

**PENERAPAN TEKNOLOGI *GENERALIZED*
MULTI-PROTOCOL LABEL SWITCHING (GMPLS)
PADA *FIBER DISTRIBUTED DATA INTERFACE (FDDI)***

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Teknik

UNIVERSITAS BRAWIJAYA



Disusun oleh:

HERU SETIAWAN
NIM. 0210630060 - 63

**DEPARTEMEN PENDIDIKAN NASIONAL
UNIVERSITAS BRAWIJAYA
FAKULTAS TEKNIK
MALANG
2009**

**PENERAPAN TEKNOLOGI *GENERALIZED*
MULTI-PROTOCOL LABEL SWITCHING (GMPLS)
PADA *FIBER DISTRIBUTED DATA INTERFACE (FDDI)***

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Teknik

UNIVERSITAS BRAWIJAYA



Disusun oleh:

HERU SETIAWAN
NIM. 0210630060 - 63

DOSEN PEMBIMBING

Dwi Fadila Kurniawan, ST, MT
NIP. 132258185

Ir. Endah Budi Purnomowati, MT
NIP. 131839362

**PENERAPAN TEKNOLOGI *GENERALIZED*
MULTI-PROTOCOL LABEL SWITCHING (GMPLS)
PADA *FIBER DISTRIBUTED DATA INTERFACE (FDDI)***

Disusun oleh:

HERU SETIAWAN
NIM. 0210630060 - 63

Skripsi ini telah diuji dan dinyatakan lulus pada
tanggal 7 Agustus 2009

Majelis Penguji:

Ir. Erfan Achmad Dahlan, MT
NIP. 131124663

Rusmi Ambarwati, ST, MT
NIP. 132258188

Ali Mustofa, ST, MT.
NIP. 132258187

Mengetahui:
Ketua Jurusan Teknik Elektro

Ir. Heru Nurwarsito, M.Kom
NIP. 131879033

ABSTRAK

HERU SETIAWAN. Agustus 2009. Penerapan Teknologi *Generalized Multi-Protocol Label Switching (GMPLS)* Pada *Fiber Distributed Data Interface (FDDI)*. Dosen Pembimbing: Dwi Fadila K. ST, MT dan Ir. Endah Budi P., MT

FDDI merupakan standar protokol yang memanfaatkan teknologi *dual-ring* yang memiliki performansi tinggi dan mampu beroperasi dengan kecepatan 100 Mbps, dimana tiap *station* digaransi untuk dapat akses ke jaringan pada periode tertentu yang diatur pada saat jaringan mulai beroperasi. Hanya saja pada FDDI tidak ada fitur manajemen QoS (*quality of service*) (Chao Chen, Fang Liu, Yong Wang, 2001:25). Dengan menerapkan GMPLS pada FDDI, GMPLS dapat menjaga kecepatan pengiriman data yang telah ada, juga dapat menjaga kualitas data yang dikirim. GMPLS sendiri merupakan konvergensi vertikal, yang tujuannya adalah untuk menyediakan jaringan dengan QoS yang konsisten dengan pengendalian penuh dan terintegrasi.

Tujuan penulisan skripsi ini adalah untuk menganalisis: 1). mekanisme kerja sistem GMPLS dan jaringan FDDI, 2). konfigurasi penerapan GMPLS pada jaringan FDDI, dan 3). pengaruh penerapan GMPLS pada jaringan FDDI terhadap kinerja sistem. Yang dianalisis dalam skripsi ini adalah: 1). media transmisi, 2). *Bit Error Rate (BER)*, 3). *Delay*, 4). *Bit Rate*, dan 5). *Throughput*. Asumsi yang digunakan analisis tersebut: 1). media transmisi yang digunakan dan dianalisis adalah serat optik *multi-mode graded index* yang bekerja pada panjang gelombang sekitar 1300 nm, 2). *Delay* propagasi pada serat optik adalah 5,085 $\mu\text{s}/\text{km}$ dan kecepatan propagasi cahaya dalam serat optik adalah 3×10^8 m, 3). nilai *Target Token Rotation Time (TTRT)* yang digunakan adalah 4 ms dan 8 ms, 4). panjang transmisi data adalah 100 km, dan 5). jumlah *node* yang dianalisis sebanyak 100 *node* dengan jarak antar *node* adalah sama. Analisis dilakukan pada perhitungan *frame* FDDI sebelum dan sesudah berada di jaringan GMPLS. Di dalam jaringan GMPLS, *frame* FDDI dienkapsulasi dengan memasang *header* GMPLS yang disebut dengan label. Proses enkapsulasi ini akan menambah jumlah *delay* pengiriman data sebesar 36 μs dan mengurangi *throughput* sebesar 0,04 Mbps (untuk panjang *frame* 500 *byte*), tetapi tidak mempengaruhi *Bit Error Rate (BER)* dan *bit rate* sistem. Pelabelan *frame* FDDI tidak mempengaruhi BER karena analisis BER dilakukan pada *transceiver* serat optik. Sedangkan pelabelan *frame* FDDI tidak mempengaruhi *bit rate* sistem karena sudah merupakan karakteristik format data dari FDDI yang menggunakan teknik *encoding digital 4B/5B*.

Kata kunci: GMPLS, FDDI, *station*, *frame*.

PENGANTAR

Alhamdulillah, segala puji dan syukur penulis panjatkan kepada *Allah Subhanahu wata'ala* yang telah melimpahkan rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan skripsi dengan judul **Penerapan Teknologi *Generalized Multi-Protocol Label Switching (GMPLS) Pada Fiber Distributed Data Interface (FDDI)*** yang diajukan untuk memenuhi sebagian persyaratan memperoleh gelar Sarjana Teknik.

Penulis ingin mengucapkan terima kasih kepada pihak-pihak yang telah membantu dan mendukung dalam penyelesaian skripsi ini, yaitu:

1. Ibunda, Ayahanda, Mbah Kung, Bu Lek, Pak De, Pak Lek, dan segenap keluarga tercinta.
2. Bapak Ir. Heru Nurwarsito, M.Kom. selaku Ketua Jurusan Teknik Elektro.
3. Bapak Rudy Yuwono., ST, MSc. selaku Sekretaris Jurusan Teknik Elektro.
4. Bapak Ir. Wahyu Adi Priyono., MSc. Selaku Ketua Kelompok Dosen dan Keahlian Dasar Jurusan Teknik Elektro.
5. Ibu Ir. Endah Budi P., MT selaku Ketua Kelompok Dosen dan Keahlian Teknik Telekomunikasi dan dosen pembimbing skripsi.
6. Bapak Dwi Fadila K., ST., MT selaku dosen pembimbing skripsi.
7. Mas Aris, Mbak Frida, Mbak Heni, segenap staf administrasi Jurusan Teknik Elektro dan Fakultas Teknik.
8. Galih, Aga, Tulus, Oky, Darus, Zaini, semua teman-teman khususnya angkatan 2002.

Penulis menyadari bahwa skripsi ini masih memiliki kekurangan sehingga saran dan kritik membangun sangat penulis harapkan. Akhirnya, semoga skripsi ini bermanfaat untuk kita semua, khususnya Jurusan Teknik Elektro Fakultas Teknik Universitas Brawijaya.

Malang, Agustus 2009

Penulis

DAFTAR ISI

ABSTRAK	i
PENGANTAR	ii
DAFTAR ISI	iii
DAFTAR GAMBAR	vi
DAFTAR TABEL	ix
BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	2
1.3. Ruang Lingkup	2
1.4. Tujuan Penulisan	2
1.5. Sistematika Penulisan	2
BAB II GMPLS (<i>Generalized Multi-protocol Label Switching</i>)	4
2.1. Pendahuluan	4
2.2. MPLS (<i>Multi-protocol Label Switching</i>)	5
2.3. GMPLS (<i>Generalized Multi-protocol Label Switching</i>)	7
2.3.1. Komponen GMPLS	7
2.3.1.1. LSR (<i>Label-Switched Router</i>) dan LER (<i>Label Edge Router</i>)	9
2.3.1.2. FEC (<i>Forward Equivalence Class</i>)	9
2.3.1.3. LSP (<i>Label-Switched Path</i>)	10
2.3.1.4. GMPLS Label	10
2.3.1.5. LDP (<i>Label Distribution Protocol</i>)	11
2.3.2. Suite Protokol GMPLS	12
2.3.3. Switching GMPLS Berdasarkan Format <i>Diverse</i>	14
2.3.4. <i>Bundling Links</i>	16
2.3.5. Fitur Dasar GMPLS	17

2.3.6. Jenis Proteksi <i>Link</i>	18
2.3.7. Perbedaan MPLS dan GMPLS	19
BAB III FDDI (FIBER DISTRIBUTED DATA INTERFACE)	21
3.1. Teknologi FDDI	21
3.2. Media Transmisi FDDI	22
3.2.1. Tipe Kabel Serat Optik	22
3.2.2. Dispersi	23
3.2.3. Redaman Serat Optik	25
3.2.4. Penguat Optik	27
3.2.5. <i>Link Budget</i> Sistem Komunikasi Serat Optik	29
3.2.5.1. <i>Bandwidth Budget</i>	29
3.2.5.2. <i>Link Budget</i>	30
3.3. Spesifikasi FDDI	31
3.4. Arsitektur FDDI	33
3.5. Tipe <i>Station-Attachment</i> FDDI	34
3.6. FDDI <i>Fault-Tolerance</i>	36
3.7. Format <i>Frame</i> FDDI	39
3.8. Mekanisme Kerja Protokol FDDI	40
3.8.1. <i>Dual counter-rotating ring</i>	41
3.8.2. Operasi <i>Ring</i>	41
3.9. FDDI <i>Physical Layer</i>	44
3.9.1. FDDI PMD (<i>Physical layer Medium Dependent Standard</i>)	44
3.9.2. FDDI PHY (<i>Physical Layer Protocol</i>)	45
3.10. FDDI <i>Data Link Layer</i>	49
3.10.1. FDDI MAC	49
3.10.2. LLC (<i>Logical Link Control</i>)	53
3.11. FDDI SMT (<i>Station Management</i>)	53
3.12. Performansi Jaringan	54
3.12.1. <i>Delay</i> jaringan FDDI	54
3.12.2. <i>Bit Rate</i>	60
3.12.3. <i>Throuhgput</i>	60

BAB IV METODOLOGI PENELITIAN	62
4.1. Pengumpulan dan penyusunan data	62
4.2. Pemodelan Sistem	62
4.3. Analisis	64
4.3.1. <i>Flow Chart</i> Analisis	65
4.3.2. <i>Flow Chart</i> Program	66
4.4. Menyimpulkan	82
BAB V ANALISIS PENERAPAN TEKNOLOGI	
GMPLS (<i>Generalized Multi-protocol Label Switching</i>)	
PADA FDDI (<i>Fiber Distributed Data Interface</i>)	83
5.1. Analisis Media Transmisi	83
5.1.1. Analisis <i>Link Budget</i>	83
5.1.2. Analisis <i>Rise-Time Budget</i>	84
5.2. Analisis <i>Bit Error Rate</i> (BER)	86
5.3. Analisis <i>Delay</i>	89
5.3.1. <i>Delay</i> Transmisi	89
5.3.2. <i>Delay</i> Propagasi	90
5.3.3. <i>Delay</i> Paketisasi	91
5.3.4. <i>Delay</i> Enkapsulasi	91
5.3.5. <i>Delay</i> Antrian (<i>Queueing delay</i>)	92
5.3.6. Analisis <i>Ring Latency</i>	94
5.3.7. Analisis Efisiensi akses dan <i>delay</i> akses maksimum	95
5.3.8. <i>Delay</i> Total	96
5.4. Analisis <i>Bit Rate</i>	99
5.5. Analisis <i>Throughput</i>	99
BAB VI PENUTUP	101
6.1. Kesimpulan	101
6.2. Saran	101
DAFTAR PUSTAKA	103
LAMPIRAN	106

DAFTAR GAMBAR

Gambar 2.1	Topologi Jaringan MPLS	6
Gambar 2.2	Mekanisme Pelabelan Paket Data	8
Gambar 2.3	Format Label GMPLS pada FDDI	10
Gambar 2.4	GMPLS <i>Protocol Stack</i>	15
Gambar 2.5	<i>Link Bundling</i>	16
Gambar 2.6	Contoh arsitektur perangkat lunak dari <i>control plane</i> GMPLS	17
Gambar 2.7	Hirarki Empat Jenis Antarmuka yang Didukung oleh GMPLS	18
Gambar 3.1	<i>Counter-rotating</i> pada FDDI. <i>Ring</i> primer dan <i>ring</i> sekunder	21
Gambar 3.2	Serat optik <i>single-mode</i> dan serat optik <i>multi-mode</i>	23
Gambar 3.3	Redaman di sepanjang serat optik	26
Gambar 3.4	<i>Rise-time</i>	30
Gambar 3.5	Standar protokol FDDI dan hubungannya dengan referensi OSI	32
Gambar 3.6	Spesifikasi FDDI dan hubungannya dengan sublapisan LLC	32
Gambar 3.7	Konfigurasi dari DAS	35
Gambar 3.8	Pemulihan <i>ring</i> ketika salah satu <i>station</i> mengalami kerusakan pada metode <i>Dual Ring</i>	36
Gambar 3.9	Pemulihan <i>ring</i> ketika terdapat kabel yang putus/rusak pada metode <i>Dual Ring</i>	37
Gambar 3.10	<i>Optical bypass switch</i> menggunakan cermin internal untuk mempertahankan jaringan	38
Gambar 3.11	Konfigurasi <i>dual-homed</i> untuk <i>device file server</i> dan <i>router</i>	38
Gambar 3.12	FDDI <i>transmission frame</i> dan <i>frame token</i>	39
Gambar 3.13	Operasi <i>counter-rotating ring</i>	41
Gambar 3.14	Sirkulasi <i>frame</i> FDDI dan <i>Token</i>	43

Gambar 3.15	Simbol 4B/5B dan contoh NRZ/NRZI coding	48
Gambar 3.16	MAC entities pada konfigurasi SAS dan DAS	50
Gambar 4.1	Konfigurasi Penerapan GMPLS pada jaringan FDDI	63
Gambar 4.2	Blok Diagram Penerapan GMPLS pada jaringan FDDI	64
Gambar 4.3	Flow Chart Analisis	65
Gambar 4.4	Flow Chart Link Budget	66
Gambar 4.5	Flow Chart Rise-Time Budget	67
Gambar 4.6	Flow Chart delay transmisi (T_r)	68
Gambar 4.7	Flow Chart delay propagasi (T_{Pro})	69
Gambar 4.8	Flow Chart delay paketisasi (T_{Paket})	70
Gambar 4.9	Flow Chart delay enkapsulasi (T_{Enkap})	71
Gambar 4.10	Flow Chart delay antrian (T_{Queue})	72
Gambar 4.11	Flow Chart delay pentransmisian frame (T_{pr})	73
Gambar 4.12	Flow Chart Ring Latency	74
Gambar 4.13	Flow Chart Efisiensi akses	75
Gambar 4.14	Flow Chart Delay akses maksimum	76
Gambar 4.15	Flow Chart delay total	77
Gambar 4.16	Flow Chart Menghitung baud rate D	78
Gambar 4.17	Flow Chart Menghitung bit rate R	79
Gambar 4.18	Flow Chart Menghitung Nilai Konstanta a	80
Gambar 4.19	Flow Chart Menghitung Throughput (ρ_{max})	81
Gambar 5.1	Link budget transceiver	83
Gambar 5.2	Pemasangan repeater untuk Transmisi Data Sejauh 100 km	83
Gambar 5.3	Rise-time Sistem	85
Gambar 5.4	Grafik hubungan P_{in} terhadap BER	88
Gambar 5.5	Grafik hubungan delay antrian terhadap Panjang Paket Data	93
Gambar 5.6	Grafik hubungan Efisiensi Akses terhadap Jumlah Node	94
Gambar 5.7	Grafik hubungan delay petransmisian frame terhadap Panjang Paket Data	95
Gambar 5.8	Grafik hubungan delay total terhadap Panjang Paket Data	97

Gambar 5.9 Grafik hubungan *delay* total terhadap Panjang Paket Data 97

Gambar 5.10 Grafik hubungan *throughput* terhadap Panjang Paket Data 100



DAFTAR TABEL

Tabel 2.1	<i>Framework</i> GMPLS	5
Tabel 2.2	Perbedaan antara <i>routing</i> konvensional dan <i>Label Switching</i>	7
Tabel 2.3	Protokol-protokol yang berhubungan dengan GMPLS	13
Tabel 2.4	Perbedaan MPLS dan GMPLS	19
Tabel 3.1	Keuntungan dan kerugian penggunaan serat optik sebagai Media transmisi	22
Tabel 3.2	Karakteristik Jaringan FDDI	33
Tabel 3.3	Simbol Pengkodean pada <i>first coding stage</i>	47
Tabel 3.4	Perhitungan parameter <i>Valid-Transmission Timer</i>	53
Tabel 5.1	BER untuk spesifikasi ILD	87
Tabel 5.2	Nilai <i>delay</i> transmisi untuk beberapa panjang paket data	89
Tabel 5.3	Nilai <i>delay</i> paketisasi untuk beberapa panjang paket data	90
Tabel 5.4	Nilai <i>delay</i> antrian untuk beberapa panjang paket data Dengan <i>utilitization</i> 10%	92
Tabel 5.5	Nilai <i>delay</i> pentransmisian <i>frame</i> untuk beberapa panjang paket data	95
Tabel 5.6	Nilai <i>delay</i> total untuk beberapa panjang paket data	96
Tabel 5.7	Nilai <i>delay throughput</i> dengan panjang paket data yang berbeda	100

BAB I PENDAHULUAN

1.1 Latar Belakang

Keandalan sistem transmisi data seperti kapasitas kanal yang besar, kecepatan tinggi, dan penerimaan data yang akurat adalah parameter yang digunakan sebagai jaringan *backbone* LAN yang besar pada suatu kawasan industri. Pemanfaatan serat optik pada sistem komunikasi data dapat memenuhi kriteria tersebut. Salah satu sistem *transport* optik adalah FDDI (*Fiber Distributed Data Interface*). FDDI merupakan standar protokol yang memanfaatkan teknologi *dual-ring* yang memiliki performansi tinggi dan mampu beroperasi dengan kecepatan 100 Mbps dalam jarak yang mencapai 100 km atau lebih dan bisa mencapai lebih dari 500 *station* terhubung. FDDI menggunakan protokol *timed-token* sehingga setiap *station* digaransi untuk dapat akses ke jaringan pada periode tertentu yang diatur pada saat jaringan mulai beroperasi. Tidak seperti akses protokol *token* umumnya yang didefinisikan oleh standar IEEE 802.5, akses protokol *timed-token* yang digunakan pada FDDI dapat melewati aliran data *synchronous* dan *asynchronous* secara bersamaan.

Penerapan GMPLS pada FDDI dapat memberikan kecepatan pengiriman data dengan tetap menjaga kualitas data yang dikirim. GMPLS merupakan konvergensi vertikal, Karena menggunakan metode *label switching* dalam *layer* 1 hingga 3. Tujuannya adalah untuk menyediakan jaringan yang secara keseluruhan mampu menangani *bandwidth* besar dengan QoS (*quality of service*) yang konsisten serta pengendalian penuh dan terintegrasi.

GMPLS (*Generalized Multi-protocol Label Switching*) merupakan evolusi dari MPLS (*Multi-protocol Label Switching*) - standar awal dari IETF (*International Engineering Task Force*) untuk meningkatkan performansi penghantaran dan rekayasa trafik dari jaringan berbasis paket (ATM, IP). GMPLS tidak hanya mendukung jaringan berbasis paket (*Packet Switch Capable*, PSC) tapi juga TDM (*Time Division Multiplex*), FSC (*Fiber Switch Capable*), dan LSC (*Lambda Switch Capable*). GMPLS dapat didukung oleh semua sistem ber-*platform* optik diantaranya SONET (*Synchronous*

Optical Network), ADMs (*Add-drop Multiplexers*), OXCs (*Optical Cross-connects*), dan DWDM (*Dense Wavelength-Division Multiplexing*).

1.2 Rumusan Masalah

Mengacu pada permasalahan yang ada, maka rumusan masalah penulisan skripsi ini ditekankan pada:

1. Bagaimana mekanisme kerja sistem GMPLS dan jaringan FDDI.
2. Bagaimana konfigurasi penerapan GMPLS pada jaringan FDDI.
3. Bagaimana pengaruh penerapan GMPLS pada jaringan FDDI terhadap kinerja sistem.

1.3 Ruang Lingkup

Analisis kajian penerapan GMPLS pada jaringan FDDI pada penulisan skripsi ini ditinjau dari aspek:

1. Membahas mekanisme komunikasi data pada GMPLS dan jaringan FDDI.
2. Membahas media transmisi serat optik yang digunakan pada jaringan FDDI.
3. Hanya membahas performansi jaringan FDDI tanpa meliputi perangkat lunak yang digunakan.
4. Hanya membahas trafik pada GMPLS.

1.4 Tujuan Penulisan

Tujuan penulisan skripsi ini adalah untuk menganalisis kinerja dari penerapan GMPLS pada jaringan FDDI.

1.5 Sistematika Penulisan

Sistematika penulisan yang digunakan dalam penyusunan skripsi ini adalah:

- BAB I Pendahuluan yang memuat latar belakang, tujuan, rumusan masalah, batasan masalah, dan sistematika penulisan.
- BAB II GMPLS, merupakan studi pustaka mengenai GMPLS yang difokuskan pada penjelasan tentang definisi, arsitektur jaringan, dan konsep pelayanannya.
- BAB III FDDI, merupakan studi pustaka yang akan membahas definisi, konsep FDDI, dan implementasi FDDI pada komunikasi data.

BAB IV Metodologi Penelitian

Membahas mengenai metodologi penelitian yang digunakan dalam skripsi ini.

BAB V Analisis penerapan teknologi GMPLS pada jaringan FDDI.

Membahas tentang analisis kinerja dan penerapan teknologi GMPLS pada FDDI.

BAB VI Penutup, memuat kesimpulan dan saran dari topik yang dianalisis dari skripsi ini.



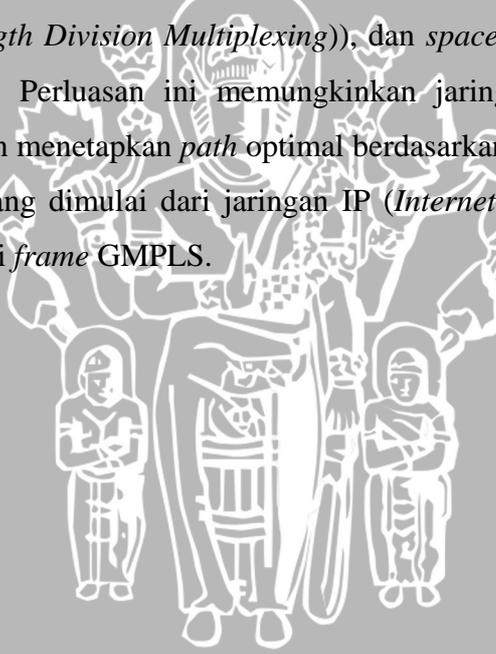
BAB II

GMPLS (*Generalized Multi-protocol Label Switching*)

2.1 Pendahuluan

GMPLS merupakan pengembangan dari MPLS untuk memberikan *control plane* (*routing* dan pensinyalan) untuk *device-device* dalam *domain* Paket (*packet*), Waktu (*time*), *Wavelength*, dan *Fiber*.

Untuk memberikan *control plane* tersebut, GMPLS disesuaikan untuk beroperasi melalui jenis jaringan yang berbeda seperti *packet*, jaringan *plesiochronous* dan jaringan optik. GMPLS adalah perluasan protokol MPLS yang di dalamnya mendukung *device-device* yang dikoneksikan dalam *domain* waktu (*time*), *wavelength* (DWDM (*Dense Wavelength Division Multiplexing*)), dan *space domain* (seperti OXC (*Optical Cross-Connect*)). Perluasan ini memungkinkan jaringan yang berbasiskan GMPLS untuk mencari dan menetapkan *path* optimal berdasarkan kebutuhan trafik *user* untuk melewati data yang dimulai dari jaringan IP (*Internet Protocol*). Tabel 2.1 merupakan rangkuman dari *frame* GMPLS.



Tabel 2.1 Framework GMPLS

<i>Switching Domain</i>	<i>Jenis Trafik</i>	<i>Skema Forwarding</i>	<i>Contoh device</i>	<i>Nama Sistem (Nomenclatur)</i>
Paket, cell	IP, ATM (Asynchronous Transfer Mode)	Label as shim header, VCC (virtual channel connection)	router IP, switch ATM	Packet switch capable (PSC)
Time	TDM/SONET	Time slot dalam repeating cycle	DCS (Digital-cross connect system), ADM (Add-drop Multiplexer)	TDM (Time Division Multiplexing) capable
Wavelength	Transparent	Lambda	DWDM (Dense Wavelength Division Multiplexing)	LSC (Lambda switch capable)
Physical space	Transparent	Fiber, line	OXC (Optical Cross-Connect)	FSC (Fiber switch capable)

Sumber:

<http://www.polarisnetworks.com/images/PDF/GMPLSWhitePaperComplete.pdf>

Dalam GMPLS, IP routers, ATM switches, Frame Relay switches, Ethernet switches, DCSs, dan OXCs semuanya disuguhkan sebagai jaringan IP tunggal dari control point of view. Hal ini berarti tidak memerlukan adanya UNIs (User Network Interfaces) dan NNIs (Network Node Interfaces), karena GMPLS adalah protokol point-to-point.

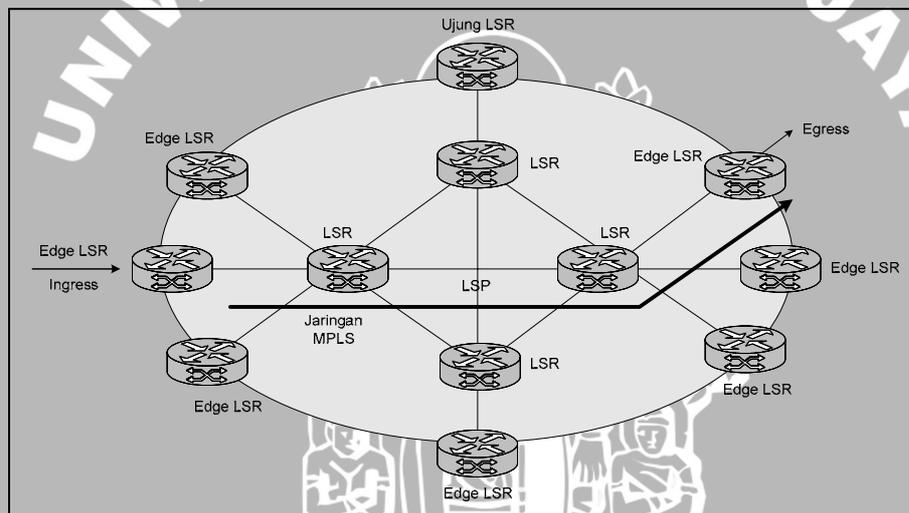
GMPLS adalah sebuah arsitektur dan implementasinya memerlukan sebuah protokol pensinyalan. RSVP-TE (Resource Reservation Protocol-Traffic Engineering) dan CR-LDP (Constraint Routing-Label Distribution Protocol) keduanya telah diperluas untuk mendukung GMPLS.

2.2 MPLS (Multi-protocol Label Switching)

MPLS adalah peningkatan metode untuk penghantaran paket data melalui jaringan menggunakan informasi yang ada dalam label yang dimasukkan ke dalam

paket IP. MPLS mengkombinasikan teknologi *switching* layer 2 dengan teknologi *routing* layer 3. Tujuan utama dari MPLS adalah untuk menciptakan struktur jaringan yang fleksibel yang menyediakan peningkatan performa dan stabilitas. Peningkatan ini meliputi kemampuan VPN (*virtual private network*) dan rekayasa jaringan (*traffic engineering*) yang menawarkan QoS dengan banyak CoS (*Classes of Service*).

Dalam jaringan MPLS, paket datang/masuk diberi label oleh *Edge Label-Switched Router*. Paket-paket diteruskan sepanjang LSP (*Label-Switched Path*) dimana tiap LSR (*Label-Switched Router*) memutuskan kemana paket diteruskan hanya berdasarkan isi dari label. Pada tiap *hop*, LSR melepaskan label yang ada dan memasang label yang baru yang mendefinisikan kemana paket akan diteruskan selanjutnya.



Gambar 2.1 Topologi Jaringan MPLS

Sumber: Vivek Alwayn, 2002:4

LSP (*Label-switched Path*) adalah sirkuit yang menghubungkan titik-titik yang disebut LSR (*Label-switched Router*). LSR pertama dan terakhir disebut *ingress* dan *egress*. Setiap LSP dikaitkan dengan sebuah FEC (*Forwarding Equivalence Class*) yang merupakan kumpulan paket yang menerima perlakuan penghantaran paket yang sama di sebuah LSR. FEC diidentifikasi dengan label.

Untuk membentuk LSP, diperlukan suatu protokol pensinyalan. Protokol ini menentukan penghantaran paket berdasarkan label pada paket. Label yang pendek dan berukuran tetap mempercepat proses penghantaran paket dan mempertinggi fleksibilitas

pemilihan *path*. Hasilnya adalah *network datagram* yang bersifat lebih *connection-oriented*.

Berikut ini diperlihatkan perbedaan antara *Label Switching* yang digunakan pada MPLS dengan *routing* konvensional yang digunakan pada IP.

Tabel 2.2 Perbedaan antara *routing* konvensional dan *Label Switching*

	Routing Konvensional	Label Switching
Analisis seluruh <i>header IP</i>	Dilakukan pada masing-masing <i>hop</i> dari jalur paket dalam jaringan	Dilakukan hanya sekali saat akan memasuki jalur paket dalam jaringan
Dukungan untuk <i>unicast</i> dan <i>multicast</i>	Membutuhkan <i>multicast routing</i> dan algoritma penghantaran paket khusus	Hanya membutuhkan satu algoritma penghantaran paket
Pengambilan keputusan <i>routing</i>	Berdasarkan alamat tujuan pada <i>header IP</i>	Berdasarkan jumlah parameter, termasuk alamat tujuan di dalam <i>IP header</i> seperti QoS, tipe data (<i>voice</i> , <i>d11</i>)

Sumber: Marcony White Paper, 2000:16

2.3 GMPLS (*Generalized Multi-protocol Label Switching*)

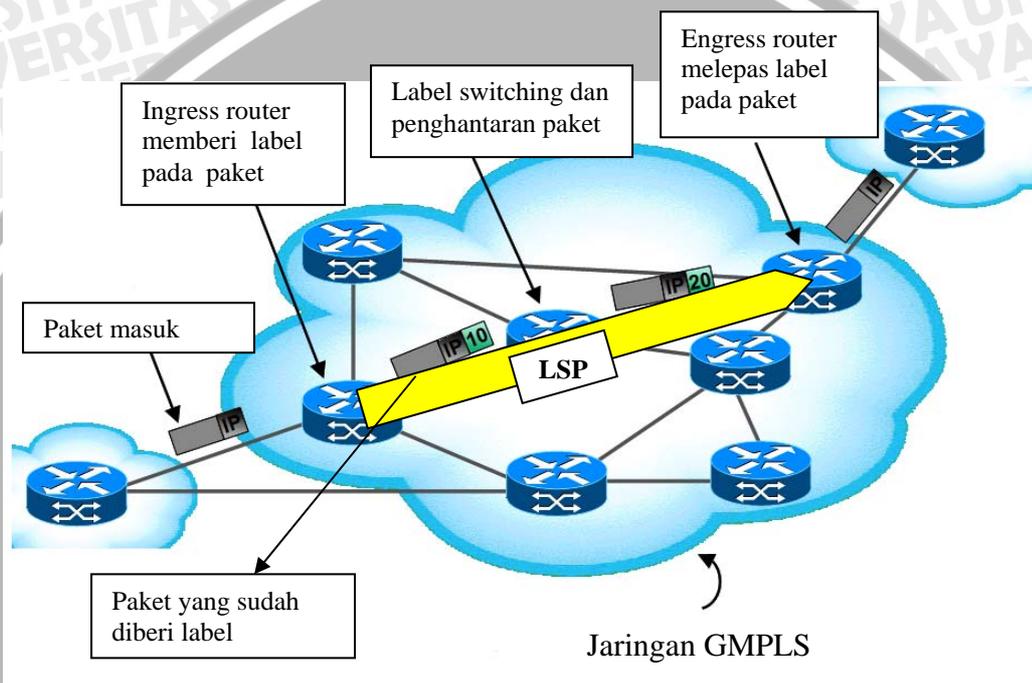
2.3.1 Komponen GMPLS

Komponen utama dalam GMPLS adalah fungsi pelabelan yang menggantikan alamat IP konvensional dengan algoritma yang lebih efisien.

- **LSP (*Label-Switched Path*):** Merupakan jalur yang melalui satu atau serangkaian LSR dimana paket diteruskan oleh *label swapping* dari satu GMPLS *node* ke GMPLS *node* yang lain.
- **LSR (*Label-Switched Router*):** GMPLS *node* yang mampu meneruskan paket-paket *layer 3*.
- **GMPLS Edge Node** atau **LER (*Label Edge Router*):** GMPLS *node* yang menghubungkan sebuah GMPLS *domain* dengan *node* yang berada di luar GMPLS *domain*
- **GMPLS Egress Node:** GMPLS *node* yang mengatur trafik saat meninggalkan GMPLS *domain*

- **GMPLS Ingress Node:** GMPLS *node* yang mengatur trafik saat akan memasuki MPLS *domain*
- **GMPLS label:** merupakan label yang ditempatkan sebagai GMPLS *header*
- **GMPLS node:** *node* yang menjalankan GMPLS. *Node* GMPLS ini sebagai protokol kendali yang akan meneruskan paket berdasarkan label.

Mekanisme kerja GMPLS dalam jaringan ditunjukkan seperti gambar berikut:



Gambar 2.2 Mekanisme Pelabelan Paket Data

Sumber: Grotto Networking, 2004

- Paket yang masuk merupakan paket yang datang dari jaringan di luar jaringan GMPLS. Jaringan tersebut bisa merupakan jaringan GMPLS yang lain atau jaringan yang bukan merupakan jaringan GMPLS, seperti FDDI.
- Jaringan secara otomatis membangun tabel *routing* yang berkaitan dengan penyedia jaringan dan proses ini dibangun oleh LSP. Proses ini diawali dengan identifikasi pencatatan label yang dilakukan dengan merutekan protokol *routing* seperti OSPF (*Open Shortest Path First*), RIP (*Routing Information Protocol*) atau BGP (*Border Gateway Protocol*).
- Paket harus dipindah dari *layer 2* ke *layer GMPLS* dengan cara menambahkan sebuah label ke *header* paket tersebut.

- d. Paket yang telah memiliki label harus bisa melewati jalur GMPLS. Jalur ini terdiri dari semua peralatan yang mengetahui bagaimana paket harus diteruskan ke dalam jaringan. Jalur ini disebut dengan LSP. Jalur ini merupakan jalur *connection-oriented* yang telah dibuat sebelum terjadi pengiriman paket.
- e. Paket harus memiliki cara untuk kembali ke *layer 2*. Proses ini meliputi proses pembuangan label dari *header* paket dan kemudian mengirimkan paket tersebut kembali ke *layer 2* untuk proses lebih lanjut.

Dalam GMPLS, *header* IP hanya dianalisa ketika memasuki LSP untuk mengklasifikasikan paket data. Paket data yang akan dikirim melalui *hop* yang sama akan dikelompokkan menjadi FEC yang kemudian dikodekan menjadi suatu nilai dengan panjang tertentu yang disebut dengan "label". Setelah paket mempunyai label, paket akan dirutekan berdasarkan label yang dimilikinya. Untuk mendistribusikan label dalam jaringan digunakan protokol pensinyalan yang disebut LDP (*Label Distribution Protocol*).

2.3.1.1 LSR (*Label-Switched Router*) dan LER (*Label Edge Router*)

LSR merupakan suatu alat yang membentuk LSP dengan menggunakan protokol pensinyalan label yang tepat dan melakukan *switching* data trafik berdasarkan jalur yang telah dibangun.

LER merupakan alat yang beroperasi diantara jaringan *access* dan jaringan MPLS. LER mampu mendukung *multiple port* yang terhubung dengan jaringan yang berbeda-beda (misalnya Frame Relay, ATM, dan *Ethernet*) dan meneruskan trafik tersebut ke jaringan MPLS setelah membangun LSP.

Setiap LSR memiliki tabel yang disebut *label-switching table*. Tabel itu berisi pemetaan label masuk, label keluar, dan *link* ke LSR berikutnya. Saat LSR menerima paket, label paket akan dibaca, kemudian diganti dengan label keluar, lalu paket dikirimkan ke LSR berikutnya.

2.3.1.2 FEC (*Forward Equivalence Class*)

FEC merupakan sekelompok paket yang membutuhkan penanganan yang sama untuk proses pengiriman. Semua paket yang berada pada kelompok yang sama akan memperoleh penanganan yang sama dan dirutekan ke tujuannya.

Pengklasifikasian FEC didasarkan pada beberapa parameter antara lain: *Address prefix*, *Host address*, dan *QoS*. Pada jaringan GMPLS, pengelompokan paket pada FEC hanya dilakukan sekali yaitu ketika paket memasuki jaringan GMPLS. Nilai FEC yang dimiliki oleh sebuah paket akan dikodekan sebagai nilai dengan panjang yang tetap yang disebut dengan "label".

2.3.1.3 LSP (*Label-Switched Path*)

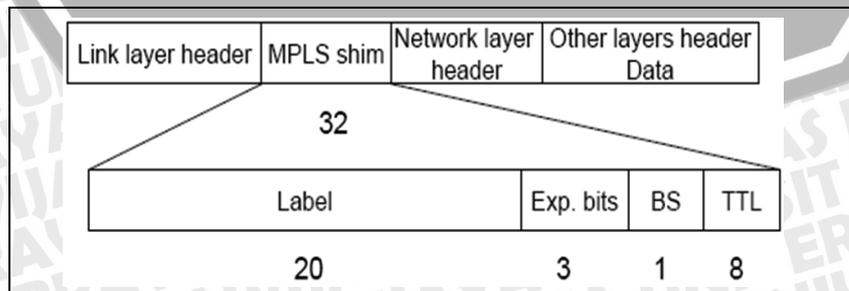
Dalam jaringan GMPLS, suatu lintasan dibangun dengan tujuan untuk mengirimkan paket ke tujuan yang didasarkan pada FEC yang dimilikinya. Sebelumnya dibangun terlebih dahulu suatu LSP untuk mengirimkan data. GMPLS memiliki dua mekanisme untuk membangun LSP yaitu:

- a. *Hop-by-hop routing*, dimana tiap-tiap LSP secara terpisah memilih *hop* yang selanjutnya akan diberi FEC
- b. *Explicit routing*, dimana LER akan merinci daftar *node-node* yang akan dilalui oleh data.

2.3.1.4 GMPLS Label

GMPLS *Label* memiliki panjang 32 bit yang digunakan untuk mengidentifikasi FEC. Tidak seperti ATM yang memecah paket-paket IP, GMPLS hanya melakukan enkapsulasi paket IP. Label adalah bagian dari *header*, memiliki panjang yang bersifat tetap, dan merupakan satu-satunya tanda identifikasi paket. Label digunakan untuk proses pengantaran paket termasuk proses rekayasa trafik.

Label, yang disematkan pada paket khusus, merepresentasikan tujuan FEC. Dalam kasus FDDI, label diletakkan pada *shim header*. *Shim label header* disisipkan antara *data link layer* dan *network layer*.



Gambar 2.3 Format Label GMPLS pada FDDI

Sumber: <http://www.bilmuh.gyte.edu.tr/BIL674/MPLS.ppt>

- Label : membawa data dalam bentuk MPLS label
- Exp. Bit : Experimental Bits, sering digunakan sebagai CoS. CoS cenderung membuat antrian dan mengabaikan algoritma yang ditetapkan pada paket yang ditransmisikan melalui jaringan
- BS : Bottom of Stack bit, di-*set* jika tidak ada label yang mengikutinya
- TTL (*Time to Live*) : menyediakan fungsi konvensional TTL IP

Selain paket IP, paket GMPLS juga bisa dienkapsulasikan kembali dalam paket MPLS. Maka sebuah paket bisa memiliki beberapa *header*. Dan bit *stack* pada *header* menunjukkan apakah suatu *header* sudah terletak di 'dasar' tumpukan *header* GMPLS itu.

2.3.1.5 LDP (*Label Distribution Protocol*)

LDP merupakan sekelompok prosedur dan *message* dimana LSR-LSR membangun LSP-LSP pada suatu jaringan dengan memetakan informasi *routing network layer* pada jalur *switch data link layer*. LDP digunakan untuk memetakan FEC ke label-label suatu LSP. LSP-LSP akan diperbesar dalam suatu jaringan ketika LSR menyambung label *incoming* yang dimiliki oleh sebuah FEC ke label *outgoing* pada FEC tersebut. *Message* yang saling dipertukarkan antara LSR-LSR mengikuti prosedur sebagai berikut:

- Discovery message* digunakan untuk memberitahukan dan memelihara keberadaan sebuah LSR dalam suatu jaringan
- Session message* digunakan untuk menciptakan, memelihara, dan menghapus *session* diantara pasangan LDP
- Advertisement message* digunakan untuk membuat, merubah, dan menghapus pemetaan label untuk FEC
- Notification messages* digunakan untuk memberikan informasi tambahan dan untuk memberitahukan informasi kesalahan.

Discovery message memberikan suatu mekanisme dimana LSR dapat mengindikasikan keberadaan mereka dalam suatu jaringan dengan cara mengirimkan *message* "HELLO" secara berkala. *Message* ini ditransmisikan sebagai sebuah paket UDP (*User Datagram Protocol*) ke *port* LDP ke alamat *multicast group*. Ketika sebuah LSR memilih untuk membuat sebuah *session* dengan LSR yang lain melalui *message* Hello, proses ini menggunakan prosedur inisialisasi LDP melalui *transport* TCP.

Setelah prosedur inisialisasi telah dilakukan secara utuh, kedua LSR tersebut menjadi pasangan LDP dan dimungkinkan untuk saling tukar *advertisement message*. LSR meminta sebuah pemetaan label dari LSR tetangganya dan melakukan pemetaan label ke LSR tetangga ketika LSR tersebut menginginkan tetangganya menggunakan label.

2.3.2 Suite Protokol GMPLS

Label yang dibentuk dalam GMPLS dibentuk untuk melakukan ekstensi dari representasi *single 32 bit number* menjadi *arbitrary length byte array* yaitu pada RSVP dalam bentuk *Generalized Label object* dan pada CR-LDP berupa *Generalized Label TLV (type length value)* dimana GMPLS membawa label dan informasi yang berhubungan untuk dikirim dari satu *node* ke *node* yang lain.

Karena pentingnya aspek kontrol dalam jaringan *core* GMPLS, sebuah *control plane* yang terpisah diterapkan pada protokol tersebut. Dengan adanya *control plane* ini maka GMPLS dapat mengontrol secara penuh jaringan optik di bawahnya. *Control plane* dalam GMPLS menyediakan 6 fungsi *top-level*, yaitu (<http://telkomcorner.blogspot.com/2008/12/gmpls-dalam-skema-konvergensi-33.html>):

1. *Neighbour discovery* – Untuk dapat mengatur jaringan maka setiap perangkat yang terhubung ke jaringan harus dikenali terlebih dahulu. Perangkat-perangkat tersebut antara lain *switch, router, multiplexer (ADM), dan OXC*.
2. *Dissemination of link status* – Selain mengetahui *hardware* apa saja yang terhubung ke jaringan, *link* yang menghubungkan perangkat-perangkat tersebut juga harus diketahui statusnya apakah *link* tersebut berjalan dengan normal atau *down*. Untuk mendapatkan informasi status *link* ini, suatu protokol *routing* harus digunakan. Dalam GMPLS, kedua protokol *routing* OSPF dan IS-IS dimodifikasi untuk keperluan ini.
3. *Topology state management* – Protokol-protokol yang mendeteksi status *link* seperti OSPF dan IS-IS dapat dipergunakan untuk mengontrol dan mengatur topologi jaringan.
4. *Path management* – MPLS menggunakan antara lain protokol RSVP untuk membangun *link end-to-end*.
5. *Link management* – Dalam MPLS, LSP digunakan untuk membangun dan membebaskan *link* dan *link-link* agregat. Dalam GMPLS dibutuhkan kemampuan untuk membangun dan membebaskan kanal-kanal optik agregat.

LMP (*Link Management Protocol*) mengembangkan fungsi-fungsi MPLS ke dalam suatu *optical plane* dimana pembangunan *link* akan meningkatkan skalabilitas.

6. *Protection and recovery* – Dengan MPLS, sebagai ganti satu *ring* dengan satu *ring backup* untuk proteksi, jaringan dapat membuat suatu struktur *mesh* yang memungkinkan terciptanya beberapa *path* yang berbeda-beda.

Satu peningkatan GMPLS dari MPLS adalah pada perluasan protokol pensinyalan (RSVP-TE, CR-LDP) dan pada protokol *routing* (OSPF-TE, IS-IS-TE). Perluasan ini membuat GMPLS kompatibel dengan jaringan *plesiochronous/synchronous* dan jaringan optik. LMP adalah protokol baru yang dikembangkan untuk dapat mendukung kedua jaringan tersebut secara bersamaan. Protokol ini mengatur dan memelihara kelangsungan *control plane* dan *data plane* antara dua *node* yang berdekatan. LMP merupakan pengembangan dari IP (*IP-based*) dan mengandung perluasan untuk RSVP-TE dan CR-LDP.

Tabel 2.3 Protokol-protokol yang berhubungan dengan GMPLS

Protokol	Penjelasan
<i>Routing</i>	OSPF-TE, IS-IS-TE Protokol <i>routing</i> untuk menentukan topologi jaringan secara otomatis, memberi tahu kondisi <i>resource</i> (jenis <i>bandwidth</i> atau proteksi). Peningkatan utama antara lain: <ol style="list-style-type: none"> 1. memberi tahu jenis proteksi saluran (1+1, 1:1, <i>unprotected</i>, trafik ekstra) 2. mencari saluran untuk menambah skalabilitas 3. menerima dan mencari saluran dengan tanpa alamat IP – saluran ID 4. <i>incoming</i> dan <i>outgoing</i> antarmuka ID 5. rute untuk <i>back-up</i> yang berbeda dari <i>primary path</i>
Pensinyalan	RSVP-TE, CR-LDP Protokol pensinyalan untuk pembentukan LSP dari rekayasa trafik. Peningkatan utama antara lain: <ol style="list-style-type: none"> 1. penukaran label untuk mendukung jaringan non-paket (<i>label generalized</i>)



Link Management

LMP

2. pembentukan LSP *bidirectional*
3. pensinyalan untuk pembentukan *path back-up* (proteksi informasi)
4. mempercepat pemberian label melalui label yang disarankan
5. mendukung *switching waveband* – menempatkan panjang gelombang yang bersebelahan yang dikoneksikan bersamaan

1. **Control-Channel Management**

Dibentuk oleh negosiasi parameter saluran (contoh: mengirim pesan hidup berulang-ulang) dan menjamin ketersediaan saluran (protokol *hello*)

2. **Link-Connectivity Verification**

Menjamin koneksi fisik saluran antara 2 *node* terdekat menggunakan PING seperti tes pesan

3. **Link-Property Correlation**

Identifikasi properti saluran dari *node-node* yang berdekatan (contoh: mekanisme proteksi)

4. **Fault Isolation**

Memisahkan kesalahan tunggal atau jamak dalam daerah optik

Sumber:

<http://www.polarisnetworks.com/images/PDF/GMPLSWhitePaperComplete.pdf>

2.3.3 Switching GMPLS Berdasarkan Format *Diverse*

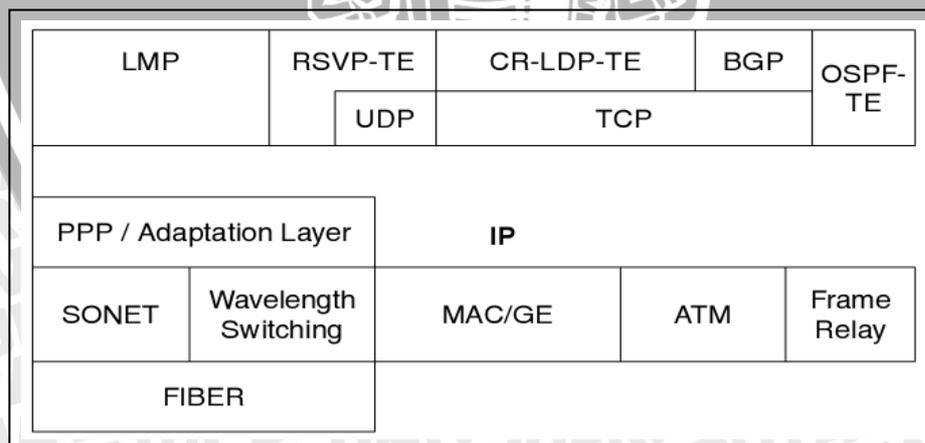
Dalam GMPLS terdapat penambahan pada format labelnya agar dapat mendukung *device* jaringan yang dikoneksikan dalam *domain* yang berbeda. Arsitektur baru dari GMPLS ini disebut sebagai *generalized label*. Arsitektur baru tersebut dapat berisi informasi yang memungkinkan *device* penerima untuk memprogram *switch* yang sesuai dan menghantarkan *frame* atau paket data tanpa memperhatikan strukturnya.

Struktur tersebut mungkin di-pack dalam DS1E1, *lambda*, dan sebagainya. *Generalized label* tersebut mungkin merepresentasikan panjang gelombang tunggal, *time slot*, atau serat optik.

Informasi yang terdapat dalam *generalized label* diantaranya:

1. Jenis encoding LSP yang mengindikasikan jenis label yang sedang dibawa (contoh: *lambda*, paket, SONET).
2. Jenis *switching* yang mengindikasikan apakah sebuah *node* dapat dikoneksikan dengan paket, *time slot*, serat optik, atau *wavelength*.
3. Sebuah *general payload identifier* untuk mengindikasikan apakah *payload* sedang dibawa oleh LSP (contoh: *virtual tributary* (VT), DS-3, ATM, *Ethernet*).

GMPLS memiliki persamaan dengan pendahulunya, MPLS, dimana distribusi label dimulai dengan meminta label *upstream* LSR dari *downstream* LSR. GMPLS mengijinkan *upstream* LSR untuk menyarankan sebuah label untuk LSP yang dapat dikesampingkan oleh *downstream* LSR. LSP yang sesuai juga dapat menyarankan label yang dimilikinya. Proses pemilihan label ini adalah krusial bagi sistem yang harus mengkonfigurasi *switch* serat optiknya, yang memerlukan waktu yang tidak sedikit. Contohnya adalah DCS yang mempunyai ketepatan *switching* tinggi, dijumpai pada DS-1 atau DS-3 (E-1 atau E-3) yang mempunyai *port* dalam jumlah sampai ribuan. Perlu diingat bahwa label dalam kasus ini digunakan untuk menentukan *path* internal antara *input port* dengan *output port* secara cepat.



Gambar 2.4 GMPLS Protocol Stack

Sumber:

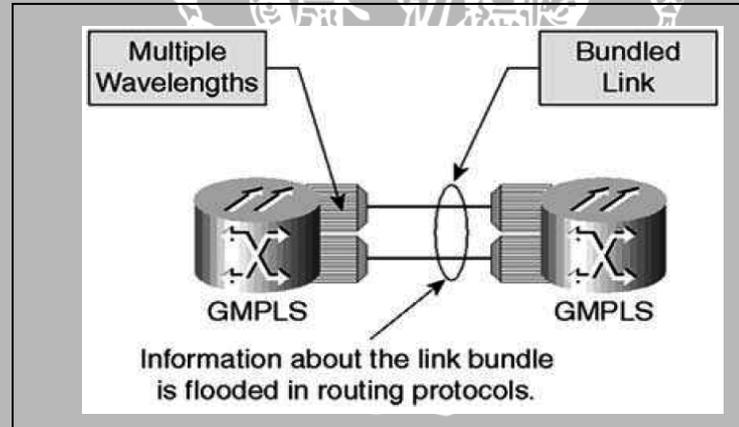
<http://www.polarisnetworks.com/images/PDF/GMPLSWhitePaperComplete.pdf>

Daripada harus menunggu menerima label dari *downstream node* dan menyelesaikan konfigurasi *hardware*, label yang diberikan digunakan mengurangi *delay* proses dan waktu *setup*. Akan tetapi, yang terjadi *downstream device* menolak label yang disarankan dan menawarkan label yang dimilikinya. Dalam kasus ini *upstream device* diperlukan untuk merekonfigurasi kembali dirinya sendiri dengan label yang baru.

2.3.4 Bundling Links

Link bundling dalam GMPLS memungkinkan beberapa *link* rekayasa trafik dikelompokkan menjadi satu *link* rekayasa trafik yang lebih besar. *Subset* beberapa *link* tersebut disebut sebagai *component link*, dan kumpulan dari beberapa *link* tersebut disebut sebagai *bundling link*. *Bundling link* hanya memerlukan satu *channel* kendali.

Link bundling meningkatkan proses *routing* dengan mengurangi jumlah *link* dan atribut *routing* yang dilewatkan ke dalam protokol *routing* seperti OSPF dan IS-IS. *Link bundling* memungkinkan beberapa *link* parallel yang mempunyai karakteristik yang sama bersama-sama dilewatkan dalam sebuah *link* tunggal.



Gambar 2.5 *Link Bundling*

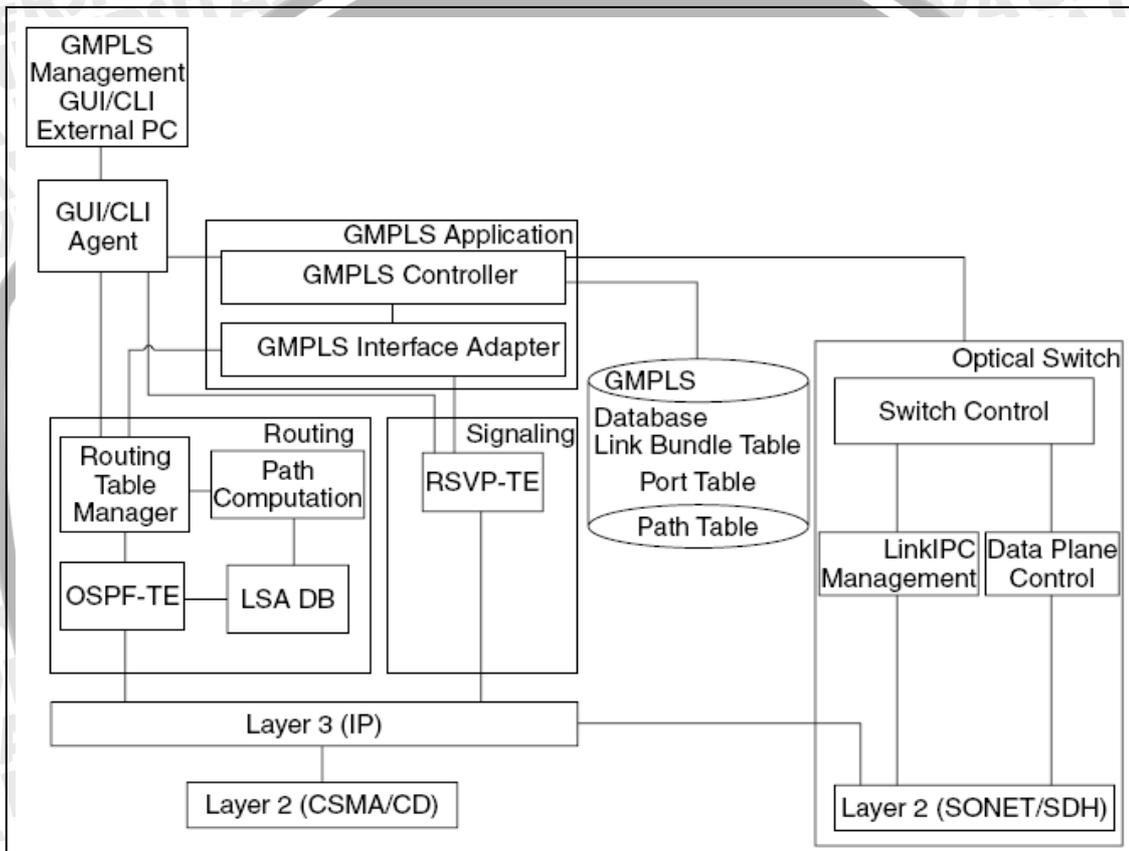
Sumber: Sam Halabi, 2003

Kriteria *link* yang dapat di-*bundle* pada GMPLS antara lain:

- Semua *link* yang termasuk dalam *bundled link* harus mempunyai tipe *link* yang sama seperti *point-to-point* atau *multicast*
- Semua *link* yang termasuk dalam *bundled link* harus mulai dan berakhir pada sepasang LSRs yang sama

- c. Semua *link* yang termasuk dalam *bundled link* harus mempunyai *traffic metric* yang sama (contoh: tipe proteksi atau *bit rate*)
- d. Semua *link* yang termasuk dalam *bundled link* harus mempunyai kapabilitas *switching* yang sama – PSC, TDM, LSC, atau FSC.

Bundled link akan tetap ada jika *component link* ada. Melihat ada tidaknya *component link* dapat dilakukan melalui protokol *routing* dan LMP. Sekali *bundled link* dianggap ada, informasi mengenai *bundled link* dilewatkan sebagai *link* rekayasa trafik.



Gambar 2.6 Contoh arsitektur perangkat lunak dari *control plane* GMPLS

Sumber:

<http://www.polarisnetworks.com/images/PDF/GMPLSWhitePaperComplete.pdf>

2.3.5 Fitur Dasar GMPLS

Beberapa antarmuka yang didukung oleh GMPLS antara lain:

- a. Antarmuka PSC (*Packet-switch Capable*)

Merupakan antarmuka yang berbeda yang digunakan untuk mengirim dan menerima paket, seperti paket IP, ATM *cells*, *frame* Frame Relay, dan *frame*



Ethernet. Paket-paket tersebut dihantarkan berdasarkan enkapsulasi label, VPI/VCI *field* dari *header ATM cells*, atau DLCI *field* dari *frame* Frame Relay.

b. Antarmuka *Time-division multiplex capable*

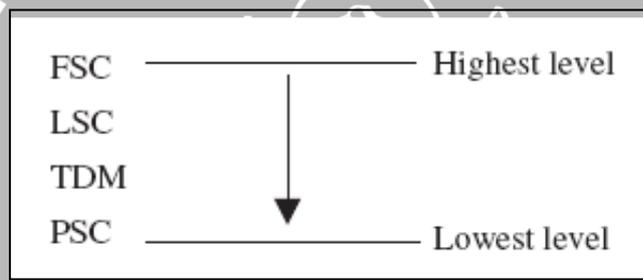
Menghantarkan data berdasarkan slot data dengan sebuah *frame*. Antarmuka ini digunakan pada SONET/SDH DCS.

c. Antarmuka LSC (*Lambda switch Capable*)

Menghantarkan data dari *incoming wavelength* ke *outgoing wavelength*. Antarmuka ini digunakan pada OXC.

d. Antarmuka FSC (*Fiber-switch Capable*)

Menghantarkan data dari satu atau lebih *incoming fibers* ke satu atau lebih *outgoing fibers*. Antarmuka ini digunakan pada OXC yang beroperasi pada satu atau lebih *fibers*.



Gambar 2.7 Hirarki Empat Jenis Antarmuka yang Didukung Oleh GMPLS

Sumber: Harry G. Perros, 2005:221

Urutan antarmuka ini digunakan oleh GMPLS untuk mendukung hirarki LSPs. Dengan menganggap LSP mulai dan berakhir pada sebuah antarmuka *packet-switching*. LSP ini dapat melewati beberapa tipe jaringan, dimana LSP tersebut dapat dikumpulkan dengan LSP-LSP lainnya ke dalam tingkatan LSP yang lebih tinggi. LSP yang lebih tinggi ini dapat mulai dan berakhir pada antarmuka *packet-switching*, antarmuka *time-division*, antarmuka *lambda switch*, atau antarmuka *fiber-switch*.

2.3.6 Jenis Proteksi *Link*

Dalam GMPLS diperkenalkan konsep **jenis proteksi link**, yang mengindikasikan kemampuan proteksi pada *link* yang eksis. Algoritma komputasi *path* menggunakan informasi ini untuk menetapkan karakteristik proteksi yang sesuai dengan *link*. Proteksi *link* tersebut diantaranya:

- a. *Extra Traffic* – *link* jenis ini memproteksi *link* lainnya. Jika terjadi kegagalan maka semua LSPs pada *link* tersebut akan hilang
- b. *Unprotected* – *link* jenis ini secara sederhana tidak diproteksi oleh *link* manapun. Jika *link* yang tidak diproteksi ini mengalami kegagalan maka semua LSPs pada *link* tersebut akan hilang
- c. *Shared* – *link* jenis ini diproteksi oleh satu atau lebih *link* yang terpisah dari *Extra Traffic*
- d. *Dedicated 1:1* – *link* jenis ini diproteksi oleh satu *link* yang terpisah dari *Extra Traffic*
- e. *Dedicated 1+1* – *link* jenis ini diproteksi oleh satu *link* yang terpisah dari *Extra Traffic*. Akan tetapi, *link* proteksi tidak diberitahukan dalam *link-state database* sehingga tidak digunakan pada *routing* LSPs
- f. *Enhanced* – *link* jenis ini mengindikasikan bahwa skema proteksinya lebih dapat dipercaya dibandingkan *dedicated 1+1*.

2.3.7 Perbedaan MPLS dan GMPLS

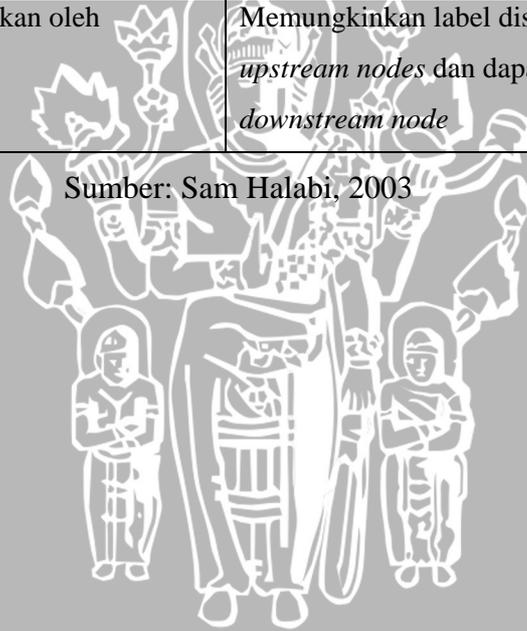
GMPLS memperluas MPLS untuk mendukung non-paket/*cell*. Dukungan tambahan terhadap antarmuka TDM, *lambda*, dan *fiber* ini berdampak pada properti dasar LSP, seperti bagaimana label dibutuhkan dan berkomunikasi dan karakteristik *unidirectional* LSP dan kesalahan propagasi.

Tabel 2.4 Perbedaan MPLS dan GMPLS

MPLS	GMPLS
Hanya mendukung antarmuka pengiriman data berbasis paket/ <i>cell</i>	Mendukung antarmuka pengiriman data berbasis paket/ <i>cell</i> , TDM, <i>lambda</i> , dan <i>fiber</i>
LSPs mulai dan berakhir pada paket/ <i>cell</i> LSRs	LSPs mulai dan berakhir pada jenis LSRs yang sama (yaitu, PSC, L2SC, TDM, LSC, FSC)
Alokasi <i>bandwidth</i> dapat dilakukan pada sejumlah unit	Alokasi <i>bandwidth</i> hanya dapat dilakukan pada unit diskrit untuk beberapa kemampuan <i>switching</i> seperti TDM, LSC, dan FSC
Tidak ada pembatasan label yang digunakan pada <i>upstream nodes</i>	Sebuah <i>ingress</i> atau <i>upstream node</i> mungkin membatasi label yang digunakan oleh LSP sepanjang <i>hop</i> tunggal atau seluruh <i>path</i>

Hanya satu format label	Menggunakan label tertentu pada antarmuka tertentu. Format label berdasarkan antarmuka yang digunakan, seperti PSC, L2SC, TDM, LSC, FSC
Label digunakan untuk penghantaran paket dan membawa data sepanjang trafik	Label hanya merupakan kontruksi <i>control plane</i> dari GMPLS dan bukan merupakan bagian dari trafik
Tidak memerlukan parameter teknologi yang spesifik, karena hanya digunakan pada antarmuka paket/ <i>cell</i>	Mendukung parameter teknologi yang spesifik yang dicantumkan pada proses pensinyalan
<i>Channel</i> data dan kendali berada pada <i>path</i> yang sama	<i>Channel</i> data dan kendali berada pada <i>path</i> yang berbeda
<i>Unidirectional</i> LSPs	<i>Bidirectional</i> LSPs
Label tidak dapat disarankan oleh <i>upstream node</i>	Memungkinkan label disarankan oleh <i>upstream nodes</i> dan dapat ditulis ulang oleh <i>downstream node</i>

Sumber: Sam Halabi, 2003



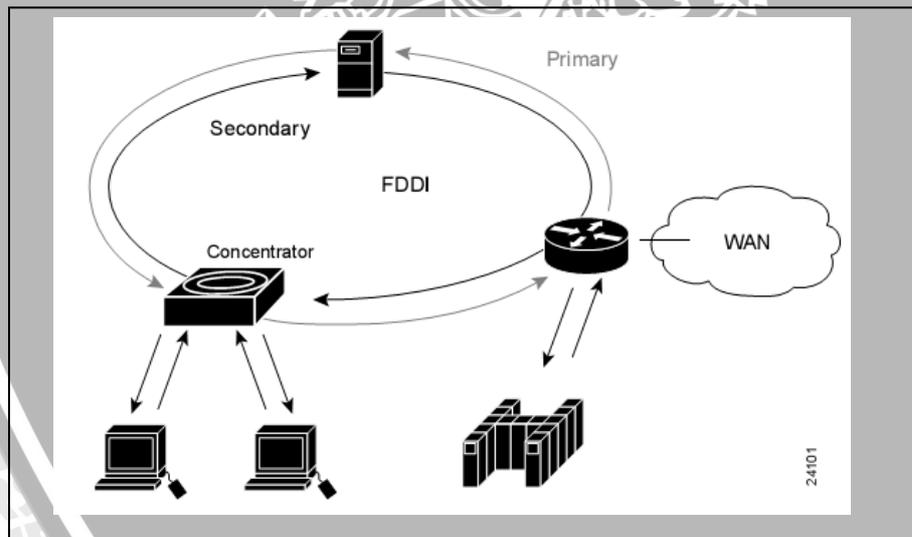
BAB III

FDDI (*Fiber Distributed Data Interface*)

3.1 Teknologi FDDI

FDDI menspesifikasikan konsep teknologi jaringan berkecepatan 100 Mbps dengan menerapkan metode *token-passing*, *dual-ring* LAN menggunakan kabel serat optik. FDDI sering digunakan sebagai teknologi *backbone* kecepatan tinggi karena mendukung *bandwidth* tinggi dan jarak yang lebih jauh daripada kabel tembaga.

FDDI menggunakan arsitektur *dual-ring* dengan trafik di tiap *ring* bergerak berlawanan arah (*counter-rotating*). *Dual-ring* terdiri dari *ring* primer dan *ring* sekunder. Selama berada pada keadaan normal, *ring* primer digunakan untuk transmisi data. Sedangkan *ring* sekunder berada dalam keadaan *idle*, kecuali dibutuhkan. Hal ini memungkinkan untuk memperkecil kesalahan pentransmisi dengan cara *beaconing* (pengidentifikasi) dan konfigurasi ulang jaringannya.



Gambar 3.1 Counter-rotating pada FDDI. Ring primer dan ring sekunder

Sumber: http://ebook.planindo.com/dl.php?f=Chapter08_-_FDDI.pdf

FDDI dikembangkan oleh komite standar ANSI (*American National Standards Institute*) X3T9.5 pada pertengahan tahun 80-an. Beberapa fitur yang dimiliki FDDI antara lain:

1. Kemampuan untuk menggunakan serat optik *multi-mode* dan *single-mode*
2. Topologi *dual-ring* untuk mengantisipasi kerusakan

3. Beroperasi dengan *data rate* 100 Mbps dan mampu mempertahankan transfer *data rate* efektif sampai 80 Mbps
4. Secara teoritis dimungkinkan untuk menyambungkan sejumlah *station*, walaupun berdasarkan standar diasumsikan penyambungan tidak boleh lebih dari 1000 *physical attachment*
5. Panjang serat optik yang digunakan mencapai 200 km
6. Kemampuan untuk mengalokasikan *bandwidth* secara dinamis sehingga layanan data baik *asynchronous* atau *synchronous* dapat dihasilkan secara simultan.

3.2 Media Transmisi FDDI

FDDI menggunakan serat optik sebagai media transmisi utamanya. Beberapa keuntungan dan kerugian dari penggunaan serat optik.

Tabel 3.1 Keuntungan dan kerugian penggunaan serat optik sebagai media transmisi

Keuntungan	Kerugian
1. Rugi-rugi transmisi rendah dan <i>bandwidth</i> lebar.	1. Konstruksi serat optik lemah sehingga dalam pemakaiannya diperlukan lapisan penguat sebagai pengaman.
2. Tahan terhadap <i>electrical interference</i> dan pengaruh frekuensi radio (RFI (<i>radio frequency interference</i>)) dan EMI (<i>electromagnetic interference</i>)).	2. Karakteristik transmisi dapat berubah bila terjadi tekanan dari luar yang berlebihan
3. Serat optik merupakan isolator listrik, sehingga tidak ada kekhawatiran adanya <i>ground loop</i> , percikan api, dan <i>crosstalk</i> .	3. Tidak dapat dialiri arus listrik, sehingga tidak dapat memberikan catuan pada pemasangan <i>repeater</i> .
4. Tingkat keamanan yang baik, karena tidak dapat dicatu sembarangan	

Sumber: http://www.itelkom.ac.id/library/index.php?option=com_content&view=article&id=410:serat-optik&catid=23:sistem-komunikasi-optik&Itemid=15

3.2.1 Tipe Kabel Serat Optik

FDDI mendefinisikan dua tipe serat optik yang dapat digunakan:

1. *Single-Mode*

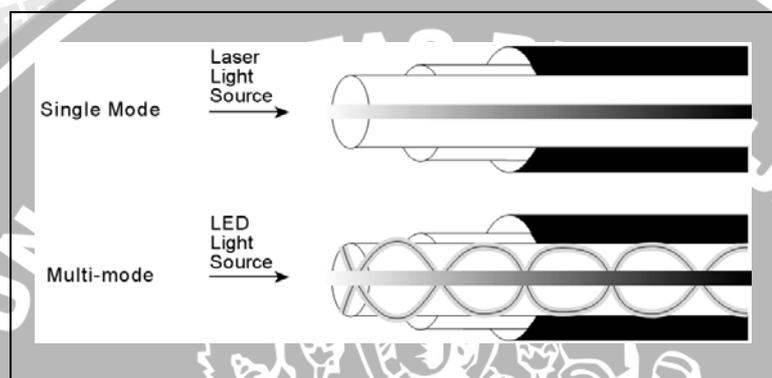
Serat optik *single-mode* memungkinkan hanya satu *mode* cahaya untuk

penghantaran melalui serat optik. Sebuah *mode* adalah sebuah cahaya yang masuk dalam serat optik pada sudut pantulan tertentu.

2. Multi-Mode

Serat optik *multi-mode* memungkinkan beberapa *mode* cahaya yang dirambatkan di dalam serat optik.

Gambar 3.2 berikut menunjukkan serat optik *single-mode* menggunakan sebuah sumber cahaya laser dan serat optik *multi-mode* menggunakan sumber cahaya LED (*Light Emitting Diode*).



Gambar 3.2 Serat optik *single-mode* dan serat optik *multi-mode*

Sumber: http://ebook.planindo.com/dl.php?f=Chapter08_-_FDDI.pdf

Serat optik *single-mode* menyediakan kapasitas *bandwidth* transmisi yang lebih besar dan rentang panjang serat optik yang lebih jauh daripada serat optik *multi-mode*. Hal ini disebabkan oleh adanya beberapa mode perambatan cahaya pada serat optik yang dapat menghantarkan pada jarak yang berbeda-beda (tergantung pada besarnya sudut pantulan). Dengan adanya kondisi tersebut menyebabkan setiap cahaya datang di tujuan pada waktu yang berbeda atau yang disebut dengan *modal dispersion*. Serat optik *single-mode* sering digunakan untuk menghubungkan antar gedung, sedangkan serat optik *multi-mode* sering digunakan untuk menghubungkan ruang atau lantai dalam satu gedung. Serat optik *multi-mode* menggunakan LEDs (*Light Emitting Diodes*) sebagai alat untuk menghasilkan cahaya, sedangkan serat optik *single-mode* secara umum menggunakan sinar laser.

3.2.2 Dispersi

Pulsa-pulsa cahaya yang dipancarkan ke dalam serat optik akan mengalami dispersi yang mengakibatkan menyebarnya pulsa (*spread*) dalam daerah waktu.

1. Dispersi Antar Ragam (dispersi *intermodal*)

Apabila cahaya dilewatkan pada serat optik *multi-mode step-index*, maka panjang jalur yang dilewati oleh setiap pancaran cahaya yang berbeda adalah tidak sama. Pancaran cahaya yang berbeda-beda ini disebut *mode*. Akibat dari adanya perbedaan waktu tiba dari *mode* itu adalah pulsa yang ditransmisikan mengalami pelebaran atau disebut dispersi *intermodal*. Dispersi ini hanya terjadi pada serat optik *multi-mode*. Besarnya dispersi *intermodal* untuk serat optik *multi-mode step-index* adalah:

$$\tau_{ims} = \frac{n_1 \cdot D \cdot \Delta_s^2}{C} \quad (3-1)$$

Selisih indeks bias, serat optik *multi-mode step-index*:

$$\Delta_s = \frac{n_1 - n_2}{n_1} \quad (3-2)$$

dengan:

τ_{ims} = dispersi intermodal serat optik *multi-mode step-index* (s)

D = panjang serat optik (m)

Δ_s = perbedaan indeks bias relatif

n_1 = indeks bias inti

n_2 = indeks bias selubung

C = kecepatan cahaya (3×10^8 m/s)

Besarnya dispersi *intermodal* untuk serat optik *multi-mode graded-index* adalah:

$$\tau_{img} = \frac{n_1 \cdot D \cdot \Delta_g^2}{8C} \quad (3-3)$$

Selisih indeks bias serat optik *multi-mode graded-index*:

$$\Delta_g = \frac{n_1^2 - n_2^2}{2n_1^2} \quad (3-4)$$

dengan:

τ_{img} = dispersi *intermodal* serat optik *multi-mode graded-index* (s)

Δ_g = perbedaan indeks bias relative

2. Dispersi Bahan (*Material Dispersion*)

Apabila cahaya yang memiliki lebih dari satu panjang gelombang menjalar dalam serat optik, maka cahaya tersebut akan menjalar dengan kecepatan yang berbeda. Besarnya dispersi material adalah:

$$\tau_m = \Delta\lambda.D.M \quad (3-5)$$

dengan:

τ_m = dispersi material (s)

$\Delta\lambda$ = lebar spektrum (nm)

D = panjang serat optik (m)

M = koefisien dispersi bahan (ps/nm.km)

3. Dispersi *Waveguide*

Dispersi *waveguide* terjadi karena karakteristik perambatan *mode* panjang gelombang sebagai fungsi dari jari-jari inti, perbedaan nilai indeks bias, dan profil indeks bias. Besarnya adalah:

$$\tau_w = \frac{D}{C\lambda_0} (n_1 - n_2) D_w \Delta\lambda \quad (3-6)$$

dengan:

τ_w = dispersi *waveguide* (s)

D = panjang serat optik (m)

n_1 = indeks bias inti

n_2 = indeks bias selubung

D_w = koefisien dispersi bahan (ps/nm.km)

$\Delta\lambda$ = lebar spektrum (nm)

λ_0 = *zero dispersion wavelength* (nm)

C = kecepatan cahaya (3×10^8 m/s)

Dispersi total yang terjadi di dalam serat optik adalah:

$$\tau_f = \sqrt{\tau_{im}^2 + \tau_m^2 + \tau_w^2} \quad (3-7)$$

dengan:

τ_f = dispersi total sistem (s)

τ_{im} = dispersi intermodal (s)

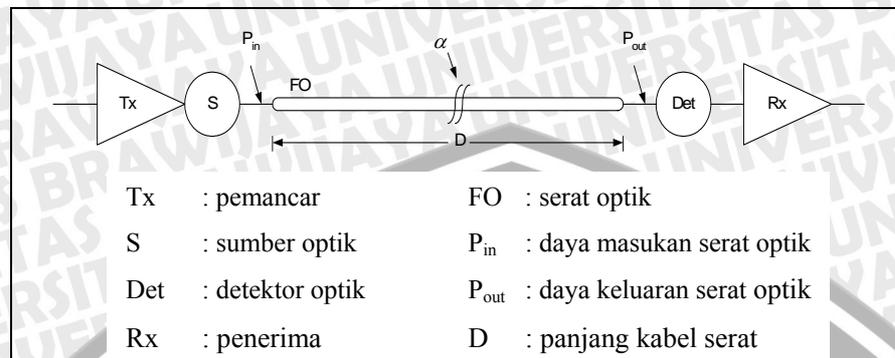
τ_m = dispersi material (s)

τ_w = dispersi *waveguide* (s)

3.2.3 Redaman Serat Optik

Redaman serat optik atau disebut juga rugi-rugi serat optik akan menimbulkan

penurunan daya keluaran serat optik, sehingga menentukan jarak transmisi maksimum antara pemancar dan penerima tanpa pemancar ulang (*repeaterless*).



Gambar 3.3 Redaman di sepanjang serat optik

Sumber: Robert J. Hoss, 1990:32

Empat macam sumber redaman serat optik yaitu:

1. Rugi-rugi hamburan (*Scattering Loss*)
2. Rugi-rugi *coupling, splicing*, dan konektor
3. Rugi-rugi pembengkokan (*Bending Loss*)
4. Rugi-rugi penyerapan bahan (*Material Absorbition Loss*)

Rugi-rugi penyerapan bahan dan hamburan merupakan faktor rugi-rugi dari dalam serat optik, sedangkan rugi-rugi pembengkokan, sambungan, dan konektor merupakan faktor rugi-rugi redaman dari luar serat optik.

1. Rugi-rugi Hamburan (*Scattering Loss*)

Ada dua macam rugi hamburan dalam serat optik, yakni: hamburan linear dan hamburan non-linear. Hamburan linear menyebabkan sebagian daya dari satu mode dipindahkan ke mode lainnya secara linear. Proses ini menghasilkan redaman karena transfer daya ini bisa saja terjadi pada mode bocor atau mode radiasi yang tidak berpropagasi dalam *core* tetapi diradiasikan dalam serat optik. Yang termasuk hamburan linear ini adalah hamburan Rayleigh. Hamburan non-linear menyebabkan transfer daya dari satu mode ke mode lainnya terjadi frekuensi yang berbeda. Hamburan Brillouin dan hamburan Raman termasuk dalam rugi-rugi hamburan non-linear.

2. Rugi-rugi *Coupling, Splicing*, dan Konektor

Rugi-rugi kopleng dihasilkan oleh *coupler*. *Coupler* adalah komponen optik pasif yang berfungsi untuk menginjeksikan energi optik dari saluran transmisi utama

ke transmisi sinyal *point to point*. *Coupler* dapat digunakan untuk membagi dan menggabungkan sinyal optik. Dalam sistem transmisi serat optik, tidak mungkin untuk menghindari adanya konektor dan *splice*. Konektor digunakan untuk menghubungkan serat optik dengan pemancar atau penerima, sehingga pemancar dan penerima dapat diambil keluar dengan mudah untuk keperluan *service*. Faktor yang mempengaruhi besarnya rugi akibat sambungan-sambungan serat optik, antara lain: ketidaksesuaian ukuran inti, kesalahan letak inti melintang, persiapan ujung serat optik yang tidak sempurna dan rugi-rugi celah optis.

3. Rugi-rugi Pembengkokan (*Bending Loss*)

Ada dua jenis pembengkokan yang menyebabkan rugi-rugi dalam serat optik, yaitu pembengkokan mikro (*microbending*) dan pembengkokan radius konstan. Pembengkokan mikro adalah suatu pembengkokan mikroskopis dari inti serat optik, yang disebabkan oleh laju penyusutan (*contraction*) *thermal* yang sedikit berbeda antara bahan inti dan bahan pelapis. Pembengkokan mikro dapat juga timbul bila serat optik berulang kali digulung menjadi suatu kabel serat optik majemuk.

4. Rugi-rugi Penyerapan bahan (*Material Absorbition Loss*)

Absorpsi bahan adalah rugi yang berhubungan dengan komposisi bahan dan proses fabrikasi serat optik. Absorpsi material menyebabkan disipasi daya sebagai panas dalam *waveguide*. Ada dua macam rugi-rugi penyerapan bahan, yaitu intrinsik dan ekstrinsik. Rugi-rugi intrinsik disebabkan oleh resonansi atomik dari bahan serat optik. Absorpsi intrinsik memiliki nilai rendah pada propagasi gelombang dengan panjang gelombang antara 0,8-1,7 μm . Sedangkan rugi-rugi ekstrinsik disebabkan oleh resonansi atomik dan partikel pada serat optik.

3.2.4 Penguat Optik

Penguat optik adalah perangkat yang akan memperbesar daya sinyal optik. Hal ini dipergunakan untuk mengatasi rugi-rugi daya sepanjang pentransmisi. Bila dibandingkan dengan penguat elektrik, penguat optik mempunyai *bandwidth* yang lebih lebar (selisih 30 dB) dan dapat memperkuat masukan-masukan optik jamak dengan panjang gelombang yang berbeda.

Syarat-syarat yang harus dipenuhi oleh penguat optik antara lain:

1. *Gain* daya yang tinggi
2. Efisiensi *pumping* eksternal yang tinggi
3. Efek saturasi yang kecil
4. *Bandwidth* yang besar
5. *Crosstalk* yang kecil
6. Rugi-rugi kopling yang rendah.

Terdapat dua jenis penguat optik yang penting, yaitu penguat semikonduktor (*semiconductor amplifier*) dan EDFA (*erbium-doped fiber amplifier*). Pada EDFA, dengan *gain* daya dan efisiensi *pumping* eksternal yang tinggi, *crosstalk* yang rendah, serta rugi-rugi kopling yang kecil, EDFA sangat bermanfaat untuk memperkuat sinyal optik pada panjang gelombang 1,55 μm . Selain memperkuat sinyal, EDFA juga membangkitkan *noise ASE (amplified spontaneous emission)* sehingga dapat menurunkan SNR [Max Ming-Kang Liu, 1996:881].

Amplifier dengan *gain* sebesar G dan sinyal masukan P_{in} akan menghasilkan sinyal keluaran $P_{out} = G.P_{in}$. SNR sinyal masukan dinyatakan [Govin P. Agrawal, 1992:334]:

$$(SNR)_{in} = \frac{P_{in}}{2hf\Delta_f} \quad (3-8)$$

dengan:

$(SNR)_{in}$ = SNR sinyal masukan (dB)

P_{in} = daya masukan penguat (W)

h = konstanta Planck ($6,626 \times 10^{-34}$ J/s)

f = frekuensi gelombang cahaya (Hz)

Δ_f = *bandwidth* detektor (Hz)

SNR sinyal yang dikuatkan dinyatakan dengan [Govin P. Agrawal, 1992:335]:

$$(SNR)_{out} = \frac{GP_{in}}{4S_{sp}\Delta_f} \quad (3-9)$$

$$S_{sp} = (G-1)n_{sp}hf \quad (3-10)$$

dengan:

$(SNR)_{out}$ = SNR sinyal yang dikuatkan (dB)

G = *gain amplifier*

- P_{in} = daya masukan penguat (W)
 S_{sp} = kerapatan spektral emisi spontan penyebab *noise*
 Δ_f = *bandwidth* detektor (Hz)
 h = konstanta Planck ($6,626 \times 10^{-34}$ J/s)
 f = frekuensi gelombang cahaya (Hz)
 n_{sp} = faktor emisi spontan

3.2.5 Link Budget Sistem Komunikasi Serat Optik

Link budget untuk sistem komunikasi serat optik yang akan dibahas berikut antara lain *bandwidth budget* dan *power budget*.

3.2.5.1 Bandwidth Budget

Kemampuan laju data suatu *link* didapatkan dari *bandwidth budget*. Parameter ini adalah suatu teknik yang diperlukan untuk menentukan pengaruh *transmitter* dan komponen sistem yang lain terhadap laju bit dalam *link*. Lebar jalur (*bandwidth*) sistem untuk format pengkodean NRZ adalah [Robert J. Hoss, 1990:159]:

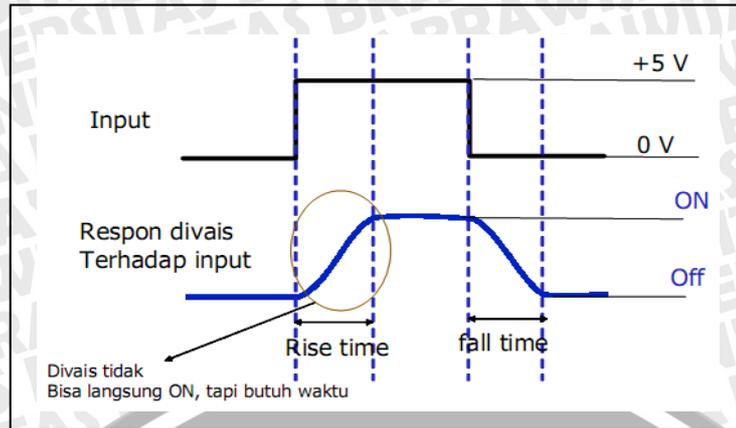
$$BW = \frac{B_{maks}}{2} \quad (3-11)$$

dengan:

- BW = *bandwidth* sistem (Hz)
 B_{maks} = laju bit transmisi maksimum (bps)

Bandwidth Budget dapat diperoleh dari perhitungan *rise-time sistem*. Analisis *rise-time budget* diperlukan untuk menentukan batas dispersi sebuah *link* serat optik, khusus dalam sistem digital. *Rise-time link* total t_{sys} adalah jumlah rms *rise-time* dari tiap kontributor t_i pada degradasi *rise-time* pulsa:

$$t_{sys} = \left(\sum_{i=1}^N t_i^2 \right)^{1/2} \quad (3-12)$$



Gambar 3.4 Rise-time

Sumber: <http://septialutfi.edublogs.org/files/2007/11/dasar-sko.pdf>

Empat elemen dasar yang membatasi kecepatan sistem adalah *rise-time transmitter* t_{TX} , *rise-time* dispersi intermodal t_{mod} , *rise-time* dispersi bahan serat optik t_{mat} , dan *rise-time* penerima t_{RX} .

$$t_{sys} = (t_{TX}^2 + D^2 \sigma_\lambda^2 L^2 + t_{RX}^2)^{1/2} \quad (3-13)$$

- Nilai t_{TX} (s) bergantung dari sumber cahaya dan rangkaian *drive*-nya
- Nilai t_{RX} dihasilkan oleh respon fotodetektor dan *bandwidth* 3 dB penerima.

$$t_{RX} = \frac{350}{B_{RX}} ns ; B_{RX} \text{ adalah } bandwidth \text{ listrik 3 dB penerima (MHz)}$$

- D adalah dispersi total (s)
- σ_λ adalah lebar spectral sumber cahaya (nm)
- L adalah panjang *link* (m)

Secara umum, degradasi waktu transisi total sebuah *link* digital tidak melebihi 70% dari sebuah periode bit NRZ atau 35% dari satu periode bit untuk data RZ.

3.2.5.2 Link Budget

Link budget memperhatikan rugi daya optik total yang diijinkan antara sumber cahaya dan detektor cahaya, serta mengalokasikannya untuk redaman kabel, rugi pada konektor, rugi pada *splice*, dan margin sistem. Margin sistem adalah tambahan penyebab rugi lain yang ditambahkan untuk mengantisipasi usia komponen, fluktuasi temperatur udara, dan rugi lain dari komponen yang mungkin ditambahkan di kemudian hari. Rugi-rugi total tidak boleh melebihi selisih antara daya optik yang dipancarkan dengan daya yang dideteksi. Panjang saluran transmisi serat optik akan dibatasi oleh

besar rugi-rugi yang timbul di sepanjang saluran. Jika P_S adalah daya optik pada ujung serat optik yang terhubung ke sumber cahaya, dan jika P_R adalah sensitivitas penerima, maka:

$$L_T = P_S - P_R \quad (3-14)$$

$$L_T = 2\alpha_c + \alpha_f L + \text{margin sistem} \quad (3-15)$$

dengan:

L_T : redaman daya total (dB),

α_c : rugi konektor (dB),

α_f : redaman serat optik (dB/km),

L : jarak transmisi (km)

margin sistem : umumnya diberikan 6 dB.

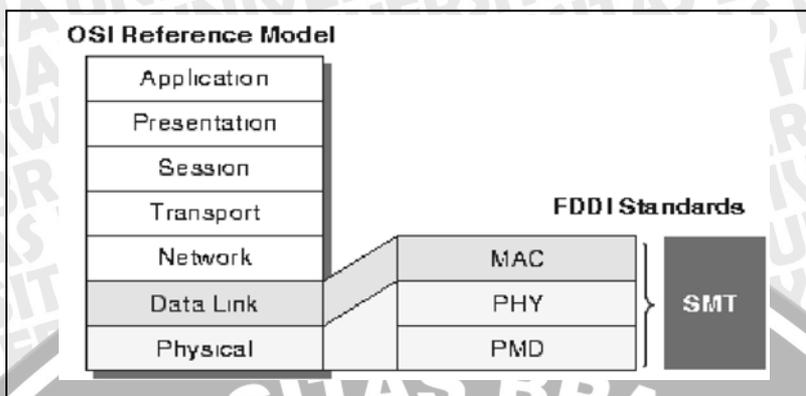
3.3 Spesifikasi FDDI

FDDI menspesifikasikan bagian fisik dan akses media dari model referensi OSI. FDDI sebenarnya bukan merupakan spesifikasi tunggal, tetapi merupakan kumpulan empat spesifikasi terpisah dimana tiap spesifikasi mempunyai fungsi tertentu. Kombinasi dari empat spesifikasi ini menghasilkan kemampuan untuk mendukung konektivitas kecepatan tinggi antara *upper-layer protocol* seperti TCP/IP dan IPX (*Internet Packet Exchange*) dan media seperti kabel serat optik.

Empat spesifikasi FDDI tersebut adalah MAC, PHY (*Physical Layer Protocol*), PMD (*Physical-Medium Dependent*), dan SMT (*Station Management*). Spesifikasi MAC mendefinisikan bagaimana *medium* dapat diakses, termasuk di dalamnya format *frame*, *token handling*, pengalamatan, algoritma untuk menghitung nilai CRC (*cyclic redundancy check*), dan mekanisme *error-recovery*. Spesifikasi PHY diantaranya mendefinisikan prosedur enkoding/dekoding data, kebutuhan *clock*, dan *framing*. Spesifikasi PMD mendefinisikan karakteristik media transmisi, termasuk di dalamnya *link* serat optik, *power level*, *bit-error rates*, komponen optik, dan konektor. Spesifikasi SMT mendefinisikan konfigurasi FDDI *station*, konfigurasi *ring*, fitur kontrol *ring*, termasuk di dalamnya *station insertion and removal*, inialisasi, pemulihan dan isolasi kesalahan (*fault isolation and recovery*), *scheduling*, dan *statistic collection*.

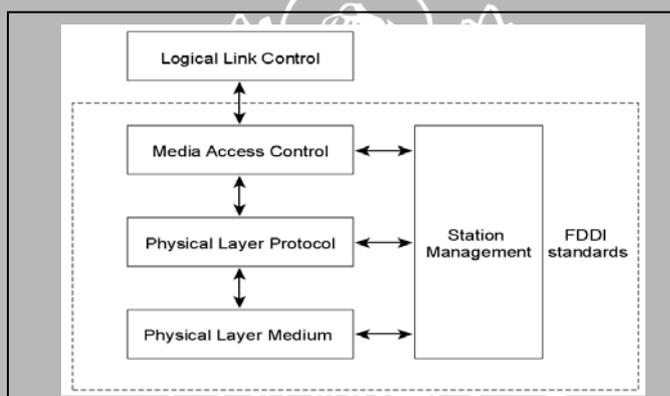
FDDI mirip dengan IEEE 802.3 *Ethernet* dan IEEE 802.5 *Token Ring* yang hubungannya sama dengan model OSI. Maksud kesamaan hubungan ini adalah untuk

memberikan konektivitas antara lapisan atas OSI (*OSI upper layer*) dari protokol yang sering digunakan dan media yang digunakan untuk menghubungkan *device* jaringan.



Gambar 3.5 Standar protokol FDDI dan hubungannya dengan referensi OSI

Sumber: <http://lecturer.ukdw.ac.id/budsus/jarkom/bab6.pdf>



Gambar 3.6 Spesifikasi FDDI dan hubungannya dengan sublapisan LLC

Sumber: http://ebook.planindo.com/dl.php?f=Chapter08_-_FDDI.pdf

Pada Gambar 3.6 menjelaskan hubungan satu dengan lainnya pada spesifikasi FDDI dan hubungannya dengan sublapisan LLC.

1. PMD (*Physical layer Medium Dependent*)

PMD merupakan *sublayer* terbawah dari OSI *physical layer*. Pada *layer* ini terdapat spesifikasi untuk level tenaga, karakteristik dari *transmitter* optik dan *receiver* optik, dan *bit error rate* yang diijinkan.

2. PHY (*Physical layer protocol*)

PHY merupakan *sublayer* teratas yang berhubungan dengan OSI *physical layer*. *Layer* ini berfungsi untuk melakukan skema enkoding, sinkronisasi *clock*, dan *framing data*.

3. MAC (*Media Access Control*)

MAC bertanggung jawab untuk menentukan aturan bagi *medium access, addressing, frame format, error checking, dan token management.*

4. SMT (*Station Management*)

SMT menyediakan pelayanan penting pada tingkat *station* untuk mengamati dan mengontrol *station* FDDI. SMT mengizinkan *station* untuk bekerja bersama dalam *ring* dan memastikan operasi *station* bekerja dengan benar. SMT terdiri atas 3 komponen utama yaitu:

- CMT (*Connection*)
- RMT (*Ring Management*)
- SMT *frame service*

5. LLC (*Logical Link Control*)

LLC ini khususnya bertugas untuk membatasi *service primitive, frame format, addressing structure, dan sejumlah prosedur kontrol.* LLC ini dibutuhkan terutama pada LANs dan MANs (*Metropolitan Area Networks*) karena menyediakan *uniform protocol interface* antara *layer* di atas dengan *layer* di bawahnya.

3.4 Arsitektur FDDI

FDDI menggunakan arsitektur *dual-ring* dengan trafik pada setiap *ring* bergerak pada arah berlawanan (*counter-rotating*). Arsitektur *dual-ring* ini terdiri atas *ring* primer dan *ring* sekunder. Saat beroperasi normal, *ring* primer digunakan untuk pentransmisi data sedangkan *ring* sekunder pada kondisi *idle*. Tujuan utama dari *dual-ring* adalah untuk menghasilkan sifat reliabilitas yang tinggi dan kuat. Karakteristik jaringan FDDI ditunjukkan dalam table 3.2 berikut:

Tabel 3.2 Karakteristik Jaringan FDDI

Karakteristik FDDI	
Media transmisi	Serat optik <i>multi-mode</i> atau <i>single-mode</i>
Sumber cahaya	LED atau laser yang beroperasi pada panjang gelombang sekitar 1300 nm
Skema encoding	4B/5B + NRZ1
Topologi	<i>Dual-ring</i> , dengan arah berlawanan
Metode akses	<i>Token passing</i> , dengan melewati banyak <i>frame</i>



Ukuran <i>frame</i>	Maksimum 4500 <i>byte</i>
<i>Layer</i>	PMD <i>optical</i> , PHY, MAC, SMT
Kinerja	Mendukung kecepatan transmisi sampai 100 Mbps, dapat menyediakan dan menjaga <i>bandwidth</i> yang diberikan, mendukung 1000 <i>node</i> pada jaringan terhubung, panjang saluran transmisi 200 km, mendukung jarak antar <i>node</i> maksimum sejauh 2 km untuk kabel serat optik <i>multi-mode</i> dan 40 km untuk kabel serat optik <i>single-mode</i>

Sumber: Werner Feibel, 1996:381

3.5 Tipe *Station-Attachment* FDDI

Dalam jaringan FDDI, *node* adalah elemen aktif pada jaringan. *Node* berfungsi untuk mengulang pentransmisi informasi yang datang tetapi tidak mempunyai kemampuan untuk menjalankan fungsi perbaikan kesalahan pada *data link layer*.

Sedangkan *station* pada jaringan FDDI adalah *node* yang mempunyai alamat di jaringan. *Station* harus mempunyai kemampuan untuk mengirim, menerima, dan membangkitkan informasi.

Station dan *node* lebih jauh lagi dijelaskan sebagai *single-attachment* atau *dual-attachment*. Jaringan FDDI dibagi menjadi *counter-rotating ring*. Satu *ring* disebut sebagai *ring primer*, yang secara logika terhubung pada seluruh *station*. Kemudian *ring* yang lain disebut sebagai *ring sekunder*, yang hanya terhubung pada *dual-attachment station*. Penggunaan *dual-ring* pada jaringan FDDI memiliki arti penting dalam kemampuan jaringan untuk memperbaiki diri bila terjadi kerusakan pada media atau *station*.

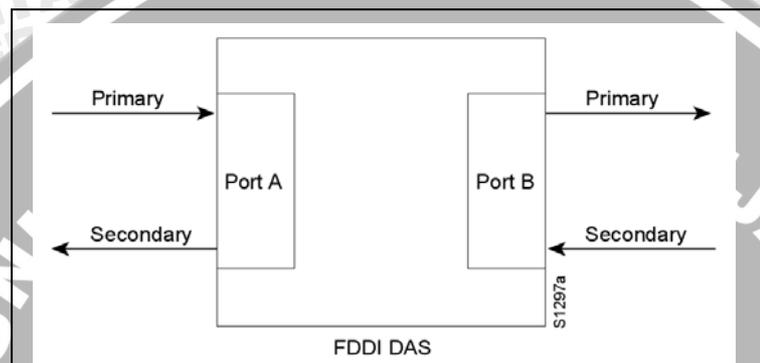
Karakteristik unik dari FDDI adalah mempunyai beberapa jalur yang dapat dihubungkan dengan *device* FDDI. FDDI mendefinisikan ada tiga tipe *device*, yaitu: SAS (*single-attachment station*), DAS (*dual-attachment station*), dan *concentrator*. Tiga tipe dasar dari *station* FDDI yaitu:

1. SAS (*Single-Attachment Station*)

Station ini terhubung hanya pada *ring primer* (melalui *concentrator*), sehingga bila terjadi kerusakan pada *ring* FDDI maka SAS akan terisolasi dari jaringan. Tetapi *device* yang terhubung dengan SAS tidak berpengaruh terhadap keadaan *ring* FDDI (*ring primer*).

2. DAS (*Dual-Attachment Station*)

Station ini terhubung ke *ring primer* dan juga ke *ring sekunder*. FDDI DAS mempunyai dua *port*, *port A* dan *port B*. Bila terjadi kerusakan pada *ring primer* atau *ring sekunder*, maka DAS pada sisi lain dari *station* yang rusak akan memperbaiki *ring*. *Server* atau *router* tersambung pada *primary* dan *secondary ring* dan terlibat dalam *wrapped ring* saat terjadi kerusakan pada salah satu *station*.



Gambar 3.7 Konfigurasi dari DAS

Sumber: http://ebook.planindo.com/dl.php?f=Chapter08_-_FDDI.pdf

3. FDDI Concentrator

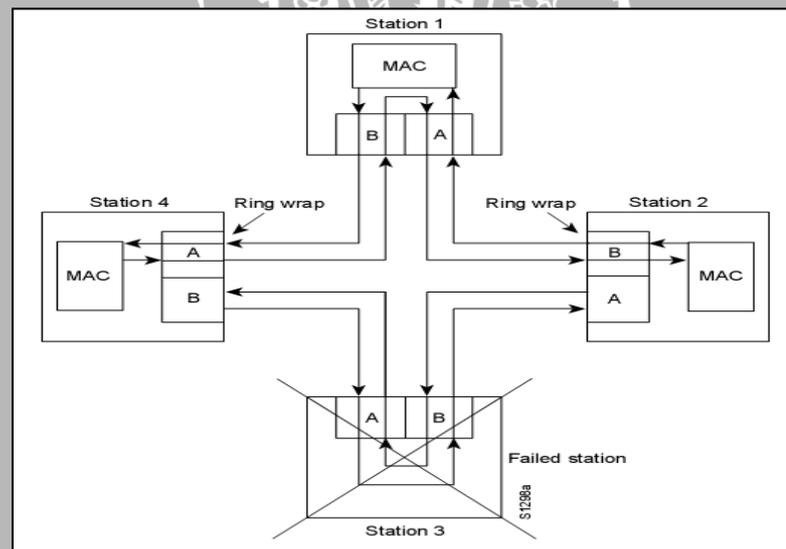
FDDI Concentrator (DAC [*dual-attachment concentrator*]) merupakan *node* lain yang mempunyai *port* tambahan di luar yang dibutuhkan untuk penyambungan ke jaringan. FDDI Concentrator berhubungan langsung dengan *ring primer* dan *sekunder* dan menjamin segala kegagalan atau *power-down* pada *device* yang terhubung dengan SAS tidak mempengaruhi *ring*. Concentrator menyediakan satu atau lebih *port* untuk SAS. SAC (*single-attachment concentrator*) digunakan untuk menghubungkan SAS dengan topologi *tree* dan kurang bagus dalam pengkoneksian dari *station* ke *backbone* karena hanya satu bagian yang tersambung. DAC dihubungkan pada kedua *ring* jaringan FDDI, sehingga bila terjadi kerusakan akan dapat segera diperbaiki, selain itu menyediakan koneksi yang kuat dari *station* ke *backbone network*. (Lihat Gambar 3.1)

3.6 FDDI *Fault-Tolerance*

Salah satu kelebihan utama dari FDDI adalah sifat *fault-tolerance*, yaitu kemampuan ring untuk memulihkan diri bila terjadi kerusakan pada salah satu *station* atau kabel putus. Sifat *self-healing* ini sangat penting pada jaringan FDDI karena akan menjamin kelangsungan transmisi data lewat jaringan dari kerusakan-kerusakan sehingga kegagalan kerja dari jaringan dapat dihindari. Beberapa *fault-tolerance features* pada FDDI yang sering digunakan, yaitu konfigurasi *dual ring*, pemakaian *optical bypass switch*, dan *dual homing*.

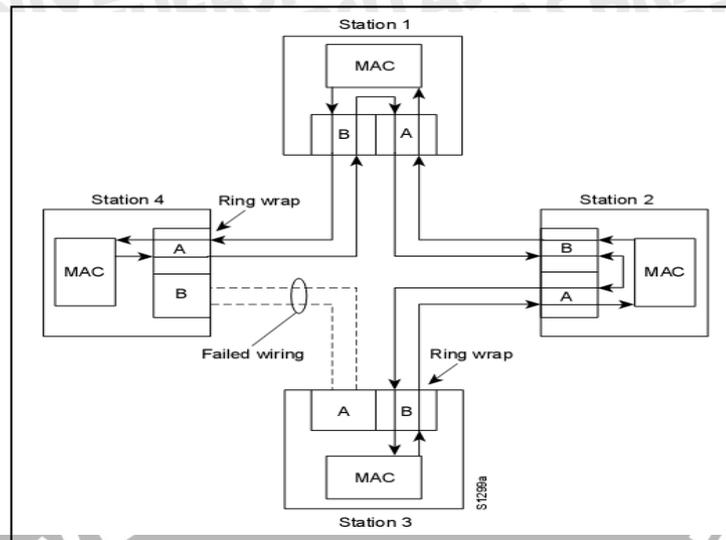
1. *Dual Ring*

Jika terdapat *station* yang rusak atau kabel yang putus pada jaringan maka secara otomatis terjadi penginterkoneksi antara *ring* primer dan *ring* sekunder (*wrapped ring*). Saat *ring* tersebut terhubung maka topologi *dual ring* berubah menjadi topologi *single ring*. Meskipun terjadi interkoneksi, data akan terus ditransmisikan dalam jaringan FDDI tanpa mempengaruhi performansi jaringan.



Gambar 3.8 Pemulihan *ring* ketika salah satu *station* mengalami kerusakan pada metode *Dual Ring*

Sumber: http://ebook.planindo.com/dl.php?f=Chapter08_-_FDDI.pdf

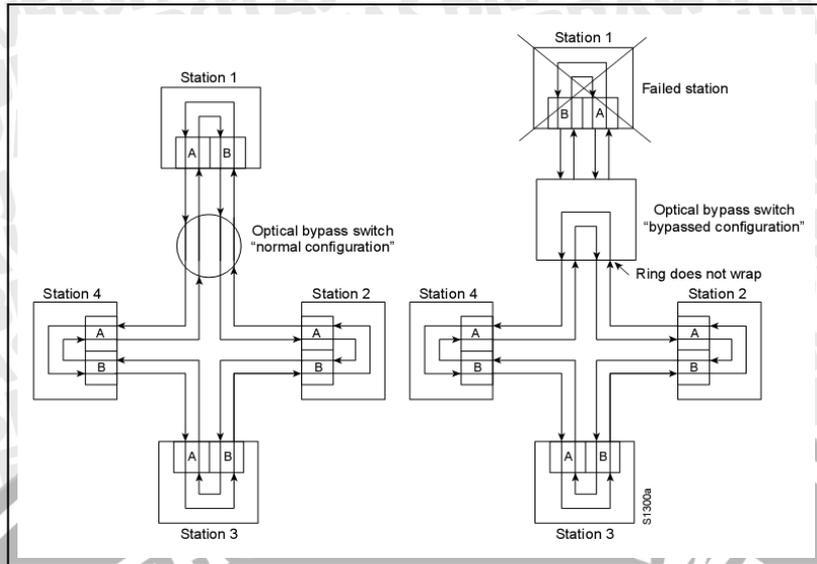


Gambar 3.9 Pemulihan *ring* ketika terdapat kabel yang putus/rusak pada metode *Dual Ring*

Sumber: http://ebook.planindo.com/dl.php?f=Chapter08_-_FDDI.pdf

2. *Optical bypass switch*

Jika terjadi kerusakan peralatan pada *dual-ring*, maka *optical bypass switch* akan menghasilkan *continuous dual-ring operation*. Hal ini dilakukan untuk mencegah *segmentasi ring* dan juga untuk memindahkan *station* yang rusak dari *ring*. *Optical bypass switch* menjalankan fungsi ini dengan menggunakan lensa optik yang melewatkan cahaya dari *ring* menuju ke peralatan DAS saat operasi normal. Jika pada peralatan DAS terjadi kerusakan, seperti matinya tenaga, *optical bypass switch* akan melewatkan cahaya lewat dirinya sendiri dengan menggunakan lensa internal sehingga akan dapat mempertahankan integritas *ring*. Keuntungan dari metode ini adalah *ring* tidak akan masuk ke kondisi *wrapped* bila terjadi kerusakan pada *ring*.

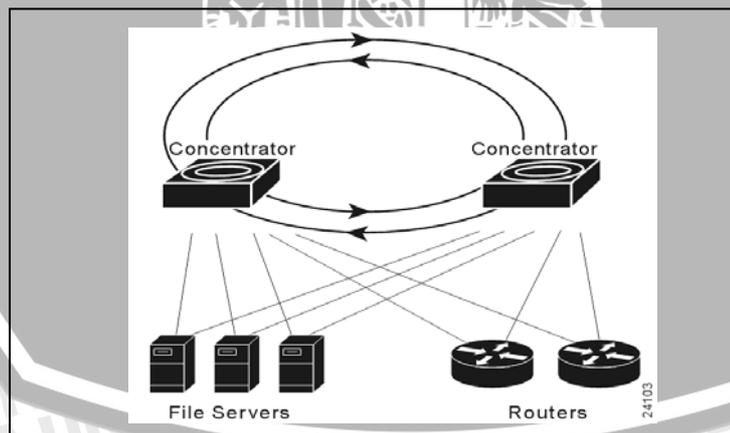


Gambar 3.10 Optical bypass switch menggunakan cermin internal untuk mempertahankan jaringan

Sumber: http://ebook.planindo.com/dl.php?f=Chapter08_-_FDDI.pdf

3. *Dual homing*

Metode ini biasanya diterapkan pada *device-device* penting seperti *router* atau *mainframe host* untuk memberikan jaminan operasi *ring*. Pada keadaan *dual-homing*, *device* penting tersebut dihubungkan dengan dua *concentrator*.

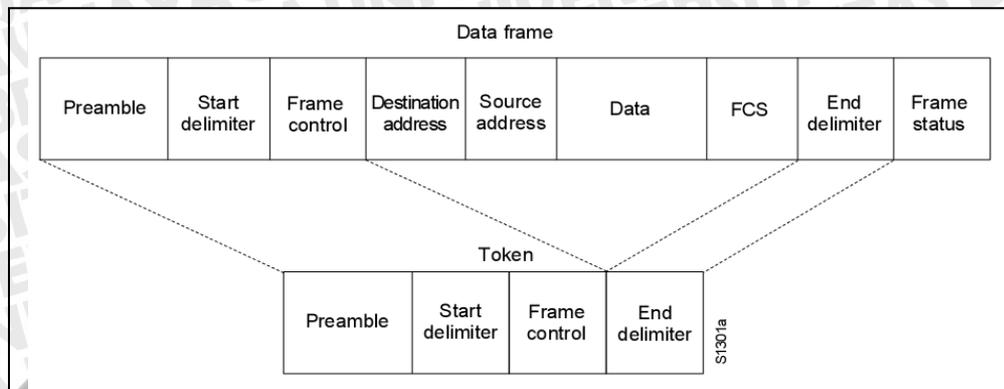


Gambar 3.11 Konfigurasi *dual-homed* untuk *device file server* dan *router*

Sumber: http://ebook.planindo.com/dl.php?f=Chapter08_-_FDDI.pdf

3.7 Format *Frame* FDDI

Format *frame* FDDI sama dengan format *frame* milik *Token Ring*. Ukuran *frame* FDDI adalah 4500 *byte*.



Gambar 3.12 FDDI *transmission frame* dan *frame token*

Sumber: http://ebook.planindo.com/dl.php?f=Chapter08_-_FDDI.pdf

- *Preamble*

Preamble (terdiri atas 16 *idle symbol* atau 64 bit (8 *byte*)) digunakan untuk sinkronisasi *clock* tiap *station* saat penransmisian *frame*.

- *Start delimiter*

Mengindikasikan permulaan dari *frame* dengan memanfaatkan pola pensinyalan yang berbeda dari *frame* lainnya. Terdiri atas 2 simbol non-data, yaitu J dan K

- *Frame Control*

Mengindikasikan besar daerah pengalamatan, memeriksa apakah *frame* berisi data *synchronous* atau *asynchnonous*, dan informasi kontrol lainnya. *Frame* kontrol mempunyai format bit CLFFZZZZ, simbol C mengidentifikasi *frame* sebagai *synchronous* atau *asynchronous*, L menentukan besar pengalamatan 16 atau 48 bit alamat, FF mengindikasikan *frame* LLC atau *frame* MAC, dan ZZZZ menyediakan informasi kontrol untuk *frame* MAC. *Frame* kontrol terdiri atas 2 simbol atau 1 *byte* (8 bit)

- *Destination address*

Berisi alamat *unicast* (*singular*), *multicast* (*group*), *broadcast* (semua *station*). Seperti halnya *Ethernet* dan *Token Ring*. *Destination address* terdiri atas 6 *byte*

- *Source address*

Mengidentifikasi *station* tunggal yang mengirim *frame*. Seperti halnya *Ethernet* dan *Token Ring*, pengalamatan FDDI *source address* panjangnya 6 *byte*

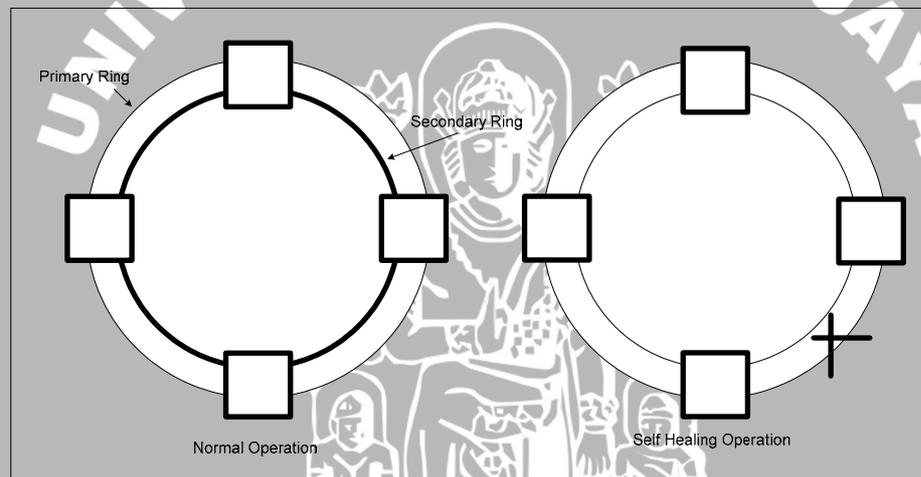
- *Data*
Berisi informasi lain atau data yang ditujukan bagi *upper-layer protocol* atau informasi kontrol. Panjang *frame* maksimum data pada FDDI adalah 4500 *byte*
- *Frame Check Sequence (FCS)*
Berisi nilai 32 bit *cyclic redundancy check*. Nilai ini dihitung berdasarkan isi daerah *frame control*, alamat tujuan, alamat sumber, dan area informasi. *Station* penerima membuat perhitungan yang sama dan membandingkan nilai tersebut dengan nilai yang tersimpan. Jika tidak sama, diperkirakan *frame* mengalami *error*
- *Ending delimiter*
Mengidentifikasi akhir dari *frame*. Berisi satu atau dua non-data *terminate*. *Ending delimiter* terdiri atas 1 *byte* (8 bit)
- *Frame status*
Berisi informasi tentang status *frame*, termasuk apakah *error* terdeteksi, pengenalan alamat, dan penggunaan *frame*. Simbol berlainan digunakan untuk mewakili masing-masing 3 kondisi. Nilai simbol R mengindikasikan nilai kondisi adalah nilai benar atau *on* dan nilai simbol S mengindikasikan bahwa kondisi adalah salah atau *off*.

3.8 Mekanisme Kerja Protokol FDDI

Pada standar FDDI menjelaskan bagaimana fungsi operasi pengontrol *ring* dan *maintenance* yang dirancang untuk menyediakan *survivability* jika terjadi kerusakan. Setelah menerima *token*, *station* yang aktif mengirimkan *frame* berupa kumpulan simbol ke *station* aktif berikutnya dalam *ring*. *Station* tersebut dapat mengirimkan banyak *frame* yang diinginkan sampai batas waktu maksimum tercapai. Ketika *station* sudah tidak mempunyai *frame* untuk dikirimkan atau telah mencapai batas waktu maksimum, *station* mengirimkan *token* kembali ke *ring*, setiap *station* pada *ring* akan mengirimkan kembali *frame* yang diterima dan men-*copy*-nya dari *ring* sesuai dengan alamat *station* yang dituju. Saat *frame* kembali ke *station* yang mengirimnya, maka *station* itu akan bertanggung jawab untuk memindahkan *frame* dari *ring*.

3.8.1 Dual counter-rotating ring

Dual counter-rotating ring merupakan salah satu konsep dasar pada standar FDDI, terdiri atas 2 *ring* yaitu *ring* primer dan *ring* sekunder. Kedua jenis *ring* tersebut dapat membawa data yang lewat ke *ring*. Pada penggambaran *dual counter-rotating ring*, data yang dilewatkan pada *ring* primer dan *ring* sekunder bergerak berlawanan arah. Pada banyak kasus, khususnya aplikasi dengan *bandwidth* tinggi, *ring* primer biasanya digunakan untuk mentransmisikan data, sedang *ring* sekunder digunakan sebagai cadangan. FDDI menggunakan serat optik dengan panjang maksimum 200 km, karena konfigurasi *dual ring*-nya maka panjang dari masing-masing *ring* maksimal 100 km. *Dual counter-rotating ring* dirancang dengan kemampuan untuk memulihkan kerja *ring* jika terdapat *station* yang rusak atau ada kabel yang putus.



Gambar 3.13 Operasi *counter-rotating ring*

Sumber: http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/fddi.html

3.8.2 Operasi Ring

Operasi kerja *ring* dari jaringan FDDI meliputi beberapa tahap, yaitu:

1. Pengoneksian Ring

Untuk membentuk jaringan *ring*, setiap *station* harus membentuk interkoneksi secara fisik antar *station*. Untuk membuat *link*, *station* harus menampilkan fungsi sebagai berikut:

- Bertukar informasi dalam hal tipe *port* dan aturan untuk koneksi
- Menegoisasikan panjang dari *link* dengan memeriksa kualitas *link* dari masing-masing *station*
- Bertukar hasil dan status dalam hal *link* dan pengoneksian.

Jika tipe pengoneksian sudah siap, *station* dapat melengkapi *physical connection* dengan mentransmisikan sejumlah *line-state sequence*. Proses ini akan diulang untuk setiap *link* dalam konfigurasi *dual ring* oleh setiap *station*.

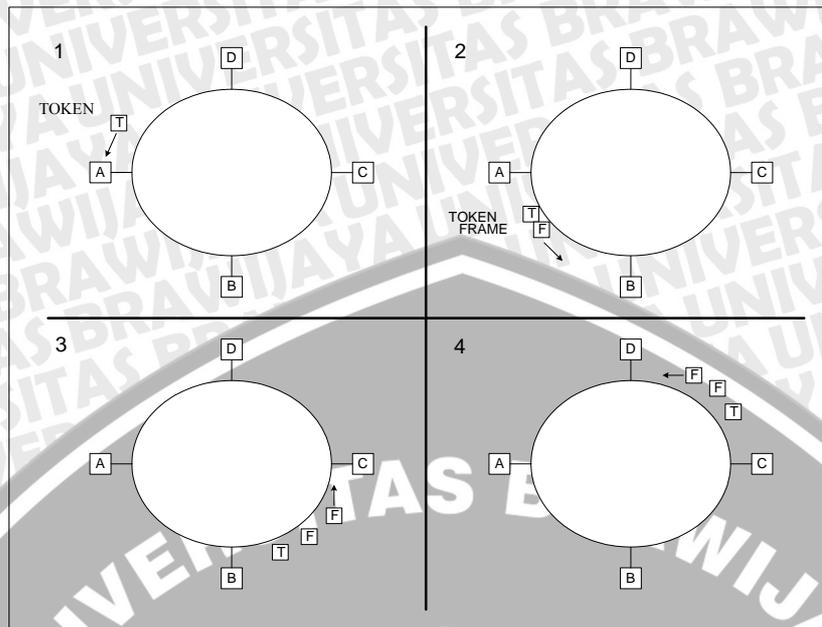
2. Inisialisasi Ring

Sekali saja koneksi ke jaringan *ring* FDDI dibuat, maka protokol FDDI membutuhkan *station* untuk mengklaim hak menginisialisasi *ring* (membangkitkan *token*) dengan menegosiasikan nilai TTRT (*Target Token Rotation Time*), untuk memastikan bahwa *station* menerima jaminan waktu layanan. Nilai TTRT diatur berdasarkan:

- Jumlah *station* yang terhubung ke jaringan *ring*
- Panjang dari *ring*
- Waktu yang dibutuhkan tiap *station* untuk mentransmisikan data lewat *ring*
- Keseimbangan *low latency* dan *bandwidth* yang memadai.

Pada saat inisialisasi *ring*, *station* menegosiasikan nilai TTRT-nya dengan *station* lainnya untuk menentukan *station* mana yang boleh membangkitkan *token* lebih dulu. Proses negosiasi ini disebut sebagai proses *claim*. *Claim* ini akan menentukan *station* mana yang akan lebih dulu menginisialisasi *ring*. Proses *claim* akan dimulai saat MAC *entity* pada satu atau lebih *station* memasuki kondisi *claim*. Pada kondisi ini, MAC pada *station* akan terus mengirimkan *claim frame*. *Claim* ini berisi *station address* dan nilai bit dari TTRT. *Station* pada *ring* akan membandingkan *incoming claim frames* dengan nilai TTRT yang ditawarkannya. Jika *frame* mempunyai waktu yang lebih pendek, *station* akan mengulang waktu yang lebih lama, *station* memindahkan *claim frame* dan meneruskan untuk mengirimkan *frame* sesuai dengan nilai TTRT yang dimilikinya.

Saat *station* menerima *claim frame*-nya sendiri, berarti *station* itu memenangkan hak untuk menginisialisasi *ring*. Jika terdapat dua atau lebih *station* membuat penawaran yang sama, *station* dengan alamat yang panjang dan tinggi yang memenangkan penawaran. *Station* pemenang akan menginisialisasi *ring* dengan mengeluarkan *token*. *Token* tersebut kemudian akan mengelilingi *ring* tanpa ditangkap oleh *station* lain. Setelah *token* kembali ke *station* yang mengirimnya, maka *station* tersebut diijinkan untuk mentransmisikan *frame*.



Gambar 3.14 Sirkulasi *frame* FDDI dan *Token*

Sumber: http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/fddi.html

3. Operasi *Steady-state*

Saat *ring* sudah terinisialisasi, jaringan FDDI berada pada kondisi *steady-state operation*. Pada kondisi ini, *station* bertukar *frame* menggunakan *timed-token protocol*. *Ring* akan mempertahankan kondisi *steady-state operation* sampai proses *claim* yang baru diinisialisasikan, seperti halnya saat *station* baru bertanggung jawab dengan *ring*.

Timed-token protocol meliputi sejumlah tahapan yang dilakukan oleh *station* yang memperoleh hak untuk memulai transmisi. Gambar 3.14 menunjukkan pentransmisian *frame* menggunakan FDDI *timed-token protocol*. Berikut ini adalah langkah-langkah *station* FDDI untuk mentransmisikan *frame*:

- Menunggu sampai dapat mendeteksi *token*
- Menangkap *token*
- Menghentikan proses pengulangan *token* (karena tidak ada *token* pada *ring* akan mencegah *station* lain untuk mentransmisikan data ke *ring*)
- Memulai mengirim *frame* (*frame* dapat dikirim sampai tidak ada data untuk dikirimkan)
- Melepaskan *token* menuju *ring* untuk digunakan oleh *station* lain.

Seluruh *station* aktif dalam jaringan FDDI menerima dan mengulang setiap *frame* kecuali *station* pengirim *frame*, kemudian tiap *station* dalam *ring*

membandingkan *destination address* tiap *frame* dengan *address* masing-masing *station* dan mendeteksi *frame error*. Jika pengalamatan cocok, *station* akan meng-copy *frame* dan memberikan simbol status (indikator kontrol) untuk menunjukkan bahwa *station* sudah terkenali alamatnya dan telah meng-copy *frame*.

Station memeriksa kembali *frame error* dan retransmit *frame* ke *station* selanjutnya. Jika *station error*, maka *station* akan memberikan indikator *error*. Setelah *frame* mengelilingi *ring* dan kembali ke *station* semula, *station* pengirim akan memindahkan *frame* dari *ring*. Pemindahan dapat menyebabkan terpecahnya *frame* yang disebut sebagai *fragment*. *Fragment* ini tertinggal dalam *ring*. Tiap *station* FDDI harus memastikan bahwa *frame fragment* tidak menurunkan kerja operasi *ring*.

4. Ring maintenance

Bertanggung jawab untuk mengawasi *ring* dan didistribusikan ke seluruh *station* pada *ring*. Setiap *station* memeriksa kondisi *ring* yang membutuhkan inisialisasi *ring*, seperti kondisi:

- *Ring* tidak aktif lebih lama daripada *valid transmission time*
- Terjadi kerusakan putusnya topologi *physical* atau *logical* pada *ring*.

3.9 FDDI Physical Layer

OSI *Physical layer protocol* menspesifikasikan banyak *item* seperti konektor, tipe media, dan karakteristik mekanik dan koneksi fisik yang meliputi jenis pensinyalan, level arus dan tegangan yang dapat dipakai, dan karakteristik listrik lainnya. Seperti arsitektur komunikasi data lainnya, FDDI juga mempunyai dua protokol *sublayer* yang berhubungan dengan OSI *Physical layer*, yaitu PMD (*Physical layer Medium Dependent Standard*) dan PHY (*Physical Layer Protocol*).

3.9.1 FDDI PMD (*Physical layer Medium Dependent Standard*)

FDDI PMD standar berhubungan dengan setengah bagian bawah dari OSI *Physical layer*. PMD menjelaskan spesifikasi media seperti karakteristik dari media serat optik konektornya dan kebutuhan performansi transmisi.

1. Physical Connection

Physical connection merupakan bangunan dasar dari jaringan FDDI. Standar PMD menjelaskan semua aspek media fisik serat optik yang digunakan. FDDI dapat menggunakan serat optik *single-mode* atau *multi-mode*. MMF-PMD

menggunakan serat optik dengan asumsi 62.5/125 μm dengan jarak *repeater* maksimum 2 km. Pada SMF-PMD menggunakan serat optik dengan ukuran 8.7/125 μm .

FDDI merupakan jaringan *ring*, terdiri atas sejumlah *station* yang dihubungkan *point-to-point* untuk membentuk *loop* tertutup. Tiap *station* menerima sinyal dari sisi input dan membangkitkannya untuk penstransmisi dari sisi *output*. Secara teoritis banyak *station* dapat dihubungkan ke jaringan, meskipun nilai *default* pada standar FDDI tidak boleh lebih dari 100 *physical attachment* atau maksimum terdapat 500 *node* dan panjang maksimum serat optik 200 km. FDDI membatasi panjang total serat optik sebesar 200 km untuk membatasi *delay propagasi token* yang bersirkulasi mengelilingi *ring*. Jika menggunakan topologi *dual-ring* maka jangkauan geografi dari jaringan FDDI dibatasi pada 100 km.

2. PMD (*Physical layer Medium Dependent*) Service

Standar PMD menjelaskan *interface* antara PMD dan *Physical Layer* dan juga pelayanan yang diberikan PMD ke PHY.

3.9.2 FDDI PHY (*Physical Layer Protocol*)

Standar PHY berhubungan dengan *upper sublayer* dari OSI *Physical layer* dan membatasi prosedur dari media yang digunakan. PHY terutama bertanggung jawab untuk membangkitkan dan mempertahankan sinkronisasi waktu, *encoding outgoing bit stream*, dan *decoding incoming bit stream*.

1. *Physical Connection*

FDDI *Physical Connection* terdiri atas *Physical Layer* dari dua *station* yang dihubungkan oleh *Primary Link* dan *Secondary Link* di sepanjang media transmisi.

2. *PHY Service*

PHY layer memberikan layanan kepada *MAC layer* yang berada di atasnya.

3. *Symbol Coding*

FDDI menggunakan *serial baseband transmission* dengan memanfaatkan media serat optik. Pola pensinyalan harus digunakan untuk mentransmisikan baik *binary data* maupun *timing information*.

Untuk memenuhi kebutuhan dari jaringan optik, FDDI *Physical Layer* menggunakan pola *two-stage coding*. *First coding stage* dibentuk oleh PHY dengan mengubah *symbol* dari MAC, ke kode 5 NRZ (*Non-Return to Zero*). Tabel 2.3 menunjukkan *binary coding* untuk *symbol* menggunakan "4-out-of-5" disebut 4B/5B.

Pada model ini, 4 bit data *sequence* dienkodkan ke 5 bit untuk pentransmisi. Saat 5 bit *symbol* digunakan mewakili 4 bit data, maka pensinyalan 4B/5B mempunyai 80% efisiensi dan *data rate* sebesar 100 Mbps.

Encoding stage kedua dibentuk oleh PHY untuk mengubah bit NRZ menjadi bit *stream* NRZ1 (*Non-Return to Zero Invert on ones*). Pada enkoding data menggunakan NRZ, dapat diasumsikan bila terdapat cahaya yang lewat pada serat optik, berarti mewakili bit 1, sedang bila tak ada cahaya yang lewat berarti mewakili bit 0. Kelemahan dari NRZ adalah sulit untuk menyediakan *clock synchronization*, sehingga bit 1 dan bit 0 dikirimkan secara terus menerus dan tidak ada cara untuk mempertahankan *dc balancing*. Sedang pada enkoding data dengan pola NRZ1, *polarity transition* mewakili bit 0. PHY mengubah polaritas sinyal NRZ1 dengan PMD menggunakan layanan PM_UNITDATA. Gambar 3.15 menunjukkan contoh simbol 4B/5B yang diubah ke NRZ, kemudian diubah ke NRZ1.



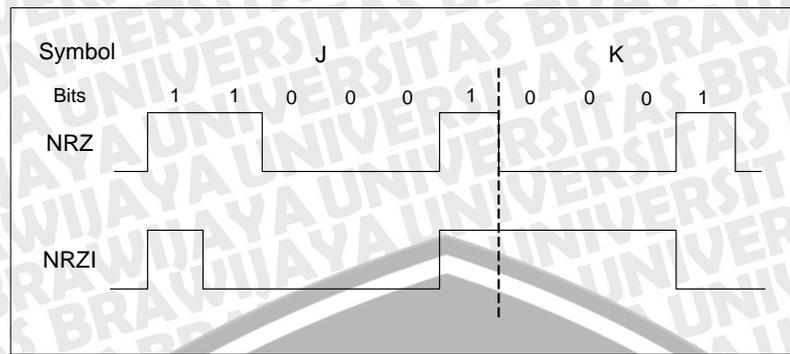
Tabel 3.3 Simbol Pengkodean pada *first coding stage*

Code Group		Symbol		
Decimal	Binary	Name	Assignment	
<i>Line State Symbols</i>				
00	00000	Q	Quiet	
04	00100	H	Halt	
31	11111	I	Idle	
<i>Starting Delimiter</i>				
24	11000	J	First symbol of JK pair	
17	10001	K	Second symbol of JK pair	
<i>Embedded Delimiter</i>				
05	00101	L	Second symbol of IL pair	
<i>Data Quartets</i>			Hexadecimal	Binary
30	11110	0	0	0000
09	01001	1	1	0001
20	10100	2	2	0010
21	10101	3	3	0011
10	01010	4	4	0100
11	01011	5	5	0101
14	01110	6	6	0110
15	01111	7	7	0111
18	10010	8	8	1000
19	10011	9	9	1001
22	10110	A	A	1010
23	10111	B	B	1011
26	11010	C	C	1100
27	11011	D	D	1101
28	11100	E	E	1110
29	11101	F	F	1111
<i>Ending Delimiter</i>				
13	01101	T	Terminate	
<i>Control Indicators</i>				
07	00111	R	Reset (logical Zero or Off)	
25	11001	S	Set (logical One or On)	

Note the 4B/5B code.

Source: Extracted from Ref. 3. X3.231-1994.

Sumber: Roger L. Freeman, 2001:311



Gambar 3.15 Simbol 4B/5B dan contoh NRZ/NRZI coding

Sumber: Gary C. Kessler and David A. A. Train, 1992:126

FDDI PHY mempunyai tujuh jenis simbol dasar, yaitu:

1. *Line State Symbols*

Terdapat 3 simbol *line state*: Q (*Quiet*) mengindikasikan tidak adanya transisi dalam media, H (*Halt*) mengindikasikan *control sequence* atau perpindahan simbol *code violation*, I (*Idle*) mengindikasikan kondisi normal saat tidak ada transmisi MAC.

2. *Starting Delimiter Symbols*

Digunakan untuk mengindikasikan permulaan urutan pentransmisi data. *Starting Delimiter Symbols* ini akan muncul saat *line state* pada kondisi *idle*, yang menandakan adanya pesan baru.

3. *Embedded Delimiter Symbols*

4. *Data Quartets*

Biasanya digunakan untuk membawa data aktual atau MAC *frame*. Setiap simbol data *quartet* mewakili data *user* 4 bit.

5. *Ending Delimiter*

Berfungsi untuk menghentikan pentransmisi data.

6. *Control Indicator*

Menerangkan kondisi logika tertentu yang berhubungan dengan urutan penstransmisi data dan terjadi pada *conjunction* dengan *ending delimiter*. Terdapat 2 simbol indikator yaitu R (*reset symbol*) menandakan logika 0 atau kondisi *reset*, dan S (*set symbol*) menandakan logika 1 atau kondisi set.

7. *Violation Symbols*

Menunjukkan *invalid code point* yang tidak sesuai dengan simbol yang dijelaskan.

3.10 FDDI Data Link Layer

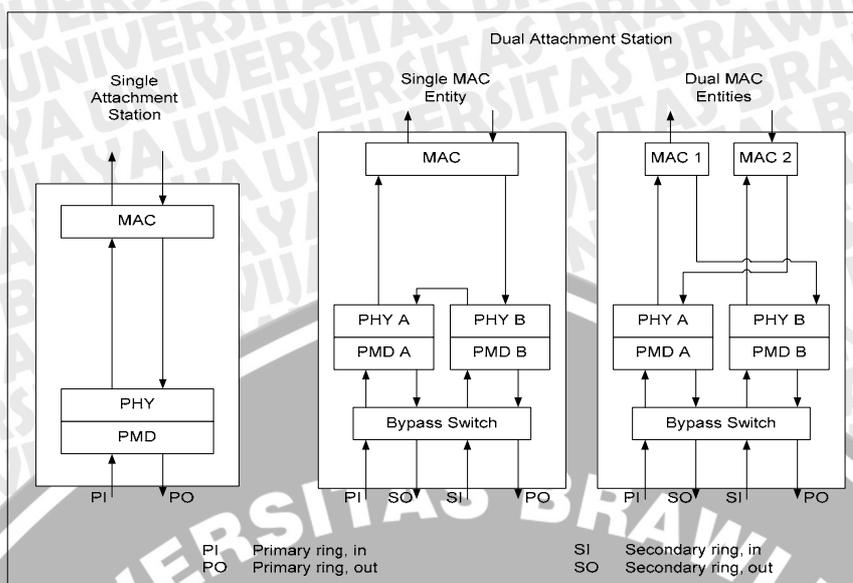
Data Link Layer pada protokol FDDI bertanggung jawab atas *error-free communication* antara *station* yang berdekatan pada jaringan. Kebutuhan ini penting karena tidak ada media transmisi yang bebas *error* secara sempurna. Untuk memenuhi fungsinya tersebut, *data link layer* mempunyai beberapa tugas tertentu yang harus dijalankan antara lain:

1. *Framing* bertugas untuk mengidentifikasi permulaan dan akhir dari penstranmisian.
2. *Addressing* bertugas untuk mengidentifikasi *node* penstransmisi dan penerima data.
3. *Logical Link Establishment and termination* bertugas untuk membuat koneksi logika antara *node* pengirim dan penerima data dan memutuskan koneksi jika diperlukan.
4. *Sequence Control* bertugas memastikan *frame* ditransmisikan dan diterima dalam *sequential order* lewat *logical link*.
5. *Error Detection* bertugas untuk mendeteksi bit *error*.
6. *Error Correction* bertugas untuk memperbaiki *error* transmisi, seperti: bit *error*, *sequence error*, dan *frame* ilegal.

Fungsi *data link layer* pada FDDI dibagi menjadi 2 macam *sublayer* yaitu *Media Access Control* dan *Logical Link Control*.

3.10.1 FDDI MAC

FDDI MAC merupakan *sublayer* yang berada pada bagian bawah protokol FDDI *data link layer*. MAC menyediakan antarmuka antara LLC *layer* dengan FDDI PHY. Gambar 3.16 menunjukkan beberapa contoh dari arsitektur protokol FDDI dalam bentuk *single* dan *dual attachment station*. SAS (*Single-Attachment Station*) hanya dihubungkan pada *ring* primer dan mempunyai satu MAC *entity*. DAS (*Dual-Attachment Station*) dapat menggunakan satu atau dua MAC *entities*.



Gambar 3.16 MAC entities pada konfigurasi SAS dan DAS

Sumber: Gary C. Kessler and David A. A. Train, 1992:136

Standar FDDI MAC membatasi fungsi-fungsi sebagai berikut:

1. Akses yang adil dan sama ke *ring* melalui penggunaan *timed-token protocol*.
2. Komunikasi antara *station* atau peralatan yang terhubung ke *ring* menggunakan *frame* dan *token*.
3. Konstruksi *frame* dan *token*.
4. Mentransmisikan, menerima, mengulang dan memindahkan *frame* dan *token* dari *ring*.
5. Mekanisme pendeteksian *error*.
6. Inisialisasi *ring* dan *ring fault isolation*.

1. PDU (*Protocol Data Unit*)

PDU adalah unit informasi yang di-*transfer* antara *communicating peer* ke *layer processes*. PDU dapat berisi informasi kontrol alamat, data, dan juga informasi pendeteksian *error*. MAC FDDI membatasi dua PDU yaitu *frame* dan *token*. *Frame* membawa MAC *supervisory frame*, *SMT control frame*, atau *LLC frame*. *Token* digunakan untuk kontrol *transfer* media ke *station* yang lain. FDDI memiliki jumlah maksimum *frame* untuk membatasi kemampuan implementasi seperti ukuran *buffer*, kebutuhan *memory*, dan probabilitas bit *error* yang terdeteksi. FDDI memiliki *frame* maksimum sebesar 9000 simbol atau 4500 *byte*.

2. Token

Token merupakan indikasi eksplisit pada *station* dalam jaringan *ring* FDDI, yang menandakan *station* tersebut mempunyai hak untuk mentransmisikan data pada jaringan. *Token* merupakan salah satu MAC PDU yang digunakan untuk layanan paket data. FDDI membatasi dua jenis *token* untuk *asynchronous transmission* yaitu *non-restricted token* dan *restricted token*. *Non-restricted token* dapat diartikan oleh *station* sebagai ijin agar dapat menggunakan *asynchronous bandwidth* yang tersedia untuk pentransmisian data. *Restricted token* digunakan saat sepasang *station* membutuhkan pelayanan seluruh *asynchronous bandwidth* jaringan yang ditujukan untuk *station-to-station dialogue*. FDDI biasanya menggunakan *non-restricted token*, karena menyediakan layanan komunikasi ke *layer* protokol yang lebih tinggi dan hanya digunakan saat ada permintaan dari protokol tersebut.

3. TTRT (*Target Token Rotation Time*)

TTRT diatur pada saat inisialisasi *ring* dan merupakan parameter *ring latency* yang menunjukkan *delay* dalam *ring*. Setiap *station* mengirimkan waktu yang dibutuhkan oleh *token* untuk mengelilingi *ring*. Nilai TTRT diambil dari waktu tercepat yang dibutuhkan oleh *token* untuk berjalan mengelilingi *ring*.

FDDI menggunakan skema alokasi kapasitas berdasarkan pada waktu yang dibutuhkan oleh *token* untuk kembali. Skema ini dibuat untuk mendukung pencampuran transmisi *stream* dan *burst*. Transmisi tersebut melibatkan komunikasi antar *pairs* dari *station*. Terdapat dua jenis *frame* yang dibagi atas *synchronous* dan *asynchronous frame*. Setiap *station* mengalokasikan panjang waktu tertentu untuk dapat mentransmit *frame synchronous*. Interval waktu ini disebut SA (*Synchronous Allocation*). Besar TTRT juga ditentukan bagi jaringan. Nilai TTRT harus besar agar dapat mencukupi penjumlahan waktu pentransmisian *synchronous* seluruh *station* ditambah waktu yang dibutuhkan *frame* maksimum untuk berjalan mengelilingi *ring* [Gary C. Kessler and David A. A. Train, 1992:155].

$$TTRT > (D_Max + F_Max + Token_Time + SA) \quad (3-16)$$

dengan:

TTRT : *Target Token Rotation Times* (s)

D_Max : *latency* maksimum atau *delay* sirkulasi (s)

F_Max : waktu yang dibutuhkan untuk men-transmit *frame* maksimum (s)

$Token_Time$: jumlah waktu yang dibutuhkan untuk men-transmit token (s)

SA : alokasi waktu transmisi *synchronous frame* (s)

4. MAC Timer

MAC FDDI pada semua *station* FDDI harus mempertahankan tiga macam *timer* untuk mengontrol kinerja *ring*. Macam-macam *timer* itu antara lain:

- TRT (*Token-Rotation Timer*)

Digunakan untuk mengawasi jumlah *elapsed time* antara kedatangan *token* berikutnya pada *station* tersebut dan mengontrol akses *asynchronous* menuju *ring* saat beroperasi normal. TRT diinisialisasikan ke TTRT dan tetap memantau jumlah waktu *token* yang datang pada *station*. Jika *token* datang sebelum nilai TRT habis, maka *station* diijinkan untuk mengirim *asynchronous frame* sesuai jumlah waktu tersisa dari TRT. Jika *token* datang setelah nilai TRT habis, maka *station* tidak bisa mengirimkan *asynchronous frame*. Nilai maksimum dari TRT adalah 165 ms.

- THT (*Token-Holding Timer*)

Digunakan untuk menentukan seberapa lama *station* dapat mentransmisikan *asynchronous frame* sesuai jumlah nilai THT. THT diinisialisasi oleh nilai yang berhubungan dengan perbedaan jumlah waktu antara kedatangan *token* dan nilai TTRT. Nilai THT dapat dihitung dengan persamaan [Gary C. Kessler and David A. A. Train, 1992:152]:

$$THT = TTRT - TRT \quad (3-17)$$

- TVX (*Valid-Transmission Timer*)

Digunakan untuk *time out* dan perbaikan kondisi *error transient* dalam *ring* yang dapat menyebabkan kondisi tak aktif. MAC *receiver* akan me-reset *timer* ini jika menerima *valid transmission*. Jika nilai *timer* habis, berarti mengindikasikan *ring* dalam kondisi tak aktif dan MAC akan mengawali prosedur inialisasi *ring* untuk mengembalikan operasi *ring* yang benar. Nilai TVX diperoleh dengan persamaan [Gary C. Kessler and David A. A. Train, 1992:152]:

$$TVX > \max(D_Max, F_Max) + Token_Time + f_Max + S_Min \quad (3-18)$$

dengan:

D_Max : *latency* maksimum atau *delay* sirkulasi (s)

F_Max : waktu untuk mentransmisikan *frame* maksimum (s)

$Token_Time$: waktu untuk mentransmisikan *token* (s)

S_Min : minimum *safety timing tolerance* (s)

Tabel 3.4 Perhitungan parameter *Valid-Transmission Timer*

Timer	Definisi	Nilai
TVX	<i>Valid-Transmission Timer</i>	3.4 ms
D_Max	<i>Maximum Ring Latency</i>	2.661 ms
F_Max	<i>Maximum Frame Transmission</i>	0.361 ms
$Token_Time$	<i>Time required to send a Token and Preamble</i>	0.88 μ s
L_Max	<i>Maximum Transmitter setup time</i>	3.5 μ s
S_Min	<i>Minimum safety timing tolerance</i>	0.3645 μ s

Sumber: Gary C. Kessler and David A. A. Train, 1992:153

3.10.2 LLC (*Logical Link Control*)

LLC mempunyai peranan penting pada jaringan dan interkoneksi LAN. LLC secara spesifik membatasi operasi layanan, format *frame*, struktur pengalamatan dan *sequence control procedures*. Meskipun LLC bukan merupakan bagian dari standar FDDI, tetapi jaringan FDDI membutuhkan LLC untuk operasi *ring* dan pentransmisian data dari pengguna. LLC membatasi layanan *link-level* yang mengizinkan pentransmisian *frame* antara dua *station*. *Frame* digunakan untuk mentransfer informasi antara MAC layer pada FDDI.

3.11 FDDI SMT (*Station Management*)

SMT menyediakan prosedur kontrol yang dibutuhkan oleh *station* atau *node* untuk mengatur proses kerja FDDI sehingga seluruh peralatan dapat bekerja bersama-sama dalam jaringan. SMT harus dapat berkomunikasi dengan seluruh elemen aktif protokol pada *node* FDDI. Fungsi SMT secara umum dapat dibagi atas tiga kategori, yaitu:

- CMT (*Connection Management*), terutama bertanggung jawab atas *establishment* dan *maintenance* dari *physical connection* dan *logical topology* dari jaringan.
- RMT (*Ring Management*), terutama bertanggung jawab untuk *establishment* dan *maintenance* dari kerja *ring*, seperti memastikan *token* mengelilingi jaringan.

- *Operation Management*, yang berhubungan dengan fungsi penting lainnya untuk *maintenance* dan *management* dari jaringan, seperti mengawasi berbagai macam *timer* dan parameter dari masing-masing protokol FDDI.

1. SMT *Service* ke *System Management*

Salah satu tujuan SMT adalah menyediakan fasilitas yang dibutuhkan untuk mengatur FDDI *subnet* dan aplikasi pada lingkungan OSI.

2. CMT (*Connection Management*)

SMT CMT bertanggung jawab atas aspek *physical layer* dan mendukung banyak tipe konfigurasi fisik yang dimungkinkan dalam jaringan FDDI. CMT mengontrol dan mengarahkan pembentukan media *attachment* ke jaringan, koneksi ke *node* lain dalam *ring*, dan konfigurasi *internal port* dan *MAC object* pada *station*. Prosedur dibuat untuk memeriksa dan mengawasi *node-to-node link* agar dapat beroperasi dengan benar, dan jika beroperasi dengan tidak benar maka akan segera dipindahkan dari *ring*.

3. RMT (*Ring Management*)

RMT mengawasi operasi MAC dan mengambil berbagai tindakan yang perlu untuk membantu pembentukan dan mempertahankan operasional *ring*. RMT dalam SMT bertugas untuk memastikan operasi logika yang benar pada *ring*. RMT menerima informasi status dan MAC dan CMT dan melaporkan status dari MAC ke SMT. *Service* yang sering diberikan oleh RMT antara lain:

- *Detection of stuck Beacon*, di sini terdapat *frame* khusus yang disebut *beacon* yang digunakan oleh MAC untuk memberitahu *station* lain bahwa *ring* mengalami kerusakan.
- *Initialization of trace function*, berfungsi vital menghasilkan mekanisme perbaikan untuk kondisi *beacon* dalam *ring* FDDI.
- Pemberitahuan bahwa MAC tersedia untuk pentransmisian data.
- *Detection of duplicagte address*, yang berfungsi untuk menjaga operasi *ring* agar bekerja dengan benar.

3.12 Performansi Jaringan

3.12.1 Delay jaringan FDDI

1. Delay Transmisi

Delay ini tergantung dari jumlah *frame* yang ditransmisikan dari media

transmisi yang digunakan. *Delay* ini tidak tergantung pada *mode transfer* yang digunakan.

$$T_r = \frac{L_{FDDI} \times (8 \text{ bit / byte})}{C} \quad (3-19)$$

dengan:

T_r = *delay transmisi* (s)

L_{FDDI} = panjang *frame* (byte)

C = kecepatan transmisi jaringan (bps)

2. Delay Propagasi

Delay propagasi adalah jumlah waktu yang diperlukan oleh gelombang cahaya untuk berpropagasi di media serat optik. Dalam berpropagasi berkas cahaya akan mengalami rugi-rugi seperti hamburan dan juga rugi pembengkokan sehingga akan mempengaruhi besar *delay* propagasi. Perhitungan rugi hamburan dan pembengkokan sudah termasuk dalam perhitungan besar *delay* propagasi. Nilai standar dari *delay* propagasi pada media serat optik adalah 5,085 $\mu\text{s/km}$ atau 65% kecepatan cahaya yang berpropagasi di dalam ruang hampa [Gary C. Kessler and David A. A. Train, 1992:130]. *Delay* propagasi pada jaringan FDDI dapat dihitung dengan menggunakan persamaan sebagai berikut:

$$T_{Pro} = dxv \quad (3-20)$$

dengan:

T_{Pro} = *delay propagasi* (s)

d = panjang serat optik yang digunakan (km)

v = *delay* propagasi gelombang cahaya pada serat optik (5,085 $\mu\text{s/km}$)

3. Delay paketisasi

Delay paketisasi adalah *delay* yang terjadi pada saat data diubah dalam bentuk paket yang tergantung pada panjang paket dan kecepatan sumber.

$$T_{Paket} = \frac{L_{FDDI} \times (8 \text{ bit / byte})}{C} \quad (3-21)$$

dengan:

T_{Paket} = *delay paketisasi* (s)

L_{FDDI} = panjang *frame* FDDI (byte)

C = kecepatan transmisi jaringan (bps)

4. Delay enkapsulasi

Delay enkapsulasi adalah *delay* yang terjadi saat paket data diberi *header* tambahan. Dalam skripsi ini *header* yang ditambahkan adalah *header label* GMPLS. Besarnya *delay* enkapsulasi adalah:

$$T_{Enkap} = \frac{(W_{frame} - MSS)}{C} \quad (3-22)$$

dengan:

T_{Enkap} = *delay* enkapsulasi (s)

W_{frame} = panjang *frame* (byte)

MSS = panjang segmen maksimum (bit)

C = kecepatan transmisi jaringan (bps)

5. Delay dekapsulasi

Delay dekapsulasi adalah *delay* yang terjadi pada saat pelepasan *label* GMPLS. Besarnya *delay* dekapsulasi adalah:

$$T_{Dekap} = \frac{(W_{frame} + MSS)}{C} \quad (3-23)$$

dengan:

T_{Dekap} = *delay* dekapsulasi (s)

W_{frame} = panjang *frame* (byte)

MSS = panjang segmen maksimum (bit)

C = kecepatan transmisi jaringan (bps)

5. Delay Antrian

Delay antrian (*queing delay*) merupakan nilai yang variabel. Antrian timbul akibat adanya kedatangan pelanggan yang menunggu untuk dilayani. Proses kedatangan pelanggan umumnya acak dengan kecepatan kedatangan adalah: λ (paket/s) dan kecepatan pelayanan adalah μ (paket/s). Berikut akan dijelaskan mengenai antrian yang terjadi dengan mengasumsikan model antrian M/M/1. Pada M pertama menunjukkan distribusi kedatangan Poisson (Markovian), dan M kedua berarti distribusi waktu pelayanan eksponensial dan 1 menunjukkan satu *server*. Disiplin antrian yang

digunakan adalah FIFO (*First In First Out*) maka akan diberikan parameter sebagai berikut:

- Kapasitas *link* adalah C bit per second dan panjang paket data adalah m bit. Besarnya kapasitas *link* akan menentukan kecepatan pelayanan.

$$\mu = \frac{C}{m} \text{ paket/s} \tag{3-24}$$

- Interval waktu tunggu untuk permintaan (*request*) merupakan distribusi *Poisson* dengan kecepatan kedatangan data adalah λ paket/s.

$$T_{Queue} = t_{queue} + t_s \tag{3-25}$$

dengan:

T_{Queue} = total *delay* antrian (s)

t_{queue} = waktu tunggu paket (s)

t_s = rata-rata kecepatan pelayanan (s)

dengan:

$$t_s = \frac{1}{\mu} \tag{3-26}$$

dengan kecepatan kedatangan adalah λ (paket/s) dan kecepatan pelayanan adalah μ (paket/s). Performansi sistem antrian ditunjukkan dalam persamaan U (*utilization*):

$$U = \frac{\lambda}{\mu} \tag{3-27}$$

Kecepatan kedatangan (λ) adalah:

$$\lambda = \frac{1}{G} \tag{3-28}$$

dengan:

G = rata-rata waktu kedatangan *frame* (s/paket)

dengan besarnya rata-rata panjang antrian paket (N) setiap saat pada sistem adalah:

$$N = \frac{\lambda}{\mu - \lambda} \tag{3-29}$$

dan *delay* rata-rata dengan menggunakan teori *Little*, didapatkan:

$$T = \frac{1}{\mu(1 - \rho)} \tag{3-30}$$

delay dari paket data dapat dirumuskan:

$$t_{queue} = T - t_s \tag{3-31}$$

$$t_{queue} = \frac{t_s}{1 - \rho} \tag{3-32}$$

$$t_{queue} = \frac{1/\mu}{1 - \rho} - \frac{1}{\mu} = \frac{\lambda/\mu^2}{1 - \rho} = \frac{\lambda}{\mu(\mu - \lambda)} \tag{3-33}$$

Maka total *delay* antrian T_{Queue} adalah:

$$T_{Queue} = \frac{\lambda/\mu}{\mu - \lambda} + \frac{1}{\mu} \text{ (s)} \tag{3-34}$$

dengan:

T_{Queue} = total *delay* antrian (s)

λ = kecepatan kedatangan *frame* (paket/s)

μ = kecepatan pelayanan *link* serat optik (paket/s)

ρ = ukuran paket data (bit)

6. Ring Latency

Station pada FDDI *token-ring network* diperbolehkan untuk mentransmisikan hanya dalam selang waktu tertentu setelah menerima *token*. Jika *station* mempunyai *frame* tambahan untuk dikirimkan, harus menunggu sampai *token* bersirkulasi mengelilingi *ring*. Jumlah waktu yang dibutuhkan oleh *token* untuk bersirkulasi mengelilingi *ring* disebut sebagai *ring latency*. *Ring latency* ini bergantung pada jumlah bit yang harus disirkulasikan dalam jaringan untuk memastikan kebenaran operasi. Perhitungan *latency* harus mempertimbangkan kontribusi *delay* dari *individual node* dan kabel. *Delay node* adalah komponen *delay* yang terjadi pada *node* jaringan, sedang *delay* kabel merupakan komponen *delay* yang terjadi pada media transmisi karena karakteristik dari media tersebut.

$$Node\ delay = N \times D_s \text{ (s)} \tag{3-35}$$

$$Cable\ delay = L \times D_c \text{ (s)} \tag{3-36}$$

Sehingga besar *ring latency* dapat dihitung dengan persamaan:

$$D = (N \times D_s) + (L \times D_c) \text{ (s)} \tag{3-37}$$

dengan:

D = *ring latency* (s)

N = jumlah *station*

D_s = *delay station* saat mengirim dan menerima bit (s/*station*)

L = panjang *total* serat optik (km)

D_c = *delay* propagasi serat optik (5,085 μ s/km)

7. Efisiensi Akses dan *Delay* Akses Maksimum

Efisiensi akses didefinisikan sebagai presentase waktu yang digunakan oleh jaringan FDDI untuk pentransmisian data. *Delay* akses maksimum adalah nilai maksimum *delay* pada *node* FDDI yang masih dapat mengirimkan data.

Untuk jaringan FDDI dengan *ring latency* D dan nilai TTRT adalah T dan N merupakan jumlah *station*, maka efisiensi dan *delay* akses maksimum dapat dihitung dengan rumus:

$$\text{Efisiensi akses} = \frac{N(T - D)}{NT + D} \times 100\% \quad (3-38)$$

$$\text{Delay akses maksimum} = (N - 1)T + 2D \quad (3-39)$$

dengan:

D = *ring latency* (s)

N = jumlah *station*

T = nilai TTRT (*Target Token Rotation Time*) (s)

8. *Delay* Total

Delay pentransmisian *frame* adalah jumlah seluruh *delay* yang terjadi selama proses proses pentransmisian data lewat *ring* jaringan. *Delay* ini dapat dihitung dengan persamaan sebagai berikut:

$$T_{pf} = T_r + T_{Pro} + T_{Queue} + T_{Paket} + T_{Enkap} + T_{Dekap} \quad (3-40)$$

dengan:

T_{pf} = *delay* pentransmisian *frame* (s)

T_r = *delay* transmisi (s)

T_{Pro} = *delay* propagasi (s)

T_{Queue} = total *delay* antrian (s)

T_{Paket} = *delay* paketisasi (s)

T_{Enkap} = *delay* enkapsulasi (s)

T_{Dekap} = *delay* dekapsulasi (s)

Kemudian kita dapat menghitung *delay* total jaringan yaitu jumlah waktu yang dibutuhkan oleh *station* untuk mentransmisikan data mulai dari waktu yang dibutuhkan

untuk mengakses jaringan, kemudian mentransmisikannya ke *station* lain hingga *frame* kembali lagi ke *station* tersebut. *Delay* total jaringan dihitung berdasarkan rumus sebagai berikut:

$$T_{total} = T_{am} + T_{pf} \quad (3-41)$$

dengan:

T_{total} = *delay* total jaringan (s)

T_{pf} = *delay* pentransmisiian *frame* (s)

T_{am} = *delay* akses maksimum (s)

3.12.2 Bit Rate

Bit rate FDDI dapat dihitung dengan menggunakan persamaan sebagai berikut:

$$\text{Baud Rate atau } D = \frac{N}{T_0} = \frac{N}{1/f} \quad (3-42)$$

$$\text{Bit Rate atau } R = (\log_2 L)D \quad (3-43)$$

dengan:

D = *baud rate* (kecepatan elemen sinyal periodik) (Baud)

R = *bit rate* (bps)

N = *bit word* (bit)

L = jumlah perbedaan elemen sinyal (bit)

T_0 = periodik (s)

f = frekuensi (Hz)

3.12.3 Throughput

Throughput adalah kecepatan maksimum jaringan saat tidak ada data yang hilang pada penstransmisiian. *Throughput* didapat dari pembagian antara jumlah data yang ditransmisikan secara benar dibagi dengan waktu transmisi.

$$\rho_{max} = \frac{1}{t_v} = \frac{(1-p)}{T_r[1+(a-1)p]} \quad (3-44)$$

dengan:

ρ_{max} = *throughput* maksimum (bps)

t_v = waktu transmisi total (s)

p = probabilitas paket data yang salah

T_r = waktu transmisi sebuah paket data (s)

a = konstanta propagasi data pada media serat optik

Konstanta a didapatkan dengan perhitungan sebagai berikut:

$$T_{out} = T_r + \frac{N \cdot b}{R} \quad (3-45)$$

$$a = \frac{T_{out}}{L/R} N \quad (3-46)$$

dengan:

T_{out} = interval waktu yang menunjukkan pengiriman paket data hingga sampai di tujuan (s)

T_r = waktu transmisi *frame* (s)

b = total *delay* proses dari b bit

R = kecepatan transmisi (bit/s)

L = panjang *frame* (bit)

N = jumlah *station*

Pada proses pentransmisian data, data yang dikirim mempunyai karakteristik tertentu, yaitu panjang *frame* pada L bit dengan total dari L' yang digunakan pada bidang sisanya (kontrol pengaman), sehingga probabilitas *frame* yang salah dapat dihitung dengan persamaan sebagai berikut:

$$p = (L + L') p_b \quad (3-47)$$

$$p = W_{frame} \cdot p_b \quad (3-48)$$

dengan:

p = probabilitas *frame error*

p_b = probabilitas bit *error* dalam sistem FDDI

q_b = probabilitas bit benar dalam sistem FDDI

L = panjang *frame* (bit)

L' = panjang *control field frame* (bit)

W_{frame} = panjang *frame* (bit)

BAB IV METODOLOGI PENELITIAN

Tujuan dari penulisan skripsi ini adalah untuk menganalisis penerapan GMPLS pada jaringan FDDI. Untuk menganalisis penerapan GMPLS pada FDDI, metode yang digunakan dalam skripsi ini adalah sebagai berikut:

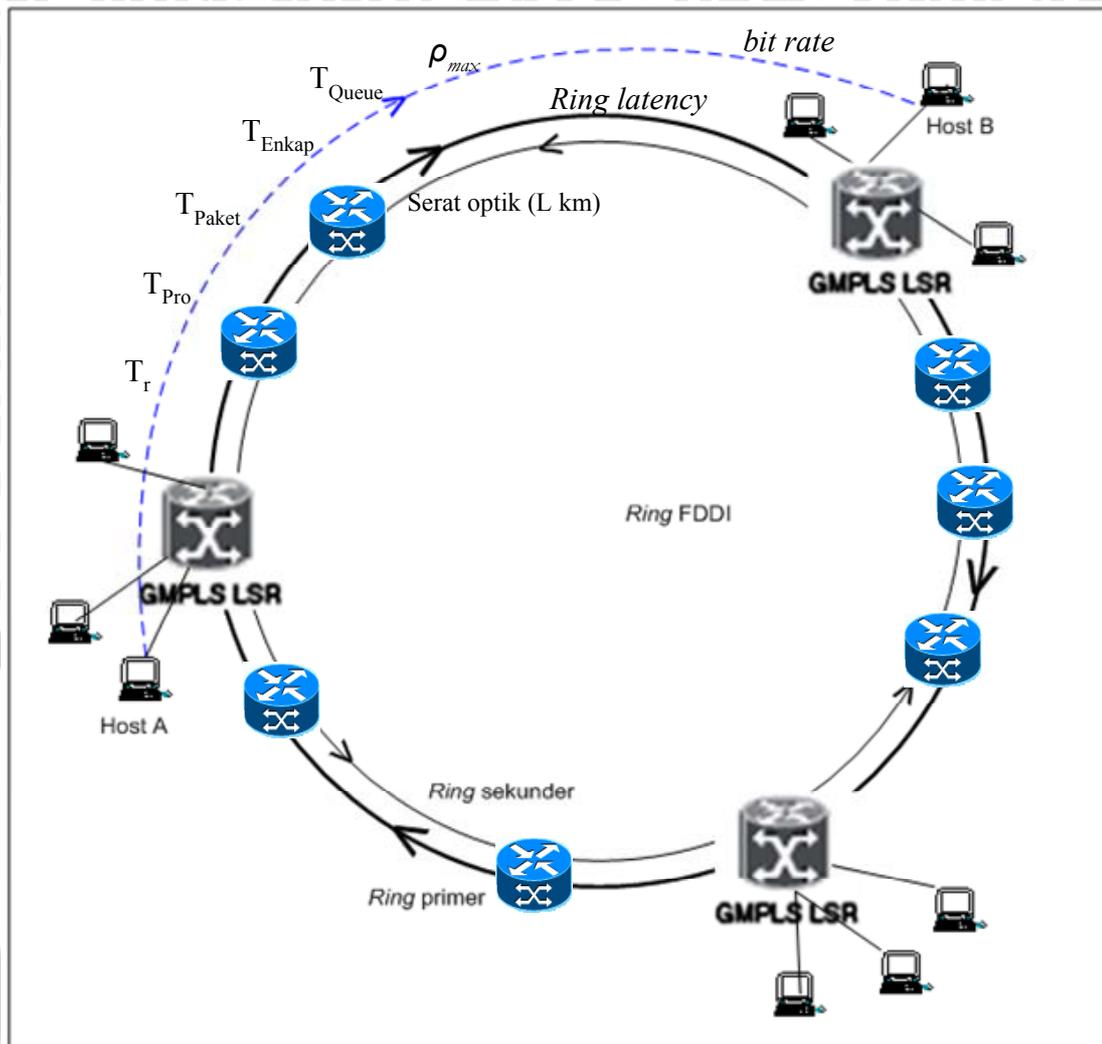
4.1 Pengumpulan dan penyusunan data

Data yang terkumpul dan tersusun merupakan data primer. Data diperoleh diantaranya dari buku-buku referensi, skripsi, *web browsing* dan literatur-literatur lain yang berhubungan dengan karakteristik dasar dari GMPLS dan FDDI.

4.2 Pemodelan Sistem

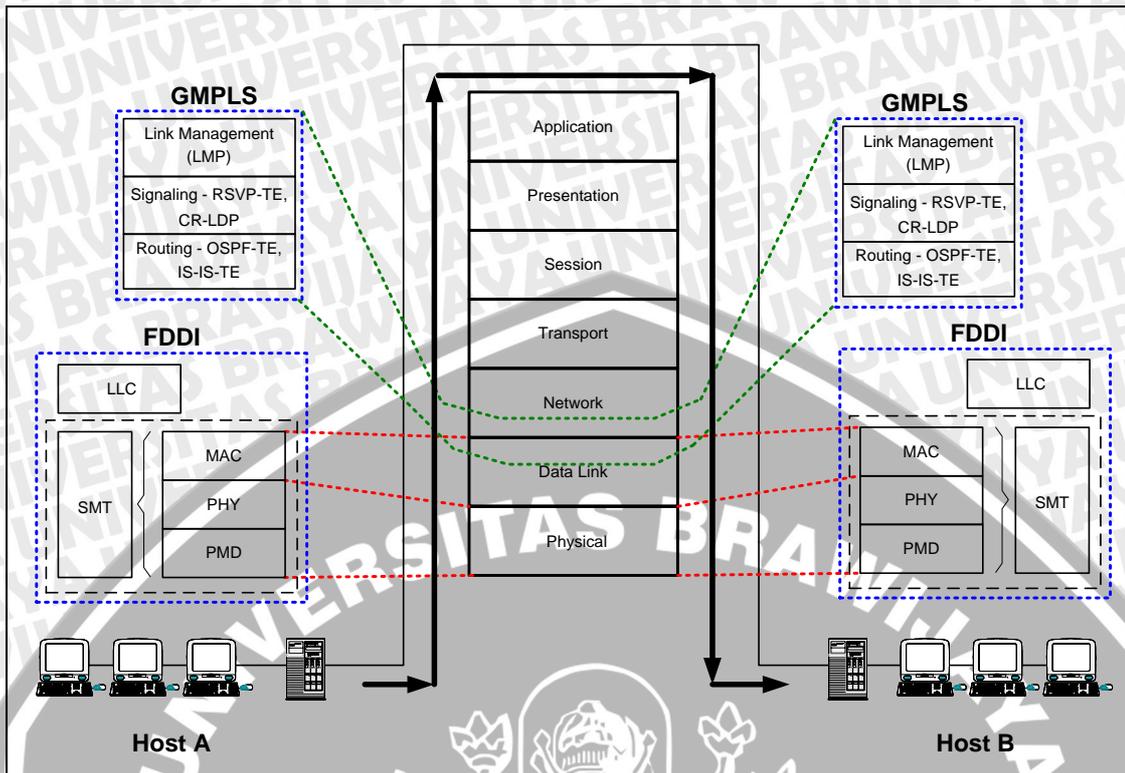
Data-data yang diperoleh tersebut kemudian dianalisis berdasarkan parameter-parameter yang mempengaruhi kinerja penerapan GMPLS pada FDDI.





Gambar 4.1 Konfigurasi Penerapan GMPLS pada jaringan FDDI

Sumber: Analisis



Gambar 4.2 Blok Diagram Penerapan GMPLS pada jaringan FDDI

Sumber: Analisis

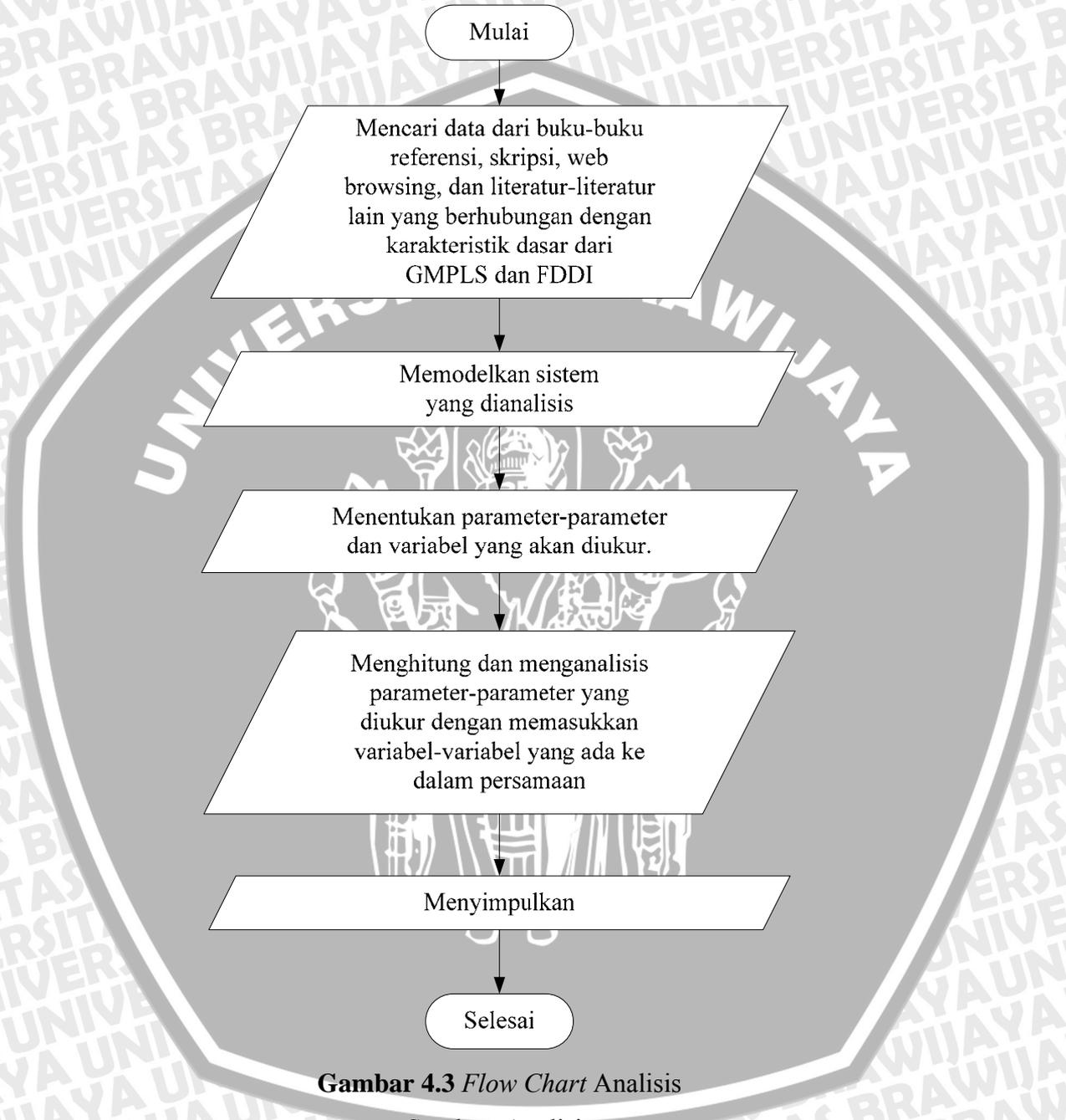
Parameter-parameter yang diukur dalam analisis penerapan GMPLS pada jaringan FDDI dalam skripsi ini adalah *Link Budget* dan *Rise-Time Budget* media transmisi serat optik, *bit error rate* (BER), *bit rate*, *throughput* (ρ_{max}), *ring latency*, efisiensi akses dan *delay* akses maksimum, dan *delay* jaringan FDDI (*delay* transmisi (T_r), *delay* propagasi (T_{Pro}), *delay* paketisasi (T_{Paket}), *delay* enkapsulasi (T_{Enkap}), *delay* antrian (T_{Queue})). Untuk dapat mengukur parameter-parameter tersebut perlu ditentukan terlebih dahulu variabel-variabel yang mempengaruhi nilai dari tiap-tiap parameter yang diukur. Variabel-variabel tersebut antara lain: sensitivitas *transceiver*, rugi-rugi dan redaman yang terjadi, t_{TX} dan t_{RX} dari *transceiver* dan dispersi total dari serat optik, waktu transmisi total (t_v), probabilitas paket salah (p), konstanta propagasi data pada media serat optik (a), interval waktu yang menunjukkan pengiriman paket data hingga sampai di tujuan (T_{out}), total *delay* akses dari b bit data (b), kecepatan transmisi (R), panjang *frame* (L), jumlah *station* (N), periode bit (T).

4.3 Analisis

Data-data yang diperoleh kemudian dianalisis berdasarkan persamaan-

persamaan dan urutan-urutan pengerjaan yang telah ditentukan.

4.3.1 Flow Chart Analisis

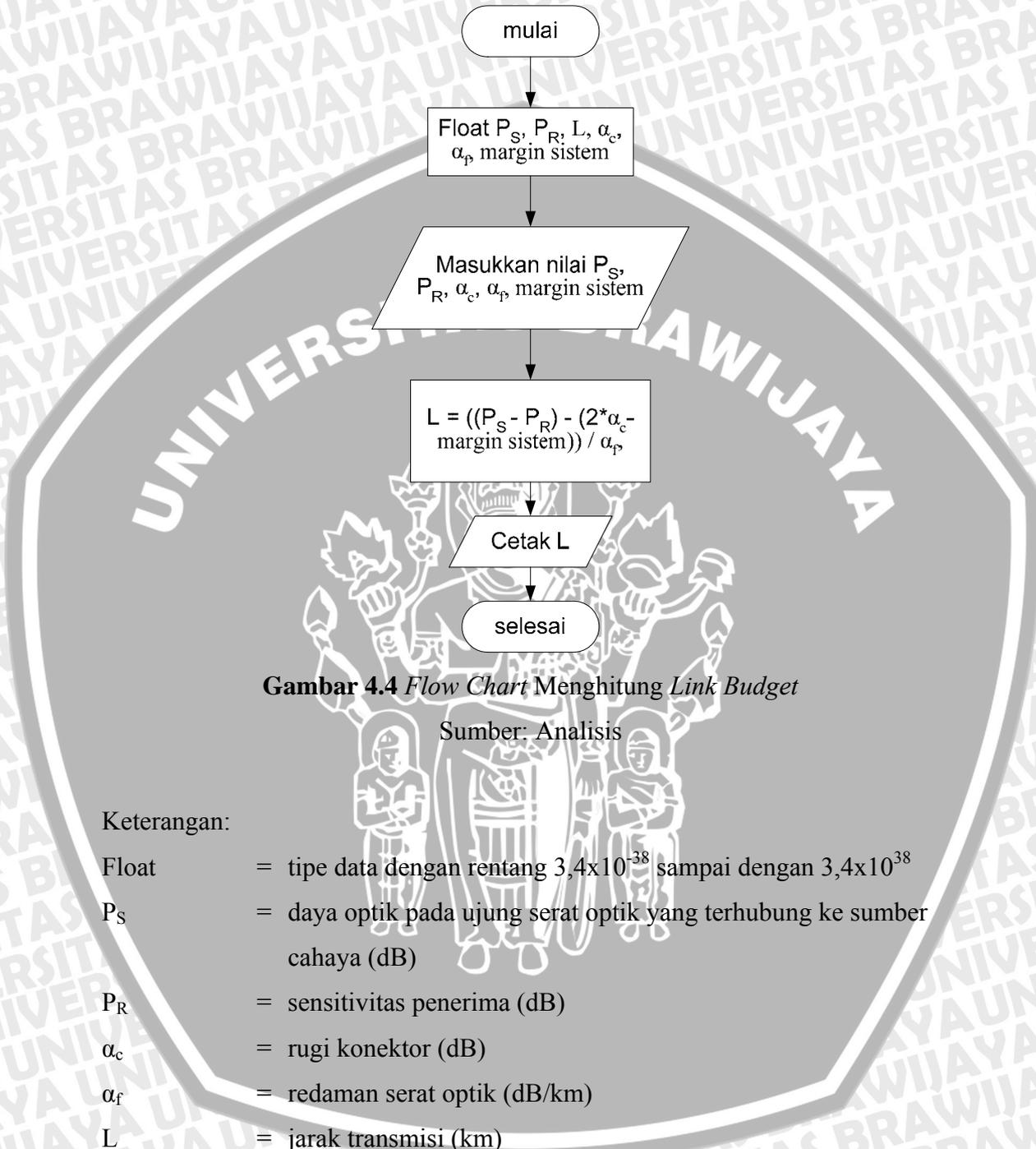


Gambar 4.3 Flow Chart Analisis

Sumber: Analisis

4.3.2 Flow Chart Program

1. Link Budget



Gambar 4.4 Flow Chart Menghitung Link Budget

Sumber: Analisis

Keterangan:

- Float = tipe data dengan rentang $3,4 \times 10^{-38}$ sampai dengan $3,4 \times 10^{38}$
- P_S = daya optik pada ujung serat optik yang terhubung ke sumber cahaya (dB)
- P_R = sensitivitas penerima (dB)
- α_c = rugi konektor (dB)
- α_f = redaman serat optik (dB/km)
- L = jarak transmisi (km)
- margin sistem = umumnya diberikan 6 dB



2. Rise-Time Budget



Gambar 4.5 Flow Chart Menghitung Rise-Time Budget

Sumber: Analisis

Keterangan:

Float = tipe data dengan rentang $3,4 \times 10^{-38}$ sampai dengan $3,4 \times 10^{38}$

t_{TX} = bergantung dari sumber cahaya dan rangkaian *drive*-nya (s)

t_{RX} = bergantung dari sumber cahaya dan rangkaian *drive*-nya (s)

D = dispersi total serat optik (s)

σ_{λ} = lebar spektral sumber cahaya (nm)

L = panjang *link* (m)

3. *delay* jaringan FDDI dan GMPLS

a. *delay* transmisi (T_r)



Gambar 4.6 *Flow Chart* *delay* transmisi (T_r)

Sumber: Analisis

Keterangan:

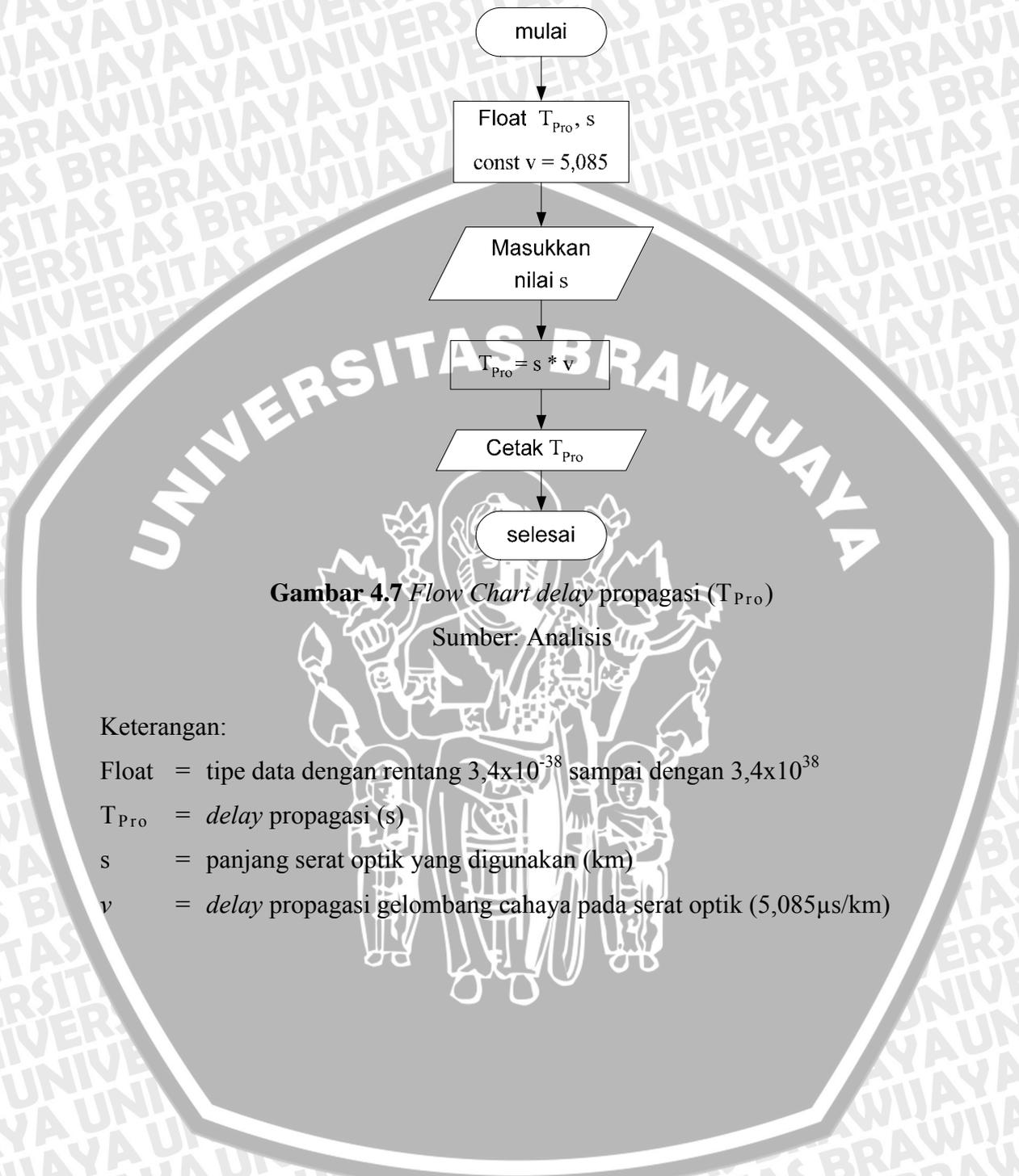
Float = tipe data dengan rentang $3,4 \times 10^{-38}$ sampai dengan $3,4 \times 10^{38}$

T_r = *delay* transmisi (s)

L = panjang *frame* (byte)

C = kecepatan akses (bit/s)

b. *delay* propagasi (T_{Pro})



Gambar 4.7 *Flow Chart* *delay* propagasi (T_{Pro})

Sumber: Analisis

Keterangan:

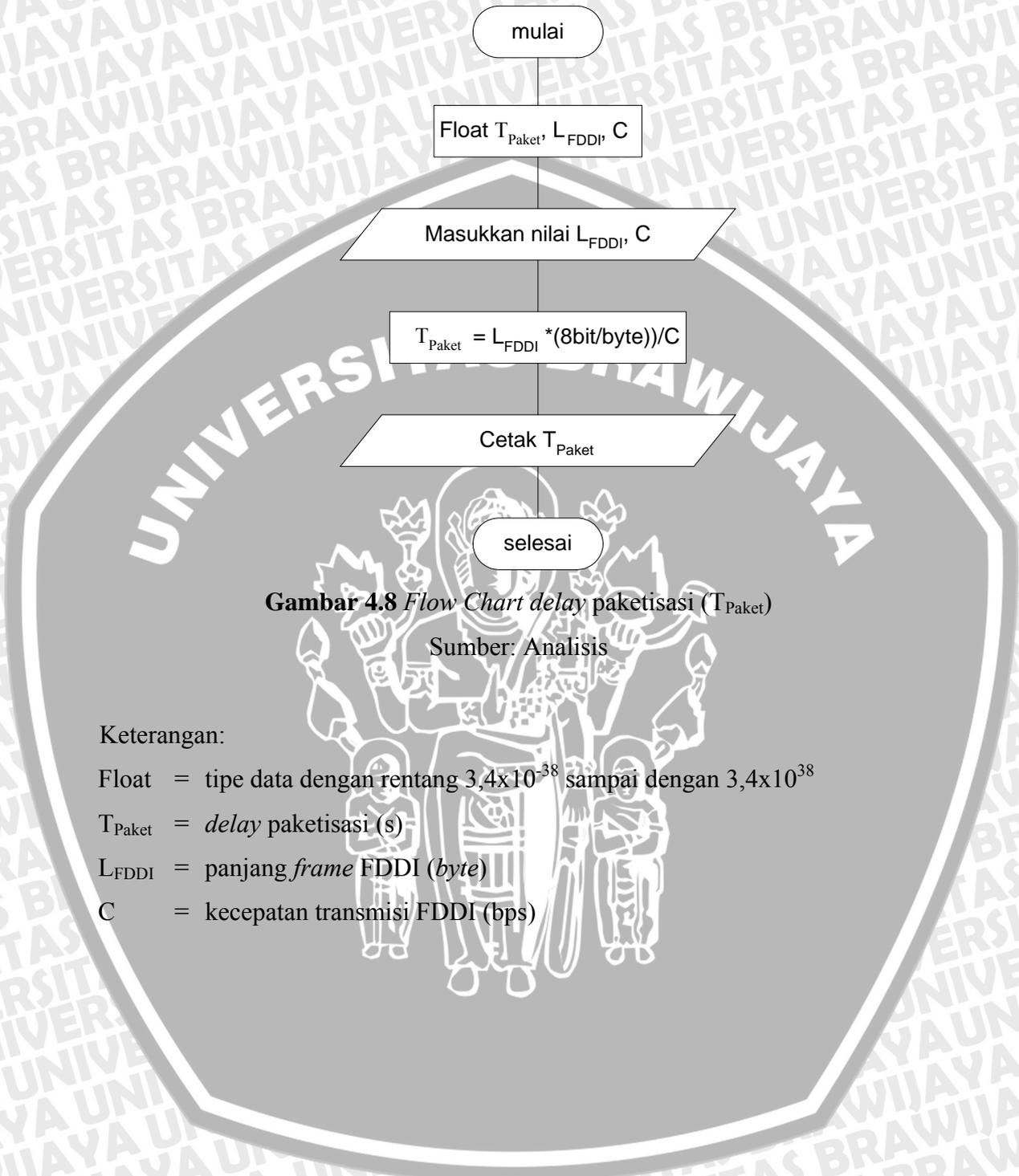
Float = tipe data dengan rentang $3,4 \times 10^{-38}$ sampai dengan $3,4 \times 10^{38}$

T_{Pro} = *delay* propagasi (s)

s = panjang serat optik yang digunakan (km)

v = *delay* propagasi gelombang cahaya pada serat optik ($5,085 \mu\text{s}/\text{km}$)

c. *delay* paketisasi (T_{Paket})



Gambar 4.8 *Flow Chart delay* paketisasi (T_{Paket})

Sumber: Analisis

Keterangan:

Float = tipe data dengan rentang $3,4 \times 10^{-38}$ sampai dengan $3,4 \times 10^{38}$

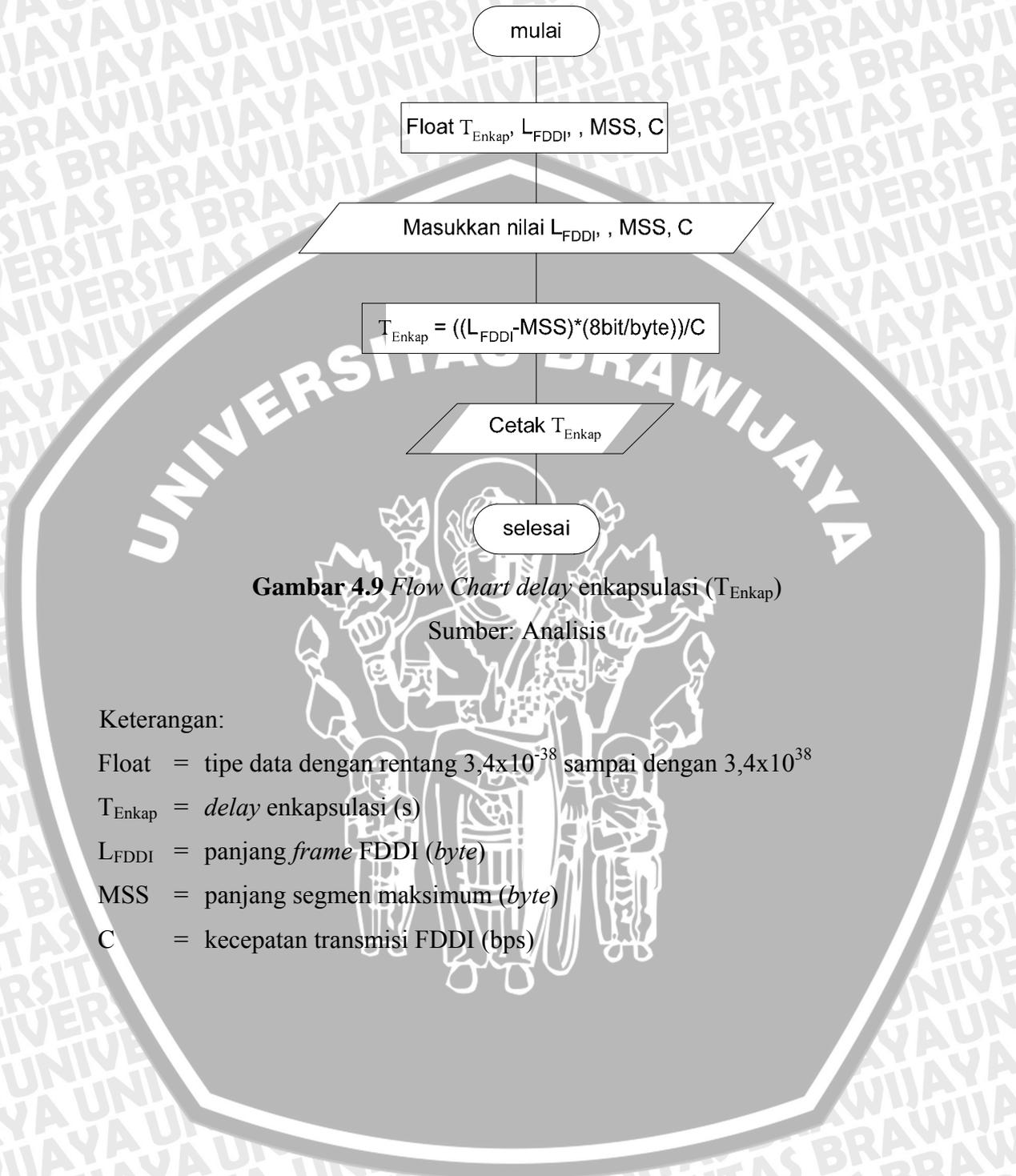
T_{Paket} = *delay* paketisasi (s)

L_{FDDI} = panjang *frame* FDDI (*byte*)

C = kecepatan transmisi FDDI (bps)



d. *delay* enkapsulasi (T_{Enkap})



Gambar 4.9 *Flow Chart* *delay* enkapsulasi (T_{Enkap})

Sumber: Analisis

Keterangan:

Float = tipe data dengan rentang $3,4 \times 10^{-38}$ sampai dengan $3,4 \times 10^{38}$

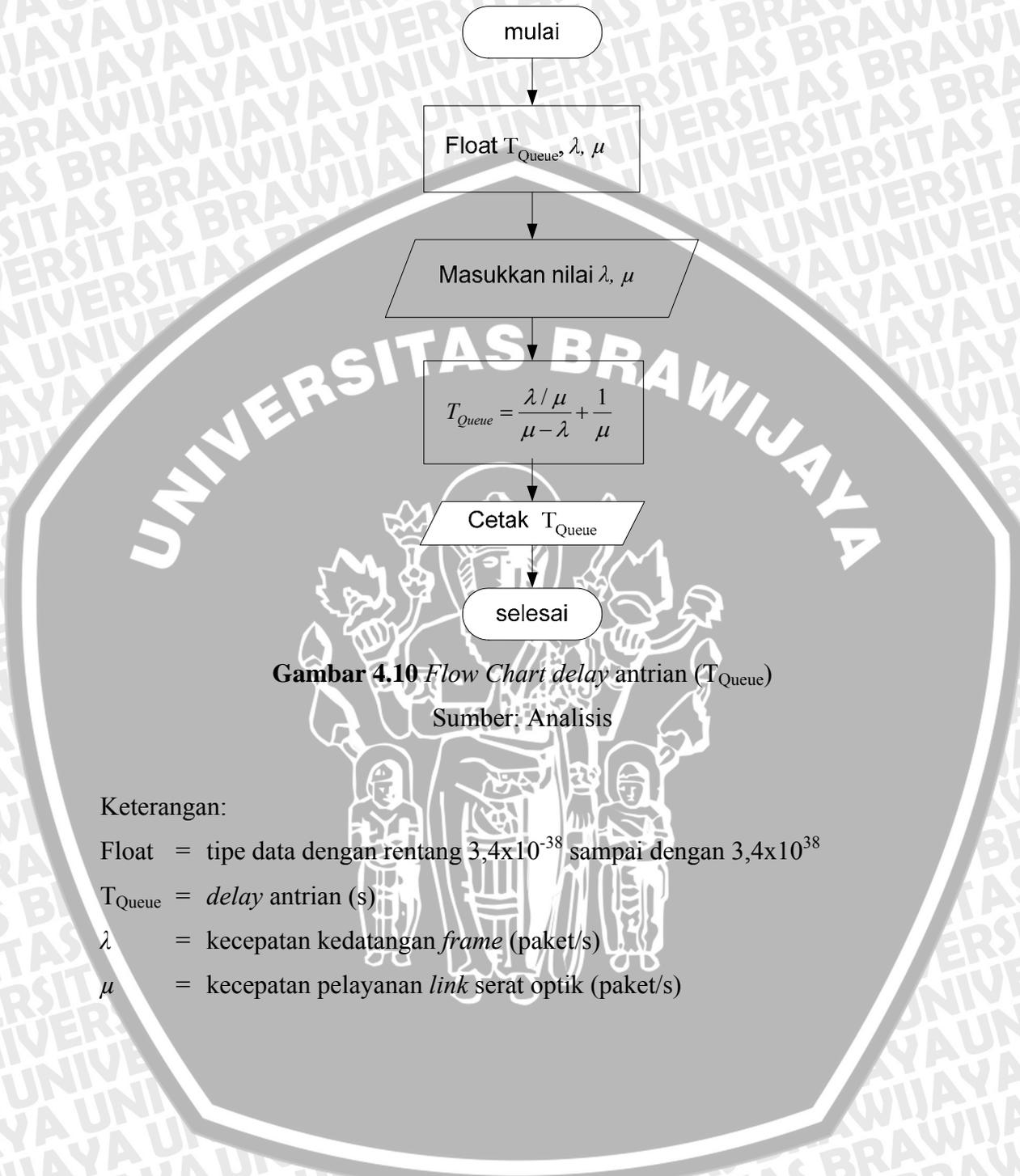
T_{Enkap} = *delay* enkapsulasi (s)

L_{FDDI} = panjang *frame* FDDI (*byte*)

MSS = panjang segmen maksimum (*byte*)

C = kecepatan transmisi FDDI (bps)

e. *delay* antrian (T_{Queue})



Gambar 4.10 *Flow Chart* *delay* antrian (T_{Queue})

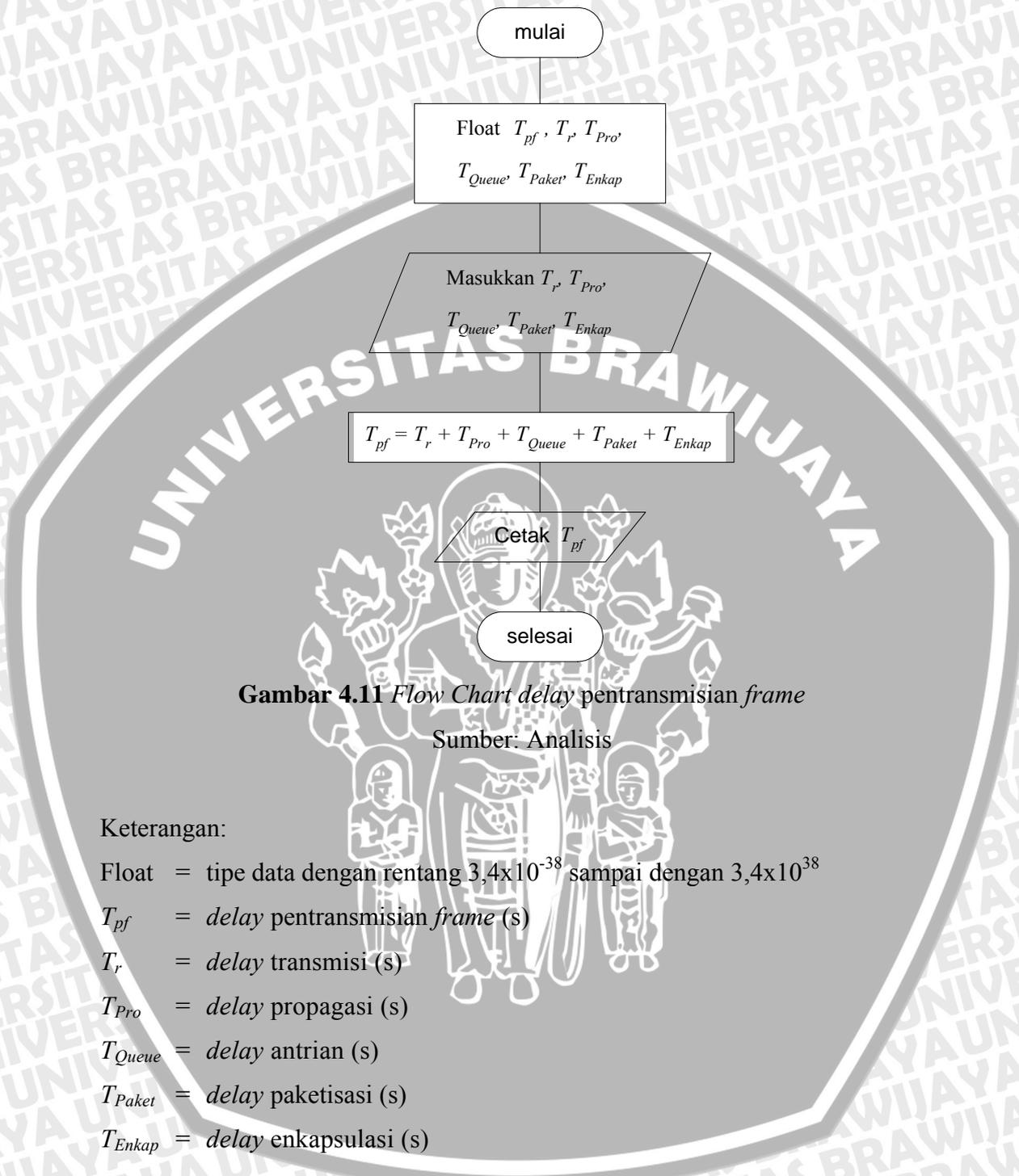
Sumber: Analisis

Keterangan:

- Float = tipe data dengan rentang $3,4 \times 10^{-38}$ sampai dengan $3,4 \times 10^{38}$
- T_{Queue} = *delay* antrian (s)
- λ = kecepatan kedatangan *frame* (paket/s)
- μ = kecepatan pelayanan *link* serat optik (paket/s)



f. *delay* pentransmisian *frame* (T_{pf})



Gambar 4.11 *Flow Chart* *delay* pentransmisian *frame*

Sumber: Analisis

Keterangan:

Float = tipe data dengan rentang $3,4 \times 10^{-38}$ sampai dengan $3,4 \times 10^{38}$

T_{pf} = *delay* pentransmisian *frame* (s)

T_r = *delay* transmisi (s)

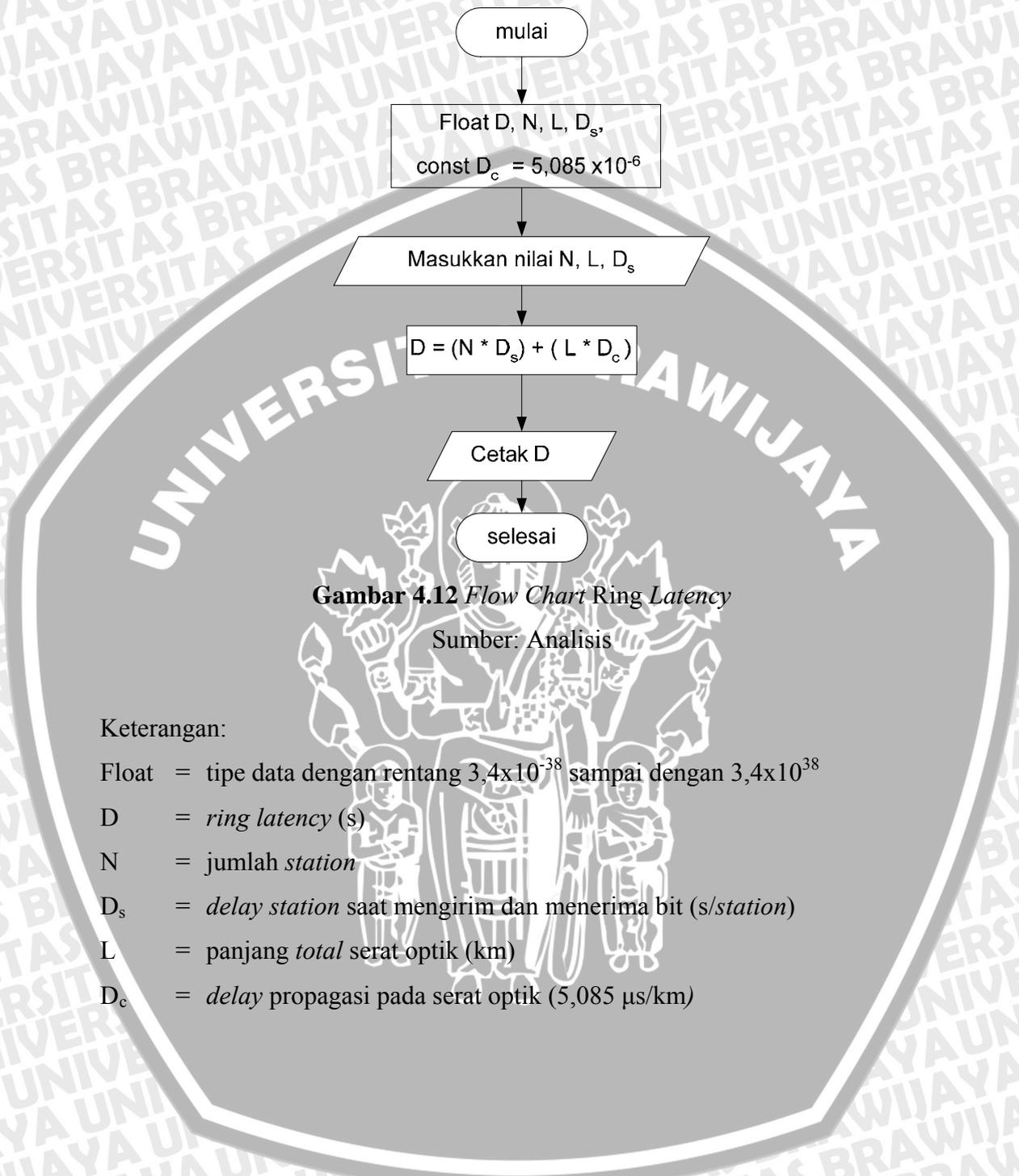
T_{Pro} = *delay* propagasi (s)

T_{Queue} = *delay* antrian (s)

T_{Paket} = *delay* paketisasi (s)

T_{Enkap} = *delay* enkapsulasi (s)

4. ring latency



Gambar 4.12 Flow Chart Ring Latency

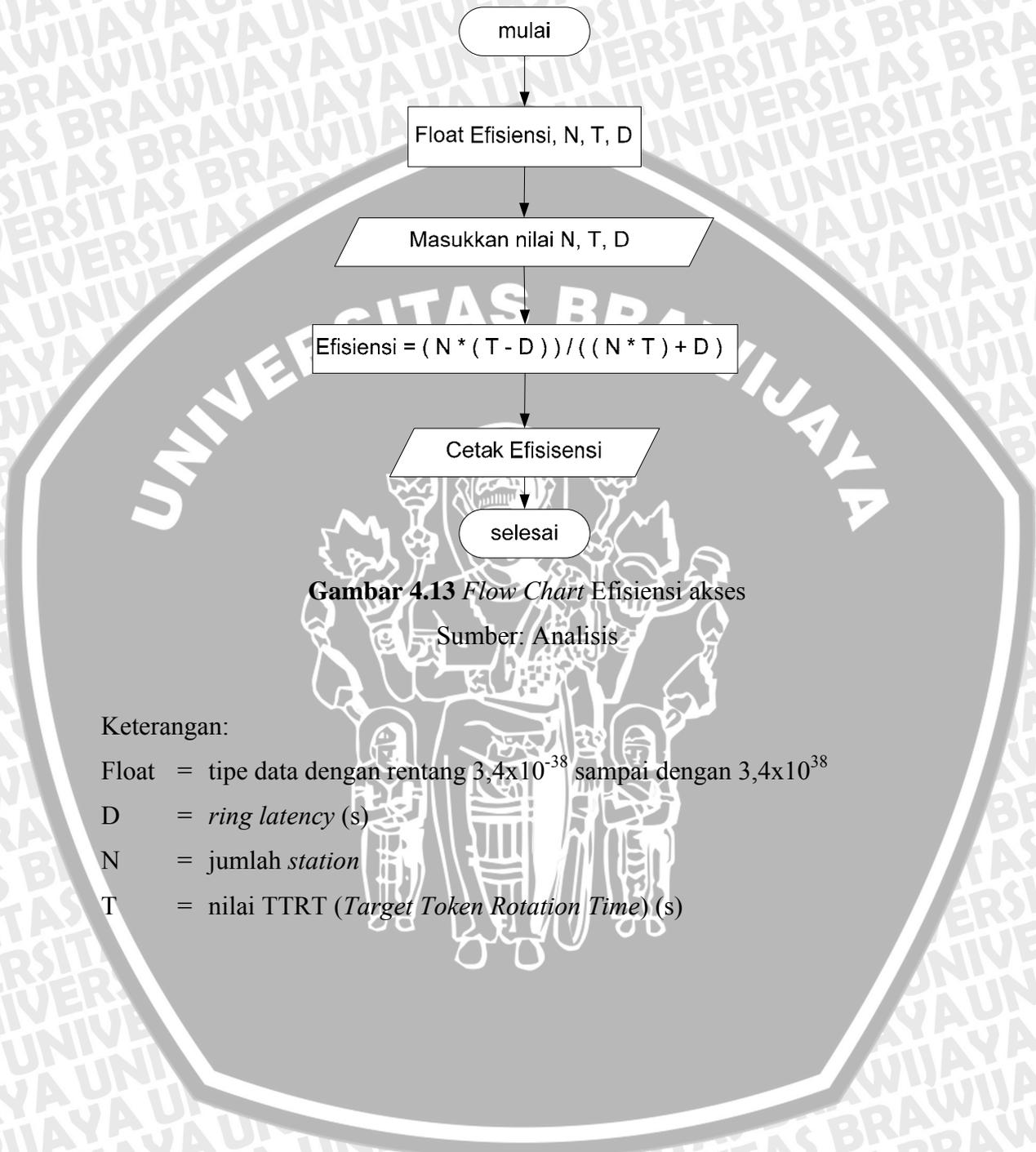
Sumber: Analisis

Keterangan:

- Float = tipe data dengan rentang $3,4 \times 10^{-38}$ sampai dengan $3,4 \times 10^{38}$
- D = ring latency (s)
- N = jumlah station
- D_s = delay station saat mengirim dan menerima bit (s/station)
- L = panjang total serat optik (km)
- D_c = delay propagasi pada serat optik (5,085 μs/km)

5. Efisiensi akses dan *Delay* akses maksimum

Efisiensi akses



Gambar 4.13 *Flow Chart* Efisiensi akses

Sumber: Analisis

Keterangan:

Float = tipe data dengan rentang $3,4 \times 10^{-38}$ sampai dengan $3,4 \times 10^{38}$

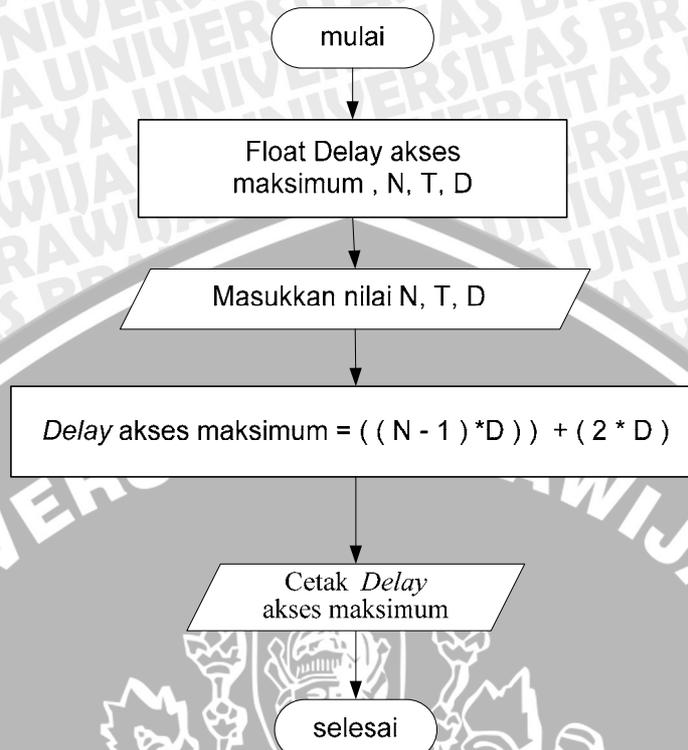
D = *ring latency* (s)

N = jumlah *station*

T = nilai TTRT (*Target Token Rotation Time*) (s)



Delay akses maksimum



Gambar 4.14 Flow Chart Delay akses maksimum

Sumber: Analisis

Keterangan:

Float = tipe data dengan rentang $3,4 \times 10^{-38}$ sampai dengan $3,4 \times 10^{38}$

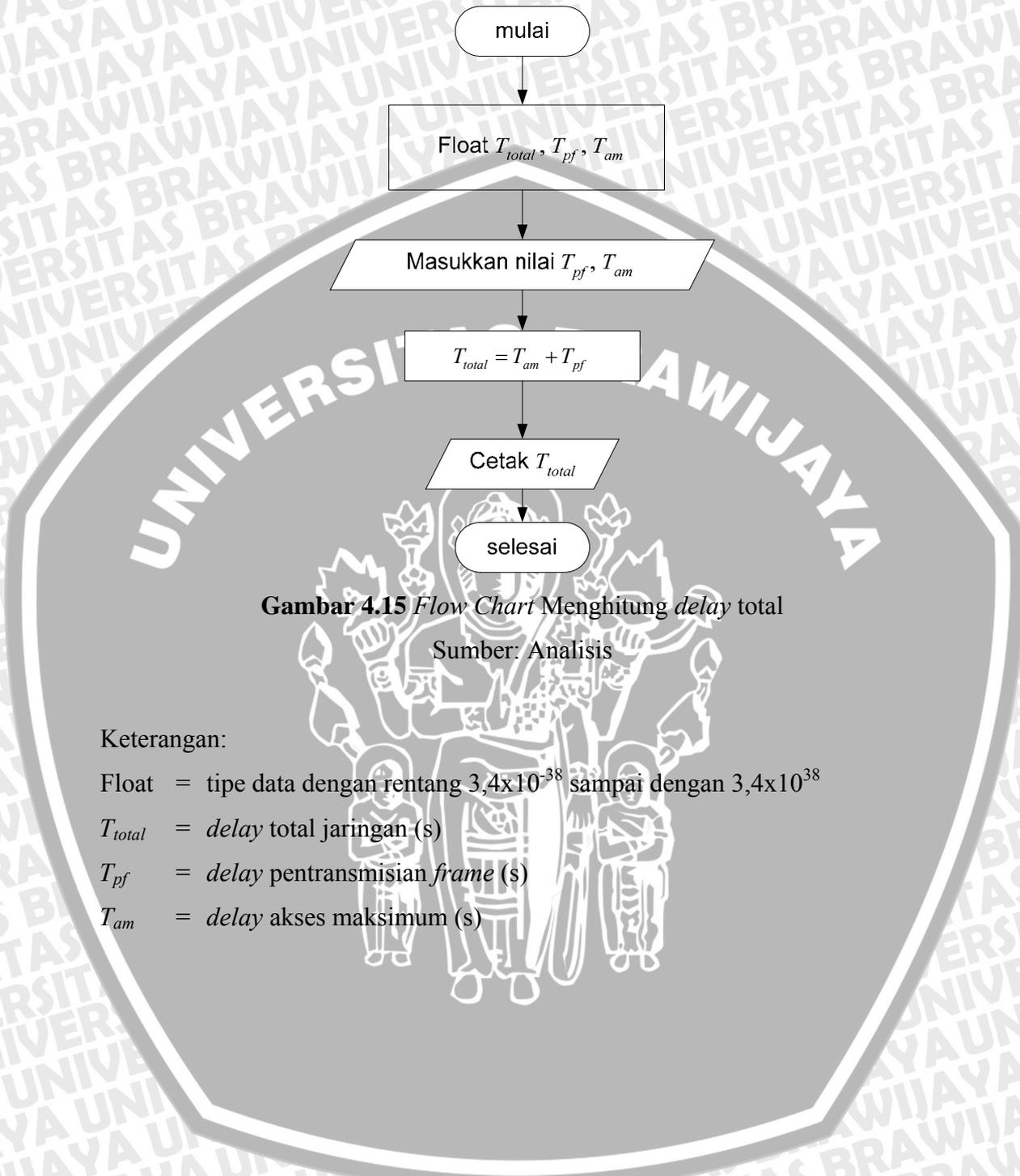
Delay akses maksimum = nilai maksimum delay pada node FDDI yang masih dapat mengirimkan data (s)

D = ring latency (s)

N = jumlah station

T = nilai TTRT (Target Token Rotation Time) (s)

6. Delay total



Gambar 4.15 Flow Chart Menghitung delay total

Sumber: Analisis

Keterangan:

Float = tipe data dengan rentang $3,4 \times 10^{-38}$ sampai dengan $3,4 \times 10^{38}$

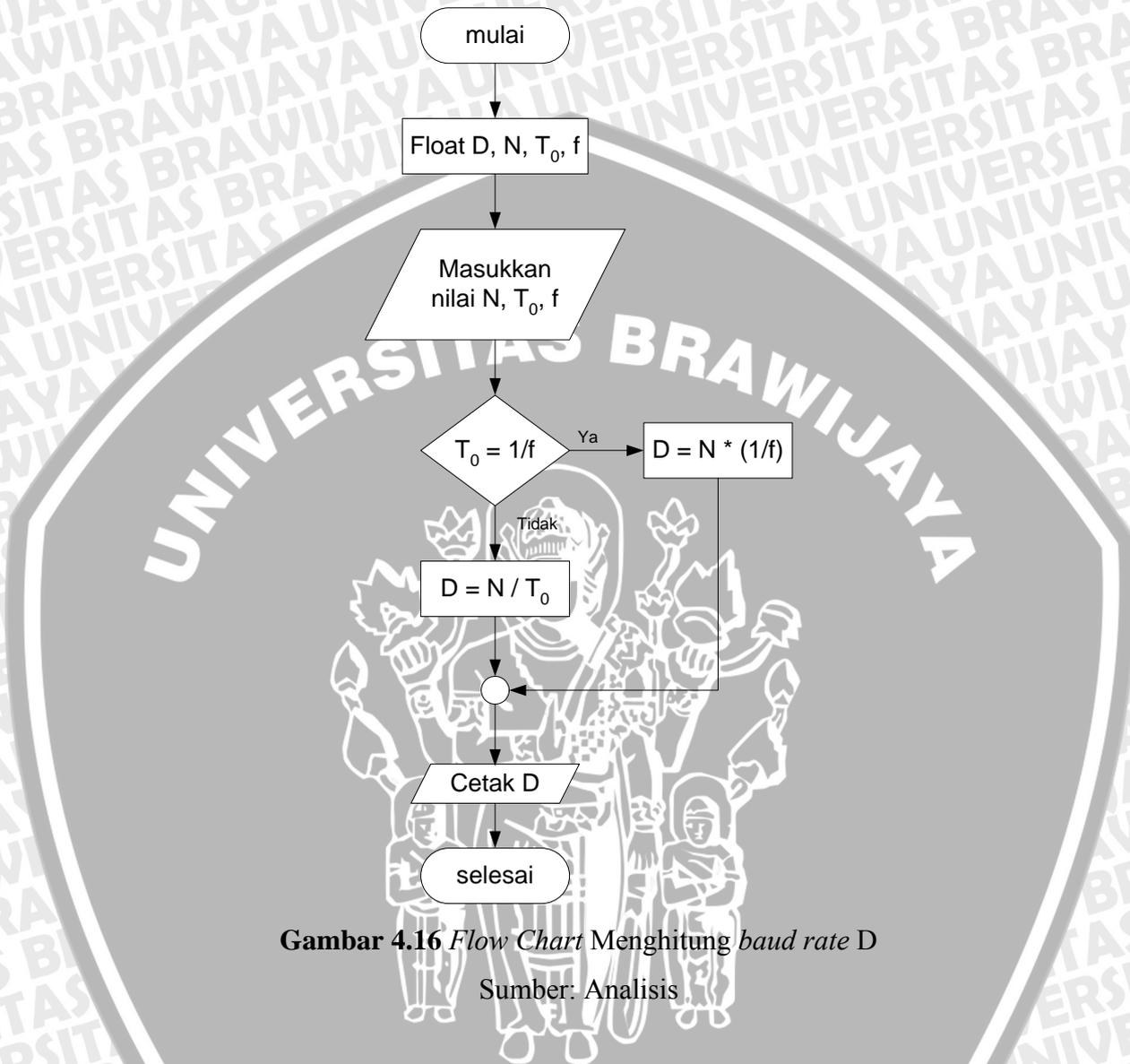
T_{total} = delay total jaringan (s)

T_{pf} = delay pentransmisian frame (s)

T_{am} = delay akses maksimum (s)

7. bit rate

Menghitung baud rate D



Gambar 4.16 Flow Chart Menghitung baud rate D

Sumber: Analisis

Keterangan:

Float = tipe data dengan rentang $3,4 \times 10^{-38}$ sampai dengan $3,4 \times 10^{38}$

D = baud rate (kecepatan elemen sinyal periodik) (Baud)

R = bit rate (bps)

N = bit words (bit)

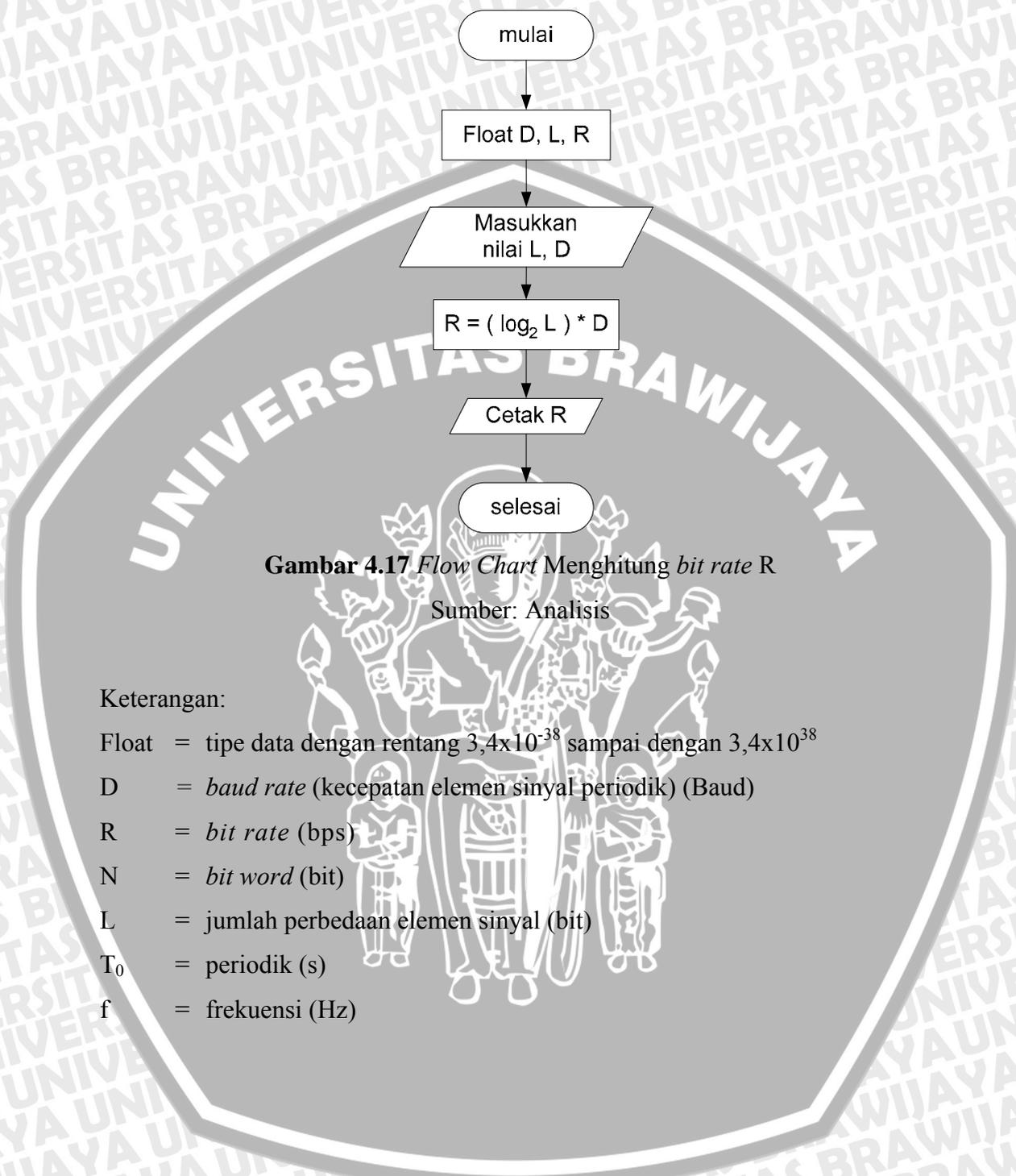
L = jumlah perbedaan elemen sinyal (bit)

T_0 = periodik (s)

f = frekuensi (Hz)



Menghitung *bit rate* R



Gambar 4.17 Flow Chart Menghitung *bit rate* R

Sumber: Analisis

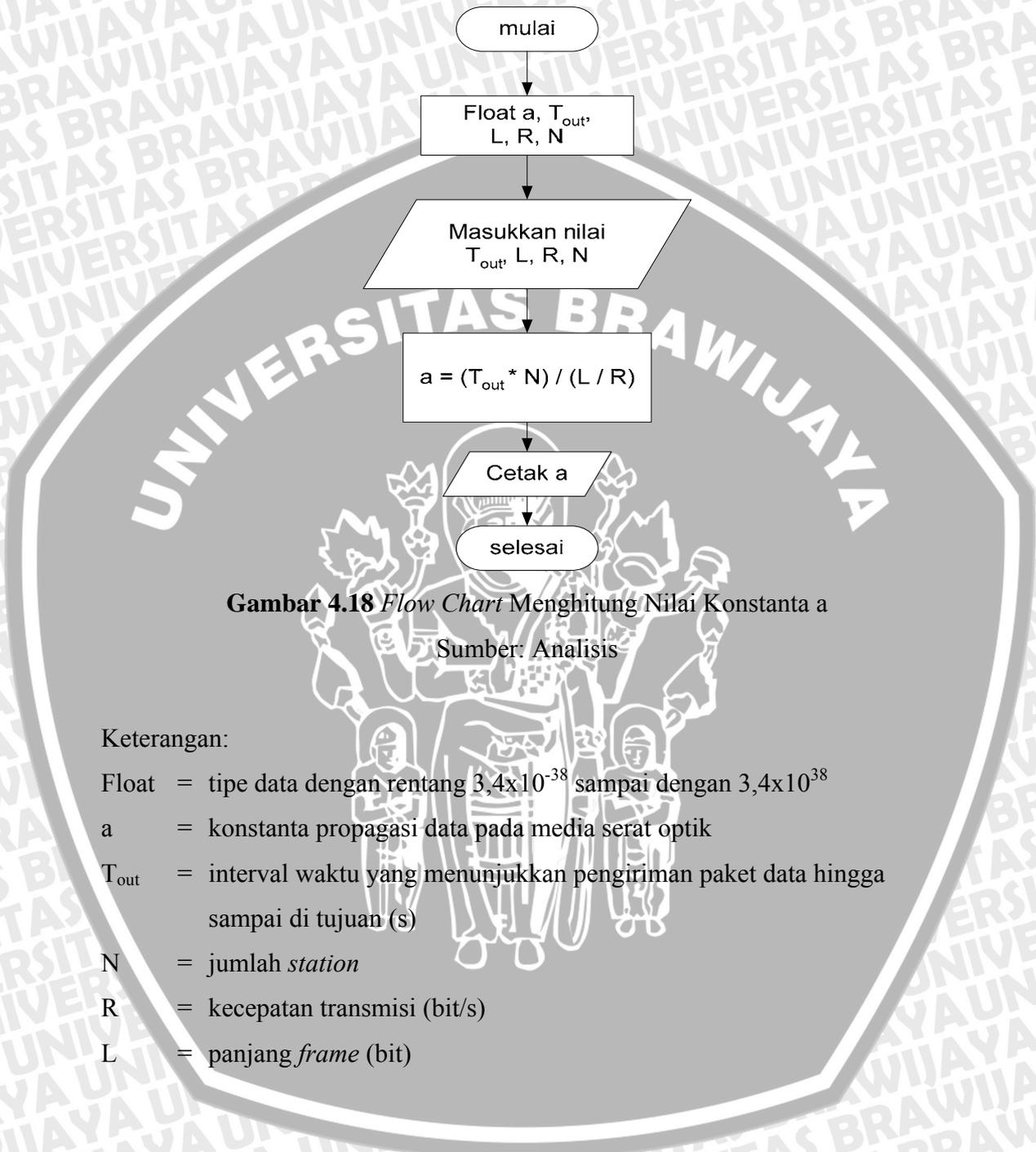
Keterangan:

- Float = tipe data dengan rentang $3,4 \times 10^{-38}$ sampai dengan $3,4 \times 10^{38}$
- D = *baud rate* (kecepatan elemen sinyal periodik) (Baud)
- R = *bit rate* (bps)
- N = *bit word* (bit)
- L = jumlah perbedaan elemen sinyal (bit)
- T_0 = periodik (s)
- f = frekuensi (Hz)



8. *throughput* (ρ_{max})

Menghitung nilai konstanta a



Gambar 4.18 Flow Chart Menghitung Nilai Konstanta a

Sumber: Analisis

Keterangan:

Float = tipe data dengan rentang $3,4 \times 10^{-38}$ sampai dengan $3,4 \times 10^{38}$

a = konstanta propagasi data pada media serat optik

T_{out} = interval waktu yang menunjukkan pengiriman paket data hingga sampai di tujuan (s)

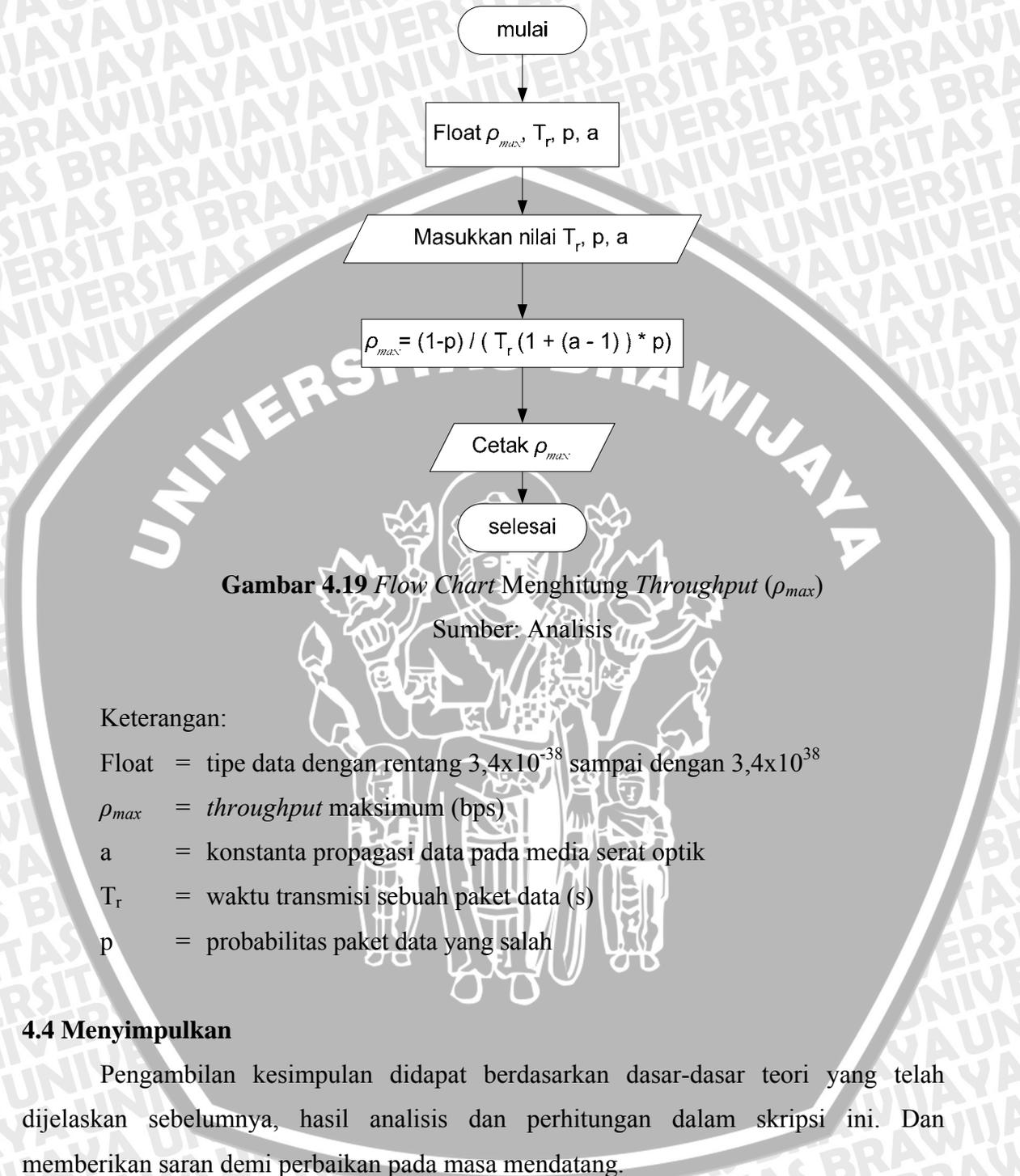
N = jumlah *station*

R = kecepatan transmisi (bit/s)

L = panjang *frame* (bit)



Menghitung *throughput* (ρ_{max})



Gambar 4.19 Flow Chart Menghitung Throughput (ρ_{max})

Sumber: Analisis

Keterangan:

Float = tipe data dengan rentang $3,4 \times 10^{-38}$ sampai dengan $3,4 \times 10^{38}$

ρ_{max} = *throughput* maksimum (bps)

a = konstanta propagasi data pada media serat optik

T_r = waktu transmisi sebuah paket data (s)

p = probabilitas paket data yang salah

4.4 Menyimpulkan

Pengambilan kesimpulan didapat berdasarkan dasar-dasar teori yang telah dijelaskan sebelumnya, hasil analisis dan perhitungan dalam skripsi ini. Dan memberikan saran demi perbaikan pada masa mendatang.

BAB V

ANALISIS PENERAPAN TEKNOLOGI GMPLS (*Generalized Multi-protocol Label Switching*) PADA FDDI (*Fiber Distributed Data Interface*)

Analisis yang dilakukan pada penerapan teknologi GMPLS pada jaringan FDDI adalah untuk mengevaluasi parameter-parameter kinerja operasi kerja jaringan FDDI. Parameter diasumsikan untuk memudahkan evaluasi kinerja sistem sebagai berikut:

1. Media transmisi yang digunakan dan dianalisis adalah serat optik *multi-mode graded index* yang bekerja pada panjang gelombang sekitar 1300 nm.
2. *Delay* propagasi pada serat optik adalah 5,085 $\mu\text{s}/\text{km}$ dan kecepatan propagasi cahaya dalam serat optik adalah 3×10^8 m/s.
3. Nilai *Target Token Rotation Time* (TTRT) yang digunakan adalah 4 ms dan 8 ms.
4. Panjang transmisi data adalah 100 km.
5. Jumlah *node* yang dianalisis sebanyak 100 *node* dengan jarak antar *node* adalah sama.

5.1 Analisis Media Transmisi

Dua analisis yang biasa dilakukan pada media transmisi serat optik untuk meyakinkan bahwa performansi sistem yang diinginkan dapat dicapai adalah: *Link Budget* dan *Rise-time Budget*.

5.1.1 Analisis *Link Budget*

Diasumsikan bahwa kabel dengan panjang L km mempunyai konektor hanya pada ujung-ujungnya. Rugi *splice* digabungkan dalam rugi kabel (margin sistem) untuk penyederhanaan. *Transceiver* yang digunakan memiliki daya keluaran -16 dBm dan memiliki sensitivitas penerimaan daya -33 dBm.

$$L_T = P_S - P_R$$

$$L_T = -16 - (-33)$$

$$L_T = 17 \text{ dB}$$

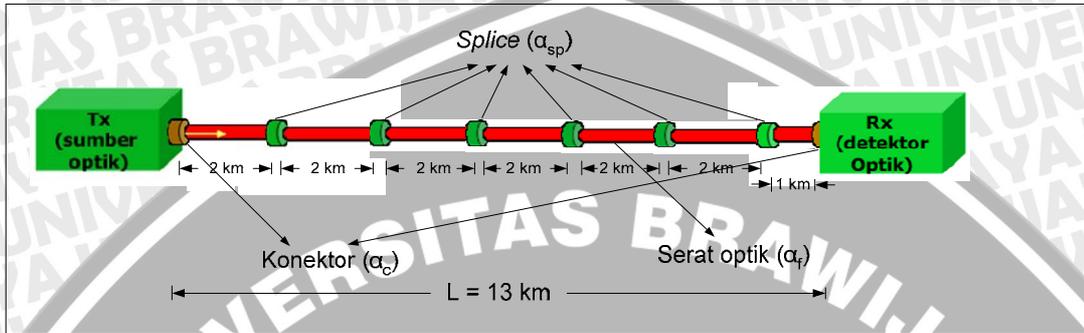
Untuk rugi konektor sebesar 0,3 dB, redaman serat optik sebesar 0,8 dB/km, maka panjang satu *link* maksimum (tanpa *repeater*) adalah:

$$L_T = 2\alpha_c + \alpha_f L + \text{margin sistem}$$

$$17 = 2(0,3) + 0,8.L + 6$$

$$L = \frac{(17 - 6,6)}{0,8}$$

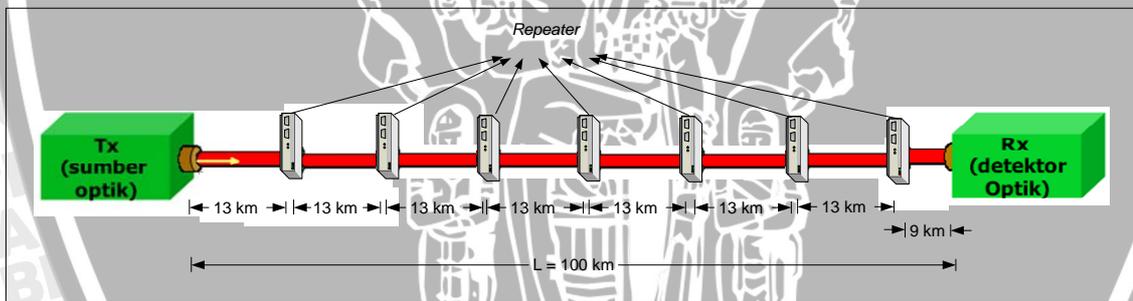
$$L = 13 \text{ km}$$



Gambar 5.1 Link Budget transceiver

Sumber: Hasil perhitungan

Untuk pentransmisi data sejauh 100 km perlu dipasangkan *repeater* sebanyak:
 $100 \text{ km}/13\text{km} = 7$ buah *repeater*.



Gambar 5.2 Pemasangan *Repeater* untuk Transmisi Data Sejauh 100 km

Sumber: Hasil perhitungan

5.1.2 Analisis *Rise-Time Budget*

Untuk menghitung *rise-time budget* sistem komunikasi serat optik yang digunakan, t_{TX} dan t_{RX} dari *transceiver* dan dispersi total dari serat optik perlu dihitung terlebih dahulu. LED beserta rangkaian *drive transceiver* yang digunakan dalam analisis skripsi ini mempunyai t_{TX} dan t_{RX} masing-masing 1,4 ns dan 1,0 ns. Dispersi intermodal dihitung dengan menggunakan persamaan (3-3):

$$\tau_{img} = \frac{n_1 \cdot D \cdot \Delta_g^2}{8C}$$

$$\tau_{img} = \frac{1,486.1x10^3 \left(\frac{n_1^2 - n_2^2}{2n_1^2} \right)^2}{8.3x10^8}$$

$$\tau_{img} = \frac{1486 \left(\frac{1,486^2 - 1,466^2}{2(1,486^2)} \right)^2}{24x10^8}$$

$$\tau_{img} = \frac{1486 \left(\frac{2,208196 - 2,149156}{4,416392} \right)^2}{24x10^8}$$

$$\tau_{img} = \frac{1486(0,013368)^2}{24x10^8}$$

$$\tau_{img} = 1,1065x10^{-10} s$$

Dispersi bahan dihitung dengan menggunakan persamaan (3-5):

$$\tau_m = \Delta\lambda.DM$$

$$\tau_m = 1300nm.1km.3,5 ps / nm.km$$

$$\tau_m = 4,55x10^{-9} s$$

Dispersi *waveguide* dihitung dengan menggunakan persamaan (3-6):

$$\tau_w = \frac{D}{C\lambda_0} (n_1 - n_2) D_w \Delta\lambda$$

$$\tau_w = \frac{1x10^3}{3x10^8.1300x10^{-9}} (1,486 - 1,466) (3,5x10^{-12} / 1x10^{-9}.1x10^3) .1300x10^{-9}$$

$$\tau_w = 3,3333x10^{-6} (1,486 - 1,466) .3,5x10^{-12} / 1x10^{-6}$$

$$\tau_w = 2,3333x10^{-13} s$$

Dispersi total yang terjadi di dalam serat optik dihitung dengan menggunakan persamaan (3-7):

$$\tau_f = \sqrt{\tau_{im}^2 + \tau_m^2 + \tau_w^2}$$

$$\tau_f = \sqrt{(1,1065x10^{-10})^2 + (4,55x10^{-9})^2 + (2,3333x10^{-13})^2}$$

$$\tau_f = \sqrt{1,224342x10^{-20} + 2,07025x10^{-17} + 5,444289x10^{-26}}$$

$$\tau_f = \sqrt{2,071474x10^{-17}}$$

$$\tau_f = 4,55x10^{-9} s$$

Rise-time budget sistem dihitung dengan menggunakan persamaan (3-13):

$$t_{sys} = (t_{TX}^2 + D^2 \sigma_\lambda^2 L^2 + t_{RX}^2)^{1/2}$$

$$t_{sys} = ((1,4 \times 10^{-9})^2 + (4,55 \times 10^{-9})^2 (1300 \times 10^{-9})^2 (1 \times 10^3)^2 + (1,0 \times 10^{-9})^2)^{1/2}$$

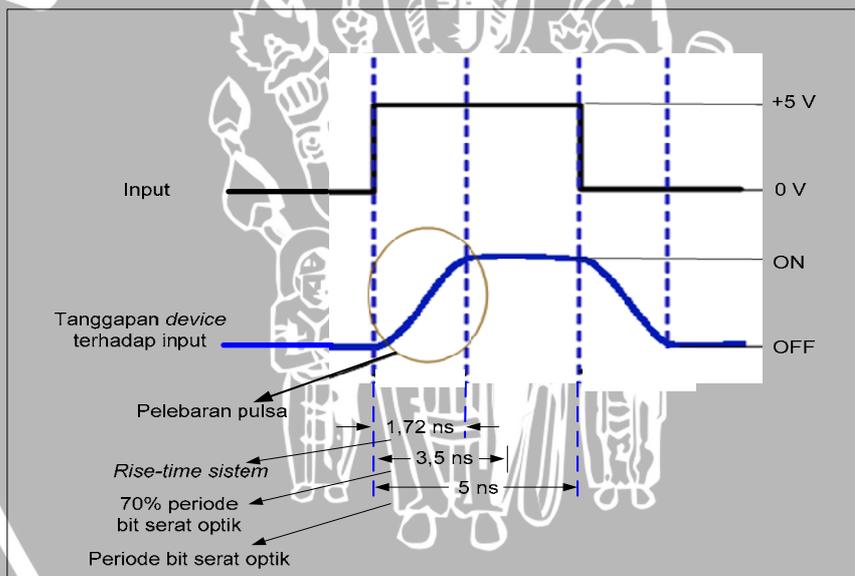
$$t_{sys} = (1,96 \times 10^{-18} + 3,4987 \times 10^{-23} + 1,0 \times 10^{-18})^{1/2}$$

$$t_{sys} = (2,96 \times 10^{-18})^{1/2}$$

$$t_{sys} = \sqrt{2,96 \times 10^{-18}}$$

$$t_{sys} = 1,72 \times 10^{-9} \text{ s} = 1,72 \text{ ns}$$

Nilai tersebut masih di bawah nilai maksimum yang diperbolehkan, yaitu: 70% dari periode bit NRZ. Serat optik yang dianalisis memiliki laju bit transmisi 200 Mbps (bandwidth-length-product sebesar 100 MHz.km), sehingga periode bit-nya adalah 5 ns dan 70%-nya adalah 3,5 ns.



Gambar 5.3 Rise-time Sistem

Sumber: Hasil perhitungan

5.2 Analisis Bit Error Rate (BER)

Amplifier dengan gain sebesar G dan sinyal masukan P_{in} akan menghasilkan sinyal keluaran $P_{out} = G \cdot P_{in}$. Berdasarkan persamaan (3-8):

$$(SNR)_{in} = \frac{P_{in}}{2hf\Delta_f}$$



Dengan nilai $P_{in} = 1 \times 10^{-5}$ W, $h = 6,626 \times 10^{-34}$ J/s, $f = 1,935 \times 10^{14}$ Hz, $\Delta_f = 21 \times 10^{19}$ Hz, maka besar nilai SNR_{in} adalah:

$$(SNR)_{in} = \frac{P_{in}}{2hf\Delta_f}$$

$$(SNR)_{in} = \frac{1 \times 10^{-5}}{2 \times (6,626 \times 10^{-34}) \times (1,935 \times 10^{14}) \times (21 \times 10^9)}$$

$$(SNR)_{in} = \frac{1 \times 10^{-5}}{5,3850 \times 10^{-9}}$$

$$(SNR)_{in} = 1857 = 32,68 \text{ dB}$$

Dengan $G = 2,0625$ dan dengan persamaan (3-9) dan (3-10), maka besar nilai SNR sinyal yang dikuatkan (SNR_{out}) adalah:

$$(SNR)_{out} = \frac{GP_{in}}{4S_{sp}\Delta_f}$$

$$(SNR)_{out} = \frac{GP_{in}}{4[(2,0625 - 1)n_{sp}hf]\Delta_f}$$

$$(SNR)_{out} = \frac{2,0625(1 \times 10^{-5})}{4[(2,0625 - 1)1,5 \cdot 6,626 \times 10^{-34} \cdot 1,935 \times 10^{14}]21 \times 10^9}$$

$$(SNR)_{out} = \frac{2,0625 \times 10^{-5}}{4[(1,0625)1,9232 \times 10^{-19}]21 \times 10^9}$$

$$(SNR)_{out} = \frac{2,0625 \times 10^{-5}}{4[2,0435 \times 10^{-18}]21 \times 10^9}$$

$$(SNR)_{out} = \frac{2,0625 \times 10^{-5}}{1,7166 \times 10^{-8}}$$

$$(SNR)_{out} = 1.201,53$$

$$(SNR)_{out} = 30,7873 \text{ dB}$$

Untuk menghitung BER, dihitung dulu nilai Q -factor:

$$Q = 20 \left(\log \sqrt{SNR_{out}} \right) \sqrt{\frac{B_o}{B_e}}$$

dengan:

B_o = bandwidth optik dari detektor optik (Hz), dalam skripsi ini nilainya = 3×10^9 Hz

B_e = bandwidth elektrik dari filter receiver (Hz), dalam skripsi ini nilainya = 21×10^9 Hz

$$Q = 20 \left(\log \sqrt{30,7873} \right) \sqrt{\frac{3 \times 10^9}{21 \times 10^9}}$$

$$Q = 20 (\log 5,5495) 0,378 = 5,6266 \text{ dB}$$

Dan BER dapat dihitung dengan persamaan:

$$\text{BER} = \frac{1}{2} \operatorname{erfc} \left(\frac{Q}{\sqrt{2}} \right)$$

$$\text{BER} = \frac{1}{2} \operatorname{erfc} \left(\frac{5,6266}{\sqrt{2}} \right)$$

$$\text{BER} = \frac{1}{2} \operatorname{erfc}(3,9786)$$

$$\text{BER} = \frac{1}{2} (1,8381 \times 10^{-8})$$

$$\text{BER} = 9,1905 \times 10^{-9}$$

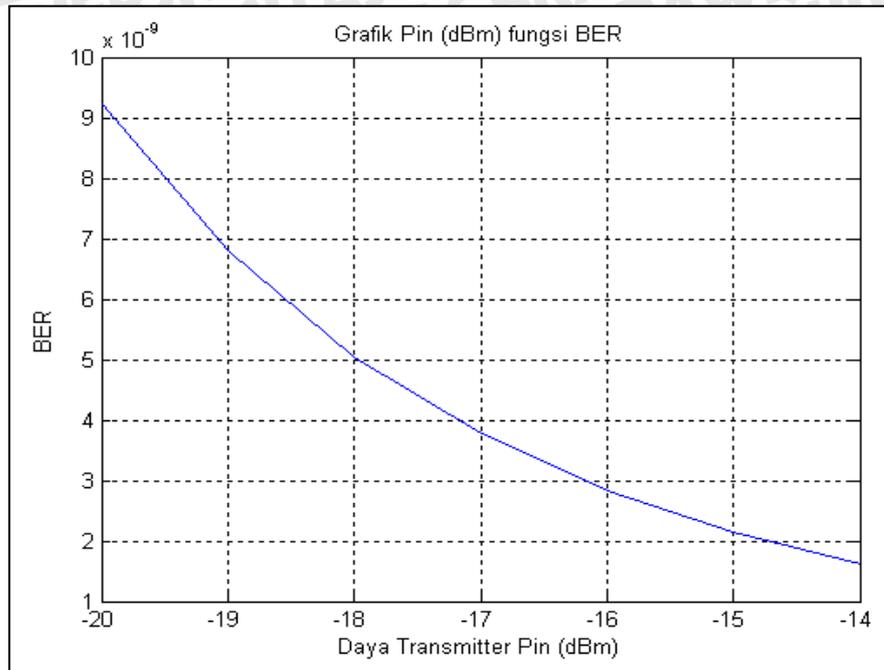
Perhitungan BER secara lengkap dapat dilihat pada tabel berikut:

Tabel 5.1 BER untuk spesifikasi LED

No.	Daya transmitter (dBm)	Q-factor (dB)	BER
1.	-20	5,4702	9,190x10 ⁻⁹
2.	-19	5,5276	6,793x10 ⁻⁹
3.	-18	5,5832	5,041x10 ⁻⁹
4.	-17	5,6371	3,765x10 ⁻⁹
5.	-16	5,6892	2,830x10 ⁻⁹
6.	-15	5,7397	2,140x10 ⁻⁹
7.	-14	5,7887	1,627x10 ⁻⁹

Sumber: Hasil perhitungan

Dari tabel di atas terlihat bahwa semakin besar nilai *Q-factor*, nilai BER semakin kecil.



Gambar 5.4 Grafik hubungan P_{in} terhadap BER

Sumber: Hasil perhitungan

Gambar 5.4 menunjukkan nilai BER makin kecil dengan bertambahnya daya transmitter P_{in} .

5.3 Analisis Delay

Delay pada jaringan merupakan jumlah waktu yang dibutuhkan untuk mengirimkan *frame* dari *station* pengirim ke *station* penerima, dan *station* penerima akan mengirimkan kembali *frame* tersebut ke *station* asal. Komponen *delay* yang akan dianalisis pada jaringan ini adalah *delay* transmisi, *delay* paketisasi, *delay* enkapsulasi, *delay* propagasi, dan *delay* antrian, termasuk *ring latency*, efisiensi dan *delay* akses maksimum.

5.3.1 Delay Transmisi

Dalam pentransmisian data, paket data akan mengalami *delay* transmisi dari *station* penerima ke *station* tujuan. Dari persamaan (3-19) maka *delay* transmisi untuk jaringan FDDI adalah:

$$T_r = \frac{L_{FDDI} \times (8 \text{ bit / byte})}{C_{FDDI}}$$

Untuk $L_{\text{FDDI}} = 500 \text{ byte}$, maka nilai *delay* transmisi adalah:

$$T_r = \frac{500 \text{ byte} (8 \text{ bit} / \text{byte})}{100 \text{ Mbps}}$$

$$T_r = \frac{4000 \text{ bit}}{100 \times 10^6 \text{ bps}} = 4 \times 10^{-5} \text{ s}$$

Nilai *delay* transmisi untuk panjang paket data yang lain dapat dilihat dalam tabel berikut:

Tabel 5.2 Nilai *delay* transmisi untuk beberapa panjang paket data

No.	Panjang paket data (byte)	T_r (s)
1.	500	4×10^{-5}
2.	1000	8×10^{-5}
3.	2000	16×10^{-5}
4.	3000	24×10^{-5}
5.	4000	32×10^{-5}
6.	4500	36×10^{-5}

Sumber: Hasil perhitungan

Tabel 5.2 menunjukkan bahwa semakin panjang ukuran data yang dilewatkan, nilai *delay* transmisi akan semakin bertambah.

5.3.2 Delay Propagasi

Dalam pentransmisian data di jaringan komunikasi serat optik, paket data akan mengalami *delay* propagasi cahaya. *Delay* ini disebabkan oleh propagasi gelombang cahaya yang membawa paket data berpropagasi dalam media serat optik. Besar dari *delay* propagasi pada jaringan ini dihitung dengan menggunakan persamaan (3-20) maka *delay* propagasi untuk jaringan FDDI adalah:

$$P_{\text{pro}} D = dxv$$

$$P_{\text{pro}} D = 100 \text{ km} (5,085 \times 10^{-6} \text{ s} / \text{km})$$

$$P_{\text{pro}} D = 5,085 \times 10^{-4} \text{ s}$$

Nilai *delay* propagasi di sini adalah tetap karena nilai *delay* propagasi gelombang cahaya pada serat optik dan jarak transmisi tetap (sudah ditentukan).

5.3.3 Delay Paketisasi

Delay paketisasi adalah banyaknya waktu yang diperlukan untuk mengubah data dalam bentuk paket. Berdasarkan persamaan (3-21) besarnya *delay* paketisasi jaringan FDDI adalah:

$$T_{Paket} = \frac{L_{FDDI} \times (8bit / byte)}{C}$$

Untuk $L_{FDDI} = 500 \text{ byte}$, besar nilai *delay* transmisi adalah:

$$T_{Paket} = \frac{500 \text{ byte} \times (8bit / byte) - 32bit}{100 \text{ Mbps}}$$

$$T_{Paket} = \frac{3968bit}{100 \times 10^6 \text{ bps}}$$

$$T_{Paket} = 3,968 \times 10^{-5} \text{ s}$$

Nilai *delay* paketisasi untuk panjang paket data yang lain dapat dilihat dalam tabel berikut:

Tabel 5.3 Nilai *delay* paketisasi untuk beberapa panjang paket data

No.	Panjang paket data (byte)	T_{Paket} (s)
1.	500	$3,968 \times 10^{-5}$
2.	1000	$7,968 \times 10^{-5}$
3.	2000	$15,968 \times 10^{-5}$
4.	3000	$23,968 \times 10^{-5}$
5.	4000	$31,968 \times 10^{-5}$
6.	4500	$35,968 \times 10^{-5}$

Sumber: Hasil perhitungan

Tabel 5.3 menunjukkan bahwa semakin panjang ukuran data yang dilewatkan, nilai *delay* paketisasi semakin bertambah.

5.3.4 Delay Enkapsulasi

Delay enkapsulasi adalah *delay* yang terjadi saat paket data diberi *header* tambahan. Dalam skripsi ini *header* yang ditambahkan adalah *header label* GMPLS. Besar *header label* GMPLS adalah 32 bit. Jadi, ukuran *frame* FDDI yang sudah ditambahkan *header label* GMPLS, $W_{frame} = 500 - 4500 \text{ byte} = 4.000 - 36.000 \text{ bit}$. *Frame* FDDI yang belum ditambahkan *header label* GMPLS disebut dengan *maximum*

segment size (MSS). Berdasarkan persamaan (3-22) besarnya *delay* enkapsulasi *frame* FDDI pada jaringan FDDI adalah:

$$T_{Enkap} = \frac{(W_{frame} - MSS)}{C}$$

$$T_{Enkap} = \frac{32}{100 \times 10^6}$$

$$T_{Enkap} = 3,2 \times 10^{-7} s$$

Nilai *delay* enkapsulasi tetap karena besarnya nilai *delay* enkapsulasi dihitung dari besarnya ukuran *header* yang ditambahkan ke paket data dibagi dengan kecepatan transmisi data.

5.3.5 Delay Antrian (*Queueing delay*)

Antrian diperlukan untuk menghindari *loss* yang berlebihan. Untuk mengetahui besar dari *delay* antrian, terlebih dahulu dihitung nilai rata-rata waktu pelayanan (μ) dengan menggunakan persamaan (3-23).

Untuk panjang data 500 *byte* maka besar rata-rata waktu pelayanannya adalah:

$$\mu = \frac{C}{m} \text{ paket / s}$$

$$\mu = \frac{C_{FDDI}}{m_{FDDI}} \text{ paket / s}$$

$$\mu = \frac{C_{FDDI}}{(L_{FDDI} \times 8 \text{ bit / byte})} \text{ paket / s}$$

$$\mu = 100 \times \frac{100 \text{ Mbps}}{500 \text{ byte} (8 \text{ bit / byte})} \text{ paket / s}$$

$$\mu = \frac{100 \times 10^8 \text{ bps}}{4000 \text{ bit}} \text{ paket / s}$$

$$\mu = 2500000 \text{ paket / s}$$

Dengan nilai *utilization* sebesar 10% maka dengan menggunakan persamaan (3-26) akan didapatkan jumlah rata-rata waktu kedatangan (λ) sebesar:

$$U = \frac{\lambda}{\mu}$$

$$\lambda = U \times \mu$$

$$\lambda = 0,1 \times 2500000$$

$$\lambda = 250000 \text{ paket / s}$$

Dari perhitungan di atas maka dapat ditentukan besar *delay* antrian untuk jaringan FDDI dengan menggunakan persamaan (3-33) adalah:

$$T_{Queue} = \frac{\lambda / \mu}{\mu - \lambda} + \frac{1}{\mu}$$

$$T_{Queue} = \frac{250000 / 2500000}{(2500000 - 250000)} + \frac{1}{2500000} \text{ s}$$

$$T_{Queue} = \frac{0,1}{2,25 \times 10^6} + \frac{1}{2500000} \text{ s}$$

$$T_{Queue} = 4,44 \times 10^{-8} + 4 \times 10^{-7} \text{ s}$$

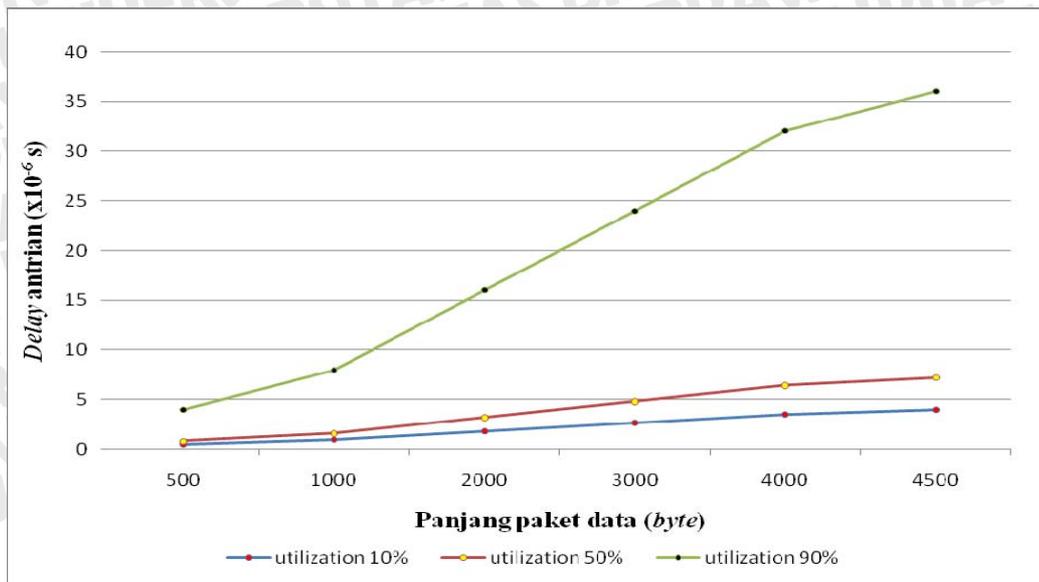
$$T_{Queue} = 4,444 \times 10^{-7} \text{ s}$$

Nilai *delay* antrian untuk panjang paket data yang lain dengan *utilization* 10% dapat dilihat dalam tabel berikut:

Tabel 5.4 Nilai *delay* antrian untuk beberapa panjang paket data dengan *utilization* 10%

No.	Panjang paket data (byte)	T_{Queue} (s)
1.	500	$4,444 \times 10^{-7}$
2.	1000	$8,889 \times 10^{-7}$
3.	2000	$17,778 \times 10^{-7}$
4.	3000	$26,667 \times 10^{-7}$
5.	4000	$35,556 \times 10^{-7}$
6.	4500	$40,000 \times 10^{-7}$

Sumber: Hasil perhitungan



Gambar 5.5 Grafik hubungan *delay* antrian terhadap Panjang paket data

Sumber: Hasil perhitungan

Dari grafik di atas dapat dilihat bahwa semakin banyak data yang dilewatkan dengan nilai utilisasi yang makin besar, nilai *delay* antrian akan semakin bertambah. Nilai *delay* antrian paling besar diperoleh pada *utilization* 90%.

5.3.6 Analisis Ring Latency

Analisis *ring latency* tergantung pada kontribusi *delay* dari masing-masing *node* dan kabel serat optik yang digunakan untuk media transmisi. *Ring latency D* dihitung dengan mengasumsikan *delay* propagasi pada serat optik pada nilai standarnya $D_c = 5,085 \mu\text{s}/\text{km}$ dan *delay interface* tiap *station* sebesar $D_s = 1 \mu\text{s}$ [Gary C. Kessler and David A. A. Train, 1992:336]. Dengan menggunakan persamaan (3-34), (3-35), dan (3-36), besar nilai *node delay*, *cable delay*, dan *ring latency* dengan jumlah *node* = 100 buah masing-masing adalah:

$$\text{Node delay} = N \times D_s = 100 \times 1 \mu\text{s} = 1 \times 10^{-4} \text{ s}$$

$$\text{Cable delay} = L \times D_c = 100 \text{ km} \times 5,085 \mu\text{s}/\text{km} = 5,085 \times 10^{-4} \text{ s}$$

Ring latency D:

$$D = (N \times D_s) + (L \times D_c) \text{ s}$$

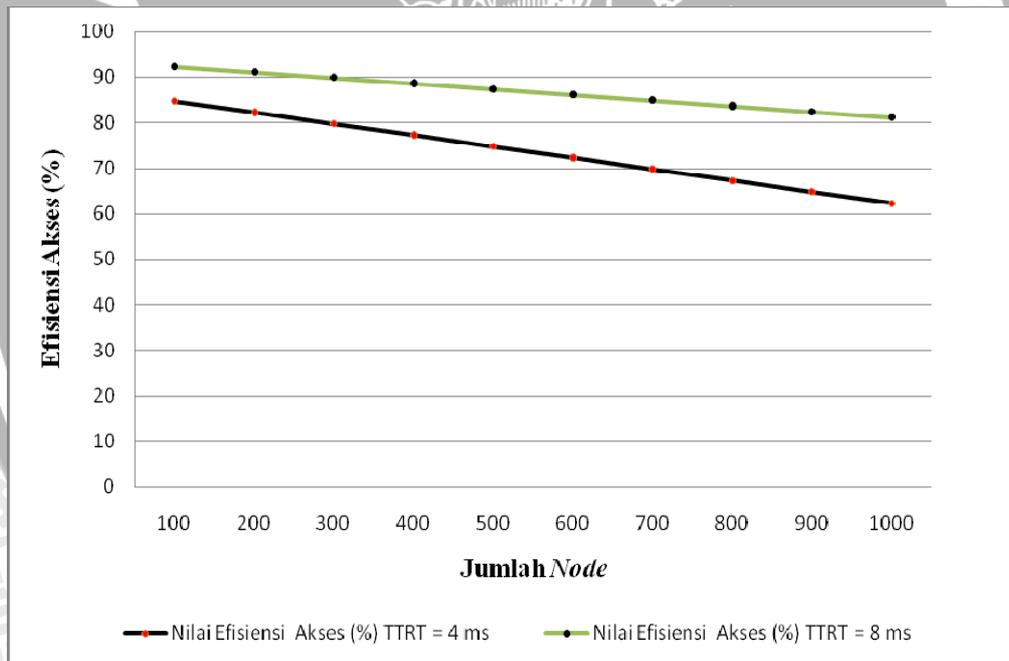
$$D = 1 \times 10^{-4} + 5,085 \times 10^{-4} = 6,085 \times 10^{-4} \text{ s}$$

5.3.7 Analisis Efisiensi Akses dan *Delay* Akses Maksimum

Dengan menggunakan nilai TTRT T sebesar 4 ms, dapat dihitung besar efisiensi akses dan *delay* akses maksimum (persamaan 3-37 dan 3-38) pada jaringan FDDI yaitu:

$$\begin{aligned} \text{Efisiensi akses} &= \frac{N(T - D)}{NT + D} \\ &= \frac{100(4 \times 10^{-3} - 6,085 \times 10^{-4})}{(100 \cdot 4 \times 10^{-3}) + 6,085 \times 10^{-4}} \\ &= 0,846587 \times 100\% \\ &= 84,6587\% \end{aligned}$$

$$\begin{aligned} \text{Delay akses maksimum} &= (N - 1)T + 2D \\ &= (100 - 1)4 \times 10^{-3} + (2 \cdot 6,085 \times 10^{-4}) \\ &= 0,396 + 0,001217 \\ &= 0,3972 \text{ s} \end{aligned}$$



Gambar 5.6 Grafik hubungan Efisiensi Akses terhadap Jumlah *Node*

Sumber: Hasil perhitungan

Grafik di atas menjelaskan bahwa semakin banyak jumlah *node* yang harus dilewati akan mengurangi nilai efisiensi akses. Nilai efisiensi akses pada TTRT = 8 ms lebih besar daripada TTRT = 4 ms.

5.3.8 Delay Total

Dengan menggunakan (3-39) besarnya *delay* pentransmisian *frame* untuk panjang paket data 500 *byte* adalah:

$$T_{pf} = T_r + T_{Pro} + T_{Queue} + T_{Paket} + T_{Enkap}$$

$$T_{pf} = 4 \times 10^{-5} + 5,085 \times 10^{-4} + 4,444 \times 10^{-7} + 3,968 \times 10^{-5} + 3,2 \times 10^{-7}$$

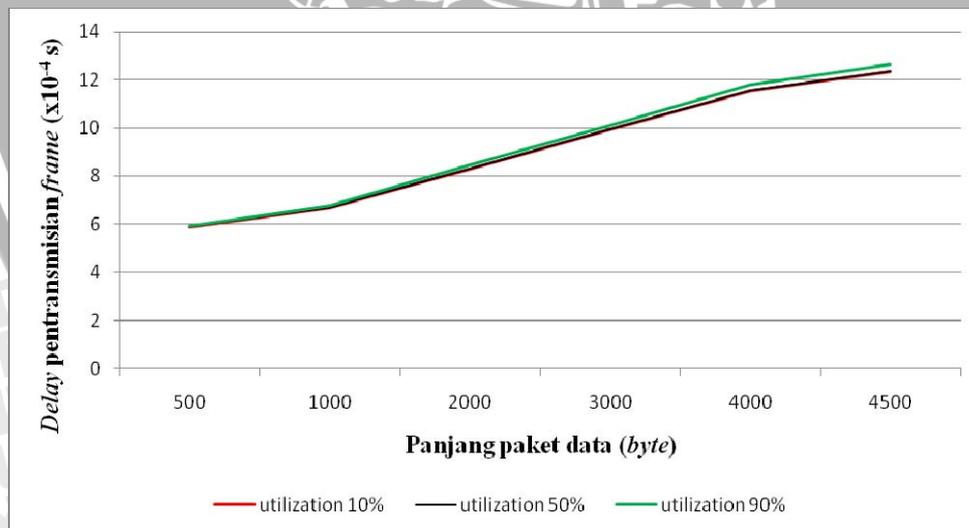
$$T_{pf} = 5,8894 \times 10^{-4} \text{ s}$$

Jumlah *delay* pentransmisian *frame* untuk panjang paket data yang lain dapat dilihat dalam tabel berikut:

Tabel 5.5 Nilai *delay* pentransmisian *frame* untuk beberapa panjang paket data

No.	Panjang paket data (<i>byte</i>)	T_{pf} (s)
1.	500	$5,8894 \times 10^{-4}$
2.	1000	$6,6939 \times 10^{-4}$
3.	2000	$8,3038 \times 10^{-4}$
4.	3000	$9,9117 \times 10^{-4}$
5.	4000	$11,5205 \times 10^{-4}$
6.	4500	$12,3250 \times 10^{-4}$

Sumber: Hasil perhitungan



Gambar 5.7 Grafik hubungan *delay* pentransmisian *frame* terhadap Panjang paket data

Sumber: Hasil perhitungan

Dari grafik di atas dapat dilihat bahwa semakin banyak data yang dilewatkan dengan nilai utilisasi mendekati 1, nilai *delay* pentransmisian *frame* akan semakin bertambah. Nilai *delay* pentransmisian *frame* paling besar diperoleh pada *utilization* 90%.

Dari hasil *delay* pentransmisian *frame*, dapat dihitung jumlah waktu yang dibutuhkan oleh sebuah *station* untuk mentransmisikan datanya mulai dari mengakses jaringan hingga selesai mengirimkan datanya yaitu:

Dengan jumlah *node* 100, panjang paket data 500 *byte*, nilai TTRT = 4 ms, dan *ring latency* sebesar 0,6085 ms akan didapatkan *delay* akses maksimum sebesar 0,3972 s, maka dapat dihitung *delay* total jaringan yaitu:

$$T_{total} = T_{am} + T_{pf}$$

$$T_{total} = 0,3972 + 5,8894 \times 10^{-4}$$

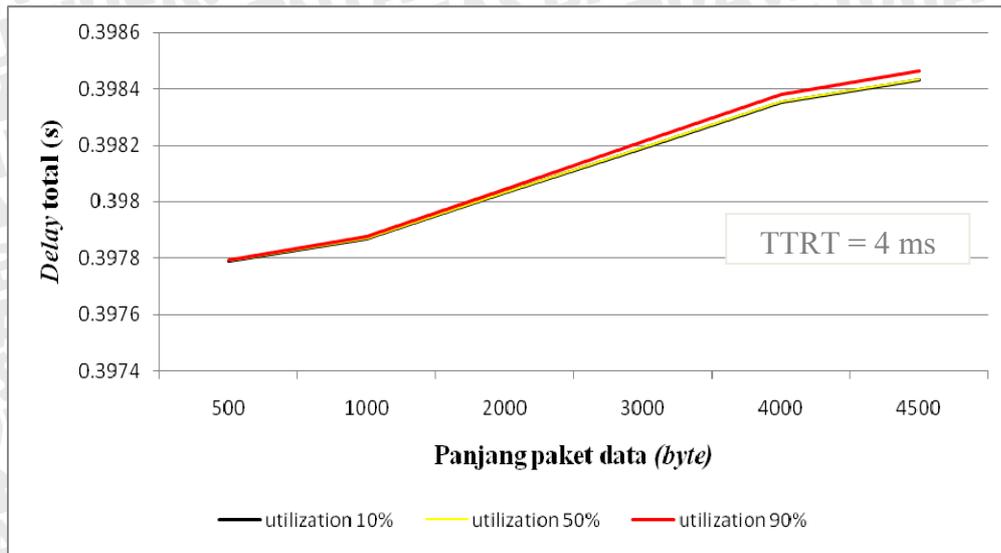
$$T_{total} = 0,397789 \text{ s.}$$

Tabel 5.6 Nilai *delay* total untuk beberapa panjang paket data

No.	Panjang paket data (byte)	T_{total} (s)
1.	500	0,397789
2.	1000	0,397869
3.	2000	0,398030
4.	3000	0,398191
5.	4000	0,398352
6.	4500	0,398432

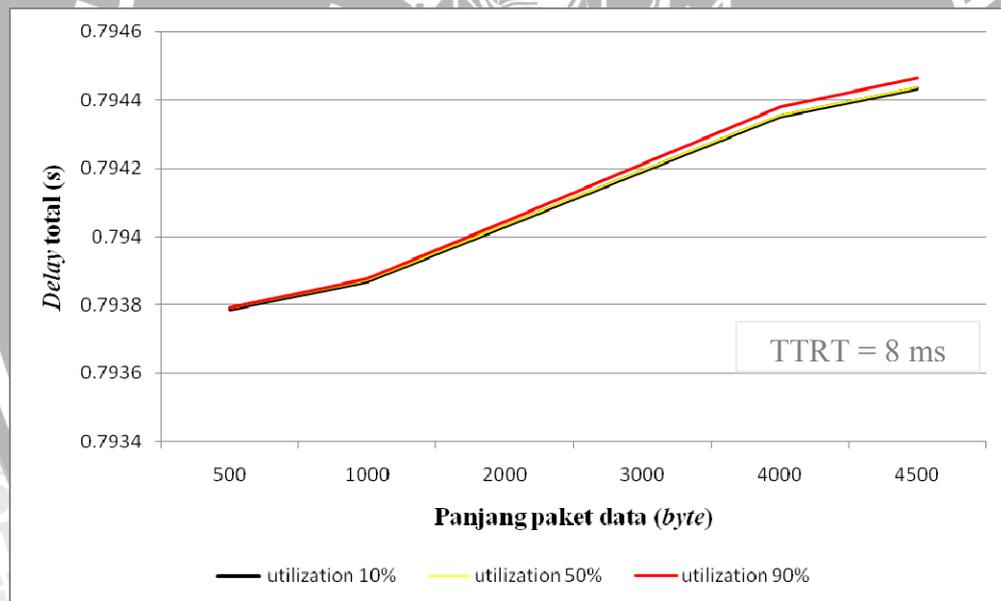
Sumber: Hasil perhitungan





Gambar 5.8 Grafik hubungan *delay* total terhadap Panjang paket data

Sumber: Hasil perhitungan



Gambar 5.9 Grafik hubungan *delay* total terhadap Panjang paket data

Sumber: Hasil perhitungan

Dari grafik gambar 5.8 dan gambar 5.9 dapat dilihat bahwa semakin banyak data yang dilewatkan dengan nilai utilisasi dan TTRT yang makin besar, nilai *delay* pentransmisian *frame* akan semakin bertambah. Nilai *delay* pentransmisian *frame* paling besar diperoleh pada *utilization* 90% dengan TTRT = 8 ms.

5.4 Analisis Bit Rate

Bit rate FDDI dapat mencapai 100 Mbps, hal ini disebabkan karena menggunakan teknik *encoding digital* 4B/5B. Teknik *encoding* ini menggunakan *line code* dimana setiap 4 bit masukan diubah menjadi 5 bit *words*. Pada *encoding* 4B/5B ada 32 bit *words* data, dan masing-masing 1 bit *words* terdiri atas 5 bit data. Untuk 8 bit data (1 *byte* *oktet*) akan diubah menjadi 10 bit terdiri atas 2 bit *words* ($N = 2$), sehingga akan terdapat $L = 32 = 2^5$ kombinasi bit *words* (simbol/bit). Pada pentransmisian bit data dengan teknik *encoding* 4B/5B pada FDDI, masing-masing bit data mempunyai alokasi frekuensi 12,5 MHz. *Baud rate* dapat dihitung dengan persamaan (3-42) adalah:

$$D = \frac{N}{T_0} = \frac{N}{1/f}$$

$$D = \frac{2}{1/12,5 \times 10^6} \text{ simbol/s}$$

$$D = 25 \times 10^6 \text{ simbol/s} = 25 \text{ Mbaud}$$

Dan besar *bit rate* dapat dihitung dengan persamaan (3-43) adalah:

$$R = (\log_2 L)D$$

$$R = (\log_2 32) 25 \times 10^6 = 5 \times 25 \times 10^6 = 125 \times 10^6 \text{ bps} = 125 \text{ Mbps}$$

Oleh karena itu, pada teknik *encoding* 4B/5B *actual transmission rate* FDDI mencapai 125 Mbps, dengan 80% atau 100 Mbps digunakan untuk pentransmisian data dan 20% untuk *redundancy* [Max Ming-Kang Liu, 1996:393].

5.5 Analisis Throughput

Analisis *throughput* akan dihitung dengan menggunakan panjang paket data, jumlah *node*, dan panjang serat optik yang berbeda untuk mengetahui kualitas pengiriman data. Nilai *throughput* jaringan FDDI dengan panjang paket data FDDI $L_{\text{FDDI}} = 500 \text{ byte}$, bit tiap *node* $b = 562,5 \text{ bit}$, $p_b = 6,19 \times 10^{-9}$, kecepatan transmisi $R = 100 \text{ Mbps}$ dihitung sebagai berikut:

- a. Probabilitas *bit error* (p) sesuai dengan persamaan (3-48) adalah:

$$W_{\text{frame}} = L_{\text{FDDI}} = 500 \text{ byte}$$

$$p = W_{\text{frame}} \cdot p_b$$

$$p = (500 \text{ byte} \times (8 \text{ bit/byte})) \times 6,19 \times 10^{-9} = 2,4760 \times 10^{-5}$$

- b. Waktu interval dapat ditentukan oleh persamaan (3-45):

$$T_{out} = T_r + \frac{N \cdot b}{R}$$

$$T_{out} = \left(\frac{L}{C} \right) + \frac{N \cdot b}{R}$$

$$T_{out} = \left(\frac{1 \times 10^5 \text{ m}}{3 \times 10^8 \text{ m/s}} \right) + \frac{100 \times 562,5 \text{ bit}}{100 \text{ Mbps}}$$

$$T_{out} = \left(\frac{1 \times 10^5 \text{ m}}{3 \times 10^8 \text{ m/s}} \right) + \frac{100 \times 562,5 \text{ bit}}{100 \times 10^6 \text{ bps}}$$

$$T_{out} = 3,3333 \times 10^{-4} + 5,625 \times 10^{-4} = 8,9583 \times 10^{-4} \text{ s}$$

- c. Nilai konstanta a dihitung dengan menggunakan persamaan (3-46) adalah:

$$a = \frac{T_{out}}{L_{FFDI} / R} N$$

$$a = \frac{8,9583 \times 10^{-4}}{(500 \text{ byte}(8 \text{ bit / byte})) / 100 \text{ Mbps}} \times 100$$

$$a = \frac{8,9583 \times 10^{-4}}{(500 \text{ byte}(8 \text{ bit / byte})) / 100 \times 10^6 \text{ bps}} \times 100$$

$$a = 2,2396 \times 10^3$$

- d. Waktu transmisi (T_r) di sini sama dengan *delay* pentransmisian *frame* (T_{pf})

$$T_r = T_{pf}$$

$$T_r = 5,8894 \times 10^{-4} \text{ s}$$

- e. *throughput* dihitung dengan persamaan (3-44) adalah:

$$\rho_{\max} = \frac{1}{t_v} = \frac{(1-p)}{T_r [1 + (a-1)p]}$$

$$\rho_{\max} = \frac{(1 - 2,4760 \times 10^{-5})}{5,8894 \times 10^{-4} [1 + (2,2396 \times 10^3 - 1) 2,4760 \times 10^{-5}]}$$

$$\rho_{\max} = \frac{0,99997524}{6,2158 \times 10^{-4}} \text{ paket / s}$$

$$\rho_{\max} = 1608,75 \text{ paket / s}$$

$$\rho_{\max} = 6,44 \text{ Mbps}$$

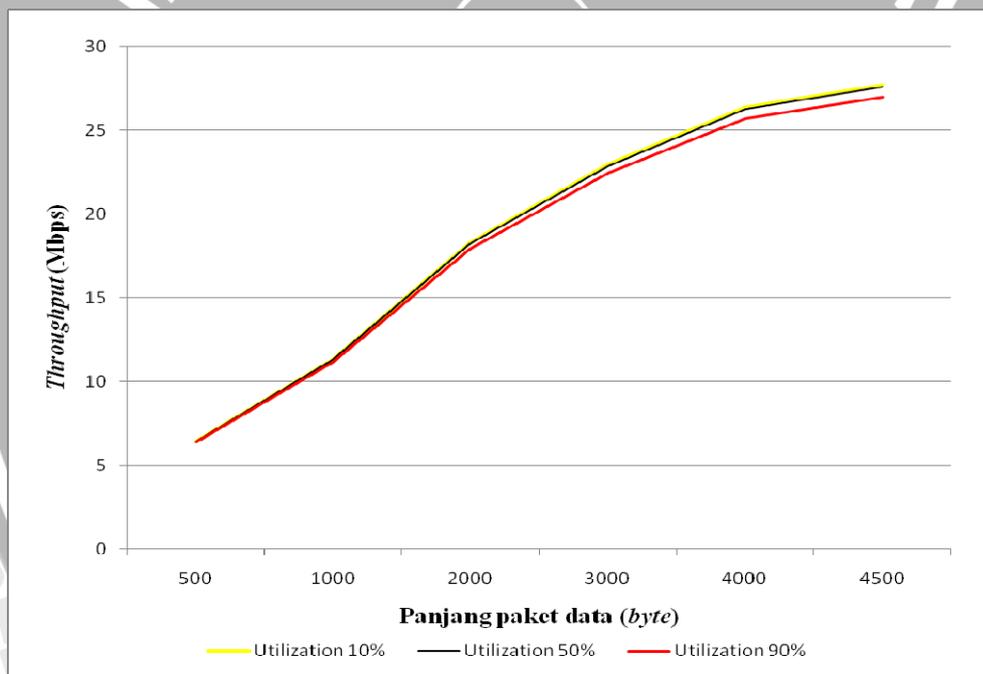
Sehingga nilai *throughput* untuk pengiriman paket data 500 byte dengan 100 node terhubung dengan *utilization* 10% dalam jaringan FDDI adalah sebesar 6,44Mbps.

Nilai *throughput* untuk panjang paket FDDI yang berbeda dapat dilihat pada tabel berikut.

Tabel 5.7 Nilai *throughput* dengan panjang paket data yang berbeda

No.	Panjang paket data (byte)	Throughput (Mbps)
1.	500	6,44
2.	1000	11,32
3.	2000	18,26
4.	3000	22,94
5.	4000	26,32
6.	4500	27,67

Sumber: Hasil perhitungan



Gambar 5.10 Grafik hubungan *Throughput* terhadap Panjang paket data

Sumber: Hasil perhitungan

Grafik di atas menunjukkan bahwa nilai *throughput* dipengaruhi oleh panjang paket data, *delay* transmisi, dan utilisasi jaringan. Semakin besar panjang paket data yang dilewatkan, nilai *throughput* akan semakin besar. Semakin besar *delay* transmisi dan utilisasi jaringan, nilai *throughput* akan semakin kecil.

BAB VI PENUTUP

6.1 Kesimpulan

1. GMPLS hanya melakukan enkapsulasi *frame* FDDI dengan memasang *header* GMPLS. Proses enkapsulasi ini akan menambah jumlah *delay* pengiriman data sebesar 36 μ s dan mengurangi *throughput* sebesar 0,04 Mbps (untuk panjang *frame* 500 *byte*), tetapi tidak mempengaruhi *Bit Error Rate* (BER) dan *bit rate* sistem. Pelabelan *frame* FDDI tidak mempengaruhi BER karena analisis BER dilakukan pada *transceiver* serat optik. Sedangkan pelabelan *frame* FDDI tidak mempengaruhi *bit rate* sistem karena sudah merupakan karakteristik format data dari FDDI yang menggunakan teknik *encoding digital* 4B/5B.
2. Proses enkapsulasi ini akan menambah jumlah *delay* pengiriman data dan mengurangi *throughput*, tetapi tidak mempengaruhi *Bit Error Rate* (BER) dan *bit rate* sistem.
3. *Delay* total jaringan akan bertambah jika jumlah data yang dilewatkan dan utilisasi jaringan bertambah. Nilai *delay* total jaringan terkecil adalah 0,397789 s pada panjang paket data yang dilewatkan sebesar 500 *byte* dengan utilisasi/penggunaan kapasitas jaringan 10%, dan TTRT = 4 ms. Nilai *delay* total jaringan terbesar adalah 0,794465 s pada panjang paket data yang dilewatkan sebesar 4500 *byte*, utilisasi/penggunaan kapasitas jaringan 90%, dan TTRT = 8 ms.
4. Nilai *throughput* dipengaruhi oleh panjang paket data, *delay* transmisi, dan utilisasi jaringan. Semakin besar panjang paket data yang dilewatkan, nilai *throughput* akan semakin besar. Semakin besar *delay* transmisi dan utilisasi jaringan, nilai *throughput* akan semakin kecil. Nilai *throughput* terbesar adalah 27,67 Mbps. Nilai ini didapat saat panjang paket data yang dilewatkan adalah 4500 *byte* dengan utilisasi jaringan 10%.
5. Dari perhitungan *ring latency*, semakin besar jumlah *node* yang terhubung ke jaringan *token ring* FDDI maka nilai *ring latency* semakin besar. Nilai *ring latency* akan mempengaruhi efisiensi akses dan *delay* maksimum. Nilai *ring*

latency untuk 100 *node* adalah sebesar $6,085 \times 10^{-4}$ s dan untuk 1000 *node* memiliki nilai *ring latency* sebesar $15,085 \times 10^{-4}$ s.

6. Parameter-parameter yang mempengaruhi kinerja FDDI adalah TTRT, panjang saluran transmisi, jumlah *node*, dan ukuran *frame* data.

6.2 Saran

Saran yang dapat diberikan berdasarkan proses analisis yang telah dilakukan:

1. Analisis penerapan teknologi GMPLS pada jaringan FDDI akan lebih baik jika dilakukan dengan simulasi tidak dengan hanya dengan menggunakan penghitungan ukuran paket data. Sehingga dapat diperoleh gambaran kinerja yang lebih nyata.
2. Penerapan GMPLS pada konfigurasi jaringan yang lebih rumit seperti SONET dan Gigabit *Ethernet* untuk mengetahui kinerjanya pada teknologi yang berbeda.



DAFTAR PUSTAKA

Agrawal, Govin P. 1992. *Fiber Optic Communication Systems*. John Wiley & Sons, Inc. Canada.

Alwayn, Vivek, CCIE™. 2002. *Advanced MPLS Design and Implementation*. Cisco Press.

Anonymous. 2000. *Building Scalable Service Provider IP Networks*. Marcony White Paper.

_____. *Fiber Distributed Data Interface*.

http://ebook.planindo.comdl.php?Chapter08_-_FDDI.pdf

_____. 2004. *Fundamental Switching Types-GMPLS*. Grotto Networking.

_____. *Generalized Multiprotocol Label Switching (GMPLS)*. Web ProForum Tutorials. <http://www.iec.org>. Copyright © The International Engineering Consortium.

_____. *GMPLS dalam Skema Konvergensi*.

<http://telkomcorner.blogspot.com/2008/12/gmpls-dalam-skema-konvergensi-33.html>.

_____. *GMPLS - The New Big Deal in Intelligent Metro Optical Networking*.

<http://www.polarisnetworks.com/images/PDF/GMPLSWhitePaperComplete.pdf>

_____. *Jaringan Komputer Model Analisis*.

<http://4sucktie.tripod.com/fiber.pdf>

_____. *MPLS Introduction*.

<http://www.bilmuh.gyte.edu.tr/BIL674/MPLS.ppt>

_____. *PT. TELEKOMUNIKASI INDONESIA, Tbk, PL 1.1 - Dasar Sistem Komunikasi Optik, OPTICAL ACCESS NETWORK*.

<http://septialutfi.edublogs.org/files/2007/11/dasar-sko.pdf>

Ayan Banerjee, John Drake, Jonathan P. Lang, and Brad Turner, Calient Networks.

Kireeti Kompella, Juniper Networks. Yakov Rekhter, Cisco Systems. January,

2001. *Generalized Multiprotocol Label Switching: An Overview of Routing and Management Enhancements*. IEEE Communication Magazine.

Chao Chen, Fang Liu, Yong Wang. April 5, 2001. *Backbone: FDDI, ATM, or Gigabit Ethernet?*.

http://www.ensc.sfu.ca/~ljlja/ENSC833/Spring01/Projects/chen_liu_wang/backbone.pdf

D, Allen. May, 2001. *How Will Multiprotocol Lambda Switching Change Optical Networks?*. Network Magazine.

Feibel, Werner. 1996. *The Encyclopedia of Networking Second Edition*. SYBEX Inc., 2021 Challenger Drive, Alameda, CA 94501.

Freeman, Roger L. 2001. *Practical Data Communications Second Edition*. A Wiley-Interscience Publication, John Wiley and Sons, Inc. New York.

Halabi, Sam. October 01, 2003. *Metro Ethernet*. Cisco Press.

Hoss, Robert J. 1990. *Fiber Optic Communication Design Handbook*. Prentice-Hall Inc. Englewood Cliffs, New Jersey, USA.

Jain, Raj. 1990. *Performance Analysis of FDDI Token Ring Networks: Effect of Parameters and Guidelines for Setting TTRT*. Copyright 1990, Association for Computing Machinery, Inc.

Keiser, Gerd. 1989. *Optical Fiber Communication*. Mc-Graw Hill Book Company. Singapore.

Kessler, Gary C and Train, David A A. 1992. *MAN Concept, Standard, and Service*. McGrawHill, Inc. Singapore.

Killen, Harold B. 1991. *Fiber Optic Communication*. Prentice-Hall International Inc. Englewood Cliffs, New Jersey.

Ming-Kang Liu, Max. 1996. *Principles And Applications of Optical Communications*. IRWIN McGraw-Hill Book Company, USA.

Partama, I Putu Sumanata. 2009. *PERENCANAAN LINK OPTIK DENPASAR-AMLAPURA UNTUK MEMENUHI KEBUTUHAN TRAFIK DI DAERAH BALI TIMUR HINGGA TAHUN 2015*.

http://www.ittelkom.ac.id/library/index.php?option=com_content&view=article&id=410:serat-optik&catid=23:sistem-komunikasi-optik&Itemid=15

Perros, Harry G. 2005. *Connection-oriented Networks SONET/SDH, ATM, MPLS, and OPTICAL NETWORKS*. John Wiley & Sons Ltd, The Atrium, Southern Gate, Chichester, West Sussex PO19 8SQ, England.

Senior, John M. 1985. *Optical Fiber: Principles and Practice Second Edition*. Prentice-Hall. New Jersey.

Watuwibowo, Kuncoro. 2003. *Pengantar MPLS*.

<http://student.eepis-its.edu/~a12ix/data/networking/koen-mpls.pdf>.

