

APLIKASI MODUL DST-52 SEBAGAI JAM DIGITAL DENGAN INPUT DARI PC KEYBOARD DAN M1632 LCD SEBAGAI PENAMPIL

Pada artikel kali ini akan dibahas contoh bagaimana menggabungkan antara modul RTC-1287, modul LCD Hitachi M1632, dan PC Keyboard ke DST-52. Sebagai contoh aplikasi dari penggabungan ini adalah Jam Digital dengan inputan PC Keyboard, yang dimaksud menggunakan inputan PC Keyboard adalah PC Keyboard digunakan sebagai inputan untuk perubahan jam, menit dan detik pada Jam Digital.

Sebagai komponen utama dari jam digital ini adalah modul RTC-1287 yang terhubung pada modul DST-52. Modul RTC ini menggunakan IC DS12887. IC tersebut berfungsi sebagai Real Time Clock, yaitu sebagai jam digital yang terus bekerja selama kurang lebih **10 tahun walaupun tidak mendapat suplay power**. Data-data waktu tersimpan di memori non-volatile pada IC tersebut.

Data-data waktu disimpan pada register-register data DS12887. Register data dibagi menjadi register waktu dan register alarm. Selain register data, terdapat register kontrol yang berfungsi sebagai seting pengaturan kerja DS1287 ketika pertama kali diaktifkan, yaitu register A,B,C dan D. Setelah register-register kontrol diinisialisasi baru register-register waktu dan alarm dapat diakses.

Pada modul DST-52, modul RCT-1287 terletak pada alamat 6000H, maka jika register jam terletak pada alamat 04H, maka pengambilan data jam dapat dilakukan dengan mengambil data pada alamat 6000H+alamat register jam, yaitu 6004H. Pada waktu pertama kali digunakan modul RTC-1287 harus diinisialisasi terlebih dahulu. Contoh dari pengambilan data data waktu adalah seperti pada potongan program 1.

Potongan Program 1

```
1.  AmbilWaktu:
2.      2.          MOV     DPTR,#ALAMAT_RTC+REGC      ;Isi DPTR dgn alamat Reg C
3.      3.          MOVX   A,@DPTR                    ;Ambil nilai Reg C
4.      4.          ANL   A,#080H                      ;Periksa bit ke 7
5.      5.          JNB   A.7,AmbilWaktu              ;Jika bernilai 0, tunggu
6.      6.          CLR   A.7
7.      7.          MOVX   @DPTR,A
8.      8.          MOV   DPTR,#ALAMAT_RTC            ;Proses pengambilan nilai
9.      9.          MOV   DPL,#JAM                    ;waktu
10.     10.         MOVX  A,@DPTR
11.     11.         MOV   VAR_JAM,A
12.     12.         MOV   DPL,#MENIT                  ;Set alamat Reg Jam
13.     13.         MOVX  A,@DPTR                    ;Ambil nilai Jam
14.     14.         MOV   VAR_MENIT,A                ;Simpan pada variabel
15.     15.         MOV   DPL,#DETIK
16.     16.         MOVX  A,@DPTR
17.     17.         MOV   VAR_DETIK,A
18.     18.         RET
```

Pada potongan program 1, pada baris 1 sampai 7 terdapat rutin untuk memeriksa register C, bit ke 7, yaitu bit IRQF, yaitu bit flag interrupt request. Setelah data-data waktu diambil, kemudian disimpan pada variable-variabel penampung.

Nilai-nilai waktu tersebut dapat dirubah sesuai dengan kebutuhan. Untuk proses perubahan tersebut, dilakukan dengan jalan melakukan proses penulisan pada register-register waktu yang sesuai. Seperti pada potongan program 2

Potongan Program 2

1.	SetWaktu:			
2.	2.	MOV	DPTR,#ALAMAT_RTC	;Set alamat RTC->6000h
3.	3.	MOV	DPL,#JAM	;tambahkan dengan alamat jam
4.	4.	MOV	A,VAR_JAM	;isi accumulaor dengan nilai jam
5.	5.	MOVX	@DPTR,A	;proses penulisan nilai jam
6.	6.	MOV	DPL,#MENIT	
7.	7.	MOV	A,VAR_MENIT	
8.	8.	MOVX	@DPTR,A	
9.	9.	MOV	DPL,#DETIK	
10.	10.	MOV	A,VAR_DETIK	
11.	11.	MOVX	@DPTR,A	
12.	12.	RET		

Pada potongan program 2, terdapat contoh proses penulisan pada register jam, menit dan detik, yang berguna untuk merubah nilai waktu jam, menit dan detik

Untuk melakukan perubahan-perubahan nilai waktu tersebut digunakan PC Keyboard sebagai inputan nilai waktu, yaitu nilai jam, nilai menit, dan nilai detik. PC Keyboard ini terhubung ke modul DST-52, melalui konektor PC Keyboard DST-52. Nilai-nilai tersebut diinputkan secara berurutan mulai dari nilai jam, menit lalu detik. Nilai tersebut diinputkan mulai dari nilai puluhan baru nilai satuan. Tiap-tiap proses perubahan nilai waktu diakhiri dengan penekanan tombol Enter pada PC Keyboard. Tombol Esc berfungsi untuk membatalkan proses perubahan nilai waktu secara keseluruhan. Proses pengubahan nilai-nilai waktu pada DS-1287 terjadi setelah penekanan tombol enter yang menandakan pengisian nilai detik telah selesai.

Untuk dapat melakukan proses pengubahan nilai waktu, diperlukan rutin untuk mendeteksi terjadinya penekanan pada PC Keyboard. Setiap terjadi penekanan tombol PC Keyboard akan mengirimkan scan code tertentu, sesuai dengan tombol yang ditekan. Selain scan code, PC Keyboard juga akan mengirimkan kode F0H setiap kali tombol dilepas. Sebagai contoh apabila tombol "x" mempunyai scan code bernilai 22H, maka jika tombol "x" ditekan, scan code 22H akan dikirimkan ke modul DST-52. Jika tombol "x" ditekan terus maka data 22H akan ditekan terus menerus.

- • Tombol "x" ditekan sekali 22 F0 22
- • Tombol "x" ditekan agak lama lalu dilepas 22 22 22
F0 22

Sistem komunikasi data yang digunakan untuk mengirimkan scan code dari PC Keyboard ke DST-52 adalah system komunikasi serial sinkron, yaitu system komunikasi serial yang diiringi sinyal clock sebagai sinkronisasi untuk setiap pengiriman bit datanya. Proses pengiriman data tersebut diawali dengan bit start berlogika 0 dan dilanjutkan dengan 8 bit mulai dari LSB hingga MSB diakhiri dengan bit parity dan bit stop. Contoh proses pengambilan 8 bit data scan code terdapat pada potongan program 3

Potongan Program 3

1.	1.	Ambil_Scancode:
2.	2.	Push 00H

```

3.      3.      CLR      A                ;Tunggu ada Interrupt keyboard
4.      4.      JNB     KbdClock,$          ;Level high
5.      5.      Mov     R0,#08H
6.      6.      Ambil_Scancode2:           ;Ambil data dari P1.4 setiap
7.      7.      JB     KbdClock,$          ;kali interrupt keyboard transisi
8.      8.      MOV    C,KbdData          ;low
9.      9.      RR     A                ;Geser Acc A
10.     10.     MOV    ACC.7,C            ;Simpan data di bit 7 Acc A
11.     11.     JNB     KbdClock,$          ;Tunggu Int0 high
12.     12.     DJNZ   R0,Ambil_Scancode2 ;Lakukan 8x
13.     13.     JB     KbdClock,$          ;Buang Stop Bit
14.     14.     JNB     KbdClock,$          ;
15.     15.     JB     KbdClock,$          ;
16.     16.     JNB     KbdClock,$          ;
17.     17.     POP    00H
18.     18.     RET

```

Untuk pendeteksian pada modul DST-52 bahwa telah terjadi penekanan tombol pada PC Keyboard, maka digunakan system interupsi eksternal, yaitu INT0 yang terhubung pada bit clock PC Keyboard. Sehingga jika ada pengiriman bit clock maka akan membangkitkan interupsi pada modul DST-52, yang mengakibatkan modul DST-52 akan melakukan proses pembacaan PC Keyboard. Sebagai contoh rutin interupsi pada modul DST-52 adalah potongan program 4

Potongan Program 4

```

1.      1.      Ada_Keyboard:
2.      2.      CLR     EA
3.      3.      CLR     EX0
4.      4.      CLR     Esc_Flag
5.      5.      ACALL  Edit_Jam
6.      6.      JB     Esc_Flag,Ada_Key_ESC
7.      7.      ACALL  Edit_Minut
8.      8.      JB     Esc_Flag,Ada_Key_ESC
9.      9.      ACALL  Edit_Detik
10.     10.     Ada_Key_ESC:
11.     11.     ACALL  Ambilwaktu
12.     12.     Ada_Keyboard_Finis:
13.     13.     SETB   EA
14.     14.     SETB   EX0
15.     15.     RET

```

Setelah terjadi interupsi maka akan dilanjutkan dengan proses pengubahan nilai jam, menit lalu detik. Pada proses pengubahan ini tombol PC Keyboard yang berpengaruh hanyalah tombol angka, Enter dan ESC. Contoh program untuk mengubah waktu detik terdapat pada potongan program 5

Potongan Program 5

```

1.      1.      Edit_Detik:
2.      2.      Edit_Detikpuluhan:
3.      3.      JB     KbdClock,$          ;Tunggu clock keyboard
4.      4.      Acall  AmbilKeyboard      ;Ambil data dari keyboard, simpan di A
5.      5.      JNC   Edit_Detikbukanenter
6.      6.      MOV    B,A                ;Simpan nilai A pada B
7.      7.      MOV    R7,#'6'
8.      8.      MOV    R6,#'0'
9.      9.      LCALL  Filter            ;Cek apakah nilai A, antara 0-6
10.     10.     JC     Edit_Detikcek_Enter ;Jika tidak , apakah tombol
Enter/ESC?
11.     11.     MOV    A,B                ;Ambil nilai A
12.     12.     LCALL  Serial_Out         ;Kirim ke serial PC
13.     13.     MOV    A,B                ;Ambil nilai A
14.     14.     ANL   A,#0Fh            ;Ambil 4 bit LSB
15.     15.     SWAP  A                ;Putar nilai A, LSB->MSB

```

16.	16.	ANL	Var_Detik,#0Fh	;Ambil 4 bit LSB
17.	17.	ORL	Var_Detik,A	;Gabung nilai variable detik dengan
	isi A			
18.	18.	LCALL	Tampil_LCD	;Tampilkan ke LCD
19.	19.	LCALL	Delay_100ms	
20.	20.	CLR	C	
21.	21.		Edit_Detiksatuan:	
22.	22.	JB	KbdClock,\$;Tunggu clock keyboard
23.	23.	ACALL	AmbilKeyboard	;Ambil data dari keyboard, simpan di A
24.	24.	JNC	Edit_Detikbukanenter	
25.	25.	MOV	B,A	;Simpan nilai A pada B
26.	26.	MOV	R7,#'9'	
27.	27.	MOV	R6,#'0'	
28.	28.	LCALI	Filter	; Cek apakah nilai A antara 0-9
29.	29.	JC	Edit_Detikcek_Enter	;Jika tidak, apakah tombol
	Enter/ESC?			
30.	30.	MOV	A,B	;Ambil nilai A pada B
31.	31.	LCALL	Serial_Out	
32.	32.	MOV	A,B	
33.	33.	ANL	A,#0Fh	;Ambil 4 bit LSB
34.	34.	ANL	Var_Detik,#0f0h	;Ambil 4 bit MSB
35.	35.	ORL	Var_Detik,A	;Gabung nilai variabel detik dengan
	isi A			
36.	36.	LCALL	Tampil_LCD	;Tampilkan pada LCD
37.	37.	LCALL	Delay_100ms	
38.	38.	CLR	C	
39.	39.	JB	KbdClock,\$;Tunggu clock keyboard
40.	40.	SJMP	Edit_Detikpuluhan	;Kembali pada pengisian detik puluhan
41.	41.		Edit_Detikcek_Enter:	
42.	42.	ACALL	Cek_Enter	;Pengecekan tombol Enter
43.	43.	JNC	Edit_Detikcekescp	;Jika tidak, apakah ESC?
44.	44.	ACALL	Setwaktu	;Jika ya, maka simpan perubahan ke DS-1287
45.	45.	SJMP	Edit_Detikfinish	;Proses selesai
46.	46.		Edit_Detikcekescp:	
47.	47.	ACALL	Cek_Esc	;Pengecekan tombol ESC
48.	48.	JNB	Esc_Flag,Edit_Detikbukanenter	;Jika tidak, proses berulang
49.	49.	LCALL	Delay_500ms	;Jika ya, proses selesai tanpa
	menyimpan			
50.	50.	SJMP	Edit_Eetikout	
51.	51.		Edit_Detikbukanenter:	
52.	52.	LCALL	Delay_100ms	
53.	53.	CLR	C	
54.	54.	JB	KbdClock,\$;Tunggu clock keyboard
55.	55.	SJMP	Edit_Detikpuluhan	
56.	56.		Edit_Detikfinish:	
57.	57.	LCALL	delay_500ms	
58.	58.	ACALL	CLRSCR	
59.	59.		Edit_Detikout:	
60.	60.	CLR	C	
61.	61.	RET		

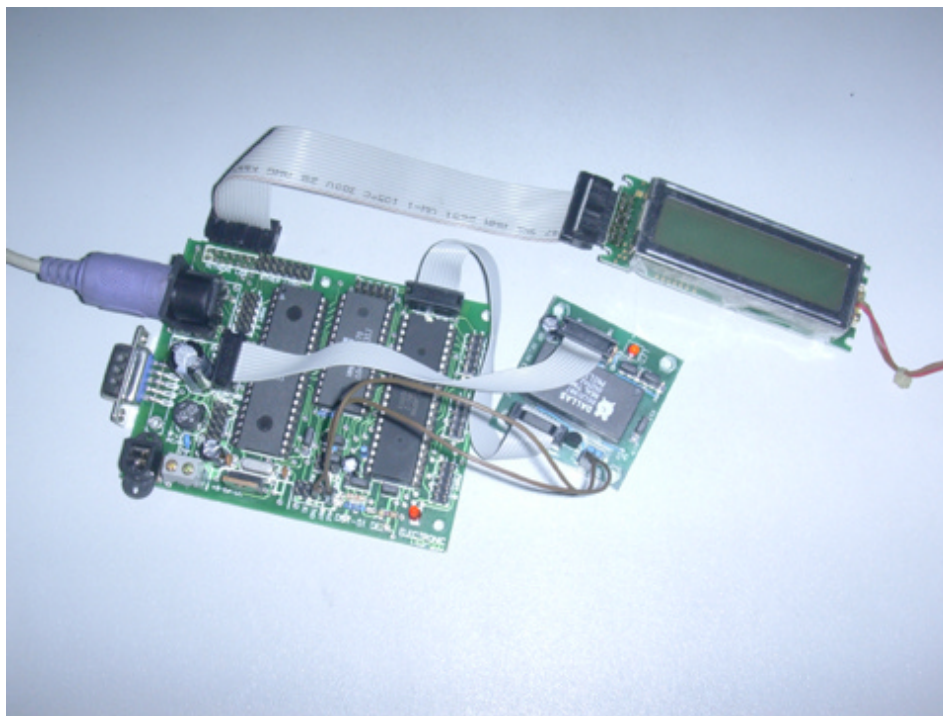
Sebagai media tampilan, digunakan modul LCD 16x2 karakter, yaitu modul Hitachi M1632 LCD. M1632 mempunyai dua buah register, yaitu register perintah dan register data. Register perintah digunakan untuk memberi perintah atau membaca status pada modul LCD, sedangkan register data digunakan untuk membaca atau menulis data dari atau ke DDRAM sesuai dengan alamat yang telah diatur sebelumnya. Modul M1632 ini terhubung pada modul DST-52 pada konektor Hitachi LCD. Pada modul DST-52 telah tersedia built-in sub rutin untuk mengakses modul LCD ini, sehingga memudahkan dalam pemakaiannya. Seperti pada modul RTC-1287,sebelum digunakan maka modul LCD harus diinisialisasi. Telah tersedia sub rutin dalam modul DST-52 untuk proses inisialisasi, pengiriman perintah, dan pengiriman data.

Proses penampilan jam digital ini adalah sebagai berikut, data waktu diambil dari modul RTC oleh modul DST-52, kemudian ditampilkan pada modul LCD, proses ini

akan diulang terus menerus. Jika terjadi penekanan tombol pada PC Keyboard maka proses akan dihentikan sementara untuk selanjutnya dilakukan proses pembacaan scan code, jika proses ini telah selesai dan terjadi penekanan tombol enter, maka data hasil pengubahan waktu tersebut akan disimpan ke modul RTC oleh modul DST, selanjutnya proses penampilan jam digital dilanjutkan kembali. Contoh program penampilan jam digital pada modul LCD adalah seperti pada potongan program 6

Potongan Program 6

1.	1.	Tampil_LCD:	
2.	2.	LCALL Posisi_Awal	;Set kursor LCD ke posisi mula2
3.	3.	MOV A,VAR_JAM	;Ambil nilai jam
4.	4.	LCALL KIRIM_LCD	;Tampilkan nilai jam pada LCD
5.	5.	MOV A,#':'	;Tampilkan karakter ":" pada LCD
6.	6.	LCALL KIRIM_KARAKTER	
7.	7.	MOV A,VAR_MENIT	;Ambil nilai menit
8.	8.	LCALL KIRIM_LCD	;Tampilkan nilai menit pada LCD
9.	9.	MOV A,#':'	;Tampilkan karakter ":" pada LCD
10.	10.	LCALL KIRIM_KARAKTER	
11.	11.	MOV A,VAR_DETIK	;Ambil nilai detik
12.	12.	LCALL KIRIM_LCD	;Tampilkan nilai detik pada LCD
13.	13.	LCALL Posisi_Awal	;Tampilkan karakter ":" pada LCD
14.	14.	RET	



Gambar 1
Instalasi Modul DST-52, M1632 LCD, Modul RTC-1287 dan PC Keyboard

Pada aplikasi ini hanya ditampilkan informasi jam, menit dan detik saja. Agar lebih bagus dapat ditambahkan tampilan informasi hari, tanggal dan alarm, karena pada modul RTC telah tersedia fasilitas tersebut. Program dan aplikasi ini dapat didownload di www.delta-electronic.com bagian application note.