PERCOBAAN 1 DISPLAY LED

TUJUAN:

- 1. Memahami rangkaian mikrokontroller untuk menghidupkan dan mematikan LED.
- 2. Memahami program assembly untuk menghidupkan dan mematikan LED.
- 3. Memahami beberapa instruksi assembly dasar, MOV, Setb, Clr, RL dan RR
- 4. Memahami pembuatan instruksi waktu tunda



Gambar 1.1 Rangkaian Display LED

Perhatikan pada gambar 1.1 tersebut. Delapan buah LED terhubung ke port 0, yang difungsikan sebagai output. Pada konfigurasi tersebut LED akan nyala bila diberi logika LOW '0' melalui port 0, dan LED akan padam bila diberi logika HIGH '1' melalui port 0.

Percobaan 1.1. Instruksi MOV

Pada percobaan 1.1 ini LED akan dihidupkan atau dimatikan dengan mengirimkan data tertentu pada port 0.

Untuk melakukan percobaan ini lakukan beberapa langkah sebagai berikut:

- 1. Hubungkan V_{CC} dengan power supply +5V, untuk mengaktifkan 8 buah LED
- 2. Hubungkan modul Microcontroller Trainer dengan power supply +5V
- 3. Hubungkan modul Microcontroller Trainer dengan rangkaian programmer
- 4. Buka Program M-IDE Studio for MCS-51, sebagai editor dan compiler program
- 5. Ketik program berikut ini

Org Oh

```
Start: Mov P0,#11110000b ; Isi P0 dengan 11110000
sjmp start ; lompat ke start
End
```

- 6. Simpanlah program yang anda ketik dan beri nama : prog11a.asm
- 7. Pada program MIDE tersebut pilih Build /F9 atau untuk melakukan kompilasi program dari *.asm ke *.hex.
- Lakukan pemrograman mikrokontroller dengan menggunakan Program ISP Software (Lihat Petunjuk Penggunaan)
- 9. Lakukan pengamatan pada LED
- 10. Gantilah data tersebut diatas dengan data seperti pada tabel berikut dan lakukan pencatatan LED mana yang padam.

Percobaan 1.2. Pembuatan Subrutine Waktu Tunda

Pada percobaan ini, 8 LED akan berkedip secara kontinu.

Untuk melakukan percobaan ini lakukan beberapa langkah sebagai berikut:

1. Hubungkan modul Microcontroller Trainer dengan power supply +5V

- 2. Hubungkan modul Microcontroller Trainer dengan rangkaian programmer
- 3. Buka Program M-IDE Studio for MCS-51, sebagai editor dan compiler program
- 4. Ketik program berikut ini:

```
Org Oh

Start: Mov PO,#11111111 ; Kirim data biner 1111111 ke PO

Call Delay ; Memanggil waktu tunda

Mov PO,#0000000b ; Kirim data biner 00000000 ke PO

Call Delay ; Memanggil waktu tunda

Sjmp start

;

Delay: mov R1,#255

Del1 : mov R2,#255

Del2 : djnz R2,del2

djnz R1,del1

ret

end
```

- 5. Simpanlah program yang anda ketik dan beri nama : prog12a.asm
- 6. Pada program MIDE tersebut pilih Build /F9 atau untuk melakukan kompilasi program dari *.asm ke *.hex.
- Lakukan pemrograman mikrokontroller dengan menggunakan Program ISP Software (Lihat Petunjuk Penggunaan)
- 8. Lakukan pengamatan pada LED
- 9. Gantilah data tersebut untuk mengedipkan sebuah LED, dua buah LED dan seterusnya.

Percobaan 1.3. Instruksi Setb dan Clr

Pada percobaan ini, sebuah LED atau lebih dapat dihidupkan atau dimatikan dengan perintah setb dan clr.

Untuk melakukan percobaan ini lakukan beberapa langkah sebagai berikut:

- 1. Hubungkan modul Microcontroller Trainer dengan power supply +5V
- 2. Hubungkan modul Microcontroller Trainer dengan rangkaian programmer
- 3. Buka Program M-IDE Studio for MCS-51, sebagai editor dan compiler program
- 4. Ketik program berikut ini:

Org 0h Start: Clr P0.0 ; Mengirimkan logika `0' ke P0.0 Sjmp start ; jump ke start end

- 5. Simpanlah program yang anda ketik dan beri nama : prog13a.asm
- 6. Pada program MIDE tersebut pilih Build /F9 atau untuk melakukan kompilasi program dari *.asm ke *.hex.
- 7. Lakukan pemrograman mikrokontroller dengan menggunakan Program ISP Software (Lihat Petunjuk Penggunaan)
- 8. Lakukan pengamatan pada LED
- 9. Gantilah data tersebut untuk menghidupkan LED yang lain: LED 2, LED 3, LED 4, LED 5, LED 6, LED 7 dan LED 8.
- 10. Lakukan percobaan berikut untuk mengedipkan LED 1, dengan mengetikka program berikut ini.

		Org	0h							
Start	::	Clr	P0.0	;	Kirim	logika	۰0،	ke	P0.0	
		Clr	P0.1	;	Kirim	logika	۰0،	ke	P0.1	
		Clr	P0.2	;	Kirim	logika	۰0،	ke	P0.2	
		Clr	P0.3	;	Kirim	logika	۰0،	ke	P0.3	
		Clr	P0.4	;	Kirim	logika	۰0،	ke	P0.4	
		Clr	P0.5	;	Kirim	logika	۰0،	ke	P0.5	
		Clr	P0.6	;	Kirim	logika	۰0،	ke	P0.6	
		Clr	P0.7	;	Kirim	logika	٬ ٥٬	ke	P0.7	
		Call	Dela	ıу	; Mema	anggil w	waktu	ı tu	nda	
		Setb	P0.0);	Kirin	n logika	a `1'	′ ke	P0.0	
		Setb	P0.1	;	Kirin	n logika	a `1'	′ ke	P0.1	
		Setb	P0.2	2;	Kirin	n logika	a `1'	′ ke	P0.2	
		Setb	P0.3	3;	Kirin	n logika	a `1'	′ ke	P0.3	
		Setb	P0.4	ŀ;	Kirin	n logika	a `1'	′ ke	P0.4	
		Setb	P0.5	5;	Kirin	n logika	a `1'	′ ke	P0.5	
		Setb	P0.6	5;	Kirin	n logika	a `1'	′ ke	P0.6	
		Setb	P0.7	';	Kirin	n logika	a `1'	′ ke	P0.7	
		Sjmp	star	t	; lomp	pat ke a	start	5		
Delay	:	mov	R1,#2	255						
Dell	:	mov	R2,#2	255						
Del2	:	djnz	R2,ċ	lel	.2					
		djnz	R1,ċ	lel	.1					
		ret								
		end								

11. Rencanakan program untuk mengedipkan dua buah LED, tiga buah LED dan seterusnya dengan instruksi Setb dan Clr.

Percobaan 1.4. Instruksi RL/ RR (Rotate Left dan Right)

Pada percobaan ini, sebuah LED akan berjalan dari kiri ke kanan dan sebaliknya, program ini memanfaatkan instruksi RL dan RR.

Untuk melakukan percobaan ini lakukan beberapa langkah sebagai berikut:

- 1. Hubungkan modul Microcontroller Trainer dengan power supply +5V
- 2. Hubungkan modul Microcontroller Trainer dengan rangkaian programmer
- 3. Buka Program M-IDE Studio for MCS-51, sebagai editor dan compiler program
- 4. Ketik program berikut ini:
 - Org OH Start: Mov A,#11111110b Mov R0,#7 Kiri: Mov P2,A Call Delay RL A DEC R0 CJNE R0,#0,Kiri Mov R0,#7 Kanan: Mov P2,A Call Delay RR A DEC R0 CJNE R0,#0,Kanan Sjmp Start Delay: mov R1,#255 Del1 : mov R2,#255 Del2 : djnz R2,del2 djnz R1,dell ret end
- 5. Simpanlah program yang anda ketik dan beri nama : prog14a.asm
- 6. Pada program MIDE tersebut pilih Build /F9 atau untuk melakukan kompilasi program dari *.asm ke *.hex.
- Lakukan pemrograman mikrokontroller dengan menggunakan Program ISP Software (Lihat Petunjuk Penggunaan)
- 8. Lakukan pengamatan pada LED
- 9. Lakukan perubahan pada program tersebut untuk menjalankan dua buah LED kekiri dan kekanan, tiga buah LED kekiri dan kekanan , dan seterusnya.