runlinc Communication Project 1: Telecommunication (STEMSEL Version)

Contents

Introduction

What is telecommunication?

Telecommunication is the transmission of signs, signals, messages, words, writings, images and sounds, or information of any nature via wire, radio, optical or other electromagnetic systems as defined by the telecommunication regulation bodies.

Why is telecommunication necessary?

The importance of telecommunication cannot be understated in the 21st century. Humans have always been communicating information between one another and this brings us together. In the past, communication between groups has always been limited due to the physical limitations of a human being and the time to communicate information. We work and communicate with people close to us because we cannot travel far. But we slowly overcame this challenge as we found new ways to communicate with one another. And as of the 21st century, we are using electromagnetic systems to communicate with one another using telecommunication technologies such as broadband, fibre optics and Wi-Fi. We have slowly reduced the barrier between us by instantly moving information around us where it would once take us months to send a message from one part of the globe to another.

Telecommunication is necessary because information is valuable. We use information to form decisions such as what to wear, what to eat, when to work and what to do. Information is an aid in decision making, and telecommunication helps us gain information quickly, which help us to lower the uncertainties when deciding. Furthermore, we use the information to find new information, which has helped us advance so much. Intuitively, thinking of science, it is the gathering of information that leads to more information. And therefore, we need telecommunication to ensure the passageway between the information and us is optimised.

Summary

We hope that the brief introduction of telecommunication has helped you understand what telecommunication is and the importance of telecommunication, as it is the building block of the modern interconnectedness of our global society.

runlinc Background

runlinc is a web page inside a Wi-Fi chip. The programming is done inside the browsers compared to programming inside a chip. The runlinc web page inside the Wi-Fi chip will command the microchips to do sensing, control, data logging Internet of Things (IoT). It can predict and command.

runlinc and Wi-Fi

Wi-Fi is a telecommunication family of wireless networking technologies which is commonly used for local connection. Wi-Fi works by connecting devices with the host device, which manages the Wi-Fi system, using particular radio waves. These radio waves can provide a high-speed wireless connection between devices. The STEMSEL board without runlinc can only be manipulated with wires such as through the USB port to program the STEMSEL board or port inputs. With the host device connected to the internet, adding a Wi-Fi chip with runlinc system on it, we can turn the STEMSEL board into a wireless board which is suitable for remote control and get the features done, which are stated above.

A Simple Application: Trapped Miners

Problem

Consider a scenario where miners are placing an explosive charge to fracture the rock and make the mine bigger. Suddenly there is a cave-in, trapping the miners with the explosives! There is a programmable microchip on the surface which is connected to the light in the cave visible to the miners. The only way the miners will remain alive is by diffusing the explosives, but the code is unknown to them and this is where you come to their rescue! The only way to communicate with the miners is by using this light, and luckily several of the miners also know the cave-in protocol of using the binary system. Use your knowledge of binary to convey the 4 digit defuse code!

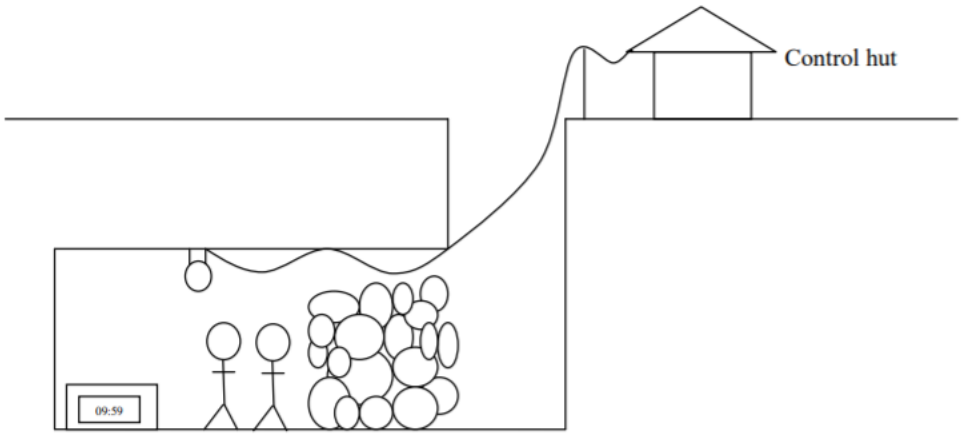

Figure 1 Problem Visualization

Ideas

How can we send the code to the miners by using just the light? How can we ensure the miners know when one number is finished, and we have started to send the next number? Should we send the code just once or multiple times?

Plan

We will use a sort of simplified Morse code to send each number to the miners by flashing the light on and off. To send numbers, we only need to blink the light the same number of times as the number we want to send, so to send a 4 we would blink the light four times. After one number has been sent, there should be a short delay before we start sending the next number. In the end, there should be a long delay before the entire code is repeated in case the miners didn't notice the first time.

Please Note: An example code [3 4 1 8] has been used for demonstration purposes and you should come up with a 4-digit code yourself.

Part A: Design the Circuit on runlinc

Note: Refer to runlinc Wi-Fi Setup Guide document to connect to runlinc

Use the left side of the runlinc web page to construct an input/output (I/O).

In our circuit design,

- C6 -> DIGITAL_OUT: Lightbulb

Figure 2 Expected I/O Configuration.

Part B: Build the Circuit

Use the runlinc I/O to connect the hardware. Remember that turning the screws clockwise will close the clamps and turning the screws anticlockwise will open them. All black wires should go in the negative (-) terminal, red wires go in the positive (+) terminal, and white wires go in the terminal we designated in the runlinc web page port.

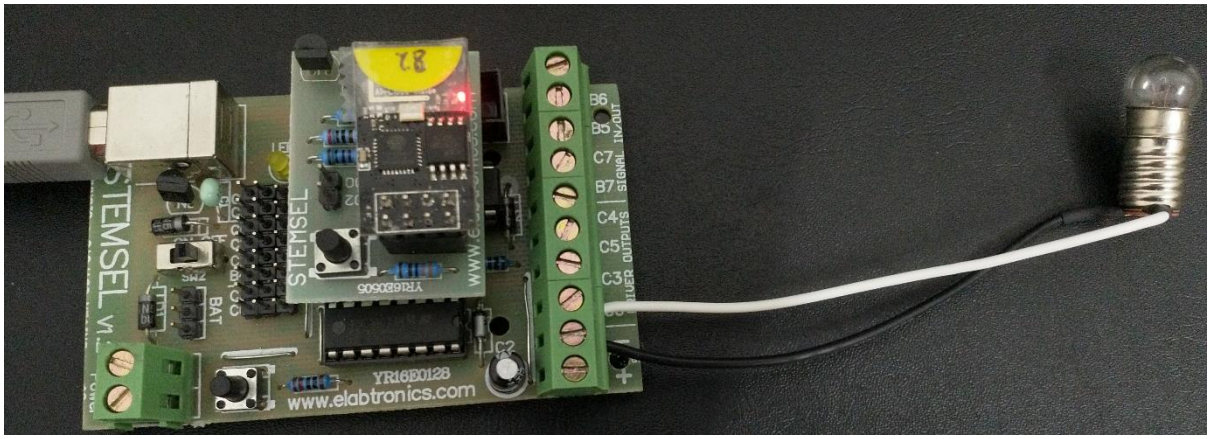

Figure 3 Circuit connection on STEMSEL.

Wiring Instructions

- Connect All White Wires to their Respective Pins
  - Lightbulb -> C6
- Connect All Black Wires to negative (-) terminal.

Part C: Program the Circuit

HTML

We will first set up a simple HTML page to show what the miners will see, such that we can confirm it is the message we want to send. We will use a circle to mimic the light bulb. Let's label the circle with a heading, so we remember that the circle represents the lightbulb.

```
<h1>This is what the miner will see.<h1>
<svg height="100" width="100">
  <circle id="circle" cx="50" cy="50" r="40" stroke="black" stroke-width="3" fill="white" />
</svg>
```

JavaScript

We will write a function in JavaScript that will help us blink the lightbulb at a certain amount of time. Using the input of number of repeats, we will do a loop for several repeats. When the Loop starts, we will turn on the light for 1 second. Then turn off the light for 1 second. As repeating the loop until the input repeat is met, this loop will be done with a for loop. We will return a promise resolve to signify that blink function has blinked for the repeated amount.

```
async function blink(repeats){
  //Loop
  for(let i = 0; i<repeats; i++){
    //Turn On
    turnOn( Lightbulb );
    document.getElementById("circle").style.fill="gold";
    await mSec( 1000 );
    //Turn Off
    turnOff( Lightbulb );
    document.getElementById("circle").style.fill="white";
    await mSec( 1000 );
  }
  return Promise.resolve();
}
```

JavaScript Loop

The JavaScript Loop will send the code to the miners using our blink function. We will type in blink(3); to blink for 3 times to tell the miner that the first digit is 3. But we need to wait for blink function to finish. Afterwards, we will add a delay of 3 seconds to notify the miner to prepare for the next number of the code sequence. Repeat this for the next 3 digits of the code. However, at the last number of the sequence, add the delay of 10 seconds instead of 3 seconds to signify the sequence is done and it will repeat.

```
await blink(3);
await mSec( 3000 );
await blink(4);
await mSec( 3000 );
await blink(1);
await mSec( 3000 );
await blink(8);
await mSec( 10000 );
```

Expected runlinc code and web page:

Figure 4 Expected runlinc

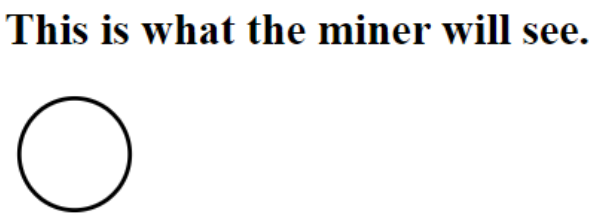

Figure 5 Expected website.

Further Thinking

What if there was a zero in the code? How would you send a 0 on your transmitter? You can't just leave the light off to send a 0 since it is likely that it will be considered mistakenly for the delays between numbers or at the end of the code. Instead, it would be better to send ten flashes since this will be different from both the other numbers and the delays between them. Can you think of another way to send a zero?

Summary

If given a tedious and repetitive task, a human would become bored and possibly make mistakes. A microchip on the other hand never gets bored and can continue doing the task all day. Because of this, microchips are very good at sending signals repeatedly. At the end of this exercise, you should be able to understand the simple application on how simple the telecommunication (sending information through a wire and thus lightbulb) is and its importance.