

## **BAB 2**

### **STUDI AWAL**

#### **2.1 Karya yang Berkaitan**

##### **2.1.1 Pewaktu Iqomah Tampilan Satu Digit**

Pewaktu iqomah satu digit adalah alat yang digunakan untuk memberi waktu antara adzan sampai dengan iqomah dengan tampilan satu digit (satu 7 segment). Tampilan yang menggunakan satu digit (berupa satuan menit) akan membingungkan orang yang melihat, lebih-lebih orang yang pertama kali melihat alat tersebut, karena mereka tidak akan menyangka kalau angka yang muncul itu sebenarnya adalah menit menuju iqomah. Selain itu, alat tersebut juga tidak memiliki menu yang digunakan untuk memilih berapa waktu yang diinginkan, serta tidak dapat digunakan untuk fungsi yang lain saat menu pewaktu iqomah sedang tidak digunakan (misalnya untuk jam atau pengukur suhu).

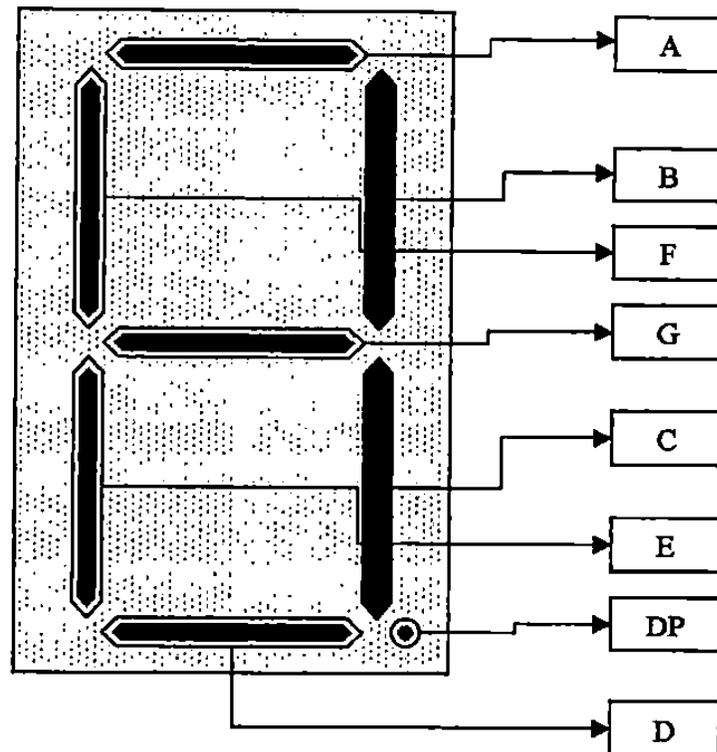
##### **2.1.2 Display Jam dan Jadwal Waktu Sholat**

Display Jam dan Jadwal Waktu Sholat adalah sebuah display yang di dalamnya berisi beberapa kelompok 7 segment ataupun *dot matrix* yang memiliki fungsi masing-

## 2.2 Dasar-Dasar Teoritis

### 2.2.1 7 Segment (7 Segment)

7 segment merupakan komponen elektronika yang terdiri atas tujuh buah LED (*light emitting diode*) yang disusun sedemikian rupa (menyerupai angka 8) sehingga jika beberapa LED tersebut menyala maka akan dapat menampilkan suatu angka, huruf atau simbol. Biasanya disertakan juga sebuah LED sebagai Dot Point yang dapat digunakan sebagai tanda titik atau koma pada display. Ada dua jenis 7 segment yang beredar yakni 7 seg Common Anode (CA) dan 7 seg Common Cathode (CC).



Gambar 2.1 Bentuk dan fungsi 7 segment

Penggunaan 7 segment ini cukup mudah, sebagai contoh jika ingin menampilkan angka 1 maka segment B dan C saja yang dinyalakan sedangkan segment yang lain dalam kondisi mati. Logika binernya adalah 0110000.

7 segment dipilih sebagai penampil data output (*display*) adalah karena mudah ditemukan dipasaran dan dapat diperoleh dengan harga yang relatif murah dibanding dengan komponen yang lain. Selain itu juga karena data outputnya berupa angka, maka akan lebih mudah pembacaannya apabila menggunakan 7 segment.

### **2.2.2 Multiplexing 7 Segment**

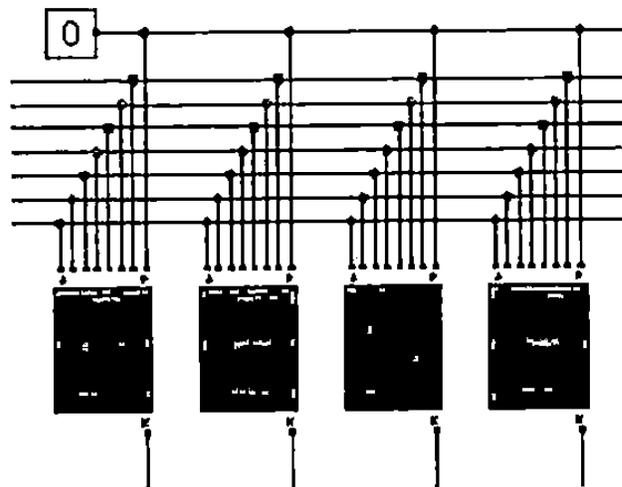
*Multiplexing 7 Segment* adalah sebuah metode untuk menggabungkan beberapa 7 segment agar dapat dikonfiigurasikan menjadi 1 bagian. Hal ini dilakukan agar penggunaan port dalam mikrokontroler ATMEGA16 yang hanya berjumlah 40 port dapat mencukupi seluruh kebutuhan.

Jika menggunakan metode yang biasa (satu segment satu pin), maka untuk menyalakan 4 buah 7 segment membutuhkan paling tidak 28 port pada mikrokontroler. Namun dengan menggunakan metode *Multiplexing* ini, kebutuhan akan port tersebut dapat diminimalisi, yakni hanya membutuhkan 7 port saja pada mikrokontroler.

Konfigurasi rangkaian dari masing-masing PIN dalam 7 segment dihubungkan secara paralel. PIN A pada semua 7 segment dihubungkan secara paralel dan selanjutnya dihubungkan satu port mikrokontroler. Demikian seterusnya

untuk PIN B, PIN C, PIN D, PIN E, PIN F, PIN G, dan PIN DP (Dot Point). Untuk pin *Common* (tegangan referensi 7 segment) tidak boleh diparalelkan. Khusus untuk *common 7 segment*, dirangkai satu *common* satu port pada mikrokontroler (karena ini nantinya yang akan dipakai untuk menyalakan setiap 7 segment).

Tatacara penampilan dari 7 segment ini sebenarnya adalah bergantian sesuai dengan urutan penampilan 7 segment per digit pada program yang telah dimasukkan ke dalam mikrokontroler. Namun urutan penampilan 7 segment dilakukan dengan sangat cepat sekali sampai mata kita tidak dapat mengikutinya. Sehingga seolah-olah angka pada 7 segment dimunculkan secara bersamaan.



**Gambar 2.2** Konfigurasi *Multiplexing 7 segment*

### 2.2.3 Mikrokontroler ATMEGA16

Mikrokontroler dapat dianalogikan dengan sebuah sistem komputer yang dikemas

ke dalam sebuah chip. Chip ini adalah sebuah IC mikrokontroler sebetulnya

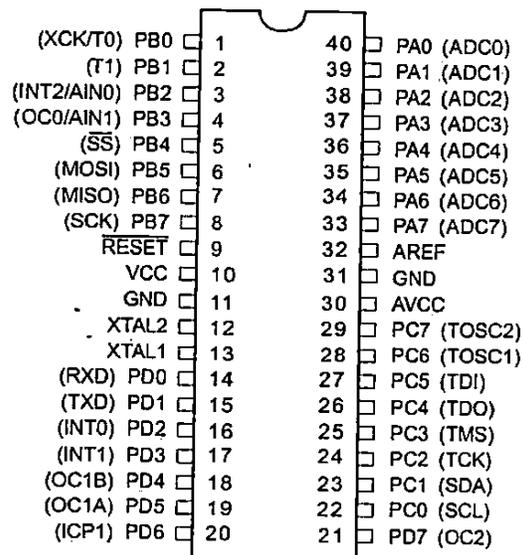
sudah terdapat kebutuhan minimal agar mikroprosesor dapat bekerja, yakni meliputi mikroprosesor, ROM, RAM, I/O dan clock seperti halnya yang dimiliki, misalnya, oleh komputer PC. Mengingat kemasannya yang sangat kecil tentu saja spesifikasi dan kemampuan yang dimiliki oleh mikrokontroler menjadi lebih rendah apabila dibandingkan dengan komputer seperti PC baik dilihat dari segi kecepatannya, kapasitas memori maupun fitur-fitur yang dimilikinya. (Agus Bejo, 2008). Menurut Agfianto Eko Putra (2005) mikrokontroler lebih banyak digunakan untuk mengendalikan sistem-sistem otomatis yang berdiri sendiri (*stand alone*).

Mikrokontroler ATMEGA16 adalah jenis mikrokontroler keluarga ATmega yang dikeluarkan oleh perusahaan ATMEL. ATMEGA16 adalah *low-power* CMOS 8-bit mikrokontroler yang berbasis arsitektur AVR (*Alf and Vegard's RISC processor*). AVR berteknologi RISC (*Reduced Instruction Set Computing*).

Berikut adalah fasilitas yang dimiliki oleh ATMEGA16:

- Saluran Input/Output sejumlah 32 buah, yaitu PORT A, Port B, Port C, dan Port D.
- Kecepatan mencapai 16 MIPS dengan kristal 16 MHZ.
- 16 Kbyte Flash Memori, yang memiliki fasilitas *in-System Programming*.
- 512 Byte internal EEPROM.
- 1K Byte SRAM.
- Programming Lock, fasilitas untuk mengamankan kode program.

- 4 Channel PWM.
- ADC (*Analog to Digital Converter*) 10 bit sebanyak 8 channel.
- *Watchdog Timer* dengan osilator internal
- Tegangan operasi 2.7 V - 5.5 V pada ATMEGA16L dan 4.5 V – 5.5 V pada ATMEGA16.
- Unit interupsi internal dan eksternal.
- Port antarmuka SPI.
- Antarmuka komparator analog.
- *Port USART programmable* untuk komunikasi serial.



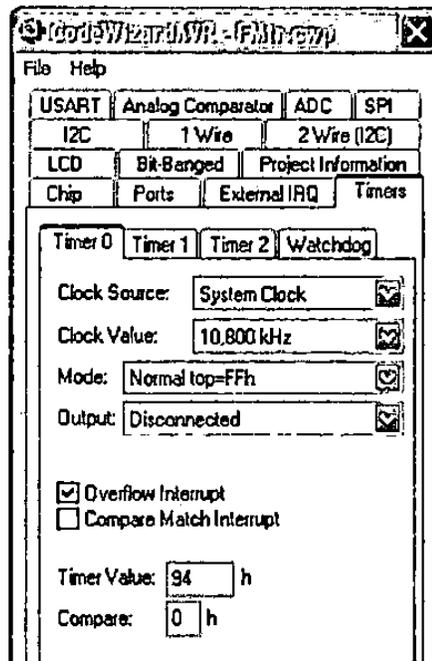
**Gambar 2.3** Konfigurasi Pin ATMega16

#### 2.2.4 Fasilitas Timer pada Mikrokontroler ATMEGA16

Atmega16 memiliki 2 buah timer/counter 8-bit (timer0 dan timer2) dan 1 buah timer/counter 16-bit (timer1). Timer0 dan timer2 merupakan modul timer/counter 8-bit yang dapat berfungsi sebagai pencacah tunggal, pembangkit PWM 8-bit, pembangkit frekuensi, pencacah kejadian eksternal, pembangkit interupsi *overflow* dan pembangkit output *compare*. Mode kerja timer/counter 2 sama persis dengan mode kerja timer/counter0, hanya saja pada timer/counter 2 memiliki fitur tambahan yaitu *asynchronous mode*. Perbedaan antara *asynchronous* dengan *synchronous mode* terletak pada sumber clock saja.

- *synchronous mode* (timer/counter0 dan 1) → sumber *clock* kristal XTAL1 dan XTAL2.
- *asynchronous mode* → sumber *clock* eksternal TOSC1 dan TOSC2.

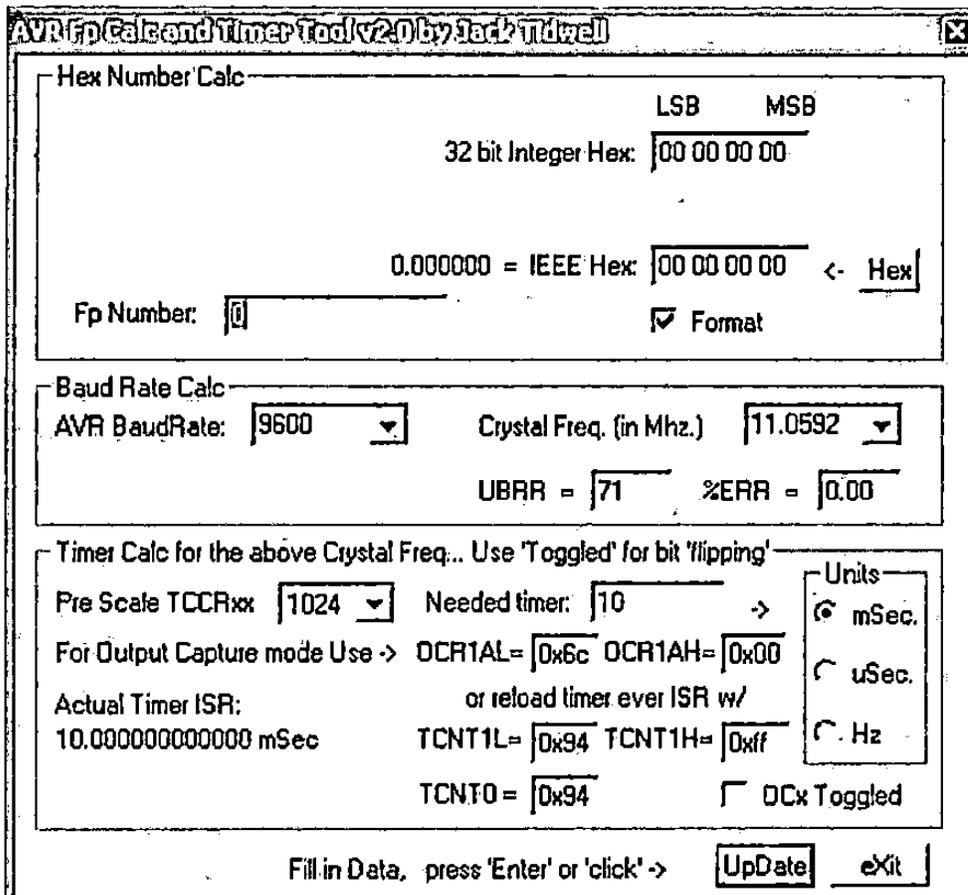
Penentuan *Timer Value* menggunakan bantuan software AVRcalc. Pada Gambar 2.4 ditunjukkan setting timer pada CodeWizardAVR



**GAMBAR 2.4** Setting timer pada AVR

Untuk menggunakan timer pada AVR, perlu dilakukan konfigurasi nilai-nilai register timer agar timer dapat berjalan sesuai yang kita inginkan. Sebagai contoh kita akan membuat timer 10ms. Konfigurasinya dapat menggunakan seperti yang ada di

1. Untuk Timer 0, kita dapat dituliskan dengan menggunakan bantuan



**GAMBAR 2.5** Setting nilai pada AVRcalc

- Crystal Freq disesuaikan dengan kristal yang digunakan dalam rangkaian mikrokontroler.
- Pre Scale TCCRxx disesuaikan dengan clock value pada CodeWizard.
- Needed Timer diisi dengan waktu yang diinginkan dengan satuan dipilih pada units.
- Klik update untuk mendapatkan nilai Timer Value pada kolom

TCNT1L dan TCNT1H untuk timer1 dan TCNT0 untuk timer0

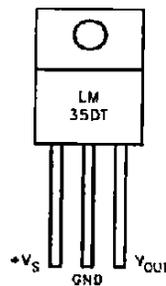
### 2.2.5 Sensor Suhu LM35

Sensor suhu LM35 adalah komponen elektronika yang diproduksi oleh *National Semiconductor*, yang memiliki fungsi untuk mengubah besaran suhu menjadi besaran listrik (tegangan). LM35 memiliki keakuratan tinggi dan kemudahan perancangan jika dibandingkan dengan sensor suhu yang lain, LM35 juga mempunyai keluaran impedansi yang rendah dan linieritas yang tinggi sehingga dapat dengan mudah dihubungkan dengan rangkaian kendali khusus serta tidak memerlukan penyetelan lanjutan.

Meskipun tegangan catu daya ke sensor ini dapat mencapai 30 volt akan tetapi yang diberikan adalah sebesar 5 volt, sehingga dapat digunakan dengan catu daya tunggal dengan ketentuan bahwa LM35 hanya membutuhkan arus sebesar 60  $\mu\text{A}$  hal ini berarti LM35 mempunyai kemampuan menghasilkan panas (*self-heating*) dari sensor yang dapat menyebabkan kesalahan pembacaan yang rendah yaitu kurang dari 0,5 °C pada suhu 25 °C . Sensor suhu ini memiliki karakteristik sebagai berikut:

- Memiliki sensitivitas suhu, dengan faktor skala linier antara tegangan dan suhu 10 mVolt/°C, sehingga dapat dikalibrasi langsung dalam *celcius*.
- Memiliki ketepatan atau akurasi kalibrasi yaitu 0,5°C pada suhu 25 °C seperti terlihat pada gambar 2.2.
- Memiliki jangkauan maksimal operasi suhu antara -55 °C sampai +150 °C.
- Bekerja pada tegangan 4 sampai 30 volt.

- Memiliki pemanasan sendiri yang rendah (*low-heating*) yaitu kurang dari 0,1 °C pada udara diam.
- Memiliki impedansi keluaran yang rendah yaitu 0,1 W untuk beban 1 mA.
- Memiliki ketidaklinieran hanya sekitar  $\pm \frac{1}{4}$  °C.



**GAMBAR 2.6** Tampak depan LM35

LM35 memiliki 3 buah pin. Pin 1 berfungsi sebagai sumber tegangan kerja, pin 2 (tengah) digunakan sebagai tegangan keluaran atau  $V_{out}$  dengan jangkauan kerja dari 0 Volt sampai dengan 1,5 Volt. Tegangan operasi LM35 antara 4 Volt sampai 30 Volt. Keluaran sensor ini akan naik sebesar 10 mV setiap derajat *celcius* sehingga diperoleh persamaan sebagai berikut :

$$V_{LM35} = \text{Suhu} * 10 \text{ mV}$$

### 2.2.6 Fasilitas Analog to Digital Converter (ADC) pada ATMEGA16

Data yang dikirimkan oleh sensor suhu LM35 merupakan data analog yang berbentuk tegangan. Data analog ini tidak akan dapat ditampilkan pada display 7 segment secara langsung. Data akan dapat ditampilkan apabila nilai keluaran dari sensor suhu LM35 diubah menjadi data digital.

Mikrokontroler ATmega16 memiliki fasilitas *Analog to Digital Converter* (ADC) yang sudah *built-in* dalam chip. Fitur ADC internal inilah yang menjadi salah satu kelebihan mikrokontroler ATmega16 bila dibandingkan dengan beberapa jenis mikrokontroler yang lain. Dengan adanya ADC internal ini kita tidak akan direpotkan lagi dengan kompleksitas hardware saat membutuhkan proses perubahan sinyal dari analog ke digital seperti yang harus dilakukan jika kita memakai komponen IC ADC eksternal (Agus Bejo, 2008).

ATmega 16 memiliki resolusi ADC 10-bit dengan 8 channel input dan mendukung 16 macam penguat beda. ADC ini bekerja dengan teknik *successive approximation*. Rangkaian internal ADC ini memiliki catu daya tersendiri yaitu pin AVCC. Tegangan AVCC harus sama dengan VCC  $\pm$  0,3V.

Data hasil konversi dirumuskan sebagai berikut:

- Untuk konversi tunggal:

$$ADC = \frac{V_{IN} \cdot 1024}{V_{REF}}$$

dengan:

$V_{in}$  : Tegangan masukan pada pin yang dipilih

$V_{ref}$  : Tegangan referensi yang dipilih

- Untuk penguat beda:

$$ADC = \frac{(V_{POS} - V_{NEG}) \cdot GAIN \cdot 512}{V_{REF}}$$

dengan:

$V_{pos}$  : Tegangan masukan pada pin positif.

$V_{neg}$  : Tegangan masukan pada pin negatif.

Gain : Faktor penguatan.

$V_{ref}$  : Tegangan referensi yang dipilih.

Untuk mengatur mode dan cara kerja ADC dilakukan melalui register *ADC Multiplexer Selection Register (ADMUX)*, *ADC Control and Status Register A (ADCSRA)*, *ADC Data Register (AD\_CONVERTER)*, *ADC Auto Trigger Source*

### **2.2.7 DS1307 Serial Real Time Clock (RTC)**

RTC yang digunakan dalam proyek ini adalah RTC DS1307 dengan antarmuka I2C. RTC DS1307 menyediakan pewaktu dalam detik, menit, jam, hari, tanggal, menit, bulan dan tahun yang memiliki akurasi hingga tahun 2100. Selain itu, RTC ini menyediakan pin baterai cadangan untuk dihubungkan pada baterai lithium 3V atau sumber energi lain sehingga ketika suplai energi utama (VCC dan GND) mati, baterai cadangan mengambil alih suplai energi pada RTC dan timer tetap berjalan sebagaimana mestinya. Konsumsi arus dari baterai cadangan 3V lithium 48mAh oleh RTC hanya kurang dari 500nA sehingga dengan baterai tersebut mampu bertahan kira-kira hingga 11 tahun. Beberapa karakteristik dari DS1307 adalah sebagai berikut:

- Suhu dan tegangan berbanding lurus (linear) yaitu  $10 \text{ mV}^\circ\text{C}$ .
- Toleransi kesalahan 0,5 V.
- Dapat mengukur suhu antara  $-55^\circ\text{C} - 150^\circ\text{C}$ .
- Bekerja pada tegangan 4-20VDC.
- Kebutuhan arus 60uA.

### **2.2.8 Komunikasi Serial I2C**

Terdapatnya 2 mikrokontroler dalam sebuah rangkaian tidak dapat dengan begitu saja dihubungkan. Apalagi seandainya masing-masing mikrokontroler itu memiliki kepentingan dengan mikrokontroler yang lain. Maka perlu sebuah metode

yang dapat digunakan untuk mengkomunikasikan 2 buah

mikrokontroler tersebut. Salah satu metode yang dapat dipakai adalah komunikasi serial I2C.

Komunikasi serial I2C merupakan salah satu jenis komunikasi *synchronous* yang hanya memerlukan 2 jalur komunikasi (*2-wire*) yaitu jalur data (SDA) dan jalur clock (SCL). Jalur data digunakan untuk mengirimkan dan menerima data (bersifat *bi-directional*) sedangkan jalur clock digunakan untuk mengirimkan sinyal sinkronisasi.

Jika kita bekerja pada level pemrograman *assembler* maka kita akan sedikit kesulitan untuk mempelajari register-register dan bagaimana menyusun format data/protokol komunikasi I2C. Hal ini berbeda apabila kita menggunakan pemrograman tingkat tinggi seperti bahasa C. Codevision AVR telah menyediakan fungsi pustaka khusus yaitu *i2c.h* yang dapat digunakan dengan mudah untuk mendukung protokol komunikasi I2C. Dengan fungsi ini kita tidak perlu repot lagi mendefinisikan protokol komunikasi serial I2C tetapi hanya cukup dengan memanggil beberapa fungsi yang sudah disediakan. Menurut (Agus Bejo, 2008) beberapa fungsi I2C adalah sebagai berikut.

**void i2c\_init (void)**

Fungsi ini digunakan untuk menginisialisasi I2C. Fungsi ini harus dipanggil pertama kali sebelum memanggil fungsi-fungsi I2C yang lain.

**unsigned i2c\_start (void)**

Fungsi ini digunakan untuk memulai protokol komunikasi I2C.

komunikasi I2C. Nilai baliknya dapat dipakai untuk melihat status jalur I2C, jika bernilai 1 berarti jalur I2C sedang kosong tetapi jika bernilai 0 berarti jalur I2C sedang sibuk.

**void i2c\_stop (void)**

Fungsi ini digunakan untuk membuat sinyal *stop* pada protokol komunikasi I2C.

**unsigned char i2c\_read (unsigned char ack)**

Fungsi ini digunakan untuk membaca/menerima data dari jalur I2C. Dalam fungsi ini terdapat sebuah argumen bernama *ack* yang dipakai untuk mengecek sinyal *unacknowledgement*. Jika argument *ack* diset '1' maka *unacknowledgement* diaktifkan sedangkan jika bernilai '0' berarti tidak diaktifkan. Nilai balik fungsi ini merupakan data hasil pembacaan dari fungsi tersebut.

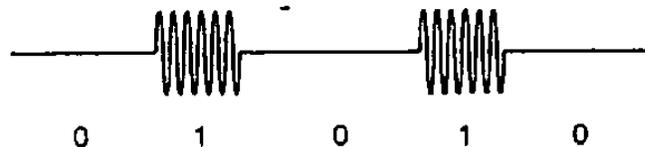
**unsigned char i2c\_write (unsigned char data)**

Fungsi ini digunakan untuk menulis/mengirimkan data ke jalur I2C. Dalam fungsi ini terdapat nilai balik yang merupakan sinyal *unacknowledgement* dimana akan bernilai 1 jika *device slave* (peripheral yang terhubung ke mikrokontroler) mengirimkan sinyal *unacknowledgement* dan sebaliknya bernilai 0 jika *device slave* tidak mengirimkan sinyal *unacknowledgement*.

### 2.2.9 Infra Red Transmitter dan Receiver

*Infra Red Receiver* (IR Receiver) digunakan sebagai sebuah sub rangkaian yang berfungsi sebagai bagian penerima data yang dikirimkan oleh pengirim data (*transmitter*). Dalam hal ini pengiriman data dilakukan dengan media cahaya infra merah. Oleh karena itu pada sisi penerima juga harus dirancang agar dapat menerima data yang dikirim melalui cahaya infra merah. Adapun pengirim data yang digunakan dalam Skripsi ini adalah *remote control* infra merah merk Sony.

Pada dasarnya, data yang dikirimkan oleh suatu *remote control* disertai dengan *carrier*-nya. Hal ini dimaksudkan supaya data dapat ditransmisikan dalam jarak yang jauh. Pada *remote control* Sony frekuensi *carrier*-nya berkisar 40 KHz. Sinyal high yang dikirimkan oleh *remote control* Sony ditumpangkan pada *carrier* sebesar 40 KHz sehingga sebetulnya di dalam pulsa high tersebut terdapat pulsa-pulsa kecil dengan frekuensi yang lebih tinggi.



Gambar 2.7 Gelombang pembawa pada data

Pada rangkaian penerima (*IR Receiver*), digunakan Infra-red Diode sebagai penerima infra merah dari *Remote control*. Akan tetapi tentu saja memerlukan piranti lain

yang dapat diterima oleh mikrokontroler sebagai suatu data

digital. Untuk itu, sebelum data tersebut diterima oleh mikrokontroler maka carrier-nya harus dihilangkan. Selain itu sinyal dari IR diode juga harus dikuatkan.

#### **2.2.10 Fasilitas Interupsi Eksternal pada ATMEGA16**

Di dalam mikrokontroler ATMEGA 16 terdapat 3 sumber interupsi eksternal yaitu INT0, INT1 dan INT2. Ketiga interupsi ini dapat terpicu dengan adanya perubahan level baik transisi naik (*rising edge*) maupun transisi turun (*falling edge*) pada pin INT0, INT1 dan INT2 sesuai dengan pengaturan mode interupsinya meskipun pada saat itu ketiga pin tersebut dikonfigurasi sebagai output. Untuk pengaturan mode dan cara kerja interupsi eksternal dilakukan melalui 2 buah register I/O yaitu register MCUCR dan register MCUCSR (*Agus Bejo, 2008*).