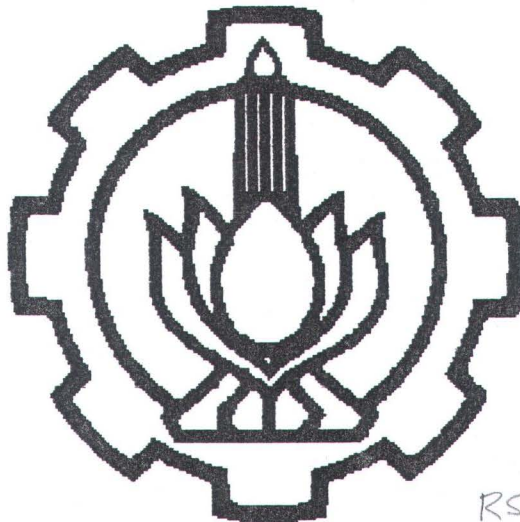


TUGAS AKHIR

(NA - 1701)

**KOMPUTERISASI TEKNIK PEMBENTANGAN KULIT DENGAN
METODE FLAT PLAN PADA PEKERJAAN MOULDLOFT**



RSPD
623.84
Sut
k-1

1999

Disusun oleh :

YUNIOR JOKO SUTOPO

Nrp : 4196 100 501

**JURUSAN TEKNIK PERKAPALAN
FAKULTAS TEKNOLOGI KELAUTAN
ITS - SURABAYA**

1999



JURUSAN TEKNIK PERKAPALAN

FAKULTAS TEKNOLOGI KELAUTAN ITS

SURAT KEPUTUSAN TUGAS AKHIR (NA 1701)

No. : 145 /PT12.FTK2/M/199 8

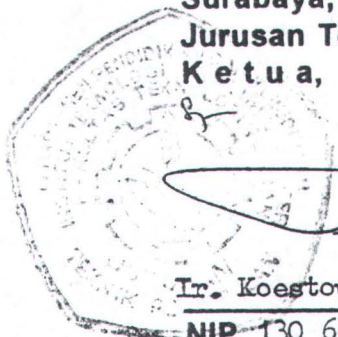
Nama Mahasiswa : Yunior Joko Sutopo
Nomor Pokok : 4196100501
Tanggal diberikan tugas : 25 September 1998
Tanggal selesai tugas : 25 Pebruari 1999
Dosen Pembimbing : 1. Ir. Djauhar Manfaat, MSc, Ph.D
2. Ir. Asjhar Imron, MSc, MSE, PED

Uraian / judul tugas akhir yang diberikan :

KOMPUTERISASI TEKNIK PEMBENTANGAN KULIT DENGAN METODE FLAT PLAN PADA PEKERJAAN MOULDLOFT,

sOn

Surabaya, 28 September 1998
Jurusan Teknik Perkapalan FTK-ITS
Ketua,



Tembusan :

1. Yth. Dekan FTK-ITS.
2. Yth. Dosen Pembimbing.
3. Arsip.

Ir. Koestowo Sastro Wiyono.

NIP. 130 687 430.



SURAT KETERANGAN PERBAIKKAN

TUGAS AKHIR (NA - 1701)

Yang bertanda tangan dibawah ini menerangkan bahwa, Mahasiswa :

Nama : YUNIOR JOKO SUTOPO

Nrp : 41916100501

Fakultas/Jurusan : FTK/ TEKNIK PERKAPALAN

Judul Tugas Akhir :

KOMPUTERISASI TEKNIK PEMBENTANGAN KULIT DENGAN METODE FLAT PLAN PADA PEKERJAAN MOULDLOFT

Telah melakukan perbaikan tugas akhir (revisi) sesuai dengan hasil sidang pada tanggal 02 Agustus 1999.



Surabaya, 12 Agustus 1999

Dosen Pembimbing I Tugas Akhir

(Ir. Djauhar Manfaat, M.Sc., Ph.D)

Nip. 131 651 444

SURAT KETERANGAN PERBAIKKAN

TUGAS AKHIR (NA - 1701)

Yang bertanda tangan dibawah ini menerangkan bahwa, Mahasiswa :

Nama : YUNIOR JOKO SUTOPO

Nrp : 41916100501

Fakultas/Jurusan : FTK/ TEKNIK PERKAPALAN

Judul Tugas Akhir :

KOMPUTERISASI TEKNIK PEMBENTANGAN KULIT DENGAN METODE

FLAT PLAN PADA PEKERJAAN MOULDLOFT

Telah melakukan perbaikan tugas akhir (revisi) sesuai dengan hasil sidang pada tanggal 02 Agustus 1999.

Surabaya, // Agustus 1999

Dosen Pembimbing II Tugas Akhir



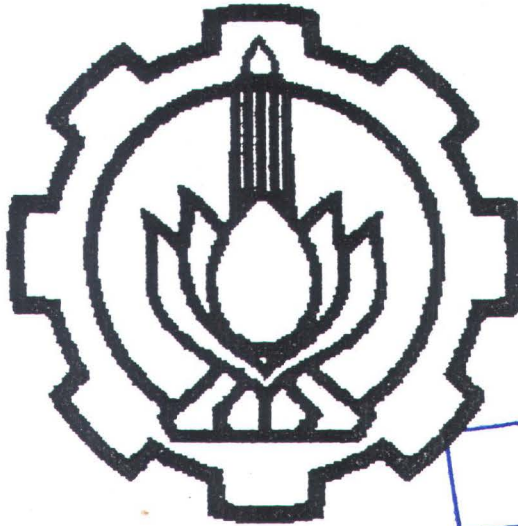
(Ir. Asjhar Imron, M.Sc, MSE, PED)

Nip. 130 687 432

LEMBAR PENGESAHAN

TUGAS AKHIR (NA -1701)

KOMPUTERISASI TEKNIK PEMBENTANGAN KULIT DENGAN
METODE FLAT PLAN PADA PEKERJAAN MOULDLOFT



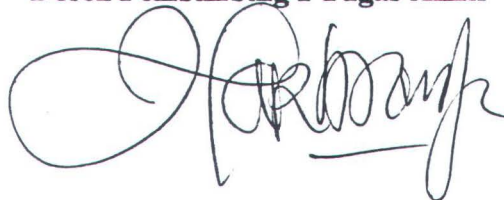
PERPUSTAKAAN ITS	
Tgl. Terima	27-6-2000
Terima Dari	H
No. Agenda Prp.	21-346

Surabaya, 12 Agustus 1999



Mengetahui dan Menyetujui

Dosen Pembimbing I Tugas Akhir



(Ir. Djauhar Manfaat, M.Sc, Phd)

Nip . 131 651 444



LEMBAR PENGESAHAN

TUGAS AKHIR (NA -1701)

**KOMPUTERISASI TEKNIK PEMBENTANGAN KULIT DENGAN
METODE FLAT PLAN PADA PEKERJAAN MOULDLOFT**



Surabaya, $\frac{11}{8}$ - 1999

**Mengetahui dan Menyetujui
Dosen Pembimbing II Tugas Akhir**

A handwritten signature in black ink, appearing to read 'Asjhar Imron', is written over a horizontal line.

(Ir. Asjhar Imron, M.Sc, MSE, PED)

Nip . 130 687 432

ABSTRAK

Dengan semakin berkembangnya teknologi komputer dewasa ini, maka usaha efisiensi dalam berbagai bidang akan semakin terwujud, dimana komputer merupakan perangkat lunak yang dapat membantu dalam proses pengambilan keputusan terhadap pekerjaan yang kita lakukan. Apalagi dengan diterapkannya expert system (sistem pakar) dimana kepandaian komputer digabung dengan kepandaian seorang ahli akan semakin dapat memberikan kepastian akan teknologi yang kita pakai.

Dalam bidang perkapalan, bagian mouldloft merupakan salah satu bagian yang amat vital dalam proses pembangunan kapal karena merupakan penghubung pekerjaan bagian perencanaan dengan bagian produksi. Salah satu tugas mouldloft adalah memberikan informasi tentang bentuk dan ukuran sebenarnya dari pelat yang dibentangkan, dimana data pelat tersebut berasal dari gambar body plan dengan menggunakan skala 1 : 1 sehingga memerlukan lantai yang cukup luas untuk menggambarkan body plan tersebut. Sedangkan pelaksanaan pekerjaan pembentangan kulit akan mengalami kesulitan karena pekerjaan tersebut dilakukan secara manual sehingga akan mempengaruhi hasil pekerjaan mouldloft dan berakibat buruk pada proses fabrikasi dan assembly kapal.

*Oleh karena itu diperlukan alternatif penggunaan program komputer bukaan kulit untuk mengatasi permasalahan tersebut, dimana dengan menggunakan program komputer ini penggambaran body plan dapat dilaksanakan cepat dan dapat menggunakan segala macam skala penggambaran, sedangkan untuk pekerjaan pembentangan kulit pelat dapat dilakukan secara cepat dan tepat untuk mendapatkan bentuk dan ukuran sebenarnya bahkan hasil pembentangan dapat langsung dicetak dan dapat ditransfer ke NC - Machine Cutting dengan menggunakan file ekstension *.dxf.*

KATA PENGANTAR

Dengan mengucapkan syukur Alhamdulillah kepada Allah SWT, penulis telah menyelesaikan Tugas Akhir berjudul “ **Komputerisasi Teknik Pembentangan Kulit Dengan Metode Flat Plan Pada Pekerjaan Mouldloft** “ yang merupakan salah satu mata kuliah di Jurusan Teknik Perkapalan.

Tugas Akhir berjudul “ **Komputerisasi Teknik Pembentangan Kulit Dengan Metode Flat Plan Pada Pekerjaan Mouldloft** “ ini bertujuan untuk memberikan suatu pemecahan permasalahan teknik pembentangan kulit yang dikerjakan secara manual di galangan kapal dan juga memberikan dasar pengetahuan bagi para mahasiswa tentang lingkup kerja mouldloft yang merupakan jembatan informasi dari tahap perencanaan ke tahap produksi.

Tugas Akhir ini dapat tersusun dengan baik tentu saja atas bantuan dari berbagai pihak, dan untuk itu penulis menyampaikan terima kasih dan penghargaan setinggi-tingginya kepada :

1. Ir. Koestowo SW, selaku Ketua Jurusan Teknik Perkapalan, FTK ITS Surabaya.
2. Ir. Andjar Suharto, selaku Dosen Wali.
3. Ir. Djauhar Manfaat, M.Sc., Ph.D., selaku Dosen Pembimbing I Tugas Akhir.
4. Ir. Asjhar Imron, M.Sc., M.SE., PED, selaku Dosen Pembimbing II Tugas Akhir.
5. Rekan-rekan POLTEK Perkapalan ITS - Surabaya bagian mouldloft yang telah memberikan bimbingan dan bantuan atas Tugas Akhir penulis.
6. Keluarga penulis di Jember yang telah memberikan dukungan moril dan materiil kepada penulis demi terselesainya Tugas Akhir ini.
7. Rekan-rekan penulis yang ada di kost-kostan, terima kasih atas dukungannya.

8. Rekan-rekan LJ '96, terima kasih atas segala bantuan dan kesetiakawanan selama ini, semoga persahabatan kita tetap kukuh selamanya.

Tugas Akhir ini telah berusaha disusun dengan sebaik-baiknya, meskipun demikian penulis menyadari masih adanya kekurangan-kekurangan, sehingga untuk itu penulis mengharapkan kritik dan saran dari berbagai pihak demi kesempurnaan Tugas Akhir ini.

Surabaya, 23 Juli 1999

Penulis

(Yunior Joko Sutopo)

DAFTAR ISI

Lembar Judul	
Lembar Pengesahan	
Abstrak.....	i
Kata Pengantar.....	ii
Daftar Isi.....	iv
Daftar Gambar.....	vi
Daftar Tabel.....	ix

BAB I. PENDAHULUAN

I. 1 Latar Belakang Permasalahan.....	1
I. 2 Perumusan Masalah.....	3
I. 3 Batasan Masalah.....	4
I. 4 Tujuan dan Manfaat.....	5
I. 5 Metodologi.....	6
I. 6 Sistematika Penulisan.....	7

BAB II STUDI LITERATUR

II. 1 Garis Besar Pekerjaan Mouldloft	
II. 1.1 Pengertian.....	8
II. 1.2 Fungsi Dan Peranan.....	11
II. 1.3 Akurasi Lofting.....	11
II. 2 Lingkup Pekerjaan Mouldloft	
II. 2.1 Aliran Informasi Dari Pekerjaan Mouldloft.....	12
II. 2.2 Scale Lofting.....	16
II. 2.3 Proses Penggambaran Rencana Garis Dalam Skala Penuh.....	18
II. 2.4 Proses Pembuatan Marking List.....	20
II. 2.5 Proses Pembuatan Rambu Film.....	21
II. 2.6 Simbol-Simbol Penandaan.....	23
II. 3 Teori - Teori Dasar Penggambaran Bukaan	
II. 3.1 Umum.....	24

II. 3.2	Matematika dan Konstruksi Geometri Praktis.....	25
II. 3.3	Konstruksi Geometri Dasar.....	31
II. 3.4	Dasar Penggambaran Bukaan.....	40
II. 4	Penggambaran Bentangan	
II. 4.1	Umum.....	41
II. 4.2	Menggambarkan Bentangan Dari Suatu Garis.....	42
II. 4.3	Menggambarkan Bukaan Plat.....	44
II. 4.4	Metode Flat Plan.....	46
BAB. III KONSEP PEMROGRAMAN KOMPUTER		
III. 1	Teori - Teori Pemrograman	
III. 1.1	AutoLISP.....	49
III. 1.2	Microsoft Visual Basic.....	50
III. 1.2.1	Visual Basic Control.....	51
III. 1.2.2	Dasar-Dasar Pemrograman.....	51
III. 2	Langkah-Langkah Pemrograman Bukaan Kulit.....	54
BAB. IV OTOMATISASI BUKAAN KULIT METODE FLAT PLAN		
IV. 1	Akurasi Hasil Bukaan Kulit Metode Flat Plan.....	83
IV. 2	Kelebihan dan Kekurangan Program Bukaan Kulit Metode Flat Plan.....	88
BAB. V KESIMPULAN DAN SARAN		
V. 1	Kesimpulan.....	91
V. 2	Saran-Saran.....	92
DAFTAR PUSTAKA.....		94
DAFTAR LAMPIRAN.....		95

DAFTAR GAMBAR

	Hal.
Gambar 2.1. Aliran Informasi Mouldloft.....	13
Gambar 2.2. Simbol Arah Penandaan.....	23
Gambar 2.3. Segitiga (pemodelan matematis).....	26
Gambar 2.4. Lingkaran (pemodelan matematis).....	27
Gambar 2.5. Bujur Telur (pemodelan matematis).....	28
Gambar 2.6. Bujur Sangkar (pemodelan matematis).....	28
Gambar 2.7. Persegi Panjang (pemodelan matematis).....	28
Gambar 2.8. Kotak (pemodelan matematis).....	29
Gambar 2.9. Silinder (pemodelan matematis).....	29
Gambar 2.10. Perbandingan.....	30
Gambar 2.11. Trigonometri Sudut.....	30
Gambar 2.12. Penggambaran Garis Tegak Lurus (a).....	31
Gambar 2.13. Penggambaran Garis Tegak Lurus (b).....	32
Gambar 2.14. Penggambaran Garis Tegak Lurus (c).....	33
Gambar 2.15. Penggambaran Garis Tegak Lurus (d).....	34
Gambar 2.16. Penggambaran Garis Tegak Lurus (e).....	35
Gambar 2.17. Penggambaran Garis Tegak Lurus (f).....	35
Gambar 2.18. Pembagian Garis Menjadi Beberapa Bagian Yang Sama.....	36
Gambar 2.19. Pembagian $\frac{1}{4}$ Lingkaran Menjadi 3 Bagian Yang Sama.....	37
Gambar 2.20. Membuat Segi Enam Beraturan.....	38
Gambar 2.21. Membuat Ellips.....	39

Gambar 2.22. Membuat Lingkaran Bulat Telur.....	40
Gambar 2.23. Garis Untuk Menentukan Panjang Bentangan.....	42
Gambar 2.24. Bentangan Upper Strake Edge Pada Pelat Kulit.....	43
Gambar 2.25. Proyeksi Garis Pada Lambung.....	44
Gambar 2.26. Metode Pengembangan Diagonal.....	45
Gambar 2.27. Bentuk Kurva Flat Plan.....	46
Gambar 2.28. Metode Flat Plan.....	48
Gambar 3.1. Flow Chart Program Bukaak Kulit.....	56
Gambar 3.2. Pembagian Titik Kurva Pada Body Plan.....	58
Gambar 3.3. Cara Pembacaan Koordinat Kurva.....	58
Gambar 3.8. Body Plan.....	64
Gambar 3.9. Perintah PICK Daerah Pelat.....	65
Gambar 3.10 Tampilan Daerah Pelat Terpilih.....	66
Gambar 3.11. Potongan Pelat.....	67
Gambar 3.12. Flow Chart Metode Flat Plan.....	68
Gambar 3.13. Metode Flat Plan.....	69
Gambar 3.14. Hasil Bukaak Kulit Metode Flat Plan.....	72
Gambar 3.15. Form Tampilan Program Utama.....	76
Gambar 3.16. Tampilan Keterangan Program Bukaak Kulit Metode Flat Plan.....	77
Gambar 3.17. Form Tabel Data Koordinat.....	78
Gambar 3.18. Form Dimensi Sebenarnya.....	79
Gambar 4.1. Posisi Kurva Gading Metode Flat Plan.....	83
Gambar 4.2. Potongan Pelat.....	84
Gambar 4.3. Hasil Bukaak Kulit Dari Potongan Pelat Dalam Gambar 4.2 Dengan Metode Flat Plan	84

Gambar 4.6. Tampilan Form Dimensi Sebenarnya Hasil Bukaan Kulit Pelat No. Gading

21-29.....87

DAFTAR TABEL

	Hal.
Tabel 3.4. Data Koordinat Kurva Gading (no. gading 21-29).....	60
Tabel 3.5. Data Koordinat Kurva Sambungan Pelat (no.gading 21-29).....	61
Tabel 3.6. Data Koordinat Kurva Sambungan Pelat (no.gading 66-74).....	61
Tabel 3.7. Data Koordinat Kurva Gading (no.gading 66-74).....	62
Tabel 4.4. Data Panjang Sebenarnya Kurva Gading Dan Las (cara manual).....	86
Tabel 4.5. Data Panjang Sebenarnya Kurva Gading Dan Las (komputer).....	87
Tabel 4.7. Data Panjang Sebenarnya Kurva Gading Dan Las (gambar kerja).....	88

BAB I

PENDAHULUAN

I.1 Latar Belakang Permasalahan

Kapal merupakan konstruksi yang dibangun dalam bentuk 3 dimensi, dengan bentuk yang rumit dan adalah hal yang sulit untuk mengetahui ukuran dan bentuk kapal yang sebenarnya dalam proyeksi 2 dimensi atau dalam bentuk yang datar. Untuk memecahkan persoalan diatas, maka suatu perencanaan bukaan dengan metode-metode tertentu perlu dilakukan. Metode-metode bukaan dari konstruksi lambung ini mempunyai tingkat kesulitan tertentu serta digunakan menurut bentuk yang akan dikembangkan.

Pekerjaan-pekerjaan pembentangan kulit merupakan pekerjaan mouldloft yang menjadi sumber informasi dari pekerjaan konstruksi lambung. Kesalahan-kesalahan yang ditemukan pada tahap fabrikasi ataupun assembly salah satunya berasal dari kesalahan-kesalahan pada tahap mouldloft (*Sentosa, 1992*), dimana ketidakteelitian (*inacuracy*) ukuran dan kesalahan bentuk merupakan kesalahan yang sering terjadi terutama pada saat proses pembentangan kulit lambung untuk mendapatkan ukuran dan bentuk yang sebenarnya dari pelat yang akan dibentangkan. Hal ini terutama dilakukan pada pekerjaan mouldloft yang hanya menggunakan tenaga manusia dengan bantuan peralatan-peralatan yang ada di mouldloft, dimana terdapat keterbatasan tenaga dan kemampuan manusia dalam melakukan suatu pekerjaan yang berskala besar seperti pembangunan kapal baru. Hal-hal tersebut merupakan kerugian-kerugian yang meliputi :

- *Kehilangan jam kerja*

Kesalahan-kesalahan dalam hal ketidakteelitian penentuan ukuran dan bentuk komponen yang ditemukan pada tahap fabrikasi maupun assembly menyebabkan pembuangan waktu yang sia-sia, karena sering terjadi “rework” (pekerjaan yang diulang-ulang).

- *Pembuangan material yang sia-sia*

Setiap kesalahan dalam penentuan ukuran maupun bentuk komponen pada tahap fabrikasi maupun assembly, seperti margin pelat yang terlalu lebar atau bentuk komponen yang tidak sesuai dengan disain yang diharapkan, menyebabkan pembuangan material pelat yang sia-sia.

- *Kehilangan jadwal pekerjaan*

Adanya “rework” akibat kesalahan-kesalahan seperti yang dijelaskan diatas menyebabkan pekerjaan yang semula diperkirakan selesai sesuai jadwal pekerjaan yang telah dibuat menjadi tidak selesai, sehingga tidak sesuai dengan yang diharapkan.

- *Mempengaruhi mental para karyawan*

Banyaknya “rework” akibat kesalahan-kesalahan tersebut diatas, maka secara psikologis akan mempengaruhi mental para karyawan seperti tingkah laku mereka yang buruk, malas, patah semangat dan lain-lain.

Untuk menghindari kerugian-kerugian tersebut sedini mungkin, maka perlu ada semacam standar ketelitian pada pekerjaan mouldloft. Prosedur penerapan standar ketelitian yang telah dibuat untuk mengatasi permasalahan diatas akan dijelaskan pada BAB II. Namun, karena dalam pelaksanaannya tetap menggunakan cara tradisional (tenaga manusia), maka kemungkinan-kemungkinan kesalahan masih

ada. Hal ini karena keterbatasan tenaga manusia, apalagi untuk proyek yang berskala besar. Disamping masalah ketelitian juga terdapat masalah dalam penggambaran body plan dalam skala penuh (1:1), dan bukaan kulitnya. Proses penggambaran body plan dan bukaan kulit yang dilakukan dengan cara manual di lantai mouldloft yang luas cukup memakan waktu dan tempat sehingga hal ini sangat tidak efisien dan tidak efektif untuk pembangunan kapal baru maupun reparasi kapal.

Berangkat dari permasalahan tersebut, maka alternatif penggunaan komputer diharapkan dapat membantu memecahkan permasalahan-permasalahan diatas. Kelebihan sistem komputer dalam hal ini adalah (*Imron, 1995*) :

- Dengan komputer, gambar akan disimpan dalam informasi digital sehingga mempunyai fleksibilitas tinggi untuk dimanipulasi.
- Kualitas disain yang lebih baik dibandingkan dari disain secara manual.

Dalam Tugas Akhir ini penulis menyusun program komputer yang ditulis dalam bahasa AutoLISP untuk mensimulasi proses bukaan kulit kapal. AutoLISP memiliki kelebihan dalam hal penggambaran, dimana penggambaran body plan dapat dilakukan secara cepat dalam skala berapapun besarnya. Penggambaran hasil bukaan kulit dari gambar body plan yang telah dibuat dapat dilakukan sesuai dengan hasil yang diharapkan dengan metode bukaan yang ditetapkan sehingga ketidakteelitian dan kesalahan-kesalahan pada tahap fabrikasi dan assembly dapat dikurangi.

I. 2 Perumusan Masalah

Keterbatasan tenaga manusia (secara kuantitas) merupakan suatu kendala dalam pelaksanaan pekerjaan mouldloft. Karena keterbatasan tersebut, maka hasil yang diharapkan dari pekerjaan-pekerjaan mouldloft sering tidak memuaskan bahkan

berakibat buruk pada proses selanjutnya baik fabrikasi maupun assembly, seperti ukuran dan bentuk sebenarnya yang salah dari pelat yang dibentangkan sehingga banyak terjadi “rework” (pekerjaan yang berulang-ulang) pada tahap-tahap tersebut. Berangkat dari permasalahan tersebut, maka penulis mencoba merumuskan permasalahan-permasalahan yang meliputi :

- Bagaimana caranya menentukan ukuran dan bentuk yang sebenarnya dari pelat yang akan dibentangkan sehingga dapat membantu bahkan mengurangi tenaga manusia pada pekerjaan mouldloft ?
- Bagaimana komputer dapat membantu menentukan standar ketelitian (accuracy standart) pada pekerjaan mouldloft ?
- Jenis program komputer apa yang sesuai untuk diterapkan dalam membantu pekerjaan mouldloft ?

I.3 Batasan Masalah

Dalam pembuatan program bukaan kulit ini, agar mencapai sasaran pemecahan permasalahan maka batasan-batasan masalah ditentukan sebagai berikut :

- Penggunaan program yang akan dibuat ini hanya terbatas pada metode bukaan kulit Flat Plan (salah satu metode bukaan dari lambung kapal).
- Macam dan bentuk kurva yang akan dikembangkan sesuai dengan ketentuan metode bukaan kulit Flat Plan yaitu diperuntukkan pada daerah / posisi bilga atau pada bagian yang agak rata dengan sebagian yang agak lengkung.
- Input data yang digunakan dalam program ini adalah data dari gambar body plan manual (Basic Plan) lengkap dengan sambungan pelat (joint plate) dan gambar body plan tersebut sudah dianggap benar.

I. 4 Tujuan dan Manfaat

Tujuan utama pembuatan program dalam Tugas Akhir ini adalah :

- Untuk mengurangi kesalahan-kesalahan masalah ketidakteelitian hasil-hasil bukaan kulit yang dilakukan oleh manusia di lantai mouldloft.
- Untuk mengurangi pekerjaan-pekerjaan manusia pada lantai mouldloft yang luas dan banyak memakan waktu (penggambaran body plan, bukaan kulit, pembuatan rambu-rambu), sehingga pekerjaan-pekerjaan tersebut cukup dilakukan dengan menggunakan komputer.

Tujuan khusus pembuatan program dalam Tugas Akhir ini adalah :

- Untuk mendapatkan bentuk dan ukuran sebenarnya dari pelat yang akan dibentangkan.
- Untuk menghasilkan rambu film yang tepat sehingga dapat dipergunakan untuk proses marking dan cutting dibagian produksi, sedang untuk proses bending hasil bukaan kulit pelat yang didapat tanpa check line (garis kontrol).

Manfaat pembuatan program dalam Tugas Akhir ini adalah :

- Dapat digunakan pada bermacam-macam ukuran kapal dengan tipe yang sama, dimana input yang digunakan diperoleh dari gambar body plan manual, tetapi tidak menutup kemungkinan untuk menggunakan gambar body plan hasil dari program pembuatan body plan.
- Memberikan kemudahan dalam menentukan daerah pelat yang akan dilakukan “replating” dalam pekerjaan reparasi kapal, dimana lengkungan gading yang dipilih sesuai batasan lengkungan gading yang diberikan oleh metode flat plan.

- Hasil bukaan kulit metode flat plan dapat ditransfer ke NC-Cutting dalam bentuk file gambar dengan ekstension dxf sehingga hal ini sangat efisien dalam pembangunan kapal baru.

I.5 Metodologi Penelitian

Dalam mengerjakan Tugas Akhir ini, langkah-langkah dari metodologi penelitian yang digunakan adalah sebagai berikut :

- Melakukan studi literatur tentang metode bukaan kulit yang benar, juga mempelajari sekaligus mempraktekkan proses penggambaran dan pembentangan melalui program AutoLISP yang merupakan fasilitas dari AutoCAD dan memasukkan data melalui text editor dan dibaca oleh Microsoft Visual Basic.
- Pembuatan program komputer dengan langkah-langkah sebagai berikut :
 - Mencari data body plan beserta joint plate (sambungan plat) dari sebuah kapal dimana pembacaan data body plan dilakukan secara manual.
 - Membuat program penggambaran body plan melalui AutoLISP.
 - Memasukkan data-data body plan dari gambar manual melalui text editor, kemudian dibaca dan ditampilkan oleh Microsoft Visual Basic.
 - Menentukan daerah pelat sesuai dengan posisi pelat yang dibentangkan.
 - Menampilkan gambar pelat yang telah terpilih.
 - Program bukaan kulit dengan metode Flat Plan.
 - Membuat program pengukuran dimensi hasil bukaan kulit.
 - Menampilkan pelat dalam bentuk rambu film lengkap dengan identitas pelat, nomor gading beserta simbol-simbol penandaan.

- Melakukan case study dengan menggunakan contoh gambar body plan dari kapal yang ada.
- Melakukan evaluasi terhadap program komputer yang telah dibuat.

I. 6 Sistematika Penulisan

1. **Bab I. Pendahuluan** : berisi dasar pemikiran dari alasan dibuatnya program ,perumusan masalah yang dihadapi,batasan masalah, tujuan dan manfaat dibuatnya program, metodologi atau langkah-langkah pengerjaan yang ditempuh dalam menyelesaikan program.
2. **Bab II. Studi Literatur** : berisi dasar-dasar teori dari pekerjaan mouldloft, dasar-dasar teori pembuatan lines plan dalam skala penuh (1:1), dasar-dasar teori yang diperlukan dalam mendukung proses penggambaran bentangan kulit, cara-cara penggambaran bentangan kulit, dan metode bukaan kulit yang digunakan dalam pembuatan program ini.
3. **Bab III. Konsep Pemrograman Komputer** : berisi teori-teori AutoLISP dan Microsoft Visual Basic yang merupakan teori pendukung pemrograman bukaan kulit beserta penjelasan mengenai langkah-langkah pemrograman bukaan kulit.
4. **Bab IV. Otomatisasi Bukaan Kulit Metode Flat Plan** : berisi pembahasan mengenai akurasi hasil bukaan kulit dan penjelasan kelebihan dan kekurangan program bukaan kulit metode flat plan.
5. **Bab VI. Kesimpulan Dan Saran** : berisi kesimpulan dari hasil bentangan kulit yang telah didapat dari hasil pemrograman dan saran-saran yang diberikan oleh penulis tentang program yang dibuat maupun hasil yang telah didapat.

Daftar Pustaka

Lampiran-Lampiran

BAB II

STUDI LITERATUR

Untuk lebih memahami mouldloft secara keseluruhan dalam proses produksi suatu kapal, maka perlu dipahami pengertian, fungsi, maupun lingkup pekerjaan mouldloft. Dalam bab ini akan dibahas beberapa hal yang berhubungan dengan mouldloft terutama pembahasan masalah pengertian, fungsi, dan lingkup pekerjaan mouldloft sehingga kita dapat mendalami mouldloft sebagai sebuah proses dalam suatu proses produksi kapal.

II.1 Garis Besar Pekerjaan Mouldloft

Untuk diketahui bahwa tidak semua permasalahan dalam desain dan produksi kapal termasuk didalam pekerjaan mouldloft. Oleh karena itu kita harus mengetahui garis besar pekerjaan mouldloft sehingga permasalahan mengenai pekerjaan mouldloft menjadi jelas. Dalam sub bab ini akan diterangkan garis besar pekerjaan mouldloft.

II.1.1 Pengertian

Dalam rangkaian proses pembuatan suatu kapal, terdapat tahap-tahap tertentu yang harus dilalui sebagai pendukung terhadap proses produksi itu sendiri. Tahap-tahap tersebut diantaranya adalah tahap perencanaan. Di dalam tahap ini pada intinya dilakukan perhitungan-perhitungan yang berhubungan dengan ukuran utama dan bentuk serta karakter teknis lainnya yang layak pada suatu kapal. Dalam

melaksanakan suatu perencanaan, data awal dari pemesan (owner) serta peraturan klasifikasi kapal dan peraturan lainnya, dijadikan sebagai bahan acuan untuk mendapatkan hasil perencanaan yang menjamin kelayakan serta memenuhi sifat-sifat teknis dan ekonomis dari suatu kapal. Dengan demikian pada tahap perencanaan ini akan dihasilkan sejumlah output yang bermanfaat untuk proses produksi selanjutnya. Hasil perencanaan tersebut berupa :

- Gambar rencana garis (Lines Plan)
- Gambar rencana umum (General Arrangement)
- Gambar penampang melintang (Midship Section)
- Gambar kurva hidrostatik dan Bonjean
- Dan lain-lain

Berdasarkan output tersebut, dilakukan pengembangan pada tahap perencanaan detail, yang intinya adalah mengembangkan apa yang telah dilakukan pada tahap perencanaan dasar (Base Plan), sehingga menjadi gambar-gambar kerja dengan ukuran serta petunjuk pelaksanaan kerja di lapangan. Selain output dari perencanaan dasar diatas, peraturan klasifikasi dan peraturan-peraturan lainnya serta informasi dari lapangan juga sangat diperlukan sebagai bahan pertimbangan untuk melaksanakan perencanaan pada tahap ini. Dengan demikian pada tahap perencanaan detail ini akan dihasilkan sejumlah output yang berupa:

- Gambar kerja (Working Drawing)
- Gambar perencanaan baja kapal (Steel Plan)
- Gambar konstruksi profil (Construction Profile)
- Gambar pembagian block (Block Division)

- Gambar rencana pemotongan (Cutting Plan)
- Prosedur pengelasan (Welding Procedure)
- Daftar material (Material List)

Setelah proses perencanaan dasar dikembangkan dalam bentuk perencanaan detail, langkah selanjutnya adalah melakukan pengembangan dari hasil rancangan tersebut untuk mendapatkan ukuran dan bentuk sebenarnya dari komponen-komponen kapal yang dilanjutkan dalam bentuk template (rambu film dan kayu) serta marking list (Santoso, 1992).

Pada umumnya kapal memiliki ukuran yang besar dengan sebagian bentuknya yang rumit dan kompleks seperti bagian haluan dan buritan. Hal ini sering merupakan kesulitan pada tahap fabrikasi dalam menentukan bentuk dan ukuran yang tepat, apabila langsung menggunakan gambar dari perencanaan dasar (Base Plan) dengan skala 1 : 50 atau skala 1 : 100 (Santoso, 1992).

Untuk memecahkan persoalan diatas, maka sangat dibutuhkan suatu bangunan khusus untuk mengembangkan output dari perencanaan dasar maupun perencanaan detail, menjadi gambar-gambar dan rambu-rambu dalam skala sebenarnya (1 : 1) dan bangunan tersebut disebut *mouldloft* (Santoso, 1992). Ini merupakan suatu bangunan khusus dengan lantainya terbuat dari kayu (papan) dengan mutu yang baik, misalnya kayu pinus atau jati, dan lain-lain. Pada lantai tersebut digambarkan rencana garis dengan skala penuh (1 : 1), serta melakukan pengembangan (expansion) terhadap kulit lambung dan bagian-bagian kapal lain berdasarkan gambar rencana garis yang telah dibuat. Hal ini dimaksudkan untuk mendapatkan ukuran dan bentuk sebenarnya sehingga mempermudah pekerjaan berikutnya.

II. 1.2 Fungsi dan Peranan

Berdasarkan pengertian diatas, maka dapat dikatakan bahwa fungsi mould loft adalah sebagai berikut :

- a) Menggambarkan bentuk rencana garis pada lantai gambar dengan skala penuh (1:1).
- b) Menggambarkan bukaan/bentangan dari bagian-bagian kapal tertentu serta membuat template (rambu kayu dan film), marking list dan mock up (model).
- c) Memeriksa gambar-gambar detail dan output lainnya sebagai petunjuk pekerjaan di lapangan.

Dengan demikian, pekerjaan mould loft memiliki peranan yang sangat penting untuk menjembatani pekerjaan perencanaan (design) dan pekerjaan produksi (production). Untuk itu ketelitian merupakan faktor yang sangat mendasar dalam pekerjaan di mould loft.

II. 2 Akurasi Lofting

Pekerjaan mould loft merupakan sumber informasi dari pekerjaan konstruksi lambung. Kesalahan-kesalahan yang terjadi pada pekerjaan di mould loft sering ditemukan setelah terjadi masalah-masalah pada pekerjaan fabrikasi atau assembly.

Hal ini akan menjadikan suatu kerugian yang meliputi :

- Kehilangan jam kerja
- Material
- Penjadwalan
- Menurunkan mental para karyawan

Untuk menghindari kesalahan tersebut sedini mungkin, maka perlu ada semacam standart akurasi pada pekerjaan mould loft. Sebagai contoh, berdasarkan pengalaman standart akurasi untuk garis-garis yang digambarkan pada lantai mould loft adalah sebagai berikut (Santoso, 1992) :

- Penggambaran garis normal sepanjang tiga meter, untuk mencapai kelurusan garis, deformasi diperkirakan mencapai 0,...(nol koma, berarti tidak sampai 1 mm).
- Penggambaran garis normal lebih dari tiga meter (± 10 m) toleransi kelurusan mencapai 0 s/d 2 mm.
- Ketebalan garis normal 1 mm dengan skala gambar 1 : 1.
- Penandaan pembuatan garis tegak lurus dan garis datar harus kita awali dari dasar (centre line atau base line).
- Pengambilan ukuran dilakukan pada pertengahan ketebalan garis.

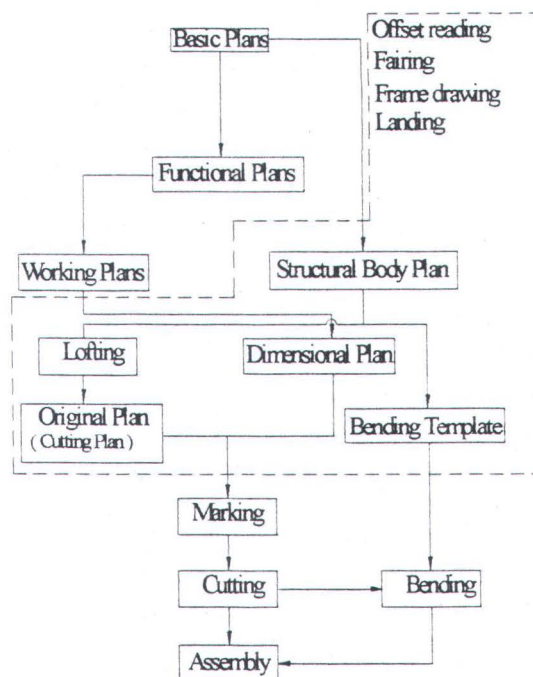
II. 2 Lingkup Pekerjaan Mould Loft

Mouldloft sebagai sumber informasi bagi pekerjaan konstruksi lambung memiliki lingkup pekerjaan yang perlu diperhatikan yaitu mulai input (masukan) yang diterima, kemudian proses pekerjaan terhadap input tersebut sampai output (hasil) dari proses pekerjaan mouldloft. Dalam sub bab ini hal-hal tersebut akan diterangkan.

II. 2.1 Aliran Informasi Dari Pekerjaan Mould Loft

Seperti telah diuraikan sebelumnya bahwa pekerjaan mould loft merupakan pekerjaan lanjutan dari pekerjaan desain konstruksi lambung. Pada prinsipnya

pekerjaan mould loft didasarkan pada penggambaran rencana garis pada lantai gambar dengan skala penuh (1 : 1). Dari gambar ini akan dilakukan pekerjaan lofting seperti : membenteng pelat dan bagian konstruksi lainnya, pembuatan rambu film, pembuatan rambu kayu dan pembuatan marking list. Sebelum melaksanakan pekerjaan, maka ada beberapa gambar serta data lainnya yang merupakan output dari bagian perencanaan yang sudah harus dipersiapkan. Secara sederhana Gambar 2.1 menunjukkan langkah kerja serta input data yang harus diterima mould loft (Santoso, 1992).



Gambar 2.1. Aliran Informasi Mouldloft (Santoso, 1992).

Pada Gambar 2.1 dapat dilihat bahwa :

- Basic Plans adalah perencanaan dasar yang dilakukan oleh bagian perencanaan.
- Functional Plans adalah perencanaan lanjutan yang dilakukan setelah perencanaan dasar (Basic Plans). Pada pokoknya functional plans dibagi atas 2 bagian utama yaitu :

a) Key Plan

Adalah perencanaan lanjutan dari basic plans terutama untuk bidang konstruksi kapal. Pada perencanaan ini lebih banyak melihat faktor-faktor fasilitas serta kemungkinan kerja lapangan, sehingga sering juga dilakukan perubahan atas perencanaan dasar.

b). Yard Plan

Adalah merupakan perencanaan lanjutan dari key plan yang bersifat lebih mendetail. Perencanaan dalam tahap ini lebih berpedoman pada pembuatan kapal secara praktis.

- Working Plans adalah merupakan perencanaan kerja yang siap untuk dilaksanakan dilapangan. Pada tahap ini dilakukan penggambaran secara detail terhadap perencanaan yang telah dilakukan oleh key plan maupun yard plan. Dengan demikian tahap ini sering disebut sebagai working drawing. Gambar-gambar yang telah dibuat disiapkan dengan tanda-tanda yang menunjukkan proses kerja di lapangan.

Dengan demikian, gambar-gambar yang diterima oleh mould loft adalah :

- Basic Plans :
 - Gambar rencana garis (Lines Plan)
 - Gambar rencana umum (General Arrangement)
 - Gambar penampang melintang (Midship Section)
 - Gambar konstruksi profil (Profile Construction)
 - Gambar rencana geladak (Deck Plan)
- Functional Plans :

- Gambar bukaan kulit (Shell Expansion)
- Gambar kerja (Working Drawing)
- Gambar rencana pemotongan (Cutting Plan)
- Gambar pembagian blok (Block Division)
- Daftar material (Material List)
- Prosedur pengelasan (Welding Procedure)

Setelah gambar-gambar tersebut diterima, maka pekerjaan mould loft dapat dilaksanakan. Urutan pekerjaan pada mould loft secara garis besar dapat dibagi menjadi 2 tahap (Santoso, 1992) yaitu :

a. Tahap pekerjaan dasar.

Pekerjaan dalam tahap ini pada pokoknya adalah pembuatan gambar rencana garis sekaligus rencana nyata dari kapal dengan skala sebenarnya pada lantai gambar. Pekerjaan ini merupakan dasar bagi pekerjaan pengembangan selanjutnya. Urutan pekerjaan dalam tahap ini adalah sebagai berikut :

- i) Penggambaran rencana garis.
- ii) Penggambaran struktur profil bagian dalam pada body plan.
- iii) Penggambaran semua lajur pelat dan sambungan-sambungannya pada body plan.
- iv) Pemberian nomor dan penandaan pada gambar yang telah dibuat.
- v) Pemeriksaan kembali terhadap semua hasil pekerjaan yang telah dilakukan diatas.

b. Tahap pekerjaan pengembangan.

Dalam tahap ini dilakukan pekerjaan-pekerjaan yang hasilnya adalah merupakan petunjuk pada pekerjaan di lapangan. Pekerjaan dalam tahap ini meliputi :

- i) Pembuatan bentangan atau bukaan dari bagian-bagian kapal untuk mengetahui ukuran dan bentuk sebenarnya berdasarkan pada gambar rencana garis dan working drawing.
- ii) Pembuatan rambu film dan rambu kayu serta model.
- iii) Pembuatan marking list.

II. 2.2 Scale Lofting

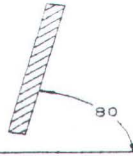
Selama 10 atau 20 tahun belakangan ini beberapa galangan di luar negeri umumnya mempunyai peralatan foto proyeksi dan mesin pemotong otomatis sehingga cara menggambar lines plan menurut ukuran sebenarnya (full size lofting) dengan adanya peralatan-peralatan dapat digantikan dengan scale lofting (Soeyitno, 1990). Scale lofting memperbaiki kondisi-kondisi pekerja dan pembuat mekanisasi dan otomatisasi memungkinkan keduanya dikerjakan dalam mould loft, khususnya di tempat-tempat dimana bagian-bagian dari badan kapal diberi tanda sebelum dipotong atau dikerjakan lebih lanjut. Tenaga yang dikerjakan menurut scale lofting hanya 10% lebih kecil dari full size lofting. Juga ruangan yang diperlukan adalah (25-100)kali lebih kecil dari pada full size lofting, sedang jumlah luas untuk seluruh jumlah pekerjaan mould loft adalah (66 - 75)% lebih kecil dari full size lofting (Soeyitno, 1990). Pemeliharaan dan penggunaannya lebih mudah, demikian pula dengan biaya material yang digunakan didalam mould loft.

Scale template (panel) dapat disimpan dalam waktu yang tidak terbatas, tidak demikian halnya dengan full size lofting. Scale lofting ekonomis mempunyai

pengaruh yang baik dalam industri perkapalan dan umumnya industri perkapalan di luar negeri telah menguasai sepenuhnya penggunaan scale lofting dan optical marking. Semua galangan yang menggunakan scale lofting umumnya selalu dilengkapi dengan peralatan foto proyeksi. Skala yang digunakan adalah 1 : 10 dan 1 : 5 dari ukuran yang sebenarnya. Pada scale lofting umumnya garis dari kapal digambarkan dalam skala diatas 3 buah panel yang terpisah. Gambar halfbreath dan sheer digambarkan diatas panel pertama, sedangkan panel kedua dan ketiga digunakan untuk menggambarkan body plan. Keuntungan dengan menggunakan 3 atau 4 panel secara terpisah untuk scale lofting adalah meja kerja yang digunakan dapat lebih sempit dan sejumlah tenaga kerja dapat mengerjakannya pada waktu bersamaan.

Panel terbuat dari "Dur aluminium", "baja" atau "plywood" yang baik yang tebalnya 4 - 6 mm (Soeyitno, 1990). Permukaannya dibuat dengan deviasi tidak boleh lebih dari 0,5 mm untuk tiap 1 meter panjang. Meja kerja harus diberi permukaan dari kayu dan meja kerja harus cukup kuat untuk mencegah lengkungan apabila 2 orang tenaga kerja bekerja diatasnya. Garis-garis tersebut bilamana perlu dapat kita hapus dan masih dapat dilihat jelas apabila kita menyimpannya dalam waktu yang lama. Jika panel tersebut terbuat dari dur aluminium, maka persiapan untuk mengerjakan permukaannya adalah sebagai berikut :

- i) Panel dimiringkan pada sudut $\pm 80^{\circ}$ dengan horisontal, kemudian debu-debu



dibersihkan dengan kain dan kemudian disemir dengan aseton.

- ii) Lapisan tipis dari minyak disikatka pada permukaan panel dan dikeringkan selama 24 jam selanjutnya digosok dan ditekan dengan udara bertekanan tinggi.
- iii) Empat sampai enam kali lapisan tipis dari cat nitro enamel putih kemudiandisemprotkan dengan alat penyemprot. Tiap-tiap lapisan dikeringkan selama 25 - 30 menit dan setelah lapisan terakhir panel dikeringkan selama 48 jam.
- iv) Panel kemudian diletakkan mendatar dan air dituangkan diatasnya kemudiankembali permukaanya digosok dengan kertas gosok sampai permukaanya menjadi licin.

Semua pekerjaan ini dikerjakan dalam suatu bangunan dengan temperatur udara tak lebih rendah dari 15°C . Untuk menjamin ketelitian pada waktu dibesarkan menurut skala full size lofting garis-garis pada scale lofting lebarnya harus tidak lebih dari 0,1 mm.

II. 2.3 Proses Penggambaran Rencana Garis Dalam Skala Penuh

Penggambaran rencana garis pada lantai gambar dapat dilakukan dengan prosedur sebagai berikut (Santoso, 1992) :

- a) Membagi ruangan atau lantai gambar sehingga dapat digambarkan semua proyeksi yang ada pada gambar rencana garis.

- b) Membuat garis dasar sebagai acuan untuk penggambaran selanjutnya.
- c) Membuat garis air dan garis buttock secara tegak lurus pada proyeksi body plan sebagai jaringan untuk penggambaran garis ordinat, camber, dan profil-profil lainnya.
- d) Membuat garis buttock dan garis-garis ordinat (frame) secara tegak lurus pada proyeksi pandangan setengah lebar kapal sebagai jaringan untuk penggambaran garis air (water line).
- e) Membuat garis air dan garis ordinat (frame) secara tegak lurus pada proyeksi pandangan samping sebagai jaringan untuk penggambaran garis buttock, sheer, deck plan, stem dan stern.
- f) Memasukkan ukuran dari tabel offset ketiga proyeksi tadi (point c, d dan e) untuk setiap titik ordinat yang ada. Hubungkan titik-titik tersebut hingga diperoleh kurva untuk garis air, frame dan buttock.
- g) Setelah semua garis yang ada pada gambar rencana garis dibuat, maka dilakukan pemeriksaan ketiga proyeksi tersebut, sehingga didapatkan ketepatan dan keselarasan garis-garis tersebut. Sesudah itu akan dibuat tabel offset yang baru dan disesuaikan dengan rencana garis yang baru dibuat dengan skala penuh.

Dalam proses penggambaran ini ada beberapa hal yang perlu diperhatikan juga yaitu :

- Apabila ukuran kapal yang akan digambarkan lebih besar dari luas lantai yang tersedia dan untuk menghemat tempat, maka gambar yang akan dibuat dapat ditumpang tindihkan satu sama lainnya ketiga proyeksi yang ada, dengan bagian

pararel middle body-nya tidak perlu digambarkan. Dalam hal ini penggambaran dilakukan dengan warna cat yang berbeda.

- Penggambaran proyeksi pandangan samping dan pandangan atas adalah lebih baik mengacu pada gambar proyeksi body plan. Hal ini dimaksudkan untuk memperkecil kesalahan penggambaran.
- Untuk gambaran nyata dari rencana garis tersebut, maka dibuat juga semua garis dari bagian-bagian konstruksi yang ada didalam lambung kapal dan garis-garis sambungan pelat sesuai petunjuk dari gambar-gambar perencanaan.
- Untuk mencegah terhapusnya garis-garis gambar yang telah dibuat pada lantai gambar, maka lebih baik dilapisi dengan vernis.

II. 2.4 Proses Pembuatan Marking List

Pada marking list ini penggambaran komponen-komponen kapal dilakukan pada kertas kalkir berukuran folio lengkap dengan ukuran dan tanda-tandanya (simbol marking). Marking list ini dibuat dengan maksud untuk mengembangkan komponen konstruksi yang mempunyai bentuk yang teratur, sehingga dapat langsung digambarkan sesuai dengan ukuran dari konstruksinya, dengan mengacu pada gambar kerja dan material list. Dalam pembuatan marking list ini seorang pelaksana dituntut mampu menguasai dan mengerti pembacaan gambar kerja (working drawing).

Pembuatan marking list dapat dibuat dengan prosedur sebagai berikut (Santoso, 1992) :

- a) Menyiapkan kertas kalkir ukuran folio (marking list table) serta semua peralatan dan bahan yang diperlukan.

- b) Menulis identitas komponen konstruksi yang akan dibuat.
- c) Memperhatikan posisi atau letak dan bentuk elemen-elemen apa saja yang menempel pada panel dari bagian atau blok yang akan dibuat listnya tersebut, sedangkan ukuran dapat dilihat pada material list serta standart yang digunakan.
- d) Mencatat pada format bagian atas dari kertas yang akan dibuat marking list hal-hal sebagai berikut : nomor kapal yang sedang dibangun, nomor blok, nomor gambar, material, nama pembuat, nama pemeriksa, dan tanggal dikeluarkan.
- e) Mengkalkulasi ukuran pelat/profil yang akan dibuat marking listnya dari ukuran-ukuran yang ada pada gambar kerja serta daftar material.
- f) Menggambar komponen-komponen dari bagian yang akan dibuat marking listnya.
- g) Memberi ukuran yang sebenarnya, tanda-tanda pengerjaan, tanda-tanda posisi serta simbol-simbol penandaan yang lain.
- h) Memeriksa secara menyeluruh hasil pekerjaan yang dilakukan diatas.

II. 2.5 Proses Pembuatan Rambu Film

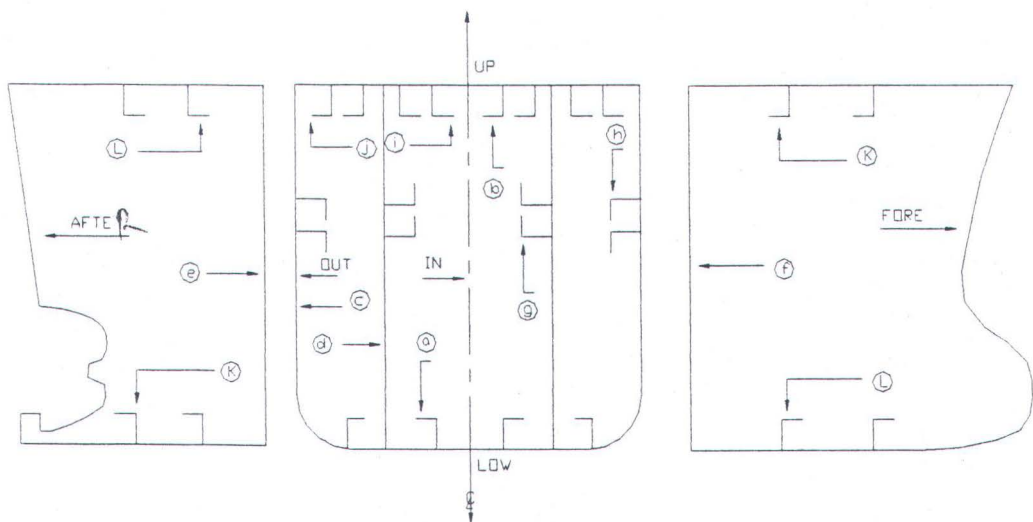
Rambu film merupakan lembaran datar yang terbuat dari lembaran polyester, dan digunakan untuk mencetak ulang bentuk komponen-komponen kapal yang telah dibentangkan dengan metode-metode bukaan kulit yang ada dengan skala penuh. Berikut ini akan dijelaskan prosedur pembuatan rambu film (Santoso, 1992):

- a) Membuat sumbu ordinat dengan garis axisnya adalah garis dasar dari metode bukaan yang digunakan.
- b) Menentukan jarak gading sebenarnya pada garis seperti point a.

- c) Memindahkan ukuran upper seam (batas sambungan atas) dan lower seam (batas sambungan bawah) dari bagian yang dibuka dengan cara mengukur panjang garis pada body plan dengan menggunakan struklat yang dilengkungkan sesuai bentuk gading, kemudian menandai semua garis yang memotong garis gading tersebut.
- d) Memindahkan ukuran-ukuran pada point c pada rambu film dengan garis axis sebagai acuan untuk menentukan upper seam dan lower seam. Pemindahan ukuran tersebut dilakukan dengan cara melengkungkan struklat, dimana pada garis struklat ditekan, kemudian pada upper seam maupun lower seam struklat digoreskan sehingga membekas pada rambu film. Lakukan dengan hal yang sama untuk semua garis gading dan batas sambungan (after butt dan fore butt).
- e) Memindahkan jarak gading sebenarnya pada upper seam dan lower seam sesuai bukaan yang telah dilakukan, dimana gading tengah dijadikan sebagai patokan pemindahan ukuran tersebut. Hal ini dilakukan dengan cara seperti pada point d, hingga berpotongan dengan apa yang telah dibuat pada point d.
- f) Menghubungkan titik potong tersebut, sehingga diperoleh bentuk dari upper seam dan lower seam.
- g) Dengan menggunakan struklat digambarkan bentuk gading, garis sambungan pada rambu film dengan kelengkungan sesuai back sheet yang telah dibuat untuk masing-masing gading. Hal ini dilakukan dengan mengacu pada garis dasar.
- h) Setelah bentuk dari bagian yang dibuka telah dibuat pada rambu film, maka langkah selanjutnya adalah menulis tanda-tanda (marking) dengan mengacu pada working drawing dan material list.
- i) Melakukan pemeriksaan secara menyeluruh terhadap hasil bukaan.

II. 2.6 Simbol-Simbol Penandaan

Untuk menghindari kesalahpahaman pelaksana di bagian produksi dengan pihak perencana maupun mould loft dalam membaca gambar kerja, maka sangat diperlukan standarisasi dari simbol-simbol penandaan. Standarisasi dari simbol-simbol penandaan tersebut memberikan petunjuk tentang : arah penandaan, letak komponen yang dibuat, ketebalan pelat, kelonggaran ukuran dari tepi pelat dan lain-lain. Untuk lebih jelasnya, simbol-simbol penandaan yang digunakan pada contoh mould loft bagian belakang, sekat dan depan kapal (Gambar 2.2) adalah sebagai berikut :



Gambar 2.2. Simbol Arah Penandaan.

Contoh simbol arah penandaan untuk pelat dan profil:

Marking pelat

- a) Up-ma tanda diatas
- b) Low-ma tanda dibawah
- c) In-ma tanda didalam
- d) Out-ma tanda diluar

- e) Aft-ma tanda ke belakang
- f) Fore-ma tanda ke depan

Marking profil

- g) Up-side flange ke atas
- h) Low-side flange kebawah
- i) In-side flange ke dalam
- j) Out-side flange ke luar
- k) Aft-side flange ke belakang
- l) Fore-side flange ke depan

II. 3 Teori-Teori Dasar Penggambaran Bukaan

Sebelum memulai penggambaran bukaan kulit, maka langkah awal yang perlu diperhatikan adalah mengerti tentang dasar-dasar dari penggambaran bukaan kulit sehingga akan memberi kejelasan baik dalam penggambaran bukaan maupun dalam perhitungan bukaan. Berikut ini penjelasan teori matematika yang merupakan teori dasar penggambaran bukaan dan penjelasan mengenai konstruksi geometris praktis yang dapat mendukung penggambaran bukaan.

II. 3.1 Umum

Pada umumnya benda yang terbuat dari pelat tidak selamanya memiliki bentuk yang datar/rata, tetapi memiliki bentuk yang bervariasi sesuai yang diinginkan oleh perancang dan tergantung dari fungsi benda itu sendiri. Hal ini sama juga untuk konstruksi lambung kapal yang terbuat dari pelat baja dengan bentuk yang bervariasi,

dimana untuk mendapatkan bentuk yang sebenarnya dari pelat tersebut, maka terlebih dahulu harus dibuka ke dalam bentuk yang datar. Hal ini beralasan, mengingat bahwa pelat yang dipesan adalah dalam bentuk yang datar. Untuk itu, maka konstruksi dari lambung kapal dengan bentuk yang bervariasi tersebut harus dapat dibuka atau dikembangkan ke dalam bentuk yang datar untuk mendapatkan ukuran dan bentuk sebenarnya. Setelah itu baru difabrikasi sesuai yang diinginkan.

Untuk menggambarkan bukaan, terlebih dahulu diketahui dan dipahami teori-teori dasar yang pada prinsipnya sangat membantu dalam menentukan ukuran dan bentuk sebenarnya. Hal ini dapat dilakukan baik melalui pengembangan dari gambar dengan proyeksi-proyeksi tertentu, maupun juga melalui perhitungan-perhitungan secara matematik. Teori-teori dasar tersebut dapat dijelaskan sebagai berikut.

II. 3.2 Matematika dan Konstruksi Geometris Praktis

Rumus-rumus matematika geometri merupakan rumus-rumus praktis yang didasarkan pada pengkajian dan pengalaman yang pernah dilakukan. Hal ini beralasan karena pekerjaan bukaan adalah merupakan pekerjaan yang dilakukan terhadap suatu konstruksi yang memiliki *ruang*. Sehingga dari alasan ini maka pemahaman tentang matematika geometri perlu dipelajari.

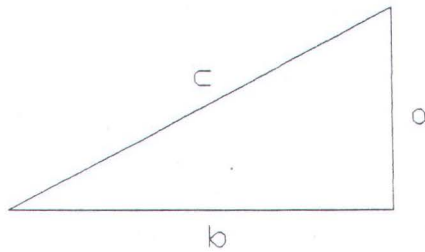
Seperti telah diuraikan sebelumnya bahwa tujuan dari bukaan adalah untuk mendapatkan ukuran dan bentuk sebenarnya. Dengan demikian dalam melakukan hal tersebut sering dibuat perhitungan-perhitungan untuk mendapatkan ukuran panjang, luas atau volume baik pada bentuk segitiga, lingkaran maupun bentuk geometri lainnya. Rumus-rumus matematika geometri praktis tersebut antara lain adalah :

a. Segitiga

$c^2 = a^2 + b^2$, dimana c = panjang sisi miring

b = panjang alas

a = panjang sisi tegak



Gambar 2.3. Segitiga.

Luas (A) = $\frac{1}{2} \cdot a \cdot b$; Keliling (P) = $a + b + c$

b. Lingkaran.

Didalam sebuah lingkaran, perbandingan pada keliling lingkaran dengan garis tengah ditunjukkan dengan huruf Yunani (pi). Nilai pi pada 4 desimal adalah 3,1416.

Beberapa persamaan fungsi pi adalah sebagai berikut :

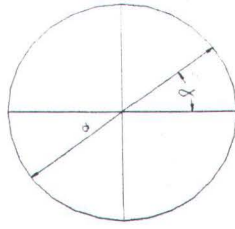
$$\pi = 3,1416 \qquad \pi^2 = 9,8969$$

$$1/\pi = 0,3183 \qquad 1/\pi^2 = 0,1013$$

$$2\pi = 6,2832 \qquad \sqrt{\pi} = 1,772$$

$$1/2\pi = 0,1591 \qquad 1/\sqrt{\pi} = 0,564$$

$$\pi/4 = 0,7854 \qquad 4\pi/3 = 4,1888$$



Gambar 2.4. Lingkaran.

$$\text{Luas lingkaran (A)} = \pi \cdot r^2 = \pi/4 \cdot d^2$$

dimana : $r = d/2$, jari-jari

$d = \text{diameter (garis tengah)}$

$$\text{Keliling lingkaran (C)} = 2\pi r$$

$$= \pi d, \text{ dimana ; } d = C/\pi$$

$$\text{Bidang lingkaran ; } l = r \cdot \alpha/180 \quad ; \quad \text{Luas (A)} = \alpha/360 \times \pi r^2$$

c. Bujur Telur.

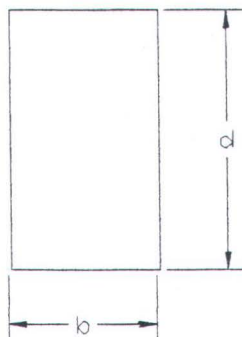
$$\text{Luas (A)} = \pi/4 \cdot d^2 + b \cdot d$$

$$\text{Keliling (P)} = \pi d + 2b$$

$$\text{dimana : } \pi = 3,1416$$

$b = \text{panjang bidang datar}$

$d = \text{lebar}$



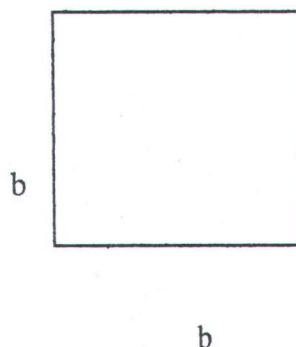
Gambar 2.5. Bujur Telur.

d. Bujur Sangkar.

$$\text{Luas (A)} = b^2$$

$$\text{Keliling (P)} = 4b$$

dimana : b = panjang sisi



Gambar 2.6. Bujur Sangkar.

e. Persegi Panjang.

$$\text{Luas (A)} = b \cdot h, \quad A/h = b ; A/b = h$$

$$\text{Keliling (P)} = 2 (b + h)$$

$$P/2 - b = h ; P/2 - h = b$$

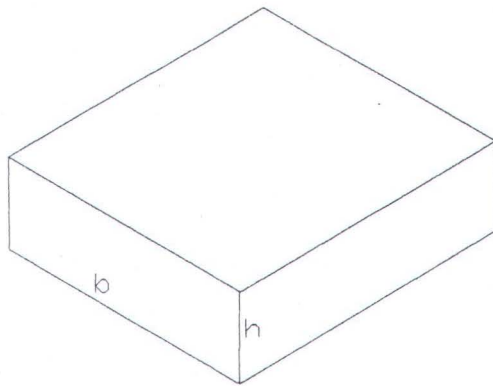


Gambar 2.7. Persegi Panjang.

f. Kotak.

$$\text{Volume (V)} = b \cdot h \cdot l$$

$$\text{Keliling kotak (P)} = 2 (lh + bl + hb)$$

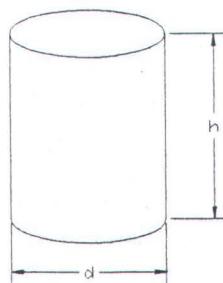


Gambar 2.8. Kotak.

g. Silinder

$$\text{Volume (V)} = \pi/4 \cdot d^2 \cdot h$$

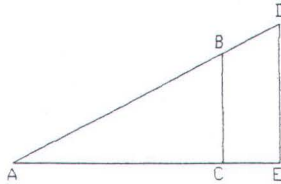
$$\text{Luas permukaan silinder (S)} = \pi \cdot d \cdot h$$



Gambar 2.9. Silinder.

h. Perbandingan.

Rumus perbandingan berikut adalah untuk mendapatkan panjang salah satu sisi sejajar pada segitiga. Misalnya; pada segitiga ADE dan ABC (lihat Gambar 2.10).



Gambar 2.10. Perbandingan.

maka didapatkan ; $AB/AD = AC/AE$

$BC/AC = DE/AE$; $BC/DE = AC/AE$

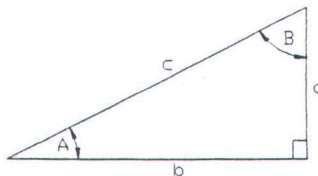
Misalnya ; $AB/AD = AC/AE$ maka $AB = AD (AC/AE)$

$$AD = AB/AC/AE$$

$$AC = AB/AD \cdot AE$$

$$AE = AC/ AB/AD$$

i. Trigonometri Sudut.



Gambar 2.11. Trigonometri Sudut.

$$\sin A = a/c \ ; \ \sin B = b/c$$

$$\cos A = b/c \ ; \ \cos B = a/c$$

$$\tan A = a/b \ ; \ \tan B = b/a$$

II. 3.3 Konstruksi Geometri Dasar

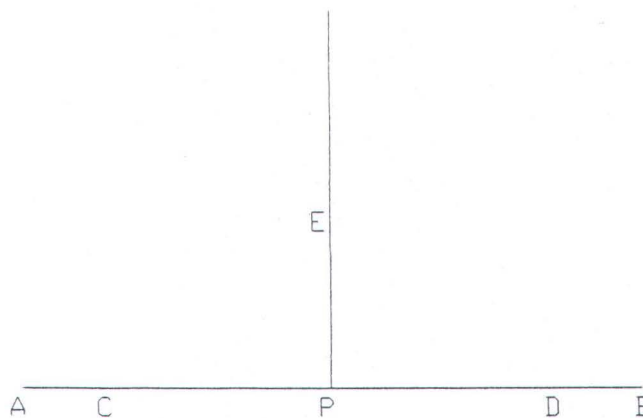
a. Membuat garis tegak lurus dari titik P terhadap garis AB (lihat Gambar 2.12).

Caranya :

⇒ Jangkauan dari titik P busur-busur lingkaran yang memotong garis AB, maka diperoleh titik CD.

⇒ Dengan jarak CD, jangkakan masing-masing dari titik C & D sehingga berpotongan dititik E.

⇒ Hubungkan titik P dengan titik E, maka diperoleh garis tegak lurus yang diinginkan.



Gambar 2.12. Penggambaran Garis Tegak Lurus.

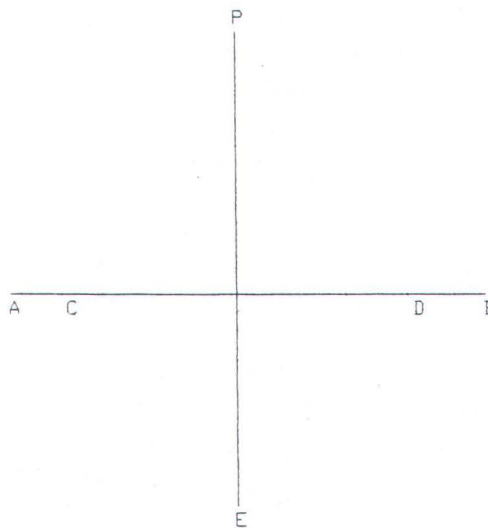
b. Membuat garis tegak lurus dari satu titik yang telah ditentukan (titik P) terhadap suatu garis (garis AB) (lihat Gambar 2.13).

Caranya :

⇒ Dari titik P buatlah busur lingkaran hingga memotong garis AB, maka didapat titik CD.

⇒ Jangkakan masing-masing dari titik C dan titik D busur lingkaran hingga berpotongan, dan diperoleh titik E.

⇒ Hubungkanlah titik P dengan titik E, maka didapat garis tegak lurus yang dimaksudkan.



Gambar 2.13. Penggambaran Garis Tegak Lurus.

c. Membuat garis tegak lurus dari salah satu ujung garis (lihat Gambar 2.14).

Caranya :

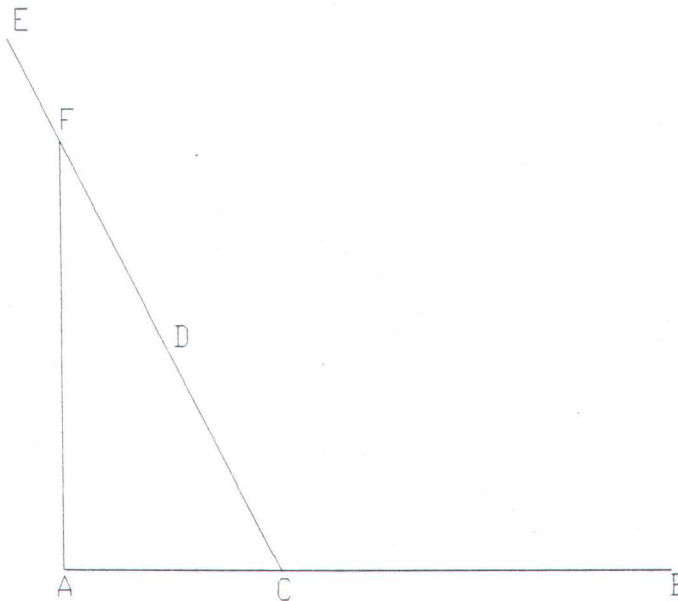
⇒ Tentukanlah salah satu titik pada garis AB, yaitu titik C.

⇒ Buatlah busur lingkaran masing-masing dari titik A & C, sehingga berpotongan dan didapat titik D.

⇒ Tarik garis dari titik C dan melewati titik D (Garis CE).

⇒ Pindahkanlah jarak CD terhadap garis CE dan diukur dari titik D, maka didapat titik F.

⇒ Hubungkanlah titik A dengan titik E, maka didapat garis tegak lurus yang dimaksudkan.



Gambar 2.14. Penggambaran Garis Tegak Lurus.

d. Membuat garis tegak lurus dari salah satu ujung garis AB dengan titik D sebagai acuan (lihat Gambar 2.15).

Caranya :

⇒ Buatlah busur lingkaran yang memotong titik A dan garis AB dengan titik D sebagai pusat, maka didapat titik C.

⇒ Hubungkanlah titik C dan D dengan perpanjangan garis yang memotong busur lingkaran (di titik E).

⇒ Hubungkanlah titik A dengan titik E, maka didapat garis tegak lurus yang dimaksudkan.



Gambar 2.15. Penggambaran Garis Tegak Lurus.

e. Membuat garis tegak lurus dan membagi garis CD menjadi dua bagian yang sama besar (lihat Gambar 2.16).

Caranya :

⇒ Prinsipnya sama dengan Gambar 2.12.

⇒ Untuk memeriksa kebenarannya dibuat lingkaran dengan titik perpotongan G sebagai pusat.

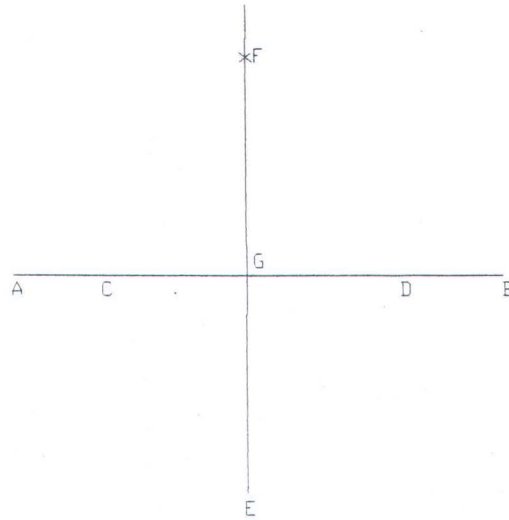
f. Membuat garis tegak lurus terhadap garis yang menghubungkan 2 buah titik yang diberikan (titik A dan B) (lihat Gambar 2.17).

Caranya :

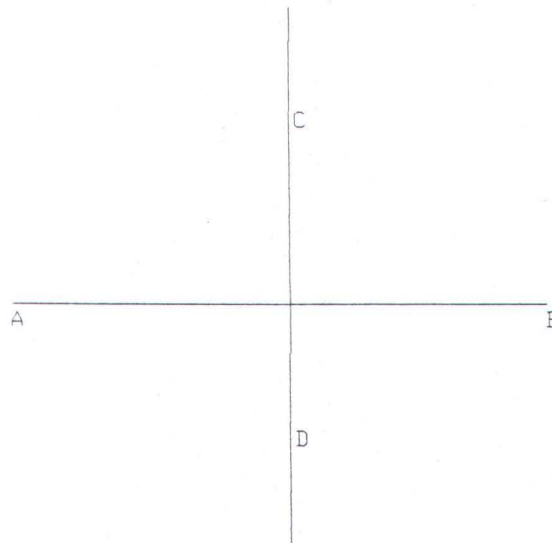
⇒ Hubungkan titik A dan B.

⇒ Buatlah busur lingkaran dengan titik A dan B sebagai pusat, maka didapat perpotongannya pada titik C dan titik D.

Hubungkanlah titik C dan D, maka didapat garis tegak lurus yang dimaksudkan.



Gambar 2.16. Penggambaran Garis Tegak Lurus.

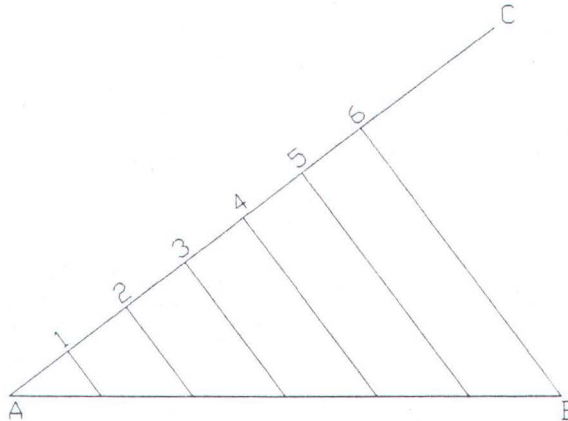


Gambar 2.17. Penggambaran Garis Tegak Lurus.

g. Membagi satu garis atas beberapa bagian yang sama (lihat Gambar 2.18).

Caranya :

- ⇒ Buatlah suatu garis dengan panjang sembarang (garis AB).
- ⇒ Dari salah satu titik (misal titik A), buatlah garis yang lain dengan sudut sembarang (garis AC).
- ⇒ Dari titik A, bagilah garis AC menjadi beberapa bagian yang sama dengan menggunakan jangka (misalnya 6 bagian yang diinginkan), maka didapat titik 1 s/d 6.
- ⇒ Hubungkanlah titik ke-6 dengan titik B.
- ⇒ Buatlah garis yang sejajar dengan garis 6B dari titik 5 sampai 1 dengan memotong garis AB, maka diperoleh pembagian garis yang dimaksudkan.

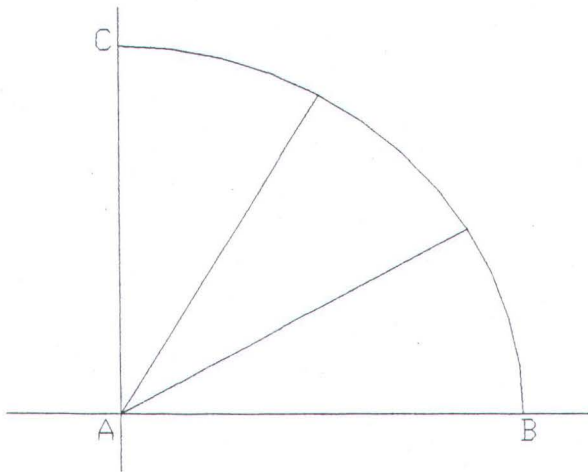


Gambar 2.18. Pembagian Garis Menjadi Beberapa Bagian Yang Sama.

h. Membagi $\frac{1}{4}$ busur lingkaran menjadi 3 bagian yang sama (lihat Gambar 2.19).

Caranya :

- ⇒ Buatlah garis AB dan garis AC yang tegak lurus garis AB pada titik A.
- ⇒ Buatlah busur lingkaran dari titik A sebagai pusat dengan AB sebagai jari-jari, yang memotong garis AC pada titik C, maka didapat busur BC.
- ⇒ Dengan jari-jari yang sama, buatlah busur lingkaran yang memotong busur BC dengan titik B dan C sebagai pusat, maka didapat pembagian busur lingkaran yang dimaksudkan.



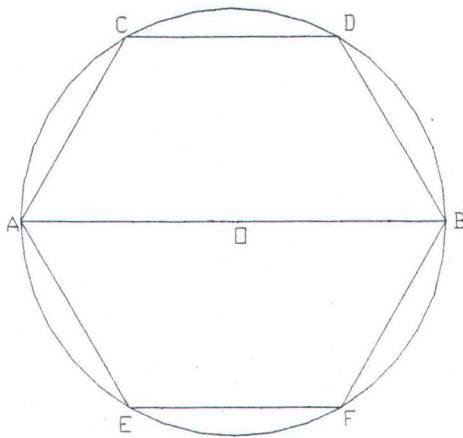
Gambar 2.19. Pembagian $\frac{1}{4}$ Lingkaran Menjadi 3 Bagian Yang Sama.

i. Membuat segi enam beraturan (lihat Gambar 2.20).

Caranya :

- ⇒ Tentukanlah panjang sisi segi enam yang dikehendaki.
- ⇒ Buatlah lingkaran dengan jari-jari yang panjangnya sama dengan panjang sisi segi enam yang dikehendaki, dengan titik O sebagai pusat.
- ⇒ Ambil sembarang titik (misal titik A) dan buatlah garis dari titik A melewati titik O dan memotong titik B.

- ⇒ Masing-masing dari titik A dan B, jangkakan pada lingkaran yang telah dibuat dengan jari-jari sesuai sisi dari segi enam yang dikehendaki hingga memotong titik O, maka didapat titik CDE dan F.
- ⇒ Hubungkan titik AC, CD, DB, BF, FE, dan EA, maka didapat segi enam beraturan yang dikehendaki.



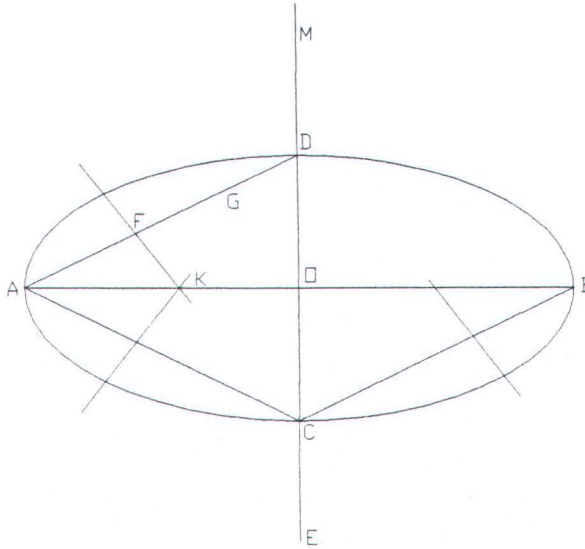
Gambar 2.20. Membuat Segi Enam Beraturan.

j. Membuat ellipsis (lihat Gambar 2.21).

Caranya :

- ⇒ Buatlah sumbu AB dan CD saling tegak lurus dan berpotongan sama panjang dititik O.
- ⇒ Buatlah busur lingkaran dengan jari-jari OA dan OE, sehingga didapat titik M dan E.
- ⇒ Dengan jari-jari sepanjang DM, buatlah lingkaran dengan titik pusat D.
- ⇒ Hubungkan titik A dengan titik D, maka didapat titik G.

⇒ Buatlah garis tegak lurus dengan membagi jarak AG menjadi 2 bagian yang sama, maka didapat titik F dan titik K, dimana titik F adalah titik pusat busur KD dan titik K adalah titik pusat busur AK.

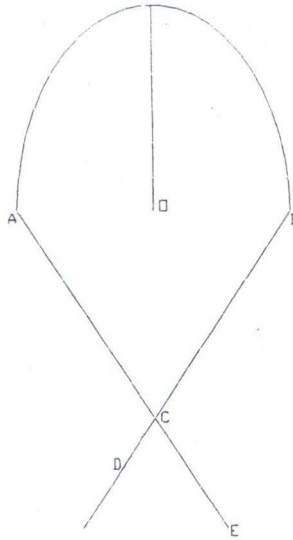


Gambar 2.21. Membuat Elips.

k. Membuat lingkaran bulat telur (lihat Gambar 2.22).

Caranya :

- ⇒ Tentukanlah garis AB dengan titik O sebagai pusat lingkaran, dengan jari-jari OA atau OB.
- ⇒ Buat garis OC tegak lurus AB.
- ⇒ Buat garis AE dan BF yang memotong titik C.
- ⇒ Jangkakan busur AD dengan pusat di B, jangkakan juga busur DH dengan pusat di titik C, maka didapat lingkaran bulat telur yang dimaksudkan.



Gambar 2.22. Membuat Lingkaran Bulat Telur.

II. 3.4 Dasar Penggambaran Bukaannya

Dasar penggambaran yang paling banyak digunakan untuk menggambarkan suatu bukaan adalah proyeksi dan basic expansion (pembentangan dasar). Pengertian mengenai kedua hal tersebut sangat perlu diperhatikan karena sebagian besar teknik-teknik bentangan dari metode-metode bukaan yang digunakan di mouldloft menggunakan proyeksi dan basic expansion (Santoso, 1992). Mengenai kedua hal tersebut dapat dijelaskan pada Lampiran A.1 (Proyeksi) dan Lampiran A.2 (Bukaan Dasar).

II. 4 Penggambaran Bentangan

Dalam penggambaran bentangan perlu dipelajari metode-metode penggambaran bentangan yang sesuai dengan bentuk komponen yang akan dibuat bentangannya. Pemilihan metode penggambaran yang benar akan memberikan bentuk

dan ukuran yang benar sehingga dapat mengurangi “reworks” pada tahap fabrikasi maupun assembly (Soeyitno, 1994). Oleh karena itu dalam sub bab ini akan diterangkan beberapa metode penggambaran bentangan sehingga akan memperjelas penggunaannya.

II. 4.1 Umum

Kapal merupakan konstruksi engineering yang dibangun dalam 3 (tiga) dimensi, dimana bentuk dari kapal merupakan komponen yang rumit dan sulit untuk dibuat atau untuk mengetahui ukuran dan bentuk yang sebenarnya dalam proyeksi 2 (dua) dimensi atau kedalam bentuk yang datar. Untuk memecahkan persoalan ini, perlu dilakukan suatu perencanaan bukaan dengan metode-metode tertentu.

Metode-metode bukaan dari lambung kapal ada beberapa macam, dimana masing-masing mempunyai tingkat kesulitan tertentu serta digunakan menurut macam dan bentuk yang akan dikembangkan. Metode-metode bukaan tersebut adalah (Santoso, 1992) :

- Metode Base Line.
- Metode Roll Line.
- Metode Flat Plan.
- Metode Squaring.
- Metode Stem/Stern.

Disamping metode-metode tersebut diatas, ada juga metode-metode tambahan yang berguna untuk pengerjaan bukaan dari konstruksi kapal lainnya. Metode-metode tambahan tersebut antara lain (Santoso, 1992) :

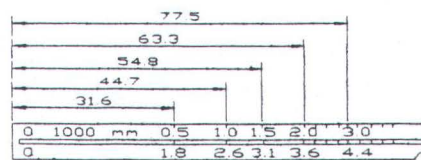
- Metode pengembangan pelat margin.
- Metode pengembangan profil.
- Metode pengembangan frame.

Dalam studi literatur ini penulis hanya menguraikan metode bukaan kulit yang akan dikembangkan secara otomatis dengan menggunakan program komputer. Penjelasan mengenai metode-metode bukaan kulit lainnya dapat dibaca (Sentosa, 1992).

II. 4.2 Menggambarkan Bentangan Dari Suatu Garis.

Panjang sebenarnya dari suatu garis lengkung yang dapat dicari dengan metode-metode yang dapat dipakai disebut “panjang bentangan” dari garis tersebut. Dalam pembuatan kapal, panjang bentangan suatu garis (Gambar 2.23) sangat diperlukan, baik panjang bentangan dari garis yang terletak dalam suatu bidang maupun dalam ruang.

Bagian-bagian dari badan kapal yang terletak pada bidang yang sejajar dengan bidang proyeksi, maka bentuk sebenarnya dapat dilihat pada body plan seperti gading, balok, floor bilge bracket dan lain-lainnya. Ukuran dan bentuknya dapat langsung ditentukan dari body plan.



Gambar 2.23. Garis Untuk Menentukan Panjang Bentangan.

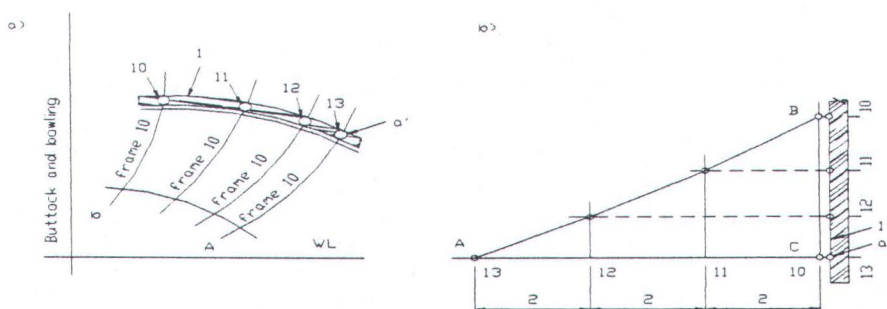
Sebagai contoh, untuk mendapatkan bentangan dari suatu gading dapat dikerjakan dengan membengkokkan papan kayu sekeliling garis sebelah dalam dan gading yang bersangkutan pada body plan dan seluruh titik-titik dimana gading memotong geladak, platform dan bagian-bagian lain diberi tanda diatas kayu tadi. Umumnya untuk menjaga agar papan dapat melengkung menurut bentuk pada garis yang dikehendaki umumnya digunakan pemberat. Papan yang telah diberi tanda ini kemudian dikirimkan ke bengkel untuk dikerjakan. Bagian-bagian yang tidak terletak pada bidang yang sejajar dengan bidang proyeksi sebagai contoh ; pelat kulit, pelat geladak atau penumpu, digambarkan diatas body plan menurut proyeksinya pada bidang tengah kapal. Untuk menentukan panjang bagian-bagian ini dipergunakan metode-metode pendekatan untuk mendapatkan panjang bentangannya.

Sebagai contoh :

Berikut ini adalah cara untuk menentukan bentangan dari "upper strake edge" dari kulit kapal yang membentang dari gading no 10 sampai 13.

- Plotting bentangan dari upper strake edge pada sebuah pelat kulit :

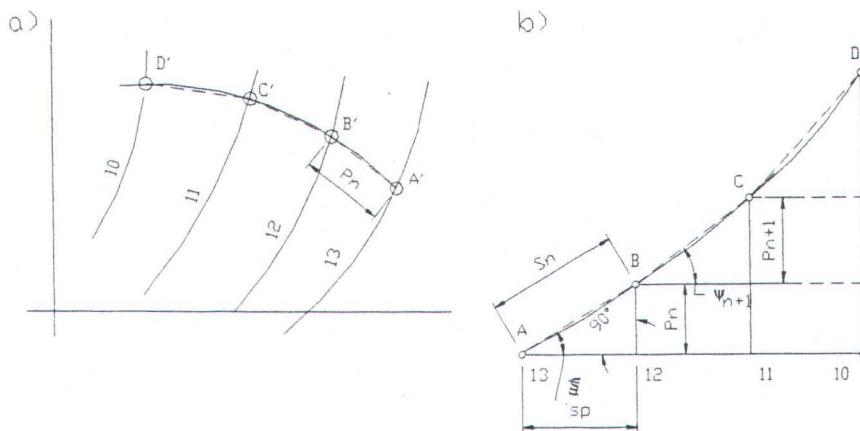
- a) Proyeksi dari pelat (sisi yang ditandai dengan bayangan) pada body plan.
- b) Plotting bentangan dari strake pada pelat AB ini.



Gambar 2.24. Bentangan Upper Strake Edge Pada Pelat Kulit.

- Menentukan bentangan dari kurva A'B'C'D dengan pengertian dari sebuah peraturan bentangan :

- a) Proyeksi dari garis pada lambung.
- b) Bentangan garis.



Gambar 2.25. Proyeksi Garis Pada Lambung.

II. 4.3 Menggambarkan Bukaannya Pelat.

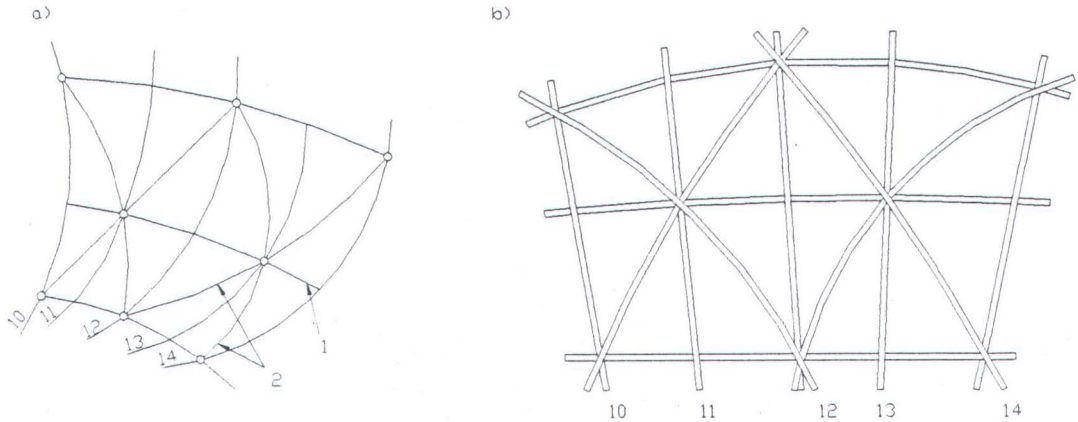
Ada bermacam-macam cara untuk menggambarkan bukaan pelat dari body plan. Untuk menggambarkan bukaan pelat ada dua golongan besar yaitu (Soeyitno, 1990) :

- Menggambarkan bukaan pelat dengan pertolongan garis-garis diatas pelat tersebut dan memotong seluruh gadingnya dengan pertolongan garis konstruksi.
- Dengan menggunakan cara diagonal atau cara Belanda (Dutch Methode).

Cara Diagonal

Keuntungan dengan pemakaian cara ini adalah :

- Pengerjaanya sederhana.
- Ketelitiannya cukup.



Gambar 2.26. Metode Pengembangan Diagonal: (a) proyeksi pelat pada body; (b) batten template.

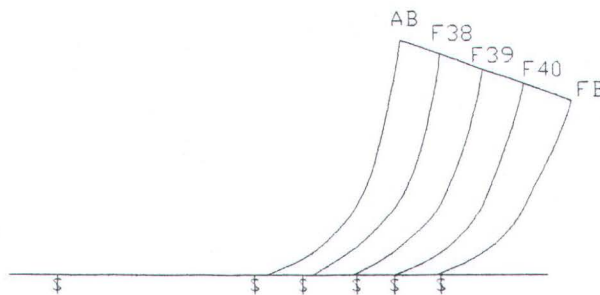
Sebuah garis pertolongan digambarkan diatas pelat pada body plan. Garis ini diusahakan agar melalui pertengahan dari lebar pelat dan kalau mungkin memotong titik-titik maksimum dari chamber frame.

Garis-garis diagonal : digambarkan untuk setiap 2 jarak gading. Diagonal tersebut melalui titik dimana gading yang ditengah memotong garis pertolongan. Kemudian digambarkan panjang sebenarnya dari garis-garis diagonal lengkap dengan titik-titik potongnya pada papan-papan atau kayu yang terpisah-pisah atau sendiri-sendiri. Juga panjang sebenarnya dari gading-gading kita gambarkan lengkap dengan titik-titik

potongnya. Papan-papan yang digunakan harus tipis dan tidak boleh lebih tebal dari pelat kulit. Jika papan-papan tersebut telah dikerjakan seperti diatas kemudian papan-papan itu kita hubungkan dengan cara dipalu dan titik potong tadi harus dibuat berimpit atau sama lain. Mal kayu ini mempunyai kelemahan karena jika disimpan dapat mengerut atau mengembang.

II. 4.4 Metode Flat Plan

Metode ini digunakan untuk bentangan pada posisi bilga atau bagian yang agak rata dengan sebagian yang agak lengkung (lihat Gambar 2.27).



Gambar 2.27. Bentuk Kurva Flat Plan.

Prosedur pengembangan :

- Menandai bagian yang akan dibentangkan pada body plan.
- Memproyeksikan bagian yang akan dikembangkan tersebut ke garis gading dengan jarak sebenarnya (proyeksikan upper seam, lower seam dan butt joint).
- Membuatkan garis tegak lurus terhadap garis proyeksi upper seam yang melalui titik potong garis tersebut dengan garis gading.

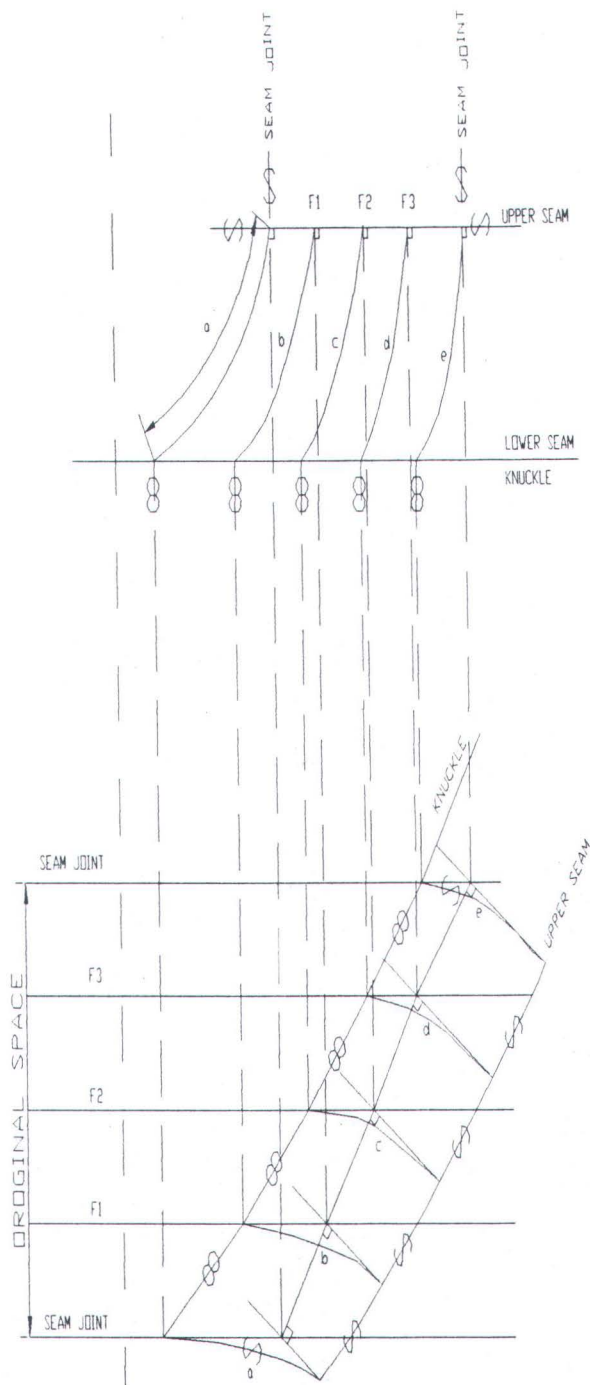
- d. Mengukur panjang gading (frame) pada body plan dengan bantuan struklat, kemudian ukuran tersebut dipindahkan ke original frame space (jarak gading sebenarnya) dan digoreskan hingga memotong garis tegak lurus yang telah dibuat sebelumnya (point c).
- e. Menghubungkan titik-titik perpotongan tersebut yang merupakan batas sebenarnya dari upper seam.
- f. Menghubungkan juga titik-titik dari proyeksi butt joint.

Dengan demikian maka ditemukanlah ukuran dan bentuk yang sebenarnya dari bagian kulit lambung yang dibentangkan. Sebagai catatan bahwa bukaan kulit pelat yang menggunakan Metode Flat Plan memiliki beberapa kondisi yang perlu diperhatikan dilapangan selain tergantung pada bentuk kurva gading yaitu :

- Luasan bagian (flat) lebih besar dari 40% dalam setiap pelat.
- Bagian diatas tersebut menerima beban "bending" pada waktu pembentukan bagaian lengkung.

Sehingga apabila ketentuan-ketentuan diatas diperhatikan maka hasil bukaan kulit pelat yang diinginkan dapat terlaksana sesuai kebutuhan pelat dilapangan. Gambar 2.28. Metode Flat Plan menunjukkan cara-cara bukaan kulit seperti yang telah diterangkan pada point-point diatas dimana cara penggunaan sistim penggambaran proyeksi lebih banyak digunakan untuk mendapatkan hasil bukaan kulit.

METODE FLAT PLAN



Gambar 2.28. Metode Flat Plan.

BAB III

KONSEP PEMROGRAMAN KOMPUTER

Konsep pemrograman komputer mempunyai tujuan untuk memberikan penjelasan mengenai proses pemrograman dan dasar-dasar pemrograman komputer yang dapat menunjang proses pemrograman. Berdasarkan tujuan tersebut maka pada bab ini akan diterangkan teori-teori dasar pemrograman dan penjelasan detail mengenai proses pemrograman.

III. 1 Teori-Teori Pemrograman

Teori-teori pendukung dari pembuatan program komputer yang dikumpulkan merupakan teori-teori yang dapat memberikan kemudahan bagi pengguna (user) dalam mengerti proses pembuatan program komputer. Teori-teori pendukung pemrograman dalam hal ini adalah AutoLISP dan Microsoft Visual Basic.

III. 1.1 AutoLISP

AutoLISP adalah implementasi dari bahasa pemrograman LISP yang merupakan bagian integral dari paket AutoCAD (Smith, 1992). AutoLISP diperuntukkan user dan developer AutoCAD dalam membuat macro dan fungsi-fungsi bahasa tingkat tinggi (high-level language) yang powerful untuk membuat dan mengakses aplikasi grafis. AutoLISP termasuk bahasa yang mudah dipelajari dan sangat fleksibel.

Kenapa LISP ?

AutoLISP sebagai bahasa pertama yang digunakan oleh AutoCAD memiliki beberapa alasan :

- Relatif mudah dipelajari dan fleksibel.
- Salah satu bahasa yang dipilih untuk riset dan pengembangan kecerdasan buatan (artificial intelligence), termasuk sistem pakar (expert systems).
- Karena LISP memiliki sintak sederhana, interpreter LISP mudah dalam mengimplementasi dan membutuhkan sedikit memori.
- LISP sangat baik dalam mengerjakan obyek yang heterogen dengan variasi ukuran grup dan tipe informasi sistem CAD seperti AutoCAD.

Sebagai pengenalan AutoCAD Development System (ADS) pada release 11 ditambahkan bahasa pemrograman alternatif yaitu bahasa C untuk mengembangkan berbagai aplikasi yang lebih kompleks. Namun demikian Autodesk (pembuat AutoCAD) masih mendukung pemakaian AutoLISP. Adapun tipe data, fungsi-fungsi, entity dan devices access dalam AutoLISP diuraikan dalam Lampiran B.1.1.

III. 1.2 Microsoft Visual Basic

Microsoft Visual Basic (Programmer's Guide, 1995) merupakan cara tercepat dan termudah untuk menciptakan aplikasi yang powerful untuk sistem operasi Microsoft Windows. Sistem Program Visual Basic dapat digunakan untuk aplikasi-aplikasi seperti untuk membuat graphical user interface (GUI). Visual Basic membantu untuk lebih produktif dengan penyediaan alat-alat untuk aspek-aspek yang berbeda dari perkembangan GUI. Software ini menciptakan graphical user interface untuk program aplikasi dengan obyek gambar dalam cara grafis. Dengan mengatur

properties pada obyek-obyek ini maka dapat menyempurnakan penampilan dan perilaku. Kemudian untuk membuat interface ini sehingga dapat berkomunikasi dengan user maka dilakukan penulisan kode yang merespon kejadian-kejadian yang terjadi dalam interface tersebut.

Dengan menggunakan Visual Basic, maka hal-hal berikut ini dapat diciptakan secara lebih, penuh dengan aplikasi-aplikasi yang istimewa :

- Akses data memberikan peluang untuk menciptakan aplikasi-aplikasi database untuk format database yang paling populer.
- OLE memberikan kemudahan untuk menggunakan fungsional yang disediakan oleh aplikasi lain, seperti Microsoft Word untuk Windows word processor, Microsoft Excel spreadsheet, dan Microsoft Project business project planning system.
- Aplikasi yang telah selesai adalah berbentuk file dengan ekstension . EXE yang menggunakan sebuah run-time dynamic-link library (DLL) yang dapat didistribusikan secara bebas.

III. 1.2.1 Visual Basic Control

Visual Basic Toolbox berisi tool yang digunakan untuk menggambar kontrol pada form. Setiap tool dalam toolbox mewakili sebuah kontrol. Beberapa tool beserta kegunaannya dijabarkan dalam Lampiran B.1.2.

III 1.2.2 Dasar-Dasar Pemrograman

Terdiri dari :

- Struktur dari sebuah aplikasi Visual Basic.
- Variabel
- Modul
- Prosedur

⇒ Struktur dari sebuah aplikasi Visual Basic

Aplikasi Visual Basic dapat berisi beberapa tipe-tipe yang berbeda dari file :

- Form module (.FRM) yang berisi elemen visual dari sebuah form, termasuk semua kontrol pada form dan kode basic yang digabungkan dengan form tersebut.
- Standart (.BAS) dan class (.CLS) modul yang berisi kode Basic.
- Custom control (.VBX atau .OCX) termasuk kontrol khusus, sebagaimana versi selanjutnya dari standart control.
- Sebuah resource file tunggal (.RES) yang berisi string dan bitmaps yang digunakan oleh aplikasi Visual Basic.

⇒ Variabel

Visual Basic, seperti kebanyakan bahasa pemrograman, menggunakan variabel untuk menyimpan nilai. Variabel memiliki sebuah *name* dan sebuah *data type*.

Cara mendeklarasikan variabel

Cara mendeklarasikan sebuah variabel dengan menggunakan *Dim* statement :

Dim variabel

Nama-nama variabel mengikuti aturan-aturan yang sama seperti yang lainnya ketika user memberi nama dalam bahasa Visual Basic. Sebagai contoh, dalam membuat sebuah variabel yang disebut *Result* dengan kode ini :

Dim Result

Tipe Data

Variabel Visual Basic merupakan tipe data **Variant**. Tipe data **Variant** dapat menyimpan arrays dan object, numerik, date/time, atau string data.

⇒ Modul

Kode dalam Visual Basic disimpan dalam modul. Ada tiga macam dari modul, yaitu :

- **Form module**

Merupakan dasar dari beberapa aplikasi Visual Basic. Form module dapat berisi gambaran grafis dari form dan kontrolnya, termasuk pengaturan property-nya, juga dapat berisi deklarasi form-level dari tipe, konstan, variabel, dan prosedur eksternal; prosedur yang mengatur kejadian; dan prosedur umum. Kenyataannya segala sesuatu yang dilakukan terhadap class module juga dilakukan terhadap form. Form merupakan hanya class module yang dapat mempunyai kontrol yang ditempatkan padanya dan menampilkan sebuah form window.

- **Standard Module**

Standard module (.BAS filename extension) merupakan kotak untuk prosedur dan deklarasi yang pada umumnya digunakan dengan bagian lainnya dari aplikasi. Mereka dapat berisi global-atau deklarasi module-level dari tipe, konstan, variabel, prosedur eksternal, dan prosedur global.

- **Class Module**

User dapat menulis kode dalam class module untuk menciptakan obyek baru. Obyek baru ini dapat memasukkan macam property dan metode, meskipun custom object

tidak dapat mempunyai kejadiannya sendiri. Semua properti dan metode yang diciptakan dapat juga digunakan dengan obyek lainnya dalam aplikasi Visual Basic.

⇒ **Prosedur**

Untuk dapat menyederhanakan perintah-perintah pemrograman maka dapat dilakukan dengan cara memecah program ke dalam komponen logika yang lebih kecil. Komponen ini disebut prosedur yang dapat kemudian menjadi blok bangunan untuk menambah dan memperpanjang Visual Basic. Prosedur sangat berguna untuk perintah-perintah pengulangan atau pembagian, seperti frekuensi yang digunakan untuk perhitungan, teks, dan manipulasi kontrol, dan operasi database.

Ada dua keuntungan utama dari pemrograman dengan prosedur :

- Prosedur dapat digunakan untuk memecah program aplikasi kedalam unit logika diskret dan dapat menemukan kesalahan lebih mudah daripada sebuah pemasangan program tanpa prosedur.
 - Prosedur digunakan dalam satu program yang dapat bekerja sebagai blok bangunan untuk program lainnya, biasanya dengan sedikit atau tanpa modifikasi.
- Ada beberapa tipe dari prosedur yang digunakan dalam Visual Basic.

Sebuah prosedur dapat juga sebuah **Sub**, **Function**, atau **Property** procedure.

- **Sub** procedure tidak menghasilkan sebuah nilai.
- **Function** procedure menghasilkan sebuah nilai.
- **Property** procedure dapat menghasilkan dan menentukan nilai, dan mengatur referensi untuk obyek.

III. 2 Langkah-Langkah Pemrograman Bukan Kulit

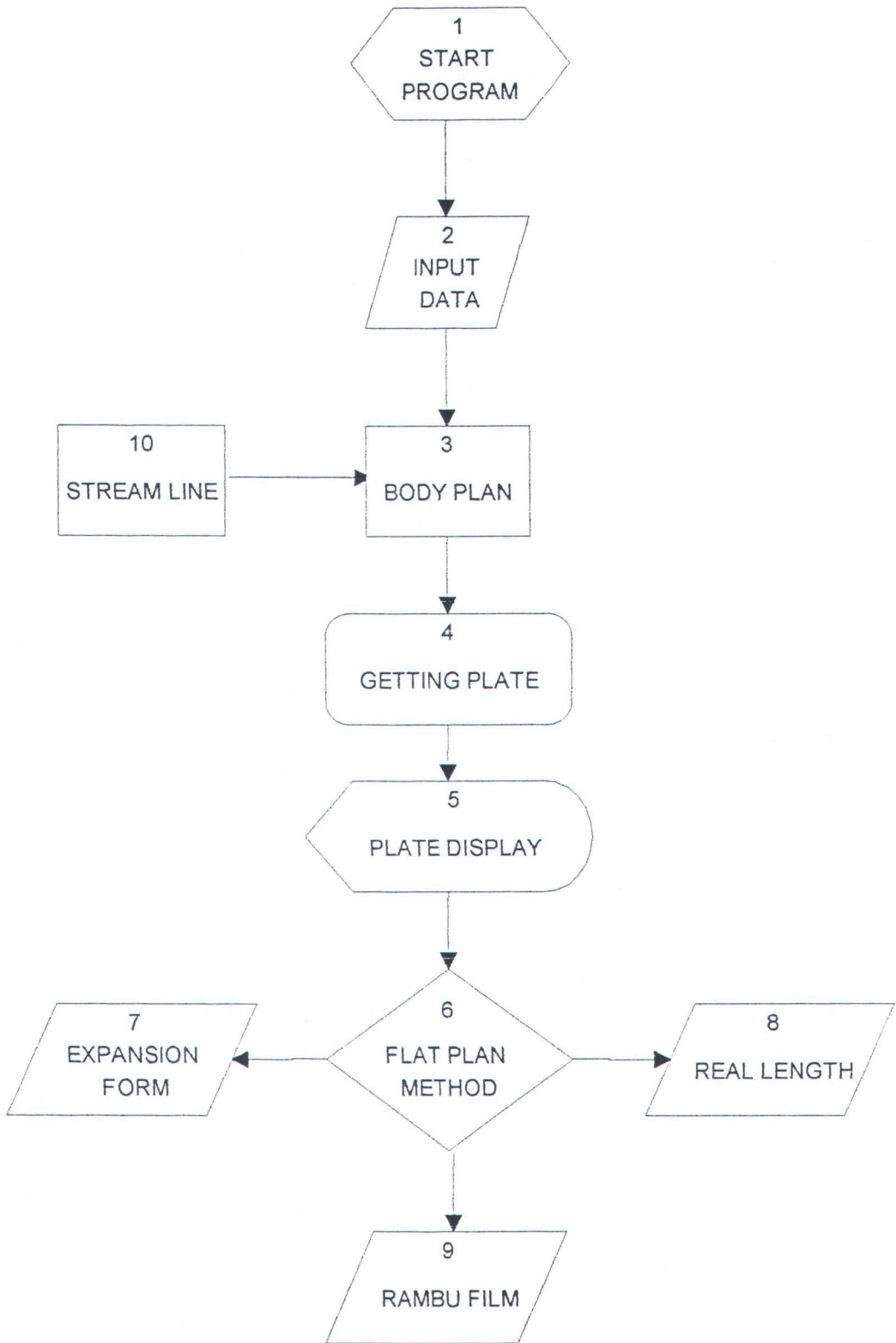
Gambar 3.1 merupakan flow chart yang menggambarkan alur pengerjaan program bukaan kulit. Alur pengerjaan bukaan kulit harus dikerjakan untuk mendapatkan hasil bukaan kulit yang diinginkan. Penjelasan singkat mengenai alur pengerjaan bukaan kulit dapat dijelaskan sebagai berikut:

1. Program penggambaran Body Plan (lengkung gading-gading dan lengkungan sambungan pelatnya) dan program stream-line/ fairing lengkung-lengkung tersebut.
2. Penentuan daerah pelat yang akan dibentangkan.
3. Menampilkan pelat sesuai pengambilan daerah pelat yang akan dibentangkan.
4. Pembuatan program bukaan kulit metode flat plan.
5. Menampilkan pelat hasil bukaan kulit dari program bukaan kulit metode flat plan.
6. Pemberian simbol-simbol penandaan dan pengukuran dimensi sebenarnya pada hasil bukaan kulit pada langkah 5.
7. Menampilkan pelat hasil bukaan didalam rambu film.

Penjelasan secara detail langkah-langkah pengerjaan program adalah sebagai berikut :

1) Program Penggambaran Body Plan

Data koordinat body plan kapal yang digunakan sebagai input data penggambaran body plan kapal dalam PROGRAM BUKAAN KULIT METODE FLAT PLAN diambil dari data sebuah kapal, dimana pengerjaan Mouldloft-nya dikerjakan di POLITEKNIK PERKAPALAN ITS - SURABAYA dengan nama kapal adalah KM BUNGA TERATAI 55. Ukuran utama KM BUNGA TERATAI 55 dapat dijelaskan sebagai berikut :



Gambar 3.1. Flow Chart Program Bukan Kulit.

Ukuran Utama KM BUNGA TERATAI 55 :

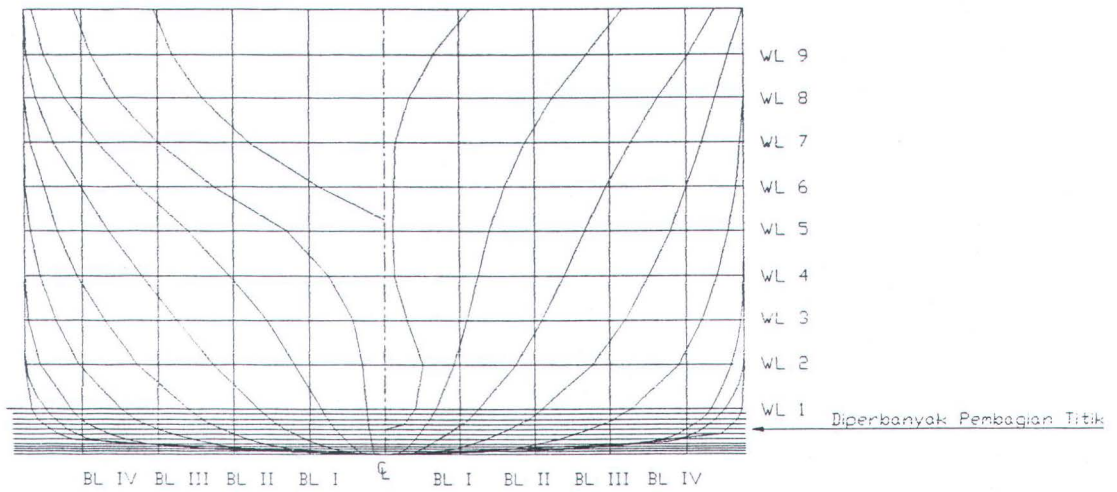
- ◆ Panjang Kapal (L_{pp}) = 54 m.
- ◆ Lebar Kapal (B_{mld}) = 9.5 m.
- ◆ Tinggi Kapal (D_{dk}) = 5.8 m.
- ◆ Sarat Kapal (d) = 3.7 m.
- ◆ Radius Bilga = 1.115 m.
- ◆ Jarak Gading = 0.6 m.

Untuk keperluan input data penggambaran body plan mouldloft, maka gambar-gambar yang diperlukan adalah :

1. Gambar Lines Plan kapal, dimana gambar lines plan tersebut telah dilengkapi dengan pembagian gading-gading.
2. Gambar Bukaan Kulit (dari tahap Key Plan), dimana data yang diperlukan adalah sambungan pelat, jarak garis las, ketebalan pelat dan nomor bagian pelat yang akan dibentangkan.

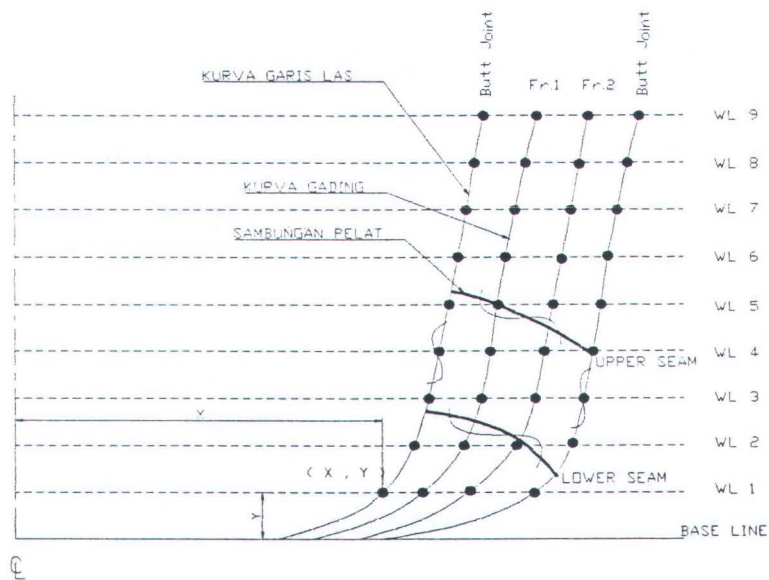
Cara pembacaan koordinat kurva :

⇒ Sebagai langkah awal dalam pembacaan kurva, maka faktor ketelitian dan kemulusan kurva menjadi hal utama dalam penggambaran kurva. Oleh karena itu sebelum melangkah ke pembacaan kurva maka langkah pembagian titik pada kurva harus diperhatikan terlebih dahulu. Terutama untuk WL1 ke bawah, karena pada bagian ini terjadi kelengkungan yang lebih besar dari kurva diatas WL1. Oleh karena itu maka pembagian titik harus diperbanyak terutama untuk garis air 1 (WL 1) ke bawah. Gambar 3.2. menunjukkan pembagian titik (WL) pada body plan kapal.



Gambar 3.2. Pembagian Titik Kurva Pada Body Plan.

⇒ Langkah selanjutnya adalah pembacaan koordinat kurva gading dan sambungan pelat yang ditunjukkan pada Gambar 3.3.



Gambar 3.3. Cara Pembacaan Koordinat Kurva.

Pembacaan koordinat kurva gading/garis las ditunjukkan dengan koordinat (x , y), dimana x merupakan jarak antara center line kapal dengan titik potong antara garis air dengan kurva, sedangkan y merupakan jarak antara base line (garis dasar) kapal dengan titik potong antara garis air dengan kurva. Sedangkan cara pembacaan koordinat kurva sambungan pelat (upper seam dan lower seam) adalah sebagai berikut, x merupakan jarak antara center line kapal dengan titik potong kurva gading/las dengan sambungan pelat, y merupakan jarak antara titik potong kurva gading/las dan sambungan pelat dengan base line. Input data koordinat kurva dari body plan KM BUNGA TERATAI 55 dapat dilihat pada Tabel 3.4. untuk data koordinat kurva gading pada pelat sebelah kiri center line kapal, Tabel 3.5. untuk data koordinat kurva sambungan pelat pada pelat sebelah kiri center line kapal, Tabel 3.7. untuk data koordinat kurva gading pada pelat sebelah kanan center line, Tabel 3.6. untuk data koordinat kurva sambungan pelat pada pelat sebelah kanan center line kapal. Dalam program bukaan kulit metode flat plan ini data-data koordinat tersebut disimpan dalam bentuk file, dimana untuk penyimpanan data dapat digunakan text editor (NOTEPAD, WORDPAD dan seterusnya). Kemudian data-data tersebut dapat dibaca oleh AutoLISP dengan menggunakan perintah “read-line”. Data-data koordinat dapat juga dimasukkan melalui Program Visual Basic yang merupakan Program Visual untuk input data. Kemudian disimpan dengan nama file yang diinginkan pengguna. Karena antara Program AutoLISP dengan Program Visual Basic terpisah, maka nama file harus dimasukkan dulu pada program AutoLISP, kemudian dapat dilakukan pengolahan data koordinat dari nama file tersebut yaitu melakukan penggambaran body plan.

**DATA KOORDINAT KURVA GADING/LAS UNTUK PELAT
PADA POSISI SEBELAH KIRI CENTER LINE BODY PLAN**

KAPAL

- ◆ Jumlah Gading = 11
- ◆ Jumlah Titik = 22
- ◆ No. Awal Gading = 21
- ◆ Jarak After Butt = 0.3 m.
- ◆ Jarak Fore Butt = 0.3 m.

KOORDINAT (X,Y) GADING No. Gading 21 - 29 (m)

	After Butt	Gading 21	Gading 22	Gading 23	Gading 24	Gading 25	Gading 26	Gading 27	Gading 28	Gading 29	Fore Butt
WL1	0, 0.35	0, 0.35	0, 0.35	0, 0.350	0, 0.350	0, 0.350	0, 0.350	0, 0.350	0, 0.350	0, 0.350	0, 0.350
WL2	0.03, 0.99	0.03, 1.03	0.03, 1.09	0.03, 1.15	0.03, 1.19	0.03, 1.24	0.03, 1.28	0.03, 1.33	0.03, 1.33	0.03, 1.33	0.03, 1.33
WL3	0.06, 1.63	0.06, 1.70	0.06, 1.83	0.06, 1.94	0.06, 2.03	0.06, 2.12	0.06, 2.21	0.06, 2.30	0.06, 2.30	0.06, 2.30	0.06, 2.30
WL4	0.10, 1.79	0.10, 1.86	0.10, 2.00	0.10, 2.15	0.10, 2.29	0.10, 2.43	0.10, 2.57	0.10, 2.71	0.10, 2.71	0.10, 2.71	0.10, 2.71
WL5	0.13, 1.94	0.13, 2.02	0.13, 2.17	0.13, 2.35	0.13, 2.54	0.13, 2.74	0.13, 2.93	0.13, 3.13	0.13, 3.13	0.13, 3.13	0.13, 3.13
WL6	0.16, 2.06	0.16, 2.14	0.16, 2.40	0.16, 2.48	0.16, 2.68	0.16, 2.88	0.16, 3.08	0.16, 3.28	0.16, 3.31	0.16, 3.35	0.16, 3.37
WL7	0.19, 2.18	0.19, 2.26	0.19, 2.42	0.19, 2.60	0.19, 2.81	0.19, 3.01	0.19, 3.22	0.19, 3.43	0.19, 3.50	0.19, 3.58	0.19, 3.61
WL8	0.26, 2.43	0.26, 2.52	0.26, 2.69	0.26, 2.87	0.26, 3.06	0.26, 3.25	0.26, 3.44	0.26, 3.63	0.26, 3.70	0.26, 3.78	0.26, 3.81
WL9	0.32, 2.60	0.32, 2.68	0.32, 2.86	0.32, 3.04	0.32, 3.22	0.32, 3.39	0.32, 3.57	0.32, 3.75	0.32, 3.82	0.32, 3.90	0.32, 3.93
WL10	0.39, 2.76	0.39, 2.86	0.39, 3.05	0.39, 3.23	0.39, 3.39	0.39, 3.55	0.39, 3.71	0.39, 3.88	0.39, 3.94	0.39, 4.01	0.39, 4.04
WL11	0.45, 2.90	0.45, 3.00	0.45, 3.20	0.45, 3.38	0.45, 3.53	0.45, 3.69	0.45, 3.84	0.45, 4.00	0.45, 4.06	0.45, 4.12	0.45, 4.15
WL12	0.52, 3.01	0.52, 3.12	0.52, 3.32	0.52, 3.50	0.52, 3.64	0.52, 3.79	0.52, 3.93	0.52, 4.08	0.52, 4.13	0.52, 4.19	0.52, 4.22
WL13	0.58, 3.10	0.58, 3.20	0.58, 3.40	0.58, 3.57	0.58, 3.72	0.58, 3.86	0.58, 4.01	0.58, 4.15	0.58, 4.21	0.58, 4.26	0.58, 4.29
WL14	1.16, 3.70	1.16, 3.78	1.16, 3.96	1.16, 4.10	1.16, 4.20	1.16, 4.30	1.16, 4.40	1.16, 4.50	1.16, 4.53	1.16, 4.56	1.16, 4.57
WL15	1.74, 4.05	1.74, 4.12	1.74, 4.26	1.74, 4.36	1.74, 4.44	1.74, 4.52	1.74, 4.60	1.74, 4.68	1.74, 4.68	1.74, 4.69	1.74, 4.70
WL16	2.32, 4.31	2.32, 4.37	2.32, 4.47	2.32, 4.55	2.32, 4.60	2.32, 4.65	2.32, 4.70	2.32, 4.75	2.32, 4.75	2.32, 4.75	2.32, 4.75
WL17	2.90, 4.49	2.90, 4.53	2.90, 4.61	2.90, 4.66	2.90, 4.68	2.90, 4.71	2.90, 4.73	2.90, 4.75	2.90, 4.75	2.90, 4.75	2.90, 4.75
WL18	3.48, 4.63	3.48, 4.65	3.48, 4.70	3.48, 4.73	3.48, 4.73	3.48, 4.74	3.48, 4.74	3.48, 4.75	3.48, 4.75	3.48, 4.75	3.48, 4.75
WL19	4.06, 4.71	4.06, 4.72	4.06, 4.74	4.06, 4.75	4.06, 4.75	4.06, 4.75	4.06, 4.75	4.06, 4.75	4.06, 4.75	4.06, 4.75	4.06, 4.75
WL20	4.64, 4.74	4.64, 4.74	4.64, 4.75	4.64, 4.75	4.64, 4.75	4.64, 4.75	4.64, 4.75	4.64, 4.75	4.64, 4.75	4.64, 4.75	4.64, 4.75
WL21	5.22, 4.75	5.22, 4.75	5.22, 4.75	5.22, 4.75	5.22, 4.75	5.22, 4.75	5.22, 4.75	5.22, 4.75	5.22, 4.75	5.22, 4.75	5.22, 4.75
WL22	5.80, 4.75	5.80, 4.75	5.80, 4.75	5.80, 4.75	5.80, 4.75	5.80, 4.75	5.80, 4.75	5.80, 4.75	5.80, 4.75	5.80, 4.75	5.80, 4.75

Tabel 3.4. Data Koodinat Kurva Gading (no. gading 21-29).

KOORDINAT (X,Y) SAMBUNGAN PELAT No. GADING 21 - 29 (m)

	Upper Seam	Lower Seam
After Butt	4.15, 1.95	3.21, 0.66
Gading 21	4.20, 1.92	3.24, 0.61
Gading 22	4.29, 1.83	3.29, 0.50
Gading 23	4.37, 1.76	3.33, 0.43
Gading 24	4.43, 1.71	3.35, 0.37
Gading 25	4.49, 1.65	3.36, 0.31
Gading 26	4.55, 1.58	3.38, 0.24
Gading 27	4.62, 1.50	3.40, 0.19
Gading 28	4.63, 1.48	3.40, 0.18
Gading 29	4.64, 1.46	3.40, 0.17
Fore Butt	4.64, 1.46	3.40, 0.17

Tabel 3.5. Data Koordinat Kurva Sambungan Pelat (no. gading 21-29).

KOORDINAT (X,Y) SAMBUNGAN PELAT No. GADING 66 - 74 (m)

	Upper Seam	Lower Seam
After Butt	3.81, 2.22	3.09, 1.02
Gading 66	3.89, 2.16	3.13, 0.94
Gading 67	4.05, 2.04	3.23, 0.79
Gading 68	4.21, 1.90	3.30, 0.63
Gading 69	4.30, 1.83	3.36, 0.51
Gading 70	4.39, 1.75	3.40, 0.39
Gading 71	4.48, 1.66	3.42, 0.27
Gading 72	4.58, 1.53	3.43, 0.17
Gading 73	4.64, 1.44	3.43, 0.14
Gading 74	4.66, 1.40	3.43, 0.13
Fore Butt	4.67, 1.39	3.43, 0.13

Tabel 3.6. Data Koordinat Kurva Sambungan Pelat (no. gading 66-74).

**DATA KOORDINAT KURVA GADING/LAS UNTUK PELAT
PADA POSISI SEBELAH KANAN CENTER LINE BODY PLAN**

KAPAL

- ◆ Jumlah Gading = 11
- ◆ Jumlah Titik = 22
- ◆ No. Awal Gading = 66
- ◆ Jarak After Butt = 0.3 m.
- ◆ Jarak Fore Butt = 0.3 m.

KOORDINAT (X,Y) GADING No, Gading 66 - 74 (m)

	After Butt	Gading 66	Gading 67	Gading 68	Gading 69	Gading 70	Gading 71	Gading 72	Gading 73	Gading 74	Fore Butt
WL1	0, 0.35	0, 0.35	0, 0.35	0, 0.35	0, 0.35	0, 0.35	0, 0.35	0, 0.35	0, 0.35	0, 0.35	0, 0.35
WL2	0.03, 1.40	0.03, 1.40	0.03, 1.40	0.03, 1.36	0.03, 1.27	0.03, 1.19	0.03, 1.10	0.03, 1.01	0.03, 0.93	0.03, 0.84	0.03, 0.80
WL3	0.06, 2.45	0.06, 2.45	0.06, 2.45	0.06, 2.36	0.06, 2.19	0.06, 2.02	0.06, 1.85	0.06, 1.68	0.06, 1.50	0.06, 1.33	0.06, 1.24
WL4	0.10, 2.93	0.10, 2.93	0.10, 2.93	0.10, 2.80	0.10, 2.56	0.10, 2.31	0.10, 2.07	0.10, 1.83	0.10, 1.64	0.10, 1.46	0.10, 1.37
WL5	0.13, 3.40	0.13, 3.40	0.13, 3.40	0.13, 3.24	0.13, 2.93	0.13, 2.61	0.13, 2.29	0.13, 1.98	0.13, 1.78	0.13, 1.59	0.13, 1.49
WL6	0.16, 3.67	0.16, 3.63	0.16, 3.57	0.16, 3.38	0.16, 3.07	0.16, 2.76	0.16, 2.45	0.16, 2.14	0.16, 1.93	0.16, 1.72	0.16, 1.62
WL7	0.19, 3.93	0.19, 3.87	0.19, 3.74	0.19, 3.52	0.19, 3.22	0.19, 2.91	0.19, 2.61	0.19, 2.30	0.19, 2.08	0.19, 1.86	0.19, 1.74
WL8	0.26, 4.08	0.26, 4.02	0.26, 3.89	0.26, 3.68	0.26, 3.38	0.26, 3.09	0.26, 2.79	0.26, 2.50	0.26, 2.27	0.26, 2.03	0.26, 1.92
WL9	0.32, 4.18	0.32, 4.13	0.32, 4.01	0.32, 3.81	0.32, 3.53	0.32, 3.24	0.32, 2.96	0.32, 2.68	0.32, 2.44	0.32, 2.20	0.32, 2.08
WL10	0.39, 4.27	0.39, 4.22	0.39, 4.11	0.39, 3.92	0.39, 3.65	0.39, 3.38	0.39, 3.12	0.39, 2.85	0.39, 2.61	0.39, 2.36	0.39, 2.24
WL11	0.45, 4.34	0.45, 4.29	0.45, 4.18	0.45, 4.00	0.45, 3.75	0.45, 3.50	0.45, 3.25	0.45, 3.00	0.45, 2.75	0.45, 2.50	0.45, 2.38
WL12	0.52, 4.40	0.52, 4.35	0.52, 4.25	0.52, 4.08	0.52, 3.84	0.52, 3.60	0.52, 3.36	0.52, 3.13	0.52, 2.87	0.52, 2.61	0.52, 2.49
WL13	0.58, 4.44	0.58, 4.39	0.58, 4.30	0.58, 4.14	0.58, 3.91	0.58, 3.68	0.58, 3.45	0.58, 3.23	0.58, 2.96	0.58, 2.70	0.58, 2.57
WL14	1.16, 4.64	1.16, 4.62	1.16, 4.57	1.16, 4.47	1.16, 4.31	1.16, 4.15	1.16, 3.99	1.16, 3.83	1.16, 3.58	1.16, 3.34	1.16, 3.21
WL15	1.74, 4.71	1.74, 4.70	1.74, 4.68	1.74, 4.62	1.74, 4.50	1.74, 4.38	1.74, 4.27	1.74, 4.15	1.74, 3.93	1.74, 3.71	1.74, 3.59
WL16	2.32, 4.74	2.32, 4.73	2.32, 4.73	2.32, 4.68	2.32, 4.60	2.32, 4.52	2.32, 4.43	2.32, 4.35	2.32, 4.15	2.32, 3.95	2.32, 3.85
WL17	2.90, 4.75	2.90, 4.75	2.90, 4.75	2.90, 4.72	2.90, 4.66	2.90, 4.60	2.90, 4.54	2.90, 4.48	2.90, 4.31	2.90, 4.14	2.90, 4.06
WL18	3.48, 4.75	3.48, 4.75	3.48, 4.75	3.48, 4.73	3.48, 4.69	3.48, 4.65	3.48, 4.61	3.48, 4.58	3.48, 4.44	3.48, 4.30	3.48, 4.23
WL19	4.06, 4.75	4.06, 4.75	4.06, 4.75	4.06, 4.74	4.06, 4.72	4.06, 4.69	4.06, 4.67	4.06, 4.65	4.06, 4.53	4.06, 4.42	4.06, 4.36
WL20	4.64, 4.75	4.64, 4.75	4.64, 4.75	4.64, 4.74	4.64, 4.73	4.64, 4.72	4.64, 4.71	4.64, 4.70	4.64, 4.61	4.64, 4.52	4.64, 4.48
WL21	5.22, 4.75	5.22, 4.75	5.22, 4.75	5.22, 4.75	5.22, 4.75	5.22, 4.75	5.22, 4.75	5.22, 4.75	5.22, 4.69	5.22, 4.64	5.22, 4.61
WL22	5.80, 4.75	5.80, 4.75	5.80, 4.75	5.80, 4.75	5.80, 4.75	5.80, 4.75	5.80, 4.75	5.80, 4.75	5.80, 4.73	5.80, 4.71	5.80, 4.69

Tabel 3.7. Data Koordinat Kurva Gading (no. gading 66-74).

Pembuatan program ini dimulai dengan suatu program startapp, yaitu program yang digunakan untuk menghubungkan data dari Microsoft Visual Basic yang telah diubah dalam bentuk project yaitu data.exe dengan AutoLISP.

```
(defun c: data ()
(startapp " c:\\ skripsi \\progam \\ data.exe " )
(princ)
) end of startapp
```

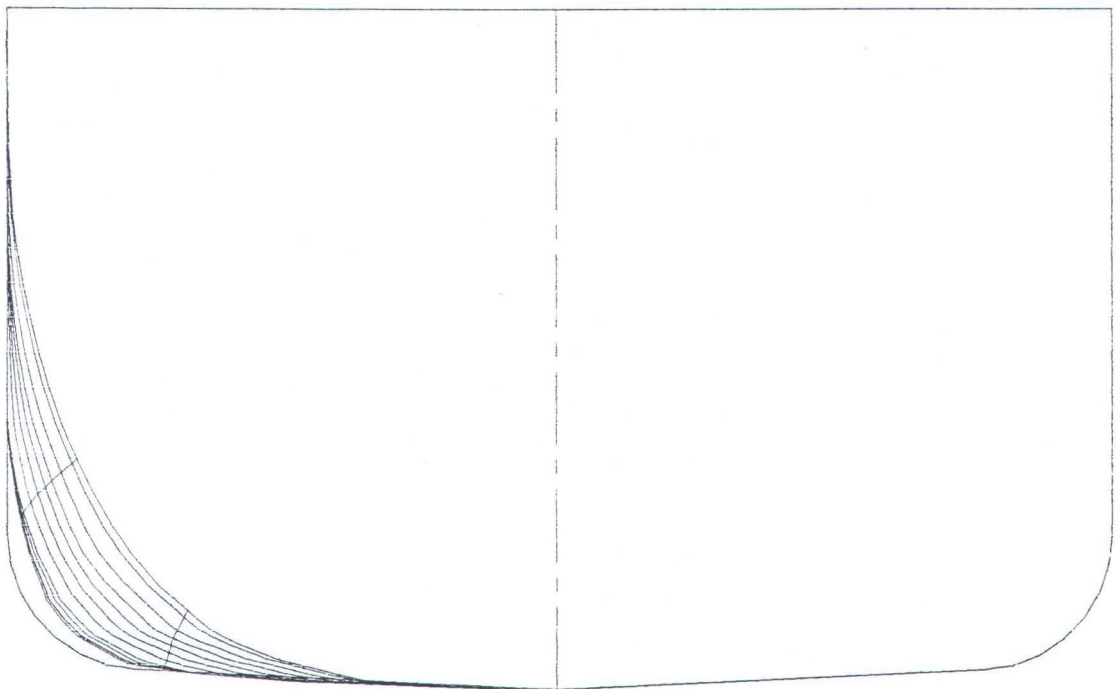
Kemudian untuk mendeteksi kesalahan akibat pemrograman digunakan fungsi * error * sehingga apabila terjadi kesalahan maka program AutoLISP akan berhenti, serta muncul keterangan tentang kesalahannya. Bentuk sub-routine pemrograman untuk mendeteksi kesalahan adalah sebagai berikut :

```
(defun myerror (s)
  (if (/= s "Function cancelled")
      (princ (strcat " \ nError: "s))
      )
  (setq p nil)
  (setq * error * olderr)
  (command "New" "y" "")
  (princ)
); end of myerror
```

⇒ Proses penggambaran dimulai dari penggambaran body kapal, dimana input data ukuran utama kapal dijadikan data masukkan untuk penggambarannya. Perintah penggambaran menggunakan perintah "LINE" yaitu menghubungkan titik-titik yang telah didefinisikan terlebih dulu dengan menggunakan (CAR) untuk pengambilan titik absis X, (CADR) untuk pengambilan titik ordinat Y. Sedangkan untuk menggambarkan radius bilga menggunakan perintah "FILLET /R". Penggambaran kurva-kurva baik untuk gading, garis las, maupun sambungan pelat menggunakan perintah "POLYLINE" yaitu dengan menghubungkan koordinat (X , Y) dari data-data koordinat yang diberikan.

Bentuk sub-routine penggambaran body plan, kurva gading-gading dan sambungan pelat dapat dilihat pada Lampiran B.2.1. Sedangkan hasil penggambaran body plan dapat dilihat pada Gambar 3.8.

⇒ Setelah semua kurva-kurva tergambar baik kurva gading maupun kurva sambungan pelat, maka langkah selanjutnya adalah memuluskan bentuk kurva-kurva tersebut sehingga kurva-kurva tersebut terlihat baik dan sesuai dengan bentuk aslinya. Dengan menggunakan perintah POLYLINE EDIT (PEDIT), dimana didalamnya terdapat perintah "FIT" yang merupakan perintah untuk memuluskan POLYLINE dengan toleransi yang lebih kecil dibandingkan "SPLINE". Bentuk sub-routine program pemulusan kurva dapat dilihat pada Lampiran B.2.2.

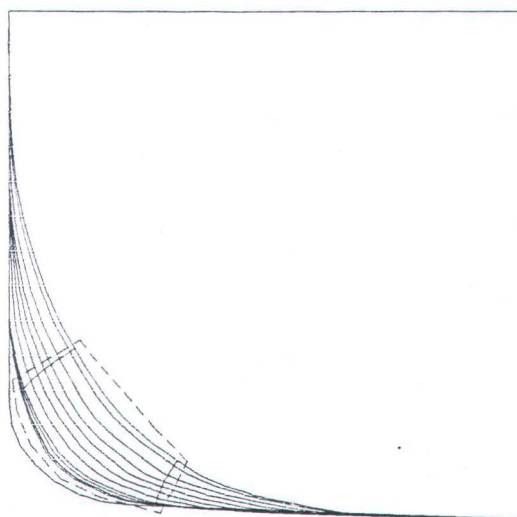


Gambar 3.8. Body Plan.

2) Program Penentuan Daerah Pelat Yang Akan Dibentangkan

Dalam menentukan daerah pelat yang akan dibentangkan, maka persyaratan posisi dan bentuk kurva dalam “ Metode Flat Plan “ harus benar-benar diperhatikan. Bentuk dan posisi kurva dalam metode ini adalah kurva-kurva gading pada daerah bilga dimana bentuk gadingnya sebagian datar dan sebagian lengkung (dapat dilihat pada Gambar 3.8. Body Plan).

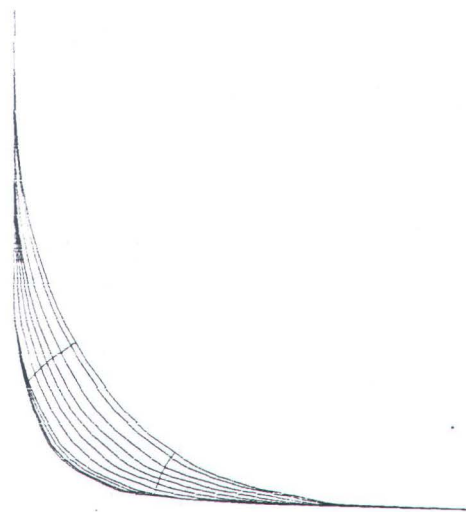
Proses pemilihan pada program ini menggunakan perintah “ PICK “ seperti halnya pada perintah AutoCAD, dimana pemilihan obyeknya menggunakan perintah “ CROSS POLYGON” yang mempunyai kelebihan dalam memilih obyek dalam posisi yang sulit. Gambar 3.9. memperlihatkan pemilihan daerah pelat dengan menggunakan perintah PICK. Bentuk sub-routine program PICK dapat dilihat pada Lampiran B.2.3. Setelah proses penentuan daerah pelat selesai, dimana dari penentuan daerah pelat tersebut menghasilkan “ am1 “ yang merupakan hasil pemilihan kurva-kurva dalam bentuk Entity. Hasil inilah yang digunakan untuk menampilkan hanya daerah pelat yang terpilih.



Gambar 3.9. Perintah PICK Daerah Pelat.

3) Program Tampilan Pelat Yang Akan Dibentangkan.

Hasil penentuan daerah pelat “am1” digunakan untuk menampilkan daerah pelat yang terpilih dalam bentuk tipe data “AutoCAD entity names”. Sebelum melakukan langkah pemotongan terhadap daerah pelat tersebut sehingga terlihat jelas daerah pelat dengan pembatas sambungan pelatnya baik itu butt joint maupun seam joint, maka langkah pertama yang harus diambil adalah menentukan titik-titik akhir dari masing-masing pembatas dan menghilangkan daerah-daerah lain selain daerah pelat yang terpilih. Setelah titik-titik akhir dari kurva gading maupun kurva sambungan pelat ditemukan, maka langkah pemotongan dapat dilakukan dengan pembatas sambungan pelat (joint plate). Gambar 3.10. memperlihatkan daerah pelat yang terpilih. Bentuk sub-routine program pengambilan daerah pelat yang dipilih dan penentuan titik awal dan akhir kurva dapat dilihat pada Lampiran B.2.4 Penggunaan perintah “SEQEND” pada *sub-routine penentuan titik* adalah perintah akhir dari AutoCAD Entity Names dimana sebelumnya harus melalui “EDIT VERTEX” atau perintah polyline yang telah diedit “FIT” (Smith,1992).



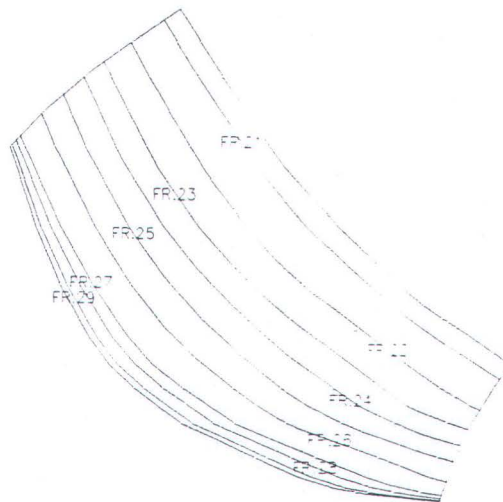
Gambar 3.10. Tampilan Daerah Pelat Terpilih.

Setelah titik awal dan akhir dari masing-masing kurva ditemukan, maka langkah pemotongan dengan menggunakan perintah “TRIM” dapat dilakukan dengan pembatas sambungan pelatnya baik butt joint maupun seam joint:

- (car am12) (cadr am12) untuk seam joint.
- (car am20) (last am20) untuk butt joint.

Bentuk sub-routine program pemotongan pelat dapat dilihat pada Lampiran B.2.5.

Setelah langkah pemotongan, maka tampak pelat dengan gading-gading yang terpilih dan sambungan pelat baik butt joint maupun seam joint. Pemberian nomor gading seperti pada Gambar 3.11 merupakan nomor gading yang diberikan pada input data penggambaran body plan. Dari pelat yang telah mengalami proses pemotongan inilah yang digunakan untuk proses pembentangan.



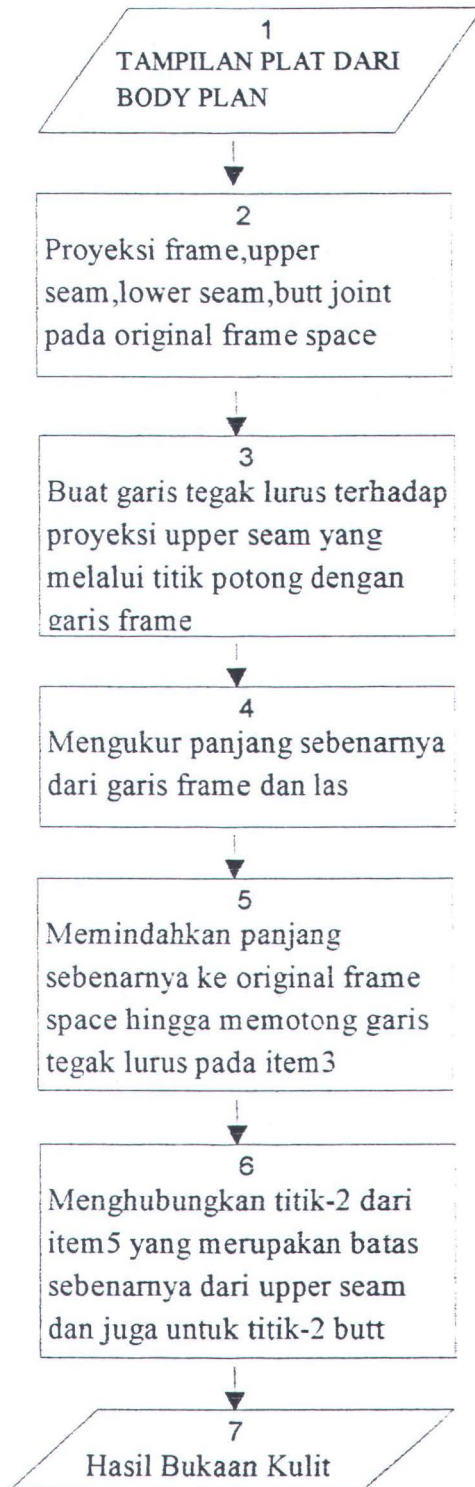
Gambar 3.11. Potongan Pelat.

4) Pembuatan Program Metode Flat Plan

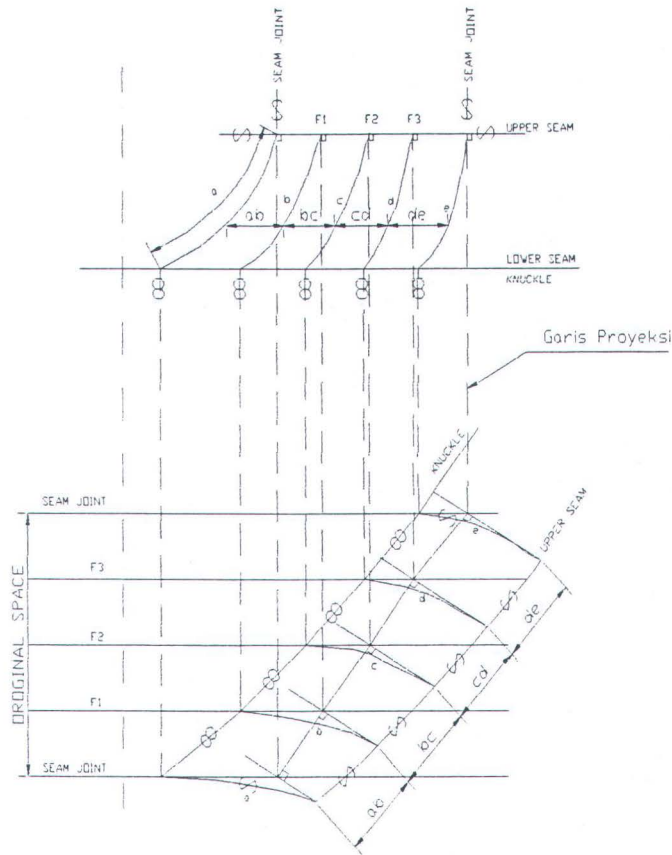
Proses pembuatan program “Metode Flat Plan” mengikuti langkah-langkah pengerjaan pembentangan kulit seperti yang telah diterangkan pada Bab II tentang “Metode Bukaam Kulit Flat Plan”. Untuk lebih jelasnya dapat dilihat Gambar 3.12



FLOW CHART
METODE BUKAAN FLAT



Gambar 3.12. Flow Chart Program Bukaan Kulit Metode Flat Plan.



Gambar 3.13. Metode Flat Plan.

Dari gambaran langkah-langkah pengerjaan bukaan kulit seperti yang dijelaskan pada Gambar 3.12, maka langkah-langkah pengerjaan program bukaan kulit metode flat plan adalah sebagai berikut :

- i. Menampilkan pelat hasil pemotongan seperti yang telah dijelaskan pada Gambar 3.11.
- ii. Memproyeksikan frame, upper seam, lower seam, after butt, dan fore butt dari pelat hasil pemotongan pada jarak gading sebenarnya.

Sebagai langkah awal dalam proyeksi adalah menentukan titik awal dan akhir dari kurva frame yang dipakai sebagai acuan proyeksi. Bentuk sub-routine program penentuan titik-titik kurva dapat dilihat pada Lampiran B.2.6. Setelah

titik awal dan akhir dari frame yang merupakan titik-titik dari upper seam dan lower seam ditemukan, maka langkah selanjutnya adalah membuat garis proyeksi yaitu dengan mendefinisikan suatu jarak tertentu dari titik upper seam maupun lower seam. Bentuk sub-routine program pembuatan garis proyeksi dapat dilihat pada Lampiran B.2.7.

Untuk menentukan titik-titik perpotongan garis proyeksi dengan jarak gading sebenarnya, dimana titik-titik tersebut dipakai sebagai titik hasil proyeksi upper seam maupun lower seam pada jarak gading sebenarnya maka langkah pemrogramannya adalah :

- ◆ Menentukan titik awal dan akhir dari garis proyeksi maupun garis gading sebenarnya.
- ◆ Menentukan titik perpotongan dari kedua garis tersebut.

Catatan : untuk entity names garis, titik awal dari garis dilambangkan dengan angka 10 sedangkan titik akhir garis dilambangkan dengan angka 11.

Bentuk pemrograman dari penentuan titik awal dan akhir garis dapat dilihat pada Lampiran B.2.8.

Titik awal dan akhir dari garis gading sebenarnya disimpan dalam *llr* sedangkan untuk garis proyeksi disimpan dalam *ccc*. Untuk menentukan perpotongan antara dua garis, AutoLISP menyediakan fungsi *inter*. Bentuk pemrograman untuk menentukan perpotongan antara dua garis dapat dilihat pada Lampiran B.2.9.

- iii. Membuat garis tegak lurus terhadap garis proyeksi upper seam yang melalui titik potong garis tersebut dengan garis gading sebenarnya. Titik potong antara garis proyeksi upper seam dengan garis gading sebenarnya (*ptr*) dijadikan sebagai titik

acuan untuk membuat garis tegak lurus. Sudut antara dua titik dalam jarak gading sebenarnya dicari untuk dijadikan sebagai sudut pembanding dari garis tegak lurus yang akan dibuat. Setelah sudut dua titik diketahui maka dengan menggunakan perintah *polar* dibuat garis tegak lurus.

Bentuk program pembuatan garis tegak lurus dapat dilihat pada Lampiran B.2.10.

- iv. Mengukur panjang sebenarnya dari gading, kemudian memindahkan ukuran panjang gading sebenarnya ke jarak gading sebenarnya dan memotong garis tegak lurus yang telah dibuat. Cara menentukan panjang kurva dalam pemrograman AutoLISP adalah membagi panjang kurva dengan 250 titik atau lebih karena makin banyak pembagian makin teliti. Setelah pembagian titik tersebut maka langkah selanjutnya adalah menentukan jarak antar titik dimana cukup jarak antar satu segmen saja yang diperlukan dan kemudian dikalikan dengan jumlah pembagian titik.

Bentuk program pembagian panjang kurva, program penentuan jarak antar titik pada kurva, program penggambaran panjang kurva gading sebenarnya, dan program penentuan titik potong lingkaran dengan garis tegak lurus dapat dilihat pada Lampiran B.2.11.

- v. Menghubungkan titik-titik potong lingkaran dengan garis tegak lurus yang merupakan batas sebenarnya dari upper seam.

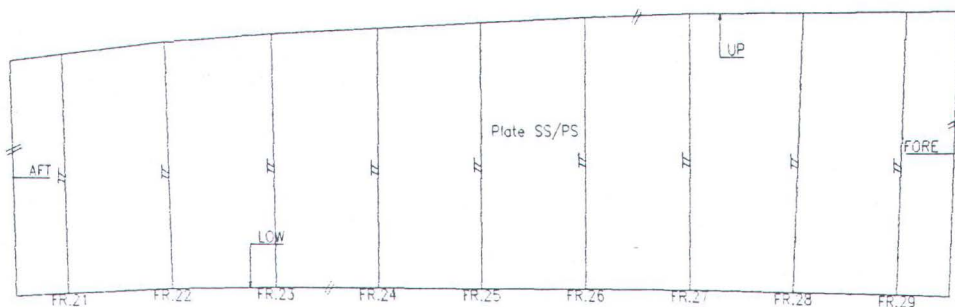
Bentuk program untuk menggambarkan batas sebenarnya dari upper seam dapat dilihat pada Lampiran B.2.12.

Langkah selanjutnya adalah menggambarkan panjang kurva frame sebenarnya dengan titik-titik potong lingkaran dengan garis tegak lurus sebagai second point

(titik kedua) dan titik-titik proyeksi lower seam sebagai first point (titik pertama). Bentuk program penggambaran panjang kurva sebenarnya dan lower seam dapat dilihat pada Lampiran B.2.13.

5). Menampilkan Hasil Bukaan Kulit Metode Flat Plan

Hasil bukaan kulit pada langkah 4 masih berada pada posisi sudut miring. Pada langkah 5 ini hasil bukaan kulit akan ditempatkan pada posisi yang datar atau pada posisi sesuai dengan keadaan sebenarnya. Langkah pemrogramannya adalah dengan memutar posisi hasil bukaan kulit dengan sudut yang sesuai yaitu pada sudut 0° atau 180° . Bentuk pemrograman tampilan pelat datar dapat dilihat pada Lampiran B.2.14. Dengan posisi yang sesuai dari hasil bukaan kulit maka pemberian simbol-simbol penandaan akan lebih baik. Hasil bukaan kulit dapat dilihat pada Lampiran D.



Gambar 3.14. Hasil Bukaan Kulit Pelat Metode Flat Plan.

6). Pemberian Simbol-Simbol Penandaan dan Pengukuran Dimensi Sebenarnya.

Pada langkah pemberian simbol-simbol penandaan ini yang perlu diperhatikan adalah simbol-simbol penandaan yang standart dari mouldloft. Pemberian simbol-

simbol penandaan disesuaikan dengan posisi bentangan kulitnya. Pada pemrograman ini simbol-simbol penandaan disimpan dengan perintah AutoCAD *wblock* seperti :

- Simbol ketebalan pelat
- Simbol batas potong pelat
- Simbol arah penandaan (aft-ma, fore-ma, low-ma, up-ma)

Selain simbol-simbol penandaan, nomer gading dari pelat yang dibentangkan juga diberikan dalam hal ini. Pemberian nomer gading diawali dari nomer gading awal pembentangan kulit. Bentuk program pemberian simbol-simbol penandaan dan nomor gading dapat dilihat pada Lampiran B.2.15.

Selanjutnya untuk langkah pengukuran hasil bukaan kulit yang perlu diukur adalah:

- Panjang kurva frame sebenarnya, panjang after butt, dan fore butt
- Panjang kurva upper seam sebenarnya
- Panjang kurva lower seam sebenarnya
- Jarak gading pada upper seam
- Jarak gading pada lower seam

Untuk panjang kurva frame, kurva after butt, dan fore butt sudah diukur pada langkah pembuatan program metode flat plan (dt12). Sedangkan untuk mengukur panjang kurva upper seam dan lower seam, langkah pemrograman sama dengan mengukur panjang kurva frame cuma tinggal mengganti “ layer frame” dengan “layer weld1” yaitu layer untuk upper seam maupun lower seam. Bentuk sub-routine program untuk mengukur jarak gading pada upper seam maupun lower seam dapat anda lihat pada Lampiran B.2.16.

Setelah semua data pengukuran didapatkan maka langkah selanjutnya adalah menuliskan data-data tersebut kedalam file. File tersebut yang akan dibaca oleh Microsoft Visual Basic untuk ditampilkan sebagai data hasil pengukuran terhadap hasil bukaan kulit. Bentuk program penulisan data pada file dapat dilihat pada Lampiran B.2.17.

7). Program Rambu Film

Pada pembuatan program ini, rambu film telah dibuat dan simpan dengan perintah wblock dan dengan cara yang sama dengan program pemberian simbol penandaan diatas yaitu menggunakan perintah Xref untuk membuka block rambu film. Bentuk program untuk rambu film dapat dilihat pada Lampiran B.2.18.

Berikut ini beberapa bentuk sub-routine program yang mendukung pemrograman utama:

Program Pembuatan Animasi Pelat Hasil Potongan

Program animasi ini memanfaatkan perintah *vpoint* dari AutoCAD yang diputar (rotate) pada sudut tertentu. Pelat hasil potongan diperoleh dengan cara membuat block pada langkah setelah pemotongan pelat. Bentuk program pembuatan block dan animasi potongan pelat dapat dilihat pada Lampiran B.2.19.

Program Pembuatan Kotak Dialog (Dialog Box)

Kotak dialog dibuat untuk memperindah tampilan perintah AutoCAD sehingga mempermudah para pengguna untuk melakukan suatu perintah pada AutoCAD. Dengan hanya menekan tombol (button) pada kotak dialog maka AutoCAD akan menjalankan perintah tersebut. Hal ini berlawanan bila tidak

menggunakan kotak dialog dimana pengguna lebih sering menuliskan perintah untuk menjalankan AutoCAD. Kotak dialog mempunyai ekstension *dcl*. Bentuk sub-routine program pembuatan kotak dialog dapat dilihat pada Lampiran B.2.20.

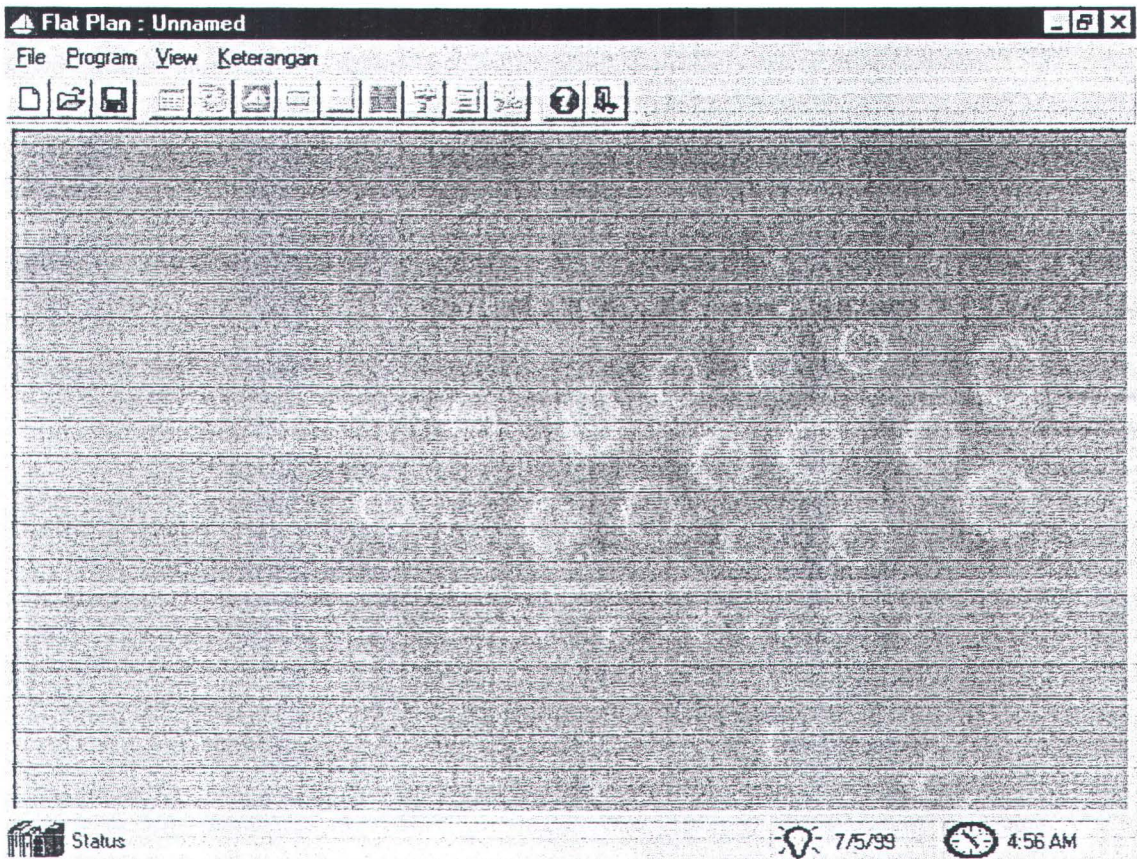
Penjelasan Singkat Pemrograman Microsoft Visual Basic

Microsoft Visual Basic dalam pemrograman bukaan kulit ini digunakan sebagai program tampilan terhadap pemrograman AutoLISP yang merupakan program proses penggambaran bukaan kulit. Fungsi Microsoft Visual Basic dalam pemrograman bukaan kulit antara lain:

- Tampilan input data (data ukuran utama dan data kurva)
- Running program AutoLISP dengan otomatisasi proses perintah AutoLISP yaitu dengan pembuatan tombol-tombol (button) yang mewakili perintah AutoLISP.
- Tampilan hasil bukaan kulit.
- Tampilan dimensi hasil bukaan kulit yang merupakan data numerik untuk dijadikan sebagai data acuan pada tahap pemotongan pelat.

Tampilan program seperti disebutkan diatas dalam Microsoft Visual Basic berbentuk *form* dimana didalamnya berisi elemen visual dan beberapa kontrol. Bentuk form bukaan kulit berisi form-form berikut ini:

⇒ **Form Program Utama** : berisi menu-menu utama program bukaan kulit metode flat plan dan tampilan proses bukaan kulit pada layar AutoCAD. Listing form program utama dapat dilihat pada Lampiran B.2.22.

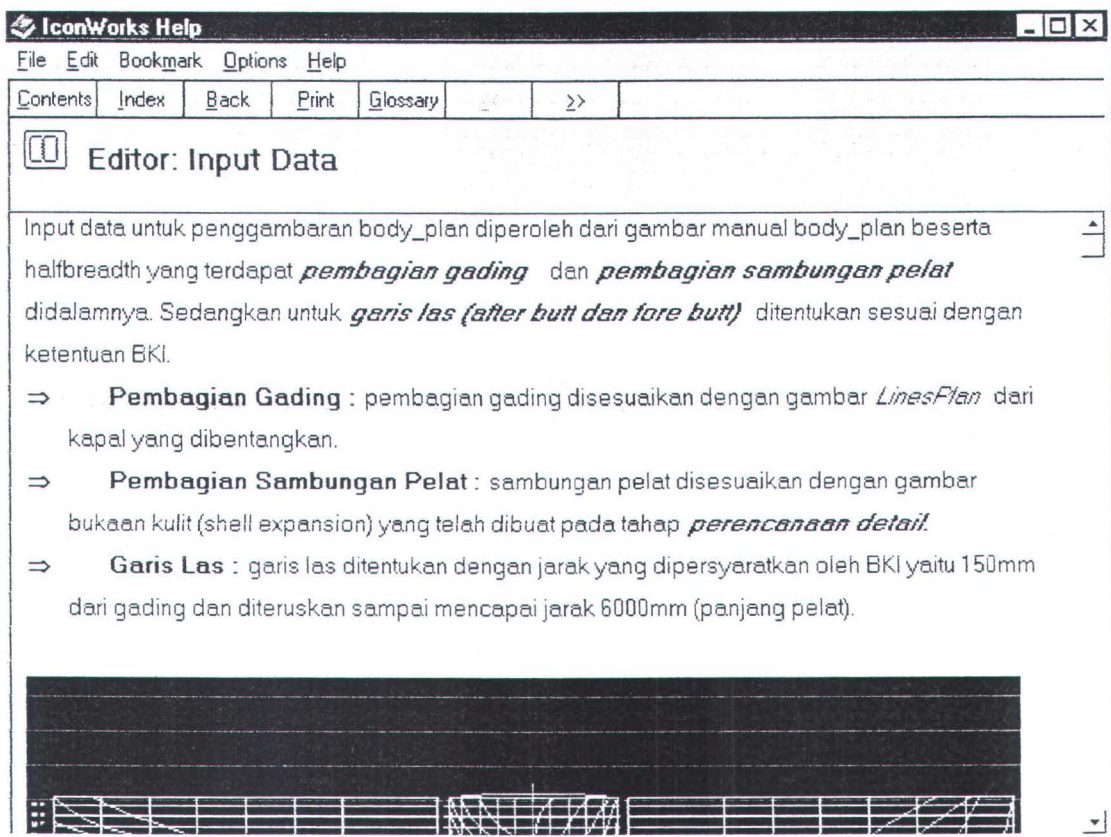


Gambar 3.15. Form Tampilan Program Utama.

Keterangan Gambar:

- ◆ **File** : berisi sub-menu editor untuk memasukkan input (buka) dengan ekstension file *.dat, sub-menu editor untuk penyimpanan data dan gambar (simpan), sub-menu editor untuk export file (*.dxf), sub-menu editor untuk mencetak gambar (plot), dan sub-menu editor untuk keluar program utama.
- ◆ **Program** : berisi sub-menu editor untuk menampilkan tabel data koordinat penggambaran body-plan dan bukaan kulit, sub-menu editor untuk proses bukaan kulit, penggambaran body plan, tampilan potongan pelat, tampilan bentuk sebenarnya pelat hasil bukaan, tampilan rambu film/marketing list, animasi pelat, dan tampilan data dimensi sebenarnya.
- ◆ **View** : sub-menu editor untuk penampilan dan penghilangan toolbar dan statusbar.

- ◆ **Keterangan** : berisi penjelasan-penjelasan mengenai program bukaan kulit metode flat plan, penjelasan-penjelasan mengenai cara-cara pemakaian program. Bentuk menu editor ini dibuat seperti *help menu* pada window.



Gambar 3.16. Tampilan Keterangan Program Bukaan Kulit Metode Flat Plan.

- ⇒ **Form Tabel Data Koordinat** : berisi data ukuran utama kapal, data keterangan bukaan kulit, data koordinat gading dan sambungan pelat, data simbol penandaan, tombol untuk menampilkan menu help, dan tombol untuk kembali ke program utama. Input data koordinat diperoleh dari perintah buka file pada *form program utama* dengan menggunakan ekstension file *.dat. Input data koordinat ini disesuaikan dengan input data koordinat pada program AutoLISP. Bentuk form tabel data koordinat dapat dilihat pada Gambar 3.17. Listing form tabel data koordinat dapat dilihat pada Lampiran B.2.23.

Tabel Data Bukaan Kulit Metode Flat Plan

Data Ukuran Utama Kapal

Panjang Kapal : 54 m

Lebar Kapal : 9.5 m

Tinggi Kapal : 5.8 m

Radius Bilga : 1.15 m

Jarak Gading : 0.6 m

Simbol Penandaan

Tanda Posisi Tanda Potong Identitas

No. Gading Tanda Ketebalan Plate SS/PS

Data Koordinat Gading X (m)

Text :

	After Butt	Gading 2
WL 1	0.35	0.35
WL 2	0.992	1.025
WL 3	1.633	1.7

Koordinat Sambungan Pelat X (m)

Text :

	Lower Seam	Upper S
After Butt	0.529	0.87
Gd. 21	0.539	0.91
Gd. 22	0.573	1.023

Data Koordinat Gading Y (m)

Text :

	After Butt	Gading 2
WL 1	0	0
WL 2	0.032	0.032
WL 3	0.064	0.064

Koordinat Sambungan Pelat Y (m)

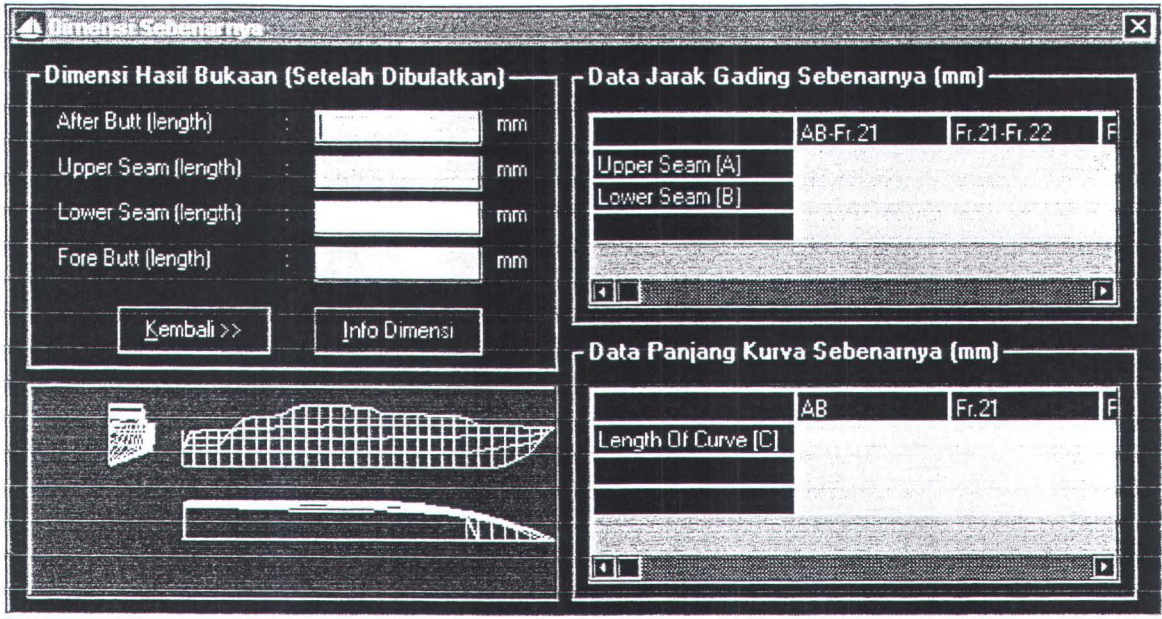
Text :

	Lower Seam	Upper S
After Butt	0.41	0.72
Gd. 21	0.39	0.7108
Gd. 22	0.32	0.6814

Info Body
Help
Kembali>

Gambar 3.17. Form Tabel Data Koordinat.

⇒ **Form Dimensi Sebenarnya** : berisi tabel data dimensi sebenarnya dari hasil bukaan kulit metode flat plan yaitu panjang sebenarnya dari after butt, fore butt, upper seam, lower seam, gading dan jarak gading pada upper seam dan lower seam. Tabel data dimensi sebenarnya ini berasal dari pengukuran dimensi pada program AutoLISP melalui perintah *hasil_buka*. Form dimensi sebenarnya ini dilengkapi dengan tombol info dimensi yang berisi keterangan masalah pengukuran dimensi sebenarnya pada hasil bukaan kulit. Dengan melihat tabel data dimensi sebenarnya, maka pengguna program bukaan kulit dapat mengerti dimensi sebenarnya dari hasil bukaan kulit. Listing form dimensi sebenarnya dapat dilihat pada Lampiran B.2.24, sedangkan form dimensi sebenarnya dapat dilihat pada Gambar 3.18.



Gambar 3.18. Form Dimensi Sebenarnya.

Cara Microsoft Visual Basic berkomunikasi dengan AutoCAD merupakan cara yang harus diperhatikan sehingga komunikasi kedua software tersebut dapat berjalan dengan baik. Perintah untuk menghubungkan kedua software tersebut menggunakan *Automation* dimana dengan menggunakan perintah tersebut, AutoCAD diaktifkan secara otomatis melalui Microsoft Visual Basic. Tampilan layar AutoCAD dapat diatur sehingga tampilan layar AutoCAD dapat tampil tepat pada layar Visual Basic yaitu dengan mengatur kondisi top, left, width dan juga height dari layar AutoCAD. Listing program untuk otomatisasi layar AutoCAD pada program Visual Basic dapat dilihat pada Lampiran B.2.25.

Running Program Keseluruhan Sebagai Petunjuk Teknis Penggunaan Program

- ➊ **Input Data :** Nama file sebagai input data koordinat penggambaran body plan dibuka dengan menu editor **Buka** atau dapat juga melalui ToolBar (Buka) yang disediakan pada Program Utama. File yang digunakan dalam program ini menggunakan ekstension *.bat. Pengguna juga dapat memasukkan input data

koordinat selain data koordinat yang ada (dari text editor) yaitu terlebih dahulu memasukkan data-data sebagai berikut pada **Form Tabel** : ukuran utama kapal, data bukaan kulit (jumlah gading yang dibentangkan, jumlah titik pembacaan, no. awal gading, jarak after butt dan fore butt, simbol-simbol penandaan). Kemudian data-data tersebut disimpan dengan menggunakan menu editor **Simpan** dengan nama file yang lain (misal : kanan1.bat, kanan2.bat dst).

- ② **Proses Bukaan Kulit** : setelah input data ada, maka data tersebut diproses oleh program yaitu dimulai dengan menggambarkan body plan, memilih daerah pelat yang akan dibentangkan, memotong pelat, melakukan proses metode flat plan dan seterusnya, dimana langkah proses ini telah diterangkan pada sub-bab langkah-langkah pembuatan program.
- ③ **Output Proses** :Setelah proses bukaan kulit dilakukan, maka hasil-hasil proses tersebut direcord untuk ditampilkan yaitu record untuk pelat hasil pemotongan, bentuk body plan, hasil bukaan kulit, rambu film. Record tersebut diwakili oleh menu editor proses, dimana didalam menu tersebut terdapat sub-menu editor tampilan body plan, tampilan potongan pelat, tampilan hasil bukaan kulit, tampilan hasil bukaan kulit dalam rambu film. Output bukaan kulit metode flat plan ini dapat dicetak sesuai skala penggambaran yang diinginkan dengan menggunakan menu editor **Plot Gambar** dan dapat juga diekspor/ditransfer dalam bentuk file dengan ekstension *.dxf dengan menggunakan menu editor **Kirim File** sehingga file gambar tersebut dapat diproses oleh NC-CUTTING. Selain menghasilkan bentuk sebenarnya dari pelat yang dibentangkan, program bukaan kulit ini juga dapat menampilkan dimensi sebenarnya dari hasil bukaan kulit,

dimana dimensi sebenarnya dapat dilihat secara langsung pada **Form Dimensi Sebenarnya** sehingga pengguna program bukaan kulit dapat melihat dimensi sebenarnya hasil bukaan kulit.

BAB IV

OTOMATISASI BUKAAN KULIT METODE FLAT PLAN

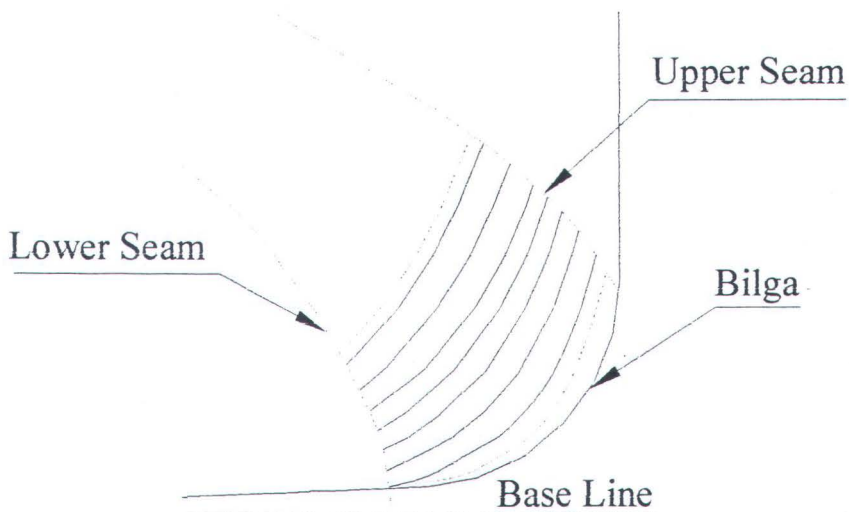
Masalah akurasi hasil bukaan kulit memerlukan perhatian yang khusus dari pihak mouldloft karena ketidakteelitian akan mengakibatkan kesalahan-kesalahan pada tahap-tahap pembangunan kapal yaitu mulai dari fabrikasi sampai ereksi. Oleh karena itu, pihak mouldloft sejak dini harus membuat langkah-langkah yang nyata dalam hal menanggulangi kesalahan-kesalahan yang akan terjadi. Contoh langkah-langkah awal dalam menanggulangi kesalahan-kesalahan tersebut diantaranya adalah adanya kesesuaian antara dimensi pada marking list dengan gambar kerja (working drawing) yang dihasilkan oleh perencanaan detail (functional plans). Sehingga pada tahap pengerjaan tidak terjadi pembuangan ataupun penambahan material.

Software program komputer bukaan kulit yang dibuat memberikan suatu pemecahan permasalahan akurasi (ketelitian) baik bentuk maupun dimensi pelat hasil bukaan. Cara pemecahan permasalahannya adalah dengan membandingkan hasil bukaan kulit yang dikerjakan secara manual (tenaga manusia) dengan hasil bukaan kulit dari komputer atau melakukan proses *matching* yaitu melakukan penyesuaian sambungan dua pelat hasil bukaan kulit. Pada bab ini akan diterangkan masalah akurasi hasil bukaan dan penjelasan masalah kelebihan dan kekurangan program komputer yang dibuat sehingga dapat memberikan kejelasan kepada pembaca mengenai pemakaian program komputer bukaan kulit yang benar dan tepat sehingga hasil bukaan kulit yang diinginkan sesuai dengan kebutuhan pelat dilapangan dan juga sekaligus mengurangi pembuangan material pelat yang sia-sia.

IV. 1 Akurasi Hasil Bukaan Kulit Metode Flat Plan

Proses akurasi (ketelitian) hasil bukaan kulit sangat erat hubungannya dengan masalah *pemilihan metode bukaan kulit* dan pemilihan metode bukaan kulit tergantung dari *bentuk kurva yang akan dibentangkan*. Oleh karena itu pengetahuan mengenai penggunaan metode-metode bukaan kulit harus diketahui sejak dini sehingga dapat mencegah kesalahan akibat penggunaan metode bukaan kulit yang salah. Pada tugas akhir ini penulis menggunakan *Metode Bukaan Kulit Flat Plan* yang memiliki ciri penggunaannya (Santoso, 1992), yaitu:

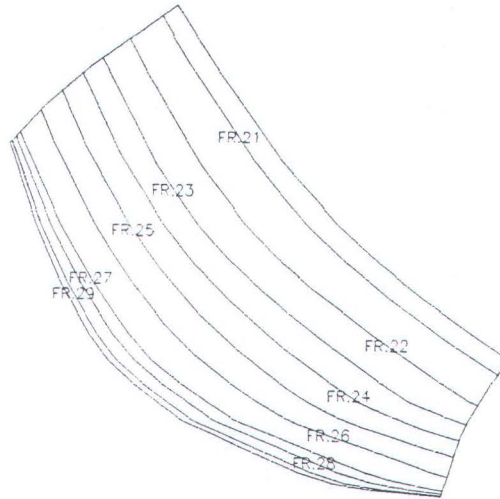
- Luasan flat adalah lebih besar dari 40% dalam sebuah pelat.
- Bagian flat tidak ditransformasikan dengan bending stress.
- Bentuk kurva yang dapat dibentangkan adalah pada posisi bilga atau pada bagian yang agak rata dengan sebagian yang agak lengkung (lihat Gambar 4.1).



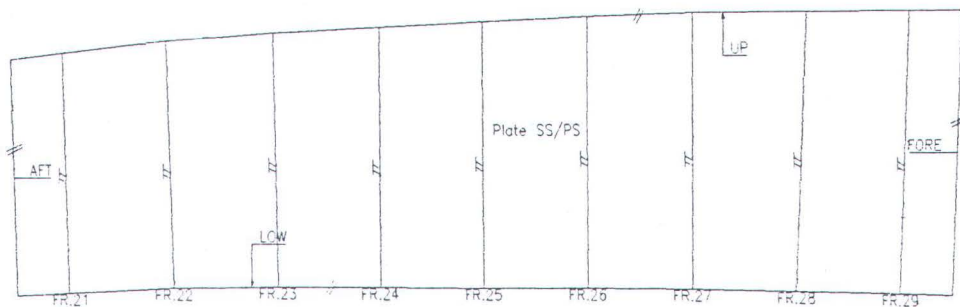
Gambar 4.1. Posisi Kurva Gading Metode Flat Plan.

Dengan mengerti ciri penggunaan metode flat plan diatas maka pemakai (user) dari program komputer bukaan kulit metode flat plan ini dapat mengerti dalam memilih bentuk kurva gading yang akan dibentangkan sehingga output (hasil) bukaan

kulit (rambu film) sesuai dengan gambar kerjanya. Untuk lebih memahami metode flat plan, Gambar 4.2 memperlihatkan bentuk potongan pelat, sedangkan Gambar 4.3 memperlihatkan hasil bukaan kulit pelat tersebut.



Gambar 4.2. Potongan Pelat.



Gambar 4.3. Hasil Bukaan Kulit dari Potongan Pelat dalam Gambar 4.2 dengan Metode Flat Plan.

Terlihat pada Gambar 4.2 pelat hasil pemotongan masih berupa lengkung kurva yang sangat sulit untuk ditentukan bentuk dan dimensi sebenarnya sehingga

tidak mungkin dilakukan langkah marking dan cutting pada pelat yang akan digunakan. Tetapi setelah pelat tersebut dibuka/dibentangkan dengan metode flat plan maka terlihat bentuk dan dimensi dari bentuk awal pelat yaitu bentuk datar, sehingga mempermudah pengerjaan pada tahap marking dan cutting. Sebagai catatan, pelat hasil bukaan kulit metode flat plan tidak memiliki back sheet (tinggi kelengkungan) sehingga garis kontrol (check line) tidak diperlukan untuk memeriksa kelengkungan pelat agar sesuai dengan kelengkungan lambung kapal.

Proses akurasi hasil bukaan kulit dapat dilakukan dengan melakukan proses *matching* yaitu melakukan penyesuaian antara dua pelat hasil bukaan kulit. Cara melakukan matching dalam hal ini adalah dengan mempertemukan sambungan antara kedua pelat hasil bukaan yaitu antara fore butt joint pelat I dengan after butt joint pelat II.

Apabila sambungan dua pelat hasil bukaan kulit cocok maka dapat dikatakan bahwa hasil bukaan kulit sesuai dengan yang diinginkan. Cara lain untuk membuktikan kebenaran hasil bukaan kulit dari pemrograman komputer adalah membandingkan hasil bukaan kulit secara manual dan hasil bukaan kulit secara komputer dengan gambar kerja. Pada dasarnya hasil bukaan kulit telah ditentukan dan direncanakan pada gambar kerja sehingga hasil bukaan kulit dari mouldloft harus disesuaikan dengan gambar kerja. Dimensi hasil bukaan kulit dari pemrograman komputer dibuat dalam bentuk Form Dimensi Sebenarnya Hasil Bukaan Kulit Pelat (lihat Gambar 4.6) dimana dalam tabel tersebut semua dimensi bukaan kulit yaitu panjang after butt dan fore butt, panjang upper seam dan lower seam, panjang gading-gading berada didalamnya. Dimensi-dimensi tersebut yang akan dijadikan acuan dasar dalam perbandingan hasil bukaan secara manual dengan secara komputer. Berikut ini Data

Dimensi Sebenarnya Hasil Bukaan Kulit Metode Flat Plan secara manual dari KM BUNGA TERATAI 55 dengan ukuran utama kapal sebagai berikut:

- Panjang Kapal (Lpp) = 54 m
- Lebar Kapal (Bmld) = 9.5 m
- Tinggi Kapal (D dk) = 5.8 m
- Sarat Kapal (d) = 3.7 m
- Radius Bilga = 1.115m
- Rise of Floor = 0.21 m

Data Dimensi Hasil Bukaan Kulit Metode Flat Plan (secara manual) pada No.

Gading 21 - 29

- Panjang sambungan batas atas (upper seam) = 5519 mm
- Panjang sambungan batas bawah (lower seam) = 5406 mm
- Panjang after butt joint = 1618 mm
- Panjang fore butt joint = 1960 mm

Data panjang sebenarnya kurva gading dan las (mm)

	AB	Fr.21	Fr.22	Fr.23	Fr.24	Fr.25	Fr.26	Fr.27	Fr.28	Fr.29	FB
Panjang Kurva	1618	1644	1697	1743	1787	1830	1867	1899	1923	1948	1960

Tabel 4.4. Data Panjang Sebenarnya Kurva Gading Dan Las (cara manual).

Data Dimensi Hasil Bukaan Metode Flat Plan (komputer) pada No. Gading 21 -

29

- Panjang sambungan batas atas (upper seam) = 5501 mm
- Panjang sambungan batas bawah (lower seam) = 5401 mm

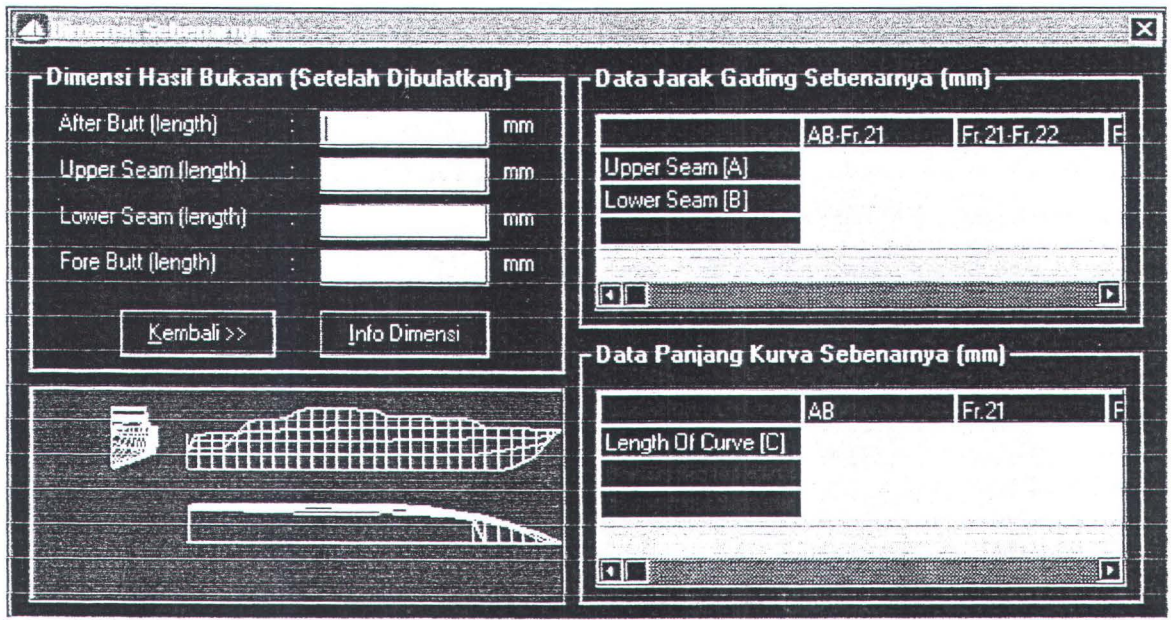
- Panjang after butt joint = 1620 mm
- Panjang fore butt joint = 1980 mm

Data panjang sebenarnya kurva gading dan las (mm)

	AB	Fr.21	Fr.22	Fr.23	Fr.24	Fr.25	Fr.26	Fr.27	Fr.28	Fr.29	FB
Panjang Kurva	1620	1650	1699	1745	1790	1839	1869	1900	1930	1950	1980

Tabel 4.5. Data Panjang Sebenarnya Kurva Gading Dan Las (komputer).

Berikut ini Form Dimensi Sebenarnya Hasil Bukaannya Kulit Metode Flat Plan dari data kapal diatas:



Gambar 4.6 Tampilan Form Dimensi Sebenarnya Hasil Bukaannya Kulit Pelat

No. Gading 21 - 29

Data Dimensi Bukaannya Kulit dari Gambar Kerja pada Functional Plans (Shell Expansion)

- Panjang sambungan batas atas (upper seam) = 5500 mm
- Panjang sambungan batas bawah (lower seam) = 5400 mm

- Panjang after butt joint = 1620 mm
- Panjang fore butt joint = 1980 mm

Data panjang sebenarnya kurva gading dan las (mm)

	AB	Fr.21	Fr.22	Fr.23	Fr.24	Fr.25	Fr.26	Fr.27	Fr.28	Fr.29	FB
Panjang Kurva	1620	1651	1699	1745	1791	1839	1869	1900	1932	1951	1980

Tabel 4.7. Data Panjang Sebenarnya Kurva Gading Dan Las (gambar kerja).

Keterangan Dimensi Hasil Bukaak Kulit :

- Data dimensi sebenarnya hasil bukaak kulit secara manual cenderung melenceng (kurang atau lebih dari hasil yang diinginkan gambar kerja). Pada data dimensi hasil bukaak kulit yang salah dilakukan langkah pembuatan Feed Back Sheet (umpan balik) dimana dalam Feed Back Sheet terdapat penunjukkan dimensi yang salah disertai dengan metode perbaiki (dengan gambar) dan juga beberapa tindakan pencegahan. Contoh Feed Back Sheet dapat dilihat pada Lampiran C.
- Data dimensi sebenarnya hasil bukaak kulit dari komputer menunjukkan kesesuaian dengan dimensi yang diminta pada gambar kerja (working drawing) meskipun dengan sedikit toleransi kesalahan (lebih atau kurang dari 1mm). Kemungkinan kelebihan dimensi karena input data program bukaak kulit masih diambil dari gambar body plan manual.

IV.2 Kelebihan dan Kekurangan Program Bukaak Kulit Metode Flat Plan

Untuk lebih memahami penggunaan program bukaak kulit ini maka perlu pengetahuan tentang kelebihan dan kekurangan program bukaak kulit metode flat plan. Hal ini dimaksudkan agar dalam penggunaan program bukaak kulit metode flat

plan tidak terjadi kesalahan sehingga akan mengakibatkan suatu masalah baru yang sebenarnya tidak perlu terjadi.

Kelebihan Program Bukaank Kulit Metode Flat Plan:

- Dapat menggunakan data kapal yang berbeda-beda untuk mendapatkan bukaan kulit kapal sehingga dapat menghemat waktu pembangunan kapal.
- Program bukaan kulit metode flat plan dapat menggambarkan body plan kapal dalam segala macam skala sehingga akan mempermudah dalam mendapatkan hasil bukaan kulit yang diinginkan.
- Program bukaan kulit metode flat plan dapat dihubungkan dengan mesin pemotong (cutting machine) dengan mentransfer file dengan ekstension dxf.
- Tampilan dimensi sebenarnya (actual dimension) dan tampilan bentuk sebenarnya (actual form) dapat memperjelas para pemakai program dalam mengetahui hasil bukaan kulit.
- Hasil (output) bukaan kulit dari program bukaan kulit yang berupa rambu film dapat dicetak melalui perintah *print* yang telah disediakan dalam program.

Kekurangan Program Bukaank Kulit Metode Flat Plan :

- Penggunaan program bukaan kulit ini hanya terbatas pada satu metode bukaan kulit mouldloft yaitu metode flat plan dimana posisi kurva gading terletak pada daerah bilga. Sehingga apabila pemakai program ini akan mendapatkan kurva gading pada daerah stem, stern dan lain-lain maka akan terjadi kesalahan hasil bukaan kulitnya.
- Data body plan masih menggunakan data body plan manual yang diperoleh pada perencanaan dasar (Base Plan). Data body plan tersebut telah dilengkapi dengan gambar sambungan pelat dan pembagian gading-gadingnya.

- Pembacaan data body plan melalui file yang selanjutnya data tersebut ditampilkan oleh Microsoft Visual Basic. Sehingga apabila pemakai akan menggunakan program dengan data kapal yang berbeda maka pemasukan data melalui file yang selanjutnya file tersebut diproses oleh program AutoLISP untuk mendapatkan hasil bukaan kulit. Jadi data penggambaran tidak dapat secara langsung dimasukkan melalui tampilan input data yang telah dibuat melalui program Microsoft Visual Basic dan menghasilkan gambar body plan.

BAB V

KESIMPULAN DAN SARAN

Setelah melalui tahapan-tahapan permasalahan, metodologi pemecahan permasalahan, dan hasil pemecahan permasalahan seperti yang telah dijelaskan pada bab-bab sebelumnya, maka penulis menarik suatu kesimpulan dari pembuatan program bukaan kulit metode flat plan dan beberapa saran yang dapat diberikan terhadap masalah pembuatan program komputer dan beberapa saran umum.

V. 1 Kesimpulan

Beberapa kesimpulan dari pemrograman bukaan kulit metode flat plan dapat dijelaskan pada penjelasan-penjelasan berikut:

- Kesalahan-kesalahan masalah ketidakteelitian hasil bukaan kulit secara manual baik kesalahan dimensi maupun bentuk sebenarnya dari pelat yang dibentangkan, dapat dikurangi dengan penerapan metode bukaan kulit yang benar pada program komputer yaitu dengan memperhatikan bentuk kurva yang akan dibentangkan sehingga dapat mengurangi “reworks” pada tahap fabrikasi dan assembly.
- Untuk penggambaran body plan yang biasanya menggunakan fasilitas lantai mouldloft yang sangat luas dimana penggambaran body plan disesuaikan dengan skala kapal yang dibuat (skala 1:1) maka dengan menggunakan program komputer bukaan kulit metode flat plan penggambaran body plan dapat digambar dengan bermacam-macam skala sesuai skala kapal yang dibuat sehingga hal ini sangat menghemat waktu dan tempat.

- Penggambaran bukaan kulit metode flat plan dalam program komputer ini dapat menghemat waktu dimana penggambaran garis-garis proyeksi, garis-garis tegak lurus kurva, panjang kurva, panjang garis sambungan pelat (upper seam dan lower seam) maupun pengukuran dimensi jarak gading sebenarnya dapat ditentukan secara cepat setelah pelat yang diinginkan dipotong. Pekerjaan penggambaran metode flat plan ini sangat lama apabila dikerjakan secara manual.
- Bentuk dan dimensi sebenarnya dari pelat yang dibentangkan dapat diperoleh secara cepat dengan menggunakan program komputer bukaan kulit metode flat plan dan memperhatikan cara-cara penggunaan yang benar dari program bukaan kulit metode flat plan tersebut.
- Hasil bukaan kulit metode flat plan dalam bentuk rambu film dapat langsung dicetak sesuai skala penggambaran bukaan kulit.
- Dengan menggunakan data kapal yang berbeda-beda dan mengacu pada pembentangan pelat bilga kapal tersebut, program komputer bukaan kulit metode flat plan dapat digunakan sehingga waktu pembangunan kapal dapat dihemat.
- Hasil bukaan kulit metode flat plan dapat ditransfer ke NC-Cutting dalam bentuk file gambar dengan ekstension dxf sehingga hal ini sangat efisien dalam pembangunan kapal baru.

V. 2 Saran-Saran.

Beberapa saran yang dapat diberikan penulis tentang pembuatan program komputer bukaan kulit metode flat plan adalah sebagai berikut:

- Input data body plan masih menggunakan data gambar manual bukan tabel offset karena hal ini sangat berpengaruh terhadap kelengkapan penggambaran kurva gading. Oleh karena itu, input data body plan secara otomatis dimana didalamnya terdapat gambar kurva gading-gading dan kurva sambungan pelat sangat diharapkan untuk menunjang kemampuan program dalam mengatasi permasalahan akurasi bukaan kulit.
- Pembuatan program masih terpisah yaitu antara Program Microsoft Visual Basic yang merupakan program tampilan dan Program AutoLISP yang merupakan program proses bukaan kulit dimana input data diambil dalam bentuk file sehingga diharapkan pembuatan program yang satu padu artinya apabila seorang pengguna akan menggambar body plan dan ingin mendapatkan hasil bukaan kulitnya dapat langsung menuliskan data body plan pada program tampilan dan langsung diproses untuk mendapatkan gambar body plan dan hasil bukaan kulitnya.
- Penulis sangat mengharapkan pembuatan program bukaan kulit dengan menggunakan semua metode bukaan kulit yang ada di mouldloft seperti metode base-line, metode roll line, metode squaring, metode flat plan, metode stem/stern, metode bukaan pelat margin, metode bukaan profil, dan metode bukaan gading dapat dibuat menjadi kesatuan program dan apabila hal ini terlaksana maka sangat membantu pekerjaan mouldloft di galangan.

DAFTAR PUSTAKA

Gesner, R, Maximing AutoLISP, USA 1992.

Guide, P, Microsoft Visual Basic, USA 1995.

Hadisiswoyo, S, Computer Aided Design & Drafting (CADD), Surabaya 1992.

Heriyanto, H, Peranan Pekerjaan Moudloft Pada Proses Pembangunan Kapal, Surabaya 1989.

Imron, A, Computer Aided Marine Design, Surabaya 1995.

Omura, G, Mastering AutoCAD 13 For Windows 95, Windows 3.1, and Windows NT, USA 1995.

Purwanto, M. D., Pedoman Pemakaian AutoLISP, Yogyakarta 1993.

Sentosa, P. I., Mould Lofting Untuk Konstruksi Lambung Kapal, Surabaya 1992.

Soeyitno, Teknologi Bangunan Baru I, Surabaya 1986.

DAFTAR LAMPIRAN

Lampiran A

Lampiran B

Lampiran C

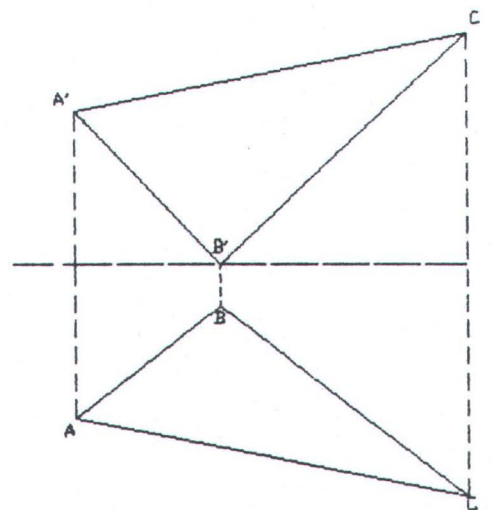
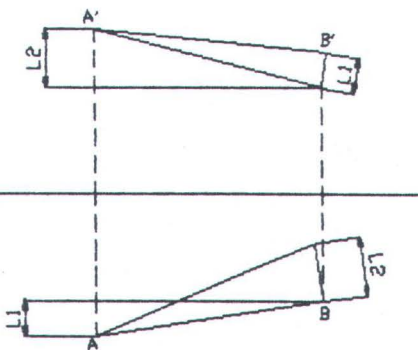
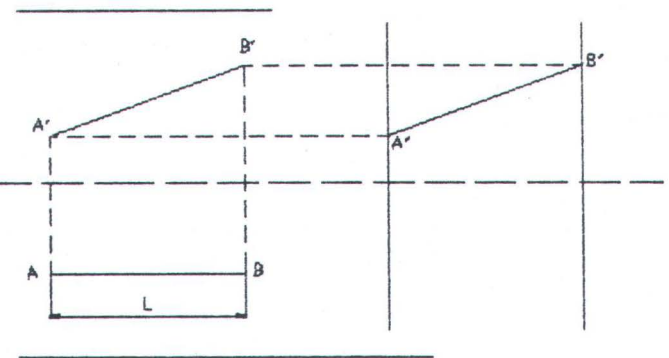
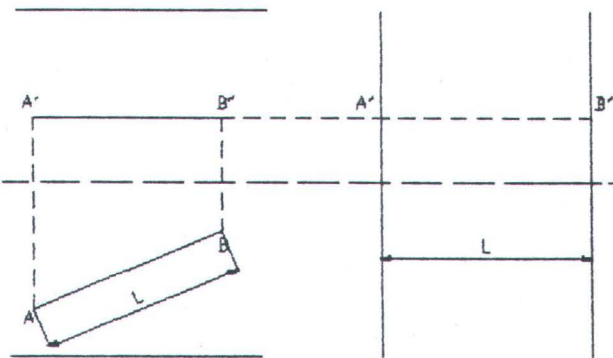
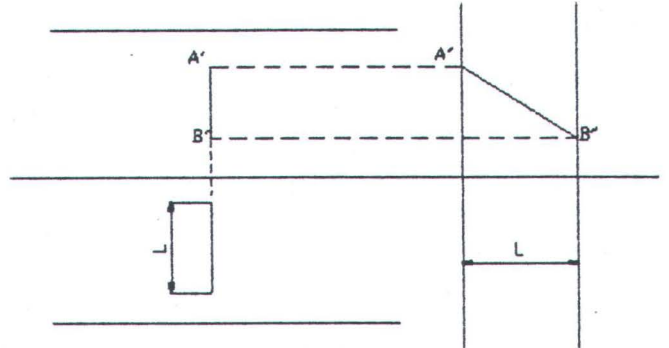
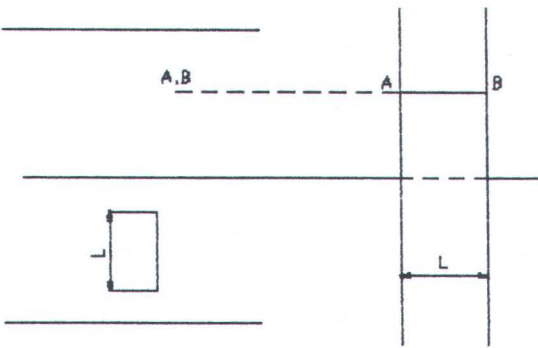
Lampiran D



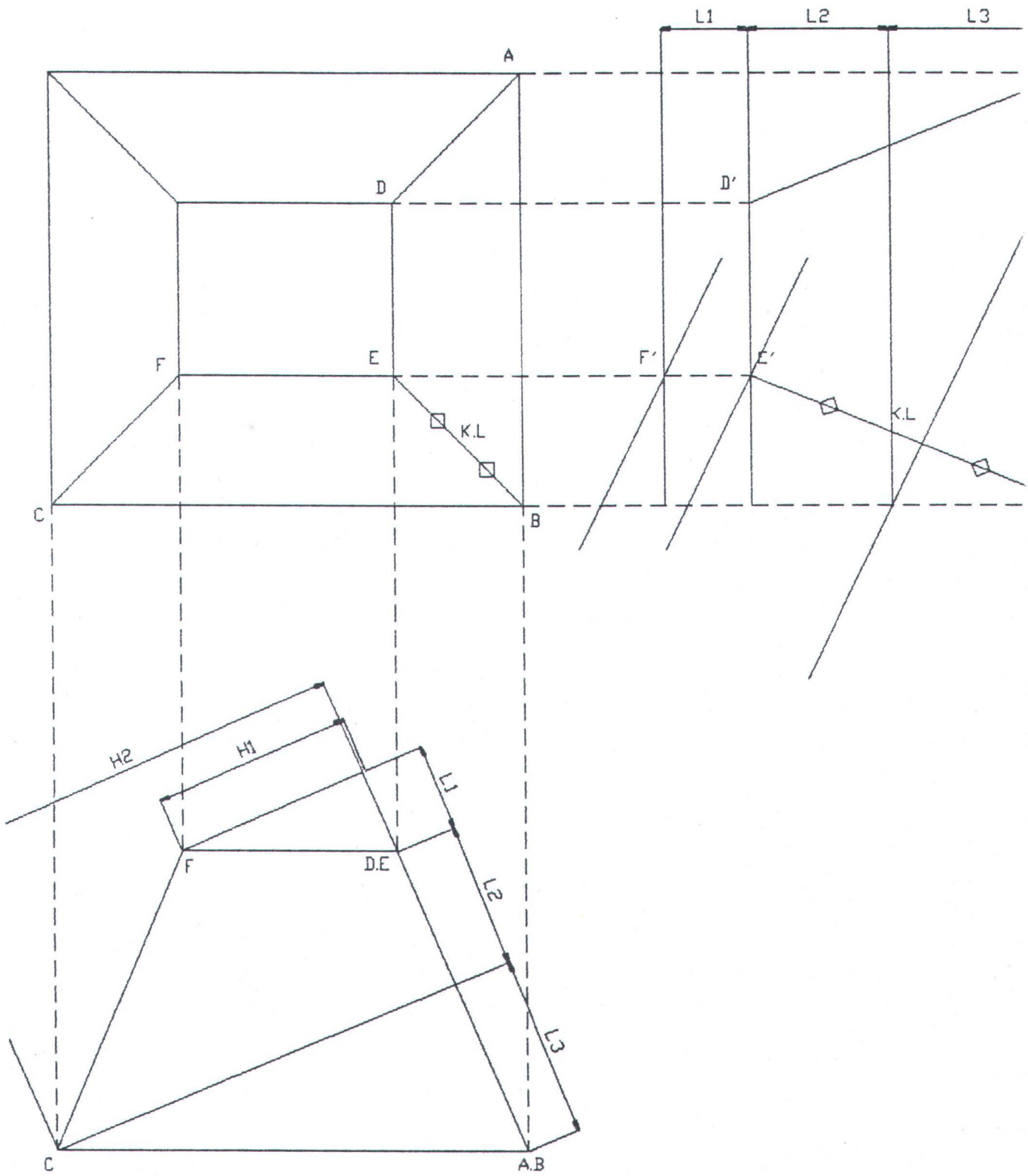
LAMPIRAN A

Lampiran A.1 (Proyeksi)

a). Bukaan Dasar I

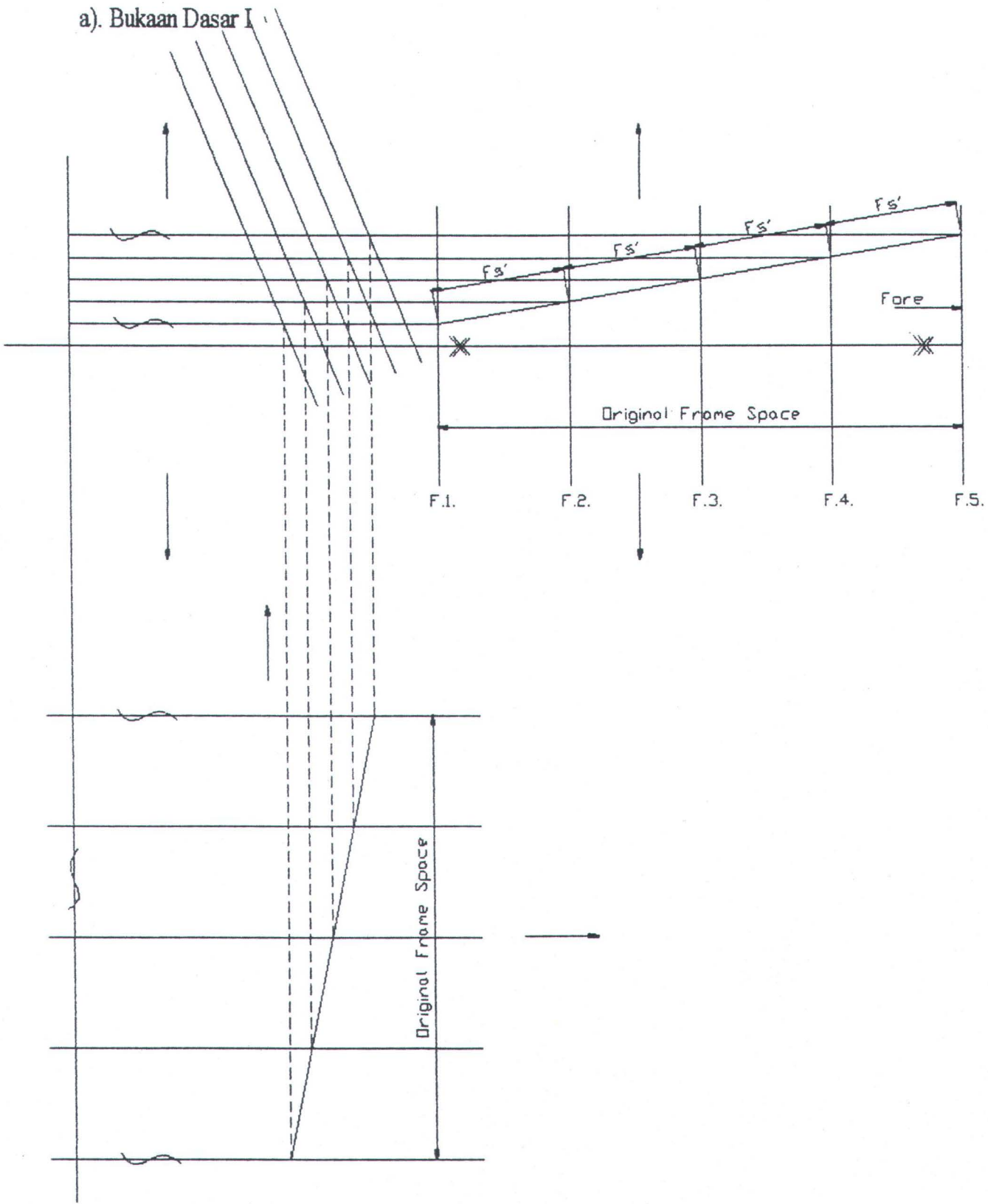


b). Proyeksi II

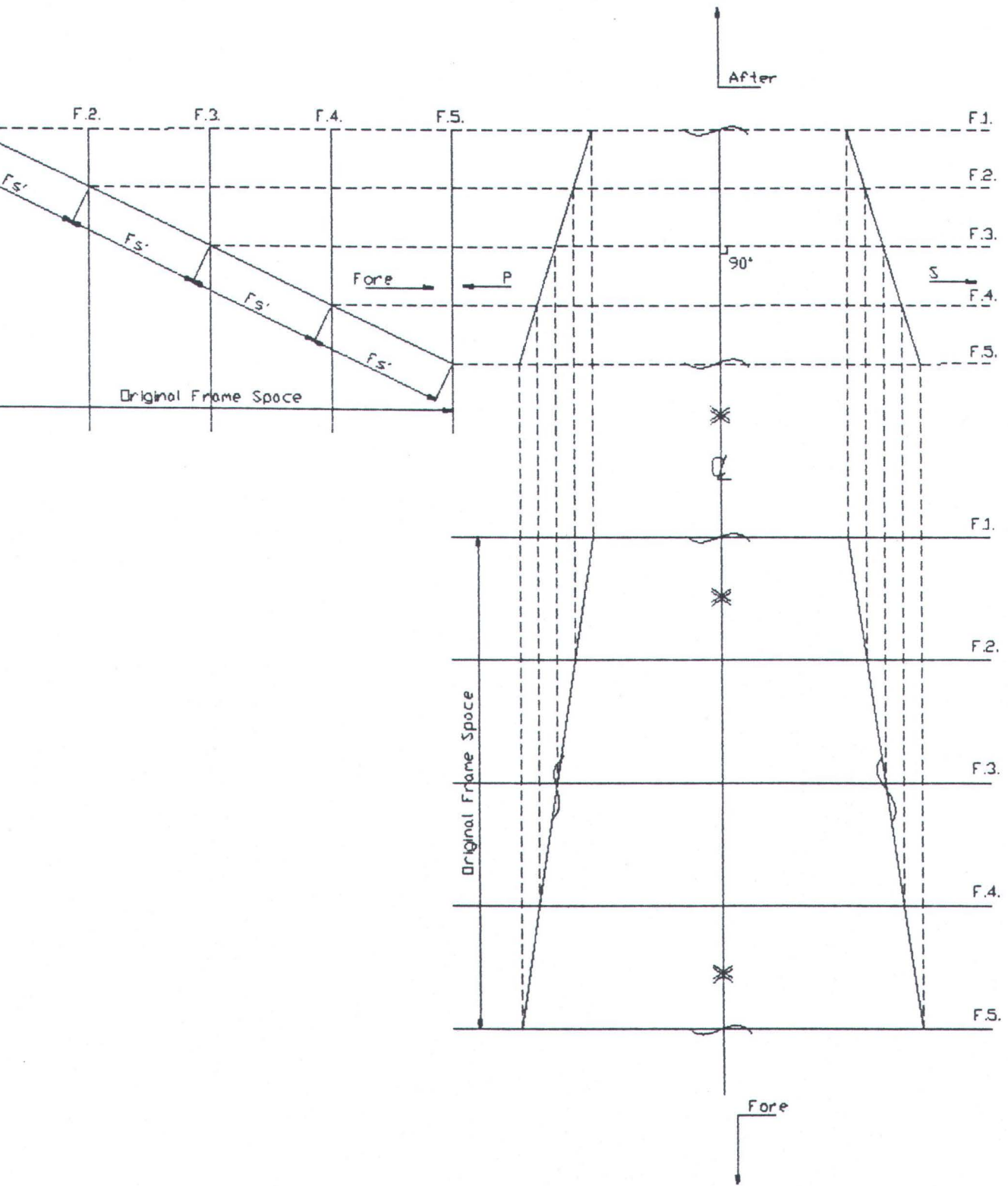


Lampiran A.2 (Bukaan Dasar)

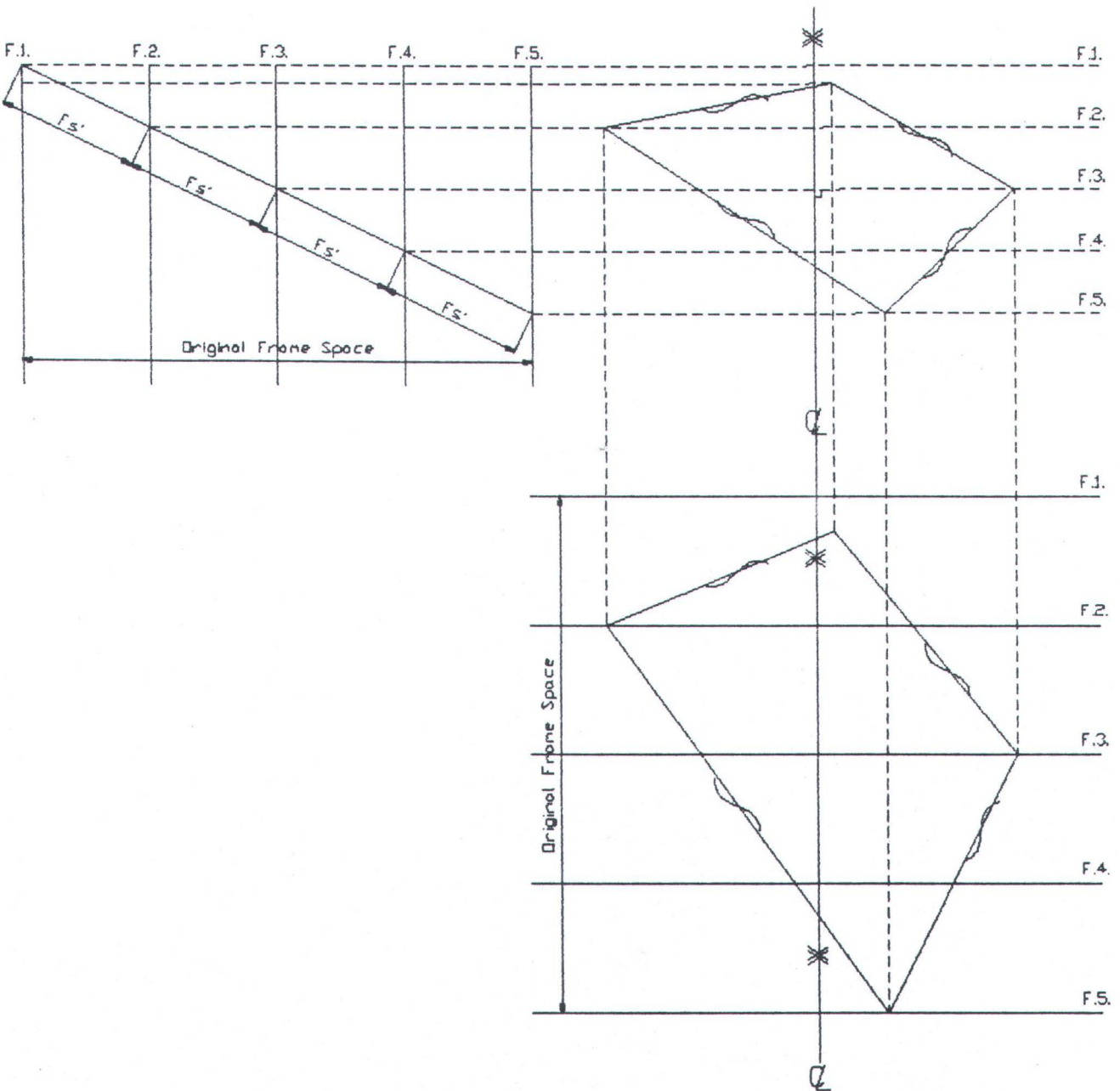
a). Bukaan Dasar I



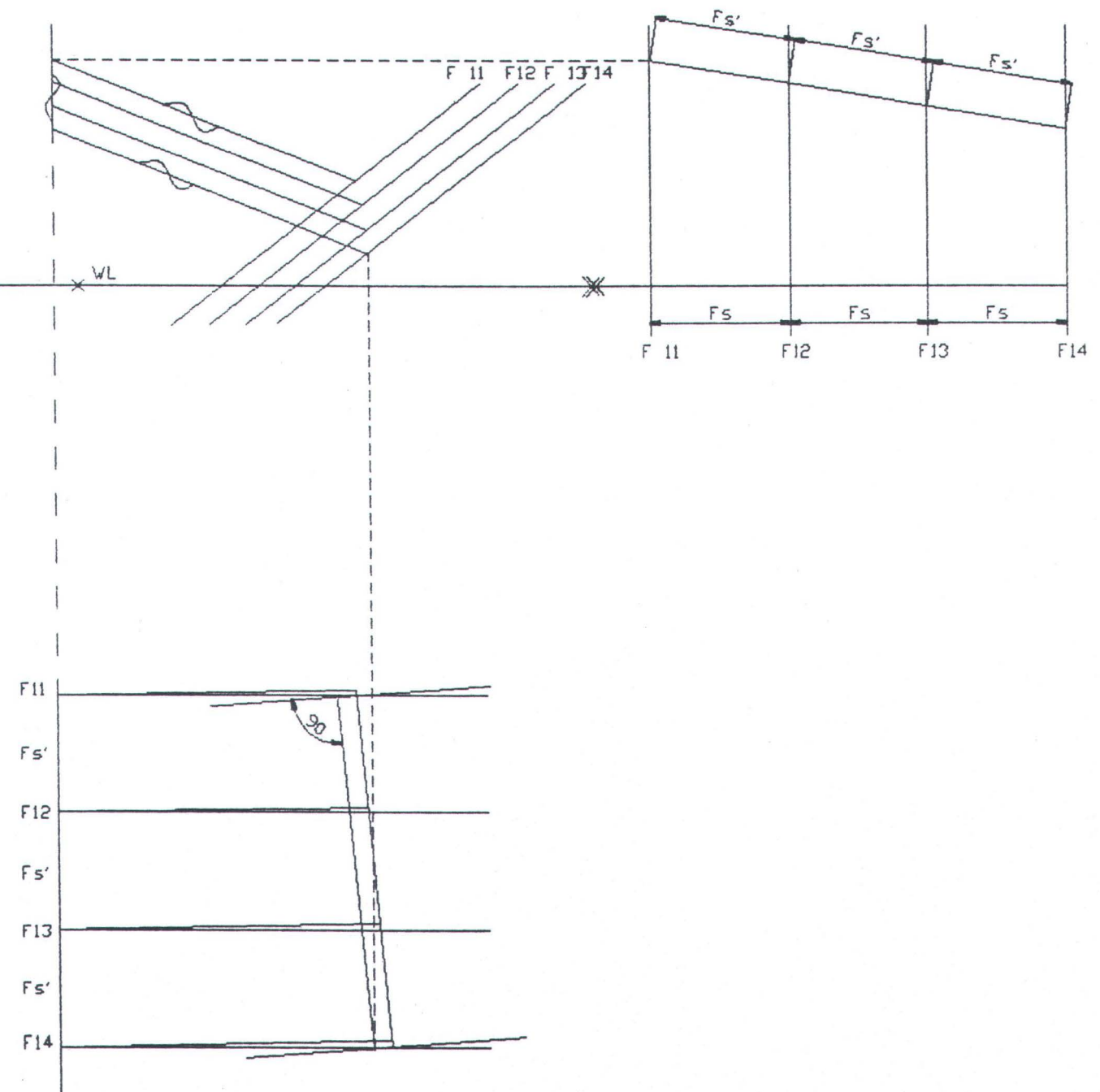
b). Bukan Dasar II



c). Bukaan Dasar III



d). Bukaan Dasar IV



LAMPIRAN B

LAMPIRAN B.1.1

Tipe Data Dalam AutoLISP

AutoLISP mendukung tipe data sebagai berikut

- Symbols
- Lists
- Strings
- Integers
- Real numbers
- File descriptors
- AutoCAD entity names
- AutoCAD selection sets

Symbols (Variabel)

AutoLISP menggunakan simbol (variabel) untuk diisi harga. Berikut ini adalah contoh fungsi 'setq' untuk memberi harga pada simbol pt1 dengan harga titik (1,2).

```
(setq pt1 '(1 2))
```

Lists

AutoCAD menggunakan list untuk menyimpan beberapa angka sekaligus yang memiliki hubungan antara satu dengan lainnya dalam satu variabel. Hal ini sangat efisien dalam pemrograman. Tipe data list ini banyak digunakan dalam memanipulasi nilai koordinat, baik koordinat dua dimensi (2D) maupun dalam koordinat tiga dimensi (3D).

2D Points adalah ekspresi list dari dua bilangan real seperti (3.4 7.52)

3D Points adalah ekspresi list dari tiga bilangan real seperti (3.4 7.52 1.0)

Strings

Tipe data string bisa memiliki panjang yang bervariasi dan memori yang digunakan dialokasi secara dinamik. Namun demikian, string konstan dibatasi sampai 132 karakter. String dengan panjang yang tak terbatas dapat dibuat dengan fungsi 'strcat' yang berguna untuk menggabungkan beberapa string sekaligus.

Integers

Bilangan integer adalah bilangan tanpa memiliki titik desimal atau sering disebut bilangan bulat. Besarnya untuk sistem 16 bit adalah antara -32768 s/d +32767.

Reals

Bilangan real adalah bilangan dengan titik desimal. Terdiri dari bilangan negatif dan positif.

File Discriptors

File discriptors adalah huruf yang digunakan untuk mewakili pembukaan file oleh AutoLISP. Jika AutoLISP melakukan proses pembacaan dan penulisan maka nama tersebut harus direferensikan. Sebagai contoh untuk membaca file 'data.dat' digunakan file discriptors 'fil' maka penulisannya :

```
(Setq Fil ( open "data.dat" "r"))
```

Entity Names

Nama entity adalah label numerik yang dipakai untuk mewakili entity gambar. Pada hakekatnya dia adalah pointer dalam file yang mengarah pada gambar AutoCAD. Dengan demikian AutoLISP bisa mendapatkan dan merubah record database entity dan vektor garis (dalam layar). Sebagai contoh untuk mendapatkan nama entity gambar yang terakhir kita gunakan fungsi 'entlast' dan kita simpan ke simbol e1: (setq e1 (entlast)) hasilnya: <entity name : 600000000016>

Selection Sets

Selection sets adalah kumpulan dari satu atau beberapa entity. Kita dapat menambah atau membuang obyek dari selection set. Berikut ini adalah contoh fungsi AutoLISP untuk mendapatkan sekelompok obyek yang terakhir terpilih :

```
(setq ss (ssget "P"))
```

hasilnya: <selection set : 1>

Fungsi-Fungsi AutoLISP

(angle pt1 pt2)

Fungsi ini menghasilkan suatu sudut dari suatu garis lurus yang ditarik dari titik pt1 ke titik pt2. Sudut diukur dari sumbu X dalam satuan radian berlawanan dengan arah jarum jam. Jika terdapat 3 titik yang membentuk 3D, maka titik-titik tersebut akan diproyeksikan dalam 2D atau plane.

Contoh :

```
(angle '(1.0 1.0) '(1.0 4.0))
```

hasil 1.5708

(append expr)

Fungsi ini mengambil beberapa bilangan / huruf dari suatu daftar kemudian mengurutkannya dalam suatu daftar.

Contoh :

```
(append '(a b) '(c d)) hasil (a b c d)
```

(apply function list)

Apply membuat suatu fungsi didefinisikan oleh fungsi lain dengan data-data dalam bentuk daftar.

Contoh :

```
(apply '+ '(1 2 3)) hasil 6
```

```
(apply 'strcat ('("a" "b" "c"))) hasil "a b c"
```

(assoc item list)

Fungsi ini mensortir suatu kumpulan data 'alist' untuk 'item' sebagai suatu elemen penunjuk dan hasilnya merupakan data masukan 'alist'. Dan jika tidak ditemukan 'item' sebagai suatu petunjuk 'alist' maka assoc akan nil.

Contoh : (asumsikan bahwa data 'a1' didefinisikan sebagai berikut :

```
(( name box) (width 3) (size 4.7263) (depth 5) )
```

kemudian

```
(assoc 'size a1) hasil (size 4.7263)
```

```
(assoc 'weigth a1) hasil nil
```

(atof string)

Fungsi ini merubah data string ke bilangan real.

Contoh :

(atof "97.1") hasil 97.1

(atoi string)

Fungsi ini merubah data string ke bilangan bulat.

Contoh :

(atoi "3.9") hasil 3

(caar list), (cadr list), (caddr list), (cadar list), etc.

AutoLISP menyediakan rangkaian 'car' dan 'cdr', sampai dengan empat rangkaian.

Contoh :

(setq x '((a b) c d))

kemudian

(caar x) ekuivalen dengan (car (car x)) hasil a

(cdar x) ekuivalen dengan (cdr (car x)) hasil (b)

(cadar x) ekuivalen dengan (car (cdr (car x))) hasil b

(cadr x) ekuivalen dengan (car (cdr x)) hasil c

(caddr x) ekuivalen dengan (cdr (cdr x)) hasil (d)

(caddr x) ekuivalen dengan (car (cdr (cdr x))) hasil d

Dalam AutoLISP, cadr seringkali digunakan untuk mendapatkan koordinat Y dari titik 2D atau 3D, car digunakan untuk mendapatkan koordinat X. Demikian juga caddr dapat digunakan untuk mendapatkan koordinat Z dari titik 3D.

(close file-desc)

Fungsi ini menutup suatu file dan menghasilkan nil. 'File-desc' adalah suatu file deskripsi yang didapatkan dari fungsi 'open'. Setelah ditutup (close) 'file-desc' tidak dapat diubah lagi.

Contoh : asumsikan X adalah sebuah file deskripsi yang benar.

(close x)

menutup suatu file dan menghasilkan nil.

(command (args).....)

Fungsi ini menjalankan perintah-perintah dalam AutoCAD dari AutoLISP dan selalu menghasilkan nil. Argumen ini mewakili perintah-perintah AutoCAD dan sub-sub perintahnya. Fungsi ini mengevaluasi masing-masing argumen dan mengirimkannya ke AutoCAD berupa respon-respon dalam tanda petik. Ia menyampaikan nama-nama perintah dan pilihan-pilihannya sebagai string, dalam titik-titik 2D maupun 3D. AutoCAD hanya mengenal nama-nama perintah ketika ia mengeluarkan Command : dalam tanda petik.

(cond (<test1> <result> ...) ...)

Fungsi ini digunakan untuk menyeleksi beberapa argumen. Jika suatu fungsi pernyataan sesuai maka proses akan dilanjutkan.

(cons <New fist element>)

Ini adalah dasar dari list CONStruktur. Dia akan meletakkan (<new fist element>) dan <list>, dan hasilnya ditambahkan diawal elemen.

Contoh :

(cons 'a '(b c d)) hasil (a b c d)

(defun <sym> <argument list> <expr>...)

Fungsi ini digunakan untuk mendefinisikan fungsi dengan nama <sym> yang didalamnya terdiri dari argumen atau ekspresi.

Contoh :

(distance <pt1> <pt2>)

Fungsi ini akan menghasilkan jarak antara 2 titik dari titik <pt1> ke titik <pt2>.

Contoh : (distance '(1.0 2.5 3.0) '(7.7 2.5 3.0)) hasil 6.7

(findfile <filename>)

Fungsi yang digunakan untuk melacak file di disk pada direktori tertentu.

Contoh : (findfile "abc.lsp") jika ada menghasilkan /acad/abc.lsp

(fix <number>)

Fungsi ini akan menghasilkan konversi <number> ke bilangan integernya baik dari bilangan integer maupun bilangan real, dengan membuang bilangan pecahannya.

Contoh : (fix 3.7) hasil 3

(float <number>)

Fungsi ini akan menghasilkan konversi <number> ke bilangan real, baik dari bilangan integer ataupun real.

Contoh : (float 3.75) hasil 3.75

(getangle [<pt>] [<prompt>])

Fungsi ini akan meminta masukan sudut pada user <prompt> sebagai opsional.

Contoh : (setq ang (getangle '(1.0 3.5)))

(getcorner <pt> [<prompt>])

Fungsi ini akan menghasilkan titik seperti getangle, akan tetapi getcorner mensyaratkan <pt> atau titik dasar sebagai acuan.

Contoh : (setq pt1 (getcorner '(1.0 1.0) "Ke sudut : "))

(getdist [<pt>] [<prompt>])

Fungsi ini akan meminta besarnya jarak atau jarak dari titik tertentu.

Contoh : (setq dist (getdist '(1.0 3.5) "berapa jaraknya : "))

(getint [<prompt>])

Fungsi ini akan meminta masukan integer. Harganya berkisar antara -32768 sampai dengan +32767. <prompt> adalah opsional untuk ditampilkan ke layar.

Contoh : (setq num (getint "Masukan sebuah bilangan : "))

(getpoint [<pt>] [<prompt>])

Fungsi ini akan meminta masukan titik.

Contoh : (setq pt (getpoint "Titik awal : "))

(getreal [<prompt>])

Fungsi ini akan meminta masukan bilangan real. <prompt> adalah opsional untuk ditampilkan di layar.

Contoh : (setq val (getreal "Masukkan faktor skala : "))

(getvar <varname>)

Fungsi ini digunakan untuk mengambil harga dari sistem variabel AutoCAD.

Contoh : (getvar "CMDECHO")

(if <testexpr> <thenexpr> [<elseexpr>])

Fungsi ini untuk mengevaluasi suatu kondisi. Jika <testexpr> tidak nol, maka akan mengevaluasi <thenexpr>, jika lain daripada itu maka akan mengevaluasi <elseexpr>. Ekspresi <elseexpr> yang terakhir itu merupakan suatu pilihan. IF akan menghasilkan nilai dari ekspresi yang dipilih; jika <elseexpr> kosong dan <testexpr> nol maka IF akan menghasilkan nol juga.

Contoh : (if (- 1 3) "YES ! ! " "no. ")
hasilnya "no. "

(inters <pt1> <pt2> <pt3> <pt4> [<onseq>])

Fungsi ini untuk menguji dua garis dan akan menghasilkan titik dimana titik-titik itu saling berpotongan atau nol jika titik-titik itu tidak saling berpotongan. <pt1> dan <pt2> berada pada titik terakhir baris pertama, sedangkan <pt3> dan <pt4> berada pada titik terakhir baris kedua. Semua titik-titik itu terdapat dalam batas UCS yang aktif. Jika sistem variabel FLATLAND nol dan ke-empat titik itu adalah 3D, INTERS check untuk 3D yang saling berpotongan; jika lain daripada itu maka INTER akan berada pada konstruksi yang telah direncanakan dan hanya menguji untuk 2D yang saling berpotongan.

(itoa <int>)

Fungsi ini akan menghasilkan konversi dari suatu integer kedalam suatu string.

Contoh : (itoa 33) hasilnya " 33 "

(last <list>)

Fungsi ini menghasilkan elemen terakhir pada <list> dan <list> tidak boleh nol.

Contoh : (last '(a b c d e)) hasilnya e

(length <list>)

Fungsi ini akan menghasilkan suatu jumlah elemen dalam <list>.

Contoh : (length '(a b c d)) hasilnya 4

(list <expr> ...)

Fungsi ini meletakkan beberapa angka dalam suatu ekspresi <expr> dan stringnya dalam suatu tempat.

Contoh : (list 'a 'b 'c) hasilnya (a b c)

(member <expr> <list>)

Fungsi ini akan meneliti <list> untuk suatu kejadian dari <expr> dan akan menghasilkan sisa dari <list> saat mulai dengan kejadian pertama <expr>. Jika tidak ada suatu kejadian <expr> dalam <list>, MEMBER akan menghasilkan nil.

Contoh : (member 'c '(a b c d e)) hasilnya (c d e), (member 'q '(a b c d e)) hasilnya nil

(nth <n> <list>)

Fungsi ini akan menghasilkan elemen "nth" dari <list>, dimana <n> adalah angka dari elemen untuk menghasilkan (angka nol yang merupakan elemen pertama). Jika <n> lebih besar dari <list> elemen angka yang paling tinggi, maka akan menghasilkan nil.

Contoh : (nth 3 '(a b c d e)) hasilnya d

(open <filename> <mode>)

Fungsi ini akan membuka suatu file untuk diakses dengan fungsi I/O AutoLISP. Fungsi ini akan menghasilkan suatu file diskriptor untuk digunakan dengan fungsi I/O yang lain; untuk itu maka harus di-SETQ dengan suatu simbol

Contoh : (setq a (open "file.ext" "r"))

(polar <pt> <angle> <distance>)

Fungsi ini digunakan untuk menentukan koordinat suatu titik, jika titik tersebut koordinat polar yang merujuk pada suatu titik tertentu diketahui harganya.

Contoh : (polar '(1.0 1.0 3.5) 0.785398 1.414212) hasilnya (2.0 2.0 3.5)

(prin1 <expr> [<file-desc>])

Fungsi ini akan mencetak ekspresi <expr> pada monitor dan akan menghasilkan <expr>, <expr> bisa pada beberapa ekspresi, fungsi itu tidak membutuhkan suatu string. Jika <file-desc> ditampilkan (dan file deskriptor untuk suatu file yang terbuka), <expr> ditulis secara pasti seperti yang tampak pada layar monitor.

Contoh : (setq a 5 b (list 1 2 3 4 5) c "Hallo")
(prin1 a) hasilnya 55

(princ <expr> [<file-desc>])

Fungsi ini sama dengan fungsi PRIN1 kecuali yang mengontrol karakter-karakter dalam <expr> yang dicetak tanpa perluasan. Pada umumnya PRIN1 ditandai untuk mencetak ekspresi dalam suatu cara yang cocok dengan LOAD, sementara PRINC akan mencetaknya dalam suatu cara yang dapat dibaca dengan fungsi sebagaimana READ-LINE.

(print <expr> [<file-desc>])

Fungsi ini sama seperti fungsi PRIN1, kecuali suatu baris baru yang dicetak sebelum <expr>, dan spasi yang dicetak berikut <expr>.

(progn <expr> ...)

Fungsi ini akan mengevaluasi setiap <expr> secara teratur dan akan menghasilkan nilai dari ekspresi yang terakhir. Anda dapat menggunakan PROGN untuk mengevaluasi beberapa ekspresi dimana hanya ada satu ekspresi yang diinginkan.

(prompt <msg>)

Fungsi ini akan memperlihatkan <msg> pada daerah prompt pada layar monitor, dan akan menghasilkan nil. <msg> adalah suatu string.

Contoh : (prompt "New value") tampil "New value" hasil nil

(read <string>)

Fungsi ini akan menghasilkan list pertama atau atom yang diperoleh dari <string>. <string> tidak boleh kosong.

Contoh : (read "hello") hasilnya *hello*

(read-line [<file-desc>])

Fungsi ini akan membaca suatu string dari keyboard atau dari file aktif yang dideskripsikan dengan <file-desc>. Jika akhir dari file itu tidak dapat dihitung, maka READ-LINE akan menghasilkan nil, sebaliknya akan menghasilkan string yang telah dibaca.

Contoh : (read-line f), dimana f merupakan suatu file terbuka yang sah

(reverse <list>)

Fungsi ini akan menghasilkan <list> dengan posisi yang terbalik.

Contoh : (reverse ' (a) b c) hasilnya (c b a)

(rtos <number> [<mode> [<precision>]])

Argumen-argumen <mode> dan <precision> merespon variabel system AutoCAD LUNITS and LUPREC. Jika anda menghilangkan argumen-argumen tersebut, setting yang aktif dari LUNITS dan LUPREC akan digunakan.

(setq <sym> <expr1> [<sym2> <expr2>] ..)

Fungsi ini akan meletakkan nilai dari <sym1> ke <expr1>, <sym2> ke <expr2> dan seterusnya. Fungsi ini berdasarkan fungsi tanda dalam AutoLISP. Fungsi ini akan menghasilkan <expr> yang terakhir.

Contoh : (setq a 5.0) hasilnya 5.0

(setvar <varname> <value>)

Fungsi ini akan meletakkan suatu sistem variabel AutoCAD terhadap <value> yang diberikan, dan akan menghasilkan nilai tersebut. Variabel nama itu harus disertakan dalam dua bagian.

Contoh : (setvar "FILLETRAD" 0.50) hasilnya 0.50

(strcase <string> [<which>])

Fungsi ini digunakan untuk membuat agar string yang berada pada argumen <string> dibuat menjadi huruf besar semua atau huruf kecil semua.

(strcat <string1> <string2> ...)

Fungsi ini digunakan untuk menggabungkan dua atau lebih string menjadi satu kesatuan.

Contoh : (strcat "a" "b" "c") hasilnya "abc"

(substs <newitem> <olditem> <list>)

Fungsi ini digunakan untuk mengganti elemen dari suatu list.

(vports)

Fungsi ini akan memberikan vport yang diaktifkan.

(write-line [<string> [file-desc]])

Fungsi ini digunakan untuk menulis sebaris <string> ke dalam file yang dibuka dengan fungsi OPEN. Adapun file yang dibuka tersebut

harus merupakan file yang digunakan untuk dijadikan keluaran. Fungsi ini juga dapat digunakan untuk menulis sebaris data pada layar.

(* error * <string>)

Fungsi ini merupakan fungsi untuk mengatasi kesalahan yang siap pakai. Jika tidak nil maka program akan diproses selama fungsi tersebut dipanggil atau ada.

Entity dan Devices Access

Pengambilan Nama Entity

AutoLISP memiliki empat fungsi untuk mengambil nama entity secara langsung dari database AutoCAD. Fungsi tersebut adalah :

entlast

Menunjukkan nama entity akhir yang belum dihapus dalam database.

(entlast)

entnext

Menunjukkan nama entity pertama yang belum dihapus dalam database. Jika ename yang dipilih tersedia, nama entity yang belum dihapus selanjutnya segera mengikuti ename tersebut yang telah ditunjukkan.

(entnext ename)

entsel

Menunjukkan sebuah list yang berisi nama entity dan koordinat titik digunakan untuk memilih / mengambil entity. Prompt string yang terpilih dapat menyediakan instruksi khusus untuk pemilihan entity.

(entsel prompt)

Perintah untuk menghapus entity

entdel

Penghapusan atau perbaikan ename tergantung pada statusnya dalam tahap edit yang dijalankan.

(entdel ename)

Kumpulan Pilihan Entity

ssget

Menunjukkan sebuah kumpulan pilihan dari entity. Jika SSGET gagal, maka akan menunjukkan nil. Dengan tanpa argumen(atau sebuah nil), SSGET menggunakan standart AutoCAD pemilihan object untuk mengambil sebuah pilihan dari pengguna. Mode argumen yang terpilih merupakan sebuah string yang mengkhususkan sebuah pilihan untuk mengotomatisasikan pilihan tanpa input dari pengguna. Mode-mode tersebut adalah P untuk Previous, L untuk Last, I untuk implikasi pickfirst sambil menunggu pengaturan pilihan, W untuk Window, C untuk Crossing, WP untuk Wpoly, CP untuk Cpoly, X untuk semua entity, dan F untuk Fence.

(ssget mode point1 point2 pointlist filterlist)

Memanipulasi Kumpulan Pilihan

sslenght

Menunjukkan jumlah dari entity dalam selection-set

(*sslenght selection-set*)

ssmemb

Menunjukkan nama entity **ename** jika **ename** termasuk dalam selection-set. Jika tidak, maka akan menunjukkan nil.

(*ssmemb ename selection-set*)

ssname

Menunjukkan nama entity dari selection-set yang menunjukkan posisi dari argumen angka. Entity pertama adalah angka 0.

(*ssname selection-set number*)

ssadd

Menambahkan nama entity ke dalam kumpulan entity yang terpilih.

(*ssadd ename selection-set*)

ssdel

Menghapus ename dari selection-set.

(*ssdel ename selection-set*)

LAMPIRAN B.1.2

Visual Basic Control

Pointer

Digunakan untuk menyediakan sebuah cara untuk memindah dan mengubah ukuran form dan kontrol.

PictureBox

Digunakan untuk menampilkan bitmaps, icons, atau Windows Metafile. PictureBox menampilkan teks atau aksi sebagai sebuah visual container untuk kontrol lainnya.

Label

Digunakan untuk menampilkan teks seorang user yang tidak dapat berpengaruh dengan atau berubah.

TextBox

Digunakan untuk menyediakan sebuah area untuk enter atau menampilkan teks.

Frame

Digunakan untuk menyediakan sebuah kotak visual dan fungsional untuk kontrol.

CommandButton

Digunakan untuk mengeluarkan sebuah perintah (command) atau aksi ketika seorang user memilikinya.

CheckBox

Digunakan untuk menampilkan sebuah pilihan True/False atau Yes/No. Anda dapat memeriksa beberapa angka dari checkbox pada sebuah form pada sekali waktu.

OptionButton

Merupakan bagian dari sebuah grup pilihan dengan optionbutton lainnya, menampilkan banyak pilihan, dari yang mana seorang user dapat memilih hanya satu.

ComboBox

Digunakan untuk mengkombinasikan sebuah textbox dengan sebuah listbox. Membolehkan seorang user untuk mengetik dalam sebuah pilihan atau memilih sebuah item dari drop-down list.

ListBox

Digunakan untuk menampilkan sebuah list dari item-item yang mana seorang user dapat memilih dari.

HScrollBar dan VScrollBar

Horizontal scroll bar dan vertikal scroll bar mengijinkan seorang user untuk memilih sebuah nilai dalam sebuah range nilai.

Timer

Digunakan untuk mengeksekusi timer events pada interval waktu khusus.

DriveListBox

Digunakan untuk menampilkan dan mengijinkan seorang user untuk memilih disk drive yang valid.

DirListBox

Kontrol Directory list box menampilkan dan mengijinkan seorang user untuk memilih directory dan paths.

FileListBox

Digunakan untuk menampilkan dan mengijinkan seorang user untuk memilih dari sebuah list dari file.

Image

Digunakan untuk menampilkan bitmaps, icons, atau Windows Metafile; beraksi seperti sebuah tombol perintah (commandbutton) ketika diklik.

Data

Membolehkan anda untuk berhubungan dengan sebuah keluaran database dan menampilkan informasi darinya pada form anda.

OLE

Digunakan untuk menyimpan data kedalam sebuah aplikasi Visual Basic.

DBCombo

Menyediakan ciri-ciri dari standart combo box control, ditambah kapabilitas dari akses data yang dinaikkan.

DBList

Menyediakan ciri-ciri dari standart list box control, ditambah kapabilitas dari akses data yang dinaikkan.

Menu

Digunakan untuk menciptakan menu-menu dalam aplikasi Visual Basic anda.

Kategori dari Kontrol

Ada tiga kategori dari kontrol dalam Visual Basic :

- *Standart controls*, seperti commandbutton dan frame controls. Kontrol-kontrol ini dimuat disamping file .EXE Visual Basic. Standart control selalu termasuk dalam ToolBox, tidak seperti custom control dan insertable objects, yang mana dapat

dipindah dari atau ditambahkan ke ToolBox.

- *Custom controls*, yang mana keluar sebagai file yang terpisah dengan filename extension .VBX atau .OCX. Mereka termasuk kontrol-kontrol khusus, seperti common dialog control dan third-party custom controls, sebaik versi tambahan dari standart controls, seperti data-bound list box dan data-bound combo box controls.
- *Insertable Objects*, sebagaimana sebuah obyek Microsoft Excel Worksheet yang berisi sebuah list dari semua pekerja perusahaan anda, atau sebuah obyek Microsoft Project Calendar yang berisi informasi schedule untuk sebuah proyek. Sejak hal-hal tersebut dapat ditambahkan ke toolbox, mereka dapat dipertimbangkan custom controls. Beberapa dari obyek-obyek ini juga mendukung Otomatisasi OLE, yang mana mengijinkan anda untuk memprogram obyek aplikasi lainnya dari dalam aplikasi Visual Basic.

LAMPIRAN B.2.1

Bentuk sub-routine program penggambaran

body plan :

```
(defun bplan ()
  (setq file (open "c:/skripsi/program/data1.txt"
    "r"))
  tg (atof (read-line file))
  lb (atof (read-line file))
  rb (atof (read-line file))
  rf (atof (read-line file))
  cm (atof (read-line file))
  gg (atof (read-line file))
  ee (atof (read-line file))
  ef (atof (read-line file))
  nmg(atof (read-line file))
  nmf(atof (read-line file))
  tanda (read-line file)
  tanda1 (read-line file)
  tanda2 (read-line file)
  tanda3 (read-line file)
  ket (read-line file)
  lbb (fix (/ lb 2))
  tk (list -4750 0)
  tk1(list (+ (car tk) lbb) (cadr tk))
  tk2(list (car tk1) (+ (cadr tk1)tg))
  tk3(list (car tk) (+ (cadr tk)tg))
  tkr(list (car tk) (+ (cadr tk)rf)))
  (command "erase" "all" ""
    "regen"
    "zoom" "e"
    "layer" "s" "0" ""
    "line" tkr tk1 tk2 tk3 "c")
  (setq pt (ssget "x" '(( 8 . "0"))))
  pt1 (ssname pt 0)
  pt2 (ssname pt 1)
```

```
pt3 (ssname pt 2)
pt4 (ssname pt 3))
(command "fillet" "r" rb "fillet" pt1 pt4)
(setq pt (ssadd (entlast) pt))
(command "change" pt3 "" "p" "c" "1" ""
  "change" pt3 "" "p" "lt" "center" ""
  "ltscale" 500)
(command "mirror" pt "" tk2 "@1<90" "")
(setq grc (polar tk2 (/ (* pi 90)180) cm))
(setq tk4 (polar tk2 (/ (* pi 0)180) lbb))
(command "arc" tk3 grc tk4)
(command "zoom" "all")
);end of bplan
Kemudian berikut ini bentuk sub-routine
penggambaran kurva gading-gading dan
sambungan pelat :
(defun frame ()
  (command "layer" "m" "frame" "c" "4" "" "")
  (setq pool (open "c:/skripsi/program/gab.txt"
    "r"))
  n (fix (atof (read-line pool)))
  nn (fix (atof (read-line pool)))
  b 0)
  (while (/= b nn)
    (command "pline")
    (setq x (atof (read-line pool))
      y (atof (read-line pool))
      xo(list x y))
    (command xo)
    (repeat (- n 1) (progn
      (setq x (atof (read-line pool))
        y (atof (read-line pool))
        xo (list x y))(command xo)))
    (prompt "-") (princ)
    (command "")
    (setq b (+ b 1))
  );end of repeat
  (command "zoom" "all")
);end of frame

(defun weld ()
  (command "layer" "m" "weld" "c" "3" "" "")
  (setq fool(open "c:/skripsi/program/data5.txt"
    "r"))
  (setq m (fix (atof (read-line fool)))
    mm (fix (atof (read-line fool)))
    c 0)
  (while (/= c mm)
    (command "pline")
    (setq x (atof (read-line fool))
      y (atof (read-line fool))
      xo(list x y))
    (command xo)
    (repeat (- m 1) (progn
      (setq x (atof (read-line fool))
        y (atof (read-line fool))
        xo (list x y))(command xo)))
    (prompt "-") (princ)
    (command "")
  (setq c (+ c 1))
```

```
); end of repeat
(command "zoom" "all")
);end of weld
```

LAMPIRAN B.2.2

Bentuk program untuk proses pemulusan kurva (stream line) adalah sebagai berikut :

```
(defun fair ()
  (setq fr (ssget "x" '((8 . "frame"))))
    fr1(sslenght fr)
    fr2 0)
  (while (/= fr2 fr1)
    (setqfr3 sname fr fr2))
  (command "pedit" fr3 "f" "")
  (setq fr2 (+ fr2 1))
  (prompt "-") (princ)
  ); while
);end of fair
```

```
(defun fair1 ()
  (setq fr (ssget "x" '((8 . "weld"))))
    fr1(sslenght fr)
    fr2 0)
  (while (/= fr2 fr1)
    (setq fr3 (sname fr fr2))
    (command "pedit" fr3 "f" ""))
    (setq fr2 (+ fr2 1))
  (prompt "-") (princ)
  ); while
);end of fair1
```

LAMPIRAN B 2.3

Bentuk program PICK adalah sebagai berikut :

```
(defun pick (/ a b c)
  (setq c ())
  (if (/= salah 1)
    (progn (setq a (cadr (grread)))
      (while (/= a 0)
        (setq b (cadr (grread)))
        (if (/= b 0)
          (grdraw a b 7 1))
          (setq a b)
          c (cons a c))
        ));while
    (progn (setq a1 (cadr (grread)))
      (while (/= a1 0)
        (setq b1 (cadr (grread)))
        (if (/= b1 0)
          (grdraw a1 b1 7 1))
          (setq a1 b1)
          c (cons a1 c))
        ));while
    )
  (GRCLEAR)
  (setq c (cdr c)
    am1 (ssget "cp" c))
);end of select
```

GRREAD digunakan untuk membaca macam alat masukan yang diterima oleh AutoCAD beserta datanya. *GRDRAW* digunakan untuk menggambar vektor dari suatu titik ke titik lain dengan menggunakan warna garis sesuai yang diinginkan. *GRCLEAR* digunakan untuk pindah ke mode grafik dan semua gambar yang ada tidak dimunculkan pada layar editor AutoCAD.

LAMPIRAN B.2.4

Program untuk pengambilan daerah pelat yang dipilih adalah sebagai berikut :

```
(defun ambil ()
  (ok4)
  (command "regen")
  (setq am (ssget "X")
    am2 (sslenght am1)
    am3 0
    am4 ()))
  (while (/= am2 am3)
    (setq am5 (sname am1 am3)
      am6 (ssmemb am5 am))
      (if (/= am6 nil)
        (setq am7 (ssdel am6 am))
        )
    (setq am3 (+ am3 1))
  )
  (command "erase" am7 "")
  (setq am (ssget "X")
    am9 (sslenght am)
    am10 0
    am12 ()
    am20 ()))
  (while (/= am10 am9)
    (setq am13 (sname am am10)
      am14 (entget am13)
      am10 (+ am10 1)
      am18 (cdr (assoc 8 am14)))
    (if (= am18 "WELD")
      (progn (setq am19 am13
        am12 (cons am19
          am12))))
    )
    (if (= am18 "FRAME")
      (progn (setq am17 am13
        am20 (cons am17
          am20))))
    )
  );while
);end of ambil

Sedang program untuk menentukan titik awal dan akhir dari masing-masing kurva yang terpilih adalah sebagai berikut :
```

```
(defun titik ()
  (command "zoom" "e")
  (setq tt1 (length am20)
    tt2 0
    tt3 0
    tt4 ()
    tt5b ()))
```

```

(while (/= tt2 tt1)
  (setq tt5a (nth tt2 am20)
        tt6 (entget tt5a))
  (IF (/= tt5a NIL)
    (progn(while (/= (cdr (assoc
0 (entget tt5a))) "SEQEND")
  (setq b (entnext tt5a)
        b1 (cdr (assoc 10 (entget b)))
        tt4 (cons b1 tt4)
        tt5a b)
  );while
  (setq tt4 (cdr tt4))
    );progn
    );if
  (setq tt2 (+ tt2 1)
        tt5b (cons tt4 tt5b)
        tt4 ())
  );while
  (setq tt11 ())
  (while (/= tt3 tt1)
    (setq tt7 (nth tt3 tt5b)
          tt8 (last tt7)
          tt9 (car tt7)
          tt10 (list tt9 tt8)
          tt11 (cons tt10 tt11)
          tt3 (+ tt3 1))
    );while
  );end of titik

```

- tt11b merupakan penyimpanan titik awal dan akhir kurva *

LAMPIRAN B.2.5

Bentuk program pemotongan pelat adalah sebagai berikut :

```

(defun potong ()
  (setq po 0
        po1 (length tt11))
  (command "trim" (car am12) (cadr am12) "")
  (while (/= po po1)
    (setq po2 (nth po tt11))
    (command (cadr po2))
    (command (car po2))
    (setq po (+ po 1)))
  );while
  (command "")
  );end of potong

```

LAMPIRAN B.2.6

Program penentuan titik-titik tersebut dapat dijelaskan sebagai berikut :

```

(defun tprok ()
  (setq tp (ssget "x" ((8 . "frame"))))
  tp1 (sslenght tp)
  tp2 0
  tpa ())
  (while (/= tp1 tp2)
    (setq tp3 (ssname tp tp2)
          tpa (cons tp3 tpa)

```

```

tp2 (+ tp2 1))
)
(setq tk1 (length tpa)
      tk2 0
      tk3 0
      tk4 ()
      tk5b ())
)
(while (/= tk2 tk1)
  (setq tk5a (nth tk2 tpa)
        tk6 (entget tk5a))
  (IF (/= tk5a NIL)
    (progn(while (/= (cdr (assoc 0 (entget
tk5a))) "SEQEND")
  (setq c (entnext tk5a)
        c1 (cdr (assoc 10 (entget c)))
        tk4 (cons c1 tk4)
        tk5a c)
  );while
  (setq tk4 (cdr tk4))
    );progn
    );if
  (setq tk2 (+ tk2 1)
        tk5b (cons tk4 tk5b)
        tk4 ())
  );while
  (setq tk11 ())
  (while (/= tk3 tk1)
    (setq tk7 (nth tk3 tk5b)
          tk8 (last tk7)
          tk9 (car tk7)
          tk10(list tk9 tk8)
          tk11(cons tk10 tk11)
          tk3 (+ tk3 1))
    );while
  );end of tprok

```

- tk11 merupakan record dari titik awal dan akhir yang telah ditemukan *

LAMPIRAN B.2.7

Program penentuan jarak garis proyeksi:

```

(defun fspace1 ()
  (command "layer" "m" "line" "c" "95" "" "")
  (setq fs (nth (/ tk1 2) tk11)
        fs1 (car fs)
        fs2 (cadr fs)
        fs3 (list (car fs1) (- (cadr fs1) 3000))
        fs4 (list (- (car fs3) 4000) (cadr fs3))
        fs5 (list (+ (car fs3) 4000) (cadr fs3)))
  (command "line" fs4 fs5 "")
  (setq ff (- tk1 2))
  (command "array" (entlast) "" "r" "2" "1" ee)
  (command "array" (entlast) "" "r" ff "1" gg)
  (command "array" (entlast) "" "r" "2" "1" ef)
  (command "zoom" "all" "")
  );fspace

```

Pendefinisian titik fs4 dan fs5 merupakan titik-titik yang digunakan untuk membuat garis gading sebenarnya. Variabel ee merupakan variabel untuk memasukkan jarak garis las dari gading, variabel gg merupakan jarak gading sebenarnya, dan variabel ef merupakan jarak garis las dari gading. Sedangkan program untuk membuat garis proyeksi adalah sebagai berikut :

```
(defun proyeksi ()
  (command "layer" "m" "bantu" "c" "2" "" "")
  (setq nn (length tk11)
        n 0
        b nn)
  (while (/= n b)
    (command "line")
    (setq d (car (nth n tk11))
          d1 (list (car d) (- (cadr d) 10000)))
    (command d d1)
    (command "")
    (setq n (+ n 1))
  )
  (command "layer" "m" "bantu" "c" "1" "" "")
  (setq mm (length tk11)
        m 0
        c mm)
  (while (/= m c)
    (command "line")
    (setq e (cadr (nth m tk11))
          e1 (list (car e) (- (cadr e) 10000)))
    (command e e1)
    (command "")
    (setq m (+ m 1))
  )
  );end of proyeksi
```

LAMPIRAN B.2.8

Bentuk pemrograman dari penentuan titik awal dan akhir garis adalah sebagai berikut:

```
(defun epoint ()
  (setq ll (ssget "x" '((8 . "line"))
                ll1 (sslenght ll)
                ll2 0
                lla ()
                llb ()
                llc ()
  )
  (while (/= ll1 ll2)
    (setq ll3 (ssname ll ll2)
          ll4 (entget ll3)
          ll5 (length ll4)
          ll6 0
          ll2 (+ ll2 1))
    (while (/= ll5 ll6)
      (setq ll7 (nth ll6 ll4)
            ll8 (car ll7)
            ll6 (+ ll6 1))
      (if (= ll8 10)
```

```
(progn (setq ll9 (cdr ll7)
          lla (cons ll9 lla)))
    (if (= ll8 11)
      (progn (setq ll10(cdr ll7)
                  llb (cons ll10 llb))))
  )
  )
  )
  (setq ll11 (length lla)
        ll12 (length llb)
        ll13 0
        llc ()
  )
  (while (/= ll13 ll11)
    (setq ll15 (nth ll13 lla)
          ll16 (nth ll13 llb)
          ll17 (list ll15 ll16)
          llc (cons ll17 llc)
          llr (reverse llc)
          ll13 (+ ll13 1))
  )
  (setq bb (ssget "x" '((8 . "bantu")))
        bb1 (sslenght bb)
        bb2 0
        bba ()
        bbb ()
        bbc ()
  )
  (while (/= bb1 bb2)
    (setq bb3 (ssname bb bb2)
          bb4 (entget bb3)
          bb5 (length bb4)
          bb6 0
          bb2 (+ bb2 1))
    (while (/= bb5 bb6)
      (setq bb7 (nth bb6 bb4)
            bb8 (car bb7)
            bb6 (+ bb6 1))
      (if (= bb8 10)
        (progn (setq bb9 (cdr bb7)
                    bba (cons bb9 bba)))
        (if (= bb8 11)
          (progn (setq bb10(cdr bb7)
                    bbb (cons bb10 bbb))))
      )
    )
  )
  (setq bb11 (length bba)
        bb12 (length bbb)
        bb13 0
        bbc ()
  )
  (while (/= bb13 bb11)
    (setq bb15 (nth bb13 bba)
          bb16 (nth bb13 bbb)
          bb17 (list bb15 bb16)
          bbc (cons bb17 bbc)
          bb13 (+ bb13 1))
  )
  (setq cc (ssget "x" '((8 . "bantu1")))
```

```

cc1 (sslenght cc)
cc2 0
cca ()
ccb ()
ccc ()
)
(while (/= cc1 cc2)
  (setq cc3 (ssname cc cc2)
    cc4 (entget cc3)
    cc5 (length cc4)
    cc6 0
    cc2 (+ cc2 1))
  (while (/= cc5 cc6)
    (setq cc7 (nth cc6 cc4)
      cc8 (car cc7)
      cc6 (+ cc6 1))
    (if (= cc8 10)
      (progn (setq cc9 (cdr cc7)
        cca (cons cc9 cca)))
      (if (= cc8 11)
        (progn (setq cc10(cdr cc7)
          ccb(cons cc10 ccb)))))))
  (setq cc11 (length cca)
    cc12 (length ccb)
    cc13 0
    ccc ())
)
(while (/= cc13 cc11)
  (setq cc15 (nth cc13 cca)
    cc16 (nth cc13 ccb)
    cc17 (list cc15 cc16)
    ccc (cons cc17 ccc)
    cc13 (+ cc13 1))
)
);end of epoint

```

LAMPIRAN B.2.9

Bentuk pemrograman untuk menentukan perpotongan antara dua garis adalah sebagai berikut :

```

(defun point2a ()
  (setq ii (length ccc)
    jj (length llc)
    b 0
    tt1()
    ttt()
    ttr()
  )
  (while (/= b jj)
    (setq jj1 (nth b llc)
      ii1 (nth b ccc))
    (progn
      (setq tt1 (inters (car jj1)(cadr jj1)(car
        ii1)(cadrii1)))
      )
    (setq b (+ b 1)
      ttt(cons tt1 ttt)
      ttr(reverse ttt))
  );end of while
);end of point2a

```

ttr merupakan titik-titik hasil perpotongan antara garis proyeksi lower seam dengan garis gading sebenarnya. Sedangkan *ptr* merupakan titik-titik hasil perpotongan garis proyeksi upper seam dengan gading sebenarnya.

LAMPIRAN B.2.10

Bentuk program pembuatan garis tegak lurus adalah sebagai berikut :

```

(defun tegak ()
  (command "layer" "m" "tegak" "c" "120" "" "")
  (setq tg (length ptr)
    tg1 0
  )
  (while (/= tg1 (- tg 1))
    (setq tg2 (nth tg1 ptr)
      tg3 (nth (+ tg1 1) ptr))
    (progn
      (setq tg4 (angle tg3 tg2)
        tg5 (angtos tg4 0 4)
        tg6 (atof tg5))
      (setq tg7 (- tg6 90)
        tg8 (+ 180 tg7)
        tg9 (polar tg2 (/ (* pi tg8) 180)
          1500))
      );progn
    (command "line" tg2 tg9 "")
    (setq tg1 (+ tg1 1))
  );while
  (setq tr (nth (- tg 1) ptr)
    tf (nth (- tg 2) ptr))
  (command "copy" (entlast) "" tf tr)
);end of tegak

```

LAMPIRAN B.2.11

Bentuk program pembagian panjang kurva:

```

(defun bagi_frame ()
  (setq fb (ssget "x" '((8 . "frame"))))
    fb1 (sslenght fb)
    fb2 0
    fbb 250
    fb3()
    fb3a()
  (while (/= fb1 fb2)
    (setq fb4 (ssname fb fb2)
      fb2 (+ fb2 1))
    (command "divide" fb4 fbb)
    (setq fb5 (ssget "x" '((0 . "point"))))
      fb6 (sslenght fb5)
    fb7 0)
  (while (/= fb7 fb6)
    (setqfb8 (ssname fb5 fb7)
      fb9 (entget fb8)
      fb10 (length fb9)
      fb11 0
      fb7 (+ fb7 1))
  (while (/= fb11 fb10)

```



```
(setq fb12 (nth fb11 fb9)
  fb13 (car fb12)
  fb11 (+ fb11 1))
(if (= fb13 10)
  (progn(setq fb12 (cdr fb12)
    fb3 (cons fb12 fb3))))))
(setq fb3a (cons fb3 fb3a)
  fb3 ())
(command "erase" fb5 "")
(prompt "-")(princ)
);while
);end of fb3
```

fb3a merupakan semua panjang kurva frame yang ada pada layar dan telah dibagi menjadi 250 titik. Langkah selanjutnya adalah menentukan jarak antar titik dengan menggunakan fungsi *distance*.

Bentuk program untuk menentukan jarak antar titik pada kurva :

```
(defun dist ()
  (setq dt (length fb3a)
    dt1 0
    dt2 ()
  )
  (while (/= dt1 dt)
    (setq dt3 (nth dt1 fb3a)
      dt4 (length dt3)
      dt1 (+ dt1 1)
      dt5 0)
    (if (/= dt5 dt4)
      (progn
        (setq dt6 (nth dt5 dt3)
          dt7 (nth (+ dt5 1) dt3)
          dt8 (distance dt6 dt7)
          dt9 (* dt8 fbb)
          dt10(rtos dt9 2 4)
          dt11(atoi dt10)
          dt12(cons dt11 dt12)
          dt5 (+ dt5 1))))))
  );end of dist
```

dt12 merupakan panjang kurva sebenarnya dari gading (frame). Setelah panjang kurva masing-masing frame ditemukan maka langkah selanjutnya adalah menggoreskan panjang kurva tersebut ke garis tegak lurus yang telah dibuat. Dalam pemrograman ini digunakan perintah AutoCAD " *circle* " untuk menggoreskan panjang kurva frame ke garis tegak lurus. Alasan penggunaan perintah *circle* adalah karena *circle* (lingkaran) memiliki jari-jari yang sama pada semua sudut.

Bentuk program untuk menggambar panjang kurva frame sebenarnya :

```
(defun circle ()
  (command "layer" "m" "gores" "c" "211" "" "")
  (setq cr (length dt12)
    crz (length ttr)
    cr1 0
    crt ()
```

```
)
  (while (/= cr1 cr)
    (setq cr2 (nth cr1 ttr)
      cr3 (nth cr1 dt12))
    (command "circle" cr2 cr3 )
    (setq cr1 (+ cr1 1)
  );while
);end of circle
```

Titik pusat untuk penggambaran lingkaran diatas adalah *ttr* yang merupakan titik-titik hasil proyeksi lower seam pada garis gading sebenarnya. Langkah pemrograman selanjutnya adalah menentukan titik potong antara lingkaran dengan garis tegak lurus dimana titik potong tersebut merupakan titik-titik untuk penggambaran upper seam sebenarnya. Cara menentukan titik potong antara lingkaran dengan garis tegak lurus adalah dengan menggunakan pointer yang dibantu object snap " *appint* " (*appint* merupakan object snap dalam AutoCAD yang biasa digunakan untuk menentukan titik potong).

Bentuk program untuk menentukan titik potong lingkaran dengan garis tegak lurus :

```
(defun gabung ()
  (setq gb (length dt12)
    ga (length avc)
    gaa (reverse avc)
    gb1 0
  )
  (while (/= gb1 gb)
    (command "layer" "m" "gores" "c" "211" "" "")
    (progn (setq gb2 (nth gb1 ttr)
      gb3 (nth gb1 dt12))
      (command "circle" gb2 gb3)
      (ide)
    )
    (command "layer" "m" "tegak" "c" "120" "" "")
    (progn (setq ga1 (nth gb1 gaa)
      ga2 (car ga1)
      ga3 (cadr ga1))
      (command "line" ga2 ga3 "")
      (ide1)
    )
  )
  (setq fx (length avc)
    dx (length hx)
    kx (car avc)
    gx 0
    px ()
  )
  (while (/= gx fx)
    (setq lx (nth gx hx)
      mx (nth gx kx)
      gx (+ gx 1))
    (progn
      (command "point" "appint" lx mx)
      (setq nx (entlast)
        ox (cdr (assoc 10 (entget nx)))
        px (cons ox px))))))
```

```
(command "erase" (ssget "x" '((8 . "gores"))))
""
(command "erase" (ssget "x" '((8 . "tegak"))))
""
(setq pxx (cons px pxx)
  gb1 (+ gb1 1))
)
);end of gabung
pxx merupakan titik-titik potong lingkaran
dengan garis tegak lurus.
```

LAMPIRAN B.2.12

Bentuk program untuk menggambarkan batas sebenarnya dari upper seam:

```
(defun upper ()
(command "layer" "m" "weld1" "c" "3" "" "")
(setq upp pxx
  up (length upp)
  up1 0
  d up)
(command "pline")
(while (/= up1 d)
  (setq up2 (car (nth up1 upp))
    up1 (+ up1 1))
  (command up2))
(command "")
);end of upper
```

LAMPIRAN B.2.13

Bentuk program penggambaran panjang kurva sebenarnya:

```
(defun panjang ()
(command "layer" "m" "frame" "c" "4" "" "")
(setq pj (length pxx)
  pk (length ttr)
  pj1 0)
)
(while (/= pj1 pj)
  (progn
    (setq pj2 (nth pj1 ttr)
      pj3 (car (nth pj1 (reverse pxx)))
      pj1 (+ pj1 1))
    )
  (command "pline" pj2 pj3 ""))
)
);end of panjang
```

Bentuk program untuk menggambarkan batas lower seam:

```
(defun seam2 ()
(command "layer" "m" "weld1" "c" "3" "" "")
(setq rr (length ttr)
  r 0
  e rr)
(command "pline")
(while (/= r e)
  (command (nth r ttr))
  (setq r (+ r 1)))
(command "")
(command "erase" (ssget "x" '((8 . "bantu"))))
"")
```

```
(command "erase" (ssget "x" '((8 . "bantu1"))))
""
);end of seam2
```

LAMPIRAN B.2.14

Bentuk pemrograman tampilan datar hasil bukaan:

```
(defun sudut ()
(setq sd (car ttr)
  sf (last ttr)
  fd (angle sf sd)
  fd1 (angtos fd 0 2)
  fd2 (atof fd1)
  fdd (- 0 fd2))
(setq ssl (ssget "X"))
(command "rotate" ssl "" sf fdd)
);end of sudut
```

LAMPIRAN B.2.15

Berikut ini program pemberian simbol-simbol penandaan:

- Simbol ketebalan pelat

```
(defun tebal ()
(setq te (ssget "X" '((8 . "frame"))))
  te1 (sslenght te)
  te2 0
  te3 ())
(while (/= te2 te1)
  (setq te4 (ssname te te2)
    te2 (+ te2 1))
  (command "divide" te4 "2")
  (setq te5 (ssget "X" '((0 . "point")))
    te6 (ssname te5 0)
    te7 (entget te6)
    te7a (length te7)
    te2a 0)
  (while (/= te2a te7a)
    (setq te8 (nth te2a te7)
      te2a (+ te2a 1))
    (if (= 10 (car te8))
      (setq te9 (cdr te8)
        te3 (cons te9 te3))
      )
    )
  (command "erase" te5 ""))
)
(setq te3a (cdr (reverse (cdr (reverse te3))))
  te2 (length te3a)
  te4 0)
(if (= ckv1 1)
  (progn (while (/= te4 te2)
    (setq te5 (nth te4 te3a)
      te4 (+ te4 1))
    (command "xref" ""
      "c:/skripsi/program/plat.dwg" te5 ".5" "" ""))
  ))
);end of tebal
* Xref (external reference) merupakan perintah
AutoCAD untuk memasukkan block *.
```

Bentuk program untuk simbol batas potong pelat sama dengan program ketebalan pelat tetapi dengan sedikit perbedaan pada masalah pembagian titik dasar (base point) dan nama block.

- Simbol arah penandaan

Simbol ini ditunjukkan dengan dimension "*leader*". Langkah awal dari pemberian simbol arah penandaan adalah setting dimensi.

Bentuk program setting dimensi:

```
(command "style" "romans" "romans" txt ".7"
"" "" "" "")
```

```
(command "dimasz" "40")
```

```
(command "dimtad" "1")
```

```
(COMMAND "DIMTXSTY" "ROMANS")
```

```
(command "dimgap" (/ txt 5))
```

Selanjutnya dilakukan pemberian simbol arah penandaan sesuai setting dimensi yang telah dibuat.

Bentuk program pemberian simbol arah penandaan:

```
(defun aful (/ tt tt1)
```

```
(setq tndc (list (car te3) (last te3)))
```

```
  tndd (length tndc)
```

```
  tnd 0
```

```
  tt "<0")
```

```
(while (/= tndd tnd)
```

```
  (setq tnd1 (nth tnd tndc))
```

```
  (if (= tnd 1)
```

```
    (progn (setq tnd2 "@75<0"
```

```
            tnd3 tanda))
```

```
    (progn (setq tnd2 "@-75<0"
```

```
            tnd3 tanda1))
```

```
  )
```

```
(setq tnd (+ tnd 1))
```

```
(command "dim" "leader" tnd1 tnd2 "" tnd3)
```

```
(command "exit")
```

```
))
```

* tanda atau tanda1 merupakan simbol yang dibaca pada langkah pembuatan body plan *

- Nomer Gading

```
(defun nomerframe1 ()
```

```
(command "style" "romans" "romans" "40"
".75" "" "" "" "" "" ""))
```

```
(setq nfr (ssget "X" '((8 . "frame"))))
```

```
  nfr1 (sslenght nfr)
```

```
  nfr2 0
```

```
  sr5 ()
```

```
  sr2 ()
```

```
  nf "ok"
```

```
  nfa 5
```

```
  nfl "no"
```

```
  nfb 15
```

```
  nfr10 ())
```

```
(while (/= nfr2 nfr1)
```

```
  (setq sr (entget (ssname nfr nfr2)))
```

```
    sr1 (cdr (assoc -1 sr))
```

```
    sr2 (cons sr1 sr2))
```

```
(command "divide" sr1 "20")
```

```
(if (= nfr2 0)
```

```
(setq sr (entget (ssname (ssget "X" '((0 .
"point")))) nfa))
```

```
  sr1 (cdr (assoc 10 sr))
```

```
  sr5 (cons sr1 sr5)
```

```
  ttd nf)
```

```
(progn
```

```
(cond ((= ttd nf)
```

```
(setq sr (entget (ssname (ssget "X" '((0 .
"point")))) nfa))
```

```
  sr1 (cdr (assoc 10 sr))
```

```
  sr5 (cons sr1 sr5)
```

```
  ttd nfr1))
```

```
((= ttd nfr1)
```

```
(setq sr (entget (ssname (ssget "X" '((0 .
"point")))) nfb))
```

```
  sr1 (cdr (assoc 10 sr))
```

```
  sr5 (cons sr1 sr5)
```

```
  ttd nfr))))))
```

```
(command "erase" (ssget "X" '((0 . "point")))
""))
```

```
(setq nfr2 (+ nfr2 1))
```

```
);while
```

```
(setq nfr13 (reverse(cdr sr5)))
```

```
  nfr13 (reverse(cdr nfr13))
```

```
  nfr13a (length nfr13)
```

```
  nfr13b 0)
```

```
(while (/= nfr13b nfr13a)
```

```
(setq nfr18 (nth nfr13b nfr13)
```

```
  nfr19 (itoa (fix (+ nmg nfr13b))))
```

```
  nfr19 (strcat "FR." nfr19)
```

```
  nfr13b (+ nfr13b 1))
```

```
(command "text" "J" "M" nfr18 "" nfr19)
```

```
)
```

```
)
```

LAMPIRAN B.2.16

Untuk mengukur jarak gading pada upper seam maupun lower seam dapat anda lihat pada program dibawah ini:

```
(defun jarak_frame ()
```

```
(setq jr (length tr11)
```

```
  jr1 0
```

```
  jr2 ()
```

```
  jr3 ()
```

```
)
```

```
(while (/= jr1 jr)
```

```
  (setq jr2 (nth jr1 tr11)
```

```
    jr3 (car jr2)
```

```
    jr4 (last jr2)
```

```
    jr5 (cons jr3 jr5)
```

```
    jr6 (cons jr4 jr6))
```

```
(setq jr1 (+ jr1 1))
```

```
)
```

```
(setq jr5 (length jr5)
```

```
  jr6 0
```

```
  jr7 ()
```

```
)
```

```
(while (/= jr6 (- jr5 1))
```

```
  (setq jr7 (nth jr6 jr5)
```

```
    jr8 (nth (+ jr6 1) jr5))
```

```

        jr9 (distance jr7 jr8)
        jru (cons jr9 jru))
(setq jr6 (+ jr6 1))
)
(setq jr10 (length jrk)
 jr11 0
 jrl ())
)
(while (/= jr11 (- jr10 1))
  (setq jr12 (nth jr11 jrk)
 jr13 (nth (+ jr11 1) jrk)
 jr14 (distance jr12 jr13)
 jrl (cons jr14 jrl))
(setq jr11 (+ jr11 1))
)
);end of jarak_frame

```

LAMPIRAN B.2.17

Bentuk program penulisan data pada file:

```

(defun hasil_buka ()
  (setq a (open "c:/skripsi/program/unnamed.dhb"
 "w"))
  bu (length dt12)
  bu1 (length jl12)
  bu2 (length jru)
  bu5 (length jrl)
  bu3 0
  jum1 (sslength (ssget "X" '((8 . "frame"))))
  (write-line (rtos (fix jum1) 2 0) a)
  (setq bb (- bu 1)
 bb1 (nth bb dt12))
  (write-line (rtos bb1 2 0) a)
  (write-line (rtos (car jl12) 2 0) a)
  (write-line (rtos (last jl12) 2 0) a)
  (setq aa (nth 0 dt12))
  (write-line (rtos aa 2 0) a)
  (write-line "Data Jarak Gading Pada Upper
Seam" a)
  (setq bu3 0
 bu6 (reverse jru))
  (while (/= bu3 bu2)
    (write-line (rtos (nth bu3 bu6) 2 0) a)
    (setq bu3 (+ bu3 1))
  )
  (write-line "Data Jarak Gading Pada Lower
Seam" a)
  (setq bu3 0
 bu7 (reverse jrl))
  (while (/= bu3 bu5)
    (write-line (rtos (nth bu3 bu7) 2 0) a)
    (setq bu3 (+ bu3 1))
  )
  (write-line "Data Panjang Kurva" a)
  (setq bu3 0
 bu4 (reverse dt12))
  (while (/= bu3 bu)
    (write-line (rtos (nth bu3 bu4) 2 0) a)
    (setq bu3 (+ bu3 1))
  )
)

```

```

(close a)
);end of hasil_buka

```

LAMPIRAN B.2.18

Bentuk program untuk marking list:

```

(defun mark ()
  (command "layer" "m" "marking" "c" "7" "" "")
  (setq mk (ssget "x" '((8 . "frame")))
 mk1 (sslength mk)
 mk2 (/ mk1 2)
 mk3 (ssname mk mk2))
  (command "divide" mk3 "2")
  (setq mk4 (ssget "X" '((0 . "point")))
 mk5 (ssname mk4 0)
 mk6 (entget mk5)
 mk6a (length mk6)
 mk2a 0)
  (while (/= mk2a mk6a)
    (setq mk7 (nth mk2a mk6)
 mk2a (+ mk2a 1))
    (if (= 10 (car mk7))
      (setq mk8 (cdr mk7)
 mk10 (cons mk8 mk10)))
    (command "erase" mk4 "")
  )
  (setq mk2 (length mk10)
 mk4 0)
  (progn (while (/= mk4 mk2)
    (setq mk5 (nth mk4 mk10)
 mk4 (+ mk4 1))
    (command "xref" "" "c:/skripsi/program/tanda2.d
wg" mk5 "1" "" "")))
  );end of mark
  Bentuk program untuk rambu film sama dengan
marking list cuma diganti nama block.

```

LAMPIRAN B.2.19

Bentuk program pembuatan block pelat hasil pemotongan:

```

(if (= (tblsearch "block" "hasil") nil)
  (progn (command "block" "hasil" '(0 0) "all" "")
    (command "oops"))
  (progn (command "block" "hasil" "Y" '(0 0)
 "all" "")
    (command "oops")))
  Bentuk program animasi:
  (defun msl ()
    (setq ms 250
 mss ms
 msi 5
 ms3 0
 ms4 ())
    (while (/= ms3 ms)
      (setq ms3 (+ mss msi))
      (if (> ms3 360)
        (progn (setq ms3 (- ms3 360)
 ms4 (cons ms3 ms4)))

```

```

        (progn(setq ms4 (cons ms3 ms4)))
        );if
(setq mss ms3)
);while
(setq ms4 (reverse (cons mss (reverse ms4))))
        ms4 (reverse ms4)
        ms5 (length ms4)
ms6 0)
(while (/= ms5 ms6)
(command "vpoint" "r" (nth ms6 ms4) ""))
(setq ms6 (+ ms6 1))
)
);end

(defun ulang ()
(setq ms4 (reverse ms4)
        ms5 (length ms4)
        ms6 0)
(while (/= ms5 ms6)
(command "vpoint" "r" (nth ms6 ms4) ""))
(setq ms6 (+ ms6 1))
);while
);end of ulang
;mengulang putaran plat dalam 3D

(defun c:v3D ()
(setq olderr *error*
        *error* myerr)
(command "erase" "all" "")
(setq ns (open "c:/base_line/filbar2.smb" "r")
        ns1 (read-line ns))
(close ns)
(if (/= (findfile "c:/skripsi/program/has1.dwg")
        nil)
(command "xref" ""
"c:/skripsi/program/has1.dwg" '(0 0) "" "" ""))
(progn (setq vdd 1)
        (command "xref" "" ns1 '(0 0) "" ""
        "")))
(command "vpoint" '(-1 -1 1))
(command "ucs" "x" "90")
(command "ucs" "y" "-90")
(command "rotate" "all" "" '(0 0) "-90")
(command "zoom" "e")
(command "ucs" "w")
(command "Vpoint" "r" "250" "")
(msl)
(ok2)
(command "erase" "all" "")
(command "plan" "")
(setq *error* olderr) (princ)
(if (= vdd 1)
        (command "New" "y" ""))
);end of v3d
Pembuatan program animasi diatas memberikan
suatu pandangan pelat dari berbagai sudut
sehingga terlihat kelengkungan pelat
sebenarnya.

```

LAMPIRAN B.2.20

Program pembuatan kotak dialog (dialog box):

```

(defun ok4 ()
(setq dcl (load_dialog
"c:/skripsi/program/ok2.dcl"))
(if (not (new_dialog "bukaan4" dcl))
        (exit))
(set_tile "zoomW" "1")
(setq zoo 0)
(action_tile "zoomW" "(setq zoo 0) ")
(action_tile "zoomP" "(setq zoo 2) ")
(action_tile "accept" "(setq zoo
1)(done_dialog)")
(action_tile "zoom" "(done_dialog)")
(start_dialog)
(unload_dialog dcl)
(dcomm)
(setq a (sslength aml)
        b 0
        c ())
(while (/= a b)
        (setq d (entget (ssname aml b))
                e (cdr (assoc 8 d)))
        (if (= e "FRAME")
                (setq c (cons e c)))
        (setq b (+ b 1)))
(setq a (sslength aml)
        b 0
        f ())
(while (/= a b)
        (setq d (entget (ssname aml b))
                e (cdr (assoc 8 d)))
        (if (= e "WELD")
                (setq f (cons e f)))
        (setq b (+ b 1)))
(cond ((< (length c) 3)(alert "Jumlah Minimal
Gading = 3")
(COMMAND "REGEN) (setq salah 1)
(GRREAD)
(OK4))
((< (length f) 2)
(alert "Jumlah Minimal Las = 2")
(COMMAND "REGEN")
(setq salah 1)
(GRREAD)
(OK4)))
);end of ok

(defun files (/ fa )
(setq fa(open"c:/skripsi/program/body.txt""w")
        fa1 1)
(close fa)
);end of file

(defun files1 (/ fa )
(setqfa(open"c:/skripsi/program/body1.txt""w)
        fa1 0)
(close fa)
);end of file

```

```

(defun ok ()
  (setq dcl (load_dialog
    "c:/skripsi/program/ok2.dcl"))
  (if (not (new_dialog "bukaan" dcl))
    (exit))
  (set_tile "zoomW" "1")
  (setq zom 1)
  (action_tile "zoomW" "(setq zom 1) ")
  (action_tile "zoomP" "(setq zom 2) ")
  (action_tile "back" "(setq zom 0)
  (done_dialog)")
  (action_tile "besar" "(done_dialog)")
  (start_dialog)
  (unload_dialog dcl)
  (cond ((= zom 0)
    (files)
    (command "erase" "all" ""))
    ((= zom 1)
    (command "zoom" "w" pause pause)
    (ok))
    ((= zom 2)
    (command "zoom" "p")
    (ok)))
  );end of ok

```

```

(defun tmp ()
  (setq dcl (load_dialog
    "c:/skripsi/program/ok2.dcl"))
  (if (not (new_dialog "buka" dcl))
    (exit))
  (set_tile "zoomW" "1")
  (setq zomm 1)
  (action_tile "zoomW" "(setq zomm 1) ")
  (action_tile "zoomP" "(setq zomm 2) ")
  (action_tile "back" "(setq zomm 0)
  (done_dialog)")
  (action_tile "besar" "(done_dialog)")
  (start_dialog)
  (unload_dialog dcl)
  (cond ((= zomm 0)
    (files)
    (command "erase" "all" ""))
    ((= zomm 1)
    (command "zoom" "w" pause pause)
    (tmp))
    ((= zomm 2)
    (command "zoom" "p")
    (tmp)))
  );end of tmp

```

```

(defun cari ()
  (setq au (ssget "X" '((0 . "mtext"))))
  (cond ((/= au nil)
    (setq b (sslenght au)
    ca 0)
    (while (/= ca b)
    (command "explode" (ssname au ca)
    (setq ca (+ 1 ca))))
    ((= au nil)

```

```

(setq a (ssget "X" '((0 . "text"))))
  b (sslenght a)
  c 0
  cl 0)))
  (setq a (ssget "X" '((0 . "text"))))
  (cond ((and (/= edt1 nil) (= edt2 nil))
    (setq e (cons 40 (atof edt1))
    disc edt1
    b (sslenght a)
    cl 0)
    (while (/= cl b)
    (setq z (entget (ssname a cl))
    g (assoc 40 z)
    cl (+ cl 1))
    i (subst e g z))
    (entmod i))
    ((and (= edt1 nil) (/= edt2 nil))
    (setq e (cons 41 (atof edt2))
    disc 30
    b (sslenght a)
    cl 0)
    (while (/= cl b)
    (setq z (entget (ssname a cl))
    g (assoc 41 z)
    cl (+ cl 1))
    i (subst e g z))
    (entmod i)))
    ((and (/= edt2 nil) (/= edt1 nil))
    (setq e (cons 40 (atof edt1))
    disc edt1
    e1 (cons 41 (atof edt2))
    b (sslenght a)
    cl 0)
    (while (/= cl b)
    (setq z (entget (ssname a cl))
    h (assoc 41 z)
    cl (+ cl 1))
    j (subst e1 h z))
    (entmod j))
    (setq cl 0)
    (while (/= cl b)
    (setq z (entget (ssname a cl))
    g (assoc 40 z)
    cl (+ cl 1))
    i (subst e g z))
    (entmod i))))
  (command "dimasz" (* (atof disc) (/ 1 (float
  3))))
  (command "zoom" "e")
  )

```

```

(defun file2 ()
  (setq dcl (load_dialog
    "c:/skripsi/program/ok2.dcl"))
  (if (not (new_dialog "bukaan5" dcl)) (exit))
  (setq a (ssget "X" '((0 . "text"))))
  (setq au (ssget "X" '((0 . "mtext"))))
  (cond ((and (= a nil) (= au nil))
    (alert "Tidak ditemukan TEXT !")
    (setq on 1)

```

```

(done_dialog 0)
(=/= a nil)(setq b (cdr (assoc 40 (entget (ssname
a 0))))
      on 0
      c (cdr (assoc 41 (entget (ssname a
0))))))
(=/= au nil)(setq b (cdr (assoc 40 (entget
(ssname au 0))))
      on 0
      c (cdr (assoc 41 (entget (ssname au
0))))))
(if (/= on 1)
  (progn (set_tile "ut" (rtos b))
         (set_tile "tt" (rtos c)))
  (action_tile "ut" "(setq edt1 $value)")
  (action_tile "tt" "(setq edt2 $value)")
  (action_tile "back" "(done_dialog 0)")
  (action_tile "ok" "(done_dialog 2)")
  (if (= (start_dialog) 2)
    (cari))
  );end of file

(defun ok11 ()
  (setq dcl (load_dialog
"c:/skripsi/program/ok2.dcl"))
  (if (not (new_dialog "bukaan1" dcl))
    (exit))
  (action_tile "exit" "(setq o11 1) (done_dialog
0)")
  (action_tile "accept" "(setq o11 0)
(done_dialog 3)")
  (action_tile "done" "(done_dialog 1)")
  (if (= (start_dialog) 1)
    (progn (file2)
           (ok11)));if
  (if (= fal 0)
    (command "erase" "all" ""));if
  );end of ok11

(defun grand ()
  (ok11)
  )

(defun ok1 ()
  (setq dcl (load_dialog
"c:/skripsi/program/ok2.dcl"))
  (if (not (new_dialog "bukaan6" dcl))
    (exit))
  (action_tile "keluar" "(files1) (done_dialog)")
  (action_tile "kembali" "(files) (done_dialog)")
  (start_dialog)
  (unload_dialog dcl)
  );end of ok1

(defun ok2 ()
  (setq dcl (load_dialog
"c:/skripsi/program/ok2.dcl"))
  (if (not (new_dialog "bukaan2" dcl))
    (exit))
  (action_tile "kk" "(setq on 0)(done_dialog)")

```

```

(action_tile "uu" "(setq on 1)(done_dialog)")
(action_tile "ku" "(setq on 2)(done_dialog)")
(start_dialog)
(unload_dialog dcl)
(cond ((= on 1)
      (ulang)(ok2))
      ((= on 0)
      (files))
      ((= on 2)
      (files1)))
);end of ok2

(defun plan ()
  (setq dcl (load_dialog
"c:/skripsi/program/ok2.dcl"))
  (if (not (new_dialog "hasil" dcl))
    (exit))
  (action_tile "kr" "(setq on 0)(done_dialog)")
  (action_tile "kn" "(setq on 1)(done_dialog)")
  (start_dialog)
  (unload_dialog dcl)
  (cond ((= on 0)
        (kiri))
        ((= on 1)
        (kanan)))
  );end of plan

(defun bentuk ()
  (setq dcl (load_dialog
"c:/skripsi/program/ok2.dcl"))
  (if (not (new_dialog "bentuk" dcl))
    (exit))
  (action_tile "bt" "(setq qw 0)(done_dialog)")
  (action_tile "kb" "(setq qw 1)(done_dialog)")
  (start_dialog)
  (unload_dialog dcl)
  (cond ((= qw 0)
        (bplan))
        ((= qw 1)
        (bplan1)))
  );end of bentuk

(defun output ()
  (setq dcl (load_dialog
"c:/skripsi/program/ok2.dcl"))
  (if (not (new_dialog "hasil1" dcl))
    (exit))
  (action_tile "rf" "(setq as 0)(done_dialog)")
  (action_tile "mrk" "(setq as 1)(done_dialog)")
  (start_dialog)
  (unload_dialog dcl)
  (cond ((= as 0)
        (rambu_film))
        ((= as 1)
        (mark)))
  );end of output

(defun output1 ()
  (setq dcl (load_dialog
"c:/skripsi/program/ok2.dcl"))

```

```
(if (not (new_dialog "hasil2" dcl))
  (exit))
(action_tile "rf" "(setq af 0)(done_dialog)")
(action_tile "mrk1" "(setq af 1)(done_dialog)")
(start_dialog)
(unload_dialog dcl)
(cond ((= af 0)
  (rambu_film))
  ((= af 1)
  (mark1)))
);end of output1
```

```
(defun ok3 ()
(setq dcl (load_dialog "c:/base_line/ok.dcl"))
(if (not (new_dialog "bukaan3" dcl))
  (exit))
(action_tile "exit" "(files1) (done_dialog)")
(action_tile "accept" "(files) (done_dialog)")
(start_dialog)
(unload_dialog dcl)
(if (= fa1 1)
  (command "erase" "all" ""));if
(if (= fa1 0)
  (command "erase" "all" ""));if
);end of ok3
```

LAMPIRAN B.2.21

Bentuk program utama untuk menjalankan sub-program dibawah ini:

```
(defun c:buka ()
(setq olderr *error* *error* myerr)
(command "erase" "all" "")
(command "plan" "")
(command "zoom" "e")
(bentuk)
(plan)
(ok11)
(if (= (findfile"c:\\skripsi\\program\\has1.dwg")
  nil)
  (command"wblock""c:\\skripsi\\program\\has1.dwg" "potongan")
  (progn(command"del""c:\\skripsi\\program\\has1.dwg")
  (command"wblock""c:\\skripsi\\program\\has1.dwg" "potongan")))
  (if(=(findfile"c:\\skripsi\\program\\has2.dwg")
  nil)
  (command"wblock""c:\\skripsi\\program\\has2.dwg" "has_buk")
  (progn(command"del""c:\\skripsi\\program\\has2.dwg")
  (command"wblock""c:\\skripsi\\program\\has2.dwg" "has_buk")))
  (if(=(findfile"c:\\skripsi\\program\\has3.dwg")
  nil)
  (command"wblock""c:\\skripsi\\program\\has3.dwg" "hasil")
  (progn(command"del""c:\\skripsi\\program\\has3.dwg")
```

```
(command"wblock""c:\\skripsi\\program\\has3.dwg" "hasil")))
  (if(=(findfile"c:\\skripsi\\program\\has4.dwg")
  nil)
  (command"wblock""c:\\skripsi\\program\\has4.dwg" "pot1")
  (progn(command"del""c:\\skripsi\\program\\has4.dwg")
  (command"wblock"
  "c:\\skripsi\\program\\has4.dwg" "pot1")))
  (if(=(findfile"c:\\skripsi\\program\\has5.dwg")
  nil)
  (command"wblock""c:\\skripsi\\program\\has5.dwg" "body")
  (progn(command"del""c:\\skripsi\\program\\has5.dwg")
  (command"wblock""c:\\skripsi\\program\\has5.dwg" "body")))
  (setq *error* olderr)(princ)
  (cond ((= o11 1) (files1))
  ((= o11 0) (files)))
  (command "erase" "all" ""))
);end of buka
```

(Display) merupakan sub-program untuk membuat body plan.

(Plan) merupakan sub-program bukaan kulit.

Program utama (c: buka ()) nantinya dapat dijalankan pada Microsoft Visual Basic dengan membuat tombol button yang memuat program utama.

Sub-program untuk menjalankan program metode flat plan dan penggambaran body plan :

```
(defun flat1 ()
(command "erase" (ssget "x" '((8 . "text")))) ""
(bagi_frame)
(dist)
(tprok)
(fspace1)
(proyeksi)
(epoint)
(point1)
(point2)
(seam1)
(seam2)
(tegak)
(circle)
(command "erase" (ssget "x" '((8 . "frame")))) ""
(command "erase" (ssget "x" '((8 . "weld")))) ""
(command "layer" "off" "line" "")
(command "layer" "off" "seam1" "")
(command "layer" "off" "weld1" "")
(ide1)
(command "erase" (ssget "x" '((8 . "gores")))) ""
(command "erase" (ssget "x" '((8 . "tegak")))) ""
(gabung)
```



```

(command "layer" "on" "seam1" "")
(command "layer" "on" "weld1" "")
(upper)
(panjang)
(command "layer" "off" "seam1" "")
(sudut)
(setq z 3
  txt 75)
(command "zoom" "e")
(command "layer" "m" "text" "")
(command "style" "romans" "romans" txt ".7"
  "" "" "" "")
(command "dimasz" "40")
(command "dimtad" "1")
(COMMAND "DIMTXSTY" "ROMANS")
(command "dimgap" (/ txt 5))
(nomerframe)
(fair)
(temu)
(jarak_frame)
(tebal)
(pot)
(pot1)
(aful)
(command "zoom" "e")
(aful2)
(setq nae (strcat
  "c:\\skripsi\\program\\bukaan.dxf"));export file
(command "export" nae "")
(command "zoom" "e")
(if (= (tblsearch "block" "hasil") nil)
  (progn(command "block" "hasil" '(0 0) "all"
  ""))
  (command "oops"))
  (progn(command "block" "hasil" "Y" '(0 0)
  "all" ""))
  (command "oops")))
(bagi_las)
(jlas)
(hasil_buka)
(cond ((= qw 0)(close file))
  ((= qw 1)(close file)))
);flat1

(defun flat2 ()
  (command "layer" "m" "text" "c" "4" "" "")
  (nomerframe3)
  (command "erase" (ssget "x" '((8 . "text"))) "")
  (bagi_frame)
  (dist)
  (tprok)
  (fspace1)
  (proyeksi)
  (epoint)
  (point1a)
  (point2a)
  (seam3)
  (seam4)
  (tegak1)
  (circle1)

```

```

(command "erase" (ssget "x" '((8 . "frame")))
  "")
(command "erase" (ssget "x" '((8 . "weld"))) "")
(command "layer" "off" "line" "")
(command "layer" "off" "seam1" "")
(command "layer" "off" "weld1" "")
(ide1)
(command "erase" (ssget "x" '((8 . "gores")))
  "")
(command "erase" (ssget "x" '((8 . "tegak")))
  "")
(gabung1)
(command "layer" "on" "seam1" "")
(command "layer" "on" "weld1" "")
(upper)
(panjang1)
(command "layer" "off" "seam1" "")
(sudut1)
(setq z 3
  txt 75)
(command "zoom" "e")
(command "layer" "m" "text" "")
(command "style" "romans" "romans" txt ".7"
  "" "" "" "")
(command "dimasz" "40")
(command "dimtad" "1")
(COMMAND "DIMTXSTY" "ROMANS")
(command "dimgap" (/ txt 5))
(nomerframe2)
(fair)
(temu)
(jarak_frame)
(tebal)
(pot)
(pot1)
(aful1)
(command "zoom" "e")
(aful2)
(setq nae (strcat
  "c:\\skripsi\\program\\bukaan.dxf"))
(command "export" nae "")
(command "zoom" "e")
(if (= (tblsearch "block" "hasil") nil)
  (progn(command "block" "hasil" '(0 0) "all"
  ""))
  (command "oops"))
  (progn(command "block" "hasil" "Y" '(0 0)
  "all" ""))
  (command "oops")))
(bagi_las)
(jlas)
(hasil_buka)
(cond ((= qw 0)(close file))
  ((= qw 1)(close file)))
);flat2

(defun diski ()
  (cond ((= qw 0)(frame1)(weld1))
  ((= qw 1)(gading1)(las1)))
  (fair)

```

```

Dim tt4(1000)
Dim ptArray(1000)
Dim batalw
Dim namafile As String
Public acadApp As Object      'The AutoCAD
application object
Public acadDoc As Object      'The AutoCAD
document (drawing) object
Public moSpace As Object
Public paSpace As Object

```

```

Private Sub Form_Activate()
If ScaleWidth < 0 Or ScaleHeight < 0 Then
Picture1.Move 50, 450, ScaleWidth - 100,
ScaleHeight - 850
End If
End Sub

```

```

Private Sub Form_Deactivate()
On Error Resume Next
If Me.WindowState = 1 Then
For i = Forms.Count - 1 To 1 Step -1
Forms(i).WindowState = 1
Next
hapusfile
bersih
SetacadApp=GetObject("AutoCAD.Applicatio
n")
acadApp.Visible = False
acadApp.quit
Set acadApp = Nothing
End If
End Sub

```

```

Private Sub Form_Load()
MousePointer = 11
ChDir "c:\skripsi\program"
ChDrive App.Path
namafile = "Unnamed"
Form6.Caption = "Flat Plan : " & namafile
bersih
MousePointer = 0
End Sub

```

```

Private Sub Form_Unload(Cancel As Integer)
On Error Resume Next
hapusfile
bersih
SetacadApp=GetObject("AutoCAD.Applicatio
n")
Set acadDoc = acadApp.ActiveDocument
If Not acadDoc.Saved Then
Set acadApp = Nothing
Dim i As Integer
For i = Forms.Count - 1 To 1 Step -1
Unload Forms(i)
Next
If Me.WindowState < vbMinimized Then
SaveSetting App.Title, "Settings", "MainLeft",
Me.Left

```

```

SaveSetting App.Title, "Settings", "MainTop",
Me.Top
SaveSettingApp.Title, "Settings",
"MainWidth", Me.Width
SaveSettingApp.Title, "Settings",
"MainHeight", Me.Height
End If
Set acadApp = Nothing
End If
For i = Forms.Count - 1 To 1 Step -1
Unload Forms(i)
Next
If Me.WindowState < vbMinimized Then
SaveSetting App.Title, "Settings", "MainLeft",
Me.Left
SaveSetting App.Title, "Settings", "MainTop",
Me.Top
SaveSettingApp.Title, "Settings",
"MainWidth", Me.Width
SaveSettingApp.Title, "Settings",
"MainHeight", Me.Height
End If
End Sub

```

```

Private Sub mnuAnimasi_Click()
'On Error GoTo ErrorHandler
f = FreeFile
Open "c:\skripsi\program\filbar2.smb" For
Output As f
Print #f, format!Text19.Text
Close f

```

```

X = Dir("c:\skripsi\program\body.txt")
Y = Dir("c:\skripsi\program\body1.txt")

```

```

If X <> nil Then
Kill "c:\skripsi\program\body.txt"
ElseIf Y <> nil Then
Kill "c:\skripsi\program\body1.txt"
Else
End If

```

```

If Dir(format!Text19.Text) = nil Then
acad_show
AppActivate "AutoCAD"
SendKeys "{esc}"
SendKeys "proses" & "{enter}", True
SendKeys "(v3d)" & "{enter}", True
st = Timer
Do While Timer > st - 1
X = Dir("c:\skripsi\program\body.txt")
Y = Dir("c:\skripsi\program\body1.txt")
If X <> nil Or Y <> 1 Then
st = Timer + Timer
Else
End If
Loop
acad_visible
AppActivate Form6.Caption
Else

```

```

(fair1)
(if (= (tblsearch "block" "body") nil)
  (progn(command "block" "body" '(0 0) "all"
    ""))
  (command "oops")))
  (progn(command "block" "body" "Y" '(0 0)
    "all" ""))
    (command "oops")))
(command "erase" (ssget "x" '((8 . "0"))) "")
(setq *error* olderr)
(princ)
);end of disk1

(defun diskan ()
(cond ((= qw 0)(frame2)(weld2))
  ((= qw 1)(gading2)(las2)))
(fair)
(fair1)
(if (= (tblsearch "block" "body") nil)
  (progn(command "block" "body" '(0 0) "all"
    ""))
  (command "oops")))
  (progn(command "block" "body" "Y" '(0 0)
    "all" ""))
    (command "oops")))
(command "erase" (ssget "x" '((8 . "0"))) "")
(setq *error* olderr)
(princ)
);end of diskan

(defun kiri())
(diski)
(command "zoom" "e")
(ambil)
(titik)
(titik1)
(potong1)
(ambill)
(titik)
(titik1)
(potong)
(if (= (tblsearch "block" "potongan") nil)
  (progn(command "block" "potongan" '(0 0)
    "all" ""))
  (command "oops")))
  (progn(command "block" "potongan" "Y"
    '(0 0) "all" ""))
    (command "oops")))
(command "layer" "m" "text" "c" "4" "" "")
(nomerframe1)

(if (= (tblsearch "block" "pot1") nil)
  (progn(command "block" "pot1" '(0 0) "all"
    ""))
  (command "oops")))
  (progn(command "block" "pot1" "Y" '(0 0)
    "all" ""))
    (command "oops")))
(command "regen")
(command "zoom" "e")

```

```

(command "regen")
(command "zoom" "all")
(flat1)
(output)
(command "zoom" "e")
);end of kiri

(defun kanan())
(diskan)
(command "zoom" "e")
(ambil)
(titik)
(titik1)
(potong1)
(ambill)
(titik)
(titik1)
(potong)
(if (= (tblsearch "block" "potongan") nil)
  (progn(command "block" "potongan" '(0 0)
    "all" ""))
  (command "oops")))
  (progn(command "block" "potongan" "Y" '(0
    0) "all" ""))
    (command "oops")))
(command "layer" "m" "text" "c" "4" "" "")
(nomerframe3)
(if (= (tblsearch "block" "pot1") nil)
  (progn(command "block" "pot1" '(0 0) "all" ""))
  (command "oops")))
  (progn(command "block" "pot1" "Y" '(0 0)
    "all" ""))
    (command "oops")))
(command "regen")
(command "zoom" "e")
(command "regen")
(command "zoom" "all")
(flat2)
(output1)
(command "zoom" "e")
);end of kanan

```

Lampiran B.2.22

Listing Form Program Utama :

```

Private Declare Function OSWinHelp% Lib
"user32" Alias "WinHelpA" (ByVal hWnd&,
ByVal HelpFile$, ByVal wCommand%,
dwData As Any)
Public xlApp As Object
Dim nomorfile
Dim st, pt, fl As Integer
Dim i, j As Integer
Dim jj, Y, p As Integer
Dim f, OK, ok1, ok2 As Integer
Dim ck1, ck2 As Integer
Dim tt(1000)
Dim ttl
Dim tt1(1000)
Dim tt2(1000)
Dim tt3(1000)

```

```

acad_show
AppActivate "AutoCAD"
SendKeys "{esc}"
SendKeys "proses" & "{enter}", True
SendKeys "(v3d)" & "{enter}", True
st = Timer
  Do While Timer > st - 1
    X = Dir("c:\skripsi\program\body.txt")
    Y = Dir("c:\skripsi\program\body1.txt")
    If X <> nil Or Y <> nil Then
      st = Timer + Timer
    Else
      End If
  Loop
  acad_visible
  AppActivate Form6.Caption
End If

If X <> nil Then
Kill "c:\skripsi\program\body.txt"
ElseIf Y <> nil Then
Kill "c:\skripsi\program\body1.txt"
Unload format: Unload frmSplash: Unload
Form6: Unload Ukuran: Unload Plan
Else
End If
MousePointer = 0
End Sub

Private Sub mnuBentuk_Click()
'On Error GoTo ErrorHandler
MousePointer = 11
f = FreeFile
Open "c:\skripsi\program\filbar1.smb" For
Output As f
Print #f, format!Text18.Text
Close f
X = Dir("c:\skripsi\program\body.txt")
Y = Dir("c:\skripsi\program\body1.txt")
If X <> nil Then
Kill "c:\skripsi\program\body.txt"
ElseIf Y <> nil Then
Kill "c:\skripsi\program\body1.txt"
Else
End If

If Dir("c:\skripsi\program\has3.dwg") <> nil
Then
acad_show
AppActivate "AutoCAD"
SendKeys "{esc}"
SendKeys "proses" & "{enter}", True
SendKeys "bki3" & "{enter}", True
st = Timer
  Do While Timer > st - 1
    X = Dir("c:\skripsi\program\body.txt")
    Y = Dir("c:\skripsi\program\body1.txt")
    If X <> nil Or Y <> nil Then
      st = Timer + Timer
    Else

```

```

End If
Loop
acad_visible
AppActivate Form6.Caption
Else
acad_show
AppActivate "AutoCAD"
SendKeys "{esc}"
SendKeys "proses" & "{enter}", True
SendKeys "ins2" & "{enter}", True
  Do While Timer > st - 1
    X = Dir("c:\skripsi\program\body.txt")
    Y = Dir("c:\skripsi\program\body1.txt")
    If X <> nil Or Y <> nil Then
      st = Timer + Timer
    Else
      End If
  Loop
  acad_visible
  AppActivate Form6.Caption
End If
If X <> nil Then
  AppActivate Form6.Caption
  Kill "c:\skripsi\program\body.txt"
ElseIf Y <> nil Then
  Kill "c:\skripsi\program\body1.txt"
  Unload format: Unload frmSplash: Unload
  Form6: Unload Ukuran: Unload Plan
Else
End If
MousePointer = 0
End Sub

Private Sub mnuDimensi_Click()
Dim rr
Ukuran.Show
rr=format!Text10.Text+Fix((format!Text6.Tex
t - 2) / 2) - 1
Ukuran!Text5.Text = "(" & "" & "Fr." & "" &
rr & "" & ")"
mtukuran
inputdata
End Sub

Private Sub mnuDXF_Click()
export_dxf
End Sub

Private Sub export_dxf()
On Error Resume Next
MousePointer = 11
Dim Counter As Integer
Dim Workarea(9) As String
Form6!ProgressBar1.Min=LBound(Workarea)
Form6!ProgressBar1.Max=UBound(Workarea)
Form6!ProgressBar1.Visible = True
Form6!ProgressBar1.Top = 8150
Form6!ProgressBar1.Value=ProgressBar1.Min
ForCounter=LBound(Workarea)ToUBound(W
orkarea)

```

```

Workarea(Counter) = "Initial value" & Counter
Form6!ProgressBar1.Value = Counter
If Counter = 1 Then
    EB = Len(filename)
    For i = 1 To EB
        If ed <> EB Then
            EG = Left(filename, i)
            EC(i) = Right(EG, 1)
            If EC(i) = "." Then
                ed = i: EB = i
            End If
        End If
    Next
    exportfile = Left(filename, ed - 1)
    expor = Left(filename, Len(filename) - 4)
ElseIf Counter = 2 Then
    If Dir(expor & ".1.dwg") <> nil Then
        Kill Dir(expor & ".1.dwg")
    End If
    If Dir(expor & ".2.dwg") <> nil Then
        Kill Dir(expor & ".2.dwg")
    End If
ElseIf Counter = 3 Then
    If Dir(expor & ".3.dwg") <> nil Then
        Kill Dir(expor & ".3.dwg")
    End If
    If Dir(expor & ".dxf") <> nil Then
        Kill Dir(expor & ".dxf")
    End If
    If Dir(expor & ".dhh") <> nil Then
        Kill Dir(expor & ".dhh")
    End If
ElseIf Counter = 4 Then
    Set doc = acadApp.ActiveDocument
ElseIf Counter = 5 Then

oldpathname=Dir("c:\skripsi\program\has1.dwg
")
newpathname = expor & ".1.dwg"
Name oldpathname As newpathname
ElseIf Counter = 6 Then
oldpathname=Dir("c:\skripsi\program\has2.dwg
")
newpathname = expor & ".2.dwg"
Name oldpathname As newpathname
ElseIf Counter = 7 Then
oldpathname=Dir("c:\skripsi\program\has3.dwg
")
newpathname = expor & ".3.dwg"
Name oldpathname As newpathname
ElseIf Counter = 8 Then
oldpathname=Dir("c:\skripsi\program\bukaan.d
xf")
newpathname = expor & ".dxf"
Name oldpathname As newpathname
ElseIf Counter = 9 Then
oldpathname=Dir("c:\skripsi\program\unnamed
.dhh")
newpathname = expor & ".dhh"
Name oldpathname As newpathname

```

```

End If
Next Counter
Form6!ProgressBar1.Visible = False
Form6!ProgressBar1.Value=
Form6!ProgressBar1.Min
MousePointer = 0
End Sub

Private Sub mnuExit_Click()
Unload Form6: Unload format: Unload
frmSplash: Unload Ukuran: Unload Plan
End Sub

Private Sub mnuNew_Click()
Dim pilihan
MousePointer = 11
SetacadApp=GetObject("AutoCAD.Applicatio
n")
Set acadDoc = acadApp.ActiveDocument
If Not acadDoc.Saved Then
If MsgBox("OK to save drawing?", 4) = vbNo
Then
GoTo SKIPNEW
Else
acadDoc.Save
End If
End If
Set acadDoc = acadDoc.New("acad")
SKIPNEW:
If filename <> "Unnamed" Then
pilihan = MsgBox("Anda ingin menyimpan
file ini ?", 3 + 32, "Pesan")
MousePointer = 11
If pilihan = 6 Then
simpan (filename)
ElseIf pilihan = 2 Then
Exit Sub
Else
MousePointer = 11
Dim Counter As Integer
Dim Workarea(4) As String
ProgressBar1.Min = LBound(Workarea)
ProgressBar1.Max = UBound(Workarea)
ProgressBar1.Visible = True
ProgressBar1.Top = 8150
ProgressBar1.Value = ProgressBar1.Min
For Counter = LBound(Workarea) To
UBound(Workarea)
Workarea(Counter) = "Initial value" & Counter
ProgressBar1.Value = Counter
If Counter = 1 Then
Unload format: Unload Ukuran: Unload Plan
filename = "Unnamed"
Form6.Caption = "Bukaan : " & filename
ElseIf Counter = 2 Then
SetacadApp=GetObject("AutoCAD.Applicatio
n")
Set acadDoc = acadApp.ActiveDocument
ElseIf Counter = 3 Then
bersih

```

```

End If
Next Counter
ProgressBar1.Visible = False
ProgressBar1.Value = ProgressBar1.Min
MousePointer = 0
End If
End If
bersih
Form6!Toolbar1.Buttons("data").Enabled =
False
Form6!Toolbar1.Buttons("v3d").Enabled =
False
Form6!Toolbar1.Buttons("hasil").Enabled =
False
Form6!Toolbar1.Buttons("potong").Enabled =
False
Form6!Toolbar1.Buttons("body").Enabled =
False
Form6!Toolbar1.Buttons("proses").Enabled =
False
Form6!Toolbar1.Buttons("rambu").Enabled =
False
Form6!Toolbar1.Buttons("export").Enabled =
False
Form6!Toolbar1.Buttons("tabel").Enabled =
False
MousePointer = 0
End Sub

```

```

Private Sub mnuOpen_Click()
MousePointer = 11
Dim f As Integer
If Dir("c:\skripsi\program") <> nil Then
ChDir "c:\skripsi\program"
End If
SetacadApp=GetObject("AutoCAD.Applicatio
n")
Set acadDoc = acadApp.ActiveDocument
If Not acadDoc.Saved Then
If MsgBox("Simpan File ?", 4) = vbNo Then
GoTo SKIPNEW
Else
acadDoc.Save
End If
End If
Set acadDoc = acadDoc.New("acad")
SKIPNEW:
MousePointer = 0
fl = 1
CmDialog1.DialogTitle = "Buka File"
CmDialog1.Filter="Bukaan Files (*.dat)*.dat|"
CmDialog1.FilterIndex = 2
CmDialog1.Action = 1
filename = CmDialog1.filename
f = FreeFile
If filename = "" Then
format.Show: format!Text1.SetFocus
Exit Sub
End If

```

```

Unload format: Unload frmSplash: Unload
Ukuran: Unload Plan
Load format: Load Ukuran: Load Plan
format!Text20.Text = filename
Form6.Caption = "Flat Plan : " & filename
bersih
format.Show: format!Text1.SetFocus
Open filename For Input As f
Do Until EOF(f)
Input #f, Lp, b, H
format!Text1.Text = Lp: format!Text2.Text =
b: format!Text3.Text = H
Input #f, t, v
format!Text4.Text = t: format!Text5.Text = v
Input #f, t1, t2
format!Text6.Text = t1: format!Text7.Text =
t2
Input #f, t3, t4
format!Text9.Text = t3: format!Text10.Text =
t4
Input #f, t5
format!Text8.Text = t5

Input #f, t6
format!Check1.Value = t6
Input #f, t7
format!Check2.Value = t7
Input #f, t8
format!Check3.Value = t8
Input #f, t9
format!Check4.Value = t9
Input #f, t10
format!Check5.Value = t10
Input #f, t11
format!Check6.Value = t11
Input #f, t12
format!Check7.Value = t12
Input #f, t13
format!Check8.Value = t13
Input #f, t14
format!Check9.Value = t14
Input #f, t15
format!Text21.Text = t15

format!tabel = True

For i = 1 To format!Text6.Text
For j = 1 To format!Text7.Text
Input #f, isa
format!Grid1.Col = i: format!Grid1.Row = j
format!Grid1.Text = isa
Next
Next

For i = 1 To format!Text6.Text
For j = 1 To format!Text7.Text
Input #f, isa1
format!Grid2.Col = i: format!Grid2.Row = j
format!Grid2.Text = isa1
Next

```

```

Next

For i = 1 To 2
For j = 1 To format!Text6.Text
Input #f, isa4
format!Grid3.Col = i: format!Grid3.Row = j
format!Grid3.Text = isa4
Next
Next

For i = 1 To 2
For j = 1 To format!Text6.Text
Input #f, isa5
format!Grid4.Col = i: format!Grid4.Row = j
format!Grid4.Text = isa5
Next
Next
format.Show: Form6.Hide
EB = Len(filename)
  For i = 1 To EB
    If ed <> EB Then
      EG = Left(filename, i)
      EC(i) = Right(EG, 1)
      If EC(i) = "." Then
        ed = i: EB = i
      End If
    End If
  Next
  exportfile = Left(filename, ed - 1)
  If Dir(exportfile & ".1.dwg") <> nil Then

Form6!Toolbar1.Buttons("potong").Enabled =
True
  Form6!mnuPelat.Enabled = True
  Else

Form6!Toolbar1.Buttons("potong").Enabled =
False
  Form6!mnuPelat.Enabled = False
  End If
  If Dir(exportfile & ".2.dwg") <> nil Then

Form6!Toolbar1.Buttons("rambu").Enabled =
True
  Form6!mnuOutput.Enabled = True
  Else

Form6!Toolbar1.Buttons("rambu").Enabled =
False
  Form6!mnuOutput.Enabled = False
  End If
  If Dir(exportfile & ".3.dwg") <> nil Then
    Form6!Toolbar1.Buttons("hasil").Enabled
= True
    Form6!mnuBentuk.Enabled = True
  Else
    Form6!Toolbar1.Buttons("hasil").Enabled
= False
    Form6!mnuBentuk.Enabled = False
  End If

  If Dir(exportfile & ".dxf") <> nil Then
    Form6!Toolbar1.Buttons("export").Enabled =
    True
    Form6!mnuDXF.Enabled = True
  Else

Form6!Toolbar1.Buttons("export").Enabled =
False
  Form6!mnuDXF.Enabled = False
  End If
  If Dir(exportfile & ".dwb") <> nil Then
    Form6!Toolbar1.Buttons("data").Enabled
= True
    Form6!mnuDimensi.Enabled = True
  Else
    Form6!Toolbar1.Buttons("data").Enabled
= False
    Form6!mnuDimensi.Enabled = False
  End If
  If Dir("c:\skripsi\program\has1.dwg") <> nil
  Then
    Kill ("c:\skripsi\program\has1.dwg")
  End If
  If Dir("c:\skripsi\program\has2.dwg") <> nil
  Then
    Kill ("c:\skripsi\program\has2.dwg")
  End If
  If Dir("c:\skripsi\program\has3.dwg") <> nil
  Then
    Kill ("c:\skripsi\program\has3.dwg")
  End If
  If Dir("c:\skripsi\program\has5.dwg") <> nil
  Then
    Kill ("c:\skripsi\program\has5.dwg")
  End If
  If Dir("c:\skripsi\program\unnamed.dhb") <>
  nil Then
    Kill ("c:\skripsi\program\unnamed.dhb")
  End If
  If Dir("c:\skripsi\program\bukaan.dxf") <> nil
  Then
    Kill ("c:\skripsi\program\bukaan.dxf")
  End If
  Loop
  Close #f, a, b, d, f, g, H, i
  Dim c(10000)
  a = filename
  b = Len(a)

  For i = 1 To b
  If f <> e Then
    c(i) = Right(a, 1)
    a = Left(a, Val(b) - i)
    If c(i) = "\" Then
      e = i - 1
      f = e
    End If
  End If
  Next

```

```

g = Right(filename, e)
hh = Left(filename, b - e) & Left(g, Len(g) - 3)
& "dhh"
ofi = Left(filename, b - e) & Left(g, Len(g) - 4)

```

```

If Dir(hh) <> nil Then
format!Text16.Text = hh
End If

```

```

nama1 = ofi & "1" & ".dwg"
nama2 = ofi & "2" & ".dwg"
nama3 = ofi & "3" & ".dwg"

```

```

If Dir(nama1) <> nil Then
format!Text17.Text = nama1
Else

```

```

format!Text17.Text = ""
End If

```

```

If Dir(nama2) <> nil Then
format!Text18.Text = nama2
Else

```

```

format!Text18.Text = ""
End If

```

```

If Dir(nama3) <> nil Then
format!Text19.Text = nama3
Else

```

```

format!Text19.Text = ""
End If

```

```

Form6!Toolbar1.Buttons("proses").Enabled =
True

```

```

Form6!Toolbar1.Buttons("tabel").Enabled =
True

```

```

Form6!mnuTabel.Enabled = True

```

```

Form6!mnuRun.Enabled = True

```

```

MousePointer = 0

```

```

End Sub

```

```

Private Sub mnuOutput_Click()

```

```

'On Error GoTo ErrorHandler

```

```

MousePointer = 11

```

```

f = FreeFile

```

```

Open "c:\skripsi\program\filbar1.smb" For
Output As f

```

```

Print #f, format!Text19.Text

```

```

Close f

```

```

X = Dir("c:\skripsi\program\body.txt")

```

```

Y = Dir("c:\skripsi\program\body1.txt")

```

```

If X <> nil Then

```

```

Kill "c:\skripsi\program\body.txt"

```

```

ElseIf Y <> nil Then

```

```

Kill "c:\skripsi\program\body1.txt"

```

```

Else

```

```

End If

```

```

If Dir("c:\skripsi\program\has2.dwg") <> nil
Then

```

```

acad_show

```

```

AppActivate "AutoCAD"

```

```

SendKeys "{esc}"

```

```

SendKeys "proses" & "{enter}", True

```

```

SendKeys "bki2" & "{enter}", True

```

```

st = Timer

```

```

Do While Timer > st - 1

```

```

X = Dir("c:\skripsi\program\body.txt")

```

```

Y = Dir("c:\skripsi\program\body1.txt")

```

```

If X <> nil Or Y <> nil Then

```

```

st = Timer + Timer

```

```

Else

```

```

End If

```

```

Loop

```

```

acad_visible

```

```

AppActivate Form6.Caption

```

```

Else

```

```

acad_show

```

```

AppActivate "AutoCAD"

```

```

SendKeys "{esc}"

```

```

SendKeys "proses" & "{enter}", True

```

```

SendKeys "ins2" & "{enter}", True

```

```

Do While Timer > st - 1

```

```

X = Dir("c:\skripsi\program\body.txt")

```

```

Y = Dir("c:\skripsi\program\body1.txt")

```

```

If X <> nil Or Y <> nil Then

```

```

st = Timer + Timer

```

```

Else

```

```

End If

```

```

Loop

```

```

acad_visible

```

```

AppActivate Form6.Caption

```

```

End If

```

```

If X <> nil Then

```

```

AppActivate Form6.Caption

```

```

Kill "c:\skripsi\program\body.txt"

```

```

ElseIf Y <> nil Then

```

```

Kill "c:\skripsi\program\body1.txt"

```

```

Unload format: Unload frmSplash: Unload

```

```

Form6: Unload Ukuran: Unload Plan

```

```

Else

```

```

End If

```

```

MousePointer = 0

```

```

End Sub

```

```

Private Sub mnuPelat_Click()

```

```

'On Error GoTo ErrorHandler

```

```

MousePointer = 11

```

```

f = FreeFile

```

```

Open "c:\skripsi\program\filbar.smb" For
Output As f

```

```

Print #f, format!Text17.Text

```

```

Close f

```

```

X = Dir("c:\skripsi\program\body.txt")

```

```

Y = Dir("c:\skripsi\program\body1.txt")

```

```

If X <> nil Then

```

```

Kill "c:\skripsi\program\body.txt"

```

```

ElseIf Y <> nil Then

```

```

Kill "c:\skripsi\program\body1.txt"

```

```

Else

```

```

End If

```



```

If Dir("c:\skripsi\program\has1.dwg") <> nil
Then
acad_show
AppActivate "AutoCAD"
SendKeys "{esc}"
SendKeys "proses" & "{enter}", True
SendKeys "bki1" & "{enter}", True
st = Timer
  Do While Timer > st - 1
  X = Dir("c:\skripsi\program\body.txt")
  Y = Dir("c:\skripsi\program\body1.txt")
  If X <> nil Or Y <> nil Then
  st = Timer + Timer
  Else
  End If
  Loop
  acad_visible
  AppActivate Form6.Caption
Else
acad_show
AppActivate "AutoCAD"
SendKeys "{esc}"
SendKeys "main" & "{enter}", True
SendKeys "ins1" & "{enter}", True
st = Timer
  Do While Timer > st - 1
  X = Dir("c:\skripsi\program\body.txt")
  Y = Dir("c:\skripsi\program\body1.txt")
  If X <> nil Or Y <> nil Then
  st = Timer + Timer
  Else
  End If
  Loop
  acad_visible
  AppActivate Form6.Caption
End If
If X <> nil Then
  AppActivate Form6.Caption
  Kill "c:\skripsi\program\body.txt"
ElseIf Y <> nil Then
  Kill "c:\skripsi\program\body1.txt"
  Unload format: Unload frmSplash: Unload
Form6: Unload Ukuran: Unload Plan
End If
MousePointer = 0
End Sub
Private Sub mnuPlan_Click()
MousePointer = 11
f = FreeFile
Open "c:\skripsi\program\filbar1.smb" For
Output As f
Print #f, format!Text18.Text
Close f
X = Dir("c:\skripsi\program\body.txt")
Y = Dir("c:\skripsi\program\body1.txt")
If X <> nil Then
Kill "c:\skripsi\program\body.txt"
ElseIf Y <> nil Then
Kill "c:\skripsi\program\body1.txt"
Else

```

```

End If
If Dir("c:\skripsi\program\has5.dwg") <> nil
Then
acad_show
AppActivate "AutoCAD"
SendKeys "{esc}"
SendKeys "proses" & "{enter}", True
SendKeys "bki4" & "{enter}", True
st = Timer
  Do While Timer > st - 1
  X = Dir("c:\skripsi\program\body.txt")
  Y = Dir("c:\skripsi\program\body1.txt")
  If X <> nil Or Y <> nil Then
  st = Timer + Timer
  Else
  End If
  Loop
  acad_visible
  AppActivate Form6.Caption
Else
acad_show
AppActivate "AutoCAD"
SendKeys "{esc}"
SendKeys "proses" & "{enter}", True
SendKeys "ins2" & "{enter}", True
  Do While Timer > st - 1
  X = Dir("c:\skripsi\program\body.txt")
  Y = Dir("c:\skripsi\program\body1.txt")
  If X <> nil Or Y <> nil Then
  st = Timer + Timer
  Else
  End If
  Loop
  acad_visible
  AppActivate Form6.Caption
End If
If X <> nil Then
  AppActivate Form6.Caption
  Kill "c:\skripsi\program\body.txt"
ElseIf Y <> nil Then
  Kill "c:\skripsi\program\body1.txt"
  Unload format: Unload frmSplash: Unload
Form6: Unload Ukuran: Unload Plan
Else
End If
MousePointer = 0
End Sub
Private Sub mnuPlot_Click()
AppActivate "AutoCAD"
SendKeys "{esc}"
SendKeys "plot" & "{enter}", True
'AppActivate Form6.Caption
End Sub
Private Sub mnuRed_Click()
CmDialog2.HelpFile="C:\DATA\HC\ICONW
RKS.HLP"
CmDialog2.HelpCommand= cdlHelpContents
CmDialog2.ShowHelp

```

End Sub

Private Sub mnuRun_Click()

MousePointer = 11

X = Dir("c:\skripsi\program\body.txt")

Y = Dir("c:\skripsi\program\body1.txt")

If X <> nil Then

Kill "c:\skripsi\program\body.txt"

ElseIf Y <> nil Then

Kill "c:\skripsi\program\body1.txt"

Else

End If

acad_show

AppActivate "AutoCAD"

SendKeys "{esc}"

SendKeys "proses" & "{enter}", True

SendKeys "(buka)" & "{enter}", True

st = Timer

Do While Timer > st - 1

X = Dir("c:\skripsi\program\body.txt")

Y = Dir("c:\skripsi\program\body1.txt")

If X <> nil Or Y <> nil Then

st = Timer + Timer

Else

End If

Loop

acad_visible

AppActivate Form6.Caption

If X <> nil Then

Kill "c:\skripsi\program\body.txt"

ElseIf Y <> nil Then

Kill "c:\skripsi\program\body1.txt"

Unload format: Unload frmSplash: Unload

Form6: Unload Ukuran: Unload Plan

Else

End If

Form6!Toolbar1.Buttons("data").Enabled =

True

Form6!Toolbar1.Buttons("v3d").Enabled =

True

Form6!Toolbar1.Buttons("hasil").Enabled =

True

Form6!Toolbar1.Buttons("potong").Enabled =

True

Form6!Toolbar1.Buttons("body").Enabled =

True

Form6!Toolbar1.Buttons("rambu").Enabled =

True

Form6!Toolbar1.Buttons("export").Enabled =

True

Form6!mnuPlan.Enabled = True

Form6!mnuPelat.Enabled = True

Form6!mnuBentuk.Enabled = True

Form6!mnuOutput.Enabled = True

Form6!mnuAnimasi.Enabled = True

Form6!mnuDimensi.Enabled = True

Form6!mnuDXF.Enabled = True

MousePointer = 0

End Sub

Private Sub mnuSave_Click()

MousePointer = 11

If Dir(filename) <> "" Then

response = MsgBox("Disimpan ditempat yang sama?", vbYesNo + vbQuestion + vbDefaultButton2)

If response = vbNo Then

Simpan_as

If format!Text20.Text <> "" Then

nama

End If

Else

simpan (filename): nama

If format!Text20.Text <> "" Then

nama

End If

End If

End If

MousePointer = 0

Form6.Show

End Sub

Private Sub mnuStatusBar_Click()

If mnuStatusBar.Checked Then

sbStatusBar.Visible = False

mnuStatusBar.Checked = False

Else

sbStatusBar.Visible = True

mnuStatusBar.Checked = True

End If

End Sub

Private Sub mnuTabel_Click()

format.Show

End Sub

Public Sub Automation()

On Error Resume Next

SetacadApp=GetObject("AutoCAD.Application")

If Err Then

Err.Clear

SetacadApp=CreateObject("AutoCAD.Application")

If Err Then

MsgBox Err.Description

Exit Sub

End If

End If

Set acadDoc = acadApp.ActiveDocument

If Not acadDoc.Saved Then

If MsgBox("OK to save drawing?", 4) = vbNo Then

GoTo SKIPNEW

Else

acadDoc.Save

End If

End If

```

Set acadDoc = acadDoc.New("acad")
SKIPNEW:
acadApp.Top = 0
acadApp.Left = 0
acadApp.Width = 400
acadApp.Height = 400
End Sub

Public Sub mtukuran()
Ukuran!Grid1.Rows = 4
Ukuran!Grid1.Cols = format!Text6.Text
For j = 1 To Val(format!Text6.Text - 1)
For i = 1 To 3
Ukuran!Grid1.Col = j: Ukuran!Grid1.Row = i
Ukuran!Grid1.FixedAlignment(j) = 2
Ukuran!Grid1.ColWidth(j) = 1200
Ukuran!Grid1.ColAlignment(j) = 1
Next
Next

Ukuran!Grid2.Rows = 4
Ukuran!Grid2.Cols = format!Text6.Text + 1
For j = 1 To Val(format!Text6.Text)
For i = 1 To 3
Ukuran!Grid2.Col = j: Ukuran!Grid2.Row = i
Ukuran!Grid2.FixedAlignment(j) = 2
Ukuran!Grid2.ColWidth(j) = 1200
Ukuran!Grid2.ColAlignment(j) = 1
Next
Next

Ukuran!Grid1.Col = 0
Ukuran!Grid1.FixedAlignment(0) = 2
Ukuran!Grid1.ColWidth(0) = 800
Ukuran!Grid1.RowHeight(0) = 250
Ukuran!Grid1.RowHeight(1) = 250
Ukuran!Grid1.RowHeight(2) = 250
Ukuran!Grid1.ColWidth(0) = 1600

Ukuran!Grid2.Col = 0
Ukuran!Grid2.FixedAlignment(0) = 2
Ukuran!Grid2.ColWidth(0) = 800
Ukuran!Grid2.RowHeight(0) = 250
Ukuran!Grid2.RowHeight(1) = 250
Ukuran!Grid1.RowHeight(2) = 250
Ukuran!Grid2.ColWidth(0) = 1600

Ukuran!Grid1.Row = 1
Ukuran!Grid1.Text = "Upper Seam [A]"
Ukuran!Grid1.Row = 2
Ukuran!Grid1.Text = "Lower Seam [B]"

Ukuran!Grid2.Row = 1
Ukuran!Grid2.Text = "Length Of Curve [C]"

nmg = format!Text8.Text
For j = 1 To Val(format!Text6.Text - 1)
Ukuran!Grid1.Col = j: Ukuran!Grid1.Row = 0
If j = 1 Then

```

```

Ukuran!Grid1.Text = "AB" & "" & "-" & "Fr."
& Val(nmg) + j - 1
ElseIf j = Val(format!Text6.Text - 1) Then
Ukuran!Grid1.Text = "Fr." & Val(nmg) + j - 2
& "" & "-" & "" & "FB"
Else
Ukuran!Grid1.Text = "Fr." & Val(nmg) + j - 2
& "" & "-" & "" & "Fr." & Val(nmg) + j - 1
End If
Next

nmg = format!Text8.Text
For j = 1 To Val(format!Text6.Text)
Ukuran!Grid2.Col = j: Ukuran!Grid2.Row = 0
If j = 1 Then
Ukuran!Grid2.Text = "AB"
ElseIf j = format!Text6.Text Then
Ukuran!Grid2.Text = "FB"
Else
Ukuran!Grid2.Text = "Fr." & Val(nmg + j - 2)
End If
Next
End Sub

Public Sub inputdata()
'Dim no As Single
Dim dtt(1000), dtt1(1000), dtt2(1000),
dtt3(1000), dtt4(1000)

zz = format!Text16.Text
If Dir("c:\skripsi\program\unnamed.dhb") <>
nil Then
f = FreeFile
Open "c:\skripsi\program\unnamed.dhb" For
Input As f
ElseIf Dir(format!Text16.Text) <> nil Then
f = FreeFile
Open Dir(zz) For Input As f
Else
End If

Do Until EOF(f)
Input #f, jml
Ukuran!Text6.Text = jml
Input #f, pp
Ukuran!Text1.Text = pp
Input #f, pp1
Ukuran!Text2.Text = pp1
Input #f, pp2
Ukuran!Text3.Text = pp2
Input #f, pp3
Ukuran!Text4.Text = pp3
Input #f, Lp
For j = 1 To Val(Ukuran!Text6.Text - 1)
Input #f, dtt(j)
Ukuran!Grid1.Col = j: Ukuran!Grid1.Row = 1
Ukuran!Grid1.Text = dtt(j)
Next

Input #f, Lp0

```

```

For j = 1 To Val(Ukuran!Text6.Text - 1)
Input #f, dtt1(j)
Ukuran!Grid1.Col = j: Ukuran!Grid1.Row = 2
Ukuran!Grid1.Text = dtt1(j)
Next

```

```

Input #f, Lp2
For j = 1 To Val(Ukuran!Text6.Text)
Input #f, dtt3(j)
Ukuran!Grid2.Col = j: Ukuran!Grid2.Row = 1
Ukuran!Grid2.Text = dtt3(j)
Next

```

```

Loop
Close f

```

```

If format!Text6.Text <> Ukuran!Text6.Text
Then

```

```

Ukuran!Grid1.Cols = Val(Ukuran!Text6.Text)
Ukuran!Grid2.Cols = Val(Ukuran!Text6.Text)
+ 1
nmg = format!Text8.Text
For j = 1 To Val(Ukuran!Text6.Text - 1)
Ukuran!Grid1.Col = j: Ukuran!Grid1.Row = 0
If j = 1 Then
Ukuran!Grid1.Text = "AB" & "" & "-" & "Fr."
& Val(nmg) + j - 1
ElseIf j = Val(Ukuran!Text6.Text - 1) Then
Ukuran!Grid1.Text = "Fr." & Val(nmg) + j - 2
& "" & "-" & "" & "FB"
Else
Ukuran!Grid1.Text = "Fr." & Val(nmg) + j - 2
& "" & "-" & "" & "Fr." & Val(nmg) + j - 1
End If
Next

```

```

nmg = format!Text8.Text
For j = 1 To Val(Ukuran!Text6.Text)
Ukuran!Grid2.Col = j: Ukuran!Grid2.Row = 0
If j = 1 Then
Ukuran!Grid2.Text = "AB"
ElseIf j = Ukuran!Text6.Text Then
Ukuran!Grid2.Text = "FB"
Else
Ukuran!Grid2.Text = "Fr." & Val(nmg + j - 2)
End If
Next
Else
End If
End Sub

```

```

Private Sub mnuToolBar_Click()
If mnuToolBar.Checked Then
ToolBar1.Visible = False
mnuToolBar.Checked = False
Else
ToolBar1.Visible = True
mnuToolBar.Checked = True
End If

```

```

End Sub

```

```

Private Sub Toolbar1_ButtonClick(ByVal
Button As ComctlLib.Button)
Select Case Button.Key
Case "baru"
mnuNew_Click
Case "buka"
mnuOpen_Click
Case "simpan"
mnuSave_Click
Case "body"
mnuPlan_Click
Case "tabel"
mnuTabel_Click
Case "proses"
mnuRun_Click
Case "potong"
mnuPelat_Click
Case "hasil"
mnuBentuk_Click
Case "rambu"
mnuOutput_Click
Case "v3d"
mnuAnimasi_Click
Case "data"
mnuDimensi_Click
Case "export"
mnuDXF_Click
Case "help"
mnuRed_Click
Case "keluar"
mnuExit_Click
End Select
End Sub

```

```

Private Sub nama()
Dim c(10000)
a = filename
b = Len(a)

```

```

For i = 1 To b
If f <> e Then
c(i) = Right(a, 1)
a = Left(a, Val(b) - i)
If c(i) = "\" Then
e = i - 1
f = e
End If
End If
Next

```

```

g = Right(filename, e)
H = Left(filename, b - e) & Left(g, Len(g) - 3)
& "dhh"
nfill = Left(filename, b - e) & Left(g, Len(g) -
4)
If Dir("c:\skripsi\program\unnamed.dhb") <>
nil Then

```

```

format!Text15.Text = H
If Dir(H) <> nil Then
Kill H
Else
Name "c:\skripsi\program\unnamed.dhb" As H
End If
End If

If Dir("c:\skripsi\program\has1.dwg") <> nil
Then
format!Text16.Text = nfil1 & "1" & ".dwg"
If Dir(format!Text16.Text) <> nil Then
Kill Dir(format!Text16.Text)
Else
Dir ("c:\skripsi\program\has1.dwg")
Name "c:\skripsi\program\has1.dwg" As nfil1
& "1" & ".dwg"
End If
End If

If Dir("c:\skripsi\program\has2.dwg") <> nil
Then
format!Text17.Text = nfil1 & "2" & ".dwg"
If Dir(format!Text17.Text) <> nil Then
Kill format!Text17.Text
Else
Name "c:\skripsi\program\has2.dwg" As nfil1
& "2" & ".dwg"
End If
End If

If Dir("c:\skripsi\program\has3.dwg") <> nil
Then
format!Text18.Text = nfil1 & "3" & ".dwg"
If Dir(format!Text18.Text) <> nil Then
Kill format!Text18.Text
Else
Name "c:\skripsi\program\has3.dwg" As nfil1
& "3" & ".dwg"

End If
End If
MousePointer = 0
End Sub

Private Sub simpan(filename As String)
If format!Text1.Text = "" Or format!Text2.Text
= "" Or format!Text3.Text = "" Or
format!Text4.Text = "" Or format!Text5.Text
= "" Then
MsgBox "Data Tidak Lengkap!", 0 + 48,
"Pesan"
Exit Sub
ElseIf IsNumeric(format!Text1.Text) = False
Or IsNumeric(format!Text2.Text) = False Or
IsNumeric(format!Text3.Text) = False Or
IsNumeric(format!Text4.Text) = False Or
IsNumeric(format!Text5.Text) = False Then
MsgBox "Isilah Dengan Angka !", 0 + 48,
"Pesan"

```

```

Exit Sub
End If
MousePointer = 11

If format!Text20 <> nil Then
filename = format!Text20.Text
Else
Unload format: Unload frmSplash: Unload
Form6: Unload Ukuran
format.Show: Text1.SetFocus
End If

f = FreeFile
Open filename For Output As f
'main
Write#f, Val(format!Text1.Text), Val(format!Te
xt2.Text), Val(format!Text3.Text)
Write#f, Val(format!Text4.Text), Val(format!Te
xt5.Text)

'data koordinat gading
Write#f, Val(format!Text6.Text), Val(format!Te
xt7.Text)
Write#f, Val(format!Text9.Text), Val(format!Te
xt10.Text)
Print #f, format!Text8.Text

If format!Check1.Value = 1 Then
Write #f, 1
ElseIf format!Check1.Value = 0 Then
Write #f, 0
Else
End If

If format!Check2.Value = 1 Then
Write #f, 1
ElseIf format!Check2.Value = 0 Then
Write #f, 0
Else
End If

If format!Check3.Value = 1 Then
Write #f, 1
ElseIf format!Check3.Value = 0 Then
Write #f, 0
Else
End If

If format!Check4.Value = 1 Then
Write #f, 1
ElseIf format!Check4.Value = 0 Then
Write #f, 0
Else
End If

If format!Check5.Value = 1 Then
Write #f, 1
ElseIf format!Check5.Value = 0 Then
Write #f, 0
Else

```

End If

```
If format!Check6.Value = 1 Then
Write #f, 1
ElseIf format!Check6.Value = 0 Then
Write #f, 0
Else
End If
```

```
If format!Check7.Value = 1 Then
Write #f, 1
ElseIf format!Check7.Value = 0 Then
Write #f, 0
Else
End If
```

```
If format!Check8.Value = 1 Then
Write #f, 1
ElseIf format!Check8.Value = 0 Then
Write #f, 0
Else
End If
```

```
If format!Check9.Value = 1 Then
Write #f, 1
ElseIf format!Check9.Value = 0 Then
Write #f, 0
Else
End If
```

Print #f, format!Text21.Text

```
For i = 1 To format!Text6.Text
For j = 1 To format!Text7.Text
format!Grid1.Row = j: format!Grid1.Col = i
If format!Grid1.Text = "" Then
isi = 0
ElseIf format!Grid1.Text <> "" And
format!Grid1.Text <> "-" Then
isi = Val(format!Grid1.Text)
ElseIf format!Grid1.Text = "-" Then
isi = "-"
End If
Write #f, isi
Next
Next
```

```
For i = 1 To format!Text6.Text
For j = 1 To format!Text7.Text
format!Grid2.Row = j: format!Grid2.Col = i
If format!Grid2.Text = "" Then
isi1 = 0
ElseIf format!Grid2.Text <> "" And
format!Grid2.Text <> "-" Then
isi1 = Val(format!Grid2.Text)
ElseIf format!Grid2.Text = "-" Then
isi1 = "-"
End If
Write #f, isi1
Next
```

Next

```
'data koordinat garis las
For i = 1 To 2
For j = 1 To format!Text6.Text
format!Grid3.Row = j: format!Grid3.Col = i
If format!Grid3.Text = "" Then
isi2 = 0
ElseIf format!Grid3.Text <> "" And
format!Grid3.Text <> "-" Then
isi2 = Val(format!Grid3.Text)
ElseIf format!Grid3.Text = "-" Then
isi2 = "-"
End If
Write #f, isi2
Next
Next
```

```
For i = 1 To 2
For j = 1 To format!Text6.Text
format!Grid4.Row = j: format!Grid4.Col = i
If format!Grid4.Text = "" Then
isi3 = 0
ElseIf format!Grid4.Text <> "" And
format!Grid4.Text <> "-" Then
isi3 = Val(format!Grid4.Text)
ElseIf format!Grid4.Text = "-" Then
isi3 = "-"
End If
Write #f, isi3
Next
Next
Close #f
filename = "Unnamed"
End Sub
```

```
Private Sub bersih()
Screen.MousePointer = 13
Automation
If Dir("c:\skripsi\program\has1.dwg") <> nil
Then
Kill Dir("c:\skripsi\program\has1.dwg")
End If
If Dir("c:\skripsi\program\has2.dwg") <> nil
Then
Kill Dir("c:\skripsi\program\has2.dwg")
End If
If Dir("c:\skripsi\program\has3.dwg") <> nil
Then
Kill Dir("c:\skripsi\program\has3.dwg")
End If
If Dir("c:\skripsi\program\bukaan.dxf") <> nil
Then
Kill Dir("c:\skripsi\program\bukaan.dxf")
End If
If Dir("c:\skripsi\program\has5.dwg") <> nil
Then
Kill Dir("c:\skripsi\program\has5.dwg")
End If
```

```

If Dir("c:\skripsi\program\unnamed.dhb") <>
nil Then
Kill ("c:\skripsi\program\unnamed.dhb")
End If
Screen.MousePointer = 0
End Sub

```

```

Private Sub hapusfile()
If Dir(CurDir & "\body.a$t") <> nil Then
Kill CurDir & "\body.a$t"
End If
End Sub

```

```

Private Sub Simpan_as()
Dim filename As String

```

```

    With CmDialog1
        .DialogTitle = "Simpan File"
        .Filter = "Bukaan Files (*.dat)|*.dat|"
        .ShowSave
        If Len(.filename) = 0 Then
            Exit Sub
        End If
        filename = .filename
    End With
    simpan (filename)
End Sub

```

Lampiran B.2.23

Listing Form Tabel Data Koordinat :

```

Private Sub Command1_Click()
Dim Counter As Integer
    Dim Workarea(250) As String
    ProgressBar1.Min = LBound(Workarea)
    ProgressBar1.Max = UBound(Workarea)
    ProgressBar1.Visible = True

```

```

'Set the Progress's Value to Min.
    ProgressBar1.Value = ProgressBar1.Min

```

```

'Loop through the array.
    For Counter = LBound(Workarea) To
UBound(Workarea)
        'Set initial values for each item in the
array.
        Workarea(Counter) = "Initial value" &
Counter
        ProgressBar1.Value = Counter

```

```

Next Counter
    ProgressBar1.Visible = False
    ProgressBar1.Value = ProgressBar1.Min
CmDialog1.HelpFile =
"C:\DATA\HC\ICONWRKS.HLP"
CmDialog1.HelpCommand = cdHelpContents
CmDialog1.ShowHelp
End Sub

```

```

Private Sub Command2_Click()

```

```

Dim Counter As Integer
    Dim Workarea(250) As String
    ProgressBar1.Min = LBound(Workarea)
    ProgressBar1.Max = UBound(Workarea)
    ProgressBar1.Visible = True

```

```

'Set the Progress's Value to Min.
    ProgressBar1.Value = ProgressBar1.Min

```

```

'Loop through the array
ForCounter=LBound(Workarea)ToUBound(W
orkarea)
'Set initial values for each item in the array.
Workarea(Counter) = "Initial value" &
Counter
ProgressBar1.Value = Counter
Next Counter
    ProgressBar1.Visible = False
    ProgressBar1.Value = ProgressBar1.Min
    Plan.Show: format.Hide
End Sub

```

```

Private Sub Command4_Click()
Form6.Show: format.Hide
End Sub

```

```

Private Sub Form_Load()
Top = Screen.Height / 2 - Height / 2
Left = Screen.Width / 2 - Width / 2
End Sub

```

```

Private Sub Grid1_Click()
Dim KeyAscii As Integer
Text11.SetFocus
SendKeys "{END}"
KeyAscii = 13
End Sub

```

```

Private Sub Grid1_KeyDown(KeyCode As
Integer, Shift As Integer)
Select Case KeyCode
Case &H71 'F2
    Text11.SetFocus
    SendKeys "{END}"
End Select
End Sub

```

```

Private Sub Grid1_KeyPress(KeyAscii As
Integer)
Text11.SetFocus
SendKeys Chr$(KeyAscii)
End Sub

```

```

Private Sub Grid1_SelChange()
Text11.Text = Grid1.Text
End Sub

```

```

Private Sub Grid2_Click()
Dim KeyAscii As Integer
Text13.SetFocus

```

```
SendKeys "{END}"
KeyAscii = 13
End Sub
```

```
Private Sub Grid2_KeyDown(KeyCode As Integer, Shift As Integer)
Select Case KeyCode
Case &H71 'F2
Text13.SetFocus
SendKeys "{END}"
End Select
End Sub
```

```
Private Sub Grid2_KeyPress(KeyAscii As Integer)
Text13.SetFocus
SendKeys Chr$(KeyAscii)
End Sub
```

```
Private Sub Grid2_RowColChange()
Text13.Text = Grid2.Text
End Sub
```

```
Private Sub Grid3_Click()
Dim KeyAscii As Integer
Text14.SetFocus
SendKeys "{END}"
KeyAscii = 13
End Sub
```

```
Private Sub Grid3_KeyDown(KeyCode As Integer, Shift As Integer)
Select Case KeyCode
Case &H71 'F2
Text14.SetFocus
SendKeys "{END}"
End Select
End Sub
```

```
Private Sub Grid3_KeyPress(KeyAscii As Integer)
Text14.SetFocus
SendKeys Chr$(KeyAscii)
End Sub
```

```
Private Sub Grid3_RowColChange()
Text14.Text = Grid3.Text
End Sub
```

```
Private Sub Grid4_Click()
Dim KeyAscii As Integer
Text12.SetFocus
SendKeys "{END}"
KeyAscii = 13
End Sub
```

```
Private Sub Grid4_KeyDown(KeyCode As Integer, Shift As Integer)
Select Case KeyCode
Case &H71 'F2
```

```
Text12.SetFocus
SendKeys "{END}"
End Select
End Sub
```

```
Private Sub Grid4_KeyPress(KeyAscii As Integer)
Text12.SetFocus
SendKeys Chr$(KeyAscii)
End Sub
```

```
Private Sub Grid4_RowColChange()
Text12.Text = Grid4.Text
End Sub
```

```
Private Sub mtabel()
Dim i, j As Integer
```

```
format!Grid1.Cols = Text6.Text + 1
format!Grid1.Rows = Text7.Text + 1
```

```
format!Grid2.Cols = Text6.Text + 1
format!Grid2.Rows = Text7.Text + 1
```

```
format!Grid3.Cols = 3
format!Grid3.Rows = Text6.Text + 1
```

```
format!Grid4.Cols = 3
format!Grid4.Rows = Text6.Text + 1
```

```
For j = 1 To Val(Text6.Text)
For i = 1 To Val(Text6.Text)
format!Grid1.Col = j: format!Grid1.Row = i
format!Grid1.FixedAlignment(j) = 2
format!Grid1.ColWidth(j) = 1000
format!Grid1.ColAlignment(j) = 1
Next
Next
```

```
For j = 1 To Val(Text6.Text)
For i = 1 To Val(Text7.Text)
format!Grid2.Col = j: format!Grid2.Row = i
format!Grid2.FixedAlignment(j) = 2
format!Grid2.ColWidth(j) = 1000
format!Grid2.ColAlignment(j) = 1
Next
Next
```

```
For j = 1 To Val(Text6.Text)
For i = 1 To 2
format!Grid3.Col = i: format!Grid3.Row = j
format!Grid3.FixedAlignment(i) = 2
format!Grid3.ColWidth(i) = 1000
format!Grid3.ColAlignment(i) = 1
Next
Next
```



```

For j = 1 To Val(Text6.Text)
For i = 1 To 2
format!Grid4.Col = i: format!Grid4.Row = j
format!Grid4.FixedAlignment(i) = 2
format!Grid4.ColWidth(i) = 1000
format!Grid4.ColAlignment(i) = 1
Next
Next

```

```

format!Grid1.Col = 0
format!Grid1.FixedAlignment(0) = 2
format!Grid1.ColWidth(0) = 800

```

```

format!Grid2.Col = 0
format!Grid2.FixedAlignment(0) = 2
format!Grid2.ColWidth(0) = 800

```

```

format!Grid3.Col = 0
format!Grid3.FixedAlignment(0) = 2
format!Grid3.ColWidth(0) = 800

```

```

format!Grid4.Col = 0
format!Grid4.FixedAlignment(0) = 2
format!Grid4.ColWidth(0) = 800

```

```

For j = 1 To Val(Text7.Text)
format!Grid1.Col = 0: format!Grid1.Row = j
format!Grid1.Text = "WL" & " " & j
Next

```

```

nmg = Text8.Text
For j = 1 To Val(Text6.Text)
format!Grid1.Col = j: format!Grid1.Row = 0
If j = 1 Then
format!Grid1.Text = "After Butt"
ElseIf j = Text6.Text Then
format!Grid1.Text = "Fore Butt"
Else
format!Grid1.Text = "Gading" & " " &
Val(nmg) + j - 2
End If
Next

```

```

For j = 1 To Val(Text7.Text)
format!Grid2.Col = 0: format!Grid2.Row = j
format!Grid2.Text = "WL" & " " & j
Next

```

```

nmg = Text8.Text
For j = 1 To Val(Text6.Text)
format!Grid2.Col = j: format!Grid2.Row = 0
If j = 1 Then
format!Grid2.Text = "After Butt"
ElseIf j = Text6.Text Then
format!Grid2.Text = "Fore Butt"
Else
format!Grid2.Text = "Gading" & " " &
Val(nmg) + j - 2
End If
Next

```

```

nmg = Text8.Text
For j = 1 To Val(Text6.Text)
format!Grid3.Col = 0: format!Grid3.Row = j
If j = 1 Then
format!Grid3.Text = "After Butt"
ElseIf j = Text6.Text Then
format!Grid3.Text = "Fore Butt"
Else
format!Grid3.Text = "Gd." & " " & Val(nmg) +
j - 2
End If
Next

```

```

format!Grid3.Col = 1: format!Grid3.Row = 0
format!Grid3.Text = "Lower Seam"
format!Grid3.Col = 2: format!Grid3.Row = 0
format!Grid3.Text = "Upper Seam"

```

```

nmg = Text8.Text
For j = 1 To Val(Text6.Text)
format!Grid4.Col = 0: format!Grid4.Row = j
If j = 1 Then
format!Grid4.Text = "After Butt"
ElseIf j = Text6.Text Then
format!Grid4.Text = "Fore Butt"
Else
format!Grid4.Text = "Gd." & " " & Val(nmg) +
j - 2
End If
Next

```

```

format!Grid4.Col = 1: format!Grid4.Row = 0
format!Grid4.Text = "Lower Seam"
format!Grid4.Col = 2: format!Grid4.Row = 0
format!Grid4.Text = "Upper Seam"

```

```

If Text6.Text <= 4 Then
format!Grid1.ScrollBars = 1
format!Grid2.ScrollBars = 1
format!Grid3.ScrollBars = 1
format!Grid4.ScrollBars = 1
Else
format!Grid1.ScrollBars = 3
format!Grid2.ScrollBars = 3
format!Grid3.ScrollBars = 3
format!Grid4.ScrollBars = 3
End If

```

```

format!Grid1.Row = 1: format!Grid1.Col = 1
format!Grid2.Row = 1: format!Grid2.Col = 1
format!Grid3.Row = 1: format!Grid3.Col = 1
format!Grid4.Row = 1: format!Grid4.Col = 1
End Sub

```

```

Private Sub tabel_Click()
If Text6.Text = "" Then
MsgBox "Isilah Jumlah Gading !", 0 + 48,
"Pesan"
Exit Sub

```

```

ElseIf IsNumeric(format!Text6.Text) = False
Then
MsgBox "Isilah Dengan Angka !", 0 + 48,
"Pesan"
Exit Sub
Else
End If

```

```

If Text7.Text = "" Then
MsgBox "Isilah Jumlah Titik !", 0 + 48,
"Pesan"
Exit Sub
ElseIf IsNumeric(Text7.Text) = False Then
MsgBox "Isilah Dengan Angka !", 0 + 48,
"Pesan"
Exit Sub
Else
End If

```

```

If Text9.Text = "" Or Text8.Text = "" Or
Text10.Text = "" Then
MsgBox "Data Tidak Lengkap!", 0 + 48,
"Pesan"
Exit Sub
ElseIf IsNumeric(Text9.Text) = False Then
MsgBox "Isilah Dengan Angka !", 0 + 48,
"Pesan"
Exit Sub
Else
End If

```

```

If Text6.Text < 3 Then
MsgBox "Jumlah Gading Minimal 3 !", 0 + 48,
"Pesan"
Exit Sub
End If

```

```

If Text7.Text < 3 Then
MsgBox "Jumlah Titik Minimal 3 !", 0 + 48,
"Pesan"
Exit Sub
End If
mtablel
End Sub

```

Lampiran B.2.24

Listing Form Dimensi Sebenarnya:

```

Private Sub Command1_Click()
Ukuran.Hide: Form6.Show
End Sub

```

```

Private Sub Command2_Click()
Form3.Show
Ukuran.Hide
End Sub

```

```

Private Sub Form_Load()
Top = Screen.Height / 2 - Height / 2
Left = Screen.Width / 2 - Width / 2
Text1.Locked = True

```

```

Text2.Locked = True
Text3.Locked = True
Text4.Locked = True
End Sub

```

```

Private Sub Form_Unload(Cancel As Integer)
Unload Form6: Unload format: Unload
frmSplash: Unload Ukuran
End Sub

```

Lampiran B.2.25

Listing Otomatisasi AutoCAD pada layar Visual Basic (dalam bentuk Modul.Bas) :

```

Sub Main()
vv = 0
Screen.MousePointer = 11
frmSplash.Show
pt = 1
st = Timer
Do While Timer < st + pt
DoEvents
Loop
Screen.MousePointer = 11
st = Dir("c:\progra~1\autoca~1\indi.txt")
If st <> nil Then
Kill "c:\progra~1\autoca~1\indi.txt"
End If
Screen.MousePointer = 11
open_cad
Do While st <> nil
DoEvents
Loop
frmSplash.Refresh
Unload frmSplash
st = Dir("c:\progra~1\autoca~1\indi.txt")
If st <> nil Then
Kill "c:\progra~1\autoca~1\indi.txt"
End If
Form6.Show
AppActivate Form6.Caption
Screen.MousePointer = 0
End Sub
Sub acad_visible()
On Error Resume Next
SetacadApp=GetObject("AutoCAD.Applicatio
n")
acadApp.Visible = False
End Sub
Sub acad_show()
On Error Resume Next
SetacadApp=GetObject("AutoCAD.Applicatio
n")
acadApp.Visible = True
With acadApp
.Top = 66: .Left = 0.9: .Width = 641: .Height =
387
End With
Tcad
End Sub

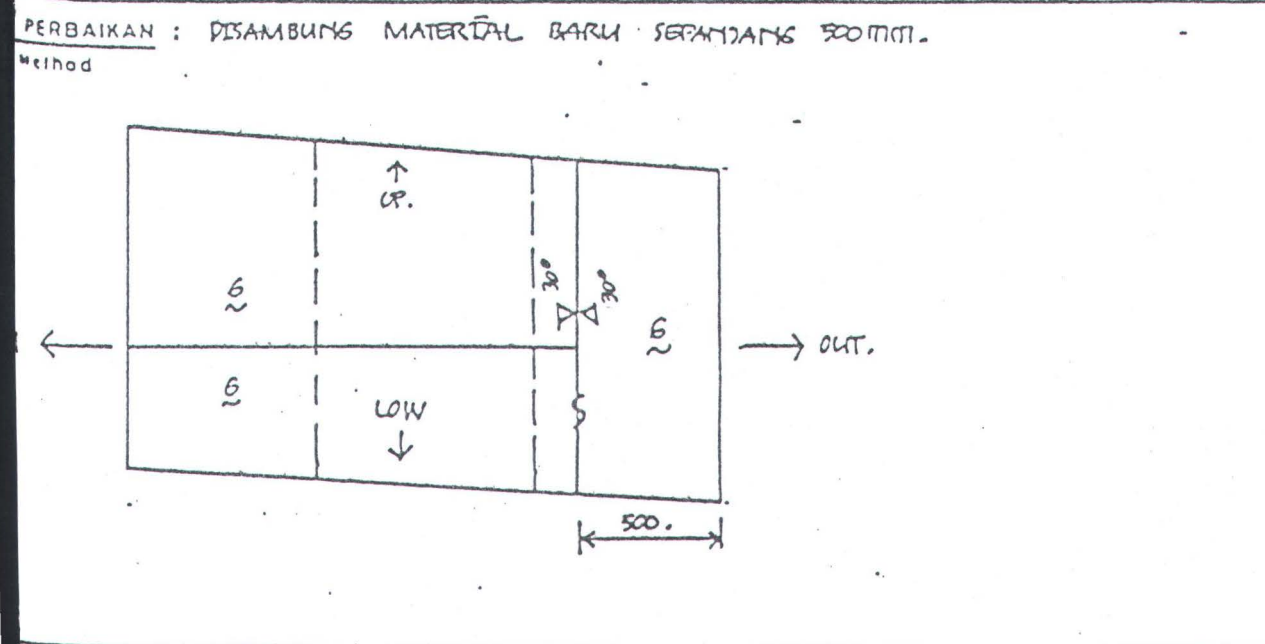
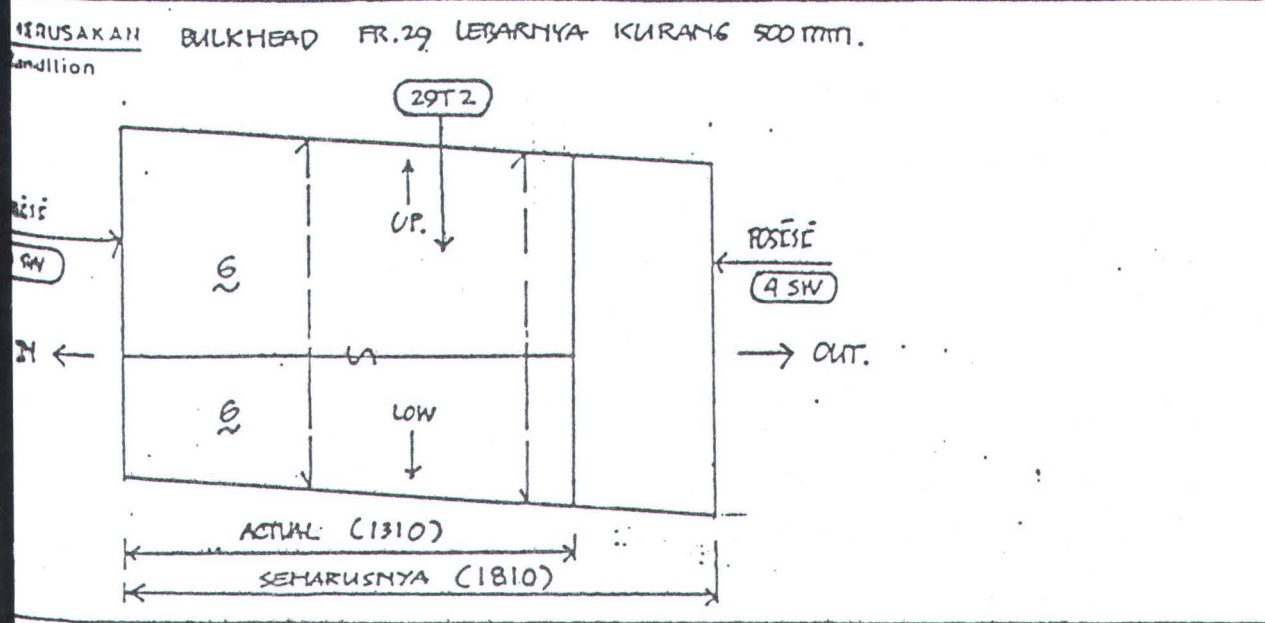
```

```
Sub open_cad()
On Error Resume Next
SetacadApp=GetObject("AutoCAD.Application")
If Err Then
Err.Clear
SetacadApp=CreateObject("AutoCAD.Application")
If Err Then
MsgBox Err.Description
Exit Sub
End If
End If
Tcad
acadApp.Visible = False
End Sub
```

```
Sub Tcad()
Dim acadPref As Object
Set acadPref = acadApp.Preferences
acadPref.CursorSize = 5
acadPref.DisplayScreenMenu = False
acadPref.DisplayScrollBars = False
acadPref.DockedVisibleLines = 1
Set activeDoc = acadApp.ActiveDocument
sysVarName = "pickbox"
Call activeDoc.SetVariable(sysVarName, 1)
End Sub
```

LAMPIRAN C

NO. S.O. : 5090.	NO. ARSIP : 02/070-FBS/03. File No. : 116/12/88	PEMERIKSA : BASDES.P. Inspector
NY	BENGKEL : ASSEMBLY. Shop	TANGGAL : 05/12/1988. Date



TINDAKAN Pencegahan
 and Counter measure

Penyebab :

- DIMENSI PADA MARKING LIST TIDAK SESUAI DG DIMENSI PADA WORKING DRAWING.
- MARKING LIST = 1310 MM (HAL : L 57-14).
- WORKING DRAWING = 1810 MM (HAL : 7/16, 13/16).

Pencegahan :

- PEMERIKSAAN ULANG PADA MARKING LIST SPY LEBIH PENTINGKATKAN, SEBELUM GAMBAR TSB DIDISTRIBUSI.

PAL INDONESIA (PERSERO)
PT. PAL INDONESIA
SHIPBUILDING INDUSTRY

FEED BACK SHEET

UMPMAN BALIK

DIVISI KAPAL NIAGA
DEPT. PENGENDALIAN MUTU
QUALITY CONTROL DEPARTMENT

NO. : 090.

NO. ARSIP : 501/090.F.35/03.
File No. : 17E/12/88

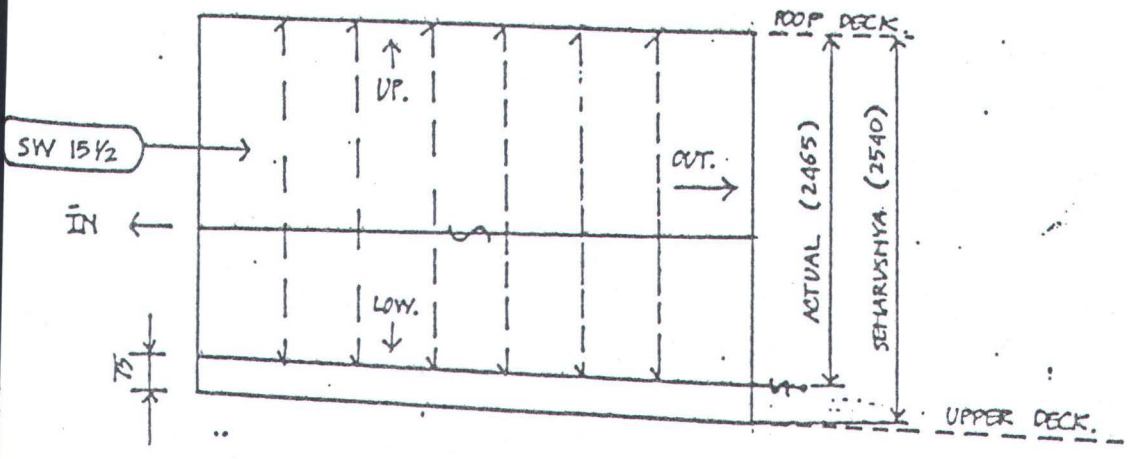
PEMERIKSA : BAGOES.P.
Inspector

1 PPO + PSO/S

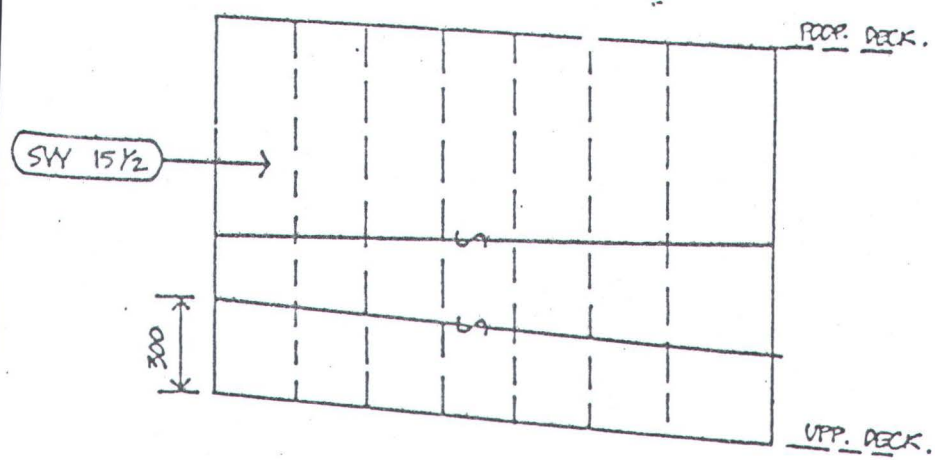
BENGKEL : ASSEMBLY.
Shop

TANGGAL : 29/11/88.
Date

KERUSAKAN BULKHEAD TR.15 1/2 PADA ARAH LOW,
Condition KURANG 75MM.



PERBAIKAN SEBAGIAN DIGANTIKAN BARU (PARTIAL RENEW). = 300MM.
Method



TINDAKAN PENCEGAHAN
Pre and Counter measure

- PENYEBAB : DIMENSI PADA MARKING LIST TIDAK SESUAI DENGAN TINGGI BANGUNAN ATAS.
- MARKING LIST = 2465 mm. (DWG NO : 2050301, HAL : L 53-8).
- TINGGI BANGUNAN ATAS = 2540 mm.
- PENCEGAHAN : PEMERIKSAAN BADA, MARKING LIST SUPAYA LEBIH DIPINGKATKAN.

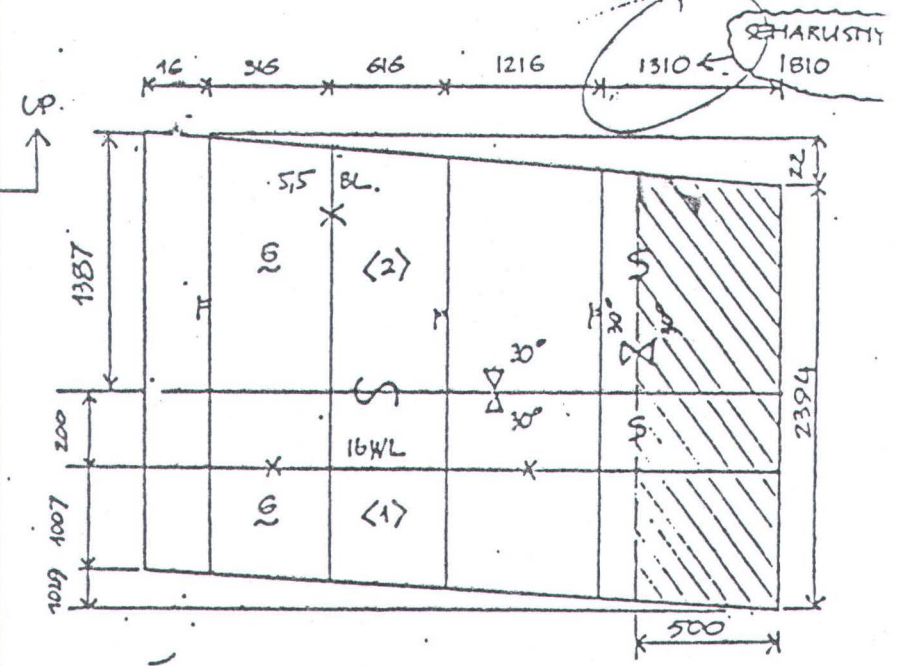
Design (MOLD LOFT)
Production Dept
Inspection Dept
QC Dept

KAPAL INDONESIA (PERSERO) (P.T. PAL INDONESIA)	LAPORAN HASIL PEMERIKSAAN DEP. DALMUTU - DIV. KANIA	PROYEK : CARAKA Project JAYA 3B
SHIPBUILDING INDUSTRY	UNTUK : - KABELAS MARKING ATTN : - KABELAS ASSEMBLY - KABELAS WELDING	TANGGAL : DATE : 05/12/8
NO. : S.090.	NO. ARSIP : 028/090/03-12/17/22 File No.	HAL : 1 dr Page 1 from 1

YANG DIPERIKSA : REKOMENDASI DAN TINDAK LANJUT

ASSEMBLY BLOCK -

IV → BULKHEAD FR. 29 NOTASI (29T2) P/S DIMENSI
LEBAR KURANG 500 MM.
DIMENSI PADA MARKING LIST = 1310 mm...
DIMENSI PADA WORKING DRAWING = 1810 mm



29T2 - 1x2, R 6 EACH P/S Q=4.

→ SELANJUTNYA PELAT TERSEBUT SPY DISAMBUNG SESUAI SKETSA YG DIARSIR ...

FABRIKASI -

→ SUPAYA MENYEDIAKAN MATERIAL PELAT :
2400 X 500 X 6 Q = 2 PCS (P/S).

DISTRIBUSI :
EST. REC.
EST. ERECTION.
EST. LAB. LAMONG.

MENGETAHUI :
KASEP. DALMUTU

[Signature]

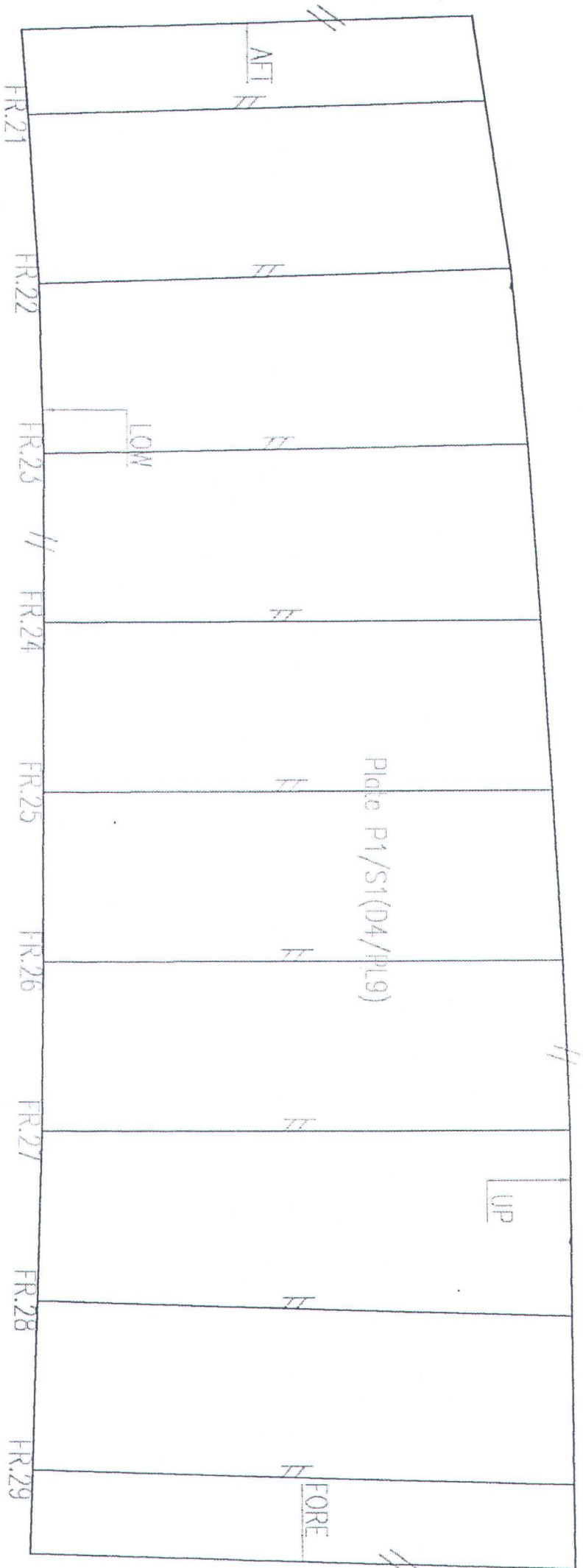
SURVEYOR :

[Signature]

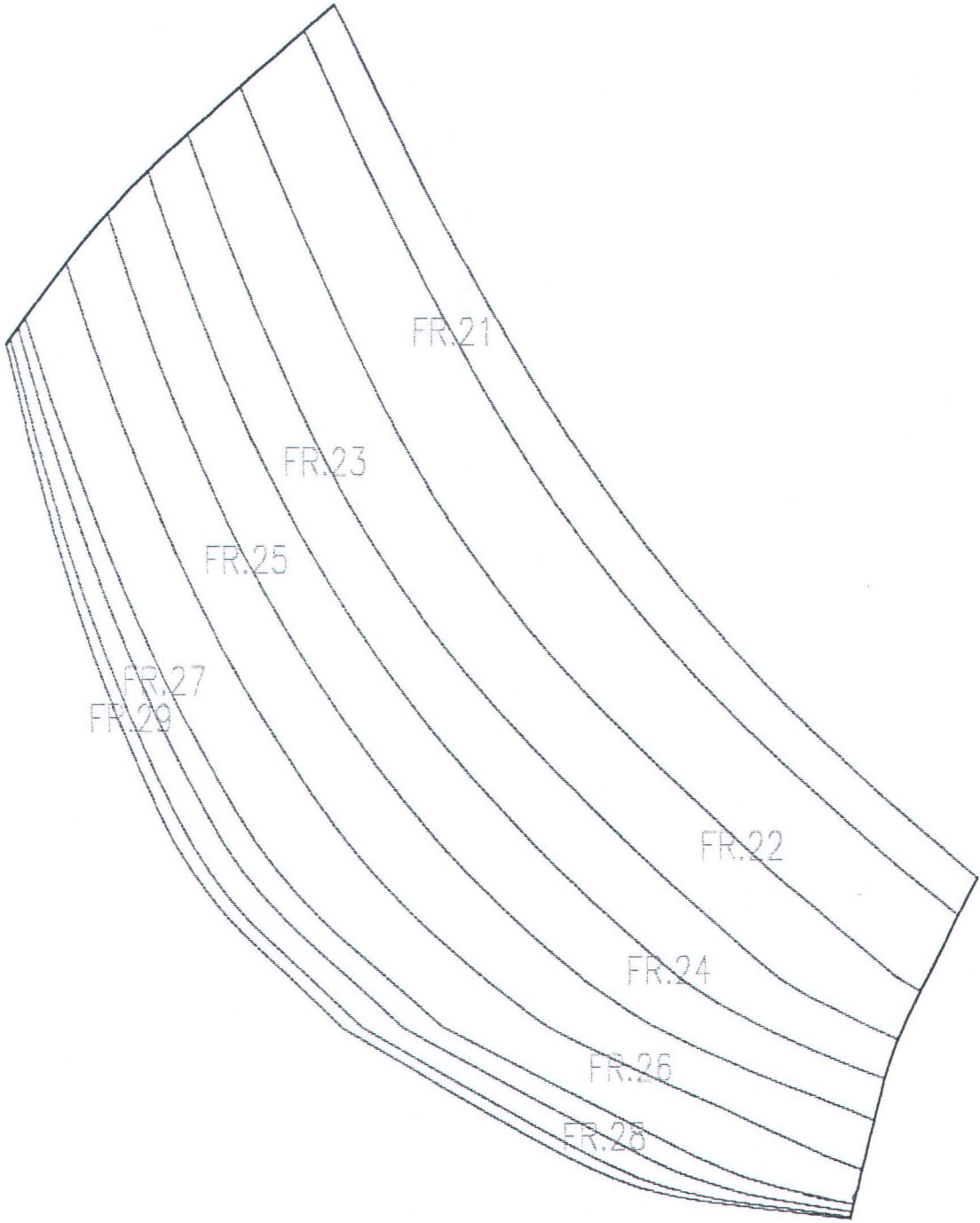
RIKTES PERJALANAN

LAMPIRAN D

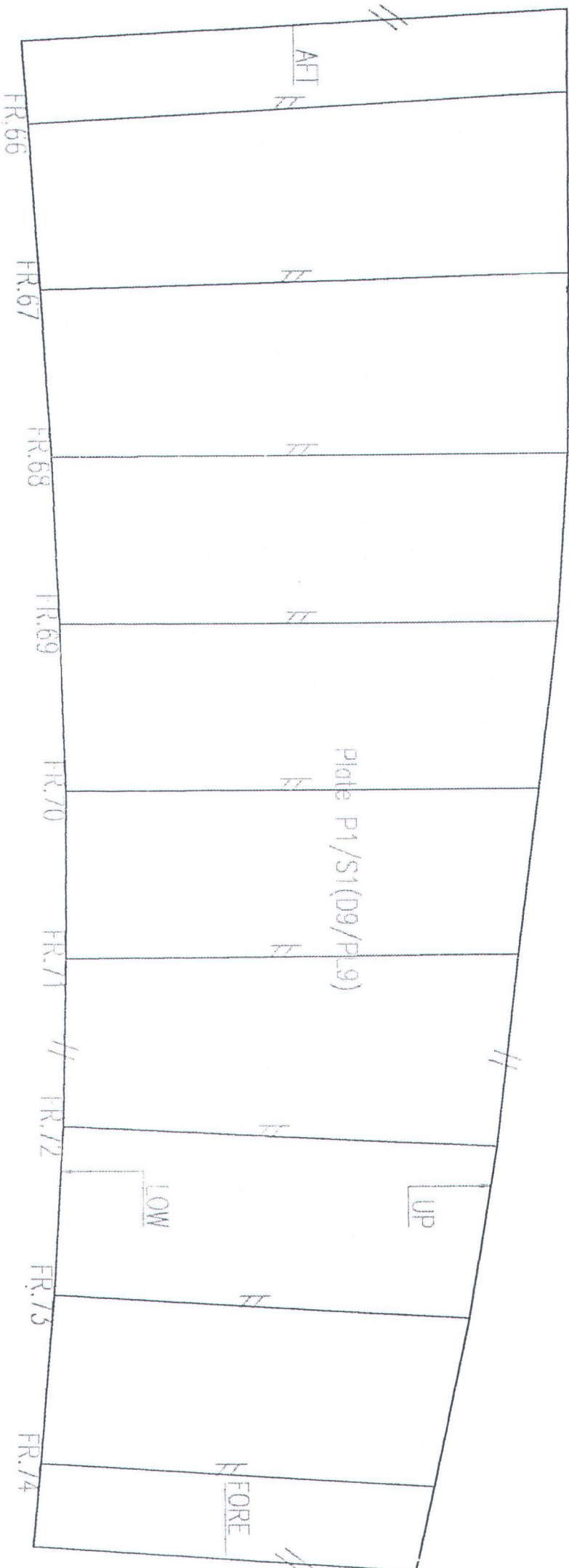
1:20



1:10



1:20



1 : 10

