src\main.c

```c
1   // Practice assignment 3, version 1
2
3   #include <stdio.h>
4   #include <avr/io.h>
5   #include <util/delay.h>
6
7   #include <i2cmaster.h>
8   #include <lcd.h>
9
10  unsigned char x, y, z1, z2; // inputs and outputs
11  unsigned char q0, q1, q2, q0_next, q1_next, q2_next;
12
13  void read_xy() {
14      if(!((PINC >> 2) & 1)) { //repeat
15          x=1;
16        }
17        else {
18          x=0;
19        }
20
21        if(!((PINC >> 3) & 1)) { //repeat
22          y=1;
23        }
24        else {
25          y=0;
26        }
27  }
28
29  char stateChar;
30
31  typedef enum {
32      A_state,
33      B_state,
34      C_state,
35      D_state,
36      E_state,
37      F_state,
38      G_state,
39      H_state
40  }state;
41
42  void show_output()
43  {
44      LCD_set_cursor(0,0);
45      printf("%hhu %hhu ",x,y);
46      LCD_set_cursor(0,1);
47      printf("%hhu %hhu ",z1,z2);
48      LCD_set_cursor(0,2);
```

```c
49    printf("%hhu %hhu %hhu",q2,q1,q0);
50    LCD_set_cursor(0,3);
51    printf("%hhu %hhu %hhu",q2_next,q1_next,q0_next);
52
53    stateChar=(0b01000000|(q2<<2)|(q1<<1)|q0)+1;
54    LCD_set_cursor(19,0);
55    printf("%c",stateChar);
56 }
57
58 void state_transition()
59 {
60    PORTB |= (1 << PINB5);
61    _delay_ms(500);
62    PORTB &= ~(1 << PINB5);
63 }
64
65 #include "i2cmaster.h"
66 #include "lcd.h" //library init
67
68 int main(void) {
69
70    i2c_init(); // initialize I2C and LCD
71    LCD_init();
72
73    q1=0;q0=0;q2=0;
74    DDRB=0b00100000;
75    PORTB=0b00000000;
76
77    DDRC = 0xF0; // set data direction for port C pins, 0-3 as input (i.e. the buttons)
78    PORTC = 0x3F; // set pull-up resistor for port C
79    DDRD = 0xFF; // set data direction for port D, all output
80    PORTD= 0x00; // set output for port D (none)
81
82    while(1) { // start program loop
83       q0_next=((!q1)&x)|(q0&(!x))|q2;
84       q1_next=((!q2)&x&q0)|(q1&(!x))|(q2&q1);
85       q2_next=((!q0)&q2)|((!q0)&(!x)&y);
86
87       show_output();
88
89       q0=q0_next;
90       q1=q1_next;
91       q2=q2_next;
92
93       z1=(!q0)|(!q1)|q2;
94       z2=(q1&q2)|((!q0)&q2);
95       read_xy();
96       state_transition();
97
98       _delay_ms(500);
```

```
 99        }
100
101        return 0;
102    }
103
```