

WATER-LEVEL CONTROLLER-CUM-MOTOR PROTECTOR



■ GURSHARANJEET SINGH KALRA

Many a time we forget to switch off the motor pushing water into the overhead tank (OHT) in our households. As a result, water keeps overflowing until we notice the overflow and switch the pump off. As the OHT is usually kept on the topmost floor, it is cumbersome to go up frequently to check the water level in the OHT.

Here's a microcontroller-based water-level controller-cum-motor protector to solve this problem. It controls 'on' and 'off' conditions of the motor depending upon the level of water in the tank. The status is displayed on an LCD module. The circuit also protects the motor from high voltages, low voltages, fluctuations of mains power and dry running.

Circuit description

Fig. 1 shows the circuit of the microcontroller-based water-level controller-cum-motor protector. It comprises operational amplifier LM324, microcontroller AT89C51, optocoupler PC817, regulator 7805, LCD module and a few discreet components.

The AT89C51 (IC2) is an 8-bit microcontroller with four ports (32 I/O lines), two 16-bit timers/counters, on-chip oscillator and clock circuitry. Eight pins of port-1 and three pins of port-3 are interfaced with data and control lines of the LCD module. Pins P3.0, P3.1 and P3.6 are connected to RS (pin 4), R/W (pin 5) and E (pin 6) of the LCD, respectively. Pin EA (pin 31) is strapped to Vcc for internal program executions. Switch S2 is used for backlight of the LCD module.

Power-on-reset is achieved by connecting capacitor C8 and resistor R14

to pin 9 of the microcontroller. Switch S1 is used for manual reset.

The microcontroller is operated with a 12MHz crystal. Port pins P2.0 through P2.2 are used to sense the water level, while pins P2.3 and P2.4 are used to sense the under-voltage and over-voltage, respectively. Pin P3.4 is used to control relay RL1 with the help of optocoupler IC3 and transistor T5 in the case of under-voltage, over-voltage and different water-level conditions. Relay RL1 operates off a 12V supply. Using switch S3, you can manually switch on the motor.

The LM324 (IC1) is a quad operational amplifier (op-amp). Two of its op-amps are used as comparators to detect under- and over-voltage. In normal condition, output pin 7 of IC1 is low, making pin P2.3 of IC2 high. When the voltage at pin 6 of N1 goes below the set reference voltage at pin 5 (say, 170 volts), output pin 7 of N1 goes high. This high output makes pin P2.3 of IC2 low, which is sensed by the microcontroller and the LCD module shows 'low voltage.'

In normal condition, pin 1 of N2 is high. When the voltage at pin 2 of N2 goes above the set voltage at pin 3, output pin 1 of N2 goes low. This low signal is sensed by the microcontroller and the LCD module shows 'high voltage.'

Presets VR1 and VR2 are used for calibrating the circuit for under- and over-voltage, respectively.

The AC mains is stepped down by transformer X1 to deliver a secondary output of 12V at 500 mA. The transformer output is rectified by a full-wave bridge rectifier comprising diodes D5 through D8, filtered by capacitor C2, and used for the under- and over-voltage detection circuitry.

PARTS LIST

Semiconductors:

IC1	- LM324 quad op-amp
IC2	- AT89C51 microcontroller
IC3	- PC817 optocoupler
IC4	- 7805, 5V regulator
T1-T4	- BC548 npn transistor
T5	- SL100 npn transistor
D1-D14	- 1N4007 rectifier diode

Resistors (all 1/4-watt, $\pm 5\%$ carbon):

R1, R2, R7,	
R11, R12	- 1-kilo-ohm
R3, R9	- 560-kilo-ohm
R4, R5, R8	- 2.7-kilo-ohm
R6	- 330-ohm
R10	- 470-ohm
R13	- 100-ohm
R14	- 10-kilo-ohm
R15-R17	- 100-kilo-ohm
R18-R20	- 2.2-kilo-ohm
R21, R22	- 33-ohm
RNW1	- 10-kilo-ohm resistor network
VR1, VR2	- 470-ohm preset
VR3	- 10-kilo-ohm preset

Capacitors:

C1-C3	- 1000 μ F, 35V electrolytic
C4	- 220 μ F, 16V electrolytic
C5, C6	- 33pF ceramic disk
C7	- 100 μ F, 35V electrolytic
C8	- 10 μ F, 16V electrolytic

Miscellaneous:

X1	- 230 AC primary to 12V, 500mA secondary transformer
RL1	- 12V, 1C/O relay
X _{TAL}	- 12MHz crystal
S1	- Push-to-on switch
S2, S3	- On/off switch
	- LCD module (1 \times 16)

The transformer output is also rectified by a full-wave bridge rectifier comprising diodes D1 through D4, filtered by capacitor C1 and regulated by IC4 to deliver regulated 5V for the circuit.

When water in the tank rises to come in contact with the sensor, the base of transistor BC548 goes high. This high signal drives transistor BC548 into saturation and its collector goes low. The low signal is sensed by

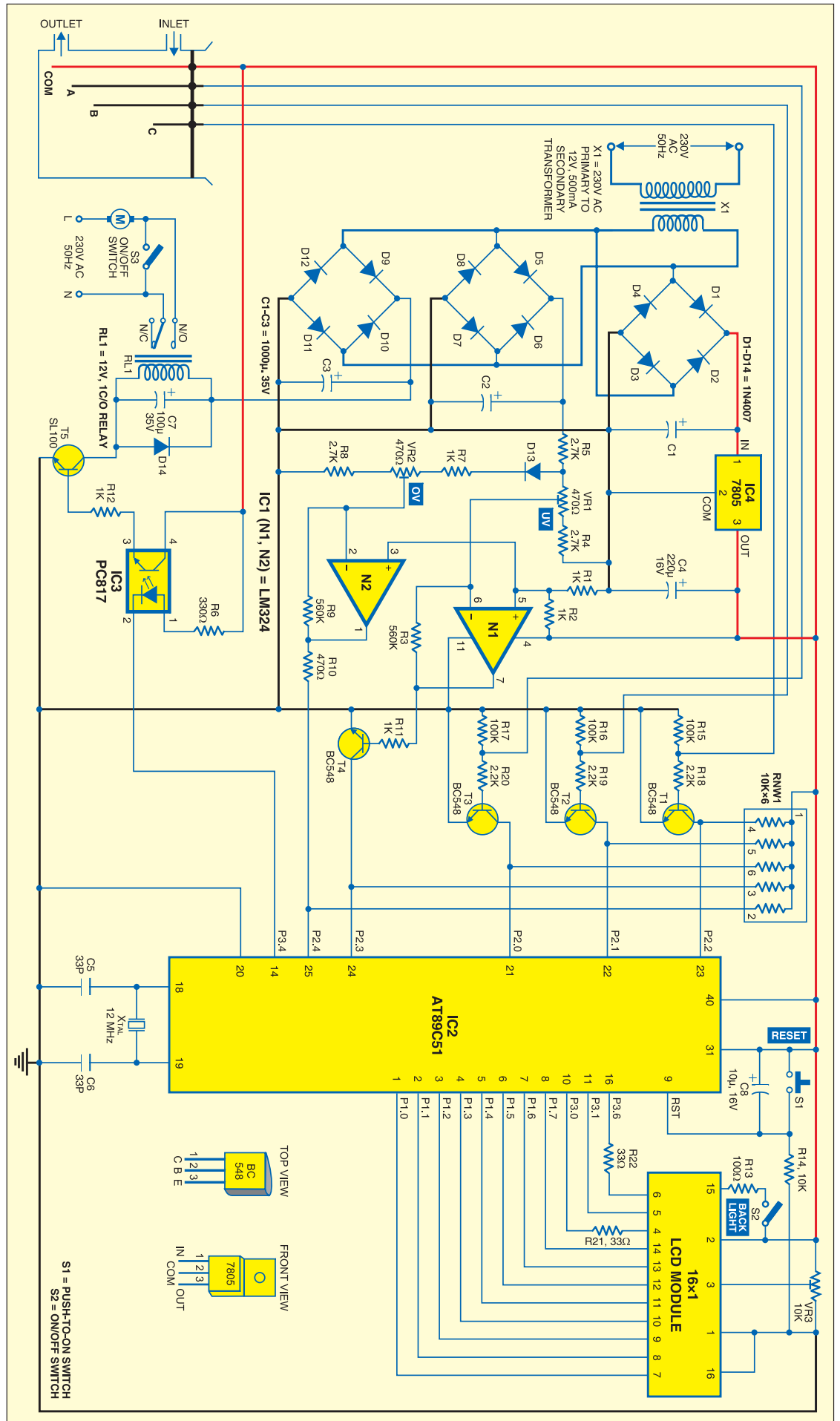
port pins of microcontroller IC2 to detect empty tank, dry sump and full tank, respectively.

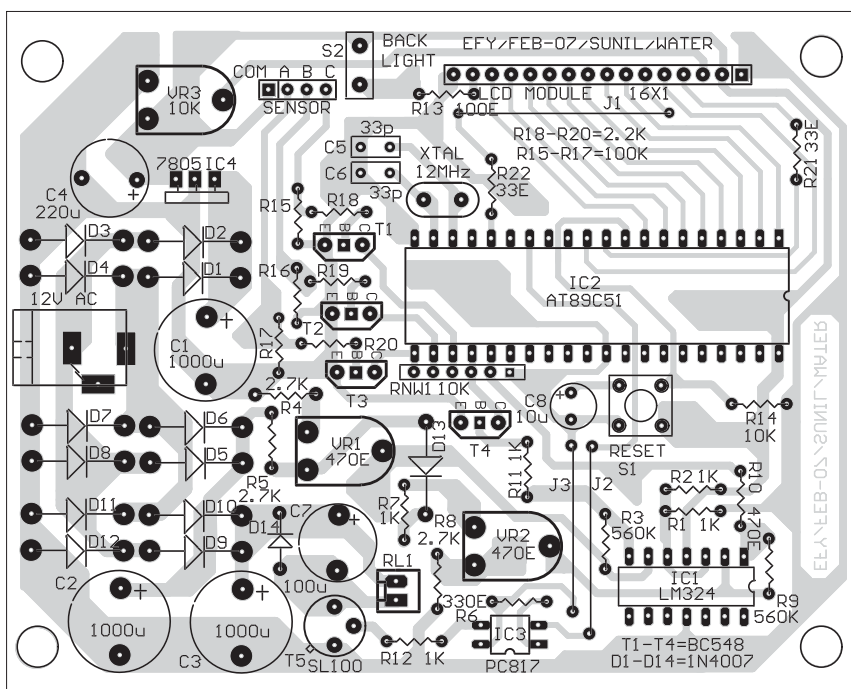
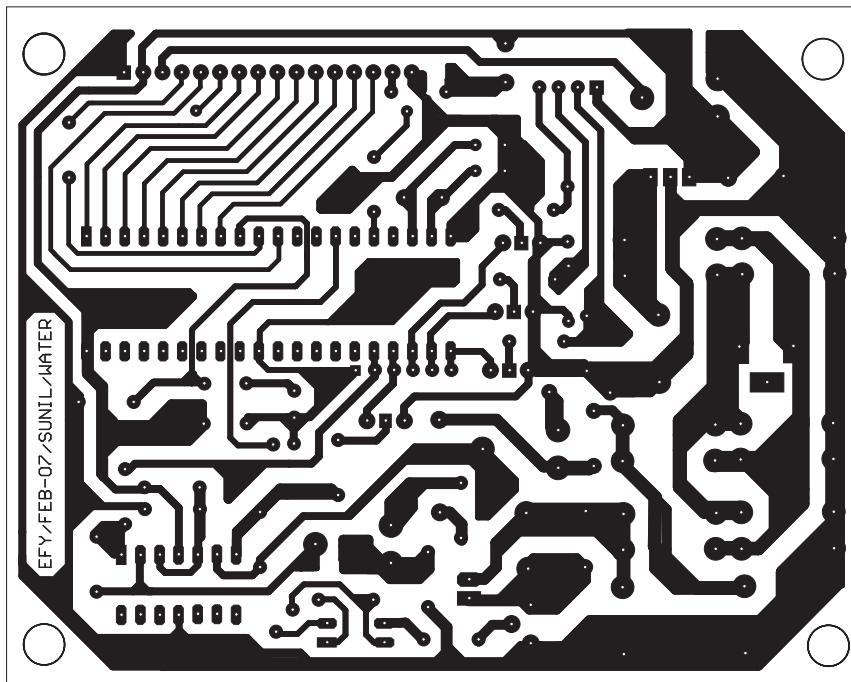
An actual-size, single-side PCB for the water-level controller-cum-motor protector (Fig. 1) is shown in Fig. 2 and its component layout in Fig. 3.

Operation

When water in the tank is below sensor A, the motor will switch on to fill water in the tank. The LCD module will show 'motor on.' The controller is programmed for a 10-minute time interval to check the dry-run condition of the motor. If water reaches sensor B within 10 minutes, the microcontroller comes out of the dry-run condition and allows the motor to keep pushing water in the tank.

The motor will remain 'on' until water reaches sensor C. Then it will stop automatically and the microcontroller will go into the standby mode. The LCD module will show 'tank full' followed by 'standby mode' after a few seconds. The 'standby mode' message is displayed until water





in the tank goes below sensor A.

In case water does not reach sensor B within 10 minutes, the microcontroller will go into the dry-running mode and stop the motor for 5 minutes, allowing it to cool down. The LCD module will show 'dry-sump1.'

After five minutes, the microcontroller will again switch on the motor for 10 minutes and check the status at sensor B. If water is still below sensor B, it will go into the dry-running mode and the LCD module will show 'dry-sump2.'

The same procedure will repeat,

and if the dry-run condition still persists, the display will show 'dry-sump3' and the microcontroller will not start the motor automatically. Now you have to check the line for water and manually reset the microcontroller to start operation.

In the whole procedure, the microcontroller checks for high and low voltages. For example, when the voltage is high, it will scan for about two seconds to check whether it is a fluctuation. If the voltage remains high after two seconds, the microcontroller will halt running of the motor. Now it will wait for the voltage to settle down. After the voltage becomes normal, it will still check for 90 seconds whether the voltage is normal or not. After normal condition, it will go in the standby mode and start the aforementioned procedure.

Practical applications

This controller is useful for single-phase operated motor-pumps and the pumps that suck water from the ground water tank. A small push-to-off manual switch in series with sensor A can also make it useful for pumps that suck water from Jal Board's supply. Because of the particular timing of this water supply, the controller must be switched on within the timing of the water supply and switched off when water is not being supplied.

When the controller is 'on' during the supply timings, it will wait for the tank to get empty before starting the motor. However, you can also start the motor using the pushbutton. The motor will turn on ignoring the status of the water level and will go through the aforementioned procedure.

Sensor positions in the tank

Four non-corrosive metallic sensors are installed in the tank as shown in Fig. 1. Sensor COM is connected to Vcc supply from the circuit. Sensor A detects the empty tank to start the motor. Sensor B detects dry-running condition of the motor and sensor C detects the full tank to stop the motor. Make sure that sensor B is around 2 cm above sensor A to check the dry-

running condition properly.

Calibration

Care must be taken when calibrating for under- and over-voltages. Always calibrate when the relay is in 'on' position. If you calibrate in the standby mode, it will trip at a voltage nearly

10 volts lower than the set voltage due to the loading effect.

Software

The source code is written in Assembly language and assembled using 8051 cross-assembler. The generated Intel hex code is burnt into

microcontroller AT89C51 using a suitable programmer. The software is well-commented and easy to understand. All the messages are displayed on the LCD module.

EFY note. All the software files of this article have been included in this month's EFY-CD.

WATER.LST

0000 02002F	1	\$MOD51		00C7 C2B1	53		CLR P3.1
	2	START:	LJMP MAIN_PGR ;GO TO MAIN	00C9 D2B7	54		SETB P3.7
			PROGRAMME	00CB C2B7	55		CLR P3.7
002F	3	ORG 002FH		00CD 22	56		RET
002F 7438	4	MAIN_PGR:	MOV A, #38H ;INITIALIZE LCD				;GO TO
0031 1200A0	5		LCALL WR_LCD	00F0	57		PROGRAMME
0034 740F	6		MOV A, #0FH	00F0 C2B7	58	SETT:	ORG 00F0H
0036 1200A0	7		LCALL WR_LCD	00F2 7590FF	59		CLR P3.7
0039 7406	8		MOV A, #06H				MOV P1, #0FFH ;SET PORT1 FOR
003B 1200A0	9		LCALL WR_LCD				INPUT
003E 7401	10		MOV A, #01H	00F5 00	60		;DELAY
0040 1200A0	11		LCALL WR_LCD	00F6 C2B0	61		NOP
0043 7480	12		MOV A, #80H	00F8 D2B1	62		CLR P3.0
0045 1200A0	13		LCALL WR_LCD	00FA C2B7	63	L1:	SETB P3.1
0048 7453	14		MOV A, #53H ;WRITE DATA	00FC D2B7	64		CLR P3.7
			TO LCD	00FE 2097F9	65		SETB P3.7
004A 1200C0	15		LCALL LCD_RAM ;i.e., "STANDBY-				JB P1.7,L1 ;IF NOT READY
			MODE"	0101 C2B7	66		JUMP TO 00FA H
004D 7454	16		MOV A, #54H	0103 22	67		RET
004F 1200C0	17		LCALL LCD_RAM				;BACK TO
0052 7441	18		MOV A, #41H	010A	68		PROGRAMME
0054 1200C0	19		LCALL LCD_RAM	010A D276	69	SCAN:	ORG 010AH
0057 744E	20		MOV A, #4EH	010C D277	70		SETB 76H ;SET USER FLAGS
0059 1200C0	21		LCALL LCD_RAM	010E D278	71		SETB 77H
005C 7444	22		MOV A, #44H	0110 75A0FF	72		SETB 78H
005E 1200C0	23		LCALL LCD_RAM				MOV P2, #0FFH ;SET PORT2 FOR
0061 7442	24		MOV A, #42H	0113 30A420	73	L4:	INPUT
0063 1200C0	25		LCALL LCD_RAM				;IF H/V THEN
0066 7459	26		MOV A, #59H	0116 30A372	74		GOTO 0136 H
0068 1200C0	27		LCALL LCD_RAM				;IF L/V THEN
006B 74B0	28		MOV A, #0B0H	0119 30A0F7	75		GOTO 018B H
006D 1200C0	29		LCALL LCD_RAM				;SCAN FOR
0070 74C0	30		MOV A, #0C0H ;JUMP TO 9TH				TANK TO BE
			POSITION	011C 8012	76		EMPTY
0072 1200A0	31		LCALL WR_LCD ;OR SECOND	011E 120750	77	LOOP:	;GOTO 0130 H
			LINE				;CALL 10 MIN.
0075 744D	32		MOV A, #4DH ;ENTER DATA	0121 307867	78		TIMER
			AGAIN	0124 30770F	79		;L/V THEN
0077 1200C0	33		LCALL LCD_RAM	0127 20765E	80		GOTO 018B H
007A 744F	34		MOV A, #4FH				;H/V THEN GOTO
007C 1200C0	35		LCALL LCD_RAM	012A 20A2FD	81		0136 H
007F 7444	36		MOV A, #44H				;DRY SUMP
0081 1200C0	37		LCALL LCD_RAM	012D 020260	82	LOOP1:	THEN GOTO
0084 7445	38		MOV A, #45H				0188 H
0086 1200C0	39		LCALL LCD_RAM	0130 C2B5	83	L130:	;WAIT UNTILL
0089 02010A	40		LJMP SCAN ;GO TO 010AH	0132 0202D0	84		TANK FULL
00A0	41		ORG 00A0H				;GOTO TANK
00A0 1200F0	42	WR_LCD:	LCALL SETT ;CHECK FOR	0135 00	85		FULL DISPLAY
			READY STATUS	0136 1201D8	86	L2:	;START MOTOR
			OF LCD				;DISPLAY
00A3 F590	43		MOV P1,A ;MOVE CONTENTS	0139 20A4D7	87		MOTOR ON
			OF A TO PORT 1				;BLANK SPACE
00A5 C2B0	44		CLR P3.0 ;WRITE				;WAIT FOR 2
			COMMANDS TO				SECONDS
			LCD				;STILL H/V
00A7 C2B1	45		CLR P3.1	013C D2B5	88	LOOP2:	THEN GOTO
00A9 D2B7	46		SETB P3.7	013E 00	89		0113 H
00AB C2B7	47		CLR P3.7	013F 00	90		;H/V THEN OFF MOTOR
00AD 22	48		RET ;RETURN TO	0140 1202B3	91		
			PROGRAMME	0143 7448	92		
00C0	49		ORG 00C0H				
00C0 1200F0	50		LCD_RAM: LCALL SETT ;CHECK	0145 1200C0	93		LCALL LCD_RAM
			READY STATUS	0148 7449	94		MOV A, #49H
			OF LCD	014A 1200C0	95		LCALL LCD_RAM
00C3 F590	51		MOV P1,A ;MOVE	014D 7447	96		MOV A, #47H
			CONTENTS OF A	014F 1200C0	97		LCALL LCD_RAM
			TO PORT 1	0152 7448	98		MOV A, #48H
00C5 D2B0	52		SETB P3.0 ;WRITE TO	0154 1200C0	99		LCALL LCD_RAM
			DISPLAY RAM	0157 74B0	100		MOV A, #0B0H
			OF LCD	0159 1200C0	101		LCALL LCD_RAM

015C 7456	102	MOV A, #56H		023A 74B0	176	MOV A, #0B0H	
015E 1200C0	103	LCALL LCD_RAM		023C 1200C0	177	LCALL LCD_RAM	
0161 744F	104	MOV A, #4FH		023F 7453	178	MOV A, #53H	
0163 1200C0	105	LCALL LCD_RAM		0241 1200C0	179	LCALL LCD_RAM	
0166 744C	106	MOV A, #04CH		0244 7455	180	MOV A, #055H	
0168 1200C0	107	LCALL LCD_RAM		0246 1200C0	181	LCALL LCD_RAM	
016B 74C0	108	MOV A, #0C0H	;MOVE TO 9TH CHARACTER	0249 744D	182	MOV A, #04DH	
016D 1200A0	109	LCALL WR_LCD		024B 1200C0	183	LCALL LCD_RAM	
0170 7454	110	MOV A, #54H		024E 7450	184	MOV A, #50H	
0172 1200C0	111	LCALL LCD_RAM		0250 1200C0	185	LCALL LCD_RAM	
0175 7441	112	MOV A, #41H		0253 74C0	186	MOV A, #0C0H	;MOVE TO 9TH CHARACTER OF LCD
0177 1200C0	113	LCALL LCD_RAM					
017A 7447	114	MOV A, #47H		0255 1200A0	187	LCALL WR_LCD	
017C 1200C0	115	LCALL LCD_RAM		0258 7431	188	MOV A, #31H	;START WRITING AGAIN
017F 7445	116	MOV A, #45H		025A 1200C0	189	LCALL LCD_RAM	
0181 1200C0	117	LCALL LCD_RAM		025D 020300	190	LJMP L300	;GOTO 0300 H
0184 020490	118	LJMP CHK_HV	;GOTO 0490 H	0260 D2B5	191	TANK_FULL:SETB P3.5	;STOP MOTOR
0187 00	119	NOP	;BLANK SPACE	0262 00	192	NOP	;BLANK SPACES FOR FURTHER EXPANSION
0188 020226	120	L5: LJMP M_STOP	;GOTO 0226 H				
018B 1201D8	121	L3: LCALL DLY_2SEC	;WAIT FOR 2 SECONDS	0263 00	193	NOP	
018E 20A382	122	JB P2.3,L4	;VOLTAGE OK THEN GOTO 0113 H	0264 00	194	NOP	
0191 D2B5	123	LOOP3:SETB P3.5	;STOP MOTOR IF ON	0265 00	195	NOP	
0193 1202B3	124	LCALL INI_LCD	;INITIALIZE LCD	0266 00	196	NOP	
0196 744C	125	MOV A, #04CH	;DISPLAY LOW-VOLTAGE	0267 00	197	NOP	
0198 1200C0	126	LCALL LCD_RAM		0268 00	198	NOP	
019B 744F	127	MOV A, #04FH		0269 1202B3	199	LCALL INI_LCD	;INITIALIZE LCD
019D 1200C0	128	LCALL LCD_RAM		026C 7454	200	MOV A, #54H	;WRITE TO DISPLAY RAM OF LCD
01A0 7457	129	MOV A, #57H		026E 1200C0	201	LCALL LCD_RAM	
01A2 1200C0	130	LCALL LCD_RAM		0271 7441	202	MOV A, #41H	
01A5 74B0	131	MOV A, #0B0H		0273 1200C0	203	LCALL LCD_RAM	
01A7 1200C0	132	LCALL LCD_RAM		0276 744E	204	MOV A, #4EH	
01AA 7456	133	MOV A, #56H		0278 1200C0	205	LCALL LCD_RAM	
01AC 1200C0	134	LCALL LCD_RAM		027B 744B	206	MOV A, #4BH	
01AF 744F	135	MOV A, #04FH		027D 1200C0	207	LCALL LCD_RAM	
01B1 1200C0	136	LCALL LCD_RAM		0280 74B0	208	MOV A, #0B0H	
01B4 744C	137	MOV A, #04CH		0282 1200C0	209	LCALL LCD_RAM	
01B6 1200C0	138	LCALL LCD_RAM		0285 7446	210	MOV A, #46H	
01B9 7454	139	MOV A, #54H		0287 1200C0	211	LCALL LCD_RAM	
01BB 1200C0	140	LCALL LCD_RAM		028A 7455	212	MOV A, #55H	
01BE 74C0	141	MOV A, #0C0H	;GOTO 9TH CHARACTER	028C 1200C0	213	LCALL LCD_RAM	
01C0 1200A0	142	LCALL WR_LCD		028F 744C	214	MOV A, #4CH	
01C3 7441	143	MOV A, #41H	;START DISPLAY AGAIN	0291 1200C0	215	LCALL LCD_RAM	
01C5 1200C0	144	LCALL LCD_RAM		0294 74C0	216	MOV A, #0C0H	;GOTO 9TH CHARACTER OF LCD
01C8 7447	145	MOV A, #47H		0296 1200A0	217	LCALL WR_LCD	
01CA 1200C0	146	LCALL LCD_RAM		0299 744C	218	MOV A, #4CH	;START DISPLAYING AGAIN
01CD 7445	147	MOV A, #45H		029B 1200C0	219	LCALL LCD_RAM	
01CF 1200C0	148	LCALL LCD_RAM		029E 1201D8	220	LCALL DLY_2SEC	;DISPLAY IT FOR 2 SECONDS
01D2 0204B0	149	LJMP CHK_LV	;GOTO 04B0 H	02A1 020000	221	LJMP START	;GOTO STANDBY MODE
01D5 00	150	NOP		02B3	222	ORG 02B3H	
01D6 00	151	NOP		02B3 7438	223	INI_LCD:MOV A, # 38H	
01D7 00	152	NOP		02B5 1200A0	224	LCALL WR_LCD	
01D8	153	ORG 01D8H		02B8 740F	225	MOV A, #0FH	
01D8 7B03	154	DLY_2SEC: MOV R3, #03H		02BA 1200A0	226	LCALL WR_LCD	
01DA 7CFF	155	L8: MOV R4, #0FFH		02BD 7406	227	MOV A, #06H	
01DC 7DFF	156	L7: MOV R5, #0FFH		02BF 1200A0	228	LCALL WR_LCD	
01DE 00	157	L6: NOP		02C2 7401	229	MOV A, #01H	
01DF 00	158	NOP		02C4 1200A0	230	LCALL WR_LCD	
01E0 00	159	NOP		02C7 7480	231	MOV A, #80H	
01E1 00	160	NOP		02C9 1200A0	232	LCALL WR_LCD	
01E2 00	161	NOP		02CC 22	233	RET	;BACK TO PROGRAMME
01E3 00	162	NOP		02CD 00	234	NOP	
01E4 DDF8	163	DJNZ R5,L6 ;01DEH		02CE 00	235	NOP	
01E6 DCF4	164	DJNZ R4,L7 ;01DCH		02CF 00	236	NOP	
01E8 DBF0	165	DJNZ R3,L8 ;01DAH		02D0 1202B3	237	DIS_M_ON: LCALL INI_LCD	;INITIALIZE LCD
01EA 22	166	RET	;BACK TO PROGRAMME	02D3 744D	238	MOV A, # 4DH	;WRITE INTO DISPLAY RAM OF LCD
0226	167	ORG 0226H		02D5 1200C0	239	LCALL LCD_RAM	
0226 D2B5	168	M_STOP: SETB P3.5	;STOP MOTOR	02D8 744F	240	MOV A, #04FH	
0228 1202B3	169	LCALL INI_LCD	;INITIALIZE LCD	02DA 1200C0	241	LCALL LCD_RAM	
022B 7444	170	MOV A, #44H	;START FILLING DISPLAY RAM OF LCD	02DD 7454	242	MOV A, #54H	
022D 1200C0	171	LCALL LCD_RAM		02DF 1200C0	243	LCALL LCD_RAM	
0230 7452	172	MOV A, #52H		02E2 744F	244	MOV A, #04FH	
0232 1200C0	173	LCALL LCD_RAM					
0235 7459	174	MOV A, #59H					
0237 1200C0	175	LCALL LCD_RAM					

02E4 1200C0	245		LCALL LCD_RAM	050A DCFA	306		DJNZ R4, L506	
02E7 7452	246		MOV A, #52H	050C DBF6	307		DJNZ R3, L504	
02E9 1200C0	247		LCALL LCD_RAM	050E DAF2	308		DJNZ R2, L502	
02EC 74B0	248		MOV A, #0B0H	0510 22	309		RET	;BACK TO MAIN PROGRAMME
02EE 1200C0	249		LCALL LCD_RAM					
02F1 744F	250		MOV A, #04FH	054D	310		ORG 054DH	
02F3 1200C0	251		LCALL LCD_RAM	054D 020642	311	M_START:	LJMP SUB_BR	;SUB BRANCH DUE TO SPACE PROBLEM
02F6 744E	252		MOV A, #04EH					;CHECK FOR HIGH VOLTAGE
02F8 1200C0	253		LCALL LCD_RAM	0550 30A447	312	MAIN:	JNB P2.4, L59A	
02FB 02011E	254		LJMP LOOP	0553 30A34D	313		JNB P2.3, L5A3	;CHECK FOR LOW VOLTAGE
			;BACK TO MAIN PROGRAMME AT 011E H	0556 C2B5	314		CLR P3.5	;IF VOLTAGE OK THEN START MOTOR
02FE 00	255		NOP					;INITIALIZE LCD
02FF 00	256		NOP	0558 1202B3	315		LCALL INI_LCD	;START WRITING TO DISPLAY RAM OF LCD
0300 120320	257	L300:	LCALL TMR_5MIN ;CALL 5 MINUTES TIMER	055B 744D	316		MOV A, #04DH	
0303 020550	258		LJMP MAIN ;GOTO MAIN PROGRAMME AT 0550 H					
0320	259		ORG 0320H	055D 1200C0	317		LCALL LCD_RAM	
0320 7A03	260		TMR_5MIN: MOV R2, #03H	0560 744F	318		MOV A, #04FH	
0322 7BFF	261	L12:	MOV R3, #0FFH	0562 1200C0	319		LCALL LCD_RAM	
0324 7CFF	262	L11:	MOV R4, #0FFH	0565 7454	320		MOV A, #54H	
0326 7DFF	263	L10:	MOV R5, #0FFH	0567 1200C0	321		LCALL LCD_RAM	
0328 00	264	L9:	NOP	056A 744F	322		MOV A, #4FH	
0329 00	265		NOP	056C 1200C0	323		LCALL LCD_RAM	
032A 00	266		NOP	056F 7452	324		MOV A, #52H	
032B 00	267		NOP	0571 1200C0	325		LCALL LCD_RAM	
032C DDFA	268		DJNZ R5, L9 ;0328	0574 74B0	326		MOV A, #0B0H	
032E DCF6	269		DJNZ R4, L10 ;0326	0576 1200C0	327		LCALL LCD_RAM	
0330 DBF2	270		DJNZ R3, L11 ;0324	0579 744F	328		MOV A, #4FH	
0332 DAEE	271		DJNZ R2, L12 ;0322	057B 1200C0	329		LCALL LCD_RAM	
0334 22	272		RET	057E 744E	330		MOV A, #4EH	
			;BACK TO MAIN PROGRAMME	0580 1200C0	331		LCALL LCD_RAM	
0430	273		ORG 0430H	0583 120750	332		LCALL TMR_10MIN	;ENTER INTO 10 MINUTES TIMER
0430 30A209	274	L430:	JNB P2.2, L43C					
			;IF TANK FULL GOTO 043C H	0586 307820	333		JNB 78H, C_LV; 05A9H	;LOW VOLTAGE THEN GOTO 05A9 H
0433 30A409	275		JNB P2.4, L43F					
			;H/V THEN GOTO 043F H	0589 307714	334		JNB 77H, C_HV; 05A0H	;HIGH VOLTAGE THEN GOTO 05A0 H
0436 30A30F	276	L436:	JNB P2.3, L448					
			;L/V THEN GOTO 0448 H	058C 20761D	335		JB 76H, DRY ;05ACH	;IF TANK DRY THEN GOTO 05ACH H
0439 80F5	277	L439:	SJMP L430					
043B 00	278		NOP	058F 30A2BB	336	L58F:	JNB P2.2, M_START	;TANK FULL THEN GOTO 054D H
043C 02012D	279	L43C:	LJMP LOOP1	0592 30A405	337		JNB P2.4, L59A	;HIGH VOLTAGE THEN GOTO 059A H
			;ACK TO MAIN PROGRAMME	0595 30A30B	338	L595:	JNB P2.3, L5A3	;LOW VOLTAGE THEN GOTO 05A3 H
043F 1201D8	280	L43F:	LCALL DLY_2SEC	0598 80F5	339	L598:	SJMP L58F	;REPEAT FROM 058F H
			;WAIT FOR 2 SECONDS	059A 1201D8	340	L59A:	LCALL DLY_2SEC	;WAIT FOR 2 SECONDS
0442 20A4F1	281		JB P2.4, L436	059D 20A4F5	341		JB P2.4, L595	;IF NOT H/V THEN GO BACK TO 0595 H
			;IF NOT H/V THEN GOTO 0436 H	05A0 02013C	342		C_HV: LJMP LOOP2	;STILL H/V THEN GOTO 013C H
0445 02013C	282		LJMP LOOP2	05A3 1201D8	343	L5A3:	LCALL DLY_2SEC	;WAIT FOR 2 SECONDS
			;H/V THEN GOTO 013C H	05A6 20A3EF	344		JB P2.3, L598	;IF NOT L/V THEN GO BACK TO 0598 H
0448 1201D8	283	L448:	LCALL DLY_2SEC	05A9 020191	345	C_LV:	LJMP LOOP3	;STILL L/V THEN GOTO 0191 H
			;WAIT FOR 2 SECONDS	05AC D2B5	346		DRY: SETB P3.5	;STOP MOTOR
044B 20A3EB	284		JB P2.3, L439	05AE 1202B3	347		LCALL INI_LCD	;INITIALIZE LCD
			;IF NOT L/V THEN GOTO 0439 H	05B1 7444	348		MOV A, #44H	;START WRITING TO DISPLAY RAM OF LCD
044E 020191	285		LJMP LOOP3					
			;IF L/V THEN GOTO 0191 H	05B3 1200C0	349		LCALL LCD_RAM	
0490	286		ORG 0490H	05B6 7452	350		MOV A, #52H	
0490 00	287		CHK_HV: NOP	05B8 1200C0	351		LCALL LCD_RAM	
0491 00	288		NOP	05BB 7459	352		MOV A, #59H	
0492 120500	289	L492:	LCALL DLY_2MIN	05BD 1200C0	353		LCALL LCD_RAM	
			;WAIT FOR 2 MINUTES	05C0 74B0	354		MOV A, #0B0H	
0495 30A4FA	290		JNB P2.4, L492					
			;CHECK FOR H/V AGAIN					
0498 020000	291		LJMP START					
			;GOTO START AGAIN					
04B0	292		ORG 04B0H					
04B0 00	293		CHK_LV: NOP					
04B1 00	294		NOP					
04B2 120500	295	L4B2:	LCALL DLY_2MIN					
			;WAIT FOR 2 MINUTES					
04B5 30A3FA	296		JNB P2.3, L4B2					
			;CHECK FOR L/V AGAIN					
04B8 020000	297		LJMP START					
			;GOTO START AGAIN					
0500	298		ORG 0500H					
0500 7AFF	299	DLY_2MIN:	MOV R2, #0FFH					
0502 7BFF	300	L502:	MOV R3, #0FFH					
0504 7CFF	301	L504:	MOV R4, #0FFH					
0506 00	302	L506:	NOP					
0507 00	303		NOP					
0508 00	304		NOP					
0509 00	305		NOP					

05C2 1200C0	355	LCALL LCD_RAM	064E 7444	409	MOV A, #44H	;START WRITING TO DISPLAY RAM OF LCD
05C5 7453	356	MOV A, #53H				
05C7 1200C0	357	LCALL LCD_RAM				
05CA 7455	358	MOV A, #55H	0650 1200C0	410	LCALL LCD_RAM	
05CC 1200C0	359	LCALL LCD_RAM	0653 7452	411	MOV A, #52H	
05CF 744D	360	MOV A, #4DH	0655 1200C0	412	LCALL LCD_RAM	
05D1 1200C0	361	LCALL LCD_RAM	0658 7459	413	MOV A, #59H	
05D4 7450	362	MOV A, #50H	065A 1200C0	414	LCALL LCD_RAM	
05D6 1200C0	363	LCALL LCD_RAM	065D 74B0	415	MOV A, #0B0H	
05D9 74C0	364	MOV A, #0C0H ;GOTO 9TH CHARACTER OF LCD	065F 1200C0	416	LCALL LCD_RAM	
			0662 7453	417	MOV A, #53H	
05DB 1200A0	365	LCALL WR_LCD	0664 1200C0	418	LCALL LCD_RAM	
05DE 7432	366	MOV A, #32H ;START WRITING AGAIN	0667 7455	419	MOV A, #55H	
05E0 1200C0	367	LCALL LCD_RAM	0669 1200C0	420	LCALL LCD_RAM	
05E3 120320	368	LCALL TMR_5MIN ;WAIT FOR 5 MINUTES	066C 744D	421	MOV A, #4DH	
05E6 30A447	369	JNB P2.4,HV ; 0630H ;H/V THEN GOTO 0630 H	066E 1200C0	422	LCALL LCD_RAM	
05E9 30A34D	370	JNB P2.3,LV ; 0639H ;L/V THEN GOTO 0639 H	0671 7450	423	MOV A, #50H	
05EC C2B5	371	CLR P3.5 ;IF VOLTAGE OK THEN START MOTOR	0673 1200C0	424	LCALL LCD_RAM	
05EE 1202B3	372	LCALL INI_LCD ;INITILIZE LCD	0676 74C0	425	MOV A, #0C0H ;GOTO 9TH CHARACTER OF LCD	
05F1 744D	373	MOV A, #4DH ;START WRITING TO DISPLAY RAM OF LCD				
			0678 1200A0	426	LCALL WR_LCD	
05F3 1200C0	374	LCALL LCD_RAM	067B 7433	427	MOV A, #33H	;START WRITING AGAIN
05F6 744F	375	MOV A, #4FH				
05F8 1200C0	376	LCALL LCD_RAM	067D 1200C0	428	LCALL LCD_RAM	
05FB 7454	377	MOV A, #54H	0680 80FE	429	SJMP \$;STAY HERE UNTILL MANUAL RESET
05FD 1200C0	378	LCALL LCD_RAM				
0600 744F	379	MOV A, #4FH	0750	430	ORG 0750H	
0602 1200C0	380	LCALL LCD_RAM	0750 7A05	431	TMR_10MIN: MOV R2, #05H	
0605 7452	381	MOV A, #52H	0752 7BFF	432	L752: MOV R3, #0FFH	
0607 1200C0	382	LCALL LCD_RAM	0754 7CFF	433	L754: MOV R4, #0FFH	
060A 74B0	383	MOV A, #0B0H	0756 7DFF	434	L756: MOV R5, #0FFH	
060C 1200C0	384	LCALL LCD_RAM	0758 00	435	L758:NOP	
060F 744F	385	MOV A, #4FH	0759 00	436	NOP	
0611 1200C0	386	LCALL LCD_RAM	075A 00	437	NOP	
0614 744E	387	MOV A, #4EH	075B 00	438	NOP	
0616 1200C0	388	LCALL LCD_RAM	075C 00	439	NOP	
0619 120750	389	LCALL TMR_10MIN ;GOTO MINUTES TIMER	075D DDF9	440	DJNZ R5, L758	
			075F DCF5	441	DJNZ R4, L756	
061C 307820	390	JNB 78H, L63F ;L/V THEN GOTO 063F H	0761 30A40C	442	JNB P2.4,L770	;H/V THEN GOTO 0670 H
061F 307714	391	JNB 77H, L636 ;H/V THEN GOTO 0636	0764 30A311	443	L764: JNB P2.3, L778	;L/V THEN GOTO 0678 H
0622 207622	392	JB 76H, L647 ;STILL DRY SUMP THEN GOTO 0647 H	0767 30A116	444	L767: JNB P2.1, L780	; NOT DRY SUMP THEN GOTO 0680 H
0625 30A21A	393	L625: JNB P2.2, SUB_BR ;0642H ;TANK FULL THEN GOTO 0642 H				
0628 30A405	394	JNB P2.4, HV ;0630H ;H/V THEN GOTO 0630 H	076A DBE8	445	L76A: DJNZ R3, L754	
062B 30A30B	395	N_HV: JNB P2.3, LV ;0639H ;L/V THEN GOTO 0639 H	076C DAE4	446	DJNZ R2, L752	
062E 80F5	396	N_LV: SJMP L625 ;REPEAT FROM 0625 H	076E 22	447	RET	;BACK TO MAIN PROGRAMME
0630 1201D8	397	HV: LCALL DLY_2SEC ;WAIT FOR 2 SECONDS	076F 00	448	NOP	
0633 20A4F5	398	JB P2.4, N_HV ;062BH ;IF NOT H/V THEN GOTO 062B H	0770 1201D8	449	L770: LCALL DLY_2SEC	;WAIT FOR 2 SECONDS
0636 02013C	399	L636:LJMP LOOP2 ;STILL H/V THEN GOTO 013C H	0773 30A412	450	JNB P2.4, 0788H	;STILL H/V THEN GOTO 0788 H
0639 1201D8	400	LV: LCALL DLY_2SEC ;WAIT FOR 2 SECONDS	0776 80EC	451	SJMP L764	; NOT H/V THEN GOTO 0764 H
063C 20A3EF	401	JB P2.3, N_LV ;062EH ;IF NOT L/V THEN GOTO 062E H	0778 1201D8	452	L778: LCALL DLY_2SEC	;WAIT FOR 2 SECONDS
063F 020191	402	L63F: LJMP LOOP3 ;STILL L/V THEN GOTO 0191 H	077B 30A30D	453	JNB P2.3, 078BH	;STILL L/V THEN GOTO 078B H
0642 D2B5	403	SUB_BR: SETB P3.5 ;STOP MOTOR	077E 80E7	454	SJMP L767	;NOT L/V THEN GOTO 0767 H
0644 020260	404	LJMP TANK_FULL ;TANK FULL THEN GOTO 0260 H	0780 1201D8	455	L780: LCALL DLY_2SEC	;WAIT FOR 2 SECONDS
			0783 30A108	456	JNB P2.1, 078EH	;STILL NOT DRY SUMP THEN GOTO 078E H
0647 D2B5	405	L647: SETB P3.5 ;STOP MOTOR	0786 80E2	457	SJMP L76A	;OTHERWISE GOTO 076A
0649 00	406	NOP	0788 C277	458	CLR 77H	;CLEAR FLAG 77H FOR H/V
064A 00	407	NOP	078A 22	459	RET	
064B 1202B3	408	LCALL INI_LCD ;INITIALIZE LCD	078B C278	460	CLR 78H	;CLEAR FLAG 78H FOR L/V
			078D 22	461	RET	
			078E C276	462	CLR 76H	;CLEAR FLAG 76H FOR DRY SUMP CHECK
			0790 22	463	RET	
				464	END	
			VERSION 1.2k ASSEMBLY COMPLETE, 0 ERRORS FOUND			