Ethan Nagelvoort
821234668
Lab 4

Description:

In this lab, I was supposed to use a keypad and transmit the key that was pressed on the keypad onto the data visualizer through the AVR. To do this, I created a program that sets the baud rate to 9600 and the UBRR using that baud rate. I also enable the transmitter. In the main, I initialize the columns of the keypad as inputs and pullups. I initialize the rows of the keypad as outputs. Then in the infinite while loop, I declare a 2D char array consisting of the characters and numbers on the keypad. I then set all the pins corresponding to the rows to 1, meaning that they are not pressed. Then I use a nested for loop to loop through all the buttons. In the first for loop, I loop through the rows and in declare the ‘i’ loop to 0, meaning on, so it can be checked. I then use the inner loop to loop through the columns. If the button is being pressed, meaning that if the pin corresponding to that column is 0, then the program will return that character. A delay happens after the character is returned, so that only one character appears and not multiple, same characters appear. Then, outside of the inner loop, I declare that pin corresponding to the current row back to 0.

Result:

After connecting the keypad to the correct pins/ports on the AVR and connecting the AVR to my laptop, I was able to get my program to work. Whenever I pressed a button on the keypad, the correct character would appear in the data visualizer. This shows that my program is operating correctly.

Source code:

```
	/*
 * LAB4.c
 *
 * Created: 2/24/2020 2:28:49 PM
 * Author : Ethan
 */
#define F_CPU 16000000
#include <avr/io.h>
#include <util/delay.h>

#define BAUD 9600
```

```
#define MYUBBR ((F_CPU)/(BAUD * 16UL) - 1) //Use UBBR for USART baud rate

void init_uart() {
        UBRR0H = (MYUBBR >> 8); //set UBBR high
        UBRR0L = (MYUBBR); //set UBBR low
        //UCSR0C = (1 << UCSZ00) | (1 << UCSZ01) | (0 << USBS0); // 8-bit 1-stop bit mode
        UCSR0B = (1 << TXEN0); //enable transmitter
}

int main(void)
{
   init_uart();
        DDRB &=~ (1<<0)&~(1<<1)&~(1<<2)&~(1<<3);//inputs

        DDRD |=         (1<<4|1<<5|1<<6|1<<7);//output
        PORTD           |=      (1<<4)|(1<<5)|(1<<6)|(1<<7);//set to high
        PORTB  |= (1<<0) | (1<<1) | (1<<2) |(1<<3);//pullups
        DDRB |=(1<<5);//B5 is output
   while (1)
   {
                char key[4][4] = {{'1', '2' , '3' , 'A' }, {'4' ,'5' ,'6' ,'B'}, {'7', '8' ,'9' ,'C'}, {'*' ,'0' , '#'
,'D'}};//keypad characters
                PORTD           |=      (1<<4)|(1<<5)|(1<<6)|(1<<7);//set to high
                for(int i=4;i<8;i++)
                {
                        PORTD &= ~(1<<i);//set to 0
                   for(int j=0; j<4; j++)
                        {
                                if(!(PINB&(1<<j)))
                                {

                                         while(!( UCSR0A & (1<<UDRE0)));
                                         UDR0 = key[i-4][j];//returns the key
                                         _delay_ms(200); //so only one character is printed at a
time


                                        /*PORTB|=(1<<5);//turn LED on
                                        _delay_ms(1000);
                                        PORTB &=~ (1<<5);//turn off led*/
                                  }
                            }

                            PORTD |= (1<<i);//set back to 1
                      }

       }
  }
```