

**RANCANG BANGUN PERANGKAT *PROGRAMMABLE*  
DENGAN MEDIA PEMROGRAMAN MENGGUNAKAN KATA  
DALAM BAHASA INDONESIA SEBAGAI MODUL  
PEMBELAJARAN DASAR PEMROGRAMAN**



**MAS KAREBET SETIAWAN  
5215127149**

**Skripsi Ini ditulis Untuk Memenuhi Sebagian Persyaratan dalam  
Memperoleh Gelar Sarjana Pendidikan**

**PROGRAM STUDI PENDIDIKAN TEKNIK ELEKTRONIKA  
FAKULTAS TEKNIK UNIVERSITAS NEGERI JAKARTA  
2017**

## HALAMAN PENGESAHAN

NAMA DOSEN	TANDA TANGAN	TANGGAL
<u>Dr. Muhammad Yusro, MT</u> (Dosen Pembimbing 1)		15/8-2017
<u>Drs. Pitoyo Yuliatmojo, MT</u> (Dosen Pembimbing 2)		15/8-2017

## PENGESAHAN PANITIA UJIAN SKRIPSI

NAMA DOSEN	TANDA TANGAN	TANGGAL
<u>Dr. Moch Sukardjo, M.Pd</u> (Ketua Penguji)		19/8-2017
<u>Dr. Baso Maruddani, MT</u> (Sekretaris)		15/8-17
<u>Syufrijal, MT</u> (Dosen Penguji Ahli)		11/8-2017

Tanggal Lulus : 8/8-2017

## HALAMAN PERNYATAAN

Dengan ini saya menyatakan bahwa:

1. Karya tulis skripsi saya ini adalah asli dan belum pernah diajukan untuk mendapatkan gelar akademik sarjana, baik di Universitas Negeri Jakarta maupun di perguruan tinggi negeri lain
2. Karya tulis ini adalah murni gagasan, rumusan dan penelitian saya sendiri dengan arahan dosen pembimbing
3. Dalam karya tulis ini tidak terdapat karya atau pendapat yang telah ditulis atau dipublikasikan orang lain, kecuali secara tertulis dengan jelas dicantumkan sebagai acuan dalam naskah dengan disebutkan nama pengarang dan dicantumkan dalam daftar pustaka
4. Pernyataan ini saya buat dengan sesungguhnya dan apabila di kemudian hari terdapat penyimpangan dan ketidak benaran dalam pernyataan ini, maka saya bersedia menerima sanksi akademik berupa pencabutan gelar yang telah diperoleh karena karya tulis ini, serta sanksi lainnya sesuai dengan norma yang berlaku di Universitas Negeri Jakarta.

Jakarta, Agustus 2017  
Yang membuat pernyataan



Mas Karebet Setiawan  
5215127149

## KATA PENGANTAR

Puji syukur penulis sampaikan kepada Allah Subhanahu Wa Ta'ala atas segala karunia dan rahmat-Nya sehingga skripsi ini dapat terselesaikan. Penulisan skripsi ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar Sarjana Pendidikan Teknik Elektronika Fakultas Teknik Universitas Negeri Jakarta. Saya menyadari bahwa tanpa bantuan dan bimbingan dari berbagai pihak, sangatlah sulit bagi saya untuk menyusun skripsi ini. Oleh karena itu, saya mengucapkan terima kasih kepada :

- 1) Drs. Pitoyo Yuliatmojo, M.T selaku Koordinator Program Studi Pendidikan Teknik Elektronika dan Pembimbing II.
- 2) Dr. Muhammad Yusro, MT selaku Pembimbing I.
- 3) Ibu dan Ayah yang telah menyayangi dan mendidik saya.

Akhir kata, semoga Allah Subhana Wa Ta'ala membalas segala kebaikan semua pihak yang telah membantu penyusunan skripsi ini dengan balasan yang lebih baik. Semoga skripsi ini membawa manfaat yang besar bagi pengembangan ilmu pengetahuan dan teknologi.

Jakarta, Agustus 2017

Penulis

## ABSTRAK

**MAS KAREBET SETIAWAN.** *Rancang Bangun Perangkat Programmable Dengan Media Pemrograman Menggunakan Kata Dalam Bahasa Indonesia Sebagai Modul Pembelajaran Dasar Pemrograman.* Skripsi. Jakarta, Program Studi Pendidikan Teknik Elektronika, Fakultas Teknik, Universitas Negeri Jakarta, 2017. Dosen Pembimbing: Dr. Muhamad Yusro, MT. dan Drs. Pitoyo Yuliatmojo, MT.

Tujuan penelitian ini adalah merancang dan membuat perangkat *programmable* yang dapat diprogram dengan menggunakan kata dalam bahasa Indonesia. Manfaat dari penelitian ini adalah menyediakan perangkat *programmable* yang memudahkan dalam pembelajaran.

Penelitian ini menggunakan metode *The Linier Sequence Model* yang meliputi menganalisis, mendesain, kode dan tes. Kebutuhan sistem yang digunakan dalam penelitian ini adalah: mikrokontroler dan compailer.

Hasil uji dari penelitian ini mendapatkan penilaian skor 95% oleh ahli media dengan kategori “Sangat Layak” untuk diuji coba dan hasil uji pengguna mendapatkan penilaian skor 84% dengan kategori “Layak” digunakan sebagai media penunjang pembelajaran.

**Kata kunci :** Perangkat, Pemrograman, *Programmable*, Media Pembelajaran, Bahasa Indonesia, kata

## **ABSTRACT**

**MAS KAREBET SETIAWAN.** *Building Programmable Device with Media Programming Using Words in Indonesian as a Basic Learning Module Programming.* Thesis. Jakarta: Electrical Engineering Education Department, Faculty of Engineering, State University of Jakarta, 2017. Advisor: Dr. Muhamad Yusro, MT. And Drs. Pitoyo Yuliatmojo, MT.

The purpose of this research is to create programmable device that can be programmed by using word in Indonesian. This research aims to provide a programmable tool that facilitates learning.

This research uses *The Linear Sequence Model* method which includes Analyze, design, code and test. System requirement used in this research is: microcontroller and compiler.

The results of this research is obtained the percentage score of 95% by media expert with the category "Very Worth" for trial and the result of the users test is obtained the percentage score of 84% with the category "Worth" is used as a support learning.

**Keywords:** Tool, Programming, Programmable, Learning Media, Indonesia, Word

## DAFTAR ISI

<b>HALAMAN JUDUL</b> .....	<b>i</b>
<b>HALAMAN PENGESAHAN</b> .....	<b>ii</b>
<b>HALAMAN PENYATAAN</b> .....	<b>iii</b>
<b>KATA PENGANTAR</b> .....	<b>iv</b>
<b>ABSTRAK</b> .....	<b>v</b>
<b>ABSTRACT</b> .....	<b>vi</b>
<b>DAFTAR ISI</b> .....	<b>vii</b>
<b>DAFTAR TABEL</b> .....	<b>ix</b>
<b>DAFTAR GAMBAR</b> .....	<b>x</b>
<b>DAFTAR LAMPIRAN</b> .....	<b>xi</b>
<b>BAB I PENDAHULUAN</b> .....	<b>1</b>
1.1. Latar Belakang .....	1
1.2. Identifikasi Masalah .....	3
1.3. Batasan Masalah.....	4
1.4. Rumusan Masalah .....	4
1.5. Tujuan Penelitian.....	4
1.6. Manfaat Penelitian.....	4
<b>BAB II TINJAUAN PUSTAKA</b> .....	<b>5</b>
2.1. Kerangka Teori.....	5
2.1.1. Media Pembelajaran.....	5
2.1.2. Pengembangan Perangkat Lunak .....	5
2.1.3. Teori Konstuktivisme Piaget.....	7
2.1.3.1.Pemikiran Operasiona Formal.....	8
2.1.4. Sistem Koordinat.....	9
2.1.5. Perangkat <i>programmable</i> .....	11
2.1.6. Mikrokontroler .....	13
2.1.7. Media Pemrograman .....	14
2.1.8. Bahasa pemrograman C .....	17
2.1.9. BootloaderHID.....	18
2.2. Kerangka Berpikir .....	19
2.2.1. Penelitian Terkait .....	21
<b>BAB III METODOLOGI PENELITIAN</b> .....	<b>22</b>
3.1. Tempat dan Waktu Penelitian .....	22
3.2. Metode penelitian .....	22
3.2.1. Tujuan pengembangan .....	22
3.2.3. Sasaran pengguna.....	22
3.2.2. Motode .....	22
3.3. Prosedur Penelitian.....	23
3.3.1. Teknik Menganalisis .....	23
3.3.2. Teknik Desain .....	24
3.3.3. Teknik Kode.....	25
3.3.4. Teknik Tes.....	25
3.4. Teknik Pengumpulan Data .....	26
3.2.4. Instrumen .....	26
3.2.4.1.Validitas Instrumen .....	30
3.2.4.2.Reliabilitas Instrumen .....	31
4.5. Teknik Analisis Data Penelitian .....	32

<b>BAB IV HASIL PENELITIAN</b> .....	35
4.1. Hasil Menganalisis Kebutuhan.....	35
4.2. Hasil Proses Desain .....	35
4.2.1. Desain Hardware .....	36
4.2.2. Desain elektrikl.....	37
4.2.4. Hasil Implementasi Disain Hardware .....	38
4.2.4. Desain Software .....	39
4.2.4. Hasil Implementasi Disain Software.....	44
4.3. Hasil Kode .....	45
4.3.1. Kode Aplikasi Media .....	45
4.3.2. Kode Firmware Perangkat.....	47
4.4. Hasil Test.....	48
4.4.1. White Box Testing .....	48
4.4.2. Black Box Testing.....	50
4.4.3. Alpa Testing .....	51
4.4.4. Beta Testing .....	51
4.5. Pembahasan .....	52
4.6. Aplikasi Hasil Penelitian .....	53
<b>BAB V KESIMPULAN DAN SARAN</b> .....	54
5.1. Kesimpulan.....	54
5.2. Keterbatasan Produk.....	54
5.3. Saran .....	55
<b>DAFTAR PUSTAKA</b> .....	<b>56</b>
<b>LAMPIRAN</b>	
<b>RIWAYAT HIDUP</b>	

## DAFTAR TABEL

Tabel 3.1. Langkah-Langkah Penelitian <i>The Linier Sequential Model</i> .....	23
Tabel 3.2. Kisi-kisi Aspek Yang Ingin Dicapai .....	27
Tabel 3.3. Kisi-kisi Pengujian.....	27
Tabel 3.4. Pernyataan Untuk Ahli Media .....	28
Tabel 3.5. Pernyataan Untuk Responden .....	29
Tabel 3.6. Validitas Pengguna Ahli .....	30
Tabel 3.7. Validitas kisi-kisi Pengguna .....	31
Tabel 3.8. Kategori Koefisien Reliabilitas.....	32
Tabel 3.9. Ketentuan Pemberian Skor .....	32
Tabel 3.10. Pedoman Konversi Skor .....	33
Tabel 3.11. Konversi Persentase Skor Menjadi Kategori Kualitatif.....	34
Tabel 4.1. Pengukuran nilai error delay .....	46
Tabel 4.2. Pengukuran Tegangan Output .....	46
Tabel 4.3. Pengukuran Arus Output .....	46
Tabel 4.4. Perhitungan daya Output.....	46
Tabel 4.5. Hasil Uji Black Box Testing .....	47
Tabel 4.6. Hasil Uji alpa testing.....	48
Tabel 4.7. Hasil Uji .....	49
Tabel 4.8. Hasil Uji beta testing.....	49

## DAFTAR GAMBAR

Gambar 2.1. Diagram The Linier Sequential Model .....	6
Gambar 2.2. Perkembangan kognitif menurut Jean Piaget .....	7
Gambar 2.3. Kuadran-kuadran pada Sistem Koordinat .....	10
Gambar 2.4. Titik pada Sistem Koordinat .....	10
Gambar 2.5. Simbol FGMOSFET .....	12
Gambar 2.6. Rangkaian dalam Sel EEPROM .....	12
Gambar 2.7. Karakteristik dan Geometri FGMOSFET .....	13
Gambar 2.8. Diagram Minimum Sistem Mikroprosesor .....	13
Gambar 2.5. Chip mikrokontroler .....	14
Gambar 2.9. OP kode dan Menemonik .....	15
Gambar 2.10. Urutan Teknik Kompilasi .....	15
Gambar 2.11. Analisa Leksikal .....	16
Gambar 2.12. Sintaks dalam bahasa BCPL .....	17
Gambar 2.13. Struktur bahasa C .....	18
Gambar 2.14. Diagram Alir BootloadHID .....	18
Gambar 2.15. Rangkaian Dasar Bootloadhid .....	19
Gambar 2.16. Diagram Blok Sistem .....	20
Gambar 2.17. Koordinat Kuadran I .....	20
Gambar 3.1. Diagram Alir Penelitian .....	23
Gambar 4.1. Diagram Analisis Kebutuhan .....	35
Gambar 4.2. Desain Perangkat .....	36
Gambar 4.3. Desain Antarmuka Pin .....	37
Gambar 4.4. Sekematik Perangkat .....	37
Gambar 4.5. Perangkat .....	38
Gambar 4.6. Rangkaian Perangkat .....	38
Gambar 4.7. Desain Sistem Pada Perangkat .....	39
Gambar 4.8. Desain struktur .....	40
Gambar 4.9. Desain Antarmuka .....	41
Gambar 4.10. Shortcut dan Halaman Media .....	43
Gambar 4.11. Lembar Kerja dan Notifikasi Software .....	43

## **DAFTAR LAMPIRAN**

**LAMPIRAN 1. Pengujian**

**LAMPIRAN 2. Program**

**LAMPIRAN 3. Buku Panduan**

# BAB I

## PENDAHULUAN

### 1.1. Latar Belakang

Perangkat *programmable* sudah digunakan secara luas di kalangan hobi, masyarakat maupun industri sebagai salah satu teknik kontrol otomatis untuk penggerak maupun sebagai kontrol instrumentasi input. Perangkat *programmable* adalah perangkat yang praktis digunakan karena dapat diprogram berulang-ulang tanpa harus mengganti atau membongkar komponen didalamnya sehingga membuat perangkat ini memudahkan penggunanya dalam merancang sebuah alat yang membutuhkan teknik kontrol otomatis. Contoh perangkat *programmable* yang ada dalam industri adalah *Programmable Logic Control* (PLC) adalah standar yang digunakan yang merupakan paket perangkat dan media pemrogramannya yang terkenal dengan pemrograman diagram *leader*. Dalam perangkatnya berbasis mikrokontroler yang pada output dan inputnya sudah disambungkan dengan rangkaian relay sehingga bisa menangani kontrol arus besar (Putri & Rifai, 2013). Dalam kehidupan masyarakat perangkat *programmable* sudah digunakan secara luas yaitu smartphone dan Komputer adalah perangkat *programmable* yang berbasis mikroprosesor didalamnya. Masyarakat menggunakannya untuk media komunikasi dan sebagai alat bantu untuk menyelesaikan suatu masalah, dikatakan perangkat *programmable* karena pengguna dapat memasang berbagai program aplikasi sehingga perangkat dapat berubah fungsinya tanpa harus membongkar komponen didalamnya.

*Programmable Logic Device* (PLD) adalah perangkat *programmable* yang jarang dikenal namun penggunaannya luas dalam sebuah sistem meskipun penggunanya

tidak pernah megetahui. PLD adalah perangkat *programmable* sirkuit logika dimana dalam penggunaanya memudahkan untuk mendisain sebuah fungsi dengan gerbang logika. Contoh perangkat *programmable* yang terkenal adalah *Arduino*, *Raspberry Pi*, *NXT Lego*, *ST Link* adalah perangkat *programmable* mikrokontroler yang dalam pemasarannya sudah disediakan media pemrogramannya. Namun dari semua yang dicontohkan tidak ada yang media pemrogramannya menggunakan bahasa Indonesia sehingga membuat kesulitan untuk mempelajari perangkat tersebut oleh karena itu penulis akan membuat perangkat progamable dan media pemrograman yang menggunakan bahasa Indonesia agar memudahkan untuk mempelajari perangkat tersebut.

Sarana dan prasarana merupakan salah satu media pendidikan yang paling berpengaruh dalam proses pembelajaran. Seperti pada Peraturan Pemerintah Republik Indonesia No. 19 Tahun 2005 Tentang Standar Nasional Pendidikan, pada BAB VII (Sarana dan Prasarana), Pasal 42 butir 1 yang berbunyi: Setiap satuan pendidikan wajib memiliki sarana yang meliputi perabot, peralatan pendidikan, media pendidikan, buku dan sumber belajar lainnya, bahan habis pakai, serta perlengkapan lain yang diperlukan untuk menunjang proses pembelajaran yang teratur dan berkelanjutan. Oleh karena itu Media pembelajaran adalah salah satu sarana dan prasarana yang harus ada untuk menunjang proses pembelajaran. Selain itu guru juga dituntut untuk bisa memaksimalkan media pembelajaran tersebut guna untuk meningkatkan kompetensi keahlian siswa (Bagus,2015).

Pembelajaran pada tahap operasional formal (11 tahun ke atas) adalah pembelajaran yang menunjang pada perkembangan kognitif individu yang secara

psikologi sudah mampu menggunakan abstraksi-abstraksi, membedakan yang konkrit dengan yang abstrak dan munculnya kemampuan nalar secara ilmiah. Menurut Piaget dalam (Papalia, 2008:555) Anak masa operasional formal memasuki level tertinggi perkembangan kognitif, yaitu ketika mengembangkan kemampuan berpikir abstrak. Anak yang ada pada masa operasional formal memiliki kemampuan dapat menggunakan simbol untuk menyimbolkan (misalnya, menjadikan huruf x sebagai angka yang tidak diketahui) dan karena itu dapat belajar aljabar dan kalkulus.

Media pembelajaran adalah salah satu sarana dan prasarana yang harus ada untuk menunjang proses pembelajaran. Selain itu pengajar juga dituntut untuk bisa mengembangkan media pembelajaran guna untuk meningkatkan kompetensi keahlian siswa berdasarkan tugas psikologi perkembangan siswa sebagai individu yang pada masa perkembangan sebelumnya sudah mempelajari matematika secara konkret dan abstrak.

## **1.2. Identifikasi Masalah**

Berdasarkan latar belakang yang dijelaskan maka dapat diidentifikasi masalah sebagai berikut :

1. Bagaimana menyediakan Produk media pembelajaran yang menunjang tugas psikologi perkembangan pada masa remaja.
2. Belum tersedia perangkat yang menggunakan bahasa indonesia sebagai media pemrograman dasar yang bersifat hanya mengantarkan ke ilmu pengetahuan selanjutnya.
3. Belum tersedianya perangkat yang megimplementasikan ilmu abstrak kepada pelaksanaan teknis yang konkret.

### **1.3. Batasan Masalah**

Agar penelitian menjadi spesifik, dibatasi masalah yakni bagaimana menyediakan media pembelajaran dalam bentuk perangkat dengan permasalahan:

1. Bagaimana merancang perangkat agar dapat memrakatkan perangkat pemrograman secara langsung dengan kata dalam bahasa indonesia?
2. Bagaimana merancang perangkat *programmable* dengan megimplementasikan ilmu matematika yang merupakan ilmu dasar?

### **1.4. Rumusan Masalah**

Berdasarkan pembatasan masalah di atas, maka rumusan masalahnya adalah bagaimana merancang bangun perangkat *programmable* dengan media pemrograman menggunakan kata dalam bahasa indonesia sebagai modul pembelajaran dasar pemrograman.

### **1.5. Tujuan Penelitian**

Tujuan penelitia ini adalah untuk merancang dan membuat perangkat *programmable* yang dapat diprogram langsung dengan media pemrogramannya yang menggunakan kata dalam bahasa Indonesia sebagai produk media pembelajaran.

### **1.6. Manfaat Penelitian**

Manfaat dari penelitian ini adalah menyediakan perangkat *programmable* dan media pemrogramannya yang memiliki kelebihan murah dalam pembuatannya dan mudah dalam penggunaannya yang nantinya diharapkan bisa digunakan sebagai media pembelajaran.

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1. Kerangka Teori**

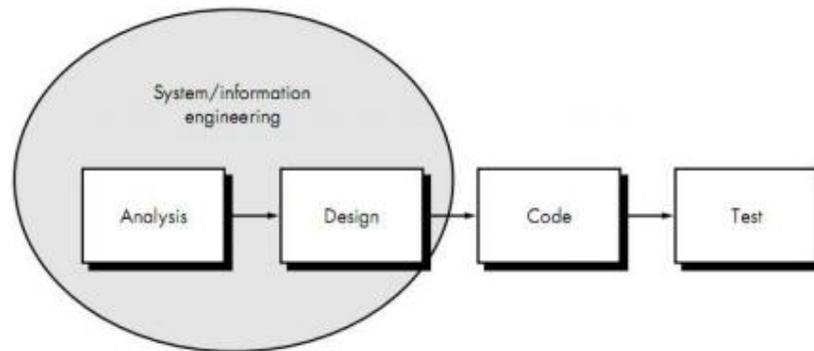
##### **2.1.1. Media Pembelajaran**

Media adalah bentuk kata jamak dari medium, yang dapat diartikan sebagai perantara atau pengantar. Media adalah salah satu komponen penting dari proses komunikasi. Media pembelajaran adalah teknologi pembawa informasi yang dapat dimanfaatkan untuk keperluan pembelajaran. Menurut Arief S. Sadiman (1986:7) media pembelajaran adalah segala sesuatu yang dapat digunakan untuk menyalurkan pesan dari pengirim ke penerima, sehingga dapat merangsang pikiran, perasaan, perhatian dan minat siswa sehingga proses belajar terjadi. Berdasarkan pengertiannya media pembelajaran yang telah dikemukakan diatas dapat disimpulkan bahwa, media pembelajaran adalah sarana untuk membantu pendidik dalam menyampaikan informasi berupa materi pembelajaran kepada siswa. Media pembelajaran memiliki manfaat secara umum untuk mempermudah proses belajar mengajar antara siswa dan pendidik agar dapat berjalan secara maksimal.

##### **2.1.2. Pengembangan Perangkat Lunak**

Menurut Pressman (2001:6) pengertian perangkat lunak adalah (1) kumpulan instruksi program komputer yang mana ketika dikerjakan akan menyediakan fungsi dan daya guna yang diinginkan, (2) kumpulan struktur data yang memungkinkan program untuk memanipulasi informasi program secukupnya, (3) kumpulan dokumen yang menggambarkan operasi dan penggunaan program. Menurut Pressman (2001: 28-30) model pengembangan

perangkat lunak yang sering digunakan adalah model *The Linier Sequential Model* atau sering disebut *Waterfall Model*. Model pengembangan ini memiliki 4 tahapan seperti dalam gambar 2.1 berikut (Pressman, 2001: 29).



**Gambar 2.1. Diagram The Linier Sequential Model**

Proses *Analysis* merupakan proses pengumpulan kebutuhan dalam perangkat lunak. Seorang pengembang harus mengetahui karakteristik dari program yang dibuat dengan memahami kebutuhan informasi, fungsi, unjuk kerja dan antarmuka yang diperlukan. Proses *Design* merupakan proses multistep yang terfokus pada empat atribut meliputi struktur data, arsitektur perangkat lunak, representasi antarmuka, dan algoritma. Proses *Code* merupakan proses perubahan desain yang telah dibuat menjadi bahasa yang dapat dipahami oleh mesin. Jika desain dibuat dengan rinci, maka proses pengkodean dapat diselesaikan secara mekanis. Setelah program selesai dibuat, maka dilanjutkan pada proses *Test* (pengujian). Proses pengujian terfokus pada logika internal perangkat lunak untuk memastikan bahwa semua pernyataan dalam program sudah diuji, dan pengujian fungsional untuk menemukan kesalahan-kesalahan dan memastikan bahwa masukan yang telah ditentukan akan menghasilkan keluaran actual yang sama dengan keluaran yang diharapkan. Pengujian perangkat lunak menurut Pressman (2001: 479)

meupakan elemen dalam pengembangan perangkat lunak yang sering disebut dengan *Verification and Validation* (V&V). Verifikasi dilakukan untuk memastikan perangkat lunak dapat berfungsi seperti yang telah ditentukan. Validasi dilakukan untuk memastikan perangkat lunak sudah sesuai dengan kebutuhan pengguna.

### 2.1.3. Teori Konstuktivisme Piaget

Menurut piaget dalam (Suparno,2001:5) pengetahuan itu dibentuk oleh murid atau orang yang sedang belajar. Pengetahuan tidak diterima begitu saja dari guru, tetapi murid sendirilah yang harus mengorganisasi, memikirkan, dan membentuk pengetahuan itu. Tanpa kegiatan aktif membentuk pengetahuan dalam pikirannya, seseorang tidak akan tahu sesuatu. Menurut piaget, seseorang itu mengalami perkembangan dari lahir sampai dewasa melewati tahap-tahap yang berkesinambungan gambar 2.2 menunjukkan ilustrasi perkembangan kognitif.



**Gambar 2.2. Perkembangan kognitif menurut Jean Piaget**

Secara garis besar, Piaget membedakan empat tahap dalam perkembangan kognitif seorang anak:

1. Tahap sensorimotor yang terjadi sejak anak lahir sampai umur 2 tahun.
2. Tahap praoperasi pada umur 2 sampai 7 tahun.
3. Tahap operasi konkret pada umur 7 sampai 11 tahun.
4. Tahap operasi formal pada umur 11 tahun ke atas.

Perkembangan tahap-tahap tersebut berurutan karena setiap tahap memerlukan tahap yang sebelumnya. Awal dan perkembangan tahap-tahap tersebut dapat berbeda untuk setiap pribadi. Teori perkembangan kognitif dan teori *konstruktivisme* piaget banyak mempengaruhi dunia pendidikan, terutama pendidikan kognitif pada masa anak-anak sampai remaja. Bagaimana menyusun kurikulum, bagaimana memilih bahan dan metode yang tepat untuk membantu murid belajar, bagaimana seorang guru membantu murid belajar.

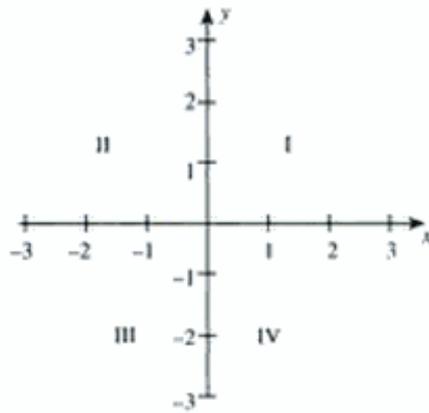
#### **2.1.3.1. Pemikiran Operasional Formal**

Tahap operasional formal adalah tahap keempat dan terakhir dari teori perkembangan kognitif piaget, yang diyakini muncul sekitar usia 11 sampai 15 tahun. Kekuatan berpikir yang sedang berkembang pada remaja membuka cakrawala pemikiran dan sosial yang baru. Pemikiran operasional formal bersifat lebih abstrak daripada pemikiran operasional konkret. Remaja tidak lagi terbatas pada pengalaman nyata dan konkret sebagai landasan berpikirnya. Mereka mampu membayangkan situasi rekaan, kejadian yang semata-mata berupa kemungkinan hipotesis ataupun proposisi abstrak, dan mencoba mengolahnya dengan pemikiran logis. Karena kemampuan berpikir abstraknya remaja seringkali bingung akan patokan ideal manakah yang akan dipegangnya (Santrock, 1996:188).

#### 2.1.4. Sistem Koordinat

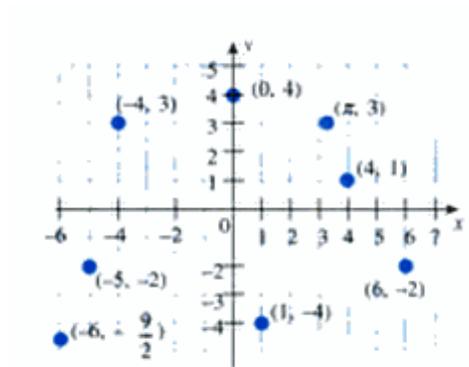
Menurut Purcell (2004:20) dua orang prancis memiliki jasa besar atas gagasan tentang system koordinat. Pierre de Fermat dan Rene Descartes, Pierre adalah pengacara yang mengemari matematika. Pada tahun 1629 Pierre menulis sebuah makalah yang pada dasarnya menggunakan sebuah koordinat untuk menggambarkan titik-titik dan kurva-kurva. Rene Descartes adalah ahli filsafat yang berfikir bahwa matematika dapat membuka kunci alam semesta. Descartes menerbitkan *La geometrie* pada tahun 1637 buku itu sangat terkenal walupun menekankan peranan aljabar dalam memecahkan masalah-masalah geometri, tetapi hanya sedikit membahas mengenai koordinat. Berdasarkan siapa yang pertamakali menjelaskan sistem koordinat secara gamblang fermat sepatasnya memperoleh pengakuan yang utama. Sejarah memang menjadi teman yang plin plan karena sekarang sistem koordinat dinamakan koordinat Cartesius menurut nama Rene Descartes.

Koordinat kartesius dalam suatu bidang, menggambarkan dua garis real; satu mendatar dan yang lainnya tegak sedemikian rupa sehingga keduanya berpotongan pada titik-titik nol dari kedua garis tersebut. Dua garis tersebut dinamakan sumbu-sumbu koordinat perpotonganya diberi label 0 dan disebut titik asal. Menurut perjanjian garis yang mendatar dinamakan sumbu-x dan garis yang tegak dinamakan sumbu-y. Setengah bagian dari sumbu-x adalah kekanan dan setengah bagian dari sumbu-y adalah ke atas sumbu-sumbu koordinat membagi empat bidang menjadi empat daerah disebut kuadran-kuadran, yang diberi label I, II, III dan IV seperti yang diperlihatkan dalam gambar 2.3.



**Gambar 2.3. Kuadrant-kuadrant pada Sistem Koordinat**

Setiap titik dalam bidang dapat dinyatakan dengan pasangan bilangan-bilangan yang disebut koordinat-koordinat cartesius-nya. Jika garis mendatar dan tegak melalui sebuah titik maka masing-masing akan memotong sumbu-x dan sumbu-y sehingga akan mempunyai koordinat  $(x,y)$  seperti pada gambar 2.4.



**Gambar 2.4. Titik pada Sistem Koordinat**

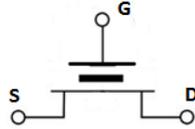
Konsep koordinat sangat penting dalam pembelajaran matematika karena merupakan dasar dari matematika turunan dan antiturunan yang merupakan matematika yang banyak digunakan dalam menyelesaikan permasalahan-permasalahan.

### 2.1.5. Perangkat programmable

Perangkat *programmable* adalah perangkat yang dapat berkerja jika perangkat tersebut sudah diberikan data program kedalamnya atau biasanya disebut perangkat lunak. Hal ini terjadi karena di dalam perangkatnya ada komponen yang memiliki fungsi memori sehingga dapat menyimpan data program yang selanjutnya data diproses sebagai operasi atau intruksi. Memori sendiri dibagi dua menurut fungsi dan cara kerjanya yaitu *Read Only Memory* (ROM) dan *Random Access Memory* (RAM). Dikatakan perangkat elektronik terprogram karena dalam sistem perangkatnya menggunakan ROM berjenis EEPROM untuk menyimpan data program utamanya yang berisi algoritma dalam penyelesaian suatu masalah dimana data program dikerjakan oleh perangkat sesuai alur algoritma sipemrogram.

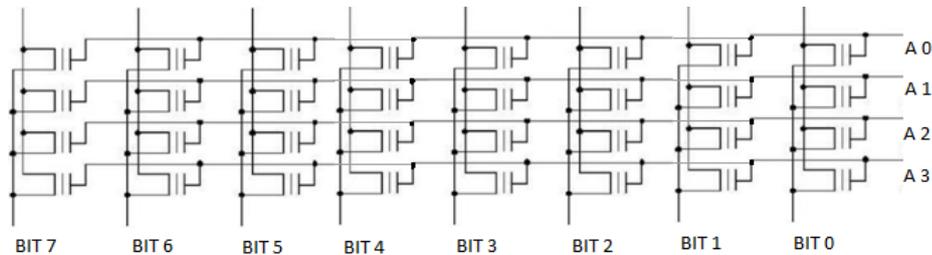
*Electrical erasable and programmable read only memory* (EEPROM) yaitu memori yang cara menulis dan menghapus datanya menggunakan implementasi ilmu kelistrikan murni dalam artian tidak menggunakan unsur mekanik bahkan mekatronik. Memori tidak berdiri sendiri sebagai sistem memori membutuhkan modul lain agar nilai kerjanya dapat bermanfaat luas. Memori yang dapat mempertahankan data tanpa membutuhkan arus listrik disebut juga *non volatile* atau dikenal dengan nama memori ROM. EEPROM atau yang terkenal dengan *flash memory* prinsip kerjanya adalah rangkaian elektronika digital yang memiliki logika dapat menyimpan data berupa delapan bit data informasi di satu alamat memori dimana setiap bit data disimpan dalam komponen elektronika *floating gate mosfet* (FGMOSFET) Gambar 2.5 adalah simbol komponen

FGMOSFET komponen ini tidak dijual dipasaran namun ada dalam setiap sel memori EEPROM.



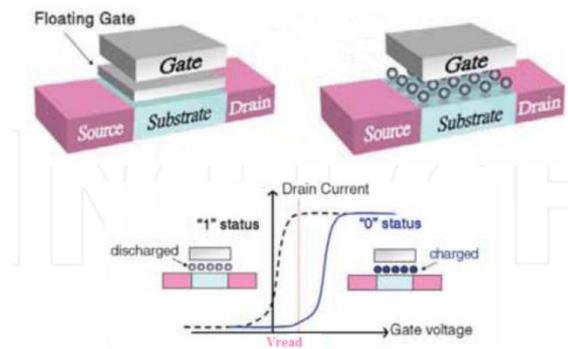
**Gambar 2.5. Simbol FGMOSFET**

Gambar 2.6 menunjukkan rangkaian dalam sel memori EEPROM dengan empat alamat dikali delapan bit, delapan bit disebut juga satu byte (Anonim, 2016).



**Gambar 2.6. Rangkaian dalam Sel EEPROM**

Prinsip kerja FGMOSFET adalah ketika pin G diberikan tegangan listrik pada nilai penulisan memori maka *floating gate* akan terisi elektron karena sifat mekanika kuantum sehingga mengakibatkan karakteristik MOSFET bergeser ke nilai yang lebih besar. Untuk menghapus data memori maka dilakukan pemberian tegangan listrik pada pin D sebesar tegangan listrik pada nilai penulisan memori maka karakteristik MOSFET kembali normal. Gambar 2.7 menunjukkan ilustrasi FGMOSFET ketika *floating gate* terisi elektron dan karakteristik tegangan pada G bergeser ke nilai yang lebih besar (Srivastava & Tyagi, 2016).

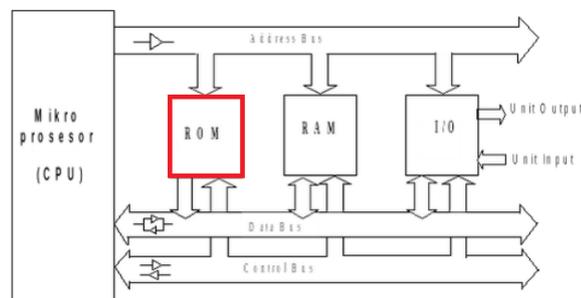


**Gambar 2.7. Karakteristik dan Geometri FG MOSFET**

Perangkat yang menggunakan komponen memori EEPROM adalah perangkat elektronik terprogram yang memiliki fitur *elektrikal eraseble* dan *elektrikal programmable*. Contoh penggunaan umumnya adalah sistem Mikrokontroler, sistem PLD, *Flash Disk* dan SSD (Anonim, 2016).

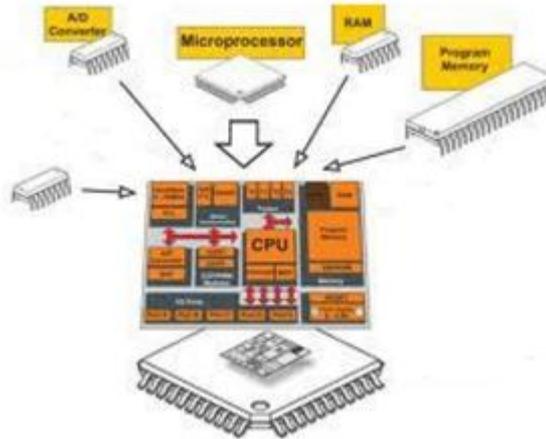
#### 2.1.6. Mikrokontroler

Mikrokontroler adalah suatu minimum sistem Mikroprosesor yang terangkai dalam satu chip sirkuit terintegrasi meliputi *Central Processing Unit* (CPU), ROM, RAM, dan Piranti input-output (I/O device) semuanya terhubung dalam sistem interkoneksi melalui jalur alamat, jalur data, dan jalur control. Gambar 2.8 menunjukkan diagram minimum sistem mikroprosesor (Wardoyo, 2011:32).



**Gambar 2.8. Diagram Minimum Sistem Mikroprosesor**

Ilustrasi sirkuit terintegrasi mikroprosesor didalam *chip* mikrokontroler ditunjukkan pada Gambar 2.5 (Wardoyo, 2011:33).



**Gambar 2.5. Chip mikrokontroler**

Mikroprosesor tidak berkerja sendiri komponen ini dapat mengolah data, melakukan perhitungan, dan mengendalikan instrumentasi input atau aktuator setelah diprogram itulah mengapa dinamakan sistem mikroprosesor.

### 2.1.7. Media Pemrograman

Pemrograman adalah penyusunan intruksi-intruksi pada perangkat terprogram untuk menyelesaikan suatu masalah teknis berdasarkan algoritma. Contoh media pemrograman adalah *Compiler* dan *Interpreter*. Pada prinsipnya pemrograman adalah menerjemahkan bahasa manusia ke bahasa mesin dalam kata lain megubah *assembly language* yang disimbolkan dengan *menemonic* ke kode berbentuk data biner yang disebut *operation code*. Hal ini dilakukan karena sangat mustahil untuk menghafal intruksi langsung dengan kode biner. Pemrograman dengan bahasa *assembly* sering disebut bahasa tigkat rendah karena sangat sulitnya dipahami oleh manusia. Gambar 2.9 menunjukkan contoh *operation code* yang disusun mengikuti menemoniknya (Prakoso, 2016).

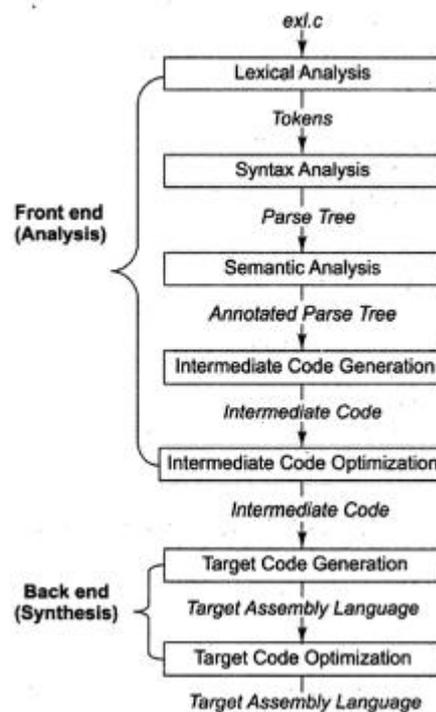
```

26
27
28 0000003E B8C007      bootloader_start:
29 00000041 052002      mov ax, 07C0h
30 00000044 FA                add ax, 544
31 00000045 8ED0                cli
32 00000047 BC0010      mov ss, ax
33 0000004A FB                mov sp, 4096
34
35 0000004B B8C007      sti
36 0000004E 8ED8                mov ax, 07C0h
37
38 00000050 80FA00      mov ds, ax
39 00000053 741D                cmp dl, 0
40 00000055 8816[EE01]      je no_change
41 00000059 B408                mov [bootdev],
                                mov ah, 8

```

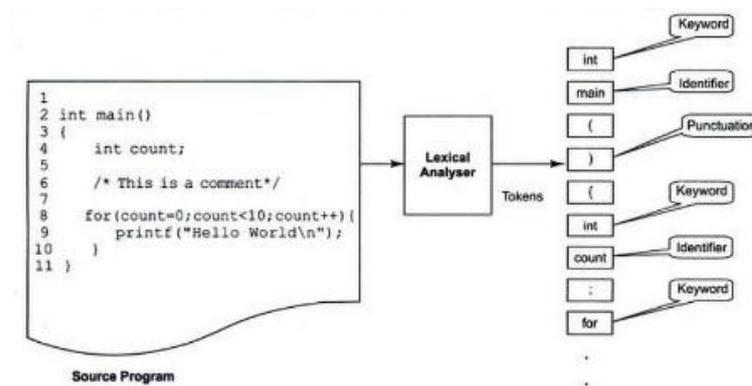
**Gambar 2.9. OP kode dan Menemonik**

Dengan kemajuan ilmu pengetahuan dan keahlian rekayasa sistem disusunlah intruksi-intruksi yang sering digunakan lalu dibuatlah sebuah fungsi intruksi yang memudahkan dalam pembuatan program inilah ide awal bahasa tingkat tinggi. Implementasi prinsip kerja kompiler adalah urutan proses manipulasi string seperti yang ditunjukkan pada gambar 2.10 (Raghavan, 2010).



**Gambar 2.10. Urutan Teknik Kompilasi**

Analisa Leksikal adalah kompailer melakukan pembacaan kode program karakter demi karakter lalu dikelompokkan menjadi satu kesatuan atau disebut token yang ditentukan kompailer yang terdiri dari *keyword*, *identifier*, *punctuation* seperti yang ditunjukkan pada gambar 2.11 (Raghavan, 2010).



**Gambar 2.11. Analisa Leksikal**

Analisa Sintaks adalah memeriksa kesesuaian pola deretan token dengan aturan sintaks yang ditentukan kompailer. Deretan token yang bersesuaian dengan sintaks tertentu akan dinyatakan sebagai pohon parsing. Analisa Semantik memeriksa token dan ekspresi pada pohon parsing lalu memberikan batasan yang ditetapkan oleh kompailer berdasarkan target mikroprosesor. Seperti memasukan konstanta ke dalam *register variable* dengan tipe data yang digunakan. Pembangkit Kode Langsung membangkitkan kode berdasarkan pohon parsing yang sudah di tentukan nilai batasan nya. Pohon parse selanjutnya diterjemahkan menjadi perintah menemonik yang merupakan representasi program untuk suatu mesin abstrak. Pembangkit Kode Mesin menerjemahkan hasil dari pembangkit kode langsung ke menemonik suatu mesin target (Slamet, S & Suhartanto, 1993:6).

### 2.1.8. Bahasa pemrograman C

Bahasa C adalah pengembangan dari bahasa BCPL yang dikembangkan oleh Martin Richards pada tahun 1967. Bahasa ini memberikan ide kepada Ken Thompson yang kemudian dikembangkan mejadi bahasa B lalu dikembangkan Dennis Ricthie pada tahun 1970 menjadi bahasa C di Bell Telephone Laboratories inc. Bahasa C pertama kali digunakan di komputer digital Equipment Corporation PDP-11 yang meggunakan sistem operaasi UNIX (Jogiyanto, 1993:1). BCPL (*Basic Combined Programming Language*) adalah bahasa pemrograman prosedural, imperatif, dan terstruktur yang dirancang oleh Martin Richards dari Universitas Cambridge pada tahun 1966. BCPL sudah tidak lagi digunakan secara umum. Namun bahasa BCPL masih berpengaruhnya karena menjadi bentuk awal dari bahasa pemrograman C gambar 2.12 menunjukkan sintaks dalam bahasa BCPL.

```

GET "LIBHDR"

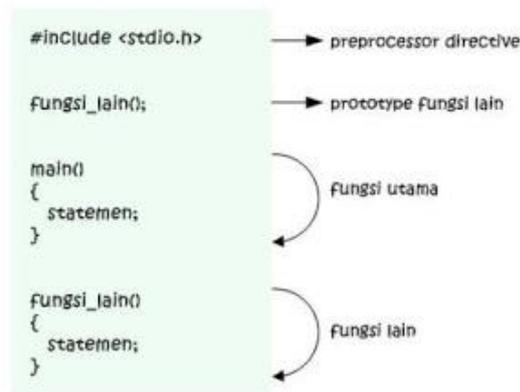
LET START() = VALOF $(
    FOR I = 1 TO 5 DO
        WRITEF("%N! = %I4*N", I, FACT(I))
    RESULTIS 0
$)

AND FACT(N) = N = 0 -> 1, N * FACT(N - 1)

```

**Gambar 2.12. Sintaks dalam bahasa BCPL**

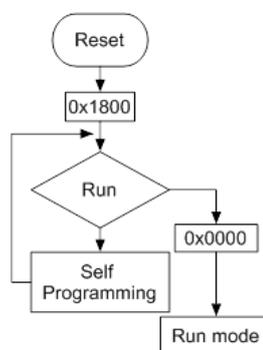
Bahasa pemrograman B dikembangkan di Bell laboratorium sekitar tahun 1969 oleh Ken Thompson dan Dennis Ritchie dengan mengurangi jumlah karakter pada bahasa BCPL dan dikembangkan terus menerus sehingga seperti bahasa C yang dikenal sekarang gambar 2.13 menunjukkan struktur bahasa C.



**Gambar 2.13. Struktur bahasa C**

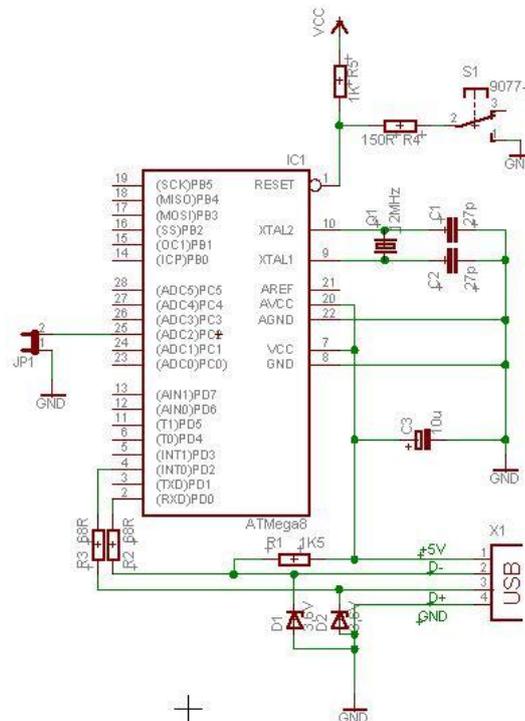
### 2.1.9. BootloaderHID

*BootloadHID* adalah USB *boot loader* untuk beberapa tipe Mikrokontroler AVR. Program ini bisa digunakan untuk mikrokontroler AVR dengan menggunakan memori 2 kB dalam sekmen *loader section*. Untuk Atmega 8 *firmware* ini di*flash* di 2 kB memori teratas pada peta memori dan akan dibaca langsung setelah tombol *reset* ditekan. Ketika perangkat dalam keadaan mode program bootloader akan menunggu data dari antar muka USB dan akan memuatnya ke memori *Flash* dibagian lain dalam peta memori (Thomas, 2007). Gambar 2.14 menunjukkan diagram alir program *bootloader* ketika pertama kali tombol *reset* ditekan.



**Gambar 2.14. Diagram Alir BootloadHID**

*BootloadHid* pada situs *open source* nya menyediakan *software* dan *hardware* yang bebas dimodifikasi oleh pengguna lain. Adapun rangkaian dasar *hardware bootloadhid avr* adalah ditunjukkan pada gambar 2.16.



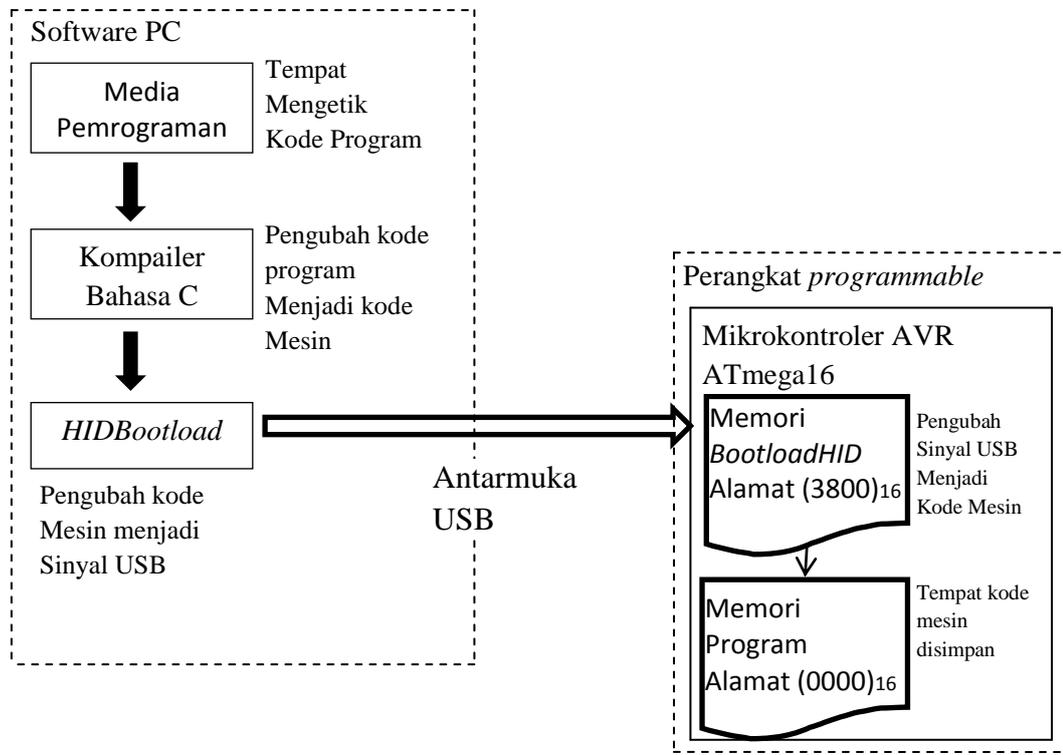
**Gambar 2.15. Rangkaian Dasar Bootloadhid**

## 2.2. Kerangka Berpikir

Perangkat *Programmable* bukan merupakan hal yang baru begitu pula media pemrogramannya yang menggunakan bahasa lain yang memudahkan dalam penggunaan perangkat tersebut. Banyak produk di pasaran yang menawarkan perangkat *programmable* oleh produsen yang eksis dibidang tersebut seperti *platform* Arduino, Rasberi Pi, PLC.

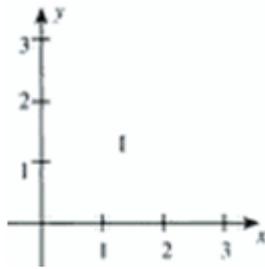
Dari teori yang dijelaskan pada kerangka teoritik dapat dibuat perangkat elektronik terprogram dan media pemrogramannya yang menggunakan bahasa indonesia dengan perancangan seperti Gambar 2.16. Perangkat yang akan dibuat

merupakan penggabungan dari hasil karya peneliti-peneliti lain yang digabung dan dimodifikasi.



**Gambar 2.16. Diagram Blok Sistem**

Peneliti akan menggunakan antarmuka pin pada perangkat dengan konsep seperti pada matematika koordinat kadrant I agar membuat alat menjadi sejalan dengan pengetahuan dasar yang dimiliki seseorang sehingga memudahkan dalam mengantar kepada pengetahuan selanjutnya. Gambar 2.17 adalah Koordinat pada kuadrant I.



**Gambar 2.17. Koordinat Kuadran I**

Peneliti menggunakan konsep koordinat karena konsep koordinat sangat penting dalam pembelajaran matematika karena merupakan dasar dari matematika turunan dan antiturunan yang merupakan matematika yang banyak digunakan dalam menyelesaikan permasalahan-permasalahan.

### **2.2.1 Penelitian Terkait**

Penelitian ini serupa dengan penelitian Hernando Barragan dari Institute Ivrea yang dilakukan pada June 2004 dengan pembimbing Massimo Banzi dan Casey Reas. Hernando Barragan melakukan penelitian dengan judul *Wiring: Prototyping Physical Interaction Design* dalam kutipan skripnya Hernando Barragan menjelaskan “Pendidikan dan praktiknya semakin mengandalkan teknologi digital. Alat-alat prototip yang ada saat ini untuk elektronik dan pemrograman sebagian besar digunakan untuk rekayasa, robotika dalam ranah teknis. Alat-alat prototip tersebut sulit dipelajari dengan bahasa pemrogramannya yang kurang berguna jika digunakan di teknologi tertentu”. Hernando Barragan membuat papan elektronik yang diberi nama *Wiring* yang dapat diprogram dengan aplikasi buatanya.

## **BAB III**

### **METODOLOGI PENELITIAN**

#### **3.1. Tempat dan Waktu Penelitian**

Penelitian ini dilakukan di gedung L1 ruang Laboratorium Lantai 4 Fakultas Teknik, Universitas Negeri Jakarta pada tanggal Januari 2017 sampai Juni 2017.

#### **3.2. Metode Penelitian**

Dengan melihat latar belakang diajukannya penelitian maka ditentukan prosedur yang tepat untuk mencapai hasil yang maksimal maka peneliti menggunakan metode *The Linier Sequential Model* untuk merancang dan membangun penelitian ini. Metode tersebut merupakan salah satu Kaidah dalam rekayasa teknik.

##### **3.2.1. Tujuan**

Tujuan penelitian ini adalah untuk merancang bangun perangkat *programmable* dan media pemrogramannya yang menggunakan kata dalam bahasa Indonesia sebagai produk media pembelajaran.

##### **3.2.3. Sasaran pengguna**

Produk hasil pengembangan akan ditujukan dalam bidang pendidikan. Hasil akan diterapkan sebagai media pembelajaran trainer perangkat *programmable* pada anak masa operasional formal yang dimulai dari usia 11 tahun ke atas.

##### **3.2.2. Metode**

Metode yang digunakan dalam penelitian ini adalah kaidah rekayasa perangkat lunak *The Linier Sequential Model*. Prosedur yang digunakan yaitu

*Analyze, Design, Code, Test*. Alur prosedur pada metode *The Linier Sequential* yang digunakan dapat dilihat pada gambar 3.1.



**Gambar 3.1. Diagram Alir Penelitian**

### 3.3. Prosedur Penelitian

Prosedur penelitian yang akan dilakukan secara garis besar berdasarkan langkah-langkah *The Linier Sequential Model*. Dari langkah-langkah tersebut peneliti menyusun tabel penelitian sabagai berikut:

**Tabel 3.1. Langkah-Langkah Penelitian *The Linier Sequential Model***

Prosedur	Tindakan
<p><i>a. Analyze</i> Melakukan analisis kerja untuk menentukan kebutuhan analisis kebutuhan.</p>	<ul style="list-style-type: none"> <li>- Menentukan Kebutuhan</li> </ul>
<p><i>b. Design</i> Menentukan kinerja yang akan dicapai berupa rencana-rencana dan pemilihan metode yang sesuai.</p>	<ul style="list-style-type: none"> <li>- Mendisain Arsitektur Perangkat.</li> <li>- Mendisain Struktur Perangkat.</li> <li>- Mendisain Antarmuka Perangkat.</li> <li>- Mendisain Algoritma.</li> </ul>
<p><i>c. Code</i> Menghasilkan Produk berupa trainer Pemrograman mikrokontroler beserta pelaksanaan teknisnya.</p>	<ul style="list-style-type: none"> <li>- Penulisan kode pada aplikasi</li> <li>- Penulisan Kode pada <i>firmware</i> perangkat keras</li> </ul>
<p><i>d. Test</i> Pengujian produk yang dibuat menggunakan metode V&amp;V</p>	<ul style="list-style-type: none"> <li>- <i>White Box Testing</i></li> <li>- <i>Black Box Testing</i></li> <li>- <i>Alpa Testing</i></li> <li>- <i>Beta Testing</i></li> </ul>

#### 3.3.1. Teknik Menganalisis

Analisis yang dilakukan merupakan penentu kebutuhan yang ditekankan pada perangkat lunak dan perangkat keras yang akan dibuat. Supaya peneliti

memahami sifat program yang akan dibuat, peneliti harus memahami perintah yang diperlukan, tingkah laku, kinerja, dan antarmuka dari program. Teknik yang digunakan pada tahap ini adalah dengan Diagram *fishbone* sering juga disebut Diagram *Cause-and-Effect* atau diagram Ishikawa. Diagram fishbone digunakan ketika kita ingin mengidentifikasi kemungkinan penyebab masalah dan terutama ketika sebuah team cenderung jatuh berpikir pada rutinitas (Tague, 2005:274) Peneliti menggunakan kaidah tersebut untuk menentukan kebutuhan perangkat yang akan dibuat.

### **3.3.2. Teknik Desain**

Pada tahap desain terdapat empat langkah atribut yang berbeda meliputi desain struksur, antarmuka, dan algoritma perangkat lunak.

1. Desain arsitektur merupakan penyajian hubungan dari setia sub sistem. Struktur merupakan bagian pokok dalam sebuah program, dalam tahapan ini desain Arsitektur program akan dibuat dengan diagram blok.
2. Desain Struktur merupakan penyajian hubungan logika dari aliran kerja subsistem. Struktur merupakan bagian penting dalam sebuah program, dalam tahapan ini desain struktur program akan dibuat dengan *flowcart*.
3. Desain Antarmuka merupakan tampilan pada suatu perangkat yang dapat dilihat dan berhubungan langsung dengan pengguna. Desain antarmuka ini bertujuan untuk memudahkan pengguna dalam mengoperasikan perangkat. Desain antar muka dibuat dengan gambar artistik.
4. Desain Algoritma merupakan cara atau langkah yang dibutuhkan untuk memecahkan suatu masalah. *Algoritma* merupakan pola pikir dari sebuah

perangkat lunak. Pada tahap ini peneliti mempresentasikan desain menggunakan *Pseudocode*.

### 3.3.3. Teknik Kode

Pada tahap Kode peneliti melakukan penerjemahkan desain ke dalam program perangkat lunak. Pada tahap pengimplementasian desain ke dalam kode program, kode harus mengikuti hasil desain perangkat lunak pada tahap sebelumnya. Kode akan ditulis dengan *Software Visual Basic 6* dan *GCC* untuk kode *firmware* pada perangkat keras akan ditulis dengan *WinAVR*.

### 3.3.4. Teknik Tes

Pada tahap *Test* peneliti melakukan pengujian-pengujian. Pengujian perangkat menurut Pressman(2001:479) merupakan elemen dalam pengembangan perangkat lunak yang sering disebut dengan *Verification and Validation (V&V)*. *Verifikasi* dilakukan untuk memastikan perangkat dapat berfungsi seperti yang telah ditentukan. *Validasi* dilakukan untuk memastikan perangkat lunak sudah sesuai dengan kebutuhan pengguna.

Tahap *Verification* meliputi:

1. *White Box Testing*, Menurut Irine (2006) merupakan pengujian untuk melakukan verifikasi program secara pengetahuan, program diuji untuk mengetahui benar tidaknya struktur dan kode dalam program.
2. *Black Box Testing*, merupakan pengujian berdasarkan kebutuhan keluaran tanpa pengetahuan struktur dalam program perangkat lunak.

Tahap *Validation Testing* menurut Pressman(2001:496) meliputi:

1. *Alpha Testing*, merupakan pengujian perangkat lunak dengan melibatkan para ahli.

2. *Beta Testing*, merupakan proses pengujian perangkat lunak oleh pengguna yang dipilih sesuai karakteristik pengguna perangkat lunak.

### **3.4. Teknik Pengumpulan Data**

Data dalam penelitian dikumpulkan dengan menggunakan kuisioner atau angket. Menurut Nasution (2006:128) angket atau *questionnaire* adalah daftar pernyataan yang didistribusikan melalui pos untuk diisi dan dikembalikan atau juga dijawab di bawah pengawasan peneliti. Angket pada umumnya meminta keterangan tentang fakta yang diketahui oleh responden atau juga mengenai pendapat atau sikap.

#### **3.2.4. Instrumen**

Jenis instrument penelitian berupa skala *likert* yang digunakan untuk mendapatkan data mengenai tingkat kelayakan. Prinsip pokok skala *likert* adalah suatu *kontinum* sikap terhadap objek sikap, mulai dari sangat negatif sampai dengan sangat positif (Widoyoko, 2009:115). Adapun pilihan jawaban dan scoring yang digunakan dalam angket yaitu:

SS = Sangat Setuju

S = Setuju

TS = Tidak Setuju

STS = Sangat Tidak Setuju

Angket penilaian ahli media terdiri dari 40 butir pernyataan yang akan diberikan kepada dosen selaku pengguna ahli sedangkan untuk angket pengguna terdiri dari 19 butir pernyataan yang akan diisi oleh mahasiswa sebagai *responden*. Adapun kisi-kisi aspek yang harus dipenuhi dapat dilihat pada tabel 3.2.

**Tabel 3.2. Kisi-kisi Aspek Yang Ingin Dicapai**

Konten materi		
No	Aspek	Indikator
1	Kualitas materi	<ul style="list-style-type: none"> <li>- Kesesuaian dengan kompetensi dasar</li> <li>- Meningkatkan kompetensi</li> <li>- Kemudahan penyajian</li> </ul>
2	Kemanfaatan	<ul style="list-style-type: none"> <li>- Alat Bantu bagi guru</li> <li>- Memudahkan peserta didik</li> </ul>
Kualitas Media		
No	Aspek	Indikator
1	Desain Media	<ul style="list-style-type: none"> <li>- Kualitas <i>hardware</i></li> <li>- Kualitas <i>Software</i></li> </ul>
2	Pegoprasian	<ul style="list-style-type: none"> <li>- Kemudahan</li> <li>- Efisiensi</li> </ul>
3	Kemanfaatan	<ul style="list-style-type: none"> <li>- Menambah motivasi Peserta didik</li> <li>- Menambah variasi Penyampaian materi</li> </ul>

**Tabel 3.3. Kisi-kisi Pengujian**

No	Aspek	Indikator
1	Kualitas media	<ul style="list-style-type: none"> <li>- <i>Hardware</i></li> <li>- <i>Software</i></li> </ul>
2	Pengoperasian media	<ul style="list-style-type: none"> <li>- Kemudahan</li> <li>- Efisiensi</li> </ul>
3	Pembelajaran	<ul style="list-style-type: none"> <li>- Menambah pegetahuan</li> <li>- Menambah motivasi</li> <li>- Meningkatkan keahlian</li> </ul>

Adapun tabel angket yang akan digunakan dalam teknik pengumpulan data dapat dilihat pada tabel 3.4 Pernyataan Untuk Ahli Media dan tabel 3.5. Pernyataan Untuk *Responden*.

Tabel 3.4. Pernyataan Untuk Ahli Media

No	Pernyataan deskriptif Konten Materi	Jawaban			
		STS	TS	S	SS
<b>Kualitas materi</b>					
1	Materi sesuai dengan konsep matematika koordinat				
2	Materi sesuai dengan konsep bilangan biner				
3	Materi sesuai dengan konsep listrik arus lemah				
4	Materi meningkatkan kemampuan kognitif siswa dalam bidang pemrograman				
5	Materi meningkatkan kemampuan Motorik siswa dalam bidang keahlian elektronika				
6	Ilustrasi dalam modul mudah dipahami				
7	Tatabahasa dalam modul mudah dimegerti				
8	Menggunakan pendekatan pegetahuan dasar				
9	Jobsheet mudah dipraktekan				
<b>Kemanfaatan materi</b>					
1	Menyediakan panduan penggunaan perangkat sebagai materi ajar				
2	Menyediakan contoh program sebagai tugas siswa				
3	Menyediakan jobsheet sebagai alat evaluasi				
4	Membuat siswa mau mengingat materi pelajaran yang sudah dilewati				
5	Menjelaskan konsep sistem otomasi				
<b>No Pernyataan deskriptif kualitas Media</b>		<b>Jawaban</b>			
<b>Hardware</b>		<b>STS</b>	<b>TS</b>	<b>S</b>	<b>SS</b>
1	Menggunakan komponen yang mudah didapat				
2	Menggunakan suplay listrik yang mudah didapat				
3	Desain <i>hardware</i> mudah dibuat				
4	Tataletak komponen rapih				
5	Memiliki kombinasi warna yang sesuai				
6	Penulisan keterangan mudah dipahami				
7	Desain <i>hardware</i> sesuai dengan filosofi matematik				
8	Dimensi perangkat praktis				
<b>Software</b>					
9	Aplikasi mudah dipasang pada PC lain				
10	Tata letak tombol pada aplikasi rapih				
11	Aplikasi memberikan animasi yang menarik				
12	Aplikasi memiliki kombinasi warna yang sesuai				
13	Mampu menampilkan materi yang bersifat abstrak				
14	Desain menggunakan bahasa indonesia sehingga memudahkan siswa				
15	Desan menggunakan karakter yang sesuai dengan tanda baca pada bahasa				
16	Desain sesuai dengan notasi matematik				
<b>Pengoperasian</b>					
1	Mudah menyambungkan <i>hardware</i> dengan PC				
2	Metode Menghubungkan Perangkat dan Properti yang lain mudah				

3	Aplikasi tidak memerlukan waktu proses yang lama				
4	Memiliki antarmuka Error pada perangkat				
5	Menampilkan antar muka pada editor yang membuat siswa tidak kebingungan dalam penggunaan				
<b>Kemanfaatan media</b>					
1	Membuat waktu praktek menjadi lebih singkat				
2	Memunculkan minat meng engineering				
3	Memunculkan daya tarik siswa untuk mencoba perangkat				
4	Menjelaskan konsep saklar elektronik				
5	Tidak Menghabiskan biaya yang besar dalam belajar				

Tabel 3.5. Pernyataan Untuk Responden

No	Pernyataan deskriptif	Jawaban			
		STS	TS	S	SS
<b>Kualitas Hardware</b>					
1	Menggunakan suplay listrik yang mudah didapat				
2	Tataletak komponen rapih				
3	Memiliki kombinasi warna yang sesuai				
4	Penulisan keterangan mudah dipahami				
5	Dimensi perangkat praktis				
<b>Kualitas Software</b>					
6	Aplikasi mudah dipasang pada PC lain				
7	Tata letak tombol pada aplikasi rapih				
8	Aplikasi memberikan animasi yang menarik				
9	Aplikasi memiliki kombinasi warna yang sesuai				
10	Mampu menampilkan materi yang bersifat abstrak				
11	Desain menggunakan bahasa indonesia sehingga mudah dipahami				
<b>Pengoperasian</b>					
1	Mudah dalam mengoperasikan perangkat				
2	Mudah dalam mengoperasikan aplikasinya				
3	Memberikan Antarmuka Error				
4	Mempersingkat waktu praktek				
<b>Kemanfaatan</b>					
1	Media pembelajaran memberi pengetahuan dalam bidang elektronika				
2	Media pembelajaran memberi pengetahuan dalam bidang pemrograman				
3	Media pembelajaran menimbulkan semangat belajar				
4	Media pembelajaran menimbulkan rasa ingintau				

### 3.2.4.1. Validitas Instrumen

Menurut Widoyoko(2009:98) alat ukur dikatakan valid apabila alat ukur itu dapat dengan tepat mengukur apa yang hendak diukur. Dengan katalain validitas berkaitan dengan “ketepatan” alat ukur tes. Instrumen yang telah divalidasi inilah yang digunakan untuk mengumpulkan data penelitian, adapun validitas instrumen dapat dilihat pada tabel 3.6 untuk pengguna ahli dan tabel 3.7 untuk responden.

**Tabel 3.6. Validitas Pengguna Ahli**

Konten materi				
No	Aspek	Indikator	Nomor Butir	Jumlah
1	Kualitas materi	<ul style="list-style-type: none"> <li>- Kesesuaian dengan kompetensi dasar</li> <li>- Meningkatkan kompetensi</li> <li>- Kemudahan penyajian</li> </ul>	1,2,3 4,5 6,7,8,9	9
2	Kemanfaatan	<ul style="list-style-type: none"> <li>- Alat Bantu bagi guru</li> <li>- Memudahkan peserta didik</li> </ul>	1,2,3 4,5	5
Kualitas Media				
No	Aspek	Indikator	Nomor Butir	Jumlah
1	Desain Media	<ul style="list-style-type: none"> <li>- Kualitas <i>Hardware</i></li> <li>- Kualitas <i>Software</i></li> </ul>	1,2,3,4,5,6, 7,8 9,10,11,12, 13,14,15,16	16
2	Pegoprasian	<ul style="list-style-type: none"> <li>- Kemudahan</li> <li>- Efesiensi</li> </ul>	1,4,5 3,2	5
3	Kemanfaatan	<ul style="list-style-type: none"> <li>- Menambah motivasi Peserta didik</li> <li>- Menambah variasi Penyampaian materi</li> </ul>	2,3 1,4,5	5
Jumlah				40

**Tabel 3.7. Validitas kisi-kisi Responden**

No	Aspek	Indikator	Nomor Butir	Jumlah
1	Kualitas media	- <i>Hardware</i> - <i>Software</i>	1,2,3,4,5 6.7.8.9.10.11	11
2	Pengoperasian media	- Kemudahan - Efisiensi	12, 13, 14 15	4
3	Pembelajaran	- Menambah pegetahuan - Menambah motivasi - Meningkatkan keahlian	17 18, 19 16	4
Jumlah				19

### 3.2.4.2. Reliabilitas Instrumen

Reliabilitas dalam bahasa indonesia diambil dari kata *reliability* yang artinya dapat dipercaya (*reliable*) jika memberikan hasil yang tetap atau ajek (konsisten) apabila diteskan berkali-kali (Widoyoko, 2009:144). Dalam penelitian ini digunakan rumus alpha untuk melakukan uji reliabilitas. Rumus alpha digunakan untuk pengujian instrumen responden untuk mengetahui tingkat reliabilitasnya. Rumus pengujian reliablitas *alpha* seperti berikut:

$$r_{11} = \left( \frac{k}{(k-1)} \right) \left( 1 - \frac{\sum \sigma_b^2}{\sigma_t^2} \right) \dots\dots\dots(1)$$

Keterangan:

$r_{11}$  = reliabilitas instrumen

$k$  = jumlah butir

$\sum \sigma_b^2$  = jumlah varians skor tiap butir

$\sigma_t^2$  = varians total

Rumus untuk varians:

$$\sigma_i^2 = \frac{\sum X^2 - \frac{(\sum X)^2}{N}}{N} \dots\dots\dots(2)$$

Keterangan:

$\sum X^2$  = jumlah kuadrat skor total

$(\sum X)^2$  = kudrat jumlah skor tiap item N

N = jumlah responden

Setelah koefisien reliabilitas diketahui, maka selanjutnya dicocokkan ke sebuah kategori acuan. Untuk menyajikan koefisien alpha digunakan kategori yang dilihat pada tabel 3.8 dikutip dari arikunto dalam (Agung 2010: 95).

**Tabel 3.8. Kategori Koefisien Reliabilitas**

Koefisien Reliabilitas	Tingkat Reliabilitas
0,00 s.d. 0,20	Kurang Reliabel
0,20 s.d. 0,39	Agak Reliabel
0,40 s.d. 0,59	Cukup Reliabel
0,60 s.d. 0,79	Reliabel
0,80 s.d. 1,00	Sangat Reliabel

#### 4.5. Teknik Analisis Data Penelitian

Teknik Analisis yang digunakan pada penelitian kali ini adalah Analisis Data kualitatif berupa pernyataan yang diberikan oleh dosen sebagai pengguna ahli dan mahasiswa sebagai *responden*. Untuk mendapatkan penilaian kelayakan media, maka data kualitatif tersebut dikonversi menjadi data kuantitatif dengan ketentuan scoring seperti pada tabel 3.9.

**Tabel 3.9. Ketentuan Pemberian Skor**

Kriteria	Skor
Sangat Tidak Setuju	1
Tidak Setuju	2
Setuju	3
Sangat Setuju	4

Kemudian data yang terkumpul dianalisis dengan cara menghitung rata-rata skor yang diperoleh dengan rumus:

$$\bar{x} = \frac{\sum x}{n} \dots\dots\dots(3)$$

Rata-rata penilaian yang diperoleh dikonversi menjadi nilai persentase kelayakan dengan rumus sebagai berikut:

$$\text{Kelayakan (\%)} = \frac{\sum \text{Hasil skor}}{\sum \text{Skor max}} \times 100\% \dots\dots\dots(4)$$

Kemudian untuk mencari kategori kelayakan media menggunakan pedoman konversi skor ideal yang dijabarkan pada Tabel 3.10 berikut ini (Widoyoko, 2009: 238).

**Tabel 3.10. Pedoman Konversi Skor**

No	Rumus	Kategori
1	$x_i + (1,8 \text{ SBi}) < X \leq \text{Skor Max}$	Sangat Layak
2	$x_i + (0,6 \text{ SBi}) < X \leq x_i + (1,8 \text{ SBi})$	Layak
3	$x_i + (-0,6 \text{ SBi}) < X \leq x_i + (0,6 \text{ SBi})$	Cukup Layak
4	$x_i + (-1,8 \text{ SBi}) < X \leq x_i + (-0,6 \text{ SBi})$	Kurang Layak
5	$\text{Skor Min} < X \leq x_i + (-1,8 \text{ SBi})$	Sangat Kurang Layak

Keterangan:

$x_i$  = (Rerata ideal)

SBi = (Simpangan Baku Ideal)

X = Skor Aktual

Berdasarkan ketentuan skor pada tabel 3.9, maka nilai skor minimum adalah 1 dan maksimum adalah 4. Sehingga diperoleh persentase nilai minimum adalah 25% dan maksimum adalah 100%. Jika persentase nilai minimum dan maksimum disubstitusikan pada rumus yang ada pada Tabel 3.10 maka diperoleh pedoman pengkonversian seperti berikut ini :

$x_i$  = (Rerata ideal)

=  $\frac{1}{2}$  (skor maksimum ideal + skor minimum ideal)

=  $\frac{1}{2}$  (100 + 25) = 62,5 SBi

SBi = (Simpangan Baku Ideal)

=  $1/6$  (Skor maksimum ideal – Skor minimum ideal)

=  $1/6$  (100 – 25) = 12,5

X = Skor Aktual

**Tabel 3.11. Konversi Persentase Skor Menjadi Kategori Kualitatif**

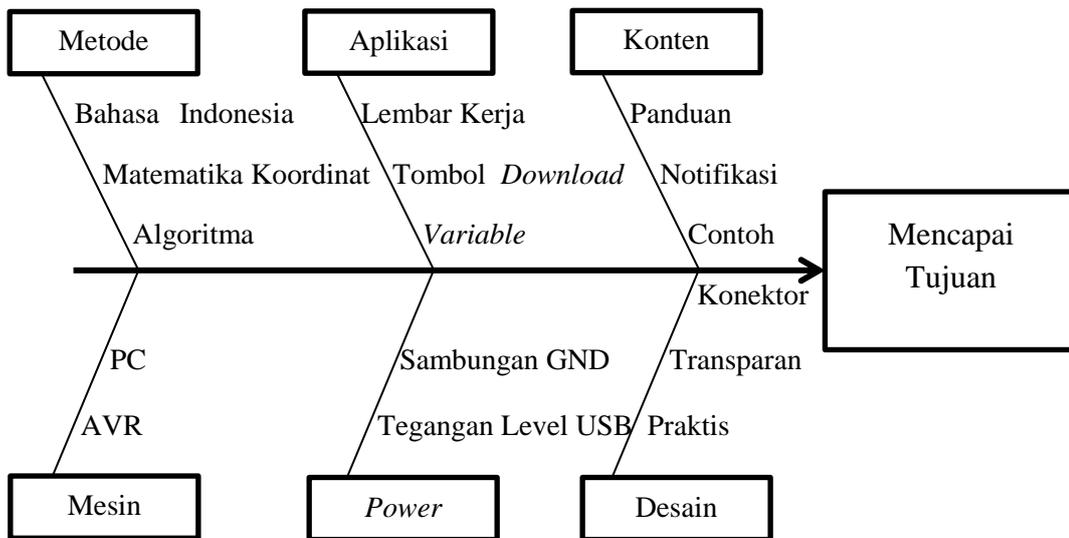
No	Nilai	Kategori
1	$85\% < X \leq 100\%$	Sangat Layak
2	$70\% < X \leq 85\%$	Layak
3	$55\% < X \leq 70\%$	Cukup Layak
4	$40\% < X \leq 55\%$	Kurang Layak
5	$25\% < X \leq 40\%$	Sangat Kurang Layak

## BAB IV

### HASIL PENELITIAN

#### 4.1. Hasil Menganalisis Kebutuhan

Dalam tahap menganalisis kebutuhan peneliti menggunakan diagram *Fishbone* untuk menentukan kebutuhan apa saja yang akan digunakan dalam pembuatan perangkat, diagram dapat dilihat pada gambar 4.1.



Gambar 4.1. Diagram Analisis Kebutuhan

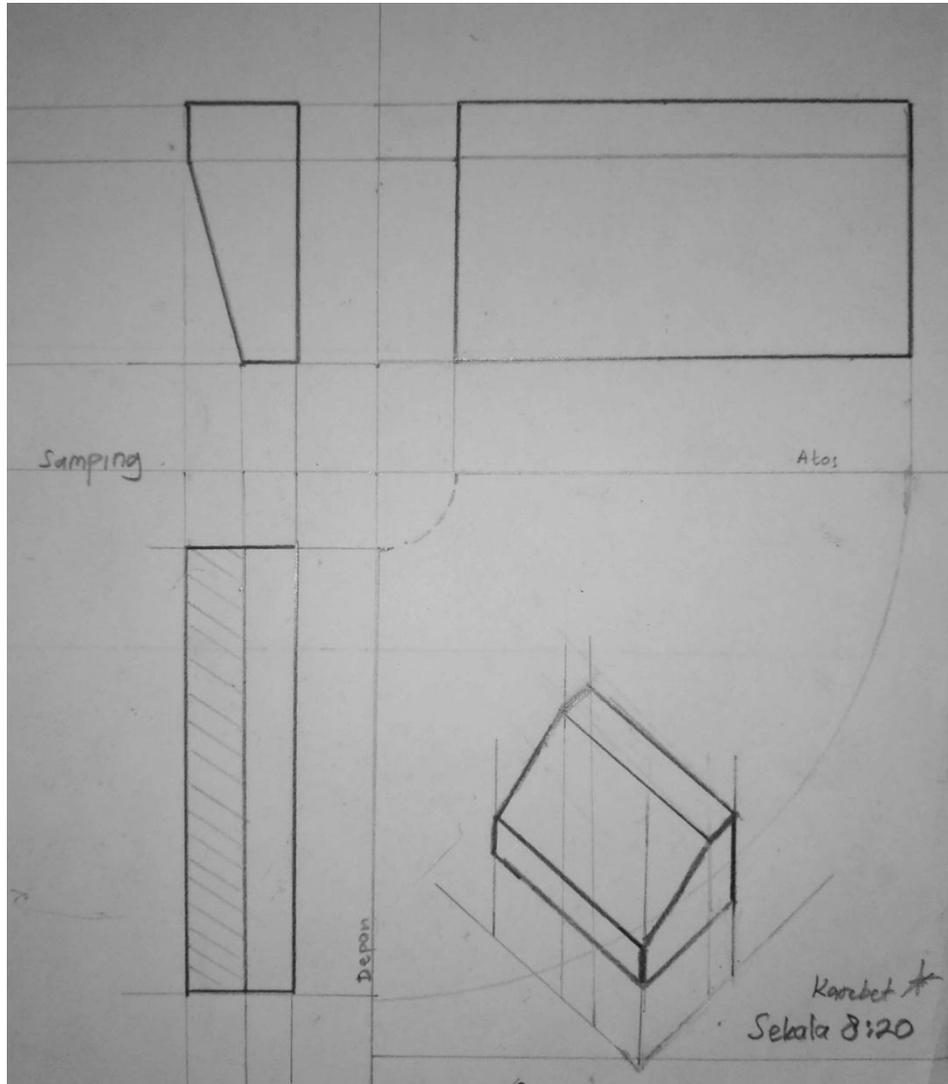
Tujuan Penelitian “Merancang dan membuat perangkat *programmable* yang dapat diprogram langsung dengan media pemrogramannya yang menggunakan kata dalam bahasa Indonesia sebagai produk media pembelajaran”.

#### 4.2. Hasil Proses Desain

Pada tahap desain dibagi menjadi dua yaitu desain *hardware* dan desain *software* desain dibuat sesuai dengan hasil analisis kebutuhan yang didapat. Peneliti mendesain *hardware* terlebih dahulu sebelum mendesain *software*.

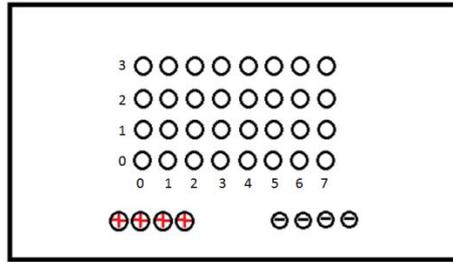
#### 4.2.1. Desain Hardware

Desain *hardware* dibuat dengan pertimbangan dimensi perangkat agar menjadi praktis. Desain dapat dilihat pada gambar 4.2.



**Gambar 4.2. Desain Perangkat**

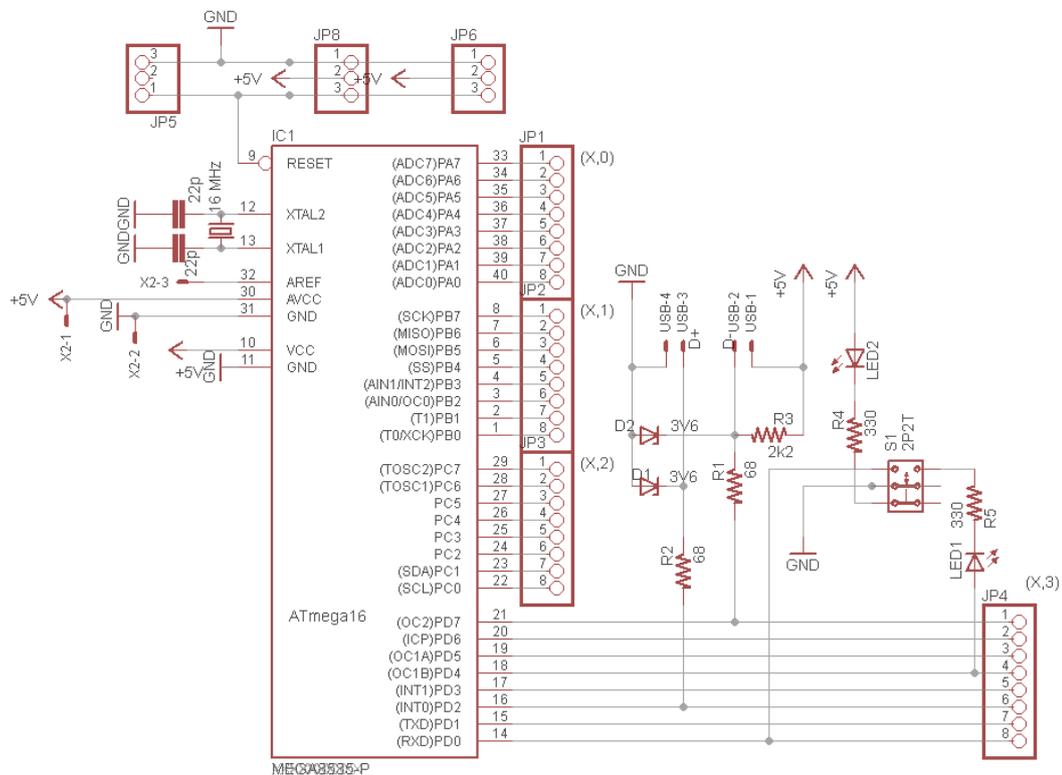
Desain dibuat agar sesuai dengan konsep matematika koordinat pada kuadran I sebagai antarmuka perangkat kerasnya desain antarmuka pin dapat dilihat pada gambar 4.3.



Gambar 4.3. Desain Antarmuka Pin

4.2.2. Desain Elektrikal

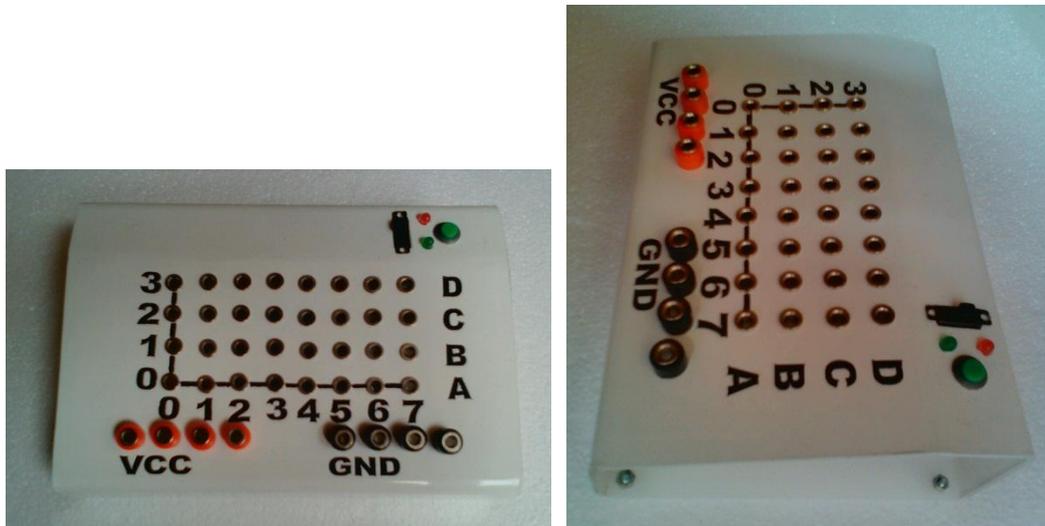
Desain Elektrikal menggunakan desain yang diperkenalkan oleh (Thomas, 2007) dalam websitenya <https://www.obdev.at/products/vusb/index.html>. Dengan adanya kebutuhan jumlah pin maka desain dimodifikasi menggunakan mikrokontroler ATmega16. Berikut adalah hasil desain elektrikal *HIDbootloader* yang telah dimodifikasi dapat dilihat pada gambar 4.4.



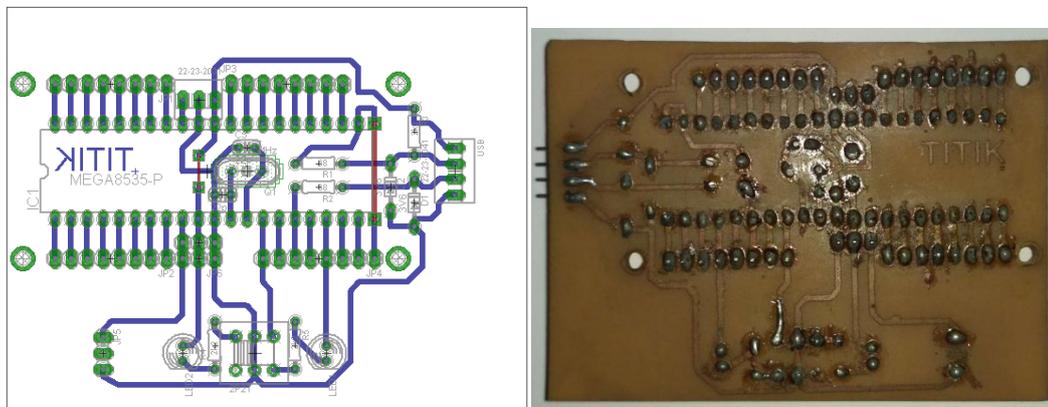
Gambar 4.4. Sekematik Perangkat

#### 4.2.4. Hasil Implementasi Desain Hardware

Desain yang dijelaskan di atas merupakan rujukan dari pembuatan perangkat yang dibuat. Hasil desain dapat dilihat pada gambar 4.5.



Gambar 4.5. Perangkat



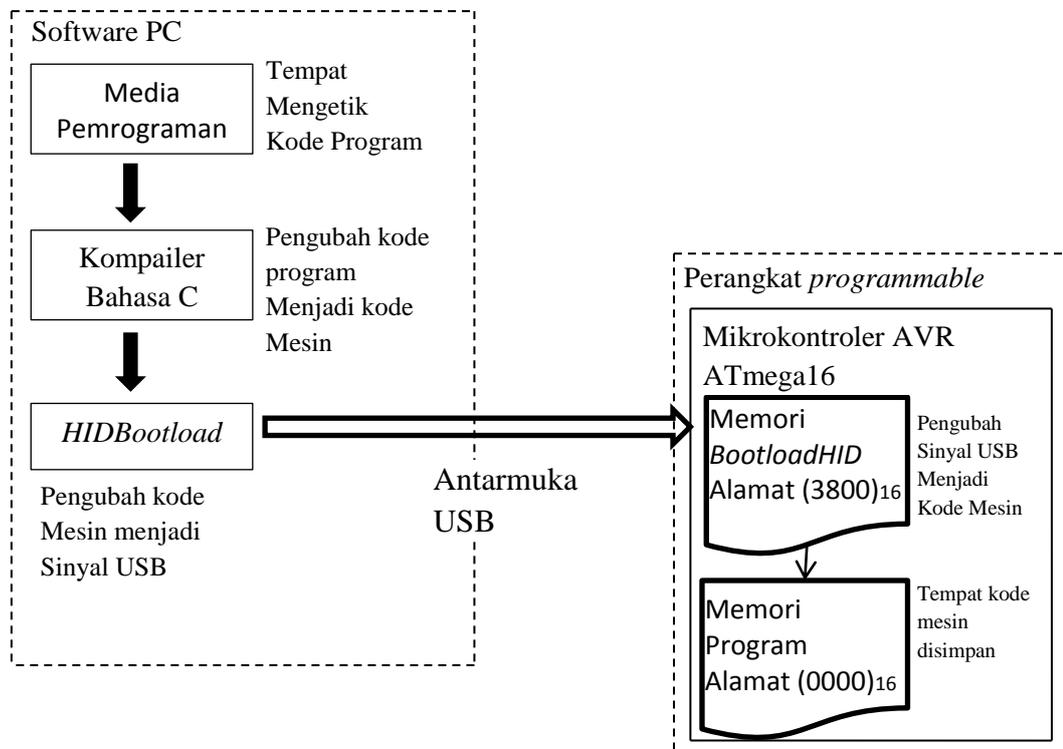
Gambar 4.6. Rangkaian Perangkat

#### 4.2.4. Desain Software

Desain *software* dibuat dengan menggunakan *flowcart* dan desain antarmuka dibuat dengan gambar artistik.

### A. Desain arsitektur perangkat

Desain arsitektur perangkat dapat dilihat pada diagram blok sistem gambar 4.7.

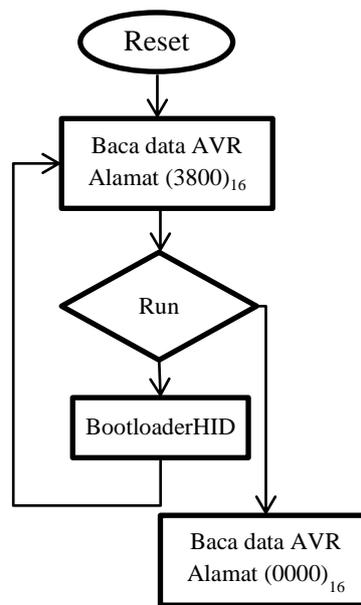


**Gambar 4.7. Desain Sistem Pada Perangkat**

Dari diagram blok *software* PC pada blok media pemrograman berbentuk antarmuka aplikasi yang nantinya digunakan untuk menulis kode program, *output* dari blok tersebut adalah seket pemrograman bahasa C. Blok kompailer bahasa C berbentuk *software* dengan antarmuka *command line*, merupakan *software open source* dari GCC yang dapat dimodifikasi, *output* blok tersebut adalah kode mesin AVR. Pada blok HIDBootload berbentuk *software* yang digunakan untuk mengubah tabel memori kode mesin menjadi sinyal NRZI yang digunakan USB. Pada blok Perangkat *Programmable* digunakan mikrokontroler dari AVR dengan seri atmega16 yang ditanam program *bootloadHID* pada alamat  $(3800)_{16}$ . *BootloadHID* digunakan untuk merubah sinyal NRZI pada antarmuka USB menjadi tabel memori dan akan ditempatkan di alamat memori  $(0000)_{16}$ .

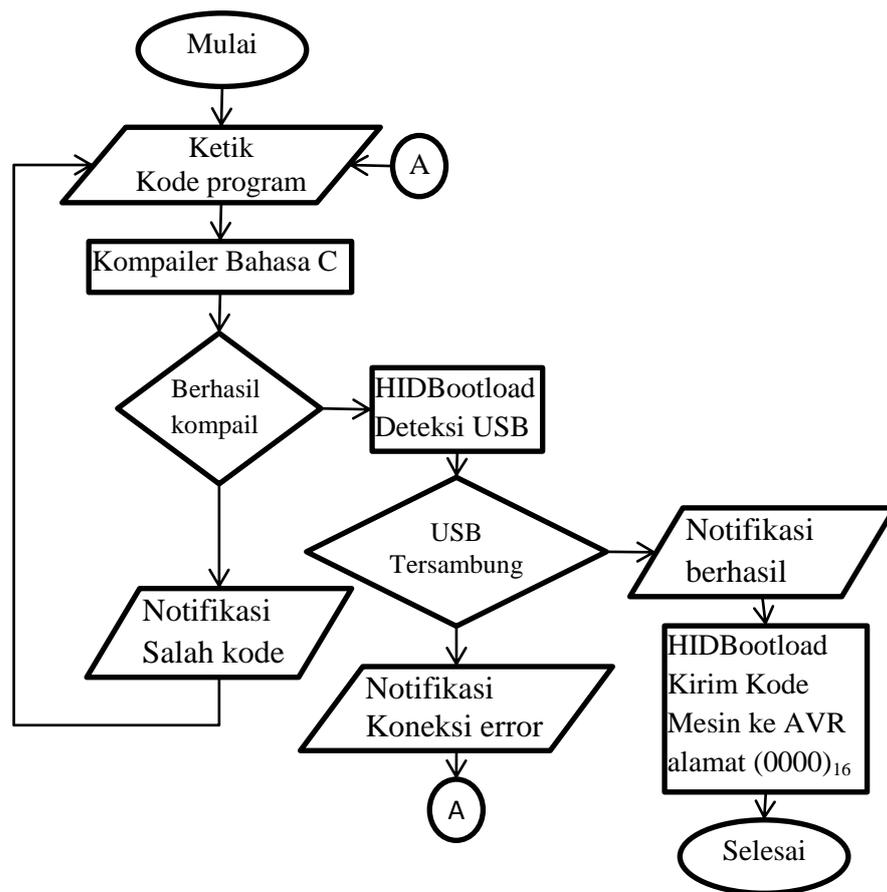
## B. Desain struktur

Desain struktur aliran program kerja perangkat dapat dilihat pada *flowcart* sistem gambar 4.8 dan untuk desain struktur kerja *software* Aplikasi dapat dilihat pada gambar 4.9.



**Gambar 4.8. Desain struktur aliran program Perangkat**

Pada *flowcart* gambar 4.8 menunjukkan aliran data dimulai dari *reset* yang diteruskan dengan pembacaan pada alamat memori  $(3800)_{16}$  yang merupakan tempat program *BootloadHID*. *BootloadHID* akan mendeteksi perangkat sedang di *mode program* atau di *mode run* dengan membaca logika pada PIN PC0 jika perangkat berada pada *mode program* ( $PC0 = Low$ ) maka aliran program akan melanjutkan pembacaan algoritma *BootloadHID* selanjutnya dan jika perangkat berada pada *mode run* maka aliran program akan membaca alamat memori  $(0000)_{16}$  yang merupakan tempat program yang akan dijalankan oleh perangkat.

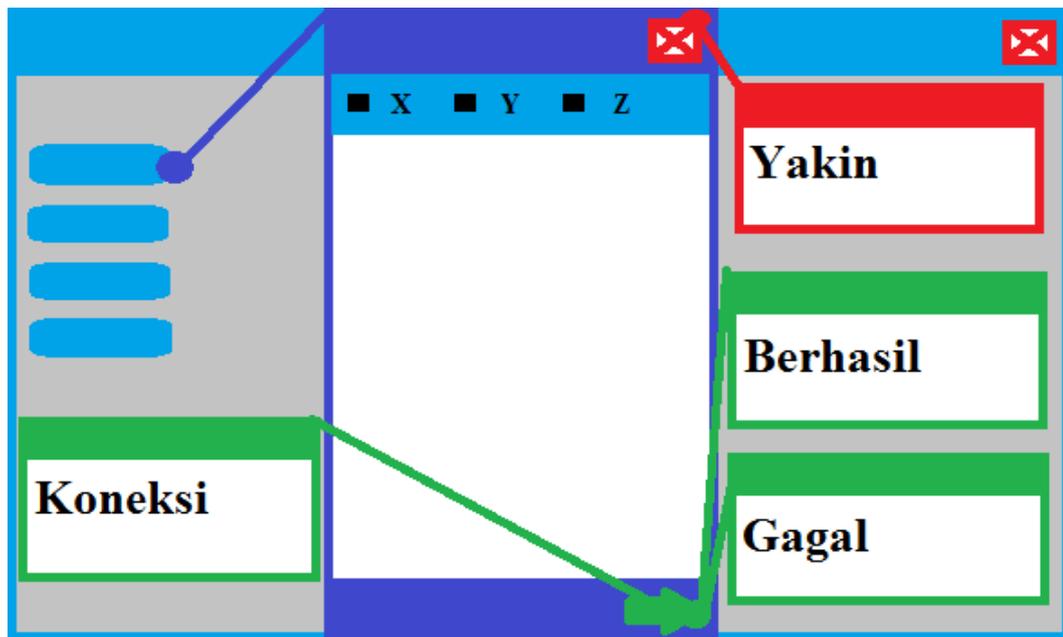


**Gambar 4.9. Desain Struktur Aliran Program Aplikasi**

Pada *flowcart* gambar 4.9 menunjukkan aliran program pada *software aplikasi* yang menyatakan aliran program dimulai dari *input* kode program yang berupa program dalam bahasa Indonesia, aliran program selanjutnya akan dilanjutkan ke proses kompilasi dengan kompailer bahasa C dengan *output* berupa kode mesin yang menandakan kompilasi berhasil. Kode mesin hasil kompilasi akan diproses oleh *HIDBootload* yang akan merubah menjadi Sinyal NRZI yang digunakan oleh USB untuk berkomunikasi dengan perangkat untuk megirim informasi berupa kode mesin yang akan ditempatkan pada alamat memori  $(0000)_{16}$  pada memori perangkat.

### C. Desain Antarmuka

Desain antarmuka perangkat dibuat praktis tanpa adanya atribut yang membingungkan agar memudahkan pengguna. Desain dapat dilihat pada gambar 4.10.



**Gambar 4.10. Desain Antarmuka**

### D. Desain Algoritma

Agar memudahkan pengguna peneliti merubah sistem PORT pada ATmega16 dengan sistem koordinat dengan desain seperti berikut :

Algoritma nyalakan koordinat

*Input (x,y)*

Jika  $y = 0$  maka gunakan PORT A

Jika  $y = 1$  maka gunakan PORT B

Jika  $y = 2$  maka gunakan PORT C

Jika  $y = 3$  maka gunakan PORT D

Jika  $x = 0$  maka  $\text{PORT X} \leftarrow \text{PORT X OR } (1)_{16}$

Jika  $x = 1$  maka  $\text{PORT X} \leftarrow \text{PORT X OR } (2)_{16}$

Jika  $x = 2$  maka  $\text{PORT X} \leftarrow \text{PORT X OR } (4)_{16}$

Jika  $x = 3$  maka  $\text{PORT X} \leftarrow \text{PORT X OR } (8)_{16}$

Jika  $x = 4$  maka  $\text{PORT X} \leftarrow \text{PORT X OR } (10)_{16}$

Jika  $x = 5$  maka  $\text{PORT X} \leftarrow \text{PORT X OR } (20)_{16}$

Jika  $x = 6$  maka  $\text{PORT X} \leftarrow \text{PORT X OR } (40)_{16}$

Jika  $x = 7$  maka  $\text{PORT X} \leftarrow \text{PORT X OR } (80)_{16}$

Algoritma matikan koordinat

*Input* (x,y)

Jika  $y = 0$  maka gunakan PORT A

Jika  $y = 1$  maka gunakan PORT B

Jika  $y = 2$  maka gunakan PORT C

Jika  $y = 3$  maka gunakan PORT D

Jika  $x = 0$  maka  $\text{PORT X} \leftarrow \text{PORT X AND } (FE)_{16}$

Jika  $x = 1$  maka  $\text{PORT X} \leftarrow \text{PORT X AND } (FD)_{16}$

Jika  $x = 2$  maka  $\text{PORT X} \leftarrow \text{PORT X AND } (FB)_{16}$

Jika  $x = 3$  maka  $\text{PORT X} \leftarrow \text{PORT X AND } (F7)_{16}$

Jika  $x = 4$  maka  $\text{PORT X} \leftarrow \text{PORT X AND } (EF)_{16}$

Jika  $x = 5$  maka  $\text{PORT X} \leftarrow \text{PORT X AND } (DF)_{16}$

Jika  $x = 6$  maka  $\text{PORT X} \leftarrow \text{PORT X AND } (BF)_{16}$

Jika  $x = 7$  maka  $\text{PORT X} \leftarrow \text{PORT X AND } (7F)_{16}$

Agar menggunakan bahasa Indonesia peneliti menyisipkan *library* pengubah sintak pada bahasa C menjadi kata dalam bahasa Indonesia dengan desain sebagai berikut :

Perubah sintak menjadi kata dalam bahasa Indonesia fiktur bahasa C

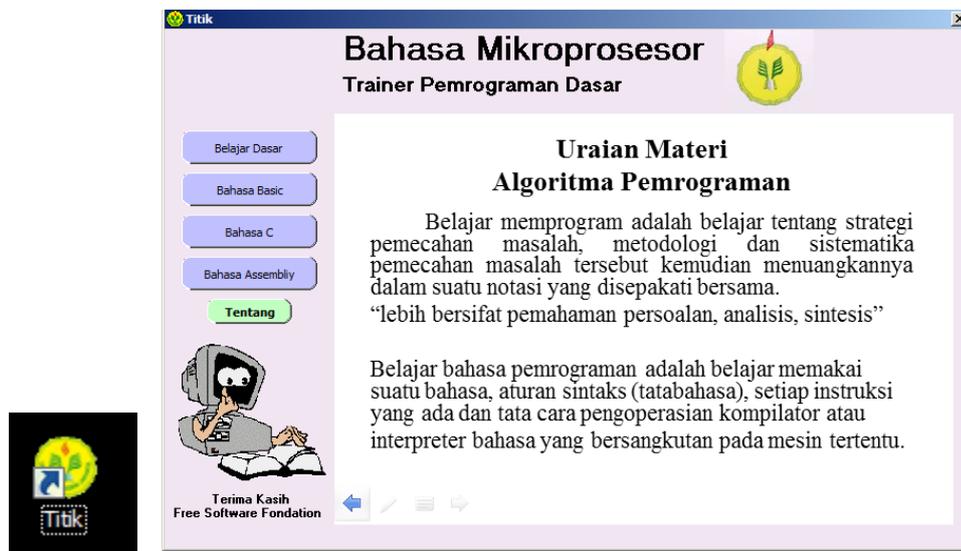
```
#define ketika while(
#define jika if(
#define maka )
#define lakukan {
#define tunda _delay_ms(
#define mili_detik );
#define detik *1000);
#define loncat_ke goto
#define mulai {
#define selesai }
```

Perubah sintak menjadi kata dalam bahasa Indonesia dengan *Visual basic 6*

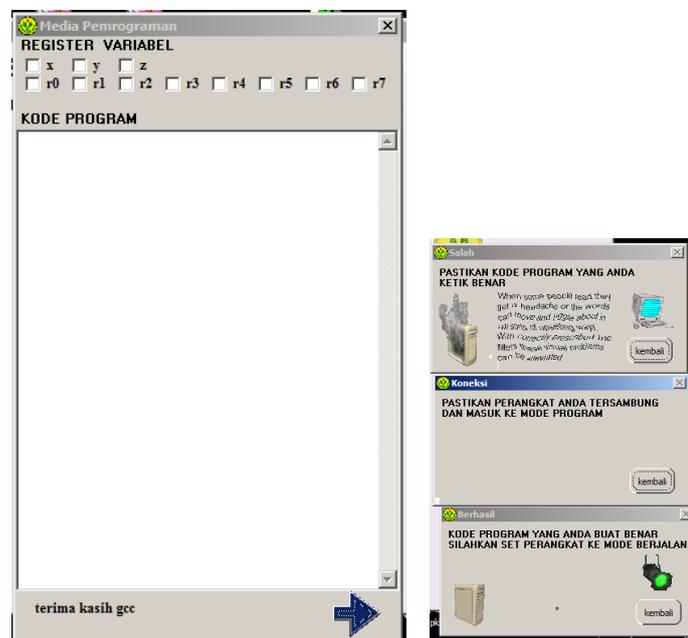
```
replace(kode, "artikan", "#define")
replace(kode, ".", ";")
replace(kode, "!", ";")
```

#### 4.2.4. Hasil Implementasi Desain Software

Desain yang dijelaskan di atas merupakan rujukan dari pembuatan *software* yang dibuat sehingga menjadi aplikasi jadi yang dilihat pada gambar dibawah :



Gambar 4.11. Shortcut dan Halaman Media



Gambar 4.12. Lembar Kerja dan Notifikasi Software

### 4.3. Hasil Kode

Untuk menerapkan hasil desain ke perangkat. Perangkat harus diprogram dengan bahasa pemrograman yang sesuai dengan komponen yang digunakan. Sebelum memulai pengkodean peneliti memasang *software Visual basic 6*, *GCC*, *Khazama AVR programmer* dan *winAVR 2010*. *Software Visual basic 6* digunakan untuk membuat antarmuka aplikasi. *Software GCC* digunakan untuk memodifikasi *HIDBootload* agar menyesuaikan tampilan antarmukanya. *Software Khazama AVR programmer* digunakan untuk menanamkan *BootloadHID* ke mikrokontroler agar sesuai dengan sistem pada perangkat. *Software winAVR 2010* digunakan untuk membuat *BootloadHID* dan digunakan sebagai kompailer bahasa C pada aplikasi. Proses modifikasi kode menggunakan *software notepad++*.

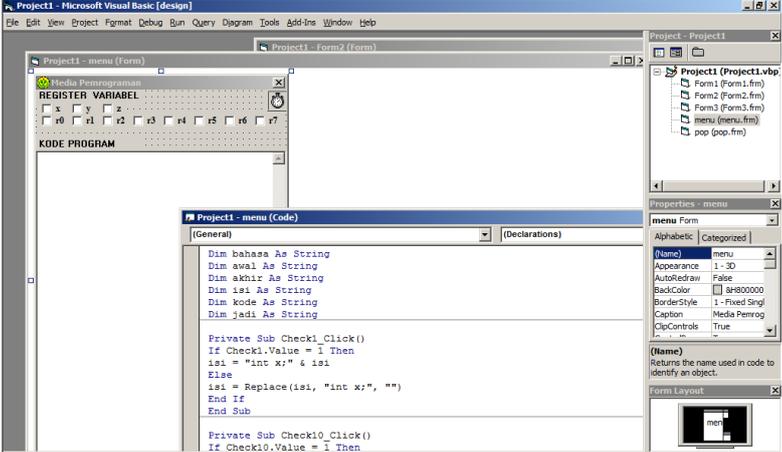
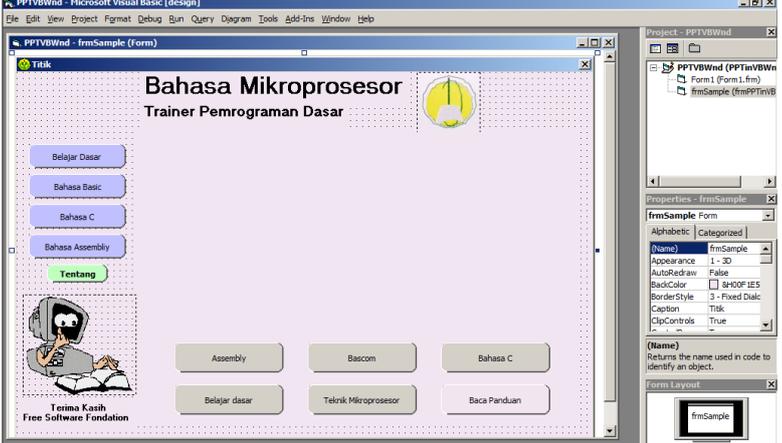
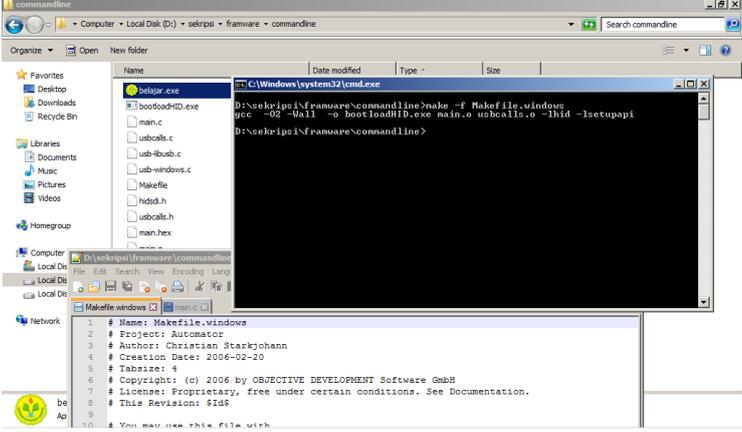
#### 4.3.1. Kode Aplikasi Media

Proses pengkodean media menggunakan *PC/Laptop* dengan *software Visual basic 6* dan kompailer *GCC*. Pengkodean disesuaikan dengan desain yang telah dibuat. Dalam poses pengkodean dihasilkan *file* hasil kode yaitu :

1. titik.exe
2. belajar.exe
3. hidbootload.exe

Ketiga *file* tersebut saling berhubungan sehingga membuat media berkerja. Untuk kode program yang telah dibuat dapat dilihat pada lampiran. Ketiga *file* tersebut akan dihubungkan dengan perintah pada “comm.bat”. Kode *file* “comm.bat” dapat dilihat pada lampiran. Adapun proses dalam pembuatan *software* dapat dilihat pada tabel 4.1.

Tabel 4.1. Pembuatan Aplikasi

Proses pembuatan	Perintah
<p>kompilasi antarmuka dengan <i>software Visual basic 6</i></p> 	<p>Klik <i>File</i> → <i>Make belajar.exe</i></p>
<p>kompilasi <i>HIDBootload</i> dengan <i>software GCC</i></p> 	<p>Klik <i>File</i> → <i>Make titik.exe</i></p>
<p>“make -f makefile.windows ” pada <i>command line</i> di tempat <i>file</i> berada.</p> 	<p>“make -f makefile.windows ” pada <i>command line</i> di tempat <i>file</i> berada.</p>

### 4.3.2. Kode Firmware Perangkat

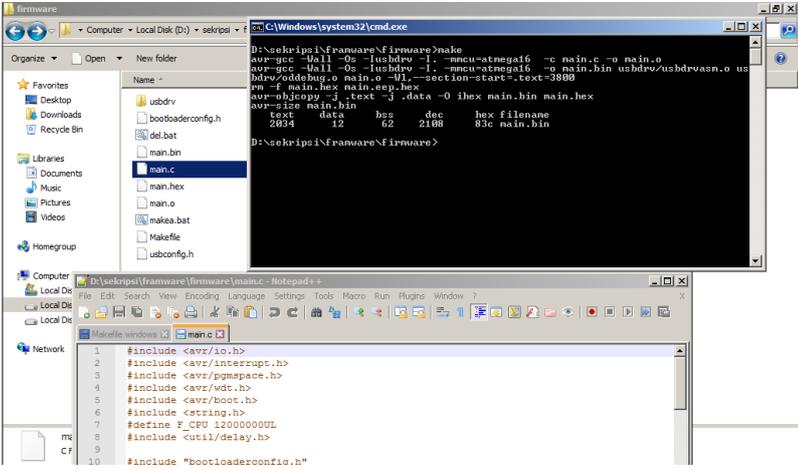
Proses pengkodean *firmware* menggunakan PC/Laptop dengan *software winAVR 2010*. Pengkodean *firmware* merupakan hasil modifikasi *BootloadHID*.

Dalam proses pengkodean dihasilkan *file* hasil kode yaitu :

1. main.hex

*File* tersebut ditanamkan kedalam ATmega16 pada alamat memori  $(3800)_{16}$  sebagai bootloader agar berkerja mengkoneksikan perangkat ke PC melalui port USB. Untuk kode program yang telah dimodifikasi dapat dilihat pada lampiran. Adapun proses dalam pembuatan *software* dapat dilihat pada tabel 4.2.

**Tabel 4.2. Pembuatan firmware**

Proses pembuatan	Perintah
<p>kompilasi <i>BootloadHID</i> dengan <i>software winAVR 2010</i></p> 	<p>“make” pada <i>command line</i> di tempat <i>file</i> berada.</p>

*file* main.hex ditanamkan ke perangkat menggunakan *USBASP* dengan *software Khazama AVR programmer*.

#### 4.4. Hasil Tes

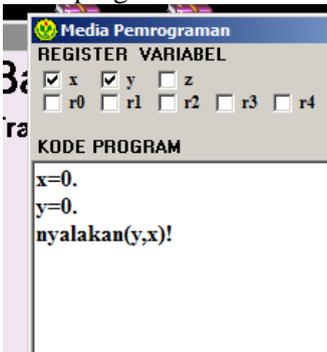
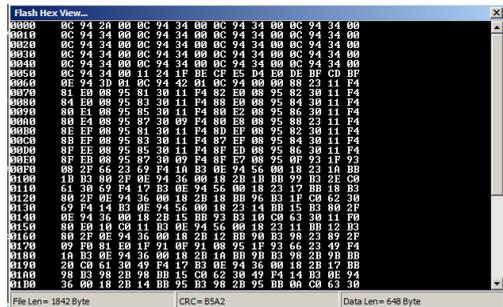
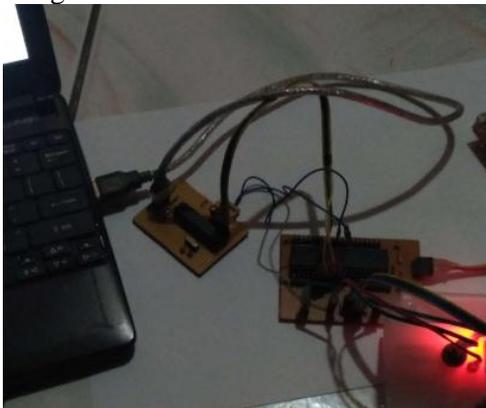
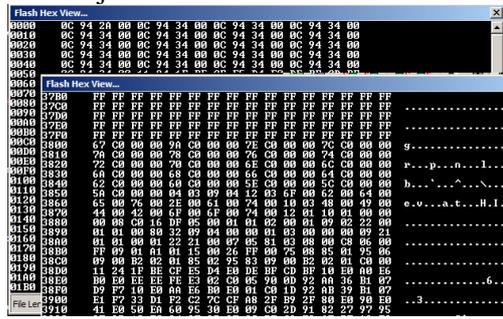
Hasil *Test* yang didapat dari uji coba perangkat terdiri dari *white box testing*, *black box testing*, *alpa testing* dan *beta testing* yang menentukan tingkat keberhasilan penelitian yang dikerjakan.

##### 4.4.1. White box Testing

Uji *White box testing* dilakukan pada saat perangkat sedang dibuat dengan tujuan agar mengetahui sistem berjalan sesuai dengan fungsinya atau tidak. Hasil uji *white box testing* dapat dilihat pada tabel berikut :

##### A. Pengujian Sub Sistem

**Tabel 4.3. Pengujian Kompailer dan Mikrokontroler**

No	Uji Compiler dengan <i>Software</i> Khazama AVR Programmer	
1	<p>Kode program</p>  <pre> x=0. y=0. nyalakan(y,x)!                     </pre>	<p>Kode mesin</p>  <pre> 0000 0C 94 20 00 0C 94 34 00 0C 94 34 00 0C 94 34 00 0010 0C 94 34 00 0C 94 34 00 0C 94 34 00 0C 94 34 00 0020 0C 94 34 00 0C 94 34 00 0C 94 34 00 0C 94 34 00 0030 0C 94 34 00 0C 94 34 00 0C 94 34 00 0C 94 34 00 0040 0C 94 34 00 0C 94 34 00 0C 94 34 00 0C 94 34 00 0050 0C 94 34 00 0C 94 34 00 0C 94 34 00 0C 94 34 00 0060 0C 94 3D 01 0C 94 42 01 0C 94 00 00 00 23 11 F4 0070 81 E0 08 95 81 30 11 F4 82 E0 08 95 82 30 11 F4 0080 84 E0 08 95 83 30 11 F4 80 08 95 84 30 11 F4 0090 80 E1 08 95 85 30 11 F4 80 E2 08 95 86 30 11 F4 00A0 80 E3 08 95 87 30 09 F4 80 E3 08 95 88 30 11 F4 00B0 80 E5 08 95 81 30 11 F4 80 EF 08 95 89 30 11 F4 00C0 80 EF 08 95 83 30 11 F4 87 EF 08 95 8A 30 11 F4 00D0 80 EF 08 95 85 30 11 F4 87 EF 08 95 8B 30 11 F4 00E0 8F EB 08 95 87 30 09 F4 8F E7 08 95 0F 93 1F 93 00F0 08 2F 66 23 69 F4 23 B3 0E 94 56 00 18 23 10 BB 0100 1B B3 09 2F 0E 94 36 00 18 2B 10 BB 79 B3 2E C0 0110 61 30 69 F4 17 B3 0E 94 56 00 18 23 17 BB 18 B3 0120 80 2F 0E 94 36 00 18 2B 18 B3 0E 94 56 00 18 23 0130 69 F4 14 B3 0E 94 56 00 18 23 14 BB 15 B3 00 2F 0140 0E 94 36 00 18 2B 15 BB 23 10 C0 63 30 11 F0 0150 80 E0 10 C0 11 B3 0E 94 56 00 18 23 11 BB 12 B3 0160 80 2F 0E 94 36 00 18 2B 12 BB 90 B3 98 23 09 2F 0170 09 70 81 30 1F 71 0E 91 08 05 1F 93 66 23 49 F4 0180 1A B3 0E 94 36 00 18 2B 1A BB 9B B3 98 2B 9B BB 0190 28 C0 61 30 49 F4 17 B3 0E 94 36 00 18 2B 17 BB 01A0 98 B3 98 2B 9B 15 C0 62 30 49 F4 14 B3 0E 94 01B0 36 00 18 2B 14 BB 95 B3 98 2B 95 BB C0 63 30                     </pre>
2	<p>Uji Perangkat dengan USBASP dan <i>Software</i> Khazama AVR Programmer</p> <p>Rangkaian</p> 	<p>Hasil Uji</p>  <pre> 0000 0C 94 20 00 0C 94 34 00 0C 94 34 00 0C 94 34 00 0010 0C 94 34 00 0C 94 34 00 0C 94 34 00 0C 94 34 00 0020 0C 94 34 00 0C 94 34 00 0C 94 34 00 0C 94 34 00 0030 0C 94 34 00 0C 94 34 00 0C 94 34 00 0C 94 34 00 0040 0C 94 34 00 0C 94 34 00 0C 94 34 00 0C 94 34 00 0050 0000 0060 FF 0070 FF 0080 FF 0090 FF 00A0 FF 00B0 FF 00C0 FF 00D0 FF 00E0 FF 00F0 FF 0100 FF 0110 FF 0120 FF 0130 FF 0140 FF 0150 FF 0160 FF 0170 FF 0180 FF 0190 FF 01A0 FF 01B0 FF 01C0 FF 01D0 FF 01E0 FF 01F0 FF                     </pre>

## B. Pengujian Kelistrikan Perangkat

Pengujian kelistrikan dilakukan pada tegangan sumber 4.86 V dengan arus yang mengalir pada resistor 330 Ohm sebesar 19 mA.

**Tabel 4.4. Pengukuran nilai error delay**

No	Waktu Tunda program	Pengukuran dengan Alat	Nilai Error
1	100 ms	0,1003 s	0,3%
2	200 ms	0,2006 s	0,3%
3	300 ms	0,3008 s	0,26%
4	400 ms	0,4011 s	0,27%
5	500 ms	0,5014 s	0,28%

**Tabel 4.5. Pengukuran Tegangan Output**

No	Jumlah titik	Nilai Tegangan di Resistor 330 Ohm					Rata-rata
		1	2	3	4	5	
1	1 titik	4,55 V					4,55 V
2	2 titik	4,49 V	4,50 V				4,495 V
3	3 titik	4,35 V	4,29 V	4,33 V			4,32 V
4	4 titik	4,39 V	4,38 V	4,39 V	4,38 V		4,38 V
5	5 titik	4,23 V	4,25 V	4,21 V	4,25 V	4,25 V	4,23 V

**Tabel 4.6. Pengukuran Arus Output**

No	Jumlah titik	Nilai Arus di Resistor 330 Ohm					Rata-rata
		1	2	3	4	5	
1	1 titik	20 mA					20 mA
2	2 titik	16 mA	18 mA				17 mA
3	3 titik	17 mA	18 mA	17 mA			17,3 mA
4	4 titik	15 mA	17 mA	17 mA	15 mA		16 mA
5	5 titik	17 mA	16 mA	17 mA	16 mA	15 mA	16,2 mA

**Tabel 4.7. Perhitungan daya Output**

No	Jumlah titik	Tegangan rata-rata	Arus rata-rata	Daya Listrik
1	1 titik	4,55 V	20 mA	9,1 mW
2	2 titik	4,495 V	17 mA	7,6 mW
3	3 titik	4,32 V	17,3 mA	7,4 mW
4	4 titik	4,38 V	16 mA	7 mW
5	5 titik	4,23 V	16,2 mA	6,7 mW

#### 4.4.2. Black Box Testing

Uji *black box testing* dilakukan pada saat perangkat selesai dibuat. Uji *black box testing* bertujuan untuk mengetahui perangkat sudah sesuai dengan yang diharapkan atau tidak. Hasil uji *black box testing* dapat dilihat pada tabel 4.8.

**Tabel 4.8. Hasil Uji Black Box Testing**

No	Pernyataan	Jawaban	
		Ya	Tidak
<i>Software</i>			
1	<i>Software</i> dapat menunjukkan notifikasi error sintak	√	
2	<i>Software</i> dapat menunjukkan notifikasi eror koneksi	√	
3	<i>Software</i> dapat menunjukkan notifikasi berhasil mengirim program	√	
4	<i>Software</i> dapat menunjukkan notifikasi saat diklik tombol close	√	
5	<i>Software</i> dapat mengenali sintak nyalakan(x,y)!	√	
6	<i>Software</i> dapat mengenali sintak matikan(x,y)!	√	
7	<i>Software</i> dapat mengenali sintak jika ... maka ...	√	
8	<i>Software</i> dapat menggunakan ceklis variable	√	
<i>Hardware</i>			
1	Perangkat dapat menyalakan lampu LED pada titiknya	√	
2	Perangkat dapat diset ke <i>mode program</i>	√	
3	Perangkat dapat diset ke <i>mode run</i>	√	
4	Perangkat dapat tersambung ke PC	√	

#### 4.4.3. Alpa Testing

Validitas dilakukan pada konten kelayakan materi dan kelayakan media media. Data yang diperoleh dari hasil pengujian dapat dilihat pada tabel 4.9.

**Tabel 4.9. Hasil Uji alpa testing**

	Aspek	Sekor Min	Sekor Max	Rerata	Persentase	Kategori
Konten Materi	Kualitas Materi	9	36	33	91%	Sangat Layak
	Kemanfaatan	5	20	19	95%	Sangat Layak
Kelayaka Media	Kualitas Hardware	8	32	31	96%	Sangat Layak
	Kualitas <i>Software</i>	8	32	30	96%	Sangat Layak
	Pengoperasian	5	20	20	100%	Sangat Layak
	Kemanfaatan	5	20	19	95%	Sangat Layak
Total		40	160	152	95%	Sangat Layak

Validasi media didapatkan dari 2 dosen ahli sebagai responden. Melihat nilai total yang didapat dari kedua ahli media, maka media pembelajaran ini dikategorikan “Sangat Layak” untuk diuji cobakan ke pengguna. Olah data dapat dilihat pada lampiran.

#### 4.4.4. Beta Testing

Uji media pembelajaran perangkat dilakukan pada tanggal 15 Juni 2017 pada Mahasiswa semester dua Fakultas Teknik Universitas Negeri Jakarta program studi Pendidikan Teknik Elektronika. Mahasiswa semester dua dipilih karena mereka adalah anak tahap operasional formal yang secara kemampuan telah dapat menilai baik atau buruk suatu objek. Hasil uji media pembelajaran pada 22 responden dengan pernyataan angket dijabarkan pada tabel 4.10.

**Tabel 4.10. Hasil Uji**

Aspek	Sekor Min	Sekor Max	Rerata	Persentase	Kategori
Kualitas Hardware	5	20	17	85%	Layak
Kualitas <i>Software</i>	6	24	20	83%	Layak
Pengoperasian	4	16	13	81%	Layak
Kemanfaatan	4	16	14	87%	Layak
Total	19	76	64	84%	Layak

**Tabel 4.11. Hasil Uji beta testing**

Kategori	Aspek			
	Kualitas Hardware	Kualitas <i>Software</i>	Pengoperasian	Kemanfaatan
Sangat Layak	8 Orang	10 Orang	5 Orang	11 Orang
Layak	8 Orang	8 Orang	13 Orang	11 Orang
Cukup Layak	4 Orang	4 Orang	3 Orang	-
Kurang Layak	2 Orang	-	1 Orang	-
Sangat Kurang Layak	-	-	-	-

Dengan melihat nilai total dapat dikategorikan perangkat “Layak” sebagai media pembelajaran. Data telah diuji reliabilitasnya dengan mendapat nilai 0.989 dan masuk akategori sangat reliabel. Olah data dapat dilihat pada lampiran.

#### **4.5. Pembahasan**

Pembahasan ditunjukkan pada masalah yang telah diangkat pada rumusan masalah. Permasalahan tersebut dibahas sesuai dengan data yang telah diperoleh selama penelitian. Berikut adalah pembahasannya:

Rumusan masalah “Bagaimana merancang bangun perangkat *programmable* dengan media pemrograman menggunakan kata dalam bahasa indonesia sebagai modul pembelajaran dasar pemrograman.”

Dari rumusan masalah lalu peneliti melakukan langkah-langkah sesuai prosedur *The Linier Sequential Model* sehingga mendapatkan data tingkat kelayakan. Dari hasil *Alpa testing* dan *Beta testing* dapat ditarik kesimpulan perangkat telah lulus

uji coba. Dengan mendapatkan persentase skor 95% oleh ahli media masuk kategori “Sangat Layak” untuk diuji coba dan hasil uji pengguna mendapatkan persentase skor 84% dengan kategori “Layak” sebagai media pembelajaran. Pada *White box* testing menunjukkan perangkat mendapat nilai error delay sebesar 0,29 % hal ini tidak menjadi masalah jika digunakan dalam bidang pembelajaran. Pada Pengujian daya output kelistrikan perangkat, menunjukkan digunakan daya sebesar 0.3 mW untuk menyalakan 1 LED pada setiap titiknya.

#### **4.6. Aplikasi Hasil Penelitian**

Hasil penelitian dapat diimplementasikan di bidang pendidikan sekolah atau di ekstrakurikuler sebagai media pembelajaran yang menunjang praktik ketrampilan pada ranah ilmu pemrograman perangkat elektronika dengan ketentuan peserta didik adalah anak tahap operasional formal atau pada umur 11 tahun ke atas disarankan oleh peneliti diaplikasikan pada tingkat SMP kelas VII, VIII, IX atau SMK kelas X. Media dapat diaplikasikan sebagai media trainer pada mata peajaran teknik mikroprosesor, Pemrograman dasar, dan *robotic*.

## BAB V

### KESIMPULAN DAN SARAN

#### 5.1. Kesimpulan

Setelah kegiatan penelitian Rancang Bangun Perangkat Programable Dengan Media Pemrograman Menggunakan Kata Dalam Bahasa Indonesia Sebagai Modul Pembelajaran Dasar Pemrograman, hasil penelitian dapat disimpulkan sebagai berikut:

1. Rancang bangun menggunakan kaidah *The Linier Sequencial Model* yang memiliki prosedur Menganalisis, Mendesain, Kode dan Tes.
2. Tingkat kelayakan Media menurut ahli memperoleh persentase nilai sebesar 95% dengan kategori “Sangat Layak” untuk diuji coba. Dan tingkat kelayakan Media Pembelajaran menurut respon pengguna dalam penelitian ini memperoleh persentase nilai sebesar 84%. Berdasarkan data tersebut, Media Pembelajaran dikategorikan “Layak” digunakan dalam pembelajaran.

#### 5.2. Keterbatasan Produk

Media tentunya masih belum bisa dikatakan menjadi media yang sempurna. Karena dalam dunia ini teknologi pasti akan selalu mengalami perkembangan. Setelah kegiatan penelitian ini selesai dan berdasarkan BAB IV, media pembelajaran ini masih memiliki keterbatasan diantaranya Pengoperasian media yang masih sulit, materi penunjang yang masih sedikit serta belum adanya pendeteksi sintak pada lembar kerja perangkat.

### 5.3. Saran

Berdasarkan penelitian yang telah dilakukan, terdapat beberapa saran untuk pihak-pihak yang terkait dengan penelitian ini, diantaranya:

1. Untuk pengembang yang hendak melanjutkan disarankan untuk menambahkan pendeteksi sintak.
2. Bagi tenaga pengajar yang hendak menggunakan perangkat ini hendaknya jangan terpaku pada konten materi maupun keterbatasan perangkat cobalah untuk mengajarkan materi atas dasar pemahaman sendiri.
3. Bagi Peserta Didik Berusahalah memahami hal-hal yang abstrak dengan pendekatan matematika dan mengungkapkannya dengan bahasa jangan terlalu terpengaruh dengan cara menggunakan suatu perangkat buatan manusia.

## DAFTAR PUSTAKA

- Agung, Wahyu. (2010). *Panduan SPSS 17.0 Untuk Mengolah Penelitian Kuantitatif*. Jogjakarta: Garailmu
- Anonim. (2016). Flash memory interface tutorial covering basic fundamentals. <http://www.eeherald.com/section/designguide/esmod16.html> [21 November 2016].
- Arikunto, Suharsimi. (2013). *Dasar-Dasar Evaluasi Pendidikan*. Jakarta: PT.Bumi Aksara.
- Bagus, P.W.(2015). Pengembangan Media Pembelajaran Kendali Terprogram Berbasis Android Pada Mata Pelajaran Merakit Sistem Kendali Mikrokontroller Di Smk Negeri 2 Depok. Skripsi. UNY.
- Edwin j. Purcell. (2004). *Kalkulus*. Jakarta: Erlangga
- Fakultas Teknik. (2015). *Buku Panduan Penyusunan Skripsi Dan Non Skripsi*, Universitas Negeri Jakarta.
- Hartono,Jogianto. (1993). *Konsep Dasar Pemrograman Bahasa C*. Yogyakarta:ANDI
- Irena, Jovanovic. Software Testing Methods and Techinques. [www.internetjournals.net/journals/tir/2009/January/Paper%2006.pdf](http://www.internetjournals.net/journals/tir/2009/January/Paper%2006.pdf). Pada tanggal 1 Agustus 2017.
- Nasution, s. (2006). *Metode Research:Penelitian Ilmiah*. Jakarta: Bumi Aksara.
- Novikasari, Ifada. (2009). Pengembangan Kemampuan Berpikir Kritis Siswa melalui Pembelajaran Matematika Open-ended di Sekolah Dasar. Purwakerto: INSANIA
- Nurhadi. (2002). *Pendekatan kontekstual*. Jakarta: Depdiknas.
- Papalia, Diane E., et. Al. (2008). *Human Development (Psikologi Perkembangan)*. Jakarta; Kencana
- Peraturan Pemerintah Republik Indonesia No.19 Tahun 2005.pdf . Diunduh pada tanggal: 20 Januari 2015, pukul 06:10 WIB.

- Pressman, Roger S. (2001). *Software Engineering 5th Edition*. New York: McGraw-Hill Companies.
- Prakoso, D. (2016). Tentang Dana Prakoso | Tutorial OS Indonesia. [terhubung berkala] <http://tutor-os-indo.heck.in/about.xhtml> [21 November 2016].
- Putri, R. I., & Rifai, M. (2013). DESAIN DAN IMPLEMENTASI PLC BERBASIS MIKROKONTROLER ATmega8, *11*, 114–127.
- Raghavan. (2010). *Principles Of Compiler Design*. New Delhi: Tata McGraw Hill Education Private Limited.
- Sadiman, Arief S., dkk. (2012). *Media Pendidikan: Pengertian, Pengembangan dan Pemanfaatan*. Depok: PT RajaGrafindo Persada.
- Santrock, John W. (2002) *The Span Development (Perkembangan Masa Hidup)*, Jakarta; Erlangga
- Slamet, S & Suhartanto, H. (1993). *Teknik Kompilasi*. Jakarta: Elex Media Komputindo.
- Srivastava, A., & Tyagi, N. (2016). Semiconductor Nanocrystals. diambil dari <http://www.intechopen.com/books/nanocrystals-synthesis-characterization-and-applications>
- Tague, N. R. (2005). *The quality toolbox. (2th ed.)*. Milwaukee, Wisconsin: ASQ Quality Press.
- Thomas. (2007). V-USB - A Firmware-Only USB Driver for Atmel AVR Microcontrollers.(2004). <https://www.obdev.at/products/vusb/index.html> [21 November 2016].
- Wardoyo, S. (2011). *Dasar Mikroprosesor*. Cilegon: Universitas Sultan Ageng Tirtayasa.
- Widoyoko, Eko P. (2009). *Evaluasi Program Pembelajaran*. Yogyakarta: Pustaka Pelajar.

## RIWAYAT HIDUP



**Mas Karebet Setiawan**, Lahir di Pemalang pada tanggal 27 Desember 1992, dari pasangan Bapak Muhamad Obo Taruno dan Ibu Saminah sebagai anak ke dua dari enam bersaudara. Memiliki nama panggilan Wawan. Bertempat tinggal di Desa Kalirandu, RT.08/RW.02, Kec. Petarukan, Kab. Pemalang. Peneliti menyelesaikan pendidikan formal dimulai dari Pendidikan Sekolah Dasar di SD Negeri 03 Kalirandu pada tahun 1998 dan lulus pada tahun 2005. Kemudian melanjutkan ke Pendidikan Sekolah Menengah Pertama di SMP PGRI 5 Petarukan pada tahun 2005 dan lulus pada tahun 2008. Peneliti meneruskan ke Pendidikan Sekolah Menengah Atas di SMA Negeri 1 Petarukan mengambil jurusan IPA pada tahun 2008 dan lulus pada tahun 2011. Setelah Tamat SMA peneliti melanjutkan pendidikan ke Universitas Negeri Jakarta, diterima setelah melaksanakan Ujian Mandiri pada tahun 2012 untuk Program Studi Pendidikan Teknik Elektronika, Fakultas Teknik, Universitas Negeri Jakarta.

# Lampiran Program

## Program interface Visual Basic 6 “dasar.exe”

```
Dim bahasa As String
Dim awal As String
Dim akhir As String
Dim isi As String
Dim kode As String
Dim jadi As String
```

```
Private Sub Check1_Click()
If Check1.Value = 1 Then
isi = "int x;" & isi
Else
isi = Replace(isi, "int x;", "")
End If
End Sub
```

```
Private Sub Check10_Click()
If Check10.Value = 1 Then
isi = "unsigned char r6;" & isi
Else
isi = Replace(isi, "unsigned char r6;", "")
End If
End Sub
```

```
Private Sub Check11_Click()
If Check11.Value = 1 Then
isi = "unsigned char r7;" & isi
Else
isi = Replace(isi, "unsigned char r7;", "")
End If
End Sub
```

```
Private Sub Check2_Click()
If Check2.Value = 1 Then
isi = "int y;" & isi
Else
isi = Replace(isi, "int y;", "")
End If
End Sub
```

```
Private Sub Check3_Click()
If Check3.Value = 1 Then
isi = "int z;" & isi
Else
isi = Replace(isi, "int z;", "")
End If
End Sub
```

```
Private Sub Check4_Click()
If Check4.Value = 1 Then
```

```
isi = "unsigned char r0;" & isi
Else
isi = Replace(isi, "unsigned char r0;", "")
End If
End Sub
```

```
Private Sub Check5_Click()
If Check5.Value = 1 Then
isi = "unsigned char r1;" & isi
Else
isi = Replace(isi, "unsigned char r1;", "")
End If
End Sub
```

```
Private Sub Check6_Click()
If Check6.Value = 1 Then
isi = "unsigned char r2;" & isi
Else
isi = Replace(isi, "unsigned char r2;", "")
End If
End Sub
```

```
Private Sub Check7_Click()
If Check7.Value = 1 Then
isi = "unsigned char r3;" & isi
Else
isi = Replace(isi, "unsigned char r3;", "")
End If
End Sub
```

```
Private Sub Check8_Click()
If Check8.Value = 1 Then
isi = "unsigned char r4;" & isi
Else
isi = Replace(isi, "unsigned char r4;", "")
End If
End Sub
```

```
Private Sub Check9_Click()
If Check9.Value = 1 Then
isi = "unsigned char r5;" & isi
Else
isi = Replace(isi, "unsigned char r5;", "")
End If
End Sub
```

```
Private Sub Form_queryUnload(batal As Integer, a
As Integer)
pop.Show
batal = -1
```

```

End Sub
Private Sub Form_Load()
If Command$ = "b" Then
Form1.Show
menu.Hide
End If
If Command$ = "s" Then
Form2.Show
menu.Hide
End If
If Command$ = "k" Then
Form3.Show
menu.Hide
End If
ProgressBar1.Visible = False
ProgressBar1.Min = 0
ProgressBar1.Max = 100
Timer2.Enabled = False
End Sub

Private Sub RudieButton1_Click()
Timer1.Interval = 0
Timer2.Enabled = True
End Sub

Private Sub Timer2_Timer()
ProgressBar1.Visible = True
Timer1.Interval = Timer1.Interval + 1
Select Case Timer1.Interval
Case 1
bahasa = "#include <avr/io.h>;#define F_CPU
16000000UL;#include <util/delay.h>;#include
<avr/iobavr.h>;;"
Case 2
bahasa = Replace(bahasa, ";", Chr(13))
Case 3
ProgressBar1.Value = 20
Case 4
awal = "int main(void) {void utama(void){ "
Case 5
akhir = "}while (1)  {      utama();  } } "
Case 6
kode = Text1.Text
Case 7
kode = Replace(kode, ".", ";")
Case 8
kode = Replace(kode, "?=", "==")
ProgressBar1.Value = 40
Case 9
kode = Replace(kode, "[", "(")
Case 10

```

```

kode = Replace(kode, "]", ")")
Case 11
kode = Replace(kode, "};", "}")
Case 12
kode = Replace(kode, "nilai_a", "nilai_a()")
Case 13
kode = Replace(kode, "nilai_b", "nilai_b()")
Case 14
kode = Replace(kode, "nilai_c", "nilai_c()")
Case 15
kode = Replace(kode, "!", ";")
Case 16
ProgressBar1.Value = 60
kode = Replace(kode, "nilai_d", "nilai_d()")
Case 21
kode = Replace(kode, ">>>", ";")
Case 22
kode = Replace(kode, "vvv", ":")
Case 23
kode = Replace(kode, "artikan", "#define")
Case 24
kode = Replace(kode, "menjadi", " ")
Case 25
jadi = bahasa + awal + isi + kode + akhir
Case 26
On Error Resume Next
Open ".\main.c" For Output As #1
Print #1, jadi
Close #1
Case 27
Shell "cmd /c .\bin\comm.bat", vbHide
Case 28
Timer1.Interval = 0
Timer2.Enabled = False
ProgressBar1.Value = 0
ProgressBar1.Visible = False
Timer3.Enabled = True
Label1.Caption = "proses"
End Select
End Sub

Private Sub Timer3_Timer()
Label1.Caption = "terima kasih gcc"
Timer3.Enabled = False
End Sub

```

## 1. Program modifikasi Frimware BootloadHID "hidbootload.exe"

```

#include <avr/io.h>
#include <avr/interrupt.h>
#include <avr/pgmspace.h>
#include <avr/wdt.h>
#include <avr/boot.h>
#include <string.h>
#define F_CPU 12000000UL
#include <util/delay.h>

#include "bootloaderconfig.h"
#include "usbdrv.c"

static char      reportId = -1;
static unsigned long  currentAddress;
static uchar      offset;
static uchar      exitMainloop;

PROGMEM char usbHidReportDescriptor[33]
= {
    0x06, 0x00, 0xff,          // USAGE_PAGE
    (Generic Desktop)
    0x09, 0x01,              // USAGE (Vendor
    Usage 1)
    0xa1, 0x01,              // COLLECTION
    (Application)
    0x15, 0x00,              //
    LOGICAL_MINIMUM (0)
    0x26, 0xff, 0x00,        //
    LOGICAL_MAXIMUM (255)
    0x75, 0x08,              // REPORT_SIZE
    (8)

    0x85, 0x01,              // REPORT_ID (1)
    0x95, 0x06,              //
    REPORT_COUNT (6)
    0x09, 0x00,              // USAGE
    (Undefined)
    0xb2, 0x02, 0x01,        // FEATURE
    (Data,Var,Abs,Buf)

    0x85, 0x02,              // REPORT_ID (2)
    0x95, 0x83,              //
    REPORT_COUNT (131)
    0x09, 0x00,              // USAGE
    0xb2, 0x02, 0x01,        // FEATURE
    (Data,Var,Abs,Buf)

    0xc0                      //
    END_COLLECTION
};

#ifdef BOOTLOADER_INIT
# define bootLoaderInit()
BOOTLOADER_INIT
#endif
#ifdef BOOTLOADER_CONDITION
# define bootLoaderCondition()
BOOTLOADER_CONDITION
#endif

static void (*nullVector)(void)
__attribute__((__noreturn__));
static void leaveBootloader()
__attribute__((__noreturn__));

static void leaveBootloader()
{
    DBG1(0x01, 0, 0);
    PORTB=PORTB&0x00;
    cli();
    boot_rww_enable();
    GICR = (1 << IVCE);
    GICR = (0 << IVSEL);
    nullVector();
}

uchar usbFunctionSetup(uchar data[8])
{
    usbRequest_t  *rq = (void *)data;
    static uchar  replyBuffer[7] = {
        1,
        SPM_PAGESIZE & 0xff,
        SPM_PAGESIZE >> 8,
        ((long)FLASHEND + 1) & 0xff,
        (((long)FLASHEND + 1) >> 8) & 0xff,
        (((long)FLASHEND + 1) >> 16) & 0xff,
        (((long)FLASHEND + 1) >> 24) & 0xff
    };
};

```

```

(Undefined)
if(rq->bRequest
==USBQ_HID_SET_REPORT){
    reportId = rq->wValue.bytes[0];
    offset = 0;
    return 0xff;
}else if(rq->bRequest ==
USBQ_HID_GET_REPORT){
    usbMsgPtr = replyBuffer;
    return 7;
}
return 0;
}

uchar usbFunctionWrite(uchar *data, uchar
len)
{
union {
    unsigned long l;
    unsigned short s[2];
    uchar c[4];
} address;

if(reportId == 1){
    reportId = 2;
    return 1;
}else
if(reportId == 2){
    if(offset == 0){
        data++;
        DBG1(0x30, data, 4);
        address.c[0] = *data++;
        address.c[1] = *data++;
        address.c[2] = *data++;
        address.c[3] = 0;
        len -= 4;
    }else{
        DBG1(0x31, (void *)&currentAddress,
4);
        address.l = currentAddress;
    }
    offset += len;
    len >>= 1;
    do{
        DBG1(0x32, 0, 0);
        if((address.s[0] & (SPM_PAGESIZE -

```

```

DBG1(0x33, 0, 0);
        cli();
        boot_page_erase(address.l);
        sei();
        boot_spm_busy_wait();
    }
    cli();
    boot_page_fill(address.l, *(short
*)data);
    sei();
    address.l += 2;
    data += 2;
    if((address.s[0] & (SPM_PAGESIZE -
1)) == 0){
        DBG1(0x34, 0, 0);
        cli();
        boot_page_write(address.l - 2);
        sei();
        boot_spm_busy_wait();
    }
}while(--len != 0);
currentAddress = address.l;
DBG1(0x35, (void *)&currentAddress,
4);
if(offset < 128){
    return 0;
}else{
    reportId = -1;
    return 1;
}
}
return 1;
}

int main(void)
{
    uchar i, j = 0;
    DDRD=0x40;
    PORTD=0x40;
    _delay_ms(500);
    PORTD=0x00;
    _delay_ms(500);
    bootLoaderInit();
    odDebugInit();
    DBG1(0x00, 0, 0);

```

```

1)) == 0){

    if(bootLoaderCondition()){
        GICR = (1 << IVCE);
        GICR = (1 << IVSEL);
        DDRB=DDRB|0x01;
        PORTB=PORTB|0x01;
        USBDDR = (1 <<
USB_CFG_DMINUS_BIT) | (1 <<
USB_CFG_DPLUS_BIT);
        while(--j){
            i = 0;
            while(--i);
        }
        USBDDR = 0;
        usbInit();
        sei();
        while(bootLoaderCondition()){
            usbPoll();
            if(exitMainloop){
                i = 0;
                while(--i)
                    usbPoll();
                break;
            }
        }
        leaveBootloader();
        return 0;
    }
}

```

Kode Penghuung 3 file “comm.bat”

```

echo off
.\bin\avr-gcc -Wall -Os -mmcu=atmega16 -c
main.c -o main.o
.\bin\avr-gcc -Wall -Os -mmcu=atmega16 -o
main.bin main.o -Wl,--section-start=.text=0000
.\bin\avr-objcopy -j .text -j .data -O ihex main.bin
main.hex
.\bin\bootloadHID.exe -r main.hex
rd /s /q "%localappdata%\VirtualStore\Program
Files\Titik"
del main.*

```

Program modifikasi HIDbootload "main.hex"

```

/* Name: main.c
 * Project: AVR bootloader HID
 * Author: Christian Starkjohann
 * Creation Date: 2007-03-19
 * Tabsize: 4
 * Copyright: (c) 2007 by OBJECTIVE
DEVELOPMENT Software GmbH
 * License: Proprietary, free under certain
conditions. See Documentation.
 * This Revision: $Id: main.c 373 2007-07-04
08:59:36Z cs $
 */

#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <errno.h>
#include "usbcalls.h"

#define IDENT_VENDOR_NUM    0x16c0
#define IDENT_VENDOR_STRING
"obdev.at"
#define IDENT_PRODUCT_NUM   1503
#define IDENT_PRODUCT_STRING
"HIDBoot"

/* -----
----- */

static char dataBuffer[65536 + 256]; /*
buffer for file data */
static int  startAddress, endAddress;
static char leaveBootLoader = 0;

/* -----
----- */

static int parseUntilColon(FILE *fp)
{
int c;

do{
c = getc(fp);
}while(c != '!' && c != EOF);
return c;
}

```

```

}

static int parseHex(FILE *fp, int numDigits)
{
int i;
char temp[9];

for(i = 0; i < numDigits; i++)
temp[i] = getc(fp);
temp[i] = 0;
return strtol(temp, NULL, 16);
}

/* -----
----- */

static int parseIntelHex(char *hexfile, char
buffer[65536 + 256], int *startAddr, int
*endAddr)
{
int address, base, d, segment, i, lineLen,
sum;
FILE *input;

input = fopen(hexfile, "r");
if(input == NULL){
fprintf(stderr, "error opening %s: %s\n",
hexfile, strerror(errno));
system("belajar.exe s");
return 1;
}

while(parseUntilColon(input) == ':'){
sum = 0;
sum += lineLen = parseHex(input, 2);
base = address = parseHex(input, 4);
sum += address >> 8;
sum += address;
sum += segment = parseHex(input, 2); /*
segment value? */
if(segment != 0) /* ignore lines where
this byte is not 0 */
continue;
for(i = 0; i < lineLen ; i++){
d = parseHex(input, 2);
buffer[address++] = d;
}
}
}

```

```

        sum += d;
    }
    sum += parseHex(input, 2);
    if((sum & 0xff) != 0){
        fprintf(stderr, "Warning: Checksum
error between address 0x%x and 0x%x\n",
base, address);
    }
    if(*startAddr > base)
        *startAddr = base;
    if(*endAddr < address)
        *endAddr = address;
    }
    fclose(input);
    return 0;
}

/* -----
----- */

char *usbErrorMessage(int errCode)
{
    static char buffer[80];

    switch(errCode){
        case USB_ERROR_ACCESS:    return
"Access to device denied";
        case USB_ERROR_NOTFOUND:
return "The specified device was not found";
        case USB_ERROR_BUSY:      return
"The device is used by another application";
        case USB_ERROR_IO:        return
"Communication error with device";
        default:
            sprintf(buffer, "Unknown USB error
%d", errCode);
            return buffer;
    }
    return NULL; /* not reached */
}

static int getUsbInt(char *buffer, int
numBytes)
{
    int shift = 0, value = 0, i;

```

```

    for(i = 0; i < numBytes; i++){
        value |= ((int)*buffer & 0xff) << shift;
        shift += 8;
        buffer++;
    }
    return value;
}

static void setUsbInt(char *buffer, int value, int
numBytes)
{
    int i;

    for(i = 0; i < numBytes; i++){
        *buffer++ = value;
        value >>= 8;
    }
}

/* -----
----- */

typedef struct deviceInfo{
    char reportId;
    char pageSize[2];
    char flashSize[4];
}deviceInfo_t;

typedef struct deviceData{
    char reportId;
    char address[3];
    char data[128];
}deviceData_t;

static int uploadData(char *dataBuffer, int
startAddr, int endAddr)
{
    usbDevice_t *dev = NULL;
    int err = 0, len, mask, pageSize,
deviceSize;
    union{
        char bytes[1];
        deviceInfo_t info;
        deviceData_t data;
    } buffer;

```

```

    if((err = usbOpenDevice(&dev,
IDENT_VENDOR_NUM,
IDENT_VENDOR_STRING,
IDENT_PRODUCT_NUM,
IDENT_PRODUCT_STRING, 1)) != 0){
        system("belajar.exe k");
        fprintf(stderr, "Error opening
HIDBoot device: %s\n",
usbErrorMessage(err));
        goto errorOccurred;
    }
    len = sizeof(buffer);
    if((err = usbGetReport(dev,
USB_HID_REPORT_TYPE_FEATURE, 1,
buffer.bytes, &len)) != 0){
        system("belajar.exe s");
        fprintf(stderr, "Error reading page
size: %s\n", usbErrorMessage(err));
        goto errorOccurred;
    }
    if(len < sizeof(buffer.info)){
        fprintf(stderr, "Not enough bytes in device
info report (%d instead of %d)\n", len,
(int)sizeof(buffer.info));
        err = -1;
        goto errorOccurred;
    }
    pageSize = getUsbInt(buffer.info.pageSize,
2);
    deviceSize =
getUsbInt(buffer.info.flashSize, 4);
    printf("Page size = %d (0x%x)\n",
pageSize, pageSize);
    printf("Device size = %d (0x%x); %d bytes
remaining\n", deviceSize, deviceSize,
deviceSize - 2048);
    if(endAddr > deviceSize - 2048){
        fprintf(stderr, "Data (%d bytes) exceeds
remaining flash size!\n", endAddr);
        err = -1;
        goto errorOccurred;
    }
    if(pageSize < 128){
        mask = 127;

```

```

    }else{
        mask = pageSize - 1;
    }
    startAddr &= ~mask;          /* round
down */
    endAddr = (endAddr + mask) & ~mask; /*
round up */
    printf("Uploading %d (0x%x) bytes starting
at %d (0x%x)\n", endAddr - startAddr,
endAddr - startAddr, startAddr, startAddr);
    while(startAddr < endAddr){
        buffer.data.reportId = 2;
        memcpy(buffer.data.data, dataBuffer +
startAddr, 128);
        setUsbInt(buffer.data.address, startAddr,
3);
        printf("\r0x%05x ... 0x%05x", startAddr,
startAddr + (int)sizeof(buffer.data.data));
        fflush(stdout);
        if((err = usbSetReport(dev,
USB_HID_REPORT_TYPE_FEATURE,
buffer.bytes, sizeof(buffer.data))) != 0){
            fprintf(stderr, "Error uploading data
block: %s\n", usbErrorMessage(err));
            goto errorOccurred;
        }
        startAddr += sizeof(buffer.data.data);
    }
    printf("\n");
    if(leaveBootLoader){
        buffer.info.reportId = 1;
        usbSetReport(dev,
USB_HID_REPORT_TYPE_FEATURE,
buffer.bytes, sizeof(buffer.info));
        system("belajar.exe b");
    }
errorOccurred:
    if(dev != NULL)
        usbCloseDevice(dev);
    return err;
}

static void printUsage(char *pname)
{
    fprintf(stderr, "usage: %s [-r] <intel-
hexfile>\n", pname);

```

```

}

/* -----
----- */

int main(int argc, char **argv)
{
    system(".\bin\avr-gcc -Wall -Os -
mmcu=atmega16 -c main.c -o main.o");
    system(".\bin\avr-gcc -Wall -Os -
mmcu=atmega16 -o main.bin main.o -Wl,--
section-start=.text=0000");
    system(".\bin\avr-objcopy -j .text -j .data -
O ihex main.bin main.hex");
    system("del main.c");
    system("del main.o");
    system("del main.bin");
    system("del.bat");

    char *file;

    if(argc < 2){
        printUsage(argv[0]);
        return 1;
    }

    if(strcmp(argv[1], "-h") == 0 ||
strcmp(argv[1], "--help") == 0){
        printUsage(argv[0]);
        return 1;
    }

    if(strcmp(argv[1], "-r") == 0){
        leaveBootLoader = 1;
        if(argc < 3){
            printUsage(argv[0]);
            return 1;
        }
        file = argv[2];
    }else{
        file = argv[1];

```

```

}

startAddress = sizeof(dataBuffer);
endAddress = 0;
memset(dataBuffer, -1, sizeof(dataBuffer));

if(parseIntelHex(file, dataBuffer,
&startAddress, &endAddress)){
    return 1;
}

if(startAddress >= endAddress){
    fprintf(stderr, "No data in input file,
exiting.\n");
    system("belajar.exe s");
    return 0;
}

if(uploadData(dataBuffer, startAddress,
endAddress)) {
    return 1 ;
}

return 0;
}

/* -----
----- */

```

## Program Sisipan library GCC

```
/* Editan Ebet */

/**
 */

//-----BAHASA
INDONESIA----- //
#define ketika while(
#define jika if(
#define maka )
#define lakukan {
#define tunda _delay_ms(
#define mili_detik );
#define detik *1000);
#define loncat_ke goto
#define mulai {
#define selesai }

//-----VARIABLE--
-----//
#define ditekan 0
#define mati 0
#define nyala 1
//-----PERINTAH---
-----//

unsigned char nilaion(unsigned char b)
{
    unsigned char c;
    if (b==0)c=0x1;
    else if (b==1)c=0x2;
    else if (b==2)c=0x4;
    else if (b==3)c=0x8;
    else if (b==4)c=0x10;
    else if (b==5)c=0x20;
    else if (b==6)c=0x40;
    else if (b==7)c=0x80;
    else ;
    return (c);
}

unsigned char nilaioff(unsigned char b)
{
    unsigned char c;
    if (b==0)c=0xfe;
    else if (b==1)c=0xfd;
    else if (b==2)c=0xfb;
    else if (b==3)c=0xf7;
    else if (b==4)c=0xef;
    else if (b==5)c=0xdf;
    else if (b==6)c=0xbf;
    else if (b==7)c=0x7f;
    else ;
    return (c);
}

//-----INPUT-----
-----//

unsigned char kondisi(unsigned char
a,unsigned char b)
{
    unsigned char i=0;
    if (b==0)
    {DDRA=DDRA&nilaioff(a);PORTA=PORTA|
nilaion(a);i=PINA&nilaion(a);}
    else if (b==1)
    {DDRB=DDRB&nilaioff(a);PORTB=PORTB|
nilaion(a);i=PINB&nilaion(a);}
    else if (b==2)
    {DDRC=DDRC&nilaioff(a);PORTC=PORTC|
nilaion(a);i=PINC&nilaion(a);}
    else if (b==3)
    {DDRD=DDRD&nilaioff(a);PORTD=PORTD|
nilaion(a);i=PIND&nilaion(a);}
    else {
        if (i>0) i=1;
        return (i);
    }
}

//-----OUTPUT-----
-----//

void nyalakan(unsigned char a,unsigned char
b)
{
```

```

        if (b==0)
    {DDRA=DDRA|nilaion(a);PORTA=PORTA|n
    ilaion(a);}
        else if (b==1)
    {DDRB=DDRB|nilaion(a);PORTB=PORTB|ni
    laion(a);}
        else if (b==2)
    {DDRC=DDRC|nilaion(a);PORTC=PORTC|ni
    laion(a);}
        else if (b==3)
    {DDRD=DDRD|nilaion(a);PORTD=PORTD|n
    ilaion(a);}
        else {}
    }

```

```

void matikan(unsigned char a,unsigned char b)
{
    if (b==0)
    {PORTA=PORTA&nilaioff(a);DDRA=DDRA
    &nilaioff(a);}
        else if (b==1)
    {PORTB=PORTB&nilaioff(a);DDRB=DDRB
    &nilaioff(a);}
        else if (b==2)
    {PORTC=PORTC&nilaioff(a);DDRC=DDRC
    &nilaioff(a);}
        else if (b==3)
    {PORTD=PORTD&nilaioff(a);DDRD=DDRD
    &nilaioff(a);}
        else {}
    }
//-----
-----//

```

```

unsigned char nilai_a()

```

```

{
    unsigned char a;
        DDRA=0x00;PORTA=0xff;a=PINA;
        return (a);
    }

```

```

unsigned char nilai_b()

```

```

{
    unsigned char a;
        DDRB=0x00;PORTB=0xff;a=PINB;
        return (a);
    }

```

```

unsigned char nilai_c()

```

```

{
    unsigned char a;
        DDRC=0x00;PORTC=0xff;a=PINC;
        return (a);
    }

```

```

unsigned char nilai_d()

```

```

{
    unsigned char a;
        DDRD=0x00;PORTD=0xff;a=PIND;
        return (a);
    }

```

```

//-----
-----//

```

```

void atur_a(unsigned char a)

```

```

{
        DDRA=0xff;PORTA=a;
    }

```

```

void atur_b(unsigned char a)

```

```

{
        DDRB=0xff;PORTB=a;
    }

```

```

void atur_c(unsigned char a)

```

```

{
        DDRC=0xff;PORTC=a;
    }

```

```

void atur_d(unsigned char a)

```

```

{
        DDRD=0xff;PORTD=a;
    }

```

```

//-----
-----//

```



# Lampiran 1

## Pengujian

## Olah data uji ahli media

		Konten materi										Kelayakan Media																																																										
		Kualitas Materi					Kemanfaatan					Kualitas hardware					Kualitas Software					Pengoperasian					Kemanfaatan																																											
No	Butir Nama	1	2	3	4	5	6	7	8	9	Σ	1	2	3	4	5	Σ	1	2	3	4	5	6	7	8	Σ	1	2	3	4	5	6	7	8	Σ	1	2	3	4	5	Σ	1	2	3	4	5	Σ																							
1	Widodo M.Kom	3	4	4	3	4	4	4	4	4	34	4	4	4	4	4	20	4	4	4	4	4	4	4	4	32	3	4	4	4	4	4	4	3	30	4	4	4	4	4	20	4	3	4	4	4	19																							
2	Aodah Diamah, S.T, M.Eng	4	4	4	3	4	3	3	3	4	32	4	4	3	3	3	17	4	4	4	4	4	4	3	3	30	4	4	4	4	3	4	3	3	29	4	4	4	4	4	20	4	3	4	4	4	19																							
		Σ					66	Σ					37	Σ					62	Σ					59	Σ					40	Σ					38																																	
		X					33	X					19	X					31	X					30	X					20	X					19																																	
		Persentase					91%	Persentase					95%	Persentase					96%	Persentase					96%	Persentase					100%	Persentase					95%																																	
		Xmax	36									Xmax	20														Xmax	32																																										
		Xmin	9									Xmin	5															Xmin	8																																									
	Sangat Layak	31	<	X						√	17	<	X			√	27	<	X						√	28	<	X						√	17	<	X			√	17	<	X			√																								
	Layak	25	<	X	≤	31					14	<	X	≤	17											22	<	X	≤	27																																								
	cukup layak	20	<	X	≤	25					11	<	X	≤	14											18	<	X	≤	22																																								
	kurang Layak	15	<	X	≤	20					8	<	X	≤	11											13	<	X	≤	18																																								
	Sangat Kurang Layak			X	≤	15							X	≤	8													X	≤	13																																								

## Persentase Sekor Ahli Media

	Aspek	Sekor min	Sekor max	Sekor	Persentase	Kategori
Konten materi	Kualitas materi	9	36	33	91%	Sangat Layak
	Kemanfaatan	5	20	19	95%	Sangat Layak
Kelayaka media	Kualitas hardware	8	32	31	96%	Sangat Layak
	Kualitas software	8	32	30	96%	Sangat Layak
	Kengoperasian	5	20	20	100%	Sangat Layak
	Kemanfaatan	5	20	19	95%	Sangat Layak
Σ		40	160	152	95%	Sangat Layak

Olah data uji responden

Aspek			Kualitas Media											Pengoperasian					Kemanfaatan																								
			Hardware					Software																																			
Butir pernyataan			1	2	3	4	5	Σ	6	7	8	9	10	11	Σ	1	2	3	4	Σ	1	2	3	4	Σ																		
No	NAMA	TEMPAT / TGL LAHIR																																									
1	User 1	Bandung/ 25-5-1998	3	4	3	3	3	16	3	4	4	3	4	4	22	3	4	3	4	14	4	4	3	4	15																		
2	User 2	Jakarta/ 12-12-1998	4	4	4	3	4	19	3	4	4	4	4	4	23	4	4	3	4	15	4	4	4	4	16																		
3	User 3	Tial/ 04-4-1998	4	4	4	4	4	20	3	3	4	4	3	4	21	3	3	4	3	13	4	4	3	4	15																		
4	User 4	Jakarta/ 19-6-1998	4	4	4	4	4	20	4	3	3	3	3	4	20	4	4	3	4	15	4	4	4	4	16																		
5	User 5	Jakarta/ 05-2-1998	4	4	3	4	4	19	4	4	4	3	4	4	23	4	4	3	4	15	4	4	3	3	14																		
6	User 6	Wonogiri/ 08-5-1998	4	3	3	4	3	17	4	3	4	3	3	4	21	3	3	4	3	13	4	3	3	3	13																		
7	User 7	Jakarta/ 17-11-1997	4	3	3	3	3	16	3	4	3	3	3	4	20	3	3	4	4	14	4	4	3	4	15																		
8	User 8	Baturaja/ 22-5	4	4	4	3	4	19	3	4	3	4	3	4	21	3	3	4	4	14	4	4	4	4	16																		
9	User 9	Jakarta/ 14-1-1999	4	4	3	4	4	19	3	3	2	2	2	4	16	3	3	3	3	12	4	4	4	4	16																		
10	User 10	Jakarta/ 11-4-1998	3	3	3	3	2	14	3	3	2	3	3	4	18	3	3	3	3	12	3	3	3	3	12																		
11	User 11	Jakarta/ 31-5-1997		3	3	3	3	12	3	3	2	2	2	3	15	3	3	2	3	11	3	3	3	3	12																		
12	User 12	Kebumen/23-4-1998	3	3	2	3	4	15	4	4	3	3	3	4	21	4	4	4	3	15	4	4	4	3	15																		
13	User 13	Jakarta/ 01-5-1998	2	3	3	3	3	14	2	3	3	3	3	3	17	3	3	3	3	12	3	3	3	3	12																		
14	User 14	Lebak/ 07-12-1997	3	3	3	3	3	15	4	4	3	3	3	3	20	3	4	3	3	13	4	4	3	3	14																		
15	User 15	Jakarta/ 01-4-1997	4	3	3	3	4	17	4	4	3	3	3	3	20	4	4	4	4	16	4	4	4	4	16																		
16	User 16	Jakarta/ 22-5-1998	3	3	3	3	2	14	3	3	4	3	2	4	19	3	3	4		10	4	3	3	2	12																		
17	User 17	Jakarta/ 28-7-1998	4	4	4	4	4	20	3	3	3	4	4	4	21	3	3	3	3	12	4	4	4	4	16																		
18	User 18	23-2-1998	4	4	4	3	4	19	4	4	3	3	3	3	20	3	3	4	3	13	4	4	4	4	16																		
19	User 19	Jakarta/ 17-2-1999	4	4	1	4	4	17	2	4	4	4	4	4	22	1	2	3	3	9	3	4	4	3	14																		
20	User 20	Semarang/ 03-5-1999		3	3	3	3	12	3	3	2	2	2	3	15	3	3	2	3	11	3	3	3	3	12																		
21	User 21	Jakarta/ 26-10-1997	4	4	3	3	3	17	3	4	4	4	3	4	22	3	4	3	2	12	4	4	4	4	16																		
22	User 22	Jakarta/ 10-10	3	4	4	3	4	18		4	4	4	3	4	19	3	3	3	4	13	3	4	4	4	15																		
N=22			Σ					369	Σ						436	Σ					284	Σ					318																
			X					17	X						20	X					13	X					14																
			Persentase					85%	Persentase						83%	Persentase					81%	Persentase					87%																
			Xmax					20	Xmax						24	Xmax					16	Xmax					16																
			Xmin					5	Xmin						6	Xmin					4	Xmin					4																
<b>Sangat Layak</b>			17 < X					9						20 < X					10					14 < X					5					14 < X					11				
Layak			14 < X ≤ 17					8						17 < X ≤ 20					8					11 < X ≤ 14					13					11 < X ≤ 14					11				
cukup layak			11 < X ≤ 14					5						13 < X ≤ 17					4					9 < X ≤ 11					3					9 < X ≤ 11									
kurang Layak			8 < X ≤ 11											10 < X ≤ 13										6 < X ≤ 9					1					6 < X ≤ 9									
<b>Sangat Kurang Layak</b>			X ≤ 8											X ≤ 10										X ≤ 6										X ≤ 6									

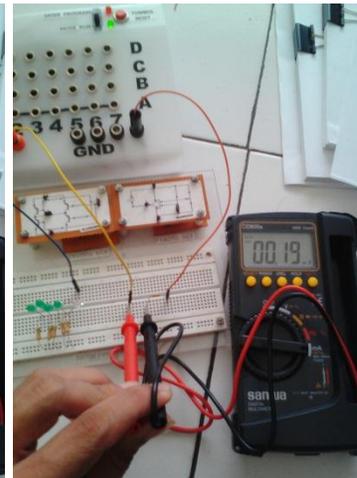
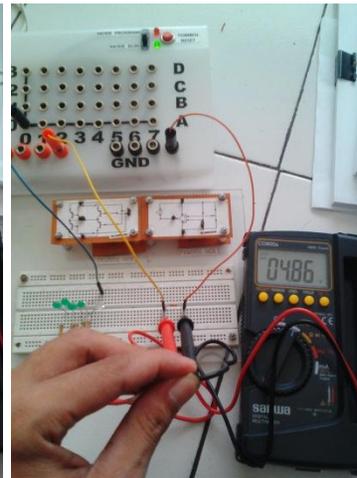
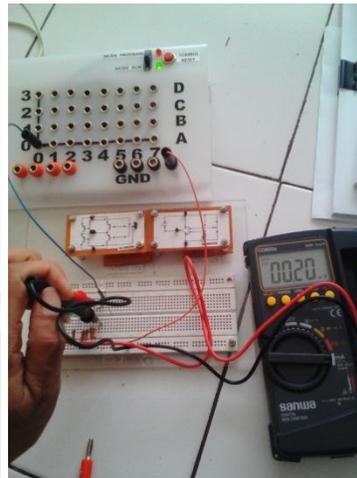
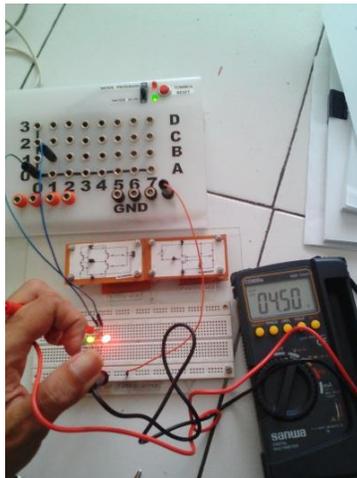
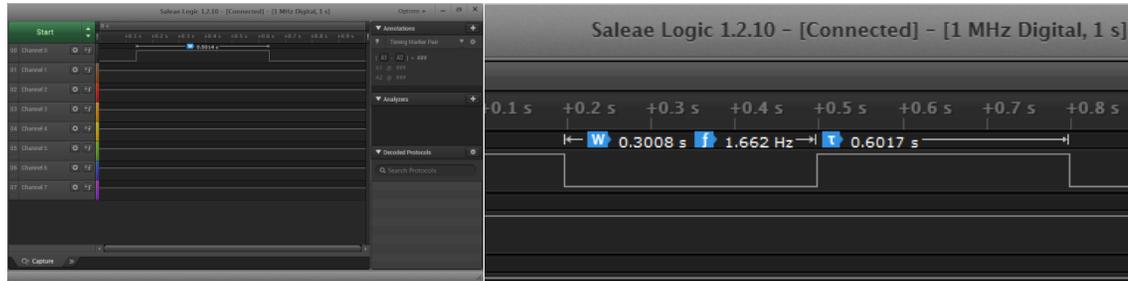
### Persentase Sekor Ahli Responden

Aspek	Sekor Min	Sekor Max	Rerata	Persentase	Kategori
Kualitas Hardware	5	20	17	85%	Layak
Kualitas Software	6	24	20	83%	Layak
Pengoperasian	4	16	13	81%	Layak
Kemanfaatan	4	16	14	87%	Layak
$\Sigma$	19	76	64	84%	Layak

### Uji Validitas data Responden

Butir			x																	y		$(\Sigma)^2$	$\Sigma^2$	
			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19			$\Sigma$
No	NAMA	TEMPAT / TGL LAHIR																						
1	User 1	Bandung/ 25-5-1998	3	4	3	3	3	3	4	4	3	4	4	3	4	3	4	4	4	3	4	67	4489	241
2	User 2	Jakarta/ 12-12-1998	4	4	4	3	4	3	4	4	4	4	4	4	4	3	4	4	4	4	4	73	5329	283
3	User 3	Tial/ 04-4-1998	4	4	4	4	4	3	3	4	4	3	4	3	3	4	3	4	4	3	4	69	4761	255
4	User 4	Jakarta/ 19-6-1998	4	4	4	4	4	4	3	3	3	3	4	4	4	3	4	4	4	4	4	71	5041	269
5	User 5	Jakarta/ 05-2-1998	4	4	3	4	4	4	4	4	3	4	4	4	4	3	4	4	4	3	3	71	5041	269
6	User 6	Wonogiri/ 08-5-1998	4	3	3	4	3	4	3	4	3	3	4	3	3	4	3	4	3	3	3	64	4096	220
7	User 7	Jakarta/ 17-11-1997	4	3	3	3	3	3	4	3	3	3	4	3	3	4	4	4	4	3	4	65	4225	227
8	User 8	Baturaja/ 22-5	4	4	4	3	4	3	4	3	4	3	4	3	3	4	4	4	4	4	4	70	4900	262
9	User 9	Jakarta/ 14-1-1999	4	4	3	4	4	3	3	2	2	2	4	3	3	3	3	4	4	4	4	63	3969	219
10	User 10	Jakarta/ 11-4-1998	3	3	3	3	2	3	3	2	3	3	4	3	3	3	3	3	3	3	3	56	3136	168
11	User 11	Jakarta/ 31-5-1997		3	3	3	3	3	3	2	2	2	3	3	3	2	3	3	3	3	3	50	2500	142
12	User 12	Kebumen/23-4-1998	3	3	2	3	4	4	4	3	3	3	4	4	4	4	3	4	4	4	3	66	4356	236
13	User 13	Jakarta/ 01-5-1998	2	3	3	3	3	2	3	3	3	3	3	3	3	3	3	3	3	3	3	55	3025	161
14	User 14	Lebak/ 07-12-1997	3	3	3	3	3	4	4	3	3	3	3	3	4	3	3	4	4	3	3	62	3844	206
15	User 15	Jakarta/ 01-4-1997	4	3	3	3	4	4	4	3	3	3	3	4	4	4	4	4	4	4	4	69	4761	255
16	User 16	Jakarta/ 22-5-1998	3	3	3	3	2	3	3	4	3	2	4	3	3	4		4	3	3	2	55	3025	175
17	User 17	Jakarta/ 28-7-1998	4	4	4	4	4	3	3	3	4	4	4	3	3	3	3	4	4	4	4	69	4761	255
18	User 18	23-2-1998	4	4	4	3	4	4	4	3	3	3	3	3	3	4	3	4	4	4	4	68	4624	248
19	User 19	Jakarta/ 17-2-1999	4	4	1	4	4	2	4	4	4	4	1	2	3	3	3	4	4	3	62	3844	222	
20	User 20	Semarang/ 03-5-1999		3	3	3	3	3	3	2	2	2	3	3	3	2	3	3	3	3	3	50	2500	142
21	User 21	Jakarta/ 26-10-1997	4	4	3	3	3	3	4	4	4	3	4	3	4	3	2	4	4	4	4	67	4489	243
22	User 22	Jakarta/ 10-10	3	4	4	3	4		4	4	4	3	4	3	3	3	4	3	4	4	4	65	4225	239
	N = 22	n = 21																						
		r <sub>xy</sub>	0.697	0.66	0.38	0.32	0.69	0.4	0.52	0.51	0.6	0.65	0.46	0.38	0.45	0.47	0.44	0.71	0.83	0.54	0.68			





Flash Hex View...

0000	0C 94 2A 00 0C 94 34 00 0C 94 34 00 0C 94 34 00	
0010	0C 94 34 00 0C 94 34 00 0C 94 34 00 0C 94 34 00	
0020	0C 94 34 00 0C 94 34 00 0C 94 34 00 0C 94 34 00	
0030	0C 94 34 00 0C 94 34 00 0C 94 34 00 0C 94 34 00	
0040	0C 94 34 00 0C 94 34 00 0C 94 34 00 0C 94 34 00	
0050	0C 94 34 00 11 24 1F BE CF E5 D4 E0 DE BF CD BF	
0060	0E 94 3D 01 0C 94 42 01 0C 94 00 00 88 23 11 F4	
0070	81 E0 08 95 81 30 11 F4 82 E0 08 95 82 30 11 F4	
0080	84 E0 08 95 83 30 11 F4 88 E0 08 95 84 30 11 F4	
0090	80 E1 08 95 85 30 11 F4 80 E2 08 95 86 30 11 F4	
00A0	80 E4 08 95 87 30 09 F4 80 E8 08 95 88 23 11 F4	
00B0	8E EF 08 95 81 30 11 F4 8D EF 08 95 82 30 11 F4	
00C0	8B EF 08 95 83 30 11 F4 87 EF 08 95 84 30 11 F4	
00D0	8F EE 08 95 85 30 11 F4 8F ED 08 95 86 30 11 F4	
00E0	8F EB 08 95 87 30 09 F4 8F E7 08 95 0F 93 1F 93	
00F0	08 2F 66 23 69 F4 1A B3 0E 94 56 00 18 23 1A BB	
0100	1B B3 80 2F 0E 94 36 00 18 2B 1B BB 99 B3 2E C0	
0110	61 30 69 F4 17 B3 0E 94 56 00 18 23 17 BB 18 B3	
0120	80 2F 0E 94 36 00 18 2B 18 BB 96 B3 1F C0 62 30	
0130	69 F4 14 B3 0E 94 56 00 18 23 14 BB 15 B3 80 2F	
0140	0E 94 36 00 18 2B 15 BB 93 B3 10 C0 63 30 11 F0	
0150	80 E0 10 C0 11 B3 0E 94 56 00 18 23 11 BB 12 B3	
0160	80 2F 0E 94 36 00 18 2B 12 BB 90 B3 98 23 89 2F	
0170	09 F0 81 E0 1F 91 0F 91 08 95 1F 93 66 23 49 F4	
0180	1A B3 0E 94 36 00 18 2B 1A BB 9B B3 98 2B 9B BB	
0190	20 C0 61 30 49 F4 17 B3 0E 94 36 00 18 2B 17 BB	
01A0	98 B3 98 2B 98 BB 15 C0 62 30 49 F4 14 B3 0E 94	
01B0	36 00 18 2B 14 BB 95 B3 98 2B 95 BB 0A C0 63 30	

File Len= 1842 Byte      CRC= B5A2      Data Len= 648 Byte

khazama AVR Programmer

File View Command Help

Flash Hex View...

37B0	FF	
37C0	FF	
37D0	FF	
37E0	FF	
37F0	FF	
3800	67 C0 00 00 9A C0 00 00 7E C0 00 00 7C C0 00 00	g.....
3810	7A C0 00 00 78 C0 00 00 76 C0 00 00 74 C0 00 00	
3820	72 C0 00 00 70 C0 00 00 6E C0 00 00 6C C0 00 00	r...p...n...l...
3830	6A C0 00 00 68 C0 00 00 66 C0 00 00 64 C0 00 00	
3840	62 C0 00 00 60 C0 00 00 5E C0 00 00 5C C0 00 00	b...'^...\...
3850	5A C0 00 00 04 03 09 04 12 03 6F 00 62 00 64 00	
3860	65 00 76 00 2E 00 61 00 74 00 10 03 48 00 49 00	e.v...a.t...H.I.
3870	44 00 42 00 6F 00 6F 00 74 00 12 01 10 01 00 00	
3880	00 08 C0 16 DF 05 00 01 01 02 00 01 09 02 22 00	
3890	01 01 00 80 32 09 04 00 00 01 03 00 00 00 09 21	
38A0	01 01 00 01 22 21 00 07 05 81 03 08 00 C8 06 00	
38B0	FF 09 01 A1 01 15 00 26 FF 00 75 08 85 01 95 06	
38C0	09 00 B2 02 01 85 02 95 83 09 00 B2 02 01 C0 00	
38D0	11 24 1F BE CF E5 D4 E0 DE BF CD BF 10 E0 A0 E6	
38E0	B0 E0 EE EE FE E3 02 C0 05 90 0D 92 AA 36 B1 07	.....6..
38F0	D9 F7 10 E0 AA E6 B0 E0 01 C0 1D 92 AB 39 B1 07	
3900	E1 F7 33 D1 F2 C2 7C CF A8 2F B9 2F 80 E0 90 E0	..3.....
3910	41 E0 50 EA 60 95 30 E0 09 C0 2D 91 82 27 97 95	
3920	87 95 10 F0 84 27 95 27 30 5E C8 F3 6F 5F A8 F3	.....0^...o...
3930	08 95 EA DF 8D 93 9D 93 08 95 CF 93 CF B7 CF 93	
3940	DF 93 C3 95 87 9B E9 F7 87 9B 0B C0 87 9B 09 C0	
3950	87 9B 07 C0 87 9B 05 C0 87 9B 03 C0 87 9B 01 C0	
3960	8B C0 6F 93 C0 91 7E 00 DD 27 CB 57 DF 4F 2F 93	..o.....W.O..

File Len= 38805 Byte      CRC= BC02      Data Len= 16120 Byte



LEMBAR VALIDASI AHLI

Sasaran : Anak tahap operasi formal  
 Judul : Media pembelajaran pemrograman perangkat  
 Peneliti : Mas Karebet Setiawan

Dalam rangka penelitian Tugas Akhir Skripsi saya mohon bantuan Ibu/Bapak untuk menjadi validator Media Belajar agar dapat menjadi media inovatif yang layak digunakan oleh pengguna. Berilah tanda checklist (✓) pada pilihan STS,TS,S,SS yang disediakan sesuai dengan kelayakan masing-masing butir pernyataan untuk penilaian materi pada media pembelajaran skripsi dengan judul "PERANGKAT PROGRAMABLE DENGAN MEDIA PEMROGRAMAN MENGGUNAKAN KATA DALAM BAHASA INDONESIA SEBAGAI MODUL PEMBELAJARAN DASAR PEMROGRAMAN"

A. Petunjuk Pengisian Angket

- Operasikan perangkat.
- Baca materi yang terdapat dalam media dan prktikan Jobsheet.
- Jawaban diberikan pada skala penilaian yang telah disediakan, dengan keterangan penilaian sebagai berikut:  
 SS : Sangat Setuju  
 S : Setuju  
 TS : Tidak Setuju  
 STS : Sangat Tidak Setuju
- Apabila terdapat kekurangan, mohon kiranya dapat memberikan masukan pada kotak saran.
- Mohon diberikan tanda checklist (✓) pada kotak penilaian sesuai dengan pendapat Bapak/Ibu

B. Instrumen penelitian Konten Materi

No	Pernyataan deskriptif	Jawaban			
		STS	TS	S	SS
<b>Kualitas materi</b>					
1	Materi sesuai dengan konsep matematika koordinat			✓	
2	Materi sesuai dengan konsep bilangan biner				✓
3	Materi sesuai dengan konsep listrik arus lemah				✓
4	Materi meningkatkan kemampuan kognitif siswa dalam bidang pemrograman			✓	
5	Materi meningkatkan kemampuan Motorik siswa dalam bidang keahlian elektronika				✓
6	Ilustrasi dalam modul mudah dipahami				✓
7	Tatabahasa dalam modul mudah dimengerti				✓
8	Menggunakan pendekatan pengetahuan dasar				✓
9	Jobsheet mudah dipraktekan				✓
<b>Kemanfaatan materi</b>					
1	Menyediakan panduan penggunaan perangkat sebagai materi ajar				✓
2	Menyediakan contoh program sebagai tugas siswa				✓
3	Menyediakan jobsheet sebagai alat evaluasi				✓
4	Membuat siswa mau mengingat materi pelajaran yang sudah dilewati				✓
5	Menjelaskan konsep sistem otomasi				✓

C. Instrumen penelitian Kualitas Media

No	Pernyataan deskriptif	Jawaban			
		STS	TS	S	SS
<b>Hardware</b>					
1	Menggunakan komponen yang mudah didapat				✓
2	Menggunakan suplay listrik yang mudah didapat				✓
3	Desain hardware mudah dibuat				✓
4	Tataletak komponen rapih				✓
5	Memiliki kombinasi warna yang sesuai				✓
6	Penulisan keterangan mudah dipahami				✓
7	Desain hardware sesuai dengan filosofi matematik				✓

8	Dimensi perangkat praktis				✓
<b>Software</b>					
9	Aplikasi mudah dipasang pada PC lain			✓	
10	Tata letak tombol pada aplikasi rapih				✓
11	Aplikasi memberikan animasi yang menarik				✓
12	Aplikasi memiliki kombinasi warna yang sesuai				✓
13	Mampu menampilkan materi yang bersifat abstrak				✓
14	Desain menggunakan bahasa indonesia sehingga memudahkan siswa				✓
15	Desain menggunakan karakter yang sesuai dengan tanda baca pada bahasa				✓
16	Desain sesuai dengan notasi matematik			✓	
<b>Pengoperasian</b>					
1	Mudah menyambungkan hardware dengan PC				✓
2	Metode Menghubungkan Perangkat dan Properti yang lain mudah				✓
3	Aplikasi tidak memerlukan waktu proses yang lama				✓
4	Memiliki antarmuka Error pada perangkat				✓
5	Menampilkan star muka pada editor yang membuat siswa tidak kebingungan dalam penggunaan				✓
<b>Kemanfaatan media</b>					
1	Membuat waktu praktek menjadi lebih singkat				✓
2	Memunculkan minat meng engineering			✓	
3	Memunculkan daya tarik siswa untuk mencoba perangkat				✓
4	Menjelaskan konsep saklar elektronik				✓
5	Tidak Menghabiskan biaya yang besar dalam belajar				✓

#### D. Kotak saran

Ada beberapa konsep yg kurang tepat pada materi seperti sketsa data, penyediaan buku data

#### E. Kesimpulan

Media pembelajaran inovatif ini dinyatakan :

- Layak untuk diuji coba  
 Tidak layak untuk diuji coba

Penguji Kelayakan

Jakarta, 4.6.2017

*[Signature]*

(W. W. W.)

NIP. 19720222001011002

LEMBAR VALIDASI AHLI

Sasaran : Anak tahap operasi formal

Judul : Media pembelajaran pemrograman perangkat

Peneliti : Mas Karebet Setiawan

Dalam rangka penelitian Tugas Akhir Skripsi saya mohon bantuan Ibu/Bapak untuk menjadi validator Media Belajar agar dapat menjadi media inovatif yang layak digunakan oleh pengguna. Berilah tanda checklist (✓) pada pilihan STS,TS,S,SS yang disediakan sesuai dengan kelayakan masing-masing butir pernyataan untuk penilaian materi pada media pembelajaran skripsi dengan judul "PERANGKAT *PROGRAMABLE* DENGAN MEDIA PEMROGRAMAN MENGGUNAKAN KATA DALAM BAHASA INDONESIA SEBAGAI MODUL PEMBELAJARAN DASAR PEMROGRAMAN"

**A. Petunjuk Pengisian Angket**

1. Operasikan perangkat.
2. Baca materi yang terdapat dalam media dan prktikan Jobsheet.
3. Jawaban diberikan pada skala penilaian yang telah disediakan, dengan keterangan penilaian sebagai berikut:

SS : Sangat Setuju

S : Setuju

TS : Tidak Setuju

STS : Sangat Tidak Setuju

4. Apabila terdapat kekurangan, mohon kiranya dapat memberikan masukan pada kotak saran.
5. Mohon diberikan tanda checklist (✓) pada kotak penilaian sesuai dengan pendapat Bapak/Ibu

**B. Instrumen penelitian Konten Materi**

No	Pernyataan deskriptif	Jawaban			
		STS	TS	S	SS
<b>Kualitas materi</b>					
1	Materi sesuai dengan konsep matematika koordinat				✓
2	Materi sesuai dengan konsep bilangan biner				✓
3	Materi sesuai dengan konsep listrik arus lemah				✓
4	Materi meningkatkan kemampuan kognitif siswa dalam bidang pemrograman			✓	
5	Materi meningkatkan kemampuan Motorik siswa dalam bidang keahlian elektronika				✓
6	Ilustrasi dalam modul mudah dipahami			✓	
7	Tatabahasa dalam modul mudah dimengerti			✓	
8	Menggunakan pendekatan pegetahuan dasar			✓	
9	Jobsheet mudah dipraktikan				✓
<b>Kemanfaatan materi</b>					
1	Menyediakan panduan penggunaan perangkat sebagai materi ajar				✓
2	Menyediakan contoh program sebagai tugas siswa				✓
3	Menyediakan jobsheet sebagai alat evaluasi			✓	
4	Membuat siswa mau mengingat materi pelajaran yang sudah dilewati			✓	
5	Menjelaskan konsep sistem otomasi			✓	

**C. Instrumen penelitian Kualitas Media**

No	Pernyataan deskriptif	Jawaban			
		STS	TS	S	SS
<b>Hardware</b>					
1	Menggunakan komponen yang mudah didapat				✓
2	Menggunakan suplay listrik yang mudah didapat				✓
3	Desain hardware mudah dibuat				✓
4	Tataletak komponen rapih				✓
5	Memiliki kombinasi warna yang sesuai				✓
6	Penulisan keterangan mudah dipahami				✓
7	Desain hardware sesuai dengan filosofi matematik			✓	

8	Dimensi perangkat praktis					✓
<b>Software</b>						
9	Aplikasi mudah dipasang pada PC lain					✓
10	Tata letak tombol pada aplikasi rapih					✓
11	Aplikasi memberikan animasi yang menarik					✓
12	Aplikasi memiliki kombinasi warna yang sesuai					✓
13	Mampu menampilkan materi yang bersifat abstrak			✓		
14	Desain menggunakan bahasa indonesia sehingga memudahkan siswa					✓
15	Desain menggunakan karakter yang sesuai dengan tanda baca pada bahasa			✓		
16	Desain sesuai dengan notasi matematik			✓		
<b>Pengoperasian</b>						
1	Mudah menyambungkan hardware dengan PC					✓
2	Metode Menghubungkan Perangkat dan Properti yang lain mudah					✓
3	Aplikasi tidak memerlukan waktu proses yang lama					✓
4	Memiliki antarmuka Error pada perangkat					✓
5	Menampilkan antar muka pada editor yang membuat siswa tidak kebingungan dalam penggunaan					✓
<b>Kemanfaatan media</b>						
1	Membuat waktu praktek menjadi lebih singkat					✓
2	Memunculkan minat meng engineering			✓		
3	Memunculkan daya tarik siswa untuk mencoba perangkat					✓
4	Menjelaskan konsep saklar elektronik					✓
5	Tidak Menghabiskan biaya yang besar dalam belajar					✓

#### D. Kotak saran

Sebaiknya keterangan mengenai bit MSB dan LSB ada juga di perangkat

#### E. Kesimpulan

Media pembelajaran inovatif ini dinyatakan :



Layak untuk diuji coba



Tidak layak untuk diuji coba

Penguji Kelayakan

Jakarta, 12 Juni 2017

  
Aodah Piamah  
NIP. 197809192005012003

## LEMBAR PENILAIAN PENGGUNA

Dalam rangka penelitian Tugas Akhir Skripsi saya mohon bantuan Saudara/i untuk menjadi responden Media Belajar inovatif. Berilah tanda checklist (✓) pada pilihan STS, TS, S, SS yang disediakan sesuai dengan kelayakan masing-masing butir pernyataan untuk penilaian materi pada media pembelajaran skripsi dengan judul "PERANGKAT PROGRAMABLE DENGAN MEDIA PEMROGRAMAN MENGGUNAKAN KATA DALAM BAHASA INDONESIA SEBAGAI MODUL PEMBELAJARAN DASAR PEMROGRAMAN"

### A. Identitas pribadi

Nama : *Deksa Rachman*

Tempat/TGL lahir : *06/01/2001*

### B. Petunjuk Pengisian Angket

1. Operasikan perangkat.
2. Baca materi dengan seksama.
3. Jawaban diberikan pada skala penilaian yang telah disediakan, dengan keterangan penilaian sebagai berikut:

SS : Sangat Setuju

S : Setuju

TS : Tidak Setuju

STS : Sangat Tidak Setuju

4. Apabila terdapat kekurangan, mohon kiranya dapat memberikan masukan pada kotak saran.

### C. Instrumen penelitian

No	Pernyataan deskriptif	Jawaban			
		STS	TS	S	SS
<b>Kualitas Hardware</b>					
2	Menggunakan suplay listrik yang mudah didapat			✓	
4	Tataletak komponen rapih			✓	
5	Memiliki kombinasi warna yang sesuai			✓	
6	Penulisan keterangan mudah dipahami			✓	
7	Dimensi perangkat praktis			✓	
<b>Kualitas Software</b>					
8	Aplikasi mudah dipasang pada PC lain				✓
9	Tata letak tombol pada aplikasi rapih				✓
10	Aplikasi memberikan animasi yang menarik			✓	
11	Aplikasi memiliki kombinasi warna yang sesuai			✓	
12	Mampu menampilkan materi yang bersifat abstrak			✓	
13	Desain menggunakan bahasa indonesia sehingga mudah dipahami			✓	
<b>Pengoperasian</b>					
1	Mudah dalam mengoperasikan perangkat			✓	
2	Mudah dalam mengoperasikan aplikasinya				✓
3	Memberikan Antarmuka Error			✓	
4	Mempercepat waktu praktek			✓	
<b>Kemanfaatan</b>					
1	Media pembelajaran memberi pengetahuan dalam bidang elektronika				✓
2	Media pembelajaran memberi pengetahuan dalam bidang pemrograman				✓
3	Media pembelajaran menimbulkan semangat belajar			✓	
4	Media pembelajaran menimbulkan rasa ingintau			✓	

D. Kotak saran

Dalam penyampaian materi tentang obeng untuk mengatasi permasalahan karena kita mungkin akan dilihat banyak orang, namun alangkah lebih baik dengan melihat seberapa jauh kita!

Jakarta, 15 Juni 2017

*Deden*  
(...Deden Rachmat...)

LEMBAR PENILAIAN PENGGUNA

Dalam rangka penelitian Tugas Akhir Skripsi saya mohon bantuan Saudara/i untuk menjadi responden Media Belajar inovatif. Berilah tanda checklist (✓) pada pilihan STS, TS, S, SS yang disediakan sesuai dengan kelayakan masing-masing butir pernyataan untuk penilaian materi pada media pembelajaran skripsi dengan judul "PERANGKAT PROGRAMABLE DENGAN MEDIA PEMROGRAMAN MENGGUNAKAN KATA DALAM BAHASA INDONESIA SEBAGAI MODUL PEMBELAJARAN DASAR PEMROGRAMAN"

A. Identitas pribadi

Nama : Adhista Cindy Rahmayani

Tempat/TGL lahir : Semarang, 3 Juni 1999

B. Petunjuk Pengisian Angket

1. Operasikan perangkat.
2. Baca materi dengan seksama.
3. Jawaban diberikan pada skala penilaian yang telah disediakan, dengan keterangan penilaian sebagai berikut:

SS : Sangat Setuju

S : Setuju

TS : Tidak Setuju

STS : Sangat Tidak Setuju

4. Apabila terdapat kekurangan, mohon kiranya dapat memberikan masukan pada kotak saran.

No	Pernyataan deskriptif	Jawaban			
		STS	TS	S	SS
<b>Kualitas Hardware</b>					
2	Menggunakan suplay listrik yang mudah didapat				
4	Tataletak komponen rapih			✓	
5	Memiliki kombinasi warna yang sesuai			✓	
6	Penulisan keterangan mudah dipahami			✓	
7	Dimensi perangkat praktis			✓	
<b>Kualitas Software</b>					
8	Aplikasi mudah dipasang pada PC lain			✓	
9	Tata letak tombol pada aplikasi rapih			✓	
10	Aplikasi memberikan animasi yang menarik		✓		
11	Aplikasi memiliki kombinasi warna yang sesuai		✓		
12	Mampu menampilkan materi yang bersifat abstrak		✓		
13	Desain menggunakan bahasa indonesia schinga mudah dipahami			✓	
<b>Pengoperasian</b>					
1	Mudah dalam mengoperasikan perangkat			✓	
2	Mudah dalam mengoperasikan aplikasi nya			✓	
3	Memberikan Antarmuka Error		✓		
4	Mempersingkat waktu praktek			✓	
<b>Kemanfaatan</b>					
1	Media pembelajaran memberi pengetahuan dalam bidang elektronika			✓	
2	Media pembelajaran memberi pengetahuan dalam bidang pemrograman			✓	
3	Media pembelajaran menimbulkan semangat belajar			✓	
4	Media pembelajaran menimbulkan rasa ingintau			✓	

#### D. Kotak saran

Tampilan kurang menarik dalam pemilihan warna

---



---



---



---



---



---



---



---



---



---

Jakarta, 15 Juni 2017

*[Signature]*  
(Adhira...Gady...R...)

**LAMPIRAN 3**  
**BUKU PANDUAN**

**PANDUAN PENGGUNAAN  
PERANGKAT PROGRAMMABLE  
DAN  
MEDIA PEMROGRAMAN TITIK**



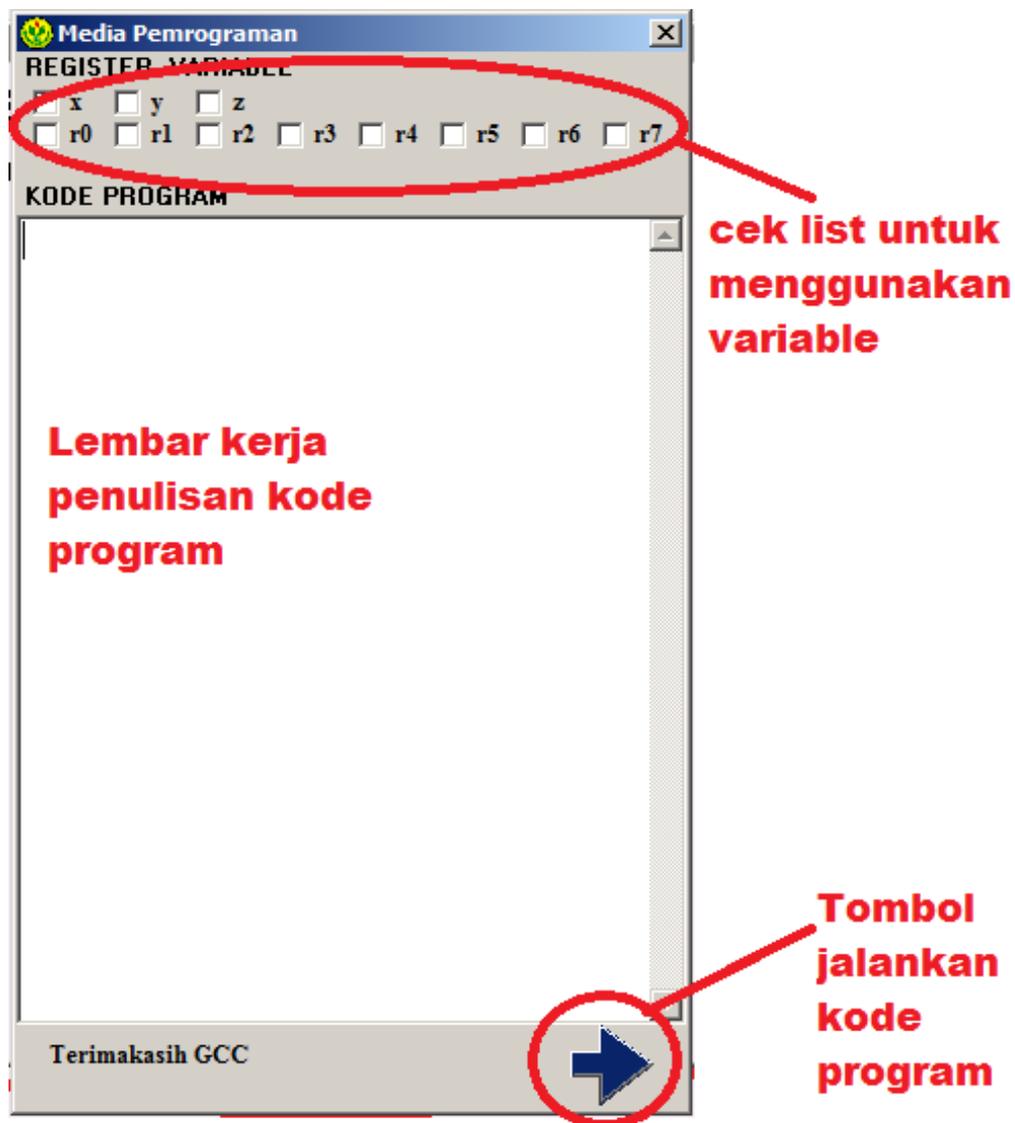
**DISUSUN OLEH MAS KAREBET SETIAWAN**

**UNTUK PENGGUNAAN PERANGKAT  
DENGAN CONTOH PEMROGRAMANNYA**

## DAFTAR ISI

<b>DAFTAR ISI</b> .....	<b>3</b>
<b>1. SOFTWARE APLIKASI</b> .....	<b>4</b>
A. Keterangan elemen aplikasi.....	4
B. Jendela notifikasi .....	6
<b>2. SINTAK</b> .....	<b>8</b>
A. fungsi .....	8
B. Setruktur .....	9
C. Operator .....	10
<b>3. PERANGKAT</b> .....	<b>11</b>
A. Seting mode .....	11
<b>CONTOH PROGRAM</b>	

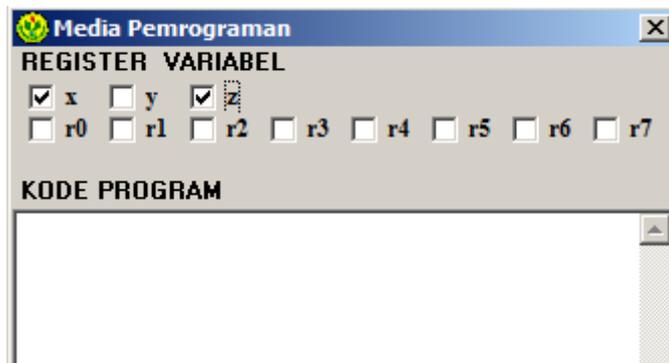
## 1. SOFTWARE APLIKASI



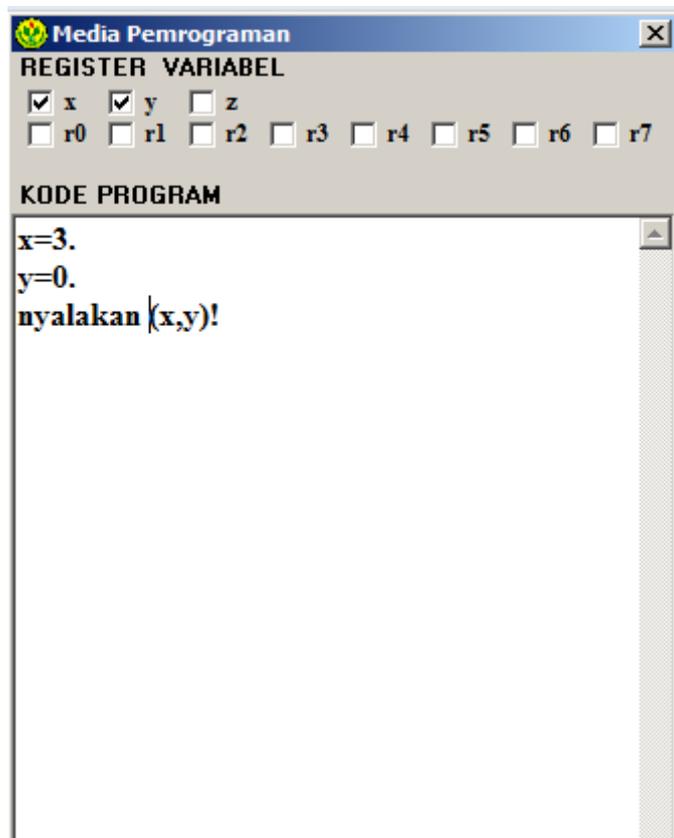
### A. Keterangan elemen aplikasi

Ceklis penggunaan variable : Digunakan saat membutuhkan adanya variable pada program yang akan dibuat cara menggunakannya cukup klik sekali variable

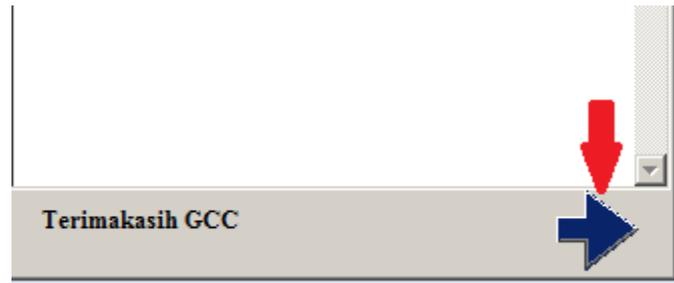
Contoh penggunaan variable x dan z:



Lembar kerja aplikasi : lembar kerja aplikasi berupa textbox pengguna dapat menuliskan program yang akan dibuat pada elemen ini contoh penulisan sebagai berikut:



Tombol Download : Digunakan ketika program selesai dibuat dan ingin ditanamkan ke perangkat, cara menggunakannya cukup dengan klik satu kali berikut gambar tombol download:

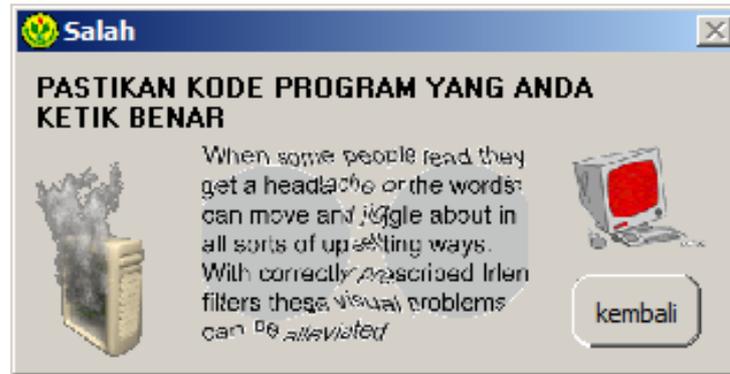


## B. Jendela notifikasi

Jendela notifikasi error koneksi : Muncul Ketika tombol download kode program diklik dan perangkat dalam keadaan mode run atau tidak tersambung, Menandakan aplikasi desktop tidak mendeteksi adanya perangkat.



Jendela notifikasi salah : Muncul Ketika tombol download kode program diklik dan program yang dibuat tidak sesuai dengan sintak pada panduan.



Jendela notifikasi berhasil : Muncul Ketika tombol download kode program diklik dan program yang dibuat sesuai dengan sintak yang ada pada buku panduan



## 2. SINTAK

### A. fungsi

**nyalakan(x,y)!** Digunakan saat pemrogram ingin menyalakan pin pada koordinat (x,y) dalam prakteknya dapat langsung diinput nilai (2,3) tanpa menggunakan variable.

**matikan(x,y)!** Digunakan saat pemrogram ingin mematikan pin pada koordinat (x,y) dalam prakteknya dapat langsung diinput nilai (2,3) tanpa menggunakan variable.

**Kondisi(x,y)** Digunakan saat pemrogram ingin menjadikan pin koordinat (x,y) seagai input pada pin perangkat. Pada prakteknya dapat langsung diinput nilai (2,3) tanpa menggunakan variable.

**tunda ...(1) detik.** Digunakan saat pemrogram ingin membuat aliran program mengandung waktu tunda dengan ...(1) sebagai nilainya.

## **B. Setruktur**

**Jika ...<sup>(1)</sup> maka lakukan ...<sup>(2)</sup>** Digunakan saat pemrogram ingin membuat aliran program percabangan dengan ...<sup>(1)</sup>  
**Selesai** seagai syarat dan ...<sup>(2)</sup> sebagai aksi programyang akan dijalankan.

**ketika ...<sup>(1)</sup> maka lakukan ...<sup>(2)</sup>** Digunakan saat pemrogram ingin membuat aliran program perulangan dengan ...<sup>(1)</sup>  
**Selesai** seagai syarat dan ...<sup>(2)</sup> sebagai aksi programyang akan dijalankan.

## C. Operator

### Operator Aritmatika

Operator	Keterangan	Contoh
+	Penjumlahan	$X=3+5.$
-	Pengurangan	$X=3-5.$
*	Perkalian	$X=3*5.$
/	Pembagian	$X=5/3.$
%	Sisa pembagian	$X=5\%3.$

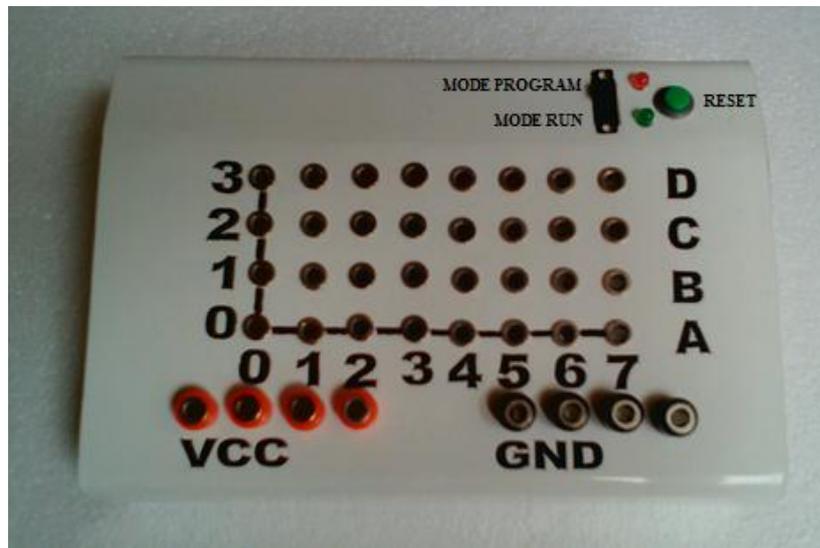
### Operator Manipulasi bit

Operator	Keterangan	Contoh
&	And	$X=F\&F.$
	Or	$X=F F.$
++	Penjumlahan 1	$X++.$
--	Pengurangan 1	$X--.$

### Operator Implikasi dan Bimplikasi

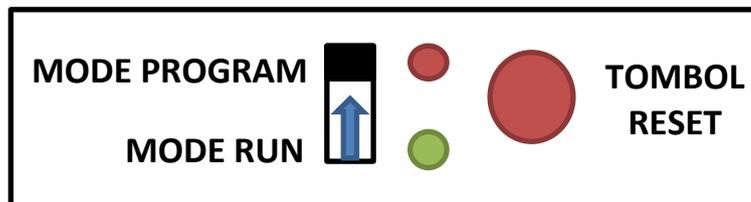
Operator	Keterangn	Contoh
>	Lebih dari	Jika $x > 7$ maka ...
<	Kurang dari	Jika $x < 7$ maka ...
==	Sama dengan	Jika $x == 7$ maka ...
&&	Dan jika	Jika $x >7 \&\& y < 5$ maka...
	Atau jika	Jika $x > 7    y <5$ maka ...

### 3. PERANGKAT

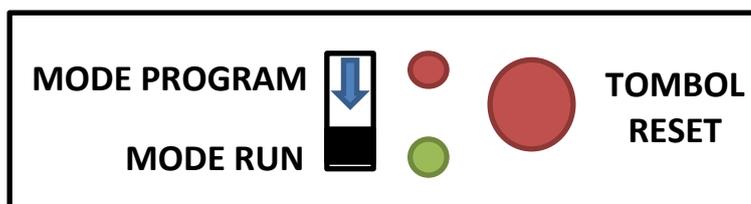


#### A. Seting mode

Mode program : Untuk masuk mode program ubah posisi saklar pada mode program dan tekan tombol reset satukali dan pastikan LED merah menyala.



Mode run : Untuk masuk mode run ubah posisi saklar pada mode run dan pastikan LED hijau menyala.



## B. Penggunaan

Untuk Memulai Menggunakan Ikuti langkah-langkah berikut ini:

1. Hubungkan Perangkat Programmable ke PC melalui port USB
2. Seting perangkat ke mode Program
3. Buka Software Titik
4. Klik Belajar Dasar
5. Ceklis Variable x dan y
6. Ketikkan program dibawah ini:

x=0.

y=1.

nyalakan(x,y)!

tunda 1 detik.

matikan(x,y)!

tunda 1 detik.

7. Klik tombol  Pada jendela aplikasi, Tunggu LED pada perangkat berkedip dan muncul notifikasi berhasil muncul.
8. Seting perangkat ke mode run

# **CONTOH PROGRAM**