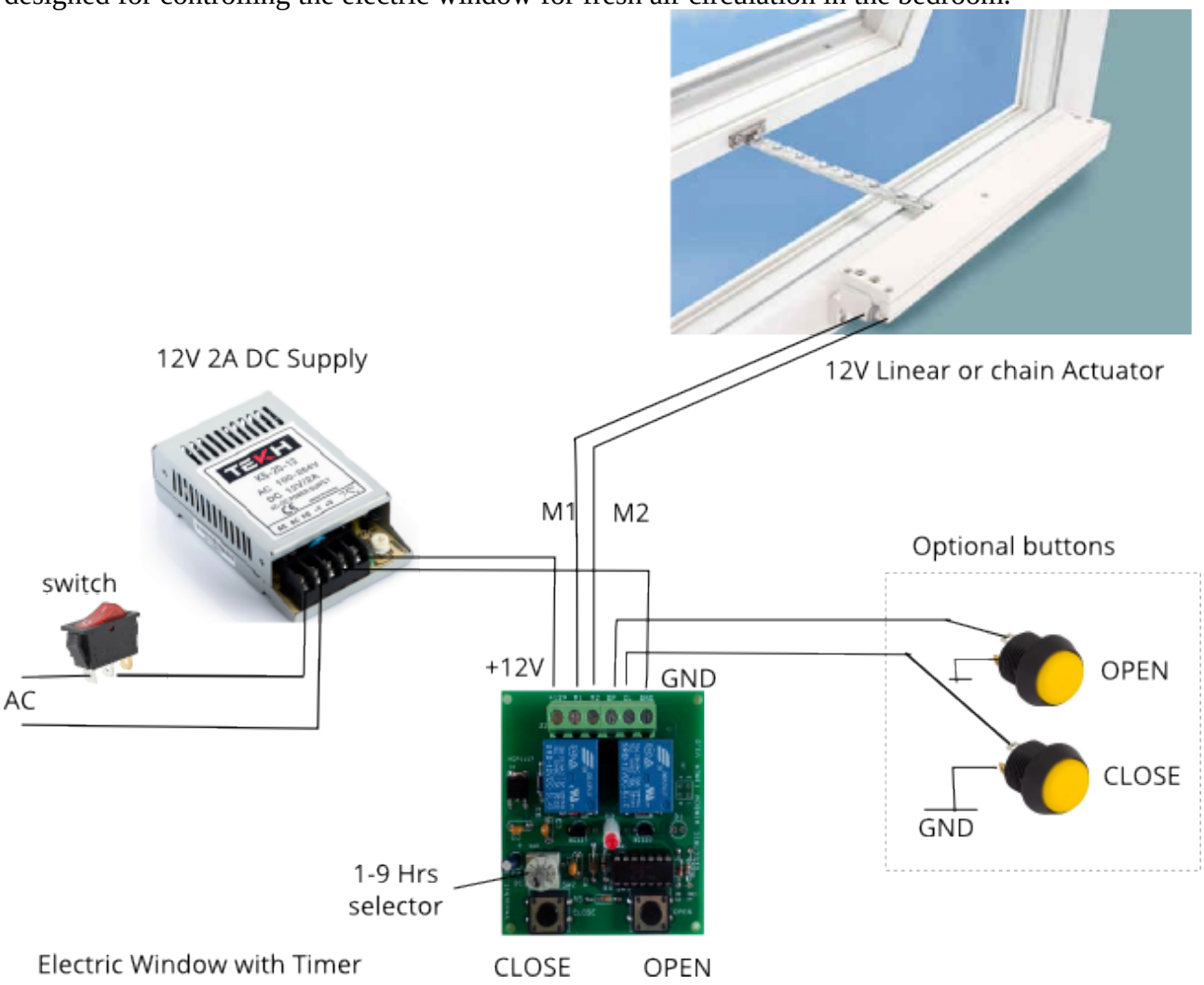
Electric window timer v1.0

The device has designed for controlling the linear, or chain actuator with delay timer. Time is set with the on-board selector 1-9 Hours. When time is reached, the actuator is activated. This timer is designed for controlling the electric window for fresh air circulation in the bedroom.



Button functions:

CLOSE — Close button is for manual control window closing. Working only when status LED is still lit.

OPEN — Open button has two functions:
First press is for starting the timer. Status LED will be blinked.
Second press is for testing, window will open until status LED will be still lit.
For normal use after window has been closed, just press once.

Hours setting: Turn the selector for hours setting, 0 position is for quick test, about 2 mins.

Technical specifications:

1. M1-M2, H-bridge power train for 12V linear actuator or chain actuator.

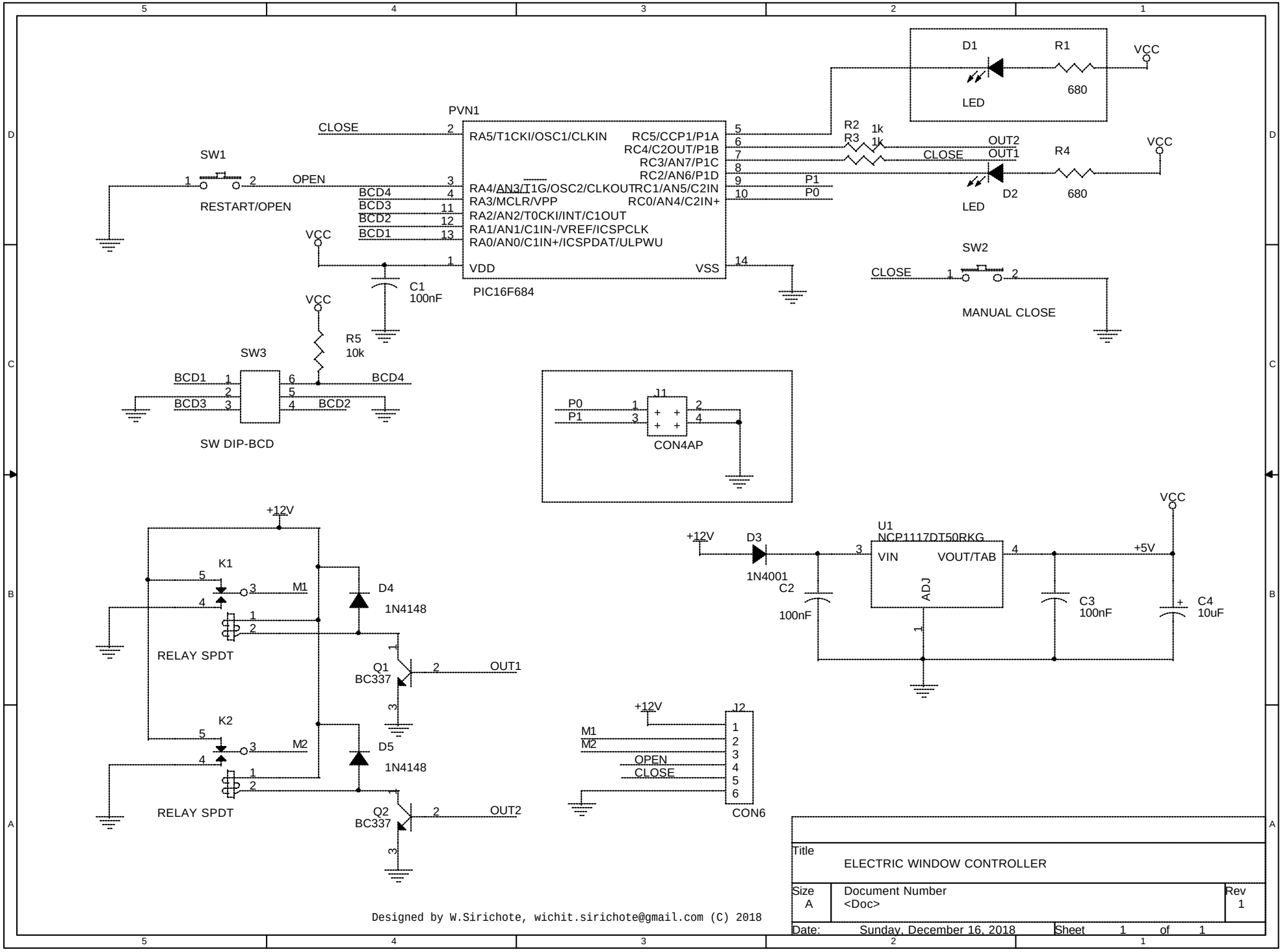
2. Preset timer: 1-9 Hours.

3. Onboard control buttons: CLOSE manual control window closing
OPEN, timer restart/ test mode

4. Microcontroller: Microchip PIC16F684

5. Stroke control byte: EEPROM location 0, default is 50.

More technical details, please contact Wichit Sirichote, wichit.sirichote@gmail.com
URL: kswichit.com


D1
R1
VCC
680
LED
PVN1
CLOSE
RA5/T1CKI/OSC1/CLKIN
RC5/CCP1/P1A
RC4/C2OUT/P1B
RC3/AN7/P1C
RC2/AN6/P1D
RA4/AN3/T1G/OSC2/CLKOUT
RC1/AN5/C2IN
RA3/MCLR/VPP
RC0/AN4/C2IN+
RA2/AN2/T0CKI/INT/C1OUT
RA1/AN1/C1IN-/VREF/ICSPCLK
RA0/AN0/C1IN+/ICSPDAT/ULPWU
VDD
VSS
PIC16F684
R2
R3
1k
1k
OUT2
OUT1
CLOSE
P1
P0
R4
D2
LED
SW1
OPEN
RESTART/OPEN
BCD4
BCD3
BCD2
BCD1
C1
100nF
SW2
MANUAL CLOSE
R5
10k
SW3
SW DIP-BCD
J1
P0
P1
CON4AP
+12V
K1
M1
D4
1N4148
RELAY SPDT
Q1
BC337
OUT1
K2
M2
D5
1N4148
RELAY SPDT
Q2
BC337
OUT2
J2
M1
M2
OPEN
CLOSE
CON6
U1
NCP1117DT50RKG
VIN
VOUT/TAB
ADJ
D3
1N4001
C2
100nF
C3
100nF
C4
10uF
+5V
Designed by W.Sirichote, wichit.sirichote@gmail.com (C) 2018
Title
ELECTRIC WINDOW CONTROLLER
Size
A
Document Number
<Doc>
Rev
1
Date:
Sunday, December 16, 2018
Sheet
1
of
1

ELECTRIC WINDOW CONTROLLER Revised: Sunday, December 16, 2018
Revision: 1

Bill Of Materials December 29,2018 17:44:51 Page1

Item	Quantity	Reference	Part
1	3	C1,C2,C3	100nF
2	1	C4	10uF
3	2	D1,D2	LED
4	1	D3	1N4001
5	2	D4,D5	1N4148
6	1	J1	CON4AP
7	1	J2	CON6
8	2	K1,K2	RELAY SPDT
9	1	PVN1	PIC16F684
10	2	Q1,Q2	BC337
11	2	R4,R1	680
12	2	R2,R3	1k
13	1	R5	10k
14	1	SW1	RESTART/OPEN
15	1	SW2	MANUAL CLOSE
16	1	SW3	SW DIP-BCD
17	1	U1	NCP1117DT50RKG

```
// PIC timer for automatic window project
// MCU PIC16F684 internal oscillator 4MHz
// preset stoke is stored at EEPROM locations 0
// user may modify it for the desired length
// test switch is dual functions: RESTART/OPEN
// add state =0 for failsafe
// new BCD switch provides negative logic, must be inverted
// compiled with Mikro-C for PIC v6.0.0
// written by W.Sirichote (C) 2018, wichit.sirichote@gmail.com


sbit LED at PORTC.B2;
sbit OUT1 at PORTC.B3;
sbit OUT2 at PORTC.B4;
sbit key1 at PORTA.B4;

char tick, tick_led;
char sec;
unsigned int min=0;
unsigned int set_time;
char press=-1;
char state =0;

unsigned int preset_length=0, save_preset;
char run=0;
char test=0;
char fire =0;
char programmed=0;
char set_hour=0;

unsigned int length;


unsigned int adc;
char release=0;

void read_key()
{
   if(key1 == 0 && release==0)
   {
   release=1;

   switch (++press)
   {
   case 0: OUT1 =1; OUT2 = 1; state=1; break;
   case 1: OUT1 =1; OUT2 = 0; run=1; press=-1; break;  // start counting for preset
   case 2: OUT1 =1; OUT2 = 1; run=0;  break;  // stop counting
   case 3: OUT1 =0; OUT2 = 1; test =1; break; // test reverse with preset
   case 4: OUT1 =1; OUT2= 1;  press =0; break;
   }
   }



}

```

```
61
62
63  read_BCD()
64  {
65    set_hour = (~PORTA)&0x0f;  // 0-9 BCD code switch is negative logic
66
67    // provide test mode at zero position
68    if(set_hour==0) set_time =2;
69    else set_time= set_hour*60;
70
71
72  }
73
74
75
76
77  // 15Hz interrupt
78  void interrupt() {
79    if (TMR0IF_bit) {
80       TMR0IF_bit = 0;
81
82      tick++;
83
84      read_key();
85
86      if(run)
87      {
88       if(--preset_length == 0)
89          {
90           OUT1 =1; OUT2= 1;  press =-1;
91           preset_length = length;
92           run=0;
93           LED =0;
94
95           fire=0;
96           state=0;
97           tick_led=0;
98
99          }
100
101
102     }
103
104
105
106     if(tick>15)
107     {
108
109     read_BCD();   // read BCD every second
110
111     tick=0;
112
113     if(key1) release=0;
114
115     if(state==1)
116     {
117
118       if(++sec>60)
119       {
120       sec=0;
```

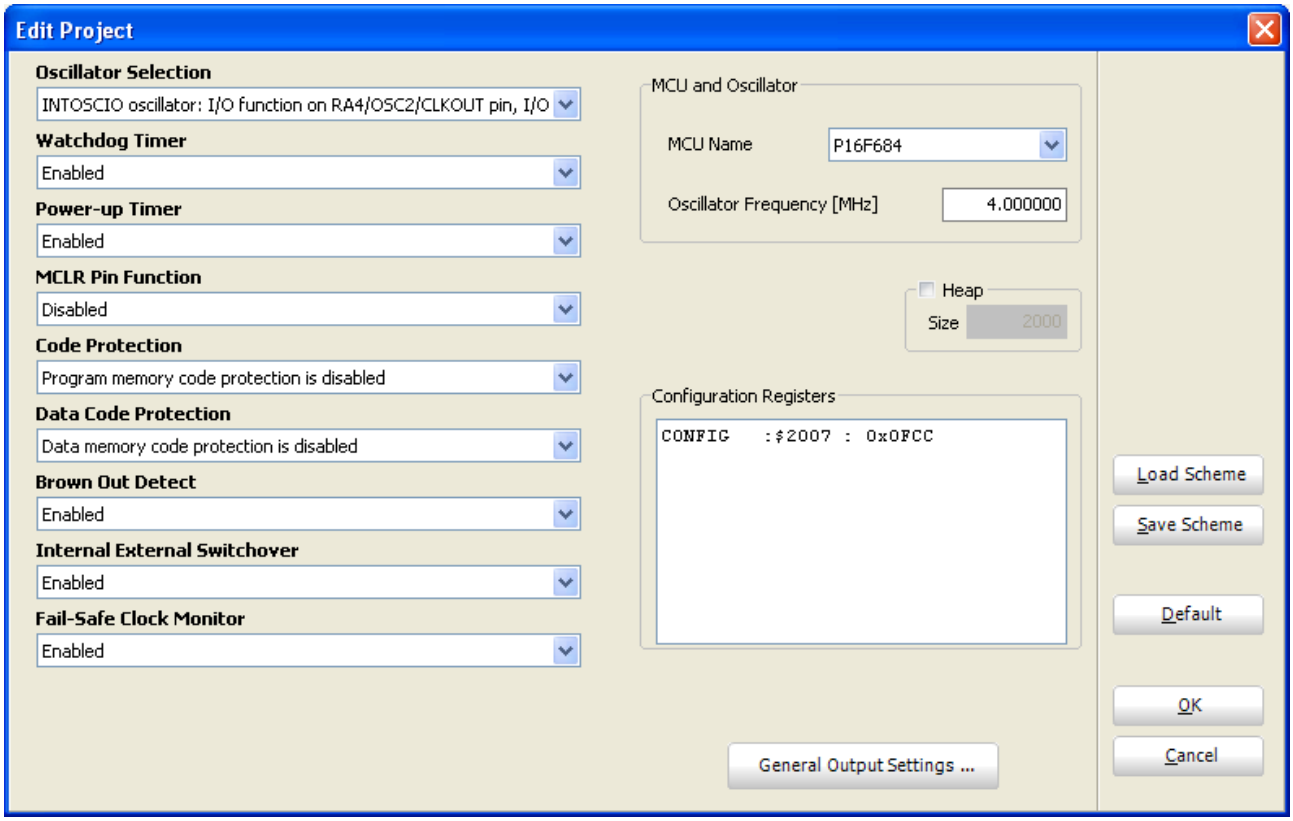
```
121       min++;
122
123       if (min>= set_time && fire == 0 )
124       {
125         OUT1 =1; OUT2 = 0; run=1;
126         fire =1;
127         min=0;
128       }
129       }
130       }
131
132     }
133
134
135
136
137    if(state==1)
138    {
139     if (++tick_led>15)
140     {
141     tick_led=0;
142     LED = 0;
143     delay_ms(5);
144     LED = 1;
145
146     }
147
148    }
149    else LED = 0; // indicate state==0
150
151   }
152 }
153
154
155
156 void main() {
157   OPTION_REG = 0x07;       // Assign prescaler to TMR0
158   ANSEL  = 0;              // Configure PORTA to digital I/O
159   CMCON0 = 7;              // ANSELH = 0;
160
161   WDTCON = 0x16;   // WDT clock/65536
162
163   WPUA = 0xff;
164
165   TRISA = 0xff;
166   TRISC = 0;               // PORTB is output
167   PORTC = 0xFF;            // Initialize PORTC
168
169   INTCON = 0xA0;           // Enable TMRO interrupt
170
171   length = EEPROM_Read(0)*2; // double length
172   preset_length = length; // double length 0-512
173
174    read_BCD(); // read set hour
175
176   while(1)
177   {
178   asm CLRWDT;
179   }
180
```

```
181
182 }
```

Project file settings

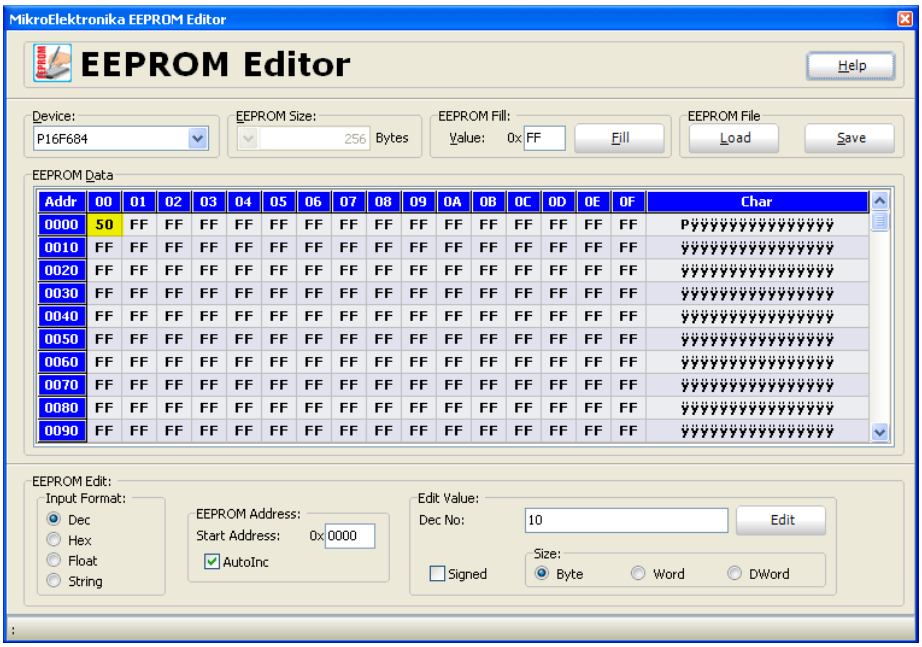
Mikro-c for PIC provides project file settings for programming the configuration fuses.

As shown, Fuse word located at $2007 is 0x0FCC.



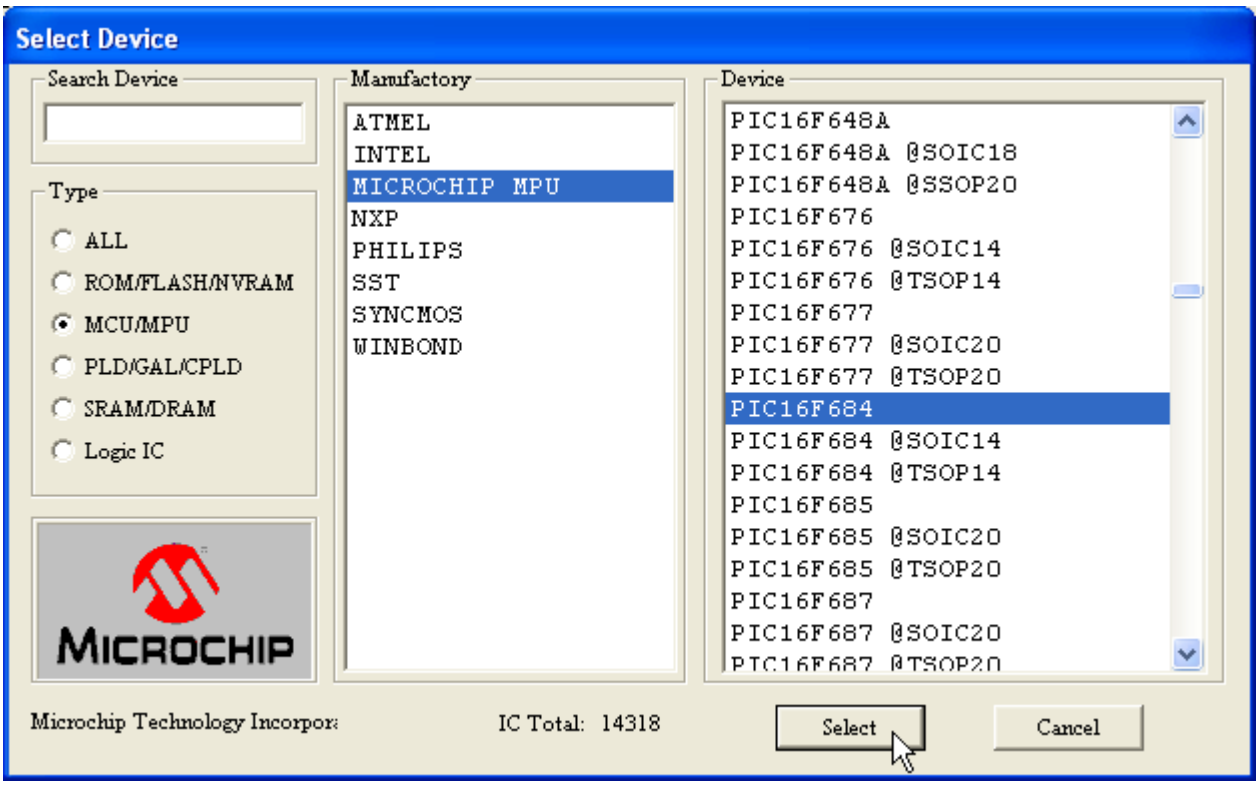
For stroke setting, edit the byte at location 00 in EEPROM editor. The value can be 00-FF.

Longer stroke, value will be 60, 80, A0, says. When complete, click save it.

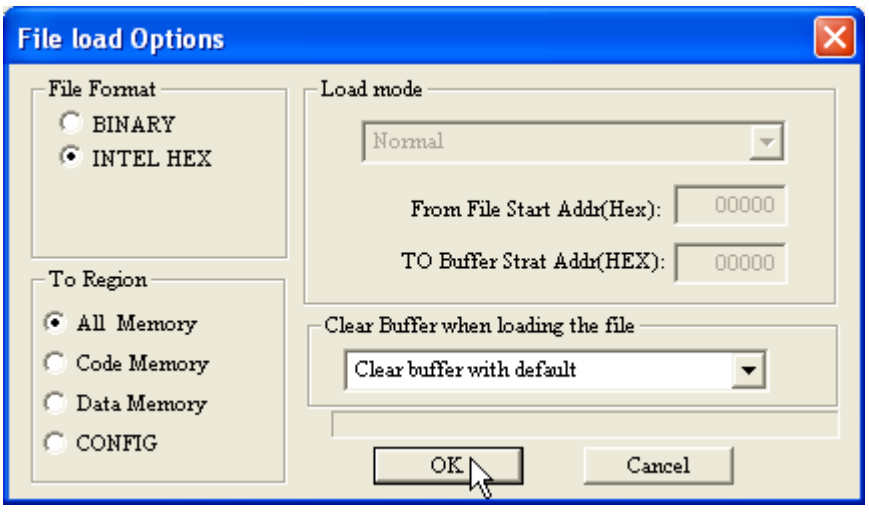


Example of programming the PIC16F684 chip using MiniPro v6.82

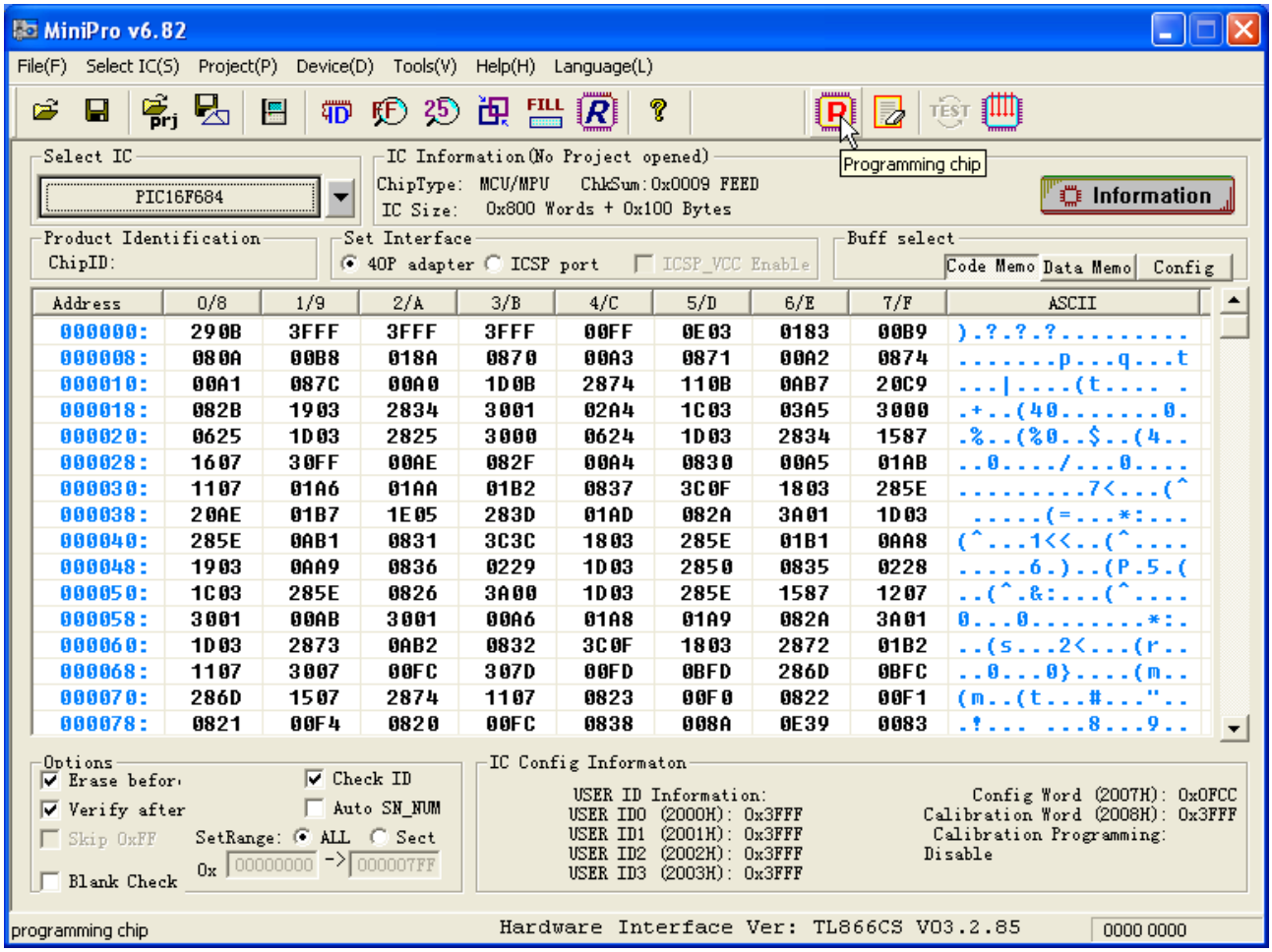
Step 1. Select Microcontroller chip, PIC16F684



Step 2. Select File Load, Intel HEX file, and To Region, All Memory



Step 3. Place the chip into the socket, then click Programming chip.



More information, please contact Wichit Sirichote, wichit.sirichote@gmail.com
URL: kswichit.com