**Lab 3 Report**

Sage Marks, Matt Krueger

3360:0001 - Embedded Systems

Professor Beichel

University of Iowa, College of Engineering

## Introduction

Embedded Systems Lab 3 builds on the concepts and circuitry implemented in lab 2. Once again, we are using the 74HC595 Shift Register, 5161AS 7-Segment Display, and an enable low push button. However, the focus of this lab was working with timer/counters in the Atmel studio environment as delay loops, lookup tables, and utilizing an RPG for user inputs. The goal of the lab using the hardware described below was to create a simple electronic door lock system that accepts a 5-digit hexadecimal code. Digits selected for the code are to be scrolled through with the RPG and then selected with a push button press of less than one second. In this report, we will outline the new concepts that were employed to achieve this result and focus less on information from the prior lab.

Listed below is an exhaustive list of hardware required to replicate the lab 3 circuit. Please note that an Arduino Uno contains the microcontroller needed and is sufficient for use in lab. We opted to use the Arduino Uno for simplicity when developing the circuit.

|  |  |  |
| --- | --- | --- |
| Hardware | Quantity | Description |
| Atmega 328P µC | 1 | Programmable µC |
| 74HC595 Shift Register | 1 | Storage of hex codes for 7-Segment display |
| 5161AS 7-Segment Display | 1 | Display current counter |
| Enable Low Push Button | 1 | Enables user input interaction with display |
| 560Ω Resistor | 8 | Resist current into 7-Segment display LEDs |
| 10KΩ Resistor | 5 | RC low pass filtering, Pull-up resistor for RPG |
| 100KΩ Resistor | 1 | Pull-up resistor for push button |
| 0.1µF Capacitor | 4 | µC Decoupling & push button RC low pass filtering |
| EVE-GA1F2012B Encoder | 1 | Rotary pulse generator |

Figure 1: Materials List

## Schematic

A computer screen shot of a computer program

AI-generated content may be incorrect.

Figure 2: Electrical circuit schematic created using KiCAD

The lab 3 circuit was built off the existing 7-segment display, shift register, and push button circuit utilized in lab 2. The main difference is found in the addition of the RPG and two new input lines to the microcontroller.

The pinout of the RPG is as shown below with pin A being the A signal, pin B being the B signal and pin C being the signal ground.

A screenshot of a computer

AI-generated content may be incorrect.

The wiring of the RPG is as follows. To read the signals from pins B and A we connected pin B to PD7 and pin A to PD6 and configured these as inputs. Each pin, A and B, has a low pass RC filter connected to help with debouncing. These were created with 0.1 microfarad capacitors and 10-kiloohm resistors. Then there is also a 10-kiloohm resistor that acts as a pullup resistor keeping the pins at defined states when the RPG is not being turned. Pin C is tied to the ground signal of the microcontroller.

Reading of the RPG was dependent on the signals that the RPG generates when it is turned either clockwise or counterclockwise. When turning the RPG, you can hear clicking sounds and there are defined points or grooves within the RPG called detents. These detents and the values (1 or 0) of the two signals are crucial to reading user RPG inputs. A signal being “off” is represented by 1, and a signal that is “on” is represented by a 0. When turning the RPG clockwise the B signal is the first to be on, followed by the A signal because of a 90-degree phase lag. When turning counterclockwise the A signal is the first to be on, followed by the B signal because of the 90-degree phase lag. Following a gray code system, which is utilized so that you are only changing one bit at a time from 1->0 or vice versa we can see the order that the signals will change when turning each direction. This knowledge is helpful when trying to decide if the current displayed digit on the 7-segment display should be incremented or decremented.

A clockwise rotation pattern for the signals in the order BA is as follows,

00 -> 01 -> 11 -> 10 -> 00 …

A counterclockwise rotation pattern for the signals in the order BA is as follows,

00->10->11->01->00 …

The difference in these signal patterns becomes relevant in the code section. A detent that represents the end of a “turn” of the RPG is the signal combination 00.

## 3. Discussion

Lookup table and modified finding

## 4. Conclusion

## 5. Appendix A: Source Code

## 6. Appendix B: References

Beichel, Reinhard. *Embedded Systems Lab 2, ECE:3360. The University of Iowa, 2025* <<https://uiowa.instructure.com/courses/248357/files/29567265?module_item_id=8134634>>

Beichel, Reinard. *Embedded Systems, Rotary Pulse Generators and Lab 3. The University of Iowa, 2025 <*[*https://uiowa.instructure.com/courses/248357/files/29778930?module\_item\_id=8162158*](https://uiowa.instructure.com/courses/248357/files/29778930?module_item_id=8162158) *>*

Beichel, Reinard. *Embedded Systems, Lecture 8: Timers. The University of Iowa, 2025 <*[*https://uiowa.instructure.com/courses/248357/files/29853139?module\_item\_id=8165716*](https://uiowa.instructure.com/courses/248357/files/29853139?module_item_id=8165716) *>*

Components101. *7 Segment Display. 22 September 2019.*  <<https://components101.com/displays/7-segment-display-pinout-working-datasheet>>

Pighixxx. *The Definitive Arduino Uno Pinout Diagram*. *May 5, 2013. <*<https://uiowa.instructure.com/courses/248357/files/29320694?module_item_id=8042318>>

Texas Instruments. *SNx4HC595 8-Bit Shift Registers With 3-State Output Register. September 2015*. <<https://www.ti.com/lit/ds/symlink/sn74hc595.pdf>>

XLITX. *5161AS Datasheet <*<http://www.xlitx.com/datasheet/5161AS.pdf>>