CIRCUITS TO SOFTWARE: AN EXPLORATION OF COMPUTING DEVELOPMENT IN THREE STAGES.

Elizabeth Forrest Curry, Mathematical and Computational Sciences
Advised by Dr. Dan Palmer

Project Overview

Goal: Showcase the development of computer science following the throughline of foundational digital logic.

Process: Create three different implementations of the same puzzle to showcase three different 'stages' of computing complexity:

1. **Raw Circuit**: Physical lights and buttons controlled by physical logic components.
2. **Arduino Circuit**: Physical lights and buttons controlled by code loaded onto an Arduino board.
3. **Software**: A version of the game made entirely in Python that receives input from a GUI.

Puzzle Game

Puzzle Requirements:

- Four buttons
- Four lights
- Each button toggles some subset of lights
- To win all lights must be turned on

Base Case Game:

Button A controls the first light, button B controls the first two lights, button C controls the first three, and button D controls all lights.

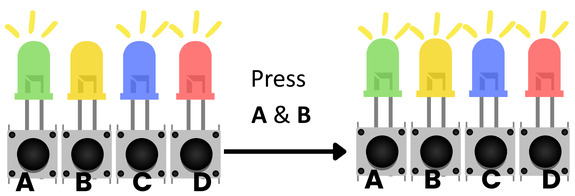

Figure 1. Example puzzle solution

Raw Circuit

Components:

- **D Flip Flops** (implemented as a toggle)
  - Takes an input signal and every time the input is asserted the output is toggled either high or low
- **Inverting Schmitt Trigger**
  - Takes an imperfect input signal (bouncing from push button) and outputs the inverse signal with a sharp rising or falling edge
- **XOR Gate**
  - Compares two input signals, outputs high if and only if one input is high and the other is low

Chalenges & Limitations:

- Button Bouncing: button inputs are imperfect, requiring special component
- Rigid structure, little room for changes
- Required specialized knowledge

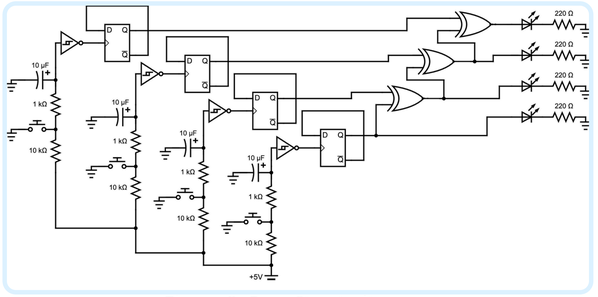

Figure 2. Raw Circuit diagram

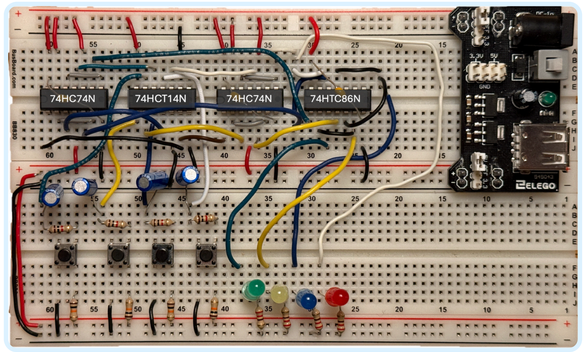

Figure 3. Image of Raw Circuit

Arduino

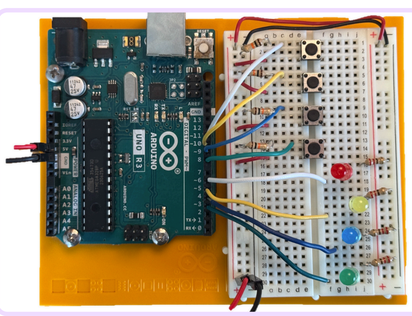

Figure 4. Image of Arduino Circuit

Software:

- Code in Arduino IDE computes logic based on inputs and outputs. It is loaded onto the microcontroller on the Arduino
- Uses rudimentary logic only capable of the base case version of the puzzle

Hardware:

- The Arduino itself, lights, and buttons used for inputs and outputs

Chalenges & Limitations:

- Button Bouncing: solved using a buffer in the software
- Specialized uses of Arduino IDE
- Could be expanded to compute more complex puzzle variants

```
// Blinking LED

const int LED = 13; // LED connected to
                    // digital pin 13

void setup()
{
  pinMode(LED, OUTPUT);     // sets the digital
                            // pin as output
}

void loop()
{
  digitalWrite(LED, HIGH);   // turns the LED on
  delay(1000);               // waits for a second
  digitalWrite(LED, LOW);    // turns the LED off
  delay(1000);               // waits for a second
}
```

Figure 5. Example of Arduino Code structure [1]

Software

Structure:

- Python code using: random, datetime, Tkinter, Numpy, and Image and ImageTk from Pillow
- Randomizes both the initial light state and actions of buttons
- Ensures solution using finite linear systems in the space $\mathbb{Z}_2$such that the only possible entries In a vector are 0 or 1.

Linear Systems:

If matrix representation is linearly Independent then a solution is accessible [2]

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \xrightarrow[\text{Row}]{\text{Reduced}} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Figure 6. Row reduction of base case game

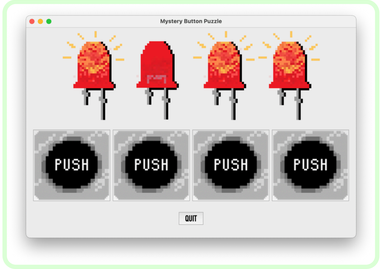

Figure 7. Image of software implementation

Conclusion

This project showcases how the underlying logic of computing remains the same despite abstracting away complexities.

Implementation	Effort	Flexibility
Raw Circuit	8	2
Arduino	4	5
Software	1	10

References:

[1] Massimo Banzi and Michael Shiloh. *Getting Started With Arduino*. en. 4th. Make: Community, Feb. 2022. isbn: 9781680456929. url:https://learning.oreilly . com / library / view / getting - started - with / 9781680456929/(pages 16–19).

[2] David Poole. "Finite Linear Games". In: Linear *Algebra: A Aodern Introduction*. 3rd ed. Boston, MA: Brooks/Cole: Cengage Learning, 2011, pp. 115–124. isbn: 978-0-538-73545-2 (pages 5, 7, 9).

THE COLLEGE OF WOOSTER