

BAB II

LANDASAN TEORI

II.1 Pengendali Jarak Jauh (*Remote Control*)

Pengendali jarak jauh atau *remote control* adalah suatu alat yang digunakan untuk mengontrol operasi dan perintah suatu perangkat atau mesin dari jarak jauh (http://id.wikipedia.org/wiki/Remote_control). Istilah pengendali jarak jauh atau *remote control* dapat juga ditunjukkan sebagai "*remote*" atau "*controller*" (http://id.wikipedia.org/wiki/Remote_control). Secara umum, *remote control* digunakan untuk mengeluarkan perintah - perintah dari jarak jauh ke perangkat - perangkat elektronik seperti televisi, stereo, VCD player, dll. *Remote control* yang digunakan untuk perangkat tersebut biasanya berupa perangkat *wireless* berukuran kecil yang memiliki beberapa tombol *array* untuk mengatur bermacam *setting* seperti *channel* televisi, nomor *track*, dan volume. Sebagian besar *remote* ini berkomunikasi dengan perangkat menggunakan sinyal *Infra - Red* (IR) dan beberapa telah dikembangkan menggunakan sinyal radio seperti *bluetooth*.

II.2 Bluetooth

II.2.1 Pengenalan Teknologi Bluetooth

Teknologi *bluetooth* adalah teknologi komunikasi yang ditujukan untuk menggantikan komunikasi dengan kabel antara *portable devices*, mesin desktop, dan *peripheral* yang menggunakan frekuensi radio sebagai media transmisi (<http://www.bluetooth.com>). Jangkauan

Teknologi *bluetooth* memungkinkan pengguna untuk melakukan komunikasi, bertukar data, melakukan sinkronisasi file tanpa harus menghubungkan perangkat dengan kabel dalam jangkauan kecil, suatu jaringan *ad hoc* yang disebut *piconet*. Digunakan jaringan *ad hoc* karena tidak mempunyai konfigurasi jaringan yang tetap. Dalam jaringan *bluetooth*, *piconet master* mengatur perubahan topologi di dalam jaringan. Disebabkan oleh perangkat yang selalu berubah, maka jaringan harus dapat melakukan konfigurasi ulang secara langsung untuk mengadaptasi topologi yang dinamik. *Routing* yang diterapkan protokol *bluetooth* memungkinkan master membuat dan mengatur perubahan jaringan. Sinyal pada *bluetooth* bersifat *omnidirectional*, dapat berpropagasi ke semua arah, jarak komunikasi lebih jauh, kemampuan sinyal yang melewati dinding.

Tabel II.1 Karakteristik Performa dari Teknologi Bluetooth

Fitur / Fungsi	Kemampuan Kerja
Tipe koneksi	<i>Spread spectrum (frequency hopping)</i>
<i>Spectrum</i>	2.4 GHz ISM band
Kekuatan Transmisi	1 milliwatt (mW)
Kecepatan Data	1 Mbps dengan <i>frequency hopping</i>
Jangkauan	Sampai dengan 3 meter
<i>Supported station</i>	Sampai dengan 8 perangkat untuk setiap <i>piconet</i>
Channel suara	Sampai dengan 3
Keamanan data	Untuk autentikasi, kunci 128 bit, untuk enkripsi, ukuran kunci antara 8 dan 128 bits

Pengalamatan	Setiap perangkat memiliki 48-bit <i>MAC address</i> yang digunakan untuk menetapkan koneksi dengan perangkat lain
--------------	---

II.2.1.1 Format Paket Bluetooth

Spesifikasi *bluetooth* menspesifikasikan kegunaan dari dua macam paket, yaitu SCO dan ACL. Paket SCO digunakan untuk *link* suara sinkron dan dirutekan untuk I/O (input/output) port suara sinkron. Tetapi paket ini tidak termasuk mekanisme *error - checking* dan tidak pernah bertransmisi ulang karena *delay* akan mengurangi kualitas suara.

Paket ACL digunakan untuk *link* asinkron. Informasi yang dibawa merupakan data *user* atau data kontrol. Karena data membawa *link* asinkron yang tidak sensitif akan *delay*, paket termasuk suatu mekanisme kontrol error dan transmisi ulang dapat digunakan untuk mengoreksi paket yang *corrupt* selama transmisi.

Format paket yang digunakan teknologi *bluetooth* terdiri dari tiga bagian, yaitu : *access code*, *header*, dan *payload* (Nathan J. Muller, 2001).



Gambar II.1 Format Paket Bluetooth

a. Access Code

Setiap paket dimulai dengan *access code*, yang berguna untuk sinyal. *Access code* berisi *preamble*, *sync word*, dan *trailer*. *Preamble* menunjukkan kedatangan paket ke *receiver*. *Sync word* digunakan untuk menghitung waktu

sinkronisasi dengan receiver. Sedang trailer ditambahkan ke sync word seperti header paket mengikuti access code. Jumlah bit access code bisa berbeda - beda, tergantung dengan header paket yang mengikuti. Jika header paket mengikuti, maka panjangnya 72 bits, selain itu hanya 68 bits.

Fungsi yang terdapat dalam access code dapat berbeda, tergantung mode operasi perangkat bluetooth. Terdapat tiga tipe access code, yaitu :

1. Channel Access Code (CAC), channel access code mengidentifikasi piconet.
2. Device Access Code (DAC), device access code digunakan untuk prosedur sinyal, seperti paging dan respon dari paging.
3. Inquiry Access Code (IAC), terdapat dua tipe inquiry access code yaitu general dan dedicated. General inquiry access code cocok untuk semua perangkat, sedang dedicated inquiry access code cocok untuk unit bluetooth yang sama karakteristiknya.

b. Header

Header berisi informasi link control (IC) dan terdiri dari enam fields, jumlah totalnya 18 bits.

Tabel II.2 Fields dalam Header

Fields	Jumlah Bit	Fungsi
Active Member Adres	3 bits	Membedakan active member yang berpartisipasi dalam piconet.
Type	4 bits	Untuk kode yang menspesifikasikan tipe paket.
Flow	1 bit	Aliran control dari dalam paket ACL link.
Automatic Repeat	1 bit	Memberitahukan kesuksesan

Request		transfer payload data.
Sequence Number	1 bit	Menghasilkan skema sequential number untuk menempatkan paket data stream tepat pada saat diterima alat penerima.
Header Error Check	8 bits	Mengecek integritas header.

c. Payload

Dalam *payload* terdapat dua tipe field, yaitu field suara (sinkron) dan field data (asinkron). Paket ACL hanya memiliki field data dan SCO paket hanya memiliki field suara. Pengecualian terdapat pada paket *Data Voice* (DV) yang memiliki kedua field tersebut. Field data terdiri dari tiga segmen :

1. Payload Header

Hanya field data yg memiliki *payload header*. Memiliki panjang 1 atau 2 bytes, dan menspesifikasikan *logical channel*, mengontrol aliran dalam *logical channel*, dan memiliki indikator panjang *payload* yang menunjukkan jumlah bytes dalam *payload body*, tidak termasuk *payload header* dan *CRC code*.

2. Payload Body

Berisikan informasi user. Panjang dari *payload body* ditunjukkan dalam panjang field dari *payload header*.

3. CRC Code Generation

Setelah generator CRC dikenali, kode 16-bit *cyclic redundancy check* dihitung dalam informasi untuk kemudian dikirim dan ditambahkan dalam informasi.

II.2.1.2 *Physical Links*

Dua macam link dapat dibuat antara *master* dengan satu atau lebih *slave* : *connection-oriented* dan *connectionless*. *Connection-oriented links* mengharuskan *session* dibuat sebelum suatu data dapat dikirim, sehingga data terjamin sampai sesuai permintaan. *Connectionless network* tidak mengharuskan *session* dibuat antara pengirim dan penerima, sehingga data bisa saja tidak sampai sesuai permintaan.

Perangkat *bluetooth* dapat menggunakan kedua link tersebut, lebih spesifik lagi yaitu *Synchronous Connection Oriented* (SCO) dan *Asynchronous Connectionless* (ACL).

a. **SCO Links**

Merupakan link *point-to-point* antara *master* dan *single slave* dalam *piconet*, sehingga bersifat simetris. Digunakan untuk mendukung informasi *time - bounded* seperti suara. Link ini dibangun oleh *master*, dengan mengirimkan pesan setup melalui protokol *Link Manager* (LM). Karena paket yang dibawa melalui SCO link berisi informasi yang *time - sensitive*, maka paket tidak pernah dikirim ulang.

b. **ACL Links**

ACL link menghasilkan koneksi yang *packet - switched* antara *master* dan semua *slave* aktif yang berada dalam *piconet*. Link ini dapat mendukung layanan sinkron dan asinkron, namun hanya *single link* yang dapat aktif. Paket dapat dikirim ulang untuk meyakinkan integritas data.

II.2.1.3 Logical Channel

Dalam sistem *bluetooth* terdapat lima *logical channels* untuk kontrol dan informasi user, yaitu :

- a. *Link Control* (LC)
- b. *Link Manager* (LM)
- c. *User Asynchronous* (UA)
- d. *User Isochronous* (UI)
- e. *User Synchronous* (US)

II.2.1.4 Jangkauan Operasi

Dari jangkauan operasinya, perangkat *bluetooth* dapat dibagi menjadi tiga kelas, seperti tampak pada Tabel II.2 di bawah ini.

Tabel II.3 Jangkauan Operasi Bluetooth

CLASS	POWER RATING	RANGE
Class 1	100 mW	100 meters
Class 2	2.5 mW	20 meters
Class 3	1 mW	10 meters

II.2.1.5 Kelebihan Bluetooth

Teknologi *bluetooth* adalah pilihan tepat untuk pemakaian komunikasi jarak dekat antar perangkat bebas kabel. Pemakaiannya telah menjadi standar koneksi tanpa kabel terhadap telepon seluler, *personal computer* (PC), mobil, *stereo headset*, *MP3 player*. *Bluetooth* memungkinkan produk tidak perlu menginstal *driver software*. Kini teknologi *bluetooth* telah mencapai spesifikasi versi ke - empat dan terus ditingkatkan kelebihannya, yaitu rendah energi, biaya murah, *built - in security*, berkekuatan tinggi, mudah pemakaiannya, dan kemampuannya menciptakan jaringan ad hoc (<http://www.bluetooth.com>).

a. Globally Available

Teknologi *bluetooth* telah tersedia secara bebas di seluruh dunia. Manufaktur dari banyak industri sibuk mengimplementasikan produk yang dapat mengurangi pemakaian kabel, koneksi tanpa interupsi, transfer data dan komunikasi dengan suara. Tidak dikenakan biaya untuk penggunaan *bluetooth*, tidak seperti berlangganan provider telepon GSM atau CDMA.

b. Jangkauan Teknologi Bluetooth

Teknologi *bluetooth* telah tersedia dalam berbagai aplikasi, mulai dari telepon seluler, mobil, bahkan perangkat kesehatan untuk semua konsumen, pasar industri, bahkan *enterprise*. Pemakaian rendah energi, ukuran kecil dan biaya murah untuk *chipsetnya*, membuat *bluetooth* dapat digunakan untuk perangkat paling kecil sekalipun.

c. Kemudahan Penggunaan

Bluetooth adalah teknologi ad hoc yang tidak membutuhkan suatu infrastruktur tertentu sehingga mudah dibuat. Tidak dibutuhkan kabel untuk melakukan koneksi. Proses untuk user baru sangat mudah, kita memiliki produk *bluetooth*, mengecek *profile* yang tersedia kemudian melakukan koneksi dengan perangkat *bluetooth* lain dengan *profile* yang sama. Seperti kita memiliki *Personal Area Network (PAN)* sendiri dan dapat melakukan koneksi dengan perangkat yang lain dengan mudah.

d. Koneksi yang Aman

Teknologi *bluetooth* didesain untuk kebutuhan keamanan. *Bluetooth* memiliki keamanan *built - in* seperti enkripsi 128 bit dan autentikasi PIN. Jadi saat produk *bluetooth* teridentifikasi, digunakan kode PIN saat

terhubung pertama kali. Sekali terhubung, maka koneksi akan aman.

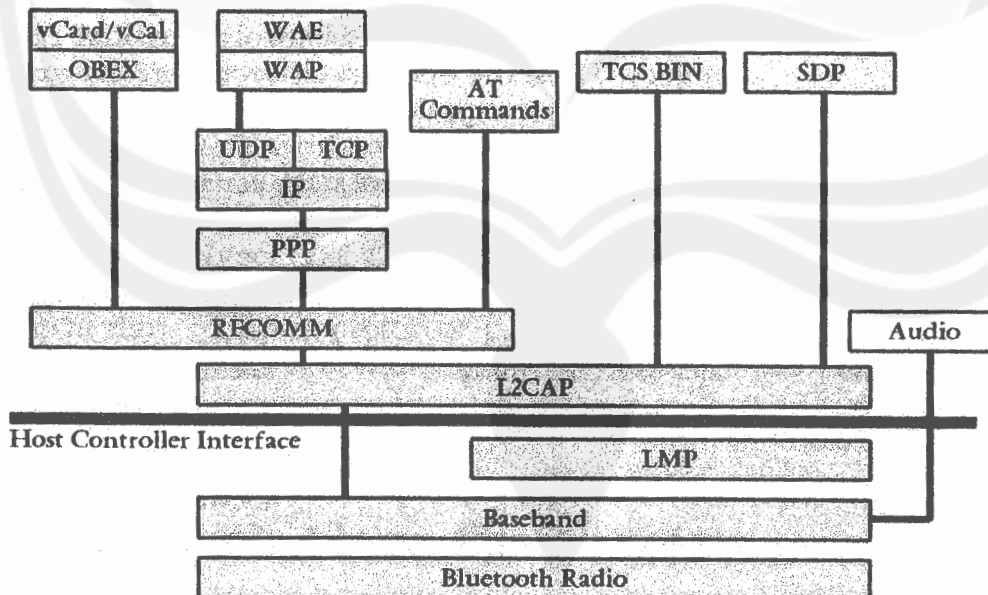
II.2.2 Arsitektur Protokol Bluetooth

II.2.2.1 Bluetooth Protocol Stack

Komputer yang terdiri dari prosesor, *memory*, bus, *hard drive*, dan peripheral lain, tetapi komputer tidak memiliki kemampuan untuk menjalankannya sendiri. Untuk itu dibutuhkan suatu *driver* yang berperan sebagai agen pengontrol untuk membantu komunikasi antara komputer dengan peripheral tersebut.

Sama halnya dengan *bluetooth stack* dan perangkat *bluetooth*. *Bluetooth stack* sebagai agen pengontrol yang mengimplementasikan protokol *bluetooth* dan juga mengizinkan kita untuk mengontrol perangkat *bluetooth* sendiri.

Dengan adanya *bluetooth protocol stack*, *bluetooth client* dan *server* untuk saling mengirim dan menerima data dalam jaringan *wireless*.



Gambar II.2 Bluetooth Protocol Stack

II.2.2.2 Protokol di Arsitektur Bluetooth

Bluetooth protokol stack terdiri dari empat layer. Layer dan protokol tersebut dapat dilihat dalam Tabel II.3.

Tabel II.4 Protokol dan Layer dalam *Bluetooth Protocol Stack*

Bluetooth Protocol Layer	Member of the Protocol Stack
Bluetooth Core Protocol	<ul style="list-style-type: none"> • Baseband • Link Management Protocol (LMP) • Logical Link Control and Adaptation Layer (L2CAP) • Service Discovery Protocol (SDP)
Cable Replacement Protocol	<ul style="list-style-type: none"> • Radio Frequency Communication (RFCOMM)
Telephony Control Protocol	<ul style="list-style-type: none"> • Telephony Control Specification Binary (TCS BIN) • AT-Commands
Adopted Protocol	<ul style="list-style-type: none"> • Point-to-Point Protocol (PPP) • UDP/TCP/IP • OBEX • WAP • vCard • vCalender • IrMC • WAE

Selain itu terdapat *Host Controller Interface (HCI)* yang melakukan perintah - perintah yang menghubungkan *baseband controller*, *link manager*, dan akses ke status *hardware* dan *control register*. HCI merupakan suatu lapisan perangkat lunak yang melewati semua data dari komputer ke perangkat *bluetooth*.

II.2.2.3 *Bluetooth Core Protocol*

Core protocol telah dikembangkan oleh Bluetooth SIG. *Core protocol*, juga *bluetooth radio* dibutuhkan oleh sebagian besar perangkat *bluetooth*, sedang protokol lain hanya digunakan jika diperlukan saja.

a. *Baseband*

Baseband layer memungkinkan *physical RF link* antara unit - unit *bluetooth* membentuk *piconet*. Sistem RF *bluetooth* adalah sistem *Frequency - Hopping Spread Spectrum* (FHSS) dimana paket ditransmisikan dalam *time slot* dan frekuensi yang telah ditetapkan. Lapisan ini menggunakan prosedur *inquiry* dan *paging* untuk mensinkronkan frekuensi *transmission - hopping* dan *clock* dari perangkat *bluetooth* yang berbeda.

b. *Link Manager Protocol* (LMP)

LMP bertanggung jawab untuk *link setup* dan kontrol antara perangkat *bluetooth*, termasuk kontrol dan negosiasi ukuran paket *baseband*. Hal ini termasuk aspek keamanan seperti autentikasi dan enkripsi. LMP juga mengatur daya dan kinerja dari perangkat radio *bluetooth* dan kondisi hubungan suatu unit *bluetooth* di dalam *piconet*.

c. *Logical Link Control and Adaptation Protocol* (L2CAP)

L2CAP merupakan adaptasi terhadap protokol di layer atas melalui *baseband*. L2CAP dapat bekerja secara paralel dengan LMP dimana L2CAP melakukan layanan ke layer atas ketika isi data tidak terkirim pada pesan LMP. L2CAP menyediakan layanan data *connection - oriented* dan *connectionless* ke protokol di layer atas dengan kemampuan *multiplexing*, *segmentation* dan *reassembly*, dan *group abstraction*. L2CAP memperbolehkan protokol di level atas untuk mengirim dan menerima paket data L2CAP yang besarnya sampai 64 Kbytes. L2CAP hanya diperuntukkan bagi saluran ACL.

d. *Service Discovery Protocol* (SDP)

SDP merupakan elemen penting dalam *bluetooth*. Layanan ini menyediakan dasar - dasar dari model penggunaan

bluetooth. Dengan SDP, informasi perangkat, layanan, karakteristik layanan dapat diminta dan setelah itu hubungan antara dua atau lebih perangkat *bluetooth* dapat dikenali.

II.2.2.4 *Cable Replacement Protocol*

a. RFCOMM

RFCOMM adalah protokol emulasi saluran serial berbasis pada ETSI 07.10, yang juga digunakan untuk perangkat komunikasi GSM. Protokol pengganti kabel ini berperan sebagai kontrol RS-232 dan sinyal data melalui baseband yang memberikan kemampuan untuk layanan level atas yang menggunakan saluran serial sebagai mekanisme transport.

II.2.2.5 *Telephony Control Protocol*

Telephony Control Protocol Binary (TCS Binary atau TCS BIN) adalah protokol *bit - oriented* yang melakukan *call control signaling* untuk pembentukan hubungan data dan suara antara perangkat *bluetooth*. Selain itu juga melakukan prosedur pengaturan mobilitas untuk menangani sekelompok perangkat TCS Bluetooth.

Bluetooth SIG juga mendefinisikan *AT commands* yang melakukan pengontrolan terhadap telepon seluler dan modem dalam *multiple usage model*. Untuk layanan faksimile, *AT commands* dispesifikasikan dalam implementasi yang dibedakan sebagai berikut :

- a. Fax Class 1.0 TIA-578-A dan ITU T.31 Service Class 1.0
- b. Fax Class 2.0 TIA-592 dan ITU T.32 Service Class 2.0
- c. Fax Service Class 2 - tanpa standar industri

II.2.2.6 Adopted Protocol

a. *Point - to - Point Protocol (PPP)*

Dibuat berdasarkan IETF, standar ini menunjukkan bagaimana IP datagram dikirim melalui link serial *point - to - point*.

b. TCP/UDP/IP

Transmission Control Protocol (TCP), *User Datagram Protocol (UDP)*, dan *Internet Protocol (IP)* dibuat berdasarkan IETF dan digunakan untuk komunikasi melalui internet.

c. *OBEX Protocol*

Merupakan protokol *session layer* yang dibuat oleh IrDA sebagai IrOBEX. Digunakan untuk membantu pertukaran objek secara sederhana dan spontan dengan menggunakan model *client - server* dan tidak tergantung pada mekanisme transport.

d. *Wireless Application Protocol (WAP)*

Merupakan spesifikasi untuk mengirim dan membaca isi internet dan pesan dalam perangkat *wireless* ukuran kecil.

II.2.3 Sistem Keamanan Bluetooth

Keamanan data merupakan aspek penting dalam semua aplikasi jaringan. Semua data yang berpindah bersifat rentan terhadap serangan, tidak peduli apakah koneksi fisik atau *wireless*. Pengembang harus sadar bahwa tidak ada cara yang dapat meyakinkan informasi yang dikirim secara *wireless* hanya akan sampai ke tempat yang dituju.

Spesifikasi *bluetooth* menempatkan keamanan dalam tiga cara, yaitu :

1. Autentikasi (Authentication)

Autentikasi dengan *bluetooth* terdiri dari mekanisme *challenge/response* yang sederhana. Spesifikasi *bluetooth* tidak termasuk untuk autentikasi user, hanya autentikasi perangkat saja. *Challenge/response* tidak melibatkan *username* dan *password* seperti skema autentikasi biasanya. Autentikasi dengan *bluetooth* hanya melibatkan *personal identification number* (PIN). Selama proses autentikasi, kode PIN tidak pernah dikirim dari *client* ke *server*. Untuk tujuan autentikasi, *client* menciptakan 128 bit *shared link key*, yang diperoleh dari PIN. Jika kode PIN dari *client* dan *server* tidak cocok, maka proses autentikasi gagal.

a. Server Authentication

Bluetooth server dapat meminta autentikasi dengan menambahkan parameter autentikasi ke koneksi URL String. Jika parameter autentikasi diset *true*, maka implementasi JSR-82 mencoba memaksa autentikasi saat perangkat terhubung ke *server*. Jika parameter diset *false*, maka implementasi tidak akan mencoba melakukan autentikasi terhadap *client*.

b. Client Authentication

Bluetooth client dapat juga meminta server melakukan autentikasi dengan melakukan setting terhadap parameter autentikasi menjadi *true*. Jika parameter diset *false* maka *client* tidak dapat meminta server melakukan autentikasi.

2. Bluetooth Server Authorization

Bluetooth server dapat meminta bahwa hanya perangkat yang memiliki kuasa untuk melakukan koneksi dan

menggunakan pelayanan istimewa. Perangkat pemegang kuasa ini disebut *trusted devices*. Server dapat meminta *client* untuk mendapat otorisasi dengan melakukan setting pada parameter otorisasi, apakah dipilih *true* atau *false*.

3. Enkripsi (*Encryption*)

Autentikasi dan otorisasi akan baik jika digunakan untuk mencegah user yang tidak diinginkan supaya tidak mengakses layanan dari perangkat *bluetooth*. Tetapi, kedua aturan keamanan tersebut tidak menjamin akan melindungi data dari *hacker*. Enkripsi, merupakan aturan keamanan yang dapat digunakan untuk melindungi perpindahan data yang sensitif.

Untuk melakukan enkripsi, diperlukan algoritma enkripsi, kunci enkripsi, dan informasi yang akan dilindungi. Algoritma enkripsi adalah prosedur sederhana untuk mengacak data. Kunci enkripsi adalah kode sederhana yang digunakan dalam algoritma untuk enkripsi data. Dalam enkripsi simetris, kunci yang sama digunakan untuk enkripsi dan dekripsi data. Dalam enkripsi asimetris, dua kunci digunakan untuk enkripsi dan dekripsi data.

II.3 Remote Personal Computer dengan Bluetooth

Remote personal computer adalah suatu sistem pengendali jarak jauh atau *remote control* yang khusus digunakan untuk mengontrol *personal computer*. Remote *personal computer* ini akan berkomunikasi dengan protokol komunikasi tanpa kabel yaitu *bluetooth*.

Bluetooth merupakan teknologi pengganti komunikasi dengan kabel. Dibandingkan dengan *infrared*, dimana perangkat yang berkomunikasi harus berjarak dekat dan *infrared receivers* harus tepat berhadapan secara "eye to eye" karena jika tidak maka transmisi akan gagal. *Bluetooth* mengatasi keterbatasan tersebut dengan memiliki jarak sampai 10 meter, kelebihan kedua karena sinyal *bluetooth* seperti radio, sehingga transmisi bersifat *omnidirectional* atau ke segala arah. Selain itu, *bluetooth* cocok digunakan untuk menghubungkan perangkat kecil seperti telepon selular dengan jarak 30 feet dengan kecepatan 1 Mb/detik, tidak seperti protokol *wireless LAN* (802.11b) yang menghubungkan dua perangkat besar berkekuatan besar dengan kecepatan tinggi. Oleh karena kecepatan data yang lebih rendah dan jarak yang lebih pendek itu maka *bluetooth* merupakan teknologi *wireless* bertenaga paling rendah.

Sebagai *remote control* digunakan perangkat bergerak telepon selular, terlebih karena kebanyakan telepon selular kini telah mendukung *bluetooth* dan mampu melakukan komunikasi dua arah dalam pertukaran data. Namun telepon selular harus telah mendukung teknologi Java dan JSR - 82 Bluetooth Java API.

Remote *personal computer* ini mampu melakukan komunikasi dan memberikan perintah dari telepon selular ke *personal computer*, seperti *shut - down*, *restart*, mengontrol aplikasi, *run command* dan mengirimkan pesan. Sistem ini

terdiri dari dua program, yaitu untuk *server* dan *client*. Program untuk *server* akan berjalan di *personal computer* dan menangani permintaan dari *client*, sedang program untuk *client* akan berjalan di telepon selular dan akan mengirimkan permintaan dari *user* ke *personal computer*.

II.4 Teknologi Wireless Java

Teknologi *wireless* dapat dibagi menjadi dua, yaitu untuk lokal dan untuk area yang luas. Untuk jaringan yang bersifat lokal hanya bekerja dalam jangkauan area yang tidak terlalu jauh, contohnya *remote control*, telepon *cordless* dan jaringan *wireless*. Sementara untuk area yang luas, jangkauannya jauh lebih jauh, contohnya *handphone*, PDA dan sejenisnya. Hal ini dikarenakan perangkat komunikasi *wireless* seperti *handphone* menggunakan jaringan yang berupa *cell - tower*, sehingga layanan diterima dari sebuah *wireless carrier* atau perusahaan yang mengoperasikan *cell - tower* tersebut.

Aplikasi komunikasi bergerak, dalam perkembangannya masing - masing *vendor* memiliki *platform* aplikasi dan sistem operasi sendiri. Namun perbedaan aplikasi ini menyebabkan suatu *platform* aplikasi maupun sistem operasi tidak dapat dijalankan untuk setiap *vendor*. Sehingga menyulitkan untuk perkembangan - perkembangan aplikasi baru.

Standarisasi yang dilakukan untuk membuat suatu bahasa pemrograman yang memiliki kebebasan *platform*. Salah satunya adalah teknologi Java yang memungkinkan untuk menulis perangkat lunak yang dapat berjalan di perangkat dan platform yang berbeda tanpa harus melakukan modifikasi, yang menciptakan ide "*write once, run everywhere*". Untuk

komunikasi bergerak, *Sun MicroSystem* mengenalkan *Java 2 Micro Edition (J2ME)* yang merupakan salah satu bagian teknologi Java yang digunakan untuk aplikasi Java yang berjalan pada *mobile device* dan perangkat jaringan *wireless*.

II.4.1 Java Virtual Machine (JVM)

Java Virtual Machine adalah perangkat lunak yang berfungsi untuk menjalankan program Java supaya dapat dimengerti oleh komputer. *Compiler* Java mengubah bahasa pemrograman Java menjadi suatu *Java bytecodes*. *Bytecodes* merupakan instruksi mesin komputer yang abstrak yang disebut *virtual machine*. JVM mengartikan *Java bytecodes* supaya dapat menjalankan suatu program Java, untuk itu bisa disebut suatu *interpreter*. JVM merupakan basis dari *Java platform* dan menjembatani antara *bytecode* dengan perangkat keras.

II.4.2 Java 2 Platform

II.4.2.1 Java 2 Platform, Standard Edition (J2SE)

Didesain untuk berjalan di *desktop* dan *workstation computers*. Platform ini berisi *class - class* inti pada Java dan *Graphic User Interface (GUI)*.

II.4.2.2 Java 2 Platform, Enterprise Edition (J2EE)

Memiliki dukungan *built - in* yang berisi *class - class* dan *interface - interface* untuk mengembangkan aplikasi Java berbasis web seperti *Servlets*, *Java Server Pages (JSP)*, dan *XML*. Edisi ini ditujukan untuk aplikasi berbasis server.

II.4.2.3 Java 2 Platform, Micro Edition (J2ME)

Didesain untuk perangkat dengan memori terbatas, tampilan, dan kekuatan proses seperti *handphone*, *Personal Digital Assistant* (PDA) dan *PocketPC*.

II.4.3 Java 2 Micro Edition (J2ME)

J2ME memungkinkan aplikasi Java untuk berjalan pada perangkat *mobile* dan berukuran kecil. Merupakan adaptasi dari teknologi Java dan diperkenalkan untuk memenuhi kebutuhan konsumen di luar J2SE dan J2EE. Komponen penting yang terdapat dalam J2ME adalah *configuration* dan *profile*.

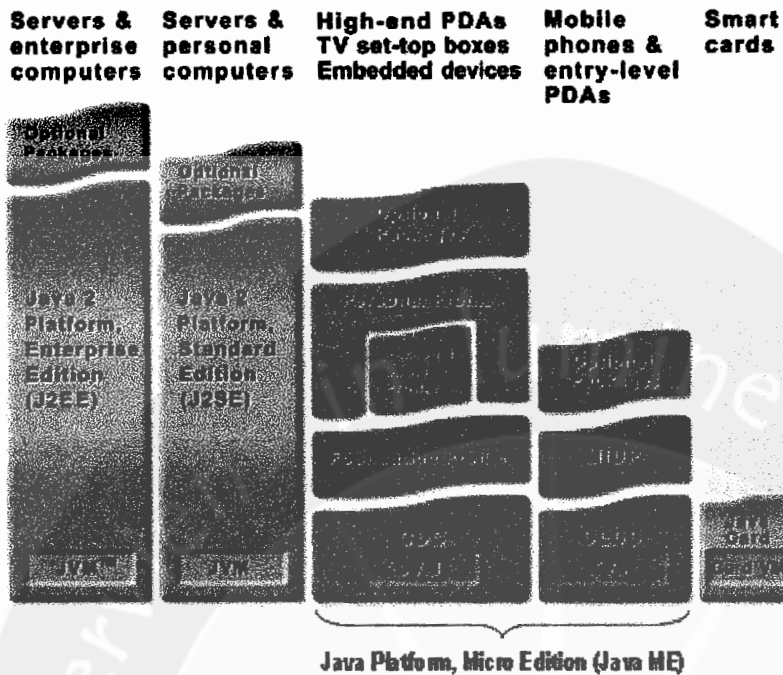
II.4.3.1 Lapisan Konfigurasi (*Configuration*)

Lapisan konfigurasi menunjukkan dasar J2ME *runtime environment*, berupa fitur Java dan dasar Java *libraries*. Konfigurasi ini berhubungan erat dengan Java *Virtual Machine* (JVM). J2ME memiliki dua konfigurasi, yaitu :

- a. *Connected Device Configuration* (CDC), mendukung aplikasi Java pada perangkat dengan 32-bit *processor* dan memori penyimpanan lebih dari 2 MB.
- b. *Connected Limited Device Configuration* (CLDC), perangkat dengan 16-bit atau 32-bit *processor* dan memori 160-512 KB.

II.4.3.2 Lapisan Profil (*Profile*)

- a. MIDP (*Mobile Information Device Profile*), menyediakan *libraries* Java untuk implementasi dasar antarmuka (GUI), implementasi jaringan (*networking*), *database*, dan *timer*. MIDP dibangun di atas CLDC.
- b. *Foundation Profile*, dibangun di atas CDC.
- c. *Personal Profile*.



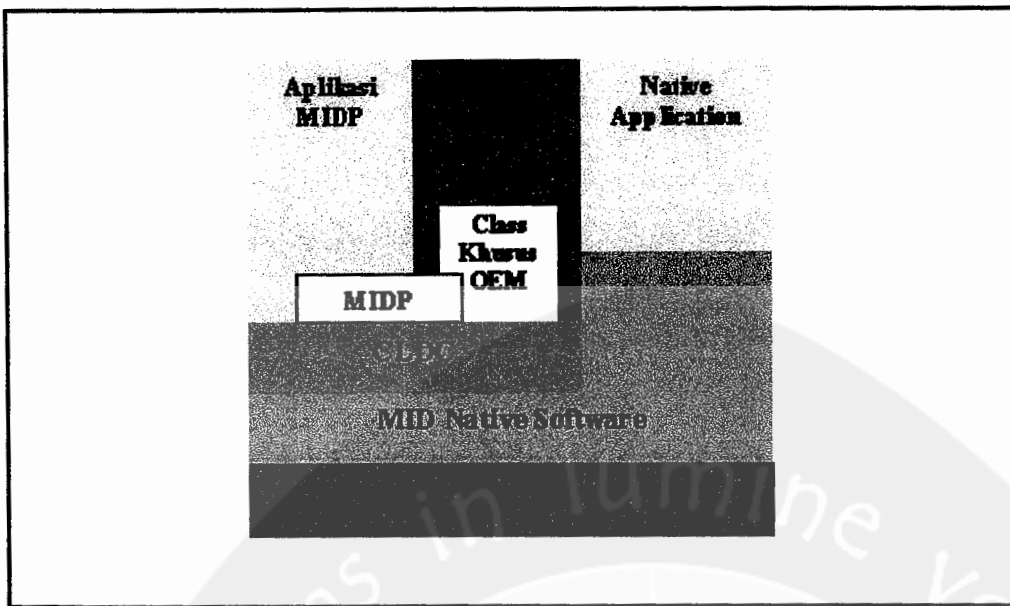
Gambar II.3 Macam - macam Java Edition

II.4.4 MIDlets

Aplikasi yang berjalan pada perangkat yang mendukung MIDP, disebut MIDlets. MIDlets merupakan aplikasi yang dibuat menggunakan *Java 2 Micro Edition* dengan CLDC dan profile MIDP. MIDlets terdiri atas :

- a. *Descriptor* (atribut aplikasi)
- b. *Java class*
- c. *Resource*

Pada Gambar II.4 menunjukkan bahwa pendukung aplikasi MIDP adalah aplikasi MIDlets (<http://java.sun.com>).

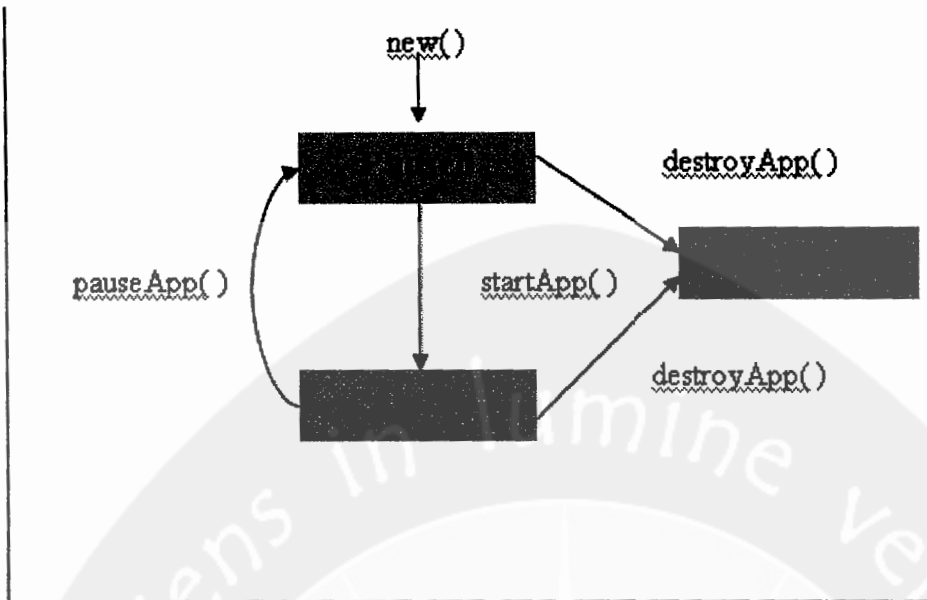


Gambar II.4 Arsitektur Aplikasi MIDP

II.4.4.1 Daur Hidup (*LifeCycle*) MIDlets

Lifecycle dari sebuah MIDlet ditangani oleh *Application Management Software* (AMS). AMS adalah sebuah lingkungan tempat siklus dari sebuah MIDlet, mampu untuk diciptakan, dijalankan, dihentikan maupun dihilangkan. AMS sering pula disebut dengan *Java Application Manager* (JAM). MIDlet memiliki beberapa *state*, yaitu *Pause*, *Active* dan *Destroy*. Ketika masing - masing *state* dipanggil, beberapa *method* yang bersesuaian dipanggil. Metode - metode tersebut merupakan bawaan dari J2ME. Untuk menjelaskan proses MIDlet dalam *Java Application Manager* (JAM) adalah pada Gambar II.4.

II.4.4.2 Status MIDlets



Gambar II.5 LifeCycle dan Perubahan Status MIDlets

II.4.5 Graphical User Interface (GUI)

Fungsi GUI pada CLDC ditangani oleh MIDP. *Library interface* MIDP dibagi atas dua yaitu *high-level API* dan *low-level API*.

User interface dengan *High-level API* dapat diimplementasikan berbeda oleh perangkat yang berbeda. *Low-level API* memungkinkan desainer "melukis" pada level *pixel*, desainer mempunyai kontrol yang lebih besar atas bagaimana tampilan *user interface*.

II.5 Java'S API for Bluetooth Wireless Technology (JABWT)

Dalam Java, API untuk *bluetooth* didefinisikan dalam JSR-82 (*Java Specification Request 82*) berdasarkan *Bluetooth Specification 1.1*. Tujuan dari spesifikasi ini adalah mendefinisikan arsitektur dan API yang dibutuhkan untuk memungkinkan pengembangan aplikasi *bluetooth* pada perangkat yang mengadopsi J2ME dan *Bluetooth - enabled* oleh pihak ketiga.

II.5.1 Kebutuhan Perangkat dan Sistem Bluetooth

- a. Total memori minimum 512 KB (ROM dan RAM).
- b. Perangkat komunikasi *bluetooth*.
- c. *Compliant* implementasi CLDC atau *superset* dari CLDC, contohnya CDC.
- d. Sistem *bluetooth* minimal mengimplementasikan *Generic Access Profile*, *Service Discovery Application Profile*, *Generic Object Exchange Profile* (GOEP) dan *Serial Port Profile*.
- e. Sistem mendukung *Service Discovery Protocol* (SDP), RFCOMM, L2CAP.
- f. Sistem harus menyediakan *Bluetooth Control Center* (BCC).

II.5.2 Ruang Lingkup JSR - 82

Kemampuan-kemampuan yang disediakan API JSR-82 :

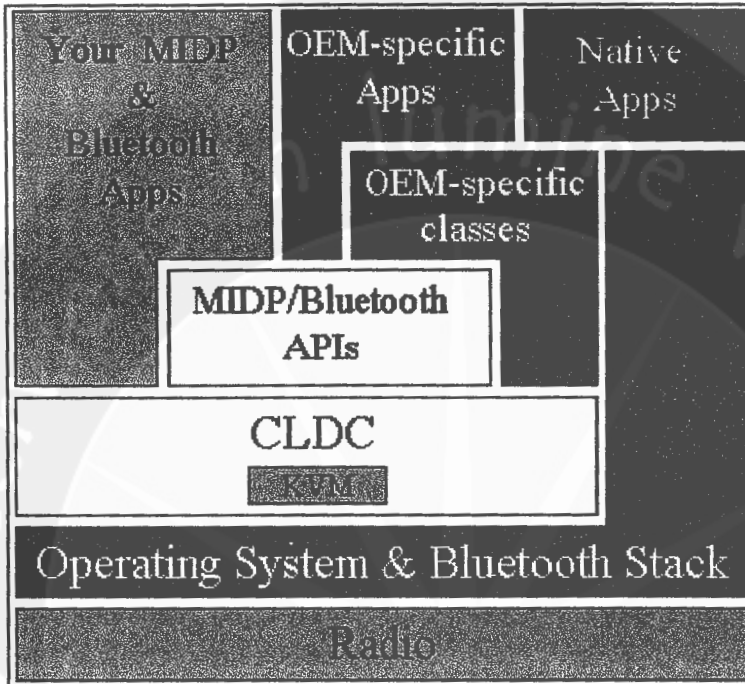
- a. Register layanan.
- b. Transmisi data.
- c. *Discover* perangkat dan layanan.
- d. Membentuk koneksi RFCOMM, L2CAP dan OBEX.

II.5.3 Arsitektur API JSR - 82

JSR-82 untuk *bluetooth* mendefinisikan dua *Package*, yaitu :

1. `javax.bluetooth` : Core API Bluetooth
2. `javax.obex` : API untuk *protocol* OBEX

Implementasi API dapat menggunakan hanya salah satu *package* atau kedua-duanya.



Gambar II.6 Posisi API JSR-82 dalam CLDC/MIDP

II.5.4 Bluetooth Control Center (BCC)

BCC adalah pusat otoritas untuk mengatur perangkat *bluetooth* lokal. BCC menyediakan satu set fungsi yang memungkinkan pengguna atau OEM menentukan nilai tertentu untuk parameter konfigurasi tertentu pada *bluetooth stack*. BCC dapat berupa *native application*, atau aplikasi dengan API sendiri, atau sekumpulan aturan yang tidak dapat diubah oleh pengguna.

BCC harus menyediakan implementasi API untuk fungsi-fungsi berikut :

- a. *Security setting* yang diperlukan oleh perangkat *local*.

- b. Daftar perangkat *pre-know*. Perangkat *pre-known* adalah perangkat-perangkat yang ditentukan *user*, biasanya perangkat yang sering diajak berkomunikasi.
- c. Mekanisme saat dua perangkat akan saling berinteraksi.
- d. Mekanisme untuk menyediakan otoritas melakukan hubungan.
- e. Perangkat yang *user - friendly*.

