

PERCOBAAN 3

APLIKASI MIKROKONTROLER

Tujuan :

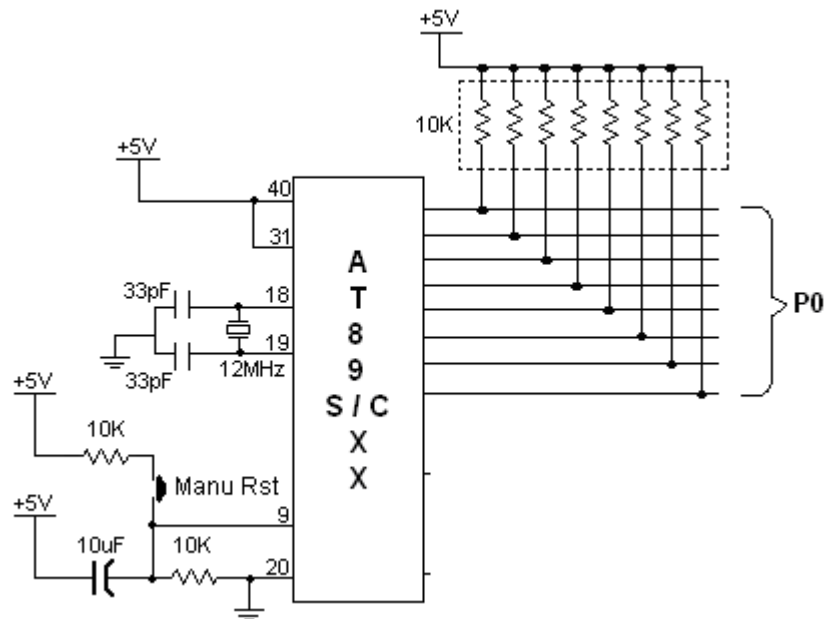
1. Mempelajari rangkaian aplikasi mikrokontroler AT89S51.
2. Memperkenalkan bentuk fisik Mikrokontroler.
3. Memperlihatkan input/output AT89S51 menggunakan Basic Training Kit.
4. Program Aplikasi Port 0 sebagai penggerak (driver) penampil 7 segment display.

Teori Dasar :

Aplikasi mikrokontroler AT89S51 mempergunakan Modul Basic Training Kit. Basic Training Kit ini terdiri dari beberapa bagian sebagai berikut.

1. Mikrokontroler AT89S51

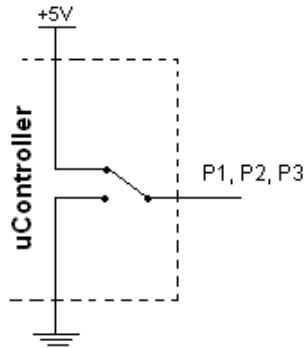
Merupakan bagian utama Basic Training Kit dimana bagian ini merupakan konfigurasi minimum sistem Mikrokontroler untuk dapat digunakan. Skematik bagian Mikrokontroler AT89S51 adalah sebagai berikut.



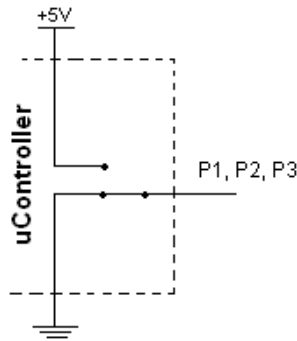
Gambar 4.1 Rangkaian Konfigurasi Standar Mikrokontroler Yang Disertai Manual Reset

Gambar di atas merupakan konfigurasi standar yang telah ditentukan sebagai dasar rangkaian dari Mikrokontroler jenis AT89S/CXX. Dari seluruh port (P0, P1, P2, P3) bila ingin menggunakan P0 harus diberi rangkaian pull

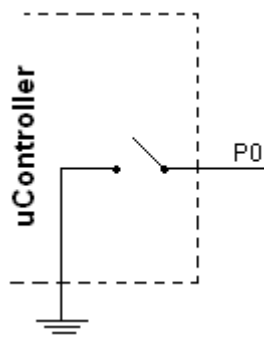
up (RPack) agar bila P0 yang dijadikan sebagai port output, saat diberikan nilai logika high maka P0 dapat mengeluarkan tegangan sebesar +5V karena bila P0 tidak diberi pull up saat kondisi high P0 tidak mengeluarkan tegangan +5V seperti port lainnya pada Mikrokontroler tetapi tidak bisa juga dikatakan kondisi low, hal ini disebut juga sebagai kondisi floating (gantung). Secara mudahnya perhatikan ilustrasi port Mikrokontroler sebagai output dibawah ini :



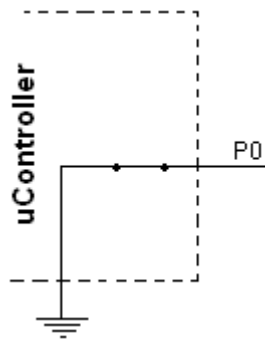
~ P1, P2, P2 saat kondisi high, port terhubung dengan Vcc ~



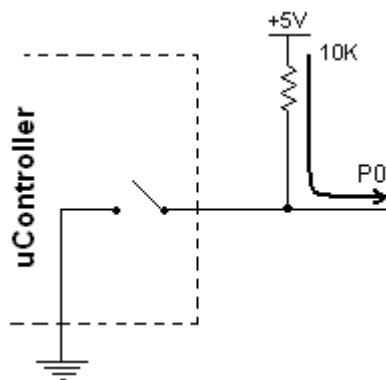
~ P1, P2, P2 saat kondisi low, port terhubung dengan ground ~



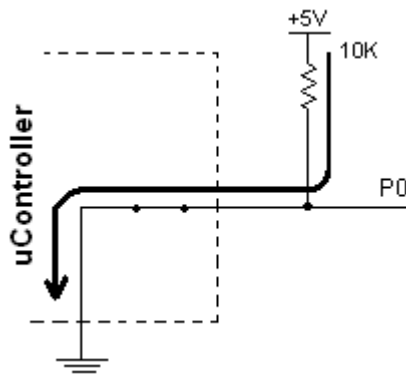
~ P0 saat kondisi high, port tidak terhubung dgn ground (floating) ~



~ P0 saat kondisi low, port terhubung dengan ground ~

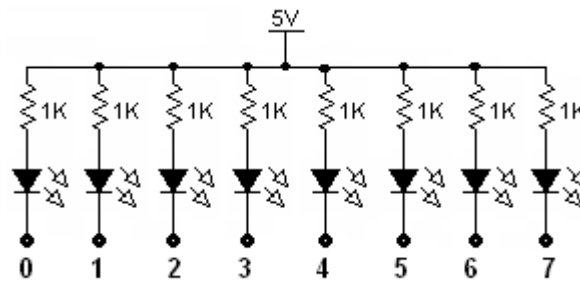


~ P0 saat kondisi high dengan penggunaan pull up, arus mengalir menuju port ~



~ P0 saat kondisi low dengan penggunaan pull up, arus mengalir menuju ground ~

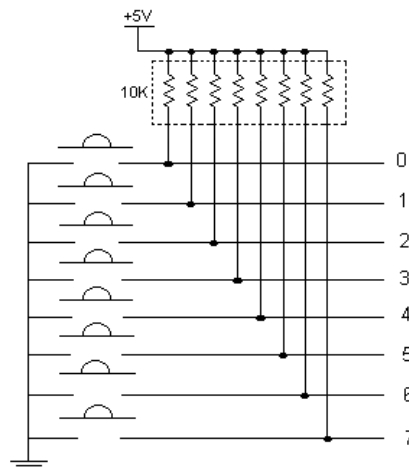
2. LED



~ Rangkaian Led ~

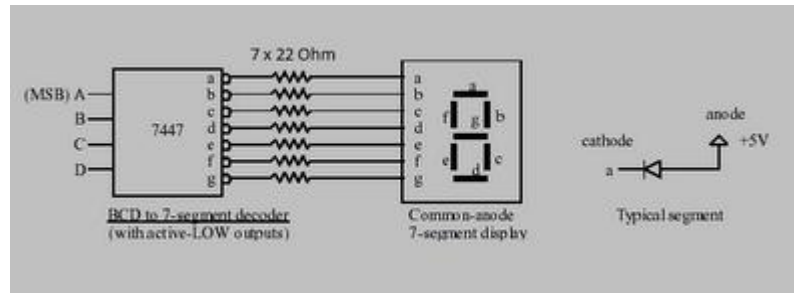
Rangkaian LED terdiri dari delapan buah LED yang dapat difungsikan sebagai keluaran 8-bit dari setiap port Mikrokontroler AT89S/CXX rangkaian LED ini erhubung ke PORT 3 Mikrokontroler. Jenis konfigurasi LED ini adalah aktif LOW.

3. PUSH-ON SWITCH



Hampir sama dengan fungsi DIP Switch, hanya saja saklar yang digunakan tidak memiliki sifat toggle, Push-On pada kondisi defaultnya memberikan logika 1 (HIGH), dan hanya memberikan logika 0 (LOW) saat saklar tersebut ditekan, kemudian akan kembali memberikan logika 1 (HIGH) apabila penekanan saklar berakhir. Rangkaian saklar ini tehubung ke PORT 2 Mikrokontroler.

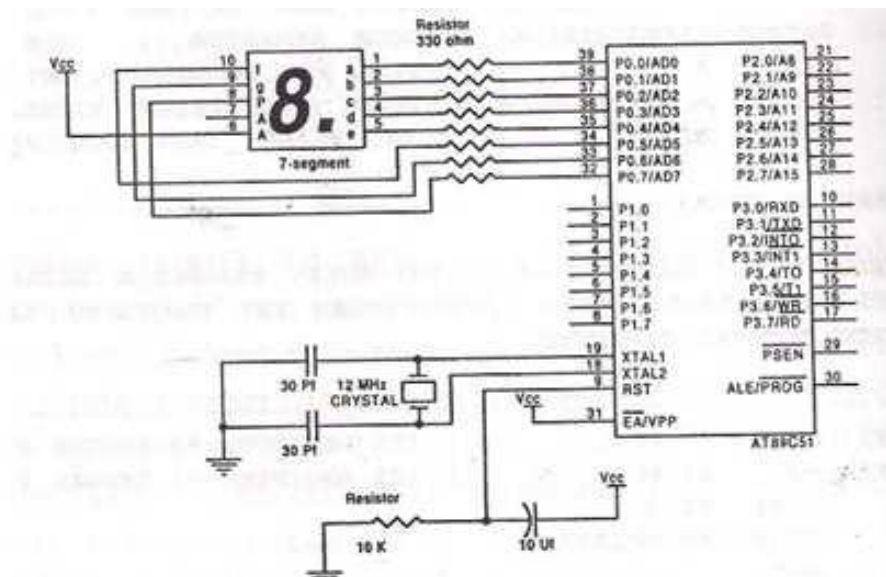
4. 7 SEGMENT BCD 74LS47



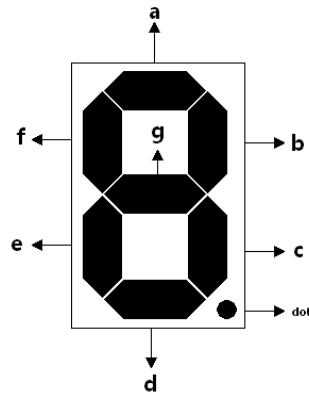
Gambar. Rangkaian 7 Segment BCD 74LS47

Rangkaian 7 segment ini merupakan rangkaian keluaran untuk mikrokontroler. Cara kerja rangkaian diatas adalah jika kita memberikan nilai biner 1 atau 0001 (4bit) pada pin A,B,C,D IC 74ls47 maka pada 7' segment akan tampil angka 1. Jika kita berikan nilai 0010 (2 dalam desimal) maka pada 7' segment akan tampil angka 2. Begitu juga selanjutnya samapai angka 9. Pada rangkaian ini pin A,B,C,D akan terhubung pada PORT mikrokontroler. Pin A,B,C,D bisa dihubungkan pada setiap pin pada port mikrokontroler tergantung pada penggunaannya.

5. Rangkaian Aplikasi



Gambar .Rangkaian Aplikasi Penggerak 7-Segment



Gambar. Struktur Penampil LED 7-segment

Catatan :

Perhatikan hubungan antara port 0 dengan penampil 7 segment yang disusun sebagai berikut :

Mikrokontroler	7 segment (Common Anode)
P0.0	Pin a
P0.1	Pin b
P0.2	Pin c
P0.3	Pin d
P0.4	Pin e
P0.5	Pin f
P0.6	Pin g
P0.7	Pin dot_point

Rangkaian penggerak penampil 7 segment ditunjukkan pada gambar. Kali ini digunakan Port 0 dengan konfigurasi *common Anode* (seperti pada contoh sebelumnya). Dilihat dari struktur Port 0 pada gambar serta sifat-sifat dari Port 0, akan lebih mudah mengkonfigurasi LED dalam 7 segment dengan *common Anode*. Dengan demikian untuk menghidupkan atau menyalakan LED diperlukan logika '0' pada Port 0.

Agar tampilan 7 segment menampilkan angka 0 maka LED a,b,c,d,e dan f harus dinyalakan untuk itu dikirimkan logika 0 pada bit-bit yang terkait yaitu **P0.0, P0.1, P0.2, P0.3, P0.4 dan P0.0 dan P0.5** sehingga data yang terkirim adalah **11000000 = C0h**. Untuk lengkapnya perhatikan tabel.

Program akan menampilkan angka 0 sampai dengan 9 secara bergantian kemudian diulang lagi. Baris-8, 10, 12, 14, 16, 18, 20, 22, 24 dan 26 digunakan untuk menyalakan 7 segment sesuai angka 0 sampai dengan 9.

Tabel daftar Heksa dari tampilan angka pada 7-segment

Tampilan Angka	P0.7 dp	P0.6 g	P0.5 f	P0.4 e	P0.3 d	P0.2 c	P0.1 b	P0.0 a	Heksa
0	1	1	0	0	0	0	0	0	C0
1	1	1	1	1	1	0	0	1	F9
2	1	0	1	0	0	1	0	0	A4
3	1	0	1	1	0	0	0	0	B0
4	1	0	0	1	1	0	0	1	99
5	1	0	0	1	0	0	1	0	92
6	1	0	0	0	0	0	1	0	82
7	1	1	1	1	1	0	0	0	F8
8	1	0	0	0	0	0	0	0	80
9	1	0	0	1	0	0	0	0	90

[illegible]

This image shows a full page of white paper with horizontal dotted lines. The lines are evenly spaced and run across the width of the page, providing a guide for handwriting practice. There are no margins, text, or other markings on the page.

This image shows a full page of white paper with horizontal dotted lines. The lines are evenly spaced and run across the width of the page, providing a guide for handwriting practice. There are no margins, text, or other markings on the page.

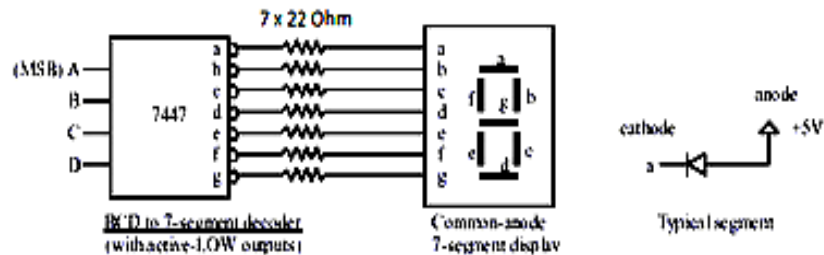
Alat-alat :

- 1 Set IBM PC dengan Sistem Operasi Windows dan Software M-IDE51, dan ISP Flash Programmer, dan TOP View Simulator.

Prosedur percobaan :


Bagian 1

Modul 3.1 Seven Segment Menggunakan BCD



1. Bukalah **M-IDE 51** yang ada di desktop kemudian klik New untuk membuat lembar kerja baru.
2. Ketik Program dan setelah program, Save as program tersebut dalam bentuk **.asm**

PROGRAM	ANALISA
\$mod51	
org 0h	
start: mov a,#00h	
mulai: mov p2,a	
acall delay	
inc a	
cjne a,#05h,mulai	
mati : mov p2,#05h	
jnb p0.1,start	
sjmp mati	
delay: mov r0,#0fh	
ulang: djnz r2,ulang	
djnz r1,ulang	
djnz r0,ulang	
ret	
end	

3. klik **F9** atau klik symbol seperti ini  (Running Program) untuk mengetahui program tersebut terjadi **error** atau **no error**.
4. Setelah program berhasil, kemudian bukalah **TOP View Simulator** untuk melihat simulasi program tersebut.
5. Pilih **Select Device : ATMEL 89C51/52** (tidak perlu mengklik Data Memory maupun Program Memory) dengan **Operating Frequency : 12000000 Hz** Kemudian **OK**.
6. Klik **File**, kemudian arahkan kursor kearah menu **EXTERNAL MODULE SETTING** → klik **Modul Led** → Cek List Pada bagian **Port 2** → Cek List **seven segment**.
7. Setelah itu, pada **Data Input Selection** : ceklist **BCD** dan **Common Anode** (di ceklist) → Ceklist **Non Multiplexed** → Klik **selection of Port Lines and Number of Digits** → Hubungkan **Digit1 A s/d Digit1 dp** untuk ke **P2.0 s/d P2.4** → **OK**
8. Buka Kembali File → **External Module Setting** → klik **Keyboard** → Cek List **P0.1** → **OK** → Setelah itu, buka **View** klik **External Modules** → Klik Module yang kita gunakan yaitu **Led** dan **Keyboard**.
9. Setelah itu Klik **File** Kemudian **Load Program**, Kemudian arahkan kursor ke menu **RUN** dan Klik **GO** untuk menjalankan Simulasi Tersebut dan Klik **Stop** untuk menghentikan simulasinya. **Tanyakan Kepada Asisten Jika Kurang Jelas !!**
10. Catat hasil keluaran pada rangkaian led dan seven segment pada Simulator !!

Output pada seven segment Jika P0.1 Ditekan :

1	2	3	4	5	6

Kesimpulan:

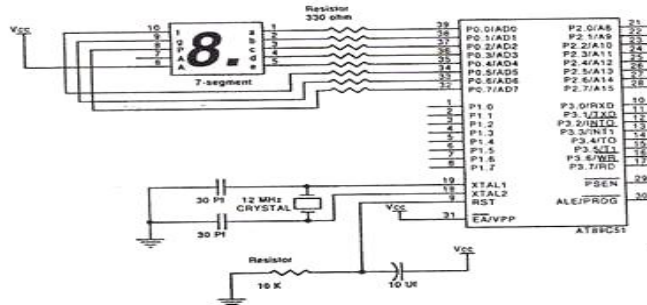
.....

Alat-alat :

- 1 Set IBM PC dengan Sistem Operasi Windows dan Software M-IDE51, dan ISP Flash Programmer, dan TOP View Simulator.

Bagian 2

Modul 3.2 Program 7 Segment (tanpa BCD) (Gambar lebih Jelas Hal V-5)



1. Ikutilah langkah-langkah pada **modul 4.2**,namun pada saat mensetting di TOP View Simulator ,pada external module setting jangan lupa **ceklist common anode** dan **ceklist 7 segment** serta **ceklist non multiplexed**.
2. Kemudian lakukan **setting selection of Port Lines and Number of Digits** lihat PORT yang dihubungkan sudah benar atau belum (P2.0 s/d P2.7). Selanjutnya ikuti seperti langkah sebelumnya.

PROGRAM	ANALISA
start : mov r7,#0ah	
mov dptr,#kata	
ulang : clr a	
movc a,@a+dptr	
mov p2,a	
acall delay	
inc dptr	
djnz r7,ulang	
sjmp start	
kata : db 0C0h, 0F9h, 0A4h, 0B0h, 99h, 92h, 82h, 0F8h, 80h, 90h	
delay : mov r0,#2fh	
lagi : djnz r2, lagi	
djnz r1, lagi	
djnz r0, lagi	
ret	
End	

Output pada seven segment Jika P0.1 Ditekan :

1	2	3	4	5	6	7	8	9	10

[illegible]

