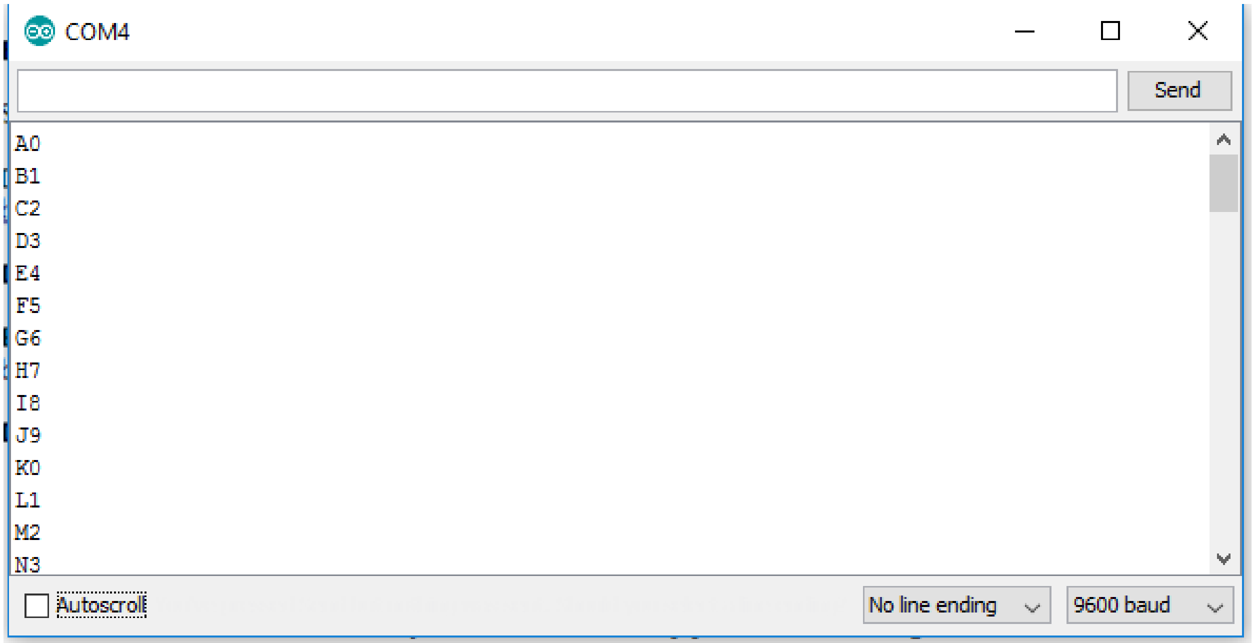
ECE 241 Lab 2 Report Spring 2018
Name: Graham Wood Lab Section A

Objective: The objective of this lab is to understand how to work with and handle ASCII characters and sending said data over a serial port to the Arduino.

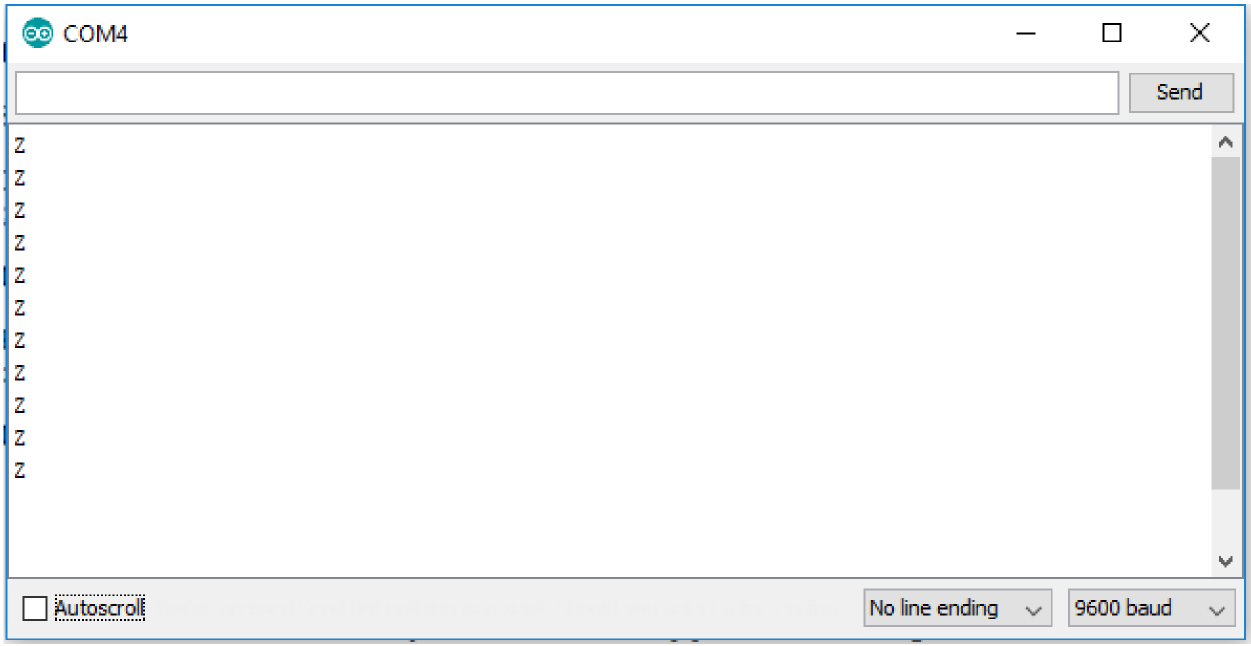
Prelab: A program was written that will serial transmit a sequence of letter and number combos to the Arduino at a 9600 Baud rate (BPS0029

Part 1) The program was uploaded to the Arduino and the output was recorded (seen below) in the console window. The program outputs alphanumeric combination with the letters incrementing from A-Z and the numbers incrementing from 0-9. The program also handles rolling over the numbers and letters once they reach the end of their range. The code that was uploaded to the Arduino can be viewed in the appendix.



```
A0
B1
C2
D3
E4
F5
G6
H7
I8
J9
K0
L1
M2
N3
```

Part 2) For part 2, the program was changed so that it no longer sends out a alphanumeric combination to the Arduino. Instead, the program just sends ‘Z’ every 500ms to the Arduino. This is done so that the transmitting of ‘Z’ can be clearly observed and analyzed. In order to observe and analyze the transmission the Arduino was connected to the logic analyzer and observed in Wave. The resulting observation and output can be seen below. Based on the outputs analyzed from my Arduino the Baud rate calculated come out to be 8,547. Significantly smaller than the set Baud rate of 9600. This equates to an error rate of .1097 or 10.67%.

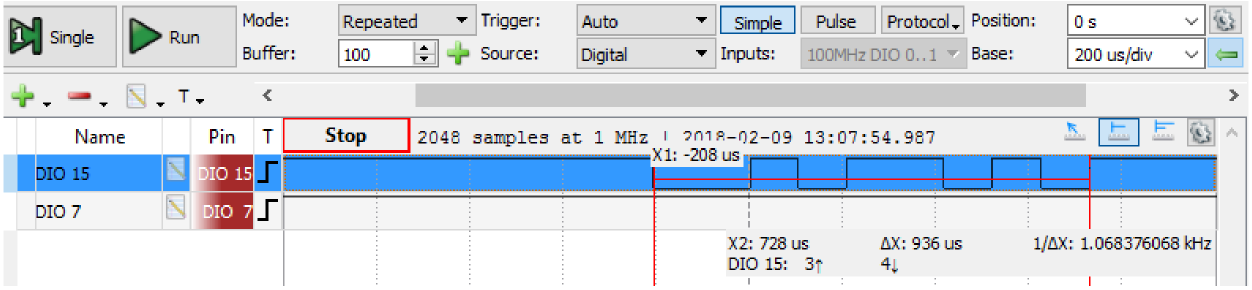


```
Z
Z
Z
Z
Z
Z
Z
Z
Z
Z
Z
```

□ Autoscroll

No line ending

9600 baud



Mode: Repeated

Trigger: Auto

Buffer: 100

Source: Digital

Inputs: 100MHz DIO 0..1

Position: 0 s

Base: 200 us/div

2048 samples at 1 MHz | 2018-02-09 13:07:54.987

X1: -208 us

X2: 728 us
DIO 15: 3↑

ΔX: 936 us
4↓

1/ΔX: 1.068376068 kHz

Part 3

Appendix A Code for Part 1)

```
// =>creates a char variable to manage the letter, creates an int variable, a
long to keep track of time, and the message that will combine the two
char letter = 'A';
int number = 0;
unsigned long timer;
// =>Sets the baud rate to 9600 and the timer equal to millis()
void setup(){
  Serial.begin(9600);
  timer = millis();

}
//=>This method will loop as long as the program is running
void loop()
{
  //=> will check if it has been 500ms since last iteration, then prints the
message to the serial port and increments number and letter.
  if (millis() - timer >= 500){
    Serial.print(letter);
    Serial.println(number);
    letter++;
    number += 1;
    if (letter > 'Z') {
      letter = 'A';
    }
    if (number > 9) {
      number = 0;
    }


    timer += 500;
  }
  //checks if Serial available and if it reads 'R' it resets the transmission
to 'A0'
  if (Serial.available()){
    if (Serial.read() == 'R');{
      letter = 'A';
      number = 0;
    }
  }


}
```

Appendix B for part 2)

```
// =>creates a char variable to manage the letter, creates an int variable, a
long to keep track of time, and the message that will combine the two
unsigned long timer;
// =>Sets the baud rate to 9600 and the timer equal to millis()
void setup(){
  Serial.begin(9600);
  timer = millis();

}
//=>This method will loop as long as the program is running
void loop()
{
  //=> will check if it has been 500ms since last iteration, then prints the
message to the serial port and increments number and letter.
  if (millis() - timer >= 500){
    Serial.print('Z');
    timer += 500;
  }
}
```