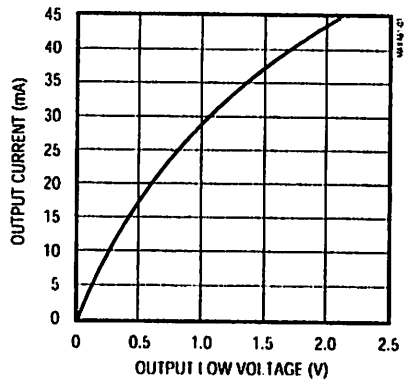
Note 4: Applies to peak current. See *Typical Operating Characteristics.*

Note 5: The MAX481/MAX483/MAX487 are put into shutdown by bringing $\overline{RE}$ high and DE low. If the inputs are in this state for less than 50ns, the parts are guaranteed not to enter shutdown. If the inputs are in this state for at least 600ns, the parts are guaranteed to have entered shutdown. See *Low-Power Shutdown Mode* section.

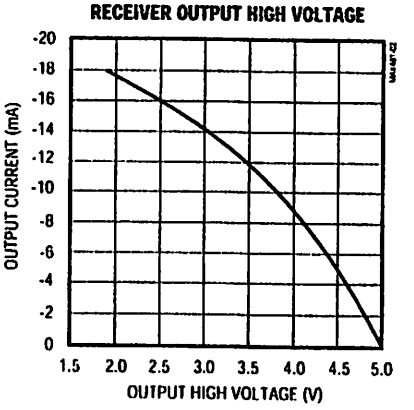
Typical Operating Characteristics

(V_{CC} = 5V, T_A = +25°C, unless otherwise noted.)

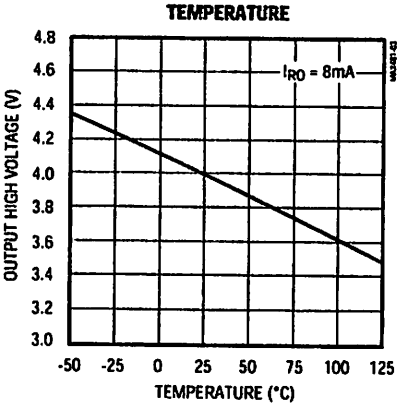
OUTPUT CURRENT vs. RECEIVER OUTPUT LOW VOLTAGE



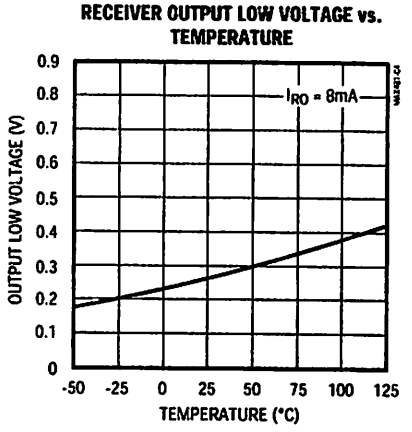
OUTPUT CURRENT vs. RECEIVER OUTPUT HIGH VOLTAGE



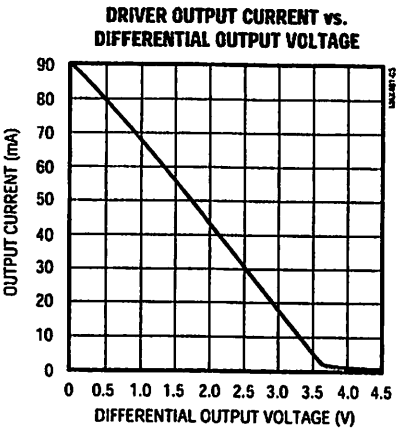
RECEIVER OUTPUT HIGH VOLTAGE vs. TEMPERATURE



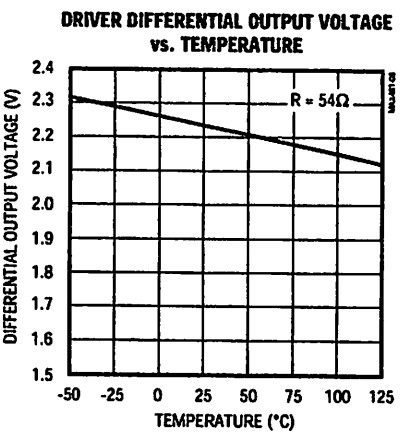
RECEIVER OUTPUT LOW VOLTAGE vs. TEMPERATURE



DRIVER OUTPUT CURRENT vs. DIFFERENTIAL OUTPUT VOLTAGE



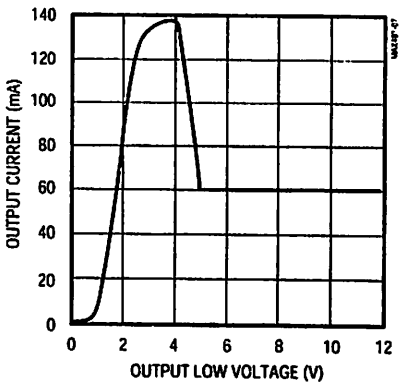
DRIVER DIFFERENTIAL OUTPUT VOLTAGE vs. TEMPERATURE



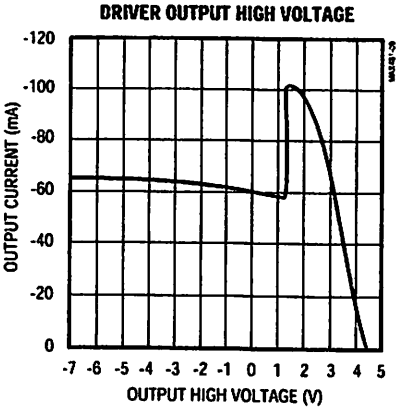
Typical Operating Characteristics (continued)

(V_{CC} = 5V, T_A = +25°C, unless otherwise noted.)

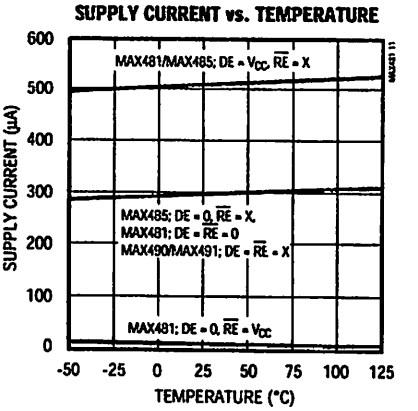
OUTPUT CURRENT vs. DRIVER OUTPUT LOW VOLTAGE



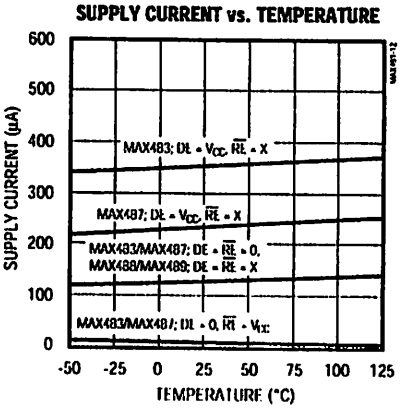
OUTPUT CURRENT vs. DRIVER OUTPUT HIGH VOLTAGE



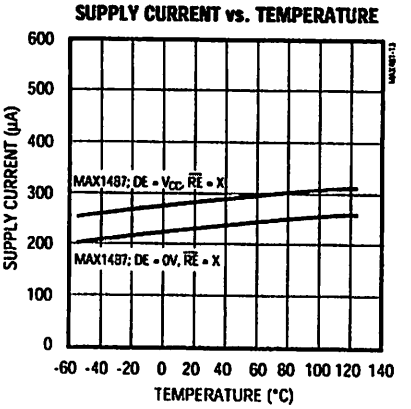
MAX481/MAX485/MAX490/MAX491 SUPPLY CURRENT vs. TEMPERATURE



MAX483/MAX487–MAX489 SUPPLY CURRENT vs. TEMPERATURE



MAX1487 SUPPLY CURRENT vs. TEMPERATURE



Pin Description

PIN					NAME	FUNCTION
MAX481/MAX483/ MAX485/MAX487/ MAX1487		MAX488/ MAX490		MAX489/ MAX491		
DIP/SO	µMAX	DIP/SO	µMAX	DIP/SO		
1	3	2	4	2	RO	Receiver Output: If A > B by 200mV, RO will be high; If A < B by 200mV, RO will be low.
2	4	—	—	3	$\overline{RE}$	Receiver Output Enable. RO is enabled when $\overline{RE}$ is low; RO is high impedance when $\overline{RE}$ is high.
3	5	—	—	4	DE	Driver Output Enable. The driver outputs, Y and Z, are enabled by bringing DE high. They are high impedance when DE is low. If the driver outputs are enabled, the parts function as line drivers. While they are high impedance, they function as line receivers if $\overline{RE}$ is low.
4	6	3	5	5	DI	Driver Input. A low on DI forces output Y low and output Z high. Similarly, a high on DI forces output Y high and output Z low.
5	7	4	6	6, 7	GND	Ground
—	—	5	7	9	Y	Noninverting Driver Output
—	—	6	8	10	Z	Inverting Driver Output
6	8	—	—	—	A	Noninverting Receiver Input and Noninverting Driver Output
—	—	8	2	12	A	Noninverting Receiver Input
7	1	—	—	—	B	Inverting Receiver Input and Inverting Driver Output
—	—	7	1	11	B	Inverting Receiver Input
8	2	1	3	14	VCC	Positive Supply: 4.75V ≤ VCC ≤ 5.25V
—	—	—	—	1, 8, 13	N.C.	No Connect—not internally connected

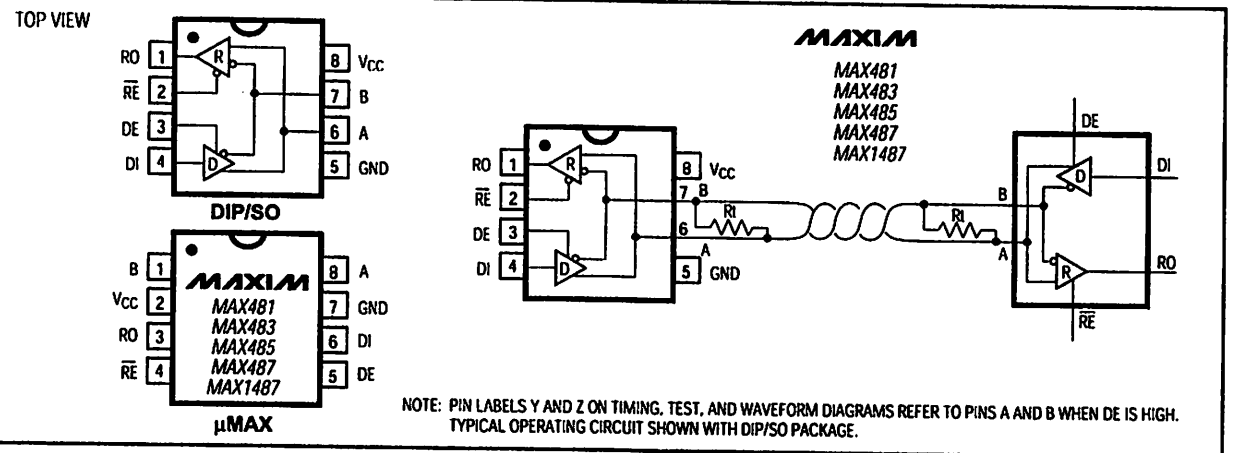


Figure 1. MAX481/MAX483/MAX485/MAX487/MAX1487 Pin Configuration and Typical Operating Circuit

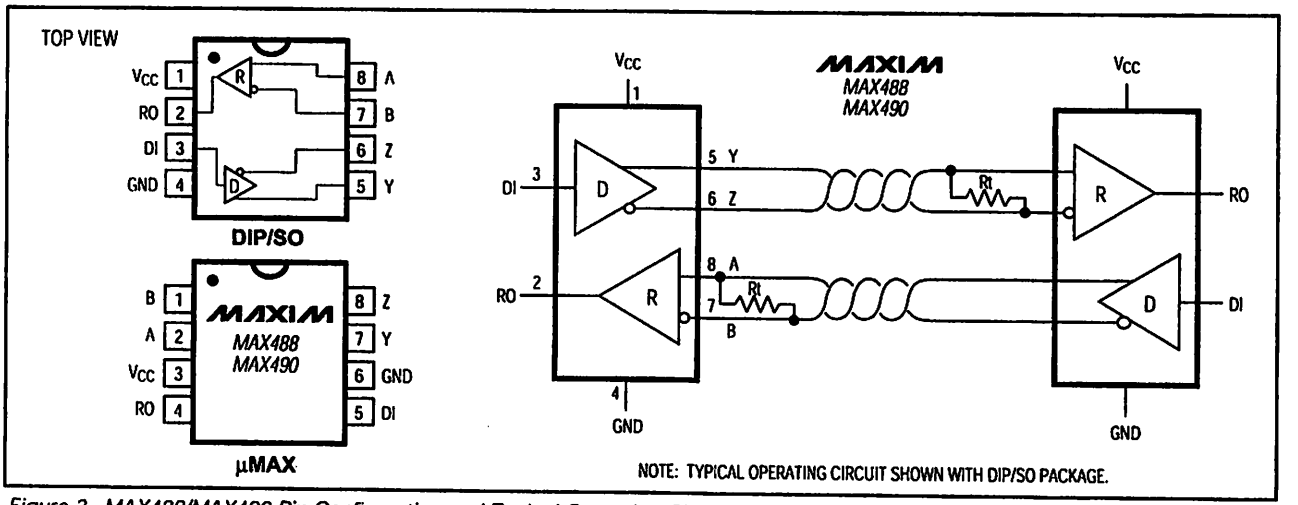


Figure 2. MAX488/MAX490 Pin Configuration and Typical Operating Circuit

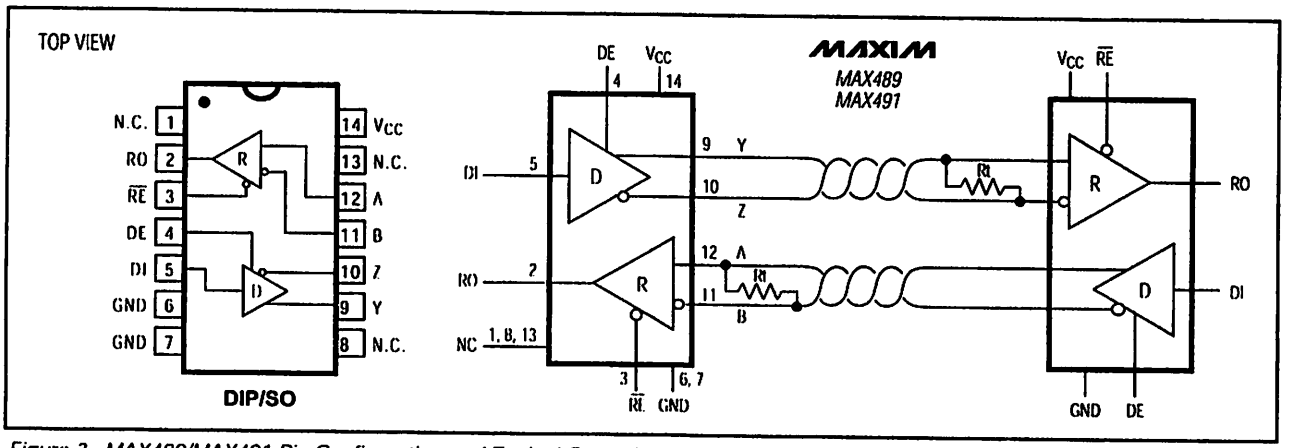


Figure 3. MAX489/MAX491 Pin Configuration and Typical Operating Circuit

Applications Information

The MAX481/MAX483/MAX485/MAX487–MAX491 and MAX1487 are low-power transceivers for RS-485 and RS-422 communications. The MAX481, MAX485, MAX490, MAX491, and MAX1487 can transmit and receive at data rates up to 2.5Mbps, while the MAX483, MAX487, MAX488, and MAX489 are specified for data rates up to 250kbps. The MAX488–MAX491 are full-duplex transceivers while the MAX481, MAX483, MAX485, MAX487, and MAX1487 are half-duplex. In addition, Driver Enable (DE) and Receiver Enable ($\overline{RE}$) pins are included on the MAX481, MAX483, MAX485, MAX487, MAX489, MAX491, and MAX1487. When disabled, the driver and receiver outputs are high impedance.

MAX487/MAX1487: 128 Transceivers on the Bus

The 48kΩ, 1/4-unit-load receiver input impedance of the MAX487 and MAX1487 allows up to 128 transceivers on a bus, compared to the 1-unit load (12kΩ input impedance) of standard RS-485 drivers (32 transceivers maximum). Any combination of MAX487/MAX1487 and other RS-485 transceivers with a total of 32 unit loads or less can be put on the bus. The MAX481/MAX483/MAX485 and MAX488–MAX491 have standard 12kΩ Receiver Input impedance.

Test Circuits

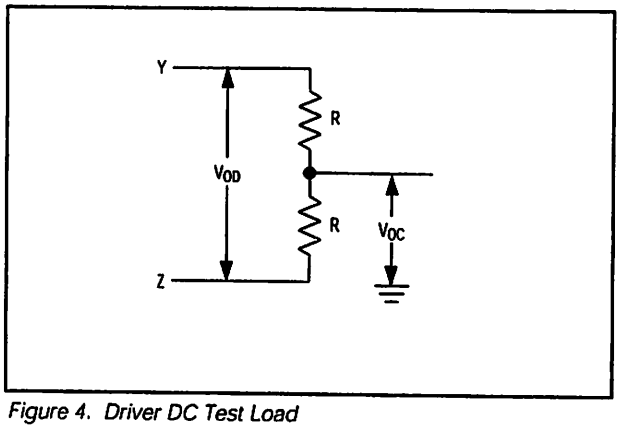


Figure 4. Driver DC Test Load

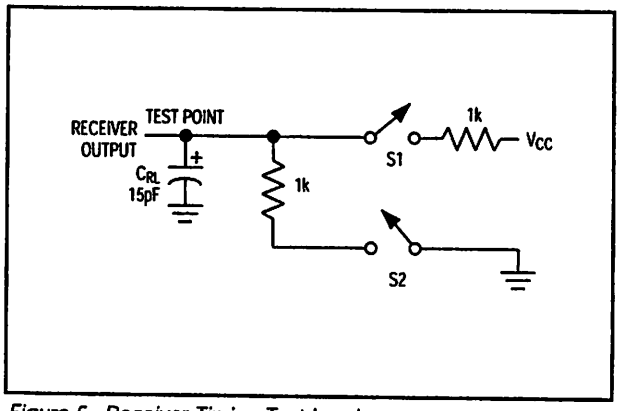


Figure 5. Receiver Timing Test Load

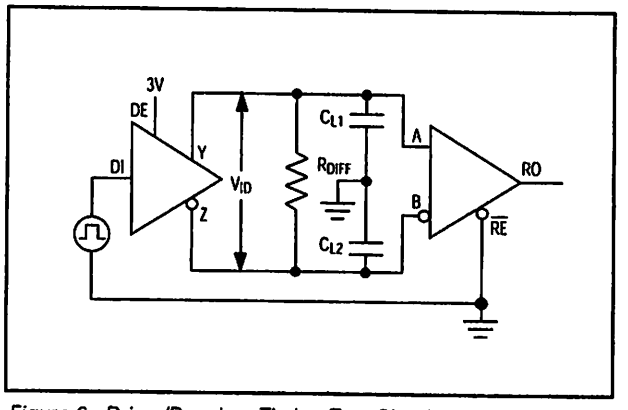


Figure 6. Driver/Receiver Timing Test Circuit

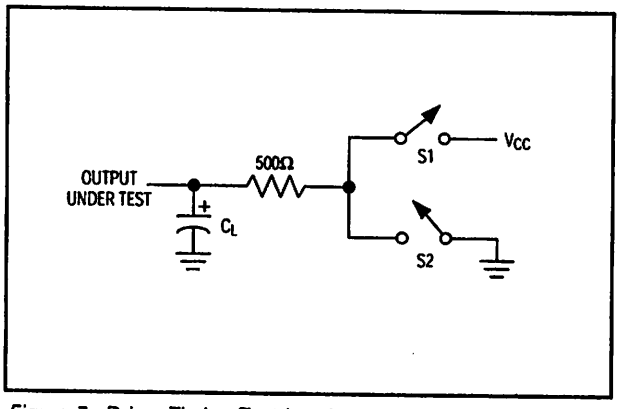


Figure 7. Driver Timing Test Load

MAX483/MAX487/MAX488/MAX489: Reduced EMI and Reflections

The MAX483 and MAX487–MAX489 are slew-rate limited, minimizing EMI and reducing reflections caused by improperly terminated cables. Figure 12 shows the driver output waveform and its Fourier analysis of a 150kHz signal transmitted by a MAX481, MAX485, MAX490, MAX491, or MAX1487. High-frequency harmonics with large amplitudes are evident. Figure 13 shows the same information displayed for a MAX483, MAX487, MAX488, or MAX489 transmitting under the same conditions. Figure 13's high-frequency harmonics have much lower amplitudes, and the potential for EMI is significantly reduced.

Switching Waveforms

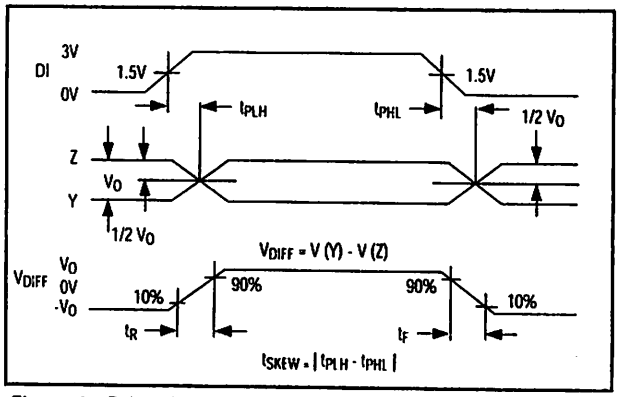


Figure 8. Driver Propagation Delays

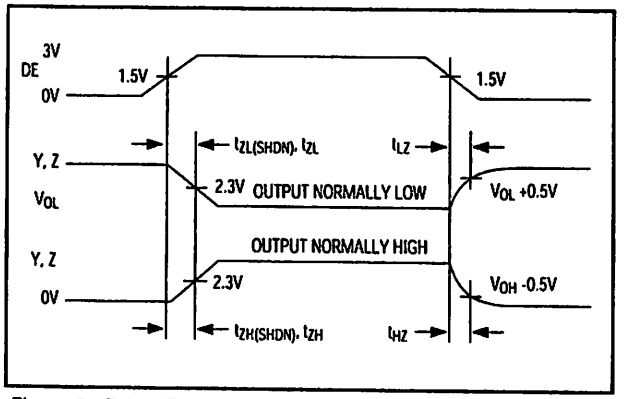


Figure 9. Driver Enable and Disable Times (except MAX488 and MAX490)

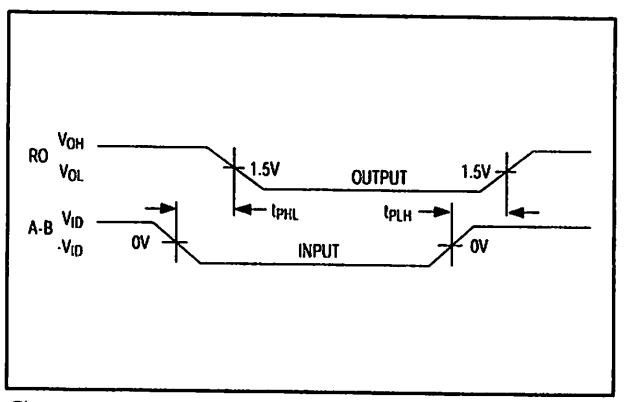


Figure 10. Receiver Propagation Delays

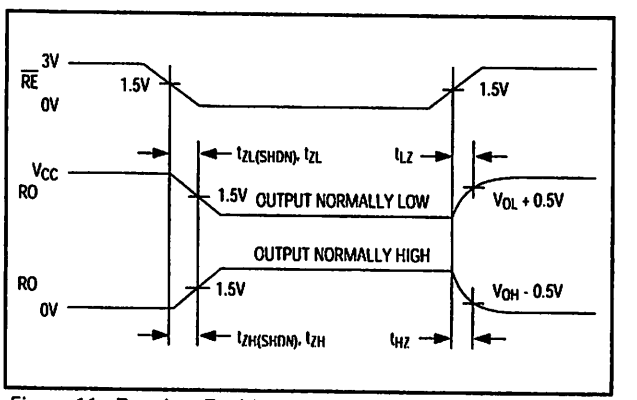


Figure 11. Receiver Enable and Disable Times (except MAX488 and MAX490)

Function Tables (MAX481/MAX483/MAX485/MAX487/MAX1487)

Table 1. Transmitting

INPUTS			OUTPUTS	
$\overline{RE}$	DE	DI	Z	Y
X	1	1	0	1
X	1	0	1	0
0	0	X	High-Z	High-Z
1	0	X	High-Z*	High-Z*

X = Don't care
High-Z = High impedance
* Shutdown mode for MAX481/MAX483/MAX487

Table 2. Receiving

INPUTS			OUTPUT
$\overline{RE}$	DE	A-B	RO
0	0	≥ +0.2V	1
0	0	≤ -0.2V	0
0	0	Inputs open	1
1	0	X	High-Z*

X = Don't care
High-Z = High impedance
* Shutdown mode for MAX481/MAX483/MAX487

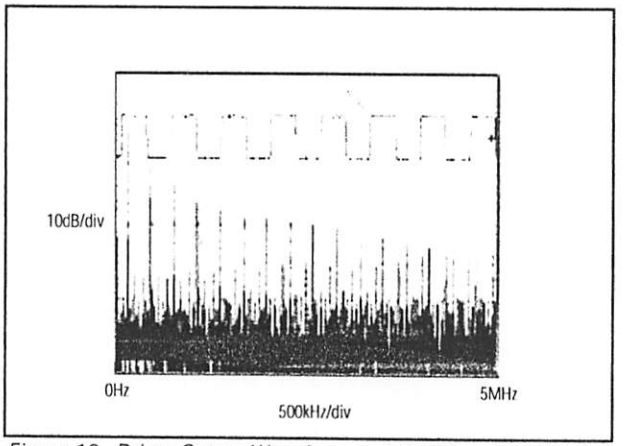


Figure 12. Driver Output Waveform and FFT Plot of MAX481/MAX485/MAX490/MAX491/MAX1487 Transmitting a 150kHz Signal

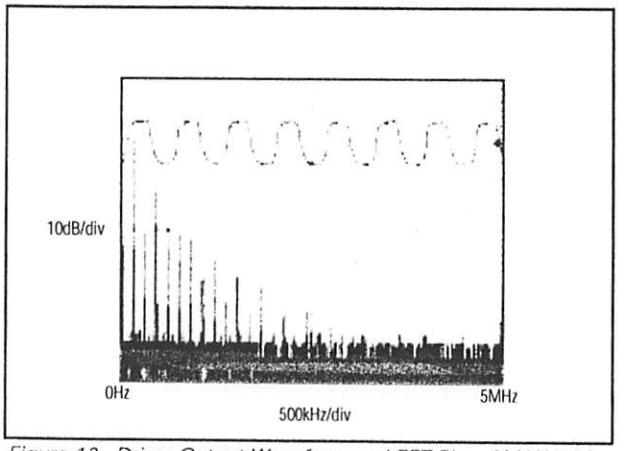


Figure 13. Driver Output Waveform and FFT Plot of MAX483/MAX487–MAX489 Transmitting a 150kHz Signal

Low-Power Shutdown Mode (MAX481/MAX483/MAX487)

A low-power shutdown mode is initiated by bringing both $\overline{RE}$ high and DE low. The devices will not shut down unless both the driver and receiver are disabled. In shutdown, the devices typically draw only 0.1µA of supply current.

$\overline{RE}$ and DE may be driven simultaneously; the parts are guaranteed not to enter shutdown if $\overline{RE}$ is high and DE is low for less than 50ns. If the inputs are in this state for at least 600ns, the parts are guaranteed to enter shutdown.

For the MAX481, MAX483, and MAX487, the t_{ZH} and t_{ZL} enable times assume the part was not in the low-power shutdown state (the MAX485/MAX488–MAX491 and MAX1487 can not be shut down). The $t_{ZH(SHDN)}$ and $t_{ZL(SHDN)}$ enable times assume the parts were shut down (see *Electrical Characteristics*).

It takes the drivers and receivers longer to become enabled from the low-power shutdown state ($t_{ZH(SHDN)}$, $t_{ZL(SHDN)}$) than from the operating mode (t_{ZH}, t_{ZL}). (The parts are in operating mode if the $\overline{RE}$, DE inputs equal a logical 0,1 or 1,1 or 0, 0.)

Driver Output Protection

Excessive output current and power dissipation caused by faults or by bus contention are prevented by two mechanisms. A foldback current limit on the output stage provides immediate protection against short circuits over the whole common-mode voltage range (see *Typical Operating Characteristics*). In addition, a thermal shutdown circuit forces the driver outputs into a high-impedance state if the die temperature rises excessively.

Propagation Delay

Many digital encoding schemes depend on the difference between the driver and receiver propagation delay times. Typical propagation delays are shown in Figures 15–18 using Figure 14's test circuit.

The difference in receiver delay times, $|t_{PLH} - t_{PHL}|$, is typically under 13ns for the MAX481, MAX485, MAX490, MAX491, and MAX1487 and is typically less than 100ns for the MAX483 and MAX487–MAX489.

The driver skew times are typically 5ns (10ns max) for the MAX481, MAX485, MAX490, MAX491, and MAX1487, and are typically 100ns (800ns max) for the MAX483 and MAX487–MAX489.

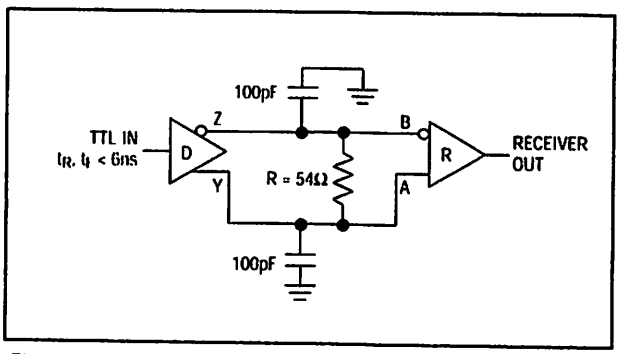


Figure 14. Receiver Propagation Delay Test Circuit

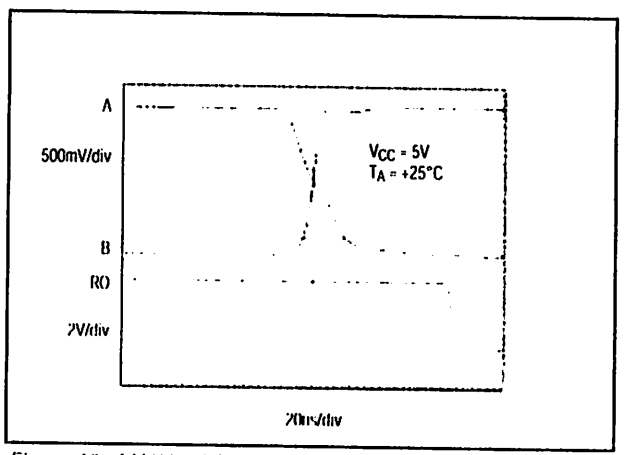


Figure 15. MAX481/MAX485/MAX490/MAX491/MAX1487 Receiver tPHL

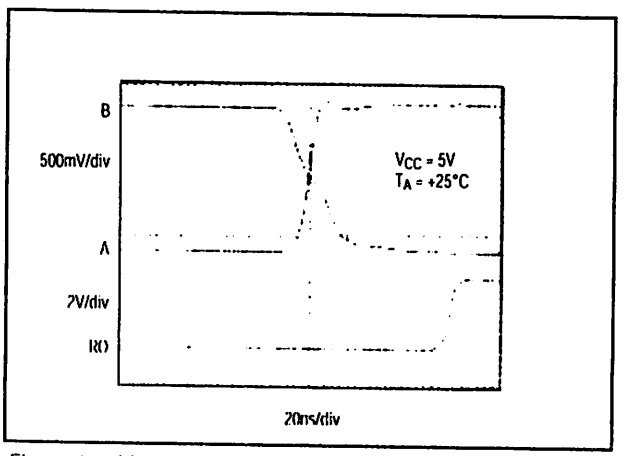


Figure 16. MAX481/MAX485/MAX490/MAX491/MAX1487 Receiver tPLH

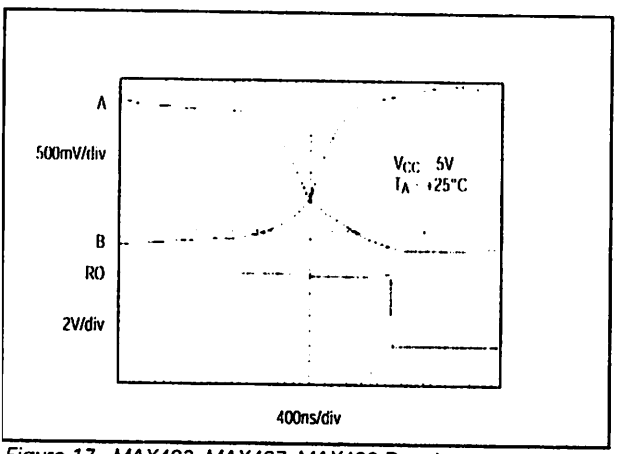


Figure 17. MAX483, MAX487–MAX489 Receiver tPHL

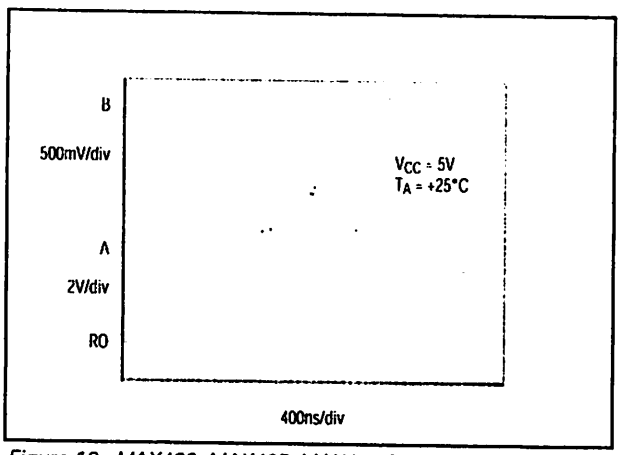


Figure 18. MAX483, MAX487–MAX489 Receiver tPLH

Line Length vs. Data Rate

The RS-485/RS-422 standard covers line lengths up to 4000 feet. For line lengths greater than 4000 feet, see Figure 23.

Figures 19 and 20 show the system differential voltage for the parts driving 4000 feet of 26AWG twisted-pair wire at 110kHz into 120Ω loads.

Typical Applications

The MAX481, MAX483, MAX485, MAX487–MAX491, and MAX1487 transceivers are designed for bidirectional data communications on multipoint bus transmission lines.

Figures 21 and 22 show typical network applications circuits. These parts can also be used as line repeaters, with cable lengths longer than 4000 feet, as shown in Figure 23.

To minimize reflections, the line should be terminated at both ends in its characteristic impedance, and stub lengths off the main line should be kept as short as possible. The slew-rate-limited MAX483 and MAX487–MAX489 are more tolerant of imperfect termination.

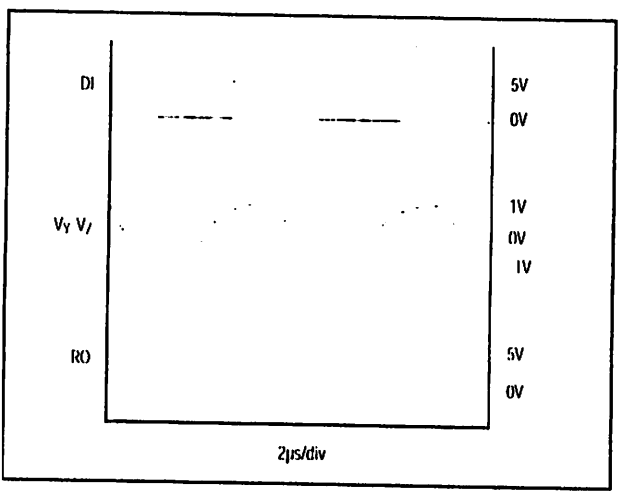


Figure 19. MAX481/MAX485/MAX490/MAX491/MAX1487 System Differential Voltage at 110kHz Driving 4000ft of Cable

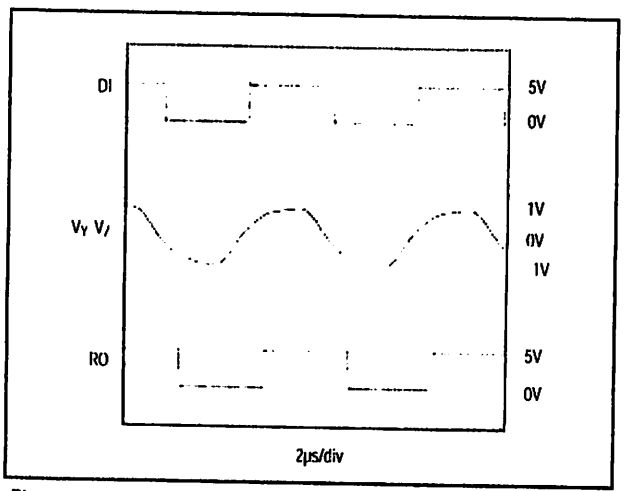


Figure 20. MAX483, MAX487–MAX489 System Differential Voltage at 110kHz Driving 4000ft of Cable

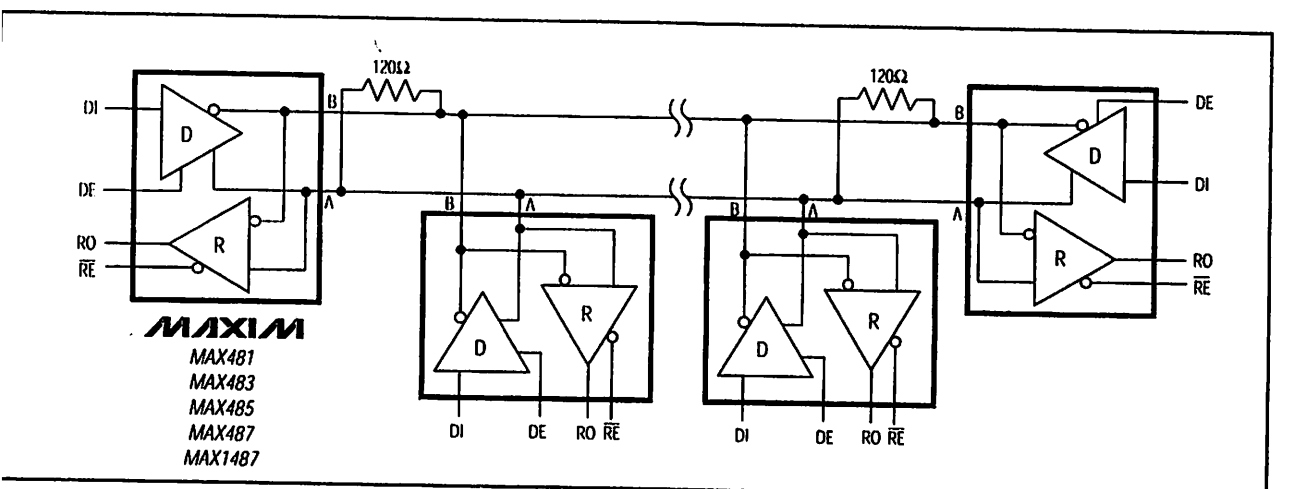


Figure 21. MAX481/MAX483/MAX485/MAX487/MAX1487 Typical Half-Duplex RS-485 Network

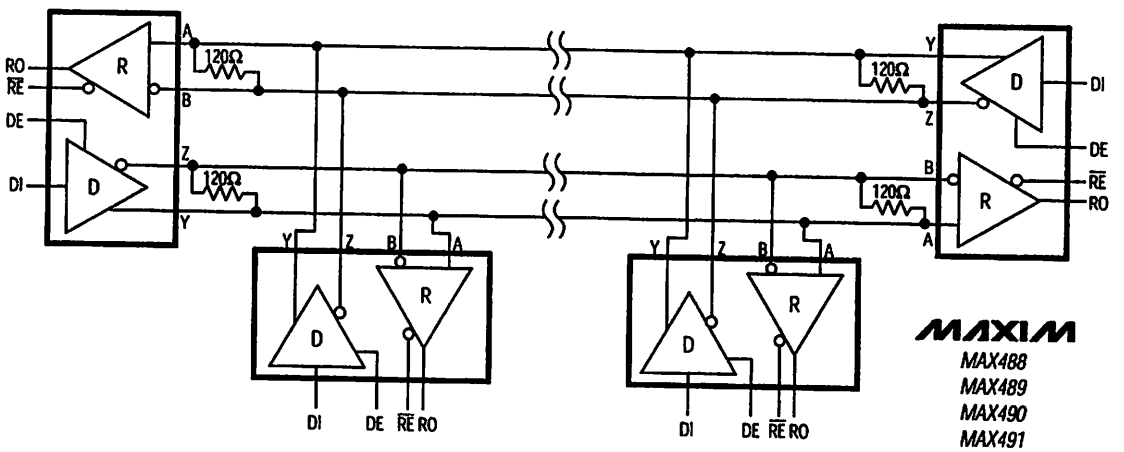


Figure 22. MAX488–MAX491 Full-Duplex RS-485 Network

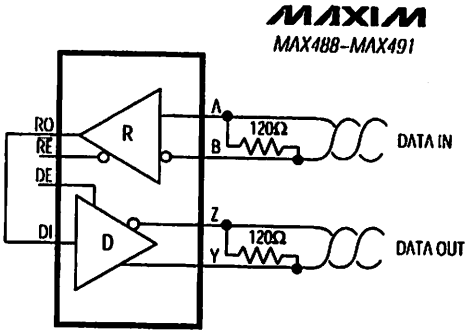


Figure 23. Line Repeater for MAX488–MAX491

Isolated RS-485

For isolated RS-485 applications, see the MAX253 and MAX1480 data sheets.

Low-Power, Slew-Rate-Limited RS-485/RS-422 Transceivers

Ordering Information (continued)

PART	TEMP. RANGE	PIN-PACKAGE
MAX481EPA	-40°C to +85°C	8 Plastic DIP
MAX481ESA	-40°C to +85°C	8 SO
MAX481MJA	-55°C to +125°C	8 CERDIP
MAX483CPA	0°C to +70°C	8 Plastic DIP
MAX483CSA	0°C to +70°C	8 SO
MAX483CUA	0°C to +70°C	8 µMAX
MAX483C/D	0°C to +70°C	Dice*
MAX483EPA	-40°C to +85°C	8 Plastic DIP
MAX483ESA	-40°C to +85°C	8 SO
MAX483MJA	-55°C to +125°C	8 CERDIP
MAX485CPA	0°C to +70°C	8 Plastic DIP
MAX485CSA	0°C to +70°C	8 SO
MAX485CUA	0°C to +70°C	8 µMAX
MAX485C/D	0°C to +70°C	Dice*
MAX485EPA	-40°C to +85°C	8 Plastic DIP
MAX485ESA	-40°C to +85°C	8 SO
MAX485MJA	-55°C to +125°C	8 CERDIP
MAX487CPA	0°C to +70°C	8 Plastic DIP
MAX487CSA	0°C to +70°C	8 SO
MAX487CUA	0°C to +70°C	8 µMAX
MAX487C/D	0°C to +70°C	Dice*
MAX487EPA	-40°C to +85°C	8 Plastic DIP
MAX487ESA	-40°C to +85°C	8 SO
MAX487MJA	-55°C to +125°C	8 CERDIP
MAX488CPA	0°C to +70°C	8 Plastic DIP
MAX488CSA	0°C to +70°C	8 SO
MAX488CUA	0°C to +70°C	8 µMAX
MAX488C/D	0°C to +70°C	Dice*
MAX488EPA	-40°C to +85°C	8 Plastic DIP
MAX488ESA	-40°C to +85°C	8 SO
MAX488MJA	-55°C to +125°C	8 CERDIP
MAX489CPD	0°C to +70°C	14 Plastic DIP
MAX489CSD	0°C to +70°C	14 SO
MAX489C/D	0°C to +70°C	Dice*
MAX489EPD	-40°C to +85°C	14 Plastic DIP
MAX489ESD	-40°C to +85°C	14 SO
MAX489MJD	-55°C to +125°C	14 CERDIP

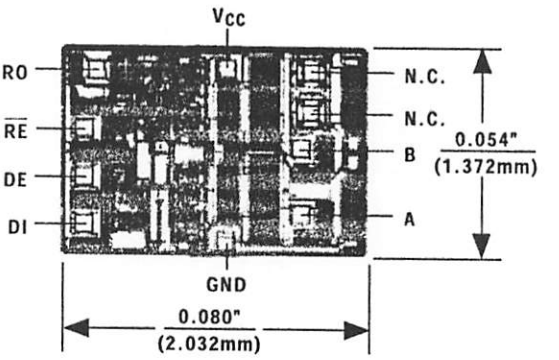
Ordering Information (continued)

PART	TEMP. RANGE	PIN-PACKAGE
MAX490CPA	0°C to +70°C	8 Plastic DIP
MAX490CSA	0°C to +70°C	8 SO
MAX490CUA	0°C to +70°C	8 µMAX
MAX490C/D	0°C to +70°C	Dice*
MAX490EPA	-40°C to +85°C	8 Plastic DIP
MAX490ESA	-40°C to +85°C	8 SO
MAX490MJA	-55°C to +125°C	8 CERDIP
MAX491CPD	0°C to +70°C	14 Plastic DIP
MAX491CSD	0°C to +70°C	14 SO
MAX491C/D	0°C to +70°C	Dice*
MAX491EPD	-40°C to +85°C	14 Plastic DIP
MAX491ESD	-40°C to +85°C	14 SO
MAX491MJD	-55°C to +125°C	14 CERDIP
MAX1487CPA	0°C to +70°C	8 Plastic DIP
MAX1487CSA	0°C to +70°C	8 SO
MAX1487CUA	0°C to +70°C	8 µMAX
MAX1487C/D	0°C to +70°C	Dice*
MAX1487EPA	-40°C to +85°C	8 Plastic DIP
MAX1487ESA	-40°C to +85°C	8 SO
MAX1487MJA	-55°C to +125°C	8 CERDIP

* *Contact factory for dice specifications.*

Chip Topographies

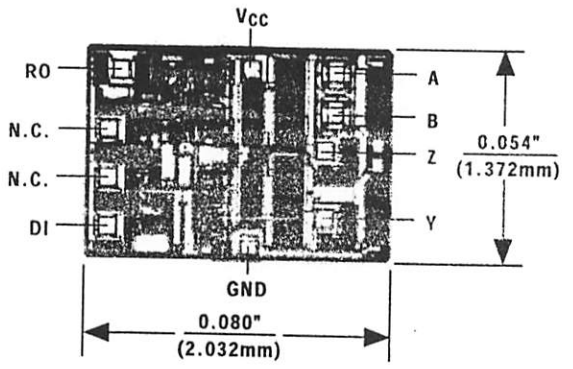
MAX481/MAX483/MAX485/MAX487/MAX1487



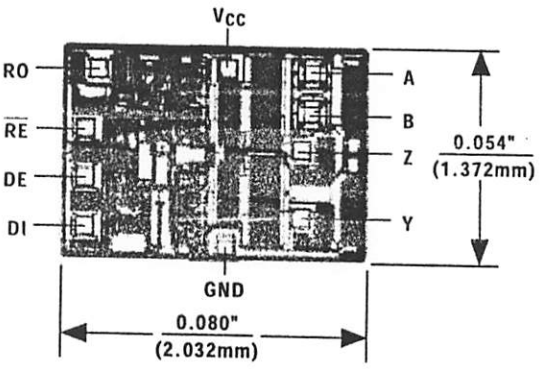
Low-Power, Slew-Rate-Limited RS-485/RS-422 Transceivers

Chip Topographies (continued)

MAX488/MAX490

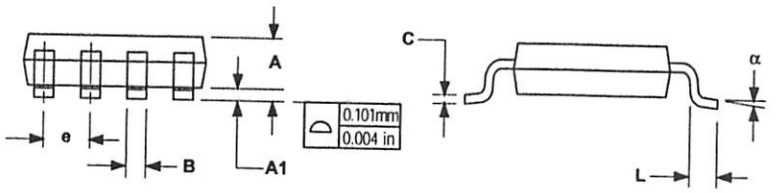


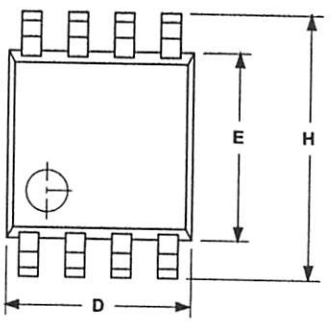
MAX489/MAX491



TRANSISTOR COUNT: 248
SUBSTRATE CONNECTED TO GND

Package Information





DIM	INCHES		MILLIMETERS	
	MIN	MAX	MIN	MAX
A	0.036	0.044	0.91	1.11
A1	0.004	0.008	0.10	0.20
B	0.010	0.014	0.25	0.36
C	0.005	0.007	0.13	0.18
D	0.116	0.120	2.95	3.05
E	0.116	0.120	2.95	3.05
e	0.0256		0.65	
H	0.188	0.198	4.78	5.03
L	0.016	0.026	0.41	0.66
α	0°	6°	0°	6°

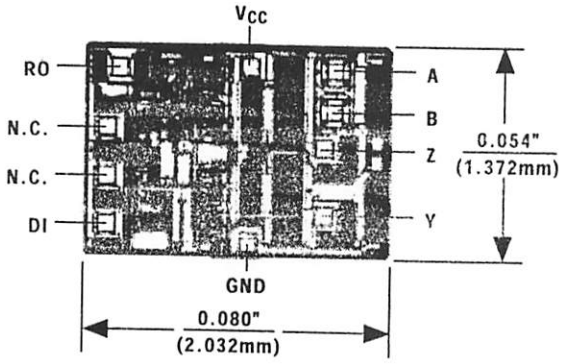
21-0036D

8-PIN μMAX
MICROMAX SMALL-OUTLINE
PACKAGE

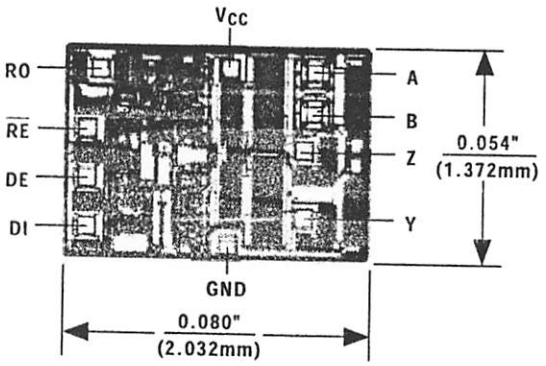
Low-Power, Slew-Rate-Limited RS-485/RS-422 Transceivers

Chip Topographies (continued)

MAX488/MAX490



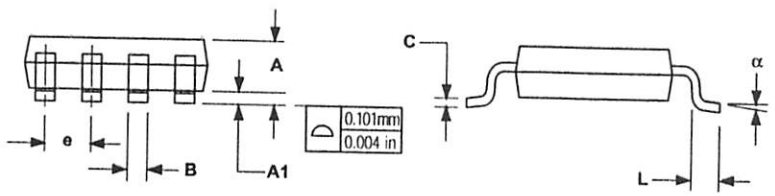
MAX489/MAX491

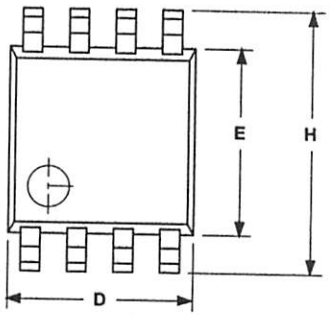


TRANSISTOR COUNT: 248

SUBSTRATE CONNECTED TO GND

Package Information





DIM	INCHES		MILLIMETERS	
	MIN	MAX	MIN	MAX
A	0.036	0.044	0.91	1.11
A1	0.004	0.008	0.10	0.20
B	0.010	0.014	0.25	0.36
C	0.005	0.007	0.13	0.18
D	0.116	0.120	2.95	3.05
E	0.116	0.120	2.95	3.05
e	0.0256		0.65	
H	0.188	0.198	4.78	5.03
L	0.016	0.026	0.41	0.66
α	0°	6°	0°	6°

21-0036D

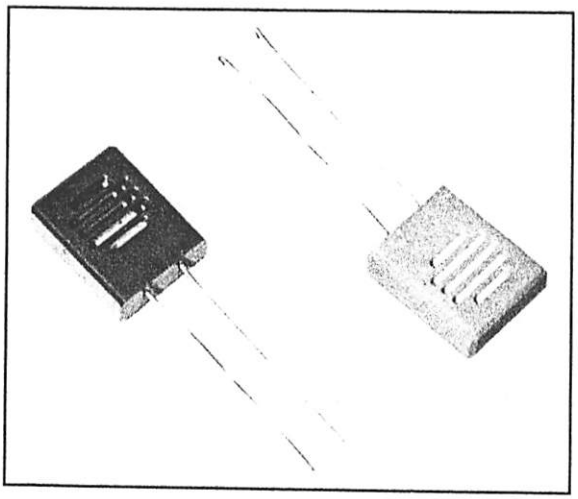
8-PIN μMAX
MICROMAX SMALL-OUTLINE
PACKAGE

HUMIDITY SENSORS: TYPE HS12P, HS15P

RELATIVE HUMIDITY SENSOR

DESCRIPTION:

Non-refresh type humidity sensor made of polymer.



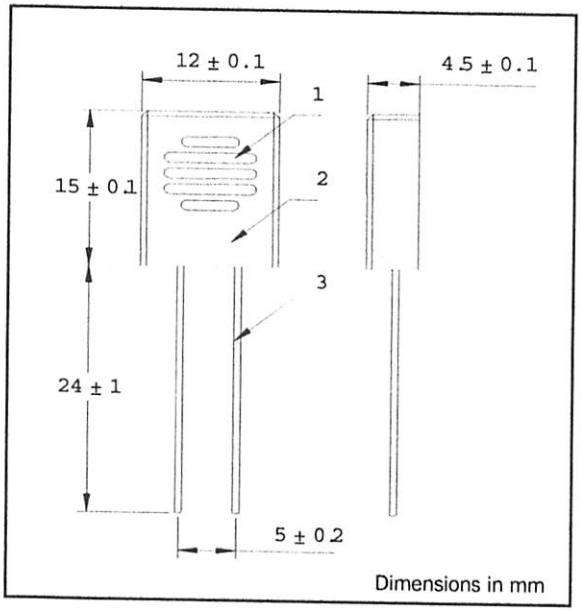
FEATURES:

- Good long term reliability
- Cost effective performance
- HS15P water resistive
- Typical applications include humidity monitors, humidity controllers, air conditioners, humidifiers, dehumidifiers, automatic ventilation

DATA:

CODE	HS12P	HS15P
Operating Temperature	0 to 50°C	
Operating Humidity	20 to 90% RH (without condensing)	20 to 100% RH
Impedance at 25°C 50% RH	60 kohm +/-30 kohm (+/-5% RH)	
Rated Voltage	AC 1 V rms	
Rated Frequency	50 Hz to 1 kHz	
Consumption Power	0.3 mW	

DIMENSIONS:



CODING:

HS12P: quick response type
HS15P: water resistant type

WARNING:

Use only within the specified limits.
Do not disassemble and change any parts.
Do not apply DC voltage or DC bias.
Do not immerse into water or any solution.

1. Filter
2. Case ..... ABS: Dark blue (HS12P)
..... Light blue (HS15P)
3. Lead wire ..... Sn-Pb plated Cu
..... Diameter: 0.6mm

rown Industrial Estate, Priorswood Rd
aunton, Somerset TA2 8QY UK
el +44 (0)1823 335200
ax +44 (0)1823 332637

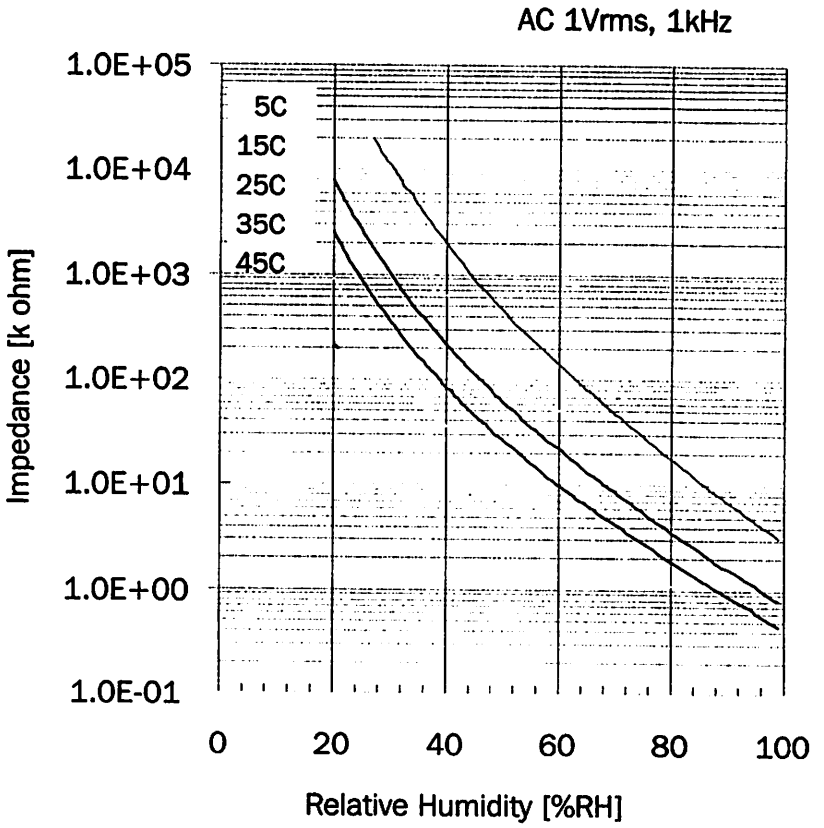
808 US Highway 1
Edison, New Jersey 08817-4695 USA
Tel +1 (732) 287 2870
Fax +1 (732) 287 8847

967 Windfall Road
St Marys, Pennsylvania 15857-3397 USA
Tel +1 (814) 834 9140
Fax +1 (814) 781 7969

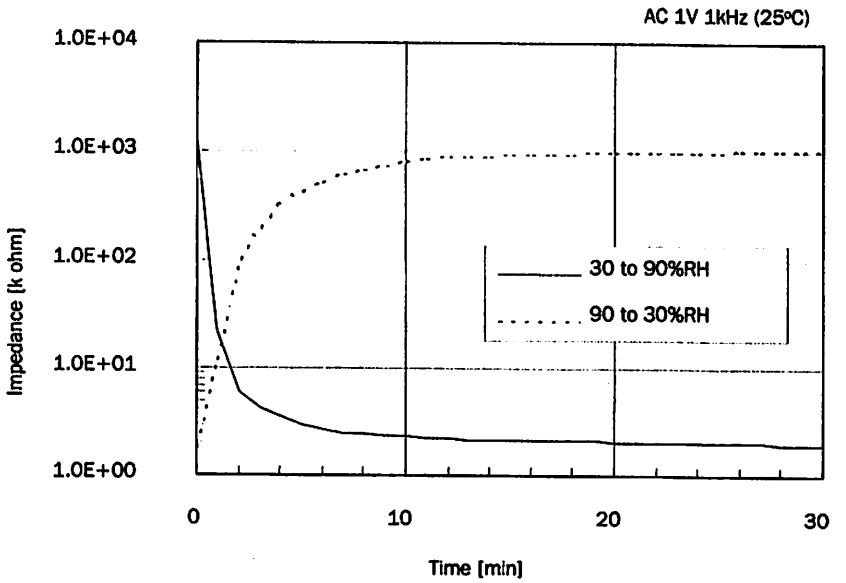
HUMIDITY SENSORS: TYPE HS12P, HS15P

RELATIVE HUMIDITY SENSOR

TYPICAL HUMIDITY CURVE:



TYPICAL RESPONSE CURVE:



rown Industrial Estate, Priorswood Rd
aunton, Somerset TA2 8QY UK
el +44 (0)1823 335200
ax +44 (0)1823 332637

808 US Highway 1
Edison, New Jersey 08817-4695 USA
Tel +1 (732) 287 2870
Fax +1 (732) 287 8847

967 Windfall Road
St Marys, Pennsylvania 15857-3397 USA
Tel +1 (814) 834 9140
Fax +1 (814) 781 7969

National Semiconductor

November 2000

LM35
Precision Centigrade Temperature Sensors

General Description

The LM35 series are precision integrated-circuit temperature sensors, whose output voltage is linearly proportional to the Celsius (Centigrade) temperature. The LM35 thus has an advantage over linear temperature sensors calibrated in ° Kelvin, as the user is not required to subtract a large constant voltage from its output to obtain convenient Centigrade scaling. The LM35 does not require any external calibration or trimming to provide typical accuracies of ±¼°C at room temperature and ±¾°C over a full −55 to +150°C temperature range. Low cost is assured by trimming and calibration at the wafer level. The LM35's low output impedance, linear output, and precise inherent calibration make interfacing to readout or control circuitry especially easy. It can be used with single power supplies, or with plus and minus supplies. As it draws only 60 µA from its supply, it has very low self-heating, less than 0.1°C in still air. The LM35 is rated to operate over a −55° to +150°C temperature range, while the LM35C is rated for a −40° to +110°C range (−10° with improved accuracy). The LM35 series is available packaged in hermetic TO-46 transistor packages, while the LM35C, LM35CA, and LM35D are also available in the plastic TO-92 transistor package. The LM35D is also available in an 8-lead surface mount small outline package and a plastic TO-220 package.

Features

- Calibrated directly in ° Celsius (Centigrade)
- Linear + 10.0 mV/°C scale factor
- 0.5°C accuracy guaranteeable (at +25°C)
- Rated for full −55° to +150°C range
- Suitable for remote applications
- Low cost due to wafer-level trimming
- Operates from 4 to 30 volts
- Less than 60 µA current drain
- Low self-heating, 0.08°C in still air
- Nonlinearity only ±¼°C typical
- Low impedance output, 0.1 Ω for 1 mA load

Typical Applications

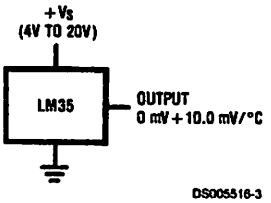
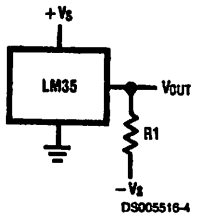


FIGURE 1. Basic Centigrade Temperature Sensor (+2°C to +150°C)

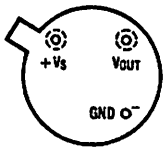


Choose $R_1 = -V_S/50\ \mu A$
V_{OUT}=+1,500 mV at +150°C
= +250 mV at +25°C
= −550 mV at −55°C

FIGURE 2. Full-Range Centigrade Temperature Sensor

Connection Diagrams

TO-46
Metal Can Package*

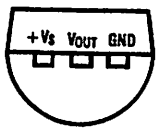


BOTTOM VIEW
DS005516-1

*Case is connected to negative pin (GND)

Order Number LM35H, LM35AH, LM35CH, LM35CAH or LM35DH
See NS Package Number H03H

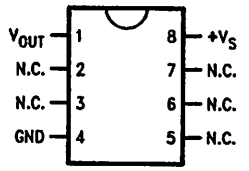
TO-92
Plastic Package



BOTTOM VIEW
DS005516-2

Order Number LM35CZ, LM35CAZ or LM35DZ
See NS Package Number Z03A

SO-8
Small Outline Molded Package

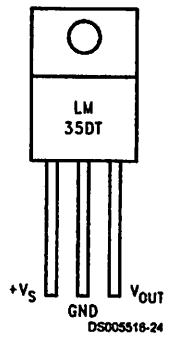


DS005516-21

N.C. = No Connection

Top View
Order Number LM35DM
See NS Package Number M08A

TO-220
Plastic Package*



DS005516-24

*Tab is connected to the negative pin (GND).
Note: The LM35DT pinout is different than the discontinued LM35DP.

Order Number LM35DT
See NS Package Number TA03F

Absolute Maximum Ratings (Note 10)

If Military/Aerospace specified devices are required, please contact the National Semiconductor Sales Office/ Distributors for availability and specifications.

Supply Voltage	+35V to −0.2V
Output Voltage	+6V to −1.0V
Output Current	10 mA
Storage Temp.;	
TO-46 Package,	−60°C to +180°C
TO-92 Package,	−60°C to +150°C
SO-8 Package,	−65°C to +150°C
TO-220 Package,	−65°C to +150°C
Lead Temp.:	
TO-46 Package, (Soldering, 10 seconds)	300°C
TO-92 and TO-220 Package, (Soldering, 10 seconds)	260°C
SO Package (Note 12)	
Vapor Phase (60 seconds)	215°C
Infrared (15 seconds)	220°C
ESD Susceptibility (Note 11)	2500V
Specified Operating Temperature Range: T_{MIN} to T_{MAX} (Note 2)	
LM35, LM35A	−55°C to +150°C
LM35C, LM35CA	−40°C to +110°C
LM35D	0°C to +100°C

Electrical Characteristics

(Notes 1, 6)

Parameter	Conditions	LM35A			LM35CA			Units (Max.)
		Typical	Tested Limit (Note 4)	Design Limit (Note 5)	Typical	Tested Limit (Note 4)	Design Limit (Note 5)	
Accuracy (Note 7)	T_A=+25°C	±0.2	±0.5		±0.2	±0.5		°C
	T_A=−10°C	±0.3			±0.3		±1.0	°C
	T_A=T_{MAX}	±0.4	±1.0		±0.4	±1.0		°C
	T_A=T_{MIN}	±0.4	±1.0		±0.4		±1.5	°C
Nonlinearity (Note 8)	$T_{MIN} \le T_A \le T_{MAX}$	**±0.18**		**±0.35**	**±0.15**		**±0.3**	°C
Sensor Gain (Average Slope)	$T_{MIN} \le T_A \le T_{MAX}$	**+10.0**	**+9.9, +10.1**		**+10.0**		**+9.9, +10.1**	mV/°C
Load Regulation (Note 3) $0 \le I_L \le 1$ mA	T_A=+25°C	±0.4	±1.0		±0.4	±1.0		mV/mA
	$T_{MIN} \le T_A \le T_{MAX}$	**±0.5**		**±3.0**	**±0.5**		**±3.0**	mV/mA
Line Regulation (Note 3)	T_A=+25°C	±0.01	±0.05		±0.01	±0.05		mV/V
	4V≤V_S≤30V	**±0.02**		**±0.1**	**±0.02**		**±0.1**	mV/V
Quiescent Current (Note 9)	V_S=+5V, +25°C	56	67		56	67		µA
	V_S=+5V	**105**		**131**	**91**		**114**	µA
	V_S=+30V, +25°C	56.2	68		56.2	68		µA
	V_S=+30V	**105.5**		**133**	**91.5**		**116**	µA
Change of Quiescent Current (Note 3)	4V≤V_S≤30V, +25°C	0.2	1.0		0.2	1.0		µA
	4V≤V_S≤30V	**0.5**		**2.0**	**0.5**		**2.0**	µA
Temperature Coefficient of Quiescent Current		**+0.39**		**+0.5**	**+0.39**		**+0.5**	µA/°C
Minimum Temperature For Rated Accuracy	In circuit of *Figure 1*, I_L=0	+1.5		+2.0	+1.5		+2.0	°C
Long Term Stability	T_J=T_{MAX}, for 1000 hours	±0.08			±0.08			°C

Electrical Characteristics

(Notes 1, 6)

Parameter	Conditions	LM35			LM35C, LM35D			Units (Max.)
		Typical	Tested Limit (Note 4)	Design Limit (Note 5)	Typical	Tested Limit (Note 4)	Design Limit (Note 5)	
Accuracy, LM35, LM35C (Note 7)	T_A=+25°C	±0.4	±1.0		±0.4	±1.0		°C
	T_A=−10°C	±0.5			±0.5		±1.5	°C
	T_A=T_{MAX}	±0.8	±1.5		±0.8		±1.5	°C
	T_A=T_{MIN}	±0.8		±1.5	±0.8		±2.0	°C
Accuracy, LM35D (Note 7)	T_A=+25°C				±0.6	±1.5		°C
	T_A=T_{MAX}				±0.9		±2.0	°C
	T_A=T_{MIN}				±0.9		±2.0	°C
Nonlinearity (Note 8)	$T_{MIN} \le T_A \le T_{MAX}$	**±0.3**		**±0.5**	**±0.2**		**±0.5**	°C
Sensor Gain (Average Slope)	$T_{MIN} \le T_A \le T_{MAX}$	**+10.0**	**+9.8, +10.2**		**+10.0**		**+9.8, +10.2**	mV/°C
Load Regulation (Note 3) $0 \le I_L \le 1$ mA	T_A=+25°C	±0.4	±2.0		±0.4	±2.0		mV/mA
	$T_{MIN} \le T_A \le T_{MAX}$	**±0.5**		**±5.0**	**±0.5**		**±5.0**	mV/mA
Line Regulation (Note 3)	T_A=+25°C	±0.01	±0.1		±0.01	±0.1		mV/V
	$4V \le V_S \le 30V$	**±0.02**		**±0.2**	**±0.02**		**±0.2**	mV/V
Quiescent Current (Note 9)	V_S=+5V, +25°C	56	80		56	80		µA
	V_S=+5V	**105**		**158**	**91**		**138**	µA
	V_S=+30V, +25°C	56.2	82		56.2	82		µA
	V_S=+30V	**105.5**		**161**	**91.5**		**141**	µA
Change of Quiescent Current (Note 3)	$4V \le V_S \le 30V$, +25°C	0.2	2.0		0.2	2.0		µA
	$4V \le V_S \le 30V$	**0.5**		**3.0**	**0.5**		**3.0**	µA
Temperature Coefficient of Quiescent Current		**+0.39**		**+0.7**	**+0.39**		**+0.7**	µA/°C
Minimum Temperature for Rated Accuracy	In circuit of *Figure 1*, I_L=0	+1.5		+2.0	+1.5		+2.0	°C
Long Term Stability	T_J=T_{MAX}, for 1000 hours	±0.08			±0.08			°C

Note 1: Unless otherwise noted, these specifications apply: $-55°C \le T_J \le +150°C$ for the LM35 and LM35A; $-40°C \le T_J \le +110°C$ for the LM35C and LM35CA; and $0°C \le T_J \le +100°C$ for the LM35D. V_S=+5Vdc and I_{LOAD}=50 µA, in the circuit of *Figure 2*. These specifications also apply from +2°C to T_{MAX} in the circuit of *Figure 1*. Specifications in **boldface** apply over the full rated temperature range.

Note 2: Thermal resistance of the TO-46 package is 400°C/W, junction to ambient, and 24°C/W junction to case. Thermal resistance of the TO-92 package is 180°C/W junction to ambient. Thermal resistance of the small outline molded package is 220°C/W junction to ambient. Thermal resistance of the TO-220 package is 90°C/W junction to ambient. For additional thermal resistance information see table in the Applications section.

Note 3: Regulation is measured at constant junction temperature, using pulse testing with a low duty cycle. Changes in output due to heating effects can be computed by multiplying the internal dissipation by the thermal resistance.

Note 4: Tested Limits are guaranteed and 100% tested in production.

Note 5: Design Limits are guaranteed (but not 100% production tested) over the indicated temperature and supply voltage ranges. These limits are not used to calculate outgoing quality levels.

Note 6: Specifications in **boldface** apply over the full rated temperature range.

Note 7: Accuracy is defined as the error between the output voltage and 10mv/°C times the device's case temperature, at specified conditions of voltage, current, and temperature (expressed in °C).

Note 8: Nonlinearity is defined as the deviation of the output-voltage-versus-temperature curve from the best-fit straight line, over the device's rated temperature range.

Note 9: Quiescent current is defined in the circuit of *Figure 1*.

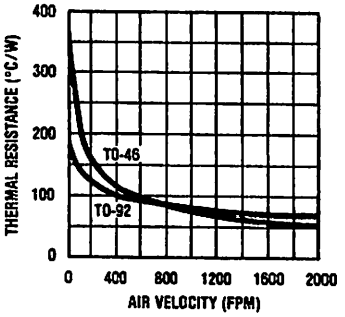
Note 10: Absolute Maximum Ratings indicate limits beyond which damage to the device may occur. DC and AC electrical specifications do not apply when operating the device beyond its rated operating conditions. See Note 1.

Note 11: Human body model, 100 pF discharged through a 1.5 kΩ resistor.

Note 12: See AN-450 "Surface Mounting Methods and Their Effect on Product Reliability" or the section titled "Surface Mount" found in a current National Semiconductor Linear Data Book for other methods of soldering surface mount devices.

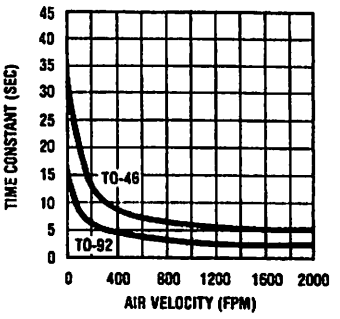
Typical Performance Characteristics

Thermal Resistance Junction to Air



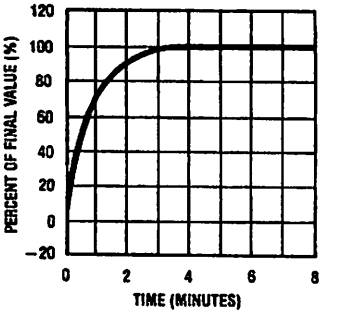
DS005516-25

Thermal Time Constant



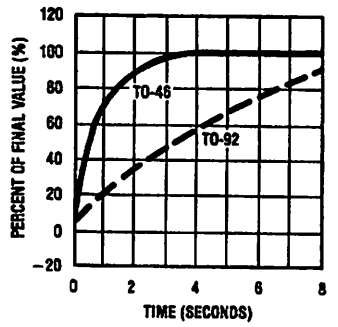
DS005516-26

Thermal Response in Still Air



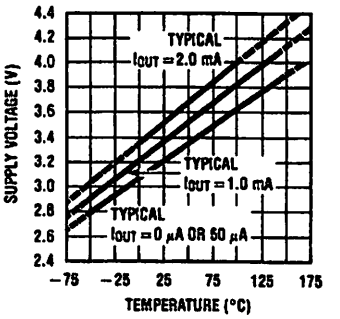
DS005516-27

Thermal Response in Stirred Oil Bath



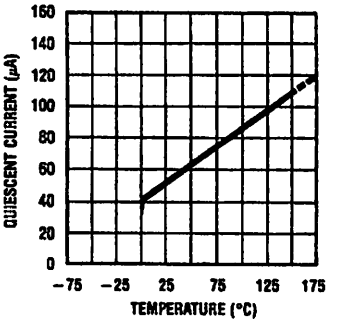
DS005516-28

Minimum Supply Voltage vs. Temperature



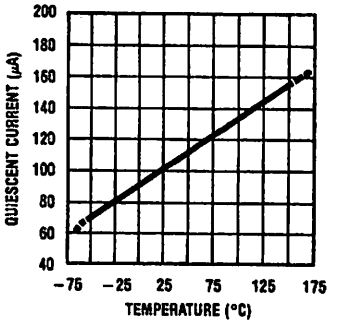
DS005516-29

Quiescent Current vs. Temperature (In Circuit of *Figure 1.*)



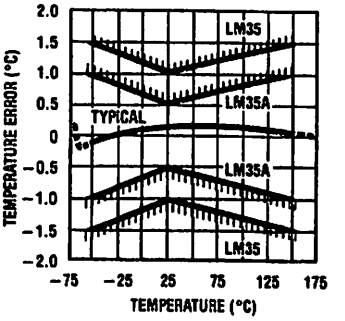
DS005516-30

Quiescent Current vs. Temperature (In Circuit of *Figure 2.*)



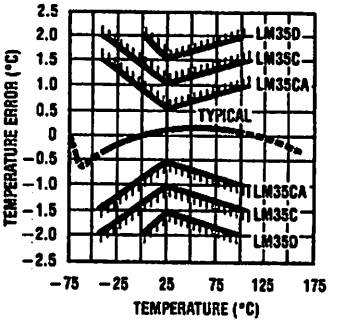
DS005516-31

Accuracy vs. Temperature (Guaranteed)



DS005516-32

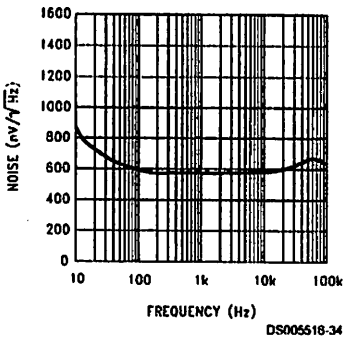
Accuracy vs. Temperature (Guaranteed)



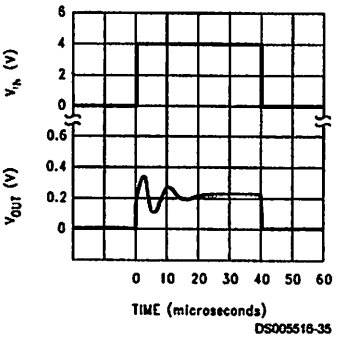
DS005516-33

Typical Performance Characteristics (Continued)

Noise Voltage



Start-Up Response



Applications

The LM35 can be applied easily in the same way as other integrated-circuit temperature sensors. It can be glued or cemented to a surface and its temperature will be within about 0.01°C of the surface temperature.

This presumes that the ambient air temperature is almost the same as the surface temperature; if the air temperature were much higher or lower than the surface temperature, the actual temperature of the LM35 die would be at an intermediate temperature between the surface temperature and the air temperature. This is expecially true for the TO-92 plastic package, where the copper leads are the principal thermal path to carry heat into the device, so its temperature might be closer to the air temperature than to the surface temperature.

To minimize this problem, be sure that the wiring to the LM35, as it leaves the device, is held at the same temperature as the surface of interest. The easiest way to do this is to cover up these wires with a bead of epoxy which will insure that the leads and wires are all at the same temperature as the surface, and that the LM35 die's temperature will not be affected by the air temperature.

The TO-46 metal package can also be soldered to a metal surface or pipe without damage. Of course, in that case the V− terminal of the circuit will be grounded to that metal. Alternatively, the LM35 can be mounted inside a sealed-end metal tube, and can then be dipped into a bath or screwed into a threaded hole in a tank. As with any IC, the LM35 and accompanying wiring and circuits must be kept insulated and dry, to avoid leakage and corrosion. This is especially true if the circuit may operate at cold temperatures where condensation can occur. Printed-circuit coatings and varnishes such as Humiseal and epoxy paints or dips are often used to insure that moisture cannot corrode the LM35 or its connections.

These devices are sometimes soldered to a small light-weight heat fin, to decrease the thermal time constant and speed up the response in slowly-moving air. On the other hand, a small thermal mass may be added to the sensor, to give the steadiest reading despite small deviations in the air temperature.

Temperature Rise of LM35 Due To Self-heating (Thermal Resistance, θ_{JA})

	TO-46, no heat sink	TO-46*, small heat fin	TO-92, no heat sink	TO-92**, small heat fin	SO-8 no heat sink	SO-8** small heat fin	TO-220 no heat sink
Still air	400°C/W	100°C/W	180°C/W	140°C/W	220°C/W	110°C/W	90°C/W
Moving air	100°C/W	40°C/W	90°C/W	70°C/W	105°C/W	90°C/W	26°C/W
Still oil	100°C/W	40°C/W	90°C/W	70°C/W			
Stirred oil	50°C/W	30°C/W	45°C/W	40°C/W			
(Clamped to metal, Infinite heat sink)		(24°C/W)				(55°C/W)	

*Wakefield type 201, or 1" disc of 0.020" sheet brass, soldered to case, or similar.
**TO-92 and SO-8 packages glued and leads soldered to 1" square of 1/16" printed circuit board with 2 oz. foil or similar.

Typical Applications

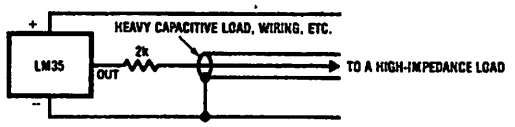


FIGURE 3. LM35 with Decoupling from Capacitive Load

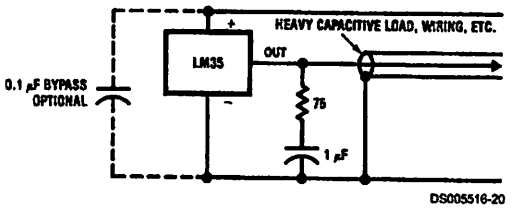


FIGURE 4. LM35 with R-C Damper

CAPACITIVE LOADS

Like most micropower circuits, the LM35 has a limited ability to drive heavy capacitive loads. The LM35 by itself is able to drive 50 pf without special precautions. If heavier loads are anticipated, it is easy to isolate or decouple the load with a resistor; see *Figure 3*. Or you can improve the tolerance of capacitance with a series R-C damper from output to ground; see *Figure 4*.

When the LM35 is applied with a 200Ω load resistor as shown in *Figure 5*, *Figure 6* or *Figure 8* it is relatively immune to wiring capacitance because the capacitance forms a bypass from ground to input, not on the output. However, as with any linear circuit connected to wires in a hostile environment, its performance can be affected adversely by intense electromagnetic sources such as relays, radio transmitters, motors with arcing brushes, SCR transients, etc, as its wiring can act as a receiving antenna and its internal junctions can act as rectifiers. For best results in such cases, a bypass capacitor from V_{IN} to ground and a series R-C damper such as 75Ω in series with 0.2 or 1 µF from output to ground are often useful. These are shown in *Figure 13*, *Figure 14*, and *Figure 16*.

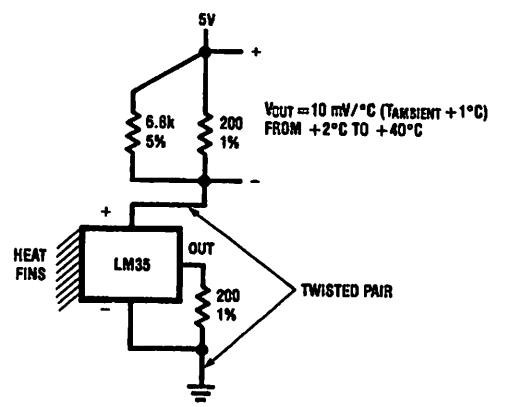


FIGURE 5. Two-Wire Remote Temperature Sensor (Grounded Sensor)

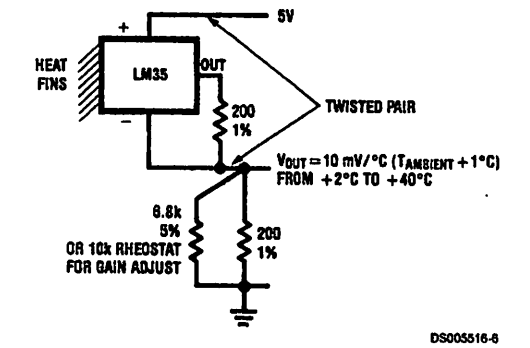


FIGURE 6. Two-Wire Remote Temperature Sensor (Output Referred to Ground)

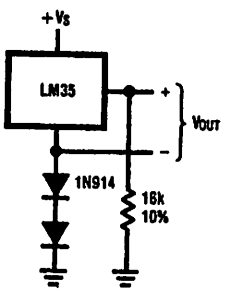


FIGURE 7. Temperature Sensor, Single Supply, −55˚ to +150˚C

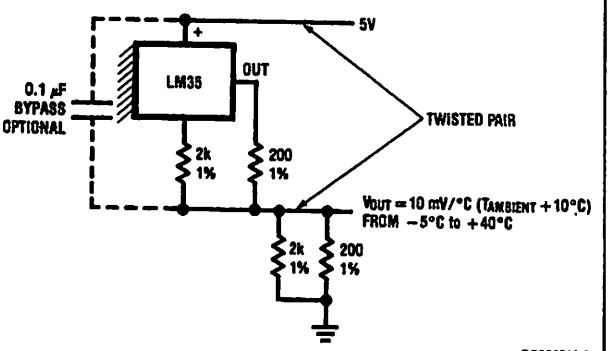


FIGURE 8. Two-Wire Remote Temperature Sensor (Output Referred to Ground)

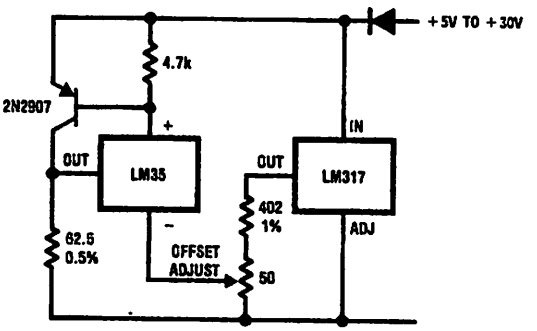


FIGURE 9. 4-To-20 mA Current Source (0˚C to +100˚C)

Typical Applications (Continued)

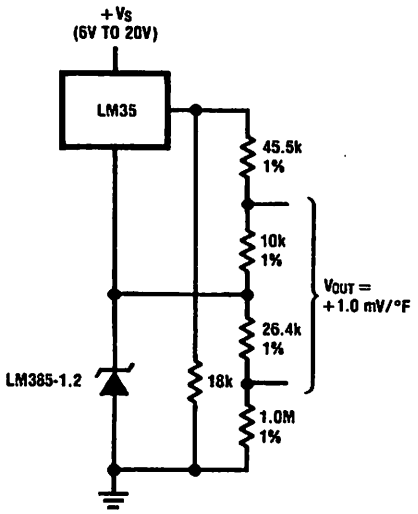


FIGURE 10. Fahrenheit Thermometer

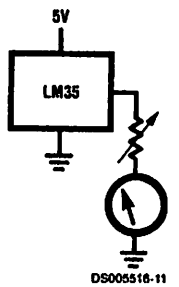


FIGURE 11. Centigrade Thermometer (Analog Meter)

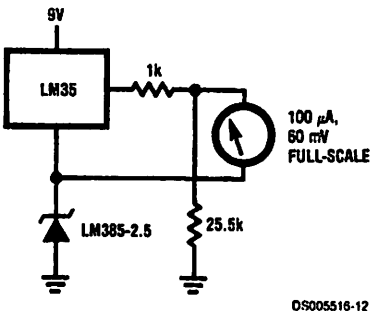


FIGURE 12. Fahrenheit ThermometerExpanded Scale Thermometer
(50˚ to 80˚ Fahrenheit, for Example Shown)

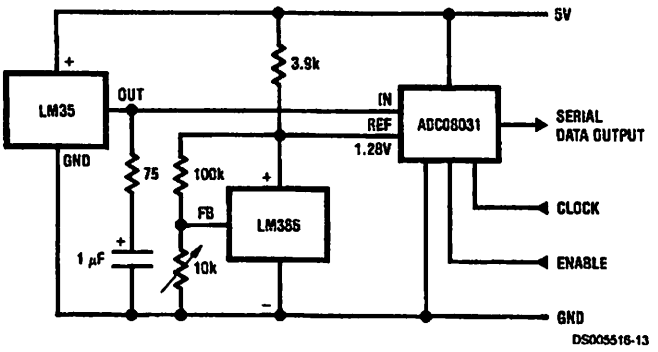


FIGURE 13. Temperature To Digital Converter (Serial Output) (+128˚C Full Scale)

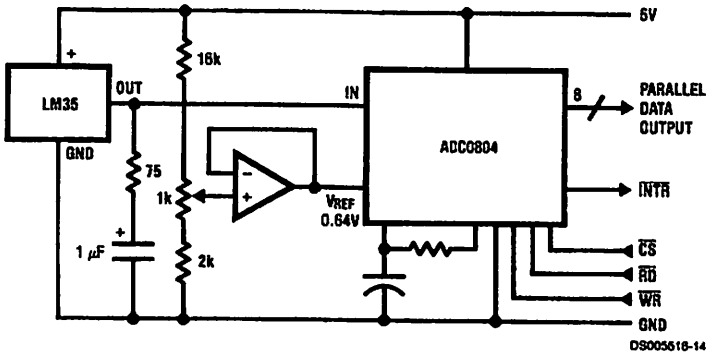
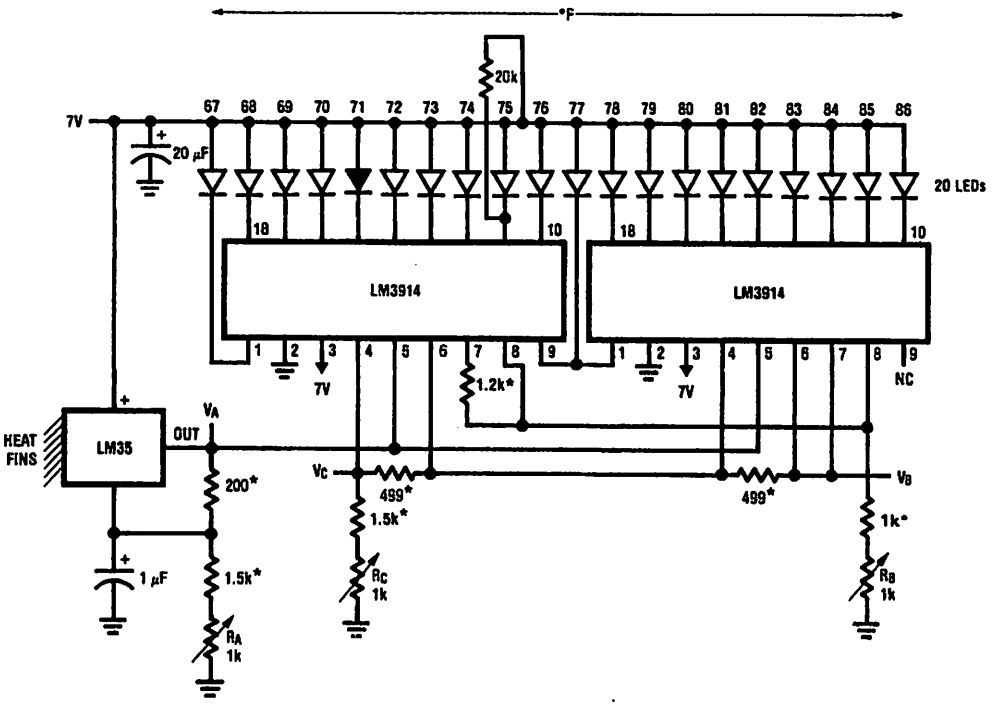


FIGURE 14. Temperature To Digital Converter (Parallel TRI-STATE™ Outputs for Standard Data Bus to µP Interface) (128˚C Full Scale)

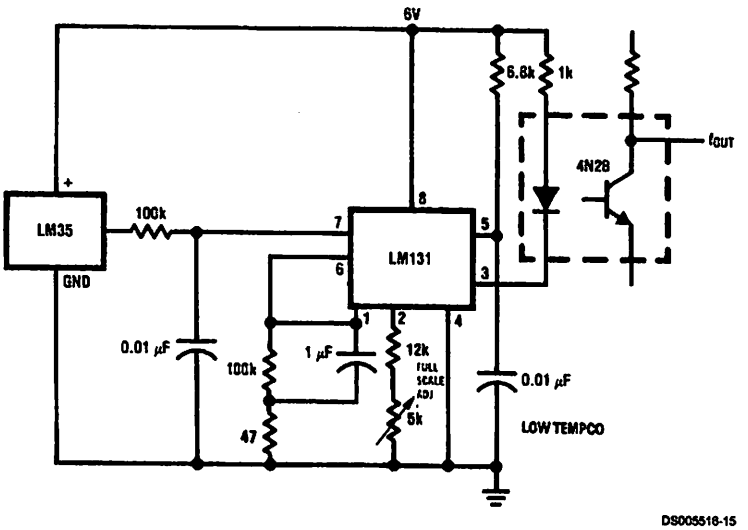
Typical Applications (Continued)



DS005516-16

*=1% or 2% film resistor
Trim R_B for V_B=3.075V
Trim R_C for V_C=1.955V
Trim R_A for V_A=0.075V + 100mV/°C x $T_{ambient}$
Example, V_A=2.275V at 22°C

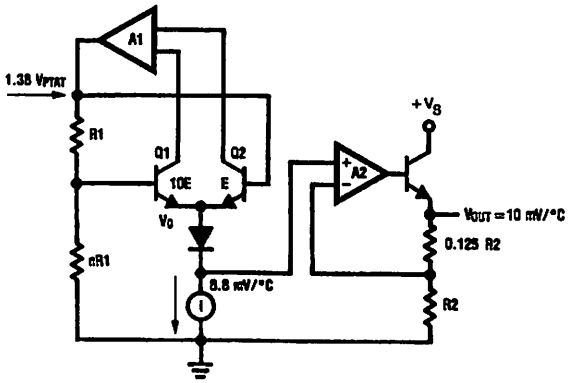
FIGURE 15. Bar-Graph Temperature Display (Dot Mode)



DS005516-15

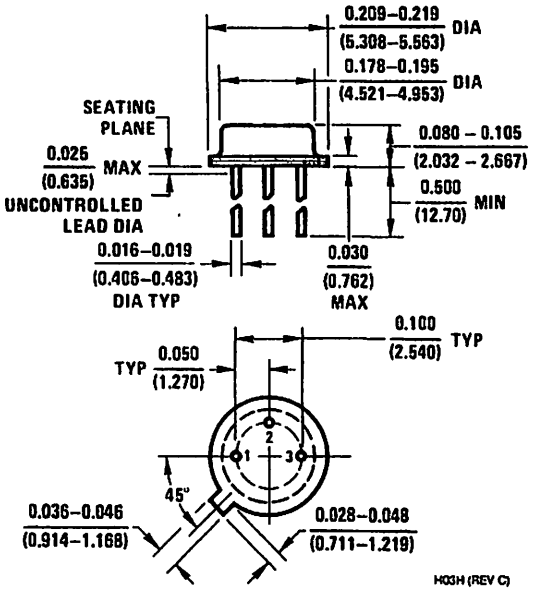
FIGURE 16. LM35 With Voltage-To-Frequency Converter And Isolated Output (2°C to +150°C; 20 Hz to 1500 Hz)

Block Diagram

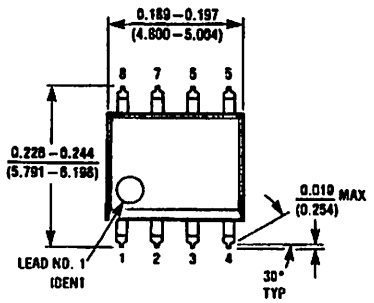


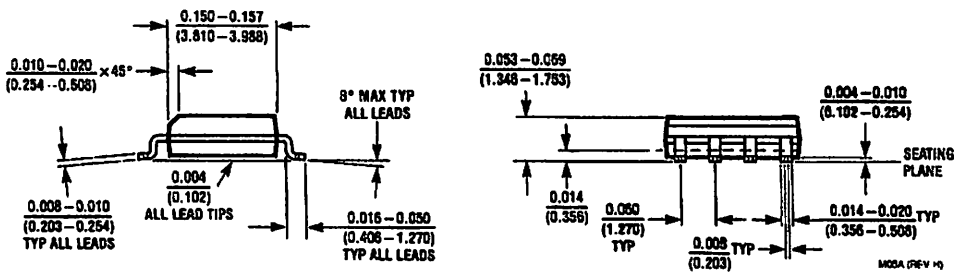
DS005516-23

hysical Dimensions inches (millimeters) unless otherwise noted



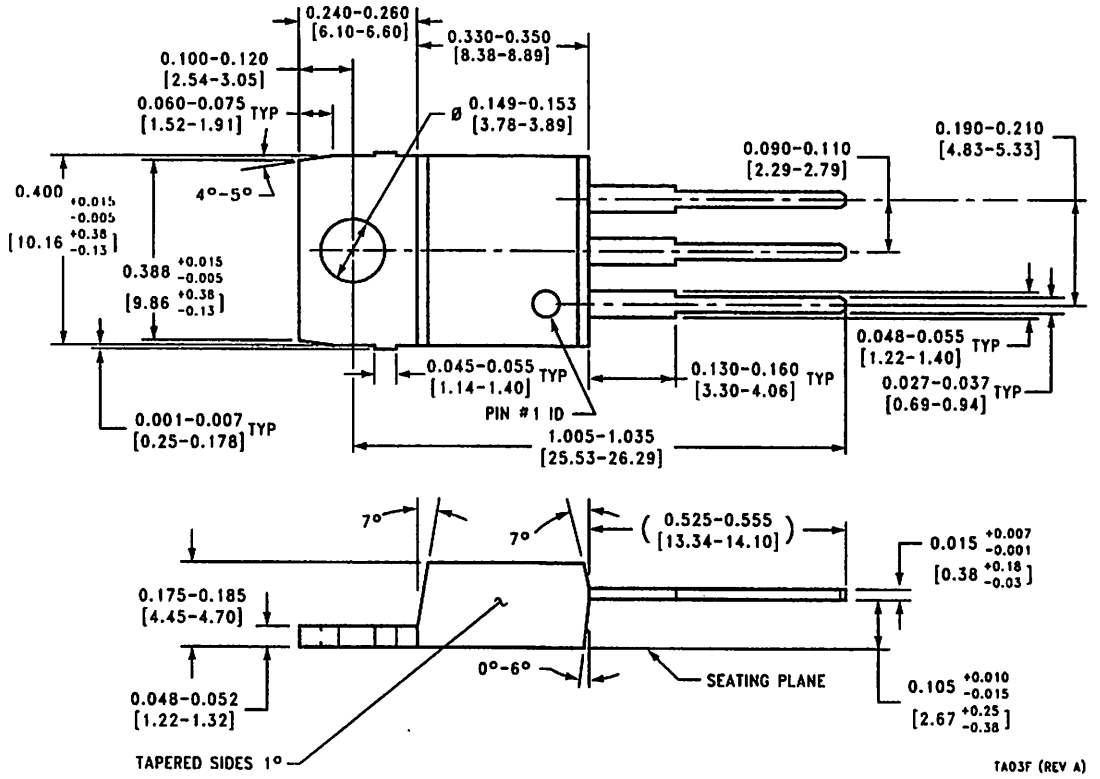
TO-46 Metal Can Package (H)
Order Number LM35H, LM35AH, LM35CH, LM35CAH, or LM35DH
NS Package Number H03H





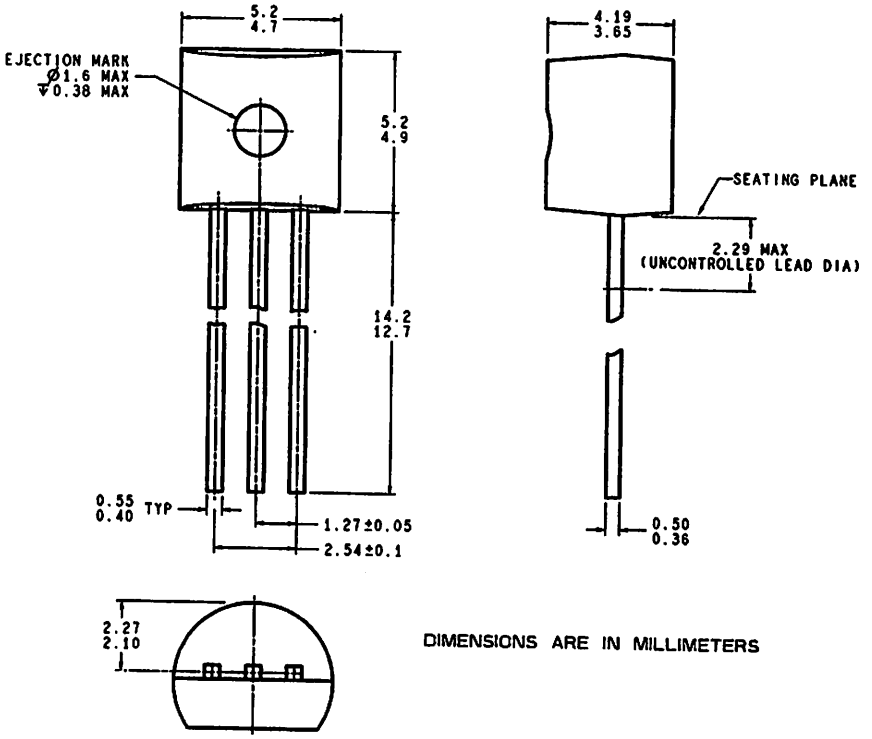
SO-8 Molded Small Outline Package (M)
Order Number LM35DM
NS Package Number M08A

Physical Dimensions inches (millimeters) unless otherwise noted (Continued)



Power Package TO-220 (T)
Order Number LM35DT
NS Package Number TA03F

hysical Dimensions inches (millimeters) unless otherwise noted (Continued)



Z03A (Rev G)

TO-92 Plastic Package (Z)
Order Number LM35CZ, LM35CAZ or LM35DZ
NS Package Number Z03A

: SUPPORT POLICY

IONAL'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT ICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF THE PRESIDENT AND GENERAL INSEL OF NATIONAL SEMICONDUCTOR CORPORATION. As used herein:

ife support devices or systems are devices or ystems which, (a) are intended for surgical implant ito the body, or (b) support or sustain life, and hose failure to perform when properly used in ccordance with instructions for use provided in the beling, can be reasonably expected to result in a gnificant injury to the user.

2. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

National Semiconductor Corporation
Americas
Tel: 1-800-272-9959
Fax: 1-800-737-7018
Email: support@nsc.com
ational.com

National Semiconductor Europe
Fax: +49 (0) 180-530 85 86
Email: europe.support@nsc.com
Deutsch Tel: +49 (0) 69 9508 6208
English Tel: +44 (0) 870 24 0 2171
Français Tel: +33 (0) 1 41 91 8790

National Semiconductor Asia Pacific Customer Response Group
Tel: 65-2544466
Fax: 65-2504466
Email: ap.support@nsc.com

National Semiconductor Japan Ltd.
Tel: 81-3-5639-7560
Fax: 81-3-5639-7507

19-0175; Rev 3; 5/96

MAXIM

±15kV ESD-Protected, +5V RS-232 Transceivers

General Description

The MAX202E–MAX213E, MAX232E/MAX241E line drivers/receivers are designed for RS-232 and V.28 communications in harsh environments. Each transmitter output and receiver input is protected against ±15kV electrostatic discharge (ESD) shocks, without latchup. The various combinations of features are outlined in the *Selection Guide*. The drivers and receivers for all ten devices meet all EIA/TIA-232E and CCITT V.28 specifications at data rates up to 120kbps, when loaded in accordance with the EIA/TIA-232E specification.

The MAX211E/MAX213E/MAX241E are available in 28-pin SO packages, as well as a 28-pin SSOP that uses 60% less board space. The MAX202E/MAX232E come in 16-pin narrow SO, wide SO, and DIP packages. The MAX203E comes in a 20-pin DIP/SO package, and needs no external charge-pump capacitors. The MAX205E comes in a 24-pin wide DIP package, and also eliminates external charge-pump capacitors. The MAX206E/MAX207E/MAX208E come in 24-pin SO, SSOP, and narrow DIP packages. The MAX232E/MAX241E operate with four 1µF capacitors, while the MAX202E/MAX206E/MAX207E/MAX208E/MAX211E/MAX213E operate with four 0.1µF capacitors, further reducing cost and board space.

Applications

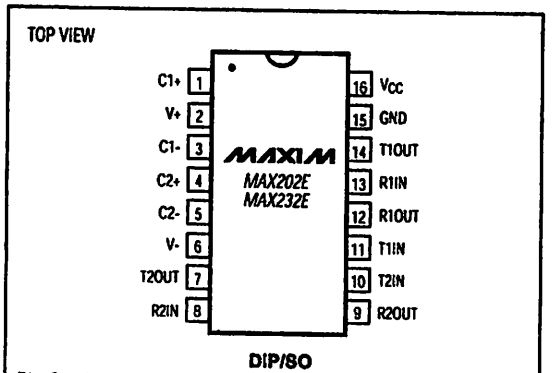
Notebook, Subnotebook, and Palmtop Computers
Battery-Powered Equipment
Hand-Held Equipment

Ordering Information appears at end of data sheet.

Features

- **ESD Protection for RS-232 I/O Pins:**
  ±15kV—Human Body Model
  ±8kV—IEC1000-4-2, Contact Discharge
  ±15kV—IEC1000-4-2, Air-Gap Discharge
- **Latchup Free (unlike bipolar equivalents)**
- **Guaranteed 120kbps Data Rate—LapLink™ Compatible**
- **Guaranteed 3V/µs Min Slew Rate**
- **Operate from a Single +5V Power Supply**

Pin Configurations



DIP/SO

Pin Configurations and Typical Operating Circuits continued at end of data sheet.

Selection Guide

PART	No. of RS-232 DRIVERS	No. of RS-232 RECEIVERS	RECEIVERS ACTIVE IN SHUTDOWN	No. of EXTERNAL CAPACITORS	LOW-POWER SHUTDOWN	TTL THREE-STATE
MAX202E	2	2	0	4 (0.1µF)	No	No
MAX203E	2	2	0	None	No	No
MAX205E	5	5	0	None	Yes	Yes
MAX206E	4	3	0	4 (0.1µF)	Yes	Yes
MAX207E	5	3	0	4 (0.1µF)	No	No
MAX208E	4	4	0	4 (0.1µF)	No	No
MAX211E	4	5	0	4 (0.1µF)	Yes	Yes
MAX213E	4	5	2	4 (0.1µF)	Yes	Yes
MAX232E	2	2	0	4 (1µF)	No	No
MAX241E	4	5	0	4 (1µF)	Yes	Yes

LapLink is a registered trademark of Traveling Software, Inc.

±15kV ESD-Protected, +5V RS-232 Transceivers

ABSOLUTE MAXIMUM RATINGS

V_{CC} ..... -0.3V to +6V
V+ ..... (V_{CC} - 0.3V) to +14V
V- ..... -14V to +0.3V
Input Voltages
T_IN ..... -0.3V to (V+ + 0.3V)
R_IN ..... ±30V
Output Voltages
T_OUT ..... (V- - 0.3V) to (V+ + 0.3V)
R_OUT ..... -0.3V to (V_{CC} + 0.3V)
Short-Circuit Duration, T_OUT ..... Continuous
Continuous Power Dissipation (T_A = +70°C)
16-Pin Plastic DIP (derate 10.53mW/°C above +70°C) 842mW
16-Pin Narrow SO (derate 8.70mW/°C above +70°C) 696mW
16-Pin Wide SO (derate 9.52mW/°C above +70°C) 762mW
20-Pin Plastic DIP (derate 11.11mW/°C above +70°C) 889mW
20-Pin SO (derate 10.00mW/°C above +70°C) 800mW
24-Pin Narrow Plastic DIP
(derate 13.33mW/°C above +70°C) 1.07W
24-Pin Wide Plastic DIP
(derate 14.29mW/°C above +70°C) 1.14W
24-Pin SO (derate 11.76mW/°C above +70°C) 941mW
24-Pin SSOP (derate 8.00mW/°C above +70°C) 640mW
28-Pin SO (derate 12.50mW/°C above +70°C) 1W
28-Pin SSOP (derate 9.52mW/°C above +70°C) 762mW
Operating Temperature Ranges
MAX2_ _EC_ _ ..... 0°C to +70°C
MAX2_ _EE_ _ ..... -40°C to +85°C
Storage Temperature Range ..... -65°C to +165°C
Lead Temperature (soldering, 10sec) ..... +300°C

Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of the specifications is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

ELECTRICAL CHARACTERISTICS

(V_{CC} = +5V ±10% for MAX202E/206E/208E/211E/213E/232E/241E; V_{CC} = +5V ±5% for MAX203E/205E/207E; C1–C4 = 0.1µF for MAX202E/206E/207E/208E/211E/213E; C1–C4 = 1µF for MAX232E/241E; T_A = T_{MIN} to T_{MAX}; unless otherwise noted. Typical values are at T_A = +25°C.)

PARAMETER	SYMBOL	CONDITIONS		MIN	TYP	MAX	UNITS
DC CHARACTERISTICS							
V_{CC} Supply Current	I_{CC}	No load, T_A = +25°C	MAX202E/203E		8	15	mA
			MAX205E–208E		11	20	
			MAX211E/213E		14	20	
			MAX232E		5	10	
			MAX241E		7	15	
Shutdown Supply Current		T_A = +25°C, Figure 1	MAX205E/206E		1	10	µA
			MAX211E/241E		1	10	
			MAX213E		15	50	
LOGIC							
Input Pull-Up Current		T_IN = 0V (MAX205E–208E/211E/213E/241E)			15	200	µA
Input Leakage Current		T_IN = 0V to V_{CC} (MAX202E/203E/232E)				±10	µA
Input Threshold Low	V_{IL}	T_IN; EN, $\overline{SHDN}$ (MAX213E) or $\overline{EN}$, SHDN (MAX205E–208E/211E/241E)				0.8	V
Input Threshold High	V_{IH}	T_IN		2.0			V
		EN, $\overline{SHDN}$ (MAX213E) or $\overline{EN}$, SHDN (MAX205E–208E/211E/241E)		2.4			
Output Voltage Low	V_{OL}	R_OUT; I_{OUT} = 3.2mA (MAX202E/203E/232E) or I_{OUT} = 1.6mA (MAX205E/208E/211E/213E/241E)				0.4	V
Output Voltage High	V_{OH}	R_OUT; I_{OUT} = -1.0mA		3.5	V_{CC} - 0.4		V
Output Leakage Current		$\overline{EN}$ = V_{CC}, EN = 0V, 0V ≤ R_{OUT} ≤ V_{CC}, MAX205E–208E/211E/213E/241E outputs disabled			±0.05	±10	µA

ELECTRICAL CHARACTERISTICS (continued)

(V_{CC} = +5V ±10% for MAX202E/206E/208E/211E/213E/232E/241E; V_{CC} = +5V ±5% for MAX203E/205E/207E; C1–C4 = 0.1µF for MAX202E/206E/207E/208E/211E/213E; C1–C4 = 1µF for MAX232E/241E; T_A = T_{MIN} to T_{MAX}; unless otherwise noted. Typical values are at T_A = +25°C.)

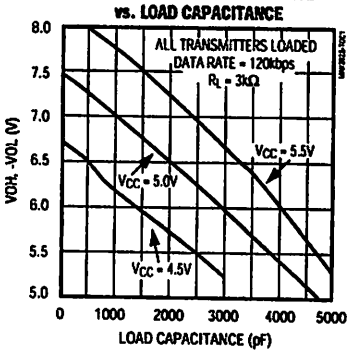
PARAMETER	SYMBOL	CONDITIONS		MIN	TYP	MAX	UNITS
EIA/TIA-232E RECEIVER INPUTS							
Input Voltage Range				-30		30	V
Input Threshold Low		T_A = +25°C, V_{CC} = 5V	All parts, normal operation	0.8	1.2		V
			MAX213E, $\overline{SHDN}$ = 0V, EN = V_{CC}	0.6	1.5		
Input Threshold High		T_A = +25°C, V_{CC} = 5V	All parts, normal operation		1.7	2.4	V
			MAX213E (R4, R5), $\overline{SHDN}$ = 0V, EN = V_{CC}		1.5	2.4	
Input Hysteresis		V_{CC} = 5V, no hysteresis in shutdown		0.2	0.5	1.0	V
Input Resistance		T_A = +25°C, V_{CC} = 5V		3	5	7	kΩ
EIA/TIA-232E TRANSMITTER OUTPUTS							
Output Voltage Swing		All drivers loaded with 3kΩ to ground (Note 1)		±5	±9		V
Output Resistance		V_{CC} = V+ = V- = 0V, V_{OUT} = ±2V		300			Ω
Output Short-Circuit Current					±10	±60	mA
TIMING CHARACTERISTICS							
Maximum Data Rate		R_L = 3kΩ to 7kΩ, C_L = 50pF to 1000pF, one transmitter switching		120			kbps
Receiver Propagation Delay	t_{PLHR}, t_{PHLR}	C_L = 150pF	All parts, normal operation		0.5	10	µs
			MAX213E (R4, R5), $\overline{SHDN}$ = 0V, EN = V_{CC}		4	40	
Receiver Output Enable Time		MAX205E/206E/211E/213E/241E normal operation, Figure 2			600		ns
Receiver Output Disable Time		MAX205E/206E/211E/213E/241E normal operation, Figure 2			200		ns
Transmitter Propagation Delay	t_{PLHT}, t_{PHLT}	R_L = 3kΩ, C_L = 2500pF, all transmitters loaded			2		µs
Transition-Region Slew Rate		T_A = +25°C, V_{CC} = 5V, R_L = 3kΩ to 7kΩ, C_L = 50pF to 1000pF, measured from -3V to +3V or +3V to -3V, Figure 3		3	6	30	V/µs
ESD PERFORMANCE: TRANSMITTER OUTPUTS, RECEIVER INPUTS							
ESD-Protection Voltage		Human Body Model			±15		kV
		IEC1000-4-2, Contact Discharge			±8		
		IEC1000-4-2, Air-Gap Discharge			±15		

Note 1: MAX211EE_ _ tested with V_{CC} = +5V ±5%.

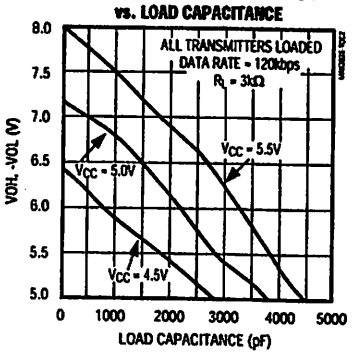
Typical Operating Characteristics

(Typical Operating Circuits, V_{CC} = +5V, T_A = +25°C, unless otherwise noted.)

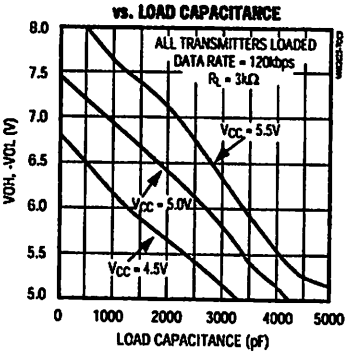
MAX232E
TRANSMITTER OUTPUT VOLTAGE
vs. LOAD CAPACITANCE



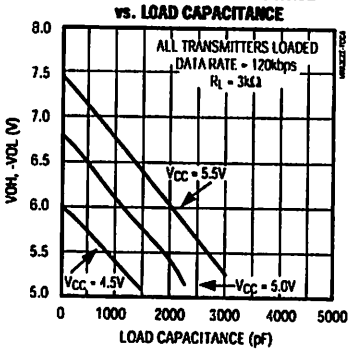
MAX202E/MAX203E
TRANSMITTER OUTPUT VOLTAGE
vs. LOAD CAPACITANCE



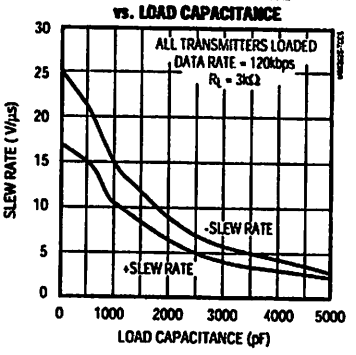
MAX241E
TRANSMITTER OUTPUT VOLTAGE
vs. LOAD CAPACITANCE



MAX211E/MAX213E
TRANSMITTER OUTPUT VOLTAGE
vs. LOAD CAPACITANCE



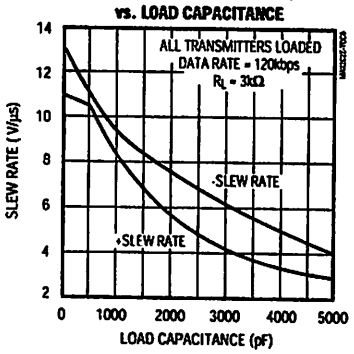
MAX211E/MAX213E/MAX241E
TRANSMITTER SLEW RATE
vs. LOAD CAPACITANCE



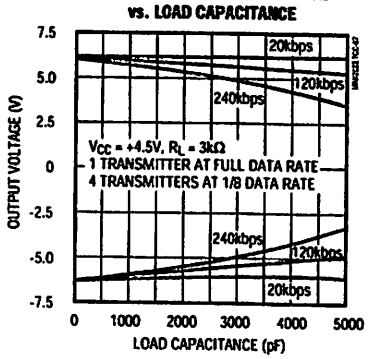
Typical Operating Characteristics (continued)

(Typical Operating Circuits, V_{CC} = +5V, T_A = +25°C, unless otherwise noted.)

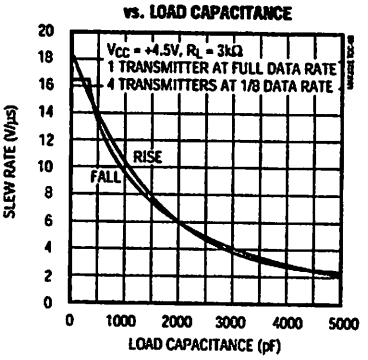
MAX202E/MAX203E/MAX232E TRANSMITTER SLEW RATE vs. LOAD CAPACITANCE



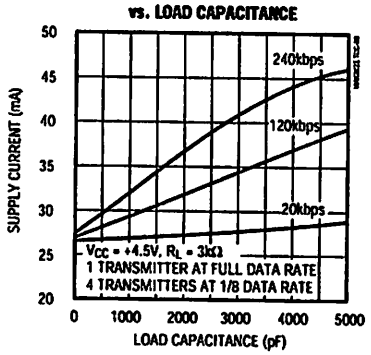
MAX205E–MAX208E TRANSMITTER OUTPUT VOLTAGE vs. LOAD CAPACITANCE



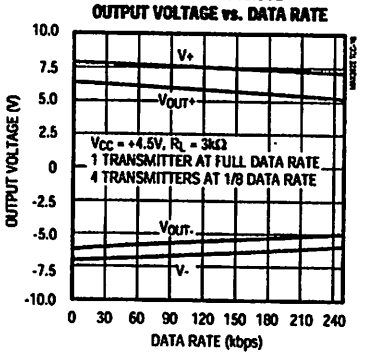
MAX205E–MAX208E TRANSMITTER SLEW RATE vs. LOAD CAPACITANCE



MAX205E–MAX208E SUPPLY CURRENT vs. LOAD CAPACITANCE



MAX205E–MAX208E OUTPUT VOLTAGE vs. DATA RATE



Pin Descriptions

MAX202E/MAX232E

PIN		NAME	FUNCTION
DIP/SO	LCC		
1, 3	2, 4	C1+, C1-	Terminals for positive charge-pump capacitor
2	3	V+	+2VCC voltage generated by the charge pump
4, 5	5, 7	C2+, C2-	Terminals for negative charge-pump capacitor
6	8	V-	-2VCC voltage generated by the charge pump
7, 14	9, 18	T_OUT	RS-232 Driver Outputs
8, 13	10, 17	R_IN	RS-232 Receiver Outputs
9, 12	12, 15	R_OUT	RS-232 Receiver Outputs
10, 11	13, 14	T_IN	RS-232 Driver Inputs
15	19	GND	Ground
16	20	VCC	+4.5V to +5.5V Supply-Voltage Input
—	1, 6, 11, 16	N.C.	No Connect—not internally connected.

MAX203E

PIN		NAME	FUNCTION
DIP	SO		
1, 2	1, 2	T_IN	RS-232 Driver Inputs
3, 20	3, 20	R_OUT	RS-232 Receiver Outputs
4,19	4, 19	R_IN	RS-232 Receiver Inputs
5,18	5, 18	T_OUT	RS-232 Transmitter Outputs
6, 9	6, 9	GND	Ground
7	7	VCC	+4.5V to +5.5V Supply-Voltage Input
8	13	C1+	Make no connection to this pin.
10, 16	11, 16	C2-	Connect pins together.
12, 17	10, 17	V-	-2VCC voltage generated by the charge pump. Connect pins together.
13	14	C1-	Make no connection to this pin.
14	8	V+	+2VCC voltage generated by the charge pump
11, 15	12, 15	C2+	Connect pins together.

MAX205E

PIN	NAME	FUNCTION
1–4, 19	T_OUT	RS-232 Driver Outputs
5, 10, 13, 18, 24	R_IN	RS-232 Receiver Inputs
6, 9, 14, 17, 23	R_OUT	TTL/CMOS Receiver Outputs. All receivers are inactive in shutdown.
7, 8, 15, 16, 22	T_IN	TTL/CMOS Driver Inputs. Internal pull-ups to VCC.
11	GND	Ground
12	VCC	+4.75V to +5.25V Supply Voltage
20	$\overline{EN}$	Receiver Enable—active low
21	SHDN	Shutdown Control—active high

Pin Descriptions (continued)

MAX206E

PIN	NAME	FUNCTION
1, 2, 3, 24	T_OUT	RS-232 Driver Outputs
4, 16, 23	R_IN	RS-232 Receiver Inputs
5, 17, 22	R_OUT	TTL/CMOS Receiver Outputs. All receivers are inactive in shutdown.
6, 7, 18, 19	T_IN	TTL/CMOS Driver Inputs. Internal pull-ups to V_{CC}.
8	GND	Ground
9	V_{CC}	+4.5V to +5.5V Supply Voltage
10, 12	C1+, C1-	Terminals for positive charge-pump capacitor
11	V+	+2V_{CC} generated by the charge pump
13, 14	C2+, C2-	Terminals for negative charge-pump capacitor
15	V-	-2V_{CC} generated by the charge pump
20	$\overline{EN}$	Receiver Enable—active low
21	SHDN	Shutdown Control—active high

MAX207E

PIN	NAME	FUNCTION
1, 2, 3, 20, 24	T_OUT	RS-232 Driver Outputs
4, 16, 23	R_IN	RS-232 Receiver Inputs
5, 17, 22	R_OUT	TTL/CMOS Receiver Outputs. All receivers are inactive in shutdown.
6, 7, 18, 19, 21	T_IN	TTL/CMOS Driver Inputs. Internal pull-ups to V_{CC}.
8	GND	Ground
9	V_{CC}	+4.75V to +5.25V Supply Voltage
10, 12	C1+, C1-	Terminals for positive charge-pump capacitor
11	V+	+2V_{CC} generated by the charge pump
13, 14	C2+, C2-	Terminals for negative charge-pump capacitor
15	V-	-2V_{CC} generated by the charge pump

MAX208E

PIN	NAME	FUNCTION
1, 2, 20, 24	T_OUT	RS-232 Driver Outputs
3, 7, 16, 23	R_IN	RS-232 Receiver Inputs
4, 6, 17, 22	R_OUT	TTL/CMOS Receiver Outputs. All receivers are inactive in shutdown.
5, 18, 19, 21	T_IN	TTL/CMOS Driver Inputs. Internal pull-ups to V_{CC}.
8	GND	Ground
9	V_{CC}	+4.5V to +5.5V Supply Voltage
10, 12	C1+, C1-	Terminals for positive charge-pump capacitor
11	V+	+2V_{CC} generated by the charge pump
13, 14	C2+, C2-	Terminals for negative charge-pump capacitor
15	V-	-2V_{CC} generated by the charge pump

Pin Descriptions (continued)

MAX211E/MAX213E/MAX241E

PIN	NAME	FUNCTION
1, 2, 3, 28	T_OUT	RS-232 Driver Outputs
4, 9, 18, 23, 27	R_IN	RS-232 Receiver Inputs
5, 8, 19, 22, 26	R_OUT	TTL/CMOS Receiver Outputs. For the MAX213E, receivers R4 and R5 are active in shutdown mode when EN = 1. For the MAX211E and MAX241E, all receivers are inactive in shutdown.
6, 7, 20, 21	T_IN	TTL/CMOS Driver Inputs. Only the MAX211E, MAX213E, and MAX241E have internal pull-ups to V_{CC}.
10	GND	Ground
11	V_{CC}	+4.5V to +5.5V Supply Voltage
12, 14	C1+, C1-	Terminals for positive charge-pump capacitor
13	V+	+2V_{CC} voltage generated by the charge pump
15, 16	C2+, C2-	Terminals for negative charge-pump capacitor
17	V-	-2V_{CC} voltage generated by the charge pump
24	$\overline{EN}$	Receiver Enable—active low (MAX211E, MAX241E)
	EN	Receiver Enable—active high (MAX213E)
25	SHDN	Shutdown Control—active high (MAX211E, MAX241E)
	$\overline{SHDN}$	Shutdown Control—active low (MAX213E)

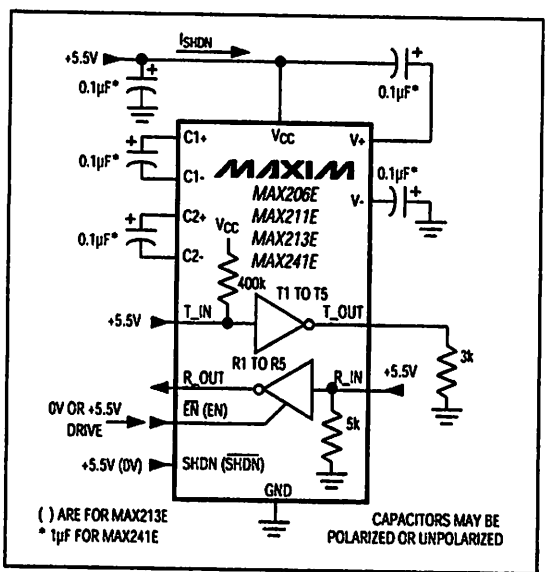


Figure 1. Shutdown-Current Test Circuit (MAX206E, MAX211E/MAX213E/MAX241E)

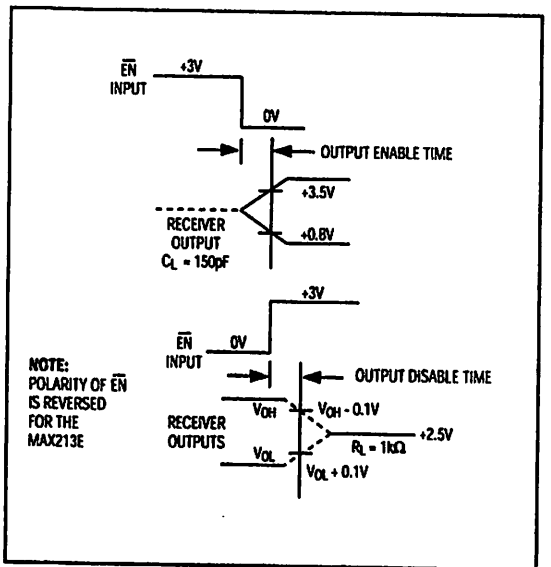


Figure 2. Receiver Output Enable and Disable Timing (MAX205E/MAX206E/MAX211E/MAX213E/MAX241E)

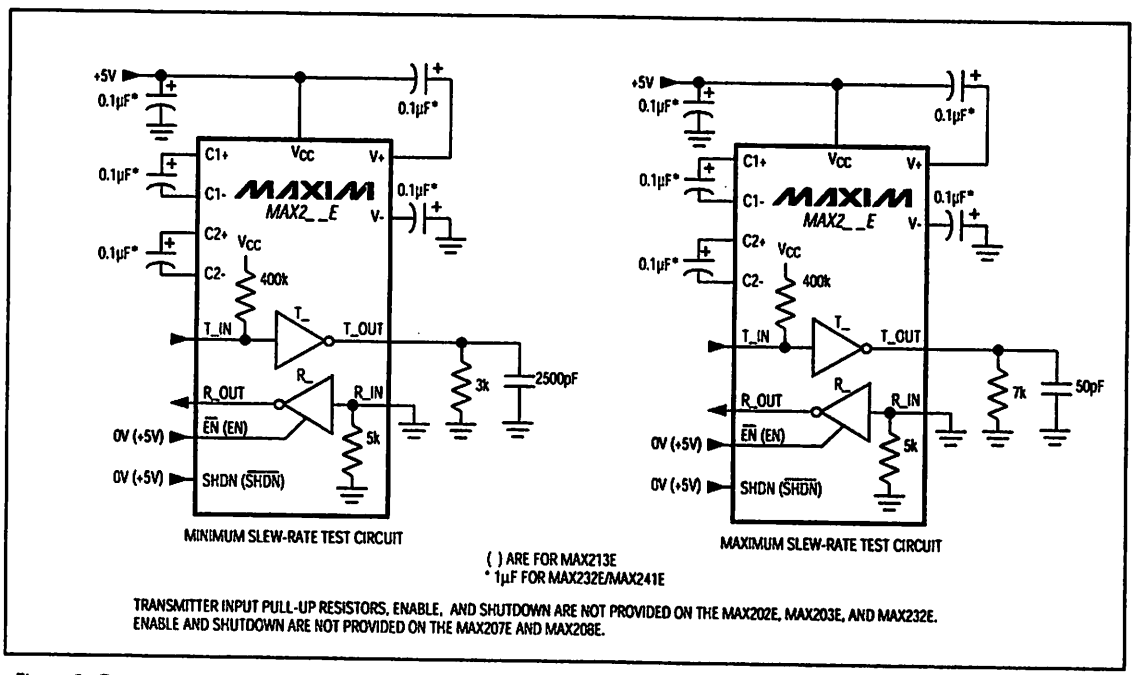


Figure 3. Transition Slew-Rate Circuit

Detailed Description

The MAX202E–MAX213E, MAX232E/MAX241E consist of three sections: charge-pump voltage converters, drivers (transmitters), and receivers. These E versions provide extra protection against ESD. They survive ±15kV discharges to the RS-232 inputs and outputs, tested using the Human Body Model. When tested according to IEC1000-4-2, they survive ±8kV contact-discharges and ±15kV air-gap discharges. The rugged E versions are intended for use in harsh environments or applications where the RS-232 connection is frequently changed (such as notebook computers). The standard (non-"E") MAX202, MAX203, MAX205–MAX208, MAX211, MAX213, MAX232, and MAX241 are recommended for applications where cost is critical.

+5V to ±10V Dual Charge-Pump Voltage Converter

The +5V to ±10V conversion is performed by dual charge-pump voltage converters (Figure 4). The first charge-pump converter uses capacitor C1 to double the +5V into +10V, storing the +10V on the output filter capacitor, C3. The second uses C2 to invert the +10V into -10V, storing the -10V on the V- output filter capacitor, C4.

In shutdown mode, V+ is internally connected to Vcc by a 1kΩ pull-down resistor, and V- is internally connected to ground by a 1kΩ pull-up resistor.

RS-232 Drivers

With Vcc = 5V, the typical driver output voltage swing is ±8V when loaded with a nominal 5kΩ RS-232 receiver. The output swing is guaranteed to meet EIA/TIA-232E and V.28 specifications that call for ±5V minimum output levels under worst-case conditions. These include a 3kΩ load, minimum Vcc, and maximum operating temperature. The open-circuit output voltage swings from (V+ - 0.6V) to V-.

Input thresholds are CMOS/TTL compatible. The unused drivers' inputs on the MAX205E–MAX208E, MAX211E, MAX213E, and MAX241E can be left unconnected because 400kΩ pull-up resistors to Vcc are included on-chip. Since all drivers invert, the pull-up resistors force the unused drivers' outputs low. The MAX202E, MAX203E, and MAX232E do not have pull-up resistors on the transmitter inputs.

When in low-power shutdown mode, the MAX205E/MAX206E/MAX211E/MAX213E/MAX241E driver outputs are turned off and draw only leakage currents—even if they are back-driven with voltages between 0V and 12V. Below -0.5V in shutdown, the transmitter output is diode-clamped to ground with a 1kΩ series impedance.

RS-232 Receivers

The receivers convert the RS-232 signals to CMOS-logic output levels. The guaranteed 0.8V and 2.4V receiver input thresholds are significantly tighter than the ±3V thresholds required by the EIA/TIA-232E specification. This allows the receiver inputs to respond to TTL/CMOS-logic levels, as well as RS-232 levels.

The guaranteed 0.8V input low threshold ensures that receivers shorted to ground have a logic 1 output. The 5kΩ input resistance to ground ensures that a receiver with its input left open will also have a logic 1 output.

Receiver inputs have approximately 0.5V hysteresis. This provides clean output transitions, even with slow rise/fall-time signals with moderate amounts of noise and ringing.

In shutdown, the MAX213E's R4 and R5 receivers have no hysteresis.

Shutdown and Enable Control (MAX205E/MAX206E/MAX211E/MAX213E/MAX241E)

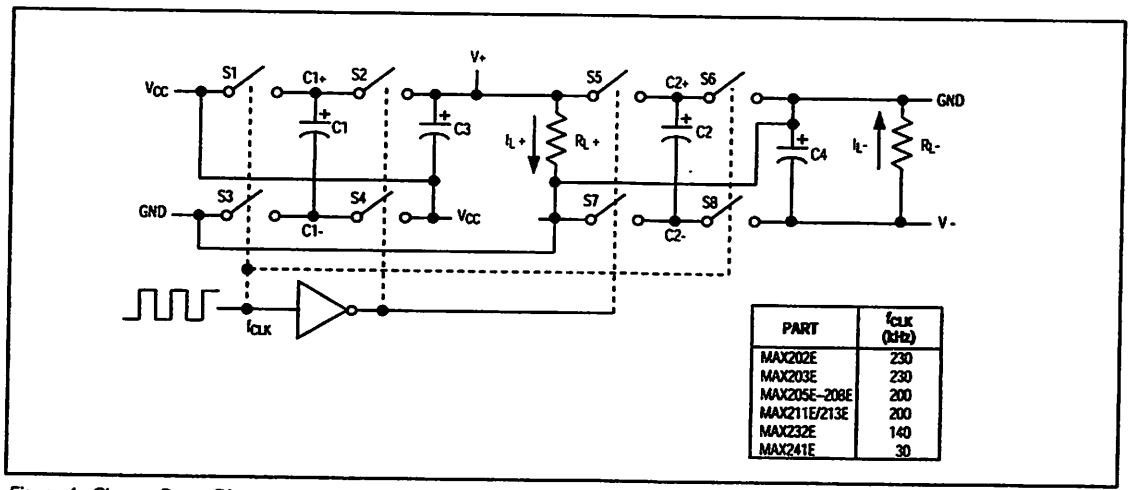
In shutdown mode, the charge pumps are turned off, V+ is pulled down to V_{CC}, V- is pulled to ground, and the transmitter outputs are disabled. This reduces supply current typically to 1µA (15µA for the MAX213E). The time required to exit shutdown is under 1ms, as shown in Figure 5.

Receivers

All MAX213E receivers, except R4 and R5, are put into a high-impedance state in shutdown mode (see Tables 1a and 1b). The MAX213E's R4 and R5 receivers still function in shutdown mode. These two awake-in-shutdown receivers can monitor external activity while maintaining minimal power consumption.

The enable control is used to put the receiver outputs into a high-impedance state, to allow wire-OR connection of two EIA/TIA-232E ports (or ports of different types) at the UART. It has no effect on the RS-232 drivers or the charge pumps.

Note: The enable control pin is active low for the MAX211E/MAX241E ($\overline{EN}$), but is active high for the MAX213E (EN). The shutdown control pin is active high for the MAX205E/MAX206E/MAX211E/MAX241E (SHDN), but is active low for the MAX213E ($\overline{SHDN}$).



PART	f_{CLK} (kHz)
MAX202E	230
MAX203E	230
MAX205E–208E	200
MAX211E/213E	200
MAX232E	140
MAX241E	30

Figure 4. Charge-Pump Diagram

The MAX213E's receiver propagation delay is typically 0.5µs in normal operation. In shutdown mode, propagation delay increases to 4µs for both rising and falling transitions. The MAX213E's receiver inputs have approximately 0.5V hysteresis, except in shutdown, when receivers R4 and R5 have no hysteresis.

When entering shutdown with receivers active, R4 and R5 are not valid until 80µs after $\overline{SHDN}$ is driven low. When coming out of shutdown, all receiver outputs are invalid until the charge pumps reach nominal voltage levels (less than 2ms when using 0.1µF capacitors).

±15kV ESD Protection

As with all Maxim devices, ESD-protection structures are incorporated on all pins to protect against electrostatic discharges encountered during handling and assembly. The driver outputs and receiver inputs have extra protection against static electricity. Maxim's engineers developed state-of-the-art structures to protect these pins against ESD of ±15kV without damage. The ESD structures withstand high ESD in all states: normal operation, shutdown, and powered down. After an ESD event, Maxim's E versions keep working without latchup, whereas competing RS-232 products can latch and must be powered down to remove latchup.

ESD protection can be tested in various ways; the transmitter outputs and receiver inputs of this product family are characterized for protection to the following limits:

1) ±15kV using the Human Body Model
2) ±8kV using the contact-discharge method specified in IEC1000-4-2
3) ±15kV using IEC1000-4-2's air-gap method.

ESD Test Conditions

ESD performance depends on a variety of conditions. Contact Maxim for a reliability report that documents test set-up, test methodology, and test results.

Human Body Model

Figure 6a shows the Human Body Model, and Figure 6b shows the current waveform it generates when discharged into a low impedance. This model consists of a 100pF capacitor charged to the ESD voltage of interest, which is then discharged into the test device through a 1.5kΩ resistor.

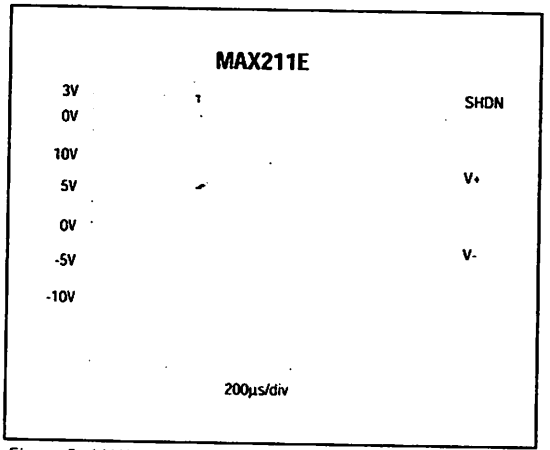


Figure 5. MAX211E V+ and V- when Exiting Shutdown (0.1µF capacitors)

Table 1a. MAX205E/MAX206E/MAX211E/ MAX241E Control Pin Configurations

SHDN	$\overline{EN}$	OPERATION STATUS	Tx	Rx
0	0	Normal Operation	All Active	All Active
0	1	Normal Operation	All Active	All High-Z
1	X	Shutdown	All High-Z	All High Z

X = Don't Care

Table 1b. MAX213E Control Pin Configurations

$\overline{SHDN}$	EN	OPERATION STATUS	Tx 1–4	Rx	
				1–3	4, 5
0	0	Shutdown	All High-Z	High-Z	High-Z
0	1	Shutdown	All High-Z	High-Z	Active*
1	0	Normal Operation	All Active	High-Z	High-Z
1	1	Normal Operation	All Active	Active	Active

**Active = active with reduced performance*

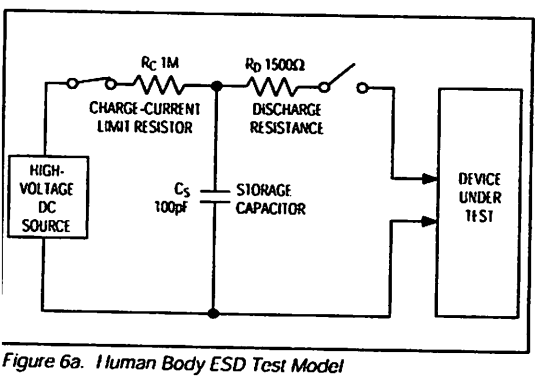


Figure 6a. Human Body ESD Test Model

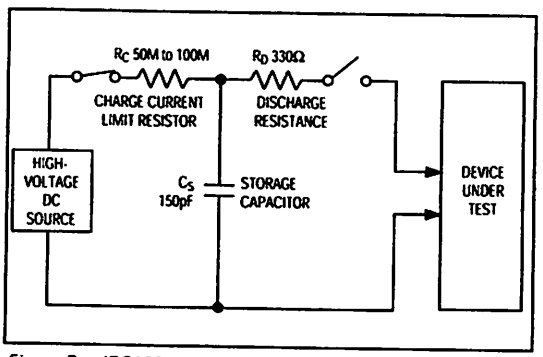


Figure 7a. IEC1000-4-2 ESD Test Model

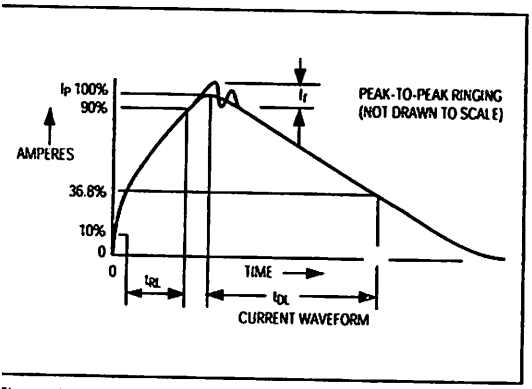


Figure 6b. Human Body Model Current Waveform

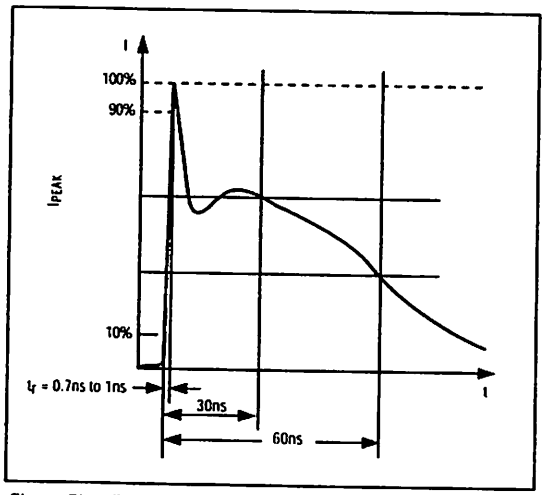


Figure 7b. IEC1000-4-2 ESD Generator Current Waveform

IEC1000-4-2

he IEC1000-4-2 standard covers ESD testing and erformance of finished equipment; it does not pecifically refer to integrated circuits. The IAX202E/MAX203E–MAX213E, MAX232E/MAX241E elp you design equipment that meets level 4 (the ighest level) of IEC1000-4-2, without the need for dditional ESD-protection components.

he major difference between tests done using the uman Body Model and IEC1000-4-2 is higher peak urrent in IEC1000-4-2, because series resistance is wer in the IEC1000-4-2 model. Hence, the ESD ithstand voltage measured to IEC1000-4-2 is enerally lower than that measured using the Human ody Model. Figure 7b shows the current waveform for e 8kV IEC1000-4-2 level-four ESD contact-discharge st.

The air-gap test involves approaching the device with a charged probe. The contact-discharge method connects the probe to the device before the probe is energized.

Machine Model

The Machine Model for ESD tests all pins using a 200pF storage capacitor and zero discharge resistance. Its objective is to emulate the stress caused by contact that occurs with handling and assembly during manufacturing. Of course, all pins require this protection during manufacturing, not just RS-232 inputs and outputs. Therefore, after PC board assembly, the Machine Model is less relevant to I/O ports.

Applications Information

Capacitor Selection

The capacitor type used for C1–C4 is not critical for proper operation. The MAX202E, MAX206–MAX208E, MAX211E, and MAX213E require 0.1µF capacitors, and the MAX232E and MAX241E require 1µF capacitors, although in all cases capacitors up to 10µF can be used without harm. Ceramic, aluminum-electrolytic, or tantalum capacitors are suggested for the 1µF capacitors, and ceramic dielectrics are suggested for the 0.1µF capacitors. When using the minimum recommended capacitor values, make sure the capacitance value does not degrade excessively as the operating temperature varies. If in doubt, use capacitors with a larger (e.g., 2x) nominal value. The capacitors' effective series resistance (ESR), which usually rises at low temperatures, influences the amount of ripple on V+ and V-.

Use larger capacitors (up to 10µF) to reduce the output impedance at V+ and V-. This can be useful when "stealing" power from V+ or from V-. The MAX203E and MAX205E have internal charge-pump capacitors.

Bypass V_{CC} to ground with at least 0.1µF. In applications sensitive to power-supply noise generated by the charge pumps, decouple V_{CC} to ground with a capacitor the same size as (or larger than) the charge-pump capacitors (C1–C4).

V+ and V- as Power Supplies

A small amount of power can be drawn from V+ and V-, although this will reduce both driver output swing and noise margins. Increasing the value of the charge-pump capacitors (up to 10µF) helps maintain performance when power is drawn from V+ or V-.

Driving Multiple Receivers

Each transmitter is designed to drive a single receiver. Transmitters can be paralleled to drive multiple receivers.

Driver Outputs when Exiting Shutdown

The driver outputs display no ringing or undesirable transients as they come out of shutdown.

High Data Rates

These transceivers maintain the RS-232 ±5.0V minimum driver output voltages at data rates of over 120kbps. For data rates above 120kbps, refer to the Transmitter Output Voltage vs. Load Capacitance graphs in the *Typical Operating Characteristics*. Communication at these high rates is easier if the capacitive loads on the transmitters are small; i.e., short cables are best.

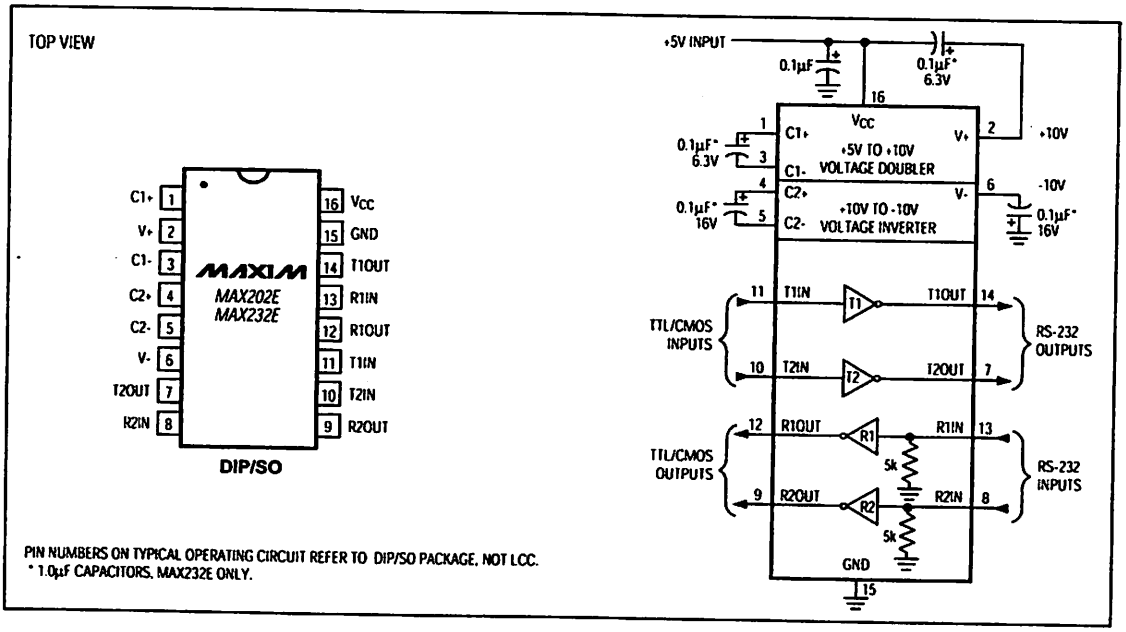
Table 2. Summary of EIA/TIA-232E, V.28 Specifications

PARAMETER		CONDITIONS	EIA/TIA-232E, V.28 SPECIFICATIONS
Driver Output Voltage	0 Level	3kΩ to 7kΩ load	+5V to +15V
	1 Level	3kΩ to 7kΩ load	-5V to -15V
Driver Output Level, Max		No load	±25V
Data Rate		$3k\Omega \le R_L \le 7k\Omega$, $C_L \le 2500pF$	Up to 20kbps
Receiver Input Voltage	0 Level		+3V to +15V
	1 Level		-3V to -15V
Receiver Input Level			±25V
Instantaneous Slew Rate, Max		$3k\Omega \le R_L \le 7k\Omega$, $C_L \le 2500pF$	30V/µs
Driver Output Short-Circuit Current, Max			100mA
Transition Rate on Driver Output		V.28	1ms or 3% of the period
		EIA/TIA-232E	4% of the period
Driver Output Resistance		$-2V < V_{OUT} < +2V$	300Ω

Table 3. DB9 Cable Connections Commonly Used for EIA/TIAE-232E and V.24 Asynchronous Interfaces

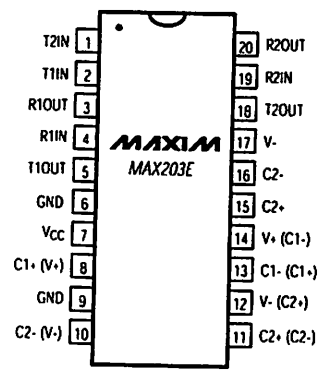
PIN	CONNECTION	
1	Received Line Signal Detector (sometimes called Carrier Detect, DCD)	Handshake from DCE
2	Receive Data (RD)	Data from DCE
3	Transmit Data (TD)	Data from DTE
4	Data Terminal Ready	Handshake from DTE
5	Signal Ground	Reference point for signals
6	Data Set Ready (DSR)	Handshake from DCE
7	Request to Send (RTS)	Handshake from DTE
8	Clear to Send (CTS)	Handshake from DCE
9	Ring Indicator	Handshake from DCE

Pin Configurations and Typical Operating Circuits (continued)



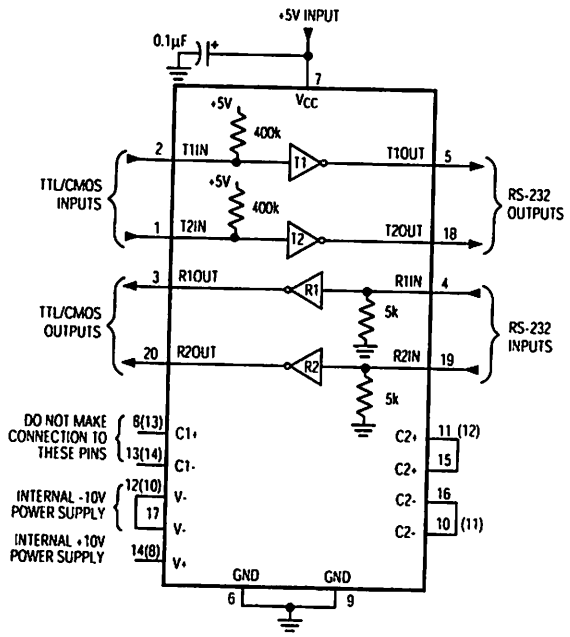
Pin Configurations and Typical Operating Circuits (continued)

TOP VIEW



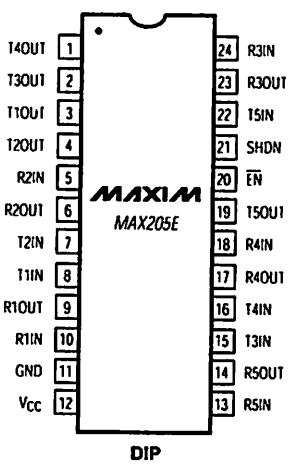
DIP/SO

PIN NUMBERS IN ( ) ARE FOR SO PACKAGE.

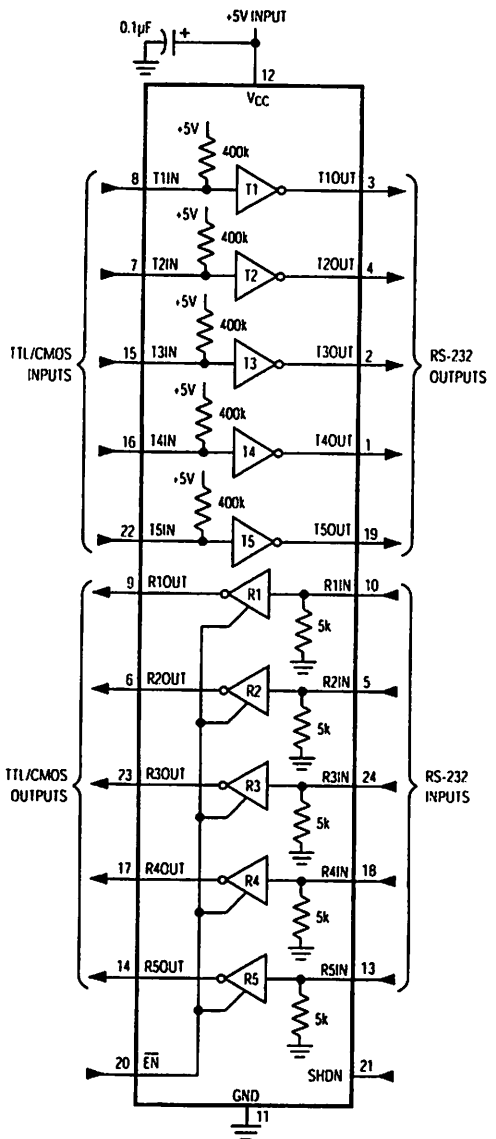


Pin Configurations and Typical Operating Circuits (continued)

TOP VIEW

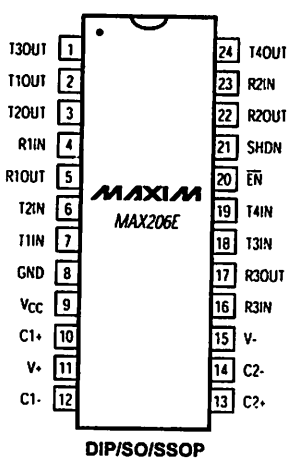


DIP

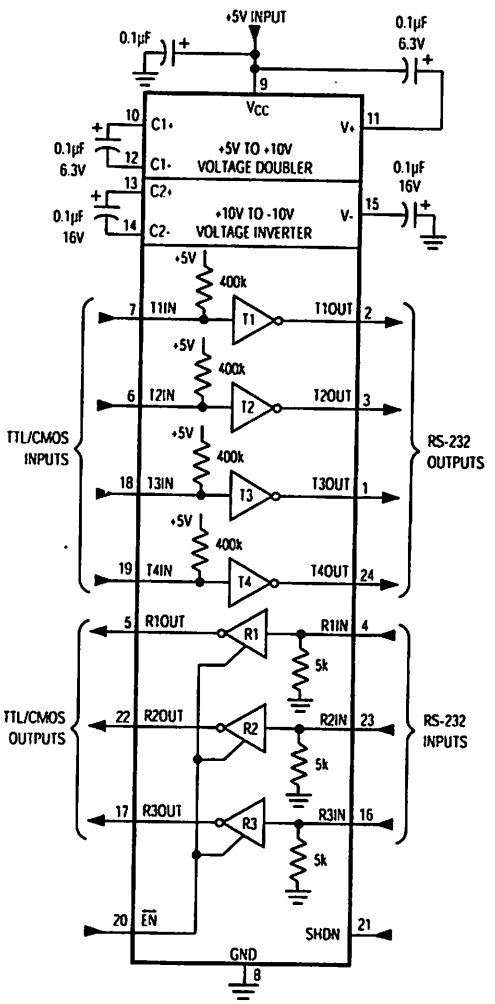


Pin Configurations and Typical Operating Circuits (continued)

TOP VIEW

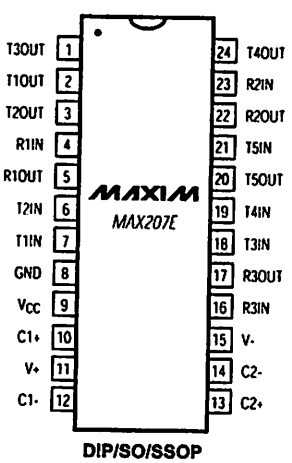


DIP/SO/SSOP

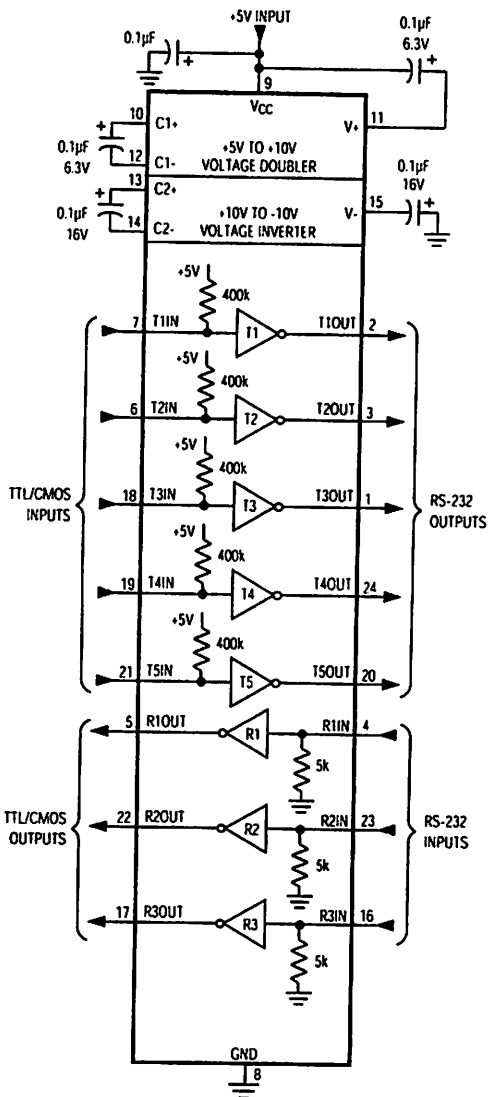


Pin Configurations and Typical Operating Circuits (continued)

TOP VIEW

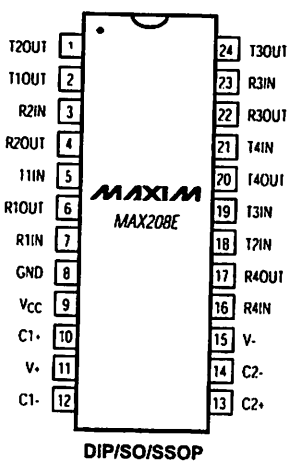


DIP/SO/SSOP

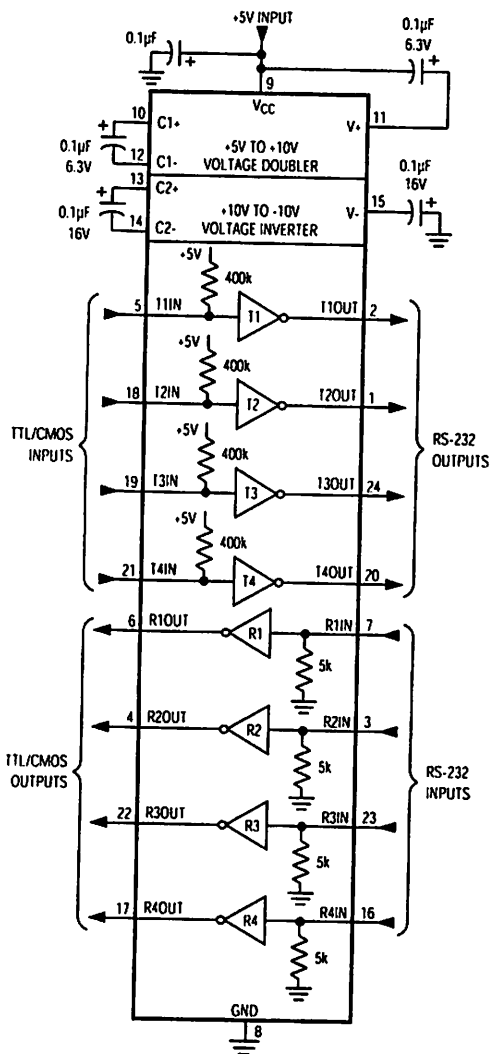


Pin Configurations and Typical Operating Circuits (continued)

TOP VIEW

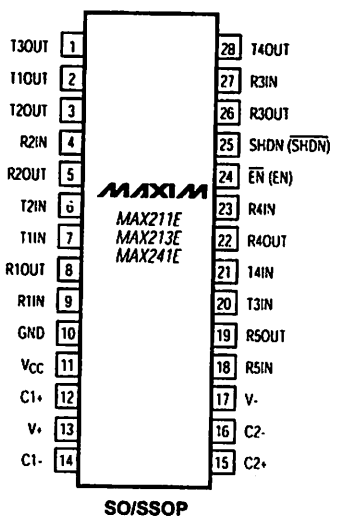


DIP/SO/SSOP

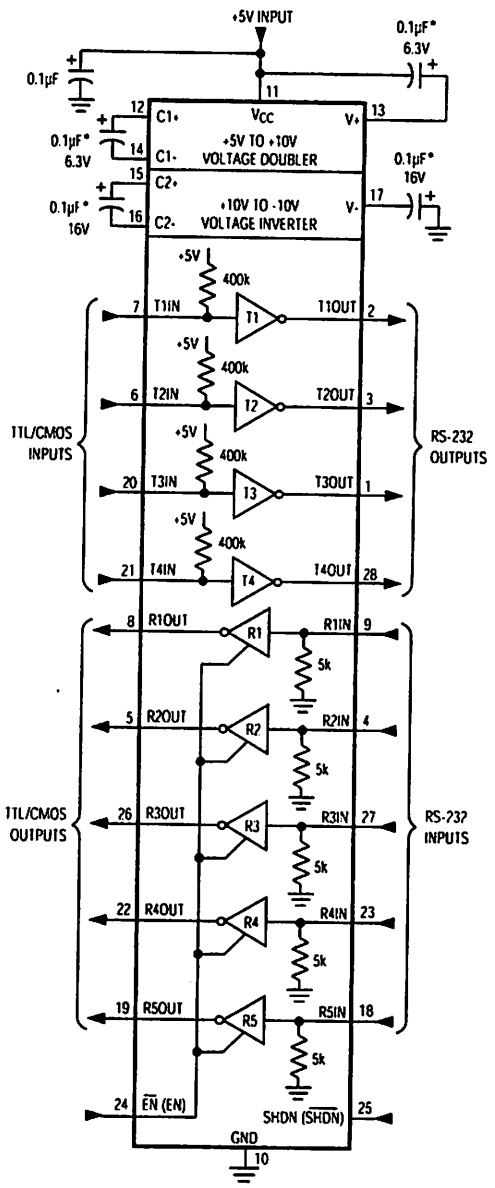


Pin Configurations and Typical Operating Circuits (continued)

TOP VIEW



SO/SSOP



( ) ARE FOR MAX213E ONLY
* 1.0µF CAPACITORS, MAX241E ONLY

Ordering Information

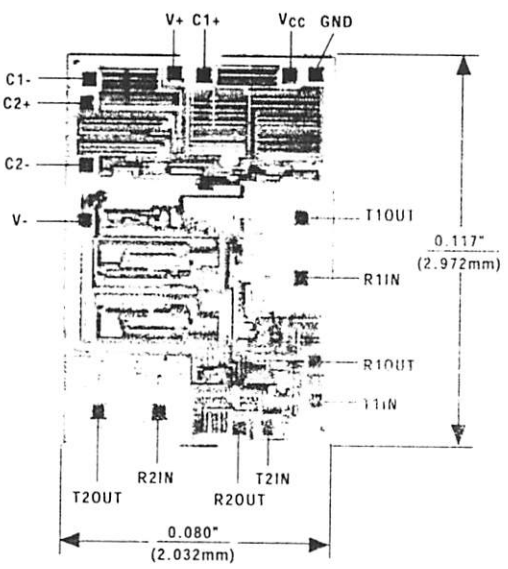
PART	TEMP. RANGE	PIN-PACKAGE
MAX202ECPE	0°C to +70°C	16 Plastic DIP
MAX202ECSE	0°C to +70°C	16 Narrow SO
MAX202ECWE	0°C to +70°C	16 Wide SO
MAX202EC/D	0°C to +70°C	Dice*
MAX202EEPE	-40°C to +85°C	16 Plastic DIP
MAX202EESE	-40°C to +85°C	16 Narrow SO
MAX202EEWE	-40°C to +85°C	16 Wide SO
MAX203ECPP	0°C to +70°C	20 Plastic DIP
MAX203ECWP	0°C to +70°C	20 SO
MAX203EEPP	-40°C to +85°C	20 Plastic DIP
MAX203EEWP	-40°C to +85°C	20 SO
MAX205ECPG	0°C to +70°C	24 Wide Plastic DIP
MAX205EEPG	-40°C to +85°C	24 Wide Plastic DIP
MAX206ECNG	0°C to +70°C	24 Narrow Plastic DIP
MAX206ECWG	0°C to +70°C	24 SO
MAX206ECAG	0°C to +70°C	24 SSOP
MAX206EENG	-40°C to +85°C	24 Narrow Plastic DIP
MAX206EEWG	-40°C to +85°C	24 SO
MAX206EEAG	-40°C to +85°C	24 SSOP
MAX207ECNG	0°C to +70°C	24 Narrow Plastic DIP
MAX207ECWG	0°C to +70°C	24 SO
MAX207ECAG	0°C to +70°C	24 SSOP
MAX207EENG	-40°C to +85°C	24 Narrow Plastic DIP
MAX207EEWG	-40°C to +85°C	24 SO
MAX207EEAG	-40°C to +85°C	24 SSOP

PART	TEMP. RANGE	PIN-PACKAGE
MAX208ECNG	0°C to +70°C	24 Narrow Plastic DIP
MAX208ECWG	0°C to +70°C	24 SO
MAX208ECAG	0°C to +70°C	24 SSOP
MAX208EENG	-40°C to +85°C	24 Narrow Plastic DIP
MAX208EEWG	-40°C to +85°C	24 SO
MAX208EEAG	-40°C to +85°C	24 SSOP
MAX211ECWI	0°C to +70°C	28 SO
MAX211ECAI	0°C to +70°C	28 SSOP
MAX211EEWI	-40°C to +85°C	28 SO
MAX211EEAI	-40°C to +85°C	28 SSOP
MAX213ECWI	0°C to +70°C	28 SO
MAX213ECAI	0°C to +70°C	28 SSOP
MAX213EEWI	-40°C to +85°C	28 SO
MAX213EEAI	-40°C to +85°C	28 SSOP
MAX232ECPE	0°C to +70°C	16 Plastic DIP
MAX232ECSE	0°C to +70°C	16 Narrow SO
MAX232ECWE	0°C to +70°C	16 Wide SO
MAX232EC/D	0°C to +70°C	Dice*
MAX232EEPE	-40°C to +85°C	16 Plastic DIP
MAX232EESE	-40°C to +85°C	16 Narrow SO
MAX232EEWE	-40°C to +85°C	16 Wide SO
MAX241ECWI	0°C to +70°C	28 SO
MAX241ECAI	0°C to +70°C	28 SSOP
MAX241EEWI	-40°C to +85°C	28 SO
MAX241EEAI	-40°C to +85°C	28 SSOP

*Dice are specified at T_A = +25°C.

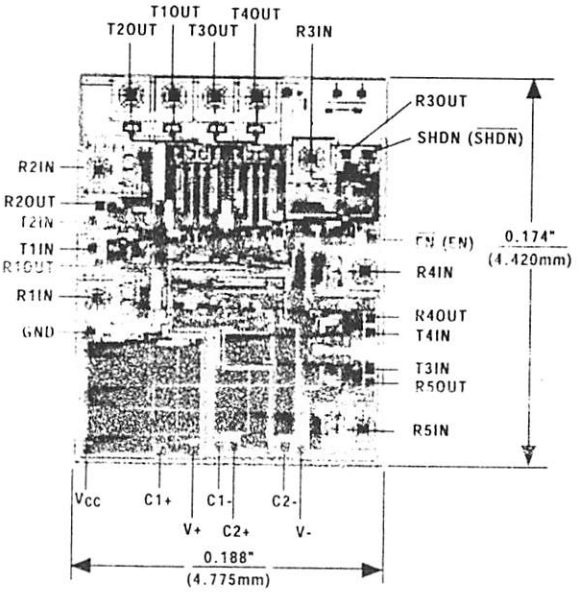
Chip Topographies

MAX202E/MAX232E



TRANSISTOR COUNT: 123
SUBSTRATE CONNECTED TO GND

MAX211E/MAX213E/MAX241E



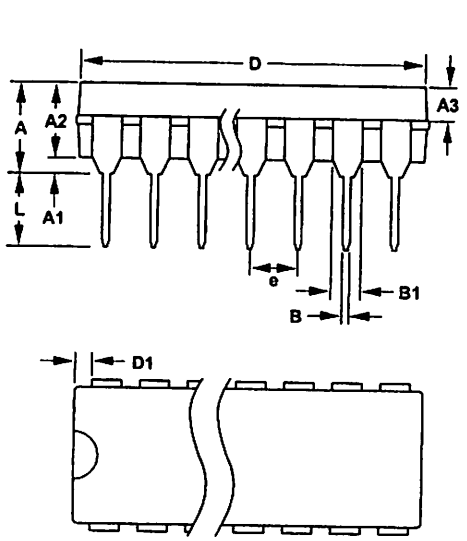
( ) ARE FOR MAX213E ONLY

TRANSISTOR COUNT: 542
SUBSTRATE CONNECTED TO GND

Chip Information

MAX205E/MAX206E/MAX207E/MAX208E
TRANSISTOR COUNT: 328
SUBSTRATE CONNECTED TO GND

Package Information

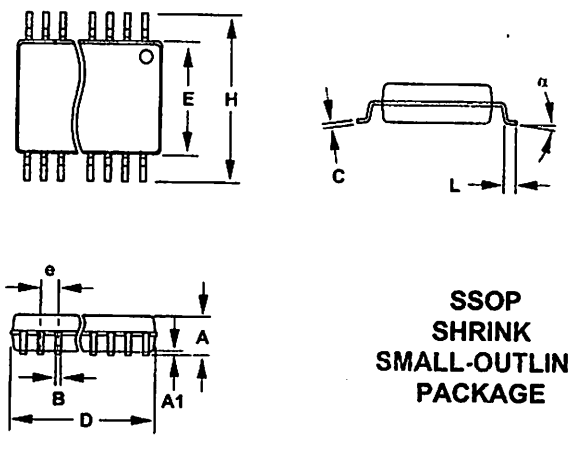


Plastic DIP
PLASTIC
DUAL-IN-LINE
PACKAGE
(0.300 in.)

DIM	INCHES		MILLIMETERS	
	MIN	MAX	MIN	MAX
A	–	0.200	–	5.08
A1	0.015	–	0.38	–
A2	0.125	0.175	3.18	4.45
A3	0.055	0.080	1.40	2.03
B	0.016	0.022	0.41	0.56
B1	0.045	0.065	1.14	1.65
C	0.008	0.012	0.20	0.30
D1	0.005	0.080	0.13	2.03
E	0.300	0.325	7.62	8.26
E1	0.240	0.310	6.10	7.87
e	0.100	–	2.54	–
eA	0.300	–	7.62	–
eB	–	0.400	–	10.16
L	0.115	0.150	2.92	3.81

PKG.	DIM	PINS	INCHES		MILLIMETERS	
			MIN	MAX	MIN	MAX
P	D	8	0.348	0.390	8.84	9.91
P	D	14	0.735	0.765	18.67	19.43
P	D	16	0.745	0.765	18.92	19.43
P	D	18	0.885	0.915	22.48	23.24
P	D	20	1.015	1.045	25.78	26.54
N	D	24	1.14	1.265	28.96	32.13

21-0043A



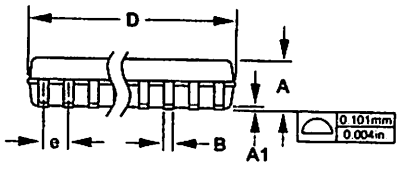
SSOP
SHRINK
SMALL-OUTLINE
PACKAGE

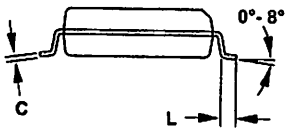
DIM	INCHES		MILLIMETERS	
	MIN	MAX	MIN	MAX
A	0.068	0.078	1.73	1.99
A1	0.002	0.008	0.05	0.21
B	0.010	0.015	0.25	0.38
C	0.004	0.008	0.09	0.20
D	SEE VARIATIONS			
E	0.205	0.209	5.20	5.38
e	0.0256 BSC		0.65 BSC	
H	0.301	0.311	7.65	7.90
L	0.025	0.037	0.63	0.95
α	0°	8°	0°	8°

DIM	PINS	INCHES		MILLIMETERS	
		MIN	MAX	MIN	MAX
D	14	0.239	0.249	6.07	6.33
D	16	0.239	0.249	6.07	6.33
D	20	0.278	0.289	7.07	7.33
D	24	0.317	0.328	8.07	8.33
D	28	0.397	0.407	10.07	10.33

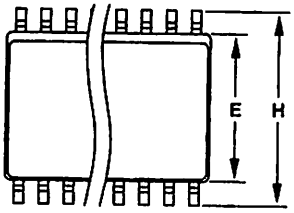
21-0056A

Package Information (continued)





DIM	INCHES		MILLIMETERS	
	MIN	MAX	MIN	MAX
A	0.093	0.104	2.35	2.65
A1	0.004	0.012	0.10	0.30
B	0.014	0.019	0.35	0.49
C	0.009	0.013	0.23	0.32
E	0.291	0.299	7.40	7.60
e	0.050		1.27	
H	0.394	0.419	10.00	10.65
L	0.016	0.050	0.40	1.27



Wide SO
SMALL-OUTLINE
PACKAGE
(0.300 in.)

DIM	PINS	INCHES		MILLIMETERS	
		MIN	MAX	MIN	MAX
D	16	0.398	0.413	10.10	10.50
D	18	0.447	0.463	11.35	11.75
D	20	0.496	0.512	12.60	13.00
D	24	0.598	0.614	15.20	15.60
D	28	0.697	0.713	17.70	18.10

21-0042A