

```

1: /*
2: Thursday, 22 May 2014
3: ATS Program - Button Controlled.
4: uC: PIC16F628A
5: Fosc: 4MHz
6: Author: N. Chitiyo (nchitiyo@sirdc.ac.zw)
7: NOTE: use COFF file for debugging in Proteus ISIS
8: Software Rev: 0.2.2sf (ATS)
9: Button Software Version: 0.1
10:
11: Software Revisions:
12: 0.0.1      : ATS. Auto only
13: 0.0.2      : ATS. With Cooldown Timer
14: 0.0.3      : ATS with key-override control
15: 0.1.0      : software v0.0.3 for board version 0.1 (Split board design)
16: 0.2.0      : software v0.1.0 for board version 0.2T(Split-board - Cable connected)
17: 0.2.1      : software v0.2.0 for board version 0.2U (ULN driver version)
18: 0.2.2sf    : ATS Software Cleaned up. for Documentation Purposes, sans feedback.
19: 0.2.2f     : v0.2.2 implementing Feedback
20:
21: Hardware Version: v0.2U (With ULN)
22: Hardware Revisions:
23: 0.0        : ATS Board - Debug Version Prototype. uni-Board, Transistor Driven
24: 0.1        : Split version. dual-board, joined by headers. Transistor Driven
25: 0.2        : Split version dual board, joined by ribbon cable/UTP
26: 0.2U       : dual, ULN-driven, Changeover Relay on board 1 with gen feedback
27:
28: */
29:
30:
31: unsigned int ProgTimer, RealTimer, RunTimer, CoolDownTimer;
32: bit Auto_Flag, Run_Flag, CoolDown_Flag;
33: bit GenFeedFlag;          //feedback indirect register
34: unsigned int RunTime, CoolDownTime;
35: unsigned short RBValue;
36:
37: /*Input-Output Table:
38:      *   PIN   |  I/O   |  Assign          |  Notes
39:      *-----|-----|-----|-----
40:      *  RA0   |Output | SQOUT/ N/A      | Clock Count
41:      *  RA1   |Input  | N/A             |
42:      *  RA2   |Input  | ZESA Sense      |
43:      *  RA3   |Output | N/A             |
44:      *  RA4   |Input  | Gen Feedback    |Unimplemented
45:      *  RB0   |Input  | OFF Button      |Manual GenSTOP interrupt en.
46:      *  RB1   |Input  | Start           |Manual GenStart
47:      *  RB2   |Input  | Auto            |GenAuto
48:      *  RB3   |Input  | ON              |GenON
49:      *  RB4   |Output | GenSTOP Control (NC!!!) |
50:      *  RB5   |Output | ChangeOver Control (NO) |
51:      *  RB6   |Output | GenStart Control (NO)   | crank.
52:      *  RB7   |Output | GenOn Control (N.O)     |
53:      *-----|-----|-----|-----
54: */
55: sbit SQOUT at RA0_Bit;
56: sbit ZESA at RA2_Bit;
57: sbit GenStop at RB4_Bit;
58: sbit ChangeOver at RB5_Bit;
59: sbit GenStart at RB6_Bit;
60: sbit GenOn at RB7_Bit;
61:
62: // ////////////Declare Functions ////////////

```

```

63: void Poll();
64: //////////////////////////////////////////
65:
66:
67: void interrupt() {           //TMR0 Interrupt Handler
68:     GIE_Bit = 0;
69:     if (T0IF_Bit) {
70:
71:         T0IF_Bit = 0;
72:         TMR0 = 0;
73:         ProgTimer++;
74:         if (progTimer == 1953)
75:         {
76:             RealTimer++;
77:
78:             //Check Run and CoolDown Status
79:             if (RunTimer == RunTime) {
80:                 RunTimer = 0;
81:                 Run_flag = 0;
82:                 CoolDown_Flag = 1;
83:             }
84:
85:             if (CoolDownTimer == CoolDownTime) {
86:                 CoolDownTimer = 0;
87:                 //Run_flag = 1; //The Run Flag Should Be Started Elsewhere
88:                 CoolDown_Flag = 0;
89:             }
90:
91:             if (Run_Flag) RunTimer++;
92:             if (CoolDown_Flag) CoolDownTimer++;
93:             SQOUT=~SQOUT;
94:             progTimer = 0;
95:         }
96:     }
97:     if (INTF_Bit) {           //Pressing "Stop" forced Stop, Turn Off all
98:         INTF_Bit = 0;
99:         PORTB = 0;
100:         RBValue = 19;         //so that it falls into "Default"
101:         Auto_Flag = 0;
102:         Run_Flag = 0;
103:     }
104:     GIE_Bit = 1;
105: }
106:
107:
108: void main() {
109:
110: //set the runtime (4 hours) and cooldown time (1 hour)
111:
112:     //RunTime = 20;           //Testing Purposes
113:     RunTime = 14388;         //4 hours - 12 seconds run time
114:     //CoolDownTime = 20;      //Testing Puroses
115:     CoolDownTime = 3599;     //1 hour (- 1 second) cooldown time
116:
117:     PORTA = 0;
118:     TRISA = 0b10110;         //following the Table Above
119:     PORTB = 0;
120:     TRISB = 0b00001111;     //Following the Table Above
121:     CMCON = 0x0F;           //PORTA all digital
122:
123:     T0IE_Bit = 1;           //Enable TMR0 Interrupt
124:     INTE_Bit = 1;           //Enable RB0 Interrupt...

```

```

125:     INTEDG_Bit = 1;           //...on rising edge of RB0
126:     GIE_bit = 1;
127:
128:     // ////////////Timer Configuration ////////////
129:
130:     T0CS_Bit = 0;           //Select Timer Mode. Timer Starts Now
131:     TMR0 = 0;               //reset the TMR0 Register
132:     PSA_Bit = 0;           //Assign Prescaler from WDT to Timer0 when value = 0
133:     OPTION_REG &= 248;      //Clear Previous Prescaler Values
134:     OPTION_REG |= 0;        //set Prescaler to TimeSet (1:2)
135:
136:     ////////////////
137:
138:
139: while(1) {                  //Main Endless Loop
140:
141:     delay_ms(100);          //delay for latency (To allow system
142:                             //to tolerate key debounces)
143:
144:     //collect the last three values input from the ignition switch:
145:     if (RBValue != 19) RBValue = PORTB & 0x07; //19 is the fallback value
146:
147:
148:     if (!Run_flag && CoolDown_flag) RBValue = 19;    //it's time to cool down
149:
150:     /* NOTE: in manual mode, When the Gen Cools down, you have to Start
151:        it manually after cooldown.
152:     */
153:
154:     /* RBValue Mode Coding Table
155:
156:         |Auto |Start |Off |
157:         |----|-----|----|-----|
158:     * Condition |RB2 |RB1 |RB0 | Hex | Dec |
159:     *-----|----|-----|----|-----|
160:     * Auto    | 1  | x  | 0  | 0x04| 4,6 |
161:     * Start   | 0  | 1  | 0  | 0x02| 2   |
162:     * OFF     | x  | x  | 1  | 0x01| 1,3,5,7 |
163:     * Maintain| 0  | 0  | 0  | 0x00| 0   |
164:     *-----|----|-----|----|-----|
165:     */
166:
167:     switch(RBValue) {
168:
169:     // ////////////Auto Mode//////////
170:     case 4:                      //Gen Auto Mode Selected
171:         Auto_Flag = 1;          // Gen is in Auto Mode
172:         RBValue = 0;
173:         if (Auto_Flag) Poll();
174:         break;
175:
176:     case 6:                      //Gen Auto Mode Selected
177:         Auto_Flag = 1;          // Gen is in Auto Mode
178:         RBValue = 0;
179:         if (Auto_Flag) Poll();
180:         break;
181:
182:     // ////////////System Start/Crank ////////////
183:
184:     case 2:                      //Start Button Pressed
185:         Auto_Flag = 0;          //Gen is in Manual Mode
186:         GenStop = 1;            //Turn Off GEN_OFF signal
187:         GenOn = 1;

```

```

187:         GenStart = 1;
188:         Run_Flag = 1;
189:         RBValue = 0;
190:         break;
191:
192:         // ///////////////////////////////////Maintain Scenario ///////////////////////////////////
193:
194:         case 0:
195:             GenStart = 0;
196:             Auto_Flag = 0;
197:             RBValue = 0;
198:             break;
199:
200:         // ///////////////////////////////////Off and Default Scenario/////////////////////////////////
201:
202:         default:
203:             Auto_Flag = 0;
204:             PORTB = 0x00;    //Turn the Gen Off if none of the conditions are met
205:             delay_ms(2000);
206:             Run_Flag = 0;
207:             RBValue = 0;
208:             break;        //We need an ERROR Condition Here
209:         } //switch
210:
211:
212:     } //While
213:
214: }
215:
216:
217: // ///////////////////////////////////Polling Function ///////////////////////////////////
218: void Poll() {
219:     if (Auto_Flag && ZESA) {
220:         ChangeOver = 0;    // make sure you revert to ZESA Supply
221:         delay_ms(2000);
222:         RBValue = 19;      //Fall to Default
223:     }
224:     else {
225:         //If ZESA is not there,
226:         delay_ms(1000);    //Just wait...
227:         if (!Run_flag) { //if the Gen is running, leave and Poll
228:             if (!CoolDown_Flag) {
229:                 GenStop = 1;    //Run the Generator Start Routine
230:                 GenOn = 1;
231:                 GenStart = 1;
232:                 delay_ms(5000);
233:                 Run_Flag = 1;
234:                 GenStart = 0;    //Stop Cranking
235:
236:                 Delay_ms(5000);    //Stabilize
237:                 if ((RBValue == 4 || RBValue == 6) && !ZESA) {
238:                     ChangeOver = 1;
239:                     Run_Flag = 1;
240:                 } //if AutoMode
241:             } //if !CoolDown_Flag
242:         } // if !Run Flag
243:     } //else ZESA is not there
244: } //Poll

```