

Project Proposal: Punched Card Reader

Yi Lyo, Patrick McCann, Alexander Thaep, Zhiwen “Michael” Zheng

October 10, 2025

Project Overview

We propose designing and implementing an Arduino-based punched card reader and puncher for our project. This device will help Malte, the instructor of CSCI 1670: Operating Systems, motivate certain OS concepts by demonstrating how slow historical I/O devices were. Please note that the earliest date that we could schedule a meeting with Malte was 10/14. Certain aspects of our design may change once we’ve met with Malte and have a firmer understanding of Malte’s requirements.

The reader will process a stack of punched cards by feeding them through an array of photointerrupters, using *timer interrupts* to drive stepper motors and precisely control the speed of the cards’ passage [9]. The analog readings from the sensors will be fed into the Arduino’s *Analog-to-Digital Converter (ADC)* to determine the arrangement of holes on the card. An *interrupt service routine (ISR)*, triggered by a sensor that detects whether a card has been inserted or by input from the host machine, will initiate the reading process.

The data read off the punch card will be transmitted to a host machine through serial communication. A program running on the host machine will receive the data and perform useful operations with it, such as displaying it on the screen or writing it to a file. A watchdog timer will be implemented as a fail-safe mechanism to reset the system in cases of malfunction, thereby ensuring the system’s reliability.

The puncher will be located on the same feed as the reader, and the device will either punch or read, depending on its mode. When in punching mode, the device will receive data from the host machine, automatically feed cards from a stack of blanks, and punch the data onto them.

History

Though punched cards can be traced back to even the late 19th century, the most iconic and arguably most ubiquitous punched card design comes from IBM in the early 19th century. The card itself was a thin piece of stiff cardboard measuring 7.375 inches by 3.25 inches packing 80 columns by 12 rows of tiny rectangular holes. Historically, a single card would contain a line of code in a program. So, these stacks of cards would comprise multitudes of encoded programs and data that once made the world go round in the early Information Age.

As such, punched card readers were once essential equipment for data processing as late as the 1960s. With speed measured in cards per minute, or CPM, early card readers used mechanical brushes that would make electrical contact for the presence of a hole, and no contact for the absence. Later punched card readers would adopt optical sensors to reduce complexity inherent in electromechanical components for more robust readers.

Design Rationale

Our project will leverage the Arduino’s built-in features, including *Timer Interrupts* to drive a stepper motor, which controls the rate at which the card is fed through the reader, and an *Analog-to-Digital Converter (ADC)* to read the analog signals from the photointerrupters [10]. These two options provide the most well-documented and robust solutions to our problem of parsing the punched card reader.

Based on an event-driven approach, the system will utilize an *Interrupt Service Routine (ISR)* to handle card insertions, thus allowing the device to initiate the reading process without continuous polling.

A *watchdog timer* will be implemented as a fail-safe to reset the system so that, in the event of a malfunction, the software can reboot itself without human intervention.

Once the data is read, it will be transmitted to a host machine via *serial communication*, making the information accessible and fulfilling the device’s input functionality. A program on the computer will process the data, such as printing out the contents of the card on the display or writing it to a file or memory buffer.

In addition, the puncher will receive data from the host machine via *serial communication* and use *timer interrupts* to automatically punch it onto punch cards, thereby fulfilling the device’s output functionality.

Deliverables and Fulfillment Criteria

A fully assembled, self-contained punched card reader prototype (though requiring a host machine) will be presented. The prototype will consist of an Arduino Uno R4 WiFi board, an array of photointerrupters, and stepper motors, together with the wires and other auxiliary components integrated into a physical device. The project’s success will be evaluated based on explicit criteria. When a punched card is inserted into the device, the prototype should detect the card using the photointerrupters, pull it through the sensor array at a controlled speed, and transmit the data encoded by the punches to the host machine via serial communication. The host machine shall receive data consistent with the decoded pattern of holes on the physical card that was read, demonstrating the device’s functionality as a reliable I/O peripheral. The device will be used to perform a short educational demonstration that matches Malte’s specifications (e.g., a realistic I/O rate, easy for onlookers to understand, etc). Finally, we will ensure that the device is well-documented and understood by Malte so that it can be easily maintained and reused in future semesters.

Hardware and Testing Plan

The following specifies the hardware components that we plan to use and the testing schema we expect to implement:

- *Arduino Uno R4 WiFi boards*: We will check whether the board is functional by running the Blink example, which will blink the built-in LED[1]. Other built-in examples can be run as well for further validation [2].
- *Array of photointerrupters*: We will test each photointerrupter by connecting it to an analog input pin and printing its readings to the Serial Monitor using `analogRead()` [3]. We will then see if there is a noticeable and sufficient change in the output value as we pull a punched card over the sensor.
- *Low-power motors*: We will verify each motor’s functionality by connecting it to the Arduino and using a standard stepper motor driver sketch. If the motors rotate as expected, then our motors are deemed functional.
- *Solenoids*: We will verify each solenoid’s functionality by driving it with the correct voltage and current and confirming that the solenoid actuates. If the solenoid actuates in the correct direction, then our solenoids are deemed functional.

Uncertainties

There are several challenges that we anticipate, especially in electromechanical integration:

- On the hardware side, achieving precise yet necessary physical alignment between the photointerrupter array and the punched card rows and/or columns, while ensuring that the stepper motors push the card forward at a reasonable speed without deviating it from the track, appears to be a hurdle.
- In addition to reading a stack of punched cards, we will also need to design a mechanism that can feed one card at a time through the reader. This may be somewhat difficult.
- To punch the cards, we will also need to design and build a hole punch array. There may be mechanical difficulties in doing this.
- On the software side, synchronizing the reading from the ADC and the card’s physical position may require non-trivial timing analysis.

A fallback solution to difficulties in *motor driving* is to manually pull the card to the correct position before reading the array of punches, thereby alleviating both hardware and software concerns. On top of that, a fallback solution to difficulties in *the punch mechanism* would be to forgo the puncher part of this project, and instead, punch cards by hand for the reader to read.

Preliminary List of Components to be Ordered

We have a few components, in addition to the Arduino board and the electronics kit, that need to be ordered:

- *An array of photointerrupters* that (1) detect and read the punches/holes of a punched card and (2) transmit the sensing information to an Arduino board [5].
- *Stepper motors* and *stepper motor drivers (driver boards)* that drive a punched card forward at a reasonable speed [6].
- *Cardstock* to manufacture punched cards [7].
- *Solenoids* to drive the hole punches [8].

References

- [1] <https://docs.arduino.cc/built-in-examples/basics/Blink/>
- [2] <https://docs.arduino.cc/built-in-examples/>
- [3] <https://docs.arduino.cc/language-reference/en/functions/analog-io/analogRead/>
- [4] <https://docs.arduino.cc/language-reference/en/functions/analog-io/analogWrite/>
- [5] <https://www.digikey.com/short/dhfpz8qc>
- [6] <https://a.co/d/bHqxbUe>
- [7] <https://ebay.us/m/qKRkb9>
- [8] <https://www.adafruit.com/product/2776>
- [9] <https://techweb.rohm.com/product/opto-electronics/photointerrupters/23775/>
- [10] <https://www.monolithicpower.com/en/learning/resources/stepper-motors-basics-types-uses?srsltid=AfmB0opcXVb-FUmDtTQE1R5hi9FugzaETDaga7aimZ2I1YAvkKzrgd9q>