

MICROCONTROLLER-BASED SOLAR CHARGER

As the sources of conventional energy deplete day by day, resorting to alternative sources of energy like solar and wind energy has become need of the hour.

Solar-powered lighting systems are already available in rural as well as urban areas. These include solar lanterns, solar home lighting systems, solar streetlights, solar garden lights and solar power packs. All of them consist of four components: solar photovoltaic module, rechargeable battery, solar charge controller and load.

In the solar-powered lighting system, the solar charge controller plays an important role as the system's overall success depends mainly on it.

It is considered as an indispensable link between the solar panel, battery and load.

The microcontroller-based solar charge controller described here has the following features:

1. Automatic dusk-to-dawn operation of the load
2. Built-in digital voltmeter (0V-20V range)
3. Parallel- or shunt-type regulation
4. Overcharge protection
5. System status display on LCD
6. Deep-discharge protection
7. Low battery lock
8. Charging current changes to 'pulsed' at full charge

9. Low current consumption
10. Highly efficient design based on microcontroller
11. Suitable for 10-40W solar panels for 10A load

The circuit of the solar charge controller is shown in Fig. 1. It comprises microcontroller AT89C2051, serial ana-

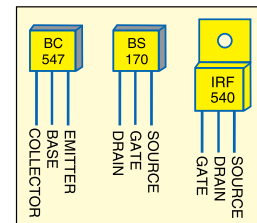


Fig. 2: Pin configurations of BC547, BS170 and IRF540

logue-to-digital converter ADC0831, optocoupler MCT2E, regulator 7805, MOSFETs BS170 and IRF540N, transistor BC547,

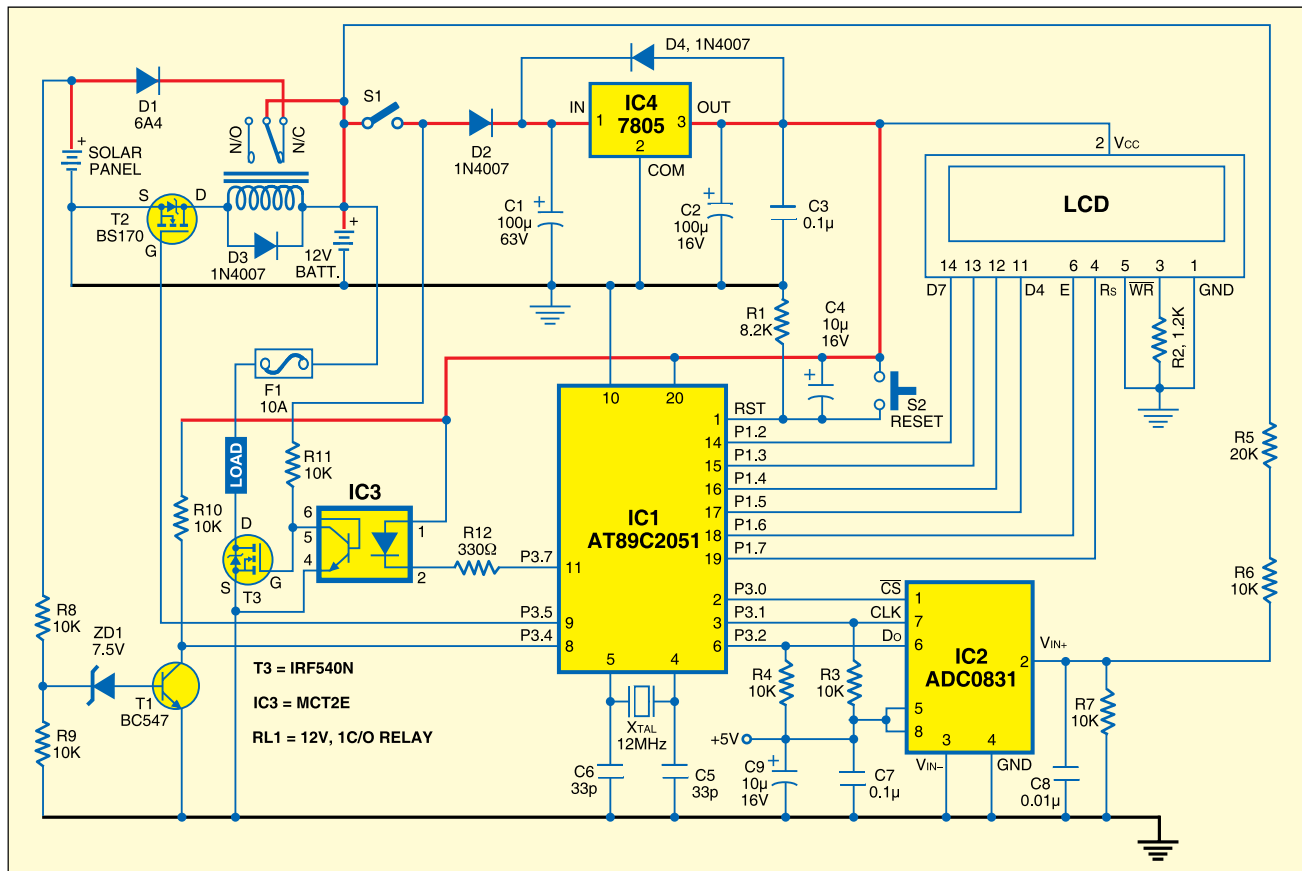


Fig. 1: Circuit of microcontroller-based solar charger

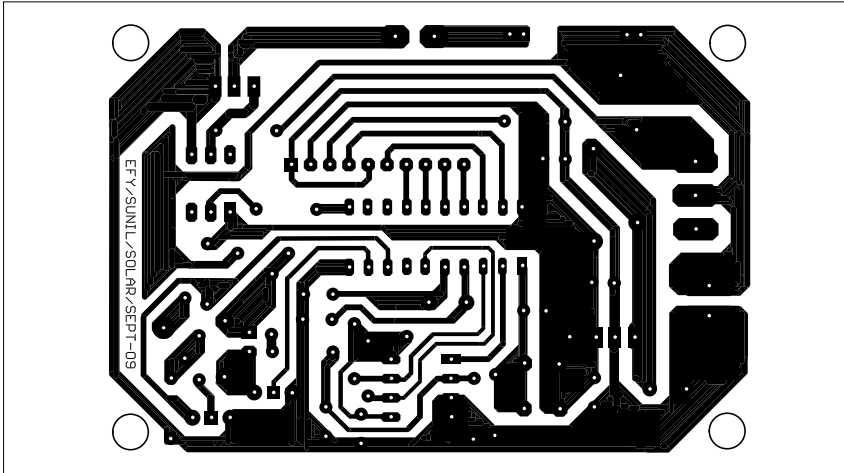


Fig. 3: A single-side, actual-size PCB layout for microcontroller-based solar charger

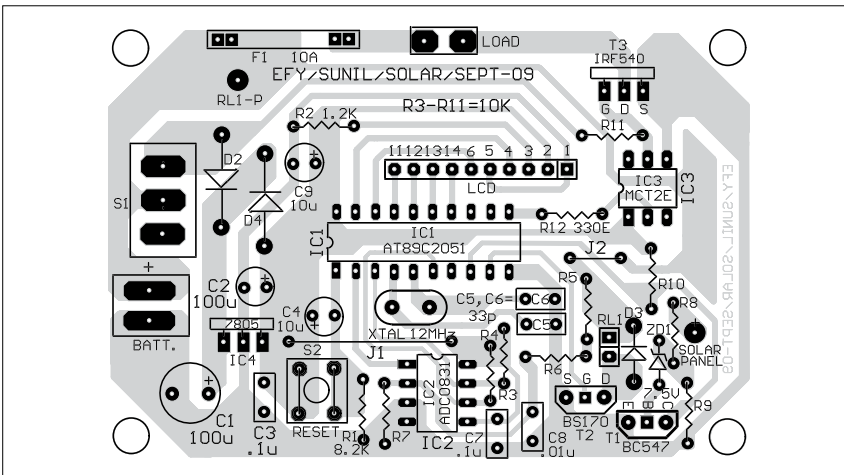


Fig. 4: Component layout for the PCB

LCD and a few discrete components. Component description is given below.

Microcontroller. Microcontroller AT89C2051 is the heart of the circuit. It is a low-voltage, high-performance, 8-bit microcontroller that features 2 kB of Flash, 128 bytes of RAM, 15 input/output (I/O) lines, two 16-bit timers/counters, a five-vector two-level interrupt architecture, a full-duplex serial port, a precision analogue comparator, on-chip oscillator and clock circuitry. A 12MHz crystal is used for providing the basic clock frequency. All I/O pins are reset to '1' as soon as RST pin goes high. Holding RST pin high for two machine cycles, while the oscillator is running, resets the device. Power-on reset is derived from resistor R1 and capacitor C4. Switch S2 is used for manual reset.

Serial ADC. The microcontroller monitors the battery voltage with the help of an analogue-to-digital converter. The ADC0831 is an 8-bit successive approximation analogue-to-digital converter with a serial I/O and very low conversion time of typically 32 µs. The differential analogue voltage input allows increase of the common-mode rejection and offsetting of the analogue zero input voltage. In addition, the voltage reference input can be adjusted to allow encoding of any smaller analogue voltage span to the full eight bits of resolution. It is available in an 8-pin PDIP package and can be interfaced to the microcontroller with only three wires.

LCD module. The system status and battery voltage are displayed on an LCD based on HD44780 controller. The backlight feature of the LCD makes it

PARTS LIST	
<i>Semiconductors:</i>	
IC1	- AT89C2051 microcontroller
IC2	- ADC0831 analogue-to-digital converter
IC3	- MCT2E optocoupler
IC4	- 7805, 5V regulator
T1	- BC547 npn transistor
T2	- BS170 n-channel MOSFET
T3	- IRF540 n-channel MOSFET
D1	- 6A4 rectifier diode
D2-D4	- 1N4007 rectifier diode
ZD1	- 7.5V zener diode
<i>Resistors (all 1/4-watt, ±5% carbon):</i>	
R1	- 8.2-kilo-ohm
R2	- 1.2-kilo-ohm
R3, R4, R6-R11	- 10-kilo-ohm
R5	- 20-kilo-ohm
R12	- 330-ohm
<i>Capacitors:</i>	
C1	- 100µF, 63V electrolytic
C2	- 100µF, 16V electrolytic
C3, C7	- 0.1µF ceramic disk
C4, C9	- 10µF, 16V electrolytic
C5, C6	- 33pF ceramic disk
C8	- 0.01µF ceramic disk
<i>Miscellaneous:</i>	
S1	- On/off switch
S2	- Push-to-on switch
RL1	- 12V, 1C/O relay
X _{TAL}	- 12MHz crystal
LCD	- 16×2 line display
Solar panel	- 10-40W
F1	- 10A fuse
	- 10-pin bergstik SIP connector (male & female)
<i>Note: 12V Battery and solar panel are not supplied by Kits 'n' Spare</i>	

readable even in low light conditions. The LCD is used here in 4-bit mode to save the microcontroller's port pins. Usually the 8-bit mode of interfacing with a microcontroller requires eleven pins, but in 4-bit mode the LCD can be interfaced to the microcontroller using only seven pins.

Solar panel. The solar panel used here is meant to charge a 12V battery and the wattage can range from 10 to 40 watts. The peak unloaded voltage output of the solar panel will be around 19 volts. Higher-wattage panels can be used with some modifications to the controller unit. Solar panel can be purchased from A.K. Electronics, Delhi (Ph: 011-41406775/76).

Rechargeable battery. The solar energy is converted into electrical energy and stored in a 12V lead-acid battery. The ampere-hour capacity ranges from 5 Ah to 100 Ah.

Dusk-to-dawn sensor. Normally, in

a solar-photovoltaic-based installation – for example, solar home lighting system, solar lantern or solar streetlight – the load (the light) is switched on at dusk (evening) and switched off at dawn (morning). During daytime, the load is disconnected from the battery and the battery is recharged with current from the solar panel. The microcontroller needs to know the presence of the solar panel voltage to decide whether the load is to be connected to or disconnected from the battery, or whether the battery should be in charging mode or discharging mode. A simple sensor circuit is built using a potential divider formed around resistors R8 and R9, zener diode ZD1 and transistor T1 for the presence of panel voltage.

Charge control. Relay RL1 connects the solar panel to the battery through diode D1. Under normal conditions, it allows the charging current from the panel to flow into the battery. When the battery is at full charge (14.0V), the charging current becomes ‘pulsed.’ To keep the overall current consumption of the solar controller low, normally-closed (N/C) contacts of the relay are used and the relay is normally in de-energised state.

Load control. One terminal of the load is connected to the battery through fuse F1 and another terminal of the load to an n-channel power MOSFET T3. MOSFETs are voltage-driven devices that require virtually no drive current. The load current should be limited to 10A. One additional MOSFET is connected in parallel for more than 10A load current.

Circuit description

Basically, there are two methods of controlling the charging current: series regulation and parallel (shunt) regulation. A series regulator is inserted between the solar panel and the battery. The series type of regulation ‘wastes’ a lot of energy while charging the battery as the control circuitry is always active and series regulator requires the input voltage to be 3-4 volts higher than the output voltage. The current and voltage output of a solar panel is governed by the angle of incidence of light, which keeps varying.

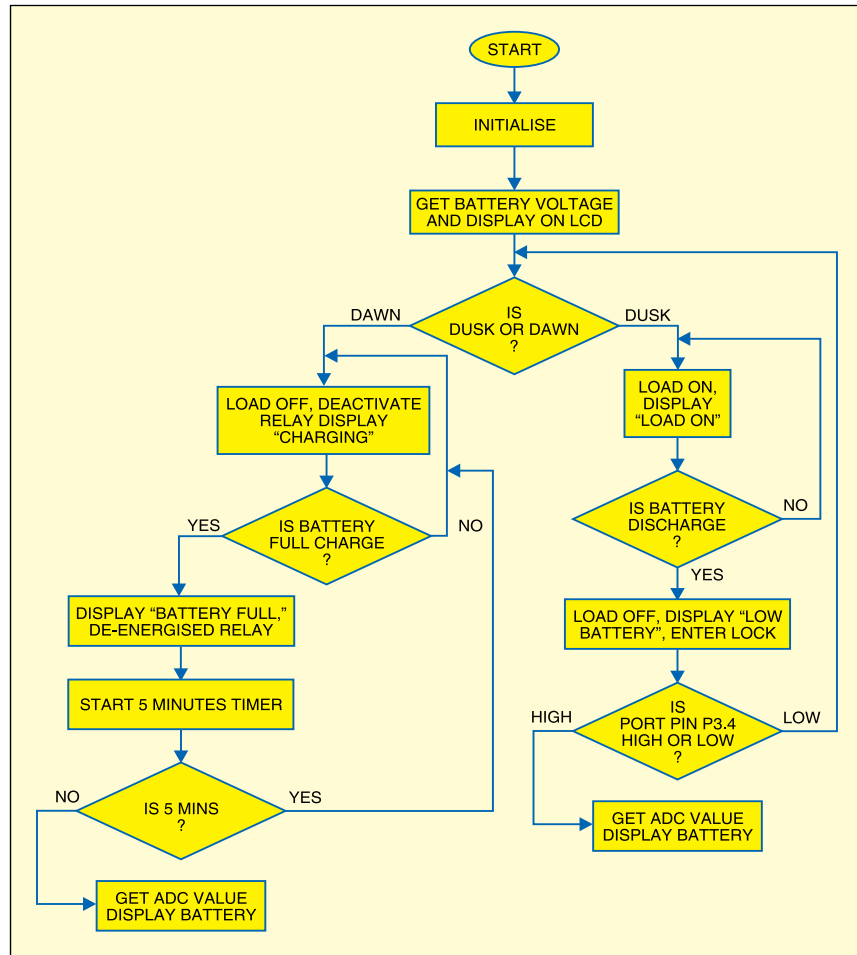


Fig. 5: Flow-chart of the source program

Parallel regulation is preferred in solar field. In parallel regulation, the control circuitry allows the charging current (even in mA) to flow into the battery and stop charging once the battery is fully charged. At this stage, the charging current is wasted by converting into heat (current is passed through low-value, high-wattage resistor); this part of the regulation dissipates a lot of heat.

In this project, we have used parallel regulation technique but instead of wasting the charging current as heat, we have made it pulsed and applied to the battery to keep the battery topped-up.

After power-on, the microcontroller reads the battery voltage with the help of the ADC and displays the values on the LCD. It monitors the input signal from the dusk-to-dawn sensor and activates the load or charging relay RL1 accordingly. The digital voltmeter

works up to 20V. As Vref of the ADC is connected to V_{CC} (5V), the input voltage to the ADC cannot exceed +5V. A potential divider is used at pin 2 of the ADC (IC2) using resistors R5, R6 and R7 to scale down the voltage from 0V-20V to 0V-05V. The ADC output is multiplied four times and displayed on the LCD as battery voltage.

When the solar panel voltage is present, the dusk-to-dawn sensor provides a signal to the microcontroller, which then displays ‘Charging’ message on the LCD. During charging, the battery voltage is continuously monitored. When the voltage reaches 14.0V, the microcontroller interrupts the charging current by energising the relay, which is connected to MOSFET BS170 (T2), and starts a 5-minute timer. During this stage, the LCD shows “Battery Full.”

After five minutes, the relay reconnects the panel to the battery. This

way, the charging current is pulsed at the intervals of five minutes and the cycle repeats until the panel voltage is present.

When the panel voltage falls below the zener diode (ZD1) voltage of the dusk-to-dawn sensor, the microcontroller senses this and activates the load by switching on MOSFET T3 via optocoupler IC3 and “Load On” message is displayed.

In this mode, the microcontroller monitors for low battery. When the battery voltage drops below 10 volts, the microcontroller turns off the load by switching off MOSFET T3 and “Battery Low—Load Off” message is displayed.

Normally, when the load is switched off, the battery voltage tends to rise back and the load oscillates between ‘on’ and ‘off’ states. To avoid this, the microcontroller employs a hysteresis control by entering into a ‘lock’ mode during low-battery state and comes out of the lock mode when the dusk-to-dawn sensor receives the panel voltage (the next morning). During lock mode, the

microcontroller keeps converting the ADC value and displays the battery voltage on the LCD.

Construction and testing

Pin configurations of transistor BC547, MOSFET BS170 and MOSFET IRF540 are shown in Fig. 2. An actual-size, single-side PCB for the microcontroller-based solar charger is shown in Fig. 3 and its component layout in Fig. 4. Wire the circuit on the PCB. Prior to inserting the programmed microcontroller into the PCB, check for soldering mistakes like shorts, and for proper connections using a multimeter. Mount power MOSFET IRF540N on a suitable heat-sink. Before switching on the controller unit, connect the leads of the battery, load and solar panel at appropriate places on the board.

Switch on the unit and the message “Solar Charge Controller-EFY” is displayed on the LCD for two seconds. The system status messages are displayed on line 1 of the LCD and the battery voltage is displayed on line 2. A small graphic representing the bat-

tery status is also displayed on line 2 of the LCD.

EFY note. 1. If the unit is switched on without the solar panel connected, the “Battery Low—Load Off” message is displayed irrespective of the battery voltage. The display changes to “Charging” as soon as the panel is connected.

2. There will be slight variation in the voltage displayed because of the tolerance levels of potential-divider resistors in the ADC section and Vref of the ADC being directly connected to V_{CC} (the output of 7805 has an accuracy of 2-5 per cent) instead of dedicated temperature-compensated voltage reference.

Software

The source program for the project is written in Assembly language and assembled using Metalink’s ASM51 assembler, which is freely available on the Internet for download. It is well commented for easy understanding and works as per the flow-chart shown in Fig. 5. The hex file ‘solar.hex’ is to be burnt into the microcontroller.

SOLAR.ASM

```
$MOD51
; LCD 4-BIT MODE CONNECTIONS
RS EQU P1.7 ;LCD REGISTER SELECT
LINE
EN EQU P1.6 ;LCD ENABLE LINE
DB4 EQU P1.5 ;
DB5 EQU P1.4 ;
DB6 EQU P1.3 ;
DB7 EQU P1.2 ;
; ADC0831 CONNECTIONS
CS EQU P3.0
CLK EQU P3.1
DO EQU P3.2
;INPUT & OUTPUT

DYI EQU P3.4 ; SOLAR PANEL VOLTAGE
SENSOR
CHG_RL EQU P3.5 ; CHARGING CONTROL
RELAY
LD_RL EQU P3.7 ; LOAD CONTROL RELAY
DSEG

ORG 0020H

VAL1: DS 1
VAL2: DS 1
VAL3: DS 1
ADC_VAL: DS 1
BUF: DS 1
CNT1: DS 1
CNT2: DS 1
IMG: DS 1
FLAGS: DS 1
OCF BIT FLAGS.0 ; OVER CHARGE FLAG
```

```
LBF BIT FLAGS.1 ; LOW BATT FLAG
CSEG

ORG 0000H
JMP MAIN

ORG 000BH ;Timer Interrupt0
JMP COUNTDOWN
MAIN: MOV SP,#50H
MOV P3,#0FFH
MOV P1,#0FFH
CLR CHG_RL
CLR LD_RL
LCALL PWR_DELAY
LCALL INIT
SETB CLK
SETB DO
SETB CS
SETB DYI
MOV VAL1,#00H
MOV VAL2,#00H
MOV VAL3,#00H
MOV FLAGS,#00H
LOADCHAR: MOV BUF,#40H
LCALL CMD
MOV DPTR,#RCHAR
REP: CLR A
MOVC A,@A+DPTR
JZ SCREEN1
MOV BUF,A
LCALL DAT
INC DPTR
SJMP REP
SCREEN1: MOV BUF,#80H
```

```
LCALL CMD
MOV DPTR,#MSG1
HERE: CLR A
MOVC A,@A+DPTR
JZ NEXT
MOV BUF,A
LCALL DAT
INC DPTR
SJMP HERE
NEXT: MOV BUF,#0C0H
LCALL CMD
MOV DPTR,#MSG2
HERE1: CLR A
MOVC A,@A+DPTR
JZ OVER
MOV BUF,A
LCALL DAT
INC DPTR
SJMP HERE1
OVER: LCALL ONE_SEC_DELAY
LCALL ONE_SEC_DELAY
LCALL CLEAR
MOV BUF,#0C0H
LCALL CMD
MOV DPTR,#MSG7
HERE2: CLR A
MOVC A,@A+DPTR
JZ CONVERT
MOV BUF,A
LCALL DAT
INC DPTR
SJMP HERE2
CONVERT:LCALL DDELAY
CLR CS ; INITIATE CONVERSION
```

```

SETB CLK
CLR CLK ; FIRST CLOCK
SETB CLK
CLR CLK ; SECOND CLOCK
MOV A,#00H ; CLEAR A
MOV R5,#08H ; 8 CLOCK PULSES
AGAIN: MOV C,DO
RLC A
SETB CLK
CLR CLK
DJNZ R5,AGAIN
SETB CS
MOV ADC_VAL,A
MOV B,#79D
MUL AB ; PRODUCT IN AB
MOV R1,B ; HIGH BYTE IN B
MOV R2,A ; LOW BYTE IN A
LCALL HEX2BCD
MOV VAL1,R7
MOV VAL2,R6
MOV VAL3,R5
LCALL SENDVAL2LCD
CHECK: JNB LBF,CHECK2 ; SEE IF ANY
FLAGS ARE SET ,i,e LOW BATT FLAG OR BATT
FULL FLAG
JB DYI,CONVERT
CHECK2:JNB OCF,PROCEED
JB DYI,NIGHT
SJMP CONVERT
PROCEED: JB DYI,NIGHT
CLR LD_RL ; OFF LOAD
CLR LBF ; CLEAR LOW BATT FLAG
MOV A,VAL2 ; SEE IF BATT.IS FULL
XRL A,#04H
JZ FULLCHG
CLR CHG_RL ; CONNECT BATT. TO
PANEL
MOV DPTR,#MSG4 ; DISPLAY CHARGING
MSG
MOV IMG,#00H
LCALL SENDSTAT2LCD
LJMP CONVERT
FULLCHG: SETB OCF ;SET OVERCHARGE
FLAG
SETB CHG_RL ;DISCONNECT BATT.FROM
PANEL
MOV TH0,#03CH ;START 5 MIN TIMER
HERE
MOV TL0,#0B0H ;DISCONNECT BATT FROM
PANEL
MOV CNT1,#200D
MOV CNT2,#30D
SETB ET0
SETB TR0
SETB EA
MOV DPTR,#MSG5 ; DISPLAY BATT.FULL
MSG
MOV IMG,#01H
LCALL SENDSTAT2LCD
LJMP CONVERT
NIGHT: CLR CHG_RL ; RECONNECT BATT.
TO PANEL
CLR TR0 ; STOP TIMER0 INCASE ITS
RUNNING
CLR OCF ; CLEAR OVER CHARGE FLAG
SETB LD_RL ; CONNECT LOAD TO BATT.
MOV A,VAL1
XRL A,#00H
JZ LOWBAT
MOV DPTR,#MSG3 ; DISPLAY LOAD ON
MSG
MOV IMG,#02H
LCALL SENDSTAT2LCD
LJMP CONVERT
LOWBAT: SETB LBF
CLR LD_RL ; DISCONNECT LOAD FROM
BATT.
MOV DPTR,#MSG6 ; DISPLAY BAT.OW AND

```

```

LOAD OFF MSG
MOV IMG,#03H
LCALL SENDSTAT2LCD
LJMP CONVERT
SENDVAL2LCD: MOV BUF,#0C7H
LCALL CMD
MOV A,VAL1
ORL A,#30H
MOV BUF,A
LCALL DAT
MOV A,VAL2
ORL A,#30H
MOV BUF,A
LCALL DAT
MOV BUF,#'. '
LCALL DAT
MOV A,VAL3
ORL A,#30H
MOV BUF,A
LCALL DAT
RET
SENDSTAT2LCD: MOV BUF,#080H
LCALL CMD
HERE3: CLR A
MOVC A,@A+DPTR
JZ PICT
MOV BUF,A
LCALL DAT
INC DPTR
SJMP HERE3
PICT: MOV BUF,#0CEH
LCALL CMD
MOV BUF,IMG
LCALL DAT
BACK: RET
;*****
;TIMER0 ISR (5 MINUTES TIMER)
;*****
COUNTDOWN: CLR TR0
MOV TH0,#03CH
MOV TL0,#0B0H
SETB TR0
DJNZ CNT1,BACK2
MOV CNT1,#200D
DJNZ CNT2,BACK2
CLR TR0 ; OFF 5 MIN TIMER
CLR ET0
CLR OCF ; CLEAR OVER CHARGE FLAG
CLR CHG_RL ; RE-CONNECT BATT TO
PANEL
BACK2: RETI
Hex2BCD: MOV R3,#00D
MOV R4,#00D
MOV R5,#00D
MOV R6,#00D
MOV R7,#00D
ACALL H2B
RET
H2B: MOV B,#10D
MOV A,R2
DIV AB
MOV R3,B ;
MOV B,#10 ; R7,R6,R5,R4,R3
DIV AB
MOV R4,B
MOV R5,A
CJNE R1,#00H,HIGH_BYTE ; CHECK FOR
HIGH_BYTE
SJMP ENDD
HIGH_BYTE: MOV A,#6
ADD A,R3
MOV B,#10
DIV AB
MOV R3,B
ADD A,#5
ADD A,R4
MOV B,#10
DIV AB

```

```

MOV R4,B
ADD A,#2
ADD A,R5
MOV B,#10
DIV AB
MOV R5,B
CJNE R6,#00D,ADD_IT
SJMP CONTINUE
ADD_IT: ADD A,R6
CONTINUE:MOV R6,A
DJNZ R1,HIGH_BYTE
MOV B,#10D
MOV A,R6
DIV AB
MOV R6,B
MOV R7,A
ENDD: RET
ONE_SEC_DELAY: MOV R0,#10D ; One second
delay routine
RZ3: MOV R1,#100D
RZ1: MOV R2,#250D
RZ2: NOP
NOP
DJNZ R2,RZ2
DJNZ R1,RZ1
DJNZ R0,RZ3
RET
PWR_DELAY: ; 15 mSec DELAY FOR LCD TO
INITIALIZE AFTER POWER-ON
MOV R4,#100D
H2: MOV R3,#250D
H1: DJNZ R3,H1
DJNZ R4,H2
RET
;*****LCD SUBROUTINES*****
CMD: PUSH ACC ; SAVE ACCUMULATOR
SETB EN
CLR RS ; SELECT SEND COMMAND
MOV A,BUF ; PUT DATA BYTE IN ACC
MOV C, ACC.4 ; LOAD HIGH NIBBLE ON
DATA BUS
MOV DB4,C ; ONE BIT AT A TIME
USING...
MOV C, ACC.5 ; BIT MOVE OPERATOINS
MOV DB5,C
MOV C, ACC.6
MOV DB6,C
MOV C, ACC.7
MOV DB7,C
CLR EN
NOP
SETB EN ; PULSE THE ENABLE LINE
MOV C, ACC.0 ; SIMILARLY, LOAD LOW
NIBBLE
MOV DB4,C
MOV C, ACC.1
MOV DB5,C
MOV C, ACC.2
MOV DB6,C
MOV C, ACC.3
MOV DB7,C
CLR EN
NOP
SETB EN ; PULSE THE ENABLE LINE
LCALL MSDELAY
POP ACC
RET
;*****LCD SUBROUTINES*****
DAT: PUSH ACC ; SAVE ACCUMULATOR
SETB EN
SETB RS ; SELECT SEND DATA
MOV A, BUF ; PUT DATA BYTE IN ACC
MOV C, ACC.4 ; LOAD HIGH NIBBLE ON
DATA BUS
MOV DB4,C ; ONE BIT AT A TIME
USING...
MOV C, ACC.5 ; BIT MOVE OPERATOINS
MOV DB5,C

```

```

MOV C, ACC.6
MOV DB6,C
MOV C, ACC.7
MOV DB7,C
CLR EN
NOP
SETB EN ; PULSE THE ENABLE LINE
MOV C, ACC.0 ; SIMILARLY, LOAD LOW
                NIBBLE

MOV DB4,C
MOV C, ACC.1
MOV DB5,C
MOV C, ACC.2
MOV DB6,C
MOV C, ACC.3
MOV DB7,C
CLR EN
NOP
SETB EN ; PULSE THE ENABLE LINENOP
LCALL MSDELAY
POP ACC
RET
;*****LCD SUBROUTINES*****
CLEAR: MOV BUF,#01H
LCALL CMD
RET
;*****LCD SUBROUTINES*****
DDELAY: MOV R5,#41D ; 4.1 mS DELAY
QT2:   MOV R6,#50D
QT1:   DJNZ R6,QT1

```

```

DJNZ R5,QT2
RET
;*****LCD SUBROUTINES*****
MSDELAY: MOV R5,#26D
QT22:   MOV R6,#50D
QT11:   DJNZ R6,QT11
DJNZ R5,QT22
RET
;*****LCD SUBROUTINES*****
INIT: MOV BUF,#30H ; FUNCTION
      SET - DATA BITS, LINES, FONTS
LCALL CMD
ACALL DDELAY ;INITIAL DELAY 4.1MSEC
MOV BUF,#30H ; FUNCTION SET -
      DATA BITS, LINES, FONTS
LCALL CMD
ACALL DDELAY ;INITIAL DELAY 4.1MSEC
MOV BUF,#30H ; FUNCTION SET - DATA
      BITS, LINES, FONTS
LCALL CMD
ACALL DDELAY ;INITIAL DELAY 4.1MSEC
MOV BUF,#28H ;2 LINES 5X7, 4-BIT
      MODE
LCALL CMD
MOV BUF,#0CH ; DISPLAY ON
LCALL CMD
MOV BUF,#01H ; CLEAR DISPLAY, HOME
      CURSOR
LCALL CMD
MOV BUF,#06H ; SET ENTRY MODE

```

```

LCALL CMD ; INCREMENT CURSOR RIGHT,
      NO SHIFT
RET

ORG 0320H
MSG1: DB ` SOLAR CHARGE `,00H
MSG2: DB ` CONTROLLER-EFY `,00H
MSG3: DB ` LOAD ON `,00H
MSG4: DB ` CHARGING `,00H
MSG5: DB ` BATTERY FULL `,00H
MSG6: DB `BAT.LOW-LOAD OFF`,00H
MSG7: DB `Volts: `,00H

ORG 0400H
RCHAR: DB 04D,31D,17D,31D,17D,31D,
      17D,31D ; CHARGING
      DB 04D,31D,31D,31D,31D,31D,
      31D,31D ; FULL
      DB 31D,31D,14D,04D,04D,14D,
      31D,31D ; LOAD
      DB 04D,31D,17D,17D,17D,17D,
      17D,31D ; LOW BATT
      DB 31D,31D,31D,31D,31D,31D,
      31D,31D
      DB 31D,31D,31D,31D,31D,31D,
      31D,31D
      DB 031D,31D,31D,31D,31D,31D,
      31D,31D,00H
END

```