

SIMULASI PERHITUNGAN GAYA APUNG DAN GAYA TENGGELAM RUMPON LAUT DALAM DI PERAIRAN SELATAN PALABUHANRATU KABUPATEN SUKABUMI

OCE JUNGJUNAN



**MAYOR TEKNOLOGI DAN MANAJEMEN PERIKANAN TANGKAP
DEPARTEMEN PEMANFAATAN SUMBERDAYA PERIKANAN
FAKULTAS PERIKANAN DAN ILMU KELAUTAN
INSTITUT PERTANIAN BOGOR
BOGOR
2009**

PERNYATAAN MENGENAI SKRIPSI DAN SUMBER INFORMASI

Dengan ini saya menyatakan bahwa skripsi Simulasi Perhitungan Gaya Apung dan Gaya Tenggelam Rumpon Laut Dalam di Perairan Selatan Palabuhanratu Kabupaten Sukabumi adalah karya saya sendiri dengan arahan komisi pembimbing dan belum diajukan dalam bentuk apa pun kepada perguruan tinggi mana pun. Sumber informasi yang berasal atau dikutip dari karya yang diterbitkan mau pun tidak diterbitkan dari penulis lain telah disebutkan dalam teks dan tercantum dalam daftar pustaka di bagian akhir skripsi ini.

Bogor, 17 Desember 2009

Oce Jungjunan

ABSTRAK

OCE JUNGJUNAN, C44051637. Simulasi Perhitungan Gaya Apung dan Gaya Tenggelam Rumpon Laut Dalam di Perairan Selatan Palabuhanratu Kabupaten Sukabumi. Dibimbing oleh RONNY IRAWAN WAHJU dan WAZIR MAWARDI.

Rumpon atau *Fish Aggregating Device* (FAD), merupakan alat bantu penangkapan yang berfungsi untuk memikat ikan agar berkumpul dalam wilayah penangkapan di sekitar rumpon tersebut dipasang. Penelitian dan informasi mengenai desain konstruksi rumpon masih terbatas, khususnya rumpon laut dalam di perairan selatan palabuhanratu kabupaten sukabumi. Selain itu, belum ada perangkat lunak yang dapat menghitung nilai *buoyancy force* dari rumpon. Tujuan dari penelitian ini adalah mendeskripsikan rumpon laut dalam, meliputi komponen dan material rumpon, menghitung *buoyancy force*, *sinking force*, dan *extra buoyancy* pada rumpon dengan asumsi arus perairan tidak mempengaruhi, dan membuat suatu perangkat lunak sederhana untuk menghitung nilai *buoyancy force*, *sinking force*, dan *extra buoyancy* dengan menggunakan perangkat lunak *Borland Delphi 7*. Metode penelitian yang digunakan adalah metode deskriptif. Data yang diamati adalah data tebal, panjang, diameter, lebar, tinggi kerucut ponton, dan jumlah. Sedangkan jenis perhitungan yang dilakukan antara lain perhitungan massa, volume, massa jenis, *buoyancy force*, *sinking force*, *total buoyancy force*, *total sinking force*, serta *extra buoyancy*. Rumpon laut dalam di perairan selatan Palabuhanratu memiliki konstruksi yang terdiri dari ponton berlapis fiber dan memiliki 2 buah sekat dan ruang pada ponton diisi *sterofoam*, 4 buah *swivel*, tali utama sepanjang 68 gulung, 9 buah hillban berlapis karung, 21 pemberat utama dengan massa per pemberat mencapai 75 kg, 25 atraktor daun kelapa, dan komponen lain seperti tali rafia, selang plastik, serta waring dan pemberat atraktor berupa sebuah batu bata. Rumpon di perairan Palabuhanratu memiliki nilai *total buoyancy force* sebesar 2.074,87 kg, nilai *total sinking force* sebesar 1.131,03 kg dan nilai *extra buoyancy* sebesar 45,49 % atau 943,84 kg. Hasil perhitungan yang dilakukan secara manual dibandingkan dengan hasil perhitungan menggunakan aplikasi SoFAD (*Software of Fish Aggregating Device*), hasil perhitungan memiliki nilai yang persis sama. Dengan demikian aplikasi SoFAD dapat digunakan untuk menghitung nilai *buoyancy force*, *sinking force*, dan *extra buoyancy* komponen rumpon.

Kata kunci : simulasi, desain konstruksi, rumpon, borland delphi



© Hak cipta IPB, Tahun 2009

Hak cipta dilindungi Undang-undang

- 1) Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan atau menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar IPB.
- 2) Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis dalam bentuk apapun tanpa seizin IPB.

**SIMULASI PERHITUNGAN GAYA APUNG
DAN GAYA TENGGELAM RUMPON LAUT DALAM
DI PERAIRAN SELATAN PALABUHANRATU
KABUPATEN SUKABUMI**

OCE JUNGJUNAN

SKRIPSI

Sebagai syarat untuk mendapatkan gelar Sarjana Perikanan
pada Departemen Pemanfaatan Sumberdaya Perikanan

**MAYOR TEKNOLOGI DAN MANAJEMEN PERIKANAN TANGKAP
DEPARTEMEN PEMANFAATAN SUMBERDAYA PERIKANAN
FAKULTAS PERIKANAN DAN ILMU KELAUTAN
INSTITUT PERTANIAN BOGOR
BOGOR
2009**

Halaman ini adalah milik IPB University. Untuk lebih jelasnya, silakan kunjungi website kami di www.ipb.ac.id.
1. Dilarang menyalin, mengutip, atau menyalin sebagian atau seluruh isi dari dokumen ini tanpa izin tertulis dari IPB University.
2. Dilarang menyalin, mengutip, atau menyalin sebagian atau seluruh isi dari dokumen ini tanpa izin tertulis dari IPB University.

Judul Skripsi : Simulasi Perhitungan Gaya Apung dan Gaya Tenggelam Rumpon Laut Dalam di Perairan Selatan Palabuhanratu Kabupaten Sukabumi

Nama : Oce Jungjunan
NRP : C44051637
Mayor : Teknologi dan Manajemen Perikanan Tangkap

Disetujui :

Pembimbing I,

Pembimbing II,

Ir. Ronny Irawan Wahyu, M. Phil
NIP: 1961 0906 198703 1 002

Ir. Wazir Mawardi, M. Si
NIP: 1965 0625 199103 1 001

Diketahui :

Ketua Departemen Pemanfaatan Sumberdaya Perikanan

Dr. Ir. Budy Wiryawan, M.Sc
NIP: 1962 1223 198703 1 001

Tanggal lulus: 17 Desember 2009

KATA PENGANTAR

Puji syukur penulis panjatkan kepada Allah SWT atas segala karunia-Nya sehingga Skripsi Simulasi Perhitungan Gaya Apung Dan Gaya Tenggelam Rumpon Laut Dalam Di Perairan Selatan Palabuhanratu Kabupaten Sukabumi dapat diselesaikan. Skripsi ini disusun sebagai syarat untuk mendapatkan gelar Sarjana Perikanan pada Mayor Teknologi dan Manajemen Perikanan Tangkap, Departemen Pemanfaatan Sumberdaya Perikanan, Fakultas Perikanan dan Ilmu Kelautan, Institut Pertanian Bogor.

Penulis mengucapkan terima kasih kepada:

1. Ir. Ronny Irawan Wahju, M.Phil dan Ir. Wazir Mawardi, M.Si selaku Komisi Pembimbing;
2. Dr. Ir. Mohammad Imron, M.Si selaku Ketua Komisi Pendidikan;
3. Ir. Zulkarnain, M.Si selaku dosen penguji tamu;
4. Dr. Ir. Budhi Hascaryo Iskandar, M.Si selaku pembimbing akademik;
5. Ir. Arik Permana, Teknisi SLK-FPIK IPB di Palabuhanratu selaku pendamping lapangan;
6. Kru PSP01 yang telah membantu selama pelaksanaan penelitian di atas kapal;
7. Orang tua tercinta atas doa dan bimbingannya yang telah diberikan tanpa henti;
8. Putri Fitrianda, S.Pt yang telah memberikan semangat dari awal hingga akhir penelitian dan selama penulis menyelesaikan studi;
9. Serta pihak-pihak yang telah membantu dalam penulisan skripsi ini yang tidak mungkin dituliskan satu-persatu.

Akhir kata, semoga skripsi ini dapat bermanfaat bagi para pembaca.

Bogor, 17 Desember 2009

Oce Jungjunan

RIWAYAT HIDUP



Penulis dilahirkan di Subang pada tanggal 6 Maret 1987 dari Bapak Edi Slamet Riyadi dan Ibu Cucu Sopirah. Penulis merupakan putra kedua dari dua bersaudara.

Penulis lulus dari SMA Negeri 3 Subang pada tahun 2005 dan pada tahun yang sama lulus seleksi masuk IPB melalui jalur Undangan Seleksi Masuk IPB. Penulis memilih Mayor Teknologi dan Manajemen Perikanan Tangkap, Departemen Pemanfaatan Sumberdaya Perikanan, Fakultas Perikanan dan Ilmu Kelautan, Institut Pertanian Bogor.

Selama mengikuti perkuliahan, penulis menjadi asisten praktikum mata kuliah Tingkah Laku Ikan pada tahun ajaran 2007/2008. Selain itu penulis juga aktif di berbagai organisasi dan kepanitiaan. Diantaranya, pada tahun ajaran 2005/2006 menjadi anggota Divisi Informasi dan Komunikasi DKM Al-Hurriyyah dan staf Divisi Pengembangan Sumberdaya Manusia, Forum Komunikasi Kulawarga Subang (FOKKUS), tahun ajaran 2006/2007 menjadi staf Divisi Informasi dan Komunikasi, Forum Komunikasi Kulawarga Subang (FOKKUS), tahun ajaran 2008/2009 menjadi Staf Divisi Kewirausahaan Himpunan Mahasiswa Pemanfaatan Sumberdaya Perikanan (HIMAFARIN) Departemen Pemanfaatan Sumberdaya Perikanan, Fakultas Perikanan dan Ilmu Kelautan, Institut Pertanian Bogor. Selama masa studi, penulis mendapatkan Beasiswa BBM (Bantuan Belajar Mahasiswa) pada tahun ajaran 2007/2008 dan 2008/2009.

Dalam rangka menyelesaikan tugas akhir, penulis melakukan penelitian dan menyusun skripsi dengan judul ” Simulasi Perhitungan Gaya Apung dan Gaya Tenggelam Rumpon Laut Dalam di Perairan Selatan Palabuhanratu Kabupaten Sukabumi”.

DAFTAR ISTILAH

- .BMP** - Disebut bitmap atau format *file* DIB (*device-independent bitmap*). Adalah format *file* gambar *bitmap* digunakan untuk menyimpan gambar digital, terutama pada sistem operasi.
- .EXE** - Adalah ekstensi nama *file* umum yang menunjukkan file eksekusi (program) di sistem operasi. banyak file *.exe* mengandung komponen lain, seperti *bitmap* dan ikon.
- .ICO** - Format *file* gambar ikon di Microsoft Windows. *.ico file* berisi satu atau lebih gambar kecil pada berbagai ukuran dan kedalaman warna.
- .NET FRAMEWORK** - Merupakan suatu lingkungan terpadu untuk pengembangan dan eksekusi untuk berbagai macam bahasa pemrograman dan kumpulan *library* untuk bekerja sama membuat dan menjalankan aplikasi berbasis Windows yang lebih mudah untuk dibuat, diatur, didistribusikan, dan diintegrasikan dengan sistem jaringan lain.
- .WMF** - Windows Metafile (WMF). Adalah format *file* grafis pada sistem Windows Microsoft, awalnya dirancang pada awal tahun 1990-an. Windows metafiles dimaksudkan untuk menjadi portabel di antara aplikasi dan mungkin mengandung komponen vektor dan *bitmap*.
- BAKTERI** - Berasal dari Bahasa Latin *bacterium* (jamak, *bacteria*). Berukuran kecil (mikroskopik) dan kebanyakan uniselular (bersel tunggal). Berada di mana-mana, di tanah, air, dan sebagai simbiosis dari organisme lain.
- BATIMETRI** - Studi tentang kedalaman dasar danau atau lautan. Dengan kata lain, batimetri adalah setara dengan *hypsometry* (pengukuran elevasi tanah relatif terhadap permukaan laut) bawah air.
- BOUYANCY** - Gaya ke atas yang ditimbulkan oleh cairan terhadap benda yang terapung.
- CCRF** - *Code Of Conduct For Responsible Fisheries*. Kesepakatan dalam konferensi *Committee on Fisheries* (COFI) ke-28 FAO di Roma pada tanggal 31 Oktober 1995. Standar internasional mengenai pola perilaku bagi praktek yang bertanggung jawab, dalam pengusahaan sumberdaya perikanan dengan maksud untuk menjamin terlaksananya aspek konservasi, pengelolaan dan pengembangan efektif sumberdaya hayati akuatik berkenaan dengan pelestarian ekosistem dan keanekaragaman hayati.
- DELPHI** - Sebuah bahasa pemrograman dan lingkungan pengembangan perangkat lunak. dikembangkan oleh CodeGear sebagai divisi pengembangan perangkat lunak milik Embarcadero. Bahasa Delphi, atau



dikenal pula sebagai *object pascal* (*pascal* dengan ekstensi pemrograman berorientasi objek (PBO/OOP)).

DIOXIN - Istilah umum untuk penamaan lain dari *polychlorinated dibenzo-p-dioxins* (PCDDs) dan *polychlorinated dibenzofurans* (PCDFs). Diklasifikasikan sebagai kelompok senyawa aromatik poliklor (*polychlorinated aromatic compounds*, PCA) yang sangat beracun. Hal ini disebabkan karena keberadaan *dioxin* dalam tubuh dapat mengakibatkan terganggunya fungsi hormonal tubuh.

EMULATOR - Memungkinkan suatu program atau piranti lunak yang dibuat pada awalnya oleh suatu sistem komputer (arsitektur dan sistem operasi) dan untuk dijalankan dalam sistem itu (atau dijalankan dalam suatu sistem yang didedikasikan), dapat dijalankan dalam sistem komputer yang sama sekali berbeda.

FISHING LURE – Sebuah benda yang diikatkan pada ujung dari tali pancing yang dirancang menyerupai dan bergerak seperti mangsa dari ikan. Fungsi dari pikatan adalah menggunakan gerakan, getaran, dan warna untuk menarik perhatian ikan sehingga memakan kail pancing.

FLOWCHART - Merupakan tipe umum dari diagram, merepresentasikan algoritma atau proses, memperlihatkan langkah-langkah dalam berbagai bentuk kotak, dan perintah yang dihubungkan dengan tanda panah. diagram dapat memberikan langkah demi langkah solusi untuk memecahkan masalah. Data diwakili dengan kotak, dan tanda panah menghubungkan kotak yang memperlihatkan arah aliran data. *Flowchart* digunakan untuk menganalisis, merancang, mendokumentasikan atau mengelola proses atau program dalam berbagai bidang.

FOOD CHAIN - Rantai makanan dalam Bhs. Indonesia. Merepresentasikan hubungan sederhana dari organisme hidup di alam.

FOOD WEB - Jaring makanan dalam Bhs. Indonesia. Merupakan rangkaian dari hubungan rantai makanan memperlihatkan pergerakan energi dan materi dalam ekosistem. Rantai makanan dan jaring makanan adalah representasi dari hubungan pemangsa dan mangsa antar spesies dalam ekosistem atau habitat.

FUSIFORM - Bentuk morfologi ikan perenang cepat yang dicirikan dengan tubuh yang meruncing pada bagian kepala dan ekor.

HILLBAN - Ban luar kendaraan bermotor khususnya mobil yang digunakan sebagai sambungan tali dalam komponen rumpon. Berfungsi mereduksi tegangan tali ketika rumpon beroperasi.

IDE - *Integrated Development Environment* atau *Integrated Design Environment* atau *Integrated Debugging Environment*. Merupakan perangkat lunak yang

menyediakan fasilitas lengkap untuk pemrograman komputer untuk pengembangan perangkat lunak.

IKA SHIBI FISHING - Memancing cumi-cumi.

JASAD RENIK - Atau mikro organisme adalah makhluk hidup yang terdiri dari satu atau beberapa kumpulan sel dengan ukuran beberapa mikron (1 mikron = 0,001 mm).

JIGGING - Metode penangkapan ikan yang menggunakan jig. Jig merupakan salah satu tipe pikatan yang terdiri dari pemberat timah dengan sebuah kail. Biasanya ditutupi oleh badan yang halus untuk memikat ikan.

KOROSI - Perkaratan.

LINUX - Merupakan salah satu contoh hasil pengembangan *free software* dan *open source*. Nama "Linux" berasal dari nama kernelnya (kernel Linux), yang diperkenalkan tahun 1991 oleh Linus Torvalds.

LONGSHORE DRIFT - Gerakan sedimen, biasanya pasir, sepanjang pantai yang sejajar dengan garis pantai. Hal ini juga disebut *longshore current* atau *littoral drift*.

MIKROALGA - Mikroorganisme nabati yang hidup melayang-layang dalam air, relatif tidak mempunyai daya gerak sehingga keberadaannya dipengaruhi oleh gerakan air serta mampu berfotosintesis. Mikroalga umumnya bersel satu atau berbentuk benang, sebagai tumbuhan dan dikenal sebagai fitoplankton. Fitoplankton memiliki zat hijau daun (klorofil) yang berperan dalam fotosintesis untuk menghasilkan bahan organik dan oksigen dalam air.

MONSOON CLIMATE - Sebuah hujan secara tradisional didefinisikan sebagai pembalikan musiman disertai angin perubahan musiman dalam curah hujan, tetapi sekarang digunakan untuk menggambarkan perubahan musiman dalam sirkulasi atmosfer dan curah hujan.

NEARSHORE CURRENT - Sistem arus yang disebabkan oleh kegiatan gelombang di dekat zona pecah gelombang, yang terdiri dari: transportasi masa air ke arah pantai, arus memanjang pantai, dan arus renggutan.

OVERFISHING - Terjadi ketika aktivitas penangkapan mengurangi ketersediaan sumberdaya ikan dibawah batas level yang dapat diterima. Mengarah kepada deplesi sumberdaya dalam kasus perikanan bersubsidi, pertumbuhan rata-rata biologis rendah, dan level biomassa kritis.

PAH - *Polycyclic Aromatic Hydrocarbons*. senyawa kimia PAH terdapat di minyak, batu bara, dan tar deposit, dan diproduksi sebagai produk

RADAR REFLECTOR - Dengan menganalisa sinyal yang dipantulkan tersebut, pemantul sinyal dapat ditentukan lokasinya dan ditentukan jenisnya. Meskipun sinyal yang diterima relatif lemah, namun radio sinyal tersebut dapat dengan mudah dideteksi dan diperkuat oleh radar.

RUMPON - Bangunan menyerupai pepohonan yang dipasang di suatu tempat di tengah laut. Merupakan alat bantu penangkapan yang berfungsi untuk memikat ikan untuk kemudian dilakukan operasi penangkapan.

SALINITAS - Tingkat keasinan atau kadar garam terlarut dalam air. Salinitas juga dapat mengacu pada kandungan garam dalam tanah.

SEINE FISHING - Penangkapan dengan menggunakan pukat. *Seine* adalah jaring ikan besar yang menggantung secara vertikal di dalam air dengan melampirkan pemberat sepanjang tepi bawah dan pelampung di bagian atas. Perahu disebut *seiners*. Jaring panjang datar seperti pagar jala yang digunakan untuk mengepung kelompok ikan, dengan menggunakan kapal di sekitar lingkaran.

SERAT KACA - (Bahasa Inggris: *fiberglass*) atau sering diterjemahkan menjadi serat gelas adalah kaca cair yang ditarik menjadi serat tipis dengan diameter sekitar 0,005 mm - 0,01 mm. Serat ini dapat dipintal menjadi benang atau ditenun menjadi kain, yang kemudian diresapi dengan resin sehingga menjadi bahan yang kuat dan tahan korosi untuk digunakan sebagai badan mobil dan bangunan kapal.

SoFAD - *Software of Fish Aggregating Device*. Perangkat lunak yang digunakan untuk menghitung *buoyancy force* rumpun.

SOLAR POWERED LIGHT - Lampu yang sumber listriknya berasal dari cahaya matahari dan memiliki kemampuan untuk mengisi ulang baterai secara otomatis sehingga lampu akan tetap menyala pada malam hari.

SPEAR FISHING - Penangkapan ikan yang menggunakan tombak.

SPINCAST FISHING - Penangkapan ikan yang menggunakan rod (jejer) dimana mata pancing dilemparkan kemudian tarik kembali secara horizontal.

SPORT FISHING - Dikenal juga sebagai *recreational fishing*. Merupakan memancing untuk hiburan dan perlombaan. Ikan ditangkap untuk dilepaskan kembali.

SWIVEL - Sambungan untuk menyambungkan benda agar mampu berputar secara horizontal dan / atau vertikal.

TOMBA JERIGEN - Termasuk dalam kategori alat tangkap pancing. Digunakan untuk menangkap tuna. Menggunakan jerigen sebagai pelampung.

DAFTAR ISI

	Halaman
DAFTAR TABEL	xv
DAFTAR GAMBAR	xvi
DAFTAR LAMPIRAN	xix
1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Tujuan	4
1.3 Perumusan Masalah	4
1.4 Manfaat	4
2 TINJAUAN PUSTAKA	5
2.1 Pengertian Rumpon	5
2.2 Klasifikasi Rumpon	6
2.3 Fungsi Rumpon	9
2.4 Konstruksi Rumpon	10
2.5 Material Rumpon	11
2.6 Metode Penangkapan	11
2.7 Peraturan Pemasangan Rumpon	15
2.8 Perangkat Lunak <i>Borland Delphi 7</i>	17
2.9 Dasar Pembuatan Program	22
2.10 Tipe Data	23
2.11 Operator	25
2.12 <i>File-file</i> Pendukung <i>Project</i>	28
3 METODE PENELITIAN	30
3.1 Waktu dan Tempat	30
3.2 Bahan dan Alat	30
3.3 Metode Penelitian	31
3.4 Analisis Data	31
4 KONDISI UMUM LOKASI PENELITIAN	67
4.1 Batas Wilayah Kabupaten Sukabumi	67
4.2 Iklim	67
4.3 Geologi	68
4.4 Hidrologi	69
4.5 Oseanografi	69
5 HASIL DAN PEMBAHASAN	73
5.1 Rumpon	73
5.1.1 Ponton	76
5.1.2 <i>Swivel</i>	78

5.1.3	Tali utama.....	79
5.1.4	Hillban.....	83
5.1.5	Pemberat utama.....	86
5.1.6	Atraktor.....	88
5.1.7	Tali hillban-pemberat, tali tambat, tali atraktor.....	89
5.1.8	Selang plastik, karung, pemberat atraktor.....	90
5.1.9	Tali tambat, tali pengikat <i>swivel</i>	91
5.1.10	Waring, tali rafia.....	92
5.2	Alat Tangkap.....	95
5.2.1	Tombo jerigen.....	95
5.2.2	Pancing layangan.....	97
5.2.3	Pancing kondo-kondo.....	99
5.2.4	Pancing taber.....	101
5.2.5	Pancing coping.....	102
5.2.6	Pancing tonda.....	103
5.3	Hasil Tangkapan.....	105
5.3.1	Cakalang.....	105
5.3.2	Madidihang.....	108
5.4	Jarak Pemasangan Rumpon.....	111
5.5	SoFAD.....	115
6	PENUTUP	122
6.1	Kesimpulan.....	122
6.2	Saran.....	123
	DAFTAR PUSTAKA	124
	LAMPIRAN	130

Hal Cipta Milik IPB University
 1. Dilindungi sebagai alat cipta milik IPB University dan merupakan sumber daya intelektual yang tidak dapat dipisahkan dari IPB University.
 2. Diperoleh dengan izin dari IPB University.
 3. Tidak diperbolehkan untuk diperjualbelikan, dipinjamkan, atau disebarkan kepada pihak lain.
 4. Penggunaan tanpa izin IPB University merupakan pelanggaran terhadap hak cipta IPB University.
 5. Penggunaan tanpa izin IPB University dapat mengakibatkan sanksi hukum yang berlaku di Indonesia.
 6. IPB University tidak bertanggung jawab atas kerugian yang timbul akibat penggunaan alat cipta milik IPB University.

DAFTAR TABEL

		Halaman
1	Komponen pokok dan bahan material rumpon.....	11
2	Derajat proses pengolahan data	26
3	Operasi aritmatika.....	26
4	Operator relasi	27
5	Data komponen rumpon yang diukur	31
6	Jenis perhitungan dari data komponen rumpon yang diukur.....	32
7	Notasi <i>flowchart</i>	47
8	Arah angin berdasarkan bulan	68
9	Posisi pemasangan rumpon.....	111
10	Jarak antar rumpon	112
11	Jarak pemasangan rumpon.....	113

Halaman ini merupakan bagian dari buku yang diterbitkan oleh IPB University dan merupakan hak cipta IPB University. Tidak diperkenankan untuk menyalin, menduplikasi, atau menyebarkan kembali tanpa izin tertulis dari IPB University.

DAFTAR GAMBAR

		Halaman
1	Rumpon tradisional dengan struktur rakit bambu	7
2	Rumpon tradisional lainnya	7
3	Rumpon modern	8
4	<i>Drop stone lining</i>	12
5	<i>Trolling</i>	13
6	<i>Pool and line fishing</i>	13
7	<i>Vertical longlining</i>	14
8	<i>Ring-netting and purse-seining</i>	15
9	Delphi 7 interface	18
10	Main window	18
11	Main menu	18
12	Toolbar	19
13	Component palette	19
14	Form designer	20
15	Object inspector	20
16	Code editor	21
17	Code explorer	21
18	Object treeview	22
19	Ilustrasi <i>buoyancy force</i>	33
20	Flowchart perhitungan lapisan besi ponton	48
21	Flowchart perhitungan lapisan fiber ponton	49
22	Flowchart perhitungan lapisan sekat ponton	50
23	Flowchart perhitungan lapisan isi ponton	51
24	Flowchart perhitungan komponen hillban	52
25	Flowchart perhitungan komponen <i>swivel</i>	53
26	Flowchart perhitungan komponen pemberat utama	54
27	Flowchart perhitungan komponen pemberat atraktor	55
28	Flowchart perhitungan komponen tali utama	56
29	Flowchart perhitungan komponen atraktor	57

30	<i>Flowchart</i> perhitungan volume waring	58
31	<i>Flowchart</i> perhitungan komponen waring.....	59
32	<i>Flowchart</i> perhitungan komponen tali hillban-pemberat	60
33	<i>Flowchart</i> perhitungan komponen selang	61
34	<i>Flowchart</i> perhitungan komponen karung.....	62
35	<i>Flowchart</i> perhitungan komponen tali rafia	63
36	<i>Flowchart</i> perhitungan <i>total buoyancy force</i>	64
37	<i>Flowchart</i> perhitungan <i>total sinking force</i>	65
38	<i>Flowchart</i> perhitungan <i>extra buoyancy</i>	66
39	Arus permukaan Teluk Palabuhanratu.....	71
40	Peta kontur batimetri Teluk Palabuhanratu	72
41	Desain dan konstruksi rumpon	74
42	Desain dan konstruksi ponton.....	76
43	<i>Swivel</i> ponton.....	78
44	Konstruksi tali utama.....	83
45	Konstruksi hillban.....	84
46	Konstruksi pemberat utama	86
47	Pemberat dari drum.....	87
48	Pemberat dari galian tanah.....	88
49	Atraktor daun kelapa.....	89
50	Waring	93
51	Tomba jerigen.....	97
52	Pancing layangan	99
53	Pancing kondo-kondo	101
54	Pancing taber	102
55	Pancing coping.....	103
56	Pancing tonda.....	104
57	Ikan cakalang	106
58	Ikan madidihang	109
59	Pola pemasangan rumpon	114
60	Aplikasi menu SoFAD.....	116
61	Aplikasi atraktor SoFAD	118

62	<i>Saved file</i> atraktor.txt.....	120
63	Aplikasi <i>extra buoyancy</i> SoFAD.....	121



Hal Cipta Plintong! Unsur-unsurnya
 1. Diambil sebagai salah satu unsur yang terdapat dalam unsur-unsurnya
 2. Diperoleh sebagai salah satu unsur yang terdapat dalam unsur-unsurnya
 3. Diperoleh sebagai salah satu unsur yang terdapat dalam unsur-unsurnya
 4. Diperoleh sebagai salah satu unsur yang terdapat dalam unsur-unsurnya
 5. Diperoleh sebagai salah satu unsur yang terdapat dalam unsur-unsurnya
 6. Diperoleh sebagai salah satu unsur yang terdapat dalam unsur-unsurnya
 7. Diperoleh sebagai salah satu unsur yang terdapat dalam unsur-unsurnya
 8. Diperoleh sebagai salah satu unsur yang terdapat dalam unsur-unsurnya
 9. Diperoleh sebagai salah satu unsur yang terdapat dalam unsur-unsurnya
 10. Diperoleh sebagai salah satu unsur yang terdapat dalam unsur-unsurnya

DAFTAR LAMPIRAN

	Halaman
1	Komponen rumpon 131
2	Alat penangkapan ikan 136
3	Hasil tangkapan 140
4	Data pengukuran dan hasil perhitungan 142
5	<i>Screenshot</i> aplikasi SoFAD 144
6	<i>Source code</i> SoFAD 154

1. PENDAHULUAN

1.1 Latar Belakang

Sumberdaya ikan merupakan sumberdaya yang dapat diperbaharui (*renewable resource*). Meskipun demikian, dalam pemanfaatannya harus dikontrol agar kelestariannya tetap terjaga sesuai dengan *Code of Conduct For Responsible Fisheries (CCRF, 1995)*.

Berhasil tidaknya usaha penangkapan ikan di laut pada dasarnya adalah bagaimana mendapatkan daerah penangkapan, gerombolan ikan, dan keadaan potensinya, untuk kemudian dilakukan operasi penangkapan ikan. Beberapa cara untuk mendapatkan kawanan ikan sebelum penangkapan dilakukan adalah menggunakan alat bantu penangkapan (*fish aggregating device, fish lure*) yang dapat berupa lampu atau rumpon (Subani dan Barus, 1989).

Rumpon yang biasa disebut *Fish Aggregating Device (FAD)*, merupakan alat bantu penangkapan ikan yang dipasang di laut, berfungsi untuk memikat ikan agar berkumpul dalam wilayah penangkapan dimana rumpon tersebut dipasang. Menurut Subani dan Barus (1989), rumpon merupakan suatu bangunan menyerupai pepohonan yang dipasang di suatu tempat di tengah laut.

Adanya ikan di sekitar rumpon berkaitan dengan pola jaringan makanan dimana rumpon menciptakan suatu arena makan yang dimulai dengan tumbuhnya bakteri dan mikroalga ketika rumpon dipasang. Kemudian makhluk renik ini bersama dengan hewan-hewan kecil lainnya menarik perhatian ikan-ikan pelagis ukuran kecil. Ikan-ikan pelagis ini akan memikat ikan yang berukuran lebih kecil untuk memakannya (Subani, 1972).

Tujuan pemasangan rumpon di suatu perairan adalah untuk memikat ikan yang beruaya agar mau singgah, beristirahat, berkumpul, atau terkonsentrasi di sekitar rumpon sehingga akan mempermudah nelayan menentukan daerah penangkapan ikan (*fishing ground*). Dengan adanya kepastian daerah penangkapan ikan, maka waktu dan biaya operasi penangkapan ikan bisa diprediksi secara akurat, sehingga usaha penangkapan ikan akan menjadi lebih efektif dan efisien (Martasuganda, 2008).

Hal yang harus diperhatikan dalam penangkapan ikan adalah: 1. Pemilihan lokasi penangkapan ikan yang sesuai dengan kondisi perairan. 2. Pemilihan waktu penangkapan ikan yang sesuai dengan siklus hidup ikan. 3. Pemilihan alat penangkapan ikan yang sesuai dengan jenis ikan. 4. Pemilihan tenaga kerja yang terampil dan berpengalaman. 5. Pemilihan lokasi penangkapan ikan yang aman dan tidak mengganggu lingkungan. 6. Pemilihan waktu penangkapan ikan yang sesuai dengan musim. 7. Pemilihan alat penangkapan ikan yang sesuai dengan jenis ikan. 8. Pemilihan tenaga kerja yang terampil dan berpengalaman. 9. Pemilihan lokasi penangkapan ikan yang aman dan tidak mengganggu lingkungan. 10. Pemilihan waktu penangkapan ikan yang sesuai dengan musim.

Monintja (1992) menyatakan bahwa manfaat yang diharapkan dengan penggunaan rumpon sebagai alat bantu penangkapan ikan adalah: (1) mengurangi waktu dan bahan bakar dalam pengejaran kelompok ikan; (2) meningkatkan hasil tangkapan per satuan upaya penangkapan; (3) meningkatkan hasil tangkapan ditinjau dari spesies dan komposisi ukuran.

Penelitian mengenai rumpon, khususnya mengenai desain dan konstruksi rumpon laut dalam masih sedikit dilakukan. Penelitian yang dilakukan sebelumnya lebih mengarah pada:

1. Mekanisme berkumpulnya ikan pelagis di dekatar rumpon (Yusfiandayani, 2004).
2. Perikanan payang dengan rumpon di Pasauran, Kabupaten Banten (Siddharta, 2004).
3. Studi Tentang kepadatan ikan pelagis di sekitar rumpon di Perairan Pasauran, Banten (Imawati, 2003).
4. Komposisi hasil tangkapan payang menurut waktu dan periode bulan di sekitar rumpon di Perairan Pasauran, Kabupaten Banten (Sianipar, 2003)
5. Pengaruh penggunaan rumpon pada bagan apung terhadap hasil tangkapan (Effendi, 2002).
6. Pengaruh penggunaan umpan terhadap hasil tangkapan cakalang (*Katsuwonus pelamis*) dengan menggunakan pancing ulur pada rumpon di perairan Pambosuang Kabupaten Polmas Sulawesi Selatan (Zainuddin, 2002).
7. Analisis pengaruh waktu pemancingan dan periode bulan terhadap jenis dan komposisi hasil tangkapan *handline* di sekitar Teluk Tomini (Nahumury, 2001).
8. Pengoperasian *purse seiner* menggunakan cahaya dan rumpon sebagai pemikat ikan (Sutyawan, 1999).
9. Telaah rumpon tradisional di Perairan Pulau Marsola Sibolga (Simanjuntak, 1993).
10. Studi tentang penangkapan madidihang (*Thunnus albacares*) di sekitar rumpon di perairan Waigeo, Sorong (Nugroho, 1992).

11. Studi tentang penggunaan rumpon pada bagan apung di Teluk Palabuhanratu, Jawa Barat (Zulkarnain, 2002).
12. Pemanfaatan rumpon laut dalam : upaya meningkatkan hasil tangkapan ikan cakalang di PT. Usaha Mina (Persero) Bacan, Kabupaten Halmahera Selatan, Maluku Utara (Ardianto, 2005).
13. Penambahan rumpon untuk meningkatkan hasil tangkapan kelong tancap di daerah Kowal, Kabupaten Tanjung Pinang kepulauan Riau (Syahaan, 2005)
14. Pengoperasian pancing tonda pada rumpon di selatan perairan Teluk Palabuhanratu, Sukabumi Jawa Barat (Handriana, 2007).
15. Konstruksi rumpon laut dangkal dengan pelampung utama jenis ponton di perairan kabupaten Pandeglang Provinsi Banten (Wahyudin, 2007).

Selatan perairan Teluk Pelabuhanratu memiliki sumberdaya ikan yang melimpah, khususnya sumberdaya ikan cakalang dan madidihang. Oleh karena itu, perairan ini dijadikan sebagai tempat pemasangan rumpon yang dibuat oleh nelayan Palabuhanratu. Rumpon yang digunakan berjenis rumpon laut dalam dengan pelampung bertipe ponton. Rumpon laut dalam ini perlu dikaji lebih dalam karena rumpon digunakan pada perairan yang memiliki arus dan gelombang yang kuat, maka rumpon harus memiliki desain dan konstruksi yang kokoh agar dapat bertahan lama. Atas dasar inilah, penulis melakukan penelitian mengenai desain dan konstruksi rumpon laut dalam di selatan perairan Teluk Palabuhanratu, Kabupaten Sukabumi.

Selain itu, sampai saat ini belum ada perangkat lunak yang dapat menghitung secara pasti nilai dari *buoyancy force* dari rumpon yang dibuat. Pembuatan rumpon lebih berdasarkan pengalaman atau perkiraan. *Buoyancy force* memperlihatkan seberapa besar kemampuan rumpon untuk mengapung di perairan. Nilai kelebihan daya apung atau *extra buoyancy* memperlihatkan apakah pelampung rumpon akan tenggelam atau pemberat akan terangkat dari dasar jika asumsi panjang tali rumpon lebih pendek dari kedalaman perairan digunakan. Oleh karena itulah, pada penelitian ini pula penulis mencoba membuat suatu perangkat lunak sederhana untuk menghitung nilai *buoyancy force*, *sinking force*, dan *extra buoyancy* rumpon berdasarkan penelitian terhadap rumpon laut dalam di selatan perairan Teluk Palabuhanratu, Kabupaten Sukabumi.

1.2 Tujuan

Tujuan dari penelitian ini adalah :

1. Mendeskripsikan rumpon laut dalam yang meliputi komponen dan material rumpon laut dalam yang digunakan oleh nelayan Palabuhanratu
2. Menghitung *buoyancy force*, *sinking force*, dan *extra buoyancy* pada rumpon dengan asumsi arus perairan tidak mempengaruhi.
3. Membuat suatu perangkat lunak sederhana untuk menghitung nilai *buoyancy force*, *sinking force*, dan *extra buoyancy* dengan bahasa pemrograman *Pascal* menggunakan perangkat lunak *Borland Delphi 7*.

1.3 Perumusan Masalah

Perumusan masalah dari penelitian ini adalah :

1. Masih terbatasnya penelitian dan informasi mengenai desain dan konstruksi beserta komponen dan material rumpon laut dalam yang dioperasikan di selatan perairan Teluk Palabuhanratu, Kabupaten Sukabumi.
2. Belum adanya aplikasi yang secara khusus menghitung menghitung nilai *buoyancy force*, *sinking force*, dan *extra buoyancy* yang bekerja pada rumpon.

1.4 Manfaat

Manfaat yang diharapkan dapat diperoleh dari penelitian ini diantaranya:

1. Dapat memberikan gambaran jelas mengenai desain dan konstruksi beserta komponen dan material rumpon laut dalam yang dioperasikan di selatan perairan Teluk Palabuhanratu, Kabupaten Sukabumi.
2. Dapat memperlihatkan apakah pelampung rumpon akan tenggelam atau pemberat akan terangkat dari dasar perairan jika asumsi panjang tali rumpon lebih pendek dari kedalaman perairan digunakan berdasarkan hasil perhitungan.
3. Dengan diselesaikannya aplikasi sederhana untuk menghitung *buoyancy force*, *sinking force*, dan *extra buoyancy* pada rumpon tersebut, diharapkan dapat memudahkan peneliti selanjutnya untuk melakukan penelitian lanjutan.

2. TINJAUAN PUSTAKA

2.1 Pengertian Rumpon

Rumpon adalah salah satu jenis alat bantu penangkapan ikan yang dipasang dilaut, baik laut dangkal maupun laut dalam. Pemasangan tersebut dimaksudkan untuk menarik gerombolan ikan agar berkumpul disekitar rumpon, sehingga ikan mudah untuk ditangkap. Dengan pemasangan rumpon maka kegiatan penangkapan ikan akan menjadi lebih efektif dan efisien karena tidak lagi berburu ikan (dengan mengikuti ruayanya), tetapi cukup melakukan kegiatan penangkapan ikan disekitar rumpon tersebut (Juklak Pemasangan dan Pemanfaatan Rumpon, 2006).

Menurut Subani dan Barus (1989), rumpon merupakan suatu bangunan menyerupai pepohonan yang dipasang di suatu tempat di tengah laut. Disebut sebagai alat bantu penangkapan, fungsinya hanya sebagai pembantu, yaitu untuk mengumpulkan ikan pada suatu titik atau tempat tertentu untuk kemudian dilakukan operasi penangkapan ikan.

Rumpon biasa disebut juga dengan *Fish Aggregating Device* (FAD), yaitu suatu alat bantu penangkapan yang berfungsi untuk memikat ikan agar berkumpul dalam suatu *catchable area* (Sudirman dan Mallawa, 2004).

Menurut Seaman (2000) diacu dalam Agus (2005) menyatakan bahwa rumpon atau yang biasa disebut terumbu buatan merupakan kerangka buatan manusia yang ditenggelamkan di dasar perairan untuk mempengaruhi proses-proses fisik maupun biologi yang berhubungan dengan sumberdaya hayati laut.

Selain itu, menurut Monintja dan Zulkarnain (1995), rumpon merupakan alat bantu yang berfungsi untuk merangsang ikan untuk datang berkumpul di sekitar rumpon, mulai dari ikan-ikan pelagis kecil hingga ikan-ikan pelagis besar. Rumpon merupakan suatu sistem tropik level yang lengkap, dimana dapat ditemukan mulai dari produsen (fitoplankton) sampai predator (ikan tuna besar) sebagai konsumen.

Rumpon dapat juga diartikan sebagai *fish shelter*, dimana rumpon dijadikan tempat perkembangbiakan ikan dan untuk menambah kelimpahan dan keanekaragaman ikan dan terumbu karang (Agus, 2005).

2.2 Klasifikasi Rumpon

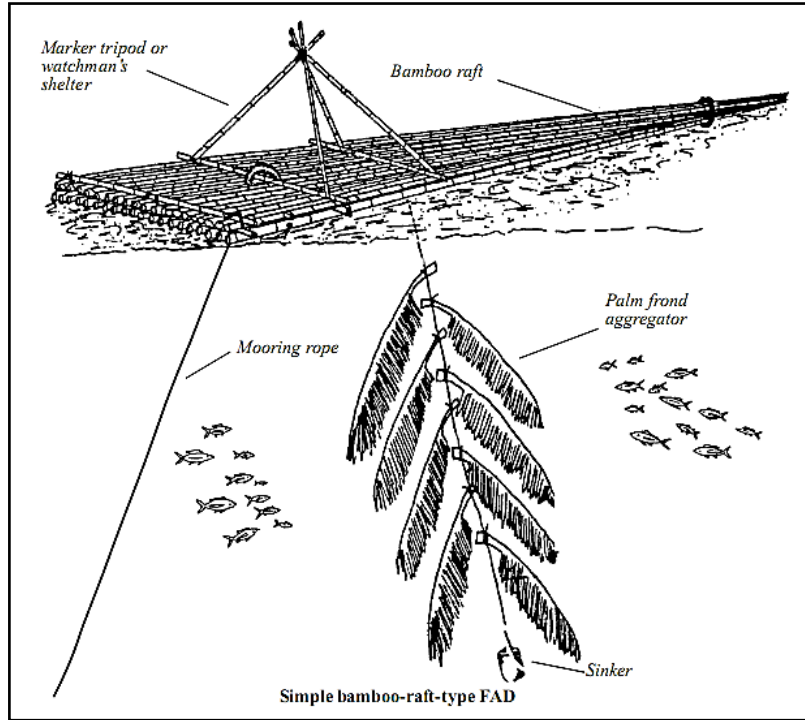
Tingkah laku dari berbagai jenis ikan untuk berkelompok di sekitar kayu terapung (seperti jenis tuna dan cakalang) merupakan awal dari adanya rumpon (Inoue, 1963 dalam Monintja, 1993). Hingga saat ini telah banyak berkembang berbagai jenis rumpon yang nama sebutannya maupun desain dan konstruksi secara detail dan lengkapnya, serta material yang digunakan berbeda signifikan antar daerah (Monintja, 1992). Agar mudah dalam mengenali rumpon, beberapa peneliti telah melakukan klasifikasi terhadap rumpon berdasarkan kriteria-kriteria tertentu.

Dilihat dari kedalaman air tempat rumpon dipasang, dibedakan menjadi rumpon laut dangkal dan rumpon laut dalam (Subani dan Barus, 1989). Rumpon laut dangkal biasanya dipasang kurang dari 100 meter, sedangkan rumpon laut dalam dipasang pada kedalaman lebih dari 600 meter, bahkan bisa mencapai 1500 meter (Monintja dan Zulkarnain 1995) diacu dalam Sianipar (2003).

Menurut Zakri (1993) diacu dalam Sianipar (2003) tipe-tipe rumpon yang dikembangkan hingga saat ini dikelompokkan atas kategori sebagai berikut:

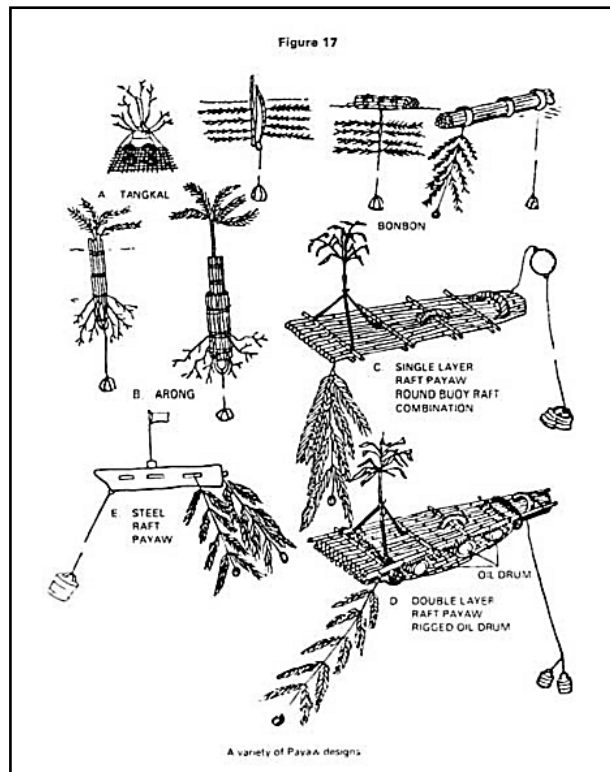
1. Berdasarkan posisi dari pemikat, rumpon dapat dibagi menjadi rumpon perairan permukaan lapisan tengah dan dasar. Rumpon perairan permukaan tengah terdiri dari jenis perairan dangkal dan perairan dalam.
2. Berdasarkan kriteria permanensi, rumpon dapat dikelompokkan menjadi rumpon yang dijangkar secara tetap (statis) dan rumpon yang dijangkar tetapi dapat dipindahkan.

Berdasarkan teknologi yang digunakan, rumpon dikelompokkan menjadi rumpon tradisional, dan modern. Menurut Anderson dan Gates (1996) dalam *South Pacific Commission Fish Aggregating Device (FAD) Manual Volume I*, rumpon tradisional berbentuk seperti rakit, mengapung, dan terbuat dari bambu. Rakit ini ditambatkan menggunakan tali serat alami yang dihubungkan dengan keranjang berisi batu sebagai jangkar di dasar perairan. Struktur ini merupakan rumpon yang pertama kali dibuat.



Sumber: Anderson dan Gates, 1996

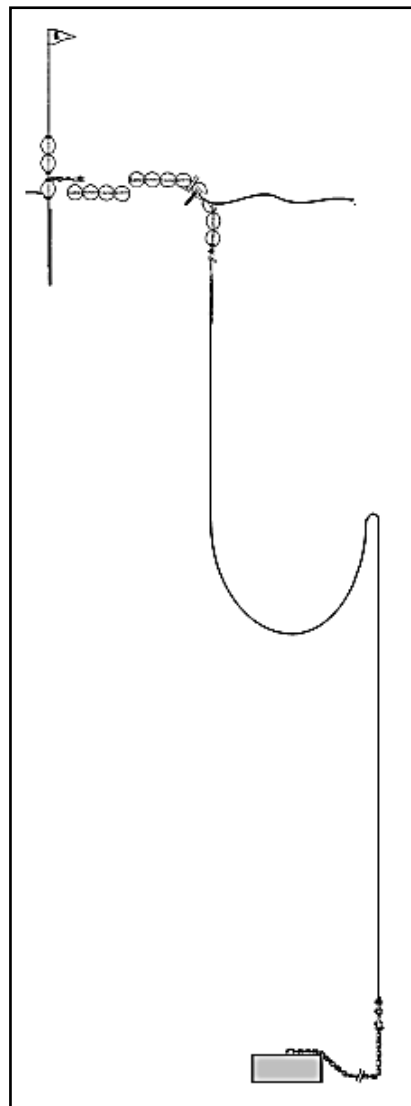
Gambar 1 Rumpon tradisional dengan struktur rakit bambu.



Sumber: Review Of Experiences With And Present Knowledge About Fish Aggregating Device

Gambar 2 Rumpon tradisional lainnya.

Rumpon modern bisa dijangkarkan di perairan hingga mencapai kedalaman lebih dari 2000 meter dan dilengkapi dengan *radar reflectors* dan *solarpowered light*. Jika rumpon sederhana terbuat dari bahan alami, maka rumpon modern terbuat dari besi, aluminium atau *fiberglass*. Beberapa rumpon modern dibuat menggunakan pelampung yang tahan terhadap arus kuat dan kondisi badai. Dibuat dengan memperhitungkan kombinasi dari tali sintetis yang sekarang biasa digunakan sebagai tali pemberat yang didesain tahan terhadap kondisi buruk dari lingkungan perairan laut. Beberapa dari rumpon ini dapat bertahan hingga lima tahun di perairan (Anderson dan Gates, 1996).



Sumber: Anderson dan Gates, 1996

Gambar 3 Rumpon modern.

2.3 Fungsi Rumpon

Fungsi utama rumpon adalah sebagai alat untuk menarik perhatian ikan agar berkumpul pada tempat tertentu yang kemudian dilakukan operasi penangkapan ikan (Subani dan Barus, 1989). Rumpon dapat berfungsi pula sebagai sumber makanan dan tempat berlindung dari predator.

Tujuan pemasangan rumpon di suatu perairan adalah untuk memikat ikan yang beruaya agar mau singgah, beristirahat, berkumpul, atau terkonsentrasi di sekitar rumpon sehingga akan mempermudah nelayan menentukan daerah penangkapan ikan (*fishing ground*). Dengan adanya kepastian daerah penangkapan ikan, maka waktu dan biaya operasi penangkapan ikan bisa diprediksi secara akurat, sehingga usaha penangkapan ikan akan menjadi lebih efektif dan efisien (Martasuganda, 2008).

Monintja (1992) menyatakan bahwa manfaat yang diharapkan dengan penggunaan rumpon sebagai alat bantu penangkapan ikan adalah : (1) mengurangi waktu dan bahan bakar dalam pengejaran kelompok ikan; (2) meningkatkan hasil tangkapan per satuan upaya penangkapan; (3) meningkatkan hasil tangkapan ditinjau dari spesies dan komposisi ukuran.

Berbagai alasan dikemukakan oleh Samples dan Sproul (1985) diacu dalam Imawati (2003) untuk menjelaskan ketertarikan ikan terhadap rumpon, antara lain:

1. Tempat berlindung dari predator bagi ikan-ikan tertentu (*hiding place*).
2. Tempat mencari makan (*feeding ground*).
3. Sebagai substrat untuk meletakkan telur bagi ikan-ikan tertentu (*spawning ground*).
4. Sebagai tempat berteduh (*shading place*).
5. Sebagai titik acuan navigasi (*meeting point*).

Adapun keuntungan yang diperoleh dari menggunakan rumpon yang terdapat dalam *Fish Aggregating Devices (FADs) and Fisheries Development in Tanzania* (2003) adalah:

1. Meningkatkan hasil tangkapan rata-rata.
2. Penggunaan biaya penangkapan lebih efisien.
3. Meningkatkan pendapatan dan kestabilan pendapatan.
4. Meningkatkan pendapatan perikanan.

5. Pembangunan pasar yang lebih luas.
6. Diversifikasi penggunaan sumberdaya laut.
7. Mengurangi ketergantungan terhadap sumberdaya tertentu.
8. Mengurangi persaingan pada perikanan pantai.
9. Merupakan perikanan berkelanjutan.
10. Meningkatkan persediaan sumber protein hewani.
11. Meningkatkan produktifitas buruh.

2.4 Konstruksi Rumpon

Menurut Rumpon *Study Group Agricultural University* (1987), persyaratan umum material rumpon adalah sebagai berikut:

1. Ponton
Ponton berfungsi sebagai pelampung, mempunyai daya apung yang baik karena sepertiga bagian mengapung di permukaan air. Konstruksinya kuat, serta memiliki kemampuan bertahan terhadap cuaca. Ponton dapat dikenali dari kejauhan dan bahan pembuatnya mudah didapat.
2. Atraktor
Memiliki daya pikat yang baik terhadap ikan (atraktor berbahan daun kelapa). Berbentuk seperti pepohonan yang dapat disusun secara vertikal. Berasal dari bahan yang kuat, murah dan mudah didapat.
3. Tali
Tali yang digunakan sebagai tali rumpon harus memiliki daya apung yang baik, tidak mudah membusuk, murah dan mudah didapat, serta tahan terhadap arus dan beban.
4. Pemberat
Bahan pemberat murah, kuat dan mudah didapat. Selain itu memiliki berat jenis yang besar dan permukaannya tidak licin.

Sama halnya dengan yang digunakan di negara lain, rumpon atau *fish aggregating device* di Indonesia pada dasarnya memiliki empat komponen utama. Keempat komponen tersebut adalah pelampung (*buoy, float*), atraktor (*aggregator, atractor*), tali tambat (*mooring line*), dan pemberat utama (*sinker*,

anchor). Namun, konstruksi secara detail dan lengkapnya, serta material yang digunakan berbeda signifikan antar daerah (Monintja, 1992).

2.5 Material Rumpon

Material rumpon merupakan bahan yang digunakan sebagai komponen penyusun konstruksi rumpon. Untuk tali, bahan yang baik digunakan adalah tali dengan jenis PP (*polypropylene*) atau PE (*polyethylene*), sedangkan untuk pemikat ikan biasanya terbuat dari daun kelapa. Namun ada pula yang menggunakan daun pinang dan daun nipah sebagai atraktor (Yusfiandayani, 2004). Sebagai pemberat digunakan pemberat berbahan semen yang diisikan kedalam drum.

Menurut Monintja (1990) yang diacu dalam Sianipar (2003) menyatakan berbagai faktor yang perlu dipertimbangkan dalam menilai prospek penggunaan rumpon antara lain:

1. Ketersediaan bahan baku rumpon.
2. Daya tahan rumpon terhadap kondisi perairan.
3. Kemudahan operasi penangkapan.

Tabel 1 Komponen pokok dan bahan material rumpon

No	Komponen	Bahan
1	Pelampung (<i>float</i>)	Plastik, Bambu
2	Tali tambat (<i>mooring line</i>)	Tali, <i>Wire</i> , Rantai, <i>Swivel</i>
3	Pemikat Ikan (<i>atractor</i>)	Daun Kelapa, Jaring Bekas
4	Pemberat (<i>bottom sinker</i>)	Batu, Beton

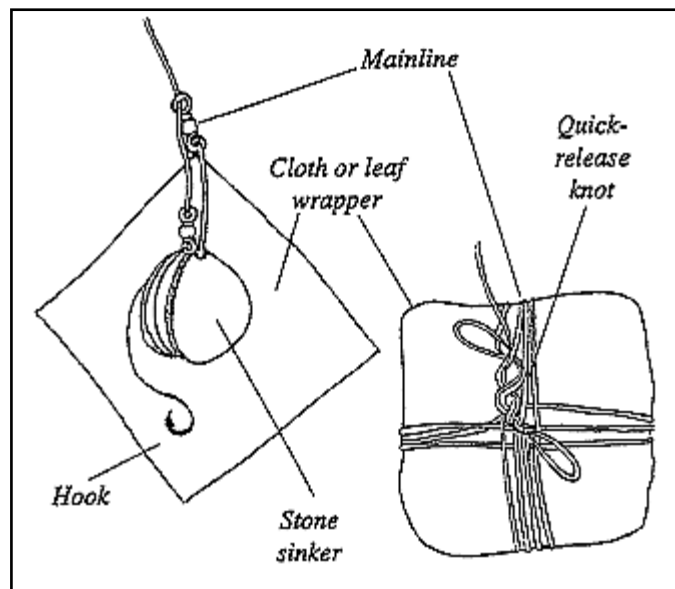
Umumnya rumpon yang dipasang di perairan yang lebih dalam memiliki konstruksi yang lebih lengkap dan kuat (Sudirman dan Mallawa, 2004).

2.6 Metode Penangkapan

Tipe alat tangkap yang biasa digunakan di sekitar rumpon akan tergantung dari alat tangkap yang biasa digunakan masyarakat pesisir yang bermukim di sekitar perairan dimana rumpon dipasang. Namun berikut merupakan alat tangkap yang biasanya digunakan menurut Anderson dan Gates (1996):

1. *Drop lining/drop stone lining*

Ini merupakan metode penangkapan untuk target tangkapan yang lebih besar, yaitu tuna. Teknik *drop stone lining* telah lama digunakan di perairan Pasifik yang merupakan wilayah berkumpulnya tuna. Umpan cincang dibalut dengan daun atau kain ditempatkan bersama dengan kail dan batu sebagai pemberat. Sebuah simpul diikatkan mengelilingi umpan menggunakan tali utama. Ketika umpan dan kail mencapai target kedalaman, tali utama disentakkan untuk melepaskan simpul, batu jatuh dan umpan bersama kail melayang terbawa arus di kedalaman dimana terdapat *yellowfin tuna* besar dan tuna lainnya.

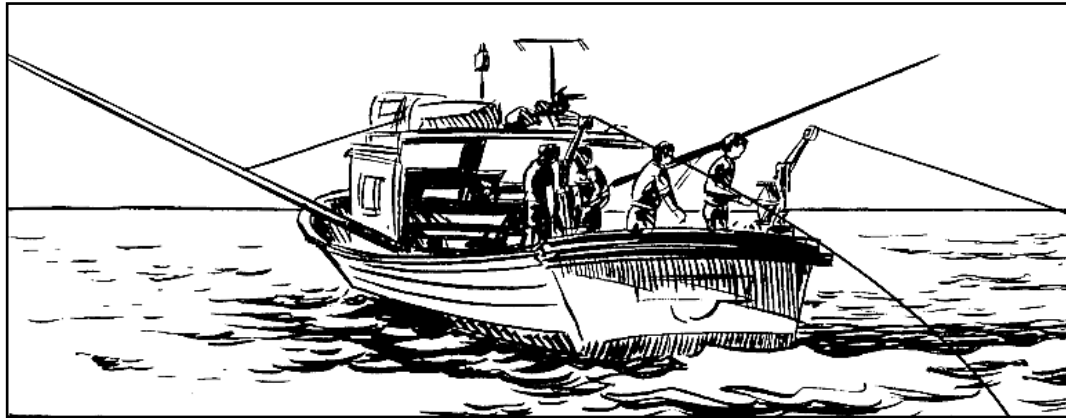


Sumber: Anderson dan Gates, 1996

Gambar 4 *Drop stone lining*.

2. *Trolling*

Metode penangkapan sederhana, menggunakan bulu, plastik berwarna atau umpan alami yang ditarik di permukaan atau sub-permukaan perairan. Alat tangkap ini murah dan dapat digunakan di atas perahu kecil.

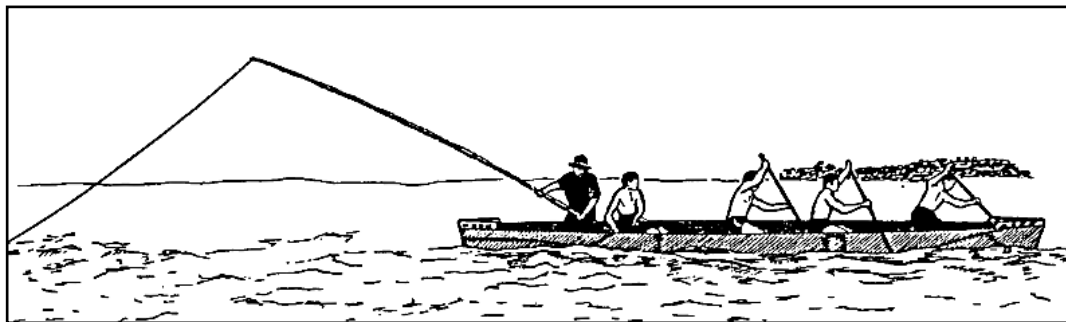


Sumber: Anderson dan Gates, 1996

Gambar 5 *Trolling*.

3. *Pole and line fishing*

Pole and line fishing merupakan metode penangkapan di permukaan yang menggunakan seni. Tidak menggunakan umpan, kait tanpa pemberat diikat pada sebuah tali dengan panjang tertentu yang terikat pada sebuah tiang pancing yang kemudian digoncang-goncangkan untuk menarik perhatian tuna.



Sumber: Anderson dan Gates, 1996

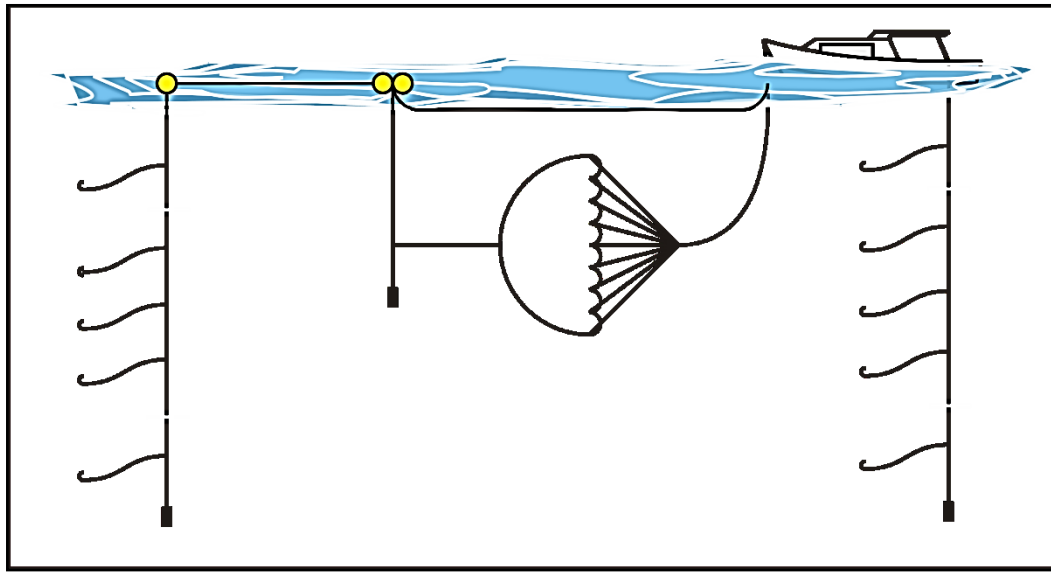
Gambar 6 *Pole and line fishing*.

4. *Trapping*

Perangkap ikan merupakan alat tangkap ikan tradisional yang kadang bisa digunakan secara efektif di dekat rumpon, khususnya di dekat rumpon laut dangkal. Digunakan untuk menangkap ikan pelagis kecil. Pada beberapa kasus, ikan hidup yang berada di dalam perangkap menarik datangnya ikan-ikan lain. Dengan kata lain, perangkap ikan merupakan cara terbaik untuk menangkap umpan untuk metode penangkapan ikan yang lain, misalnya *vertical longlining*.

5. *Vertical longlining*

Vertical longlining (dalam kasus ini, menggunakan jangkar laut atau parasut) merupakan teknik penangkapan ikan di rumpon laut sedang dengan menggunakan kapal kecil untuk menangkap ikan secara serempak pada kedalaman tertentu. Tali utama dengan pemberat di goncangkan setiap beberapa meter dengan cabang pendek dengan kait berumpan. Tali utama dapat berada di atas kapal atau dihanyutkan dekat rumpon dengan bantuan pelampung.

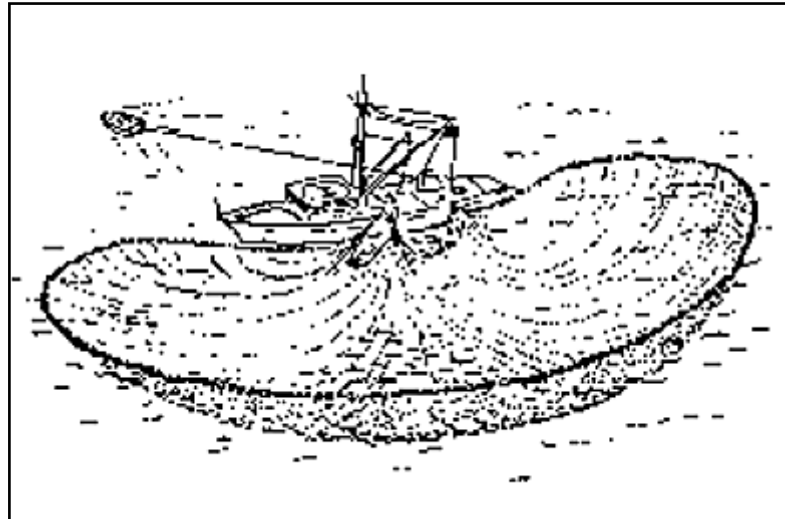


Sumber: Anderson dan Gates, 1996

Gambar 7 *Vertical longlining*.

6. *Ring-netting and purse-seining*

Ring-netting merupakan metode penangkapan ikan yang umumnya digunakan di Philippina. Kawanan ikan umpan atau pelagis kecil bergerombol di sekitar rumpon dengan menggunakan cahaya. Jaring melingkar diset mengelilingi kawanan dan menutup di bawah ikan. Metode yang sama digunakan dengan skala yang lebih besar oleh *purse seiners* yang dapat menangkap berton-ton tuna dalam sekali seting.



Sumber: Anderson dan Gates, 1996

Gambar 8 *Purse seining*.

7. Teknik lainnya

Banyak metode penangkapan lain yang juga dapat secara produktif digunakan di sekitar rumpon. Hal ini termasuk kedalam teknik penangkapan komersial dan *sport fishing*, termasuk *deep trolling*, *ika shibi fishing*, *jigging*, *spin casting* dan bahkan *spear fishing*.

2.7 Peraturan Pemasangan Rumpon

Sejalan dengan perkembangan rumpon, pemasangan rumpon tak hanya menimbulkan efek positif dengan meningkatkan produksi perikanan. Akibat dari pemasangan rumpon yang tidak teratur dan lokasi pemasangan yang berdekatan dapat merusak pola ruaya ikan yang bermigrasi. Kondisi tersebut dapat merusak keseimbangan ekosistem dan menimbulkan konflik baik antar nelayan rumpon maupun antara nelayan rumpon dengan bukan nelayan rumpon. Selain itu, kemudahan menangkap ikan di sekitar rumpon dapat mengakibatkan *overfishing* (Juklak Petunjuk Pemasangan dan Pemanfaatan Rumpon, 2006).

Berdasarkan keadaan tersebut dan sebelum kondisi tersebut berlangsung lebih lama sehingga berdampak lebih buruk, pemerintah mengeluarkan Buku Juklak Petunjuk Pemasangan dan Pemanfaatan Rumpon, dengan tujuan:

1. Menata pemasangan rumpon agar kelestarian ikan terjaga.

2. Menerbitkan Pedoman Pembinaan Sarana Perikanan Tangkap khususnya rumpon.
3. Mencegah terjadinya konflik sosial antara nelayan pemilik rumpon dengan nelayan bukan pemilik rumpon.
4. Membina pengelolaan rumpon yang melibatkan unsur-unsur terkait baik pusat maupun daerah maupun antara perusahaan perikanan dengan nelayan sehingga tercapai kesinambungan dan keserasian usaha di lapangan dengan tujuan untuk kelestarian sumberdaya ikan.
5. Menyusun mekanisme pendataan dan penandaan dalam pemasangan rumpon serta mekanisme evaluasi produktivitas penangkapan di sekitar rumpon.

Berdasarkan Juklak Pemasangan dan Pemanfaatan Rumpon (2006), berikut merupakan tata cara pemasangan rumpon :

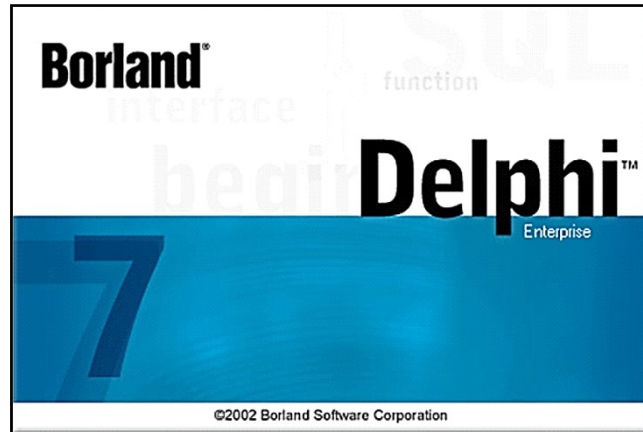
1. Rumpon dapat dipasang di wilayah :
 1. Perairan dua mil laut sampai dengan empat mil laut, diukur dari garis pantai pada titik surut terendah.
 2. Perairan di atas empat mil laut sampai dengan 12 mil laut, diukur dari garis pantai pada titik surut terendah.
 3. Perairan di atas 12 mil laut dan ZEE Indonesia.
 4. Perorangan atau perusahaan berbadan hukum yang akan memasang rumpon wajib terlebih dahulu memperoleh ijin.
2. Pengusaha atau nelayan yang akan memasang rumpon mengajukan permohonan ijin kepada Direktorat Jendral Perikanan Tangkap, Dinas Perikanan dan Kelautan Provinsi atau Kabupaten atau Kota sesuai dengan kewenangan pemberi ijin. Sesuai dengan Kepmen Kelautan dan Perikanan No. Kep 30/MEN/2004 tentang Pemasangan dan Pemanfaatan Rumpon. Dalam Permohonan ijin harus dilakukan penilaian baik terhadap administrasi pemohon maupun lokasi perairan. Penilaian lokasi pemasangan rumpon harus memperhatikan:

1. Apakah daerah tersebut merupakan alur pelayaran atau kepentingan lainnya seperti daerah suaka, atau daerah lainnya. Pemasangan rumpon tidak boleh dilakukan pada daerah perairan tersebut.
2. Apakah daerah tersebut merupakan konsentrasi penangkapan ikan nelayan-nelayan yang tidak menggunakan rumpon, rumpon tidak boleh dipasang pada perairan tersebut.
3. Apakah daerah tersebut berbatasan dengan provinsi, untuk itu maka Dinas Perikanan dan Kelautan dari domisili pemohon ijin rumpon ditujukan kepada provinsi tersebut.

2.8 Perangkat Lunak *Borland Delphi 7*

Delphi merupakan suatu perangkat lunak pemrograman yang memberikan berbagai fasilitas pembuatan aplikasi visual. Keunggulan bahasa pemrograman ini terletak pada produktivitas, kualitas, pengembangan perangkat lunak, kecepatan kompilasi, pola desain yang menarik serta diperkuat dengan pemrogramannya yang terstruktur. Keunggulan lain dari *Delphi* adalah dapat digunakan untuk merancang program aplikasi yang memiliki tampilan seperti program aplikasi lain yang berbasis *Windows*.

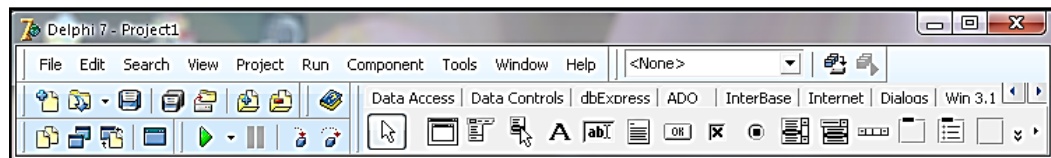
Lembar kerja *Delphi* atau dikenal dengan IDE (*Integrated Development Environment*) merupakan sebuah lingkungan dimana semua tombol perintah yang diperlukan untuk mendesain aplikasi, menjalankan dan menguji sebuah aplikasi disajikan dengan baik untuk memudahkan dalam pengembangan program. *Integrated Development Environment* (IDE) dalam program *Delphi* terbagi menjadi delapan bagian utama, yaitu *main window*, *toolbar*, *component palette*, *form designer*, *code editor*, *object inspector*, *code explorer*, dan *object treeview*.



Sumber: Pemrograman *Borland Delphi 7*, 2002

Gambar 9 *Delphi 7 interface*.

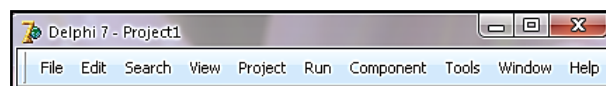
1. *Main Window*, merupakan bagian dari IDE yang mempunyai fungsi yang sama dengan semua fungsi utama dari program aplikasi *Windows* lainnya. Jendela utama terbagi menjadi tiga bagian, yaitu *main menu*, *toolbar* dan *compenent palette*.



Sumber: Pemrograman Borland Delphi 7, 2002

Gambar 10 *Main window*.

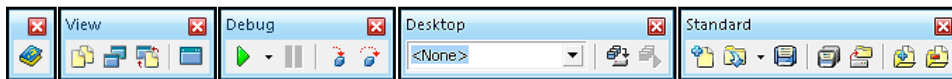
2. *Main Menu*, memiliki fungsi yang sama seperti program aplikasi *Windows* lainnya. Pada dasarnya semua perintah dapat ditemukan pada bagian menu utama ini.



Sumber: Pemrograman Borland Delphi 7, 2002

Gambar 11 *Main menu*.

3. *Toolbar*, memiliki berbagai fungsi yang berbeda-beda. Setiap tombol pada bagian *toolbar* berfungsi sebagai pengganti suatu menu perintah yang sering digunakan. *Toolbar* sering disebut juga *speedbar*. Pada kondisi *default*, *toolbar* terdiri dari enam bagian yaitu *standard*, *view*, *debug*, *desktops*, *custom*, dan *component palette*. Tombol-tombol pada bagian *toolbar* dapat dikurangi atau ditambah sesuai kebutuhan.



Sumber: Pemrograman Borland Delphi 7, 2002

Gambar 12 *Toolbar*.

4. *Component Palette*, berisi kumpulan *icon* yang melambangkan komponen-komponen yang terdapat pada VCL (*Visual Component Library*). Pada *component palette* terdapat beberapa *page control*, seperti *standard*, *additional*, *win32*, *data access* dan lain-lain.



Sumber: Pemrograman Borland Delphi 7, 2002

Gambar 13 *Component palette*.

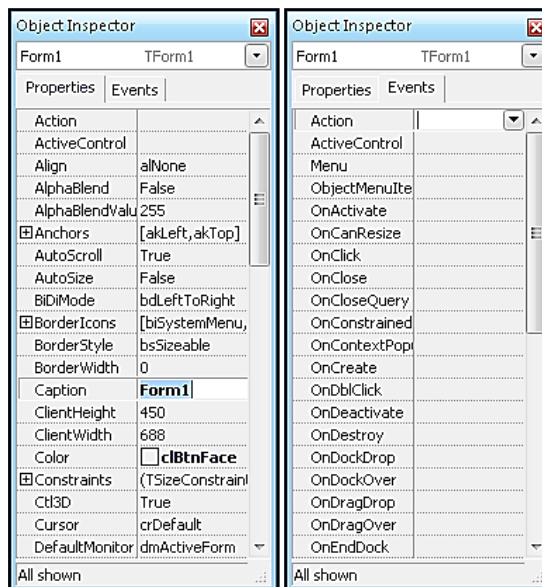
5. *Form Designer*, merupakan suatu obyek yang dapat dipakai sebagai tempat untuk merancang program aplikasi. *Form* berbentuk seperti sebuah meja kerja yang dapat diisi dengan komponen-komponen yang dapat diambil dari *component palette*. Dalam sebuah *form* terdapat titik-titik yang disebut *grid* yang berguna untuk membantu pengaturan tata letak objek yang dimasukkan kedalam *form*. Sebuah *form* mengandung *unit* yang berfungsi untuk mengendalikan *form*.



Sumber: Pemrograman Borland Delphi 7, 2002

Gambar 14 *Form designer*.

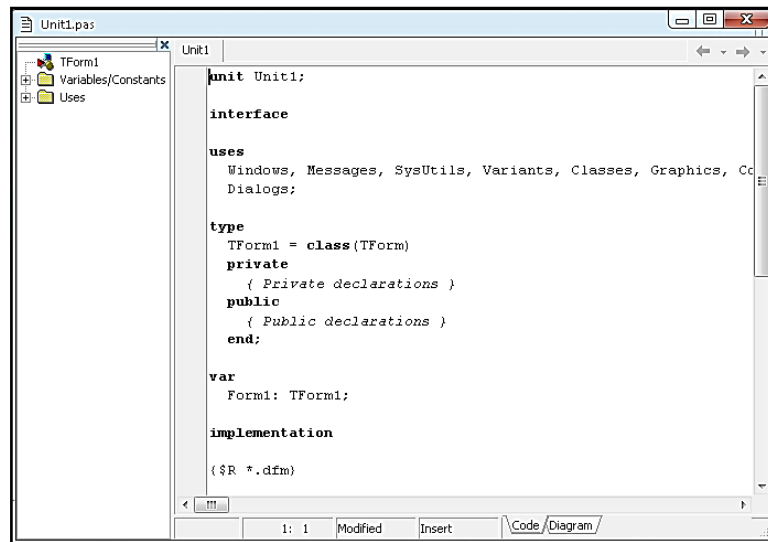
6. *Object Inspector*, digunakan untuk mengubah *property* atau karakteristik dari sebuah komponen. *Object inspector* terdiri dari dua *tab*, yaitu *properties* dan *events*.



Sumber: Pemrograman Borland Delphi 7, 2002

Gambar 15 *Object inspector*.

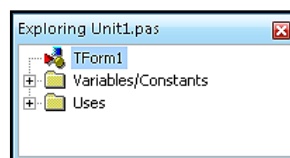
7. *Code Editor*, merupakan tempat menuliskan kode program. Pada bagian ini dituliskan pernyataan-pernyataan dalam *object pascal*. Keuntungan lain menggunakan *Delphi* adalah tidak perlu menuliskan kode-kode sumber, karena *Delphi* telah menyediakan kerangka penulisan sebuah program. *Code editor* dilengkapi dengan fasilitas *highlight* yang memudahkan dalam menemukan kesalahan.



Sumber: Pemrograman Borland Delphi 7, 2002

Gambar 16 *Code editor*.

8. *Code Explorer*, merupakan lembar kerja baru yang terdapat di dalam *Delphi* dan yang tidak ditemukan pada versi-versi sebelumnya. *Code explorer* digunakan untuk memudahkan pemakai berpindah antar *unit file* yang terdapat di dalam jendela *code editor*. *Code explorer* berisi diagram pohon yang menampilkan semua *type*, *class*, *property*, *method*, *variable global*, dan *routin global* yang telah didefinisikan di dalam *unit*.



Sumber: Pemrograman Borland Delphi 7, 2002

Gambar 17 *Code explorer*.

Mengatur tata letak komponen dilakukan dengan cara memilih *view*, *alignment palette*, maka akan muncul sebuah *tool box align*, pilih salah satu dengan klik untuk merapikan *alignment palette*.

5. Mengubah Nilai *Property*

Beberapa aturan penamaan komponen/*variable/identifier* adalah sebagai berikut :

- a. Diawali alphabet, angka atau garis bawah.
- b. Tidak menggunakan spasi, tanda baca, *reserved word*, dan operator.
- c. Bisa menggunakan kapital atau kecil, tidak ada perbedaan.
- d. Memberi nama komponen diawali dengan kelompok komponennya.

6. Membuat *Method* atau *Procedure* Melalui *Event*

Event yang biasanya digunakan pada komponen *button* adalah *onclick* yang berarti jika *button* tersebut ditekan akan melakukan *procedure* tertentu. Teknik yang digunakan untuk membuat *event onclick* pada komponen *button* adalah :

- a. Klik ganda pada komponen tersebut.
- b. Pilih *button* lalu pilih *object inspector's combo box (object selector)*, pilih tab *event*, klik ganda pada area putih di sebelah kanan *event onclick*.
- c. Pilih *button*, pilih tab *event* dan masukkan nama *method* yang diinginkan pada area putih di sebelah kanan *event onclick*.

2.10 Tipe Data

Delphi menyediakan tipe data yang cukup banyak dan lengkap, sehingga memberi keleluasaan bagi pemogram. Pemilihan tipe data yang tepat sangat penting dilakukan agar proses berjalan cepat dan stabil. Berikut merupakan tipe-tipe data yang pada *Delphi*.

1. Tipe *Integer*

Tipe data *integer* digunakan untuk menyatakan bilangan yang tidak mempunyai angka desimal.

2. Tipe *Real*



Tipe data real merupakan *floating point* (ditulis dalam $a \times 10^b$) yang digunakan untuk menyatakan bilangan yang mempunyai angka desimal.

3. Tipe *Boolean*

Tipe data *boolean* digunakan untuk menyatakan data logika, yaitu *true* (benar) dan *false* (salah).

4. Tipe *Character*

Tipe data *character* digunakan untuk menyatakan karakter satu huruf.

5. Tipe *String*

Tipe data *string* digunakan untuk menyatakan sederetan karakter yang membentuk satu kesatuan. Tipe data string bisa juga berisi karakter kosong atau satu karakter.

6. Tipe *Array*

Array adalah variabel tunggal yang dapat dipakai untuk menyimpan sekumpulan data sejenis. Untuk membedakan tempat menyimpan satu data dengan data lainnya, *array* menggunakan nomor elemen di sebelah kanan nama *array*. Suatu *array* mempunyai jumlah komponen yang jumlahnya tetap. *Array* dapat bertipe data sederhana *byte*, *word*, *integer*, *real*, *boolean*, *char* atau *string* dan tipe data *scalar* atau *subrange*. *Array* sering disebut sebagai larik.

7. Tipe *Record*

Tipe data *record* dipakai untuk menyimpan sekumpulan data yang mungkin berbeda tipe, tetapi saling berhubungan.

8. Tipe Terenumerasi dan *Subrange*

Tipe data terenumerasi dan tipe data *subrange* dipakai untuk menyatakan data berurutan yang bertipe sama.

9. Tipe *Ordinal*

Tipe *ordinal* terdiri atas tipe *integer*, *character*, *boolean*, *enumerated*, dan *subrange*. Tipe *ordinal* mempunyai sifat terurut. Tiap anggota *ordinal*, kecuali urutan pertama akan mempunyai nilai sebelumnya.

10. Konstanta

Konstanta adalah suatu nilai yang bersifat tetap. Nilai yang diberikan terhadap suatu konstanta akan disimpan di memori. Salah satu keuntungan penggunaan konstanta adalah program lebih mudah dimengerti.

11. Variabel

Variabel adalah suatu tempat yang dialokasikan dalam memori yang diberi nama (sebagai pengenalan) untuk menampung suatu data. Perbedaan konstanta dan variabel adalah konstanta dipergunakan saat kompilasi program, sedangkan variabel digunakan saat pelaksanaan program. Nilai konstanta tidak dapat berubah pada saat program berjalan, sedangkan nilai variabel dapat berubah.

12. Prosedur *Increment* dan *Decrement*

Prosedur *increment* dan *decrement* menghasilkan program yang sudah dioptimalkan untuk proses penambahan dan pengurangan pada *integer variable*. Setelah kompilasi, prosedur *increment* dan *decrement* akan menghasilkan suatu perintah dalam bahasa mesin.

13. Himpunan

Tipe himpunan atau set digunakan untuk menyimpan kumpulan nilai atau dapat dikatakan sebagai anggota himpunan yang bertipe sama.

2.11 Operator

Operator digunakan untuk memanipulasi dan mengolah data. *Delphi* menyediakan bermacam-macam operator. Setiap operator mempunyai tingkat hierarki, yaitu urutan pelaksanaan jika ada beberapa operator dalam satu proses. Sebagai contoh, jika terdapat ekspresi $X+Y*Z$, maka akan dilaksanakan terlebih dahulu operasi $Y*Z$, baru setelah itu, hasil operasi $Y*Z$ ditambah X . Hal ini disebabkan operator perkalian memiliki hierarki yang lebih tinggi dari operator penjumlahan. Jika dalam suatu ekspresi terdapat beberapa operator dengan tingkat hierarki yang sama, maka pelaksanaannya didahulukan mulai dari yang paling kiri. Selain itu, dapat pula menggunakan tanda kurung untuk mendahulukan suatu operasi, mendahului semua tingkat hierarki.

Tabel 2 Derajat proses pengolahan data

Operator	Hierarki
@, <i>not first</i>	tertinggi
*, /, div, mod, and, shl, shr, as	kedua
+, -, or, xor	ketiga
=, <>, >, <, >=, <=, in, is	keempat

Sumber: Pemrograman Borland Delphi 7, 2002

1. Operator Penugasan

Operator penugasan (*assignment*) disimbolkan dengan tanda titik dua diikuti dengan tanda sama dengan dan ditulis tanpa spasi. Operator penugasan berfungsi untuk memasukkan suatu data kedalam suatu variabel.

2. Operator Aritmatika

Operator aritmatika digunakan untuk melakukan operasi aritmatika.

Tabel 3 Operasi aritmatika

Operator	Operasi	Tipe Proses	Tipe Hasil
*	perkalian	<i>integer, real</i>	<i>integer, real</i>
/	pembagian	<i>integer, real</i>	<i>real</i>
<i>div</i>	pembagian bulat	<i>integer</i>	<i>integer</i>
<i>mod</i>	sisanya pembagian	<i>integer</i>	<i>integer</i>
+	penambahan	<i>integer, real</i>	<i>integer, real</i>
-	pengurangan	<i>integer, real</i>	<i>integer, real</i>

Sumber: Pemrograman Borland Delphi 7, 2002

Operator aritmatika di atas disebut juga sebagai operator aritmatika *binari*. Selain itu, ada juga operator aritmatika *unari*, yaitu tanda positif dan tanda negatif. Operator aritmatika *unari* dapat dipakai pada tipe data *integer* maupun *real*. Hierarki operator *unari* lebih tinggi dari semua operator *binari*.

3. Operator Relasi

Operator relasi atau perbandingan digunakan untuk membandingkan suatu ekspresi atau data dengan ekspresi atau data lain. Hasil penggunaan operator

relasi adalah logika (*boolean*) benar atau salah. Ekspresi atau data yang dibandingkan harus memiliki tipe yang sama.

Tabel 4 Tabel operator relasi

Operator	Keterangan
=	sama dengan
≠	tidak sama dengan
<	lebih kecil
>	lebih besar
≤	lebih kecil atau sama dengan
≥	lebih besar atau sama dengan

Sumber: Pemrograman Borland Delphi 7, 2002

Operator relasi hierarkinya lebih rendah dari operator aritmatika, sehingga operator aritmatika selalu dijalankan lebih dahulu sebelum operator relasi.

4. Operator Logika

a. Operator Logika *Not*

Operator logika *not* dipakai untuk menghasilkan nilai kebalikan dari suatu ekspresi atau data logika. Operator *not* mempunyai hierarki yang paling tinggi di antara operator logika lainnya.

b. Operator Logika *And*

Operator logika *and* akan menghasilkan nilai *boolean true* hanya jika dua buah ekspresi di kiri dan kanan operator bernilai *true*.

c. Operator Logika *Or*

Operator logika *or* akan menghasilkan nilai *boolean true* jika salah satu dari dua buah ekspresi atau kedua ekspresi di kiri dan kanan operator *or* bernilai *true*. Sebaliknya, operator *or* akan menghasilkan nilai *false* hanya jika kedua ekspresi di kiri dan kanan operator *or* bernilai *false*.

d. Operator Logika *Xor*

Operator logika *xor* akan menghasilkan nilai *true* jika ekspresi di sebelah kiri operator *xor* berbeda dengan ekspresi di sebelah kanannya.

2.12 File-file Pendukung *Project*

Ketika sebuah program aplikasi dibuat dengan menggunakan *Delphi*, tidak hanya *file project* dan sebuah *unit* yang dibentuk, melainkan ada beberapa *file* lain yang mendukung dalam proses pembuatan sebuah program aplikasi.

1. *File Project (.dpr)* dan *file Unit (.pas)*

Dipakai untuk menyimpan informasi mengenai *form* dan *unit*. *File* tersebut terbentuk saat membuat desain program *project* yang berisikan inisialisasi *form* utama dan *form-form* lain yang dibuat secara otomatis.

2. *File Form (.dfm)*

Merupakan *biner file* yang dibuat *Delphi* untuk menyimpan informasi yang berkaitan dengan *form*, dan setiap *form* mempunyai sebuah *file unit (.pas)*.

3. *File Resource (.res)*

Merupakan *biner file* yang berisi sebuah *icon* yang digunakan oleh *project*. *File* ini terus menerus diperbaharui selama proses pembuatan program aplikasi.

4. *File Project Options (.dof)* dan *file desktop settings (.dsk)*

File project options merupakan *file* yang berisi *option* dari suatu *project* yang dinyatakan melalui perintah *Option* dari menu *project*. *File* ini tersimpan saat pada saat *file project* disimpan. *File desktop settings* berisi *options* yang dinyatakan melalui perintah *Environment Options* dari menu *Tools*.

5. *File Backup (.~dp, .~df, .~pa)*

File dengan ekstensi di atas akan terbentuk pada saat proses penyimpanan untuk yang kedua kalinya. Disebut *file backup*, karena berisi salinan terakhir dari *file* utama sebelum disimpan lebih lanjut.

6. *File* jenis lain

a. *File executable (.exe)*

Dibentuk oleh *compiler* dan merupakan *file* eksekusi (*executable*) dari program aplikasi.

b. *File unit object (.dcu)*

Merupakan *file unit (.pas)* yang telah dikompilasi oleh *compiler* dan akan dihubungkan dengan *file* eksekusi.

c. *File dynamic link library (.dll)*

File dibentuk oleh *compiler* apabila dalam membuat program aplikasi *file* ini juga dibuat.

d. *File help (.hlp)*

File ini merupakan *file Windows* dan merupakan *file help* standar yang dapat dipakai di program aplikasi *Delphi*.

e. *File image (.wmf, .bmp, .ico)*

Merupakan *file Windows* dari program aplikasi selain *Delphi* yang dapat digunakan untuk mendukung program aplikasi yang dibuat.

3.3 Metode Penelitian

Metode penelitian yang digunakan adalah metode deskriptif. Urutan kegiatan penelitian diawali dengan menuliskan komponen penyusun rumpon, menuliskan bahan komponen rumpon, mengukur komponen rumpon, dan menghitung data hasil pengukuran komponen rumpon. Tahap selanjutnya adalah membuat aplikasi sederhana untuk menghitung gaya apung (*buoyancy force*) dan gaya tenggelam (*sinking force*) pada rumpon dengan bahasa pemrograman *pascal* menggunakan perangkat lunak *Borland Delphi 7*.

3.4 Analisis Data

Data yang akan dianalisis dalam penelitian ini adalah data aspek teknis yang meliputi desain dan konstruksi rumpon laut dalam yang kemudian dilakukan pembuatan program sederhana dengan bahasa pemrograman *pascal*. Data yang diperoleh akan dihitung menggunakan perhitungan matematis sederhana mengenai gaya apung (*buoyancy force*), gaya tenggelam (*sinking force*) dan *extra buoyancy* pada rumpon dengan asumsi arus perairan tidak mempengaruhi.

Tabel 5 Data Komponen Rumpon yang Diukur

Komponen	Data						
	Jumlah	Panjang	Lebar	Tinggi	Diameter	Tebal	Jumlah Sekat
Ponton	√	√	-	-	√	√	√
Hillban	√	-	√	-	√	√	-
Swivel	√	√	-	-	√	-	-
Pemberat	√	√	√	√	-	-	-
Tali	√	√	-	-	√	-	-
Atraktor	√	√	√	-	-	-	-

Tabel 6 Jenis Perhitungan dari Data Komponen Rumpon yang Diukur

Komponen	Perhitungan				
	Massa	Volume	Massa Jenis	<i>Bouyancy Force</i>	<i>Sinking Force</i>
Ponton	√	√	•	√	√
Hillban	•	√	√	√	√
Swivel	√	√	•	√	√
Pemberat	•	√	√	√	√
Tali	√	√	•	√	√
Atraktor	√	√	√	√	√
				<i>Total Bouyancy Force</i>	<i>Total Sinking Force</i>
				√	√
				<i>Extra Bouyancy</i>	
				√	

Pada tabel 6 jenis perhitungan dari data komponen yang diukur, tanda centang pada tabel menandakan nilai yang didapat merupakan hasil perhitungan, sedangkan tanda titik menandakan nilai yang didapat merupakan hasil studi literatur atau kuesioner. Setelah semua komponen rumpon diukur, maka tahap selanjutnya adalah melakukan perhitungan dengan menggunakan persamaan-persamaan berikut.

Persamaan *buoyancy force* didapat berdasarkan *principle of buoyancy* atau yang lebih dikenal sebagai *Archimedes principle* dalam *The Works of Archimedes* (1897). Pada dasarnya, prinsip ini menyatakan bahwa benda yang terbenam dalam fluida akan diberi tekanan keatas oleh sebuah kekuatan yang sama dengan berat fluida yang dipindahkan. Apakah benda tertentu akan mengapung, tenggelam, atau melayang di dalam fluida tertentu tergantung dari berat dan volume benda. Perbandingan antara berat dengan volume benda menghasilkan nilai densitas relatif. Dengan kata lain apakah benda tertentu akan mengapung, tenggelam, atau melayang di dalam fluida tertentu tergantung densitas relatif benda.

Ada tiga kemungkinan ketika suatu benda dimasukkan ke dalam suatu fluida, yaitu:

1. Jika gaya apung melebihi berat benda, benda akan bergerak ke atas, dengan kata lain benda akan mengambang, sehingga dapat dikatakan *buoyancy force* positif.

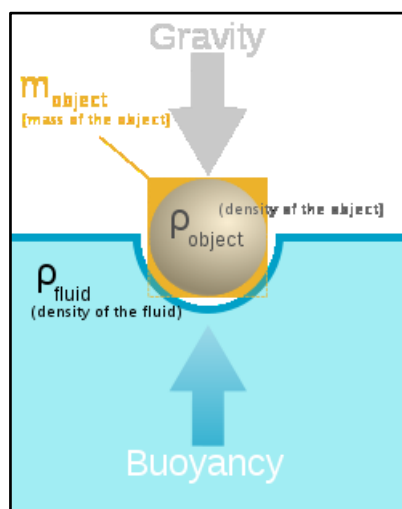
2. Jika berat benda melebihi gaya apung, benda akan bergerak ke bawah, dengan kata lain benda akan tenggelam, sehingga dapat dikatakan *buoyancy force* negatif.
3. Jika gaya apung sama dengan berat benda, benda akan berada pada kondisi statis, melayang di dalam fluida, sehingga dapat dikatakan *buoyancy force* netral.

Ada cara lain untuk menyatakan tiga kemungkinan tersebut adalah:

1. Jika berat benda melebihi berat fluida yang dipindahkan, benda akan tenggelam.
2. Jika berat fluida yang dipindahkan melebihi berat benda, objek akan mengambang.
3. Jika berat fluida yang dipindahkan sama dengan berat benda, ia akan tetap stasioner, melayang di dalam fluida.

Dengan kata lain:

1. Jika densitas benda lebih besar daripada densitas fluida, benda akan tenggelam, sehingga dapat dikatakan *buoyancy force* negatif.
2. Jika densitas benda lebih kecil daripada densitas fluida, benda akan terapung, sehingga dapat dikatakan *buoyancy force* positif.
3. Jika densitas benda sama dengan densitas fluida, benda akan melayang di dalam fluida, sehingga dapat dikatakan *buoyancy force* netral.



Sumber: <http://en.wikipedia.org>

Gambar 19 Ilustrasi *buoyancy force*.

$$P_{lfp} = \text{Massa Jenis Lapisan Fiber Ponton (kg/m}^3\text{)}$$

$$V_{lfp} = \text{Volume Lapisan Fiber Ponton (m}^3\text{)}$$

b. Volume Lapisan Fiber Ponton

$$V_{lfp} = P_{tp} \times T_{lfp} \times (\pi \times D_{Ponton}) + 1/3 \times (T_{Kerucut} \times (\pi \times D_{Ponton}) \times T_{lfp}) \\ \times ((\pi \times (1/2 \times D_{Ponton})^2) \times T_{lfp})$$

Keterangan:

$$V_{lfp} = \text{Volume Lapisan Fiber Ponton (m}^3\text{)}$$

$$P_{tp} = \text{Panjang Tabung Ponton (m)}$$

$$T_{lfp} = \text{Tebal Lapisan Fiber Ponton (m)}$$

$$\Pi = \text{Phi (3,1428)}$$

$$D_{Ponton} = \text{Diameter Ponton (m)}$$

$$T_{Kerucut} = \text{Tinggi Kerucut Ponton (m)}$$

c. *Buoyancy Force* Lapisan Fiber Ponton

$$F_{b\ lfp} = \rho_{air\ laut} \times V_{lfp} - M_{lfp}$$

Keterangan:

$$F_{b\ lfp} = \text{Buoyancy Force Lapisan Fiber Ponton (kg)}$$

$$\rho_{air\ laut} = \text{Rho Air laut (kg/m}^3\text{)}$$

$$V_{lfp} = \text{Volume Lapisan Fiber Ponton (m}^3\text{)}$$

$$M_{lfp} = \text{Massa Lapisan Fiber Ponton (kg)}$$

3. Sekat Ponton

a. Massa Sekat Ponton

$$M_{sp} = \rho_{sp} \times V_{sp}$$

Keterangan:

$$M_{sp} = \text{Massa Sekat Ponton (kg)}$$

$$P_{sp} = \text{Massa Jenis Sekat Ponton (kg/m}^3\text{)}$$

$$V_{sp} = \text{Volume Sekat Ponton (m}^3\text{)}$$

b. Volume Sekat Ponton

$$V_{sp} = J_{sp} \times (\pi \times (1/2 \times D_{Ponton})^2) \times T_{sp}$$

Keterangan:

J_{sp}	= Jumlah Sekat Ponton
V_{sp}	= Volume Sekat Ponton (m^3)
T_{sp}	= Tebal Sekat Ponton (m)
Π	= <i>Phi</i> (3,1428)
D_{Ponton}	= Diameter Ponton (m)

c. *Buoyancy Force* Sekat Ponton

$$F_{b\ sp} = \rho_{air\ laut} \times V_{sp} - M_{Ifp}$$

Keterangan:

$F_{b\ sp}$	= <i>Buoyancy Force</i> Sekat Ponton (kg)
$\rho_{air\ laut}$	= <i>Rho</i> Air laut (kg/m^3)
V_{sp}	= Volume Sekat Ponton (m^3)
M_{sp}	= Massa Sekat Ponton (kg)

4. Isi Ponton (Sterofoam)

a. Massa Isi Ponton

$$M_{ip} = \rho_{ip} \times V_{ip}$$

Keterangan:

M_{ip}	= Massa Isi Ponton (kg)
ρ_{ip}	= Massa Jenis Isi Ponton (kg/m^3)
V_{ip}	= Volume Isi Ponton (m^3)

b. Volume Isi Ponton

$$V_{ip} = (\pi \times r_{ponton}^2 \times P_{tp}) + (\pi \times r_{ponton}^2 \times T_{kerucut}) - V_{sp}$$

Keterangan:

P_{tp}	= Panjang Tabung Ponton (m)
V_{ip}	= Volume Isi Ponton (m^3)
Π	= <i>Phi</i> (3,1428)
r_{Ponton}	= Jari-jari Ponton (m)

T_{Kerucut} = Tinggi Kerucut Ponton (m)
 V_{sp} = Volume Sekat Ponton (m³)

c. *Buoyancy Force* Isi Ponton

$$F_{b \text{ ip}} = \rho_{\text{air laut}} \times V_{\text{ip}} - M_{\text{ip}}$$

Keterangan:

$F_{b \text{ ip}}$ = *Buoyancy Force* Isi Ponton (kg)
 $\rho_{\text{air laut}}$ = *Rho* Air laut (kg/m³)
 V_{ip} = Volume Isi Ponton (m³)
 M_{ip} = Massa Isi Ponton (kg)

5. **Hillban**

a. Massa Jenis Hillban

$$\rho_{\text{Hillban}} = M_{\text{Hillban}}/V_{\text{Hillban}}$$

Keterangan:

M_{Hillban} = Massa Hillban (kg)
 ρ_{Hillban} = Massa Jenis Hillban (kg/m³)
 V_{Hillban} = Volume Hillban (m³)

b. Volume Hillban

$$V_{\text{Hillban}} = 2 \times (T_{\text{Hillban}} \times (\pi \times r_1^2_{\text{Hillban}}) - T_{\text{Hillban}} \times (\pi \times r_2^2_{\text{Hillban}})) \\ + ((\pi \times D_{1\text{Hillban}}) \times T_{\text{Hillban}} \times L_{\text{Hillban}})$$

Keterangan:

V_{Hillban} = Volume Hillban (m³)
 T_{Hillban} = Tebal Hillban (m)
 L_{Hillban} = Lebar Hillban (m)
 Π = *Phi* (3.1428)
 r_1 dan r_2 = Jari-jari Hillban (m)
 $D_{1 \text{ Hillban}}$ = Diameter 1 Hillban (m)

c. *Buoyancy Force Hillban*

$$F_{b \text{ Hillban}} = \rho_{\text{air laut}} \times V_{\text{Hillban}} - M_{\text{Hillban}}$$

Keterangan:

$$F_{b \text{ Hillban}} = \text{Buoyancy Force Hillban (kg)}$$

$$\rho_{\text{air laut}} = \text{Rho Air Laut (kg/m}^3\text{)}$$

$$V_{\text{Hillban}} = \text{Volume Hillban (m}^3\text{)}$$

$$M_{\text{Hillban}} = \text{Massa Hillban (kg)}$$

6. *Swivel*a. *Massa Swivel*

$$M_{\text{Swivel}} = \rho_{\text{Swivel}} \times V_{\text{Swivel}}$$

Keterangan :

$$M_{\text{Swivel}} = \text{Massa Swivel (kg)}$$

$$\rho_{\text{Swivel}} = \text{Massa Jenis Swivel (kg/m}^3\text{)}$$

$$V_{\text{Swivel}} = \text{Volume Swivel (m}^3\text{)}$$

b. *Volume Swivel*

$$V_{\text{Swivel}} = P_{\text{Swivel}} \times (\pi \times r_{\text{Swivel}}^2)$$

Keterangan:

$$V_{\text{Swivel}} = \text{Volume Swivel (m}^3\text{)}$$

$$P_{\text{Swivel}} = \text{Panjang Swivel (m)}$$

$$r_{\text{Swivel}} = \text{Jari-jari Swivel (m)}$$

$$\Pi = \text{Phi (3.1428)}$$

c. *Buoyancy Force Swivel*

$$F_{b \text{ Swivel}} = \rho_{\text{air laut}} \times V_{\text{Swivel}} - M_{\text{Swivel}}$$

Keterangan:

$$F_{b \text{ Swivel}} = \text{Buoyancy Force Swivel (kg)}$$

$$\rho_{\text{air laut}} = \text{Rho Air Laut (kg/m}^3\text{)}$$

$$V_{\text{Swivel}} = \text{Volume Swivel (m}^3\text{)}$$

$$M_{\text{Swivel}} = \text{Massa Swivel (kg)}$$

7. Pemberat Utama

a. Volume Pemberat Utama

$$V_{pu} = P_{pu} \times L_{pu} \times T_{pu}$$

Keterangan:

V_{pu} = Volume Pemberat Utama (m^3)

P_{pu} = Panjang Pemberat Utama (m)

L_{pu} = Lebar Pemberat Utama (m)

T_{pu} = Tinggi Pemberat Utama (m)

b. Massa Jenis Pemberat Utama

$$\rho_{pu} = M_{pu}/V_{pu}$$

Keterangan:

P_{pu} = *Rho* Pemberat Utama (kg/m^3)

M_{pu} = Massa Pemberat Utama (kg)

V_{pu} = Volume Pemberat Utama (m^3)

c. *Buoyancy Force* Pemberat Utama

$$F_{b pu} = \rho_{air laut} \times V_{pu} - M_{pu}$$

Keterangan:

$F_{b pu}$ = *Buoyancy Force* Pemberat Utama (kg)

$P_{air laut}$ = *Rho* air laut (kg/m^3)

V_{pu} = Volume Pemberat Utama (m^3)

M_{pu} = Massa Pemberat Utama (kg)

8. Pemberat Atraktor

a. Volume Pemberat Atraktor

$$V_{pa} = P_{pa} \times L_{pa} \times T_{pa} \times J_{pa}$$

Keterangan:

V_{pa} = Volume Pemberat Atraktor (m^3)

P_{pa} = Panjang Pemberat Atraktor (m)

L_{pa} = Lebar Pemberat Atraktor (m)

T_{pa} = Tinggi Pemberat Atraktor (m)

J_{pa} = Jumlah Pemberat Atraktor (buah)

b. Massa Jenis Pemberat Atraktor

$$\rho_{pa} = M_{pa}/V_{pa}$$

Keterangan:

$$P_{pa} = \text{Rho Pemberat Atraktor (kg/m}^3\text{)}$$

$$M_{pa} = \text{Massa Pemberat Atraktor (kg)}$$

$$V_{pa} = \text{Volume Pemberat Atraktor (m}^3\text{)}$$

c. *Buoyancy Force* Pemberat Atraktor

$$F_{b\ pa} = \rho_{air\ laut} \times V_{pa} - M_{pa}$$

Keterangan:

$$F_{b\ pa} = \text{Buoyancy Force Pemberat Atraktor (kg)}$$

$$P_{air\ laut} = \text{Rho Air Laut (kg/m}^3\text{)}$$

$$V_{pa} = \text{Volume Pemberat Atraktor (m}^3\text{)}$$

$$M_{pa} = \text{Massa Pemberat Atraktor (kg)}$$

9. Tali Utama

a. Massa Tali Utama

$$M_{Tali} = M_{Tali\ Sampel} \times (P_{Tali}/P_{Tali\ Sampel})$$

Keterangan:

$$M_{Tali} = \text{Massa Tali (kg)}$$

$$M_{Tali\ Sampel} = \text{Massa Tali Sampel (kg)}$$

$$P_{Tali} = \text{Panjang Tali (m)}$$

$$P_{Tali\ Sampel} = \text{Panjang Tali Sampel (m)}$$

b. Volume Tali Utama

$$V_{Tali} = (\pi \times r_{ct}^2) \times P_{Tali} \times J_{ct}$$

Keterangan:

$$V_{Tali} = \text{Volume Tali (m}^3\text{)}$$

$$P_{Tali} = \text{Panjang Tali (m)}$$

$$J_{ct} = \text{Jumlah Cabang Tali (cabang)}$$

$$\Pi = \text{Phi (3.1428)}$$

$$r_{ct} = \text{Jari-jari Cabang Tali (m)}$$

c. *Buoyancy Force* Tali Utama

$$F_{b\ Tali} = \rho_{air\ laut} \times V_{Tali} - M_{Tali}$$

Keterangan:

$$F_{b\ Tali} = \text{Buoyancy Force Tali (kg)}$$

$$\rho_{air\ laut} = \text{Rho Air laut (kg/m}^3\text{)}$$

$$V_{Tali} = \text{Volume Tali (m}^3\text{)}$$

$$M_{Tali} = \text{Massa Tali (kg)}$$

10. **Atraktor**a. **Volume Atraktor**

$$V_{Atraktor} = P_{Bak} \times L_{Bak} \times \Delta t_{Air\ Bak}$$

Keterangan:

$$V_{Atraktor} = \text{Volume Atraktor (m}^3\text{)}$$

$$P_{Bak} = \text{Panjang Bak (m)}$$

$$L_{Bak} = \text{Lebar Bak (m)}$$

$$\Delta t_{Air\ Bak} = \text{Perubahan Ketinggian Air Bak (m)}$$

b. **Massa Jenis Atraktor**

$$\rho_{Atraktor} = M_{Atraktor} / V_{Atraktor}$$

Keterangan:

$$\rho_{Atraktor} = \text{Rho Atraktor (kg/m}^3\text{)}$$

$$M_{Atraktor} = \text{Massa Atraktor (kg)}$$

$$V_{Atraktor} = \text{Volume Atraktor (m}^3\text{)}$$

c. *Buoyancy Force* Atraktor

$$F_{b\ Atraktor} = \rho_{air\ laut} \times V_{Atraktor} - M_{Atraktor}$$

Keterangan:

$$F_{b\ Atraktor} = \text{Buoyancy Force Atraktor (kg)}$$

$$\rho_{air\ laut} = \text{Rho air laut (kg/m}^3\text{)}$$

$$V_{Atraktor} = \text{Volume Atraktor (m}^3\text{)}$$

$$M_{Atraktor} = \text{Massa Atraktor (kg)}$$

11. Waring

a. Jumlah Tali per Lebar Waring

$$Jt_{plw} = \frac{L_w}{MS_w + D_w}$$

Keterangan:

Jt_{plw} = Jumlah Tali per Lebar Waring (tali)

L_w = Lebar Waring (m)

MS_w = *Mesh size* Waring (m)

D_w = Diameter Benang Waring (m)

b. Jumlah Tali per Panjang Waring

$$Jt_{ppw} = \frac{P_w}{MS_w + D_w}$$

Keterangan:

Jt_{ppw} = Jumlah Tali per Panjang Waring (tali)

P_w = Panjang Waring (m)

MS_w = *Mesh size* Waring (m)

D_w = Diameter Waring (m)

c. Volume Waring

$$V_w = (\pi \times (\frac{1}{2}D_w)^2 \times Jt_{ppw} \times L_w) + (\pi \times (\frac{1}{2}D_w)^2 \times Jt_{plw} \times P_w) - (\pi \times (\frac{1}{2}D_w)^2 \times Jt_{ppw} \times Jt_{plw} \times D_w)$$

Keterangan:

V_w = Volume Waring

Π = *Phi* (3.1428)

D_w = Diameter Waring (m)

Jt_{ppw} = Jumlah Tali per Panjang Waring (tali)

Jt_{plw} = Jumlah Tali per Lebar Waring (tali)

L_w = Lebar Waring (m)

P_w = Lebar Waring (m)

d. Massa Waring

$$M_{Waring} = \rho_{Atraktor} \times M_{Waring}$$

Keterangan:

$$P_{Waring} = \text{Rho Waring (kg/m}^3\text{)}$$

$$M_{Waring} = \text{Massa Waring (kg)}$$

$$V_{Waring} = \text{Volume Waring (m}^3\text{)}$$

e. *Buoyancy Force* Waring

$$F_{b \text{ Atraktor}} = \rho_{air \text{ laut}} \times V_{Waring} - M_{Waring}$$

Keterangan:

$$F_{b \text{ Waring}} = \text{Buoyancy Force Waring (kg)}$$

$$\rho_{air \text{ laut}} = \text{Rho air laut (kg/m}^3\text{)}$$

$$V_{Waring} = \text{Volume Waring (m}^3\text{)}$$

$$M_{Waring} = \text{Massa Waring (kg)}$$

12. Tali Hillban-Pemberat

a. Massa Tali Hillban-Pemberat

$$M_{thp} = M_{Tali \text{ Sampel}} \times (P_{thp} / P_{Tali \text{ Sampel}})$$

Keterangan:

$$M_{thp} = \text{Massa Tali Hillban-Pemberat (kg)}$$

$$M_{Tali \text{ Sampel}} = \text{Massa Tali Sampel (kg)}$$

$$P_{thp} = \text{Panjang Tali Hillban-Pemberat (m)}$$

$$P_{Tali \text{ Sampel}} = \text{Panjang Tali Sampel (m)}$$

b. Volume Tali Hillban-Pemberat

$$V_{thp} = M_{thp} / \rho_{thp}$$

Keterangan:

$$V_{thp} = \text{Volume Tali Hillban-Pemberat (m}^3\text{)}$$

$$M_{thp} = \text{Massa Tali Hillban-Pemberat (kg)}$$

$$P_{thp} = \text{Massa Jenis Tali Hillban-Pemberat (kg/m}^3\text{)}$$

c. *Buoyancy Force* Tali Hillban-Pemberat

$$F_{b\ thp} = \rho_{air\ laut} \times V_{thp} - M_{thp}$$

Keterangan:

$F_{b\ thp}$ = *Buoyancy Force* Tali Hillban-Pemberat (kg)

$\rho_{air\ laut}$ = *Rho* air laut (kg/m^3)

V_{thp} = Volume Tali Hillban-Pemberat (m^3)

M_{thp} = Massa Tali Hillban-Pemberat (kg)

13. Selang

a. Volume Selang

$$V_s = P_s \times T_s \times D_s \times \pi$$

Keterangan:

V_s = Volume Selang (m^3)

P_s = Panjang Selang (m)

T_s = Tebal Selang (m)

D_s = Diameter Selang (m)

Π = *Phi* (3.1428)

b. Massa Selang

$$M_s = V_s \times \rho_s$$

Keterangan:

ρ_s = *Rho* Selang (kg/m^3)

M_s = Massa Selang (kg)

V_s = Volume Selang (m^3)

c. *Buoyancy Force* Selang

$$F_{b\ s} = \rho_{air\ laut} \times V_s - M_s$$

Keterangan:

$F_{b\ s}$ = *Buoyancy Force* Selang (kg)

$\rho_{air\ laut}$ = *Rho* air laut (kg/m^3)

V_s = Volume Selang (m^3)

M_s = Massa Selang (kg)

14. Karung

a. Volume Karung

$$V_k = M_k / \rho_k$$

Keterangan:

$$V_k = \text{Volume Karung (m}^3\text{)}$$

$$M_k = \text{Massa Karung (kg)}$$

$$\rho_k = \text{Massa Jenis Karung (kg/m}^3\text{)}$$

b. *Buoyancy Force* Karung

$$F_{b k} = \rho_{\text{air laut}} \times V_k - M_k$$

Keterangan:

$$F_{b k} = \text{Buoyancy Force Karung (kg)}$$

$$\rho_{\text{air laut}} = \text{Rho air laut (kg/m}^3\text{)}$$

$$V_k = \text{Volume Karung (m}^3\text{)}$$

$$M_k = \text{Massa Karung (kg)}$$

15. Tali Rafia

a. Volume Tali Rafia

$$V_{tr} = M_{tr} / \rho_{tr}$$

Keterangan:

$$V_{tr} = \text{Volume Tali Rafia (m}^3\text{)}$$

$$M_{tr} = \text{Massa Tali Rafia (kg)}$$

$$\rho_{tr} = \text{Massa Jenis Tali Rafia (kg/m}^3\text{)}$$

b. *Buoyancy Force* Tali Rafia

$$F_{b tr} = \rho_{\text{air laut}} \times V_{tr} - M_{tr}$$

Keterangan:

$$F_{b tr} = \text{Buoyancy Force Tali Rafia (kg)}$$

$$\rho_{\text{air laut}} = \text{Rho air laut (kg/m}^3\text{)}$$

$$V_{tr} = \text{Volume Tali Rafia (m}^3\text{)}$$

$$M_{tr} = \text{Massa Tali Rafia (kg)}$$

16. *Total Buoyancy Force*

$$F_{b\ Total} = F_{b_i} + \dots F_{b_p}$$

Keterangan:

$F_{b\ Total}$ = *Total Buoyancy Force* (gr)

i,p = Komponen rumpon yang memiliki nilai *buoyancy force* positif

17. *Total Sinking Force*

$$F_{s\ Total} = F_{s_j} + \dots F_{s_n}$$

Keterangan:

$F_{s\ Total}$ = *Total Sinking Force* (gr)

j,n = Komponen rumpon yang nilai *buoyancy force* bernilai negatif

18. *Extra Buoyancy*

$$E_b = \frac{F_{b\ Total} - F_{s\ Total}}{F_{b\ Total}} \times 100\%$$

Keterangan:

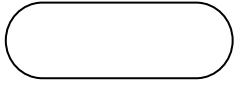

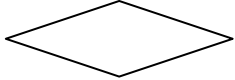
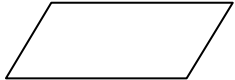
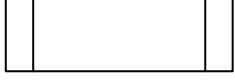
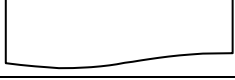

E_b = *Extra Buoyancy Force* (%)

$F_{b\ Total}$ = *Total Buoyancy Force* (gr)

$F_{s\ Total}$ = *Total Sinking Force*(gr)

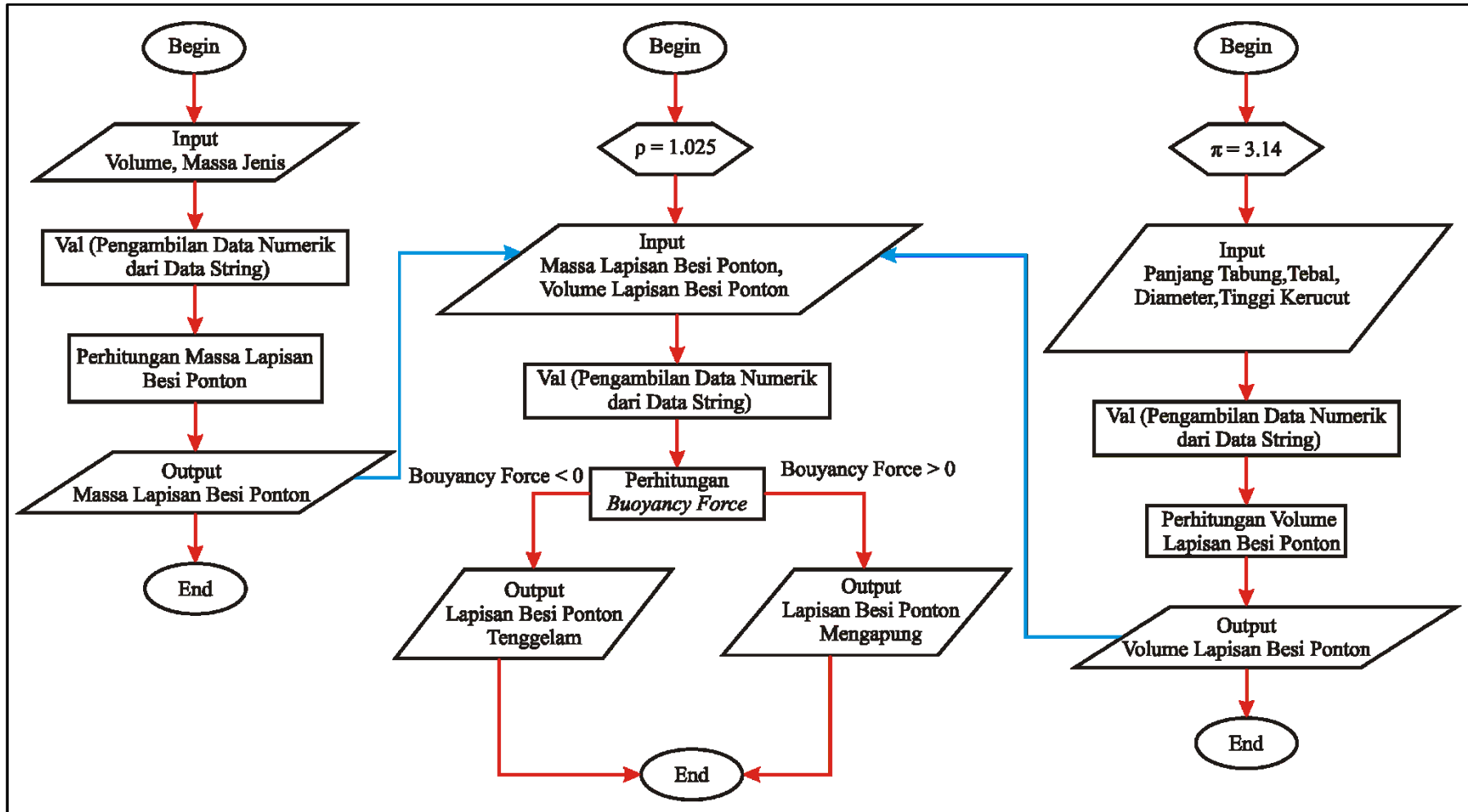
Setelah data hasil pengukuran komponen rumpon berhasil dikumpulkan, maka tahap selanjutnya adalah membuat aplikasi sederhana untuk menghitung gaya-gaya yang bekerja pada rumpon dengan dasar pemograman *Delphi*. Pada proses pembuatan aplikasi diawali dengan menuliskan notasi algoritma dengan menggunakan *flowchart* (diagram alir). Berikut merupakan notasi-notasi yang sering digunakan dalam *flowchart* beserta maknanya.

Tabel 7 Notasi *Flowchart*

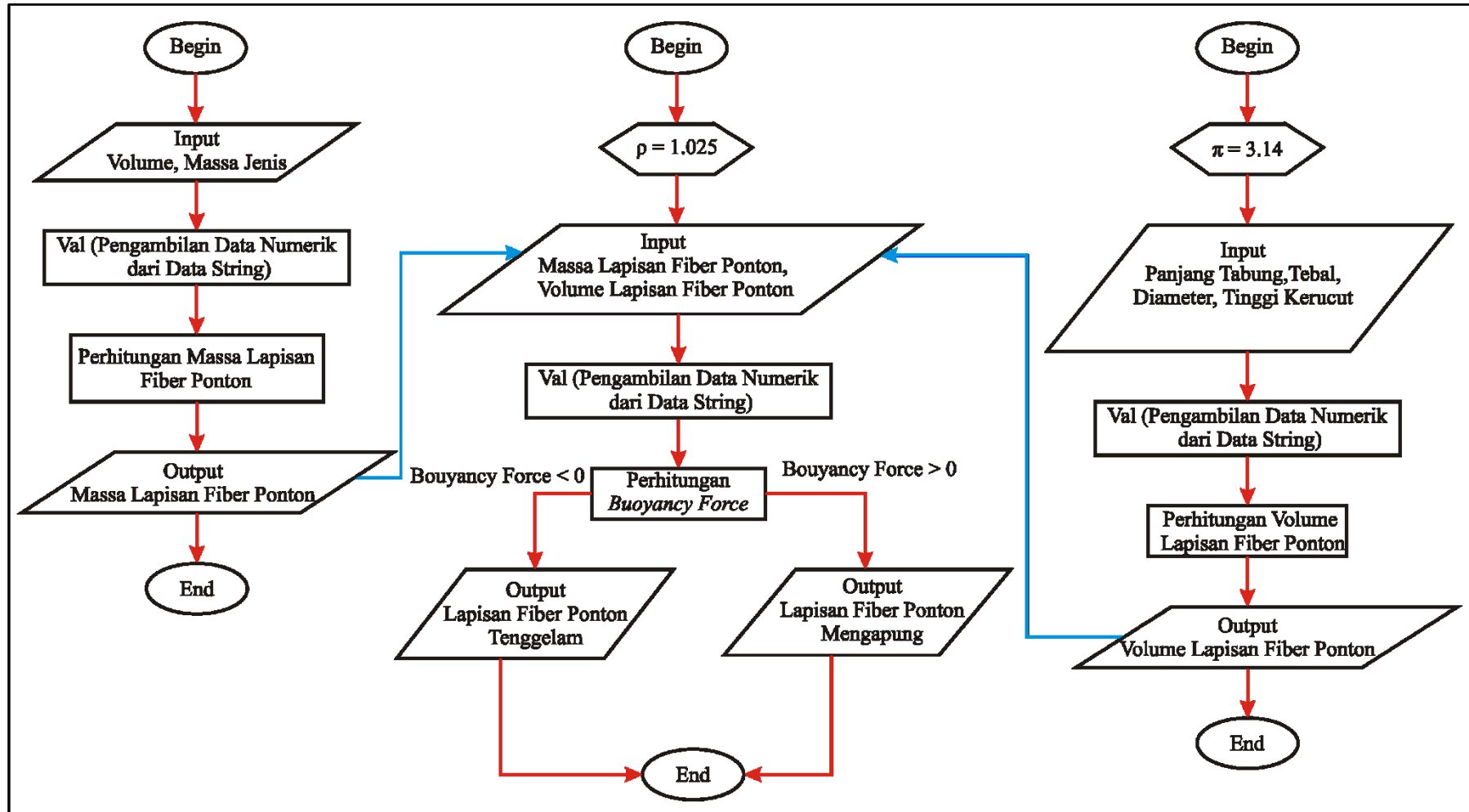
Notasi	Makna
	Menyatakan awal dimulainya suatu algoritma (<i>Begin</i>) dan menyatakan akhir dari suatu algoritma (<i>End</i>)
	Menyatakan proses yang ada dalam algoritma
	Menyatakan pemilihan terhadap suatu keputusan
	Menyatakan bentuk data, baik yang berperan sebagai <i>input</i> maupun <i>output</i>
	Menyatakan <i>predefined proces</i>
	Menyatakan Suatu bentuk dokumen
	Menyatakan langkah persiapan, menunjukkan nilai awal dari suatu variabel yang digunakan dalam algoritma

Sumber : Penuntun Praktikum Penerapan Komputer

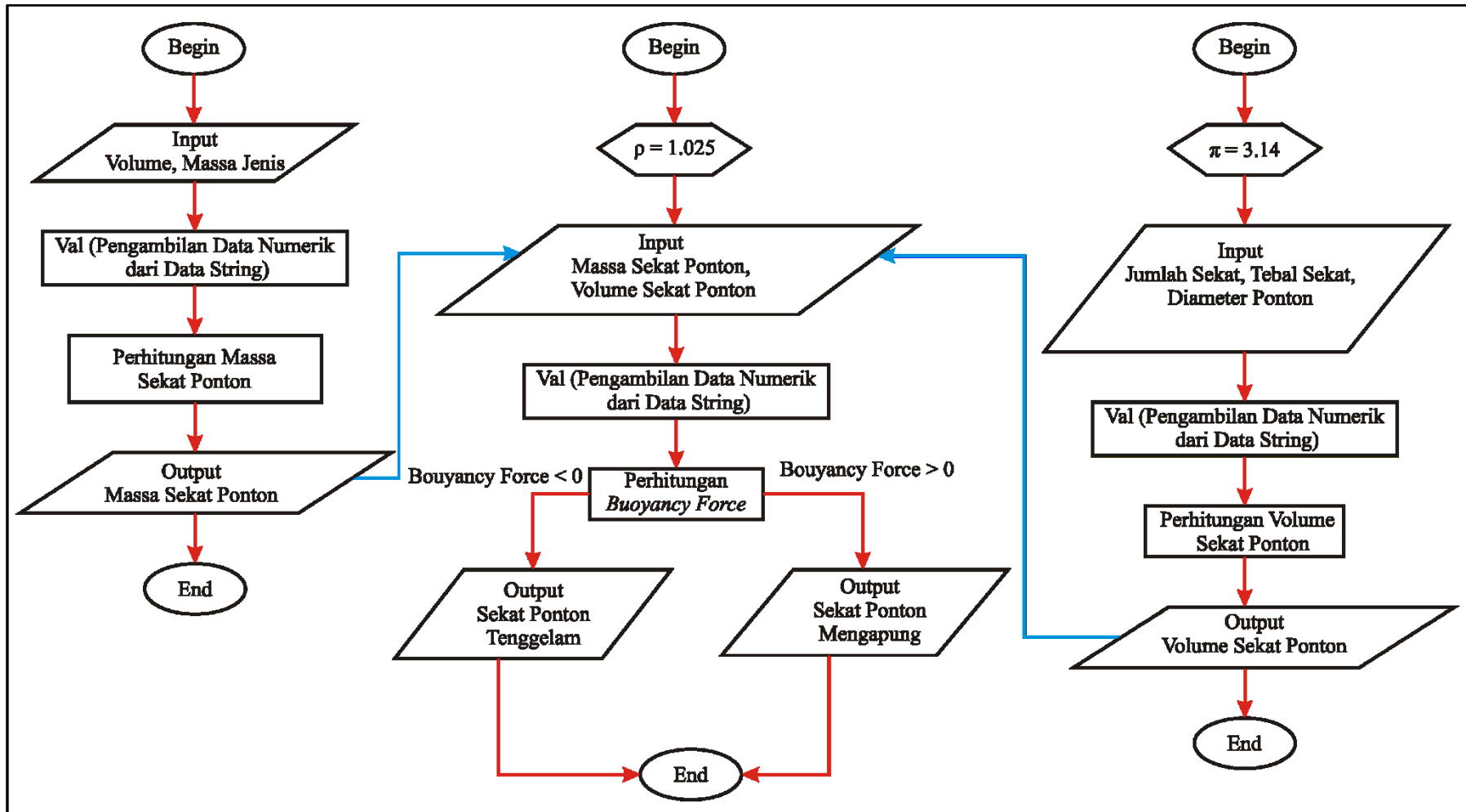
Berikut merupakan algoritma aplikasi perhitungan komponen rumpon yang dinotasikan dalam bentuk *flowchart*.



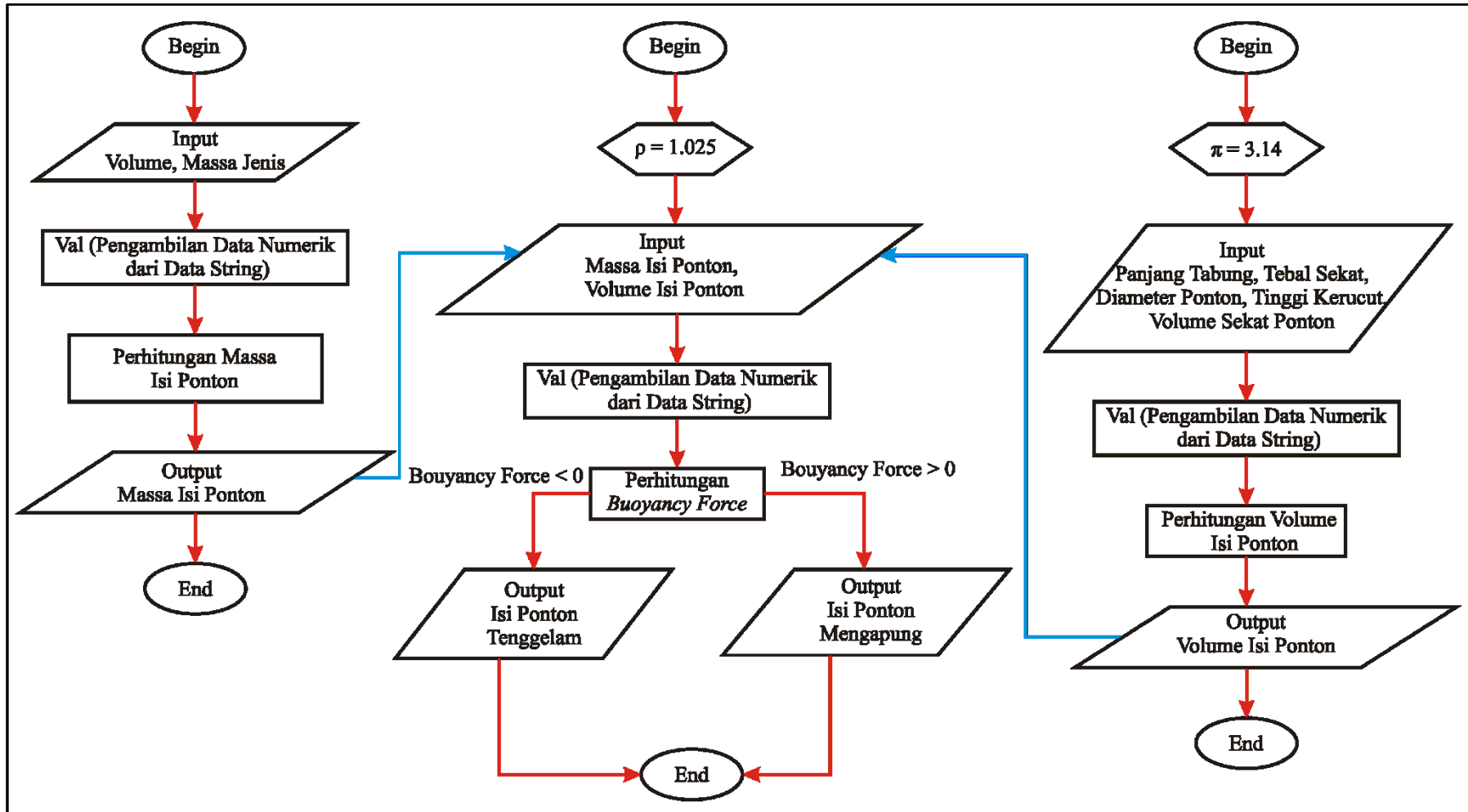
Gambar 20 Flowchart perhitungan lapisan besi ponton.



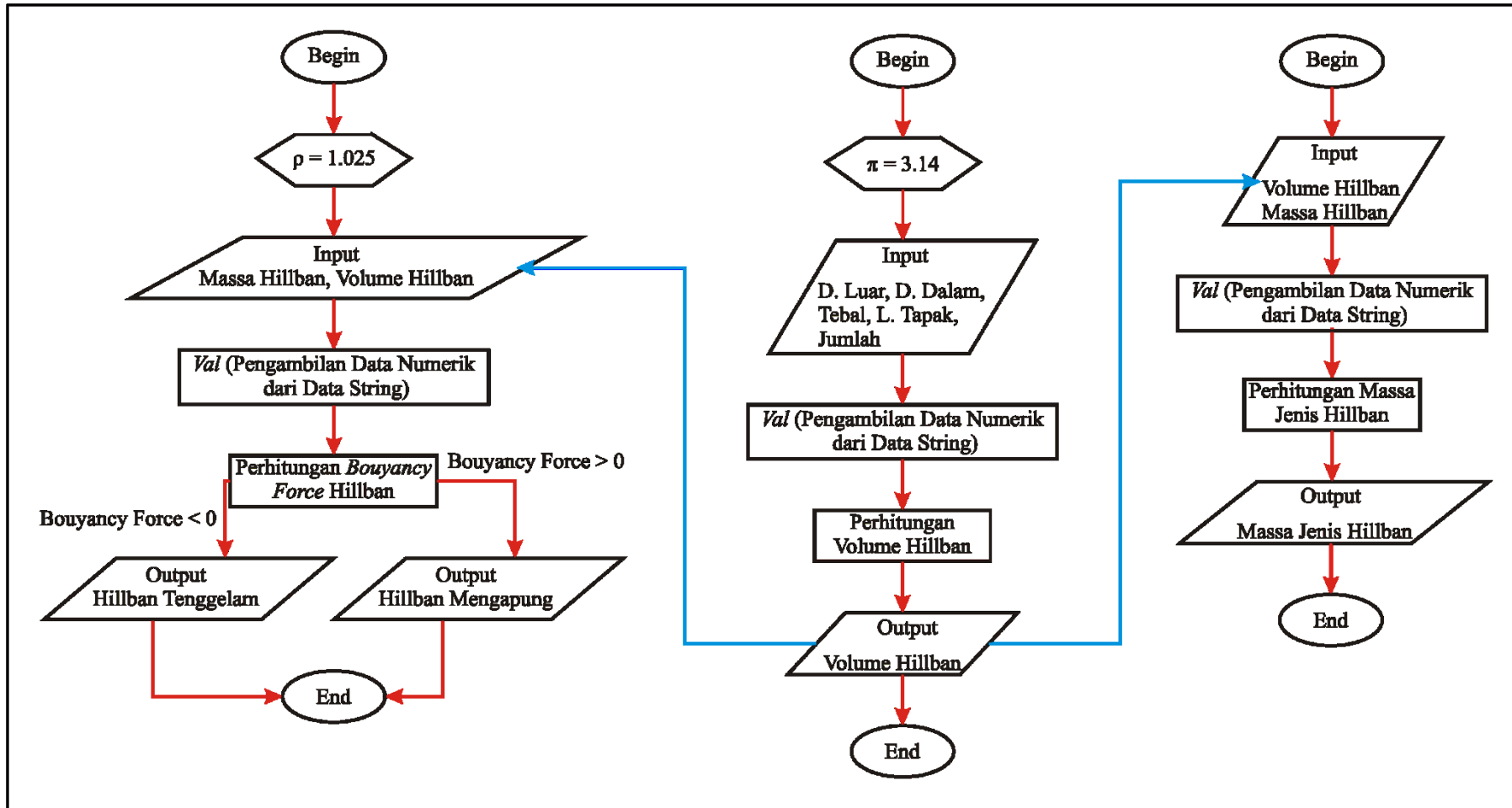
Gambar 21 Flowchart perhitungan lapisan fiber ponton.



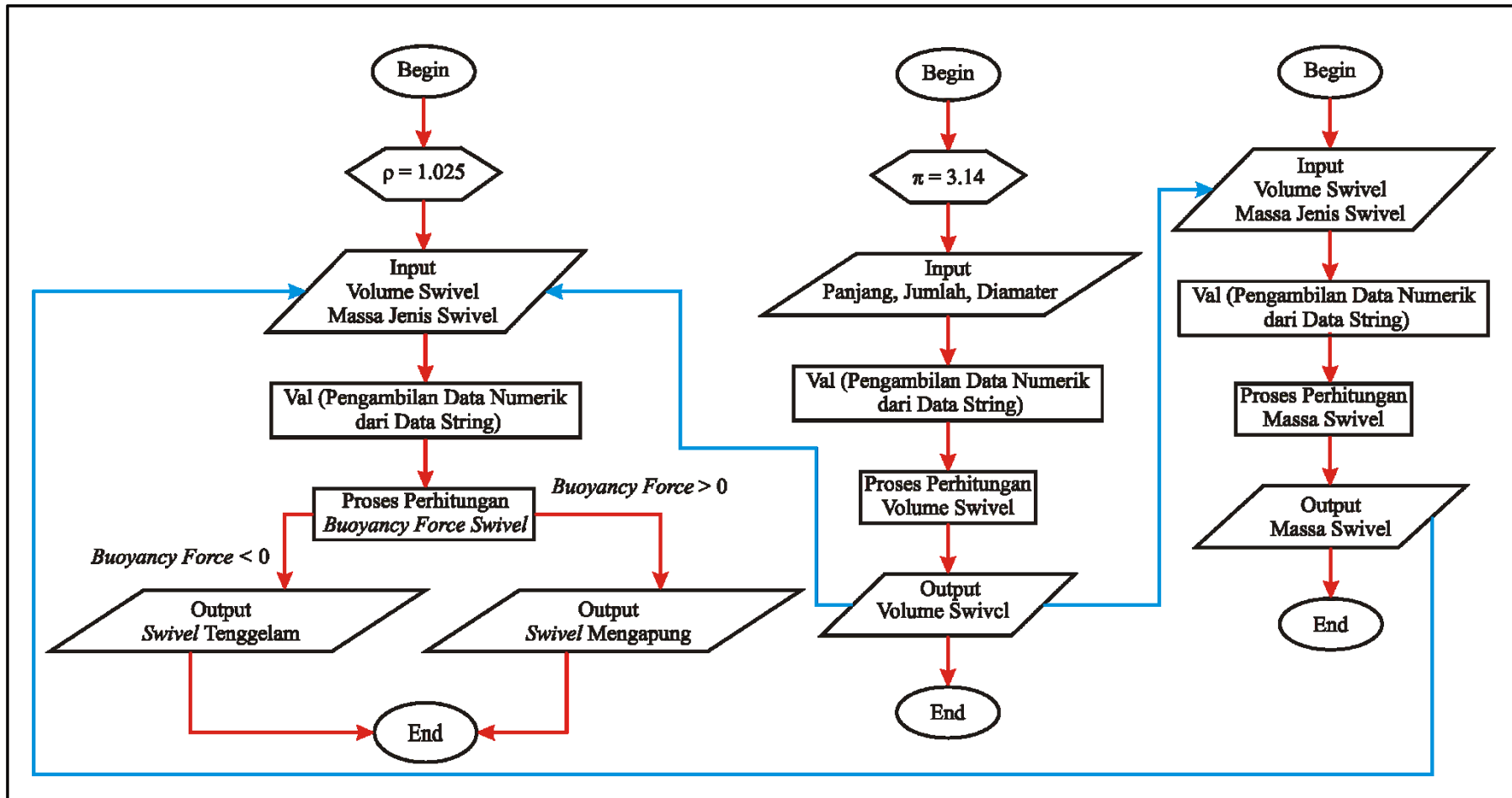
Gambar 22 Flowchart perhitungan sekat pontoon.



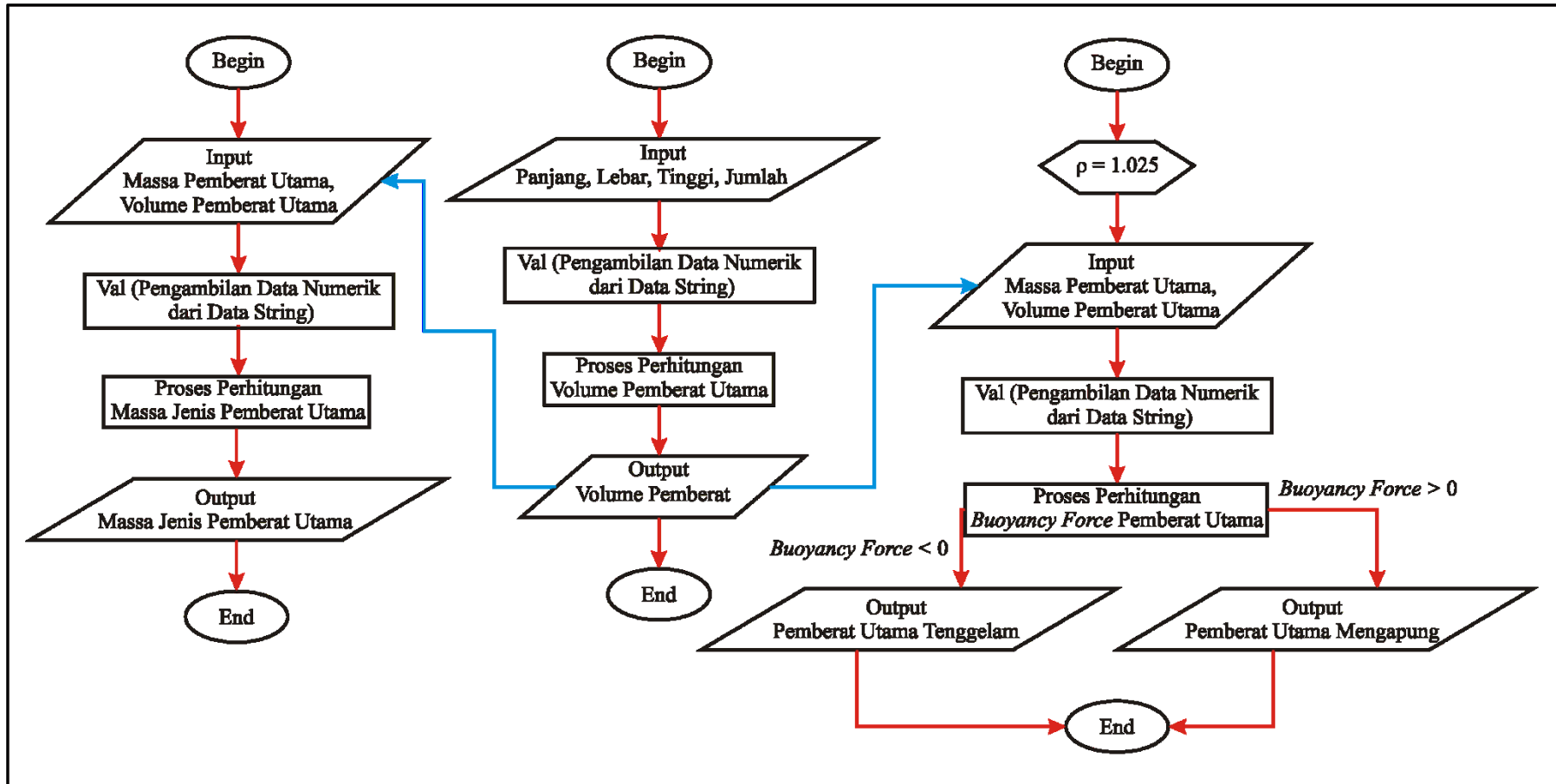
Gambar 23 Flowchart perhitungan isi ponton.



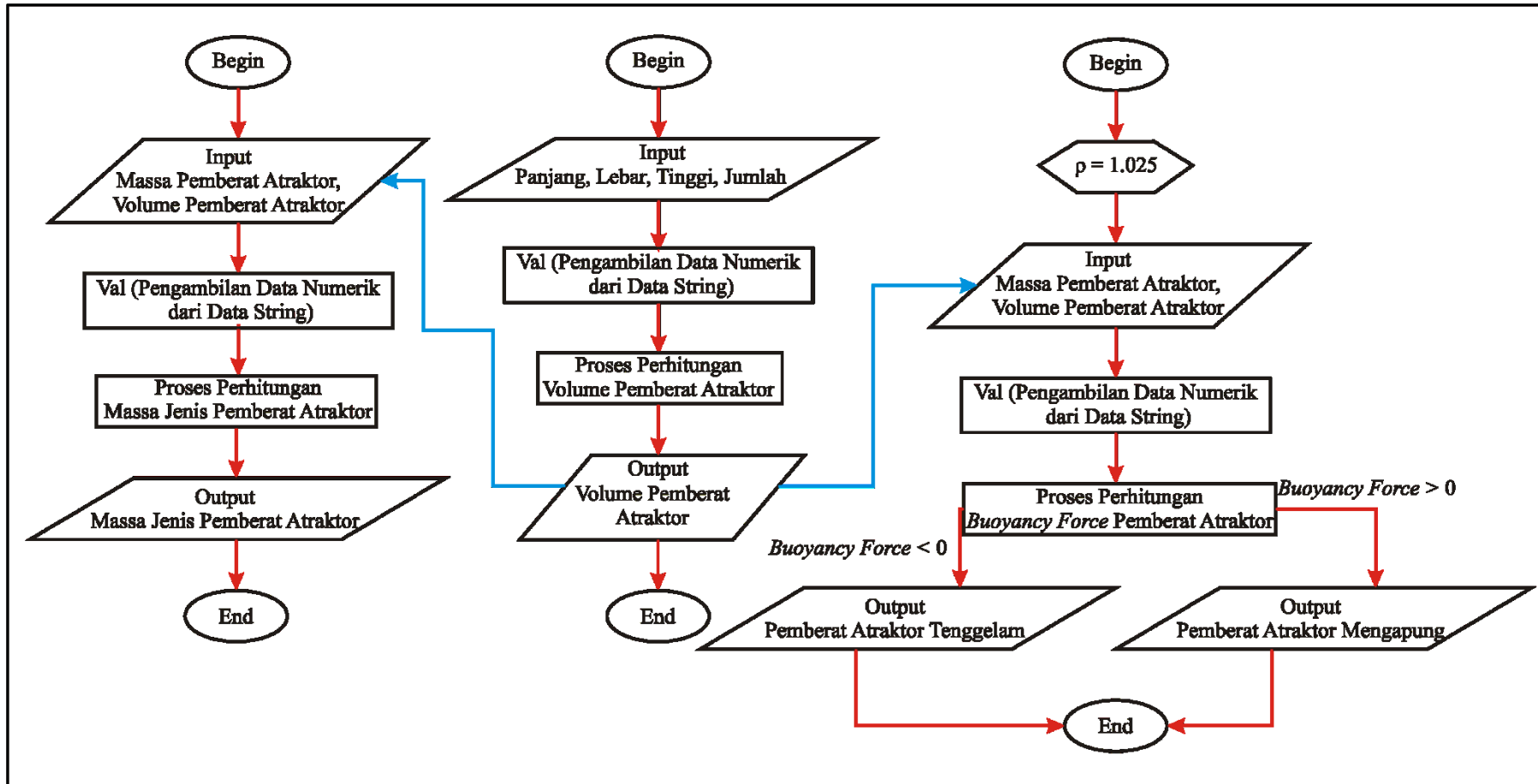
Gambar 24 Flowchart perhitungan komponen hillban.



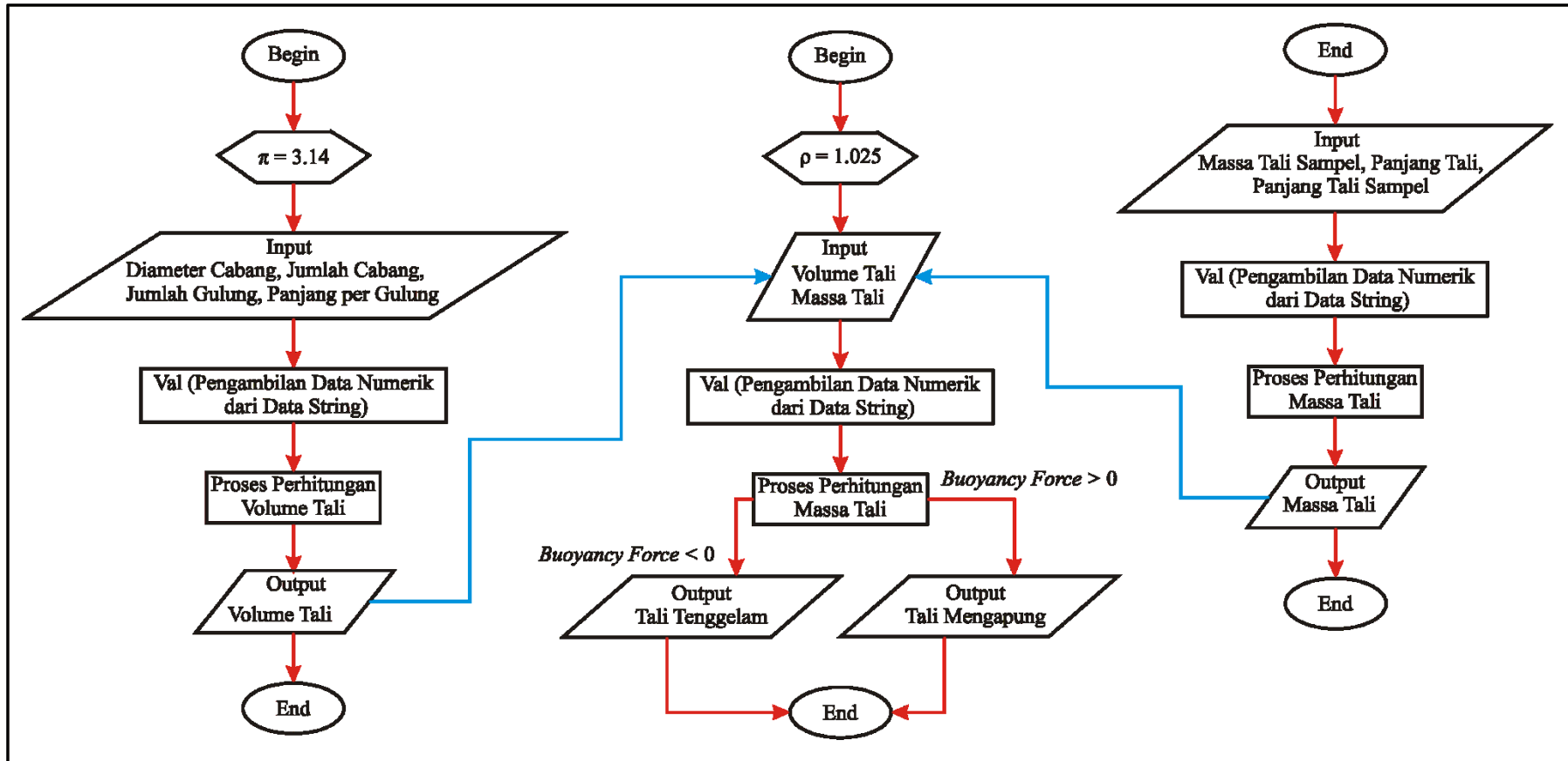
Gambar 25 Flowchart perhitungan komponen swivel.



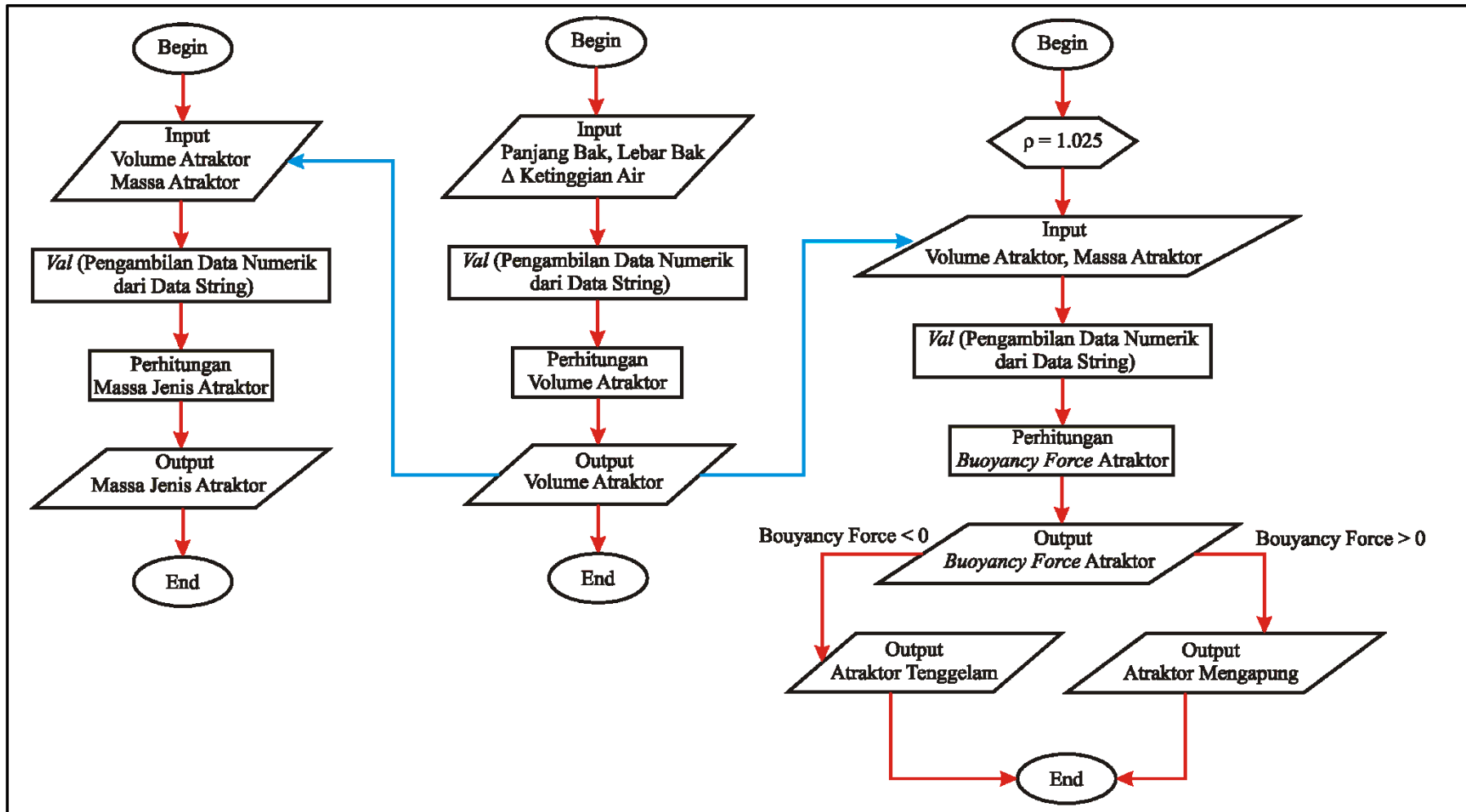
Gambar 26 Flowchart perhitungan komponen pemberat utama.



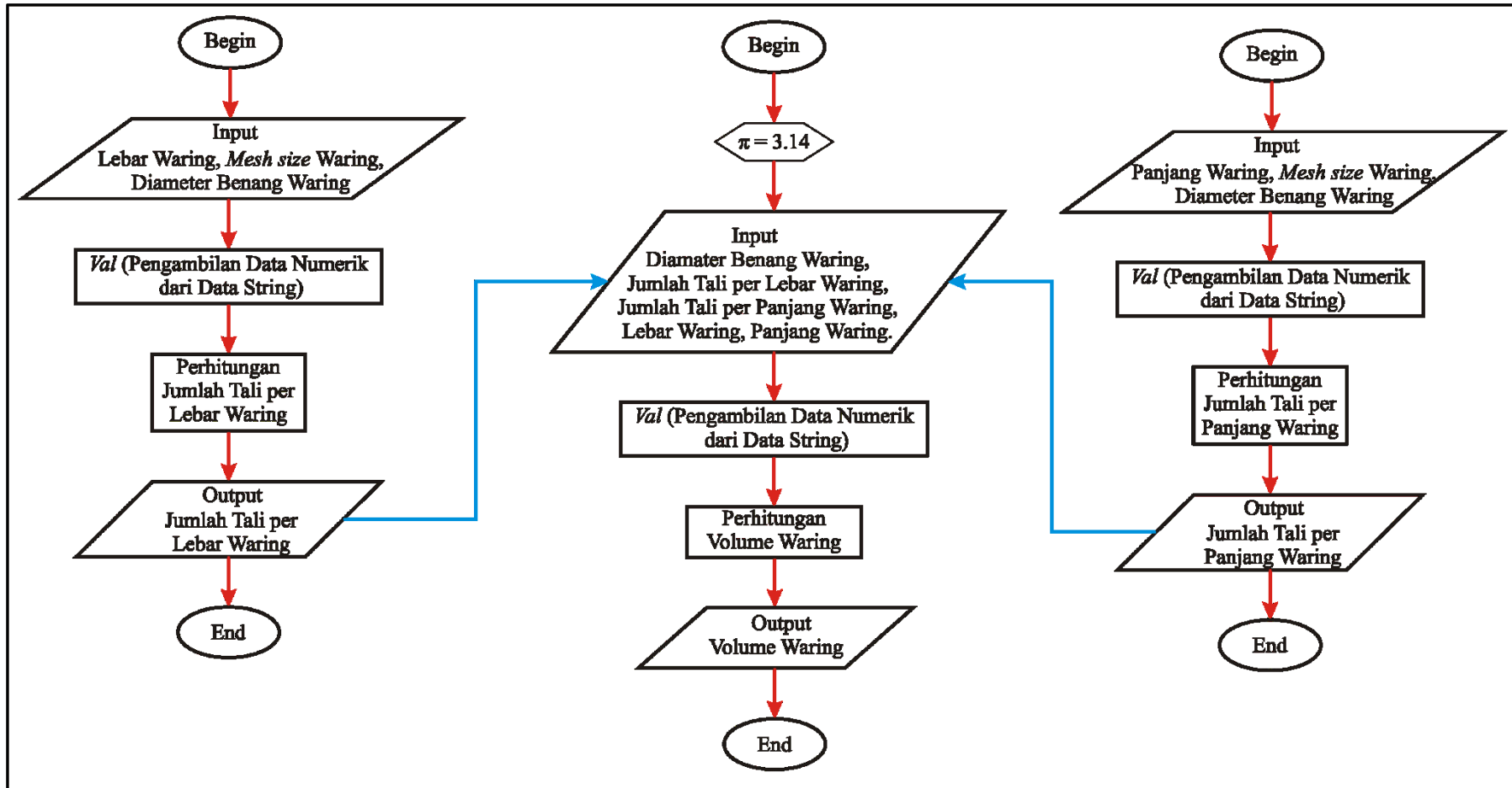
Gambar 27 Flowchart perhitungan komponen pemberat atraktor.



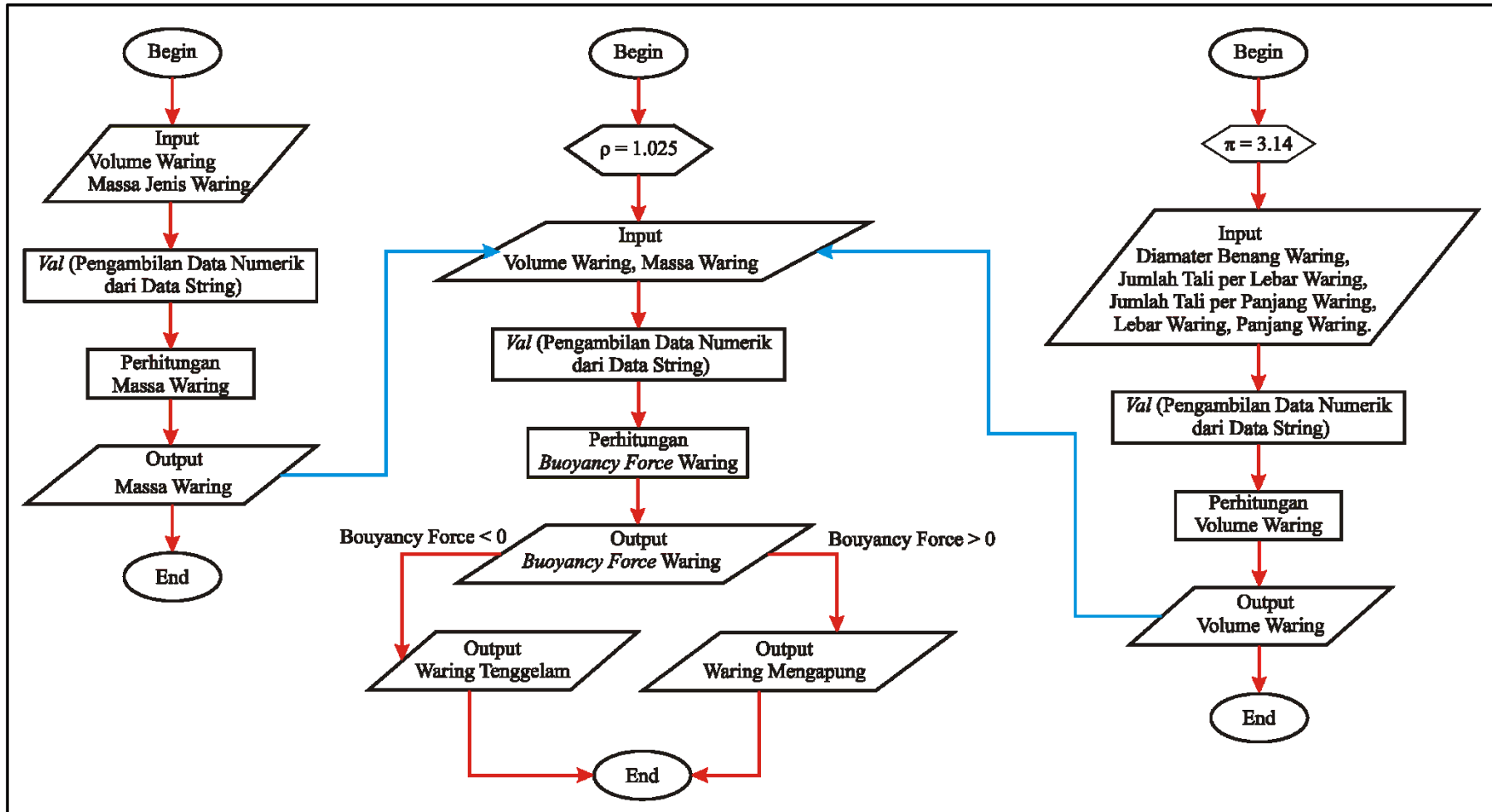
Gambar 28 Flowchart perhitungan komponen tali utama.



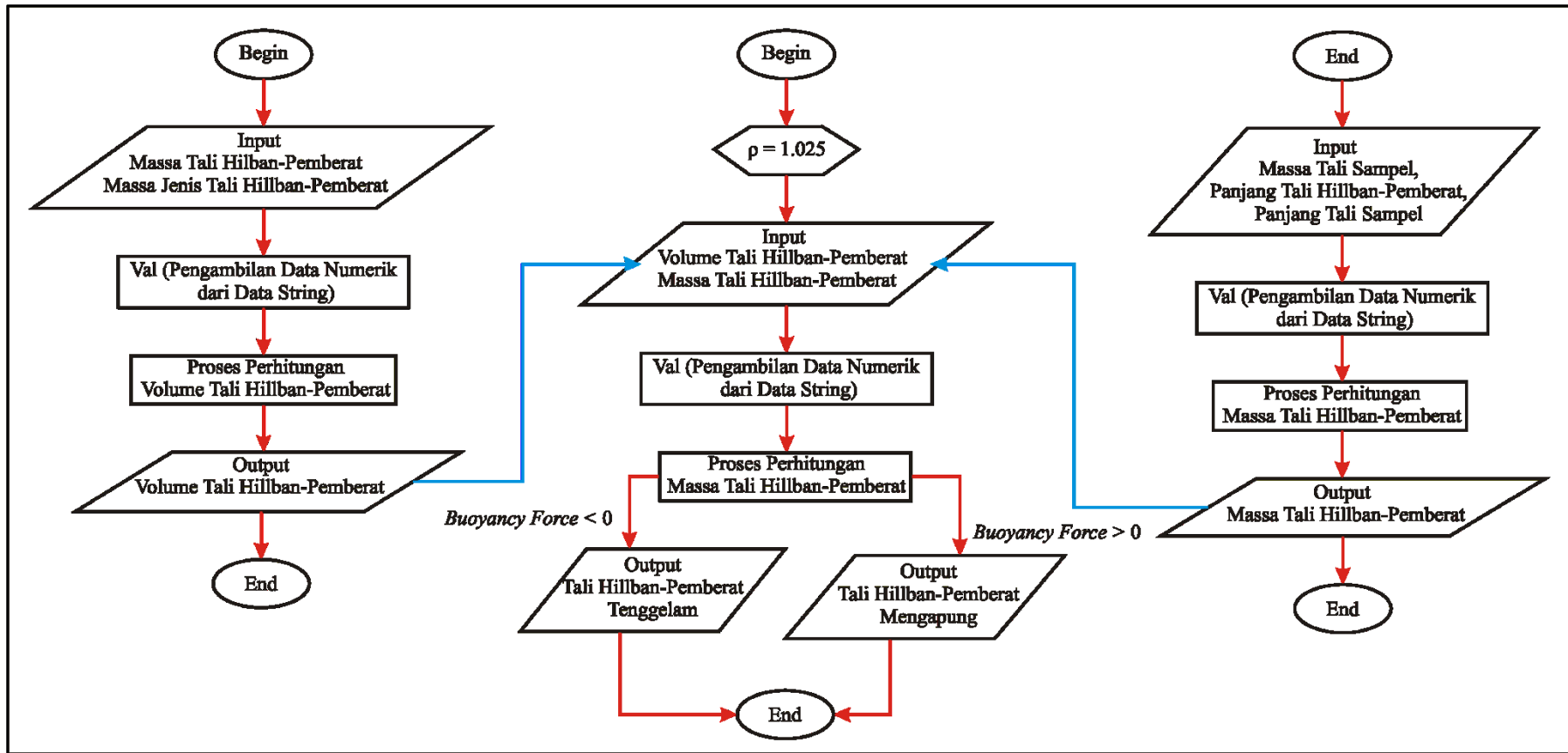
Gambar 29 Flowchart perhitungan komponen atraktor.



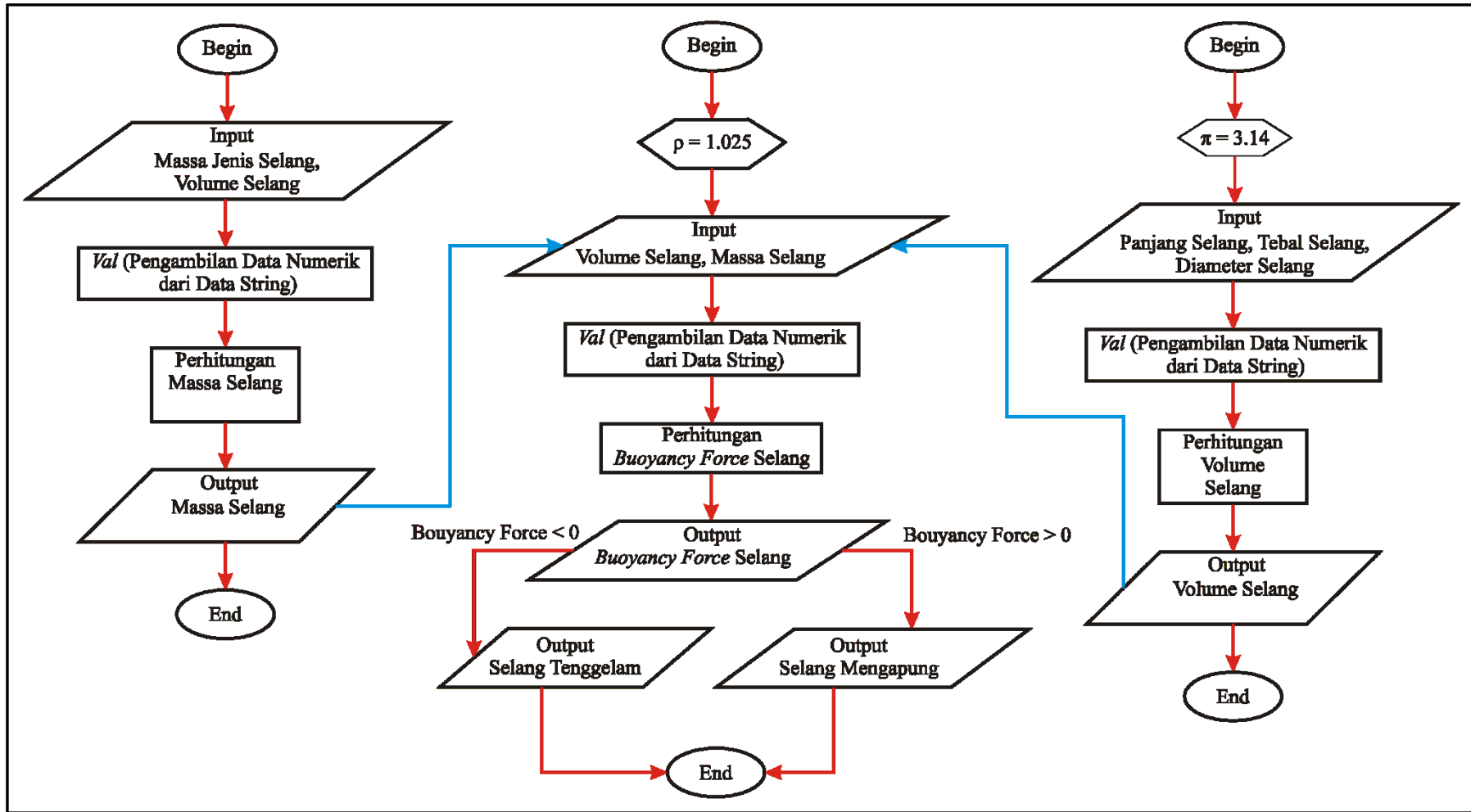
Gambar 30 Flowchart perhitungan volume waring.



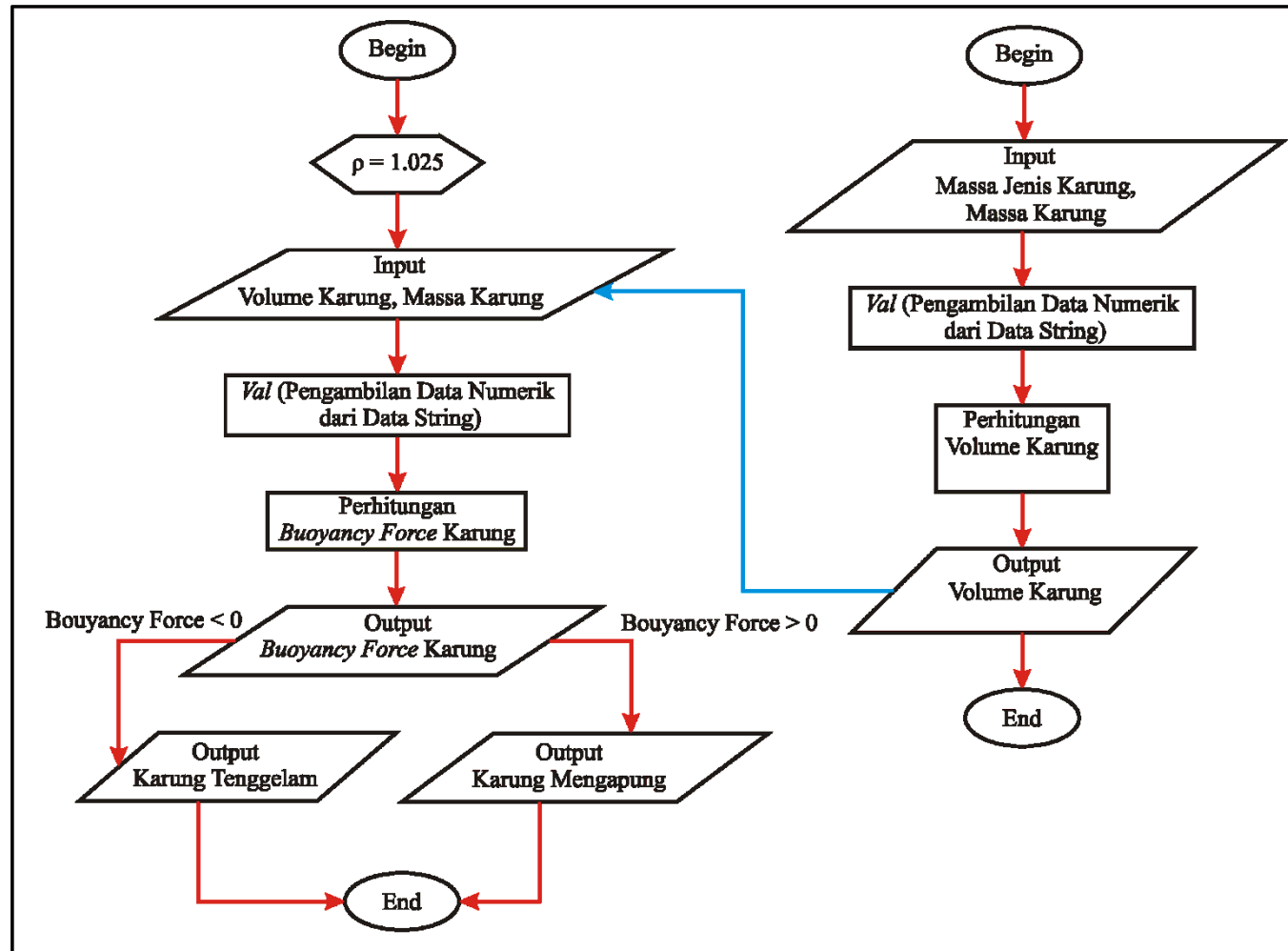
Gambar 31 Flowchart perhitungan komponen waring.



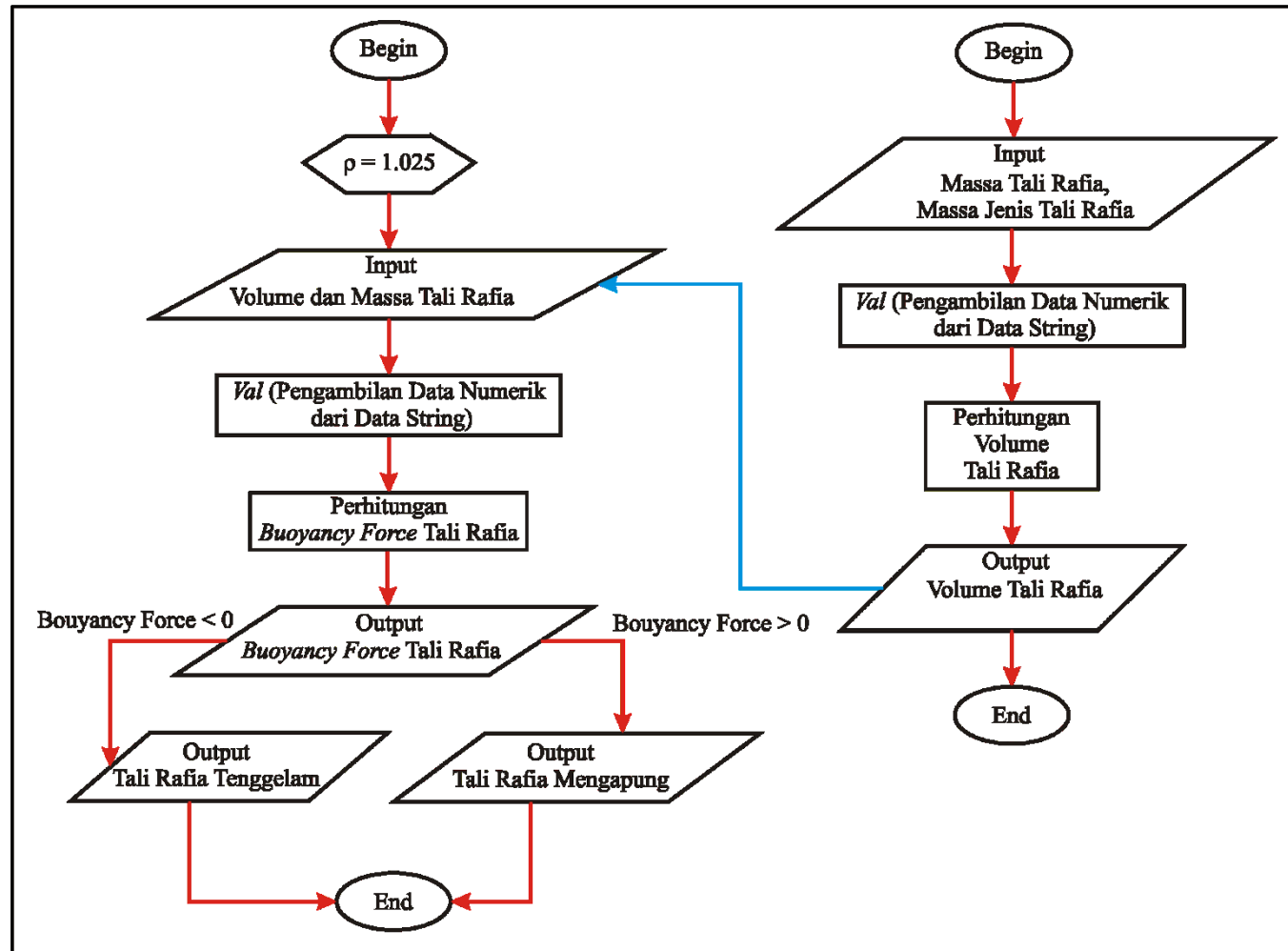
Gambar 32 Flowchart perhitungan komponen tali hillban-pemberat.



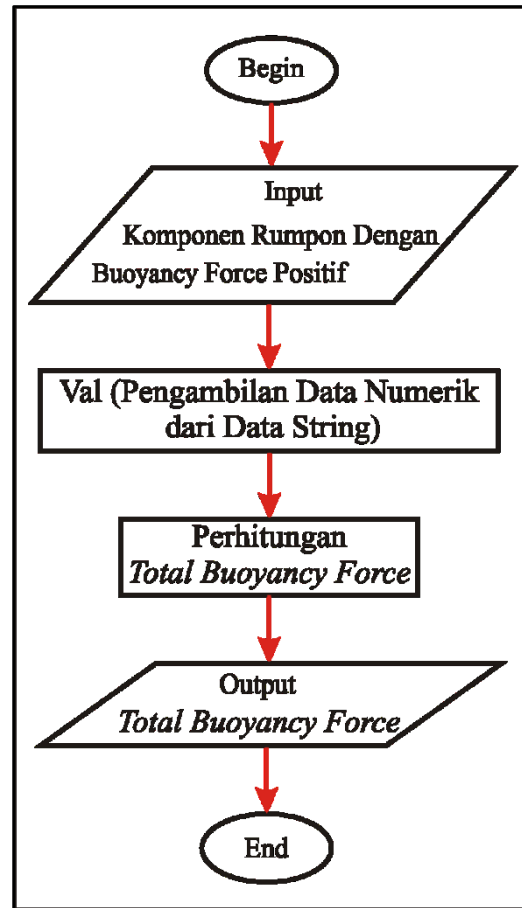
Gambar 33 Flowchart perhitungan komponen selang.



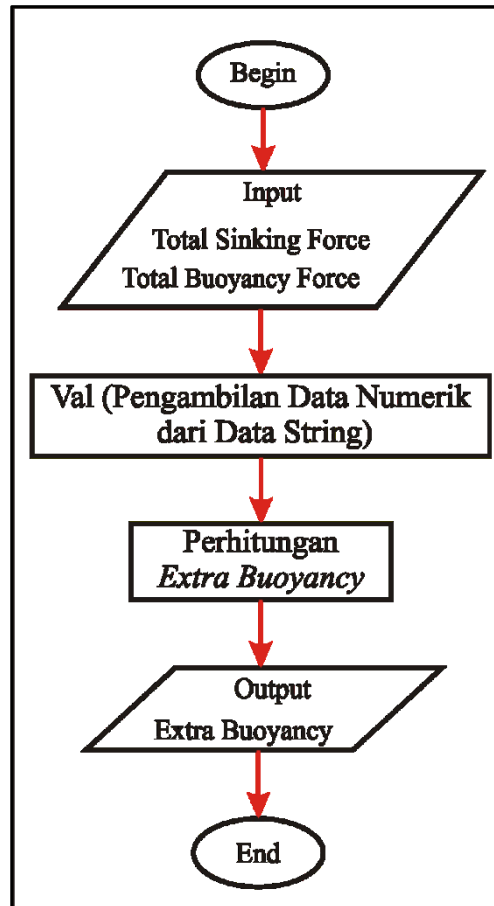
Gambar 34 *Flowchart* perhitungan komponen karung.



Gambar 35 Flowchart perhitungan komponen tali rafia.



Gambar 36 *Flowchart* perhitungan *total buoyancy force*.



Gambar 38 *Flowchart* perhitungan *extra buoyancy*.

hujan berlangsung dari bulan November hingga April, dengan 1.662 mm (71%) dari curah hujan bulanannya mencapai 192 mm.

- b. Temperatur dan Kelembaban Udara; temperatur rata-rata bulanan berkisar antara 25,8-28,8°C dengan kelembaban tertinggi terjadi pada bulan Oktober hingga Maret.
- c. Kecepatan Angin; kawasan Palabuhanratu dan sekitarnya mempunyai musim *Monsoon Climate* dan pola angin yang dipengaruhi oleh musim Barat dan musim Timur. Secara umum angin biasanya berhembus ke arah Barat-Barat Daya selama musim Timur, selama periode ini angin biasanya sangat kencang dengan kecepatan hingga 20 m/detik. Pada musim Barat angin berhembus ke arah Timur-Tenggara, selama periode ini dan selama waktu transisi kecepatan angin bervariasi dari lemah sampai sedang dan jarang mencapai kecepatan 10 m/detik. Secara keseluruhan angin dominan bertiup dari Tenggara (22,6%) dan Barat (13,6%). Untuk lebih jelasnya mengenai arah tiupan angin di Teluk Palabuhanratu yang terjadi selama satu tahun dapat dilihat pada Tabel 8 berikut.

Tabel 8 Arah Angin Berdasarkan Bulan di Wilayah Teluk Palabuhanratu.

Bulan	Arah Tiupan Angin
Januari	Dari Barat dan Barat Laut
Februari	Dari Barat Laut
Maret	Dari Tenggara
April	Dari Tenggara
Mei	Dari Tenggara
Juni	Dari Tenggara
Juli	Dari Tenggara
Agustus	Dari Tenggara
September	Dari Tenggara
Oktober	Dari Tenggara
November	Dari Tenggara dan Barat
Desember	Dari Barat Laut

Sumber: Badan Lingkungan Hidup Kabupaten Sukabumi, 2003

4.3 Geologi

Secara fisiografi wilayah pesisir Palabuhanratu merupakan dataran pantai yang berada pada muara Sungai Cimandiri, Sungai Civalabuan-Cigangsa, Sungai Citepus,

Sungai Sukawayana, Sungai Cimaja, Sungai Cipawenang, Sungai Cisolok, Sungai Citiis, Sungai Cibangan, Sungai Cihaur dan Sungai Cibareno serta dikelilingi oleh Gunung Butak, Gunung Cabe, Gunung Handeuleum, Gunung Gado dan Gunung Habibi. Sedangkan sebelah Utara dan Selatan berbatasan dengan Samudra Hindia. Lahan di daerah lorong perbukitan ditutupi oleh hutan, perkebunan dan lahan pertanian, sedangkan dataran dan lembah sungai banyak dipergunakan untuk persawahan, pemukiman serta pariwisata.

4.4 Hidrologi

- a. Air Tanah; Air tanah dangkal pada umumnya tersebar mengikuti bentuk topografi, di daerah datar air tanah dangkal ini relatif dangkal, sedangkan di daerah perbukitan air tanahnya lebih dalam. Air tanah dangkal ini merupakan sumber air bagi masyarakat setempat. Sedangkan air tanah dalam tidak langsung dipengaruhi oleh curah hujan yang sifatnya lokal. Kedalaman air tanah dalam ini mencapai 50 meter. Air tanah dalam ini hanya dimanfaatkan oleh pengusaha seperti perhotelan (Badan Lingkungan Hidup Kabupaten Sukabumi, 2003).
- b. Air Permukaan; Air permukaan di wilayah pesisir Palabuhanratu bersumber dari sungai yang berjumlah ± 10 sungai yang bermuara langsung ke laut.

4.5 Oseanografi

- a. Oksigen Terlarut
Dari laporan hasil penelitian dan analisa Badan Lingkungan Hidup Kabupaten Sukabumi (2003) menunjukkan bahwa oksigen terlarut rata-rata di wilayah pesisir Teluk Palabuhanratu berkisar antara 12,0 – 12,2 mg/l.
- b. Salinitas
Salinitas di perairan Teluk Palabuhanratu dipengaruhi oleh keadaan musim dengan faktor utama adanya masukan massa air sungai yang bermuara. Transpor massa air sungai yang terutama pada musim barat mengakibatkan turunnya salinitas perairan pantai Teluk Palabuhanratu.

Namun demikian di perairan teluk bagian tengah nilai perbedaan salinitas permukaan laut pada musim timur dan musim barat relatif kecil. Hasil pengukuran memperlihatkan nilai salinitas rata-rata pada periode Agustus Oktober dan Mei-Juli masing-masing sebesar 32,96‰ dan 32,33‰ (Pariwono et al., 1988).

c. Suhu Perairan

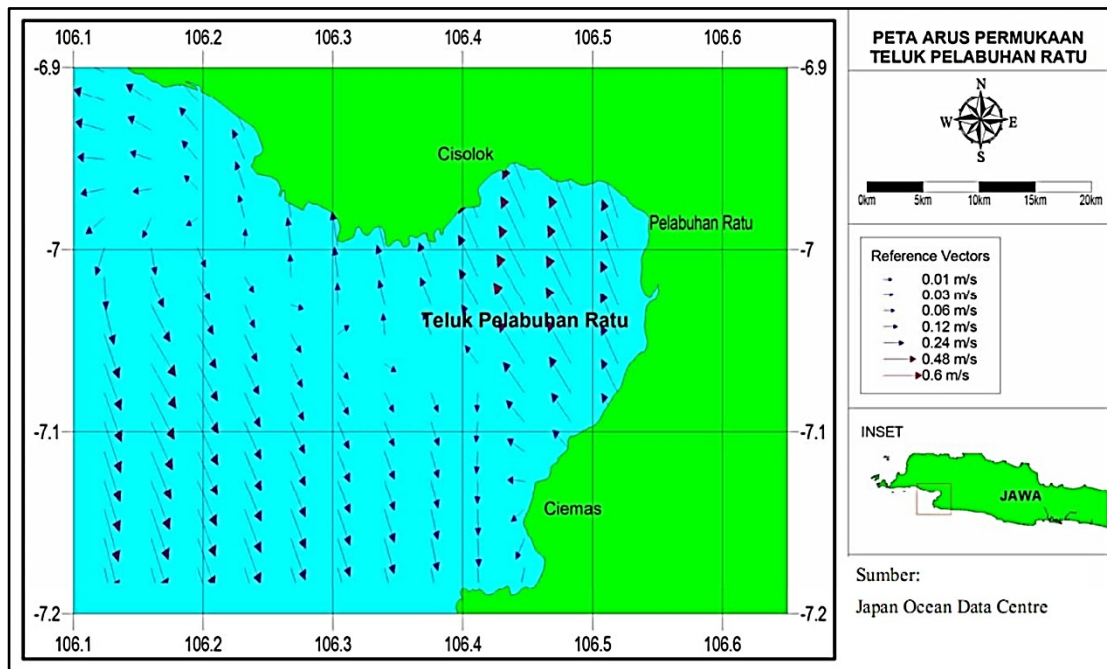
Rata-rata perubahan suhu perairan di wilayah pesisir Teluk Palabuhanratu berkisar antara 28-29°C. Perubahan suhu rata-rata di dekat pantai berkisar antara 28,1-28,6°C, sedangkan suhu di lepas pantai berkisar antara 28,24-28,7°C (Badan Lingkungan Hidup Kabupaten Sukabumi, 2003).

d. pH

pH di perairan Teluk Palabuhanratu berkisar antara 7,00-8, (Pariwono et al., 1988). Jika dibandingkan dengan baku mutu pH perairan untuk biota laut berdasarkan Kep-51/MENKLH/2004, nilai pH yang terukur masih berada dalam kisaran yang diinginkan yaitu 6,50- 8,50. Hal ini mengindikasikan bahwa kualitas perairan ditinjau dari segi pH dapat dikatakan baik.

e. Kecepatan Arus

Arus pantai dapat terjadi karena gelombang yang datang menuju pantai, dan hal ini mempengaruhi proses sedimentasi dan atrofi pantai. Pola arus pantai ini ditentukan oleh besarnya sudut yang dibentuk antara gelombang yang datang dengan garis pantai. Jika sudut datang cukup besar, maka akan terbentuk arus menyusur pantai (*longshore current*) yang disebabkan oleh perbedaan tekanan hidrostatis. Berdasarkan hasil penelitian geologi kelautan diperairan pesisir Palabuhanratu, arus permukaan dekat pantai (*nearshore current*) bergerak ke Timur Laut mulai dari daerah Karang Hawu sampai daerah Tanjung Karang dan berbelok ke Barat Laut mulai daerah Tanjung Pamipiran sampai daerah Palabuhanratu serta arus berbelok lagi ke Timur Barat melalui daerah Citepus. Arus permukaan dekat pantai pada umumnya memperlihatkan pola pergerakan arus Barat Daya-Timur Laut dengan kecepatan rata-rata 0,4 m/det (Badan Lingkungan Hidup Kabupaten Sukabumi, 2003).



Gambar 39 Arus permukaan Teluk Palabuhanratu.

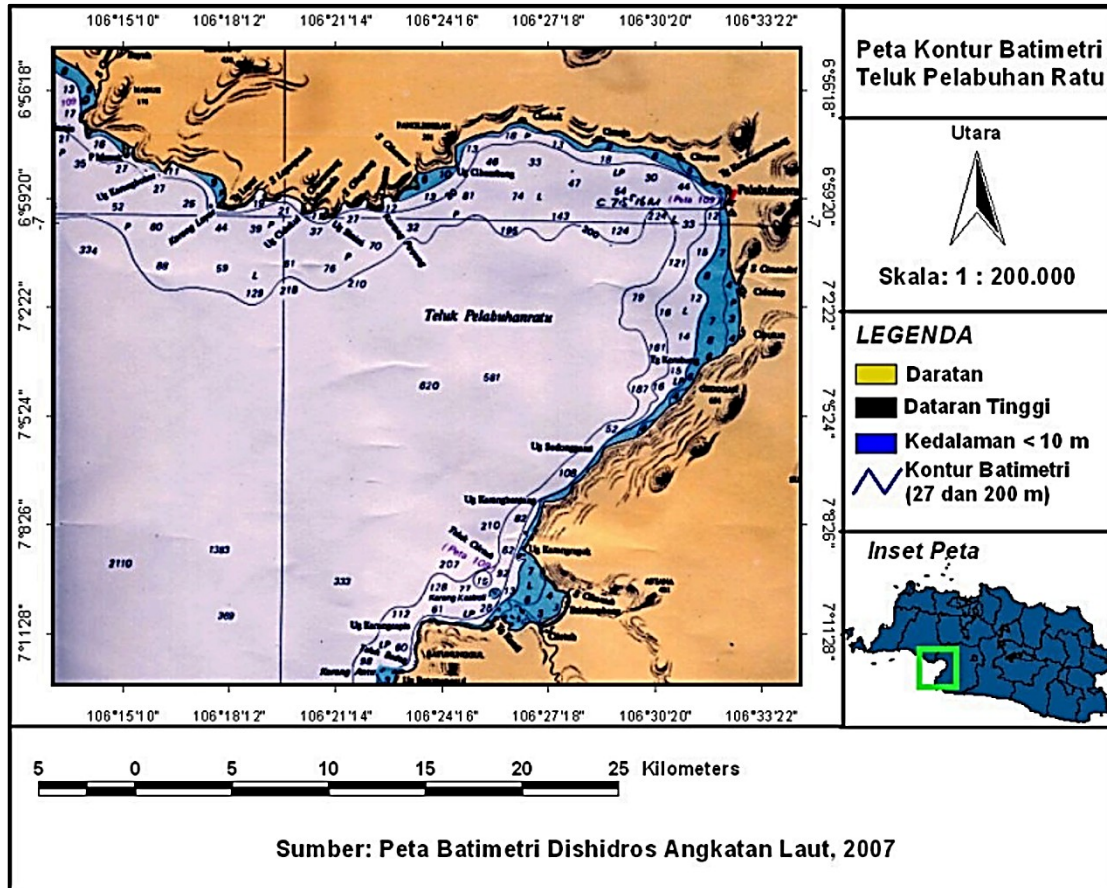
f. Pasang-Surut

Pasang Surut adalah proses naik-turunnya muka air laut diakibatkan oleh pengaruh gravitasi benda-benda angkasa, terutama bulan dan matahari. Pasang-Surut mempengaruhi arus dan sirkulasi perairan, terutama di perairan semi tertutup seperti selat dan teluk. Pasang-surut di perairan pesisir Palabuhanratu bertipe campuran dengan unsur ganda lebih menonjol dengan bilangan $E = 0,25$. Hal ini menunjukkan bahwa perairan pesisir Palabuhanratu pada umumnya mengalami dua kali pasang dan dua kali surut setiap harinya dengan ketinggian yang berbeda. Kedudukan air terendah adalah 90 cm dan kedudukan air tertinggi mencapai 249 cm dengan tunggangan airnya adalah 159 cm (Badan Lingkungan Hidup Kabupaten Sukabumi, 2003).

g. Batimetri

Dengan batas 250 meter kearah laut, kedalaman wilayah pesisir Palabuhanratu rata-rata berkisar antara 0-50 meter, pada kedalaman 10 meter dicapai pada

jarak 50-100 meter, kedalaman 25 meter dicapai pada jarak 100-150 meter dari garis pantai ke arah laut (Badan Lingkungan Hidup Kabupaten Sukabumi, 2003).



Gambar 40 Peta kontur batimetri Teluk Palabuhanratu.

h. Gelombang

Gelombang yang terbentuk pada umumnya disebabkan oleh adanya proses alih energi dari angin menuju permukaan laut. Gelombang ini merambat ke segala arah membawa energi yang kemudian dilepaskan ke pantai dalam bentuk hampasan ombak (*breakers*). Tinggi gelombang di Teluk Palabuhanratu dapat berkisar antara 1-3 meter (Pariwono et al., 1988).

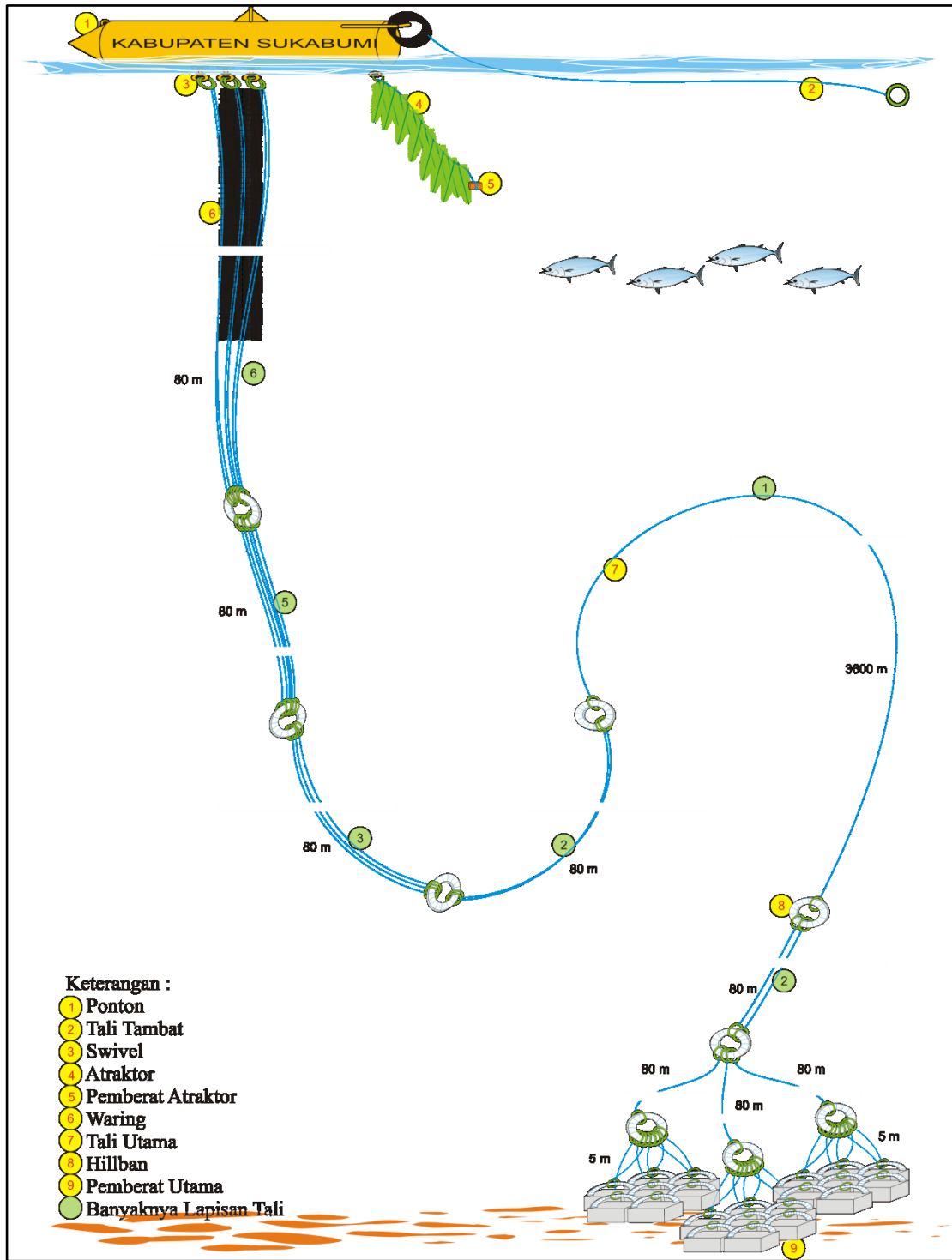
5. HASIL DAN PEMBAHASAN

5.1 Rumpon

Rumpon adalah suatu bangunan menyerupai pepohonan yang dipasang di suatu tempat di tengah laut. Disebut alat bantu penangkapan, karena fungsinya hanya sebagai pembantu, yaitu untuk mengumpulkan ikan pada suatu titik atau tempat untuk kemudian dilakukan operasi penangkapan ikan. Pada prinsipnya, rumpon terdiri dari empat komponen utama, yaitu pelampung, tali panjang, atraktor, dan pemberat (Subani dan Barus, 1989). Namun, konstruksi secara detail dan lengkapnya, serta material yang digunakan berbeda signifikan antar daerah (Monintja, 1992).

Menurut hasil pengamatan di lokasi penelitian, rumpon yang digunakan oleh nelayan Palabuhanratu memiliki komponen utama yang terdiri dari pelampung yang terbuat dari ponton, tali utama yang terbuat dari tali PP (*polypropylene*), atraktor alami yang terbuat dari pelepah daun kelapa, sedangkan atraktor permanennya terbuat dari bahan sintetis yang terbuat dari waring bekas dan tali plastik rafia, serta pemberat utama yang terbuat dari campuran semen. Komponen pelengkap rumpon terdiri dari *swivel* yang terbuat dari besi, hillban yang terbuat dari ban luar mobil bekas, karung plastik, selang plastik, tali PE, dan pemberat ataktor yang terbuat dari batu bata merah.

Berdasarkan posisi dari pemikat, rumpon ini dikategorikan sebagai rumpon perairan permukaan lapisan tengah perairan dalam. Berdasarkan kriteria permanensi, rumpon ini dikelompokkan menjadi rumpon yang dijangkar secara tetap (statis). Berdasarkan teknologi yang digunakan, rumpon ini dikelompokkan menjadi rumpon modern. Hal ini berdasarkan material bahan pembuat rumpon yang menggunakan bahan sintetis, seperti pelampung yang terbuat dari ponton besi, tali utama yang terbuat dari *polypropylene*, pemberat yang terbuat dari campuran semen, dan atraktor yang dikombinasikan antara bahan alami dengan bahan sintetis berupa waring dan tali plastik. Rumpon ini masih bisa dikategorikan sebagai rumpon modern walaupun tidak menggunakan *radar reflectors* atau *solarpowered light*.



Gambar 41 Desain dan konstruksi rumpon.

Berbeda halnya dengan rumpon laut dalam di Kabupaten Halmahera Selatan, Maluku Utara (Ardianto, 2005), rumpon ini memiliki komponen utama yang terdiri dari pelampung yang terbuat dari:

1. Ponton besi yang memiliki dimensi panjang 2,44 meter, lebar 1,22, tinggi 0.5 meter.
2. Atraktor sintetis yang terbuat dari irisan ban bekas, yaitu lima buah ban bekas yang dirangkai sepanjang 15 meter. kemudian dirangkai dengan tali timba dengan jarak tiga meter melalui empat sisi yang telah dilubangi. Pada tali timban dan ban diikat tali rafia yang dirumbai-rumbai. Atraktor dan ponton dihubungkan dengan rantai berdiameter 13 milimeter dengan panjang 20 meter. Pemberat atraktor dihubungkan pada atraktor agar tegak lurus kebawah.
3. Tali rumpon yang digunakan memiliki panjang 1,5 kali kedalaman perairan. Terbuat dari tali baja (*wire rope*) dan tali PE atau PP berdiameter 18-24 milimeter. Tali rantai besi yang menghubungkan pontoon, atraktor, dan pemberat antara disambungkan dengan *wire rope* berdiameter 16 milimeter sepanjang 50 meter menggunakan segel dan *swivel* yang ujungnya dibuat *eye spicing*. Selanjutnya kawat kabel baja ini disambungkan lagi dengan tali PE dan disambungkan lagi dengan kawat kabel baja berdiameter 16 milimeter sepanjang 40 meter. Setiap sambungan menggunakan segel dan *swivel* serta dibuatkan *eye spicing*. Untuk menghubungkan dengan pemberat digunakan kawat kabel baja berdiameter 16 milimeter sepanjang lima meter.
4. Pemberat terbuat dari drum ekas yang diisi dengan semen dan diberi pegangan atau kuping dari potongan salah satu sisi ban mobil bekas. Jumlah pemberat enam buah dengan berat masing-masing 100 kilogram. Untuk mengikat pemberat digunakan kawat kabel baja.

Berdasarkan hasil perhitungan terhadap komponen rumpon, diperoleh nilai *buoyancy force* total 2.074.878,24 gr, *sinking force* total 1.131.032,90 gr, dan *extra buoyancy force* 943.845,34 gr atau 45,49 %. Kelebihan *buoyancy force* sebesar 45,49% menandakan bahwa jika panjang tali utama lebih pendek daripada kedalaman perairan tempat rumpon dipasang, maka keadaan rumpon akan tergantung, pemberat

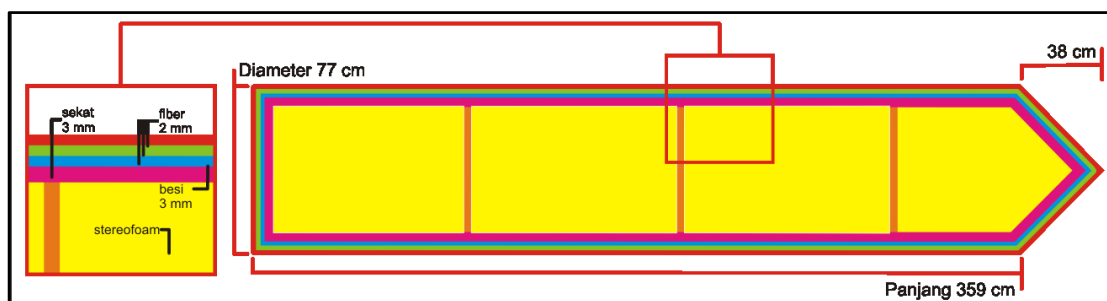
utama terangkat dari dasar perairan. Jika memperhitungkan faktor arus, dengan keadaan rumput yang terangkat, maka rumput dapat berpindah atau bergeser dari tempat semula dipasang.

Dari hasil penelitian yang telah dilakukan, rumput laut dalam di perairan Palabuhanratu memiliki desain dan konstruksi sebagai berikut:

5.1.1 Ponton

Ponton memiliki bentuk tabung silinder dengan ujung meruncing, dengan panjang tabung 359 cm, tinggi kerucut 38 cm, dan diameter 77 cm. Ponton dilapisi dengan tiga lapis fiber dengan ketebalan setiap lapisan fiber sekitar 1 mm, sedangkan untuk ponton terbuat dari baja dengan ketebalan 3 mm. Ponton memiliki dua sekat dengan ketebalan masing-masing sekat sama dengan ketebalan ponton, sehingga memiliki tiga ruang kosong. Tiga ruang kosong tersebut diisi penuh dengan *sterofoam*.

Perhitungan ponton dilakukan pada setiap lapisan. Perhitungan pada lapisan fiber menghasilkan nilai volume fiber sebesar $28.354,75 \text{ cm}^3$ dengan massa fiber sebesar 32.324,41 gr, lapisan besi memiliki volume $27.834,31 \text{ cm}^3$ dan massa 218.777,70 gr, lapisan sekat memiliki volume sebesar $2.663,52 \text{ cm}^3$ dengan massa sekat sebesar 20.935,30 gr, sedangkan lapisan *sterofoam* memiliki volume sebesar $1.645.909,76 \text{ cm}^3$ dengan massa *sterofoam* sebesar 197.509,17 gr.



Gambar 42 Desain dan konstruksi ponton.

Ponton adalah perahu berdasar datar atau pelampung yang dapat digunakan untuk mendukung struktur di atas air. Dapat terbuat dari silinder tertutup yang terbuat pipa atau tong atau fiber untuk pelapis ponton yang terbuat dari logam atau beton. Dapat digunakan untuk mendukung bangunan sederhana, seperti membuat rakit. Sebuah rakit dapat mendukung bangunan menyerupai rumah, salah satu contohnya adalah rumah kapal yang dapat bergherak atau sebuah bangunan seperti dermaga yang dibangun secara permanen (<http://en.wikipedia.org>). Di beberapa daerah di Indonesia, ponton dikenal juga sebagai tongkang yang digunakan untuk alat transportasi menyebrangi sungai (<http://id.wikipedia.org>).

Ponton dipilih sebagai pelampung rumpon karena strukturnya yang kuat sehingga tahan terhadap arus and gelombang yang besar. Selain itu bahan pembuat ponton yaitu pipa besi dapat dibeli dengan mudah. Namun, terdapat kekurangan dari bahan pipa besi ini, yaitu tidak tahan terhadap korosi air laut. oleh karena itu, dibandingkan dengan membeli pipa baja anti karat, nelayan rumpon di Palabuhanratu lebih memilih untuk melapisi ponton dengan serat fiber yang dicat dengan cat anti bocor. Cat anti bocor yang digunakan adalah cat anti bocor yang biasa digunakan sebagai cat anti bocor pada genting rumah. Menurut hasil wawancara, ponton dicat dengan cat anti bocor dengan tiga kali pengecatan. Pemberian tiga lapisan fiber yang dicat anti bocor pada ponton dimaksudkan agar lapisan besi yang merupakan bahan utama ponton tidak cepat mengalami korosi akibat terkena air laut.

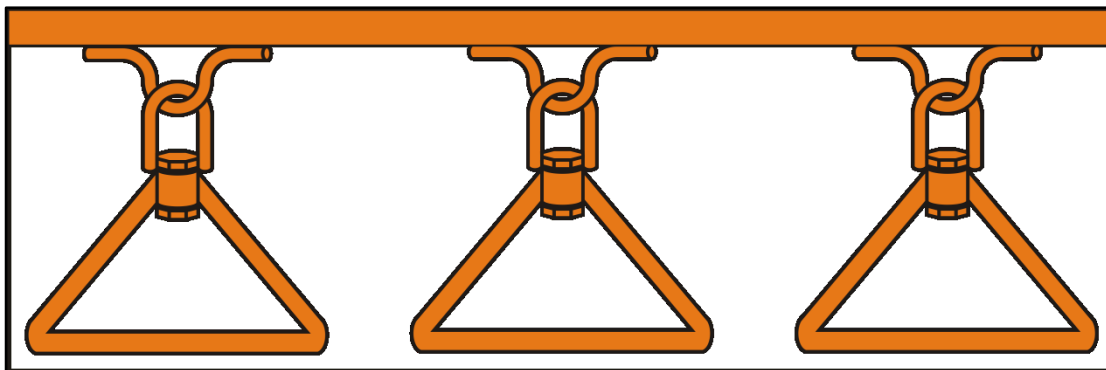
Nelayan rumpon Palabuhanratu tidak selalu melakukan pengecekan terhadap kondisi rumpon. Oleh karena itu, sebagai pengaman tambahan, selain diberi lapisan cat anti bocor, nelayan rumpon Palabuanaratu memberi sekat pada ponton tersebut. Seiring dengan waktu, pada akhirnya lapisan fiber dan cat terkikis sehingga lapisan besi bersentuhan langsung dengan air laut dan mengalami kebocoran akibat korosi, sekat membatasi daerah yang mengalami kebocoran sehingga air tidak mengisi seluruh ruang pada ponton. Sekat tidak diberi lapisan fiber yang dicat anti bor seperti halnya lapisan terluar ponton, oleh karena itu sekat akan mengalami kebocoran akibat korosi lebih cepat daripada lapisan ponton. Mempertimbangkan kondisi tersebut, maka seluruh ruang kosong pada ponton diisi oleh *sterofoam*. Jika ponton dan sekat

mengalami kebocoran, maka *sterofoam* akan mempertahankan kondisi ponton agar tidak tenggelam setidaknya hingga rumpun ditarik ke daratan untuk diperbaiki.

5.1.2 Swivel

Ketika melakukan pengukuran terhadap komponen ponton, *swivel* telah terpasang di ponton sehingga sulit untuk mengukur berat *swivel* seutuhnya. Jumlah *swivel* yang digunakan adalah empat buah. Pengukuran panjang *swivel* menggunakan tali dengan mengikuti bentuknya, kemudian tali tersebut diukur menggunakan penggaris. Data berat *swivel* didapat dengan mengalikan massa jenis bahan (ρ) *swivel* dengan volume *swivel*. *Swivel* memiliki diameter 2,38 cm dengan panjang masing-masing *swivel* 24 cm. Nilai massa jenis bahan *swivel* sama dengan massa jenis bahan yang digunakan sebagai ponton, yaitu $7,8 \text{ kg/m}^3$. Dengan demikian dapat diketahui massa *swivel* adalah 3.355,16 gr atau sekitar tiga kg dengan volume $426,87 \text{ cm}^3$. Berdasarkan massa *swivel* tersebut dapat diketahui nilai *buoyancy force* adalah -2.917,62 gr.

Swivel berfungsi sebagai tempat bergantungnya tali. Tiga *swivel* pada bagian tengah berfungsi sebagai tempat bergantung tali utama, sedangkan satu *swivel* pada bagian belakang berfungsi sebagai tempat bergantung tali atraktor. Pada *swivel* bagian tengah, setiap satu *swivel* digantungi oleh dua tali utama dan diikat oleh tali yang melilit badan badan ponton. Pengikatan *swivel* oleh tali pada badan ponton dimaksudkan untuk memperkuat *swivel*.



Gambar 43 Swivel ponton.

Swivel adalah sebuah perangkat kecil yang terdiri dari dua cincin terhubung agar dapat berputar bersama. Perangkat ini biasanya terbuat dari logam, dan bagian persambungannya berbentuk bola atau silinder. Fungsi utama dari *swivel* adalah untuk mencegah tali terlilit saat penarikan, mencegah kekusutan yang tidak diinginkan. Fungsi tambahan dari penggunaan *swivel* adalah dapat dijadikan patokan untuk pemasangan pemberat, apakah sebelum atau sesudah *swivel*. Namun ada pula kelemahan dari *swivel*, yaitu jika pemasangan *swivel* terlalu dekat dengan kail, terkadang kail akan terkait pada *swivel*. Selain itu, penggunaan bahan pembuat *swivel* yang tidak berkualitas membuat *swivel* menjadi titik lemah pada tali yang dipasangi *swivel* (<http://en.wikipedia.org>).

Fungsi *swivel* sebenarnya adalah untuk menjaga tali agar tidak terlilit saat rumpon bergerak bebas di perairan akibat terkena arus dan gelombang. Namun pemasangan yang tidak tepat dari *swivel* pada badan ponton mengakibatkan fungsi *swivel* hanya sebagai tempat bergantungnya tali utama. Gambar pemasangan *swivel* pada badan ponton dapat dilihat pada gambar 43. Dengan posisi pemasangan seperti itu, tali utama yang tergantung pada *swivel* akan tetap terlilit jika ponton berputar.

Fungsi *swivel* sebagai pereduksi puntiran semakin tidak ada lagi sejak ketiga *swivel* diikat ke badan ponton. Menurut nelayan rumpon, pengikatan *swivel* ke badan ponton bertujuan untuk memperkuat *swivel*. Jika suatu saat bagian *swivel* yang dilakukan pengelasan dengan badan ponton terlepas, maka tali tersebut akan mempertahankan agar tali utama tetap terikat dengan ponton, sehingga ponton tidak hilang terbawa arus. Jika fungsi *swivel* ingin tetap di pertahankan, maka bukan tiga *swivel* yang dipasang pada badan ponton, melainkan hanya satu saja namun berdiameter lebih besar.

5.1.3 Tali utama

Tali utama digunakan sebagai tali yang menghubungkan ponton dengan pemberat. Tali utama berjenis PP (*polypropylene*) yang terdiri dari empat cabang. Masing-masing cabang tali memiliki diameter 1,2 cm. Jumlah tali yang disediakan adalah 68 gulung, namun yang digunakan sebagai tali utama sebanyak 67,3 gulung

dengan panjang per gulung 80 meter, sisa tali digunakan sebagai penguat *swivel* (mengikat *swivel* dengan badan ponton) dan tali tambatan kapal. Dengan kata lain, tali yang digunakan sebagai tali utama adalah sepanjang 538.400 cm. Panjang tali utama tersebut bukan merupakan panjang tali yang bisa diasumsikan untuk mengetahui kedalaman perairan dimana rumpon dipasang. Panjang tali dari ponton hingga pemberat setelah dihitung berdasarkan konstruksinya adalah sepanjang 4.085 meter.

Subani dan Barus (1989) menyatakan bahwa panjang tali yang umum digunakan memiliki panjang 1,5 kali kedalaman perairan dimana rumpon dipasang. Sedangkan menurut Martasuganda (2008), panjang tali yang sebaiknya memiliki panjang dua hingga tiga kali kedalaman perairan dimana rumpon dipasang. Jika menggunakan pernyataan yang dikemukakan oleh Subani dan Barus, panjang tali sama dengan 1,5 kali kedalaman perairan, maka dapat diperkirakan kedalaman perairan adalah 2.723 meter. Namun, jika menggunakan pernyataan yang dikemukakan oleh Martasuganda, panjang tali sama dengan dua hingga tiga kali kedalaman perairan, maka kedalaman perairan dapat diperkirakan 2.042 hingga 1.362 meter.

Berdasarkan kedalamannya, laut dibedakan menjadi empat zona (<http://geografi.sekolahvirtual.or.id>).

1. Zona Pesisir

Wilayah laut antara garis batas air pasang naik dengan garis batas air pasang surut. Wilayah ini tergenang pada saat pasang naik sedangkan pada surut wilayah ini tidak tergenang air laut.

2. Zona Laut Dangkal

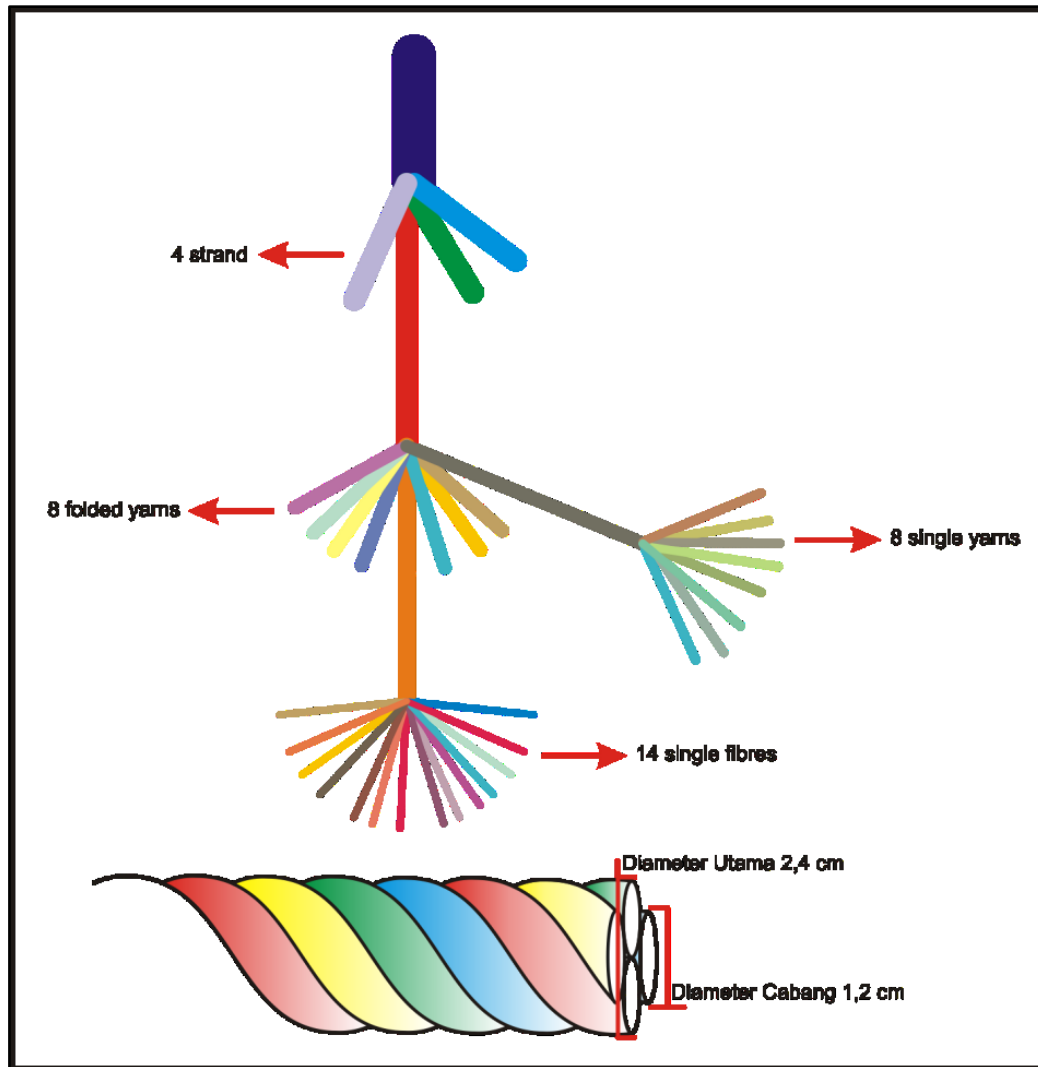
Wilayah laut yang dangkal antara batas pasang surut sampai kedalaman 200 meter. Zona ini kaya akan ikan dan tumbuh-tumbuhan laut, karena masih terdapat sinar matahari yang menyebabkan fotosintesis dapat berjalan baik (matahari dapat menembus air laut hingga kedalaman 90 meter). Pada zona ini pula plankton dapat tumbuh dengan subur karena terdapat banyak oksigen, dan masih terdapat ombak yang menyebabkan tersebarnya plankton sebagai makanan utama ikan.

Bagian ketiga memiliki konstruksi rangkap tiga, bagian keempat memiliki konstruksi tali rangkap dua. Tali bagian ketiga dan keempat memiliki panjang yang sama dengan tali bagian pertama maupun kedua, serta persambungan tali tiap bagian dihubungkan oleh sebuah hillban. Penggunaan konstruksi tali rangkap ini bertujuan agar meningkatkan umur teknis dari rumpon. Pada saat nelayan rumpon melakukan operasi penangkapan di sekitar rumpon, tak jarang alat tangkap mengenai tali utama sehingga menyebabkan alat tangkap terpaksa dipotong. Selain itu arus dan gelombang yang kuat sebagai ciri khas dari perairan selatan karena berhubungan langsung dengan Samudera Hindia membuat rumpon yang dibuat harus memiliki konstruksi yang kokoh agar dapat bertahan lama.

Bagian selanjutnya adalah bagian kelima. Tali pada bagian ini berbeda dengan konstruksi tali bagian sebelumnya, yaitu hanya terdiri atas satu tali saja, tidak rangkap, namun memiliki panjang mencapai 45 gulung tali atau setara dengan 3.600 meter. Penggunaan hanya satu rangkap tali pada bagian ini karena selain untuk menghemat biaya pembuatan rumpon, pada kedalaman bagian tali ini alat tangkap tidak ada yang dapat beroperasi.

Bagian tali keenam memiliki konstruksi rangkap dua dengan panjang 1 gulung tali atau setara dengan 80 meter. Pada bagian tali ini terhubung dengan tiga tali cabang pemberat oleh sebuah hillban. Setiap tali cabang pemberat memiliki panjang 80 meter dan hanya terdiri dari satu rangkap dan setiap tali cabang pemberat terhubung dengan tujuh tali pemberat oleh sebuah hillban. Panjang tali pemberat hanya sekitar lima meter yang terhubung langsung dengan pemberat.

Massa tali utama diperoleh dengan membandingkan panjang tali utama terhadap panjang tali contoh (55 cm) kemudian dikalikan dengan massa tali contoh (198 gr), sehingga diperoleh massa total tali adalah 1.967.800,32 gr. Begitu halnya dengan volume total tali, diperoleh dengan membandingkan panjang total tali (544.000 cm) terhadap panjang tali contoh (55 cm) kemudian dikalikan dengan volume tali contoh (248,69 cm³), sehingga diperoleh volume total tali adalah 2.459.750,40 cm³. Dengan memperhitungkan massa jenis air laut ($\rho_{sw}=1,025$) sebagai konstanta, maka diperoleh nilai *buoyancy force* tali utama sebesar 553.443,84 gr.

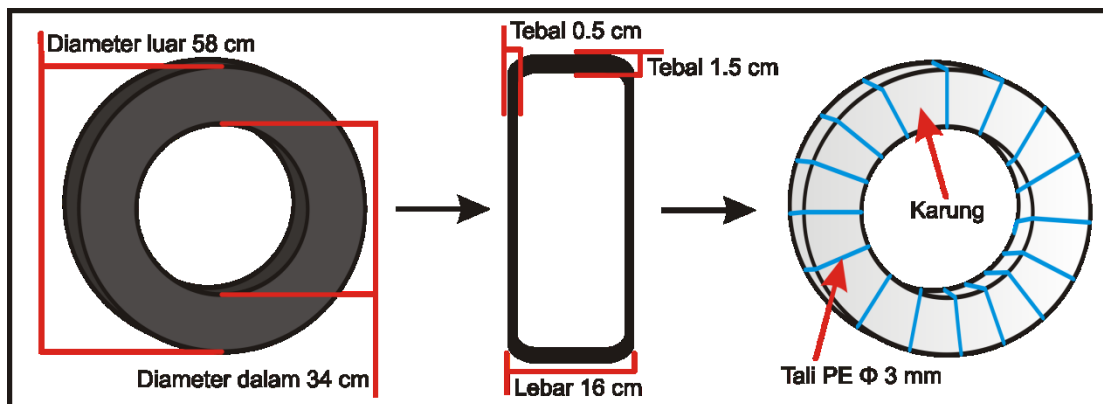


Gambar 44 Konstruksi tali utama.

5.1.4 Hillban

Hillban terbuat dari ban berukuran 13 inchi. Memiliki diameter dalam 34 cm, diameter luar 58 cm, lebar tapak 16 cm, tebal tapak 1,5 cm, dan tebal samping 0,5 cm. Hillban yang digunakan berjumlah 9 buah dengan berat masing-masing sekitar 8 kg. Berdasarkan hasil perhitungan, diketahui volume total hillban sebesar 54.937,44 cm³. Dengan memperhitungkan massa jenis air laut, diperoleh *buoyancy force* hillban sebesar -15.689,12 gr.

Hillban berfungsi sebagai persambungan antara tali utama dengan tali utama, tali utama dengan tali cabang pemberat, maupun tali cabang pemberat dengan tali pemberat yang memiliki jumlah tali yang berbeda. Selain itu, hillban berfungsi layaknya pegas yang dapat mengurangi ketegangan tali apabila tali mengalami *stretching* akibat ponton yang bergerak sebagai akibat dari pengaruh arus ataupun gelombang.



Gambar 45 Konstruksi hillban.

Hillban merupakan sebutan nelayan rumpon Palabuhanratu untuk ban bekas yang digunakan sebagai komponen rumpon. Sampai saat ini belum ditemukan nama lain untuk komponen ini. Dari beberapa sumber, ban bekas biasanya diiris tipis kemudian diikat dengan tali tambat, di sekeliling ban diikat tali plastik berwarna-warni sebagai atraktor permanen (Wahyudin, 2007).

Penggunaan ban sebagai karang buatan (*artificial coral reef*) maupun bahan pembuat rumpon menuai kontroversi. Penggunaan ban bekas dapat memperkaya sumberdaya hayati laut namun disisi lain dapat mencemari lingkungan. Beberapa waktu yang lalu, pimpinan tertinggi negara menyatakan pelarangan penggunaan ban bekas sebagai karang buatan dan bahan pembuat rumpon karena dapat mencemari perairan. Diklaim bahwa bahwa ban bekas dapat mengeluarkan senyawa *dioxin* yang berbahaya.

Robert M. Kellough (1991) dari *Deep/Quest 2 Underseas Research Society Inc.* yang melakukan penelitian mengenai *The Effect of Scrap Automobile Tires in Water*, menyatakan bahwa :

1. Terdapat kemungkinan terdapat faktor yang berasosiasi dengan ban mobil bekas sehingga beracun. Namun, senyawa alami yang merupakan faktor yang berasosiasi dengan ban mobil bekas sehingga menimbulkan racun belum bisa diketahui.
2. Hasil tes bahan kimia membuktikan bahwa sedikit kemungkinan bagi ban mobil bekas untuk mengeluarkan senyawa logam seperti PCB dan PAH ketika direndam dalam jangka waktu relatif singkat (60 hari).
3. Hasil penelitian memperlihatkan tidak terdapat perbedaan yang signifikan dari bioakumulasi dari bahan yang diujikan.
4. Senyawa yang beracun bisa jadi bukan berasal dari ban itu sendiri, namun dari kontaminasi eksternal seperti minyak atau gas.
5. Jumlah air yang sangat banyak yang merendam ban memiliki efek terhadap jumlah senyawa yang dihasilkan.
6. Ikan mengalami adaptasi selama hidup dalam media air yang berisi ban mobil bekas.

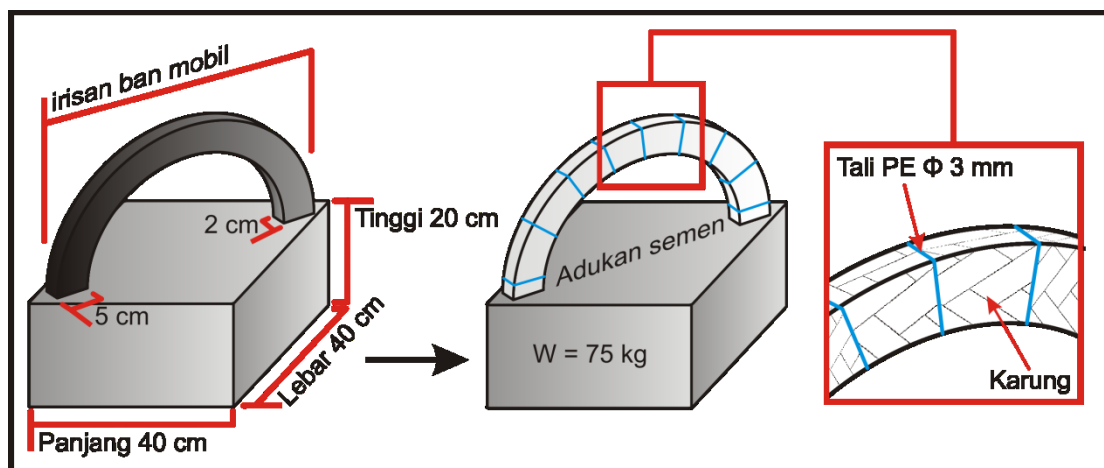
Dari hasil penelitiannya dapat disimpulkan bahwa bisa jadi ban mobil bekas beracun, namun jika jumlah ban mobil bekas yang terendam sangat banyak dan telah terendam dalam waktu yang sangat lama, berasosiasi dengan bahan lain sehingga beracun. Terlepas dari apakah ban mobil bekas beracun atau tidak, jelas bahwa karang yang terbuat dari ban mobil bekas tidak lebih indah daripada karang alami.

5.1.5 Pemberat utama

Pemberat layaknya jangkar, mempertahankan posisi rumpon agar tetap pada posisi semula. Pemberat memiliki terbuat dari campuran antara semen, pasir, batu, dan air. Perbandingan penggunaan bahan yang digunakan untuk membuat pemberat oleh nelayan dilakukan berdasarkan pengalaman atau perkiraan. Sebagai bahan rujukan mengenai perbandingan penggunaan bahan dalam pembuatan beton sebagai

pemberat rumpon dapat melihat perbandingan penggunaan bahan dalam pembuatan beton setara K225. Beton K225 merupakan beton yang biasa digunakan dalam pembuatan gedung bertingkat. Beton ini memiliki perbandingan antara semen, pasir, aggregate kasar dan air sebanyak 1 bagian semen, 2 bagian pasir, 3 bagian aggregate kasar, dan 0,5 bagian air. Sehingga jika ingin membuat beton setara K225 dalam 1 m^3 diperlukan 358 kg semen, 715 kg pasir, 1073 kg aggregate kasar, dan 179 kg air. Berat jenis beton segar setara K225 adalah 2.325 kg/m^3 .

Pemberat yang digunakan berjumlah 21 buah dengan massa per buah rata-rata 75.000 gr. Sehingga massa pemberat total yang digunakan sebesar 1.575.000 gr. Pemberat berbentuk menyerupai kubus dengan dimensi panjang 40 cm, lebar 40 cm, dan tinggi 20 cm sehingga volume total pemberat adalah 672.000 cm^3 . Perbandingan antara massa dengan volume pemberat menghasilkan massa jenis pemberat sebesar 2.34 gr/cm^3 . Perhitungan *buoyancy force* pemberat menghasilkan nilai *buoyancy* sebesar -886.200 gr.



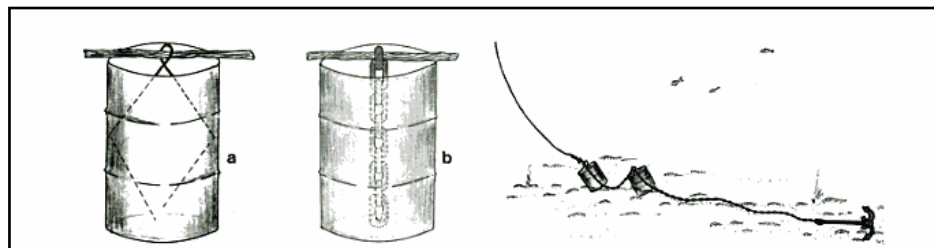
Gambar 46 Konstruksi pemberat utama.

Proses pembuatan pemberat dilakukan dengan mencetak adukan semen ke dalam cetakan kayu berbentuk kotak berukuran kecil. Pencetakan adukan semen ke dalam kotak-kotak yang berukuran kecil tersebut dimaksudkan agar mudah dalam pengangkutan dari darat menuju ke atas kapal jika adukan telah kering. Selain itu, ada

pula proses pembuatan pemberat lain menurut Ben-Yami M (1989) dalam *How to Make and Sets FADs* yang diterbitkan oleh FAO. Pembuatan pemberat pertama dilakukan pada drum minyak atau oli bekas. Bahan yang dibutuhkan dalam pembuatan pemberat pertama antara lain:

1. Drum minyak atau oli bekas berkapasitas 200 liter,
2. Sepuluh hingga dua puluh mata rantai besi bekas berukuran 18 mm atau satu hingga dua meter batang besi berdiameter 20 mm,
3. Satu kantong semen (50 kg),
4. Pasir, kerikil dan batu.

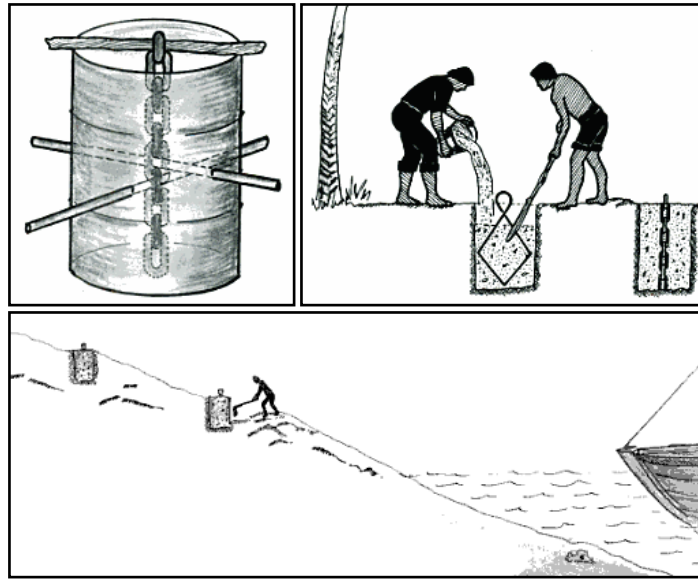
Ketika kering, satu drum dapat memiliki bobot hingga 500 kg. Sebagai tambahan dapat menggunakan jangkar besi yang dihubungkan dengan rantai pada pemberat.



Sumber: Ben-Yami M, 1989

Gambar 47 Pemberat dari drum.

Cara lain untuk membuat jangkar adalah dengan melubangi drum, kemudian memasukkan batangan besi atau pipa besi. Jika tidak memiliki drum, proses pembuatan pemberat dapat dilakukan dengan menggali lubang di tanah dan menuangkan adukan kedalamnya. Namun tidak memiliki *crane* atau mesin untuk mengangkat pemberat yang telah mengering, akan lebih baik jika menggali lubang pada sisi bukit.



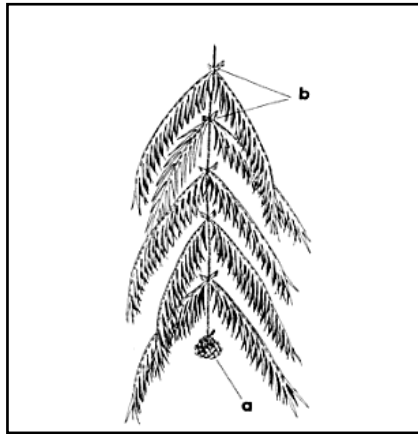
Sumber: Ben-Yami M (1989)

Gambar 48 Pemberat dari galian tanah.

5.1.6 Atraktor

Bahan yang digunakan sebagai atraktor adalah daun kelapa. Pada saat mengikuti operasi penangkapan di sekitar rumpon, daun kelapa yang digunakan berjumlah 25 buah dengan kondisi daun masih hijau namun bukan dalam kondisi segar. Menurut hasil wawancara dengan nelayan rumpon, sebenarnya tidak ada batasan jumlah daun kelapa yang digunakan sebagai atraktor dan kondisi daun kelapa. Hanya saja, jika daun kelapa yang dijadikan sebagai atraktor masih muda, ikan tidak langsung mendekati rumpon karena menunggu daun kelapa busuk terlebih dahulu. Tapi jika daun kelapa yang digunakan sudah kering, ikan lebih cepat mendatangi rumpon, namun atraktor harus sering diganti. Atraktor dapat bertahan paling lama dua hingga tiga minggu dan operasi penangkapan dimulai setelah satu minggu pemasangan rumpon. Selain karena pembusukan, atraktor dapat rusak karena keadaan alam. Berdasarkan hasil evaluasi terhadap nelayan di Teluk Semangka, Lampung (Yusfiandayani, 2004), daya tahan atraktor daun kelapa dapat mencapai 3-4 minggu dengan operasi penangkapan dilakukan setelah 10 hari pemasangan.

Menurut Yusfiandayani (2004) selain daun kelapa (*Cocos nucifera*), bahan alami yang biasa digunakan sebagai atraktor adalah daun nipah (*Nypa fructican*) dan daun pinang (*Areca catechu*). Diantara ketiga daun ini, yang memiliki daya tahan yang baik di perairan pada segala musim adalah daun kelapa. Daun kelapa tahan diperairan hingga 24-28 hari. Hal ini disebabkan oleh kandungan ketebalan lapisan *epidermis* dan lapisan *cuticle* yang lebih tebal dibandingkan dengan daun nipah maupun daun pinang. Penggunaan bahan alami sebagai atraktor tergantung dari sumberdaya alam yang tersedia di daerah perkampungan nelayan tersebut. Tak hanya daun kelapa, daun nipah, atau daun pinang yang dapat dijadikan atraktor, daun lontar pun dapat dijadikan atraktor. Berdasarkan penelitian Arsyad (1999) yang membandingkan hasil tangkapan *purse seine* yang menggunakan daun lontar dan daun kelapa menunjukkan hasil tangkapan yang berbeda nyata, dimana daun lontar memberikan hasil tangkapan yang lebih banyak.



Sumber: Ben-Yami M (1989)

Gambar 49 Atraktor daun kelapa.

5.1.7 Tali hillban, tali tambatan pemberat, dan tali atraktor

Tali hillban dan tali pemberat dipergunakan untuk mengikat karung yang membungkus hillban dan tambatan pemberat agar karung tidak lepas saat rumpun dipasang di perairan. Pada saat pengambilan data pengukuran, data spesifik mengenai berapa panjang tali yang digunakan sebagai tali pengikat hillban, tali pengikat

layaknya pada simpul tali yang mengikat hillban. Tali tambatan kapal ini diikat secara permanen pada ponton. Kapal yang akan bertambat pada ponton harus mengambil ujung tali tambat yang mengapung di perairan.

Tali pengikat *swivel* digunakan untuk memperkuat *swivel*. *Swivel* diikat dengan tali mengelilingi badan ponton, namun hanya *swivel* bagian tengah yang diikat. Pengikatan pada *swivel* bagian tengah ini dilakukan karena pada *swivel* bagian tengah digantungi oleh dua tali utama yang memiliki rumbai dan waring. Panjang tali yang digunakan setelah memperhitungkan keliling badan ponton adalah 967,12 cm dengan volume 4.372,93 cm³. Perhitungan terhadap gaya apung tali pengikat *swivel* menghasilkan nilai *buoyancy force* sebesar 3.515,13 gr.

Operasi penangkapan disekitar rumpon dapat berlangsung selama seminggu. Pada saat istirahat di siang hari dan di malam hari, kapal akan menambatkan diri di rumpon. Kapal yang paling dekat dengan rumpon akan mengambil tali tambat dan menambatkannya pada kapal. Jika kemudian datang kapal lain untuk bertambat, maka kapal yang baru datang akan menambatkan diri pada kapal yang telah bertambat lebih dulu. Begitu seterusnya dan kapal yang saling bertambat akan terlihat seperti pagar kapal di tengah laut.

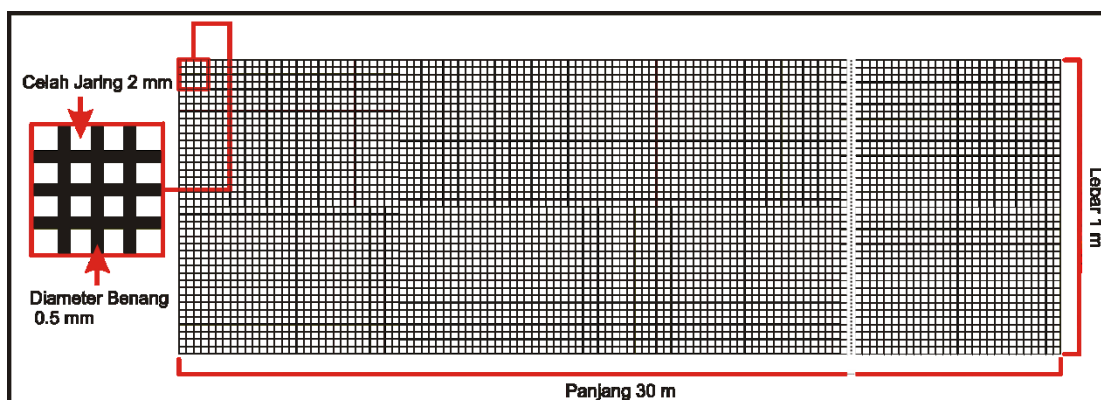
5.1.10 Waring dan tali rafia

Menurut wawancara dengan nelayan pembuat rumpon, pemasangan waring dimaksudkan agar ikan tetap berada di sekitar rumpon jika atraktor utama yaitu atraktor daun kelapa hilang atau rusak.

Waring yang digunakan memiliki dimensi lebar 100 cm dan panjang 3.000 cm. pada saat pengambilan data, waring telah dipasang pada tali utama sehingga penimbangan massa waring tidak dapat dilakukan. Karena kondisi tersebut, sehingga hanya dapat mengamati diameter tali dan celah jaring. Dari hasil pengamatan tersebut diperoleh data diameter tali waring, yaitu 0.5 mm dengan bukaan celah waring 2 mm. Berdasarkan data-data waring yang kemudia dilakukan perhitungan, diperoleh data jumlah tali per lebar waring adalah 400 tali dan jumlah tali per panjang waring adalah 12.000 tali. Dengan demikian panjang tali per panjang waring adalah 1.200.000 cm

dan panjang tali per lebar waring adalah 1.200.000 cm. Karena waring dibuat oleh mesin, sehingga tidak terdapat simpul pada jaringnya. Oleh karena itu harus menghitung panjang tali yang hilang akibat perpotongan antar tali tersebut. Perhitungan tersebut menghasilkan nilai 240.000 cm. Dengan demikian panjang total tali yang terdapat pada waring tersebut dapat diperkirakan mencapai 2.160.000 cm dengan volume waring 4.239 cm³. Perhitungan gaya apung waring menghasilkan nilai *buoyancy force* sebesar 275,54 gr.

Tali rafia digunakan untuk mengikat waring dengan tali utama dan membuat rumbai-rumbai yang diikatkan pada tali utama. Rumbai berfungsi untuk menarik perhatian ikan dengan gerakan rumbainya ketika terkena arus air. Data mengenai perhatian ikan dengan gerakan rumbainya ketika terkena arus air. Data mengenai berapa panjang tali rafia yang digunakan untuk mengikat waring pada tali utama dan berapa panjang tali yang digunakan untuk membuat satu rumbai serta jumlah rumbai yang dibuat tidak didapat. Data yang diperoleh hanya jumlah tali rafia yang dipergunakan. Tali rafia yang digunakan berjumlah 2 pack. Satu pack terdiri dari 50 bungkus dan tiap bungkusnya memiliki panjang 300 cm. Massa per bungkus adalah 0,5 gr. Berdasarkan literatur yang diperoleh, tali rafia terbuat dari bahan plastik PP (*polypropylene*) dengan massa jenis 0,91 gr/cm³. Dengan demikian massa total tali rafia yang digunakan adalah 50 gr dan volume 549,45 cm³. Perhitungan gaya apung rafia menghasilkan nilai *buoyancy force* sebesar 63,19 gr.



Gambar 50 Waring.

Literatur mengenai penggunaan waring dan bahan-bahan sintetis lain seperti tali rafia sebagai atraktor, menyatakan bahwa bahan sintetis ini digunakan sebagai atraktor permanen. Atraktor yang terbuat dari bahan alami lebih cepat rusak dan memerlukan penggantian secara berkala. Oleh karena itu digunakanlah waring, tali rafia atau bahan sintetis lain sebagai atraktor. Menurut Subani dan Barus (1989), khusus untuk rumbai (*apandages*), disamping menggunakan daun nyiur juga menggunakan bahan sintetis, seperti serabut bekas tali PE atau bekas benang jaring nilon bahkan bekas ban mobil. Pernyataan lain dikemukakan oleh Martasuganda (2008), bahwa bagian atraktor bisa menggunakan dedaunan seperti pelepah daun kelapa, pinang, rumbia, dedaunan lain, atau bekas alat tangkap yang terbuat dari jaring. Pernyataan serupa juga dinyatakan oleh Boy dan Smith (1984) dalam Yusfiandayani (2004), bahwa *apandage* atau atraktor dapat berupa daun kelapa, *tyrewall*, jaring dan kumpulan tali-temali yang diikatkan pada bagian rakit telah berhasil meningkatkan efektifitas rumpon dalam memikat ikan.

Monintja (2003) mengemukakan bahwa perbedaan morfologi, keawetan fisik dan biologis dari bahan alami dan bahan buatan dari atraktor merupakan penyebab perbedaan komunitas perifiton pada atraktor dan mengakibatkan terjadinya perbedaan keefektifan bahan atraktor dalam penangkapan ikan. *Perifiton* merupakan komunitas organisme yang hidup menempel di atas permukaan sekitar substrat yang tenggelam. Substrat tersebut dapat berupa batu, kayu, tumbuhan air yang tenggelam dan kadangkala hewan air (Odum, 1971). Sedangkan menurut Weitzel (1979), *perifiton* terdiri dari mikroflora yang tumbuh pada semua substrat yang tenggelam.

Menurut Ruttner (1974) dalam Yusfiandayani (2004), perkembangan *perifiton* menuju kestabilan komunitasnya sangat ditentukan oleh kestabilan substrat. Substrat berupa benda hidup sering bersifat sementara karena adanya proses pertumbuhan dan kematian. Setelah tumbuh cepat kemudian stabil, selanjutnya mengalami kematian dan pembusukan. Setiap saat pada substrat hidup akan terjadi perubahan lingkungan sebagai akibat respirasi dan asimilasi, sehingga mempengaruhi komunitas perifiton. Pada substrat berupa benda mati akan lebih stabil atau permanen meskipun pembentukan komunitas lambat laun akan lebih stabil sehingga tidak mengalami

perubahan. Singkatnya, tipe substrat sangat menentukan proses kolonisasi dan komposisi perifiton.

Ikan-ikan berkumpul di rumpon antara lain disebabkan oleh proses pembentukan rantai makanan lokal. Proses ini diawali dengan terbentuknya akumulasi atau kolonisasi *perifiton* yang diikuti dengan berkumpulnya pemangsa *perifiton* dan kemudian *plankton feeder*. Kolonisasi oleh mikroorganisme, baik mikroba maupun mikroalga akan menarik perhatian juvenil ikan, ikan berukuran kecil sampai ikan berukuran besar sehingga menyebabkan terjadinya *food web* di sekitar rumpon (Yusfiandayani, 2004).

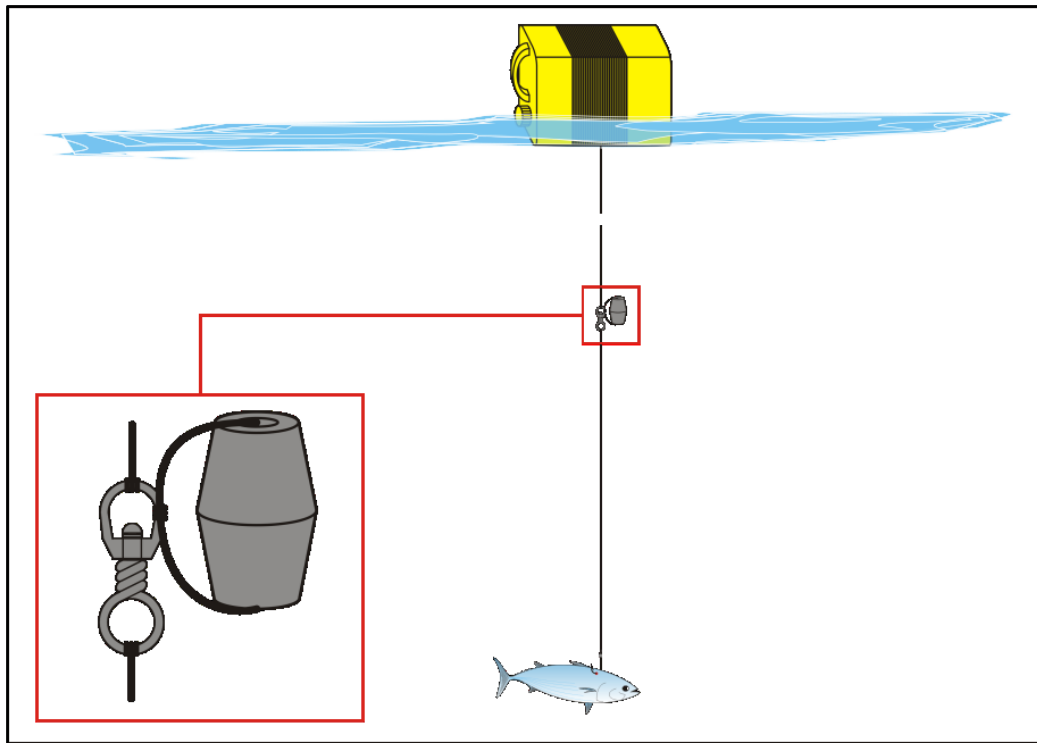
5.2 Alat Tangkap

Alat tangkap yang digunakan oleh nelayan rumpon di perairan palabuhanratu khususnya yang digunakan oleh nelayan rumpon *PSP 01* diantaranya tomba jerigen, pancing tonda, pancing layangan, pancing kondo-kondo, pancing coping, dan pancing taber. Semuanya memiliki konstruksi, metode penangkapan, dan waktu penangkapan yang berbeda-beda.

5.2.1 Tomba jerigen

Tomba jerigen merupakan alat tangkap utama yang digunakan dalam operasi penangkapan di sekitar rumpon yang termasuk dalam kategori alat tangkap pancing. Alat tangkap ini memiliki konstruksi yang terdiri dari jerigen, tali nilon monofilamen, *swivel*, pemberat, kail, dan umpan hidup. Jerigen yang digunakan memiliki kapasitas 25 liter dan biasanya memiliki warna yang cerah, kontras terhadap warna perairan sehingga mudah dikenali. Selain itu, jerigen diberi tanda, dapat berupa tulisan cat minyak ataupun plastik berwarna yang mencirikan pemilik jerigen tersebut. Jerigen berfungsi sebagai pelampung tanda. Pada saat umpan tidak dimakan oleh ikan tujuan tangkapan, posisi jerigen terbaring dengan tali masih tergulung pada jerigen. Sedangkan pada saat umpan dimakan oleh ikan tujuan penangkapan, posisi jerigen tersungkur dengan ikatan tali terulur seluruhnya karena ditarik oleh ikan tujuan penangkapan.

berlaku apabila ada salah satu umpan yang dimakan oleh ikan tujuan penangkapan dan terjerat, maka jerigen tersebut yang pertama diangkat ke atas kapal. Ikan umpan yang sudah mati dikumpulkan di keranjang untuk kemudian dimasukkan kedalam palkah ketika operasi penangkapan berakhir.



Gambar 51 Tomba jerigen.

5.2.2 Pancing layangan

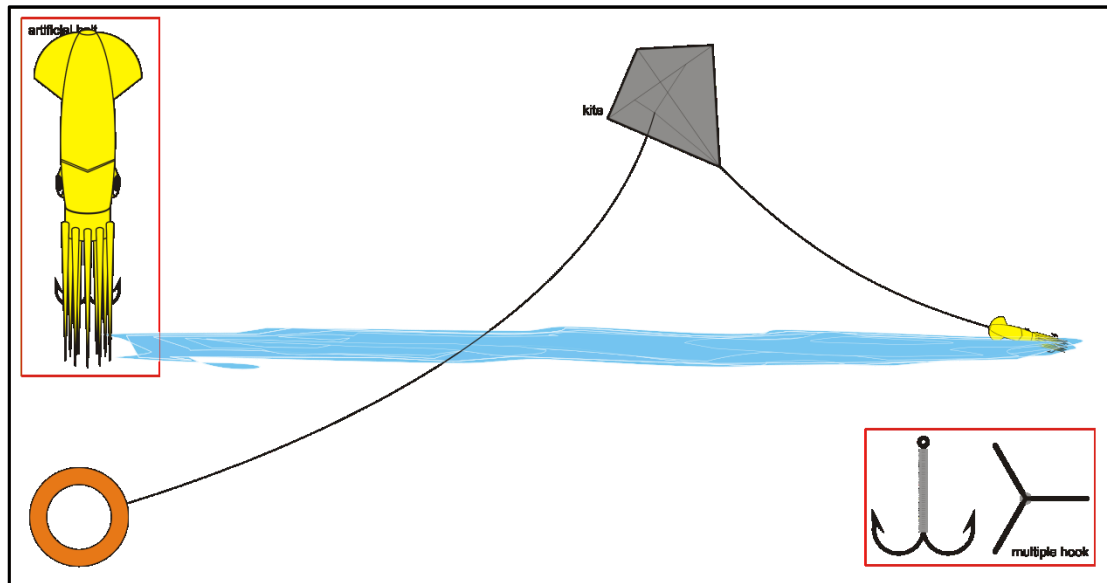
Alat tangkap pancing layangan biasanya dioperasikan pada saat istirahat makan siang ataupun menjelang sore hari. Tapi kadang dioperasikan pula pada saat nelayan jenuh karena ikan umpan tidak tertangkap oleh tonda dengan lama pengoperasian tak tentu. Dengan kata lain, alat tangkap pancing layangan juga dioperasikan untuk menangkap ikan umpan. Alat tangkap ini memiliki konstruksi yang terdiri dari layangan plastik, umpan buatan berupa cumi-cumi, multiple hook, dan tali.

Multiple hook yang digunakan merupakan buatan atau rakitan sendiri dengan ukuran mata pancing yang digunakan adalah mata pancing no dua. Jumlah mata

pancing yang dirakit menjadi satu kesatuan terdiri dari tiga mata pancing yang diikat menggunakan benang nilon monofilamen. Mata pancing disembunyikan di dalam tangan cumi-cumi umpan buatan agar tidak terlihat oleh ikan. Umpan buatan yang berupa cumi-cumi tersebut memiliki warna yang cerah dan menarik perhatian ikan, serta masih memiliki tangan yang lengkap, jika tangan cumi-cumi tidak lengkap, maka mata pancing akan terlihat oleh ikan.

Layangan yang digunakan terbuat dari plastik dengan rangka terbuat bambu. Umumnya berwarna hitam. Layangan diikat dengan tali nilon monofilamen dengan ukuran tali no 200 yang terhubung dengan *line roller*. Tali yang tersedia pada line roller dapat mencapai 100 meter, namun pada saat digunakan tidak mencapai 100 meter. Pada bagian bawah nelayan diikat tali nilon monofilamen dengan ukuran tali no 200 yang terhubung dengan umpan buatan dan kail. Panjang tali antara layangan dengan umpan buatan mencapai sembilan meter.

Alat tangkap ini dioperasikan dengan arah angin tidak datang dari arah haluan atau buritan kapal, melainkan dari sisi kanan atau kiri kapal. Hal ini dilakukan agar mudah dalam pengoperasian alat tangkap dan proses pengangkatan ikan hasil tangkapan ke atas kapal. Pengoperasian alat tangkap ini diawali dengan penurunan umpan buatan ke perairan, kemudian berulah layangan diterbangkan. Layangan diterbangkan dengan arah menyamping agar umpan buatan menyusur di permukaan air. Umpan buatan yang memiliki warna menarik serta riakan air akibat gerakan menyusur umpan di permukaan air menarik perhatian ikan. Dengan cepat ikan menyambar umpan dan menariknya ke perairan yang lebih dalam, pada saat itu nelayan menarik tali. Pada saat umpan disambar ikan, tak jarang layangan ikut masuk kedalam air dan terjadi saling tarik antara nelayan dan ikan. Akibat dari tarik-menarik tersebut, layangan menjadi rusak.

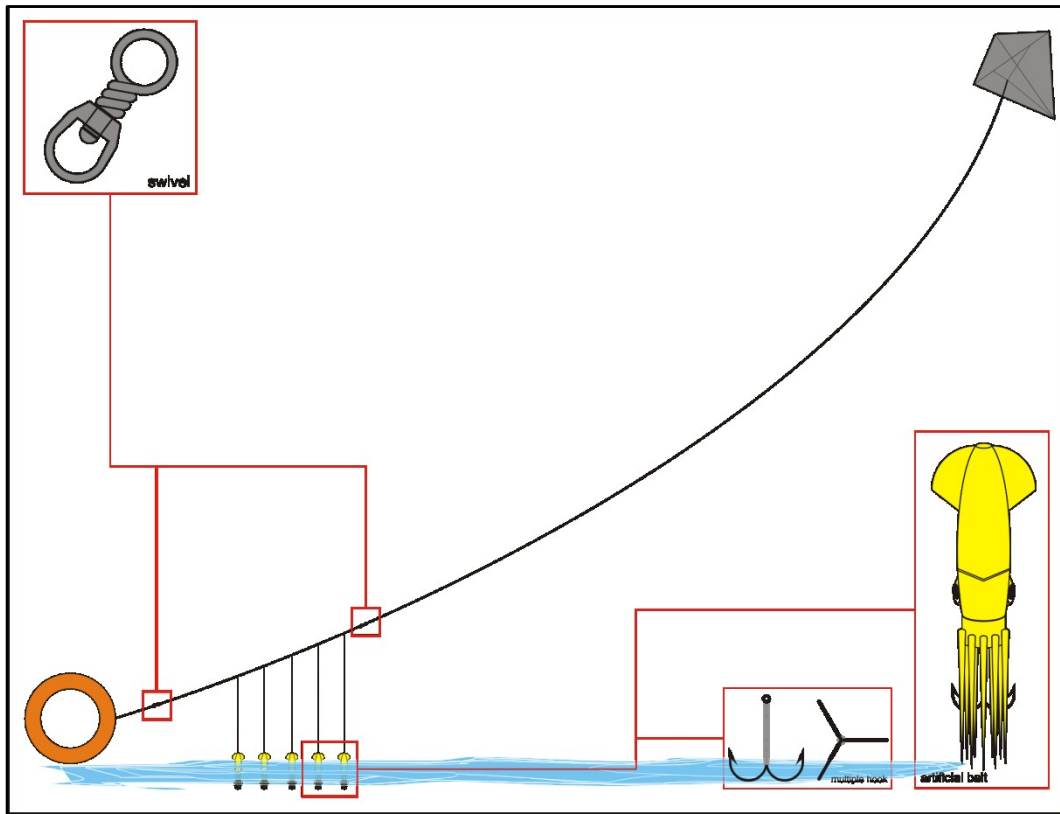


Gambar 52 Pancing layangan.

5.2.3 Pancing kondo-kondo

Pancing kondo-kondo merupakan alat tangkap sejenis pancing layangan. Beda antara pancing layangan dan pancing kondo-kondo terdapat pada metode penangkapan dan konstruksinya. Pancing kondo-kondo terdiri dari layangan plastik, tali nilon monofilamen, *swivel*, umpan buatan dan *multiple hook*.

Layangan pada kondo-kondo diikat oleh tali nilon monofilamen sepanjang 100 meter dengan ukuran tali yang digunakan adalah no 70. Ujung lain dari tali diberi *swivel* dan dihubungkan dengan tali nilon monofilamen sepanjang 20 meter dengan ukuran tali no 100 dan diberi *swivel*. *Swivel* diikat oleh tali nilon monofilamen sepanjang satu rol dengan ukuran tali no 100. Pada tali sepanjang 20 meter ini digantungi lima tali cabang dengan ujung mengikat pada umpan buatan yang berukuran kecil dan *hook* dengan ukuran *hook* no 7. Panjang tali cabang ini bervariasi. Tali cabang pertama memiliki panjang 2 meter, tali cabang kedua memiliki panjang 1,5 meter, tali cabang ketiga memiliki panjang 1,5 meter, tali cabang keempat memiliki panjang satu meter, dan tali cabang kelima memiliki panjang setengah meter. Panjang tali cabang yang berbeda dimaksudkan agar pada

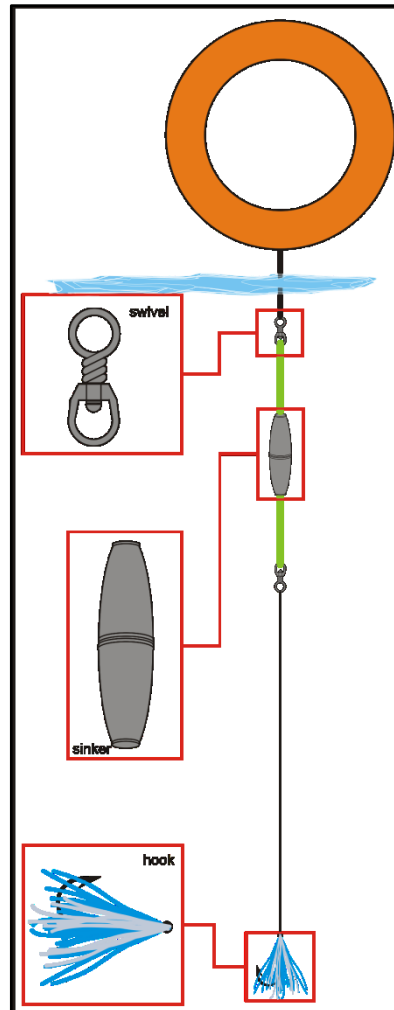


Gambar 53 Pancing kondo-kondo.

5.2.4 Pancing Taber

Pancing taber memiliki konstruksi yang berbeda dengan pancing tonda. Pada pancing taber terdapat 15 tali cabang, *swivel* dan pemberat. Panjang tali cabang 30 cm dengan jarak antar tali cabang sepanjang 1 meter. Jarak antara *swivel* pertama dengan tali cabang pertama adalah sepanjang 1,5 meter, sedangkan dari tali cabang terakhir hingga *swivel* kedua berjarak hanya 1 meter. Pada *swivel* kedua ini digantungi pemberat 100 gr. Tali cabang menggunakan tali nilon monofilamen berukuran no 70, sedangkan tali utamanya memiliki ukuran tali nilon monofilamen no 150. Kail yang digunakan memiliki ukuran no 7 dengan umpan buatan berupa tali rafia berwarna. Cara pengoperasian alat tangkap pancing taber sama dengan cara pengoperasian alat tangkap pancing tonda, yaitu ditarik pada buritan kapal.

terlebih dahulu kemudian diikuti dengan pemberat. Pemberat dibiarkan tenggelam hingga kedalaman tertentu tanpa ditahan, dengan kata lain kail akan bergerak karena tertarik oleh pemberat. Penarikan coping dilakukan dengan cepat. Tangan kanan menarik dan tangan kiri menahan. Jika dibayangkan, kail yang diikatkan benang berwarna-warni akan bergerak layaknya ikan yang naik atau turun di perairan.



Gambar 55 Pancing coping.

5.2.6 Pancing tonda

Alat tangkap pancing tonda merupakan alat tangkap utama yang digunakan untuk memancing ikan umpan. Operasi alat tangkap tonda ini ditarik dengan kapal

yang ditonda sekitar 20 meter dan kecepatan perahu rata-rata empat hingga lima knot. Pada pukul tujuh hingga pukul Sembilan, jarak rumpon dengan lokasi menonda sekitar 50-100 meter dengan panjang tali 45-60 meter dengan kecepatan perahu tujuh hingga delapan knot. Sedangkan pada siang hari, jarak antara rumpon dengan lokasi menonda mencapai 180 meter dengan panjang tali 45-60 meter dan kecepatan rata-rata kapal tujuh hingga delapan knot (Nugroho, 1992).

Zainuddin (2002) menyatakan bahwa penggunaan umpan yang berbeda memberikan pengaruh nyata terhadap hasil tangkapan ikan cakalang, baik dalam satuan jumlah (ekor) maupun dalam satuan berat (kilogram). Berdasarkan hal tersebut maka dapat dikemukakan bahwa ikan cakalang mempunyai kecenderungan memilih makanannya (jenis umpan). Dilihat dari jumlah dan berat hasil tangkapan dengan menggunakan umpan plastic mengkilap menunjukkan hasil yang lebih baik daripada umpan plastik transparan. Hal ini sesuai dengan yang dikemukakan oleh Djatikusumo (1975) bahwa ikan cakalang sangat peka terhadap rangsangan cahaya yang mengkilat.

5.3 Hasil Tangkapan

Hasil tangkapan yang didapat selama operasi penangkapan terdiri dari hasil tangkapan utama dan hasil tangkapan sampingan. Hasil tangkapan utama yaitu ikan jenis tuna sirip kuning (*Thunnus albacares*), sedangkan hasil tangkapan sampingan berupa ikan umpan yaitu jenis cakalang (*Katsuwonus pelamis*) dan tuna sirip kuning yang memiliki berat beragam yang tidak masuk dalam standar ekspor. Hasil tangkapan ketika operasi penangkapan, memperoleh hasil tangkapan 98 kg tuna (masuk kategori ekspor) dan 90 kg cakalang (termasuk tuna yang tidak kategori ekspor).

5.3.1 Cakalang

Cakalang atau *skipjack tuna* merupakan salah satu ikan ekonomis penting. *Skipjack tuna* pertama kali dideskripsikan oleh Carl Linnaeus tahun 1758, yang saat itu bernama *Scomber pelamis*. Sampai saat ini masih banyak perdebatan mengenai penempatan umum dari *skipjack*, sebagian mengajukan untuk dikelompokkan dengan

genus *Euthynnus*, sedangkan yang lain merekomendasikan untuk ditempatkan di genus tersendiri, *Katsuwonus*. Nama spesies *pelamis* berasal dari bahasa Latin yang berarti ikan tongkol atau cakalang. Nama latin lain yang digunakan untuk menyebut tuna jenis ini diantaranya: *Scomber pelamides* oleh Lacepede (1880), *Scomber pelamys* oleh Bloch and Synder (1801), *Thynnus pelamys* oleh Cuvier (1817), *Thynnus vagans* oleh Lesson (1826), *Orcynus pelamy* oleh Poey (1875), dan *Gymnosarda pelamis* Dresslar and Fesler (1889). Di Indonesia ikan ini lebih dikenal sebagai cakalang, namun oleh nelayan Palabuhanratu ikan ini dikenal sebagai *salur*. Nama sebutan lain untuk cakalang diantaranya: *aku* (Hawai), *atu* (Tahiti), *atun* (Spanyol), *balaya* (Sinhala), *bonite* (Perancis), *bonito del sur* (Spanyol), *cachurreta* (Spanyol), *faolua* (Samoan), *gaiado* (Portugis), *jodari* (Swahili), *listado* (Spanyol), *merma* (Spanyol), *nzirru* (Italia), *palamatu* (Italia), *serra* (Portugis), and *tunna* (Arab) (<http://www.flmnh.ufl.edu>).

Klasifikasi cakalang menurut Matsumoto, et al. (1984) adalah sebagai berikut:

Filum: Vertebrata

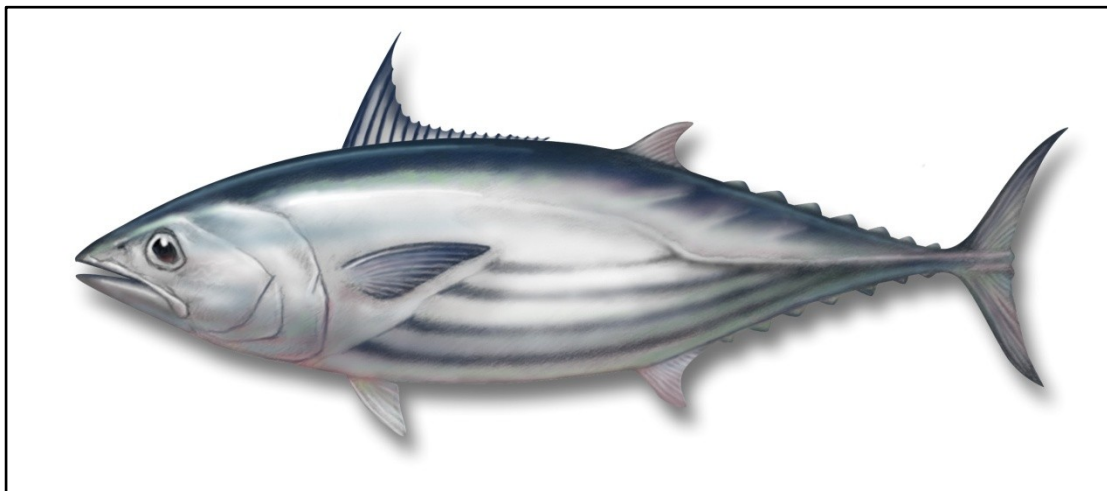
Kelas: Teleostei

Ordo: Perciformes

Famili: Scombridae

Genus: *Katsuwonus*

Spesies: *Katsuwonus pelamis*



Sumber: <http://www.sustainablesushi.net>

Gambar 57 Ikan cakalang.

Cakalang menyebar di sekitar daerah tropis. Menurut Gunarso (1996), suhu ideal untuk ikan cakalang antara 26°C – 32°C, dan suhu ideal untuk melakukan pemijahan berkisar antara 28°C – 29°C dengan salinitas 33‰. Sedangkan menurut Jones dan Silas (1962), cakalang hidup pada suhu antara 16°C – 30°C dengan suhu optimum 28°C. Ikan cakalang menyebar luas diseluruh perairan tropis dan sub tropis pada lautan Atlantik, Hindia dan Pasifik, kecuali laut Mediterania. Penyebaran ini dapat dibedakan menjadi dua macam yaitu penyebaran horizontal atau penyebaran menurut letak geografis perairan dan penyebaran vertikal atau penyebaran menurut kedalaman perairan. Penyebaran tuna dan cakalang sering mengikuti penyebaran atau sirkulasi arus garis konvergensi diantara arus dingin dan arus panas merupakan daerah yang kaya akan fitoplankton dan diduga daerah tersebut merupakan *fishing ground* yang sangat baik untuk perikanan tuna dan cakalang. Dalam perikanan tuna dan cakalang pengetahuan tentang sirkulasi arus sangat diperlukan, karena kepadatan populasi pada suatu perairan sangat berhubungan dengan arus-arus tersebut (Nakamura, 1969).

Penyebaran cakalang di perairan Samudra Hindia meliputi daerah tropis dan sub tropis, penyebaran cakalang ini terus berlangsung secara teratur di Samudra Hindia di mulai dari Pantai Barat Australia, sebelah selatan Kepulauan Nusa Tenggara, sebelah selatan Pulau Jawa, Sebelah Barat Sumatra, Laut Andaman, diluar pantai Bombay, diluar pantai Ceylon, sebelah Barat Hindia, Teluk Aden, Samudra Hindia yang berbatasan dengan Pantai Sobali, Pantai Timur dan selatan Afrika (Jones dan Silas, 1963).

Cakalang termasuk jenis ikan tuna dalam famili Scombridae, species *Katsuwonus pelamis*. Ciri-ciri morfologi cakalang yaitu tubuh berbentuk *fusiform*, memanjang dan agak bulat, tapis insang (*gill rakes*) berjumlah 53-63 pada helai pertama. Mempunyai dua sirip punggung yang terpisah. Pada sirip punggung yang pertama terdapat 14-16 jari-jari keras, jari-jari lemah pada sirip punggung kedua diikuti oleh 7-9 *finlet*. Sirip dada pendek, terdapat dua *flops* diantara sirip perut. Sirip anal diikuti dengan 7-8 *finlet* (sirip antara sirip *dorsal* terakhir dan sirip *caudal*). Badan tidak bersisik kecuali pada barut badan (*corselets*) dan *lateral line* terdapat

titik-titik kecil. Bagian punggung berwarna biru kehitaman (gelap) disisi bawah dan perut keperakan, dengan 4-6 buah garis-garis berwarna hitam yang memanjang pada bagian samping badan. Cakalang termasuk ikan perenang cepat dan mempunyai sifat makan yang rakus. Ikan jenis ini sering bergerombol, berenang bersama melakukan ruaya disekitar pulau maupun jarak jauh dan senang melawan arus, ikan ini biasa bergerombol diperairan pelagis hingga kedalaman 200 m. Ikan ini mencari makan berdasarkan penglihatan dan rakus terhadap mangsanya (Collete, 1983).

5.3.2 Madidihang

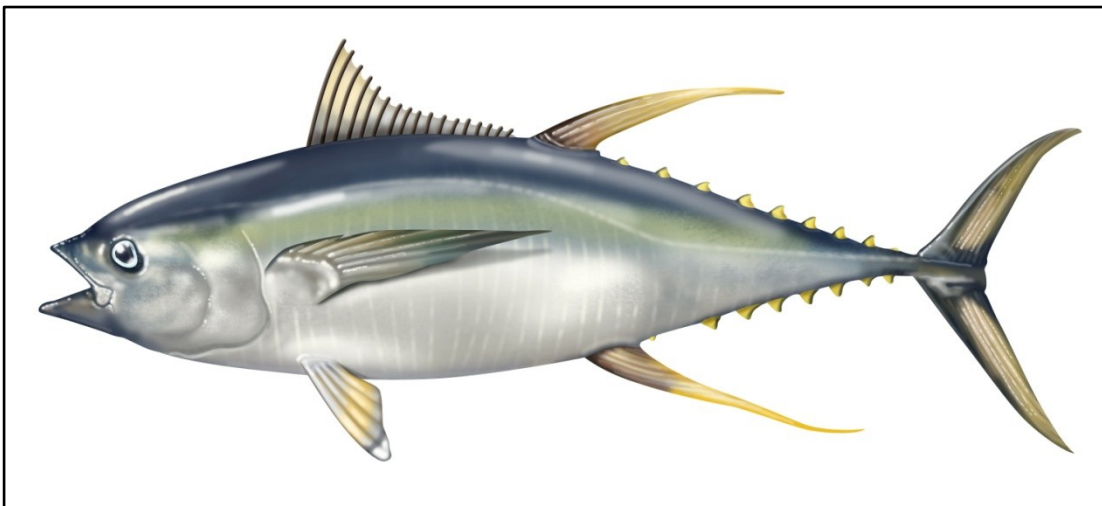
Yellowfin tuna pertama kali dideskripsikan oleh Bonnaterre pada tahun 1788, kala itu bernama *Scomber albacores*. Ginsburg pertama kali menggunakan kombinasi *Thunnus albacores* pada 1953. Nama genus *Thunnus* berasal dari bahasa Yunani “thynnos” yang berarti tuna. Banyak nama yang telah digunakan untuk menyebut tuna ini diantaranya *Scomber albacores*, *Thynnus argentivittatus*, *Scomber sloanei*, *Thynnus albacore*, *Thynnus macropterus*, *Thunnus argentivittatus*, *Orcynus subulatus*, *Orcynus albacore*, *Orcynus macropterus*, *Germo macropterus*, *Thunnus macropterus*, *Thunnus allisoni*, *Germo argentivittatus*, *Germo allisoni*, *Neothunnus macropterus*, *Neothunnus catalinae*, *Neothunnus albacores*, *Neothunnus allisoni*, *Kishinoella zacalles*, *Semathunnus guildi*, *Semathunnus itosibi*, *Neothunnus argentivittatus*, *Germo albacore*, *Thunnus albacore*, *Germo itosibi*, *Neothunnus albacore brevipinna*, *Neothunnus albacore longipinna*, *Neothunnus macropterus macropterus*, *Neothunnus macropterus itosibi*, *Neothunnus brevipinna*, *Thunnus zacalles*, *Thunnus catalinae*, *Neothunnus albacores*, *Thunnus albacores*, *Neothunnus albacore macropterus*, *Thunnus albacores macropterus*, and *Thunnus itosibi*.

Di Indonesia ikan ini lebih dikenal sebagai madidihang, namun oleh nelayan Palabuhanratu ikan ini lebih dikenal sebagai *jabrig*. Sebutan lain untuk madidihang diantaranya adalah *a'ahi* (Tahiti), *ahi* (Hawai), *albacore* (Portugis), *badla-an* (Tagalog), *bakulan* (Malaysia), *chefarote* (Portugis), *gelang kawung* (Malaysia), *jaydher* (Arabi), *kababa* (Arabi), *kelawalla* (Tamil), *kihada* (Jepang), *lamatra* (Malagasi), *maha'o* (Hawai), *otara* (Tahiti), *palaha* (Hawai), *rabil* (Spanyol), *te*

baibo (Kiribati), *thon jaune* (Perancis), *thunfisch* (Jerman), *tonno albacore* (Italia), *tonnos macropteros* (Yunani), *tuna* (Afrika), *tunczyk zółtopletwy a. albakora* (Polski), *yatu* (Fijian), *zheltokhvostyj tunets* (Russia), and *zutorepi tunj* (Serbia).

Menurut Saanin (1984), klasifikasi ikan tuna sirip kuning adalah sebagai berikut:

Phylum: Chordata
 Subphylum: Vertebrata Thunnus
 Class: Teleostei
 SubClass: Actinopterygii
 Ordo: Perciformes
 Subordo: Scombroidae
 Genus: Thunnus
 Species: *Thunnus albacores*



Sumber: <http://www.sustainablesushi.net>

Gambar 58 Ikan Madidihang.

Menurut literatur yang bersumber dari situs web luar negeri yang ditemukan (<http://www.flmnh.ufl.edu>) *Yellowfin tuna* dapat ditemukan di perairan tropis maupun subtropis, dari lintang antara 40°U hingga 35°S namun tidak terdapat di Laut Mediterania. *Yellowfin tuna* merupakan ikan yang bermigrasi.

Ikan *yellowfin tuna* merupakan jenis *epipelagic oceanic fish*, hidup di atas dan di bawah *thermocline*, pada temperatur 65 hingga 88°F (18-31°C). Secara umum

dapat ditemukan di atas 330 kaki (100 meter) dari kolom air dan hidup secara berkelompok. Kecenderungan untuk berkelompok dengan organisme yang berukuran sama lebih kuat daripada kecenderungan untuk berkelompok berdasarkan spesies. Mereka sering berenang bersama bercampur dengan kelompok lain seperti *skipjack*, *bigeye*, dan tuna lainnya. Di sebelah timur laut pasifik, *yellowfin* yang berukuran lebih besar secara sering berkelompok berasosiasi dengan lumba-lumba, khususnya dengan *spotted dolphin*, *spinner dolphin*, dan dolphin biasa. Asosiasi dengan lumba-lumba seperti ini belum pernah diteliti di laut Pasifik, Hindia, dan Atlantik. *Yellowfin* akan segera bergerombol di bawah benda yang terapung seperti kayu hanyut, petak-petak rumput laut, perahu, atau mamalia laut yang mati. Terdapat banyak hipotesis mengenai alasan bergelombol di bawah benda-benda ini. *Yellowfin* bisa jadi tertarik oleh objek untuk memakan mangsa yang lebih kecil yang berkeliaran di sekitar benda terapung tersebut. Objek yang melayang memberikan keteduhan dan perlindungan dari predator. *Yellowfin* tuna dapat memanfaatkan objek sebagai substrat untuk bertelur atau sebagai "stasiun pembersihan," di mana parasit dikeluarkan oleh ikan lain. Selain itu, ikan dapat melihat objek sebagai satu kelompok yang sama dengan mereka. Ketika *Yellowfin tuna* berenang lebih jauh dari permukaan, mereka cenderung untuk tidak bergerombol, cenderung menyebar. Ada kemungkinan mengapa mereka tidak bergerombol, karena terdapat lebih sedikit predator dan berusaha untuk mendapatkan makanan di kedalaman.

Yellowfin tuna adalah ikan tuna besar. Tubuhnya sangat fusiformis, Terdapat dua sirip punggung. Pada saat dewasa, sirip dorsal kedua sangat panjang, seperti sirip anus, yang langsung di bawah dorsal kedua. Sirip ini menjadi relatif lebih panjang pada individu yang lebih besar. Sirip dada panjang, mencapai luar ruang antara sirip punggung. *Caudal peduncle* sangat ramping dan termasuk tiga set *keels*. Terdapat tujuh sampai sepuluh dorsal dan ventral finlets. Sebuah pita membentuk lingkaran di sekeliling tubuh di belakang kepala. Mata kecil, gigi kecil dan kerucut.

Tubuh metalik biru gelap atau kehijauan di atas, sementara perut dan sisi bawah berwarna putih perak dan dipotong oleh banyak garis vertikal. Tanda yang paling paling jelas, terdapat garis emas di sepanjang sisi. Dorsal kedua dan sirip anal dan

finlets berwarna kuning cerah, dan *finlets* dibatasi oleh pita tipis berwarna hitam. Panjang maksimum dapat mencapai 110 inch (280 cm) dan berat maksimum 880 lbs (400 kg). Ukuran saat dewasa variatif tergantung wilayah, dan dapat pula terjadi perbedaan antara individu yang ditemukan di dekat pantai dan lepas pantai. Semua yellowfin dewasa bereproduksi saat mencapai panjang 47 inch (120 cm) atau berumur sekitar 2-3 tahun), namun beberapa diantaranya dewasa saat memiliki panjang 20-23 inch (50-60 cm) atau sekitar umur 12-15 bulan.

5.4 Jarak Pemasangan Rumpon

Berdasarkan data GPS lokasi pemasangan rumpon, terdapat enam lokasi pemasangan rumpon. Semua rumpon dipasang pada koordinat sekitar 7° lintang selatan dan di antara 105° hingga 107° bujur timur. Masing-masing rumpon pada GPS diberi nama, diantaranya adalah PSP, Bintang Mas, Setia, Perkasa, Budi, dan Baen. Rumpon dipasang dengan jarak berbeda satu sama lain. Berikut merupakan data GPS pemasangan rumpon.

Tabel 9 Posisi Pemasangan Rumpon

POSISI	RUMPON					
	PSP	Bintang Mas	Setia	Perkasa	Budi	Baen
Lintang (Selatan)	$7^{\circ} 33.540'$	$7^{\circ} 44.031'$	$7^{\circ} 20.463'$	$7^{\circ} 14.291'$	$7^{\circ} 26.514'$	$7^{\circ} 15.133'$
Bujur (Timur)	$106^{\circ} 12.370'$	$106^{\circ} 48.416'$	$105^{\circ} 26.916'$	$105^{\circ} 16.954'$	$105^{\circ} 37.595'$	$105^{\circ} 33.619'$

Jarak antar dua rumpon dihitung menggunakan rumus Teorema Pitagoras (*Pythagoras' theorem*), dimana perbedaan lintang diasumsikan sebagai “y” dan perbedaan bujur diasumsikan sebagai “x”, sedangkan jarak kedua rumpon tersebut diasumsikan sebagai “z”. Perhitungan diawali dengan melakukan konversi nilai satuan derajat menjadi satuan menit, sehingga hasil akhir memiliki satuan menit. Hasil akhir tersebut dikalikan dengan jarak per satuan menit, dimana satu menit sama dengan satu mil laut dan satu mil laut sama dengan 1,852 km.

Misalnya antara Rumpon Perkasa dengan Rumpon Setia memiliki jarak antar lintang $-6,17$ menit dan jarak antar bujurnya $9,96$ menit, maka proses perhitungannya adalah sebagai berikut:

$$x = 9.96' \text{ dan } y = -6.17'$$

$$z = \sqrt{x^2 + y^2}$$

$$z = \sqrt{9.96^2 + (-6.17)^2}$$

$$z = \sqrt{137.27}$$

$$z = 11.72'$$

$$1' = 1 \text{ mil laut}$$

$$z = 11.72' \times 1 \text{ mil}$$

$$z = 11.72 \text{ mil}$$

$$1 \text{ mil laut} = 1.852 \text{ km}$$

$$z = 11.72 \times 1.852$$

$$z = 21.70 \text{ km}$$

Perhitungan jarak antar dua rumpon berdasarkan jarak dua rumpon yang terdekat dimana pada jarak tersebut tidak terhalangi oleh rumpon lain. Dengan asumsi seperti itu, maka hanya ada tujuh jarak antar dua rumpon terdekat. Selain itu tanda negatif pada lintang menunjukkan bahwa lintang tersebut terletak pada posisi lintang selatan. Penggunaan tanda negatif pada lintang tidak mempengaruhi hasil perhitungan.

Tabel 10 Jarak Antar Rumpon

	perkasa-setia	setia-baen	baen-budi	budi-pp	pp-bintang mas	perkasa-baen	setia-budi
lintang	-6.17	-5.33	-11.38	-7.03	-10.49	-0.84	-6.05
bujur	9.96	6.70	3.98	34.77	36.05	16.67	10.68
jarak (menit)	11.72	8.56	12.06	35.48	37.54	16.69	12.27
jarak (km)	21.70	15.86	22.33	65.70	69.53	30.90	22.73

Dari keenam rumpon tersebut, jarak antar rumpon yang paling dekat adalah jarak rumpon Setia-Baen dengan jarak 15.86 kilometer. Jarak rumpon yang paling

jauh adalah jarak rumpon PSP-Bintang Mas dengan jarak 69.53 kilometer. Pola pemasangan diilustrasikan oleh gambar di bawah ini yang merupakan hasil pengolahan aplikasi *Google Earth*.

Di dalam KEP.30/MEN/2004 tentang pemasangan dan pemanfaatan rumpon, jarak pemasangan antar rumpon tidak boleh kurang dari 10 mil atau sekitar 18.52 km, tidak mengganggu alur pelayaran dan cara pemasangan tidak mengakibatkan efek pagar (*zigzag*). Berdasarkan perhitungan jarak antar rumpon dan pengamatan cara pemasangan rumpon, jarak pemasangan antar rumpon dan cara pemasangan rumpon di perairan selatan Palabuhanratu secara umum memenuhi ketentuan, kecuali jarak antara rumpon Setia dengan rumpon Baen.

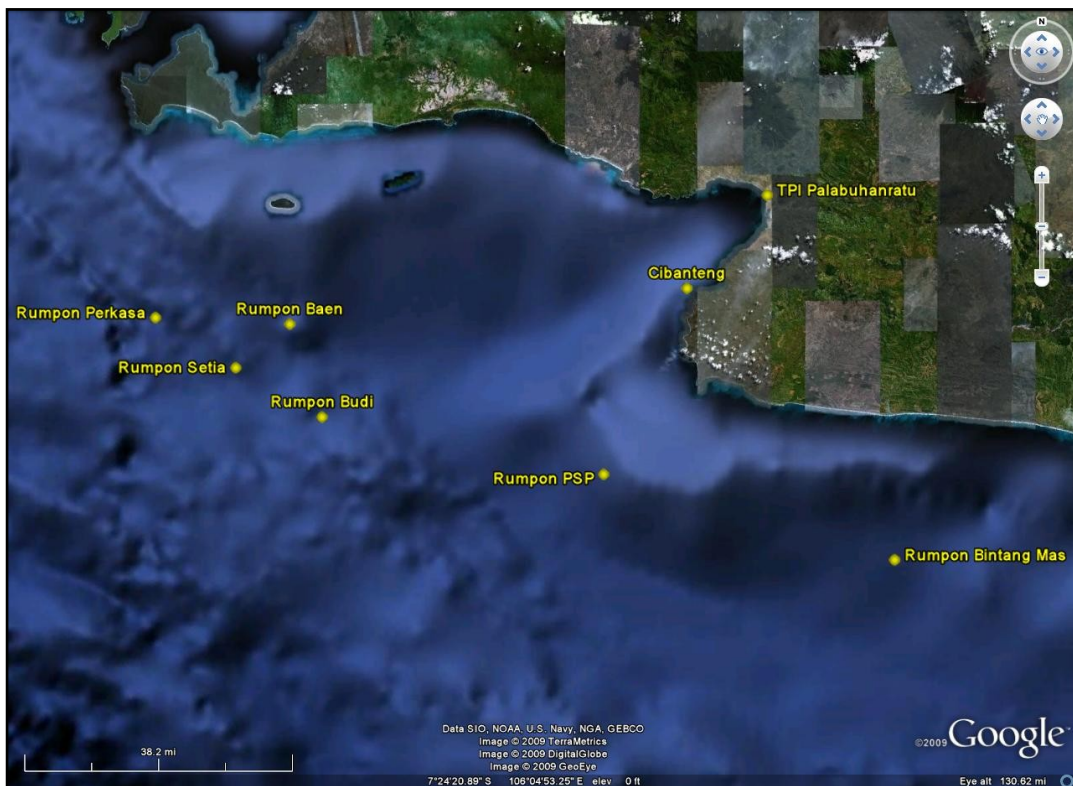
Tabel 11 Jarak Pemasangan Rumpon

Rumpon	<i>Fishing Base</i> (mil)	Pantai Terdekat (mil)
Perkasa	88.18	22.27
Baen	69.86	16.10
Setia	78.84	23.23
Budi	70.14	29.77
PSP	45.19	17.32
Bintang Mas	54.23	20.09

Pada ketentuan tersebut tidak disebutkan secara jelas mengenai garis pantai yang dijadikan sebagai patokan pengukuran wilayah pemasangan rumpon. Dalam ketentuan hanya disebutkan pemasangan rumpon dapat dilakukan di perairan 2 mil hingga 4 mil laut, 4 mil hingga 12 mil laut, dan 12 mil laut hingga ZEE yang diukur dari garis pantai pada titik surut terendah. Jika melihat kondisi geografinya, perairan Palabuhanratu merupakan teluk, sehingga dari perhitungan mengenai wilayah pemasangan rumpon akan timbul pertanyaan. Apakah wilayah pemasangan rumpon diukur pada garis pantai pada titik surut terendah pantai *fishing base* atau wilayah pemasangan rumpon diukur pada garis pantai pada titik surut terendah pantai pulau terluar. Jika berdasarkan tabel jarak pemasangan rumpon, baik pengukuran dari pantai *fishing base* maupun pantai terdekat dimana rumon tersebut dipasang,

menunjukkan bahwa pemasangan rumpon berada pada wilayah 12 mil laut hingga ZEE.

Pembuatan peraturan dalam hal ini adalah keputusan menteri haruslah jelas, ringkas, dan mudah dipahami karena peraturan merupakan patokan atau dasar sebelum melakukan kegiatan, dalam hal ini adalah pemasangan dan pemanfaatan rumpon. Selain itu, harus ada sosialisasi kepada pihak-pihak yang terkait dalam peraturan tersebut agar peraturan bisa diterapkan dengan baik. Terakhir, dalam penerapannya harus diawasi, pelanggar terhadap peraturan tersebut harus ditindak dengan tegas, serta evaluasi peraturan secara berkelanjutan.



Sumber: Google Earth

Gambar 59 Pola pemasangan rumpon.

Saat berangkat dari *fishing base* menuju *fishing ground*, kapal tidak langsung menuju lokasi *fishing ground* sesuai data posisi yang terdapat pada GPS, melainkan menuju data posisi Cibanteng terlebih dahulu. Dari posisi Cibanteng inilah, kapal

langsung menuju ke lokasi *fishing ground* sesuai dengan data posisi pada GPS. Hal ini dilakukan karena jika dari *fishing base*, kapal langsung diarahkan menuju posisi *fishing ground*, maka kapal akan menabrak daratan. Begitu juga ketika akan pulang menuju *fishing base*, kapal diarahkan menuju Cibanteng terlebih dahulu, baru setelah itu kapal diarahkan menuju *fishing base*.

5.5 SoFAD

SoFAD merupakan kependekan dari *Software of Fish Aggregating Device*. Perangkat lunak ini dibuat untuk menghitung volume, massa, massa jenis, *buoyancy force*, *sinking force*, serta *extra buoyancy* dari komponen rumpon laut dalam yang telah diukur. Perangkat lunak ini dibuat menggunakan perangkat lunak *Borland Delphi 7* dengan bahasa pemrograman *pascal*. Pada aplikasi ini tidak memperhitungkan arus perairan, dengan kata lain menggunakan asumsi bahwa kondisi perairan tenang tidak berarus.

Ketika aplikasi SoFAD dijalankan, muncul aplikasi Menu yang terdiri dari enam belas pilihan tombol yang diawali dengan munculnya tampilan *splash screen* selama 10 detik. Pada saat *splash screen* ini muncul, aplikasi SoFAD yang terdiri dari beberapa aplikasi ini melakukan inisialisasi aplikasi agar siap digunakan. Masing-masing tombol memiliki nama tersendiri yang terhubung dengan aplikasi komponen rumpon sesuai dengan nama komponen yang terdapat pada tombol. Misalnya, ketika tombol atraktor ditekan, maka aplikasi untuk menghitung komponen atraktor akan muncul. Terdapat tombol Bantuan dan Info pada sebelah kanan atas aplikasi. Tombol Bantuan berisi mengenai keterangan bahan dan petunjuk penggunaan, sedangkan tombol Info berisi informasi mengenai aplikasi.

Pada perangkat lunak Delphi, memunculkan aplikasi lain pada aplikasi SoFAD dilakukan pada *procedure event TsButton onclick*.

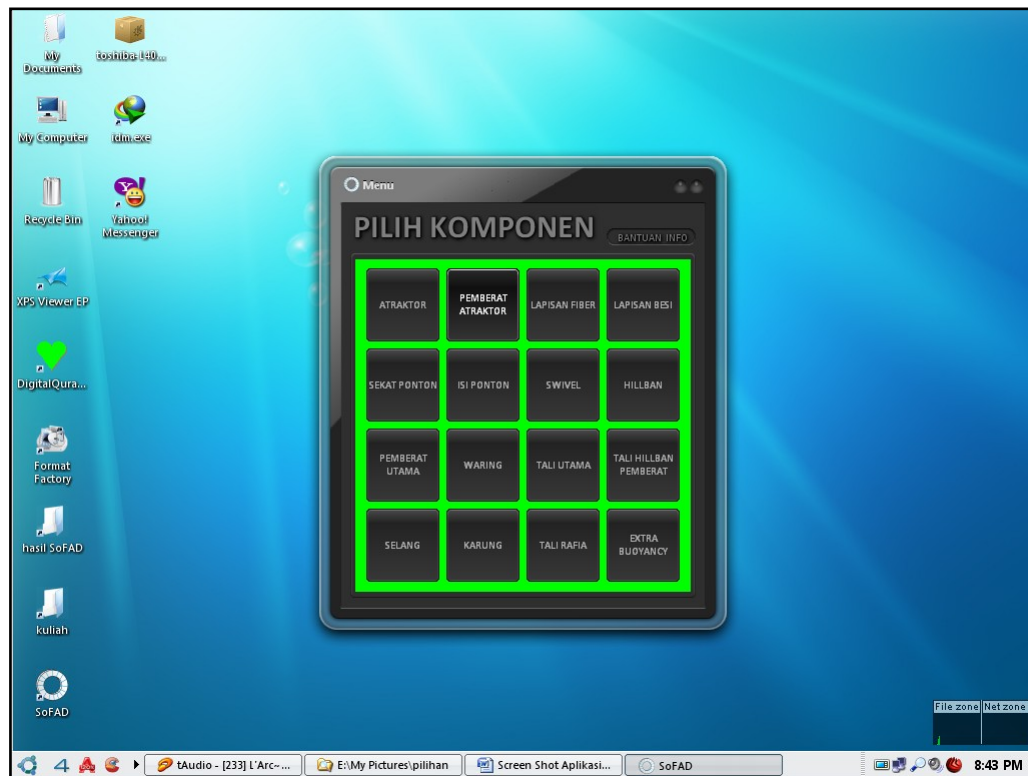
```
procedure TMain_Menu.TsButton1Click(Sender: TObject);
Var
pa:TPemberat_Atraktorv;
begin
```

```

pa:=Pemberat_Atraktorv;
pa.Show;
end;

```

Berdasarkan *source code* di atas, aplikasi atraktor yang dideklarasikan dalam *variable* dan diinisialisasikan dengan 'pa' akan muncul jika *TsButton1* pada aplikasi Main Menu ditekan.



Gambar 60 Aplikasi menu SoFAD.

Pada aplikasi atraktor dan aplikasi lain selain aplikasi menu dan *extra buoyancy* terdiri dari tiga bagian. Bagian pertama adalah bagian input, bagian kedua adalah bagian output, dan bagian ketiga untuk menampilkan input dan output. Khusus untuk bagian ketiga ini hanya dimunculkan pada aplikasi *extra buoyancy*. Bagian ketiga ini merupakan komponen *TsListBox* yang berfungsi selain untuk menampilkan input dan

output, juga sebagai penampung data sementara yang nantinya akan digunakan dalam proses penyimpanan dokumen kedalam bentuk *text document*.

Bagian input dan output terdiri dari komponen *TEdit*, sedangkan bagian untuk melihat nilai input dan output merupakan komponen *TsListBox*. Selain itu, terdapat tiga komponen *TsButton*. *TsButton* pertama memiliki *caption* Hitung, *TsButton* kedua memiliki *caption* Hapus, dan *TsButton* ketiga memiliki *caption* Simpan.

Karakter yang bisa dimasukkan ke dalam komponen *TEdit* pada bagian input maksimal berjumlah 19 karakter. Komponen *TEdit* bisa dimasukkan karakter baik angka maupun huruf. Jika bagian input dimasukkan karakter huruf, pemasukkan karakter huruf sama dengan tidak memasukkan angka, dengan kata lain akan bernilai nol. Hal tersebut disebabkan *source code* yang memiliki fungsi mengambil data numerik atau angka dari data *string* yang dimasukkan.

```
begin
  Val(Edit1.Text,jatr,Kode);
  Val(Edit2.Text,matr,Kode);
  Val(Edit3.Text,vatr,Kode);
  mtatr:=jatr*matr;
  str(mtatr:0:2,masstotatr);
  Edit4.Text:=masstotatr;
```

Pengambilan data numerik dari data *string* ini berfungsi pada saat menekan *TsButton* Hitung yang dilanjutkan dengan perhitungan komponen berdasarkan data input, memperlihatkan hasil perhitungan pada bagian output dan memperlihatkan seluruh data, baik input dan output pada komponen *TListBox*. Jika saat menekan *TsButton* Hitung, ada nilai nol atau tidak diisi, maka akan muncul *error message invalid floating point operation*. Hal tersebut dikarenakan aplikasi tidak memungkinkan menjalankan perhitungan dengan angka nol. Perhitungan dapat diulangi dengan menekan tombol Hapus.

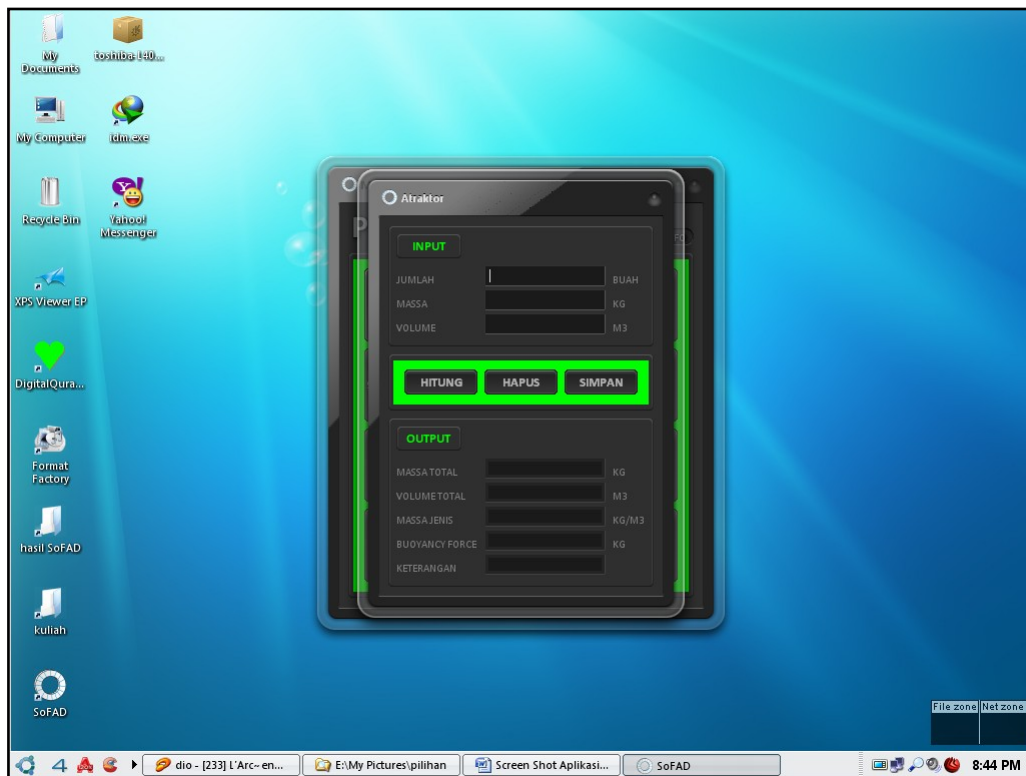
TsButton kedua yang memiliki *caption* Hapus, seperti namanya memiliki fungsi untuk menghapus data, baik pada komponen *TEdit* maupun komponen *TListBox*.

Pada saat *TsButton* Hapus diklik, data pada komponen *TEdit* maupun komponen *TListBox* yang dihapus.

```

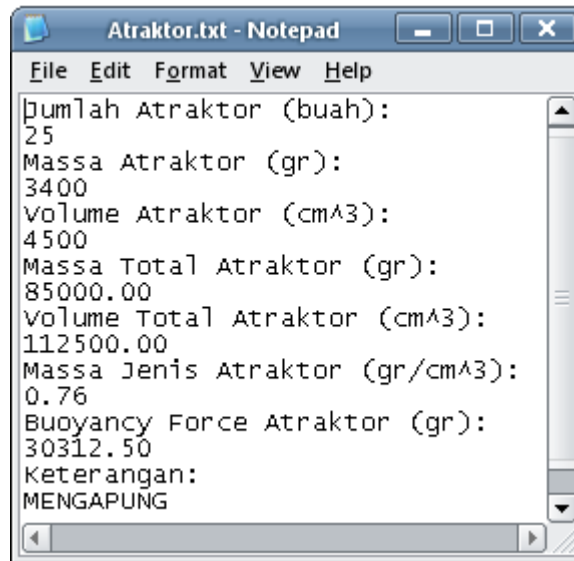
procedure Tatraktorv.sButton2Click(Sender: TObject);
begin
    edit1.Clear;
    edit2.Clear;
    edit3.Clear;
    edit4.Clear;
    Edit5.clear;
    Edit6.Clear;
    Edit7.Clear;
    Edit8.Clear;
    sListBox1.clear; end;

```



Gambar 61 Aplikasi atraktor SoFAD.

end; end;



```

File Edit Format View Help
jumlah Atraktor (buah):
25
Massa Atraktor (gr):
3400
Volume Atraktor (cm^3):
4500
Massa Total Atraktor (gr):
85000.00
Volume Total Atraktor (cm^3):
112500.00
Massa Jenis Atraktor (gr/cm^3):
0.76
Buoyancy Force Atraktor (gr):
30312.50
Keterangan:
MENGAPUNG

```

Gambar 62 *Saved File* Atraktor.txt.

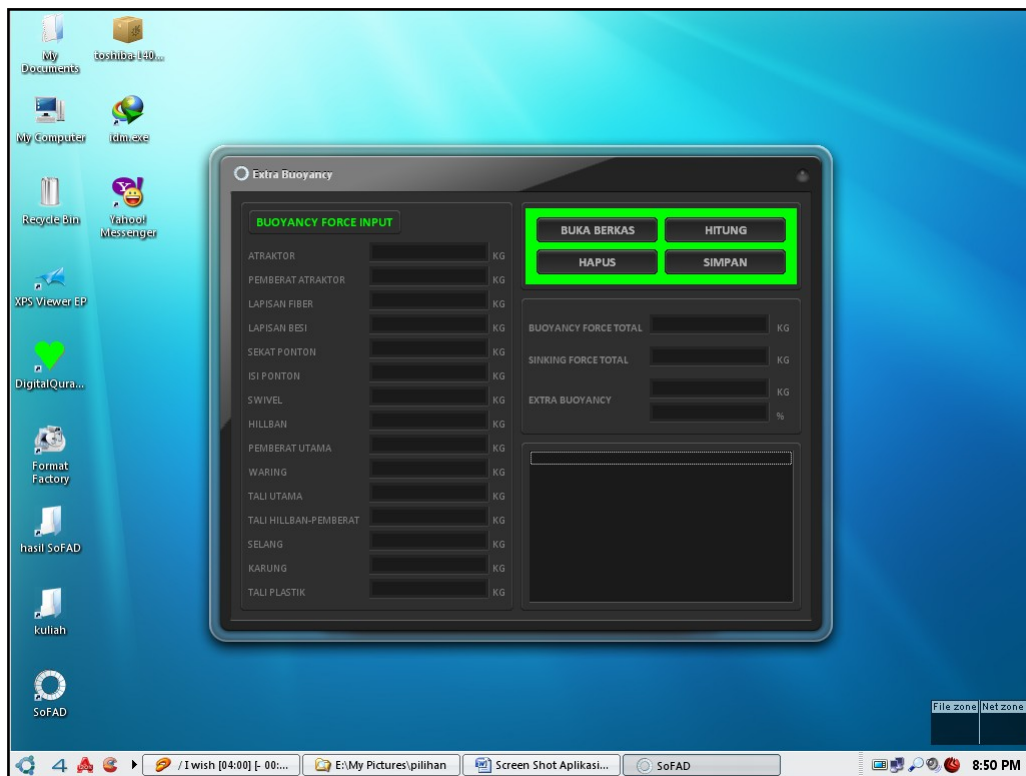
Source code di atas memiliki arti bahwa *item* yang terdapat pada komponen *TListBox* berasal dari *file* Atraktor.txt. Selanjutnya, komponen *TEdit* hanya membaca dan menampilkan data yang berada pada *lines* ke-13, yaitu *lines* yang memiliki data *buoyancy force*. *Lines* pertama pada komponen *TListBox* memiliki nilai nol, sehingga data *buoyancy force* terdapat pada *lines* ke-13.

Saved file komponen atraktor yang telah dihitung, akan lebih baik jika berada dalam satu *folder*. Jika berbeda *folder*, ketika *TsButton* Buka Berkas diklik akan muncul *error message* yang memberitahukan bahwa *saved file* komponen tertentu tidak ditemukan. Aplikasi tetap berjalan, namun pada aplikasi akan ada komponen *TEdit* yang tidak terisi atau kosong yang merupakan komponen yang tidak ditemukan tersebut.

Ketika *TsButton* hitung diklik, maka *TEdit* *Bouyancy Force Total*, *Sinking force Total*, *Extra Buoyancy* akan terisi. Semua nilai *buoyancy force* dan *sinking force* dari semua komponen rumpon dan hasil dari perhitungan *extra buoyancy* dapat dilihat

pada bagian bawah yaitu pada komponen *TListBox*. Seperti aplikasi sebelumnya, hasil perhitungan disimpan dengan nama *file* “Extra Buoyancy.txt”.

Aplikasi ini bersifat *portable* sehingga tidak memerlukan instalasi kedalam *harddisk*. Memiliki ukuran sekitar 4 Mb, dengan hanya melakukan klik ganda, maka aplikasi langsung bisa digunakan. Selain itu, aplikasi SoFAD kompatibel di semua *Windows OS*, bisa juga dijalankan di *Linux OS* bahkan *Mac OS* dengan menggunakan *Windows OS Emulator*. Tema yang digunakan pada SoFAD berasal dari aplikasi *AlphaLite* yang merupakan pengaya dari *Delphi*. Aplikasi ini masih dalam pengembangan dan belum melalui tahap uji publik.



Gambar 63 Aplikasi *extra buoyancy* SoFAD.



Hik Cipta (Indonesia) Limited/United Kingdom

1. Diizinkan mengutip sebagian atau seluruh karya kami jika tanpa memutarbalikkan dan diperbolehkan menyertakan :
- a. Pengutipan harus menyebutkan sumbernya, penulisnya, penerbitan karya ilmiah, jenis-jenis laporan, jenis-jenis karya ilmiah, jenis-jenis karya ilmiah atau tujuan suatu masalah
- b. Pengutipan tidak merugikan kepentingan yang wajar IPB University.
2. Diizinkan mengutip dan memutarbalikkan sebagian atau seluruh karya kami jika dalam bentuk apapun tanpa izin IPB University.

LAMPIRAN

1. Komponen Rumpon

a. Ponton



b. Pemberat utama



d. Tali utama



e. Hillban



f. Atraktor daun kelapa



g. Ponton



2. Alat Penangkapan Ikan

a. Alat tangkap pancing taber



b. Alat tangkap tomba jerigen



c. Alat tangkap pancing coping



d. Alat tangkap pancing tonda



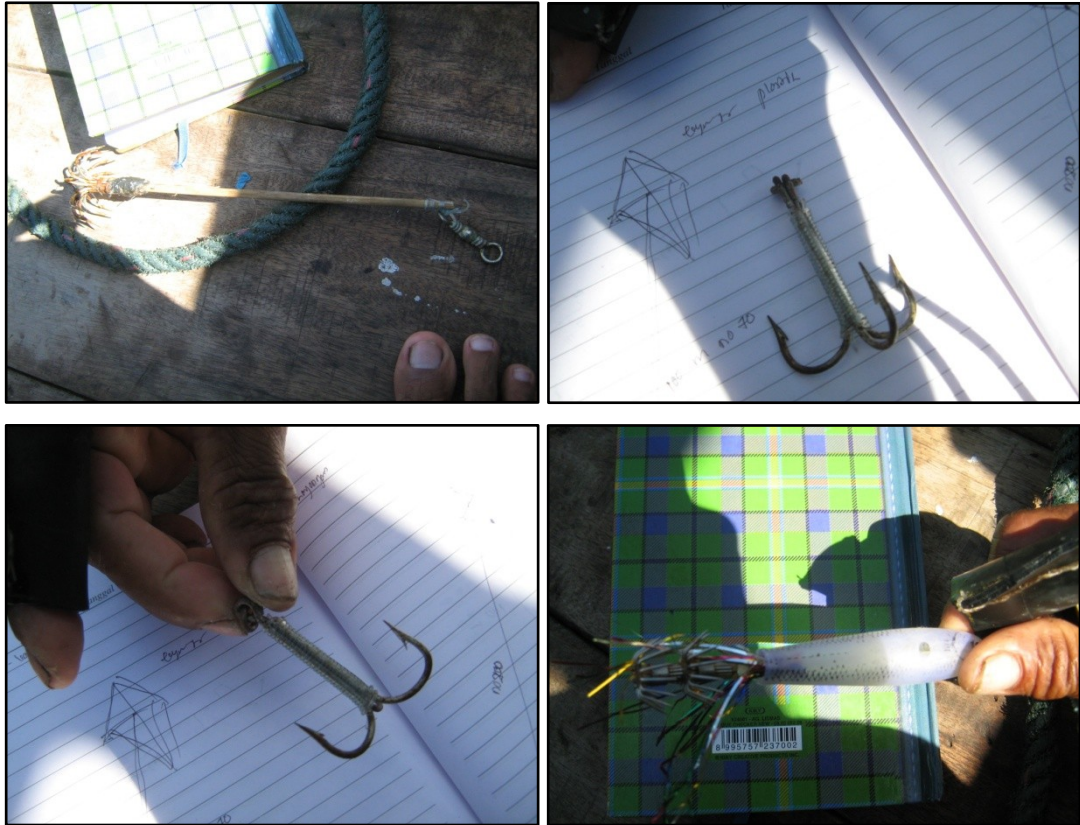
e. Line roller



f. Umpan cumi buatan



g. Multiple hook



h. Layangan plastik



3. Hasil Tangkapan
a. Ikan umpan hidup





b. Kumpulan ikan umpan



c. Hasil tangkapan utama



4. Data Pengukuran dan Hasil Perhitungan

ATRAKTOR			PEMBERAT ATRAKTOR		
No	Data Pengukuran	Ukuran	No	Data Pengukuran	Ukuran
1	JumlahAtraktor (buah):	25	1	JumlahPemberatAtraktor (buah):	1
2	Massa Atraktor (gr):	3400	2	PanjangPemberatAtraktor (cm):	24
3	Volume Atraktor (cm ³):	4500	3	LebarPemberatAtraktor (cm):	10
4	Massa Total Atraktor (gr):	85000	4	TinggiPemberatAtraktor (cm):	5
5	Volume Total Atraktor (cm ³):	112500	5	Massa PemberatAtraktor (gr):	2500
6	Massa JenisAtraktor (gr/cm ³):	0,76	6	Massa Total PemberatAttraktor (gr):	2500
7	Buoyancy Force Atraktor (gr):	30312,5	7	Volume Total PemberatAtraktor (cm ³):	1200
8	Keterangan:	MENGAPUNG	8	Massa JenisPemberatAtraktor (gr/cm ³):	2,08
			9	Buoyancy Force PemberatAtraktor (gr):	-1270
			10	Keterangan:	TENGGELAM
LAPISAN BESI			LAPISAN FIBER		
No	Data Pengukuran	Ukuran	No	Data Pengukuran	Ukuran
1	KetebalanBesi (cm):	0,3	1	Ketebalan Fiber (cm):	0,3
2	PanjangTabung (cm):	358,4	2	PanjangTabung (cm):	359
3	TinggiKerucut (cm):	37,4	3	Diameter (cm):	77
4	Jari-jari (cm):	37,9	4	TinggiKerucut (cm):	38
5	Massa Jenis (gr/cm ³):	7,86	5	Massa Jenis (gr/cm ³):	1,14
6	Volume TabungBesi (cm ³):	25591,05	6	Volume Kerucut (cm ³):	918,76
7	Volume Kerucut (cm ³):	890,16	7	Volume Penutup (cm ³):	1396,28
8	Volume Penutup (cm ³):	1353,1	8	Volume Tabung (cm ³):	26039,71
9	Volume LapisanBesi (cm ³):	27834,31	9	Volume Fiber (cm ³):	28354,75
10	Massa Besi (gr):	218777,7	10	Massa Fiber (gr):	32324,41
11	Buoyancy Force (gr):	-190247,53	11	Buoyancy Force (gr):	-3260,80
12	Keterangan:	TENGGELAM	12	Keterangan:	TENGGELAM

TALI UTAMA			TALI HILLBAN-PEMBERAT		
No	Data Pengukuran	Ukuran	No	Data Pengukuran	Ukuran
1	PanjangprGulung (cm):	8000	1	JumlahGulungTali (gulung):	2
2	JumlahGulung (gulung):	68	2	Panjang per GulungTali (cm):	5000
3	Jari-jariCabang (cm):	0,6	3	Jari-jariTali (cm):	0,15
4	JumlahCabang (cabang):	4	4	Massa JenisTali (gr/cm ³):	0,96
5	Massa JenisTali (gr/cm ³):	0,8	5	Panjang Total Tali (cm):	10000
6	PanjangTali (cm):	544000	6	Volume Total Tali (cm ³):	706,5
7	Volume Tali (cm ³):	2459750,4	7	Massa Total Tali (gr):	678,24
8	Massa Tali (gr):	1967800,32	8	Buoyancy Force Tali (gr):	45,92
9	Buoyancy Force Tali (gr):	553443,84	9	Keterangan:	MENGAPUNG
10	Keterangan:	MENGAPUNG			
SEKAT PONTON			STEROFOAM		
No	Data Pengukuran	Ukuran	No	Data Pengukuran	Ukuran
1	KetebalanSekat (cm):	0,3		PanjangTabung (cm):	358,4
2	Jumlahsekat (buah):	3		Jari-jari (cm):	37,6
3	Jari-jari (cm):	37,6		TinggiKerucut (cm):	37,1
4	Massa Jenis (gr/cm ³):	7,86		Massa Jenis (gr/cm ³):	0,12
5	Volume Sekat (cm ³):	3995,29		Volume Isi Tabung (cm ³):	1591011,57
6	Massa sekat (gr):	31402,95		Massa Isi Tabung (gr):	197509,17
7	Buoyancy Force (gr):	-27307,78		Volume Isi Kerucut (cm ³):	54898,19
8	Keterangan:	TENGGELAM		Volume Isi Ponton (cm ³):	1645909,76
				Buoyancy Force Isi (gr):	1489548,33
				Keterangan:	MENGAPUNG

PEMBERAT UTAMA			HILLBAN		
No	Data Pengukuran	Ukuran	No	Data Pengukuran	Ukuran
1	JumlahPemberat (buah):	21	1	JumlahHillban (buah):	9
2	PanjangPemberat (cm):	40	2	Jari-jariLuarHillban (cm):	29
3	LebarPemberat (cm):	40	3	Jari-jariDalamHillban (cm):	17
4	TinggiPemberat (cm):	20	4	LebarTapakHillban (cm):	16
5	Massa per BuahPemberat (gr):	75000	5	TebalTapakHillban (cm):	1,5
6	Massa Total Pemberat (gr):	1575000	6	TebalSampingHillban (cm):	0,5
7	Volume per BuahPemberat (cm ³):	32000	7	Massa per Hillban (gr):	8000
8	Volume Total Pemberat (cm ³):	672000	8	Volume TapakHillban (cm ³):	4370,88
9	Massa JenisPemberat (gr/cm ³):	2,34	9	Volume SampingHillban (cm ³):	1733,28
10	Buoyancy Force Pemberat (gr):	-886200	10	Volume per Hillban (cm ³):	6104,16
11	Keterangan:	TENGGELAM	11	Volume Total Hillban (cm ³):	54937,44
			12	Buoyancy Force Hillban (gr):	-15689,12
			13	Keterangan:	TENGGELAM
WARING			KARUNG		
No	Data Pengukuran	Ukuran	No	Data Pengukuran	Ukuran
1	PanjangWaring (cm):	3000	1	JumlahKarung (buah):	42
2	LebarWaring (cm):	100	2	Massa per Karung (gr):	224
3	Massa Waring (gr):	4069,44	3	Massa JenisKarung (gr/cm ³):	0,91
4	Massa JenisWaring (gr/cm ³):	0,96	4	Massa Total Karung (gr):	9408
5	Volume Waring (cm ³):	4239	5	Volume per Karung (cm ³):	246,15
6	Buoyancy Force Waring (gr):	275,54	6	Volume Total Karung (cm ³):	10338,46
7	Keterangan:	MENGAPUNG	7	Buoyancy Force Karung (gr):	1188,92
			8	Keterangan:	MENGAPUNG

SWIVEL			SELANG		
No	Data Pengukuran	Ukuran	No	Data Pengukuran	Ukuran
1	Jumlah Swivel (buah):	4	1	PanjangSelang (cm):	19800
2	Massa per Swivel (gr):	838,79	2	Diameter Selang (cm):	3
3	Massa Jenis (gr/cm ³):	7,86	3	KetebalanSelang (cm):	0,2
4	Massa Total Swivel (gr):	3355,16	4	Massa Jenisselang (gr/cm ³):	1,38
5	Volume per Swivel (cm ³):	106,72	5	Volume Selang (cm ³):	37303,2
6	Volume Total Swivel (cm ³):	426,87	6	Massa Selang (gr):	51478,42
7	Buoyancy Force Swivel (gr):	-2917,62	7	Buoyancy Force Selang (gr):	-13242,64
8	Keterangan:	TENGGELAM	8	Keterangan:	TENGGELAM
TALI RAFIA			EXTRA BUOYANCY		
No	Data Pengukuran	Ukuran	No	Data Pengukuran	Ukuran
1	JumlahGulungRafia (gulung):	100	1	Buoyancy Force Total (gr):	2074878,24
2	Panjang per GulungRafia (cm):	3000	2	Sinking Force Total (gr):	1131032,90
3	Massa per GulungRafia (gr):	0,5	3	Extra Buoyancy Force (gr):	943845,34
4	Massa JenisRafia (gr/cm ³):	0,91	4	Extra Buoyancy Force (%):	45,49
5	Panjang Total Rafia (cm):	300000			
6	Massa Total Rafia (gr):	50			
7	Volume Total Rafia (cm ³):	54,95			
8	Buoyancy Force Rafia (gr):	6,32			
9	Keterangan:	MENGAPUNG			

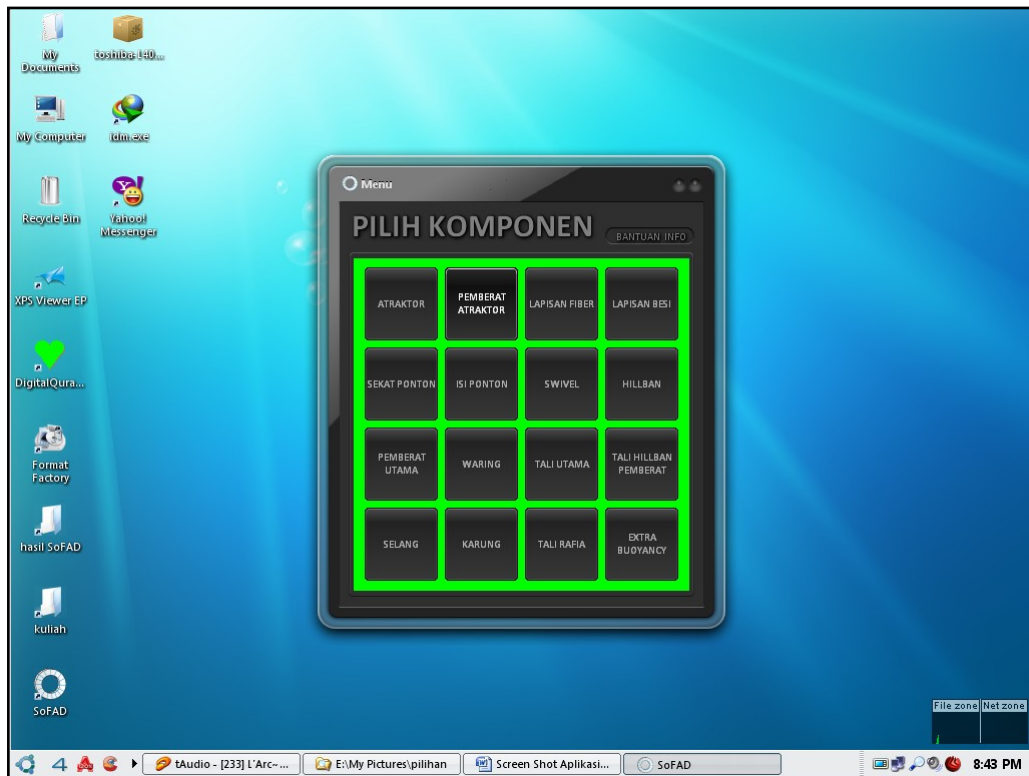


5. **Screenshot Aplikasi SoFAD**

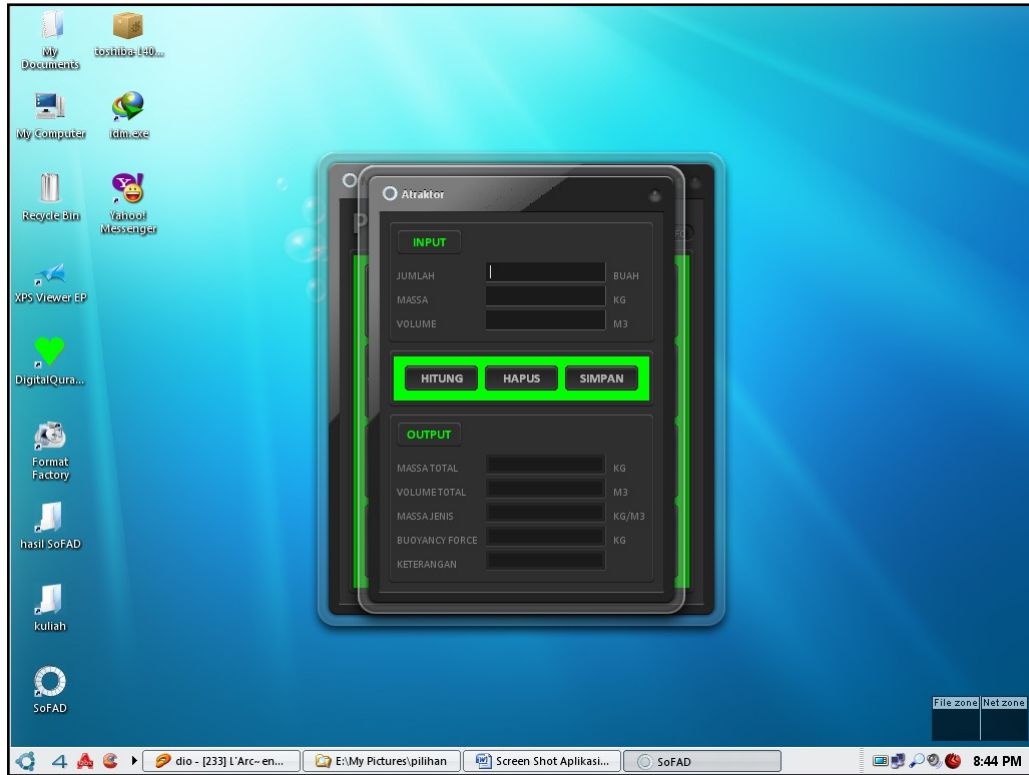
a. *Splash screen*



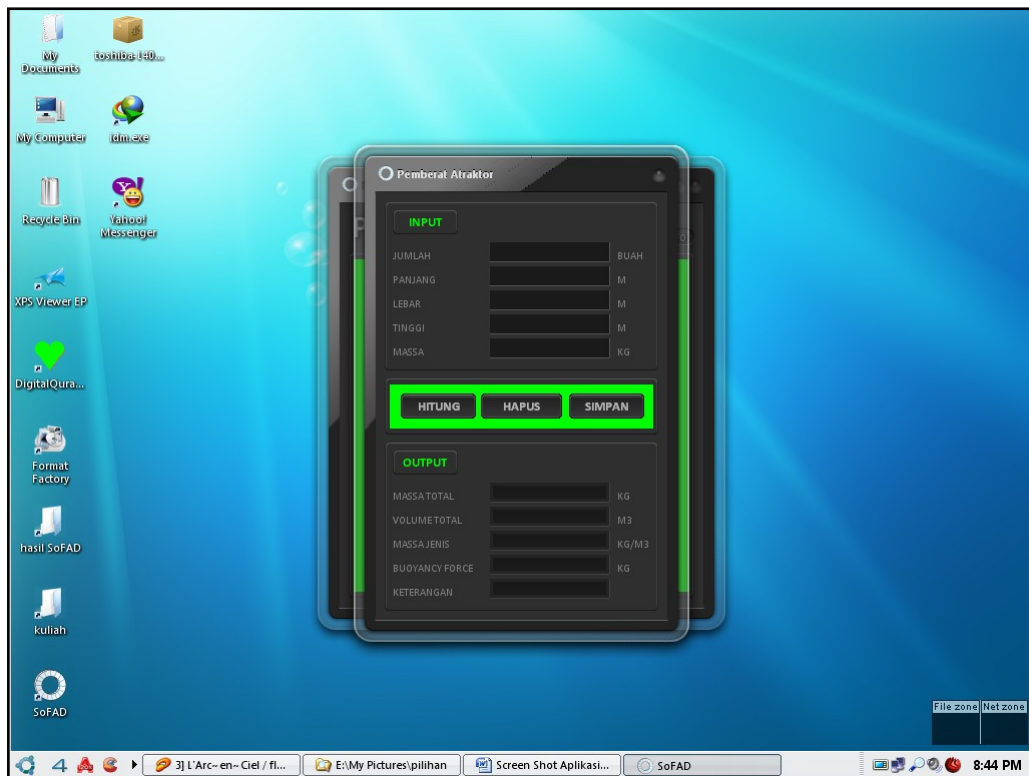
b. **Menu utama**



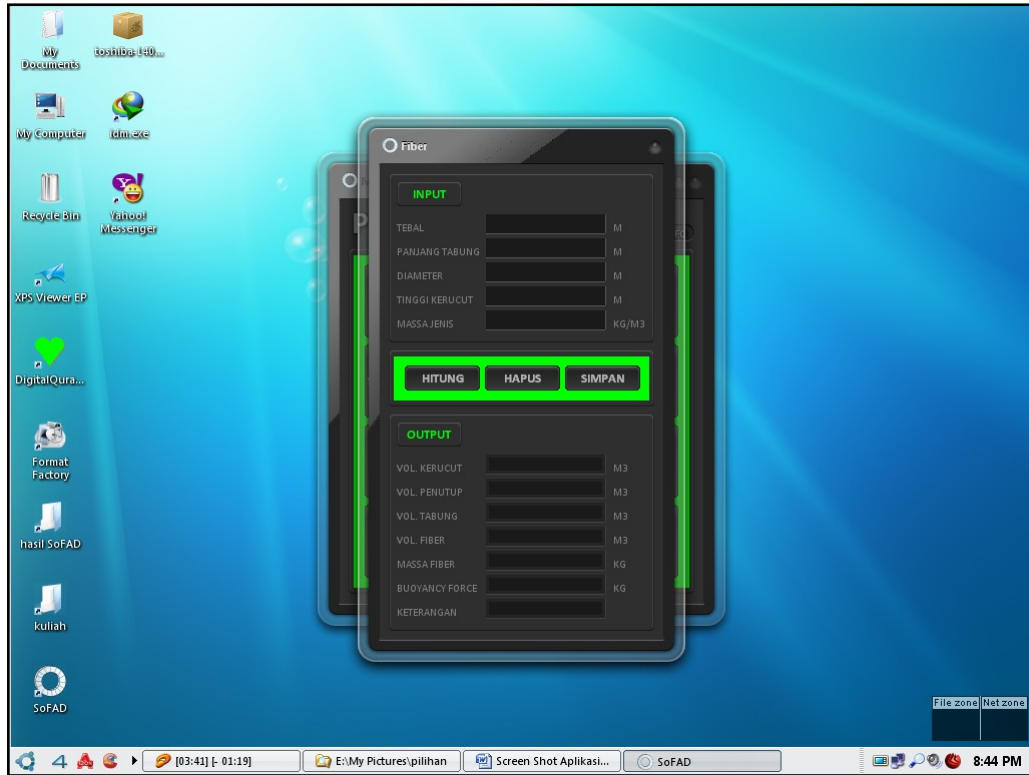
c. Atraktor



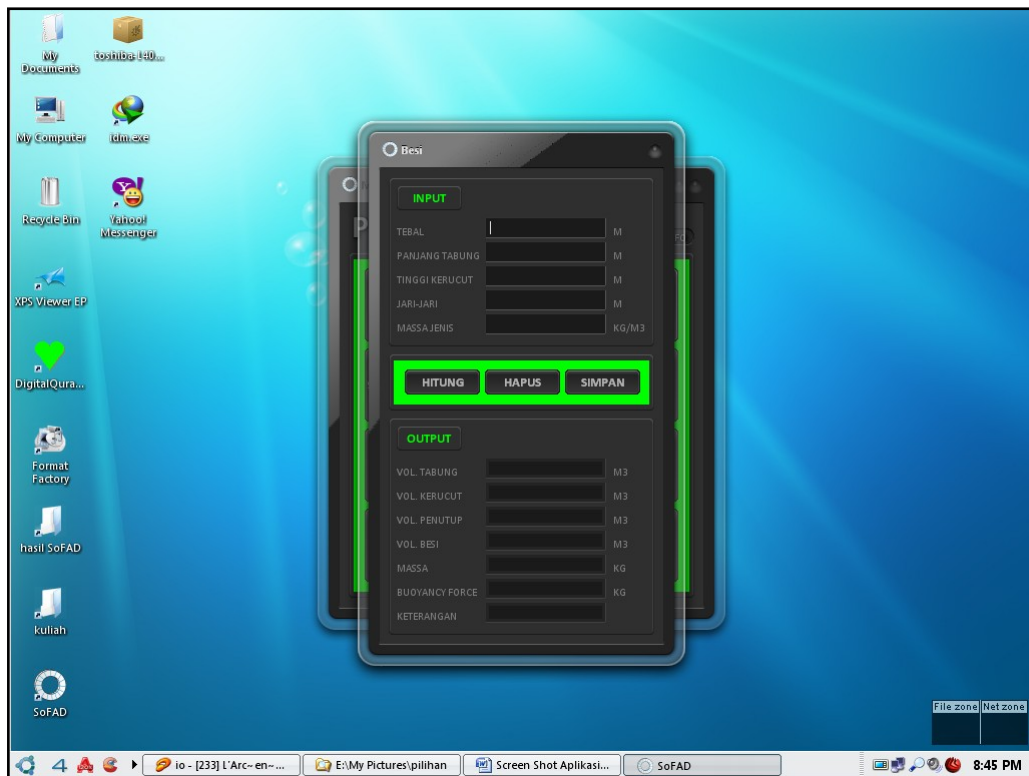
d. Pemberat atraktor



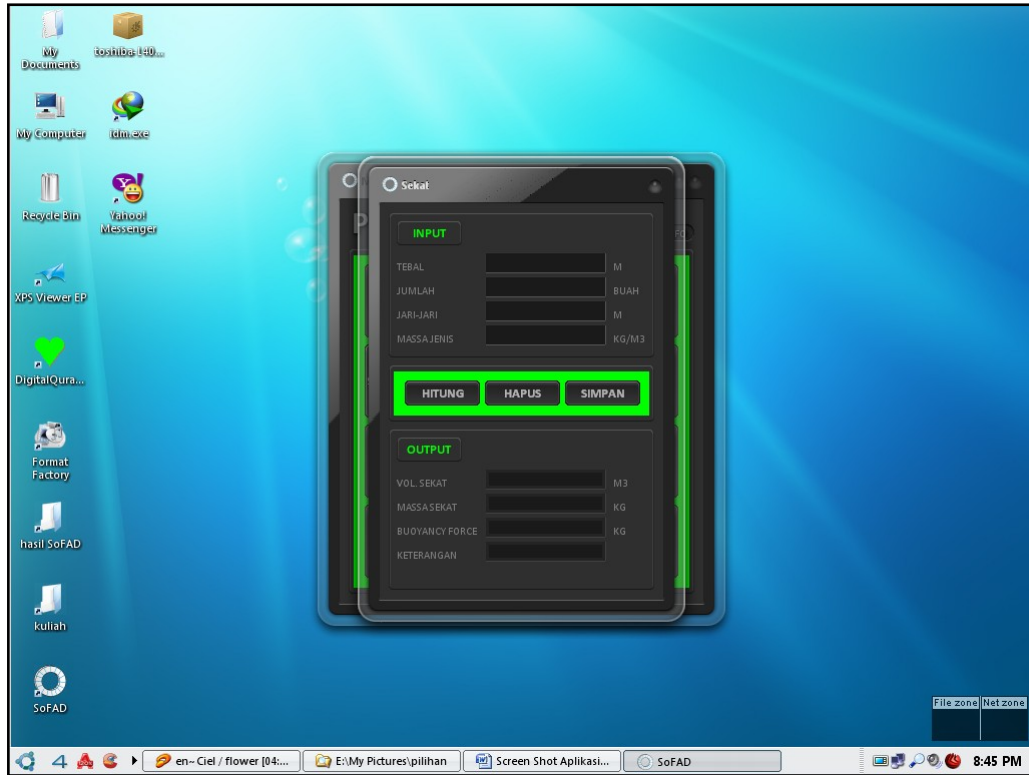
e. Lapisan fiber



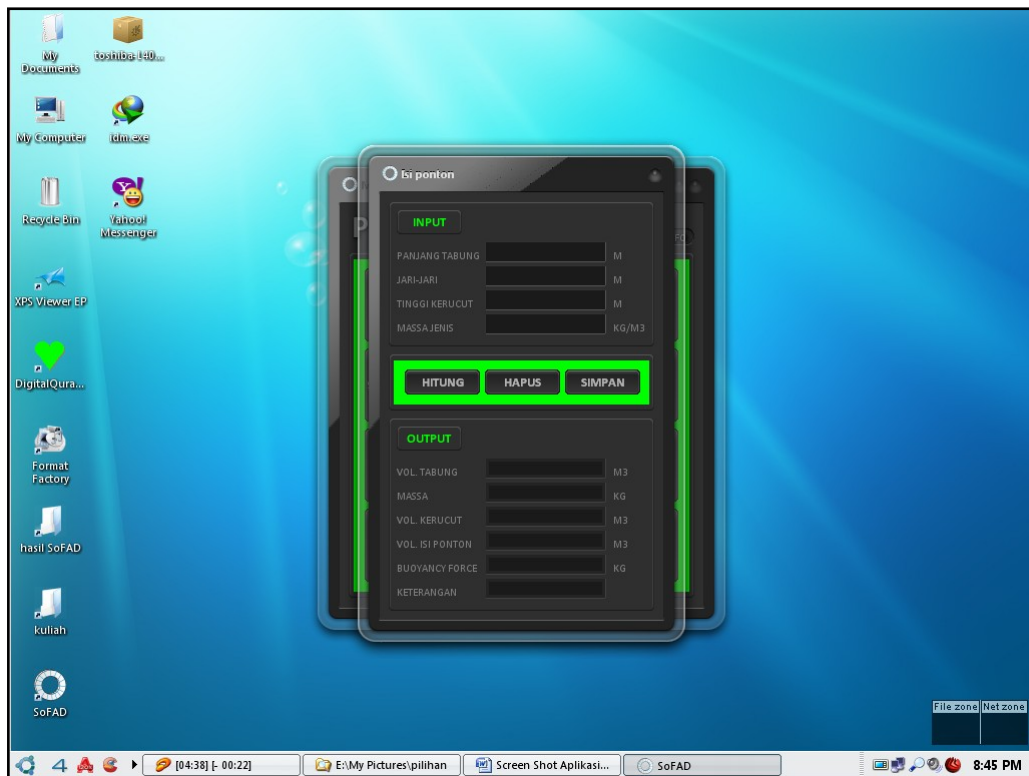
f. Lapisan besi



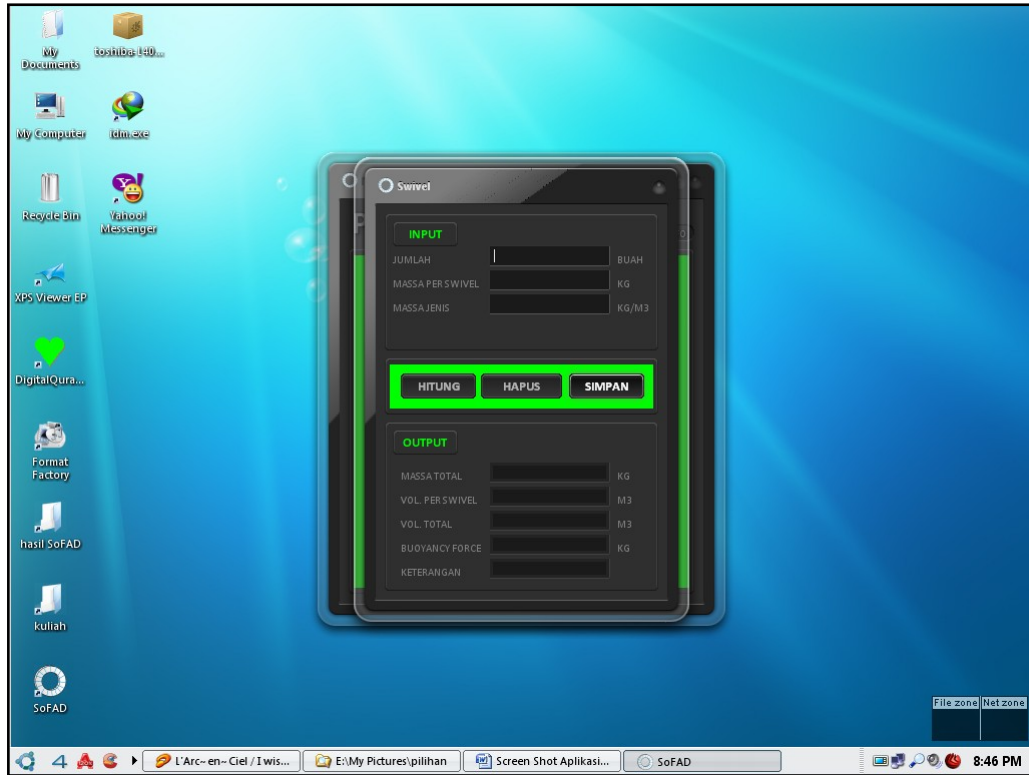
g. Sekat ponton



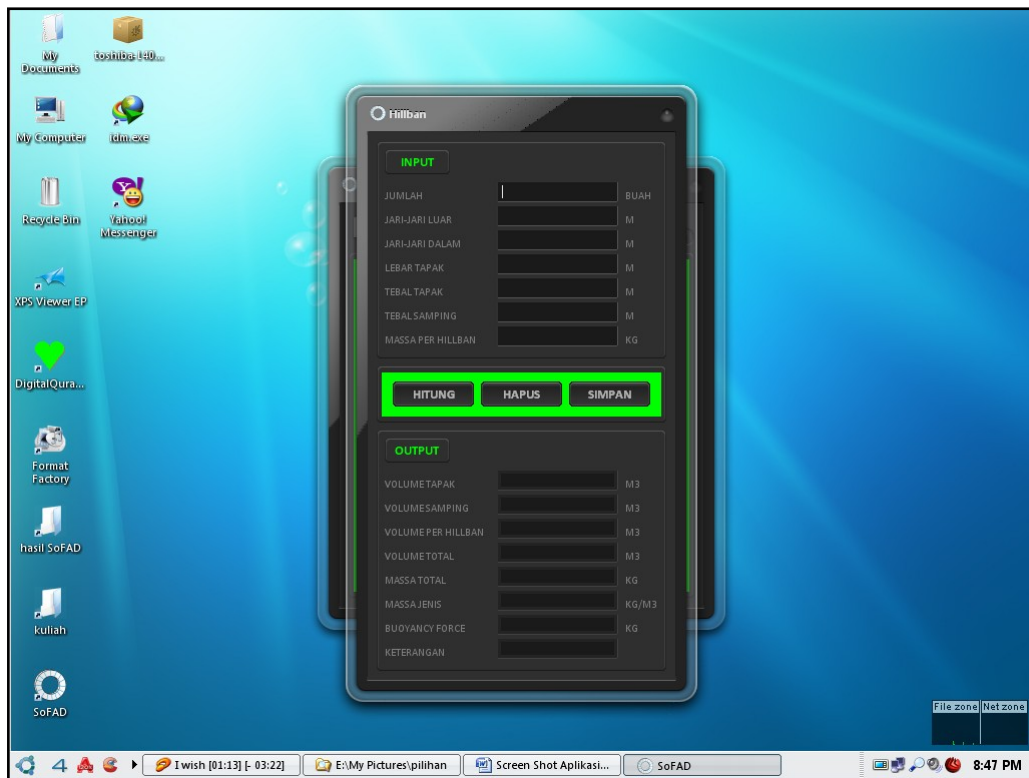
h. Isi ponton



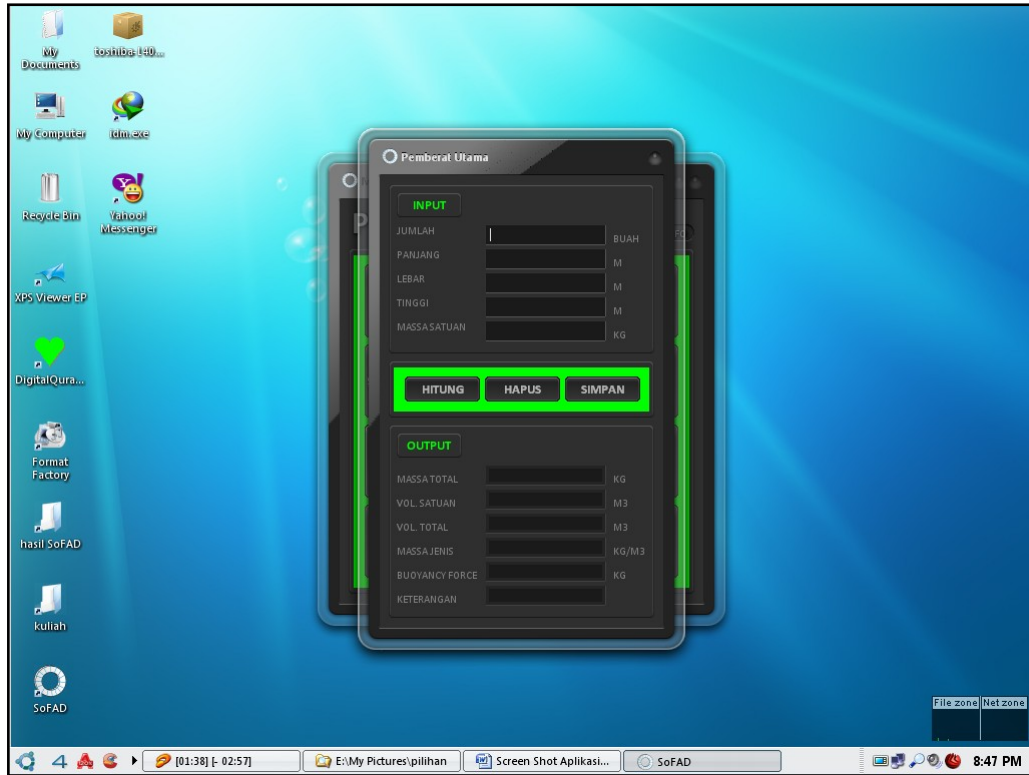
i. *Swivel*



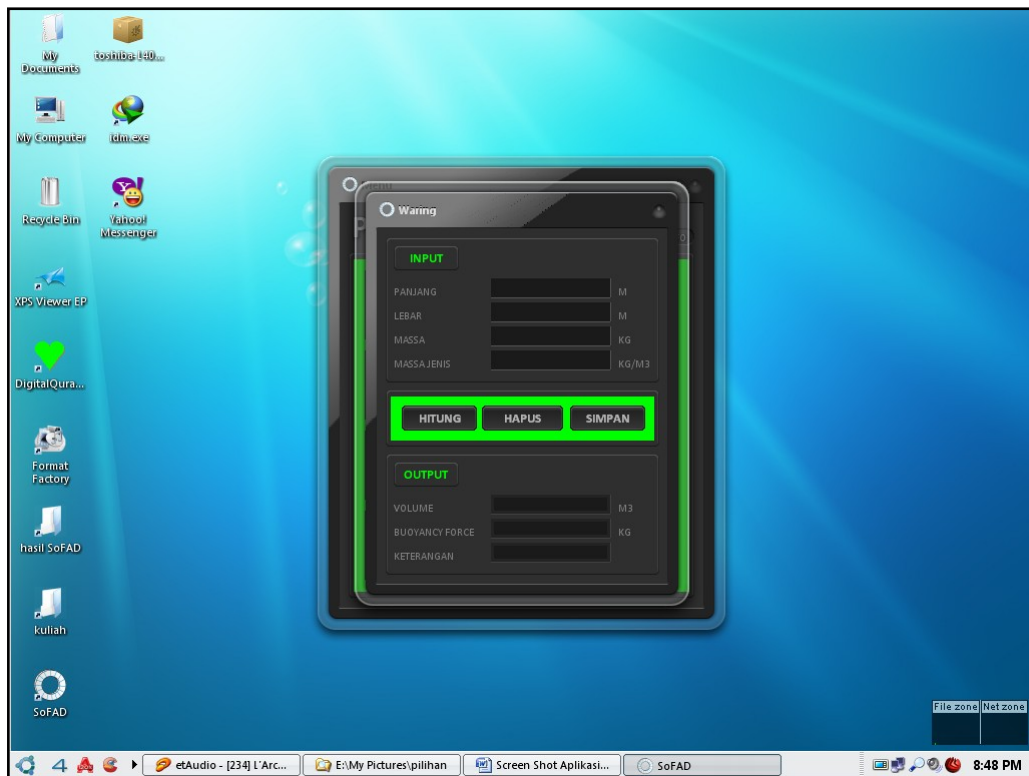
j. *Hillban*



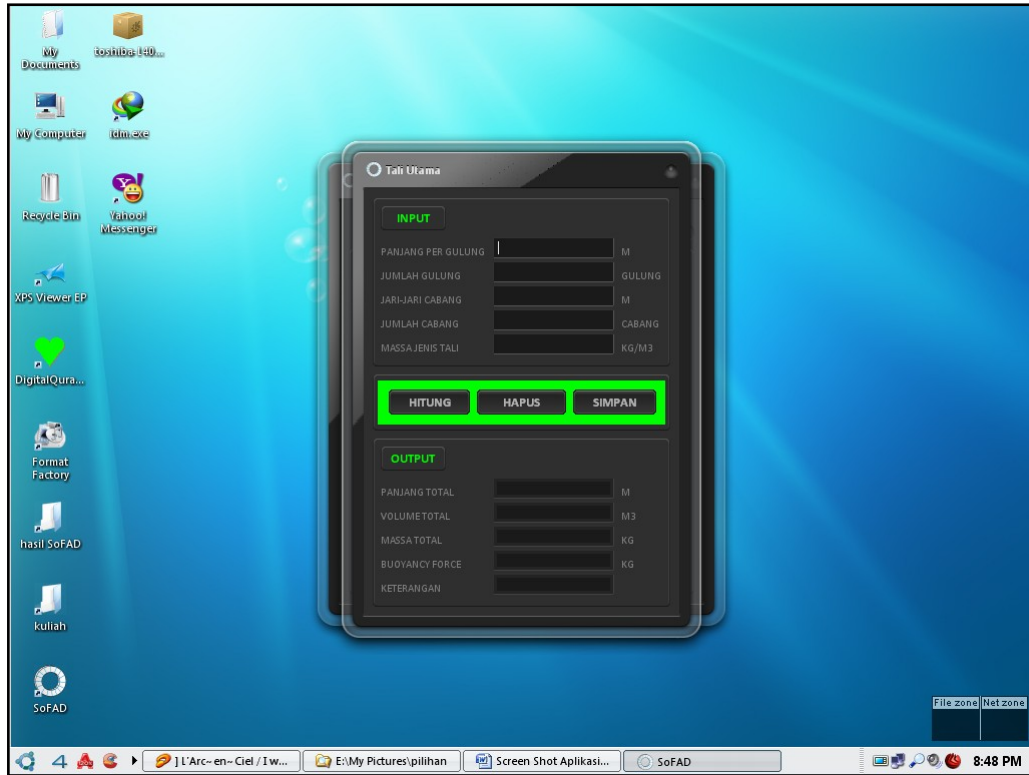
k. Pemberat utama



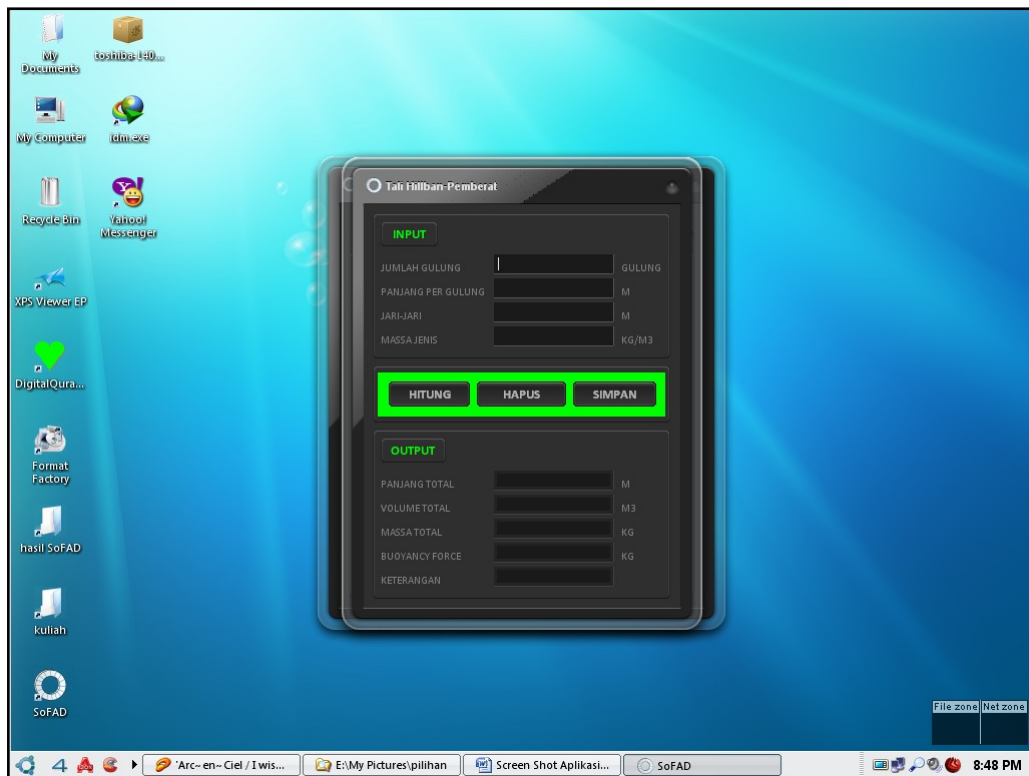
l. Waring



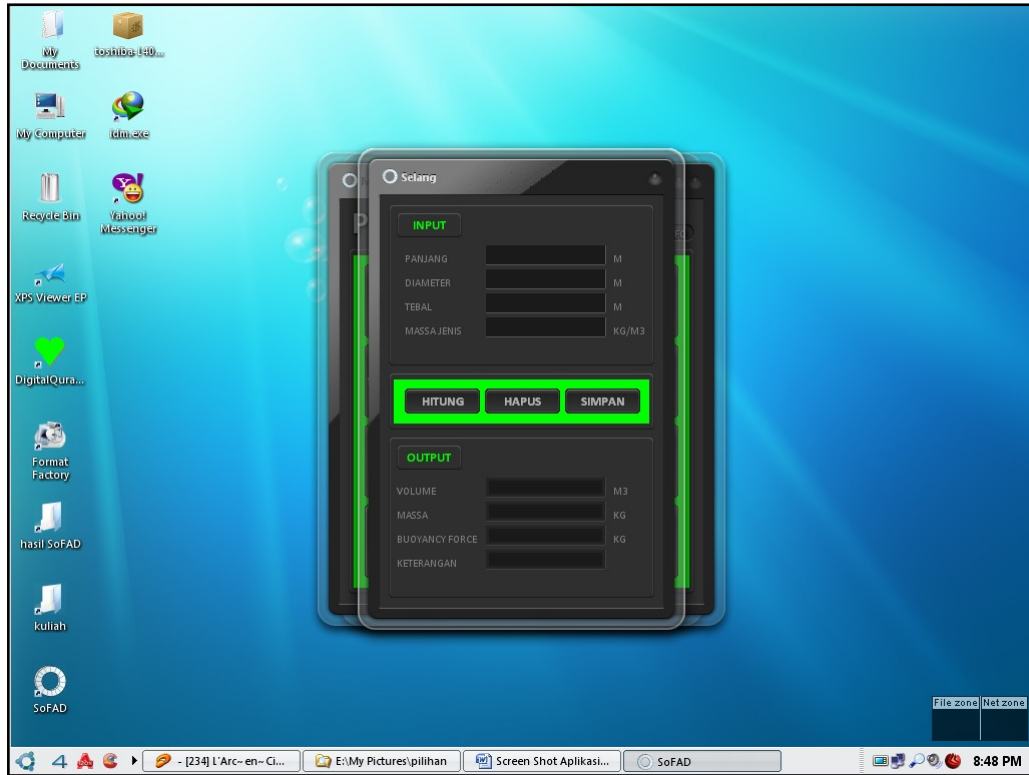
m. Tali utama



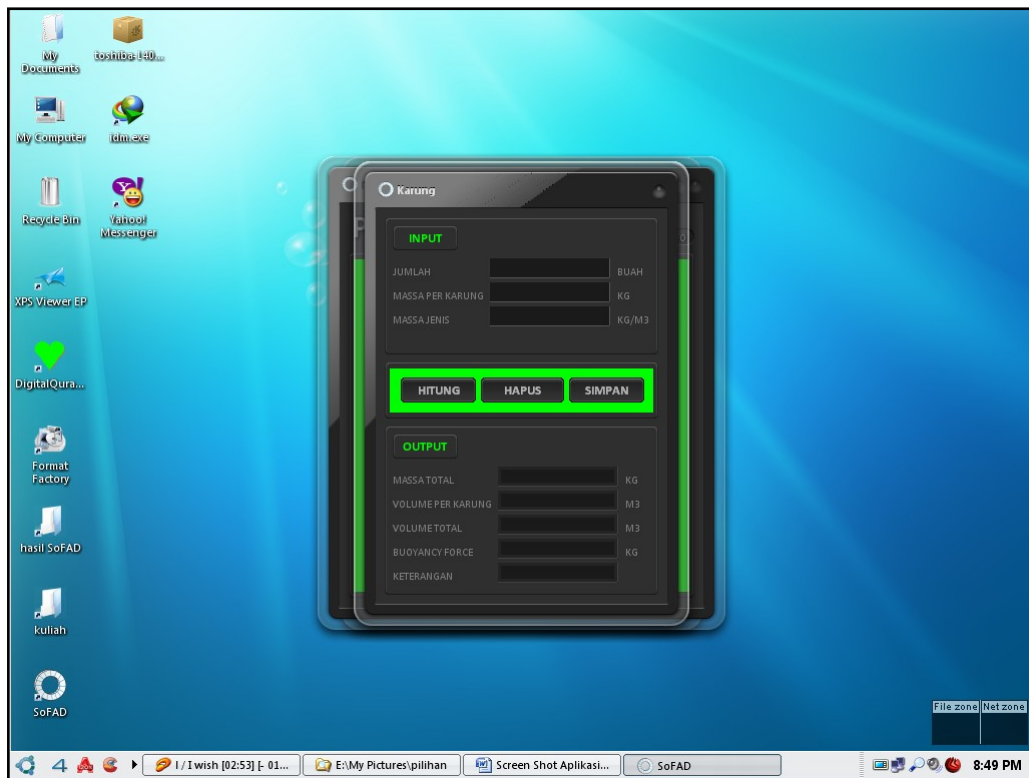
n. Tali hillban-pemberat



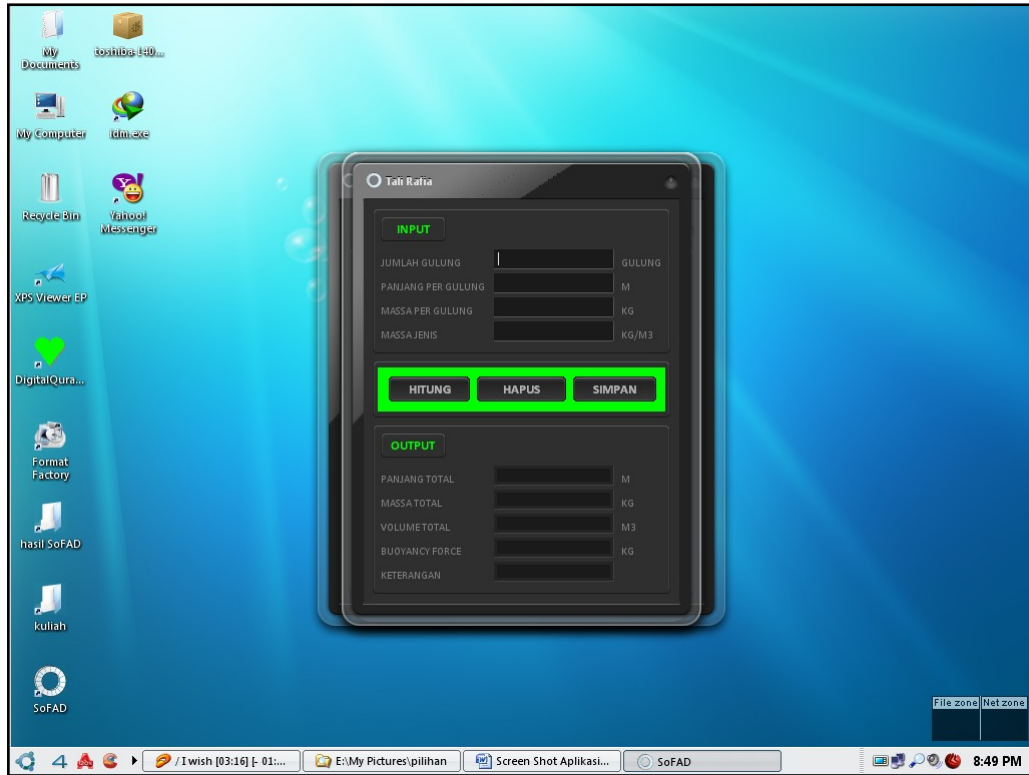
o. Selang



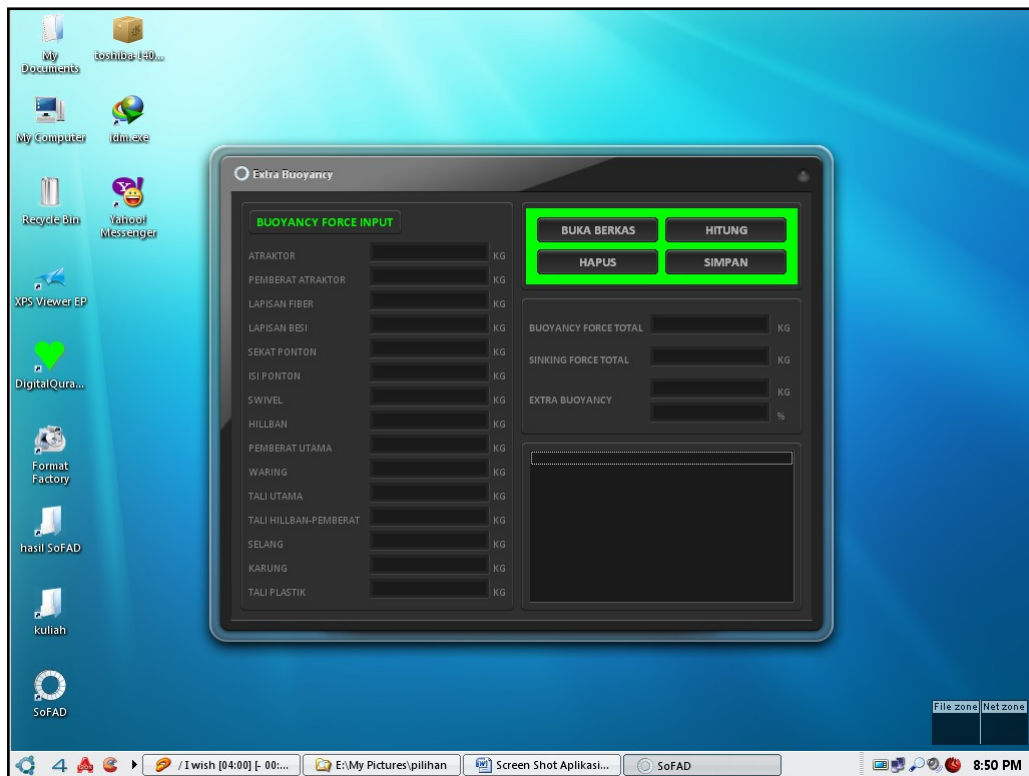
p. Karung



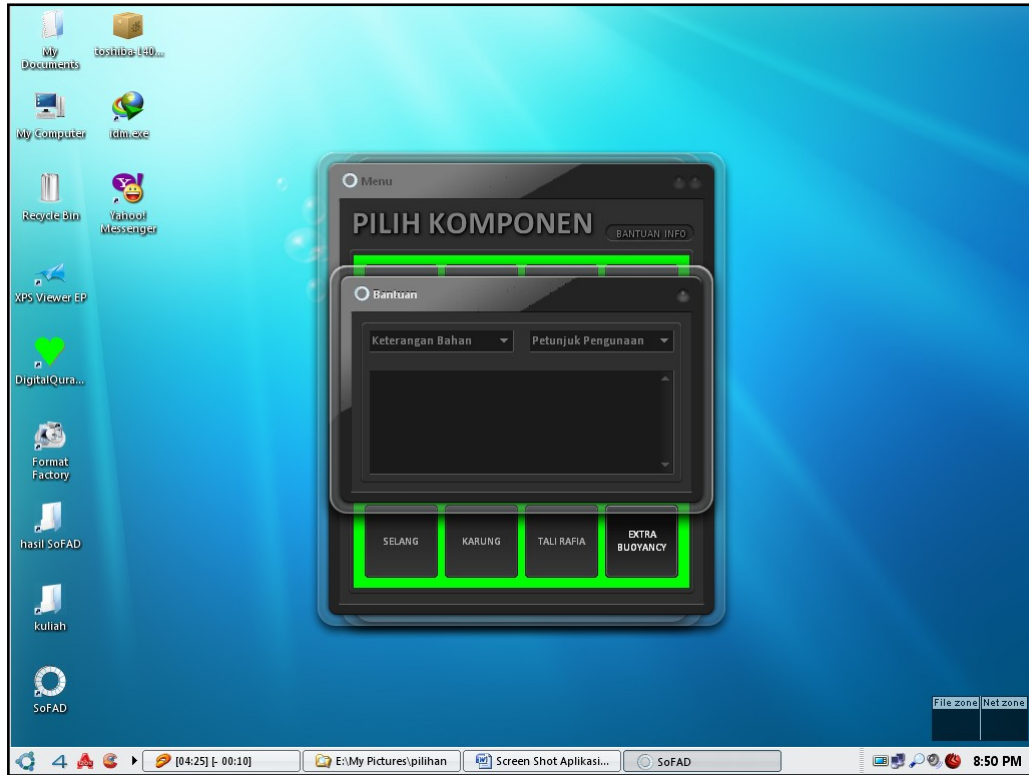
q. Tali rafia



r. Extra buoyancy



s. Bantuan



t. Info



6. Source Code SoFAD

a. Source code unit

atraktor

unit Atraktor;

interface

uses

Windows, Messages,
SysUtils, Variants, Classes,
Graphics, Controls, Forms,
Dialogs, StdCtrls, ExtCtrls,
sSkinManager, sLabel,
sPanel, sAlphaListBox,
sButton, sSkinProvider;

type

Tatraktorv = class(TForm)

Edit1: TEdit;
Edit2: TEdit;
Edit3: TEdit;
Edit4: TEdit;
Edit5: TEdit;
Edit6: TEdit;
Edit7: TEdit;
Edit8: TEdit;
SaveDialog1:

TSaveDialog;

sPanel1: TPanel;
sLabel1: TLabel;
sLabel2: TLabel;
sLabel3: TLabel;
sLabel4: TLabel;
sLabel5: TLabel;
sLabel6: TLabel;
sPanel2: TPanel;
sPanel3: TPanel;
sLabel7: TLabel;
sLabel8: TLabel;
sLabel9: TLabel;
sLabel10: TLabel;
sLabel11: TLabel;
sPanel4: TPanel;
sLabel12: TLabel;
sLabel13: TLabel;
sLabel14: TLabel;

sLabel15: TLabel;
sButton1: TButton;
sButton2: TButton;
sButton3: TButton;
sPanel6: TPanel;
sListBox1: TListBox;
sSkinProvider1:
TsSkinProvider;
procedure
sButton1Click(Sender:
TObject);
procedure
sButton2Click(Sender:
TObject);
procedure
sButton3Click(Sender:
TObject);
private
{ Private declarations }
public
{ Public declarations }
end;
var
atraktorv: Tatraktorv;
implementation
{SR *.dfm}
procedure
Tatraktorv.sButton1Click(Se
nder: TObject);
Var
kode: Integer;
jatr, matr, vatr, mtatr, vtatr, mjat
r, bfatr, sfatr: real;
masstotatr, voltotatr, massjatr,
sinkatr: string;
begin
Val(Edit1.Text, jatr, Kode);
Val(Edit2.Text, matr, Kode);
Val(Edit3.Text, vatr, Kode);
mtatr:=jatr*matr;
str(mtatr:0:2,masstotatr);
Edit4.Text:=masstotatr;

vtatr:=jatr*vatr;
str(vtatr:0:2,voltotatr);
edit5.Text:=voltotatr;
mjatr:=mtatr/vtatr;
str(mjatr:0:2,massjatr);
edit6.Text:=massjatr;
bfatr:=1.025*vtatr;
sfatr:=bfatr-mtatr;
str(sfatr:0:2,sinkatr);
edit7.Text:=sinkatr;
begin
if sfatr>0 then
edit8.Text:='MENGAPUNG'
else
edit8.Text:='TENGCELAM';
end;
sListBox1.Clear;
sListBox1.Items[0]:=JUML
AH ATRAKTOR (BUAH):;
sListBox1.items[1]:=edit1.T
ext;
sListBox1.Items[2]:=MASS
A ATRAKTOR (KG):;
sListBox1.Items[3]:=edit2.T
ext;
sListBox1.Items[4]:=VOLU
ME ATRAKTOR (M3):;
sListBox1.items[5]:=edit3.T
ext;
sListBox1.Items[6]:=MASS
A TOTAL ATRAKTOR
(KG):;
sListBox1.items[7]:=edit4.T
ext;
sListBox1.Items[8]:=VOLU
ME TOTAL ATRAKTOR
(M3):;
sListBox1.items[9]:=edit5.T
ext;

```

sListBox1.Items[10]:='MAS
SA JENIS ATRAKTOR
(KG/M3)';
sListBox1.items[11]:=edit6.
Text;

sListBox1.Items[12]:='BUO
YANCY FORCE
ATRAKTOR (KG)';
sListBox1.items[13]:=edit7.
Text;

sListBox1.Items[14]:='KET
ERANGAN :';
sListBox1.items[15]:=edit8.
Text;
end;

procedure
Tatraktorv.sButton2Click(Se
nder: TObject);
begin
edit1.Clear;
edit2.Clear;
edit3.Clear;
edit4.Clear;
Edit5.clear;
Edit6.Clear;
Edit7.Clear;
Edit8.Clear;
sListBox1.clear;
end;

procedure
Tatraktorv.sButton3Click(Se
nder: TObject);
begin
if
SaveDialog1.Execute
then
sListBox1.items.SaveToFile(
'ATRAKTOR.txt');
end;
end.

interface

uses
Windows, Messages,
SysUtils, Variants, Classes,
Graphics, Controls, Forms,
Dialogs, StdCtrls, ExtCtrls,
sSkinProvider,
sAlphaListBox, sButton,
sLabel, sPanel;

type
TPemberat_Atraktorv =
class(TForm)
Edit1: TEdit;
Edit2: TEdit;
Edit3: TEdit;
Edit4: TEdit;
Edit5: TEdit;
Edit6: TEdit;
Edit7: TEdit;
Edit8: TEdit;
Edit9: TEdit;
Edit10: TEdit;
SaveDialog1:
TSaveDialog;
sSkinProvider1:
TSkinProvider;
sPanel2: TsPanel;
sLabel1: TsLabel;
sLabel2: TsLabel;
sLabel3: TsLabel;
sLabel4: TsLabel;
sLabel5: TsLabel;
sLabel6: TsLabel;
sLabel7: TsLabel;
sLabel8: TsLabel;
sLabel9: TsLabel;
sPanel3: TsPanel;
sLabel10: TsLabel;
sLabel11: TsLabel;
sLabel12: TsLabel;
sLabel13: TsLabel;
sLabel14: TsLabel;
sLabel15: TsLabel;
sLabel16: TsLabel;
sLabel17: TsLabel;
sLabel18: TsLabel;

sLabel19: TsLabel;
sPanel4: TsPanel;
sButton1: TsButton;
sButton2: TsButton;
sButton3: TsButton;
sPanel5: TsPanel;
sPanel6: TsPanel;
sListBox1: TsListBox;
procedure
sButton1Click(Sender:
TObject);
procedure
sButton2Click(Sender:
TObject);
procedure
sButton3Click(Sender:
TObject);
private
{ Private declarations }
public
{ Public declarations }

end;

var
Pemberat_Atraktorv:
TPemberat_Atraktorv;

implementation

{$R *.dfm}

procedure
TPemberat_Atraktorv.sButto
n1Click(Sender: TObject);
Var
kode:Integer;
jpa,ppa,lpa,tpa,mpa,mtpa,vpa
,mjpa,bfpa,sfpa:real;
masstotpa,voltotpa,massjpa,s
inkpa:string;

begin
Val(Edit1.Text,jpa,Kode);
Val(Edit2.Text,ppa,Kode);
Val(Edit3.Text,lpa,Kode);
Val(Edit4.Text,tpa,Kode);
Val(Edit5.Text,mpa,Kode);

```

- b. *Source code* unit pemberat atraktor
unit pemberat_Atraktor;



```

mtpa:=jpa*mpa;
str(mtpa:0:2,masstotpa);
edit6.Text:=masstotpa;

vpa:=jpa*ppa*lpa*tpa;
str(vpa:0:2,voltotpa);
Edit7.Text:=voltotpa;

mjpa:=mtpa/vpa;
str(mjpa:0:2,massjpa);
edit8.Text:=massjpa;

bfpa:=1.025*vpa;

sfpa:=bfpa-mtpa;
str(sfpa:0:2,sinkpa);
edit9.Text:=sinkpa;

begin
if sfpa>0 then
edit10.Text:='MENGAPUN
G'
else
edit10.Text:='TENGGELA
M'
end;

sListBox1.Clear;
sListBox1.Items[0]:='JUML
AH PEMBERAT
ATRAKTOR (BUAH)';
sListBox1.Items[1]:=edit1.T
ext;

sListBox1.Items[2]:='PANJ
ANG PEMBERAT
ATRAKTOR (M)';
sListBox1.Items[3]:=edit2.T
ext;

sListBox1.Items[4]:='LEBA
R PEMBERAT
ATRAKTOR (M)';
sListBox1.Items[5]:=edit3.T
ext;

sListBox1.Items[6]:='TING
GI PEMBERAT
ATRAKTOR (M)';

sListBox1.Items[7]:=edit4.T
ext;

sListBox1.Items[8]:='MASS
A PEMBERAT
ATRAKTOR (M)';
sListBox1.Items[9]:=edit5.T
ext;

sListBox1.Items[10]:='MAS
SA TOTAL PEMBERAT
ATRAKTOR (M)';
sListBox1.Items[11]:=edit6.
Text;

sListBox1.Items[12]:='VOL
UME TOTAL PEMBERAT
ATRAKTOR (M)';
sListBox1.Items[13]:=edit7.
Text;

sListBox1.Items[14]:='MAS
SA JENIS PEMBERAT
ATRAKTOR (KG/M3)';
sListBox1.Items[15]:=edit8.
Text;

sListBox1.Items[16]:='BUO
YANCY FORCE
PEMBERAT ATRAKTOR
(KG)';
sListBox1.Items[17]:=edit9.
Text;

sListBox1.Items[18]:='KET
ERANGAN';
sListBox1.Items[19]:=edit10
.Text;

procedure
TPemberat_Atraktorv.sButto
n2Click(Sender: TObject);
Begin
edit1.Clear;
edit2.Clear;
edit3.Clear;
edit4.Clear;
Edit5.clear;

Edit6.Clear;
Edit7.Clear;
Edit8.Clear;
Edit9.Clear;
Edit10.Clear;

sListBox1.Clear;
end;

procedure
TPemberat_Atraktorv.sButto
n3Click(Sender: TObject);
begin
if
SaveDialog1.Execute
then
sListBox1.Items.SaveToFile(
'PEMBERAT_ATRAKTOR.
txt');
end;
end.

c. Source code unit
fiber

unit fiber;

interface

uses
Windows, Messages,
SysUtils, Variants, Classes,
Graphics, Controls, Forms,
Dialogs, StdCtrls, ExtCtrls,
sSkinProvider,
sAlphaListBox, sLabel,
sButton, sPanel;

type
Tfiberv = class(TForm)
Edit1: TEdit;
Edit2: TEdit;
Edit3: TEdit;
Edit4: TEdit;
Edit5: TEdit;
Edit6: TEdit;
Edit7: TEdit;
Edit8: TEdit;
Edit9: TEdit;

```

```

Edit10: TEdit;           { Private declarations }
Edit11: TEdit;          public      bff:=vff*1.025;
Edit12: TEdit;          { Public declarations }
SaveDialog1:           sff:=bff-mf;
TSaveDialog;           end;      str(sff:0:2,sinkf);
  sSkinProvider1:      edit11.Text:=sinkf;
TsSkinProvider;        var
  sPanel2: TsPanel;     fiberv: Tfiberv;      begin
  sPanel3: TsPanel;     implementation
  sPanel4: TsPanel;     {SR *.dfm}           else
  sButton1: TsButton;   {SR *.dfm}           edit12.Text:='MENGAPUN
  sButton2: TsButton;   {SR *.dfm}           G'
  sButton3: TsButton;   {SR *.dfm}           edit12.Text:='TENGGELE
  sLabel1: TLabel;      procedure           M'
  sLabel2: TLabel;      Tfiberv.sButton1Click(Sende
  sLabel3: TLabel;      r: TObject);
  sLabel4: TLabel;      var
  sLabel5: TLabel;      kode: integer;
  sLabel6: TLabel;      tf,ptf,df,tkf,mjf,vkf,vpf,vtf,vf
  sLabel7: TLabel;      f,mf,bff,sff:real;
  sLabel8: TLabel;      volkf,volpf,voltf,volff,massf,
  sLabel9: TLabel;      sinkf:string;
  sLabel10: TLabel;     begin
  sPanel5: TsPanel;     val(edit1.Text,tf,kode);
  sPanel6: TsPanel;     val(edit2.Text,ptf,kode);
  sLabel11: TLabel;     val(edit3.Text,df,kode);
  sLabel12: TLabel;     val(edit4.Text,tkf,kode);
  sLabel13: TLabel;     val(edit5.Text,mjf,kode);
  sLabel14: TLabel;
  sLabel15: TLabel;
  sLabel16: TLabel;     vkf:=(1/3)*3.14*tf*df*tkf;
  sLabel17: TLabel;     str(vkf:0:2,volkf);
  sLabel18: TLabel;     edit6.Text:=volkf;
  sLabel19: TLabel;
  sLabel20: TLabel;     vpf:=3.14*(df/2)*(df/2)*tf;
  sLabel21: TLabel;     str(vpf:0:2,volpf);
  sLabel22: TLabel;     edit7.Text:=volpf;
  sLabel23: TLabel;
  sListBox1: TListBox;  vtf:=3.14*df*ptf*tf;
  procedure            str(vtf:0:2,voltf);
  sButton1Click(Sender: edit8.Text:=voltf;
  TObject);           vff:=vkf+vpf+vtf;
  procedure            str(vff:0:2,volff);
  sButton2Click(Sender: edit9.Text:=volff;
  TObject);           mf:=mjf*vff;
  procedure            str(mf:0:2,massf);
  sButton3Click(Sender: edit10.Text:=massf;
  TObject);           private
  private

```



```
sListBox1.Items[12]='VOL
UME PENUTUP (M3)';
sListBox1.items[13]:=edit7.
Text;
```

```
sListBox1.Items[14]='VOL
UME TABUNG (M3)';
sListBox1.items[15]:=edit8.
Text;
```

```
sListBox1.Items[16]='VOL
UME FIBER (M3)';
sListBox1.items[17]:=edit9.
Text;
```

```
sListBox1.Items[18]='MAS
SA FIBER (KG)';
sListBox1.Items[19]:=edit10
.Text;
```

```
sListBox1.Items[20]='BUO
YANCY FORCE (KG)';
sListBox1.items[21]:=edit11.
Text;
```

```
sListBox1.Items[22]='KET
ERANGAN .';
sListBox1.items[23]:=edit12.
Text;
end;
```

```
procedure
Tfiberv.sButton2Click(Sende
r: TObject);
begin
edit1.Clear;
edit2.Clear;
edit3.Clear;
edit4.Clear;
Edit5.clear;
Edit6.Clear;
Edit7.Clear;
Edit8.Clear;
edit9.Clear;
Edit10.clear;
Edit11.Clear;
Edit12.Clear;
```

```
sListBox1.clear;
```

```
end;
procedure
Tfiberv.sButton3Click(Sende
r: TObject);
begin
if
SaveDialog1.Execute
then
sListBox1.items.SaveToFile(
'LAPISAN_FIBER.txt');
end;
end.
```

d. *Source code unit besi*

```
unit besi;
```

```
interface
```

```
uses
Windows, Messages,
SysUtils, Variants, Classes,
Graphics, Controls, Forms,
Dialogs, StdCtrls, ExtCtrls,
sSkinProvider, sButton,
sPanel,
sAlphaListBox, sLabel;
```

```
type
```

```
TForm1 = class(TForm)
Edit1: TEdit;
Edit2: TEdit;
Edit3: TEdit;
Edit4: TEdit;
Edit5: TEdit;
Edit6: TEdit;
Edit7: TEdit;
Edit8: TEdit;
Edit9: TEdit;
Edit10: TEdit;
Edit11: TEdit;
Edit12: TEdit;
SaveDialog1:
TSaveDialog;
sPanel1: TsPanel;
sPanel2: TsPanel;
sPanel3: TsPanel;
```

```
sPanel4: TsPanel;
sPanel5: TsPanel;
sButton1: TsButton;
sButton2: TsButton;
sButton3: TsButton;
sLabel1: TsLabel;
sLabel2: TsLabel;
sLabel3: TsLabel;
sLabel4: TsLabel;
sLabel5: TsLabel;
sLabel6: TsLabel;
sLabel7: TsLabel;
sLabel8: TsLabel;
sLabel9: TsLabel;
sLabel10: TsLabel;
sSkinProvider1:
TsSkinProvider;
sLabel11: TsLabel;
sLabel12: TsLabel;
sLabel13: TsLabel;
sLabel14: TsLabel;
sLabel15: TsLabel;
sLabel16: TsLabel;
sLabel17: TsLabel;
sLabel18: TsLabel;
sLabel19: TsLabel;
sLabel20: TsLabel;
sLabel21: TsLabel;
sLabel22: TsLabel;
sLabel23: TsLabel;
sListBox1: TsListBox;
procedure
sButton1Click(Sender:
TObject);
procedure
sButton2Click(Sender:
TObject);
procedure
sButton3Click(Sender:
TObject);
private
{ Private declarations }
public
{ Public declarations }
end;
var
Form1: TForm1;
```

```

implementation
    {$R *.dfm}

procedure
    TForm1.sButton1Click(Sender: TObject);
    var
        kode:integer;
        tb,ptb,tkb,rb,mjb,vtb,vkb,vpb
        ,vb,mb,bfb,sfb:real;
        voltb,volkb,volpb,volb,mass
        b,sinkb:string;

    begin
        val(edit1.Text,tb,kode);
        val(edit2.Text,ptb,kode);
        val(edit3.Text,tkb,kode);
        val(edit4.Text,rb,kode);
        val(edit5.Text,mjb,kode);

        vtb:=3.14*(2*rb)*tb*ptb;
        str(vtb:0:2,voltb);
        edit6.Text:=voltb;

        vkb:=(1/3)*3.14*(2*rb)*tb*tkb;
        str(vkb:0:2,volkb);
        edit7.Text:=volkb;

        vpb:=3.14*rb*rb*tb;
        str(vpb:0:2,volpb);
        edit8.Text:=volpb;

        vb:=vtb+vkb+vpb;
        str(vb:0:2,volb);
        edit9.Text:=volb;

        mb:=mjb*vb;
        str(mb:0:2,massb);
        edit10.Text:=massb;

        bfb:=1.025*vb;
        sfb:=bfb-mb;
        str(sfb:0:2,sinkb);
        edit11.Text:=sinkb;

        begin
            if sfb>0 then
                edit12.Text:='MENGAPUN
                G'
            else
                edit12.Text:='TENGGELA
                M'
            end;

            sListBox1.Clear;
            sListBox1.Visible:=true;
            sListBox1.Items[0]:='KETE
            BALAN BESI (M)';
            sListBox1.items[1]:=edit1.T
            ext;

            sListBox1.Items[2]:='PANJ
            ANG TABUNG (M)';
            sListBox1.Items[3]:=edit2.T
            ext;

            sListBox1.Items[4]:='TING
            GI KERUCUT (M)';
            sListBox1.items[5]:=edit3.T
            ext;

            sListBox1.Items[6]:='JARI-
            JARI (M)';
            sListBox1.items[7]:=edit4.T
            ext;

            sListBox1.Items[8]:='MASS
            A JENIS (KG/M3)';
            sListBox1.items[9]:=edit5.T
            ext;

            sListBox1.Items[10]:='VOL
            UME TABUNG BESI
            (M3)';
            sListBox1.items[11]:=edit6.
            Text;

            sListBox1.Items[12]:='VOL
            UME KERUCUT (M3)';
            sListBox1.items[13]:=edit7.
            Text;

            sListBox1.Items[14]:='VOL
            UME PENUTUP (M3)';

            sListBox1.items[15]:=edit8.
            Text;

            sListBox1.Items[16]:='VOL
            UME LAPISAN BESI
            (M3)';
            sListBox1.items[17]:=edit9.
            Text;

            sListBox1.Items[18]:='MAS
            SA LAPISAN BESI (KG)';
            sListBox1.items[19]:=edit10.
            Text;

            sListBox1.Items[20]:='BUO
            YANCY FORCE (KG)';
            sListBox1.items[21]:=edit11.
            Text;

            sListBox1.Items[22]:='KET
            ERANGAN';
            sListBox1.items[23]:=edit12.
            Text;
            end;

        procedure
            TForm1.sButton2Click(Sender: TObject);
            begin
                edit1.Clear;
                edit2.Clear;
                edit3.Clear;
                edit4.Clear;
                Edit5.clear;
                Edit6.Clear;
                Edit7.Clear;
                Edit8.Clear;
                Edit9.clear;
                Edit10.Clear;
                Edit11.Clear;
                Edit12.Clear;
                sListBox1.Clear;
            end;

        procedure
            TForm1.sButton3Click(Sender: TObject);
            begin
                if

```

```

SaveDialog1.Execute
then
sListBox1.items.SaveToFile(
'LAPISAN_BESI.txt');
end;
end.

```

e. *Source code* unit sekat

```

unit sekat;

interface

uses
  Windows, Messages,
  SysUtils, Variants, Classes,
  Graphics, Controls, Forms,
  Dialogs, StdCtrls, ExtCtrls,
  ComCtrls, sSkinProvider,
  sAlphaListBox,
  sButton, sPanel, sLabel;

type
  TSekatv = class(TForm)
    Edit1: TEdit;
    Edit2: TEdit;
    Edit3: TEdit;
    Edit4: TEdit;
    Edit5: TEdit;
    Edit6: TEdit;
    Edit7: TEdit;
    Edit8: TEdit;
    SaveDialog1:
    TSaveDialog;
    sSkinProvider1:
    TsSkinProvider;
    sListBox1: TsListBox;
    sPanel1: TsPanel;
    sButton1: TsButton;
    sButton2: TsButton;
    sButton3: TsButton;
    sPanel2: TsPanel;
    sPanel3: TsPanel;
    sPanel4: TsPanel;
    sLabel1: TsLabel;
    sLabel2: TsLabel;
    sLabel3: TsLabel;
    sLabel4: TsLabel;

```

```

    sLabel5: TsLabel;
    sLabel6: TsLabel;
    sLabel7: TsLabel;
    sLabel8: TsLabel;
    sPanel5: TsPanel;
    sLabel9: TsLabel;
    sLabel10: TsLabel;
    sLabel11: TsLabel;
    sLabel12: TsLabel;
    sLabel13: TsLabel;
    sLabel14: TsLabel;
    sLabel15: TsLabel;

    procedure
    sButton1Click(Sender:
    TObject);
    procedure
    sButton2Click(Sender:
    TObject);
    procedure
    sButton3Click(Sender:
    TObject);
  private
    { Private declarations }
  public
    { Public declarations }

  end;

var
  Sekatv: TSekatv;

implementation
  {$R *.dfm}

  procedure
  TSekatv.sButton1Click(Send
  er: TObject);
  var
    kode: integer;
    ts,js,rs,mjs,vs,ms,bfs,sfs:real;
    vols,masss,sinks:string;

  begin
    val(edit1.Text,ts,kode);
    val(edit2.Text,js,kode);
    val(edit3.Text,rs,kode);
    val(edit4.Text,mjs,kode);
    vs:=ts*js*rs*rs*3.14;

```

```

    str(vs:0:2,vols);
    edit5.Text:=vols;

    ms:=mjs*vs;
    str(ms:0:2,masss);
    edit6.Text:=masss;

    bfs:=1.025*vs;

    sfs:=bfs-ms;
    str(sfs:0:2,sinks);
    edit7.Text:=sinks;

  begin
  if sfs>0 then
    edit8.Text:='MENGAPUNG'
  else
    edit8.Text:='TENGCELAM'
  end;

  sListBox1.Clear;
  sListBox1.Items[0]:='KETE
  BALAN SEKAT (M):';
  sListBox1.items[1]:=edit1.T
  ext;

  sListBox1.Items[2]:='JUML
  AH SEKAT (BUAH):';
  sListBox1.Items[3]:=edit2.T
  ext;

  sListBox1.Items[4]:='JARI-
  JARI (M):';
  sListBox1.items[5]:=edit3.T
  ext;

  sListBox1.Items[6]:='MASS
  A JENIS (KG/M3):';
  sListBox1.items[7]:=edit4.T
  ext;

  sListBox1.Items[8]:='VOLU
  ME SEKAT (M3):';
  sListBox1.items[9]:=edit5.T
  ext;

  sListBox1.Items[10]:='MAS
  SA SEKAT (KG):';

```

```

sListBox1.items[11]:=edit6.
Text;

sListBox1.Items[12]:='BUO
YANCY FORCE SEKAT
(KG)';
sListBox1.items[13]:=edit7.
Text;

sListBox1.Items[14]:='KET
ERANGAN';
sListBox1.items[15]:=edit8.
Text;
end;

procedure
TSekativ.sButton2Click(Send
er: TObject);
begin
edit1.Clear;
edit2.Clear;
edit3.Clear;
edit4.Clear;
Edit5.clear;
Edit6.Clear;
Edit7.Clear;
Edit8.Clear;
sListBox1.clear;
end;

procedure
TSekativ.sButton3Click(Send
er: TObject);
begin
if
SaveDialog1.Execute
then
sListBox1.items.SaveToFile(
'SEKAT_PONTON.txt');
end;
end.

interface

uses
Windows, Messages,
SysUtils, Variants, Classes,
Graphics, Controls, Forms,
Dialogs, StdCtrls, ExtCtrls,
sSkinProvider, sButton,
sPanel,
sAlphaListBox, sLabel;

type
TIsi = class(TForm)
Edit1: TEdit;
Edit2: TEdit;
Edit3: TEdit;
Edit4: TEdit;
Edit5: TEdit;
Edit6: TEdit;
Edit7: TEdit;
Edit8: TEdit;
Edit9: TEdit;
Edit10: TEdit;
SaveDialog1:
TSaveDialog;
sSkinProvider1:
sSkinProvider;
sListBox1: TsListBox;
sPanel1: TsPanel;
sPanel2: TsPanel;
sPanel3: TsPanel;
sButton1: TsButton;
sButton2: TsButton;
sButton3: TsButton;
sPanel4: TsPanel;
sLabel1: TsLabel;
sLabel2: TsLabel;
sLabel3: TsLabel;
sLabel4: TsLabel;
sLabel5: TsLabel;
sLabel6: TsLabel;
sLabel7: TsLabel;
sLabel8: TsLabel;
sPanel5: TsPanel;
sLabel9: TsLabel;
sLabel10: TsLabel;
sLabel11: TsLabel;
sLabel12: TsLabel;
sLabel13: TsLabel;
sLabel14: TsLabel;
sLabel15: TsLabel;
sLabel16: TsLabel;
sLabel17: TsLabel;
sLabel18: TsLabel;
sLabel19: TsLabel;
procedure
sButton1Click(Sender:
TObject);
procedure
sButton2Click(Sender:
TObject);
procedure
sButton3Click(Sender:
TObject);
private
{ Private declarations }
public
{ Public declarations }
end;

var
Isi: TIsi;

implementation

{$R *.dfm}

procedure
TIsi.sButton3Click(Sender:
TObject);
begin
if
SaveDialog1.Execute
then
sListBox1.items.SaveToFile(
'ISI_PONTON.txt');
end;

procedure
TIsi.sButton1Click(Sender:
TObject);
var
kode:integer;
pi,ri,ti,mi,vti,m,vk,vi,bf,sf:re
al;
volt,massi,volk,voli,sinki:stri
ng;
begin

```

f. *Source code* unit

isi ponton

```
unit isiponton;
```

```
interface
```

```

val(edit1.Text,pi,kode);
val(edit2.Text,ri,kode);
val(edit3.Text,ti,kode);
val(edit4.Text,mi,kode);

vti:=3.14*ri*ri*pi;
str(vti:0:2,volt);
edit5.Text:=volt;

vk:=(1/3)*3.14*ri*ri*ti;
str(vk:0:2,volk);
edit7.Text:=volk;

vi:=vti+vk;
str(vi:0:2,voli);
edit8.Text:=voli;

m:=mi*vi;
str(m:0:2,massi);
edit6.Text:=massi;

bf:=vi*1.025;

sf:=bf-m;
str(sf:0:2,sinki);
edit9.Text:=sinki;

begin
if sf>0 then
edit10.Text:='MENGAPUN
G'
else
edit10.Text:='TENGGE LA
M'
end;

sListBox1.Clear;
sListBox1.Items[0]:='PANJ
ANG TABUNG (M)';
sListBox1.items[1]:=edit1.T
ext;

sListBox1.Items[2]:='JARI-
JARI (M)';
sListBox1.Items[3]:=edit2.T
ext;

sListBox1.Items[4]:='TING
GI KERUCUT (M)';

```

```

sListBox1.items[5]:=edit3.T
ext;

sListBox1.Items[6]:='MASS
A JENIS (KG/M3)';
sListBox1.items[7]:=edit4.T
ext;

sListBox1.Items[8]:='VOLU
ME ISI TABUNG (M3)';
sListBox1.items[9]:=edit5.T
ext;

sListBox1.Items[10]:='MAS
SA ISI TABUNG (KG)';
sListBox1.items[11]:=edit6.
Text;

sListBox1.Items[12]:='VOL
UME ISI KERUCUT (M3)';
sListBox1.items[13]:=edit7.
Text;

sListBox1.Items[14]:='VOL
UME ISI PONTON (M3)';
sListBox1.items[15]:=edit8.
Text;

sListBox1.Items[16]:='BUO
YANCY FORCE ISI
PONTON (KG)';
sListBox1.items[17]:=edit9.
Text;

sListBox1.Items[18]:='KET
ERANGAN';
sListBox1.items[19]:=edit10.
Text;
end;

procedure
TIsi.sButton2Click(Sender:
TObject);
begin
edit1.Clear;
edit2.Clear;
edit3.Clear;
edit4.Clear;
edit5.clear;

```

```

Edit6.Clear;
Edit7.Clear;
Edit8.Clear;
Edit9.Clear;
Edit10.Clear;
sListBox1.clear;
end;
end.
```

g. *Source code unit swivel*

```

unit Swivel;

interface

uses
  Windows, Messages,
  SysUtils, Variants, Classes,
  Graphics, Controls, Forms,
  Dialogs, StdCtrls, ExtCtrls,
  sSkinProvider,
  sAlphaListBox, sButton,
  sPanel, sLabel;

type
  TSwivelv = class(TForm)
    Edit1: TEdit;
    Edit2: TEdit;
    Edit3: TEdit;
    Edit4: TEdit;
    Edit5: TEdit;
    Edit6: TEdit;
    Edit7: TEdit;
    Edit8: TEdit;
    SaveDialog1:
    TSaveDialog;
    sSkinProvider1:
    TSkinProvider;
    sListBox1: TListBox;
    sPanel1: TPanel;
    sButton1: TButton;
    sButton2: TButton;
    sButton3: TButton;
    sPanel2: TPanel;
    sPanel3: TPanel;
    sPanel4: TPanel;
    sLabel1: TLabel;
    sLabel2: TLabel;

```



```

sLabel3: TLabel;
sLabel4: TLabel;
sLabel5: TLabel;
sLabel6: TLabel;
sPanel5: TPanel;
sLabel7: TLabel;
sLabel8: TLabel;
sLabel9: TLabel;
sLabel10: TLabel;
sLabel11: TLabel;
sLabel12: TLabel;
sLabel13: TLabel;
sLabel14: TLabel;
sLabel15: TLabel;
procedure
sButton1Click(Sender:
TObject);
    procedure
sButton2Click(Sender:
TObject);
    procedure
sButton3Click(Sender:
TObject);
    private
    { Private declarations }
    public
    { Public declarations }
end;
var
    Swivelv: TSwivelv;
implementation
{SR *.dfm}
procedure
TSwivelv.sButton1Click(Sen
der: TObject);
Var
kode: Integer;
jswi,mpswi,mjswi,mtswi,vps
wi,vtswi,bfswi,sfswi:real;
matotswi,volpswi,voltotswi,s
inkswi:string;

begin
Val(Edit1.Text,jswi,Kode);

Val(Edit2.Text,mpswi,Kode)
;
Val(Edit3.Text,mjswi,Kode);

mtswi:=jswi*mpswi;
str(mtswi:0:2,matotswi);
Edit4.Text:=matotswi;

vpswi:=mpswi/mjswi;
str(vpswi:0:2,volpswi);
edit5.Text:=volpswi;

vtswi:=jswi*vpswi;
str(vtswi:0:2,voltotswi);
edit6.Text:=voltotswi;

bfswi:=1.025*vtswi;

sfswi:=bfswi-mtswi;
str(sfswi:0:2,sinkswi);
edit7.Text:=sinkswi;

begin
if sfswi>0 then
edit8.Text:='MENGAPUNG'
else
edit8.Text:='TENGSELAM'
end;

sListBox1.Clear;
sListBox1.Items[0]:='JUML
AH SWIVEL (buah)';
sListBox1.Items[1]:=edit1.T
ext;

sListBox1.Items[2]:='MASS
A PER SWIVEL (KG)';
sListBox1.Items[3]:=edit2.T
ext;

sListBox1.Items[4]:='MASS
A JENIS SWIVEL
(KG/M3)';
sListBox1.Items[5]:=edit3.T
ext;

sListBox1.Items[6]:='MASS
A TOTAL SWIVEL (KG)';

sListBox1.Items[7]:=edit4.T
ext;

sListBox1.Items[8]:='VOLU
ME PER SWIVEL (M3)';
sListBox1.Items[9]:=edit5.T
ext;

sListBox1.Items[10]:='VOL
UME TOTAL SWIVEL
(M3)';
sListBox1.Items[11]:=edit6.
Text;

sListBox1.Items[12]:='BUO
YANCY FORCE SWIVEL
(KG)';
sListBox1.Items[13]:=edit7.
Text;

sListBox1.Items[14]:='KET
ERANGAN';
sListBox1.Items[15]:=edit8.
Text;
end;

procedure
TSwivelv.sButton2Click(Sen
der: TObject);
begin
edit1.Clear;
edit2.Clear;
edit3.Clear;
edit4.Clear;
Edit5.clear;
Edit6.Clear;
Edit7.Clear;
Edit8.Clear;
sListBox1.clear;
end;

procedure
TSwivelv.sButton3Click(Sen
der: TObject);
begin
if
SaveDialog1.Execute
then

```

```

sListBox1.items.SaveToFile(
'SWIVEL.txt');
end;

end.

h. Source code unit
hillban
unit hillban;

interface

uses
  Windows, Messages,
  SysUtils, Variants, Classes,
  Graphics, Controls, Forms,
  Dialogs, StdCtrls, ExtCtrls,
  sSkinProvider, sButton,
  sLabel, sPanel,
  sAlphaListBox, sEdit;

type
  THillbanv = class(TForm)
    Edit1: TEdit;
    Edit2: TEdit;
    Edit3: TEdit;
    Edit4: TEdit;
    Edit5: TEdit;
    Edit6: TEdit;
    Edit7: TEdit;
    Edit8: TEdit;
    Edit9: TEdit;
    Edit10: TEdit;
    Edit11: TEdit;
    Edit12: TEdit;
    Edit13: TEdit;
    Edit14: TEdit;
    Edit15: TEdit;
    SaveDialog1:
    TSaveDialog;
    sSkinProvider1:
    TsSkinProvider;
    sPanel1: TsPanel;
    sPanel2: TsPanel;
    sPanel3: TsPanel;
    sLabel1: TsLabel;
    sLabel2: TsLabel;
    sLabel3: TsLabel;
    sLabel4: TsLabel;
    sLabel5: TsLabel;
    sLabel6: TsLabel;
    sLabel7: TsLabel;
    sLabel8: TsLabel;
    sLabel9: TsLabel;
    sLabel10: TsLabel;
    sLabel11: TsLabel;
    sLabel12: TsLabel;
    sLabel13: TsLabel;
    sLabel14: TsLabel;
    sPanel4: TsPanel;
    sButton1: TsButton;
    sButton2: TsButton;
    sButton3: TsButton;
    sPanel5: TsPanel;
    sLabel15: TsLabel;
    sLabel16: TsLabel;
    sLabel17: TsLabel;
    sLabel18: TsLabel;
    sLabel19: TsLabel;
    sLabel20: TsLabel;
    sLabel21: TsLabel;
    sLabel22: TsLabel;
    sLabel23: TsLabel;
    sLabel24: TsLabel;
    sLabel25: TsLabel;
    sLabel26: TsLabel;
    sLabel27: TsLabel;
    sLabel28: TsLabel;
    sLabel29: TsLabel;
    sListBox1: TsListBox;
    procedure
    sButton1Click(Sender:
    TObject);
    procedure
    sButton2Click(Sender:
    TObject);
    procedure
    sButton3Click(Sender:
    TObject);
    private
    { Private declarations }
    public
    { Public declarations }
  end;
var
  Hillbanv: THillbanv;

implementation

{$R *.dfm}

procedure
THillbanv.sButton1Click(Se
nder: TObject);
  Var
  kode:Integer;
  jh,rlh,rdh,lth,tth,tsh,mh,vth,v
sh,vph,vtoth,mth,mjh,bfh,sfh
:real;
  volth,volsh,volph,voltoth,ma
toth,massjh,sinkh:string;

begin
  Val(Edit1.Text,jh,Kode);
  Val(Edit2.Text,rlh,Kode);
  Val(Edit3.Text,rdh,Kode);
  Val(Edit4.Text,lth,Kode);
  Val(Edit5.Text,tth,Kode);
  Val(Edit6.Text,tsh,Kode);
  Val(Edit7.Text,mh,Kode);

  vth:=3.14*2*rlh*lth*tth;
  str(vth:0:2,volth);
  edit8.Text:=volth;

  vsh:=2*((3.14*rlh*rlh*tsh)-
(3.14*rdh*rdh*tsh));
  str(vsh:0:2,volsh);
  edit9.Text:=volsh;

  vph:=vth+vsh;
  str(vph:0:2,volph);
  edit10.Text:=volph;

  vtoth:=jh*vph;
  str(vtoth:0:2,voltoth);
  edit11.Text:=voltoth;

  mth:=jh*mh;
  str(mth:0:2,matoth);
  edit12.Text:=matoth;

  mjh:=mth/vtoth;
  str(mjh:0:2,massjh);

```

```

edit13.Text:=massjh;

bfh:=1.025*vtoth;

sfh:=bfh-mth;
str(sfh:0:2,sinkh);
edit14.Text:=sinkh;

begin
if sfh>0 then
edit15.Text:='MENGAPUN
G'
else
edit15.Text:='TENGGELA
M';
end;

sListBox1.Clear;
sListBox1.Items[0]:='JUML
AH HILLBAN (BUAH)';
sListBox1.items[1]:=edit1.T
ext;

sListBox1.Items[2]:='JARI-
JARI LUAR HILLBAN
(M)';
sListBox1.Items[3]:=edit2.T
ext;

sListBox1.Items[4]:='JARI-
JARI DALAM HILLBAN
(M)';
sListBox1.items[5]:=edit3.T
ext;

sListBox1.Items[6]:='LEBA
R TAPAK HILLBAN (M)';
sListBox1.items[7]:=edit4.T
ext;

sListBox1.Items[8]:='TEBA
L TAPAK HILLBAN (M)';
sListBox1.items[9]:=edit5.T
ext;

sListBox1.Items[10]:='TEB
AL SAMPING HILLBAN
(M)';

sListBox1.items[11]:=edit6.
Text;

sListBox1.Items[12]:='MAS
SA PER HILLBAN (KG)';
sListBox1.items[13]:=edit7.
Text;

sListBox1.Items[14]:='VOL
UME TAPAK HILLBAN
(M3)';
sListBox1.items[15]:=edit8.
Text;

sListBox1.Items[16]:='VOL
UME SAMPING HILLBAN
(M3)';
sListBox1.items[17]:=edit9.
Text;

sListBox1.Items[18]:='VOL
UME PER HILLBAN
(M3)';
sListBox1.items[19]:=edit10.
Text;

sListBox1.Items[20]:='VOL
UME TOTAL HULLBAN
(M3)';
sListBox1.items[21]:=edit11.
Text;

sListBox1.Items[22]:='MAS
SA TOTAL HILLBAN
(KG)';
sListBox1.items[23]:=edit12.
Text;

sListBox1.Items[24]:='MAS
SA JENIS HILLBAN
(KG/M3)';
sListBox1.items[25]:=edit13.
Text;

sListBox1.Items[22]:='BUO
YANCY FORCE HILLBAN
(KG)';
sListBox1.items[23]:=edit14.
Text;

sListBox1.Items[24]:='KET
ERANGAN';
sListBox1.items[25]:=edit15.
Text;

end;

procedure
THillbanv.sButton2Click(Se
nder: TObject);
Begin
edit1.Clear;
edit2.Clear;
edit3.Clear;
edit4.Clear;
Edit5.clear;
Edit6.Clear;
Edit7.Clear;
Edit8.Clear;
edit9.Clear;
edit10.Clear;
edit11.Clear;
edit12.Clear;
Edit13.clear;
Edit14.Clear;
Edit15.Clear;
sListBox1.Clear;
end;

procedure
THillbanv.sButton3Click(Se
nder: TObject);
begin
if
SaveDialog1.Execute
then
sListBox1.items.SaveToFile(
'HILLBAN.txt');
end;
end.

i. Source code unit
pemberat utama
unit Pemberat_Utama;

interface

```

```

uses
    Windows, Messages,
    SysUtils, Variants, Classes,
    Graphics, Controls, Forms,
    Dialogs, StdCtrls, ExtCtrls,
    sSkinProvider,
    sAlphaListBox, sButton,
    sPanel, sLabel;

type
    TPemberatUtamax =
class(TForm)
    Edit1: TEdit;
    Edit2: TEdit;
    Edit3: TEdit;
    Edit4: TEdit;
    Edit5: TEdit;
    Edit6: TEdit;
    Edit7: TEdit;
    Edit8: TEdit;
    Edit9: TEdit;
    Edit10: TEdit;
    Edit11: TEdit;
    SaveDialog1:
TSaveDialog;
    sSkinProvider1:
TsSkinProvider;
    sListBox1: TsListBox;
    sButton1: TsButton;
    sButton2: TsButton;
    sButton3: TsButton;
    sPanel1: TsPanel;
    sPanel2: TsPanel;
    sPanel3: TsPanel;
    sLabel1: TsLabel;
    sLabel2: TsLabel;
    sLabel3: TsLabel;
    sLabel4: TsLabel;
    sLabel5: TsLabel;
    sLabel6: TsLabel;
    sLabel7: TsLabel;
    sLabel8: TsLabel;
    sLabel9: TsLabel;
    sLabel10: TsLabel;
    sPanel4: TsPanel;
    sLabel11: TsLabel;
    sLabel12: TsLabel;
    sLabel13: TsLabel;
    sLabel14: TsLabel;
    sLabel15: TsLabel;
    sLabel16: TsLabel;
    sLabel17: TsLabel;
    sLabel18: TsLabel;
    sLabel19: TsLabel;
    sLabel20: TsLabel;
    sLabel21: TsLabel;
    procedure
sButton1Click(Sender:
TObject);
    procedure
sButton2Click(Sender:
TObject);
    procedure
sButton3Click(Sender:
TObject);

    private
        { Private declarations }
    public
        { Public declarations }
    end;

var
    PemberatUtamax:
TPemberatUtamax;

implementation
    {$R *.dfm}

    procedure
TPemberatUtamax.sButton1
Click(Sender: TObject);
    Var
    kode:Integer;
    jpu,ppu,lpu,tpu,bpbpu,btpu,v
pbpu,vtpu,mjpu,bfpu,sfpu:re
al;
    betotpu,volpbpu,voltotpu,ma
ssjpu,sinkpu:string;

    begin
    Val(Edit1.Text,jpu,Kode);
    Val(Edit2.Text,ppu,Kode);
    Val(Edit3.Text,lpu,Kode);
    val(edit4.Text,tpu,kode);
    val(edit5.Text,bpbpu,kode);
    btpu:=jpu*bpbpu;
    str(btpu:0:2,betotpu);
    edit6.Text:=betotpu;

    vpbpu:=ppu*lpu*tpu;
    str(vpbpu:0:2,volpbpu);
    edit7.Text:=volpbpu;

    vtpu:=jpu*vpbpu;
    str(vtpu:0:2,voltotpu);
    edit8.Text:=voltotpu;

    mjpu:=btpu/vtpu;
    str(mjpu:0:2,massjpu);
    edit9.Text:=massjpu;

    bfpu:=1.025*vtpu;

    sfpu:=bfpu-btpu;
    str(sfpu:0:2,sinkpu);
    edit10.Text:=sinkpu;

    begin
    if sfpu>0 then
    edit11.Text:='MENGAPUN
G'
    else
    edit11.Text:='TENGGELA
M';
    end;

    sListBox1.Clear;
    sListBox1.Items[0]:='JUML
AH PMBERAT (BUAH):';
    sListBox1.items[1]:=edit1.T
ext;

    sListBox1.Items[2]:='PANJ
ANG PEMBERAT (M):';
    sListBox1.Items[3]:=edit2.T
ext;

    sListBox1.Items[4]:='LEBA
R PEMBERAT (M):';
    sListBox1.items[5]:=edit3.T
ext;

```

```

sListBox1.Items[6]:= 'TING
GI PEMBERAT (M)';
sListBox1.items[7]:=edit4.T
ext;

sListBox1.Items[8]:= 'MASS
A PER BUAH PEMBERAT
(KG)';
sListBox1.items[9]:=edit5.T
ext;

sListBox1.Items[10]:= 'MAS
SA TOTAL PEMBERAT
(KG)';
sListBox1.items[11]:=edit6.
Text;

sListBox1.Items[12]:= 'VOL
UME PER BUAH
PEMBERAT (M3)';
sListBox1.items[13]:=edit7.
Text;

sListBox1.Items[14]:= 'VOL
UME TOTAL PEMBERAT
(M3)';
sListBox1.items[15]:=edit8.
Text;

sListBox1.Items[16]:= 'MAS
SA JENIS PEMBERAT
(KG/M3)';
sListBox1.Items[17]:=edit9.
Text;

sListBox1.Items[18]:= 'BUO
YANCY FORCE
PEMBERAT (KG)';
sListBox1.Items[19]:=edit10
.Text;

sListBox1.Items[20]:= 'KET
ERANGAN';
sListBox1.Items[21]:=edit11
.Text;
end;

procedure
TPemberatUtamax.sButton2
Click(Sender: TObject);
begin
edit1.Clear;
edit2.Clear;
edit3.Clear;
edit4.Clear;
Edit5.clear;
Edit6.Clear;
Edit7.Clear;
Edit8.Clear;
edit9.Clear;
edit10.Clear;
sListBox1.clear;
end;

procedure
TPemberatUtamax.sButton3
Click(Sender: TObject);
begin
if
SaveDialog1.Execute
then
sListBox1.items.SaveToFile(
'PEMBERAT_UTAMA.txt');
end;
end.

```

j. *Source code unit*

```

waring
unit waring;

interface

uses
  Windows, Messages,
  SysUtils, Variants, Classes,
  Graphics, Controls, Forms,
  Dialogs, StdCtrls, ExtCtrls,
  sSkinProvider, sButton,
  sPanel,
  sAlphaListBox, sLabel;

type
  Twaringv = class(TForm)
    Edit1: TEdit;
    Edit2: TEdit;
    Edit3: TEdit;
    Edit4: TEdit;
    Edit5: TEdit;
    Edit6: TEdit;
    Edit7: TEdit;
    SaveDialog1:
      TSaveDialog;
    sSkinProvider1:
      TsSkinProvider;
    sListBox1: TsListBox;
    sPanel1: TsPanel;
    sButton1: TsButton;
    sButton2: TsButton;
    sButton3: TsButton;
    sPanel2: TsPanel;
    sLabel1: TsLabel;
    sLabel2: TsLabel;
    sLabel3: TsLabel;
    sLabel4: TsLabel;
    sPanel3: TsPanel;
    sLabel5: TsLabel;
    sLabel6: TsLabel;
    sLabel7: TsLabel;
    sLabel8: TsLabel;
    sPanel4: TsPanel;
    sLabel9: TsLabel;
    sLabel10: TsLabel;
    sLabel11: TsLabel;
    sPanel5: TsPanel;
    sLabel12: TsLabel;
    sLabel13: TsLabel;
    procedure
      sButton1Click(Sender:
        TObject);
    procedure
      sButton2Click(Sender:
        TObject);
    procedure
      sButton3Click(Sender:
        TObject);
    private
      { Private declarations }
    public
      { Public declarations }
  end;
var
  waringv: Twaringv;

```


implementation

```
{SR *.dfm}
```

procedure

```
Twaringv.sButton1Click(Sender: TObject);
```

```
Var
```

```
kode:Integer;
```

```
pwar,lwar,mwar,mjwar,vwar
```

```
,bfwar,sfwar:real;
```

```
volwar,sinkwar:string;
```

```
begin
```

```
Val(Edit1.Text,pwar,Kode);
```

```
Val(Edit2.Text,lwar,Kode);
```

```
Val(Edit3.Text,mwar,Kode);
```

```
Val(Edit4.Text,mjwar,Kode)
```

```
;
```

```
vwar:=mwar/mjwar;
```

```
str(vwar:0:2,volwar);
```

```
Edit5.Text:=volwar;
```

```
bfwar:=1.025*vwar;
```

```
sfwar:=bfwar-mwar;
```

```
str(sfwar:0:2,sinkwar);
```

```
edit6.Text:=sinkwar;
```

```
begin
```

```
if sfwar>0 then
```

```
edit7.Text:='MENGAPUNG'
```

```
else
```

```
edit7.Text:='TENGSELAM'
```

```
end;
```

```
sListBox1.Clear;
```

```
sListBox1.Items[0]:='PANJANG WARING (M)';
```

```
sListBox1.items[1]:=edit1.T
```

```
ext;
```

```
sListBox1.Items[2]:='LEBAR
```

```
WARING (M)';
```

```
sListBox1.Items[3]:=edit2.T
```

```
ext;
```

```
sListBox1.Items[4]:='MASS
```

```
A WARING (KG)';
```

```
sListBox1.items[5]:=edit3.T
```

```
ext;
```

```
sListBox1.Items[6]:='MASS
```

```
A JENIS WARING
```

```
(KG/M3)';
```

```
sListBox1.items[7]:=edit4.T
```

```
ext;
```

```
sListBox1.Items[8]:='VOLUME
```

```
WARING (M3)';
```

```
sListBox1.items[9]:=edit5.T
```

```
ext;
```

```
sListBox1.Items[10]:='BUOYANCY
```

```
FORCE WARING
```

```
(KG)';
```

```
sListBox1.items[11]:=edit6.
```

```
Text;
```

```
sListBox1.Items[12]:='KETERANGAN';
```

```
sListBox1.items[13]:=edit7.
```

```
Text;
```

```
end;
```

```
procedure
```

```
Twaringv.sButton2Click(Sender: TObject);
```

```
begin
```

```
edit1.Clear;
```

```
edit2.Clear;
```

```
edit3.Clear;
```

```
edit4.Clear;
```

```
Edit5.clear;
```

```
Edit6.Clear;
```

```
Edit7.Clear;
```

```
sListBox1.clear;
```

```
end;
```

```
procedure
```

```
Twaringv.sButton3Click(Sender: TObject);
```

```
begin
```

```
if
```

```
SaveDialog1.Execute
```

```
then
```

```
sListBox1.items.SaveToFile('
```

```
WARING.txt');
```

```
end;
```

```
end.
```

k. *Source code unit selang*

unit Selang;

interface

uses

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms, Dialogs, StdCtrls, ExtCtrls, sSkinProvider, sAlphaListBox, sButton, sPanel, sLabel;

type

```
TSelangv = class(TForm)
```

```
Edit1: TEdit;
```

```
Edit2: TEdit;
```

```
Edit3: TEdit;
```

```
Edit4: TEdit;
```

```
Edit5: TEdit;
```

```
Edit6: TEdit;
```

```
Edit7: TEdit;
```

```
Edit8: TEdit;
```

```
SaveDialog1:
```

```
TSaveDialog;
```

```
sSkinProvider1:
```

```
TsSkinProvider;
```

```
sListBox1: TsListBox;
```

```
sPanel1: TsPanel;
```

```
sButton1: TsButton;
```

```
sButton2: TsButton;
```

```
sButton3: TsButton;
```

```
sPanel2: TsPanel;
```

```
sPanel3: TsPanel;
```

```
sPanel4: TsPanel;
```

```
sLabel1: TsLabel;
```

```
sLabel2: TsLabel;
```

```
sLabel3: TsLabel;
```

```

sLabel4: TLabel;
sLabel5: TLabel;
sLabel6: TLabel;
sLabel7: TLabel;
sLabel8: TLabel;
sPanel5: TPanel;
sLabel9: TLabel;
sLabel10: TLabel;
sLabel11: TLabel;
sLabel12: TLabel;
sLabel13: TLabel;
sLabel14: TLabel;
sLabel15: TLabel;
procedure
sButton1Click(Sender:
TObject);
    procedure
sButton2Click(Sender:
TObject);
    procedure
sButton3Click(Sender:
TObject);
private
    { Private declarations }
public
    { Public declarations }
end;
var
    Selangv: TSelangv;

implementation

{$R *.dfm}
procedure
TSelangv.sButton1Click(Se-
nder: TObject);
var
    kode: Integer;
    psel,dsel,tset,msjssel,vsel,mss-
    el,bfssel,sfssel:real;
    volsel,massssel,sinksel:string;

begin
    Val(Edit1.Text,psel,Kode);
    Val(Edit2.Text,dsel,Kode);
    Val(Edit3.Text,tset,Kode);
    Val(Edit4.Text,msjssel,Kode)
    ;
    vsel:=3.14*psel*dsel*tset;
    str(vsel:0:2,volsel);
    edit5.Text:=volsel;

    mssel:=msjssel*vsel;
    str(mssel:0:2,massssel);
    edit6.Text:=massssel;

    bfssel:=1.025*vsel;

    sfssel:=bfssel-mssel;
    str(sfssel:0:2,sinksel);
    edit7.Text:=sinksel;

begin
if sfssel>0 then
    edit8.Text:='MENGAPUNG'
else
    edit8.Text:='TENGCELAM'
end;

sListBox1.Clear;
sListBox1.Items[0]:='PANJ-
ANG SELANG (M)';
sListBox1.items[1]:=edit1.T-
ext;

sListBox1.Items[2]:='DIAM-
ETER SELANG (M)';
sListBox1.Items[3]:=edit2.T-
ext;

sListBox1.Items[4]:='KETE-
BALAN SELANG (M)';
sListBox1.items[5]:=edit3.T-
ext;

sListBox1.Items[6]:='MASS-
A JENIS SELANG
(KG/M3)';
sListBox1.items[7]:=edit4.T-
ext;

sListBox1.Items[8]:='VOLU-
ME SELANG (M3)';

sListBox1.items[9]:=edit5.T-
ext;

sListBox1.Items[10]:='MAS-
SA SELANG (KG)';
sListBox1.items[11]:=edit6.
Text;

sListBox1.Items[12]:='BOU-
YANCY FORCE SELANG
(KG)';
sListBox1.items[13]:=edit7.
Text;

sListBox1.Items[14]:='KET-
ERANGAN:';
sListBox1.items[15]:=edit8.
Text;
end;

procedure
TSelangv.sButton2Click(Se-
nder: TObject);
begin
edit1.Clear;
edit2.Clear;
edit3.Clear;
edit4.Clear;
Edit5.clear;
Edit6.Clear;
Edit7.Clear;
Edit8.Clear;
sListBox1.clear;
end;

procedure
TSelangv.sButton3Click(Se-
nder: TObject);
begin
if
SaveDialog1.Execute
then
sListBox1.items.SaveToFile(
'SELANG.txt');
end;
end.

```

1. Source code unit karung

unit karung;

interface

uses

Windows, Messages,
SysUtils, Variants, Classes,
Graphics, Controls, Forms,
Dialogs, StdCtrls, ExtCtrls,
Menus, sSkinProvider,
sAlphaListBox, sPanel,
sLabel, sButton;

type

Tkarungv = class(TForm)

Edit1: TEdit;
Edit2: TEdit;
Edit3: TEdit;
Edit4: TEdit;
Edit5: TEdit;
Edit6: TEdit;
Edit7: TEdit;
Edit8: TEdit;
SaveDialog1:
TSaveDialog;

sSkinProvider1:

TsSkinProvider;

sListBox1: TsListBox;
sPanel1: TsPanel;
sPanel2: TsPanel;
sPanel3: TsPanel;
sButton1: TsButton;
sButton2: TsButton;
sButton3: TsButton;
sPanel4: TsPanel;
sLabel1: TsLabel;
sLabel2: TsLabel;
sLabel3: TsLabel;
sLabel4: TsLabel;
sLabel5: TsLabel;
sLabel6: TsLabel;
sPanel5: TsPanel;
sLabel7: TsLabel;
sLabel8: TsLabel;
sLabel9: TsLabel;
sLabel10: TsLabel;

sLabel11: TsLabel;

sLabel12: TsLabel;

sLabel13: TsLabel;

sLabel14: TsLabel;

sLabel15: TsLabel;

procedure

sButton1Click(Sender:
TObject);

procedure

sButton2Click(Sender:
TObject);

procedure

sButton3Click(Sender:
TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

karungv: Tkarungv;

implementation

{SR *.dfm}

procedure

Tkarungv.sButton1Click(Sen-
der: TObject);

Var

kode: Integer;

jukar,makar,mjekar,mtokar,v

pekar,vtokar,bfkar,sfkar:real;

matotkar,volperkar,voltofkar,

sinkkar:string;

begin

Val(Edit1.Text,jukar,Kode);

Val(Edit2.Text,makar,Kode);

Val(Edit3.Text,mjekar,Kode)

;

mtokar:=jukar*makar;

str(mtokar:0:2,matotkar);

Edit4.Text:=matotkar;

vpekar:=makar/mjekar;

str(vpekar:0:2,volperkar);

edit5.Text:=volperkar;

vtokar:=jukar*vpekar;

str(vtokar:0:2,voltofkar);

edit6.Text:=voltofkar;

bfkar:=1.025*vtokar;

sfkar:=bfkar-mtokar;

str(sfkar:0:2,sinkkar);

edit7.Text:=sinkkar;

begin

if sfkar>0 then

edit8.Text:='MENGAPUNG'

else

edit8.Text:='TENGCELAM'

end;

sListBox1.Clear;

sListBox1.Items[0]:=JUML

AH KARUNG (BUAH):';

sListBox1.items[1]:=edit1.T

ext;

sListBox1.Items[2]:=MASS

A PER KARUNG (KG):';

sListBox1.Items[3]:=edit2.T

ext;

sListBox1.Items[4]:=MASS

A JENIS KARUNG

(KG/M3):';

sListBox1.items[5]:=edit3.T

ext;

sListBox1.Items[6]:=MASS

A TOTAL KARUNG

(KG):';

sListBox1.items[7]:=edit4.T

ext;

sListBox1.Items[8]:=VOLU

ME PER KARUNG (M3):';

sListBox1.items[9]:=edit5.T

ext;



```

sListBox1.Items[10]:='VOL
UME TOTAL KARUNG
(M3)';
sListBox1.items[11]:=edit6.
Text;

sListBox1.Items[12]:='BUO
YANCY FORCE KARUNG
(KG)';
sListBox1.items[13]:=edit7.
Text;

sListBox1.Items[14]:='KET
ERANGAN.';
sListBox1.items[15]:=edit8.
Text;
end;

procedure
Tkarungv.sButton2Click(Sen
der: TObject);
begin
edit1.Clear;
edit2.Clear;
edit3.Clear;
edit4.Clear;
Edit5.clear;
Edit6.Clear;
Edit7.Clear;
Edit8.Clear;
sListBox1.clear;
end;

procedure
Tkarungv.sButton3Click(Sen
der: TObject);
begin
if
SaveDialog1.Execute
then
sListBox1.Items.SaveToFile(
'KARUNG.txt');
end;
end.

```

m. *Source code unit rafia*

```

unit Rafia;

interface

uses
  Windows, Messages,
  SysUtils, Variants, Classes,
  Graphics, Controls, Forms,
  Dialogs, StdCtrls, ExtCtrls,
  sSkinProvider, sPanel,
  sAlphaListBox,
  sButton, sLabel;

type
  TRafiav = class(TForm)
    Edit1: TEdit;
    Edit2: TEdit;
    Edit3: TEdit;
    Edit4: TEdit;
    Edit5: TEdit;
    Edit6: TEdit;
    Edit7: TEdit;
    Edit8: TEdit;
    Edit9: TEdit;
    SaveDialog1:
TSaveDialog;
    sSkinProvider1:
TSkinProvider;
    sListBox1: TsListBox;
    sPanel1: TsPanel;
    sPanel2: TsPanel;
    sButton1: TsButton;
    sButton2: TsButton;
    sButton3: TsButton;
    sPanel3: TsPanel;
    sLabel1: TsLabel;
    sLabel2: TsLabel;
    sLabel3: TsLabel;
    sLabel4: TsLabel;
    sLabel5: TsLabel;
    sLabel6: TsLabel;
    sLabel7: TsLabel;
    sLabel8: TsLabel;
    sPanel4: TsPanel;
    sLabel9: TsLabel;
    sLabel10: TsLabel;
    sLabel11: TsLabel;
    sLabel12: TsLabel;
    sLabel13: TsLabel;
    sLabel14: TsLabel;
    sLabel15: TsLabel;
    sLabel16: TsLabel;
    sLabel17: TsLabel;

  procedure
sButton1Click(Sender:
TObject);
  procedure
sButton2Click(Sender:
TObject);
  procedure
sButton3Click(Sender:
TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

  var
    Rafiav: TRafiav;

  implementation

{$R * .dfm}

  procedure
TRafiav.sButton1Click(Send
er: TObject);
  Var
kode:Integer;
jgraf,ppgraf,mpgraf,mjraf,ptr
af,mtraf,vtraf,bfrac,sfrac:real;
patotraf,matotraf,vototraf,sin
kraf:string;

  begin
    Val(Edit1.Text,jgraf,Kode);
    Val(Edit2.Text,ppgraf,Kode)
    ;
    Val(Edit3.Text,mpgraf,Kode
    );
    Val(edit4.Text,mjraf,kode);

    ptraf:=jgraf*ppgraf;
    str(ptraf:0:2,patotraf);
    Edit5.Text:=patotraf;

    mtraf:=jgraf*mpgraf;

```

```

str(mtraf:0:2,matotraf);
Edit6.Text:=matotraf;

vtraf:=mtraf/mjraf;
str(vtraf:0:2,vototraf);
Edit7.Text:=vototraf;

bfraf:=1.025*vtraf;

sfraf:=bfraf-mtraf;
str(sfraf:0:2,sinkraf);
Edit8.Text:=sinkraf;

begin
if sfraf>0 then
edit9.Text:='MENGAPUNG'
else
edit9.Text:='TENGSELAM'
end;

sListBox1.Clear;
sListBox1.Items[0]:='JUMLAH GULUNG RAFIA (GULUNG)';
sListBox1.Items[1]:=edit1.Text;

sListBox1.Items[2]:='PANJANG PER GULUNG RAFIA (M)';
sListBox1.Items[3]:=edit2.Text;

sListBox1.Items[4]:='MASSA PER GULUNG RAFIA (KG)';
sListBox1.Items[5]:=edit3.Text;

sListBox1.Items[6]:='MASSA JENIS RAFIA (KG/M3)';
sListBox1.Items[7]:=edit4.Text;

sListBox1.Items[8]:='PANJANG TOTAL RAFIA (M)';
sListBox1.Items[9]:=edit5.Text;

```

```

sListBox1.Items[10]:='MASA TOTAL RAFIA (KG)';
sListBox1.Items[11]:=edit6.Text;

sListBox1.Items[12]:='VOLUME TOTAL RAFIA (M3)';
sListBox1.Items[13]:=edit7.Text;

sListBox1.Items[14]:='BUOYANCY FORCE RAFIA (KG)';
sListBox1.Items[15]:=edit8.Text;

sListBox1.Items[16]:='KETERANGAN:';
sListBox1.Items[17]:=edit9.Text;
end;

procedure TRafiav.sButton2Click(Sender: TObject);
begin
edit1.Clear;
edit2.Clear;
edit3.Clear;
edit4.Clear;
edit5.Clear;
edit6.Clear;
edit7.Clear;
edit8.Clear;
edit9.Clear;
sListBox1.Clear;
end;

procedure TRafiav.sButton3Click(Sender: TObject);
begin
if SaveDialog1.Execute then
sListBox1.Items.SaveToFile('RAFIA.txt');
end;

```

end.

n. *Source code unit tali hillban*

```

unit Tali_hillban;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms, Dialogs, StdCtrls, ExtCtrls, sSkinProvider, sAlphaListBox, sButton, sPanel, sLabel;

type
  TTali_Hillbanv =
class(TForm)
  Edit1: TEdit;
  Edit2: TEdit;
  Edit3: TEdit;
  Edit4: TEdit;
  Edit5: TEdit;
  Edit6: TEdit;
  Edit7: TEdit;
  Edit8: TEdit;
  Edit9: TEdit;
  SaveDialog1: TSaveDialog;
  TSaveDialog: sSkinProvider1;
  TsSkinProvider: sListBox1: TsListBox;
  sPanel1: TsPanel;
  sButton1: TsButton;
  sButton2: TsButton;
  sButton3: TsButton;
  sPanel2: TsPanel;
  sPanel3: TsPanel;
  sLabel1: TsLabel;
  sLabel2: TsLabel;
  sLabel3: TsLabel;
  sLabel4: TsLabel;
  sPanel4: TsPanel;
  sLabel5: TsLabel;
  sLabel6: TsLabel;

```



```

sLabel7: TLabel;
sLabel8: TLabel;
sLabel9: TLabel;
sLabel10: TLabel;
sLabel11: TLabel;
sLabel12: TLabel;
sLabel13: TLabel;
sPanel5: TsPanel;
sLabel14: TLabel;
sLabel15: TLabel;
sLabel16: TLabel;
sLabel17: TLabel;
procedure
sButton1Click(Sender:
TObject);
    procedure
sButton2Click(Sender:
TObject);
    procedure
sButton3Click(Sender:
TObject);
private
    { Private declarations }
public
    { Public declarations }
end;
var
    Tali_Hillbanv:
TTali_Hillbanv;

implementation

{$R *.dfm}

procedure
TTali_Hillbanv.sButton1Clic
k(Sender: TObject);
Var
kode: Integer;
jgthp,ppgthp,rthp,mjthp,ptth
p,vtthp,mtthp,bfthp,sfthp:real
;
pantotthp,voltotthp,masstotth
p,buoythp,sinkthp:string;

begin
Val(Edit1.Text,jgthp,Kode);

Val(Edit2.Text,ppgthp,Kode)
;
Val(Edit3.Text,rthp,Kode);
Val(Edit4.Text,mjthp,Kode);

ptthp:=jgthp*ppgthp;
str(ptthp:0:2,pantotthp);
Edit5.Text:=pantotthp;

vtthp:=3.14*rthp*rthp*ptthp;
str(vtthp:0:2,voltotthp);
Edit6.Text:=voltotthp;

mtthp:=mjthp*vtthp;
str(mtthp:0:2,masstotthp);
Edit7.Text:=masstotthp;

bfthp:=1.025*vtthp;

sfthp:=bfthp-mtthp;
str(sfthp:0:2,sinkthp);
Edit8.Text:=sinkthp;

begin
if sfthp>0 then
edit9.Text:='MENGAPUNG'
else
edit9.Text:='TENGSELAM'
end;

sListBox1.Clear;
sListBox1.Items[0]:='JUML
AH GULUNG TALI
(GULUNG)';
sListBox1.items[1]:=edit1.T
ext;

sListBox1.Items[2]:='PANJ
ANG PER GULUNG TALI
(M)';
sListBox1.Items[3]:=edit2.T
ext;

sListBox1.Items[4]:='JARI-
JARI TALI (M)';
sListBox1.items[5]:=edit3.T
ext;

sListBox1.Items[6]:='MASS
A JENIS TALI (KG/M3)';
sListBox1.items[7]:=edit4.T
ext;

sListBox1.Items[8]:='PANJ
ANG TOTAL TALI (M)';
sListBox1.items[9]:=edit5.T
ext;

sListBox1.Items[10]:='VOL
UME TOTAL TALI (M3)';
sListBox1.items[11]:=edit6.
Text;

sListBox1.Items[12]:='MAS
SA TOTAL TALI (KG)';
sListBox1.items[13]:=edit7.
Text;

sListBox1.Items[14]:='BUO
YANCY FORCE TALI
(KG)';
sListBox1.items[15]:=edit8.
Text;

sListBox1.Items[16]:='KET
ERANGAN';
sListBox1.Items[17]:=edit9.
Text;

end;

procedure
TTali_Hillbanv.sButton2Clic
k(Sender: TObject);
begin
edit1.Clear;
edit2.Clear;
edit3.Clear;
edit4.Clear;
Edit5.clear;
Edit6.Clear;
Edit7.Clear;
Edit8.Clear;
Edit9.Clear;
sListBox1.clear;
end;

```



```

procedure
TTali_Hillbanv.sButton3Click(Sender: TObject);
begin
if
SaveDialog1.Execute
then
sListBox1.Items.SaveToFile(
'TALI_HILLBAN-
PEMBERAT.txt');
end;

end.

```

o. *Source code* unit

tali utama

```
unit Tali_utama;
```

```
interface
```

```
uses
```

```

Windows, Messages,
SysUtils, Variants, Classes,
Graphics, Controls, Forms,
Dialogs, ExtCtrls, StdCtrls,
sSkinProvider,
sAlphaListBox, sButton,
sPanel, sLabel;
```

```
type
```

```

TTaliUtama =
class(TForm)

```

```

    Edit1: TEdit;
    Edit2: TEdit;
    Edit3: TEdit;
    Edit4: TEdit;
    Edit5: TEdit;
    Edit6: TEdit;
    Edit7: TEdit;
    Edit8: TEdit;
    Edit9: TEdit;
    Edit10: TEdit;
    SaveDialog1:

```

```
TSaveDialog;
```

```
    sSkinProvider1:
```

```
TsSkinProvider;
```

```
    sListBox1: TsListBox;
```

```
    sPanel1: TsPanel;
```

```

sButton1: TsButton;
sButton2: TsButton;
sButton3: TsButton;
sPanel2: TsPanel;
sLabel1: TsLabel;
sLabel2: TsLabel;
sLabel3: TsLabel;
sLabel4: TsLabel;
sLabel5: TsLabel;
sPanel3: TsPanel;
sLabel6: TsLabel;
sLabel7: TsLabel;
sLabel8: TsLabel;
sLabel9: TsLabel;
sLabel10: TsLabel;
sPanel4: TsPanel;
sLabel11: TsLabel;
sLabel12: TsLabel;
sLabel13: TsLabel;
sLabel14: TsLabel;
sLabel15: TsLabel;
sPanel5: TsPanel;
sLabel16: TsLabel;
sLabel17: TsLabel;
sLabel18: TsLabel;
sLabel19: TsLabel;
procedure
sButton1Click(Sender:
TObject);
    procedure
sButton2Click(Sender:
TObject);
    procedure
sButton3Click(Sender:
TObject);
    private
        { Private declarations }
    public
        { Public declarations }
end;

var
    TaliUtama: TTaliUtama;

implementation
{$SR *.dfm}

```

```

procedure
TTaliUtama.sButton1Click(S
ender: TObject);
Var
kode:Integer;
ppgt,jgt,rct,jct,mjt,pt,vt,mt,bf
t,sft:real;
panjt,volt,masst,sinkt:string;

begin
Val(Edit1.Text,ppgt,Kode);
Val(Edit2.Text,jgt,Kode);
Val(Edit3.Text,rct,Kode);
Val(Edit4.Text,jct,kode);
Val(Edit5.Text,mjt,Kode);

pt:=ppgt*jgt;
str(pt:0:2,panjt);
Edit6.Text:=panjt;

vt:=3.14*rct*rct*jct*pt;
str(vt:0:2,volt);
edit7.Text:=volt;

mt:=mjt*vt;
str(mt:0:2,masst);
edit8.Text:=masst;

bft:=1.025*vt;

sft:=bft-mt;
str(sft:0:2,sinkt);
edit9.Text:=sinkt;

begin
if sft>0 then
edit10.Text:='MENGAPUN
G'
else
edit10.Text:='TENGGE
LA
M'
end;

sListBox1.Clear;
sListBox1.Items[0]:='PANJ
ANG PER GULUNG (M)';
sListBox1.Items[1]:=edit1.T
ext;

```

```
sListBox1.Items[2]='JUMLAH
GULUNG (GULUNG)';
sListBox1.Items[3]=edit2.Text;
```

```
sListBox1.Items[4]='JARI-JARI
CABANG (M)';
sListBox1.items[5]=edit3.Text;
```

```
sListBox1.Items[6]='JUMLAH
CABANG (CABANG)';
sListBox1.items[7]=edit4.Text;
```

```
sListBox1.Items[8]='MASSA
JENIS TALI (KG/M3)';
sListBox1.items[9]=edit5.Text;
```

```
sListBox1.Items[10]='PANJANG
TALI (M)';
sListBox1.items[11]=edit6.Text;
```

```
sListBox1.Items[12]='VOLUME
TALI (M3)';
sListBox1.items[13]=edit7.Text;
```

```
sListBox1.Items[14]='MASSA
TALI (KG)';
sListBox1.items[15]=edit8.Text;
```

```
sListBox1.Items[16]='BOUYANCY
FORCE TALI (KG)';
sListBox1.Items[17]=edit9.Text;
```

```
sListBox1.Items[18]='KETERANGAN';
sListBox1.Items[19]=edit10.Text;
```

```
procedure
TTaliUtama.sButton2Click(Sender: TObject);
begin
edit1.Clear;
edit2.Clear;
edit3.Clear;
edit4.Clear;
Edit5.clear;
Edit6.Clear;
Edit7.Clear;
Edit8.Clear;
Edit9.Clear;
Edit10.Clear;
sListBox1.Clear;
end;
```

```
procedure
TTaliUtama.sButton3Click(Sender: TObject);
begin
if
SaveDialog1.Execute
then
sListBox1.items.SaveToFile('TALI_UTAMA.txt');
end;
end.
```

p. *Source code unit extra bouyancy*

```
unit extra;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms, Dialogs, StdCtrls, ExtCtrls, ComCtrls, jpeg, sAlphaListBox, sPanel, sLabel, sButton, sSkinProvider;

type
  Textra = class(TForm)
```

```
Edit1: TEdit;
Edit2: TEdit;
Edit3: TEdit;
Edit4: TEdit;
Edit5: TEdit;
Edit6: TEdit;
Edit7: TEdit;
Edit8: TEdit;
Edit9: TEdit;
Edit10: TEdit;
Edit11: TEdit;
Edit12: TEdit;
Edit13: TEdit;
Edit14: TEdit;
Edit15: TEdit;
ListBox1: TListBox;
ListBox2: TListBox;
ListBox3: TListBox;
ListBox4: TListBox;
ListBox5: TListBox;
ListBox6: TListBox;
ListBox7: TListBox;
ListBox8: TListBox;
ListBox9: TListBox;
ListBox10: TListBox;
ListBox11: TListBox;
ListBox12: TListBox;
ListBox13: TListBox;
ListBox14: TListBox;
ListBox15: TListBox;
OpenDialog1: TOpenDialog;
Label1: TLabel;
Label2: TLabel;
Label3: TLabel;
Label4: TLabel;
Label5: TLabel;
Label6: TLabel;
Label7: TLabel;
Label8: TLabel;
Label9: TLabel;
Label10: TLabel;
Label11: TLabel;
Label12: TLabel;
Label13: TLabel;
Label14: TLabel;
Label15: TLabel;
Edit31: TEdit;
Edit32: TEdit;
```



```

end;

Val(Edit12.Text,lb,Kode);
begin
if lb<0 then lb:=0
end;

Val(Edit13.Text,mb,Kode);
begin
if mb<0 then mb:=0
end;

Val(Edit14.Text,nb,Kode);
begin
if nb<0 then nb:=0
end;

Val(Edit15.Text,ob,Kode);
begin
if ob<0 then ob:=0
end;

{sink}
Val(Edit1.Text,aas,Kode);
begin
if aas>0 then aas:=0
end;

Val(Edit2.Text,bs,Kode);
begin
if bs>0 then bs:=0
end;

Val(Edit3.Text,cs,Kode);
begin
if cs>0 then cs:=0
end;

Val(Edit4.Text,ds,Kode);
begin
if ds>0 then ds:=0
end;

Val(Edit5.Text,es,Kode);
begin
if es>0 then es:=0
end;

Val(Edit6.Text,fs,Kode);
begin
if fs>0 then fs:=0
end;

Val(Edit7.Text,gs,Kode);
begin
if gs>0 then gs:=0
end;

Val(Edit8.Text,hs,Kode);
begin
if hs>0 then hs:=0
end;

Val(Edit9.Text,iis,Kode);
begin
if iis>0 then iis:=0
end;

Val(Edit10.Text,js,Kode);
begin
if js>0 then js:=0
end;

Val(Edit11.Text,ks,Kode);
begin
if ks>0 then ks:=0
end;

Val(Edit12.Text,ls,Kode);
begin
if ls>0 then ls:=0
end;

Val(Edit13.Text,ms,Kode);
begin
if ms>0 then ms:=0
end;

Val(Edit14.Text,ns,Kode);
begin
if ns>0 then ns:=0
end;

Val(Edit15.Text,os,Kode);
begin
if os>0 then os:=0
end;

xb:=ab+bb+cb+db+eb+fb+g
b+hb+ib+jb+kb+lb+mb+nb+
ob;
str(xb:0:2,xbuoy);
Edit31.Text:=xbuoy;

xs:=aas+bs+cs+ds+es+fs+gs
+hs+iis+js+ks+ls+ms+ns+os;
xsp:=-xs;
str(xsp:0:2,xsink);
edit32.Text:=xsink;

xbx:=((xb+xs)/xb)*100;
str(xbx:0:2,xbuoyx);
edit33.Text:=xbuoyx;

xb2:=xb+xs;
str(xb2:0:2,xbuoy2);
edit16.Text:=xbuoy2;

sListBox1.Items[0]:='BUOY
ANCY FORCE
ATRAKTOR (KG)';
sListBox1.Items[1]:=Edit1.T
ext;
sListBox1.Items[2]:='BUOY
ANCY FORCE
PEMBERAT ATRAKTOR
(KG)';
sListBox1.Items[3]:=Edit2.T
ext;
sListBox1.Items[4]:='BUOY
ANCY FORCE LAPISAN
FIBER (KG)';
sListBox1.Items[5]:=Edit3.T
ext;
sListBox1.Items[6]:='BUOY
ANCY FORCE LAPISAN
BESI (KG)';
sListBox1.Items[7]:=Edit4.T
ext;
sListBox1.Items[8]:='BUOY
ANCY FORCE LAPISAN
SEKAT (KG)';
sListBox1.Items[9]:=Edit5.T
ext;
sListBox1.Items[10]:='BUO
YANCY FORCE ISI
PONTON (KG)';

```



```

sListBox1.Items[11]:=Edit6.
Text;
sListBox1.Items[12]:=‘BUO
YANCY FORCE SWIVEL
(KG)’;
sListBox1.Items[13]:=Edit7.
Text;
sListBox1.Items[14]:=‘BUO
YANCY FORCE HILLBAN
(KG)’;
sListBox1.Items[15]:=Edit8.
Text;
sListBox1.Items[16]:=‘BUO
YANCY FORCE
PEMBERAT UTAMA
(KG)’;
sListBox1.Items[17]:=Edit9.
Text;
sListBox1.Items[18]:=‘BUO
YANCY FORCE WARING
(KG)’;
sListBox1.Items[19]:=Edit10
.Text;
sListBox1.Items[20]:=‘BUO
YANCY FORCE TALI
UTAMA (KG)’;
sListBox1.Items[21]:=Edit11
.Text;
sListBox1.Items[22]:=‘BUO
YANCY FORCE TALI
HILLBAN-PEMBERAT
(KG)’;
sListBox1.Items[23]:=Edit12
.Text;
sListBox1.Items[24]:=‘BUO
YANCY FORCE SELANG
(KG)’;
sListBox1.Items[25]:=Edit13
.Text;
sListBox1.Items[26]:=‘BUO
YANCY FORCE KARUNG
(KG)’;
sListBox1.Items[27]:=Edit14
.Text;
sListBox1.Items[28]:=‘BUO
YANCY FORCE TALI
PLASTIK (KG)’;
sListBox1.Items[29]:=Edit15
.Text;

sListBox1.Items[30]:=‘BUO
YANCY FORCE TOTAL
(KG)’;
sListBox1.Items[31]:=Edit31
.Text;
sListBox1.Items[32]:=‘SINK
ING FORCE TOTAL
(KG)’;
sListBox1.Items[33]:=Edit32
.Text;
sListBox1.Items[34]:=‘EXT
RA BUOYANCY FORCE
(KG)’;
sListBox1.Items[35]:=Edit16
.Text;
sListBox1.Items[36]:=‘EXT
RA BUOYANCY FORCE
(%)’;
sListBox1.Items[37]:=Edit33
.Text;
end;

procedure
Textrav.sButton3Click(Sender: TObject);
begin
edit1.Clear;
edit2.Clear;
edit3.Clear;
edit4.Clear;
Edit5.clear;
Edit6.Clear;
Edit7.Clear;
Edit8.Clear;
edit9.Clear;
edit10.Clear;
edit11.Clear;
edit12.Clear;
Edit13.clear;
Edit14.Clear;
Edit15.Clear;
Edit16.Clear;
Edit31.Clear;
Edit32.Clear;
Edit33.Clear;

sListBox1.clear;
end;

end;

procedure
Textrav.sButton4Click(Sender: TObject);
begin
if
SaveDialog1.Execute
then
sListBox1.items.SaveToFile(
‘EXTRA_BUOYANCY.txt’);
end;

procedure
Textrav.sButton1Click(Sender: TObject);
begin
if OpenFileDialog1.Execute then
begin
{listbox untuk buoyancy
force}
Listbox1.Items.LoadFromFile(
‘ATRAKTOR.txt’);
ListBox2.Items.LoadFromFile(
‘PEMBERAT_ATRAKTO
R.txt’);
Listbox3.Items.LoadFromFile(
‘LAPISAN_FIBER.txt’);
ListBox4.Items.LoadFromFile(
‘LAPISAN_BESI.txt’);
Listbox5.Items.LoadFromFile(
‘SEKAT_PONTON.txt’);
ListBox6.Items.LoadFromFile(
‘ISI_PONTON.txt’);
Listbox7.Items.LoadFromFile(
‘SWIVEL.txt’);
ListBox8.Items.LoadFromFile(
‘HILLBAN.txt’);
Listbox9.Items.LoadFromFile(
‘PEMBERAT_UTAMA.txt
’);
ListBox10.Items.LoadFromF
ile(‘WARING.txt’);
Listbox11.Items.LoadFromF
ile(‘TALI_UTAMA.txt’);
ListBox12.Items.LoadFromF
ile(‘TALI_HILLBAN-
PEMBERAT.txt’);

```

```
Listbox13.Items.LoadFromF
ile('SELANG.txt');
ListBox14.Items.LoadFromF
ile('KARUNG.txt');
ListBox15.Items.LoadFromF
ile('RAFIA.txt');
```

```
{Tedit untuk buoyancy
Force}
edit1.Text:=ListBox1.Items[
13];
edit2.Text:=ListBox2.Items[
17];
edit3.Text:=ListBox3.Items[
21];
edit4.Text:=ListBox4.Items[
21];
edit5.Text:=ListBox5.Items[
13];
edit6.Text:=ListBox6.Items[
17];
edit7.Text:=ListBox7.Items[
13];
edit8.Text:=ListBox8.Items[
23];
edit9.Text:=ListBox9.Items[
19];
edit10.Text:=ListBox10.Item
s[11];
edit11.Text:=ListBox11.Item
s[17];
edit12.Text:=ListBox12.Item
s[15];
edit13.Text:=ListBox13.Item
s[13];
edit14.Text:=ListBox14.Item
s[13];
edit15.Text:=ListBox15.Item
s[15];
end;
end;
end.
```

q. *Source code* unit
menu utama
 unit menu_utama;
 interface

```
uses
  Windows, Messages,
  SysUtils, Variants, Classes,
  Graphics, Controls, Forms,
  Dialogs, ExtCtrls, StdCtrls,
  jpeg, sSkinManager,
  sButton, sPanel,
  sSkinProvider, sLabel,
  sGauge;
```

```
type
  TMain_Menu =
  class(TForm)
    sButton1: TsButton;
    sButton2: TsButton;
    sButton3: TsButton;
    sButton4: TsButton;
    sButton5: TsButton;
    sButton6: TsButton;
    sButton7: TsButton;
    sButton8: TsButton;
    sButton9: TsButton;
    sButton10: TsButton;
    sButton11: TsButton;
    sButton12: TsButton;
    sButton13: TsButton;
    sButton14: TsButton;
    sButton15: TsButton;
    sButton16: TsButton;
    sPanel1: TsPanel;
    sSkinProvider1:
  TsSkinProvider;
    sGauge1: TsGauge;
    sLabelFX2: TsLabelFX;
    sLabelFX3: TsLabelFX;
    sLabelFX1: TsLabelFX;
```

```
procedure
  FormClose(Sender: TObject);
  var Action: TCloseAction);
  procedure
    sButton1Click(Sender:
  TObject);
  procedure
    sButton2Click(Sender:
  TObject);
  procedure
    sButton3Click(Sender:
  TObject);
```

```
procedure
  sButton4Click(Sender:
  TObject);
  procedure
  sButton5Click(Sender:
  TObject);
  procedure
  sButton6Click(Sender:
  TObject);
  procedure
  sButton7Click(Sender:
  TObject);
  procedure
  sButton8Click(Sender:
  TObject);
  procedure
  sButton9Click(Sender:
  TObject);
  procedure
  sButton10Click(Sender:
  TObject);
  procedure
  sButton11Click(Sender:
  TObject);
  procedure
  sButton12Click(Sender:
  TObject);
  procedure
  sButton13Click(Sender:
  TObject);
  procedure
  sButton14Click(Sender:
  TObject);
  procedure
  sButton15Click(Sender:
  TObject);
  procedure
  sButton16Click(Sender:
  TObject);
  procedure
  sLabelFX2Click(Sender:
  TObject);
  procedure
  sLabelFX3Click(Sender:
  TObject);
  private
    { Private declarations }
  public
    { Public declarations }
```

```

end;

var
  Main_Menu: TMain_Menu;

implementation

uses Atraktor,
  pemberat_Atraktor, fiber,
  besi, extra, Rafia, karung,
  Selang, Tali_hillban,
  Tali_utama, waring,
  Pemberat_Utama, hillban,
  Swivel, isiponton, sekat,
  help, infov;

{$R *.dfm}

procedure
  TMain_Menu.FormClose(Se
nder: TObject; var Action:
  TCloseAction);
begin
  showmessage('TERIMA
  KASIH TELAH
  MENGGUNAKAN UNIT
  SoFAD');
end;

procedure
  TMain_Menu.sButton1Click
(Sender: TObject);
Var
  pa:TPemberat_Atraktorv;
begin
  pa:=Pemberat_Atraktorv;
  pa.Show;
end;

procedure
  TMain_Menu.sButton2Click
(Sender: TObject);
var a:Tatraktorv;
begin
  a:=atraktorv;
  a.Show;
end;

procedure
  TMain_Menu.sButton3Click
(Sender: TObject);
var
  lp:Tfiberv;
begin
  lp:=fiberv;
  lp.show;
end;

procedure
  TMain_Menu.sButton4Click
(Sender: TObject);
var
  lb:Tform1;
begin
  lb:=form1;
  lb.show;
end;

procedure
  TMain_Menu.sButton5Click
(Sender: TObject);
var
  sek:Tsekatv;
begin
  sek:=sekatv;
  sek.Show;
end;

procedure
  TMain_Menu.sButton6Click
(Sender: TObject);
var
  si:Tisi;
begin
  si:=isi;
  si.show;
end;

procedure
  TMain_Menu.sButton7Click
(Sender: TObject);
var
  swi:Tswivelv;
begin
  swi:=swivelv;
  swi.show;
end;

procedure
  TMain_Menu.sButton8Click
(Sender: TObject);
var
  h:Thillbanv;
begin
  h:=hillbanv;
  h.show;
end;

procedure
  TMain_Menu.sButton9Click
(Sender: TObject);
var
  pu:TPemberatUtamax;
begin
  pu:=PemberatUtamax;
  pu.show;
end;

procedure
  TMain_Menu.sButton10Clic
k(Sender: TObject);
var
  w:Twaringv;
begin
  w:=waringv;
  w.show;
end;

procedure
  TMain_Menu.sButton11Clic
k(Sender: TObject);
var
  tu:TtaliUtama;
begin
  tu:=TaliUtama;
  tu.show;
end;

procedure
  TMain_Menu.sButton12Clic
k(Sender: TObject);
var
  thp:TTali_Hillbanv;
begin
  thp:=Tali_Hillbanv;
  thp.show;
end;

```

```

end;

procedure
TMain_Menu.sButton13Click(Sender: TObject);
var
s:Tselangv;
begin
s:=selangv;
s.show;
end;

procedure
TMain_Menu.sButton14Click(Sender: TObject);
var
k:Tkarungv;
begin
k:=karungv;
k.show;
end;

procedure
TMain_Menu.sButton15Click(Sender: TObject);
var
raf:Trafiav;
Begin
raf:=rafiav;
raf.show;
end;

procedure
TMain_Menu.sButton16Click(Sender: TObject);
var
x:Textrav;
begin
x:=extrav;
x.show;
end;

procedure
TMain_Menu.sLabelFX2Click(Sender: TObject);
var
hlp:Thehelpv;
begin
hlp:=helpv;
hlp.show;
end;

end;

procedure
TMain_Menu.sLabelFX3Click(Sender: TObject);
var
hlp:TInfo;
begin
hlp:=Info;
hlp.show;
end;

end.

```

r. *Source code unit splash screen*

```

unit splashfix;

interface

uses
  Windows, Messages,
  SysUtils, Classes, Graphics,
  Controls, Forms, Dialogs,
  sLabel, StdCtrls, ExtCtrls,
  sPanel, sSkinProvider,
  ComCtrls, sGauge,
  sButton, Buttons, sBitBtn;

type
  TfmSplash = class(TForm)
    sSkinProvider1:
      TSkinProvider;
    Timer1: TTimer;
    Timer2: TTimer;
    sPanel1: TsPanel;
    sLabelFX1: TsLabelFX;
    sLabelFX2: TsLabelFX;
    sGauge1: TsGauge;
    sLabel1: TsLabel;
  procedure
    Timer1Timer(Sender:
      TObject);
  procedure
    Timer2Timer(Sender:
      TObject);
  private
    end;

```

```

var
  fmSplash: TfmSplash;

implementation

{$R *.DFM}

procedure
  TfmSplash.Timer1Timer(Sender:
    TObject);
begin
  sgauge1.Progress:=sgauge1.
    Progress+1;

  if sgauge1.Progress=1 then
    lbl1.Caption:='MOHON
    MENUNGGU...';
  if sgauge1.Progress=5 then
    lbl1.Caption:='MEMBUK
    A APLIKASI...';
  if sgauge1.Progress=10 then
    lbl1.Caption:='ATRAKT
    OR.EXE';
  if sgauge1.Progress=12 then
    lbl1.Caption:='PEMBER
    AT ATRAKTOR.EXE';
  if sgauge1.Progress=14 then
    lbl1.Caption:='LAPISAN
    BESI.EXE';
  if sgauge1.Progress=16 then
    lbl1.Caption:='LAPISAN
    FIBER.EXE';
  if sgauge1.Progress=18 then
    lbl1.Caption:='SEKAT
    PONTON.EXE';
  if sgauge1.Progress=20 then
    lbl1.Caption:='ISI
    PONTON.EXE';
  if sgauge1.Progress=22 then
    lbl1.Caption:='TALI
    UTAMA.EXE';
  if sgauge1.Progress=24 then
    lbl1.Caption:='TALI-
    HILLBAN
    PEMBERAT.EXE';
  if sgauge1.Progress=26 then
    lbl1.Caption:='PEMBER
    AT UTAMA.EXE';

```

```

if sgaugel.Progress=28 then
  lbl1.Caption:='HILLBAN
.EXE';
if sgaugel.Progress=30 then
  lbl1.Caption:='WARING.
EXE';
if sgaugel.Progress=32 then
  lbl1.Caption:='KARUNG.
EXE';
if sgaugel.Progress=34 then
  lbl1.Caption:='SWIVEL.
EXE';
if sgaugel.Progress=36 then
  lbl1.Caption:='SELANG.
EXE';
if sgaugel.Progress=38 then
  lbl1.Caption:='TALI
PLASTIK.EXE';
if sgaugel.Progress=40 then
  lbl1.Caption:='EXTRA
BUOYANCY.EXE';
if sgaugel.Progress=42 then
  lbl1.Caption:='MEMBUK
A APLIKASI SELESAI';
if sgaugel.Progress=45 then
  lbl1.Caption:='SEKARA
NG MEMBUKA
IDENTITAS PEMBUAT
APLIKASI...';
if sgaugel.Progress=55 then
  lbl1.Caption:='INISIAL:
OJ';
if sgaugel.Progress=60 then
  lbl1.Caption:='OCE
JUNJUNAN';
if sgaugel.Progress=65 then
  lbl1.Caption:='C4405163
7';
if sgaugel.Progress=70 then
  lbl1.Caption:='MAYOR
TEKNOLOGI DAN
MANAJEMEN
PERIKANAN TANGKAP';
if sgaugel.Progress=75 then
  lbl1.Caption:='DEPARTE
MEN PEMANFAATAN
SUMBERDAYA
PERIKANAN';

```

```

if sgaugel.Progress=80 then
  lbl1.Caption:='FAKULT
AS PERIKANAN DAN
ILMU KELAUTAN';
if sgaugel.Progress=85 then
  lbl1.Caption:='INSTITUT
PERTANIAN BOGOR';
if sgaugel.Progress=90 then
  lbl1.Caption:='MEMBUK
A IDENTITAS PEMBUAT
APLIKASI SELESAI';
if sgaugel.Progress=95 then
  lbl1.Caption:='MENYIA
PKAN TEMA
APLIKASI...';
end;

procedure
TfmSplash.Timer2Timer(Se
nder: TObject);
begin
  timer2.Enabled:=false;
end;

end.

```

S. *Source code data module*

```

unit datamodul;

interface

uses
  Windows, Messages,
  SysUtils, Classes, Graphics,
  Controls, Forms, Dialogs,
  sSkinManager;

type
  TDataModule1 =
class(TDataModule)
  sSkinManager1:
  TsSkinManager;
end;

var
  DataModule1:
  TDataModule1;

```

implementation

```
{SR *.DFM}
```

end.

t. *Source code unit bantuan*

unit help;

interface

uses

Windows, Messages,
SysUtils, Variants, Classes,
Graphics, Controls, Forms,
Dialogs, ComCtrls,
StdCtrls, sComboBox,
sAlphaListBox,
sSkinProvider,
ExtCtrls, sPanel;

type

```

Thelpv = class(TForm)
  sListBox1: TsListBox;
  sListBox2: TsListBox;
  sSkinProvider1:
  TsSkinProvider;
  sPanel1: TsPanel;
  sComboBox1:
  TsComboBox;
  sComboBox2:
  TsComboBox;
  Memo1: TMemo;
  procedure
  FormCreate(Sender:
  TObject);
  procedure
  sComboBox1Change(Sender
  : TObject);
  procedure
  sComboBox2Change(Sender
  : TObject);
  private
  { Private declarations }
  public
  { Public declarations }

```



```
sampai 40 mm (No. 1½ inci).
';
sListBox1.Items[25]='Meme-
liki massa jenis 2.6 g/cm3. ';
```

```
sComboBox1.Items[18]='Be-
ton K225';
sListBox1.Items[26]='Meru-
pakan campuran antara
semen-pasir-agregat kasar-
air. ';
```

```
sListBox1.Items[27]='Deng-
an perbandingan 1:2:3:0.5. ';
```

```
sListBox1.Items[28]='Meme-
liki massa jenis 2.325
g/cm3. ';
```

```
sComboBox2.Items[0]='Um-
um';
sListBox2.Items[0]='Aplika-
si ini tidak bisa menghitung
angka nol. ';
```

```
sListBox2.Items[1]='Akan
muncul peringatan "Invalid
Floating Point Operation". ';
```

```
sListBox2.Items[2]='Tekan
tombol "HAPUS" untuk
mengulangi perhitungan. ';
```

```
end;
```

```
procedure
Thelpv.sComboBox1.Change
(Sender: TObject);
begin
memo1.Clear;
case
sComboBox1.ItemIndex of
0:memo1.Lines.Add(slistbox
1.Items[0]+slistbox1.Items[1
]);
1:memo1.Lines.Add(slistbox
1.Items[2]+slistbox1.Items[3
]);
2:memo1.Lines.Add(slistbox
1.Items[4]+slistbox1.Items[5
]);
3:memo1.Lines.Add(slistbox
1.Items[6]+slistbox1.Items[7
]);
```

```
4:memo1.Lines.Add(slistbox
1.Items[8]+slistbox1.Items[9
]);
5:memo1.Lines.Add(slistbox
1.Items[10]+slistbox1.Items[
11]);
6:memo1.Lines.Add(slistbox
1.Items[12]+slistbox1.Items[
13]);
7:memo1.Lines.Add(slistbox
1.Items[14]);
8:memo1.Lines.Add(slistbox
1.Items[15]);
9:memo1.Lines.Add(slistbox
1.Items[16]);
10:memo1.Lines.Add(slistbo
x1.Items[17]);
11:memo1.Lines.Add(slistbo
x1.Items[18]);
12:memo1.Lines.Add(slistbo
x1.Items[19]);
13:memo1.Lines.Add(slistbo
x1.Items[20]);
14:memo1.Lines.Add(slistbo
x1.Items[21]);
15:memo1.Lines.Add(slistbo
x1.Items[22]);
16:memo1.Lines.Add(slistbo
x1.Items[23]);
17:memo1.Lines.Add(slistbo
x1.Items[24]+slistbox1.Items
[25]);
18:memo1.Lines.Add(slistbo
x1.Items[26]+slistbox1.Items
[27]+slistbox1.Items[28]);
```

```
end;
```

```
end;
```

```
procedure
Thelpv.sComboBox2.Change
(Sender: TObject);
begin
memo1.Clear;
case
sComboBox2.ItemIndex of
0:Memo1.Lines.Add(slistbox
2.Items[0]+slistbox2.Items[1
]+slistbox2.Items[2]);
```

```
end;
end;
end.
```

u. *Source code unit info*

```
unit infov;
```

```
interface
```

```
uses
```

```
Windows, Messages,
SysUtils, Variants, Classes,
Graphics, Controls, Forms,
Dialogs, jpeg, ExtCtrls,
sPanel, sSkinProvider,
StdCtrls, sLabel;
```

```
type
```

```
TInfo = class(TForm)
sPanel1: TsPanel;
sSkinProvider1:
TsSkinProvider;
sPanel2: TsPanel;
sPanel3: TsPanel;
Image1: TImage;
sLabelFX1: TsLabelFX;
sLabelFX2: TsLabelFX;
sLabelFX3: TsLabelFX;
sLabelFX4: TsLabelFX;
sLabelFX5: TsLabelFX;
sLabelFX6: TsLabelFX;
sLabelFX7: TsLabelFX;
sLabelFX8: TsLabelFX;
sLabelFX9: TsLabelFX;
sLabelFX10: TsLabelFX;
sLabelFX11: TsLabelFX;
sLabelFX12: TsLabelFX;
```

```
private
```

```
{ Private declarations }
```

```
public
```

```
{ Public declarations }
```

```
end;
```

```
var
```

```
Info: TInfo;
```

```

implementation
    {SR *.dfm}

end.

v. Source code
program menu
program menu;

uses
    Forms,
    menu_utama in
'menu_utama.pas'
{Main_Menu},
    Atraktor in 'Atraktor.pas'
{atraktor},
    besi in 'besi.pas' {Form1},
    extra in 'extra.pas' {extrav},
    fiber in 'fiber.pas' {fiberv},
    hillban in 'hillban.pas'
{Hillbanv},
    isiponton in 'isiponton.pas'
{Isi},
    karung in 'karung.pas'
{karungv},
    pemberat_Atraktor in
'pemberat_Atraktor.pas'
{Pemberat_Atraktorv},
    Pemberat_Utama in
'Pemberat_Utama.pas'
{PemberatUtamax},
    Rafia in 'Rafia.pas'
{Rafiav},
    sekat in 'sekat.pas'
{Sekatv},
    Selang in 'Selang.pas'
{Selangv},
    Swivel in 'Swivel.pas'
{Swivelv},
    Tali_hillban in
'Tali_hillban.pas'
{Tali_Hillbanv},
    Tali_utama in
'Tali_utama.pas'
{TaliUtama},
    waring in 'waring.pas'
{waringv},

    datamodul in
'datamodul.pas'
{DataModule1:
TDataModule},
    splashfix in 'splashfix.pas'
{fmSplash},
    help in 'help.pas' {helpv},
    infov in 'infov.pas' {Info};
{SE .exe}

{SR *.res}

begin
    Application.CreateForm(TD
ataModule1, DataModule1);

    fmsplash:=Tfmsplash.Create
(application);
    fmsplash.Show;
    fmsplash.Update;
    while
    fmsplash.Timer2.Enabled do

    Application.ProcessMessage
s;
        Application.Initialize;

    Application.Title:='SoFAD';

    Application.CreateForm(TM
ain_Menu, Main_Menu);

    Application.CreateForm(Tatr
aktorv, atraktorv);

    Application.CreateForm(TFo
rm1, Form1);

    Application.CreateForm(Tex
trav, extrav);

    Application.CreateForm(Tfib
erv, fiberv);

    Application.CreateForm(THi
llbanv, Hillbanv);

    Application.CreateForm(TIsi
, Isi);

    Application.CreateForm(Tka
rungv, karungv);

    Application.CreateForm(TPe
mberat_Atraktorv,
Pemberat_Atraktorv);

    Application.CreateForm(TPe
mberatUtamax,
PemberatUtamax);

    Application.CreateForm(TRa
fiav, Rafiav);

    Application.CreateForm(TSe
katv, Sekatv);

    Application.CreateForm(TSe
langv, Selangv);

    Application.CreateForm(TS
wivelv, Swivelv);

    Application.CreateForm(TTa
li_Hillbanv, Tali_Hillbanv);

    Application.CreateForm(TTa
liUtama, TaliUtama);

    Application.CreateForm(Tw
aringv, waringv);

    Application.CreateForm(The
lpv, helpv);

    Application.CreateForm(TIn
fo, Info);
        fmsplash.Hide;
        fmsplash.Free;
        Application.Run;
    end.

```