

TUGAS AKHIR

TRANSFORMASI KOORDINAT SISTEM LOKAL KE SISTEM KOORDINAT TM-3° DENGAN MENGGUNAKAN PROGRAM VISUAL BASIC 6.0



**MILIK
PERPUSTAKAAN
ITN MALANG**

Disusun Oleh :

MUHAMMAD NAF'AN

98 . 25 . 051

**JURUSAN TEKNIK GEODESI
FAKULTAS TEKNIK SIPIL DAN PERENCANAAN
INSTITUT TEKNOLOGI NASIONAL MALANG
2005**

**TRANSFORMASI KOORDINAT SISTEM LOKAL
KE SISTEM KOORDINAT TM-3⁰
DENGAN MENGGUNAKAN PROGRAM VISUAL BASIC 6.0**

TUGAS AKHIR

Diajukan Sebagai Salah Satu Syarat Untuk Menyelesaikan
Program Pendidikan Sarjana Srata 1 (satu)
Bidang Teknik Geodesi

Disusun Oleh :

MUHAMMAD NAF' AN

98 . 25 . 051

Disetujui Oleh :

Dosen Pembimbing I



(Ir . Rinto Sasongko, MT)

Dosen Pembimbing II



(Ir . Leo Partimena, MSc)



Mengetahui :

Plh. Ketua Jurusan



(Ir . Leo Partimena, MSc)

**TRANSFORMASI KOORDINAT SISTEM LOKAL
KE SISTEM KOORDINAT TM-3⁰
DENGAN MENGGUNAKAN PROGRAM VISUAL BASIC 6.0**

TUGAS AKHIR

Dipertahankan di depan Penguji Tugas Akhir Jurusan Teknik Geodesi
Fakultas Teknik Sipil dan Perencanaan, Institut Teknologi Nasional Malang
Dan diterima untuk memenuhi salah satu syarat guna memperoleh gelar Sarjana
Strata 1 (Satu) Bidang Teknik Geodesi

Disusun Oleh :

MUHAMMAD NAF' AN

98 . 25 . 051

Disahkan Oleh :

Panitia Penguji Tugas Akhir



Ketua

(Ir . Agustina Nurul Hidayati, MTP)
Dekan Fakultas teknik Sipil dan Perencanaan

Penguji I

(Ir . Jasmani, MKom)

Sekretaris

(Ir . Leo Pantimena, MSc)
Plh. Kajar Teknik Geodesi

Penguji II

(Ir . Rinto Sasongko, MT)

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

LEMBAR PERSEMBAHAN

Segala Puji dan Syukur kepada sumber dari suara-suara hati yang bersifat mulia, sumber ilmu pengetahuan, sumber segala kebenaran, Sang Maha Cahaya, pilar nalar kebenaran dan kebaikan yang terindah, Sang Kekasih Tercinta yang tak terbatas pencahayaan cinta-Nya bagi umat-Nya Allah S.W.T.. Sholawat dan Salam untuk Rasulullah Muhammad S.A.W, beserta keluarga, sahabat dan semua para pengikutnya yang istiqomah. Waba'du.

Dibalik selesainya Laporan Tugas Akhir ini ada seorang yang begitu menjadi sumber kehidupan Saya, pembimbing utama Saya, pendidik Saya, yang telah membesar dan mendidik Saya "*Sang Matahariku*" Yang Mulia Ibunda Tercinta. Terima Kasih atas ridho, do'a dan kerja keras Ibunda sehingga Saya dapat setapak menggapai cita.

Tak lupa juga rasa terima kasih Saya untuk Kakakku yang tercantik, sedikit cerewet tapi baik hati, Adikku yang *rodo* manja, keras kepala, dan yang terpenting dialah "*Jubir*" Saya selama ini. Thank's For All.....!!!

Saya juga tak akan melupakan segala jasa dan kebaikan dari: "*Yang Sabar dan Baik Hati*" Binda Arifah, "*Yang Murah Hati*" Om Hasby, "*Yang Baik hati dan Murah Senyum*" Bi Hafni, "*Yang Baik Hati*" Binda Aminah, "*Yang Penyabar*" Om Buden, "*Yang Rendah Hati*" Binda Nurul, "*Yang Disiplin*" Om Lukman "*Sang Petualang*" Om Rony, "*Pamanku yang Penyabar*" Om Senuden, "*Si Kalem dan Penyabar*" Pak Demung, "*Ayah nya Anis*" Abang As, "*Bu Guru yang Cantik*" Kak Ily, "*Bu Guru yang periang*" Ka Idel, "*Yang Ganteng*" Dedy, "*Gadis Pemalu*" Nung, Adik-adikku: "*Si Trouble Maker*" Didit, "*Kutu Komik*" Fajri, "*Calon Tukang Insinyur*" Rizki, "*Si Cantik*" Ica, "*Si Otak Encer*" Evi, "*Gadis Keibuan*" Dian, "*Si Pembangkang*" Imam, "*Santri Calon Ustad*" Andri. Dan semuanya yang apabila tidak dapat Saya ingat, bukan maksud Saya untuk melupakan segala jasa dan dukungannya.

Thank's 4 U'r Support

Kata Pengantar

Segala Puji dan Syukur kepada sumber dari suara-suara hati yang bersifat mulia, sumber ilmu pengetahuan, sumber segala kebenaran, Sang Maha Cahaya, pilar nalar kebenaran dan kebaikan yang terindah, Sang Kekasih Tercinta yang tak terbatas pencahayaan cinta-Nya bagi umat-Nya Allah S.W.T.. Sholawat dan Salam untuk Rasulullah Muhammad S.A.W, beserta keluarga, sahabat dan semua para pengikutnya yang istiqomah. Waba'du.

Laporan Tugas Akhir ini berisi tentang *Program Hitung untuk melakukan Transformasi Koordinat dari Sistem Koordinat Lokal menjadi Sistem Koordinat TM-3*, dengan menggunakan Metode Transformasi Affine dan Metode Transformasi Lauf menggunakan bahasa pemrograman Visual Basic 6.0.

Penulis ingin mengucapkan banyak terima kasih kepada pihak-pihak yang telah banyak membantu penulis, baik selama proses penulisan laporan, pembuatan program dan analisa data, terutama kepada:

1. Ibu Ir. Agustina Nurul Hidayati, MTP,s selaku Dekan Fakultas Teknik Sipil dan Perencanaan, Institut Teknologi Nasional Malang.
2. Bapak Ir. Leo Pantimena, MSc, selaku Ketua Jurusan Teknik Geodesi sekaligus sebagai Pembimbing II (dua) kami.
3. Bapak Ir. Rinto Sasongko, MT, selaku Dosen Pembimbing I (satu), sekaligus sebagai Dosen Penguji II (dua) kami.
4. Bapak Christian T. Siahaan, ST., selaku Sekretaris Jurusan teknik Geodesi.

Geodesi

- 4. Bapak Christian T. Siaboo, ST, selaku Sekretaris Jurusan Teknik sekaligus sebagai Dosen Pengajar II (dua) kami
- 3. Bapak Ir. Kinto Sasongko, MT, selaku Dosen Pengajar I (satu) sekaligus sebagai Pengajar II (dua) kami
- 2. Bapak Ir. Leo Pambunga, MSc, selaku Ketua Jurusan Teknik Geodesi dan Perencanaan, Institut Teknologi Nasional Malang.
- 1. Ibu Ir. Yudianta Murni Hidayati, MT, S, selaku Dekan Fakultas Teknik Sipil

bemampuan program dan analisis data, terutama kepada:

telah banyak membantu penulis baik selama proses penulisan laporan

Penulis ingin mengucapkan banyak terima kasih kepada pihak-pihak yang

Transformasi dan menggunakan bahasa pemrograman Visual Basic 6.0

TM-3, dengan menggunakan Metode Transformasi Affine dan Metode

Transformasi Koordinat dan Sistem Koordinat Lokal menjadi Sistem Koordinat

Laporan Tugas Akhir ini berisi tentang Program Hitung untuk melakukan

asahat dan semua data yang bersangkutan yang terdapat pada

Sholawat dan Salam untuk Keseluruhan Muhammad S.A.W beserta keluarga,

Terakhir yang tak terlupa berterimakasih pada orang tua saya Ayah S.W.T.

Orang tua, pihak lain, kerabat dan kerabat yang terhormat, yang kesemuanya

bersifat mulia, sumber ilmu pengetahuan, sumber segala kerendahan, yang Maha

segala Puji dan Syukur kepada sumber dan sumber-sumber puji yang

Mats Pengantar

5. Bapak Ir. Jasmani, MKom, selaku Dosen Penguji I (satu) kami.
6. Bapak-bapak Dosen Pengajar di Jurusan Teknik Geodesi, terima kasih atas semua ilmunya dan keikhlasannya.
7. Mas Nur, terima kasih atas kesabaran dan kebaikannya, tanpa bantuan dari Mas Nur mungkin Tugas Akhir ini tidak akan pernah selesai.

Pada kesempatan ini juga Penulis ingin menyampaikan banyak terima kasih buat *teman-teman angkatan Songo Walu*. Thank's for All, *Gogon, Gabler, Dek Jun, Wuri, Syamsul, Wahyu, Maika, Syafii, Joko, Johan, Chasman, Topan, Dani, Akbar, Lulu', Lala, Dina, Ayu, Yohana, Teguh, Willy, Rully, Agus*, dan semuanya yang apabial tidak dapat Saya ingat bukan maksud Saya untuk melupakan teman-teman. Thank's U So Much.

Penulis juga tak akan melupakan jasa teman-teman yang selalu bersama baik dalam senang maupun duka *"HABITAT 269"*; *Shadoet yang baik hati "kapan kita PS_an lagi, Juayut yang telah dengan kebaikannya rela meminjam komputernya hingga jebol, tanpa komputer U ngkac taulah..?, Redro "Telo" terima kasih juga telah minjamin komputer 'n motornya. "Kang" Kholif (Slanker genteg), ojo turu ae, ndang garap "bendungannya". Yoppl kapan kita PS-an..., Bledozz terima kasih "Shogunnya" ojo pacaran ae, Ria, Sigit "Batman" ojo turu 'n game tok kuliah sing rajin, Cak Herman, Sony "sontong", Andry terima kasih atas backup-an CD islamnya*. Thank's 4 U'r Support.

Penulis menyadari bahwa laporan Tugas Akhir ini masih banyak kekurangannya baik dalam penulisan maupun isi dari laporan ini, karena itu Penulis sanagt mengharapkan saran dan kritik yang bisa kami gunakan untuk

5. Bapak Ir. Jasmari, M.Kom. selaku Dosen Pengaji I (satu) kami.
6. Bapak-bapak Dosen Pengajar di Jurusan Teknik Geodesi, terima kasih atas semua bimbingan dan keikhlasannya

7. Mas Nur, terima kasih atas kesabaran dan kebaikannya, tanpa bantuan dari Mas Nur mungkin Tugas Akhir ini tidak akan pernah selesai.

Pada kesempatan ini juga Penulis ingin menyampaikan banyak terima kasih buat teman-teman angkatan Songo Walaik. Thank's for All, Gagon, Gadien, Dek Jun, Wuri, Syamsul, Waiyul, Maika, Syaifi, Joko, Johan, Chasman, Topan, Dani, Akbar, Lulu, Lala, Dina, Ayu, Yohana, Teguh, Willy, Rully, Agus, dan semuanya yang apdial tidak dapat saya ingat bukan maksud saya untuk melupakan teman-teman. Thank's U So Much.

Penulis juga tak akan melupakan jasa teman-teman yang selalu bersama baik dalam senang maupun duka. HABIBAT 2007, Shadot yang baik hati "kapan kita PS-an lagi, Juayut yang telah dengan kebaikannya rela menjerim komputeranya hingga jebol, tanpa komputer U ngak lahirlah...? Rendra "Telo" terima kasih juga telah meminjam komputer in motomya. "Kang" Kholif (Stiker gentel), ojo tunu ad, abang garap "bendungannya", Yoppi kapan kita PS-an... Bledox terima kasih "Shodunya" ojo pacaran de, Riel, Sigit "Berman" ojo tunu in game lok kuliah sing rajin, Cak Herman, Sony "sontong", Andy terima kasih atas backup-an CD isminya. Thank's 4 UR Support.

Penulis menyadari bahwa laporan Tugas Akhir ini masih banyak kekurangannya baik dalam penulisan maupun isi dari laporan ini, karena itu Penulis sangat mengharapkan saran dan kritik yang bisa kami gunakan untuk

perbaikan kelak. Akhi kata, Penulis menyampaikan banyak terima kasih semoga Laporan Tuga Ahir ini dapat bermanfaat bagi pembaca semuanya.

Terima kasih atas segala bantuan yang telah diberikan kepada penulis.

Malang, Juli 2005

Penulis

Daftar Isi

Lembar Persetujuan	i
Lembar Pengesahan	ii
Lembar Persembahan	iii
Kata Pengantar	iv
Daftar Isi	v
BAB I. PENDAHULUAN	
I. 1. Latar Belakang	1
I. 2. Tujuan Penelitian	2
I. 3. Faedah Penelitian	2
I. 4. Batasan Masalah	3
I. 5. Tinjauan Pustaka	3
BAB II. DASAR TEORI	
II. 1. Sistem Pemetaan Nasional	5
II. 2. Sistem Koordinat Nasional	7
II. 3. Pengertian Umum Transformasi Koordinat	15
II. 4. Transformasi Koordinat 2 Dimensi (2-D)	20
II. 5. Transformasi Koordinat Metode Affine	26
II. 6. Transformasi Koordinat Metode Lauf	28
II. 6. 1. Metode Lauf Selisih Terbagi (Bilangan Kompleks)	30
II. 6. 2. Koreksi Harga Pendekatan Metode Numeris	32
II. 7. Bahasa Pemrograman Visual Basic 6.0	34
II. 7. 1. Elemen Program Dalam Visual Basic 6.0	35
II. 7. 2. Struktur Program Dalam Visual Basic 6.0	40

BAB III. PELAKSANAAN PENELITIAN

III. 1. Peralatan Penelitian	47
III. 2. Materi Penelitian	47
III. 2. 1. Diagram Alir Penelitian	49
III. 2. 2. Diagram Alir Transformasi Koordinat Metode Affine	50
III. 2. 3. Diagram Alir Transformasi Koordinat Metode Lauf Selisih Terbagi	51
III. 2. 4. Diagram Alir Transformasi Koordinat Metode Lauf Koreksi Sudut Jarak	52
III. 2. 5. Diagram Alir Paket Program	53
III. 2. 6. Penjelasan Diagram Alir Penelitian	54
III. 2. 7. Penjelasan Diagram Alir Paket Program	60
III. 3. Pembuatan Form Menu Utama	64
III. 3. 1. Langkah Pembuatan Menu Utama Program	64
III. 3. 2. Langkah Pembuatan Form Data Daerah	65
III. 3. 3. Langkah Pembuatan Form Metode Affine	67
III. 3. 4. Langkah Pembuatan Form Metode Lauf	71
III. 3. 5. Langkah Pembuatan Form Petunjuk Program	77

BAB IV. HASIL dan ANALISA

IV. 1. Menu Transformasi Koordinat Metode Affine	79
IV. 1. 1. Form Input Data	79
IV. 1. 2. Form Proses Hitung Transformasi Koordinat Metode Affine	81
IV. 2. Program Transformasi Koordinat Metode Lauf	82

IV. 2. 1. Sub Menu Input Data	83
IV. 2. 2. Sub Menu Proses Hitung Transformasi Koordinat	
Metode Lauf Selisih Terbagi	83
IV. 2. 3. Form Metode Lauf Koreksi Sudut dan Jarak	85
IV. 3. Uji Ketelitian	85
IV. 4. Analisa Hasil Perhitungan Transformasi Koordinat	
Metode Affine dan Metode Lauf	86
 BAB V. KESIMPULAN dan SARAN	
V. 1. Kesimpulan	92
V. 2. Saran	94
 Daftar Pustaka	
Lampiran-lampiran	

B A B I

PENDAHULUAN

I. 1. Latar Belakang

Peraturan Menteri Negara Agraria/ Kepala BPN No. 2 Tahun 1996 tentang pengukuran dan pemetaan untuk penyelenggaraan Pendaftaran Tanah, menyebutkan bahwa *Proyeksi Transverse Mercator* dengan lebar zone 3^0 merupakan sistem koordinat Nasional. Sebelum dikeluarkannya peraturan Menteri Negara Agraria/ Kepala BPN tentang pengukuran dan pemetaan untuk penyelenggaraan Pendaftaran Tanah yaitu peraturan PMNA/ Kepala BPN No. 2 tahun 1996, BPN dalam hal ini melakukan pengukuran dan pemetaan untuk penyelenggaraan pendaftaran tanah banyak menggunakan sistem lokal untuk daerah-daerah relative kecil. Hal ini dilakukan karena perlu adanya percepatan pelaksanaan pengukuran dan pemetaan dalam rangka pembuatan peta pendaftaran tanah, serta keterbatasan sumberdaya manusia dan peralatan yang digunakan pada lingkungan BPN.

Proyeksi TM- 3^0 merupakan sistem koordinat nasional, sehingga peta-peta dasar pendaftaran dan peta-peta pendaftaran tanah yang telah dibuat dalam sistem lokal perlu ditransformasikan ke sistem proyeksi nasional. Untuk melakukan transformasi peta ke sistem proyeksi TM- 3^0 dibutuhkan waktu yang cukup lama dan biaya yang cukup besar, oleh karena itu diperlukan metode transformasi yang dapat dilakukan secara sederhana dengan menggunakan fasilitas peralatan yang telah dimiliki oleh BPN di tingkat daerah.

Pada pasal 4, ayat (3), (4) dan (5) PMNA N0.2 Tahun 1996, menyebutkan bahwa pengukuran titik dasar teknik Orde 4, pada prinsipnya dilaksanakan dalam sistem koordinat nasional, dengan mengikatkan ke titik dasar teknik Orde 3. Apabila tidak memungkinkan, pengukuran titik dasar teknik Orde 4, dilaksanakan dalam sistem koordinat lokal, dimana dikemudian hari harus ditransformasikan kedalam sistem koordinat nasional.

Mengingat tata-cara transformasi untuk “memindahkan” koordinat dari sistem koordinat lokal ke dalam sistem koordinat nasional (TM-3⁰) tersebut di atas dalam hal ini menggunakan metode Affine dan Metode Lauf, melalui proses penyelesaian yang panjang serta untuk mempermudah melakukan transformasi koordinat titik maka diperlukan suatu *Software* yang dapat mempercepat dan mempermudah melakukan proses transformasi koordinat titik tersebut.

I. 2. Tujuan Penelitian

Tujuan dari penelitian ini adalah untuk melakukan Transformasi Koordinat dari Sistem Lama (Lokal) ke Sistem Proyeksi Baru (TM-3⁰) dengan metode Transformasi Affine dan Metode Lauf menggunakan bahasa pemrograman Visual Basic 6.0

I. 3. Faedah Penelitian

Dengan adanya penelitian ini akan dapat diketahui cara dan metode yang dapat digunakan untuk melakukan transformasi koordinat sistem lama (lokal) ke sistem proyeksi baru (TM-3⁰) dengan mudah dan cepat.

I. 4. Batasan Masalah

Batasan masalah yang digunakan penulis dalam melakukan penelitian ini adalah:

1. Penentuan besaran-besaran parameter untuk transformasi koordinat peta dari sistem lama (lokal) ke sistem proyeksi baru (TM-3⁰).
2. Penggunaan Metode Transformasi Affine dan Metode Transformasi Lauf untuk melakukan transformasi koordinat.
3. Dalam proses perhitungan transformasi koordinat dari sistem lama (lokal) ke sistem proyeksi baru (TM-3⁰) menggunakan bahasa pemrograman Visual Basic 6.0.

I. 5. Tinjauan Pustaka

Peraturan Menteri Negara Agraria/ Kepala BPN No. 2 Tahun 1996 tentang pengukuran dan pemetaan untuk penyelenggaraan Pendaftaran Tanah, menyebutkan bahwa Proyeksi Transverse Mercator dengan Lebar Zone 3⁰ (TM-3⁰) merupakan sistem koordinat nasional, sehingga Peta-peta Dasar Pendaftaran dan Peta-peta Pendaftaran Tanah yang telah dibuat dalam sistem lokal perlu ditransformasikan ke Sistem Proyeksi Nasional. [Hendriatiningsih, 2002].

Dalam rangka pelaksanaan pengadaan jaring kontrol geodesi nasional orde-2 untuk densifikasi 10 Km dan Orde-3 untuk densifikasi 2 Km dengan metode pengukuran Global Positioning System (GPS) dan penetapan sistem proyeksi *Transverse Mercator* dengan lebar zone 3 (tiga) derajat (TM-3⁰), maka perlu segera disusun dan disiapkan mengenai segala sesuatu yang berhubungan dengan pemakaian sistem proyeksi tersebut dan penggunaan

jaring titik kontrol geodesi yang merupakan titik-titik Dasar Teknik Nasional sebagai referensi ikatan untuk berbagai kegiatan Pengukuran dan Pemetaan Bidang Pendaftaran Tanah terutama untuk pembuatan Peta Dasar Pendaftaran Tanah [*Perawiranagara, 2000*].

Transformasi koordinat 2 (dua) dimensi banyak digunakan untuk mentransformasikan sistem koordinat lokal 2 (dua) dimensi kedalam sistem koordinat 2 (dua) dimensi lainnya [*Tjahjadi, 1999*]

Visual Basic 6.0 merupakan salah satu bahasa pemrograman tercepat dan termudah untuk membuat aplikasi dalam Microsoft Windows. Dengan menggunakan metode GUI (*Graphical User Interface*), Visual Basic 6.0 memudahkan pemrograman untuk berinteraksi langsung dengan elemen-elemen untuk setiap bentuk pemrograman [*Pamungkas, 2000*].

B A B II

DASAR TEORI

II. 1. Sistem Pemetaan Nasional.

Menteri Negara Agraria/ Kepala Badan Pertanahan Nasional mengeluarkan peraturan tentang pengukuran dan pemetaan untuk penyelenggaraan pendaftaran tanah (PMNA/ Kepala BPN No. 3 Tahun 1997) berdasarkan pertimbangan-pertimbangan sebagai berikut :

1. Bahwa untuk percepatan penyelenggaraan pendaftaran tanah, perlu percepatan pelaksanaan pengukuran dan pemetaan dalam rangka pembuatan peta pendaftaran;
2. Bahwa untuk menuju tercapainya sistem pencatatan nasional, diperlukan keterpaduan dalam sistem pengukuran dan pemetaan dengan instansi lain, khususnya Bakosurtanal yang menyediakan sistem kerangka dasar pemetaan nasional (JKGN);
3. Bahwa dengan kemajuan teknologi sekarang ini, mengenai pengukuran dan pemetaan yang ada, perlu disempurnakan, agar dapat menunjang percepatan pembuatan peta pendaftaran yang memenuhi syarat.

Sedangkan pada pasal 2 (dua) dan pasal 3 (tiga) Peraturan Menteri Negara Agraria tersebut menyebutkan bahwa JKGN Orde 0 (nol) dan Orde 1 (satu) Bakosurtanal dirapatkan menjadi titik dasar teknik Orde 2 (dua), Orde 3 (tiga) dan Orde 4 (empat). Titik dasar teknik tersebut mempunyai fungsi sebagai titik kontrol atau sebagai titik ikat pengukuran untuk pemetaan dalam rangka penyelenggaraan pendaftaran tanah serta untuk keperluan

rekonstruksi batas. Titik dasar teknik Orde 2 (dua) dilaksanakan dengan kerapatan ± 10 kilometer dengan mengikat ke titik dasar teknik Orde 0 (nol) dan Orde 1 (satu). Selanjutnya titik dasar Orde 2 (dua) dirapatkan menjadi titik dasar teknik Orde 3 (tiga) dengan kerapatan $\pm 1-2$ kilometer dengan mengikat ke titik dasar teknik Orde 2 (dua). Titik dasar teknik Orde 3 (tiga) dirapatkan menjadi titik dasar teknik Orde 4 (empat) dengan kerapatan hingga 150 meter dengan mengikat ke titik dasar teknik Orde 3 (tiga). Dengan cara pengikatan di atas, titik-titik dasar teknik Orde 0 (nol) sampai dengan Orde 4 (empat) merujuk pada Datum Geodesi Nasional 1995. Titik-titik dasar teknik tersebut kemudian disebut sebagai *titik dasar teknik nasional*.

Pada pasal 3 PMNA No. 3 Tahun 1997 menyebutkan antara lain bahwa pengukuran titik dasar teknik Orde 4 pada prinsipnya dilaksanakan dalam sistem nasional dengan mengikat ke titik dasar teknik Orde 3. Apabila tidak memungkinkan, pengukuran titik dasar teknik Orde 4 dapat dilaksanakan dalam sistem Koordinat lokal yang selanjutnya harus *ditransformasikan* ke dalam sistem koordinat nasional. Disamping itu, pemetaan dalam sistem nasional, harus memperhatikan pula *zoning* proyeksi TM-3⁰.

Pada Pasal 7, Peraturan Menteri Negara Agraria / Kepala BPN No. 3 Tahun 1997 menyebutkan bahwa :

1. Sistem koordinat nasional menggunakan sistem koordinat Proyeksi *Transverse Mercator Nasional* dengan lebar zone 3⁰ (tiga derajat) dan selanjutnya disebut sebut TM-3⁰.
2. Meridian sentral zone TM-3⁰ terletak 1,5 derajat di barat dan timur meridian sentral zone UTM yang bersangkutan.

3. Besaran faktor skala di meridian sentral yang digunakan dalam zone TM-3⁰ adalah 0,9999.
4. Titik nol semu yang digunakan mempunyai koordinat (x) = 200.000 meter barat dan (y) = 1.500.000 meter selatan.
5. Model matematik bumi sebagai bidang referensi adalah spheroid pada datum WGS '84 dengan parameter $a = 6.378.137$ meter dan $f = 1/298,25722357$.

Dengan ketetapan-ketetapan di atas, Badan Pertanahan Nasional telah mempunyai *Sistem Pemetaan Nasional*. Penyelenggaraan pemetaan dalam Sistem Nasional, harus dilakukan secara konsisten, misalnya prosedur dalam pengolahan data.

Data ukuran terestris, secara teoritis tidak dapat digunakan langsung untuk keperluan pemetaan dalam sistem nasional, akan tetapi perlu proses bertahap, yaitu: **proses reduksi** data ukuran ke ellipsoid referensi (WGS '84), kemudian **proses pemberian koreksi** pada besaran ellipsoid referensi untuk menjadikan besaran yang bersangkutan pada proyeksi TM-3⁰.

II. 2. Sistem Koordinat Nasional

Sistem koordinat adalah sistem yang digunakan untuk menyatakan posisi titik-titik tertentu dipermukaan bumi relatif terhadap sebuah sistem tertentu yang didefinisikan berdasarkan suatu fenomena tertentu yang dapat diamati, mantap, dan dapat disusun kembali dengan mudah bila diperlukan.

Metode-metode proyeksi peta yang umumnya dipakai di Indonesia untuk pemetaan adalah proyeksi Polyeder, proyeksi Mercator dan proyeksi Universal Transverse Mercator (UTM).

Proyeksi Mercator adalah proyeksi silinder normal konform yang menyinggung pada lingkaran ekuator dan beracuan seperti proyeksi Polyeder. Proyeksi Mercator mulai digunakan tahun 1886 bagi pemetaan Kalimantan Barat dan kemudian untuk pemetaan Irian Barat. Selain itu, proyeksi Mercator telah digunakan untuk perataan jaringan triangulasi Nusa Tenggara Barat.

Dalam melakukan Hitung Geodesi untuk keperluan pemetaan (jaring kontrol geodesi), bumi fisis geodesi diwakili oleh permukaan matematika yang mempunyai ukuran dan bentuk yang mendekati geoid. Model matematik bumi tersebut adalah sebuah ellips putar dengan sumbu pendek sebagai sumbu putar yang disebut *ellipsoida* atau *spheroid*. Ellipsoida yang mempunyai ukuran dan bentuk yang digunakan untuk hitung geodesi disebut *Ellipsoid Referensi* atau *Ellipsoid Rujukan* (ER). Ellipsoid referensi sebagai bidang referensi yang digunakan oleh Badan Pertanahan Nasional (BPN) adalah ellipsoid/spheroid pada datum WGS '84. Bentuk dan ukuran ellipsoid referensi, secara umum ditentukan oleh besaran parameter tertentu.

Parameter tetapan Ellipsoida Referensi WGS '84 adalah sebagai berikut :

Simbol	Nilai	Satuan
a	6.378.137	Meter
f	1 / 298,257.223.570	Meter
b	6.356.752,314	Meter
c	6.399593,626	Meter
e²	0,006.694.380	Meter
e'²	0,006.739.497	Meter

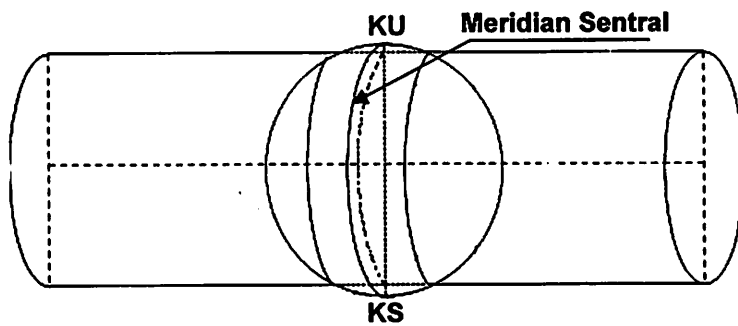
Tabel 1. Parameter Tetapan Referensi WGS '84

Badan Pertanahan Nasional telah menetapkan melalui PMNA No. 2 Tahun 1996 bahwa untuk pembuatan Peta Dasar Pendaftaran dan Peta Pendaftaran guna penyelenggaraan pendaftaran tanah digunakan Proyeksi Transverse Mercator dengan ketentuan-ketentuan tertentu yaitu, faktor skala di meridian sentral adalah sebesar 0,9999 dan wilayah cakupannya (lebar zone) adalah selebar 3° . Proyeksi ini disebut juga Proyeksi TM- 3° .

Penggunaan sistem Proyeksi TM- 3° untuk keperluan pembuatan peta Pendaftaran Tanah di seluruh Indonesia sangat cocok karena, tidak terlalu banyak reduksi ukuran (sudut, jarak) terhadap bidang proyeksi, jadi yang perlu diperhitungkan hanya:

1. Reduksi jarak terhadap permukaan air laut rata-rata (MSL) atau yang biasa disebut reduksi jarak ke geoid.
2. Reduksi jarak akibat perbesaran skala (faktor skala)
3. Reduksi azimuth akibat konvergensi meridian diperhitungkan apabila azimuth yang digunakan untuk hitungan adalah hasil pengamatan matahari, tetapi dengan adanya densifikasi titik dasar teknis per 2 Km penggunaan metode hitung polygon tanpa azimuth pengamatan tidak perlu lagi koreksi konvergensi meridian.

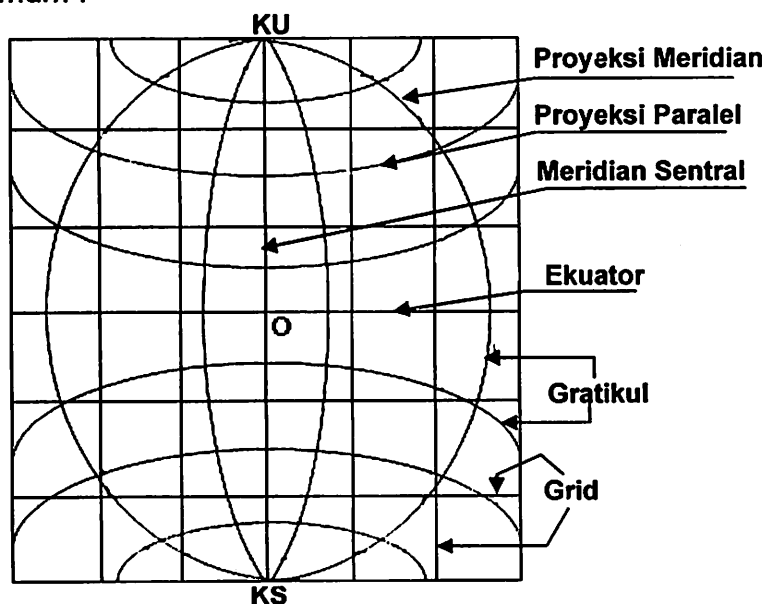
Secara Geometrik, Proyeksi TM- 3° sama dengan proyeksi UTM. Perbedaannya terletak pada penetapan faktor skala di meridian sentral dan penetapan lebar wilayah cakupannya. Proyeksi UTM dapat dibayangkan sebagai proyeksi silinder transversal yang memotong model matematis bumi seperti pada gambar dibawah ini, dan mempunyai ciri sebagai proyeksi konform (proyeksi yang menghasilkan sudut sama)



Gambar 1. Proyeksi Silinder Transversal Yang Memotong

Proyeksi meridian sentral dan ekuator masing-masing merupakan garis-garis lurus yang tegak lurus. Sedangkan proyeksi meridian dan paralel lainnya masing-masing merupakan kurva-kurva yang juga saling tegak lurus.

Gambar dibawah ini, menunjukkan tatanan gratikul dan grid pada proyeksi TM secara umum :



Gambar 2. Tatanan Gratikul dan Grid Pada Proyeksi TM

Proyeksi TM pada umumnya menunjukkan distorsi jarak yang makin membesar ke arah timur maupun ke arah barat meridian sentral. Masalah inilah yang membatasi lebar wilayah cakupan.

Pada PMNA No. 2 Tahun 1996 menetapkan bahwa batas wilayah cakupan adalah $1^{\circ},5$ ke arah timur dan $1^{\circ},5$ ke arah barat dari meridian sentrai. Wilayah cakupan selebar 3° disebut dengan zone.

Di dalam sistem TM- 3° , wilayah Indonesia tercakup dalam 16 (enam belas). Di dalam setiap zone, didefinisikan sistem koordinat kartesian dua dimensi yang dinyatakan dalam bentuk sumbu X dan Y untuk **Koordinat Sejati** sumbu U dan T untuk **Koordinat Semu**.

Sebagai sumbu Y adalah proyeksi meridian sentral dan sebagai sumbu X adalah proyeksi lingkaran ekuator. Titik potongnya merupakan titik nol sejati. Di samping titik nol sejati didefinisikan pula adanya titik nol semu yang **200.000 meter sebelah barat meridian sentral dan 1.500.000 meter sebelah selatan ekuator**.

Ketentuan-ketentuan proyeksi TM- 3° adalah sebagai berikut:

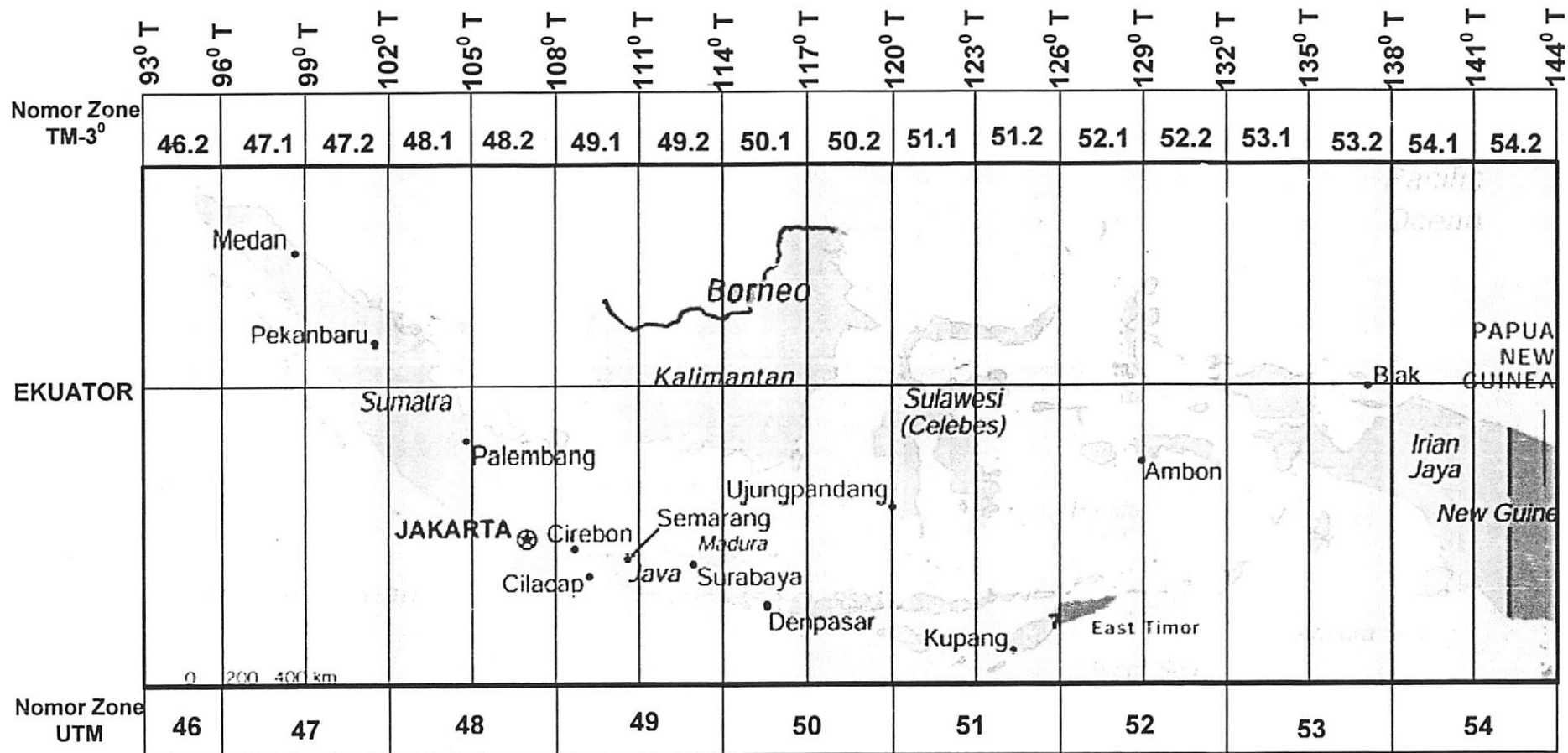
- a. Proyeksi : Transverse Mercator dengan Lebar Zone 3°
- b. Sumbu Pertama (X) : Meridian sentral dari setiap zone
- c. Sumbu Kedua (Y) : Ekuator
- d. Satuan : Meter
- e. Absis Semu (T) : 200. 000 meter + X
- f. Ordinat Semu (U) : 1. 500. 000 meter + Y
- g. Faktor Skala pada Meridian Sentral : 0, 9999
- h. Penomoran Zone : - Dimulai **Zone no. 46.2** mulai bujur **93° Timur s/d bujur 96° Timur.**

- **Zone no. 47.1** mulai bujur **96⁰**
Timur s/d 99⁰ Timur.
- **Zone no. 47.2** mulai bujur **99⁰**
Timur s/d 102⁰ Timur.
- **Zone no. 48.1** mulai bujur **102⁰**
Timur s/d 105⁰ Timur.
- **Demikian sampai Zone terakhir no.**
54.1 mulai bujur **138⁰ Timur s/d**
141⁰ Timur.

i. Batas Lintang : Lintang 6⁰ U dan lintang 11⁰ S.

Penomoran zone TM-3⁰ mengikuti penomoran zone UTM. Dua angka pertama merupakan nomor zone UTM_nya dan satu angka terakhir dengan didahului tanda titik merupakan nomor zone TM-3⁰l.

Dalam penerapan sistem grid UTM, seluruh wilayah Indonesia terbagi dalam 9 (sembilan) zone, yang masing-masing memiliki lebar 6⁰ bujur, mulai dari meridian 90⁰ bujur timur sampai dengan meridian 144⁰ bujur timur Meridian. Seperti pada gambar dibawah ini:



Tata Letak dan Penomoran Zone UTM & TM-3°

Wilayah Indoneisa

Penomoran zone proyeksi TM-3⁰ berbasis nomor zoner UTM 46-54.

NOMOR ZONE		MERIDIAN CENTRAL		MERIDIAN BATAS ZONE			
				BARAT		TIMUR	
UTM	TM-3 ⁰	UTM	TM-3 ⁰	UTM	TM-3 ⁰	UTM	TM-3 ⁰
46		93 ⁰		90 ⁰		96 ⁰	
	46.2		94 ⁰ 30'		93 ⁰		96 ⁰
47	47.1	99 ⁰	97 ⁰ 30'	96 ⁰	96 ⁰	102 ⁰	99 ⁰
	47.2		100 ⁰ 30'		99 ⁰		102 ⁰
48	48.1	105 ⁰	103 ⁰ 30'	102 ⁰	102 ⁰	108 ⁰	105 ⁰
	48.2		106 ⁰ 30'		105 ⁰		108 ⁰
49	49.1	111 ⁰	109 ⁰ 30'	108 ⁰	108 ⁰	114 ⁰	111 ⁰
	49.2		112 ⁰ 30'		111 ⁰		114 ⁰
50	50.1	117 ⁰	115 ⁰ 30'	114 ⁰	114 ⁰	120 ⁰	117 ⁰
	50.2		118 ⁰ 30'		117 ⁰		120 ⁰
51	51.1	123 ⁰	121 ⁰ 30'	120 ⁰	120 ⁰	126 ⁰	123 ⁰
	51.2		124 ⁰ 30'		123 ⁰		126 ⁰
52	52.1	129 ⁰	127 ⁰ 30'	126 ⁰	126 ⁰	132 ⁰	129 ⁰
	52.2		130 ⁰ 30'		129 ⁰		132 ⁰
53	53.1	135 ⁰	133 ⁰ 30'	132 ⁰	132 ⁰	138 ⁰	135 ⁰
	53.2		136 ⁰ 30'		135 ⁰		138 ⁰
54	54.1	141 ⁰	139 ⁰ 30'	138 ⁰	138 ⁰	144 ⁰	141 ⁰
	54.2						

Tabel 2. Penomoran Zone Untuk Sistem Grid UTM dan TM-3⁰ di Indonesia

Di dalam penyelenggaraan Kerangka Horizontal Geodesi (misalnya, Kerangka Dasar Geodesi Nasional), terlebih dahulu perlu didefinisikan suatu *Sistem Referensi* yang lengkap dan jelas. Sistem referensi yang menyangkut aspek geometrik meliputi :

1. Ellipsoid Referensi yang digunakan, meliputi kedudukan dan orientasinya terhadap bumi (geoid), ukuran dan bentuknya (jari-jari ekuator = a, dan penggempangan = f).
2. Sistem Koordinat yang digunakan (sistem koordinat proyeksi, sistem koordinat geodesi, dan lain-lain).
3. Prosedur pengukuran serta model perhitungan.
4. Konstanta-konstanta lain yang digunakan.

Untuk mengetahui koordinat dari suatu sistem ke sistem lainnya disebut dengan *Transformasi Koordinat*. Untuk dapat melakukan transformasi koordinat diperlukan pengetahuan tentang hubungan geometrik antara sistem koordinat satu dengan lainnya yang meliputi :

1. Pergeseran titik nol (*translation*).
2. Perbesaran jarak (*scalling*).
3. Perputaran sumbu koordinat (*rotation*).

Besaran-besaran tersebut dinamakan dengan *parameter transformasi*.

Transformasi mempunyai peranan penting di dalam pemetaan karena adanya kebutuhan untuk mengubah nilai koordinat suatu titik dari suatu sistem koordinat atau datum ke sistem koordinat atau datum lainnya. Transformasi koordinat adalah perubahan suatu koordinat ke suatu sistem koordinat lainnya yang diakibatkan oleh adanya faktor rotasi, translasi, dan skala [Perawiranagara,2000]

Sistem proyeksi peta TM-3⁰ adalah sistem proyeksi Universal Transverse Mercator dengan ketentuan faktor skala di meridian sentral = 0,9999 dan lebar zone = 3⁰. Sistem proyeksi ini, sejak tahun 1997 digunakan oleh Badan Pertanahan Nasional (BPN) sebagai sistem koordinat nasional menggunakan datum absolut DGN-95.

II. 3. Pengertian Umum Transformasi Koordinat

Cukup banyak kemungkinan dalam mengartikan “transformasi” sehingga untuk mencegah salah pengertian, sebaiknya ditegaskan terlebih dahulu pengertian istilah transformasi ini. Transformasi secara umum adalah

perubahan suatu bentuk dan ukuran ke bentuk dan ukuran lain, baik secara fisik maupun secara non-fisik [Soedomo & Sudarman, 2004]

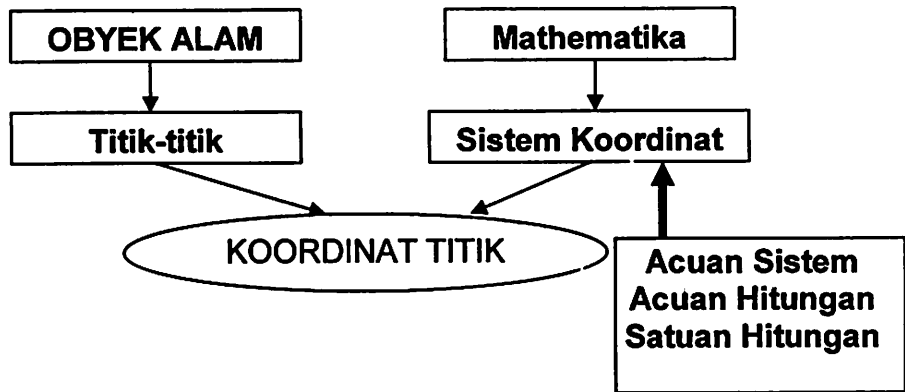


Diagram 3. Terjadinya Koordinat Titik

Setiap metoda transformasi memiliki perbedaan hasil tersendiri (spesifik) mengingat dasar pemikiran dan jenis kasus yang berbeda. Yang perlu ditekankan, berdasarkan diagram 1 di atas, bahwa dalam mengartikan sistem koordinat terdapat beberapa aspek yang penting artinya, yaitu:

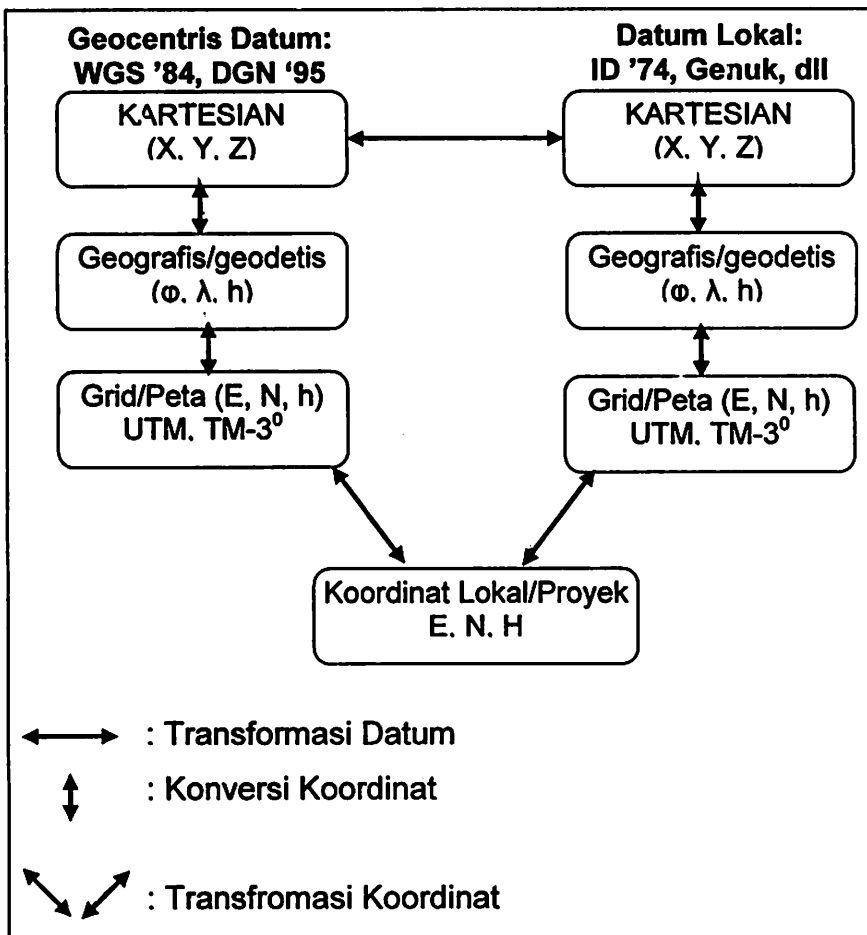
- a. Acuan sistem koordinat
- b. Satuan (unit) untuk pernyataan koordinat
- c. Acuan hitungan koordinat, yang erat kaitannya dengan :
 - 1. Besaran ukuran/parameter, dan
 - 2. Bentuk dasar hitungan

Dari beberapa aspek penting maka dapat ditarik pengertian “sistem koordinat”. Sistem Koordinat adalah suatu kesatuan yang dibentuk sedemikian rupa dalam menyatakan letak atau posisi obyek yang tidak tergantung pada obyek lainnya. Pada dasarnya, hal pertama yang perlu diperhatikan pada sistem koordinat adalah:

- a. Bidang acuan hitungan yang digunakan
- b. Peletakan (penempatan) garis atau bidang referensi koordinat
- c. Tata cara menyatakan posisi obyek

1. Besaran
2. Satuan

Dengan mengacu pada gambar dibawah ini, dapat dibedakan antara pengertian konversi koordinat dan transformasi koordinat, serta transformasi datum.



Gambar 4. Konsep Konversi Koordinat dan Transformasi Koordinat serta Transformasi Datum

Transformasi koordinat dilakukan dari sistem koordinat grid/ peta ke sistem koordinat lokal atau sebaliknya. Konversi koordinat adalah mengkonversi nilai koordinat suatu obyek dari sistem koordinat kartesian ke

sistem koordinat geografis/ geodetis ke sistem koordinat grid/ peta dan sebaliknya (kartesian <-> geografis/ geodetis <-> grid/ peta) didalam suatu sistem datum yang sama.

Transformasi datum adalah mengubah nilai suatu koordinat (koordinat kartesian) dari suatu datum ke datum lainnya yang berbeda (misalnya: ID'74<-> DGN'95). Koordinat yang ditransformasikan haruslah dalam sistem koordinat kartesian, karena hanya dalam sistem koordinat ini dapat dihasilkan perhitungan yang teliti.

Transformasi koordinat adalah perubahan suatu sistem koordinat ke suatu sistem koordinat lainnya yang diakibatkan oleh adanya faktor rotasi, translasi dan skala. Aplikasi transformasi koordinat banyak sekali digunakan pada bidang Geodesi yang berhubungan dengan sistem koordinat dua dimensi maupun sistem koordinat tiga dimensi (koordinat ruang). Beberapa persyaratan yang harus dipenuhi pada transformasi koordinat antara lain :

1. Koordinat titik ikat (*definitif*) dengan koordinat yang akan ditransformasikan harus memenuhi syarat geometris, artinya pada saat sistem koordinat yang akan ditransformasikan ditentukan pengukurannya baik dengan cara terestris atau dengan metode lainnya (misal metode GPS) harus terhindar dari kesalahan besar (blunder) dan memenuhi spesifikasi teknis ketelitian yang telah ditetapkan berdasarkan orde ketelitian pengukuran, misalnya dari hasil pengukuran polygon. Salah penutup sudut (f_s), salah penutup koordinat (f_x, f_y) dan ketelitian linier koordinat, sedangkan untuk

pengukuran GPS: standart deviasi setiap koordinat dan seluruh matrik varian covarian (VCV) dari setiap pengukuran baseline.

2. Mempunyai hubungan (korelasi) antara sistem koordinat yang ditransformasi misalnya sistem koordinat lokal dengan sistem koordinat definitif.
3. Orde ketelitian koordinat definitif sebagai titik ikat harus lebih tinggi atau minimum sama dengan ketelitian koordinat yang akan ditransformasikan, karena ketelitian hasil transformasi selain ditentukan oleh sistem koordinat sementara juga oleh sistem koordinat definitif yang digunakan sebagai titik ikat. Untuk memenuhi syarat geometris (rotasi, skala dan translasi), maka jumlah titik yang digunakan adalah 3 (tiga) titik, apabila jumlah titik ikat yang digunakan lebih tiga, maka penyelesaian harus dengan metode perataan kuadrat terkecil.

Beberapa transformasi koordinat yang banyak hubungannya dengan kegiatan bidang pengukuran dan pemetaan antara lain:

1. Transformasi dari sistem koordinat tiga dimensi (X, Y, Z) dengan pusat koordinat geosentris, hasil penentuan posisi dengan satelit GPS ke sistem koordinat Geodetik (L, B, h).
2. Transformasi dari sistem koordinat geodetik (L, B) ke sistem koordinat proyeksi (X, Y) atau sebaliknya (X, Y) ke (L, B), misalnya transformasi koordinat geodetik (L, B, h) ke sistem koordinat proyeksi (X, Y) UTM atau TM.

3. Transformasi dari suatu sistem koordinat proyeksi ke sistem proyeksi lainnya, misal dari UTM ke TM, dari UTM ke Polyeder, dan lain-lain.
4. Transformasi koordinat antara satu zone ke zone lainnya yang berdekatan dalam sistem proyeksi UTM maupun proyeksi TM.
5. Transformasi dari suatu sistem koordinat ke sistem koordinat lainnya dalam bentuk aplikasi banyak digunakan untuk transformasi:
 - Sistem koordinat tiga dimensi: Triangulasi udara dan perataan bloknya.
 - Sistem koordinat dua dimensi: Transformasi dari sistem koordinat lokal ke sistem koordinat nasional baik secara numeris maupun secara grafis.

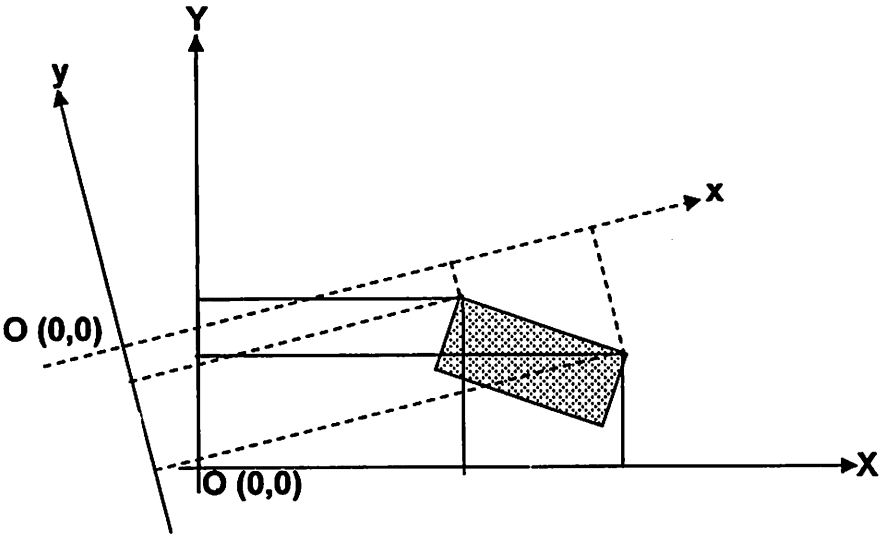
II. 4. Transformasi Koordinat 2 Dimensi (2-D)

Pada pasal 4, ayat (3), (4) dan (5) PMNA No. 2 tahun 1996, menyebutkan bahwa pengukuran titik dasar teknik Orde 4, pada prinsipnya dilaksanakan dalam sistem koordinat nasional, dengan mengikatkan ke titik dasar teknik Orde 3. Apabila tidak memungkinkan, pengukuran titik dasar teknik Orde 4, dilaksanakan dalam sistem koordinat lokal, yang kemudian hari harus ditransformasikan ke dalam sistem koordinat nasional.

Titik dasar teknik Orde 1, Orde 2, Orde 3, disebut *Titik Dasar Teknik Nasional*, disingkat *TDT.N*, sedangkan titik dasar teknik Orde 4 bila koordinat titiknya belum ditransformasikan ke dalam sistem koordinat nasional, disebut *Titik Dasar Teknik Lokal* disingkat *TDT.L*.

Suatu persyaratan utama untuk dapat mentransformasikan sekumpulan koordinat dari sistem (x, y) menjadi sekumpulan koordinat di sistem lainnya (X, Y) adalah ketersediaan titik-titik sekutu (*commont point*). Kedua sistem koordinat (x, y) , (X, Y) tersebut mungkin dapat tidak paralel, titik pusat salib sumbunya mungkin tidak brimpit, dan mungkin ada perubahan skala salib-salib sumbunya.

II. 4. 1. Paramater Transformasi



Gambar 5. Ilustrasi Perubahan Koordinat

Keterangan Gambar 5:

- x, y : Sistem Koordinat Obyek Posisi Lama
- X, Y : Sistem Koordinat Obyek Posisi Baru

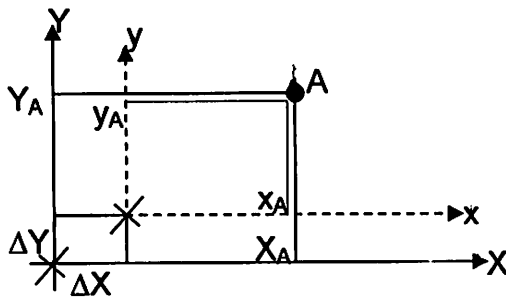
Seperti pada gambar (5) di atas, bahwa perubahan dalam suatu transformasi yang terlihat langsung, adalah terjadinya pergeseran dan perputaran. Pada transformasi yang umum, terdapat parameter lainnya yang tidak terlihat, yaitu perbesaran skala sistem koordinat. Dengan demikian parameter transformasi yang harus diperhatikan adalah:

- a. Parameter translasi (pergeseran)
- b. Parameter rotasi (perputaran)
- c. Parameter perbesaran (scaling factor)

A. Translasi

Translasi sistem koordinat, merupakan besarnya pergeseran yang terjadi pada semua titik yang dinyatakan dalam sistem koordinat tersebut. Translasi ini dapat diartikan pula sebagai besarnya pergeseran **titik pusat koordinat** sistem lama ke sistem baru.

Akibat dari bergesernya pusat koordinat, maka koordinat seluruh titik yang dinyatakan terhadap sistem koordinat tersebut akan ikut bergeser, dengan pergeseran yang sama.



Gambar 6. Faktor Translasi

Keterangan Gambar 6 :

- x, y : Sistem Koordinat Obyek Posisi Lama
- X, Y : Sistem Koordinat Obyek Posisi Baru
- x_A, y_A : Koordinat Titik "A" pada Sistem Koordinat Lama
- X_A, Y_A : Koordinat Titik "A" pada Sistem Koordinat Baru

Gambar (6) diatas memperlihatkan perubahan pusat koordinat sistem koordinat lama (salib sumbu x, y) ke dalam sistem koordinat baru (salib sumbu X, Y).

Pergeseran/translasi transformasi dinyatakan sebesar ΔX , ΔY . Besaran translasi ini merupakan “**besar koordinat titik pusat koordinat lama pada sistem koordinat baru**”. Hal ini akan berakibat pada koordinat titik yang dinyatakan di dalam sistem koordinat lama (misal titik A), yaitu $x_A \rightarrow X_A$ dan $y_A \rightarrow Y_A$.

Secara matematis, dinyatakan sebagai :

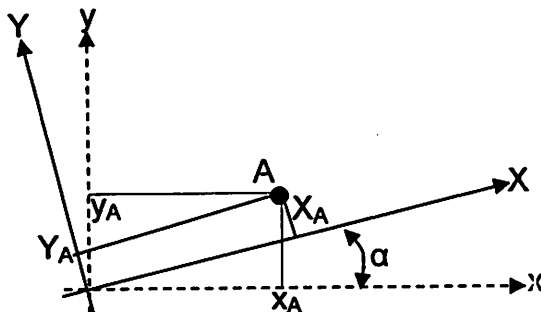
$$\begin{aligned} X_A &= x_A + \Delta X \\ Y_A &= y_A + \Delta Y \end{aligned} \quad (2.1)$$

Rumus di atas, digunakan untuk menghitung koordinat titik-titik dari sistem lama ke sistem baru.

B. Rotasi

Parameter transformasi berikutnya berupa “putaran” salib sumbu lama terhadap salib sumbu baru. Gambar (7) di bawah ini memperlihatkan putaran salib sumbu xoy menjadi XOY dengan besar putaran (rotasi) α . Rotasi pada gambar (7) di bawah merupakan **rotasi absolut**, yaitu rotasi yang dinyatakan pada pusat koordinat. Akibat dari rotasi absolut ini, maka seluruh obyek yang dinyatakan terhadap sistem koordinat tersebut akan ikut berputar. Secara matematis, dinyatakan sebagai:

$$\begin{aligned} X_A &= x_A \cdot \cos \alpha - y_A \cdot \sin \alpha \\ Y_A &= y_A \cdot \cos \alpha + x_A \cdot \sin \alpha \end{aligned} \quad (2.2)$$



Gambar 7. Faktor Rotasi

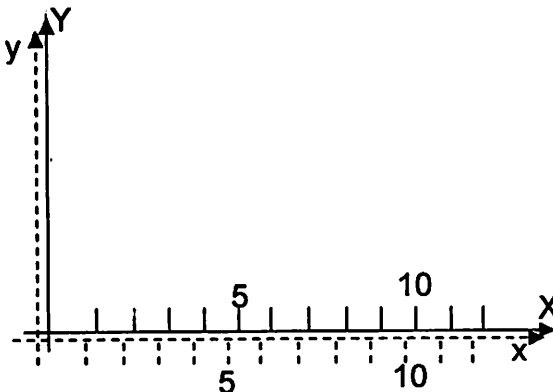
Keterangan Gambar 7 :

- α : besarnya sudut rotasi dari sistem lama ke sistem baru, positif untuk perputaran berlawanan jarum jam
- x, y : Sistem Koordinat Obyek Posisi Lama
- X, Y : Sistem Koordinat Obyek Posisi Baru
- x_A, y_A : Koordinat Titik "A" pada Sistem Koordinat Lama
- X_A, Y_A : Koordinat Titik "A" pada Sistem Koordinat Baru

C. Perbesaran Skala

Suatu sistem koordinat, tidak mungkin terlepas dari besarnya selang/rentang skala yang digunakan, mengingat sistem koordinat harus mengandung faktor skala. Selang skala yang berbeda, akan berakibat pada perbedaan pernyataan koordinat titik, sehingga apabila terdapat 2 (dua) sistem koordinat yang menggunakan selang yang berbeda, maka dalam menyatakan koordinat suatu titik akan berbeda pula.

Gambar (8) dibawah ini, memperlihatkan 2 (dua) salib sumbu yang berimpit, namun menggunakan selang skala yang berbeda.



Gambar 8. Faktor Perbesaran (Scaling factor)

Keterangan Gambar 7:

- x, y : Sistem Koordinat Obyek Posisi Lama
- X, Y : Sistem Koordinat Obyek Posisi Baru

Pernyataan matematis untuk titik A adalah:

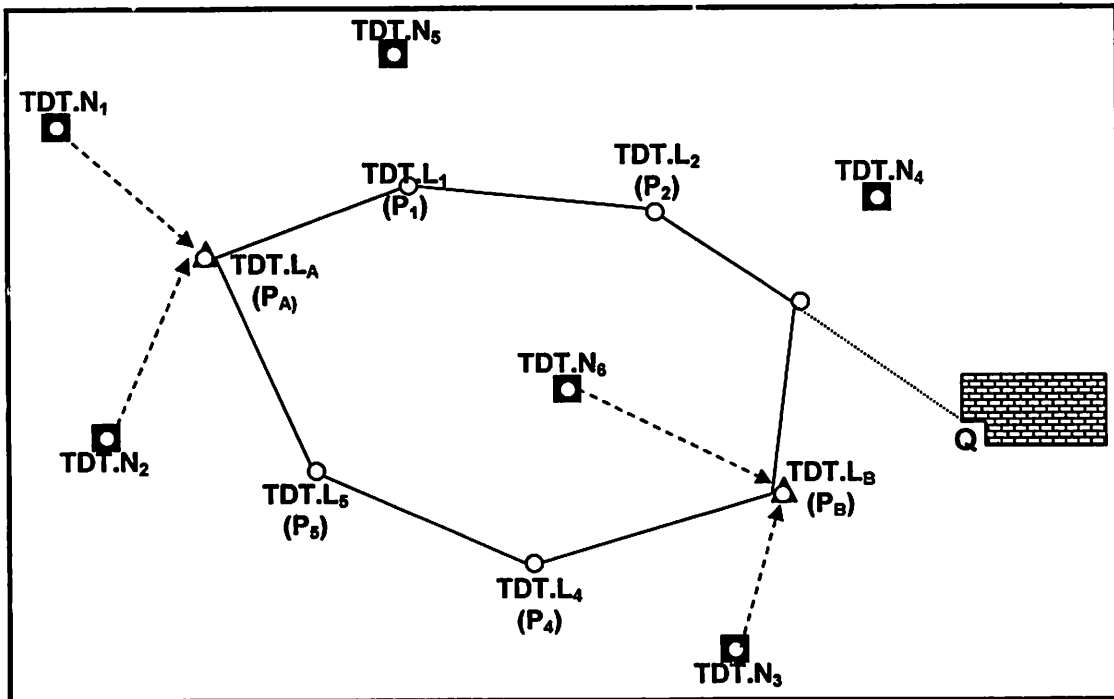
$$X_A = \lambda x_A$$

$$Y_A = \lambda y_A \quad (2.3)$$

Dalam hal ini:

λ = Parameter perbesaran, yaitu perbandingan antara selang skala sistem lama terhadap selang skala sistem baru.

Titik sekutu adalah *titik yang mempunyai koordinat dalam sistem koordinat lokal dan sistem koordinat nasional*. Titik sekutu tersebut, digunakan untuk mengetahui besarnya *parameter-parameter transformasi* (skala, rotasi dan translasi). Setelah parameter-parameter transformasi diketahui nilainya (berdasarkan hitungan), maka koordinat titik lainnya dalam sistem koordinat lokal dapat ditransformasikan ke dalam sistem koordinat nasional. Bila TDT.L telah terikat pada TDT.N, transformasi koordinat dapat langsung dilakukan. Akan tetapi bila belum terikat, terlebih dahulu beberapa TDT.L diikatkan pada TDT.N. Pada gambar dibawah ini diilustrasikan , TDT.L_A dan TDT.L_B diikatkan pada TDT.N terdekat. Dengan demikian, TDT.L_A dan TDT.L_B menjadi titik sekutu dan TDT.L lainnya dapat ditransformasikan ke dalam sistem koordinat nasional:



Gambar 9. Distribusi dan Kerapatan Titik

Keterangan :

- : Titik Dasar Teknik Nasional (TDT.N)
- : Titik yang akan ditransformasikan (TDT.L)
- △ : Titik sekutu (P_A , P_B)
- > : Garis Pengikat (base line)
- Q : Titik Detail

Dalam pemetaan mempertahankan bentuk obyek di kedua sistem, bentuk antar titik-titik yang ditransformasikan tidak mengalami perubahan sifat dikenal dengan istilah *konform*.

II. 5. Transformasi Koordinat Metode Affine

Metode Affine, merupakan metode transformasi dengan memasukkan ketiga unsur transformasi, yaitu, **translasi, rotasi, dan faktor perbesaran.**

Persamaan dasar transformasi Affine metoda dasar adalah sebagai

berikut :

$$\begin{matrix} X = a x + b y + C_1 \\ Y = c x + d y + C_2 \end{matrix} \quad (2.4)$$

Keterangan : x, y : Sistem koordinat lama (lokal)

X, Y : Sistem Koordinat baru (TM-3⁰)

a, b, c, d, C_1, C_2 : Parameter-parameter

transformasi dengan $a \neq d$; $b \neq c$

Persamaan transformasi Affine di atas, berlaku untuk setiap titik. Dapat

dilihat bahwa :

a. Parameter transformasi adalah : a, b, c, d, C_1, C_2 .

b. Parameter tersebut sebagai 6 (enam) variabel dari kedua persamaan, sehingga untuk transformasi Koordinat Metode Affine diperlukan minimal 3 (tiga) titik sekutu.

Bila rumus (2.4) dinyatakan dalam bentuk matrik, untuk jumlah titik

banyak (n buah), maka:

$$\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} x & y & 0 & 0 & 1 & 0 \\ 0 & 0 & x & y & 0 & 1 \end{bmatrix} * \begin{bmatrix} a \\ b \\ c \\ d \\ C_1 \\ C_2 \end{bmatrix} \quad (2.5)$$

Unsur matrik X (matrik parameter) dapat dihitung, dengan

menggunakan rumus :

$$\text{Matrik } X = \begin{bmatrix} a \\ b \\ c \\ d \\ C1 \\ C2 \end{bmatrix} = [A^T A]^{-1} * A^T F \quad (2.6)$$

Keterangan:

[A] : merupakan matrik disain

[F] : adalah matrik konstanta

[X] : matrik parameter

Selanjutnya, koordinat titik obyek yang akan ditransformasi dapat dihitung dengan formula yang sama setelah nilai parameter dihitung, dengan menggunakan rumus:

$$F2 = A2 * [X] \quad (2.7)$$

Keterangan **A2** = Matrik koefisien untuk titik obyek (seperti juga matrik A)

F2 = Matrik koordinat titik obyek hasil transformasi

[X] = Matrik parameter

Bila ingin diketahui nilai atau harga faktor skala (λ) dan harga putaran salib sumbu (ω), dapat digunakan rumus dibawah ini:

$$\begin{array}{l} \lambda = (a^2 + b^2)^{0,5} \\ \omega = \text{ArcTan}\left(\frac{-b}{a}\right) \end{array} \quad (2.8)$$

II. 6. Transformasi Koordinat Metode Lauf

Metode Lauf ditujukan untuk transformasi koordinat titik-titik obyek, dengan titik sekutu yang berjauhan jaraknya (dapat mencapai ± 80 Km). Transmoramsi untuk titik yang begjarak jauh, harus diperhitungkan factor

kelengkungan bumi yang akan berpengaruh besar pada hasil transformasi tersebut.

Pada dasarnya, metode ini memberikan koreksi pada jarak dalam bentuk penambahan suku persamaan sebagai suku koreksi, semakin banyak suku persamaan diperhitungkan, maka semakin jauh jarak antar titik sekutu.

Pada mulanya, transformasi Lauf, merupakan transformasi berderajat banyak (misal n), persamaan berikut ini untuk transformasi berderajat dua.

$$X = ax - by + c(x^2 - y^2) - 2dxy + C_1 \quad (2.9)$$

$$Y = bx - ay + d(x^2 - y^2) + 2cxy + C_2$$

Keterangan:

a, b, c, d, C1, C2 : adalah parameter transformasi

X, Y : Sistem Koordinat baru (nasional)

x, y : sistem koordinat lama (local)

Mode penyelesaian diatas merupakan cara mendasar atau disebut juga **Model Lauf-1**.

Persamaan di atas berlaku untuk semua jenis sekutu (titik sekutu maupun untuk titik obyek).

Bila persamaan (2.9) dinyatakan dalam bentuk matrik untuk setiap titik, maka dapat ditulis:

$$\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} x & -y & (x^2 - y^2) & -2xy & 1 & 0 \\ y & x & 2xy & (x^2 - y^2) & 0 & 1 \end{bmatrix} * \begin{bmatrix} a \\ b \\ c \\ d \\ C1 \\ C2 \end{bmatrix} \quad (2.10)$$

dengan "n" adalah jumlah titik sekutu atau titik obyek.

Persamaan diatas dapat pula ditulis sebagai:

$$A * [X] - F = 0 \quad (2.11)$$

Sehingga diperoleh

$$[X] = [A^T x A]^{-1} x A^T x F \quad (2.12)$$

Keterangan:

A : merupakan matrik disain

F : adalah matrik konstanta

[X] : matrik parameter

II. 6. 1. Meode Lauf Selisih Terbagi (Bilangan Kompleks).

Metode ini merupakan penerapan matematik dengan bilangan kompleks. Persamaan (2.9), dinyatakan dalam bilangan kompleks sebagai:

$$Z = Az^2 + Bz + C \quad (2.13)$$

Dalam hal ini:

Z = X + iY = Koordinat hasil Transformasi

z = x + iy = Koordinat lama (local)

A = e + id = Koefisien variable derajat 2 (kuadrat)

B = a + ib = Koefisien variable derajat 1 (linier)

C = c₁ + ic₂ = Konstanta

Pemecahan persamaan diatas dilakukan melalui interpolasi selisih terbagi (*divided difference*). Metode ini baik digunakan untuk **titik sekutu minimal 4 (empat) titik.**

Diagram Hitungan Selisih Terbagi

Titik Sekutu	Z	Z	Selisih Terbagi		
			Ke-1	Ke-2	Ke-3
A	$X_A + iY_A$	$x_A + iy_A$	$\Delta^{(1)}Z_1$ $\Delta^{(1)}Z_2$ $\Delta^{(1)}Z_3$	$\Delta^{(2)}Z_1$ $\Delta^{(2)}Z_2$	$\Delta^{(3)}Z_1$
B	$X_B + iY_B$	$x_B + iy_B$			
C	$X_C + iY_C$	$x_C + iy_C$			
D	$X_D + iY_D$	$x_D + iy_D$			

Dengan :

$$\begin{aligned}
 \Delta^{(1)}Z_1 &= \frac{Z_B - Z_A}{z_B - z_A} & ; & \Delta^{(2)}Z_1 = \frac{\Delta^{(1)}Z_2 - \Delta^{(1)}Z_1}{z_C - z_A} & ; & \Delta^{(3)}Z_1 = \frac{\Delta^{(2)}Z_2 - \Delta^{(2)}Z_1}{z_D - z_A} \\
 \Delta^{(1)}Z_2 &= \frac{z_C - z_B}{z_C - z_B} & ; & \Delta^{(2)}Z_2 = \frac{\Delta^{(1)}Z_3 - \Delta^{(1)}Z_2}{z_C - z_A} \\
 \Delta^{(1)}Z_3 &= \frac{z_D - z_C}{z_D - z_C} & \dots\dots\dots & (2.14)
 \end{aligned}$$

Selanjutnya titik obyek ditransformasikan dengan interpolasi dari harga titik sekutu pertama dan terakhir. Nilai titik obyek (titik i) dinyatakan terhadap/ dari nilai sekutu pertama atau titik sekutu akhir, seperti pada persamaan dibawah ini:

Bila dari titik A (titik awal)

$$\begin{aligned}
 Z_i &= Z_A + (Z_i - Z_A) \Delta^{(1)}Z_1 + (Z_i - Z_A)(Z_i - Z_B) \Delta^{(2)}Z_1 + \\
 &\quad (Z_i - Z_A)(Z_i - Z_B)(Z_i - Z_C) \Delta^{(3)}Z_1 + \dots\dots\dots & (2.15a)
 \end{aligned}$$

Bila dari titik D (titik akhir)

$$\begin{aligned}
 Z_i &= Z_D + (Z_i - Z_D) \Delta^{(1)}Z_3 + (Z_i - Z_D)(Z_i - Z_C) \Delta^{(2)}Z_2 + \\
 &\quad (Z_i - Z_D)(Z_i - Z_C)(Z_i - Z_B) \Delta^{(3)}Z_1 + \dots\dots\dots & (2.15b)
 \end{aligned}$$

Dalam melaksanakan hitungan dengan bilangan kompleks ini, ada beberapa operasi hitungan yang harus diperhatikan sebagai berikut:

Operasi Hitungan	Pelaksanaan Hitungan
Penjumlahan	$Z_1 + Z_2 = (X_1 + X_2) + i(Y_1 + Y_2)$
Pengurangan	$Z_1 - X_2 = (X_1 - X_2) + i(Y_1 - Y_2)$
Perkalian	$Z_1 * Z_2 = (X_1X_2 - Y_1Y_2) + i(X_2Y_1 + X_1Y_2)$
Pembagian	$Z_1/Z_2 = \{(X_1X_2 + Y_1Y_2) + i(X_2Y_1 - X_1Y_2)\}/(X_2^2 + Y_2^2)$

6. 2. Koreksi Harga Pendekatan Metode Numeris

Pada metode koreksi ini (metode numerik), harga pendekatan setiap titik koordinat baru (Koordinat lokal), diperoleh dengan hitungan metode Helmert (*transformasi, dilakukan dengan menggunakan syarat cukup, yaitu dengan 2 (dua) titik sekutu yang dipilih*).

Bentuk persamaan Transformasi metode Helmert adalah:

$$ax - by + C_1 - X = 0$$

$$bx + ay + C_2 - Y = 0$$

Bila persamaan tersebut diterapkan pada suatu titik sekutu (misal titik A dan titik B) dan dinyatakan dalam bentuk matrik, maka dapat ditulis:

$$\begin{bmatrix} x_A & -y_A & 1 & 0 \\ x_B & -y_B & 1 & 0 \\ y_A & x_A & 0 & 1 \\ y_B & x_B & 0 & 1 \end{bmatrix} * \begin{bmatrix} a \\ b \\ C1 \\ C2 \end{bmatrix} - \begin{bmatrix} X_A \\ X_B \\ Y_A \\ Y_B \end{bmatrix} = 0 \quad (2.16)$$

Matrik A Matrik X Matrik F

$$\text{Dengan : } A * [X] - F = 0 \quad (2.17)$$

$$[X] = [A^T * A]^{-1} * A^T * F \quad (2.18)$$

Keterangan:

A = Matrik Desain

F = Matrik Konstanta

[X] = Matrik Parameter

Langkah-langkah pelaksanaan koreksi hasil transformasi adalah sebagai berikut:

1. Hitung koreksi titik sekutu terakhir dari harga pendekatan dengan menggunakan persamaan:

$$\Delta Z_n = Z_n - Z_n^0$$

Atau (2.19)

$$Z_i = Z_i^0 + \Delta Z_i$$

Dengan:

ΔZ_n = Koreksi Koordinat pendekatan titik ke-n (titik sekutu terakhir)

ΔZ_i = Koreksi koordinat pendekatan titik i

Z_i = Koordinat titik-i

Z_n = Koordinat titik ke-n

2. Hitung koreksi setiap titik obyek dengan persamaan:

$$\Delta Z_i = \frac{(Z_A^0 - Z_i^0) - (Z_B^0 - Z_i^0)}{(Z_A^0 - Z_C^0) - (Z_B^0 - Z_C^0)} \Delta Z_C \quad (2.20)$$

3. Hitung jarak vektor (modulus) " $|\Delta Z_n|$ " dengan menggunakan persamaan:

$$|\Delta Z_n| = \sqrt{\Delta X_n^2 + \Delta Y_n^2} \quad (2.21)$$

4. Hitung sudut jurusan (argumen) " $\{\Delta Z_i\}$ " dengan menggunakan persamaan:

$$\{\Delta Z_n\} = \text{Arc Tan}(\Delta X_n / \Delta Y_n) \quad (2.22)$$

5. Penghitungan jarak vektor (D_o) dan sudut jurusan (α_o) antara koordinat titik sekutu dengan koordinat titik yang akan ditransformasikan dengan menggunakan persamaan :

$$\begin{aligned} \alpha_{oi} &= \text{ArcTan} \frac{\Delta X_i}{\Delta Y_i} \\ D_{oi} &= \Delta X_i * \text{Sin} \alpha_{oi} \end{aligned} \quad (2.23)$$

6. Hitung Modulus ($|\Delta Z_i|$) dan argumen ($\{\Delta Z_i\}$) ke setiap titik obyek dengan menggunakan persamaan:

$$|\Delta Z_i| = \frac{D^0_{Ai} * D^0_{Bi}}{D^0_{AC} * D^0_{BC}} |\Delta Z_C|$$

$$\{\Delta Z_i\} = \alpha^0_{iA} + \alpha^0_{iB} - \alpha^0_{CB} + \{\Delta Z_C\}$$
(2.24)

7. Hitung koreksi titik obyek dengan persamaan:

$$\Delta X_i = |\Delta X_i| \sin\{\Delta Z_i\} \quad ; \quad \Delta Y_i = |\Delta Y_i| \cos\{\Delta Z_i\}$$
(2.25)

II. 7. Bahasa Pemrograman Visual Basic . 6.0

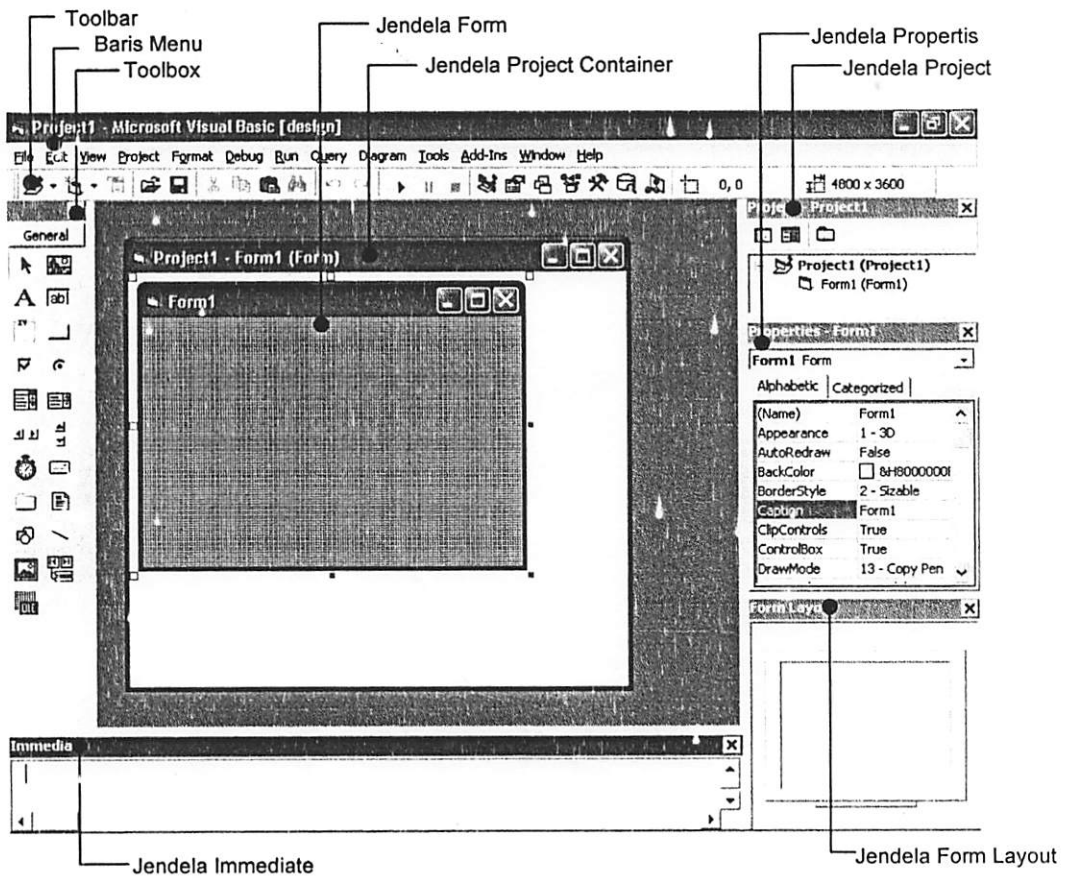
Definisi sebuah program adalah daftar instruksi secara rinci. Sebuah program hanyalah sederetan instruksi yang memberitahukan komputer tentang tindakan yang harus dilakukannya.

Visual Basic 6.0 merupakan salah satu bahasa pemrograman tercepat dan termudah untuk membuat suatu aplikasi dalam Microsoft Windows, dengan menggunakan metode GUI (*Graphical User Interface*). Visual Basic 6.0 memudahkan pemrogram untuk berinteraksi langsung dengan elemen-elemen untuk setiap bentuk pemrograman [Pamungkas, 2000].

Secara mendasar Visual Basic 6.0 mirip dengan bahasa pemrograman yang lain misalnya pascal (tetapi tentu saja sintaks dari tiap-tiap bahasa tidak sama persis). Visual Basic tidak memerlukan programan khusus untuk menampilkan jendela (*windows*), dan cara penggunaannya juga berbasis visual seperti aplikasi windows lainnya [Wiryanto, 2003].

Visual Basic 6.0 menyediakan prasarana yang dapat dipergunakan secara cepat untuk menciptakan aplikasi komputer dengan antar muka yang berbasis visual dilingkungan Windows dan telah menyediakan obyek-obyek

bantu pemrograman yang paling efisien dan relatif lebih mudah untuk mengembangkan aplikasi perangkat lunak (*Software*) yang berbasis pada sistem operasi windows.



Gambar 10. Tampilan Visual Basic

2. 7. 1. Elemen Program Dalam Visual Basic 6.0

Visual Basic 6.0 memiliki beberapa elemen utama antara lain sebagai berikut :

1. Menu Baris (Menu Bar)


Menampilkan menu perintah untuk pengembangan aplikasi. Selain perintah standar windows seperti *File*, *View*, *Window*, dan *Help*, terdapat juga menu-menu khusus dalam pemrograman seperti halnya

Project, Format Debug, atau Run. Seperti terlihat pada gambar 11 dibawah ini:

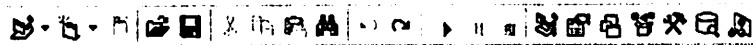
File Edit View Project Format Debug Run Query Diagram Tools Add-Ins Window Help

Gambar 11. Sistem Menu dalam Visual Basic

2. Toolbar

Toolbar adalah tombol-tombol yang mewakili suatu perintah tertentu dari *Visual Basic*. Setiap tombol tersebut dapat langsung diklik untuk melakukan perintah tertentu. Biasanya tombol-tombol ini merupakan perintah-perintah yang sering digunakan dan terdapat pula menu *Visual Basic*. Sebagai contoh, dari pada memilih perintah **Open**, dapat langsung mengklik tombol  (*open*).






Secara default (yang ditetapkan oleh pembuat program) toolbar standart akan ditampilkan, toolbar yang lain adalah **Edit** untuk editing, **Form Editor** untuk desain form. **Debug** untuk melacak kesalahan.



Gambar 12. Toolbar Standart Visual Basic

Fungsi dari masing-masing *tiolbar standart yang ada di Visual Basic*

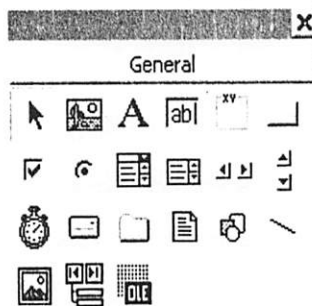
antara lain :

TOOLBAR	NAMA	FUNGSI
	Add Project	Menambah proyek ke dalam proyek yang sudah ada
	Add Item	Menambahkan komponen atau obyek ke dalam jendela Form
	Menu Editor	Menampilkan Menu Editor untuk mengubah tampilan menu
	Open Poject	Membuka Proyek Visual Basic yang sudah ada
	Save Proyek Group	Menyimpan proyek Visual Basic pada Komputer

	Cut	Memotong Elemen yang dipilih di layar
	Copy	Mengcopy elemen yang dipilih di layar
	Paste	Menyalin Elemen sebelumnya yang sudah dipotong
	Find	Mencari teks Tertentu
	Undo	Membatalkan perintah atau tindakan yang terakhir
	Redo	Mengulangi perintah atau tindakan terakhir yang dibatalkan
	Start	Menjalankan proyek yang dibuat di Visual Basic
	Break	Menghentikan <i>Running</i> program untuk sementara
	End	Menghentikan <i>Running</i>
	Project Explorer	Menampilkan jendela project explorer
	Propertis Windows	Menampilkan jendela propertis
	Form Layout Windows	Menampilkan jendela form layout windows
	Object Browser	Menampilkan jendela object browser
	Tollbox	Menampilkan jendela tollbox
	Data View Windows	Menampilkan data view windows
	Visual Component Manager	Menampilkan jendela visual component manager

3. Toolbox

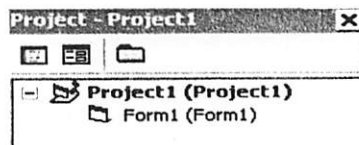
Toolbox adalah sebuah piranti yang berfungsi sebagai alat untuk menempatkan berbagai macam tombol kontrol di atas jendela form.



Gambar 13. Toolbox

4. Project Explorer

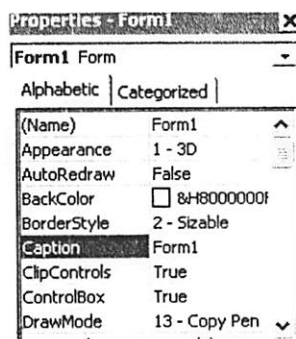
Jendela **Project Explorer** adalah tempat untuk memonitor jumlah form, module, class yang digunakan dalam suatu atau beberapa project. Didalam jendela tersebut ditampilkan nama *Project*, nama-nama objek form dan nama-nama module (hanya menyimpan kode program saja) yang digunakan. Di belakang setiap nama tersebut di dalam tanda kurung adalah nama file penyimpanannya.



Gambar14. Project Explorer

5. Jendela Properties Windows

Jendela Properties Windows adalah tempat untuk mengedit karakteristik dari sebuah objek yang terdapat di dalam project, misalnya ukuran, warna, dan text dari suatu tombol.

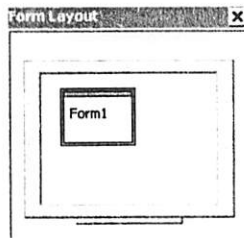


Gambar 15. Jendela Propertis Windows

6. Form Layout Windows

Form Layout Windows adalah jendela yang dipakai untuk mengatur posisi tampilan interface relative terhadap layar monitor komputer. Untuk mengubah posisinya, digunakan cara drag-drop objek yang

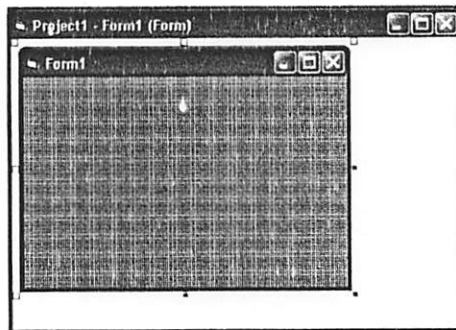
terlihat di jendela, yaitu klik form tersebut dan dengan tetap mengklik, geser form dalam bidang layer.



Gambar 16. Form Layout

7. Jendela Form

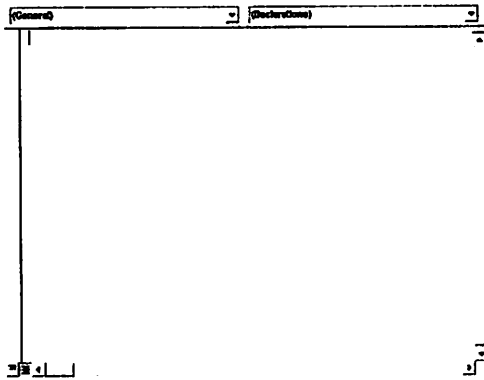
Jendela Form adalah tempat untuk merancang interface dari suatu aplikasi. Pada Form ini akan dimasukkan berbagai macam object interaktif seperti teks, bergambar tombol-tombol perintah *scrollbar* dan sebagainya sesuai dengan kebutuhan.



Gambar 17. Jendela Form

8. Jendela Code

Jendela Code tempat untuk mengetik kode-kode program yang merupakan instruksi-instruksi untuk aplikasi Visual Basic, dan berfungsi sebagai program pengolah kata untuk menuliskan baris-baris kode pemrograman pada aplikasi yang sedang dibuat. Suatu jendela **Code Editor** yang terpisah akan dihasilkan untuk setiap form atau module.



Gambar 18. Jendela Code

2. 7. 2. Struktur Program dalam Visual Basic 6.0

Dalam program Visual Basic 6.0 memiliki beberapa struktur kontrol program antara lain sebagai berikut :

1. Decision

Struktur decision dipakai untuk menguji suatu kondisi, dan kode program baru akan dieksekusi tergantung dari hasil tes. Ada tiga macam struktur decision, antara lain :

a. If....Then

Pada decision kondisi yang diuji biasanya berupa perbandingan dari sembarang ekspresi untuk mengevaluasi nilai suatu bilangan. Ada dua macam penulisannya :

If condition Then statement

Atau

If condition Then

Statement

End If

b. If...Then....Else

Select Case adalah bentuk variasi dan control struktur

If...Then...Else. Format penulisannya sebagai berikut :

```
Select Case testexpression  
    [Case expression-1  
        [statementblok-1]  
    [Case expression-2  
        [statementblok-2]  
    .....  
    [Case Else  
        [statementblok-n]  
  
End Select
```

Contoh :

```
If OptMode (0) . Value Then  
    strMetode = "INTEGRASI"  
    Hitung 1  
Else  
    strMetode = "URAIAN DERET"  
    Hitung 2  
End If
```

c. Select Case

Select Case adalah bentuk variasi dari kontrol struktur

If...Then....Else.

Format penulisannya sebagai berikut :

```
Select Case testexpression  
    [Case expression-1
```

[statementblok-1]

[Case expression-2

[statementblok-2]

.....

[Case Else

[statementblok-n]

End Select

Contoh :

```
Private Sub lblMenu_Click (Index As Integer)
    Select Case Index
    Case 0
        InputFrm.Show vbModal, MainForm
        InputFrm.Zorder 0
    Case 1
        ProcessFrm.Show vbModal, MainForm
        ProcessFrm.Zorder 0
    Case 2
        ReportFrm.Show cbModeless, MainForm
        ReportFrm.Zorder 0
    Case 3
        PetunjukFrm.OLE1.DoVerb
    Case 4
        End
    End Select
End Sub
```

2. Loop

Dengan loop struktur memungkinkan kita untuk mengeksekusi beberapa baris kode secara berulang-ulang. Loop struktur dalam Visual Basic 6.0, meliputi :

a. DO.....Loop

Digunakan untuk mengulang suatu perintah sampai suatu kondisi terpenuhi, kondisi dapat ditetapkan pada akhir atau awal perintah.

Format penulisannya sebagai berikut :

Do While *condition*

Statement

Loop

Atau

Do

Statement

Loop While *condition*

Contoh :

```
Do
    i = i + 1
Loop While I < 5
```

Sedangkan pada variasi lainnya seperti :

Do Until *condition*

Statement

Loop

Atau

Do

Statement

Loop Until condition

Pada variasi ini loop akan dieksekusi selama *condition* terus bernilai salah.

Contoh :

```
Do Until i = 5
    i = i + 1
Loop
```

b. For...Next

Dugunakan untuk mengulang suatu perintah sebanyak yang ditetapkan. Format penulisannya :

```
For counter = start To end [Step increment]
    [statements]
Next [counter]
```

Contoh :

```
For I = 1 To 5
    Print I
Next I
```

c. For Each.....Next

Berguna untuk looping berdasarkan banyaknya element dari suatu koleksi objek atau array. Format penulisannya adalah :

```
For Each element in group
    Statements
Next element
```

Contoh :

```
Private Sub Form_Click ( )
```

```

For Each Control In Form1. Controls
    List1. Additem Control.Name
Next Control

End Sub

```

3. Array

Array merupakan suatu variabel yang mempunyai banyak ruang yang diisi data. Untuk menandai data di ruang-ruang tersebut digunakan indeks (angka). Cara kerjanya sama seperti konsep matrik, mempunyai batas atas dan batas bawah (upper-lower bound), setiap elemen (ruang) mempunyai indeks yang berurutan diantara batas-batas tersebut dan untuk mengaksesnya digunakan perintah looping. Dalam Visual Basic ada dua macam array yaitu :

a. Array Statis

Pembuatannya tergantung pada rencana cakupan pemakaiannya. Ukurannya selalu sama. Array statis terdiri dari :

- *Array Public*, agar variabel dapat diakses dari setiap prosedur dari semua form atau modul yang ada (form/module biasa lebih dari satu), pakai perintah **Public** dibagian **Declaration** dari suatu **Module**.

Contoh :

```

Public Counters (14) As Integer

Public Sums (20) As Double

Public Sums (20) otomatis pakai type Variant

```

- Jika Array hanya perlu diakses dari **Module** itu sendiri, maka ganti perintah di atas dengan **Private**, **Static** atau **Dim** di bagian **Declaration** dari suatu module.

Contoh :

```
Dim Counters (14) As Integer
Dim Sums (20) as Double
Static MatrixA (9, 9) as Double
Static MatrixA (1 To 10, 1 to 10) As Double
Dim MultiD (3, 1 To 10, 1 To 10)
```

b. Array Dinamis

Array Dinamis ukurannya (jumlah elemen) dapat diubah ketika dijalankan, sangat berguna misalnya ketika program sedang berjalan mendapat input data yang memerlukan penyuaian ukuran variabel.

Contoh :

```
Dim Mtrx ( )
Private Sub Commad1_Click ( )
    l = InputBox ("Jumlah baris")
    k = InputBox ("Jumlah kolom")
    ReDim Mtrx (l, k)
    For i = 1 To l
        For j = 1 To k
            Mtrx (I, j) = Str (i) + Str (j)
        Next j
    Print
    Next i
End Sub
```

BAB III

PELAKSANAAN PENELITIAN

III. 1. Peralatan Penelitian

Adapun peralatan yang digunakan dalam penelitian , meliputi:

1. Perangkat Keras

- a. Central Processing Unit (CPU)
- b. Processor AMD Athlon 1, 73 GHz
- c. Memory 256 MB
- d. Hard Disk 80 GB
- e. Monitor
- f. Keyboard dan Mouse
- g. Disk Drive 1, 44 MB

2. Perangkat Lunak

- a. Microsoft Word
- b. Microsoft Excel
- c. Program Visual Basic V.6.0

III. 2. Materi Penelitian

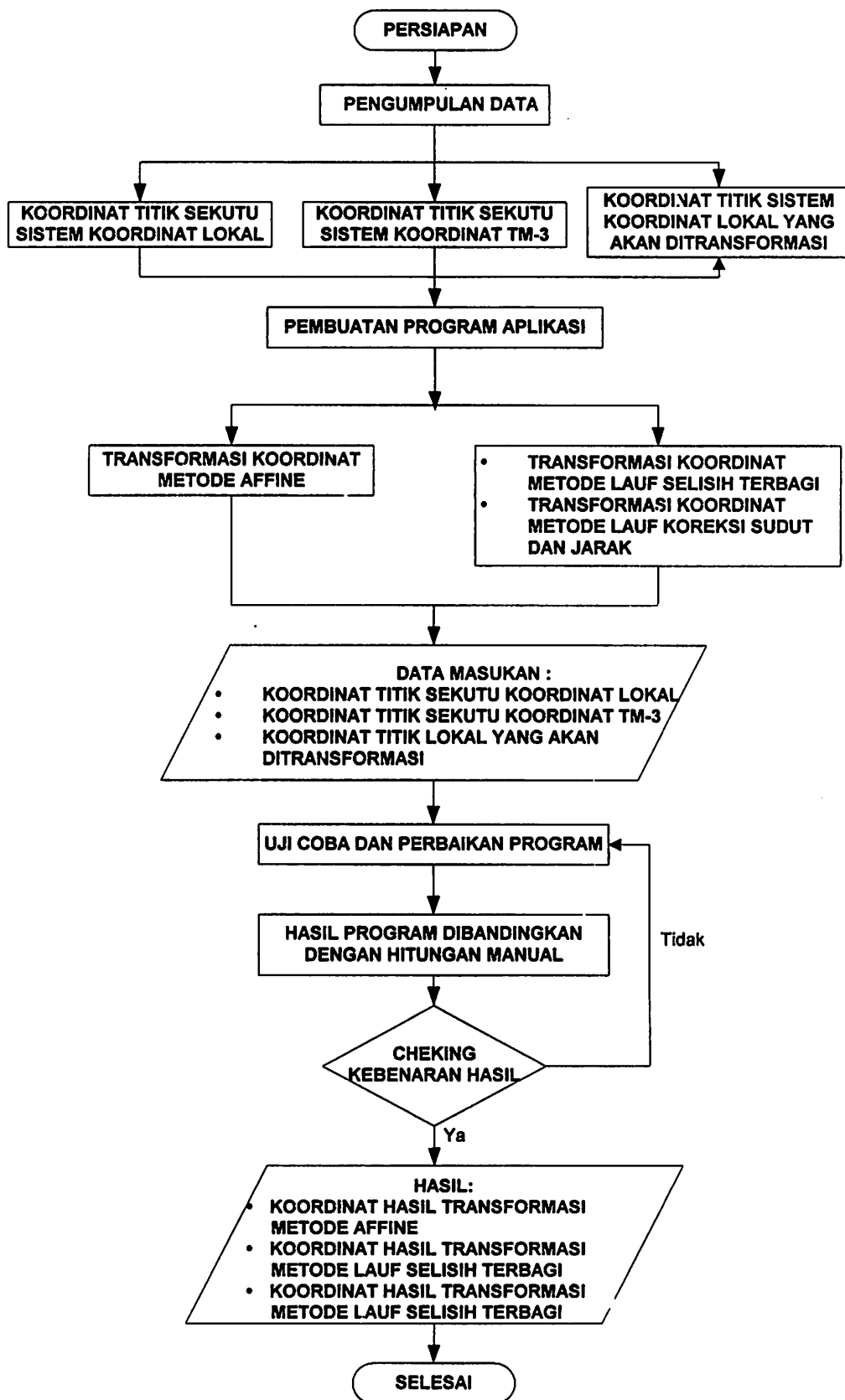
Materi penelitian terdiri dari beberapa tahap, yaitu:

1. Pengumpulan data sampel. Data yang digunakan untuk menguji apakah program hitung Transformasi Koordinat ini *running*, serta memenuhi ketelitian adalah beberapa data Koordinat titik-titik GPS Orde-3 daerah Kodya Surabaya, Sistem Koordinat Lokal dan Sistem Koordinat TM-3⁰.

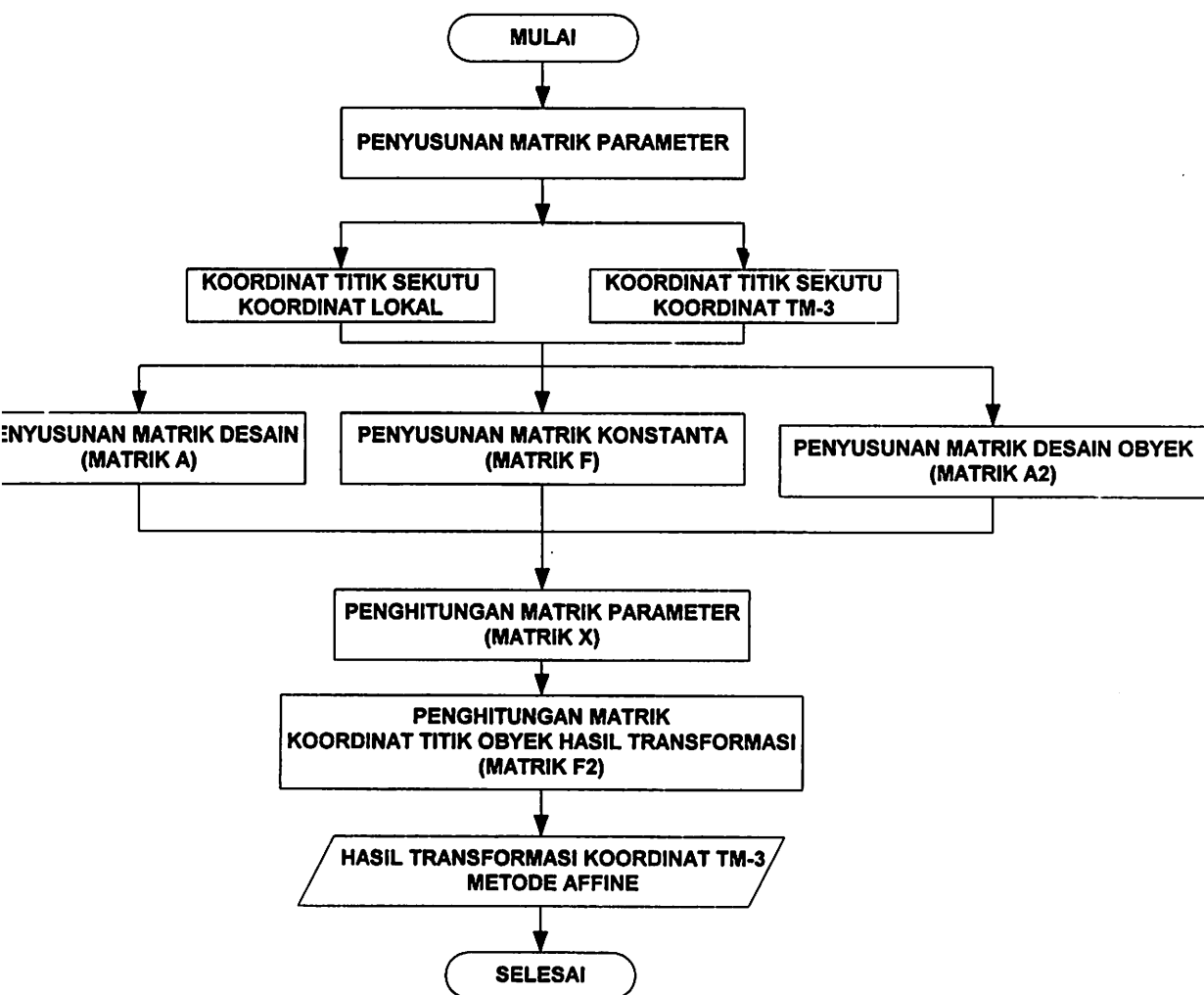
2. Penghitungan Transformasi Koordinat dengan Metode Lauf (selisih terbagi), Metode Lauf Koreksi Sudut dan Jarak, serta Metode Affine dan Pembuatan Program Aplikasi untuk Transformasi Koordinat Lokal menjadi Sistem Koordinat Nasional TM-3⁰, dengan menggunakan bahasa pemrograman Visual Basic V.6.0.
3. Untuk menguji ketelitian hasil dari Program Hitung Transformasi Koordinat ini, maka Koordinat hasil Transformasi akan dibandingkan dengan hasil perhitungan menggunakan Software Microsoft Excel.

Langkah-langkah dalam penelitian dapat digambarkan pada diagram alir dibawah ini:

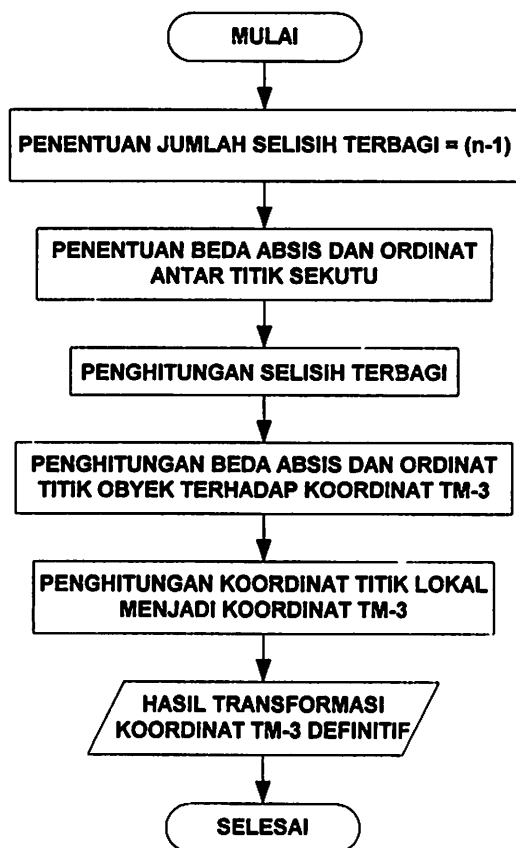
1. Diagram Alir Penelitian



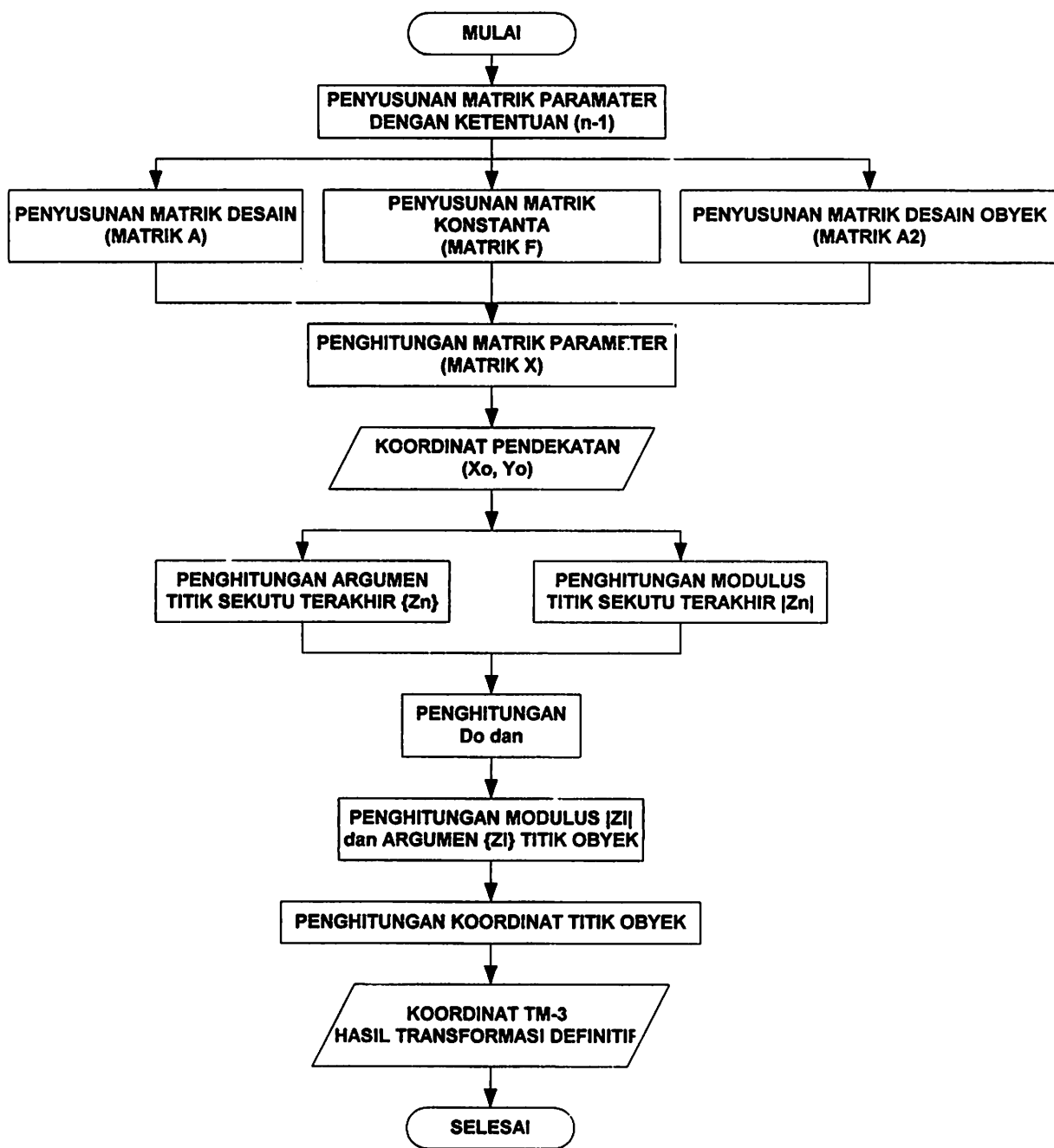
2. Diagramm Alir Transformasi Koordinat Metode Affine



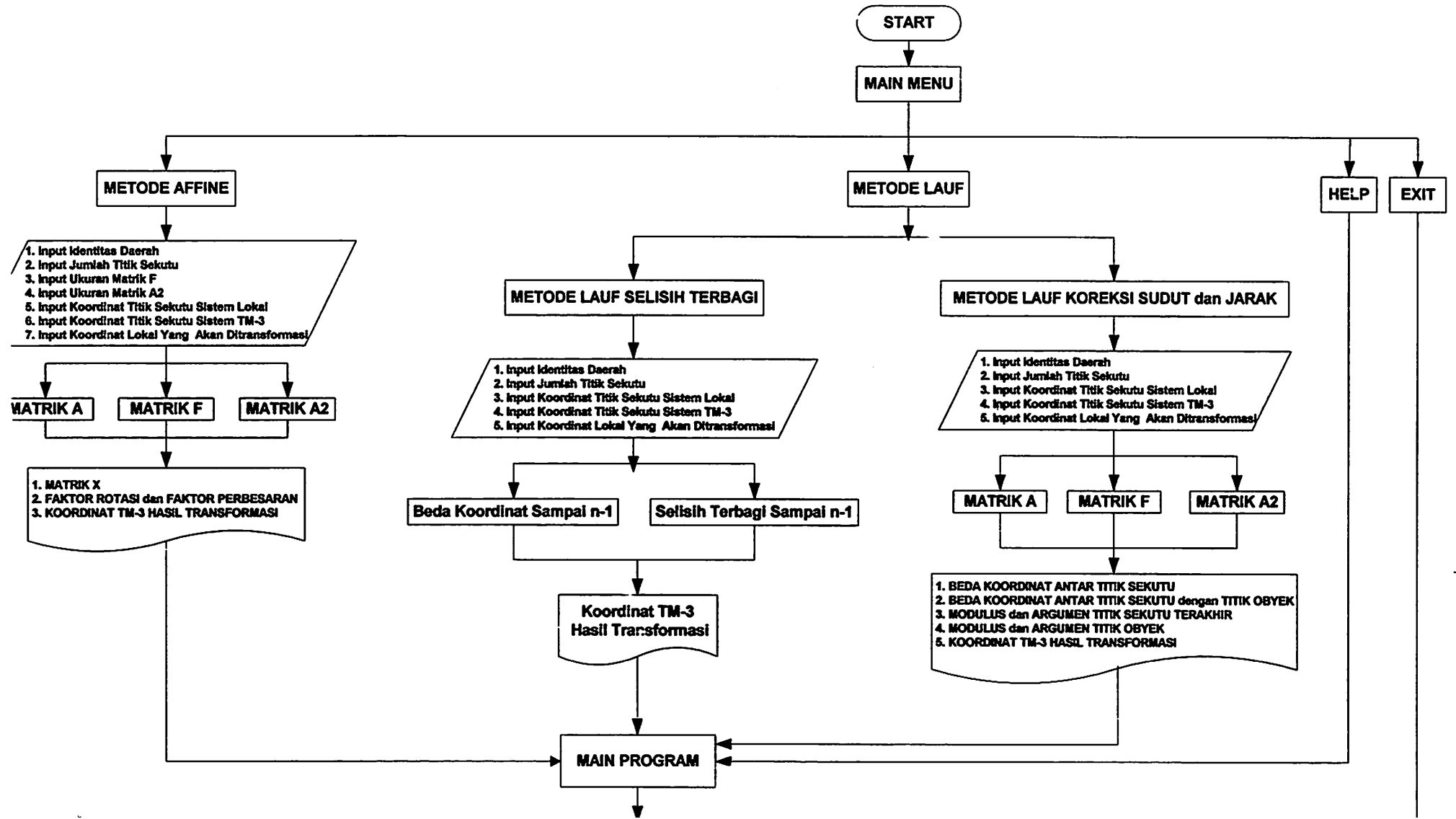
2. 3. Diagram Alir Transformasi Koordinat Metode Lauf Selisih Terbagi



2. 4. Diagram Alir Metode Lauf Koreksi Sudut dan Jarak



II. 2 . 5. Diagram Alir Paket Program



III. 3. 1. Penjelasan Diagram Alir Penelitian

1. *Persiapan*, dalam tahap persiapan penelitian ini dilakukan persiapan-persiapan sebelum penelitian, yaitu persiapan materi teori dengan pembacaan literature yang berhubungan dengan topik penelitian.
2. *Pengumpulan Data*, tahap ini dilakukan untuk pengumpul data yang diperlukan dalam penelitian yang selanjutnya akan diproses dalam penelitian ini. Data-data yang diperlukan untuk penelitian ini adalah :
 - a. Data koordinat titik sekutu sistem koordinat lokal
 - b. Data koordinat titik sekutu sistem koordinat TM-3⁰
 - c. Data koordinat sistem koordinat lokal yang akan ditransformasikan.
3. Pembuatan program aplikasi perhitungan transformasi koordinat, yang didalamnya terdiri dari Program Hitung Transformasi Koordinat dengan Metode Affine dan Program Hitung Transformasi Koordinat dengan Metode Lauf.
4. Data masukan yang akan dipakai dalam Program Hitung Transformasi Koordinat adalah :
 - a. Koordinat Titik Sekutu Sistem Koordinat Lokal
 - b. Koordinat Titik Sekutu Sistem Koordinat TM-3⁰
 - c. Koordinat Titik Sistem Koordinat Lokal yang akan ditransformasikan.
5. Uji coba dan perbaikan program, dalam hal ini untuk mengetahui apakah Program Hitung Transformasi Koordinat tersebut *Running* atau tidak maka perlu dilakukan ujicoba dengan menggunakan data-data yang sudah valid serta perbaikan apabila diperlukan.

6. Hasil Program dibandingkan dengan hitungan manual. Dalam hal ini apabila program hitung transformasi koordinat tersebut telah *running* maka perlukan dilakukan uji hasil program dengan membandingkan hasil hitung Program dengan hitungan manual dalam hal ini menggunakan *Microsoft Excel*.
7. Cheking kebenaran hasil merupakan pengecekan uji coba hasil hitungan dengan menggunakan program hitung transformasi apabila hasil hitungan telah sesuai dengan hasil hitungan manual, maka program tersebut sudah benar, tapi apabila selisih antara hasil hitungan dengan menggunakan program hitung transformasi koordinat dengan hitungan manual berbeda jauh maka perlu dilakukan perbaikan kembali pada program hitung transformasi koordinat tersebut.
8. Hasil dari program hitung transformasi koordinat tersebut adalah:
 - a. Koordinat TM-3⁰ hasil transformasi dengan metode Affine
 - b. Koordinat TM-3⁰ hasil transformasi dengan metode Lauf Selisih Terbagi
 - c. Koordinat TM-3⁰ hasil transformasi dengan metode Lauf Koreksi Sudut dan Jarak.

A. Penjelasan Diagram Alir Transformasi Koordinat Affine

Dalam proses penghitungan transformasi koordinat dengan menggunakan metode Affine yang diterapkan akan diperoleh enam parameter transformasi. Adapun langkah-langkah dalam penghitungan transformasi koordinat dengan metode Affine, yaitu:

1. *Penyusunan Matrik Parameter*, untuk menyusun matrik parameter diperlukan Koordinat Titik Sekutu Sistem Koordinat Lokal dan Titik Sekutu Sistem Koordinat TM-3⁰.
2. *Penyusunan Matrik Desain (Matrik A)*, komposisi matrik A adalah matrik desain yang berisi titik sekutu koordinat lama (lokal), ordo matrik A adalah 6 (enam) kolom dan 2n baris (n jumlah titik sekutu). Bentuk matrik A adalah seperti pada persamaan (2.5)
3. *Penyusunan Matrik Konstanta (Matrik F)*, *Matrik F* merupakan matrik konstanta yang berisikan titik sekutu koordinat baru (TM-3⁰), ordo matriknya " ${}_{2n}F_1$ " dengan "n" jumlah titik sekutu. Bentuk matrik F adalah seperti pada persamaan (2.5)
4. *Penyusunan Matrik Koefisien Untuk titik Obyek (Matrik A2)*, komposisi *Matrik A2* adalah matrik desain yang berisi koefisien untuk titik obyek yang akan ditransformasikan atau koordinat lama (lokal), ordo *Matrik A2* adalah 6 (enam) kolom dan 2n baris (n jumlah titik yang ditransformasikan). Bentuk *Matrik A2* adalah seperti *Matrik A*.
5. *Penghitungan Matrik Parameter (Matrik X)*, *Matrik X* diperoleh dari proses matematik pada persamaan (2.6), hasilnya adalah 6 (enam) parameter transformasi. Dimensi *Matrik X* adalah 6 (enam) baris dan 1 (satu) kolom.
6. *Penghitungan Matrik Koordinat Titik Obyek yang Ditransformasikan (Matrik F2)*. Hasil *Matrik F2* diperoleh dari proses matematik dengan menggunakan persamaan (2.4). *Matrik F2* berisikan koordinat hasil transformasi. Ordo *Matrik F2* adalah " ${}_{2n}F2_1$ ", "n" merupakan banyaknya titik obyek yang akan ditransformasikan.

7. Hasil dari perhitungan matrik F2 adalah Koordinat Titik dalam Sistem Koordinat TM-3⁰.

B. Penjelasan Diagram Alir Transformasi Koordinat Metode Lauf Selisih Terbagi.

Dalam Transformasi Koordinat Metode Laut Selisih Terbagi ada beberapa langkah yang dilakukan:

1. Penentuan Jumlah selisih terbagi hingga didapatkan satu 1 (satu) harga. Bila titik sekutu diketahui sebanyak "n", maka selisih terbagi bertingkat hingga (n-1), bentuk penyusunan selisih terbagi seperti pada persamaan (2.14).
2. Penentuan beda Absis (Δx) dan Ordinat (Δy) titik yang akan ditransformasikan (Koordinat Lama atau Lokal) terhadap salah satu titik sekutu (X, Y) awal atau akhir.
3. *Penghitungan selisih terbagi.* Penghitungan selisih terbagi dilakukan mulai dari selisih terbagi antar titik sekutu ($\Delta^{(1)}Z_{(n)}$) sampai dengan ($\Delta^{(n-1)}Z_{(1)}$), sehingga diperoleh beda Koordinat antar titik sekutu. Bentuk persamaannya seperti pada persamaan (2.14).
4. Perhitungan beda Absis dan Ordinat koordinat titik-titik yang akan ditransformasikan (Sistem Koordinat Lama atau Lokal) dengan koordinat titik sekutu (Sistem Koordinat Baru atau TM-3³), hingga didapatkan satu harga selisih terbagi ($\Delta^{(n-1)}Z_{(1)}$). Bentuk persamaannya sama seperti pada langkah 3.c, dengan bentuk persamaannya adalah persamaan (2.14).

5. *Penghitungan Koordinat Titik Lama (Lokal) menjadi Koordinat Baru (TM-3⁰), penghitungan koordinat Lama (lokal) menjadi sistem koordinat Baru (TM-3⁰) dengan menggunakan rumus 2.15a bila menggunakan Titik Sekutu Pertama atau menggunakan rumus 2.15b bila menggunakan Titik Sekutu terakhir.*

C. Penjelasan Diagram Alir Transformasi Koordinat Metode Lauf Koreksi Sudut dan Jarak

Ada beberapa langkah yang dilakukan dalam Transformasi Lauf koreksi

Sudut dan Jarak Dengan Metode Numerik, yaitu:

1. *Penyusunan Matrik Parameter.* Penyusunan Matrik Parameter ini menggunakan metode Helmert dengan ketentuan sebagai berikut:

- a. *Penyusunan Matrik Sekutu Koordinat Lama (Lokal) yaitu (Matrik*

- A), Matrik A merupakan matrik deesain yang berisi nilai koordinat sistem lama (lokal) dengan dimensi dengan ketentuan (n-1), jadi apabila ada "n" titik sekutu koordinat, maka dimensi dari Matrik A adalah ada 4 (empat) kolom dan baris 2(n-1). Bentuk matrik A adalah seperti pada persamaan (2.16)

- b. *Penyusunan Matrik Sekutu Koordinat Baru (TM-3⁰) (Matrik F).*

- Matrik F merupakan matrik sekutu yang berisi nilai koordinat sistem TM-3⁰ dengan dimensinya adalah 1 (satu) kolom dengan barisnya yaitu "2n-1". Bentuk matrik F adalah seperti pada persamaan (2.16)

- c. *Penyusunan Matrik Titik Koordinat Lama (Lokal) yang akan ditransformasikan (Matrik A2).* Matrik A2 merupakan matrik yang

berisikan koordinat titik sistem lokal dan 1 (satu) koordinat dalam system TM-3⁰, jadi dimensi matrik A2 adalah "2(n+1) untuk jumlah barisnya dan ada 4 (empat) jumlah kolom.

d. *Penghitungan Matrik Parameter (Matrik X)*, Matrik X diperoleh dari hasil proses matematik dengan metode Helmert, hasilnya adalah ada 4 (empat) parameter transformasi. Dimensi Matrik X adalah 4 (empat) baris, 1 (satu) kolom). Bentuk matrik parameter X adalah seperti pada persamaan (2.18)

e. *Penghitungan Koordinat Pendekatan (X₀, Y₀)*, Koordinat Pendekatan diperoleh dari proses matematik dengan menggunakan perhitungan matrik (metode helmert), koordinat pendekatan merupakan hasil perkalian antara Matrik A2 dengan Matrik X.

2. *Penghitungan Argumen ($\{\Delta Z_n\}$) Titik Sekutu Terakhir*. Argumen ($\{\Delta Z_n\}$) merupakan sudut jurusan vector koreksi titik sekutu terakhir, bentuk persamaan adalah seperti pada persamaan (2.22).

3. *Penghitungan Modulus ($|\Delta Z_n|$)*. Modulus merupakan jarak vector koreksi titik sekutu terakhir, bentuk persamaan adalah seperti pada persamaan (2.21).

4. *Penghitungan D₀ dan α_0* antara titik sekutu dengan titik yang akan ditransformasi koordinatnya, bentuk persamaannya adalah seperti pada persamaan (2.23).

5. *Penghitungan Modulus ($|\Delta Z_n|$) dan Argumen ($\{\Delta Z_n\}$) Titik Koordinat Obyek (Koordinat yang akan ditransforamsi)*, bentuk persamaannya seperti pada persamaan (2.24).

6. Penghitungan Koordinat Titik Obyek sehingga diperoleh kopordinat hasil transformasi yang definitif, bentuk persamaannya seperti pada persamaan (2.25).

III. 4. Penjelasan Diagram Alir Paket Program :

1. Transformasi Koordinat adalah tampilan menu utama paket program ini, yang terdiri dari Metode Affine dan Metode Lauf.
2. Metode Affine adalah menu untuk melakukan penghitungan Transformasi Koordinat 2-Dimensi dengan Metode Affine.
 - a. Input Identitas Daerah adalah menu untuk memasukkan Nama Daerah yang dilakukan Transformasi Koordinatnya
 - b. Input Jumlah Titik Sekutu adalah menu untuk memasukkan jumlah titik sekutu yang akan digunakan
 - c. Ukuran Matrik F adalah menu untuk memasukkan ukuran dari Matrik F yang akan digunakan
 - d. Ukuran Matrik A2 adalah menu untuk memasukkan ukuran dari Matrik A2 yang akan digunakan
 - e. Input Koordinat Titik Sekutu Koordinat Lama (Lokal) adalah menu untuk memasukkan Koordinat Titik Sekutu Koordinat Lama (Lokal) yang akan digunakan
 - f. Input Koordinat Titik Sekutu Koordinat Baru (TM-3⁰) adalah menu untuk memasukkan Koordinat Titik Sekutu Koordinat Baru (TM-3⁰) yang akan digunakan

- g. Input Koordinat Titik Koordinat Lama (Lokal) adalah menu untuk memasukkan Koordinat Titik Koordinat Lama (Lokal) yang akan ditransformasi koordinatnya menjadi Koordinat Baru (TM-3⁰)
 - h. Matrik A adalah menu untuk menampilkan Matrik A yang terbentuk
 - i. Matrik F adalah menu untuk menampilkan Matrik F yang terbentuk
 - j. Matrik A2 adalah menu untuk menampilkan Matrik A2 yang terbentuk
 - k. Matrik X adalah menu untuk menampilkan Matrik X yang terbentuk dari hasil hitungan
 - l. Faktor Rotasi dan Faktor Perbesaran adalah menu untuk menampilkan Faktor Rotasi dan Faktor Perbesaran terhadap sumbu X dan Surnbu Y dari hasil hitungan
3. Metode Lauf Selisih Terbagi adalah menu untuk melakukan penghitungan Transformasi Koordinat 2-Dimensi dengan Metode Lauf Selisih Terbagi.
- a. Input Identitas Daerah adalah menu untuk memasukkan Nama Daerah yang dilakukan Transformasi Koordinatnya
 - b. Input Jumlah Titik Sekutu adalah menu untuk memasukkan jumlah titik sekutu yang akan digunakan
 - c. Input Koordinat Titik Sekutu Koordinat Lama (Lokal) adalah menu untuk memasukkan Koordinat Titik Sekutu Koordinat Lama (Lokal) yang akan digunakan
 - d. Input Koordinat Titik Sekutu Koordinat Baru (TM-3⁰) adalah menu untuk memasukkan Koordinat Titik Sekutu Koordinat Baru (TM-3⁰) yang akan digunakan

- e. Input Koordinat Lama (Lokal) Yang Akan Ditransformasi adalah menu untuk memasukkan Koordinat Titik Koordinat Lama (Lokal) yang akan ditransformasi koordinatnya menjadi Koordinat Baru (TM-3⁰)
 - f. Beda Koordinat adalah menu untuk menampilkan hasil perhitungan Beda Koordinat sampai dengan $n-1$, dengan n adalah jumlah titik sekutu
 - g. Selisih Terbagi adalah menu untuk menampilkan hasil perhitungan Selisih Terbagi sampai dengan $n-1$, dengan n adalah jumlah titik sekutu
4. Metode Lauf Koreksi Sudut dan Jarak adalah menu untuk melakukan penghitungan Transformasi Koordinat 2-Dimensi dengan Metode Lauf Koreksi Sudut dan Jarak.
- a. Input Identitas Daerah adalah menu untuk memasukkan Nama Daerah yang dilakukan Transformasi Koordinatnya
 - b. Input Jumlah Titik Sekutu adalah menu untuk memasukkan jumlah titik sekutu yang akan digunakan
 - c. Input Koordinat Titik Sekutu Koordinat Lama (Lokal) adalah menu untuk memasukkan Koordinat Titik Sekutu Koordinat Lama (Lokal) yang akan digunakan
 - d. Input Koordinat Titik Sekutu Koordinat Baru (TM-3⁰) adalah menu untuk memasukkan Koordinat Titik Sekutu Koordinat Baru (TM-3⁰) yang akan digunakan
 - e. Input Koordinat Lama (Lokal) Yang Akan Ditransformasi adalah menu untuk memasukkan Koordinat Titik Koordinat Lama (Lokal)

yang akan ditransformasi koordinatnya menjadi Koordinat Baru (TM-3⁰)

- f. Matrik A adalah menu untuk menampilkan Matrik A yang terbentuk
- g. Matrik F adalah menu untuk menampilkan Matrik F yang terbentuk
- h. Matrik A2 adalah menu untuk menampilkan Matrik A2 yang terbentuk
- i. Beda Koordinat Antar Titik Sekutu adalah menu untuk menampilkan hasil perhitungan yang berupa nilai beda koordinat antar titik sekutu
- j. Beda Koordinat Antar Titik Sekutu dan Titik Obyek adalah menu untuk menampilkan hasil perhitungan yang berupa nilai beda koordinat antar titik sekutu terakhir dengan titik obyek yang akan ditransformasi
- k. Modulus dan Argumen Titik Sekutu Terakhir adalah menu yang menampilkan hasil hitungan yang berupa Modulus dan Argumen titik sekutu terakhir.
- l. Modulus dan Argumen Titik Obyek adalah menu yang menampilkan hasil hitungan yang berupa Modulus dan Argumen titik obyek yang akan ditransformasi.

5. Koordinat Hasil Transformasi adalah menu yang menampilkan Koordinat Hasil Transformasi.

III. 3. Pembuatan Form Menu Utama

III. 3. 1. Langkah pembuatan menu utama program

1. sebelum masuk Visual Basic, terlebih dahulu buatlah desain gambar untuk menjadi background serta gambar tombol yang akan digunakan pada form menu utama atau form-form yang lain dengan menggunakan program Macromedia Fireworks 3.0 kemudian simpan dengan format (*.bmp)
2. Jalankan Visual Basic dan pilih project Standart EXE.
3. Pada Form tersebut aturlah properti sebagai berikut :

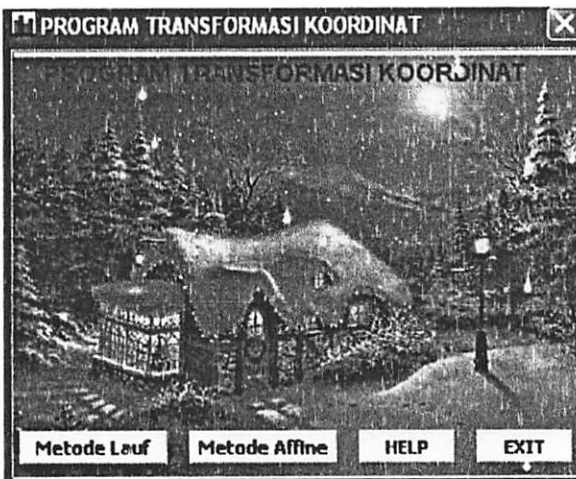
Komponen	Property	
	Type	Nilai
Form	Name	FormSnow
	Border style	1-Fixed Single
	Picture	None

4. Panggillah Connectics Irregular Form Shaper Control melalui pilihan Componen dari project menu. Control tersebut berfungsi untuk menghilangkan background gambar yang tidak diperlukan.
5. Pada Form tersebut tambahkan 4 (empat) CoomandButton yang akan digunakan sebagai tombol-tombol untuk kontrol program. Atur propertinya sebagai berikut:

Komponen	Property	
	Type	Nilai
CommandButton	Name	cmdMenuAffine
	Button Type	1-3D
	Caption	Metode Affine
	Index	0
	Name	cmdMenuLauf
CommandButton	Button Type	1-3D
	Caption	Metode Lauf
	Index	0
	Name	cmdPetunjuk
CommandButton	Button Type	1-3D

	Caption	HELP
	Index	0
CommandButton	Name	cmdExit
	Button Type	1-3D
	Caption	EXIT
	Index	0

6. Tulis kode program pada jendela code FormSnow yang dapat dilihat pada lampiran Listing Program.
7. Hasil pembuatan Menu Utama Program dapat dilihat pada gambar 3.1 dibawah ini :



Gambar 3. 1. Menu Utama Program






III. 3. 2. Langkah pembuatan Form Data Daerah :

1. Tambahkan form baru yang akan digunakan untuk input identitas daerah. Atur propertinya sebagai berikut :

Komponen	Property	
	Type	Nilai
Form	Name	FormDataDaerah
	Border style	3-Fixed Dialog
	Caption	Data Daerah
	Picture	None

2. Tambahkan 6 (enam) buah icon Toolbar yaitu:

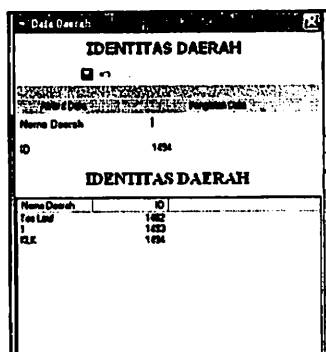
- a.  (new) tool untuk input data baru.

- b.  (edit) tool untuk mengedit data yang sudah ada.
- c.  (delete) tool untuk menghapus data daerah yang sudah tidak digunakan.
- d.  (undo) tool untuk membatalkan perintah atau tindakan yang terakhir pada pengisian data daerah.
- e.  (save) tool untuk menyimpan data daerah yang baru atau yang diedit.
- f.  (print) untuk mencetak hasil hitungan transformasi koordinat.

3. Tambahkan 2 (dua) TextBox yang akan digunakan sebagai kotak isian Nama Daerah dan ID Daerah.
4. Tambahkan List View yang digunakan untuk menampilkan identitas daerah serta Id Daerah baik data daerah yang baru di input maupun data daerah yang telah diedit. Atur propertinya sebagai berikut:

Komponen	Property	
	Type	Nilai
ListView	Name	LV
	Border style	1-ccFixed Single
	Picture	None
	View	3-lvwReport

5. Tulis kode program pada jendela code FormDataDaerah, yang dapat dilihat pada lampiran Listing Program.
6. Hasil Pembuatan Form Data Daerah dapat dilihat pada gambar 3.2 dibawah ini.



Gambar 3. 2. Form Data Daerah

III. 3. 3. Langkah Pembuatan Form Metode Affine :

A. Langkah Pembuatan Form Input Data

1. Buat Form baru yang akan digunakan untuk input data koordinat daerah yang akan dilakukan transformasi koordinat. Atur propertinya sebagai berikut :

Komponen	Property	
	Type	Nilai
Form	Name	FormInputData
	Border style	1-Fixed Single
	Caption	Input Data Metode Affine

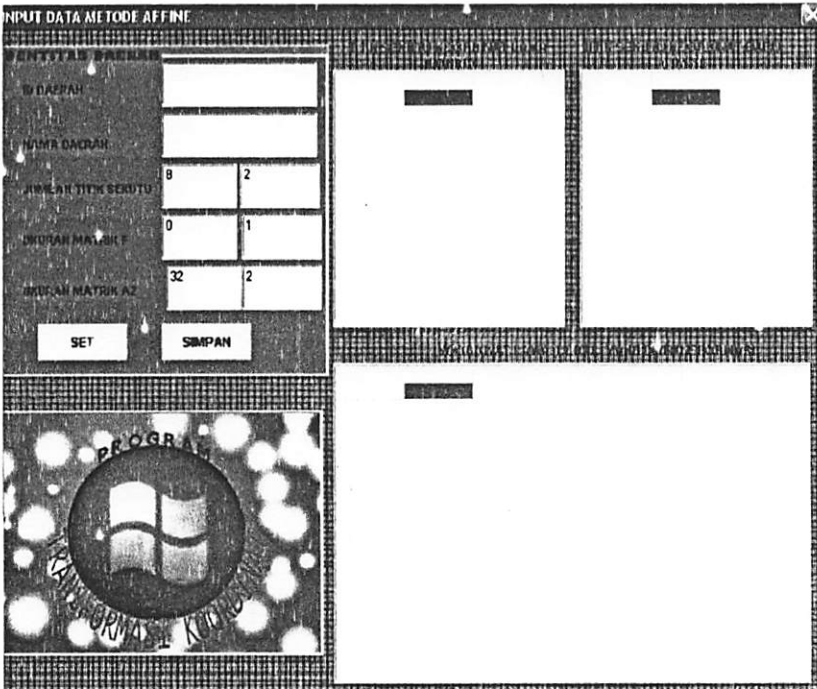
2. Pada Form tersebut tambahkan frame yang didalamnya ditambahkan 8 (delapan) Text Box yang akan digunakan sebagai kotak isian ID Daerah, Nama Daerah, Jumlah Titik Sekutu, Ukuran Matrik F, Ukuran Matrik A2, dan 2 (dua) Command Button yang akan digunakan sebagai tombol-tombol untuk kontrol program. Atur properti 2 (dua) CommandButton tersebut sebagai berikut :

Komponen	Property	
	Type	Nilai
CommandButton	Name	cmdSet
	Border style	1-Fixed Single
	Caption	SET
	Index	0
CommandButton	Name	cmdSimpan
	Border style	1-Fixed Single
	Caption	SIMPAN
	Index	0

3. Pada form tersebut tambahkan 3 (tiga) VSFlexGrid yang akan digunakan untuk pengisian koordinat-koordinat titik sekutu serta koordinat titik obyek yang akan ditransformasikan. Atur propertinya sebagai berikut :

Komponen	Property	
	Type	Nilai
VSFlexGrid	Name	flxA
	Border style	1-FlexBorderFlat
	Index	0
VSFlexGrid	Name	flxB
	Border style	1-FlexBorderFlat
	Index	0
VSFlexGrid	Name	flxA2
	Border style	1-FlexBorderFlat
	Index	0

4. Tulis kode program pada jendela code FormInputData, yang dapat dilihat pada lampiran Listing Program.
5. Hasil Pembuatan Form Input Data Metode Affine dapat dilihat pada gambar 3.3 dibawah ini.



Gambar 3. 3. Form Input Data Metode Affine

B. Langkah Pembuatan Proses Hitung

1. Tambahkan Form baru yang akan digunakan untuk proses hitung. Atur propertinya sebagai berikut :

Komponen	Property	
	Type	Nilai
Form	Name	FormUtama
	Border style	1-Fixed Single
	Caption	Metode Affine

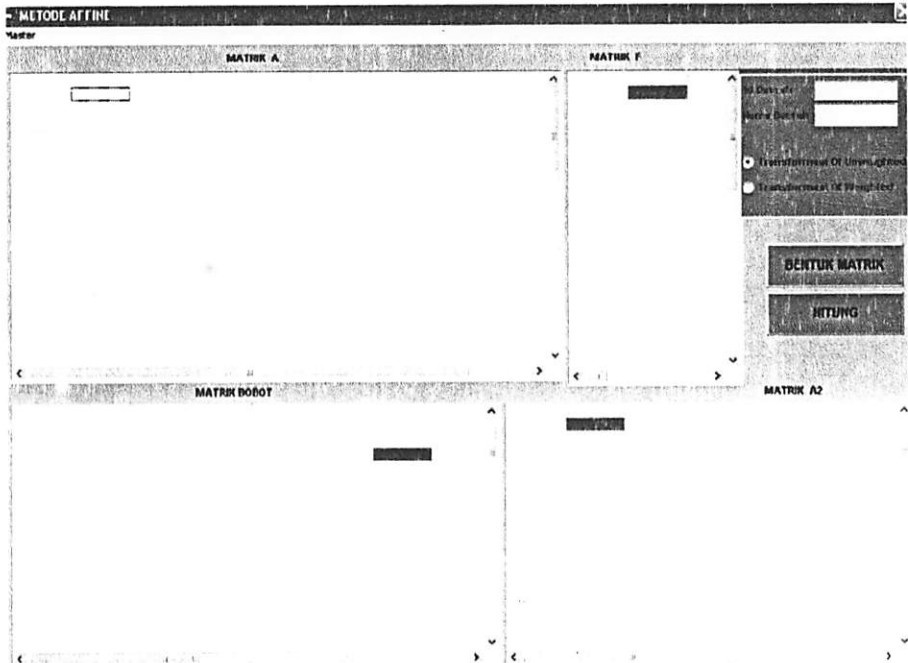
2. Tambahkan 2 (dua) TextBox yang akan digunakan untuk pengisian Id Daerah dan Nama Daerah.
3. Tambahkan 2 (dua) CommandButton yang akan digunakan sebagai tombol-tombol untuk kontrol program. Atur propertinya sebagai berikut:

Komponen	Property	
	Type	Nilai
CommandButton	Name	cmdBentukMatrik
	Border style	1-3D
	Caption	Bentuk Matrik
	Index	0
CommandButton	Name	cmdHitung
	Border style	1-3D
	Caption	HITUNG
	Index	0

4. Tambahkan 4 (empat) VSFlexGrid yang akan digunakan untuk menampilkan hasil dari proses hitung transformasi koordinat. Atur propertinya sebagai berikut :

Komponen	Property	
	Type	Nilai
VSFlexGrid	Name	flxA
	Border style	1-FlexBorderFlat
	Index	0
VSFlexGrid	Name	flxB
	Border style	1-FlexBorderFlat
	Index	0
VSFlexGrid	Name	flxA2
	Border style	1-FlexBorderFlat
	Index	0
VSFlexGrid	Name	flxMatrikBobot
	Border style	1-FlexBorderFlat
	Index	0

- 5. Tulis kode program pada jendela code FormUtama, yang dapat dilihat pada lampiran Listing Program.
- 6. Hasil pembuatan Form Proses Hitung dapat dilihat pada gambar 3.4 dibawah ini:



Gambar 3. 4. Form Proses Hitung Transformasi Metode Affine

C. Langkah Pembuatan Form Hasil Metode Affine

- 1. Tambahkan Form baru yang akan digunakan untuk menampilkan hasil transformasi koordinat metode Affine. Atur propertinya sebagai berikut:

Komponen	Property	
	Type	Nilai
Form	Name	FormHasil
	Border style	1-Fixed Single
	Caption	Hasil Transformasi

- 2. Tambahkan 4 (empat) Text Box yang digunakan untuk menampilkan Faktor Perbesaran Terhadap X, Faktor Perbesaran Terhadap Y, Faktor Rotasi Terhadap X, dan Faktor Rotasi Terhadap Y.

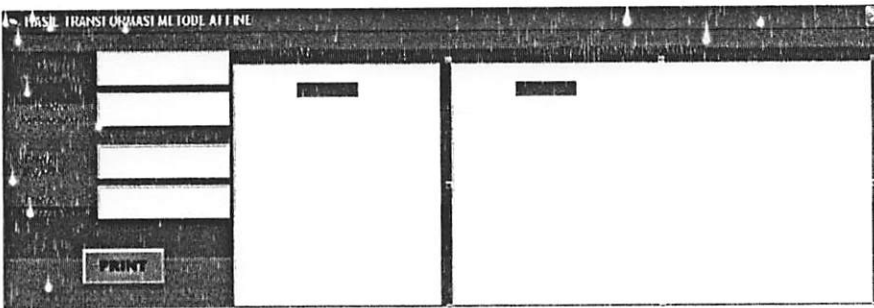
- 3. Tambahkan 2 (dua) VSFlexGrid yang digunakan untuk menampilkan Hasil Transformasi, serta hasil dari Matrik X. Atur propertinya sebagai berikut:

Komponen	Property
----------	----------

	Type	Nilai
VSFlexGrid	Name	flxX
	Border style	1-FlexBorderFlat
	Index	0
VSFlexGrid	Name	flxF
	Border style	1-FlexBorderFlat
	Index	0

4. Tulis kode program pada jendela code FormHasil, yang dapat dilihat pada lampiran Listing Program.

5. Hasil pembuatan Form Hasil dapat dilihat pada gambar 3.5 dibawah ini:



Gambar 3. 5. Form Hasil Transformasi Metode Affine

III. 3. 4. Langkah Pembuatan Form Metode Lauf :

A. Langkah Pembuatan Form Input Data Metode Lauf

1. Tambahkan Form baru yang akan digunakan untuk input data transformasi koordinat Metode Lauf baik Metode Lauf Selisih Terbagi maupun Metode Lauf Koreksi Sudut Jarak. Atur propertinya sebagai berikut :

Komponen	Property	
	Type	Nilai
Form	Name	FormLauf
	Border style	1-Fixed Single
	Caption	Metode Lauf

2. Tambahkan 3 (tiga) TextBox yang akan digunakan sebagai kotak isian Id Daerah, Nama Daerah dan Jumlah Titik Obyek.

3. Tambahkan 3 (tiga) VSFlexGrid yang akan digunakan sebagai kotak isian Titik Sekutu Koordinat Lokal, Titik Sekutu Koordinat Nasional serta Koordinat Titik Obyek yang ditransformasi. Atur propertinya sebagai berikut :

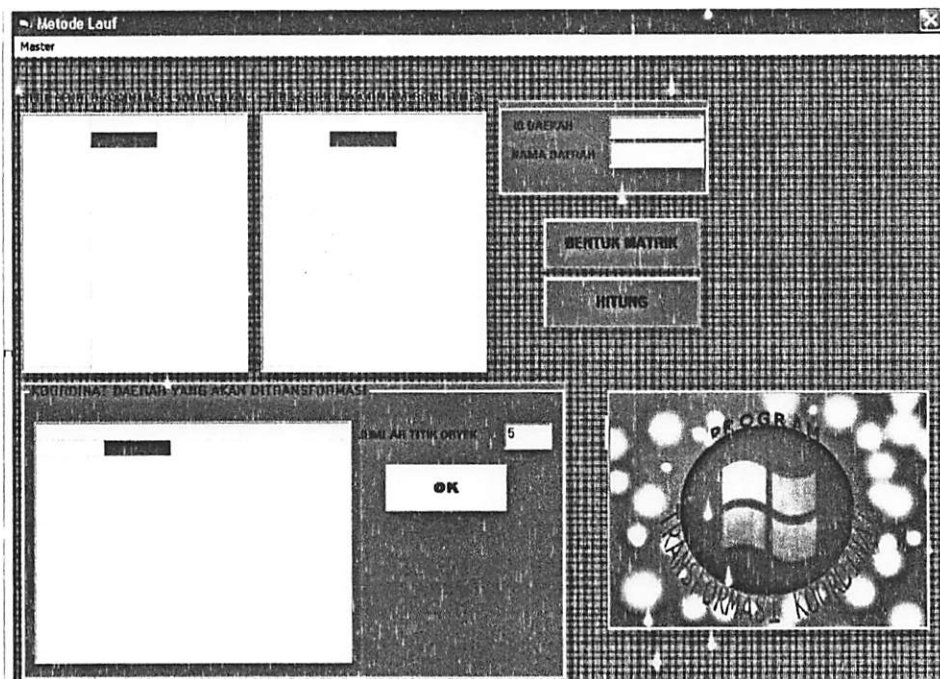
Komponen	Property	
	Type	Nilai
VSFlexGrid	Name	flxA
	Border style	1-FlexBorderFlat
	Index	0
VSFlexGrid	Name	flxTM-3
	Border style	1-FlexBorderFlat
	Index	0
VSFlexGrid	Name	flxCalon
	Border style	1-FlexBorderFlat
	Index	0

4. Tambahkan 3 (tiga) CommandButton yang akan digunakan sebagai tombol-tombol untuk kontrol program. Atur propertinya sebagai berikut:

Komponen	Property	
	Type	Nilai
CommandButton	Name	cmdBentukMatrikA
	Border style	1-3D
	Caption	Bentuk Matrik
	Index	0
CommandButton	Name	cmdHitung
	Border style	1-3D
	Caption	HITUNG
	Index	0
CommandButton	Name	cmdSetInput
	Border style	1-3D
	Caption	OK
	Index	0

5. Tulis kode program pada jendela code frmLauf yang dapat dilihat pada lampiran Listing Program

6. Hasil pembuatan Form Lauf dapat dilihat pada gambar 3.6. dibawah ini:



Gambar 3. 6. Form Input Data Metode Lauf

B. Langkah Pembuatan Form Hasil Transformasi Koordinat Metode Lauf Selisih Terbagi

1. Tambahkan form baru yang akan digunakan untuk menampilkan hasil transformasi koordinat Metode Lauf Selisih Terbagi. Atur propertinya sebagai berikut :

Komponen	Property	
	Type	Nilai
Form	Name	FormHasilSelisihTerbagi
	Border style	1-Fixed Single
	Caption	Hasil Metode Lauf Selisih Terbagi

2. Tambahkan 7 (tujuh) VSFlexGrid yang akan digunakan untuk menampilkan hasil dari perhitungan transformasi dengan Metode Lauf Selisih Terbagi. Atur propertinya sebagai berikut :

Komponen	Property	
	Type	Nilai
VSFlexGrid	Name	flxBedaAbsisSatu
	Border style	1-FlexBorderFlat
	Index	0

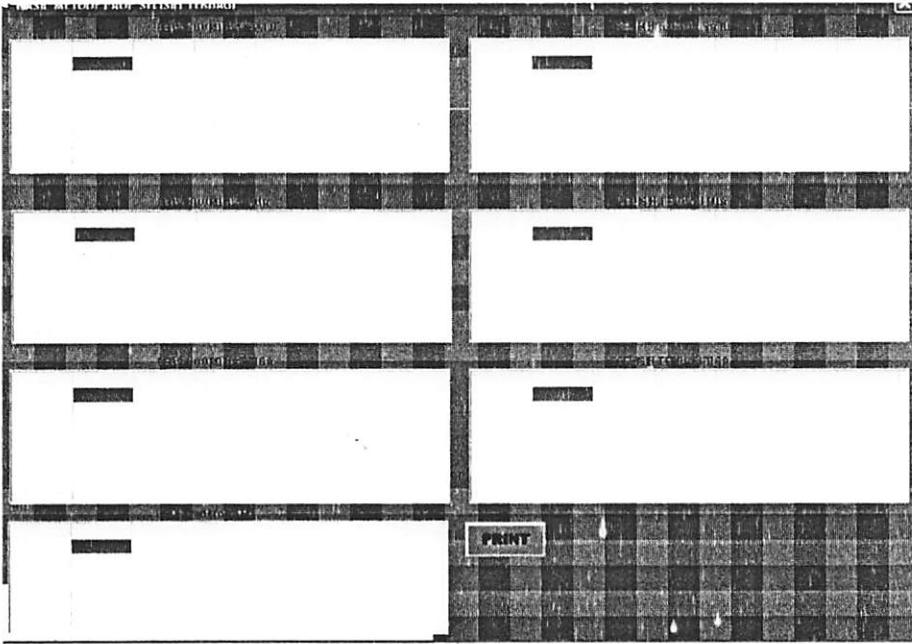
VSFlexGrid	Name	flxSelisihSatu
	Border style	1-FlexBorderFlat
	Index	0
VSFlexGrid	Name	flxBedaDua
	Border style	1-FlexBorderFlat
	Index	0
VSFlexGrid	Name	flxSelisihDua
	Border style	1-FlexBorderFlat
	Index	0
VSFlexGrid	Name	flxBedaTiga
	Border style	1-FlexBorderFlat
	Index	0
VSFlexGrid	Name	flxSelisihTiga
	Border style	1-FlexBorderFlat
	Index	0
VSFlexGrid	Name	flxHasil
	Border style	1-FlexBorderFlat
	Index	0

3. Tambahkan 1 (satu) **CommandButton** yang digunakan sebagai tombol untuk kontrol program. Atur propertinya sebagai berikut :

Komponen	Property	
	Type	Nilai
CommandButton	Name	cmdPrint
	Border style	1-3D
	Caption	PRINT
	Index	0

4. Tulis kode program pada jendela code **FormHasilSelisihTerbagi** yang dapat dilihat pada lampiran Listing Program.

5. Hasil pembuatan **Form Hasil Selisih Terbagi** dapat dilihat pada gambar 3.7 dibawah ini :



Gambar 3. 7. Form Hasil Transformasi

Metode Lauf Selisih Terbagi

C. Langkah Pembuatan Form Hasil Transformasi Koordinat Metode Lauf Koreksi Sudut Jarak

1. Tambahkan form baru yang akan digunakan untuk menampilkan hasil transformasi koordinat Metode Lauf Koreksi Sudut Jarak. Atur propertinya sebagai berikut :

Komponen	Property	
	Type	Nilai
Form	Name	FormKoreksi
	Border style	1-Fixed Single
	Caption	Hasil Metode Lauf Koreksi

2. Tambahkan 6 (enam) VSFlexGrid yang akan digunakan untuk menampilkan hasil dari perhitungan transformasi dengan Metode Lauf Koreksi Sudut Jarak. Atur propertinya sebagai berikut :

Komponen	Property	
	Type	Nilai
VSFlexGrid	Name	flxAHelmert
	Border style	1-FlexBorderFlat
	Index	0

VSFlexGrid	Name	flxFHelmert
	Border style	1-FlexBorderFlat
	Index	0
VSFlexGrid	Name	flxA2Helmert
	Border style	1-FlexBorderFlat
	Index	0
VSFlexGrid	Name	flxX
	Border style	1-FlexSelisihDua
	Index	0
VSFlexGrid	Name	flxZD
	Border style	1-FlexBorderFlat
	Index	0
VSFlexGrid	Name	flxSudut
	Border style	1-FlexBorderFlat
	Index	0
VSFlexGrid	Name	flxModulus
	Border style	1-FlexBorderFlat
	Index	0
VSFlexGrid	Name	flxArgumen
	Border style	1-FlexBorderFlat
	Index	0
VSFlexGrid	Name	flxHasil
	Border style	1-FlexBorderFlat
	Index	0

3. Tambahkan 2 (dua) TextBox yang akan digunakan untuk menampilkan hasil dari Modulus Titik Sekutu Terakhir dan Argumen Titik Sekutu Terakhir.

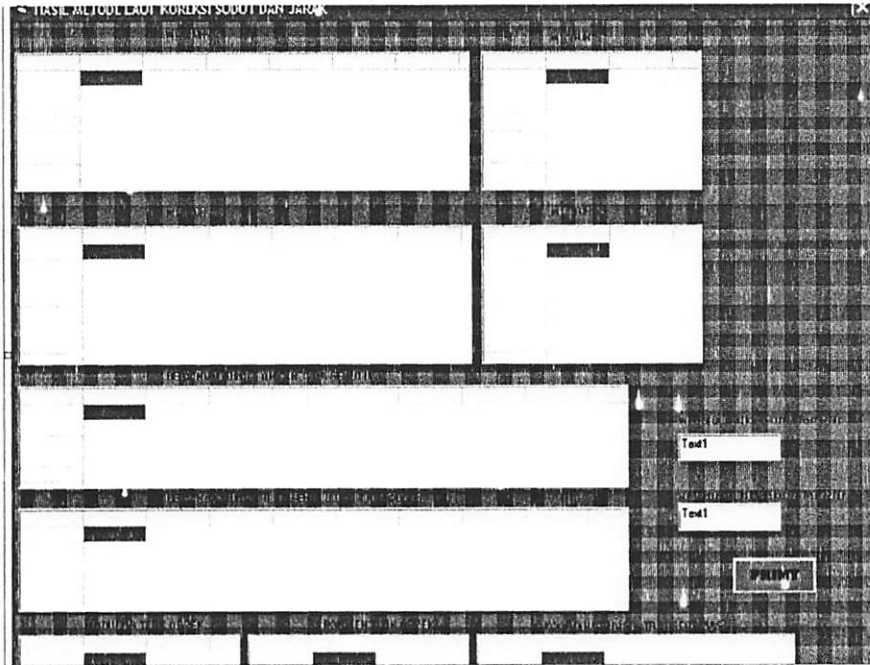
4. Tambahkan 1 (satu) CommandButton yang akan digunakan sebagai tombol kontrol program. Atur propertinya sebagai berikut :

Komponen	Property	
	Type	Nilai
CommandButton	Name	cmdPrint
	Border style	1-3D
	Caption	PRINT
	Index	0

5. Tulis kode program pada jendela code FormKoreksi yang dapat dilihat pada lampiran Listing Program.

6. Hasil pembuatan Form Koreksi dapat dilihat pada gambar 3.8 dibawah ini

:



Gambar 3. 8. Form Hasil Transformasi

Metode Lauf Koreksi Sudut Jarak

III. 3. 5. Langkah Pembuatan Form Petunjuk program :

1. Sebelumnya buatlah petunjuk program terlebih dahulu yang dilakukan dengan bantuan software Microsoft Word 2003. Hasilnya disimpan dengan nama file (Help.rtf).
2. Tambahkan form baru yang akan digunakan untuk menampilkan hasil petunjuk program yang telah dibuat tersebut. Aturilah propertinya sebagai berikut :

Komponen	Property	
	Type	Nilai
Form	Name	FormRtf
	Border style	1-Fixed Single
	Caption	Help Transformasi

3. Tambahkan kontrol OLE yang akan digunakan untuk memanggil file (Help.rtf). Klik data source pada propertinya dan cari file tempat (Help.rtf) disimpan.
4. Tulislah kode program pada jendela code FormRtf yang dapat dilihat pada lampiran listing program.

BAB IV

HASIL DAN ANALISA

Hasil yang diperoleh dari penelitian ini adalah suatu program perhitungan Transformasi Koordinat dengan menggunakan Metode Lauf Selisih Terbagi, Metode Lauf Koreksi Sudut Jarak serta Transformasi Affine. Tujuan utama dari program ini adalah supaya benar-benar memudahkan pengguna khususnya dalam melakukan hitung Transformasi Koordinat, sehingga program ini dibuat sedemikian rupa untuk memudahkan penggunaanya, sehingga tampilannya, penempatan tombol-tombol, maupun proses hitungannya.

Form Utama Program Hitung Transformasi Koordinat terdiri dari beberapa Menu Utama yaitu Metode Lauf, Metode Affine, Help, Exit. Berikut Tampilan dari Menu Utama Program Hitung Transformasi Koordinat seperti pada gambar IV.1 dibawah ini.



Gambar IV.1. Menu Utama Program


V.1. Menu Transformasi Koordinat Metode Affine




Pada Menu Utama Program Hitung Transformasi Koordinat Klik Menu **Metode Affine** untuk memilih program hitung transformasi dengan menggunakan metode Affine untuk melakukan proses hitung Transformasi Koordinat Metode Affine, pada Menu Utama seperti pada gambar IV.1, masuk pada Menu Metode Affine maka akan tampil Form Perhitungan Transformasi Koordinat Metode Affine yang berisi Sub Menu antara lain Id Daerah, Bentuk Matrik, Hitung, Transformasi Of Unweighted dan Transformasi Of Weighted, bentuk tampilan Sub Menu Transformasi Koordinat Metode Affine.

IV.1.1. Form Input Data

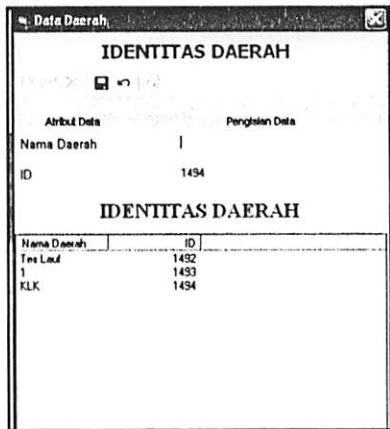
Pada menu Master di Sub Menu Utama terdapat Form Input Data yang terdapat Sub Menu yang terdiri antara lain, Id Daerah, Nama Daerah, Banyak Titik Sekutu, Ukuran Matrik B, Ukuran Matrik A2, Set, Simpan. Form Input Data digunakan untuk mengisi Id Daerah yang akan ditransformasikan koordinatnya, mengisi banyaknya titik sekutu, menentukan ukuran Matrik B, menentukan ukuran Matrik A2, serta menyimpan data input. Form Input Data digunakan untuk menginput data-data yang diperlukan dalam proses hitung transformasi koordinat metode Affine seperti data banyaknya titik sekutu.

Pada Form Identitas Daerah ini digunakan untuk untuk mengisi identitas daerah yang akan dilakukan Transformasi Koordinat. Ada beberapa sub menu yang penting dalam Form Identitas Daerah yaitu:

1.  adalah sub menu untuk memasukan identitas daerah baru yang akan dilakukan proses hitung transformasi koordinat titik.

2.  adalah sub menu untuk mengedit identitas daerah dan data yang lama pada daerah yang sama.
3.  adalah sub menu untuk menghapus idenstitas daerah dan data lama yang sudah tidak diperlukan lagi.
4.  adalah sub menu untuk meyimpan data input identitas daerah baik data input yang baru maupun data editan.

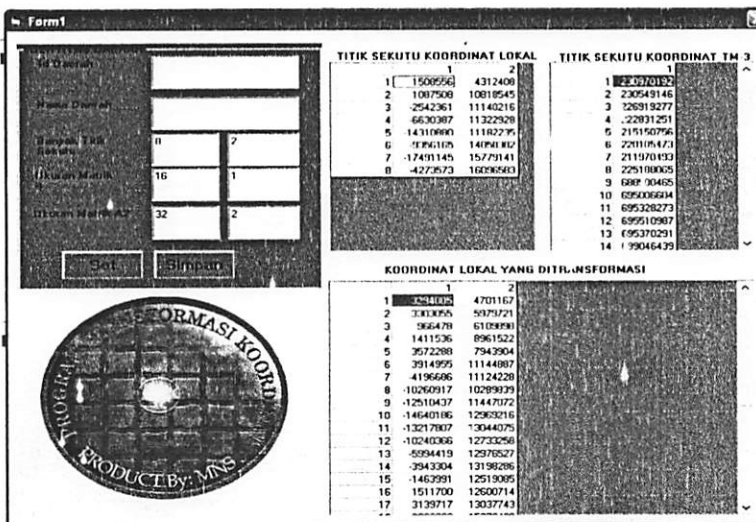
Bentuk tampilan Sub Menu Identitas Daerah seperti pada gambar IV.2. dibawah ini:



IDENTITAS DAERAH	
Atribut Data	
Nama Daerah	
ID	1494
IDENTITAS DAERAH	
Nama Daerah	ID
Tes Luf	1492
1	1493
KLK	1494

Gambar IV.2. Tampilan Form Identitas Daerah

Bentuk tampilan dari Sub Menu Input Data seperti pada gambar IV.3. dibawah ini:



TITIK SEKUTU KOORDINAT LOKAL	
1	2
1	1506956 4312400
2	1087500 10818545
3	2542361 11140216
4	6630387 11322928
5	14310660 11182795
6	5186105 14180182
7	-17491145 15729141
8	-4273573 16280503

TITIK SEKUTU KOORDINAT TM 3	
1	2
1	23070182
2	230549146
3	326918277
4	228931291
5	215150796
6	220105473
7	211970193
8	225108055
9	688 90465
10	695006634
11	695328273
12	695910987
13	695702291
14	79046439

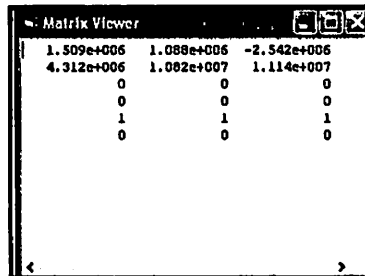
KOORDINAT LOKAL YANG DITRANSFORMASI	
1	2
1	2581038 4701167
2	3313365 5979721
3	965478 6108891
4	1411536 8961522
5	3672288 7942904
6	3914955 11144887
7	4196696 11124228
8	-10260917 10289399
9	-12510437 11447072
10	-14640186 12963216
11	-13217887 13044075
12	-10240366 12733256
13	-5894419 12976527
14	-3943304 13198286
15	-1462991 12519385
16	1511700 12600714
17	3139717 13037743

Gambar IV.3. Sub Menu Input Data Transformasi Koordinat Metode

Affine

IV.1.2. Form Proses Hitung Transformasi Koordinat Metode Affine

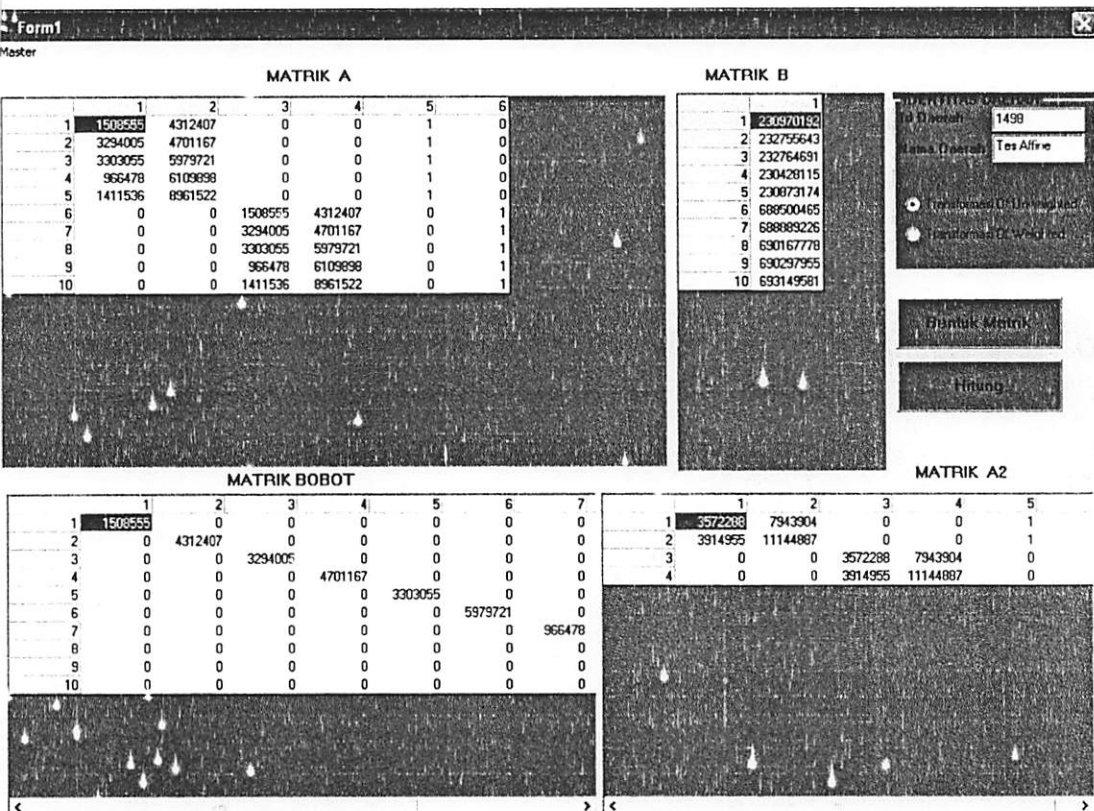
Pada form ini terdapat beberapa menu yaitu Id Daerah, Nama Daerah, Hitung, Transformasi Of Unweighted dan Transformasi Of Weighted, serta Bentuk Matrik. Form Proses Hitung ini digunakan untuk melakukan proses hitung transformasi koordinat sesuai dengan keinginan pengguna, sehingga diperoleh hasil proses hitung transformasi koordinat seperti nilai matrik A Transpose, dan nilai hasil proses hitung yang lain yang ditampilkan dalam bentuk message box. Contoh tampilan nilai hasil hitung dalam bentuk message box seperti pada gambar IV.4. dibawah ini:



Matrix Viewer		
1.509e+006	1.088e+006	-2.542e+006
4.312e+006	1.082e+007	1.114e+007
0	0	0
0	0	0
1	1	1
0	0	0

Gambar IV.4. Tampilan Message Box Nilai Matrik A Transpose

Sedangkan bentuk tampilan dari Sub Menu Proses Hitung adalah seperti pada gambar IV.5. dibawah ini:



Gambar IV.4. Sub Menu Proses Hitung dan Hasil Hitung Transformasi

Koordinat Metode Affine

IV.2. Program Transformasi Koordinat Metode Lauf

Pada Menu Utama Program Hitung Transformasi Koordinat Klik Menu

Metode Lauf untuk memilih program hitung transformasi dengan menggunakan metode Lauf. Pada program Transformasi Koordinat Metode Lauf didalamnya berisi program Transformasi Metode Selisih Terbagi dan Koreksi Sudut dan Jarak dan beberapa Sub Menu antara lain Form Identitas

Daerah, Input Data Daerah, Bentuk Matrik, Hitung, Print. Setelah Masuk pada Menu "Metode Lauf" maka akan tampak Form Metode Lauf, untuk mengisi identitas daerah pada menu "Master" pilih "Identitas Daerah".

IV. 2. 1. Sub Menu Input Data

Setelah mengisi identitas daerah form "identitas daerah" maka pada menu "Master" pilih "Data Daerah" untuk selanjutnya mengisi data koordinat titik sekutu dan koordinat titik lama (lokal) yang akan ditransformasi pada Form Input Data. Bentuk tampilan dari Form Input Data seperti pada gambar IV.5. dibawah ini:

The screenshot shows a software window titled "Input Data" with three main sections:

- IDENTITAS DAERAH:** Contains input fields for "ID DAERAH" (value: 1495), "NAMA DAERAH" (value: Baru), and "BANYAK TITIK SEKUTU" (values: 4 and 2). Below these are "SET" and "SIMPAN" buttons.
- TITIK SEKUTU KOORDINAT LAMA (LOKAL):** A table with columns 'x' and 'y' and rows 1-4.
- TITIK SEKUTU KOORDINAT BARU (TM-3):** A table with columns 'X' and 'Y' and rows 1-4.

	x	y
1	-48076,83	28946,225
2	9610,75	60421,075
3	24007,8	-21853,52
4	-39427,87	-31853,75

	X	Y
1	-614308,95	42228,565
2	-564910,78	36435,206
3	600740,331	-44978,367
4	652046,664	-6276,488

Gambar IV.5. Tampilan Form Input Data

IV.2.2. Sub Menu Proses Hitung Transformasi Koordinat Metode Lauf Selisih Terbagi

Pada Sub Menu proses hitung transformasi koordinat Metode Lauf Selisih Terbagi ini terdapat beberapa sub menu yakni:

1. Sub Menu Id Daerah dan Nama Daerah yang berfungsi untuk mencari Data Identitas Daerah yang akan dilakukan Proses Hitung Transformasi Koordinat.

2. Form Banyak Titik Obyek dan Daerah Yang Ditransformasi yang berfungsi sebagai tempat input data banyaknya titik yang akan ditransformasi.
3. Sub Menu Hitung yang berfungsi untuk proses hitung tranformasi koordinat Metode Lauf Selisih Terbagi.

Bentuk tampilan dari Sub Menu Proses Hitung seperti pada gambar

IV.6. dibawah ini:

(a) Metode Lauf

Mater

TITIK SEKUTU KOORDINAT LAMA (LOKAL)		TITIK SEKUTU KOORDINAT BARU (TM-3)			
x	y	X	Y		
1	-49076,83	29546,225	1	614308,98	42228,565
2	3610,75	60421,075	2	-564910,78	36435,206
3	24007,8	-21853,52	3	600740,331	-44978,367
4	-39427,87	-31853,75	4	652046,654	-6276,488

ID ZONE: 1495
 NAMA DAERAH: Bmw

BENTUK Matrik
 HITUNG

KOORDINAT DAERAH (TAMBAH DAERAH DITRANSFORMASI)

x	y
1	-41614,288
2	-9955,936
3	13707,882
4	-26183,472
5	-20538,72

JUMLAH TITIK OBYEK: 5
 OK

(b) HASIL METODE LAUF SELISIH TERBAGI

BEDA KOORDINAT SATU				SELISIH TERBAGI SATU			
x	y	X	Y	Zx	Zy	Delta x	Delta y
1	57587,58	31474,85	49388,17	-0,437419197	0	0	0
2	14387,35	40274,595	26623,951	0,096153001	0,590562634	0,269544631	-0,153141347
3	43405,67	-10000,23	-81306,333	0,6993395148	-0,719711645	-0,190898743	-0,129151111

BEDA KOORDINAT DUA				SELISIH TERBAGI DUA			
x	y	X	Y	Zx	Zy	Delta x	Delta y
1	72084,83	60795,745	0,26946821	0,153141347	1,0000000000000000	0	0
2	-49038,62	-9274,825	-0,190868743	-0,129151111	2,00001946530862	0,00001032843774	0,00001541045098

BEDA KOORDINAT TIGA				SELISIH TERBAGI TIGA			
x	y	X	Y	Zx	Zy	Delta x	Delta y
1	6640,96	40793,575	0,0001541045098	0,0001367564531	1,861182961296402	-2,797962403E-11	0

Hasil Transformasi

X	Y
1	620779,83414758
2	612953,414041257
3	5981025,888489971
4	453211,686728973
5	599925,601214675

OK

Gambar III.6. Tampilan Form Proses Hitung Transformasi Koordinat Selisih Terbagi

IV.2.3. Form Metode Lauf Koreksi Sudut Dan Jarak

Pada Menu "Master" Pilih "Koreksi Sudut Jarak" apabila perhitungan transformai menggunakan metode Lauf Koreksi Sudut dan Jarak. Bentuk tampilan dari form proses Hitung Koreksi Sudut dan Jarak adalah seperti pada gambar IV.7. dibawah ini.

HASIL METODE LAUF KOREKSI SUDUT DAN JARAK

Matrik A					Matrik F				
1	0	0	1	0	1	82017,574			
2	1802,776	-1,248	1	0	2	82820,323			
3	1005,984	-1503,462	1	0	3	83025,349			
4		0	0	1	4	590662,93			
5	1,248	1802,776	0	1	5	590663,849			
6	1503,462	1005,984	0	1	6	589160,701			

Matrik Az					Matrik X				
1	2813,017	-1504,928	1	0	1	84140,7476757605			
2	-41614,288	-5028,108	1	0	2	52548,3019707622			
3	-9955,936	21782,363	1	0	3	71157,084979802			
4	13707,882	-17795	1	0	4	94643,8102300982			
5	-28183,472	-49073,883	1	0	5	69345,7530785289			
6	-20608,72	-22901,756	1	0	6	70696,7065996197			
7	1504,928	2813,017	0	1	7	589737,707848372			
8	5028,108	-41614,288	0	1	8				

BEDA KOORDINAT ANTAR TITIK SEKUTU						
	XD	YO	X	Y	Delta Zx0	Delta Zy0
1	84140,7476757605	-589737,707848372	84832,355	589161,407	7324239482	3848372048
2	52548,3019707622	-580050,545380268	0	0	0	0
3	71157,084979802	-604542,822293476	0	0	0	0
4	94643,8102300982	-579696,426528299	0	0	0	0
5	69345,7530785289	-550325,859123816	0	0	0	0

BEDA KOORDINAT TITIK SEKUTU DAN TITIK OBYEK				
	X	Y	D0	Alpha
1	2123,17367576051	925,222151627997	2316,01003609776	1,15983619743143
2	1320,4246757605	926,141151627991	1612,84182643434	0,959133776809261
3	1115,3986757605	-577,006848372053	1255,80695529071	358,906603162319
4	-29469,2720292378	10612,784619732	31321,8885326275	358,774862789897
5	-30272,0210292378	10613,303619732	32078,6139182982	358,786410944106

MODULUS TITIK OBYEK		ARGUMEN TITIK OBYEK		KOORDINAT HASIL TRANSFORMASI	
1	6133760,7975213	1	368,111489621524	1	8445510,34248155
2	1191291,66228593	2	1,87049659302999	2	1209346,24610582
3	775642,75497554	3	2,416314335278	3	609161,574724485

Modulus Titik Sekutu Terakhir: 900,24627674

Argumen Titik Sekutu Terakhir: 0,8769098

PRINT

Gambar IV.7. Form Proses Hitung Koreksi Sudut dan Jarak

IV.3. Uji Ketelitian

Untuk mengetahui kebenaran dari hasil program perhitungan, maka hasil perhitungan dari program tersebut perlu dibandingkan dengan hasil perhitungan dengan menggunakan hitungan manual (Excel) sehingga hasil dari program perhitngan Transformasi Koordinat ini benar-benar telah teruji kebenarannya.

V.4. Analisa Hasil Perhitungan Transformasi Koordinat Metode Affine Dan Metode Lauf

Perujian program pada perhitungan Transformasi Koordinat Metode Lauf di dalamnya juga termasuk pengujian Selisih Terbagi dan koreksi Sudut Jarak serta Transformasi Koordinat Metode Affine, dengan cara membandingkan hasil pengolahan dari program perhitungan dengan menggunakan Visual Basic 6.0 dengan hasil perhitungan menggunakan perangkat lunak Microsoft Excel.

TRANSFORMASI KOORDINAT METODE AFFINE

Hasil perhitungan Transformasi Koordinat Metode Affine dengan Jumlah Titik Sekutu ada 5 (lima) dan jumlah Titik yang ditransformasikan ada 2 (dua) menggunakan program Visual Basic 6.0. yaitu:

Data Yang Di Input:

Ukuran Matrik A : (5 x 2)

Ukuran Matrik F : (10 x 1)

Ukuran Matrik A2 : (2 x 2)

Titik Sekutu Koordinat Lama (Lokal) dan Koordinat Baru (TM-3⁰)

Titik	x	y	X	Y
A	1508555	4312407	230970192	688500465
B	3294005	4701167	232755643	688889226
C	3303055	5979721	232764691	690167778
D	966478	6109898	230428115	690297955
E	1411536	8961522	230873174	693149581

Koordinat Titik yang ditransformasi, yaitu:

Titik	x	y
1	3572288	7943904
2	3914955	11144887

Matrik A:

1508555	4312407	0	0	1	0
3294005	4701167	0	0	1	0
3303055	5979721	0	0	1	0
966478	6109898	0	0	1	0
1411536	8961522	0	0	1	0
0	0	1508555	4312407	0	1
0	0	3294005	4701167	0	1
0	0	3303055	5979721	0	1
0	0	966478	6109898	0	1
0	0	1411536	8961522	0	1

Matrik F :

Matrik A2 :

230970192
232755643
232764691
230428115
230873174
688500465
688889226
690167778
690297955
693149581

$\begin{bmatrix} 3572288 & 7943904 & 0 & 0 & 1 & 0 \\ 3914955 & 11144887 & 0 & 0 & 1 & 0 \\ 0 & 0 & 3572288 & 7943904 & 0 & 1 \\ 0 & 0 & 3914955 & 11144887 & 0 & 1 \end{bmatrix}$

Hasil Perhitungan

Nilai Matrik X : $\begin{bmatrix} 0,999999920639779 \\ 1,090732164588 E - 07 \\ 1,7196897261762 E - 07 \\ 1,00000015351022 \\ 229461636,710545 \\ 684188056,71638 \end{bmatrix}$

Koordinat Hasil Transformasi:

Titik	X	Y
1	233033925,293515	692131962,550173
2	233376592,615462	695332946,100485

Faktor Perbesaran Terhadap X : 0,9999999920639779

Faktor Perbesaran Terhadap Y : 1,7196897261762E-07

Faktor Rotasi Terhadap X : -1,09073225114885E-07

Faktor Rotasi Terhadap Y : -1,57079615482595

Hasil Hitungan dengan menggunakan Software Microsoft Excel dengan

data yang sama yaitu:

Matrik X :

$$\begin{bmatrix} 0,9999999920639766 \\ 0,000000109073312 \\ 0,000000171966803 \\ 1,000000153510880 \\ 229461636,710545 \\ 684188056,716381 \end{bmatrix}$$

Koordinat Hasil transformasi :

Titik	X	Y
1	233.033.925,2493515	692.131.962,550172
2	233.376.592,615463	695.332.946,100485

Faktor Perbesaran Terhadap X : 0,9999999920639779

Faktor Perbesaran Terhadap Y : 1,719689726E-07

Faktor Rotasi Terhadap X : -1,090732251E-07

Faktor Rotasi Terhadap Y : -1,57079615482595

Hasil perhitungan Transformasi Koordinat Metode Lauf menggunakan program Visual Basic 6.0. yaitu:

TRANSFORMASI KOORDINAT METODE SELISIH TERBAGI

Data Yang Di Input:

Jumlah Titik Sekutu : 5

Jumlah Titik Yang Ditransformasi : 3

Titik Sekutu Koordinat Lama (lokal) dan Koordinat Baru (TM-3⁰)

Titik	Sistem Lama (Lokal)		Sistem Baru (TM-3 ⁰)	
	x	y	X	Y
A	27085,345	35160,745	46024,5	143860,35
B	47850,765	-2319,535	76224,5	113465
C	-4811,185	-45169,04	37240,93	57875,455
D	-33712,93	26511,975	-10110,545	118960,17
E	3487,06	-1680	33370,06	101980,199

Koordinat Titik Yang Ditransformasi:

Titik	Sistem Lama (Lokal)	
	x	y
1	9232	41169,5
2	-16717,58	14723,65
3	13381,13	-33017,685

Hasil Perhitungan :

Nilai beda Absis dan Ordinat I antar Titik Sekutu

Antara	Δx	Δy	ΔX	ΔY
AB	20765,42	-37480,28	30200	-30395,35
BC	-52661,95	-42849,505	-38983,57	-55589,545
CD	-28901,745	71681,015	-47351,475	61084,715
DE	37199,99	-28191,975	43480,605	-16979,971

Nilai Selisih Terbagi I

	Zx	Zy	Δx	Δy
$\Delta^{(1)}Z_1$	0,962073460	0,272733836		
$\Delta^{(1)}Z_2$	0,962158366	0,272711802	0,000084906	-0,000022034
$\Delta^{(1)}Z_3$	0,962111835	0,272663049	-0,000046531	-0,000048754
$\Delta^{(1)}Z_4$	0,962155187	0,272717384	0,000043352	0,000054335

Beda Absis dan Ordinat II Untuk Titik Sekutu dan Selisih terbagi II

Koordinat Titik Hasil Transformasi Koordinat Baru (TM-3⁰)

X	Y
27209,4798	144772,0024
9456,7557	112251,8235
51434,6032	74530,0888

Hasil perhitungan Transformasi Koordinat Metode Lauf menggunakan program Microsoft Excel. yaitu:

Hasil Perhitungan :

Koordinat Titik Hasil Transformasi Koordinat Baru (TM-3⁰)

X	Y
27209,4798072095	144772,0024450
9456,7557133829	112251,8235443800
51434,6032057314	74530,0888437104

Hasil perhitungan Transformasi Koordinat Metode Lauf Koreksi Sudut dan Jarak dengan data yang sama menggunakan program Visual Basic 6.0. yaitu:

KOREKSI SUDUT (ARGUMEN) DAN JARAK (MODULUS)

Jumlah Titik Sekutu : 4

Jumlah Titik Yang Ditransformasi : 5

Titik Sekutu Koordinat Lama (lokal) dan Koordinat Baru (TM-30) yaitu :

Titik	Koordinat Lama		Koordinat Baru	
	x	y	X	Y
A	-48.076,830	28.946,225	-614.308,950	42.228,565
B	9.610,750	60.421,075	-564.910,780	36.435,206
C	24.007,800	-21.853,520	-600.740,331	-44.978,367
D	-39.427,870	-31.853,750	-652.046,664	-6.276,488

Koordinat Titik yang akan ditransformasi, yaitu:

Titik	x	y
1	-41.614,288	5.028,108
2	-9.955,936	-21.782,363
3	13.707,882	17.795,000
4	-28.183,472	49.073,883
5	-20.508,720	22.901,756

Hasil Koordinat Titik Obyek hasil Koreksi Sudut dan Jarak

Titik	X	Y
1	-630898,866236403	21013,03241895
2	-623187,17728559	-18624,3028827652
3	-581757,26466488	-3466,47842401652
4	-594326,602442799	45365,4274153977
5	-604793,609546595	21421,1861253297

Hasil perhitungan dengan menggunakan Software Microsoft Excel

dengan menggunakan data yang sama yaitu:

Hasil Koordinat Titik Obyek dengan Metode Lauf Koreksi Sudut dan Jarak

Titik	X	Y
1	-630898,8662	21013,03242
2	-623187,1773	-18624,30288
3	-581757,2647	-3466,478424
4	-594326,6024	45365,42742
5	-604793,6095	21421,18613

BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Kesimpulan yang dapat diambil dari hasil penelitian ini adalah, sebagai berikut::

1. Program hitung yang dihasilkan dapat digunakan untuk:
 - a. Menghitung transformasi koordinat dari Sistem Koordinat Lokal menjadi Sistem Koordinat TM-3⁰ dengan menggunakan metode Affine
 - b. Menghitung transformasi koordinat dari Sistem Koordinat Lokal menjadi Sistem Koordinat TM-3⁰ dengan menggunakan metode Lauf Selisih Terbagi
 - c. Menghitung koreksi transformasi koordinat dari Sistem Koordinat Lokal menjadi Sistem Koordinat TM-3⁰ dengan menggunakan metode Lauf Koreksi Sudut dan Jarak
2. Untuk mengetahui perbedaan nilai hasil hitungan dengan menggunakan software lain (Microsoft Excel) maka dalam hal ini pengujiannya menggunakan beberapa koordinat titik sekutu dan koordinat titik yang akan ditransformasi, sehingga diperoleh rata-rata perbedaan nilai hasil hitungan sebagai berikut:
 - a. Bila menggunakan metode Affine dengan ketentuan 5(lima) titik sekutu dengan 2 (dua) titik yang akan ditransformasi, 4 (lima) sekutu dengan 4 (empat) titik yang akan ditransformasi diketahui hasil masing-masing koordinat hasil transformasi dengan

menggunakan Program Hitung serta hasil dengan Microsoft Excel yaitu:.

Koordinat Hasil Transformasi				Keterangan
Hasil Program Hitung		Hasil Microsoft Excel		
X	Y	X	Y	
233033925,293	692131962,550	233033925,249	692131962,550	5 (lima) titik sekutu dan 2 (dua) titik obyek
233376592,615	695332946,100	233376592,615	695332946,100	
97945.2617493	15086.3432458	97945.2617493	15086.3432458	4 (empat) titik sekutu dan 4 (empat) titik obyek
97811.4920235	15120.4139564	97811.4920235	15120.4139564	
97875.1006857	15218.2611851	97875.1006857	15218.2611851	
97989.5061501	15209.527809	97989.5061501	15209.527809	

Sehingga diperoleh rata-rata perbedaan nilai hasil hitungan sebagai berikut:

$$X = 0,0000002831 \text{ meter ; } Y = 0,0000006557 \text{ meter}$$

- b. Bila menggunakan Metode Lauf Selisih Terbagi rata-rata perbedaan nilainya dengan menggunakan 4 (empat) titik sekutu dengan 2 (dua) titik yang akan ditransformasi, 5 (lima) sekutu dengan 3 (tiga) titik yang akan ditransformasi diketahui hasil masing-masing koordinat hasil transformasi dengan menggunakan Program Hitung serta hasil dengan Microsoft Excel yaitu:

Koordinat Hasil Transformasi				Keterangan
Hasil Program Hitung		Hasil Microsoft Excel		
X	Y	X	Y	
-21603,266	-627426,4634	-21603,266	-627426,459	4 (empat) titik sekutu dan 2 (dua) titik obyek
-63367,8974	-699561,7739	-63367,8974	-699561,7767	
27209,4798	144772,0024	27209,4798	144772,0024	5 (lima) titik sekutu dan 3 (tiga) titik obyek
9456,7557	112251,8235	9456,7557	112251,8235	
51434,6032	74530,0888	51434,6032	74530,0888	

Sehingga diperoleh rata-rata perbedaan nilai hasil hitungan sebagai berikut:

$$X = 0,0000247010 \text{ meter ; } Y = 0,000357301 \text{ meter}$$

- c. Bila menggunakan Lauf Koreksi Sudut dan Jarak rata-rata perbedaan nilainya dengan menggunakan 4 (empat) titik sekutu dengan 5 (lima) titik yang akan ditransformasi, 4 (empat) sekutu dengan 1 (satu) titik yang akan ditransformasi diketahui hasil

masing-masing koordinat hasil transformasi dengan menggunakan Program Hitung serta hasil dengan Microsoft Excel yaitu:

Koordinat Hasil Transformasi				Keterangan
Hasil Program Hitung		Hasil Microsoft Excel		
X	Y	X	Y	
-630898,866236	21013,032419	-630898,866236	21013,032418	4 (empat) titik sekutu dan 5 (lima) titik obyek
-323187,177286	-18624,302883	-623187,177286	-18624,302882	
-581757,264665	-3466,478424	-581757,264665	-3466,478424	
-594326,602443	45385,427415	-594326,602443	45385,427415	
-604793,609547	21421,186125	-604793,609547	21421,186125	
83083,207582	-590707,083586	83083,207582	-590707,083586	4 (empat) titik sekutu dan 1 (satu) titik obyek

Sehingga diperoleh rata-rata perbedaan nilai hasil hitungan sebagai berikut:

$$X = 0,0000000001 \text{ meter ; } Y = 0,0000000001 \text{ meter}$$

3. Apabila ada perbedaan pada hasil hitungan (decimal nilai) dengan menggunakan program yang lain (EXCEL) adalah akibat dari perbedaan pembulatan dan alat hitung yang digunakan.

V. 2. Saran

1. Program Hitung Transformasi Koordinat Titik (Software) ini hanya digunakan untuk transformasi koordinat titik dalam satu zone, sehingga untuk selanjutnya Program Hitung Transformasi Koordinat Titik ini dapat dikembangkan untuk dapat melakukan transformasi koordinat titik antar zone.
2. Program Hitung Transformasi Koordinat Titik ini hanya menggunakan 2 (dua) metode yaitu Metode Affine dan Metode Lauf, sehingga untuk selanjutnya Software ini dapat dikembangkan dengan menambah metode transformasi yang lainnya

Daftar Pustaka

- Hendriatiningsih, Sri. (2002), *Transformasi Peta Sistem Lokal Ke Sistem Proyeksi TM-3⁰ Dengan Metode Numerik*, Staf Pengajar Jurusan Teknik Geodesi, Fakultas Teknik Sipil Dan Perencanaan, Institut Teknologi Bandung, Bandung.
- Kahar, Joenil. (2002), *Teknik Perataan Parameter*, Staf Pengajar Jurusan Teknik Geodesi, Fakultas Teknik Sipil Dan Perencanaan, Institut Teknologi Bandung, Bandung.
- Kahar, Joenil. (2002), *Penurunan Variansi Baku Hasil Perataan*, Staf Pengajar Jurusan Teknik Geodesi, Fakultas Teknik Sipil Dan Perencanaan, Institut Teknologi Bandung, Bandung.
- Perawiranagara, K. (2002), *Reduksi Jarak, Sudut dan Hitungan Koordinat Pada Proyeksi Mercator 3⁰*, Staf Deputi IV Direktorat Pengukuran dan Pemetaan Deputi Bidang Pengukuran dan Pendaftaran Tanah. Badan Pertanahan Nasional, Jakarta.
- Prihandito, Aryono (1988), *Proyeksi Peta*, Kanisius, Yogyakarta.
- Purwohardjo, Umaryono. (1994), *Sistem Dan Transformasi Koordinat*, ITB, Bandung.
- Purwohardjo, U., Hedriatiningsih S., Soedomo, A S., Hakim D M., Handoro S. (1997), *Petunjuk Penggunaan Sistem Proyeksi TM-3⁰*, Badan Pertanahan Nasional, Lembaga Pengabdian Kepada Masyarakat, ITB, Bandung.
- Razaq, A. (2003), *Pemrograman Microsoft Visual Basic 6.0*, Penerbit dan Percetakan Offset Indah, Surabaya.

Soedomo, Agoes S., Sudarman. (2004), *Sistem & Transformasi Koordinat, Penyelesaian dan Soal Latihan*, Departemen Teknik Geodesi, Fakultas Teknik Sipil dan Perencanaan, ITB, Bandung

Gjajhyadi, Edwin. (2002), *Buku Ajar Matematika*, Institut Teknologi Nasional, Malang.

Jmaryono P, S. Hedriatiningsih, Sudarman, Agoes Soedomo, D. Muhally H, Saptono H. (1997), *Petunjuk Penggunaan Sistem Proyeksi TM-3⁰*, Badan Pertanahan Nasional, Lembaga Pengabdian Kepada Masyarakat, ITB, Bandung.

Wolf, P. R. (1993), *Transformasi Koordinat, Element Fotogrametry*, Gajah Mada University Press, Yogyakarta.

LAMPIRAN



INSTITUT TEKNOLOGI NASIONAL MALANG
KAMPUS I : JL. BEND. SIGURA-GURA NO. 2
KAMPUS II: JL. RAYA KARANGLO, KM 2

PROJ MALANG
A MALANG

NAMA

: MUHAMMAD NAF' AN

NIM

: 98 . 25 . 051

JURUSAN

: TEKNIK GEODESI (S 1)

DOSEN PEMBIMBING

: Ir. RINTO SASONGKO, MT

Ir. LEO PANTIMENA, MSc

LEMBAR ASISTENSI
TUGAS AKHIR / SKRIPSI

No.	Tanggal	Keterangan	Paraf
1.	27-09-2004	Sempurnaan Redaksi bab I Buat Rencana Isi Skripsi selengkapnya (draft)	
2.	03-01-2005	Bab I Lanjutan bab II dan Rancangan program	
3.	15-01-2005	Bab II lanjutan bab III	
4.	12-03-2005	Lanjutan proses data dan uji program, Konsultasi ke P. Leo.	


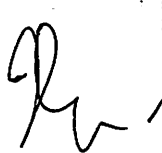



INSTITUT TEKNOLOGI NASIONAL MALANG
KAMPUS I : JL. BEND. SIGURA-GURA NO. 2
KAMPUS II: JL. RAYA KARANGLO, KM 2

ER) MALANG
A MALANG

NAMA : MUHAMMAD NAF' AN
NIM : 98 . 25 . 051
JURUSAN : TEKNIK GEODESI (S 1)
DOSEN PEMBIMBING : Ir. RINTO SASONGKO, MT
Ir. LEO PANTIMENA, MSc

LEMBAR ASISTENSI
TUGAS AKHIR / SKRIPSI

No.	Tanggal	Keterangan	Paraf
	7/05 /05	Perbaiki bab I & IV, V, dan IV, V, 	
	12/05 /05	All seminar hasil	

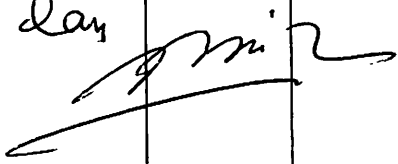


INSTITUT TEKNOLOGI NASIONAL MALANG
KAMPUS I : JL. BEND. SIGURA-GURA No.2
KAMPUS II: JL. RAYA KARANGLO, KM.2

BNI (PERSERO) MALANG
BANK NIAGA MALANG

NAMA : MUHAMMAD NAF'AN
N . I . M : 98 . 25 . 051
JURUSAN : TEKNIK GEODESI (S1)
DOSEN PENGUJI I (satu) : Ir. JASMANI, M.Kom.

LEMBAR REVISI TUGAS AKHIR

No	Tanggal	Keterangan	Paraf
1/-	14/10/05 7	Sudah di revisi dan Aee	




INSTITUT TEKNOLOGI NASIONAL MALANG
KAMPUS I : JL. BEND. SIGURA-GURA No.2
KAMPUS II: JL. RAYA KARANGLO, KM.2

BNI (PERSERO) MALANG
BANK NIAGA MALANG

NAMA : MUHAMMAD NAF'AN
N.I.M : 98 . 25 . 051
JURUSAN : TEKNIK GEODESI (S1)
DOSEN PENGUJI II (dua) : Ir. RINTO SASONGKO, MT

LEMBAR REVISI TUGAS AKHIR

No	Tanggal	Keterangan	Paraf
	16/07 '05	Revisi bab IV Revisi bab V	

Algoritma Program Proses Hitung dan Hasil Transformasi Koordinat Metode Affine

```
Option Explicit

Private Sub cmdPrint_Click()
    Ngeprint
End Sub

Private Sub Form_Load()
    isiData
End Sub

Sub isiData()
    txtPx = Px
    txtPy = Py
    txtWx = Wx.Simple
    txtWy = Wy.Simple

    flxX.rows = X.rows + 1
    flxF.rows = (F.rows / 2) + 1
    flxX.cols = X.cols + 1

    flxF.cols = F.cols + 2
    flxX.ColWidth(1) = 2000
    flxF.ColWidth(1) = 2000
    flxF.ColWidth(2) = 2000

    Dim i As Long, k As Long

    For i = 1 To flxX.rows - 1
        For k = 1 To flxX.cols - 1
            flxX.TextMatrix(i, k) = X.r2(i, k)
        Next
    Next

    ' For i = 1 To (flxF.rows - 1) / 2
    '     For k = 1 To flxF.cols - 1
    '         flxF.TextMatrix(i, k) = F.r2(i, k)
    '     Next
    ' Next

    For i = 1 To (flxF.rows - 1)
        flxF.TextMatrix(i, 1) = F.r2(i, 1)

    Next
    Dim z As Long
    z = 0
    For i = F.rows / 2 To F.rows
        flxF.TextMatrix(z, 2) = F.r2(i, 1)
        z = z + 1
    Next

    ' Pemberian nama kolom
```

```

For i = 1 To (flxX.cols - 1)
    flxX.TextMatrix(0, i) = i
Next

```

```

' Pemberian nama baris
For i = 1 To (flxX.rows - 1)
    flxX.TextMatrix(i, 0) = i
Next

```

```

' Pemberian nama kolom
'For i = 1 To (flxF.cols - 1)
    flxF.TextMatrix(0, 1) = "X"
    flxF.TextMatrix(0, 2) = "Y"
'Next

```

```

' Pemberian nama baris
For i = 1 To (flxF.rows - 1)
    flxF.TextMatrix(i, 0) = i
Next

```

End Sub

```

Sub Ngeprint()
    frmVsPrinter.VS.FontName = "Courier new"
    frmVsPrinter.VS.StartDoc
    frmVsPrinter.VS.TextAlign = taCenterMiddle
    frmVsPrinter.VS.FontSize = 14
    frmVsPrinter.VS.Paragraph = "Hasil Transformasi Local TM-3
Derajat"
    frmVsPrinter.VS.Paragraph = "Id Daerah: " & frmUtama.txtID
    frmVsPrinter.VS.Paragraph = "Nama Daerah: " &
frmUtama.txtNamaDarah
    frmVsPrinter.VS.Paragraph = ""
    frmVsPrinter.VS.FontSize = 12

    frmVsPrinter.VS.Paragraph = ""
    frmVsPrinter.VS.Paragraph = "Matrik A"

    Dim StringKolom As String
    Dim StringKolomJudul As String

    StringKolom = ""

    frmVsPrinter.VS.StartTable
    Dim i As Long

    For i = 1 To frmUtama.flxA.cols - 1
        StringKolom = StringKolom & "|>1300"
        StringKolomJudul = StringKolomJudul & "|Kolom " & i
    Next
    StringKolomJudul = "|a|b|c|d|C1|C2 "

    '    frmVsPrinter.VS.AddTable "100|1000|" & _
    '                                "1000|1000|>1000|>1000", "No|No
Rek|Keterangan|D/K" & _

```

```

        "|Debet|Kredit", ""
frmVsPrinter.VS.AddTable StringKolom, StringKolomJudul, ""

frmVsPrinter.VS.TableCell(tcCols) = frmUtama.flxA.cols

    For i = 0 To frmUtama.flxA.cols
        frmVsPrinter.VS.TableCell(tcColWidth, , i + 1)
= "1300" 'no
    Next

frmVsPrinter.VS.FontSize = 10

    Dim kolom As Long, baris As Long
    baris = 0
    frmVsPrinter.VS.TextAlign = taCenterMiddle

    Dim nomer As Long

    For i = 1 To frmUtama.flxA.rows - 1
        nomer = nomer + 1
        baris = baris + 1

        frmVsPrinter.VS.TableCell(tcRows) = baris

        Dim k As Long

        frmVsPrinter.VS.TableCell(tcText, baris,
1) = "Baris: " & nomer
        For k = 2 To frmUtama.flxA.cols
            frmVsPrinter.VS.TableCell(tcText,
baris, k) = frmUtama.flxA.TextMatrix(i, k - 1)
        Next

        seling baris

    Next

frmVsPrinter.VS.EndTable
'*****
frmVsPrinter.VS.Paragraph = ""
frmVsPrinter.VS.Paragraph = "Matrik F"

StringKolom = ""

frmVsPrinter.VS.StartTable

For i = 1 To frmUtama.flxB.cols - 1
    StringKolom = StringKolom & ">1300"
    StringKolomJudul = StringKolomJudul & "|Kolom " & i
Next
StringKolomJudul = "|Kolom 1 "

```

```

frmVsPrinter.VS.AddTable StringKolom, StringKolomJudul, ""
frmVsPrinter.VS.TableCell(tcCols) = frmUtama.flxB.cols
    For i = 0 To frmUtama.flxB.cols
        frmVsPrinter.VS.TableCell(tcColWidth, , i + 1)
= "1300" 'no
    Next

frmVsPrinter.VS.FontSize = 10

    baris = 0
    nomer = 0

    For i = 1 To frmUtama.flxB.rows - 1
        nomer = nomer + 1
        baris = baris + 1

        frmVsPrinter.VS.TableCell(tcRows) = baris

            frmVsPrinter.VS.TableCell(tcText, baris,
1) = "Baris: " & nomer
            For k = 2 To frmUtama.flxB.cols
                frmVsPrinter.VS.TableCell(tcText,
baris, k) = frmUtama.flxB.TextMatrix(i, k - 1)
            Next

            seling baris

    Next

frmVsPrinter.VS.EndTable

'((((((((((((((((((((((((((((

frmVsPrinter.VS.Paragraph = ""
frmVsPrinter.VS.Paragraph = "Matrik A' "

StringKolom = ""

frmVsPrinter.VS.StartTable

For i = 1 To frmUtama.flxA2.cols - 1
    StringKolom = StringKolom & ">1300"
    StringKolomJudul = StringKolomJudul & "|Kolom " & i
Next

```



```

StringKolom = StringKolom & "|>2300"
StringKolomJudul = StringKolomJudul & "|Kolom " & i
Next
StringKolomJudul = "|X|Y "
frmVsPrinter.VS.AddTable StringKolom, StringKolomJudul, ""

frmVsPrinter.VS.TableCell(tcColWidth, , 1) = "1300" 'no
frmVsPrinter.VS.TableCell(tcCols) = flxF.cols

    For i = 1 To flxF.cols
        frmVsPrinter.VS.TableCell(tcColWidth, , i + 1)
= "2300" 'no
    Next

frmVsPrinter.VS.FontSize = 10

    baris = 0
    nomer = 0

    For i = 1 To flxF.rows - 1
        nomer = nomer + 1
        baris = baris + 1

        frmVsPrinter.VS.TableCell(tcRows) = baris

        frmVsPrinter.VS.TableCell(tcText, baris,
1) = "Baris: " & nomer
        For k = 2 To flxF.cols
            frmVsPrinter.VS.TableCell(tcText,
baris, k) = Format(flxF.TextMatrix(i, k - 1), "#####.#####0")
        Next

        seling baris

    Next

frmVsPrinter.VS.EndTable

frmVsPrinter.VS.TextAlign = taLeftMiddle
frmVsPrinter.VS.Paragraph = ""

frmVsPrinter.VS.Paragraph = "Nilai Px: " & txtPx
frmVsPrinter.VS.Paragraph = "Nilai Py: " & txtPy
frmVsPrinter.VS.Paragraph = "Nilai Wx: " & txtWx
frmVsPrinter.VS.Paragraph = "Nilai Wy: " & txtWy

```

```
frmVsPrinter.VS.EndDoc
frmVsPrinter.VS.TextAlign = taLeftMiddle
Dim z As Integer
For z = 1 To frmVsPrinter.VS.PageCount
    frmVsPrinter.VS.StartOverlay z
    frmVsPrinter.VS.CurrentX = frmVsPrinter.VS.MarginLeft
    frmVsPrinter.VS.CurrentY = frmVsPrinter.VS.MarginTop -
300    frmVsPrinter.VS.Paragraph = "Halaman " & z & " dari "
& frmVsPrinter.VS.PageCount
    frmVsPrinter.VS.EndOverlay
Next

frmVsPrinter.Show 1

End Sub

Private Sub Form_Resize()
    Move 0, 0
End Sub
```

Algoritma Program Transformasi Koordinat Metode Lauf

```
Option Explicit

Private Sub affine_Click()
    frmUtama.Show 1
End Sub

Private Sub cmdBentukMatrikA_Click()
    If Trim(txtId) = "" Then
        MsgBox "Id Daerah Harus Diisi", vbInformation
        txtId.SetFocus
        Exit Sub
    End If
    isiSetting
    BuatMatrikA
    BuatMatrikTM3
    'BuatMatrikB
    'BuatMatrikBobot
    'BuatMatrikA2
    'IsiMatrikA2
End Sub

Private Sub cmdHitung_Click()
    If Trim(txtId) = "" Then
        MsgBox "Id Daerah Harus Diisi", vbInformation
        txtId.SetFocus
        Exit Sub
    End If
    cmdBentukMatrikA_Click
    cmdSetInput_Click
    HitungLauf
End Sub

Private Sub cmdSetInput_Click()
    BuatMatrikCalon
    'MsgBox flxCalon.TextMatrix(1, 1) * 10
End Sub

Private Sub dataDaerah_Click()
    frmInputData2.Show 1
End Sub

Private Sub Form_Load()

    AwalCalon
End Sub

Private Sub inputDaerah_Click()
    frmDataDaerah2.Show 1
End Sub
```

```

Private Sub koreksi_Click()
    If Trim(txtId) = "" Then
        MsgBox "Id Daerah Harus Diisi", vbInformation
        txtId.SetFocus
    Exit Sub
End If

frmTes.BentukFleksiAhelmet
frmTes.BentukMatrikFhelmet
frmTes.BentukFleksiA2helmet
frmTes.IsiMatrikAhelmet
frmTes.IsiMatrikFhelmet
frmTes.IsiMatrikA2helmet
frmTes.HitungHelmert
frmTes.BentukFleksiX
frmTes.BentukFleksiZDhelmet
frmTes.BentukFleksiSudut
frmTes.BentukFleksiModulus
frmTes.BentukFleksiArgumen
frmTes.BentukFleksiAkhir

frmKoreksi.Show 1
End Sub

Private Sub selisihTerbagi_Click()
    cmdHitung_Click
End Sub

Private Sub tentangProgram_Click()
    frmRtf.Show 1
End Sub

Private Sub txtId_KeyDown(KeyCode As Integer, Shift As
Integer)
    If KeyCode = vbKeyReturn Then
        kataSql = "SELECT * FROM dataDaerah2 ORDER BY id"

        If Trim(txtId) <> "" Then
            KataPotong = Trim(txtId) & "*"
        Else
            KataPotong = ""
        End If

        Kolom1 = ""
        Kolom2 = ""
        frmListView.Label1.Caption = "Daftar Daerah"
        frmListView.ListView1.ColumnHeaders(1).Text = "Id
Daerah"
        frmListView.ListView1.ColumnHeaders(2).Text = "Nama
Daerah"
        frmListView.Show 1
        txtId = Kolom1
        txtNamaDarah = Kolom2
        cmdBentukMatrikA_Click
        cmdSetInput_Click
    End If
End Sub

```

```

Exit Sub
End If
End Sub

Sub BuatMatrikA()
Dim i As Long, k As Long
flxA.cols = 3
flxA.rows = JumlahBarisTitikSekutu + 1

' Pemberian nama kolom
' For i = 1 To (flxA.cols - 1)
'   flxA.TextMatrix(0, i) = i
' Next

flxA.TextMatrix(0, 1) = "x"
flxA.TextMatrix(0, 2) = "y"

' Pemberian nama baris
For i = 1 To (flxA.rows - 1)
    flxA.TextMatrix(i, 0) = i
Next

' mengenkodkan semua
' kolom 6 sudah fix
For i = 1 To 2
    For k = 1 To JumlahBarisTitikSekutu
        flxA.TextMatrix(k, i) = 0
    Next
Next

' memberikan nilai satu
'
' For i = 1 To JumlahBarisTitikSekutu
'   flxA.TextMatrix(i, 5) = 1
' Next
'
'           For i = JumlahBarisTitikSekutu + 1 To
(JumlahBarisTitikSekutu * 2)
'   flxA.TextMatrix(i, 6) = 1
' Next

' pengisian Data dari database
kataSql = "SELECT * FROM titikSekutu2 WHERE id=" & txtId
TutupRs
rs.Open kataSql, koneksi, adOpenStatic

Dim ketemu As Long, ketemu2 As Long
Dim baris As Long, kolom As Long, data As Double

Do While rs.EOF = False
    'ketemu = InStr(rs.Fields("data"), "|")
    kolom = rs.Fields("kolom")
    'MsgBox baris

    'ketemu2 = InStr(rs.Fields("data"), "?")

```

```

    baris = rs.Fields("baris")
    'MsgBox kolom

    data = rs.Fields("data")
    'MsgBox data

    flxA.TextMatrix(baris, kolom) = data
    rs.MoveNext
Loop
End Sub

Sub BuatMatrikTM3()
Dim i As Long, k As Long
flxTM3.cols = 3
flxTM3.rows = JumlahBarisTitikSekutu + 1

' Pemberian nama kolom
'For i = 1 To (flxTM3.cols - 1)
'    flxTM3.TextMatrix(0, i) = i
'Next

flxTM3.TextMatrix(0, 1) = "X"
flxTM3.TextMatrix(0, 2) = "Y"

' Pemberian nama baris
For i = 1 To (flxTM3.rows - 1)
    flxTM3.TextMatrix(i, 0) = i
Next

' mengenkodkan semua
' kolom 6 sudah fix
For i = 1 To 2
    For k = 1 To JumlahBarisTitikSekutu
        flxTM3.TextMatrix(k, i) = 0
    Next
Next

' pengisian Data dari database
kataSql = "SELECT * FROM TM3 WHERE id=" & txtId
TutupRs
rs.Open kataSql, koneksi, adOpenStatic

Dim ketemu As Long, ketemu2 As Long
Dim baris As Long, kolom As Long, data As Double

Do While rs.EOF = False
    kolom = rs.Fields("kolom")
    baris = rs.Fields("baris")
    data = rs.Fields("data")
    flxTM3.TextMatrix(baris, kolom) = data
    rs.MoveNext
Loop
End Sub

```

```

Sub isiSetting()

    kataSql = "SELECT
namaDaerah,DimensiTitikSekutuBaris,DimensiTitikSekutuKolom," &
    "DimensiTMbaris,DimensiTMkolom FROM " & _
    "dataDaerah2 WHERE id=" & txtId
    TutupRs
    rs.Open kataSql, koneksi, adOpenStatic

    JumlahBarisTitikSekutu =
rs.Fields("DimensiTitikSekutuBaris")
    JumlahKolomTitikSekutu =
rs.Fields("DimensiTitikSekutuKolom")

    JumlahBarisTM = rs.Fields("DimensiTMBaris")
    JumlahKolomTM = rs.Fields("DimensiTMKolom")

    MsgBox JumlahBarisTitikSekutu & " " &
JumlahKolomTitikSekutu

End Sub

Sub HitungLauf()

    Dim i As Long, k As Long, z As Long

    ReDim BedaAbsisSatu(flxA.rows - 3, (flxA.cols - 1) * 2)
    ReDim SelisihSatu(flxA.rows - 3, (flxA.cols - 1) * 2)

    For i = 1 To flxTM3.rows - 2 Step 1
        BedaAbsisSatu(z, 0) = flxA.TextMatrix(i + 1, 1) -
flxA.TextMatrix(i, 1)
        BedaAbsisSatu(z, 1) = flxA.TextMatrix(i + 1, 2) -
flxA.TextMatrix(i, 2)
        BedaAbsisSatu(z, 2) = flxTM3.TextMatrix(i + 1, 1) -
flxTM3.TextMatrix(i, 1)
        BedaAbsisSatu(z, 3) = flxTM3.TextMatrix(i + 1, 2) -
flxTM3.TextMatrix(i, 2)
        z = z + 1
    Next
    IsiBedaAbsisSatu
    IsiSelisihSatu
    IsiBedaAbsisDua
    IsiSelisihDua
    IsiBedaAbsisTiga
    IsiSelisihTiga
    HitungTransformasi
    HitungMatrikKoreksi

    frmTes.Show 1

End Sub

Sub IsiBedaAbsisSatu()

```

```
Dim i As Long, k As Long, z As Long
```

```
frmTes.flxBedaAbsisSatu.cols = 5
```

```
frmTes.flxBedaAbsisSatu.rows = UBound(BedaAbsisSatu) + 2
```

```
For i = 0 To 4
```

```
    frmTes.flxBedaAbsisSatu.ColWidth(i) = 1300
```

```
Next
```

```
' pemberian nomor
```

```
For i = 1 To frmTes.flxBedaAbsisSatu.rows - 1
```

```
    frmTes.flxBedaAbsisSatu.TextMatrix(i, 0) = i
```

```
Next
```

```
'For k = 1 To frmTes.flxBedaAbsisSatu.cols - 1
```

```
'    frmTes.flxBedaAbsisSatu.TextMatrix(0, k) = k
```

```
'Next
```

```
frmTes.flxBedaAbsisSatu.TextMatrix(0, 1) = "x"
```

```
frmTes.flxBedaAbsisSatu.TextMatrix(0, 2) = "y"
```

```
frmTes.flxBedaAbsisSatu.TextMatrix(0, 3) = "X"
```

```
frmTes.flxBedaAbsisSatu.TextMatrix(0, 4) = "Y"
```

```
For i = 0 To UBound(BedaAbsisSatu)
```

```
    For k = 0 To 3
```

```
        frmTes.flxBedaAbsisSatu.TextMatrix(i + 1, k + 1) =
```

```
Round(BedaAbsisSatu(i, k), 4)
```

```
    Next
```

```
Next
```

```
End Sub
```

```
Sub IsiSelisihSatu()
```

```
Dim i As Long, k As Long
```

```
frmTes.flxSelisihSatu.cols = 5
```

```
frmTes.flxSelisihSatu.rows = UBound(SelisihSatu) + 2
```

```
' Pengisian Data Zx dan Zy
```

```
For i = 0 To UBound(SelisihSatu)
```

```
    SelisihSatu(i, 0) = Round((BedaAbsisSatu(i, 0) *  
BedaAbsisSatu(i, 2) + BedaAbsisSatu(i, 1) * BedaAbsisSatu(i,  
3)) / _  
        ((BedaAbsisSatu(i, 0) ^ 2) + (BedaAbsisSatu(i, 1)  
^ 2)), 9)
```

```
    SelisihSatu(i, 1) = Round((BedaAbsisSatu(i, 3) *  
BedaAbsisSatu(i, 0) - BedaAbsisSatu(i, 2) * BedaAbsisSatu(i,  
1)) / _  
        ((BedaAbsisSatu(i, 0) ^ 2) + (BedaAbsisSatu(i, 1)  
^ 2)), 9)
```

```
Next
```

```
' Pengisian deltax dan deltay
```

```
For i = 1 To (UBound(SelisihSatu))
```



```

        SelisihSatu(i, 2) = Round(SelisihSatu(i, 0), 9) -
Round(SelisihSatu(i - 1, 0), 9)
        SelisihSatu(i, 3) = Round(SelisihSatu(i, 1), 9) -
Round(SelisihSatu(i - 1, 1), 9)

Next

' pengaturan lebar sell
For i = 0 To 4
    frmTes.flxSelisihSatu.ColWidth(i) = 1300
Next

' pemberian nomor
For i = 1 To frmTes.flxSelisihSatu.rows - 1
    frmTes.flxSelisihSatu.TextMatrix(i, 0) = i
Next
frmTes.flxSelisihSatu.TextMatrix(0, 1) = "Zx"
frmTes.flxSelisihSatu.TextMatrix(0, 2) = "Zy"
frmTes.flxSelisihSatu.TextMatrix(0, 3) = "Delta x"
frmTes.flxSelisihSatu.TextMatrix(0, 4) = "Delta y"

'For k = 1 To frmTes.flxSelisihSatu.cols - 1
'    frmTes.flxSelisihSatu.TextMatrix(0, k) = k
'Next

For i = 0 To UBound(SelisihSatu)
    For k = 0 To 3
        frmTes.flxSelisihSatu.TextMatrix(i + 1, k + 1) =
Round(SelisihSatu(i, k), 10)
    Next
Next

End Sub

Sub IsiBedaAbsisDua()
    Dim i As Long, k As Long, z As Long

    'ReDim BedaAbsisDua((flxA.rows - 1) / 2, 3)
    ReDim BedaAbsisDua(flxA.rows - 1 - 2 - 1, 3)

    'textmatrik dimulai dari 0

    For i = 1 To frmTes.flxBedaAbsisSatu.rows - 1 - 1
'((flxA.rows - 1) / 2) ' berdasarkan tabel
        BedaAbsisDua(i - 1, 0) = flxA.TextMatrix(i + 2, 1) -
flxA.TextMatrix(i, 1)
        BedaAbsisDua(i - 1, 1) = flxA.TextMatrix(i + 2, 2) -
flxA.TextMatrix(i, 2)
    Next

    For i = 0 To UBound(BedaAbsisDua)
        BedaAbsisDua(i, 2) =
frmTes.flxSelisihSatu.TextMatrix(i + 2, 3)
    Next

```

```

        BedaAbsisDua(i, 3) =
frmTes.flxSelisihSatu.TextMatrix(i + 2, 4)
Next

    frmTes.flxBedaDua.cols = 5
    frmTes.flxBedaDua.rows = frmLauf.flxA.rows - 2
' ((flxA.rows - 1) / 2) + 1

    For i = 0 To frmTes.flxBedaDua.cols - 1
        frmTes.flxBedaDua.ColWidth(i) = 1300
    Next

    ' pemberian nomor
    For i = 1 To frmTes.flxBedaDua.rows - 1
        frmTes.flxBedaDua.TextMatrix(i, 0) = i
    Next

    'For k = 1 To frmTes.flxBedaDua.cols - 1
    '    frmTes.flxBedaDua.TextMatrix(0, k) = k
    'Next

    frmTes.flxBedaDua.TextMatrix(0, 1) = "x"
    frmTes.flxBedaDua.TextMatrix(0, 2) = "y"
    frmTes.flxBedaDua.TextMatrix(0, 3) = "X"
    frmTes.flxBedaDua.TextMatrix(0, 4) = "Y"

    ' pengisian data ke flexi flxBedaDua
    For i = 0 To UBound(BedaAbsisDua)
        For k = 0 To 3
            frmTes.flxBedaDua.TextMatrix(i + 1, k + 1) =
Round(BedaAbsisDua(i, k), 10)
        Next

    Next
End Sub

Sub IsiSelisihDua()
    Dim i As Long, k As Long

    ReDim SelisihDua(flxA.rows - 1 - 2 - 1, 3)

    frmTes.flxSelisihDua.cols = 5
    frmTes.flxSelisihDua.rows = frmLauf.flxA.rows - 2

    For i = 0 To UBound(SelisihDua)
        SelisihDua(i, 0) = ((frmTes.flxBedaDua.TextMatrix(i +
1, 1) * frmTes.flxBedaDua.TextMatrix(i + 1, 3)) +
        (frmTes.flxBedaDua.TextMatrix(i + 1, 2) *
frmTes.flxBedaDua.TextMatrix(i + 1, 4))) /
        ((frmTes.flxBedaDua.TextMatrix(i + 1, 1) ^ 2) +
(frmTes.flxBedaDua.TextMatrix(i + 1, 2) ^ 2))

        SelisihDua(i, 1) = ((frmTes.flxBedaDua.TextMatrix(i +
1, 1) * frmTes.flxBedaDua.TextMatrix(i + 1, 4)) -

```

```

        (frmTes.flxBedaDua.TextMatrix(i + 1, 2) *
frmTes.flxBedaDua.TextMatrix(i + 1, 3))) /
        ((frmTes.flxBedaDua.TextMatrix(i + 1, 1) ^ 2) +
(frmTes.flxBedaDua.TextMatrix(i + 1, 2) ^ 2))

Next

For i = 1 To (UBound(SelisihDua))

    SelisihDua(i, 2) = Round(SelisihDua(i, 0), 15) -
Round(SelisihDua(i - 1, 0), 15)
    SelisihDua(i, 3) = Round(SelisihDua(i, 1), 15) -
Round(SelisihDua(i - 1, 1), 15)

Next

' pengaturan lebar sell
For i = 1 To 4
    frmTes.flxSelisihDua.ColWidth(i) = 1400
Next

' pemberian nomor
For i = 1 To frmTes.flxSelisihDua.rows - 1
    frmTes.flxSelisihDua.TextMatrix(i, 0) = i
Next

'For k = 1 To frmTes.flxSelisihDua.cols - 1
'    frmTes.flxSelisihDua.TextMatrix(0, k) = k
'Next
frmTes.flxSelisihDua.TextMatrix(0, 1) = "Zx"
frmTes.flxSelisihDua.TextMatrix(0, 2) = "Zy"
frmTes.flxSelisihDua.TextMatrix(0, 3) = "Delta x"
frmTes.flxSelisihDua.TextMatrix(0, 4) = "Delta y"

For i = 0 To UBound(SelisihDua)
    For k = 0 To 3
        frmTes.flxSelisihDua.TextMatrix(i + 1, k + 1) =
Round(SelisihDua(i, k), 15)
    Next
Next

End Sub
Sub IsiBedaAbsisTiga()
Dim i As Long, k As Long, z As Long

ReDim BedaAbsisTiga(flxA.rows - 1 - 3 - 1, 3)

'textmatrik dimulai dari 0

For i = 0 To UBound(BedaAbsisTiga)

```

```

        BedaAbsisTiga(i, 0) = flxA.TextMatrix(i + 3 + 1, 1) -
flxA.TextMatrix(i + 1, 1)
        BedaAbsisTiga(i, 1) = flxA.TextMatrix(i + 3 + 1, 2) -
flxA.TextMatrix(i + 1, 2)
    Next

    For i = 0 To UBound(BedaAbsisTiga)
        BedaAbsisTiga(i, 2) =
frmTes.flxSelisihDua.TextMatrix(i + 2, 3)
        BedaAbsisTiga(i, 3) =
frmTes.flxSelisihDua.TextMatrix(i + 2, 4)
    Next

    frmTes.flxBedaTiga.cols = 5
    frmTes.flxBedaTiga.rows = frmLauf.flxA.rows - 3
' ((flxA.rows - 1) / 2) + 1

    For i = 0 To frmTes.flxBedaTiga.cols - 1
        frmTes.flxBedaTiga.ColWidth(i) = 1300
    Next

    ' pemberian nomor
    For i = 1 To frmTes.flxBedaTiga.rows - 1
        frmTes.flxBedaTiga.TextMatrix(i, 0) = i
    Next

    'For k = 1 To frmTes.flxBedaTiga.cols - 1
    '    frmTes.flxBedaTiga.TextMatrix(0, k) = k
    'Next
    frmTes.flxBedaTiga.TextMatrix(0, 1) = "x"
    frmTes.flxBedaTiga.TextMatrix(0, 2) = "y"
    frmTes.flxBedaTiga.TextMatrix(0, 3) = "X"
    frmTes.flxBedaTiga.TextMatrix(0, 4) = "Y"

    ' pengisian data ke flexi flxBedaDua
    For i = 0 To UBound(BedaAbsisTiga)
        For k = 0 To 3
            frmTes.flxBedaTiga.TextMatrix(i + 1, k + 1) =
Round(BedaAbsisTiga(i, k), 15)
        Next
    Next

End Sub

Sub IsiSelisihTiga()

    Dim i As Long, k As Long

    ReDim SelisihTiga(flxA.rows - 1 - 3 - 1, 3)

    frmTes.flxSelisihTiga.cols = 5
    frmTes.flxSelisihTiga.rows = frmLauf.flxA.rows - 3

    For i = 0 To UBound(SelisihTiga)

```

```

SelisihTiga(i, 0) = ((frmTes.flxBedaTiga.TextMatrix(i
+ 1, 1) * frmTes.flxBedaTiga.TextMatrix(i + 1, 3)) +
(frmTes.flxBedaTiga.TextMatrix(i + 1, 2) *
frmTes.flxBedaAbsisSatu.TextMatrix(i + 1, 4))) /
((frmTes.flxBedaTiga.TextMatrix(i + 1, 1) ^ 2) +
(frmTes.flxBedaTiga.TextMatrix(i + 1, 2) ^ 2))

```

```

SelisihTiga(i, 1) = ((frmTes.flxBedaTiga.TextMatrix(i
+ 1, 1) * frmTes.flxBedaTiga.TextMatrix(i + 1, 4)) -
(frmTes.flxBedaTiga.TextMatrix(i + 1, 2) *
frmTes.flxBedaTiga.TextMatrix(i + 1, 3))) /
((frmTes.flxBedaTiga.TextMatrix(i + 1, 1) ^ 2) +
(frmTes.flxBedaTiga.TextMatrix(i + 1, 2) ^ 2))

```

```
Next
```

```

' For i = 1 To (UBound(SelisihDua))
' SelisihDua(i, 2) = frmTes.flxSelisihDua.TextMatrix(i
+ 1, 1) - frmTes.flxBedaDua.TextMatrix(i + 1 - 1, 1)
' SelisihDua(i, 3) = frmTes.flxBedaDua.TextMatrix(i +
1, 2) - frmTes.flxBedaDua.TextMatrix(i + 1 - 1, 2)
' Next

```

```
For i = 1 To (UBound(SelisihTiga))
```

```

SelisihTiga(i, 2) = Round(SelisihTiga(i, 0), 15) -
Round(SelisihTiga(i - 1, 0), 15)
SelisihTiga(i, 3) = Round(SelisihTiga(i, 1), 15) -
Round(SelisihTiga(i - 1, 1), 15)

```

```
Next
```

```
' pengaturan lebar sell
```

```

For i = 1 To 4
frmTes.flxSelisihTiga.ColWidth(i) = 1400
Next

```

```
' pemberian nomor
```

```

For i = 1 To frmTes.flxSelisihTiga.rows - 1
frmTes.flxSelisihTiga.TextMatrix(i, 0) = i
Next

```

```
'For k = 1 To frmTes.flxSelisihTiga.cols - 1
```

```

' frmTes.flxSelisihTiga.TextMatrix(0, k) = k
'Next

```

```
frmTes.flxSelisihTiga.TextMatrix(0, 1) = "Zx"
```

```
frmTes.flxSelisihTiga.TextMatrix(0, 2) = "Zy"
```

```
frmTes.flxSelisihTiga.TextMatrix(0, 3) = "Delta x"
```

```
frmTes.flxSelisihTiga.TextMatrix(0, 4) = "Delta y"
```

```
For i = 0 To UBound(SelisihTiga)
```

```

    For k = 0 To 3
        frmTes.flxSelisihTiga.TextMatrix(i + 1, k + 1) =
Round(SelisihTiga(i, k), 20)
    Next
Next
End Sub

Sub HitungTransformasi()
Dim i As Long, k As Long
ReDim HasilTrans(flxCalon.rows - 2, 2)

Dim Zax As Double
Dim Zay As Double
Dim suku2 As Double

For i = 0 To flxCalon.rows - 2
    ' mencari X terlebih dahulu
    Zax = flxTM3.TextMatrix(1, 1)

    'suku2 = flxCalon.TextMatrix(i + 1, 1) *
frmTes.flxSelisihSatu.TextMatrix(i + 1, 1) -
    ' flxCalon.TextMatrix(i + 1, 2) *
frmTes.flxSelisihSatu.TextMatrix(i + 1, 2)
    ' suku2 = suku2 - (flxA.TextMatrix(i + 1, 1) *
frmTes.flxSelisihSatu.TextMatrix(i + 1, 1) -
    ' flxA.TextMatrix(i + 1, 2) *
frmTes.flxSelisihSatu.TextMatrix(i + 1, 2))

    suku2 = flxCalon.TextMatrix(i + 1, 1) *
frmTes.flxSelisihSatu.TextMatrix(1, 1) -
    flxCalon.TextMatrix(i + 1, 2) *
frmTes.flxSelisihSatu.TextMatrix(1, 2)
    suku2 = suku2 - (flxA.TextMatrix(1, 1) *
frmTes.flxSelisihSatu.TextMatrix(1, 1) -
    flxA.TextMatrix(1, 2) *
frmTes.flxSelisihSatu.TextMatrix(1, 2))

    HasilTrans(i, 0) = Zax + suku2

    Zay = flxTM3.TextMatrix(1, 2)
    ' suku2 = frmTes.flxSelisihSatu.TextMatrix(i + 1, 1) *
flxCalon.TextMatrix(i + 1, 2) +
    ' flxCalon.TextMatrix(i + 1, 1) *
frmTes.flxSelisihSatu.TextMatrix(i + 1, 2)
    ' suku2 = suku2 - (frmTes.flxSelisihSatu.TextMatrix(i +
1, 1) * flxA.TextMatrix(i + 1, 2) +
    ' flxA.TextMatrix(i + 1, 1) *
frmTes.flxSelisihSatu.TextMatrix(i + 1, 2))

    suku2 = frmTes.flxSelisihSatu.TextMatrix(1, 1) *
flxCalon.TextMatrix(i + 1, 2) +
    flxCalon.TextMatrix(i + 1, 1) *
frmTes.flxSelisihSatu.TextMatrix(1, 2)
    suku2 = suku2 - (frmTes.flxSelisihSatu.TextMatrix(1,
1) * flxA.TextMatrix(1, 2) +

```

```

        flxA.TextMatrix(1,
frmTes.flxSelisihSatu.TextMatrix(1, 2))
        HasilTrans(i, 1) = Zay + suku2
1)
*
Next

frmTes.flxHasil.rows = flxCalon.rows
frmTes.flxHasil.cols = flxCalon.cols

' Pemberian nama kolom
'For i = 1 To (frmTes.flxHasil.cols - 1)
'    frmTes.flxHasil.TextMatrix(0, i) = i
'Next
frmTes.flxHasil.TextMatrix(0, 1) = "X"
frmTes.flxHasil.TextMatrix(0, 2) = "Y"

' Pemberian nama baris
For i = 1 To (frmTes.flxHasil.rows - 1)
    frmTes.flxHasil.TextMatrix(i, 0) = i
Next

    For i = 1 To flxCalon.cols - 1
        frmTes.flxHasil.ColWidth(i) = 1600
    Next

For i = 0 To UBound(HasilTrans)
    frmTes.flxHasil.TextMatrix(i + 1, 1) = HasilTrans(i,
0)
    frmTes.flxHasil.TextMatrix(i + 1, 2) = HasilTrans(i,
1)
Next
End Sub

Sub BuatMatrikCalon()

Dim i As Long, k As Long
flxCalon.cols = 3
flxCalon.rows = txtBanyakDaerah + 1

' Pemberian nama kolom
'For i = 1 To (flxCalon.cols - 1)
'    flxCalon.TextMatrix(0, i) = i
'Next

flxCalon.TextMatrix(0, 1) = "x"
flxCalon.TextMatrix(0, 2) = "y"

' Pemberian nama baris
For i = 1 To (flxCalon.rows - 1)
    flxCalon.TextMatrix(i, 0) = i
Next

' mengenkodkan semua

```

```

' For i = 1 To 2
'     For k = 1 To txtBanyakDaerah
'         flxCalon.TextMatrix(k, i) = 0
'     Next
' Next
For i = 1 To flxCalon.cols - 1
    flxCalon.ColWidth(i) = 1600
Next

```

End Sub

Sub AwalCalon()

```

' flxCalon.TextMatrix(1, 1) = 966.737
' flxCalon.TextMatrix(1, 2) = 1678.967
' flxCalon.TextMatrix(2, 1) = -40719.525
' flxCalon.TextMatrix(2, 2) = -70521

```

```

flxCalon.TextMatrix(1, 1) = -41614.288
flxCalon.TextMatrix(1, 2) = 5028.108
flxCalon.TextMatrix(2, 1) = -9955.936
flxCalon.TextMatrix(2, 2) = -21782.363

```

```

flxCalon.TextMatrix(3, 1) = 13707.882
flxCalon.TextMatrix(3, 2) = 17795

```

```

flxCalon.TextMatrix(4, 1) = -28183.472
flxCalon.TextMatrix(4, 2) = 49073.883

```

```

flxCalon.TextMatrix(5, 1) = -20508.72
flxCalon.TextMatrix(5, 2) = 22901.756

```

End Sub

Sub HitungMatrikKoreksi()

```

frmTes.flxKoreksi.rows = (flxA.rows - 2) * (flxCalon.rows
- 1) + 1
frmTes.flxKoreksi.cols = 5

```

```

frmTes.flxKoreksi.TextMatrix(0, 1) = "Zx"
frmTes.flxKoreksi.TextMatrix(0, 2) = "Zy"
frmTes.flxKoreksi.TextMatrix(0, 3) = "Delta x"
frmTes.flxKoreksi.TextMatrix(0, 4) = "Delta y"

```

Dim i As Long, k As Long, z As Long

```

'For i = 1 To frmTes.flxKoreksi.rows Step (flxA.rows - 1 -
1)
For i = 1 To flxCalon.rows - 1
    For k = 1 To flxA.rows - 1 - 1
        frmTes.flxKoreksi.TextMatrix(z + 1, 1) =
flxCalon.TextMatrix(i, 1) - flxA.TextMatrix(k, 1)
        frmTes.flxKoreksi.TextMatrix(z + 1, 2) =
flxCalon.TextMatrix(i, 2) - flxA.TextMatrix(k, 2)
        z = z + 1
    Next
Next

```


Algoritma Program Proses Hitung dan Hasil Transformasi Koordinat Metode Lauf Selisih Terbagi

Option Explicit

```
Private Sub cmdKoreksi_Click()  
    BentukFleksiAhelmet  
    BentukMatrikFhelmet  
    BentukFleksiA2helmet  
    IsiMatrikAhelmet  
    IsiMatrikFhelmet  
    IsiMatrikA2helmet  
    HitungHelmert  
    BentukFleksiX  
    BentukFleksiZDhelmet  
    BentukFleksiSudut  
    BentukFleksiModulus  
    BentukFleksiArgumen  
    BentukFleksiAkhir  
  
    frmKoreksi.Show 1  
End Sub  
  
Private Sub cmdPrint_Click()  
    Ngeprint  
End Sub  
  
Private Sub Form_Resize()  
    Move 0, 0  
End Sub  
  
Sub BentukFleksiAhelmet()  
  
    Dim i As Long, k As Long, z As Long  
  
    ' mengeset ukuran Flexi  
    frmKoreksi.flxAhelmet.rows = (2 * (frmLauf.flxA.rows - 1))  
- 1  
    frmKoreksi.flxAhelmet.cols = 5  
  
    For i = 1 To frmKoreksi.flxAhelmet.rows - 1  
        frmKoreksi.flxAhelmet.TextMatrix(i, 0) = i  
    Next  
    ' mengeset semua  
  
    For i = 1 To frmKoreksi.flxAhelmet.rows - 1  
        For k = 1 To 4  
            frmKoreksi.flxAhelmet.TextMatrix(i, k) = 0  
        Next  
    Next  
  
    ' pengisian kolom satu bagian pertama  
    For i = 1 To frmLauf.flxA.rows - 1 - 1
```

```

        frmKoreksi.flxAhelmet.TextMatrix(i,          1)          =
frmLauf.flxA.TextMatrix(i, 1)
        frmKoreksi.flxAhelmet.TextMatrix(i, 3) = 1
Next

' pengisian kolom satu bagian dua
z = 1
For k = i To 2 * (frmLauf.flxA.rows - 1 - 1)
    frmKoreksi.flxAhelmet.TextMatrix(k,          1)          =
frmLauf.flxA.TextMatrix(z, 2)
    frmKoreksi.flxAhelmet.TextMatrix(k, 4) = 1
    z = z + 1
Next

' pengisian kolom dua bagian pertama
For i = 1 To frmLauf.flxA.rows - 1 - 1
    frmKoreksi.flxAhelmet.TextMatrix(i,          2)          = -
frmLauf.flxA.TextMatrix(i, 2)

Next

' pengisian kolom dua bagian dua
z = 1
For k = i To 2 * (frmLauf.flxA.rows - 1 - 1)
    frmKoreksi.flxAhelmet.TextMatrix(k,          2)          =
frmLauf.flxA.TextMatrix(z, 1)
    z = z + 1
Next

End Sub

Sub BentukMatrikFhelmet()

Dim i As Long, k As Long, z As Long

For i = 1 To frmKoreksi.flxFhelmet.rows - 1
    frmKoreksi.flxFhelmet.TextMatrix(i, 0) = i
Next

' mengeset ukuran Flexi
frmKoreksi.flxFhelmet.rows = (2 * (frmLauf.flxA.rows - 1))
- 1
frmKoreksi.flxFhelmet.cols = 2

' pengisian kolom satu bagian pertama
For i = 1 To frmLauf.flxTM3.rows - 1 - 1
    frmKoreksi.flxFhelmet.TextMatrix(i,          1)          =
frmLauf.flxTM3.TextMatrix(i, 1)
Next

' pengisian kolom satu bagian dua
z = 1
For k = i To 2 * (frmLauf.flxTM3.rows - 1 - 1)
    frmKoreksi.flxFhelmet.TextMatrix(k,          1)          =
frmLauf.flxTM3.TextMatrix(z, 2)
    z = z + 1

```

Next

End Sub

Sub BentukFleksiA2helmet()

Dim i As Long, k As Long, z As Long, m As Long

' mengeset ukuran Flexi

frmKoreksi.flxA2helmet.rows = 2 + ((frmLauf.flxCalon.rows
- 1) * 2) + 1

frmKoreksi.flxA2helmet.cols = 5

For i = 1 To frmKoreksi.flxA2helmet.rows - 1
 frmKoreksi.flxA2helmet.TextMatrix(i, 0) = i

Next

' mengenkolkan semua

For i = 1 To frmKoreksi.flxA2helmet.rows - 1
 For k = 1 To 4
 frmKoreksi.flxA2helmet.TextMatrix(i, k) = 0

Next

Next

' pengisian kolom satu bagian pertama

frmKoreksi.flxA2helmet.TextMatrix(1, 1) =
frmLauf.flxA.TextMatrix(frmLauf.flxA.rows - 1, 1)
frmKoreksi.flxA2helmet.TextMatrix(1, 3) = 1

' di sini z menunjukkan kolom A2Helmet

z = 2

For i = 1 To frmLauf.flxCalon.rows - 1
 frmKoreksi.flxA2helmet.TextMatrix(z, 1) =
frmLauf.flxCalon.TextMatrix(i, 1)
 frmKoreksi.flxA2helmet.TextMatrix(z, 3) = 1
 z = z + 1

Next

' pengisian kolom satu bagian dua

frmKoreksi.flxA2helmet.TextMatrix(z, 2) =
frmLauf.flxA.TextMatrix(frmLauf.flxA.rows - 1, 2)
frmKoreksi.flxA2helmet.TextMatrix(z, 4) = 1
z = z + 1
For k = 1 To (frmLauf.flxCalon.rows - 1)
 frmKoreksi.flxA2helmet.TextMatrix(z, 1) =
frmLauf.flxCalon.TextMatrix(k, 2)
 frmKoreksi.flxA2helmet.TextMatrix(z, 4) = 1
 z = z + 1

Next

```

' pengisian kolom kedua bagian pertama

frmKoreksi.flxA2helmet.TextMatrix(1,      2)      =      -
frmLauf.flxA.TextMatrix(frmLauf.flxA.rows - 1, 2)
' di sini z menunjukkan kolom A2Helmet
z = 2
For i = 1 To frmLauf.flxCalon.rows - 1
    frmKoreksi.flxA2helmet.TextMatrix(z,      2)      =      -
frmLauf.flxCalon.TextMatrix(i, 2)
    z = z + 1
Next

' pengisian kolom satu bagian dua
frmKoreksi.flxA2helmet.TextMatrix(z,      2)      =
frmLauf.flxA.TextMatrix(frmLauf.flxA.rows - 1, 1)
z = z + 1
For k = 1 To (frmLauf.flxCalon.rows - 1)
    frmKoreksi.flxA2helmet.TextMatrix(z,      2)      =
frmLauf.flxCalon.TextMatrix(k, 1)
    z = z + 1
Next

End Sub

Sub IsiMatrikAhelmet()
Dim i As Long, k As Long
MatrikAhelmet = zeros(frmKoreksi.flxAhelmet.rows - 1,
frmKoreksi.flxAhelmet.cols - 1)

For i = 1 To frmKoreksi.flxAhelmet.rows - 1
    For k = 1 To frmKoreksi.flxAhelmet.cols - 1
        MatrikAhelmet.r2(i,      k)      =
frmKoreksi.flxAhelmet.TextMatrix(i, k)
    Next
Next

'MatrikA.Show
End Sub

Sub IsiMatrikFhelmet()
Dim i As Long, k As Long
MatrikFhelmet = zeros(frmKoreksi.flxFhelmet.rows - 1,
frmKoreksi.flxFhelmet.cols - 1)

For i = 1 To frmKoreksi.flxFhelmet.rows - 1
    For k = 1 To frmKoreksi.flxFhelmet.cols - 1
        MatrikFhelmet.r2(i,      k)      =
frmKoreksi.flxFhelmet.TextMatrix(i, k)
    Next
Next

'MatrikA.Show
End Sub

Sub IsiMatrikA2helmet()

```

```

Dim i As Long, k As Long
MatrikA2helmet = zeros(frmKoreksi.flxA2helmet.rows - 1,
frmKoreksi.flxA2helmet.cols - 1)

For i = 1 To frmKoreksi.flxA2helmet.rows - 1
    For k = 1 To frmKoreksi.flxA2helmet.cols - 1
        MatrikA2helmet.r2(i, k) =
frmKoreksi.flxA2helmet.TextMatrix(i, k)
    Next
Next

'MatrikA.Show
End Sub

Sub HitungHelmert()
Dim sementara As Variant, sementara1 As Variant,
sementara2 As Variant, sementara3 As Variant
Dim Temp As Variant
sementara = transpose(MatrikAhelmet)
'sementara.Show

sementara1 = mtimes(sementara, MatrikAhelmet)
'sementara1.Show

sementara2 = inv(sementara1)
'sementara2.Show

sementara3 = mtimes(transpose(MatrikAhelmet),
MatrikFhelmet)
'sementara3.Show

MatrikXhelmet = mtimes(sementara2, sementara3)
'MatrikXhelmet.Show
MatrikXhelmet = mtimes(MatrikA2helmet, MatrikXhelmet)
'MatrikXhelmet.Show

End Sub

Sub BentukFleksiZDhelmet()

Dim i As Long, k As Long, z As Long

' mengeset ukuran Flexi
frmKoreksi.flxZD.rows = (frmLauf.flxCalon.rows - 1) + 1 +
1
frmKoreksi.flxZD.cols = 7

frmKoreksi.flxZD.ColWidth(1) = 1700
frmKoreksi.flxZD.ColWidth(2) = 1700

frmKoreksi.flxZD.TextMatrix(0, 1) = "X0"
frmKoreksi.flxZD.TextMatrix(0, 2) = "Y0"
frmKoreksi.flxZD.TextMatrix(0, 3) = "X"
frmKoreksi.flxZD.TextMatrix(0, 4) = "Y"
frmKoreksi.flxZD.TextMatrix(0, 5) = "Delta ZxD"
frmKoreksi.flxZD.TextMatrix(0, 6) = "Delta ZyD"

```

```

For i = 1 To frmKoreksi.flxZD.rows - 1
    frmKoreksi.flxZD.TextMatrix(i, 0) = i
Next

```

```

' mengenkolkan semua
For i = 1 To frmKoreksi.flxZD.rows - 1
    For k = 1 To 6
        frmKoreksi.flxZD.TextMatrix(i, k) = 0
    Next
Next

```

```

' pengisian kolom satu
For i = 1 To (frmLauf.flxCalon.rows)
    frmKoreksi.flxZD.TextMatrix(i, 1) =
frmKoreksi.flxX.TextMatrix(i, 1)
Next

```

```

' pengisian kolom dua
For k = 1 To (frmLauf.flxCalon.rows)
    frmKoreksi.flxZD.TextMatrix(k, 2) =
frmKoreksi.flxX.TextMatrix(i, 1)
    i = i + 1
Next

```

```

' pengisian kolom ke tiga
frmKoreksi.flxZD.TextMatrix(1, 3) =
frmLauf.flxTM3.TextMatrix(frmLauf.flxA.rows - 1, 1)

```

```

' pengisian kolom ke empat
frmKoreksi.flxZD.TextMatrix(1, 4) =
frmLauf.flxTM3.TextMatrix(frmLauf.flxA.rows - 1, 2)

```

```

' pengisian kolom ke lima delta Zx
frmKoreksi.flxZD.TextMatrix(1, 5) =
frmKoreksi.flxZD.TextMatrix(1, 3) -
frmKoreksi.flxZD.TextMatrix(1, 1)

```

```

' pengisian kolom ke lima delta Zy
frmKoreksi.flxZD.TextMatrix(1, 6) =
frmKoreksi.flxZD.TextMatrix(1, 4) -
frmKoreksi.flxZD.TextMatrix(1, 2)

```

```

Dim tes As Double
tes = (180 * 22) / 7

```

```

Zd = Sqr(frmKoreksi.flxZD.TextMatrix(1, 5) ^ 2 +
frmKoreksi.flxZD.TextMatrix(1, 6) ^ 2)
'MsgBox Zd

```

```

Argumen = atan(frmKoreksi.flxZD.TextMatrix(1, 5) /
frmKoreksi.flxZD.TextMatrix(1, 6))
If Argumen.Simple < 0 Then
    Argumen = plus(Argumen, 360)
End If

```

```
'Zd = times(Zd, tes)
'MsgBox Zd

frmKoreksi.lblModulus = Round(Zd, 8)
frmKoreksi.lblSudut = Round(Argumen.Simple, 8)
```

```
End Sub
```

```
Sub BentukFleksiX()
```

```
Dim i As Long, k As Long, z As Long
```

```
' mengeset ukuran Flexi
frmKoreksi.flxX.rows = (MatrikXhelmet.rows) + 1
frmKoreksi.flxX.cols = 2
frmKoreksi.flxX.ColWidth(1) = 1700
```

```
For i = 1 To frmKoreksi.flxX.rows - 1
    frmKoreksi.flxX.TextMatrix(i, 0) = i
Next
```

```
' pengisian kolom satu
```

```
For i = 1 To (MatrikXhelmet.rows)
    frmKoreksi.flxX.TextMatrix(i, 1) = MatrikXhelmet(i,
1).Simple
Next
```

```
End Sub
```

```
Sub BentukFleksiSudut()
```

```
Dim i As Long, k As Long, z As Long, tes As Variant
```

```
' mengeset ukuran Flexi
frmKoreksi.flxSudut.rows = ((frmLauf.flxA.rows - 1 - 1) *
```

```
-
    (frmLauf.flxCalon.rows) -
1 + 1) + 1
frmKoreksi.flxSudut.cols = 5
```

```
frmKoreksi.flxSudut.ColWidth(1) = 1700
frmKoreksi.flxSudut.ColWidth(2) = 1700
frmKoreksi.flxSudut.ColWidth(3) = 1700
frmKoreksi.flxSudut.ColWidth(4) = 1700
```

```
frmKoreksi.flxSudut.TextMatrix(0, 1) = "X"
frmKoreksi.flxSudut.TextMatrix(0, 2) = "Y"
frmKoreksi.flxSudut.TextMatrix(0, 3) = "D0"
frmKoreksi.flxSudut.TextMatrix(0, 4) = "Alpha"
```

```
For i = 1 To frmKoreksi.flxSudut.rows - 1
    frmKoreksi.flxSudut.TextMatrix(i, 0) = i
Next
```

```

z = 1
For i = 1 To frmKoreksi.flxZD.rows - 1

    For k = 1 To frmLauf.flxA.rows - 2
        frmKoreksi.flxSudut.TextMatrix(z,          1)          =
frmKoreksi.flxZD.TextMatrix(i, 1) - _
        frmLauf.flxTM3.TextMatrix(k, 1)

        frmKoreksi.flxSudut.TextMatrix(z,          2)          =
frmKoreksi.flxZD.TextMatrix(i, 2) - _
        frmLauf.flxTM3.TextMatrix(k, 2)

        ' mengisi kolom ke 3 D0
        frmKoreksi.flxSudut.TextMatrix(z,          3)          =
Sqr(frmKoreksi.flxSudut.TextMatrix(z, 1) ^ 2 + _
        frmKoreksi.flxSudut.TextMatrix(z, 2) ^ 2)

        ' mengisi kolom ke 4 sudut
        tes = atan(frmKoreksi.flxSudut.TextMatrix(z, 1) /
-
        frmKoreksi.flxSudut.TextMatrix(z, 2))
        If tes.Simple < 0 Then
            tes = plus(tes, 360)
        End If
        frmKoreksi.flxSudut.TextMatrix(z, 4) = tes.Simple

        z = z + 1
    Next

Next

Next

' pengisian kolom satu
'   For i = 1 To (MatrikXhelmet.rows)
'       frmKoreksi.flxX.TextMatrix(i, 1) = MatrikXhelmet(i,
1).Simple
'   Next

End Sub

Sub BentukFleksiModulus()

    Dim i As Long, k As Long, z As Long

    ' mengeset ukuran Flexi

    frmKoreksi.flxModulus.rows = frmLauf.flxCalon.rows

    frmKoreksi.flxModulus.cols = 2
    frmKoreksi.flxModulus.ColWidth(1) = 1700

    For i = 1 To frmKoreksi.flxModulus.rows - 1
        frmKoreksi.flxModulus.TextMatrix(i, 0) = i
    Next

```



```

Dim pembagi As Double, penyebut As Double
' pengisian kolom satu
For i = 1 To frmKoreksi.flxModulus.rows - 1

    frmKoreksi.flxModulus.TextMatrix(i, 1) = 1

    For k = 1 To (frmLauf.flxA.rows - 1 - 1)
        frmKoreksi.flxModulus.TextMatrix(i, 1) =
frmKoreksi.flxModulus.TextMatrix(i, 1) * _
frmKoreksi.flxSudut.TextMatrix(((frmLauf.flxA.rows - 1 - 1) *
i) + k, 3)
    Next

    penyebut = 1
    For k = 1 To frmLauf.flxA.rows - 2
        penyebut = penyebut *
frmKoreksi.flxSudut.TextMatrix(k, 3)
    Next

    'MsgBox penyebut
    frmKoreksi.flxModulus.TextMatrix(i, 1) =
(frmKoreksi.flxModulus.TextMatrix(i, 1) / _
    penyebut) * Zd

Next

'MsgBox Zd

End Sub

Sub BentukFleksiArgumen()
Dim i As Long, k As Long, z As Long
' mengeset ukuran Flexi

frmKoreksi.flxArgumen.rows = frmLauf.flxCalon.rows

frmKoreksi.flxArgumen.cols = 2
frmKoreksi.flxArgumen.ColWidth(1) = 1700
'frmKoreksi.flxArgumen.ColWidth(2) = 1700

For i = 1 To frmKoreksi.flxArgumen.rows - 1
    frmKoreksi.flxArgumen.TextMatrix(i, 0) = i
Next

Dim pembagi As Double, penyebut As Double
' pengisian kolom satu

For i = 1 To frmKoreksi.flxArgumen.rows - 1

    frmKoreksi.flxArgumen.TextMatrix(i, 1) = 0

    For k = 1 To (frmLauf.flxA.rows - 1 - 1)
        frmKoreksi.flxArgumen.TextMatrix(i, 1) =
CDbl(frmKoreksi.flxArgumen.TextMatrix(i, 1)) + _

```

```

Cdbl(frmKoreksi.flxSudut.TextMatrix(((frmLauf.flxA.rows - 1 -
1) * i) + k, 4))
Next

penyebut = 0
For k = 1 To frmLauf.flxA.rows - 2
    penyebut = penyebut +
frmKoreksi.flxSudut.TextMatrix(k, 4)
Next
penyebut = -1 * (penyebut)
penyebut = Cdbl(penyebut)
'MsgBox penyebut
'MsgBox frmKoreksi.flxArgumen.TextMatrix(i, 1)

frmKoreksi.flxArgumen.TextMatrix(i, 1) =
Cdbl((frmKoreksi.flxArgumen.TextMatrix(i, 1)) +
Cdbl(penyebut)) + Argumen.Simple

If frmKoreksi.flxArgumen.TextMatrix(i, 1) > 360 Then
    frmKoreksi.flxArgumen.TextMatrix(i, 1) =
frmKoreksi.flxArgumen.TextMatrix(i, 1) - 360
End If

If frmKoreksi.flxArgumen.TextMatrix(i, 1) < 0 Then
    frmKoreksi.flxArgumen.TextMatrix(i, 1) =
frmKoreksi.flxArgumen.TextMatrix(i, 1) + 360
End If

'MsgBox Argumen.Simple
Next

'MsgBox Argumen.Simple

End Sub

Sub BentukFleksiAkhir()

Dim i As Long, k As Long, z As Long

' mengeset ukuran Flexi
frmKoreksi.flxHasil.rows = frmLauf.flxCalon.rows
frmKoreksi.flxHasil.cols = 3
frmKoreksi.flxHasil.ColWidth(1) = 1700
frmKoreksi.flxHasil.ColWidth(2) = 1700

frmKoreksi.flxHasil.TextMatrix(0, 1) = "X"
frmKoreksi.flxHasil.TextMatrix(0, 2) = "Y"

For i = 1 To frmKoreksi.flxHasil.rows - 1
    frmKoreksi.flxHasil.TextMatrix(i, 0) = i
Next

' pengisian kolom satu
For i = 1 To frmKoreksi.flxHasil.rows - 1

```

```

        frmKoreksi.flxHasil.TextMatrix(i,          1)          =
frmKoreksi.flxX.TextMatrix(i + 1, 1) + _
        frmKoreksi.flxModulus.TextMatrix(i,          1)          *
Sin(frmKoreksi.flxArgumen.TextMatrix(i, 1))
Next

' pengisian kolom dua (y)

For i = 1 To frmKoreksi.flxHasil.rows - 1
    frmKoreksi.flxHasil.TextMatrix(i,          2)          =
frmKoreksi.flxX.TextMatrix((frmLauf.flxCalon.rows - 1) + 2 +
i, 1) + _
    frmKoreksi.flxModulus.TextMatrix(i,          1)          *
Cos(frmKoreksi.flxArgumen.TextMatrix(i, 1))
Next

End Sub

Sub Ngeprint()

    frmVsPrinter.VS.FontName = "Courier new"
    frmVsPrinter.VS.StartDoc
    frmVsPrinter.VS.TextAlign = taCenterMiddle
    frmVsPrinter.VS.FontSize = 14
    frmVsPrinter.VS.Paragraph = "Hasil Transformasi Koordinat
Metode Selisih Terbagi"
    frmVsPrinter.VS.Paragraph = "Id Daerah: " & frmLauf.txtId
    frmVsPrinter.VS.Paragraph = "Nama Daerah: " &
frmLauf.txtNamaDaerah
    frmVsPrinter.VS.Paragraph = ""
    frmVsPrinter.VS.FontSize = 12

    frmVsPrinter.VS.Paragraph = ""
    frmVsPrinter.VS.Paragraph = "Beda Koordinat Satu"

    Dim StringKolom As String
    Dim StringKolomJudul As String

    StringKolom = ""

    frmVsPrinter.VS.StartTable
    Dim i As Long

    For i = 1 To flxBedaAbsisSatu.cols - 1
        StringKolom = StringKolom & ">1300"
        StringKolomJudul = StringKolomJudul & "|Kolom " & i
    Next
    StringKolomJudul = "|Delta x|Delta y|Delta X|Delta Y "

    '    frmVsPrinter.VS.AddTable "100|1000|" &
    '    "1000|1000|>1000|>1000", "No|No
Rek|Keterangan|D/K" & _
    '    "|Debet|Kredit", ""
    frmVsPrinter.VS.AddTable StringKolom, StringKolomJudul, ""

    frmVsPrinter.VS.TableCell(tcCols) = flxBedaAbsisSatu.cols

```

```

        For i = 0 To flxBedaAbsisSatu.cols
            frmVsPrinter.VS.TableCell(tcColWidth, , i + 1)
= "1400" 'no
        Next

    frmVsPrinter.VS.FontSize = 10

    Dim kolom As Long, baris As Long
    baris = 0
    frmVsPrinter.VS.TextAlign = taCenterMiddle

    Dim nomer As Long

    For i = 1 To flxBedaAbsisSatu.rows - 1
        nomer = nomer + 1
        baris = baris + 1

        frmVsPrinter.VS.TableCell(tcRows) = baris

        Dim k As Long

        frmVsPrinter.VS.TableCell(tcText, baris,
1) = "Baris: " & nomer
        For k = 2 To flxBedaAbsisSatu.cols
            frmVsPrinter.VS.TableCell(tcText,
baris, k) = flxBedaAbsisSatu.TextMatrix(i, k - 1)
        Next

        seling baris
    Next

    frmVsPrinter.VS.EndTable
    !*****
    frmVsPrinter.VS.Paragraph = ""
    frmVsPrinter.VS.Paragraph = "Selisih Terbagi Satu"

    StringKolom = ""

    frmVsPrinter.VS.StartTable

    For i = 1 To flxSelisihSatu.cols - 1
        StringKolom = StringKolom & ">1400"
        StringKolomJudul = StringKolomJudul & "|Kolom " & i
    Next
    StringKolomJudul = "|Zx|Zy|Delta X|Delta Y "
    frmVsPrinter.VS.AddTable StringKolom, StringKolomJudul, ""

    frmVsPrinter.VS.TableCell(tcCols) = flxSelisihSatu.cols

        For i = 0 To flxSelisihSatu.cols
            frmVsPrinter.VS.TableCell(tcColWidth, , i + 1)
= "2000" 'no
        Next

    frmVsPrinter.VS.FontSize = 10

```

```

baris = 0
nomer = 0

For i = 1 To flxSelisihSatu.rows - 1
    nomer = nomer + 1
    baris = baris + 1

    frmVsPrinter.VS.TableCell(tcRows) = baris

    frmVsPrinter.VS.TableCell(tcText,    baris,
1) = "Baris: " & nomer
    For k = 2 To flxSelisihSatu.cols
        frmVsPrinter.VS.TableCell(tcText,
baris, k) = flxSelisihSatu.TextMatrix(i, k - 1)
    Next

    seling baris

Next
frmVsPrinter.VS.EndTable
'
' '((((((((((((((((((((((((((((
' Print Hasil
'
frmVsPrinter.VS.Paragraph = ""
frmVsPrinter.VS.Paragraph = "Hasil Transformasi"

StringKolom = ""

frmVsPrinter.VS.StartTable

For i = 1 To flxHasil.cols - 1
    StringKolom = StringKolom & ">1300"
    StringKolomJudul = StringKolomJudul & "|Kolom " & i
Next
StringKolomJudul = "|X |Y"
frmVsPrinter.VS.AddTable StringKolom, StringKolomJudul, ""

frmVsPrinter.VS.TableCell(tcCols) = flxHasil.cols

    For i = 0 To flxHasil.cols
        frmVsPrinter.VS.TableCell(tcColWidth, , i + 1)
= "2100" 'no
    Next

frmVsPrinter.VS.FontSize = 10

baris = 0
nomer = 0

For i = 1 To flxHasil.rows - 1
    nomer = nomer + 1
    baris = baris + 1

    frmVsPrinter.VS.TableCell(tcRows) = baris

```



```

'           seling baris'
'
'           Next
'
'   frmVsPrinter.VS.EndTable
'
'   frmVsPrinter.VS.TextAlign = taLeftMiddle
'   frmVsPrinter.VS.Paragraph = ""
'
'   frmVsPrinter.VS.Paragraph = "Nilai Px: " & txtPx
'   frmVsPrinter.VS.Paragraph = "Nilai Py: " & txtPy
'   frmVsPrinter.VS.Paragraph = "Nilai Wx: " & txtWx
'   frmVsPrinter.VS.Paragraph = "Nilai Wy: " & txtWy
'
'
'   frmVsPrinter.VS.EndDoc
'   frmVsPrinter.VS.TextAlign = taLeftMiddle
'   Dim z As Integer
'   For z = 1 To frmVsPrinter.VS.PageCount
'       frmVsPrinter.VS.StartOverlay z
'       frmVsPrinter.VS.CurrentX = frmVsPrinter.VS.MarginLeft
'       frmVsPrinter.VS.CurrentY = frmVsPrinter.VS.MarginTop -
300
'       frmVsPrinter.VS.Paragraph = "Halaman " & z & " dari "
& frmVsPrinter.VS.PageCount
'       frmVsPrinter.VS.EndOverlay
'   Next
'
'   frmVsPrinter.Show 1
'
End Sub

'To Convert degrees to radians, multiply degrees by pi/180.
'To convert radians to degrees, multiply radians by 180/pi.
Private Sub Label1_Click()
End Sub

```


Algoritma Program Proses Hitung dan Hasil Transformasi Koordinat Metode Lauf Koreksi Sudut dan Jarak

Option Explicit

```
Private Sub cmdPrint_Click()  
    Ngeprint  
End Sub
```

```
Private Sub Form_Resize()  
    Move 0, 0  
End Sub
```

```
Sub Ngeprint()  
  
    frmVsPrinter.VS.FontName = "Courier new"  
    frmVsPrinter.VS.StartDoc  
    frmVsPrinter.VS.TextAlign = taCenterMiddle  
    frmVsPrinter.VS.FontSize = 14  
    frmVsPrinter.VS.Paragraph = "Hasil Transformasi Koreksi  
Sudut Dan Jarak"  
    frmVsPrinter.VS.Paragraph = "Id Daerah: " & frmLauf.txtId  
    frmVsPrinter.VS.Paragraph = "Nama Daerah: " &  
frmLauf.txtNamaDaerah  
    frmVsPrinter.VS.Paragraph = ""  
    frmVsPrinter.VS.FontSize = 12  
  
    frmVsPrinter.VS.Paragraph = ""  
    frmVsPrinter.VS.Paragraph = "Matrik A"  
  
    Dim StringKolom As String  
    Dim StringKolomJudul As String  
  
    StringKolom = ""  
  
    frmVsPrinter.VS.StartTable  
    Dim i As Long  
  
    For i = 1 To flxAhelmet.cols - 1  
        StringKolom = StringKolom & ">1300"  
        StringKolomJudul = StringKolomJudul & "|Kolom " & i  
    Next  
  
'    frmVsPrinter.VS.AddTable "100|1000|" &  
'                                "1000|1000|>1000|>1000", "No|No  
Rek|Keterangan|D/K" &  
'                                "|Debet|Kredit", ""  
    frmVsPrinter.VS.AddTable StringKolom, StringKolomJudul, ""  
  
    frmVsPrinter.VS.TableCell(tcCols) = flxAhelmet.cols  
  
    For i = 0 To flxAhelmet.cols
```

```

        frmVsPrinter.VS.TableCell(tcColWidth, , i + 1)
= "1400" 'no
        Next

    frmVsPrinter.VS.FontSize = 10

    Dim kolom As Long, baris As Long
    baris = 0
    frmVsPrinter.VS.TextAlign = taCenterMiddle

    Dim nomer As Long

    For i = 1 To flxAhelmet.rows - 1
        nomer = nomer + 1
        baris = baris + 1

        frmVsPrinter.VS.TableCell(tcRows) = baris

        Dim k As Long

        frmVsPrinter.VS.TableCell(tcText, baris,
1) = "Baris: " & nomer

        For k = 2 To flxAhelmet.cols
            frmVsPrinter.VS.TableCell(tcText,
baris, k) = flxAhelmet.TextMatrix(i, k - 1)
        Next

        seling baris
    Next

    frmVsPrinter.VS.EndTable
    '*****

    frmVsPrinter.VS.Paragraph = ""
    frmVsPrinter.VS.Paragraph = "Matrik F"

    StringKolom = ""

    frmVsPrinter.VS.StartTable

    For i = 1 To flxFhelmet.cols - 1
        StringKolom = StringKolom & ">1400"
        StringKolomJudul = StringKolomJudul & "|Kolom " & i
    Next

    frmVsPrinter.VS.AddTable StringKolom, StringKolomJudul, ""

    frmVsPrinter.VS.TableCell(tcCols) = flxFhelmet.cols

    For i = 0 To flxFhelmet.cols
        frmVsPrinter.VS.TableCell(tcColWidth, , i + 1)
= "2000" 'no

```

Next

frmVsPrinter.VS.FontSize = 10

baris = 0
nomer = 0

For i = 1 To flxFhelmet.rows - 1
 nomer = nomer + 1
 baris = baris + 1

 frmVsPrinter.VS.TableCell(tcRows) = baris

 frmVsPrinter.VS.TableCell(tcText, baris,
1) = "Baris: " & nomer

 For k = 2 To flxFhelmet.cols

 frmVsPrinter.VS.TableCell(tcText,

baris, k) = flxFhelmet.TextMatrix(i, k - 1)

 Next

 seling baris

Next

frmVsPrinter.VS.EndTable

,

' '((((((((((((((((((((((((((((

' Print Hasil

,

frmVsPrinter.VS.Paragraph = ""

frmVsPrinter.VS.Paragraph = "Matrik A2"

StringKolom = ""

frmVsPrinter.VS.StartTable

For i = 1 To flxA2helmet.cols - 1

 StringKolom = StringKolom & ">1300"

 StringKolomJudul = StringKolomJudul & "|Kolom " & i

Next

frmVsPrinter.VS.AddTable StringKolom, StringKolomJudul, ""

frmVsPrinter.VS.TableCell(tcCols) = flxA2helmet.cols

 For i = 0 To flxA2helmet.cols

 frmVsPrinter.VS.TableCell(tcColWidth, , i + 1)

= "2100" 'no

 Next

frmVsPrinter.VS.FontSize = 10

```

baris = 0
nomer = 0

For i = 1 To flxA2helmet.rows - 1
    nomer = nomer + 1
    baris = baris + 1

    frmVsPrinter.VS.TableCell(tcRows) = baris

    frmVsPrinter.VS.TableCell(tcText,    baris,
1) = "Baris: " & nomer
    For k = 2 To flxA2helmet.cols
        frmVsPrinter.VS.TableCell(tcText,
baris, k) = flxA2helmet.TextMatrix(i, k - 1)
    Next

    seling baris

Next

frmVsPrinter.VS.EndTable
'
' '11111111111111111111111111111111
'
frmVsPrinter.VS.Paragraph = ""
frmVsPrinter.VS.Paragraph = "Matrik X "

StringKolom = ""

frmVsPrinter.VS.StartTable

For i = 1 To flxX.cols - 1
    StringKolom = StringKolom & ">1300"
    StringKolomJudul = StringKolomJudul & "|Kolom " & i
Next

frmVsPrinter.VS.AddTable StringKolom, StringKolomJudul, ""

frmVsPrinter.VS.TableCell(tcColWidth, , 1) = "1300" 'no
frmVsPrinter.VS.TableCell(tcCols) = flxX.cols

    For i = 1 To flxX.cols
        frmVsPrinter.VS.TableCell(tcColWidth, , i + 1)
= "2700" 'no
    Next

frmVsPrinter.VS.FontSize = 10

```



```

        For i = 1 To flxZD.cols
            frmVsPrinter.VS.TableCell(tcColWidth, , i + 1)
= "1600" 'no
        Next

    frmVsPrinter.VS.FontSize = 10

    baris = 0
    nomer = 0

    For i = 1 To flxZD.rows - 1
        nomer = nomer + 1
        baris = baris + 1

        frmVsPrinter.VS.TableCell(tcRows) = baris

        frmVsPrinter.VS.TableCell(tcText, baris,
1) = "Baris: " & nomer
        For k = 2 To flxZD.cols
            frmVsPrinter.VS.TableCell(tcText,
baris, k) = Round(flxDZD.TextMatrix(i, k - 1), 4)
'Format(Round(flxDZD.TextMatrix(i, k - 1), 5), "#####.###0")
        Next
        seling baris
    Next
    frmVsPrinter.VS.EndTable

    frmVsPrinter.VS.Paragraph = ""
    frmVsPrinter.VS.Paragraph = "Beda Koordinat Titik Sekutu Dan
Titik Obyek "

    StringKolom = "1300"

    frmVsPrinter.VS.StartTable

    For i = 1 To flxSudut.cols - 1
        StringKolom = StringKolom & ">2300"
        StringKolomJudul = StringKolomJudul & "|Kolom " & i
    Next
    StringKolomJudul = "|X|Y|Do|Alpha"
    frmVsPrinter.VS.AddTable StringKolom, StringKolomJudul, ""

    frmVsPrinter.VS.TableCell(tcColWidth, , 1) = "1300" 'no
    frmVsPrinter.VS.TableCell(tcCols) = flxSudut.cols

    For i = 1 To flxSudut.cols
        frmVsPrinter.VS.TableCell(tcColWidth, , i + 1)
= "1600" 'no

```

Next

frmVsPrinter.VS.FontSize = 10

baris = 0
nomer = 0

For i = 1 To flxSudut.rows - 1
nomer = nomer + 1
baris = baris + 1

frmVsPrinter.VS.TableCell(tcRows) = baris

frmVsPrinter.VS.TableCell(tcText, baris,
1) = "Baris: " & nomer
For k = 2 To flxSudut.cols
frmVsPrinter.VS.TableCell(tcText,
baris, k) = Round(flxSudut.TextMatrix(i, k - 1), 4)
'Format(flxSudut.TextMatrix(i, k - 1), "#####.#####0")
Next
seling baris
Next

frmVsPrinter.VS.EndTable

' Modulus

frmVsPrinter.VS.Paragraph = ""
frmVsPrinter.VS.Paragraph = "Modulus Titik Obyek "

StringKolom = "1300"

frmVsPrinter.VS.StartTable

For i = 1 To flxModulus.cols - 1
StringKolom = StringKolom & ">2300"
StringKolomJudul = StringKolomJudul & "|Kolom " & i
Next
StringKolomJudul = "|Modulus"
frmVsPrinter.VS.AddTable StringKolom, StringKolomJudul, ""
frmVsPrinter.VS.TableCell(tcColWidth, , 1) = "1300" 'no
frmVsPrinter.VS.TableCell(tcCols) = flxModulus.cols

For i = 1 To flxModulus.cols
frmVsPrinter.VS.TableCell(tcColWidth, , i + 1)
= "1600" 'no
Next

```
frmVsPrinter.VS.FontSize = 10
```

```
baris = 0  
nomer = 0
```

```
For i = 1 To flxModulus.rows - 1  
    nomer = nomer + 1  
    baris = baris + 1
```

```
    frmVsPrinter.VS.TableCell(tcRows) = baris
```

```
        frmVsPrinter.VS.TableCell(tcText,    baris,  
1) = "Baris: " & nomer  
        For k = 2 To flxModulus.cols  
            frmVsPrinter.VS.TableCell(tcText,  
baris,    k) = Format(flxModulus.TextMatrix(i,    k    -    1),  
"#####.#####0")  
        Next  
        seling baris  
    Next
```

```
frmVsPrinter.VS.EndTable
```

```
' Argumen
```

```
frmVsPrinter.VS.Paragraph = ""
```

```
frmVsPrinter.VS.Paragraph = "Argumen Titik Obyek "
```

```
StringKolom = "1300"
```

```
frmVsPrinter.VS.StartTable
```

```
For i = 1 To flxArgumen.cols - 1
```

```
    StringKolom = StringKolom & "|>2300"
```

```
    StringKolomJudul = StringKolomJudul & "|Kolom " & i
```

```
Next
```

```
StringKolomJudul = "|Argumen"
```

```
frmVsPrinter.VS.AddTable StringKolom, StringKolomJudul, ""
```

```
frmVsPrinter.VS.TableCell(tcColWidth, , 1) = "1300" 'no
```

```
frmVsPrinter.VS.TableCell(tcCols) = flxArgumen.cols
```

```
    For i = 1 To flxArgumen.cols
```

```
        frmVsPrinter.VS.TableCell(tcColWidth, , i + 1)
```

```
= "1600" 'no
```

```
    Next
```

```
frmVsPrinter.VS.FontSize = 10
```



```
baris = 0
nomer = 0
```

```
For i = 1 To flxArgumen.rows - 1
    nomer = nomer + 1
    baris = baris + 1
```

```
    frmVsPrinter.VS.TableCell(tcRows) = baris
```

```
        frmVsPrinter.VS.TableCell(tcText,    baris,
1) = "Baris: " & nomer
        For k = 2 To flxArgumen.cols
            frmVsPrinter.VS.TableCell(tcText,
baris,    k) = Format(flxArgumen.TextMatrix(i,    k - 1),
"0.#####0")
        Next
        seling baris
    Next
```

```
frmVsPrinter.VS.EndTable
```

```
' Hasil
```

```
frmVsPrinter.VS.Paragraph = ""
frmVsPrinter.VS.Paragraph = "Hasil Transformasi Koordinat
Koreksi Sudut Dan Jarak"
```

```
StringKolom = "1300"
```

```
frmVsPrinter.VS.StartTable
```

```
For i = 1 To flxHasil.cols - 1
    StringKolom = StringKolom & ">2300"
    StringKolomJudul = StringKolomJudul & "|Kolom " & i
Next
StringKolomJudul = "|X|Y"
```

```
frmVsPrinter.VS.AddTable StringKolom, StringKolomJudul, ""
```

```
frmVsPrinter.VS.TableCell(tcColWidth, , 1) = "1300" 'no
```

```
frmVsPrinter.VS.TableCell(tcCols) = flxHasil.cols
```

```
    For i = 1 To flxHasil.cols
        frmVsPrinter.VS.TableCell(tcColWidth, , i + 1)
```

```
= "2000" 'no
```

```
    Next
```

```
frmVsPrinter.VS.FontSize = 10
```

```
baris = 0
```

```
nomer = 0
```

```
For i = 1 To flxHasil.rows - 1
```

```
    nomer = nomer + 1
```

```
    baris = baris + 1
```

```
    frmVsPrinter.VS.TableCell(tcRows) = baris
```

```
    frmVsPrinter.VS.TableCell(tcText, baris,
```

```
1) = "Baris: " & nomer
```

```
    For k = 2 To flxHasil.cols
```

```
        frmVsPrinter.VS.TableCell(tcText,
```

```
baris, k) = Format(flxHasil.TextMatrix(i, k - 1),
```

```
"#####.####0")
```

```
    Next
```

```
    seling baris
```

```
Next
```

```
frmVsPrinter.VS.EndTable
```

```
frmVsPrinter.VS.TextAlign = taLeftMiddle
```

```
frmVsPrinter.VS.Paragraph = ""
```

```
frmVsPrinter.VS.Paragraph = "Nilai Modulus Titik Sekutu  
Terakhir: " & lblModulus
```

```
frmVsPrinter.VS.Paragraph = "Nilai Argumen Titik Sekutu  
Terakhir: " & lblSudut
```

```
' frmVsPrinter.VS.Paragraph = "Nilai Wx: " & txtWx
```

```
' frmVsPrinter.VS.Paragraph = "Nilai Wy: " & txtWy
```

```
frmVsPrinter.VS.EndDoc
```

```
frmVsPrinter.VS.TextAlign = taLeftMiddle
```

```
Dim z As Integer
```

```
For z = 1 To frmVsPrinter.VS.PageCount
```

```
    frmVsPrinter.VS.StartOverlay z
```

```
    frmVsPrinter.VS.CurrentX = frmVsPrinter.VS.MarginLeft
```

```
    frmVsPrinter.VS.CurrentY = frmVsPrinter.VS.MarginTop -
```

```
300
```

```
    frmVsPrinter.VS.Paragraph = "Halaman " & z & " dari "
```

```
& frmVsPrinter.VS.PageCount
```

```
    frmVsPrinter.VS.EndOverlay
```

```
Next
```

```
frmVsPrinter.Show 1
```

```
End Sub
```