05-333: Gadgets Sensors and Activity Recognition in HCI
Reverse Final Exam

Q/A Due Tues 30 April at 11:59pm

Long Questions:

Q1) You're working on a home project over the summer which has you keep track of states through the press of a button. However, you notice that sometimes when you press the button, it skips through multiple counts. What could be the reason as to why that might be? Write a brief function called *corrector()* that would be called in *loop()*, which could be used to rectify this issue. The following boiler code turns makes it so that when you press the button the LED turns off.

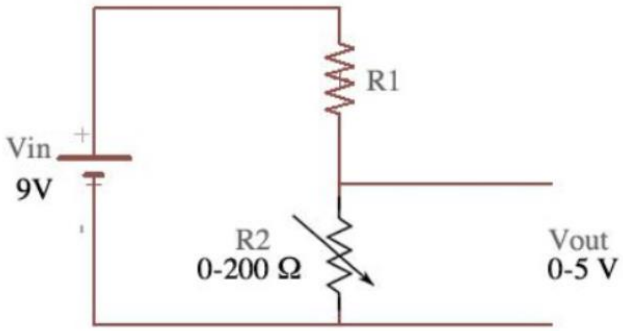
```
const int buttonPin = 2;  // the number of the pushbutton pin
const int ledPin = 13;    // the number of the LED pin


int ledState = HIGH;        // the current state of the output pin
int lastButtonState = LOW;  // the previous reading from the input pin


void setup() {

  pinMode(buttonPin, INPUT);
  pinMode(ledPin, OUTPUT);
  digitalWrite(ledPin, ledState);

}


void loop() {

  int reading = corrector(buttonPin);                    // Implement this

  digitalWrite(ledPin, ledState);
  lastButtonState = reading;

}
```

Q2) Timmy is implementing a keyboard clone for his HCI project where he must map 40 keys to a micro controller. Unsurprisingly there aren't enough pins for him to use. He remembers from class that you're able to map multiple buttons to an analog pin but does remember how. Can you help Timmy by drawing a circuit diagram to help map and explain how to do so?

In this particular case Timmy needs to know how to map 3 distinct buttons to a pin.

Multiple Choice Question

Q1) If your variable resistance sensor R2 ranges from 0-200 Ohms, and you'd like your Vout to be between 0 and 5 Volts, what value should you choose for R1?



a) 140

b) 150

c) 160

d) 170

Q2) As the frequency of change for Alternating Current decreases, how does the resistance in a capacitor change?

a) It increases

b) It decreases

c) It stays the same

d) none of the above

Q3) How would you set D8 with output set to HIGH:

a) DDRB |= 0b000001; PORTB |= 0b000001;

b) DDRB &= ~0b000001; PORTB &= ~0b000001;

c) DDRC |= 0b000001; PORTC |= 0b000001;

d) DDRD |= 0b000001; PORTD &= ~0b000001;

Q4) Let's say you want to output a pulse width with a Duty cycle of 20% out one of the pins of your micro controller. Which line of code correctly does so

a) analogWrite(10, 127)

b) analogWrite(4, 51)

c) analogWrite(10, 51)

d) digitalWrite(10, 51);

Q5) You want to establish long range communication via wire to your device. What is the resistance of a copper wire of 0.5mm radius and length 10 km? [Coppers resistivity is $1.71 \times 10^{-8} \Omega m$]

a) 250 ohm

b) 200 ohm

c) 210 ohm

d) 225 ohm

Answers:

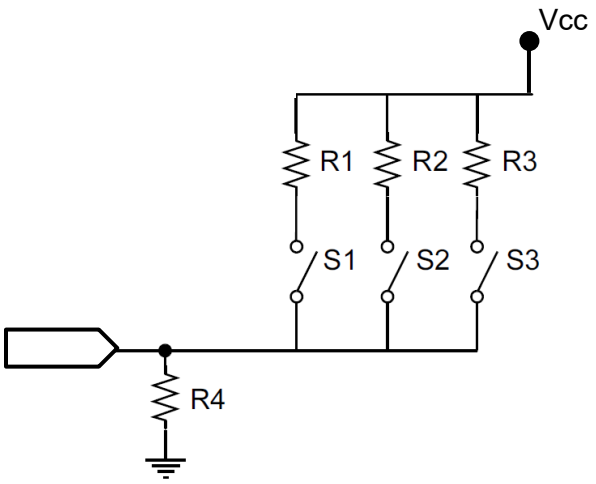
Long Form:

A1) The issue the user is experiencing with the button is due to excessive bouncing. This occurs due to the conductor's elasticity causing it to bounce. We can fix this by implementing a process known as debouncing. Which essentially has you wait for a fixed estimate to let the bouncing pass by. This delay occurs within milliseconds and is not perceivable by the naked eye, which makes it an effective fix.

//Solution Code

```
int buttonState;             // the current reading from the input pin
unsigned long lastDebounceTime = 0;  // the last time the output pin was toggled
unsigned long debounceDelay = 100;    // the debounce time; increase if the output
flickers


int corrector (int ButtonPin){
  int reading = digitalRead(buttonPin);


  if (reading != lastButtonState) {
    lastDebounceTime = millis();
  }


  if ((millis() - lastDebounceTime) > debounceDelay) {
    if (reading != buttonState) {
      buttonState = reading;

      if (buttonState == HIGH) {
        ledState = !ledState;
      }
    }
  }
```

A2) Since Timmy is using an analog pin, he is able to distinguish between voltages. So the goal is to create different voltages for each button. Here is a sample diagram that the tester can come up with.



Here we note that since R1, R2 and R3 are all connected in parallel the total will be different depending on which combination of switches are active. We can set R1 = 1, R2 = 2, and R3 = 3.

We note that each switch illicit a different resistance and hence a different set of voltages to match:

S1 = R1 = 1ohm S1 & S2 = 1/(1/R1 + 1/R2) = 2/3ohm
S2 = R2 = 2ohm S2 & S3 = 1/(1/R2 + 1/R3) = 6/5ohm
S3 = R3 = 3ohm S3 & S1 = 1/(1/R3 + 1/R1) = 3/4ohm

S1, S2 & S3 = 1/(1/R1 + 1/R2 + 1/R3) = 6/11ohm

Multiple Choice:

1 c) 160

Using the voltage divider equation, we can use one of the range boundaries to solve for R1.
Vout = Vin * R2/(R1+R2) where 5 = 9 * 200/(200 + R1)

2 a) The resistance increases as we reduce the frequency of change. This is mainly due to the fact that slow change in flow creates impedance in the circuit due to the capacitor. (Slides 3, page 17)

3 a) DDRB |= 0b000001; PORTB |= 0b000001; Since D8 is the first pin for port B and setting DDRB sets the relevant pin in memory to output, and setting 1 in PORTB sets the output pin to HIGH.

4c) analogWrite(10, 51). We can only send a pulse width to via analogWrite to certain pins. Those pins are 3,5,6,9,10, and 11. And we can set the duty cycle from 0-100% by changing the second argument in analogWrite to 0-255.

5d) 225 Using the resistivity formula $R = \rho L/A$ we plug in the values and get ~ 225 ohms