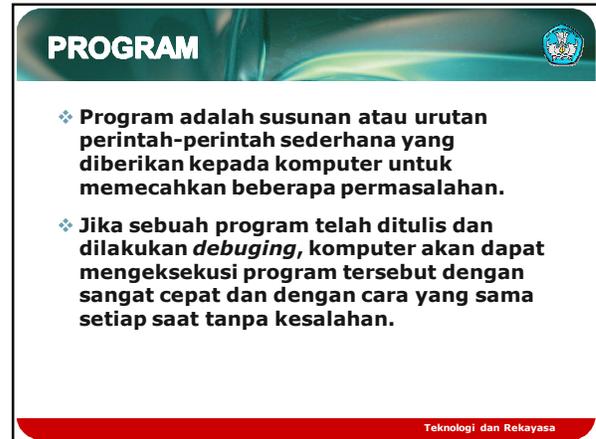




**Microprocessor & Microcontroller  
Programming**

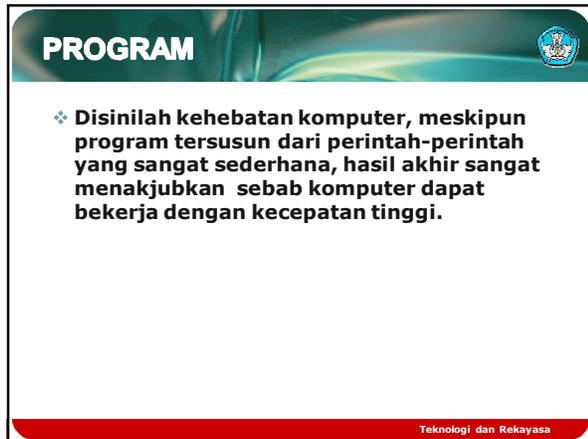
**BAB V**  
BAHASA DAN PENGEMBANGAN  
PROGRAM MIKROPROSESOR



**PROGRAM**

- ❖ Program adalah susunan atau urutan perintah-perintah sederhana yang diberikan kepada komputer untuk memecahkan beberapa permasalahan.
- ❖ Jika sebuah program telah ditulis dan dilakukan *debuging*, komputer akan dapat mengeksekusi program tersebut dengan sangat cepat dan dengan cara yang sama setiap saat tanpa kesalahan.

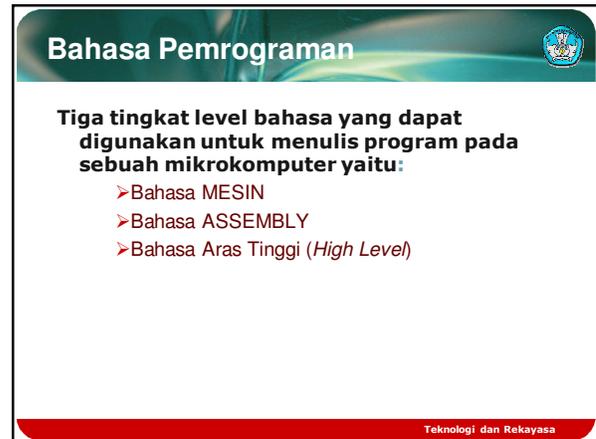
Teknologi dan Rekayasa



**PROGRAM**

- ❖ Disinilah kehebatan komputer, meskipun program tersusun dari perintah-perintah yang sangat sederhana, hasil akhir sangat menakjubkan sebab komputer dapat bekerja dengan kecepatan tinggi.

Teknologi dan Rekayasa

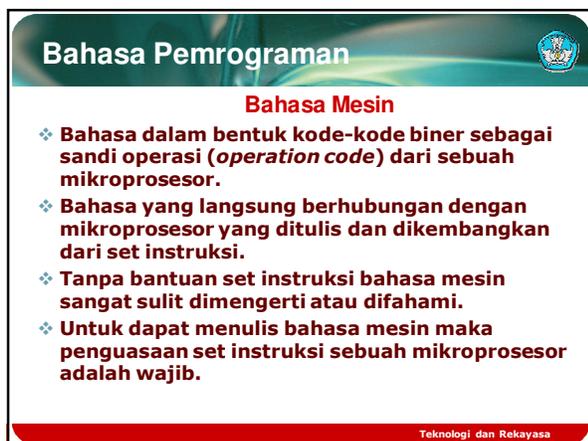


**Bahasa Pemrograman**

Tiga tingkat level bahasa yang dapat digunakan untuk menulis program pada sebuah mikrokomputer yaitu:

- Bahasa MESIN
- Bahasa ASSEMBLY
- Bahasa Aras Tinggi (*High Level*)

Teknologi dan Rekayasa

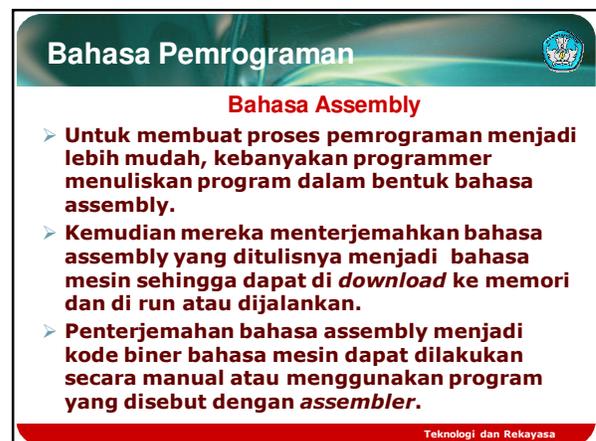


**Bahasa Pemrograman**

**Bahasa Mesin**

- ❖ Bahasa dalam bentuk kode-kode biner sebagai sandi operasi (*operation code*) dari sebuah mikroprosesor.
- ❖ Bahasa yang langsung berhubungan dengan mikroprosesor yang ditulis dan dikembangkan dari set instruksi.
- ❖ Tanpa bantuan set instruksi bahasa mesin sangat sulit dimengerti atau difahami.
- ❖ Untuk dapat menulis bahasa mesin maka penguasaan set instruksi sebuah mikroprosesor adalah wajib.

Teknologi dan Rekayasa



**Bahasa Pemrograman**

**Bahasa Assembly**

- Untuk membuat proses pemrograman menjadi lebih mudah, kebanyakan programmer menuliskan program dalam bentuk bahasa assembly.
- Kemudian mereka menterjemahkan bahasa assembly yang ditulisnya menjadi bahasa mesin sehingga dapat di *download* ke memori dan di run atau dijalankan.
- Penterjemahan bahasa assembly menjadi kode biner bahasa mesin dapat dilakukan secara manual atau menggunakan program yang disebut dengan *assembler*.

Teknologi dan Rekayasa

## Bahasa Pemrograman

### Bahasa Assembly

- ❖ Bahasa assembly menggunakan sejumlah mnemonik untuk merepresentasi kan instruksi–instruksi. Mnemonik adalah singkatan dari suatu perintah atau instruksi sebagai piranti untuk membantu ingatan. Sebagai contoh :
- Load disingkat LD
- Add → ADD
- Add With Carry → ADC
- Subtract → SUB
- Subtract With Carry → SBC
- Complement → CPL

Teknologi dan Rekayasa

## Bahasa Pemrograman

### Pernyataan bahasa assembly

Label	Mnemonik	Operand	Komentar
Mulai:	LD	A, 3F	Isi Register A dengan data 3Fh
	LD	B, 5D	Isi Register B dengan data 5Dh
	ADD	A,B	Jumlahkan data A dengan data B

Teknologi dan Rekayasa

## Bahasa Pemrograman

- ❖ Label adalah simbol atau kelompok simbol yang digunakan untuk merepresentasikan alamat yang tidak diketahui secara spesifik pada saat pernyataan-pernyataan ditulis.
- ❖ Label tidak dipersyaratkan dalam setiap pernyataan.
- ❖ Label dimasukkan bila diperlukan saja.

Teknologi dan Rekayasa

## Bahasa Pemrograman

- ❖ Mnemonik adalah kolom singkatan dari setiap perintah bahasa assembly.
- ❖ Bagian operand dapat memuat nama Register, alamat memori, alamat port, atau data immediate dari sebuah instruksi.
- ❖ Operand adalah sasaran dari instruksi.
- ❖ Pada bagian operand terbagi menjadi dua bagian yaitu sumber data yang disebut dengan source dan tujuan data atau destinasi.
- ❖ Pada umumnya source ada disebelah kanan dari destinasi.
- ❖ Bagian komentar biasanya digunakan untuk memberi penjelasan singkat maksud atau sasaran dari instruksi disebelah kirinya.

Teknologi dan Rekayasa

## Bahasa Pemrograman

- ❖ Jika kita sudah menulis program dalam bahasa assembly, pertanyaannya bagaimana menterjemahkan menjadi bahasa mesin yang siap dieksekusi ke sistem mikroprosesor.
- ❖ Ada dua jawaban atas pertanyaan ini.
- ❖ Pertama secara manual dengan menterjemahkan setiap instruksi bahasa assembly menjadi bahasa mesin melalui set instruksi.
- ❖ Kemudian kedua secara otomatis dengan menggunakan **ASSEMBLER**.
- ❖ Assembler adalah sebuah program yang dapat dijalankan pada sebuah mikrokomputer atau pada sebuah *Microcomputer Development System*.

Teknologi dan Rekayasa

## Bahasa Aras Tinggi

- ❖ Cara lain dalam menulis program pada mikrokomputer adalah dengan bahasa aras tinggi seperti Bahasa C, bahasa Pascal, BASIC, FORTRAN, dan sebagainya. Secara umum bahasa aras tinggi dapat dikelompokkan menjadi dua yaitu sebagai *Compiler* atau *Interpreter*.

Teknologi dan Rekayasa

## Tools Pengembangan Program Bahasa Assembly

### Editor

- ❖ program yang digunakan untuk mengetik atau mengedit program dalam bentuk pernyataan-pernyataan.
- ❖ Contoh program editor adalah TV Demo, EDLIN, ALTER. Fungsi utama program editor adalah untuk membantu programmer dalam membangun program dalam bahasa assembly dalam bentuk format yang benar sehingga assembler dapat menterjemahkan menjadi bahasa mesin yang benar. Hasil atau bentuk program hasil dari editor disebut *Source Program*.

Teknologi dan Rekayasa

## Tools Pengembangan Program Bahasa Assembly

### Assembler

- ❖ program yang digunakan untuk menterjemahkan mnemonik bahasa assembly menjadi kode biner yang benar untuk setiap instruksinya.
- ❖ Assembler akan membaca File Source Program di disk dimana source program tersebut disimpan setelah diedit menggunakan editor.
- ❖ Assembler biasanya membaca file source program lebih dari sekali.
- ❖ Pertama untuk menentukan penggantian nama item data dan offset dari sebuah label, dan meletakkan informasi tersebut dalam sebuah *Symbol Table*.
- ❖ Kemudian yang kedua assembler membangkitkan kode biner (Op-code) untuk setiap instruksi.

Teknologi dan Rekayasa

## Tools Pengembangan Program Bahasa Assembly

### Assembler

- ❖ Assembler membangkitkan dua file yaitu *Object File* yang berisi kode biner untuk instruksi-instruksi dan informasi tentang alamat instruksi.
- ❖ File yang kedua adalah *Assembler List File* yang memuat pernyataan bahasa assembly, kode biner untuk setiap instruksi, dan offset untuk setiap instruksi.
- ❖ File yang kedua ini yang biasanya di cetak ke printer sebagai pegangan pada saat melakukan testing dan troubleshooting.

Teknologi dan Rekayasa

## Tools Pengembangan Program Bahasa Assembly

### Linker

- ❖ program yang digunakan untuk menggabungkan beberapa object file menjadi satu object file yang lebih besar.
- ❖ Biasanya dalam membangun program yang sangat besar agar lebih efisien program tersebut dipecah dan dibagi-bagi menjadi beberapa program yang lebih kecil yang disebut dengan *Modul*.
- ❖ Setiap modul dapat ditulis, ditest, dan di debugged secara individual. Jika semua modul-modul telah berfungsi dengan baik, selanjutnya dapat di *linked* menjadi program dalam fungsinya yang besar.

Teknologi dan Rekayasa

## Tools Pengembangan Program Bahasa Assembly

### Linker

- ❖ Linker menghasilkan *Link file* yang berisi kode biner untuk semua modul-modul yang dikombinasikan.
- ❖ Linker juga menghasilkan *Link map file* yang berisi informasi alamat tentang file-file yang digabungkan.
- ❖ Linker tidak menetapkan alamat absolut pada program, ia hanya menetapkan alamat relatif mulai nol.
- ❖ Bentuk dari program ini dikatakan sebagai *relocatable* = dapat direlokasi, sebab dapat ditempatkan disetiap tempat di memori untuk dijalankan.

Teknologi dan Rekayasa

## Tools Pengembangan Program Bahasa Assembly

### Lokator

- ❖ Lokator adalah program yang digunakan untuk menetapkan alamat spesifik dimana object code diletakkan dalam memori.
- ❖ Lokator pada DOS IBM PC disebut EXE2BIN.

Teknologi dan Rekayasa

## Tools Pengembangan Program Bahasa Assembly

### Debugger

- ❖ Jika program kita tidak mempersyaratkan adanya perangkat keras luar atau hanya perangkat keras internal saja yang diakses secara langsung, kita membutuhkan debugger untuk menjalankan program tersebut.
- ❖ Debugger adalah sebuah program yang dapat digunakan untuk men-load program dalam bentuk object code pada sistem memori, mengesekusi, dan melakukan troubleshooting atau *debug*.
- ❖ Dalam hal ini melakukan pelacakan ada tidaknya kesalahan dalam sebuah program.

Teknologi dan Rekayasa

## Tools Pengembangan Program Bahasa Assembly

### Debugger

- ❖ Debugger menyediakan fasilitas untuk melihat isi register dan isi lokasi memori setelah sebuah instruksi dari program dijalankan. Disamping juga ada fasilitas untuk mengganti isi register dan data suatu lokasi memori.

Teknologi dan Rekayasa

## Tools Pengembangan Program Bahasa Assembly

### Emulator

- ❖ Cara lain untuk menjalankan sebuah program adalah menggunakan emulator.
- ❖ Emulator adalah gabungan diantara hardware dan software.
- ❖ Emulator biasanya digunakan untuk mengetes dan debug hardware dan software dari sebuah sistem eksternal seperti prototype dari sebuah instrumen berbasis mikroprosesor.

Teknologi dan Rekayasa

## Langkah-Langkah Pengembangan Program

- ❖ Menurut Douglas ada empat langkah yang harus diperhatikan dan dilakukan dalam mengembangkan sebuah program komputer. Keempat langkah itu adalah:
  1. Pendefinisian permasalahan.
  2. Representasi kerja program.
  3. Penemuan instruksi-instruksi yang benar, dan
  4. Penulisan program.

Teknologi dan Rekayasa

## Langkah-Langkah Pengembangan Program

### 1. Pendefinisian permasalahan.

- Langkah pertama yang harus dilakukan dalam menulis program adalah memikirkan dan mendefinisikan secara cermat permasalahan yang ingin diselesaikan menggunakan program komputer.
- Dengan kata lain apakah sesungguhnya yang ingin dikerjakan oleh sebuah program.
- Jika anda telah melakukan identifikasi permasalahan dan mendefinisikan permasalahan dengan jelas dan benar maka ini merupakan langkah awal yang sangat baik dalam menulis apa yang diinginkan dalam pembuatan program.

Teknologi dan Rekayasa

## Langkah-Langkah Pengembangan Program

### 1. Pendefinisian permasalahan.

- Mari kita lihat satu contoh permasalahan berikut ini:

"Menyeberang di jalan yang sangat ramai"

Teknologi dan Rekayasa

## Langkah-Langkah Pengembangan Program

"Menyeberang di jalan yang sangat ramai"

1. Pendefinisian permasalahan.

**Algoritma:**

- Untuk memecahkan permasalahan bagaimana menyeberang di jalan yang sangat ramai diperlukan langkah-langkah atau sekuen atau formula kerja.
- Formula kerja yang digunakan untuk memecahkan masalah pemrograman disebut **Algoritma program**.
- Seorang programmer harus menggunakan daftar urutan pekerjaan.

Teknologi dan Rekayasa

## Langkah-Langkah Pengembangan Program

"Menyeberang di jalan yang sangat ramai"

**Algoritma:**  
 Dalam kasus permasalahan menyeberang jalan step perintah-perintah sederhana dapat dinyatakan seperti Gambar

STEP	PERINTAH
1	Start
2	Berjalajah ke sudut jalan dan berhenti
3	Lihat dan cermati lampu pengatur lalu lintas
4	Apakah lampu penyeberangan pada arah anda menyala merah
5	Jika "ya", kembali ke Step 3 (Untuk keadaan lain teruskan ke Step 6)
6	Lihat ke arah kiri
7	Apakah masih ada kendaraan yang lewat
8	Jika "ya", kembali ke Step 6
9	Lihat ke arah kanan
10	Apakah masih ada kendaraan yang lewat
11	Jika "ya", kembali ke Step 9
12	Menyeberanglah dengan hati-hati
13	Stop

Teknologi dan Rekayasa

## Langkah-Langkah Pengembangan Program

"Menyeberang di jalan yang sangat ramai"

1. Pendefinisian permasalahan.

**Algoritma:**

- Perintah dalam program mengalir dari step awal (step 1) yaitu start ke step 2, step 3 dan seterusnya sampai dengan terminal stop. Kecuali ada perintah untuk melompat atau memanggil subrutin atau jika ada interupsi program akan mengalir terus kelangkah-langkah berikutnya. Pada step 3, step 4, dan step 5 terjadi proses pendeteksian dan pengambilan keputusan melangkah ke langkah 6 yaitu untuk memperhatikan apakah ada kendaraan yang lewat dari arah sebelah kiri.

Teknologi dan Rekayasa

## Langkah-Langkah Pengembangan Program

"Menyeberang di jalan yang sangat ramai"

1. Pendefinisian permasalahan.

**Algoritma:**

- Pada step 6, step 7, dan step 8 juga terjadi proses pendeteksian dan pengambilan keputusan apakah melangkah step 9 untuk melihat ke arah kanan atau masih tetap melihat arah kiri pada step 6. Demikian juga pada step 9, step 10, dan step 11 juga terjadi proses pendeteksian dan pengambilan keputusan apakah melangkah step 12 untuk melangkah menyeberang atau masih tetap melihat arah kanan pada step 9.

Teknologi dan Rekayasa

## Langkah-Langkah Pengembangan Program

"Menyeberang di jalan yang sangat ramai"

1. Pendefinisian permasalahan.

**Algoritma:**

- Ketigabelas langkah ini adalah bahasa bayi atau bahasa aras rendah atau bahasa level rendah, yang pada kenyataannya dilakukan langkah-langkah inilah yang dilakukan pada setiap menyeberang jalan di jalan yang sibuk dan ada lampu pengatur lalu lintas. Tidak akan pernah terjadi melihat lampu penyeberangan, melihat arah kiri, dan melihat arah kanan dilakukan secara serentak. Memang nampaknya seakan-akan demikian. Namun sesungguhnya tidaklah pernah demikian karena proses yang terjadi pasti merupakan sekuen langkah demi langkah. Ketigabelas sekuen step perintah ini disebut juga dengan Algoritma Program.

Teknologi dan Rekayasa

## Langkah-Langkah Pengembangan Program

"Menyeberang di jalan yang sangat ramai"

1. Pendefinisian permasalahan.

**Algoritma:**

- Semakin detail algoritma suatu program semakin baik karena program menghendaki kesuksesan 100% sesuai kebutuhan tuntutan permasalahan.
- Bisa saja step 5, step 6 dan step 7 tidak perlu ada manakala jalur lalu lintas hanya satu arah.
- Sangat berbahaya jika step 2, step 3, dan step 4 diabaikan. Jika ingin menyeberang jalan dan algoritma diatas digunakan dapat dipastikan penyeberang akan sukses dan selamat

Teknologi dan Rekayasa

## Langkah-Langkah Pengembangan Program

**"Menyeberang di jalan yang sangat ramai"**

1. Pendefinisian permasalahan.

**Algoritma:**

- ❖ Kasus menyeberang di jalan yang sangat ramai dengan 13 step penyelesaian merupakan bentuk bahasa aras rendah.
- ❖ Mikroprosesor di dalam sistem bekerja dengan bahasa aras rendah.
- ❖ Agar anda sukses dan mudah belajar memprogram mikroprosesor maka anda harus memiliki kesiapan merubah kebiasaan berfikir pada aras tinggi kekebiasaan berfikir di aras rendah. Inilah kendala utama dalam belajar memprogram mikroprosesor.

Teknologi dan Rekayasa

## Langkah-Langkah Pengembangan Program

1. Representasi kerja program.

**Flowchart**

- ❖ Flowchart atau diagram alir digunakan untuk menunjukkan aliran proses sebuah program.
- ❖ Untuk menyajikan jenis operasi sebuah program digunakan bentuk-bentuk grafis standar.
- ❖ Ada dua puluh delapan jenis bentuk grafis yang digunakan untuk menyusun flowchart seperti pada Gambar.

Teknologi dan Rekayasa

## Langkah-Langkah Pengembangan Program

**Flowchart**

Teknologi dan Rekayasa

## Langkah-Langkah Pengembangan Program

**Flowchart**

- Bentuk-bentuk grafis penyusun flowchart dapat dilihat pada AutoShapes Flowchart program Microsoft Word.
- Pemilihan bentuk-bentuk grafis flowchart pada Gambar tidak boleh sembarangan atau asal pilih apalagi membuat sendiri bentuk-bentuk lain secara bebas.
- Hal ini tidak dibenarkan karena semua bentuk grafis flowchart telah disepakati dan distandardkan secara internasional sebagai alat komunikasi.

Teknologi dan Rekayasa

## Langkah-Langkah Pengembangan Program

**Flowchart**

- ❖ Flowchart kasus menyeberang di jalan yang sangat ramai ditunjukkan seperti Gambar disamping.

Teknologi dan Rekayasa

## Langkah-Langkah Pengembangan Program

**Flowchart**

- ❖ Flowchart menyeberang di jalan ramai diawali dengan perintah START sebagai tanda mulai.
- ❖ Perintah start sebagai tanda mulai menggunakan bentuk grafis terminal.
- ❖ Selanjutnya diteruskan dengan dua proses "berhenti di sudut jalan" dan "melihat lampu pengatur lalu lintas"

Teknologi dan Rekayasa

Langkah-Langkah Pengembangan Program 

**SELESAI**  
**TERIMAKASIH**

Teknologi dan Rekayasa