/\*

\* serialMidi.c

\* Description:

\* Pressing button SW0-2 lights up the corresponding LEDS for 0.5s

\* Repeat

\* Hardware:

\* Outputs: PORTB 0~7 connected to STK600 LEDs 0~7 (active low)

\* Inputs: PORTD 0~7 connected to STK600 pushbuttons 0~7 (active low)

\* PORTC 0~7 connected to L&S board Switches 0~7 (active low)

\*

\* Created: 9/13/2015

\* Author: Jakob Andren

\* Ver: 0.1

\*/

#define F\_CPU 8000000UL

#include <avr/io.h>

#include <util/delay.h>

#include <avr/interrupt.h>

void UART\_out(uint8\_t ch);

void Init\_UART0(uint16\_t BAUDRATE);

void Init\_ports(void);

// global variables

volatile uint16\_t rx\_ch = 0xFF;

int main(void)

{

Init\_ports();

Init\_UART0(9600);

sei();

uint8\_t pressed;

while (1)

{

PORTB = rx\_ch;

pressed = ~PIND;

if(pressed)

{

UART\_out(0b10010001); // Command

uint8\_t converted = Conversion(pressed);

UART\_out(converted, ~PINC); // Note 7bit

UART\_out(0b01001000); // Velocity 7 bit

\_delay\_ms(300);

}

}

// End of while(1)

}

uint8\_t Conversion(uint8\_t pressed, uint8\_t switches)

{

uint8\_t converted;

uint8\_t mask = 0b00000001;

switch(pressed)

{

case 0x01:

converted = 0;

break; // C

case 0x03:

converted = 1;

break; // C# or Db

case 0x02:

converted = 2;

break; // D

case 0x06:

converted = 3;

break; // D# or Eb

case 0x04:

converted = 4;

break; // E

case 0x08:

converted = 5;

break; // F

case 0x18:

converted = 6;

break; // F# or Gb

case 0x10:

converted = 7;

break; // G

case 0x30:

converted = 8;

break; // G# or Ab

case 0x20:

converted = 9;

break; // A

case 0x60:

converted = 10;

break; // A# or Bb

case 0x40:

converted = 11;

break; // B

case 0x80:

converted = 12;

break; // C

default:

break;

}

for(uint8\_t i = 0; i < 7; i++)

{

if((switches & mask) == mask)

{

converted += 12;

}

mask = mask << 1;

}

return converted;

}

ISR(USART0\_RX\_vect){

rx\_ch = UDR0; //read UART register into value

}

void UART\_out(uint8\_t ch){

while((UCSR0A & (1<<UDRE0)) == 0); // Wait until empty

UDR0 = ch; // Put character to be sent in the output register

}

void Init\_UART0(uint16\_t BAUDRATE){

//ubrr = F\_CPU/16uL/BAUDRATE;

// Set baud rate to 9600 @ 8MHz

UBRR0L = 51;

UBRR0H = 0;

// No parity bit, 8 bit char, one stop bit

UCSR0A = 0x00;

UCSR0B = 0b10011000;

UCSR0C = 0b00000110;

}

void Init\_ports(void){

// Hard coded setup

// Set up port B LEDS

DDRB = 0xFF; // No inputs, all ports as outputs

PORTB = 0xFF; // All LEDS are off

// Set up port D Buttons

DDRD = 0x00;

PORTD = 0xFF; // Set pull up on the inputs

// Set up port C Switches from L&S board

DDRC = 0x00;

PORTC = 0xFF; // Set pull up on the inputs

}