src\main.c

```
// Practice assignment 3, version 2

#include <stdio.h>
#include <avr/io.h>
#include <util/delay.h>

#include <i2cmaster.h>
#include <lcd.h>

unsigned char x, y, z1, z2; // inputs and outputs

void read_xy() {
  if(!((PINC >> 2) & 1)) { //repeat
      x=1;
    }
    else {
      x=0;
    }

    if(!((PINC >> 3) & 1)) { //repeat
      y=1;
    }
    else {
      y=0;
    }
}

char stateChar;

typedef enum {
  A_state,
  B_state,
  C_state,
  D_state,
  E_state,
  F_state,
  G_state,
  H_state
}state;

void show_output(state state_to_print)
{
  LCD_set_cursor(0,0);
  printf("%hhu %hhu ",x,y);
  LCD_set_cursor(0,1);
  switch(state_to_print){
    case A_state: printf("A_STATE"); break;
    case B_state: printf("B_STATE"); break;
```

```
49      case C_state: printf("C_STATE"); break;
50      case D_state: printf("D_STATE"); break;
51      case E_state: printf("E_STATE"); break;
52      case F_state: printf("F_STATE"); break;
53      case G_state: printf("G_STATE"); break;
54      case H_state: printf("H_STATE"); break;
55    }
56    LCD_set_cursor(0,2);
57    printf("%hhu %hhu ",z1,z2);
58  }
59
60
61  state state_transition(state current_state)
62  {
63    switch (current_state) {
64      case (A_state):
65        if((!x)&&(!y))
66          return A_state;
67        if(x)
68          return B_state;
69        if((!x)&&y)
70          return E_state;
71      case (B_state):
72        if((!x))
73          return B_state;
74        if(x)
75          return D_state;
76      case (C_state):
77        if(x)
78          return A_state;
79        if((!x)&&y)
80          return G_state;
81        if((!x)&&(!y))
82          return C_state;
83      case (D_state):
84        if((!x))
85          return D_state;
86        if(x)
87          return C_state;
88      case (E_state):
89        return F_state;
90      case (F_state):
91        return B_state;
92      case (G_state):
93        return H_state;
94      case (H_state):
95        return D_state;
96   }
97
98  }
```

```
 99
100 void state_z(state current_state)
101 {
102   switch (current_state) {
103     case (A_state):
104       z1=1;z2=0;break;
105     case (B_state):
106       z1=1;z2=0;break;
107     case (C_state):
108       z1=1;z2=0;break;
109     case (D_state):
110       z1=0;z2=0;break;
111     case (E_state):
112       z1=1;z2=1;break;
113     case (F_state):
114       z1=1;z2=0;break;
115     case (G_state):
116       z1=1;z2=1;break;
117     case (H_state):
118       z1=1;z2=1;break;
119   }
120
121 }
122
123 #include "i2cmaster.h"
124 #include "lcd.h" //library init
125
126 int main(void) {
127
128   i2c_init(); // initialize I2C and LCD
129   LCD_init();
130
131   state current_state, next_state;
132
133   current_state=A_state;
134
135   DDRB=0b00100000;
136   PORTB=0b00000000;
137
138   DDRC = 0xF0; // set data direction for port C pins, 0-3 as input (i.e. the buttons)
139   PORTC = 0x3F; // set pull-up resistor for port C
140   DDRD = 0xFF; // set data direction for port D, all output
141   PORTD= 0x00; // set output for port D (none)
142
143   while(1) { // start program loop
144     read_xy();
145     next_state=state_transition(current_state);
146     current_state=next_state;
147     state_z(current_state);
148     show_output(current_state);
```

```
149
150     _delay_ms(1000);
151   }
152
153   return 0;
154 }
155
```