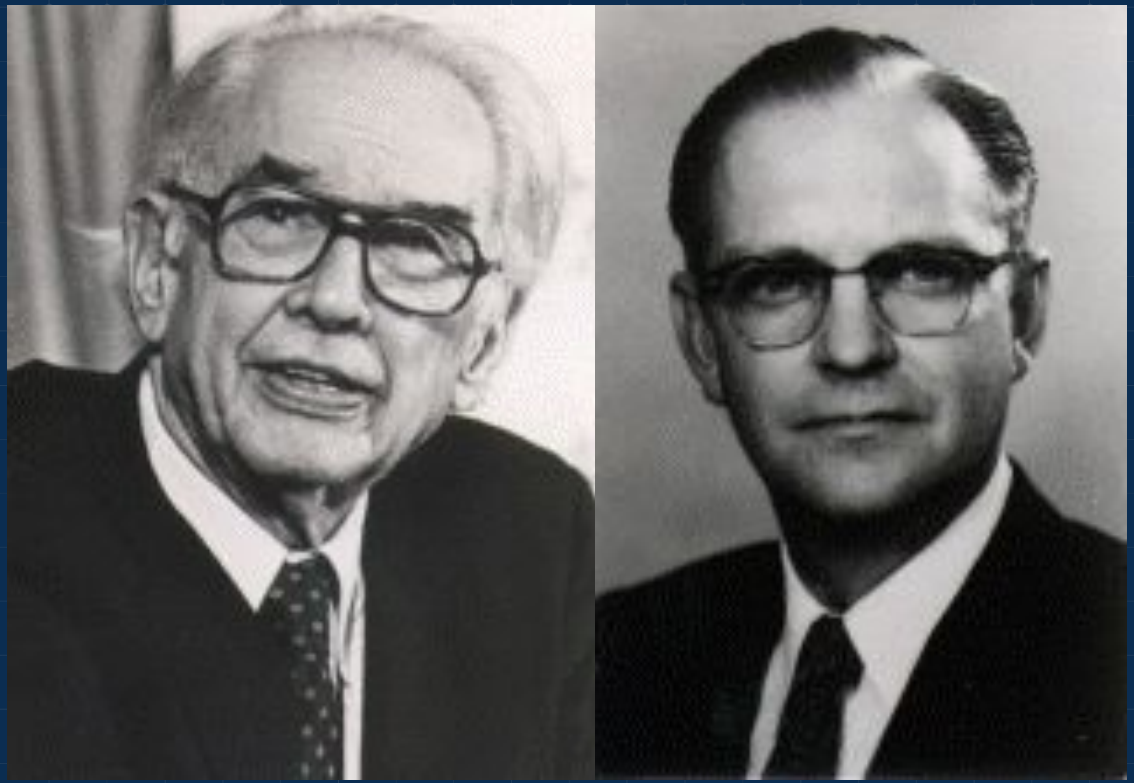
Implementing the ABC Computer with Modern Technology (Phase 2)

sdmay26-12

Samuel Schroeder, Ryan McFadden, Bryan York
Kyle Koch, Vance Arnold, Gavin Bunker

Faculty Advisor: Dr. Stoytchev

The Creators

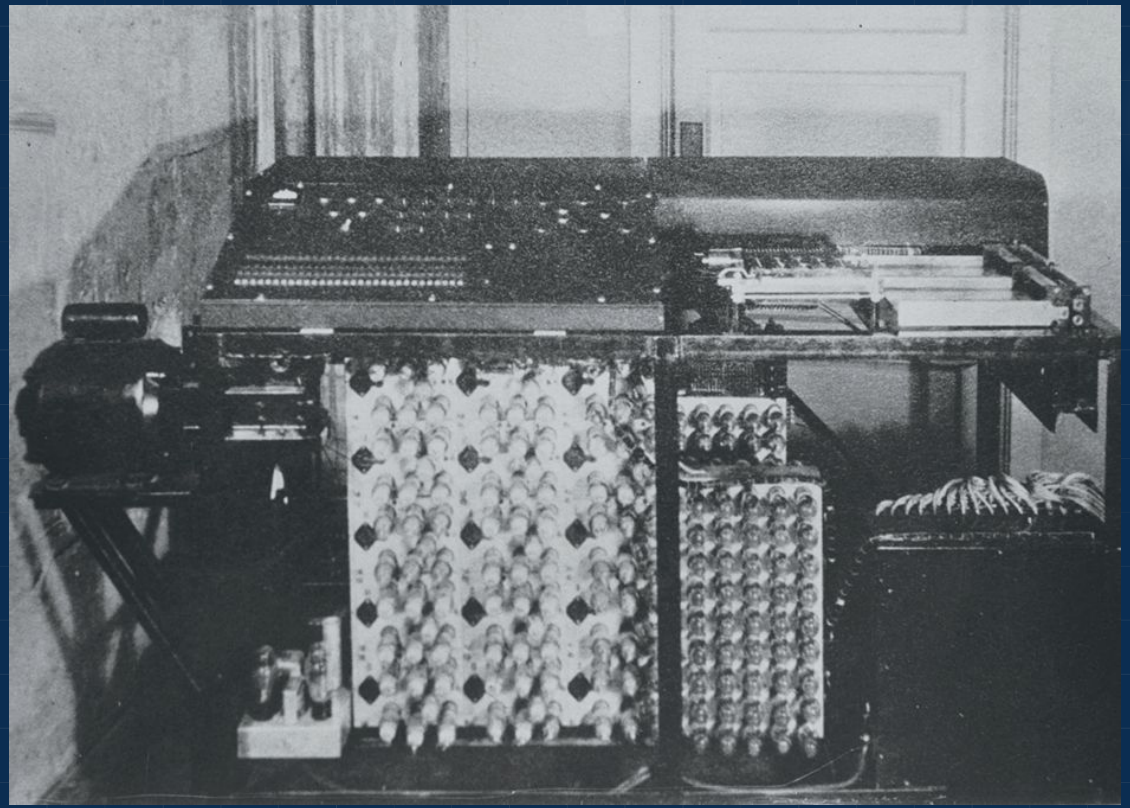


John Vincent Atanasoff
https://jva.cs.iastate.edu/jvabio.php

Clifford Berry
https://jva.cs.iastate.edu/berrybio.php

Atanasoff-Berry Computer

ABC, built
1939-1942



https://jva.cs.iastate.edu/operation.php

Technology



Rotating Memory Drum
https://jva.cs.iastate.edu/operation.php



Vacuum Tube Circuit
https://en.wikipedia.org/wiki/Atanasoff%E2%80%93Berry_computer

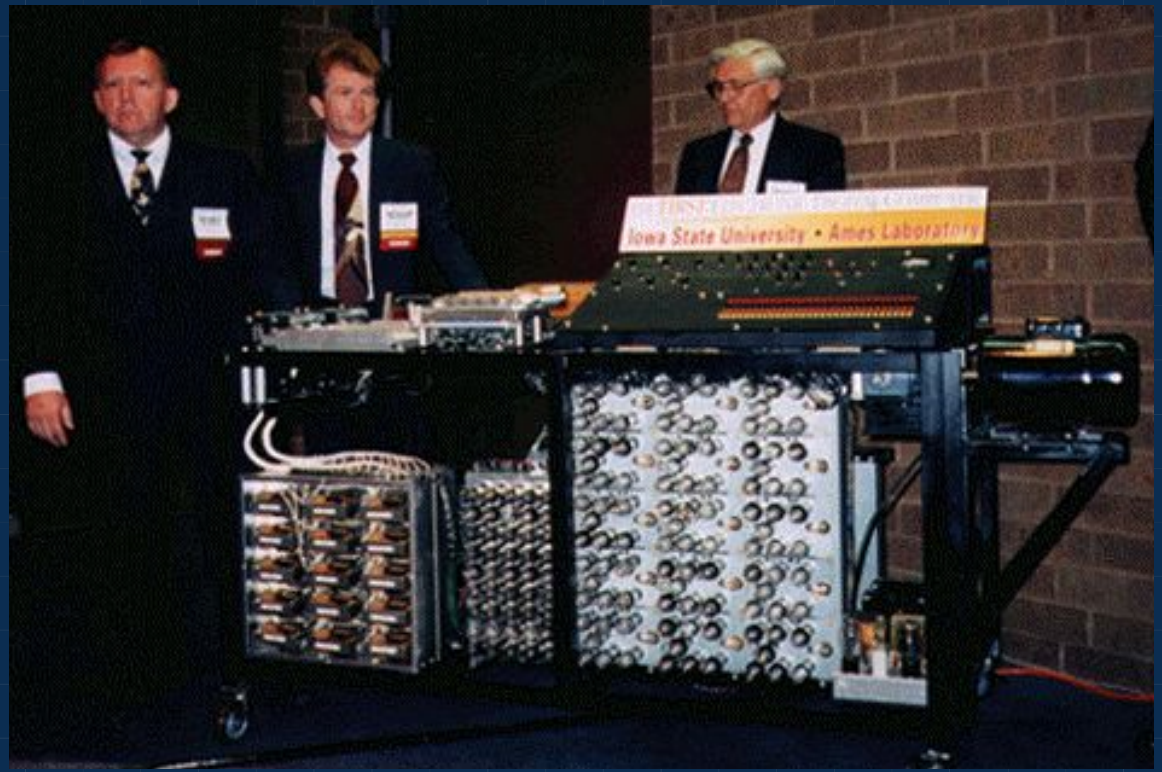
ABC Reconstruction (Ames Lab, 1994-1997)

Successes:

- Original mechanical design
- Looks like the original ABC

Shortcomings:

- $300,000 budget (~$650,000 today)
- Lacks technical documentation
- No longer at ISU



https://jva.cs.iastate.edu/team.php

Project Directors: John Gustafson and Del Bluhm

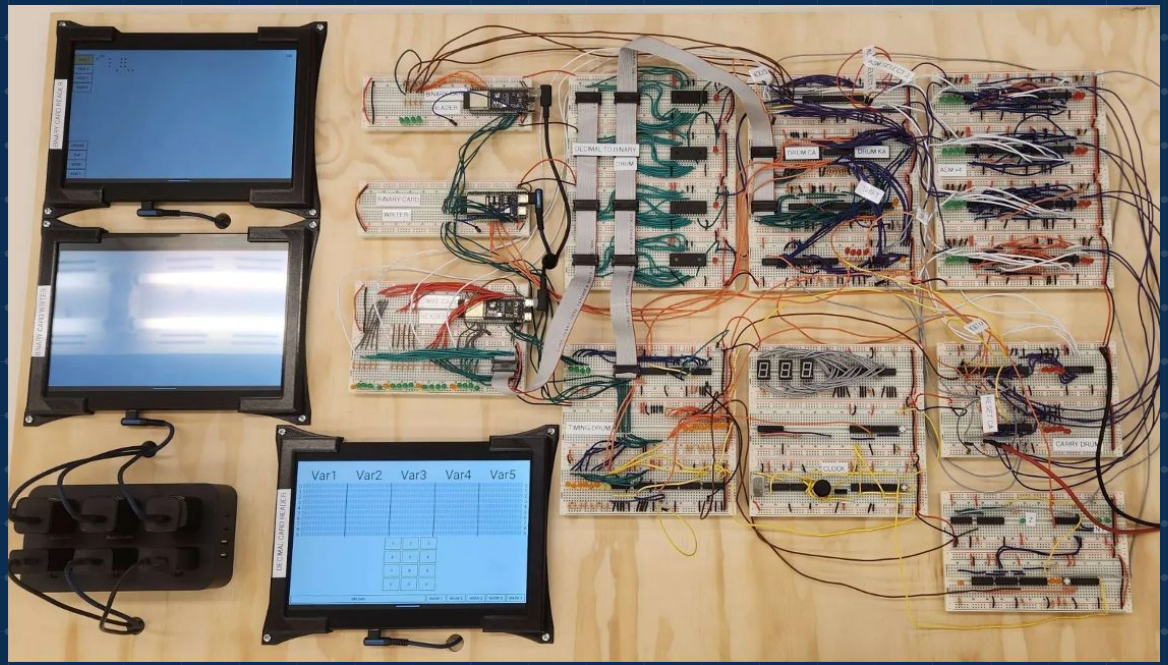
Partial Breadboard Recreation - sdmay25-19

Successes:

- Incredible discovery work on ABC's design
- Created proof of concept designs for most modules

Shortcomings:

- Scaled down
- Did not implement all of the modules of the ABC
- Very difficult to operate



https://sdmay25-29.sd.ece.iastate.edu/

Team Members: Connor Hand, Zach Scurlock, William Mayer, Noah Butler, and Peter Hurd

Problem Statement

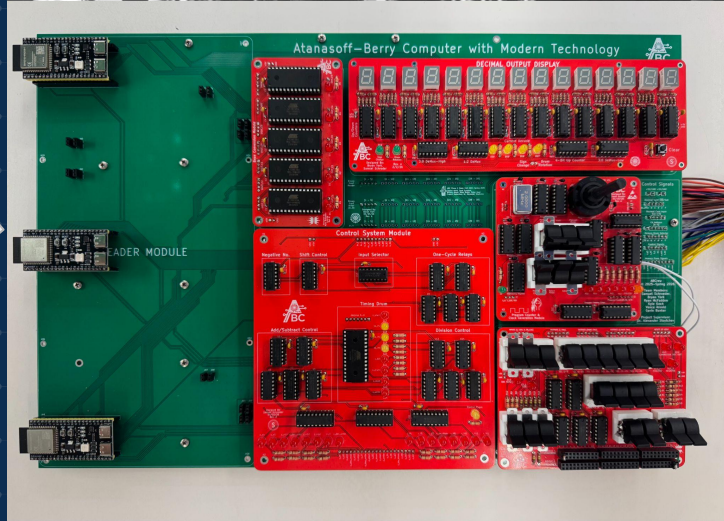
Implement the logical design of the ABC using modern components

Design PCBs for each of the ABC's modules and integrate them onto a motherboard for a full-scale logical recreation of the ABC

Purpose:

- Restore lost history
- Teach students and ISU visitors about computing logic and design





https://jva.cs.iastate.edu/berrybio.php https://sd-archive.ece.iastate.edu/sdmay24-14/

Requirements

Functional

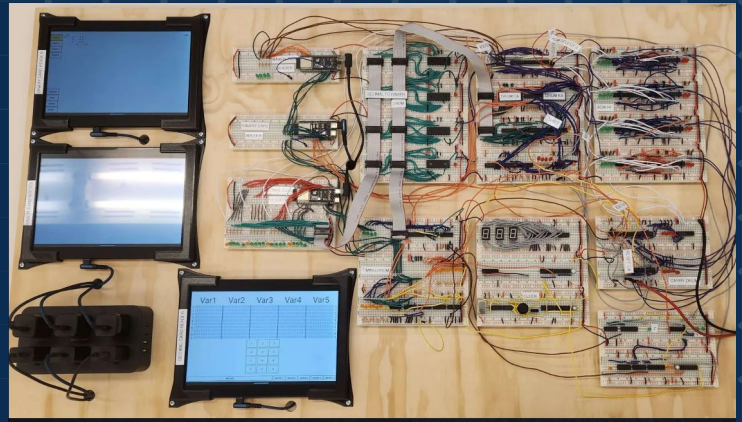
- Recreate the original logic of the ABC as accurately as possible
- Design for full 29x29 system of equations, fabricate scaled-down version for 5-coefficients

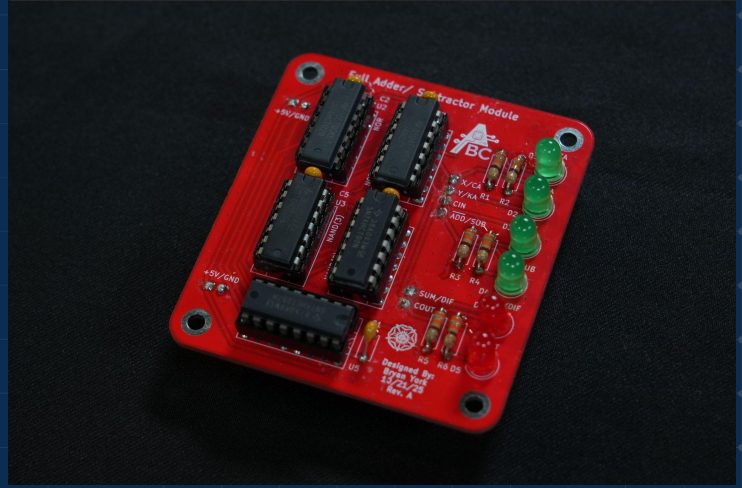
Physical

- Clean, modular design
- Full system must take only 1-person to transport

User Experience

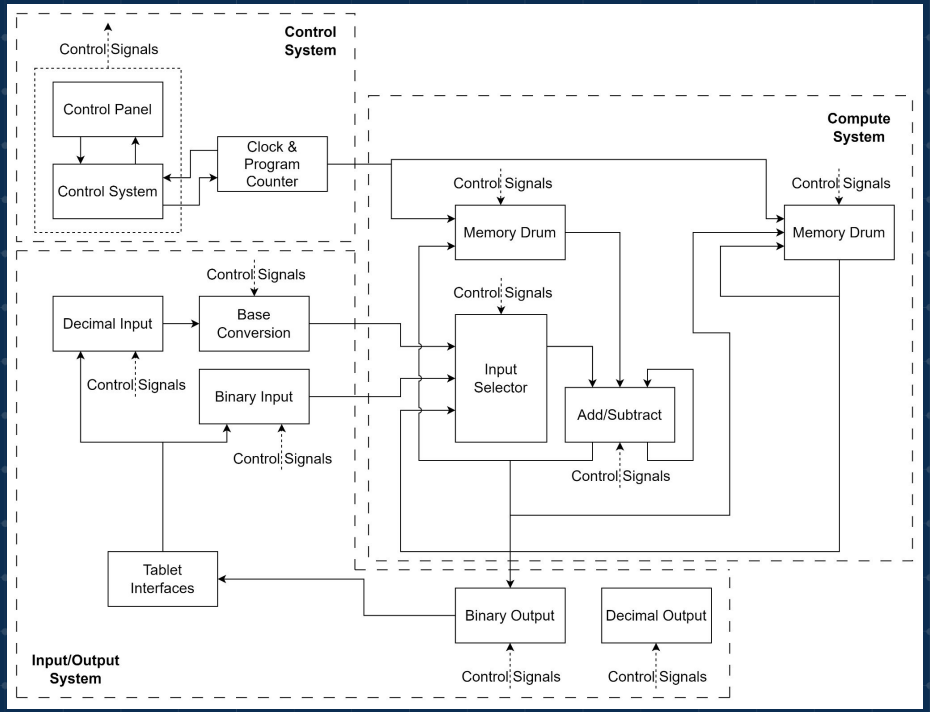
- Provide operation instructions and thorough documentation for future users
- System is visually appealing and informative





Design & Implementation

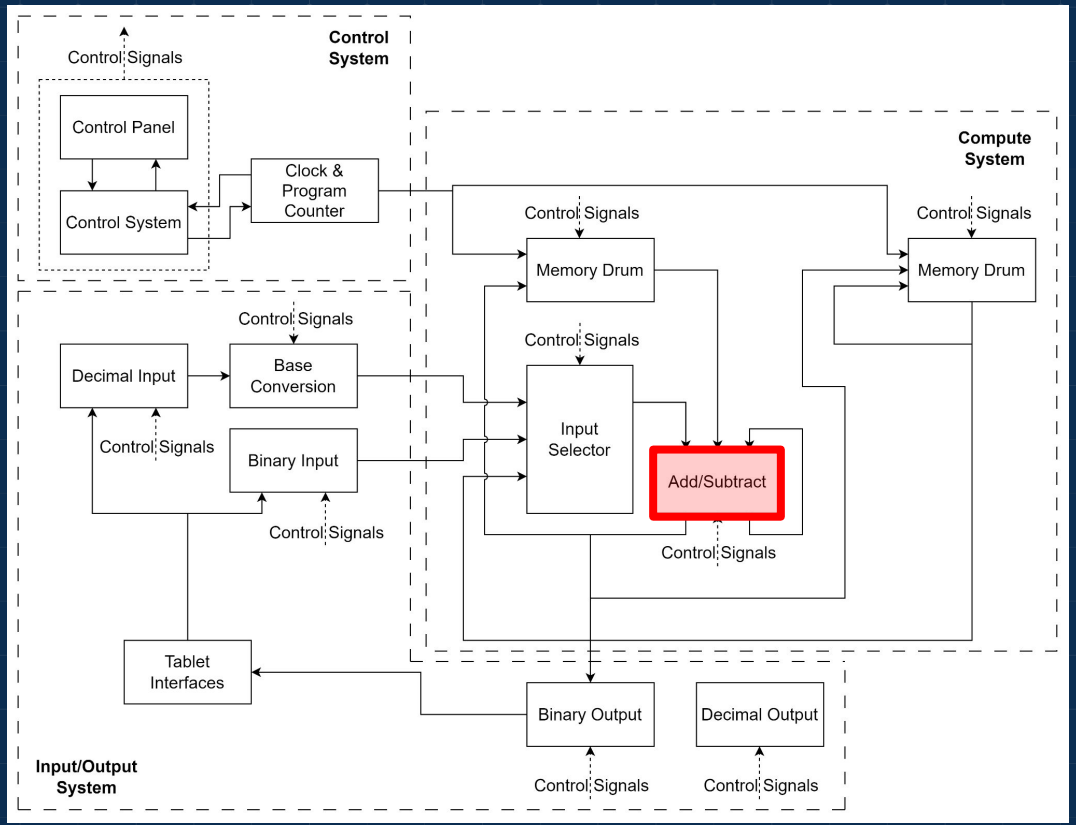
High Level Design



Design & Testing Process

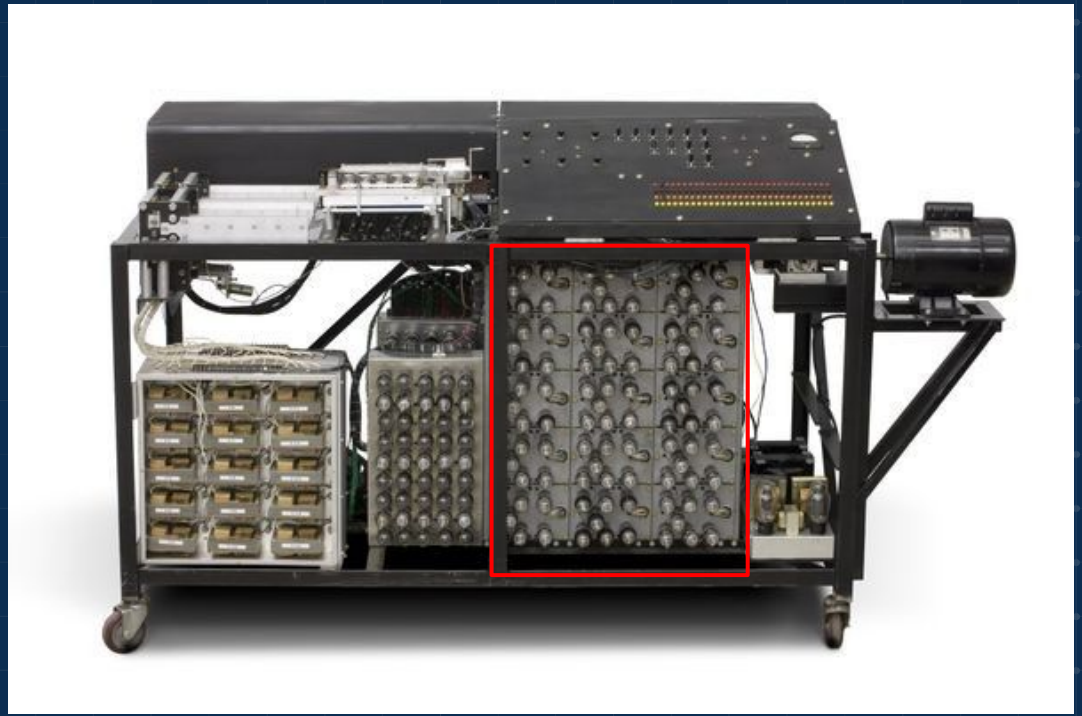


Add/Sub Module

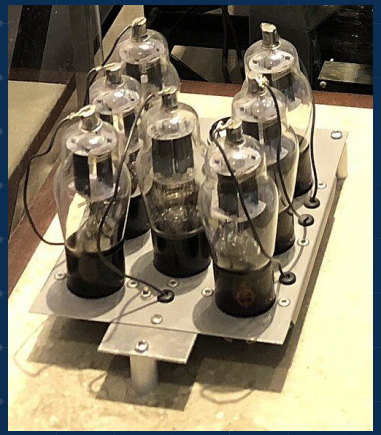


Add/Sub Module: Design

1997 ABC Replica



https://sdmay25-29.sd.ece.iastate.edu/



https://en.wikipedia.org/wiki/Atanasoff%E2%80%93Berry_computer

Add/Sub Module: Design Verification

```
entity AddSubModule is
port (x, y, cin, AddSub: in std_logic; Sum, Cout : out std_logic);
end entity AddSubModule;

architecture Structural of AddSubModule is

    -- Declare internal signals
    signal P1, P2, P3, P4, P5, P6, P7, P8, P10, P11,
    P12, P13, m1A, m1B, m4a, m4b : STD_LOGIC;
        begin
        P1 <= NOT x;
        P2 <= NOT y;
        P3 <= NOT cin;
        P4 <= NOT P1;
        P5 <= P2 NAND P3;
        P6 <= P2 NOR P3;
        P7 <= NOT P6;
        P8 <= NOT (P4 AND P5 AND P7);
        P10 <= NOT(P1 AND P5 AND P7);
        P11 <= P4 NAND P8;
        m1a <= P7 NOR P10;
        m1b <= P10 NAND P7;
        m4a <= P7 NAND P8;
        m4b <= P8 NOR P7;
        P12 <= ((NOT AddSub) AND m1a) OR (AddSub AND m1b);
        P13 <= ((NOT AddSub) AND m4a) OR (AddSub AND m4b);
        Cout <= NOT (P12 NOR P13);
        Sum <= P10 NAND P11;


end Structural;
```

Addition Implementation

F = X + Y + Cin

Subtraction Implementation

F = (X-Y) -Cin

Inputs:

X => 1'b0

Y => 1'b1

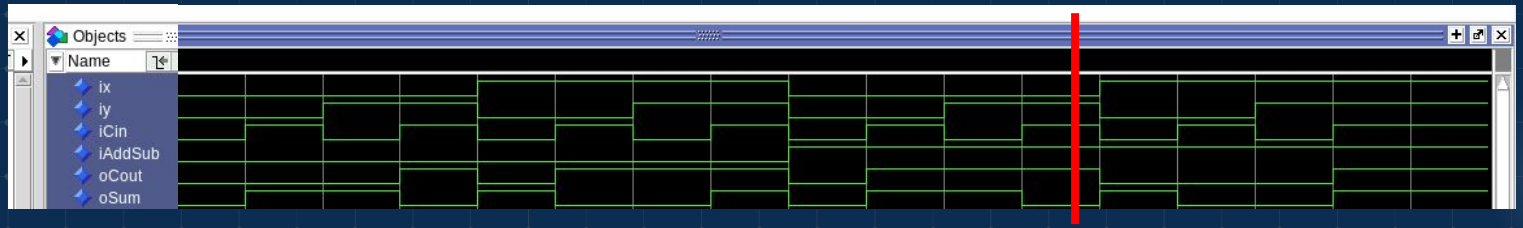
Cin => 1'b1

Add/Sub => 1b'1

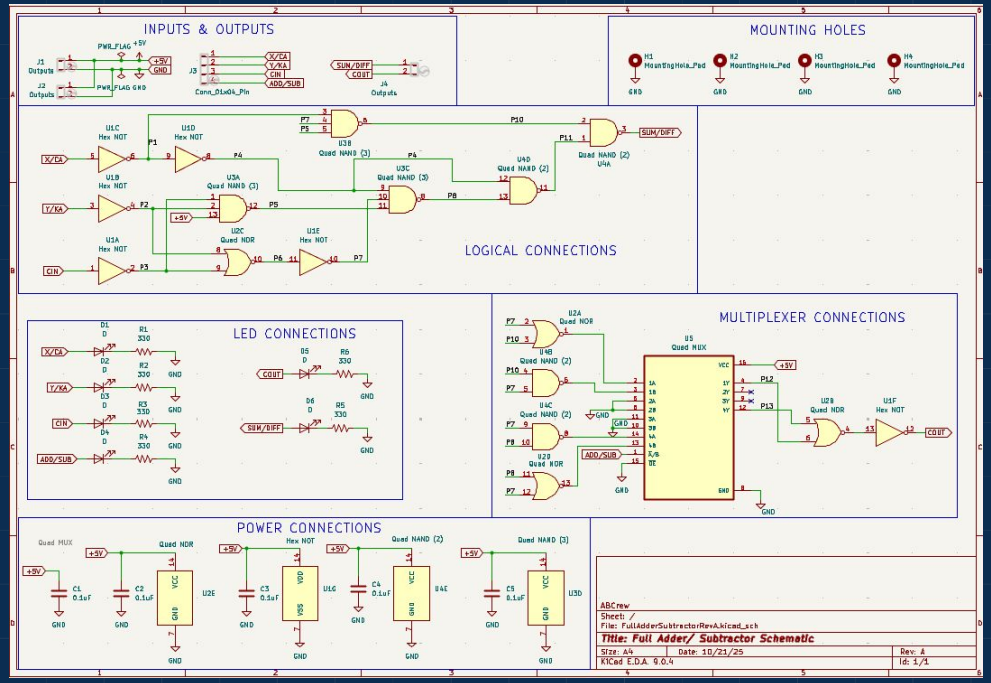
Outputs:

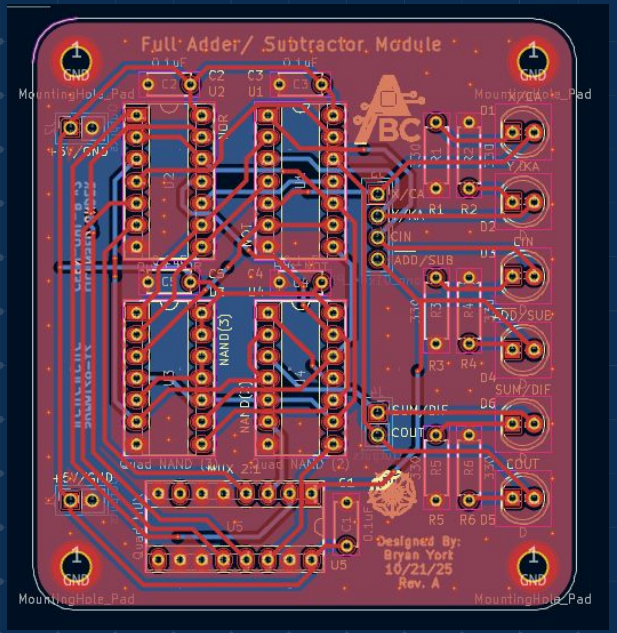
Cout => 1'b1

Sum => 1'b0

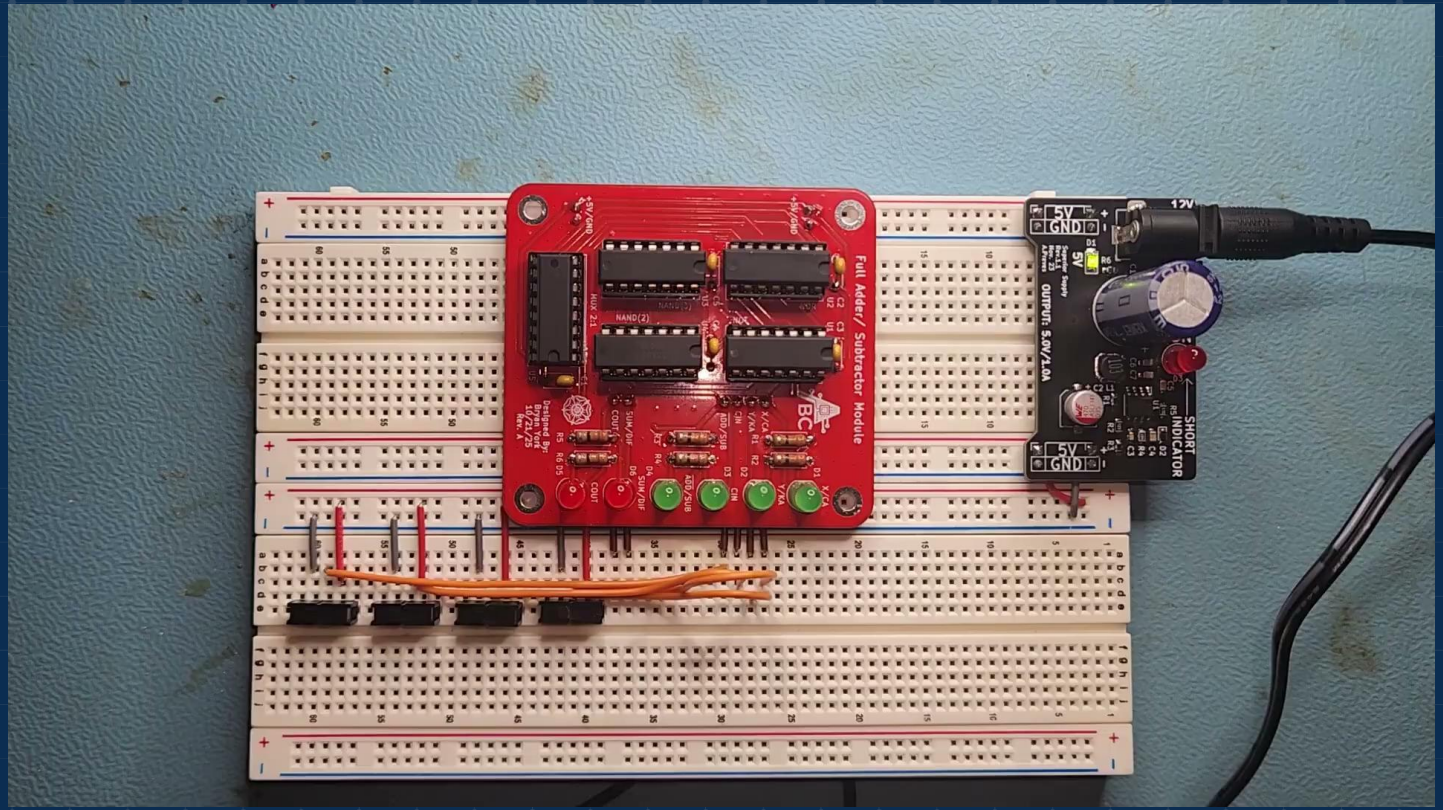


Add/Sub Module: PCB Design



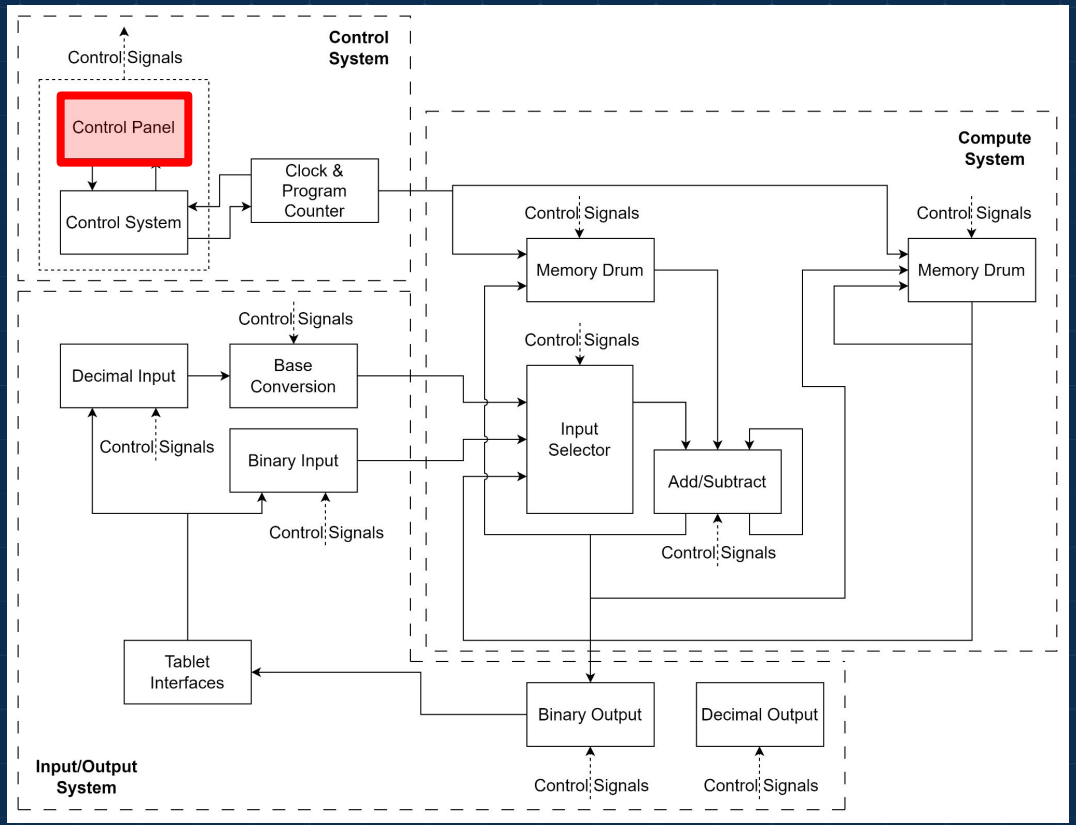


Add/Sub Module: PCB Verification

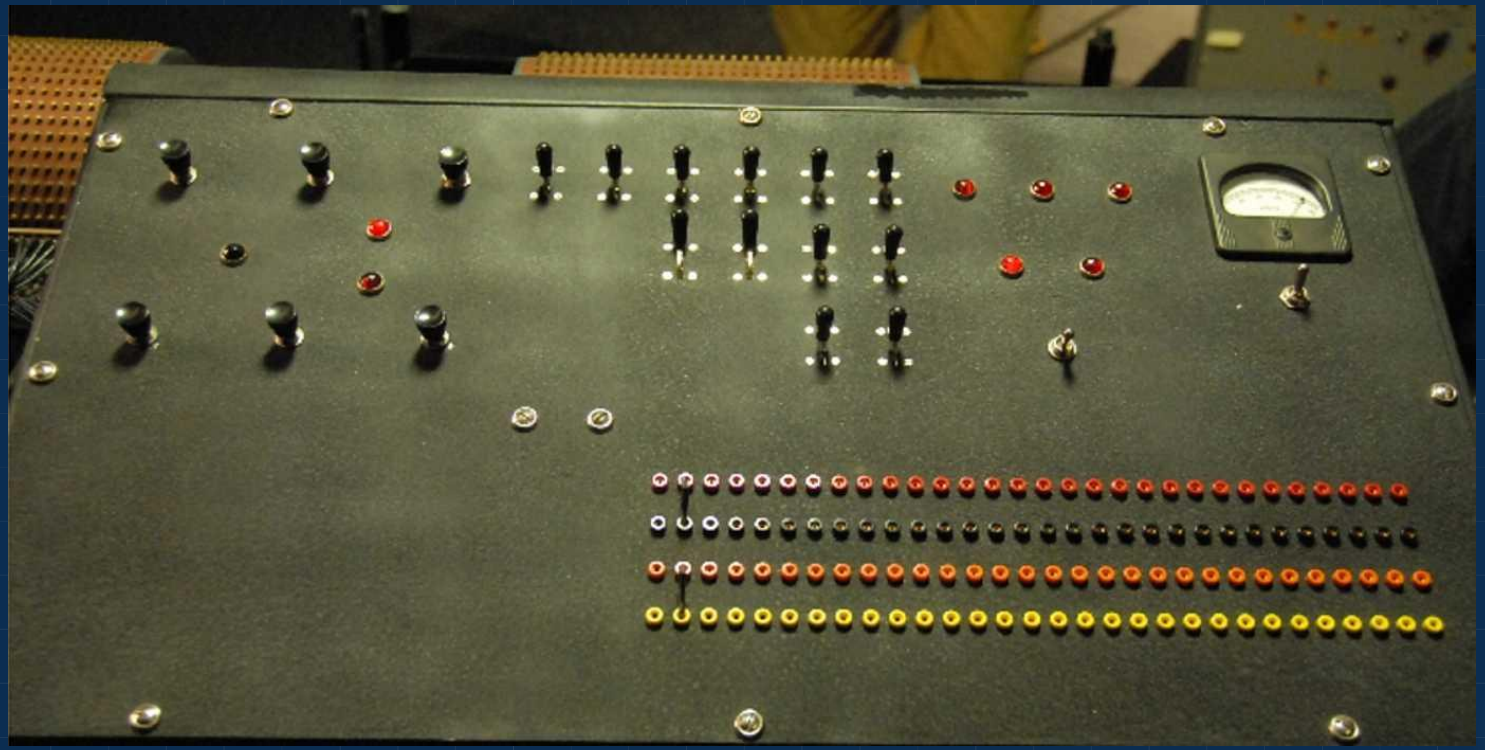


Overview of Modules

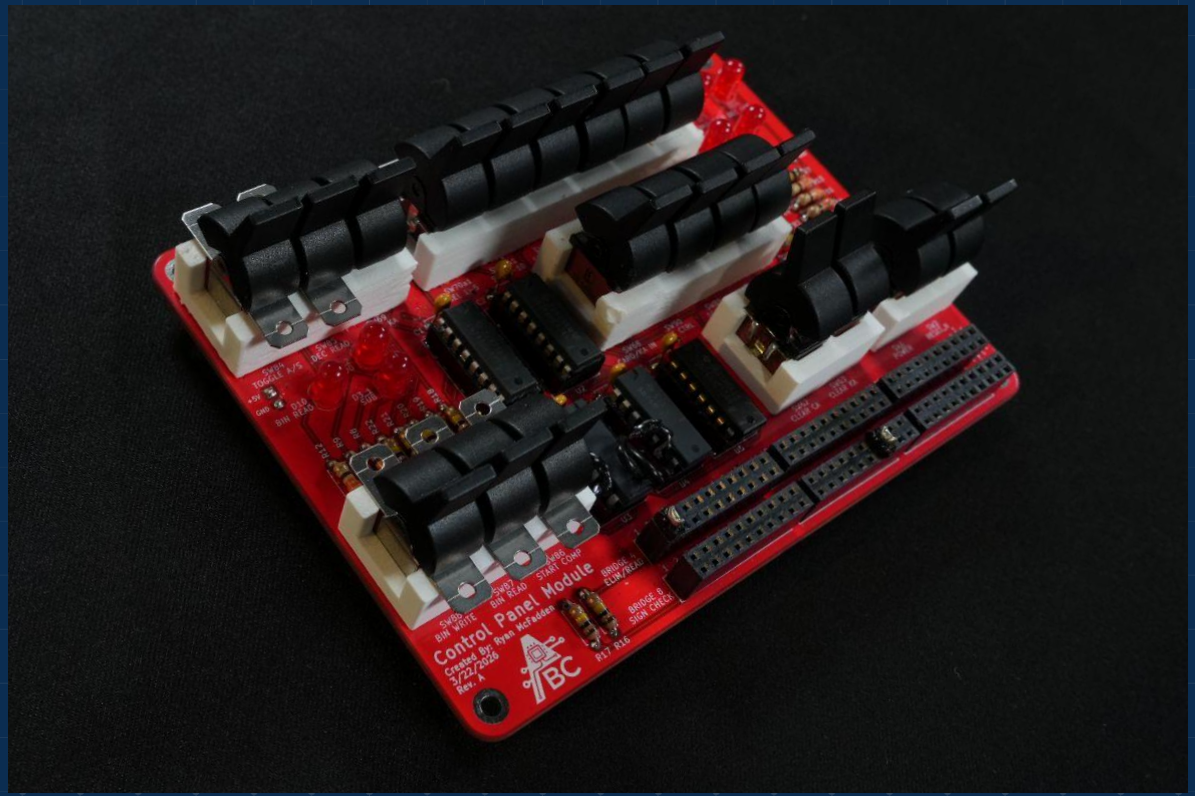
Control Panel



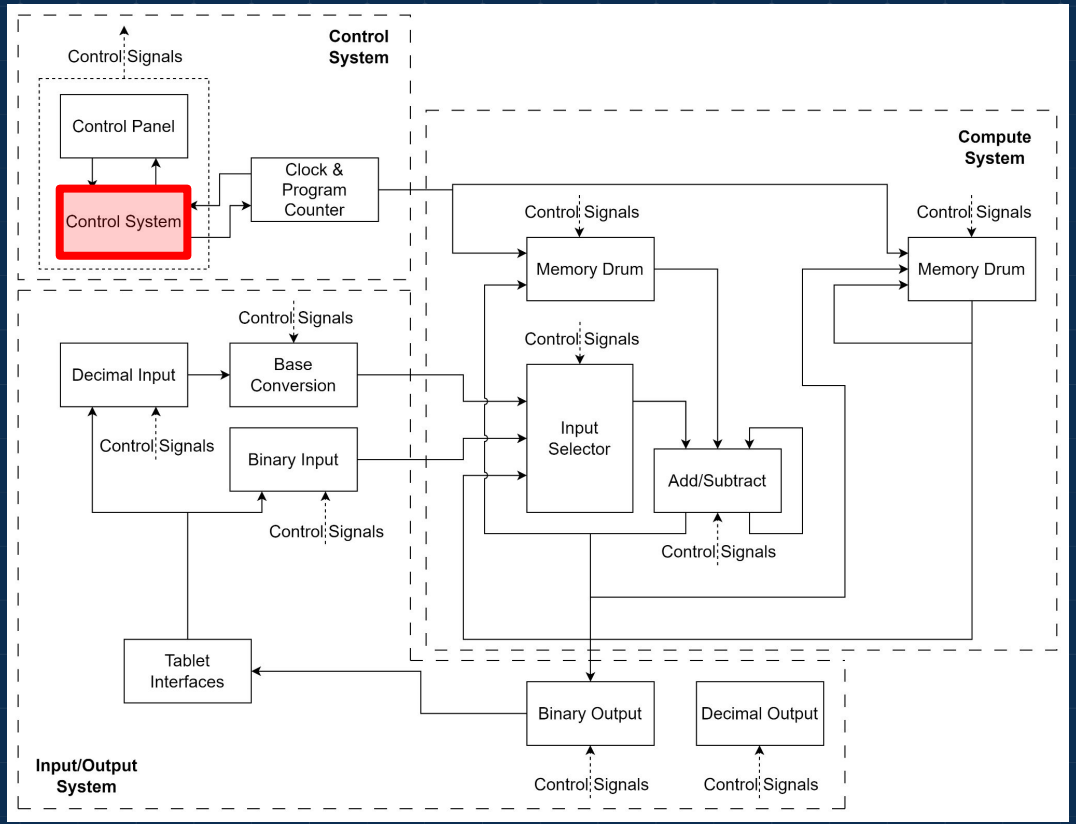
Control Panel



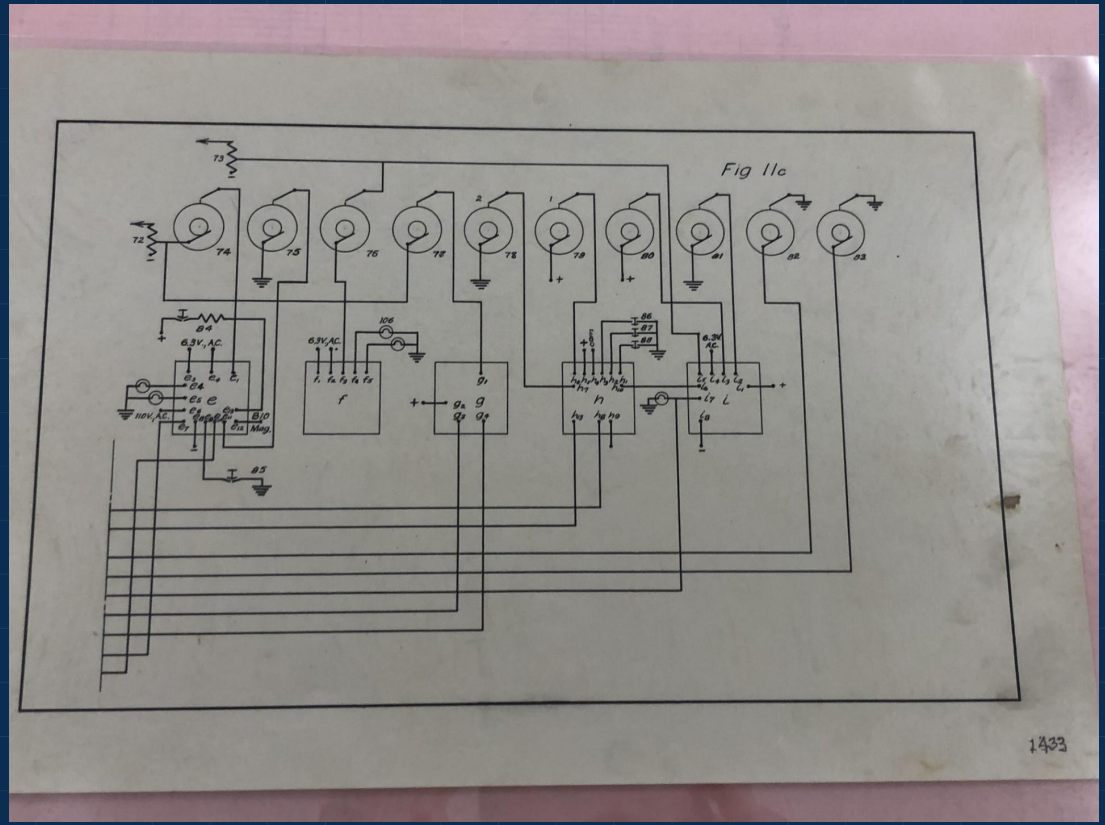
Control Panel



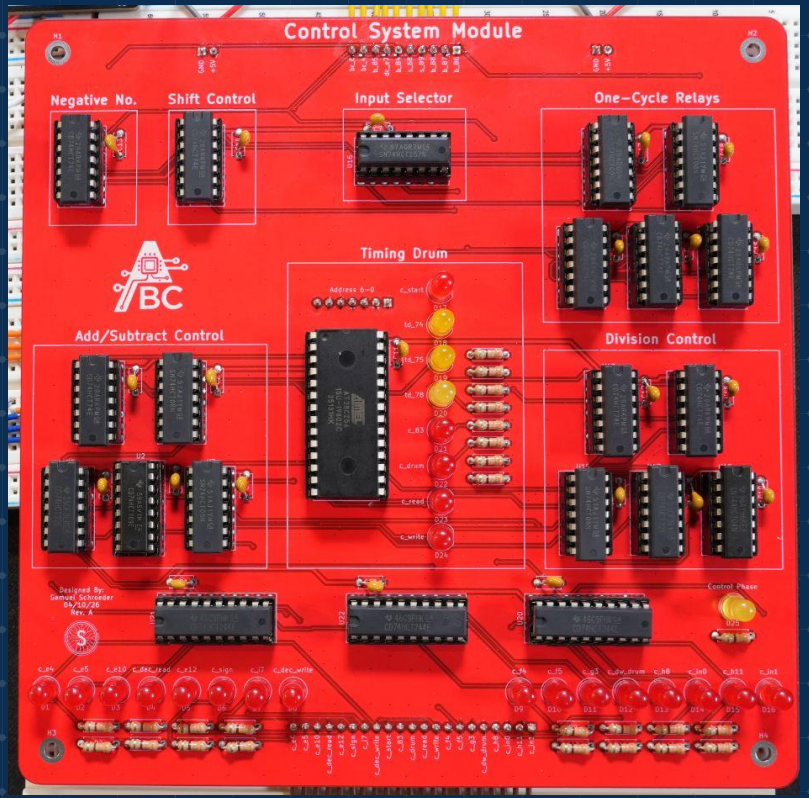
Control System



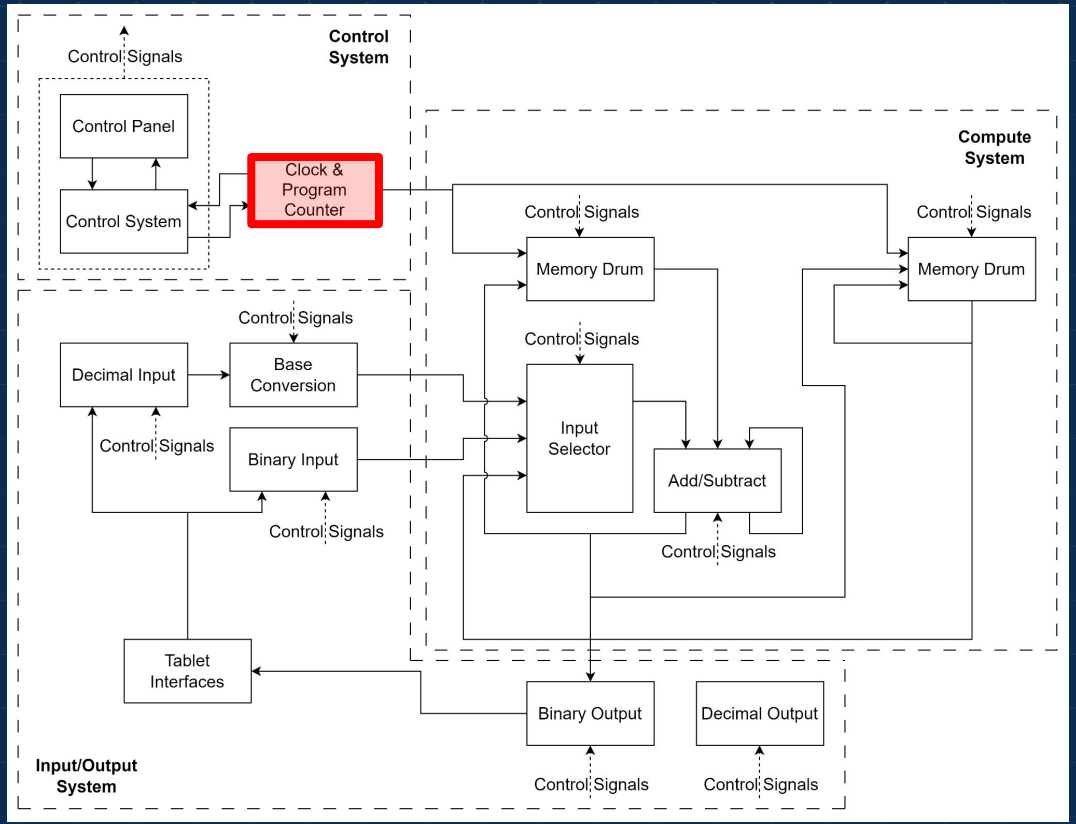
Control System



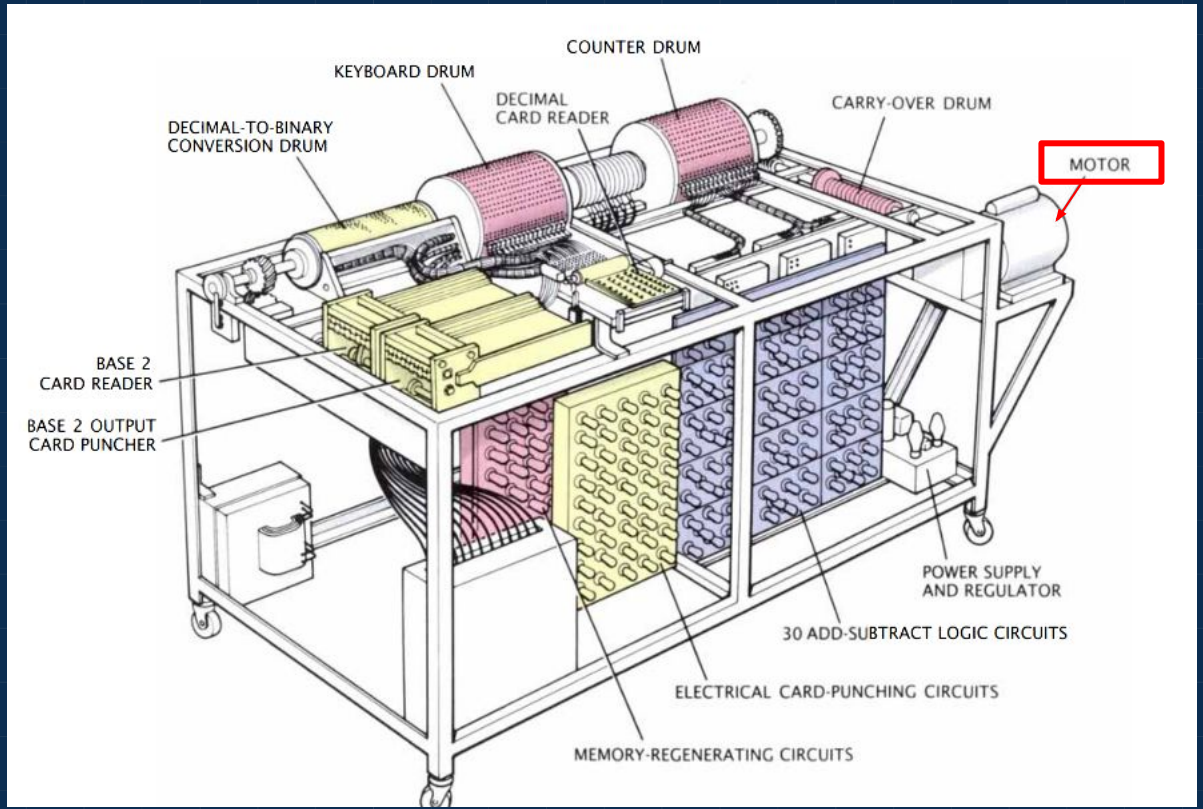
Control System



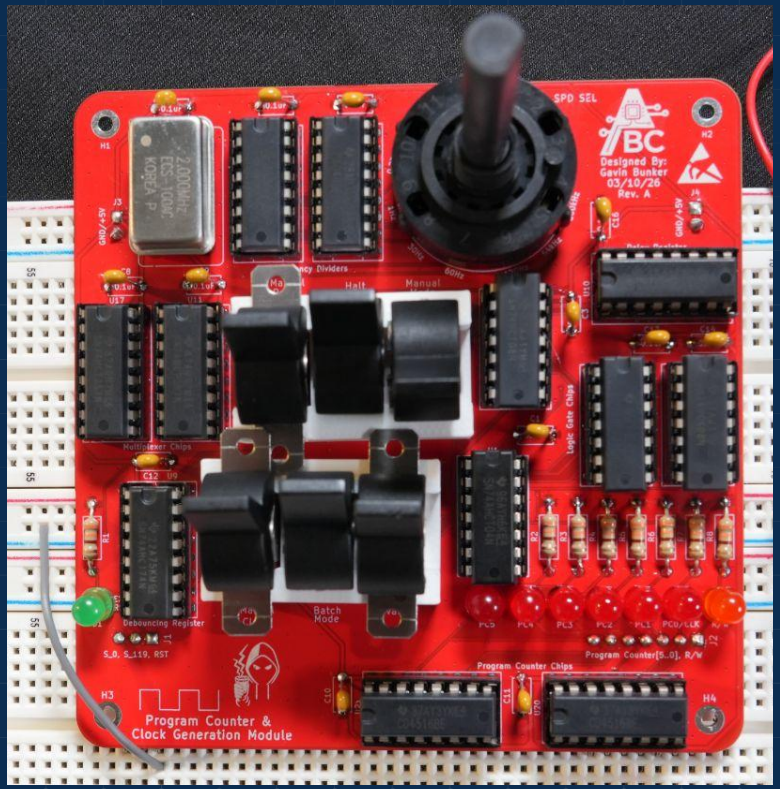
Clock


Control System
Control Signals
Control Panel
Control System
Clock & Program Counter
Compute System
Control Signals
Memory Drum
Control Signals
Memory Drum
Control Signals
Decimal Input
Base Conversion
Control Signals
Input Selector
Control Signals
Binary Input
Add/Subtract
Control Signals
Control Signals
Tablet Interfaces
Binary Output
Decimal Output
Input/Output System
Control Signals
Control Signals

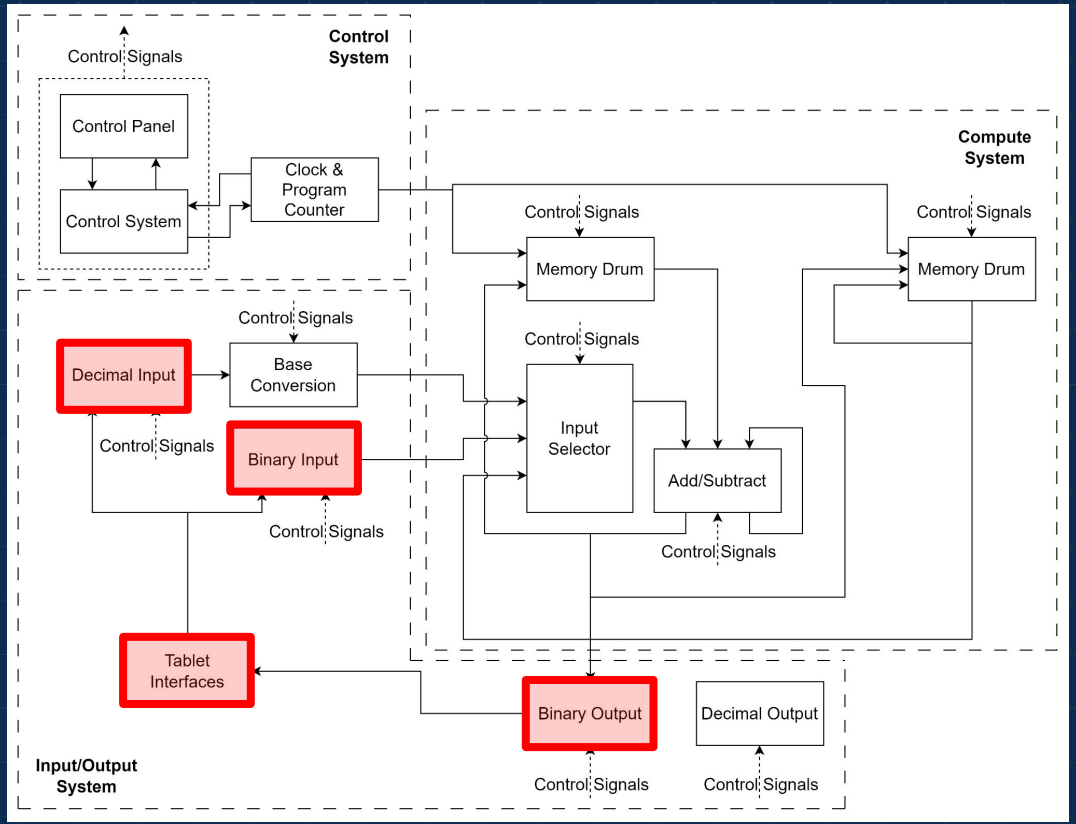
Clock


COUNTER DRUM
KEYBOARD DRUM
DECIMAL
CARD READER
CARRY-OVER DRUM
DECIMAL-TO-BINARY
CONVERSION DRUM
MOTOR
BASE 2
CARD READER
BASE 2 OUTPUT
CARD PUNCHER
POWER SUPPLY
AND REGULATOR
30 ADD-SUBTRACT LOGIC CIRCUITS
ELECTRICAL CARD-PUNCHING CIRCUITS
MEMORY-REGENERATING CIRCUITS

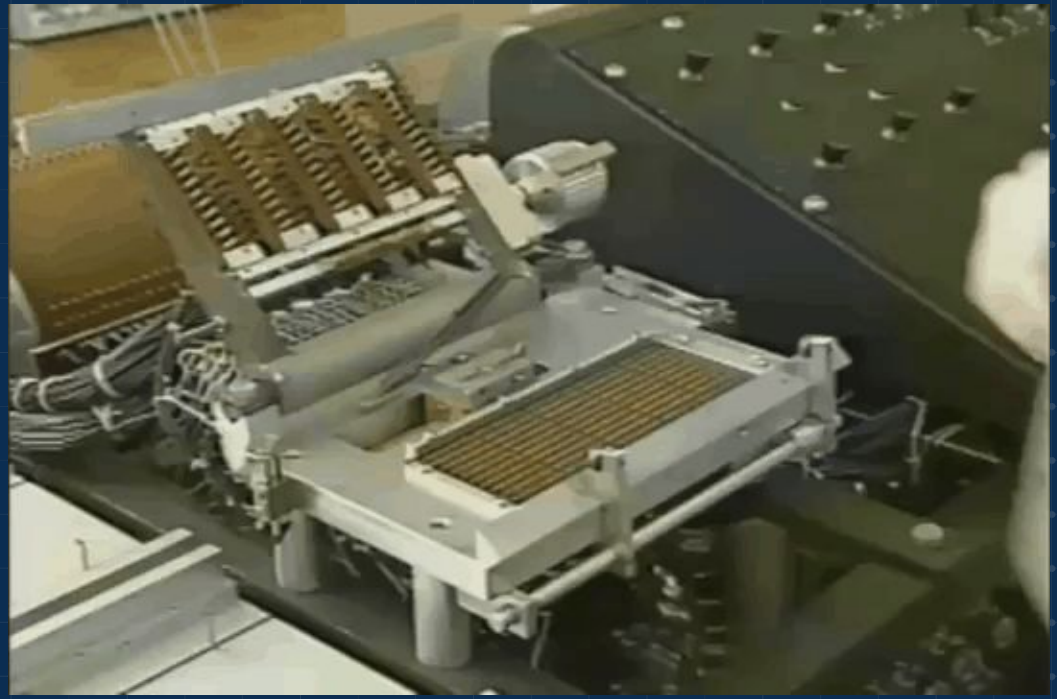
Clock


SPD SEL
Designed By:
Gavin Bunker
03/10/26
Rev. A
Halt
Manual
Batch
Mode
Debouncing Register
Program Counter Chips
Program Counter[5..0], R/W
Program Counter &
Clock Generation Module

Card Reader/Writer

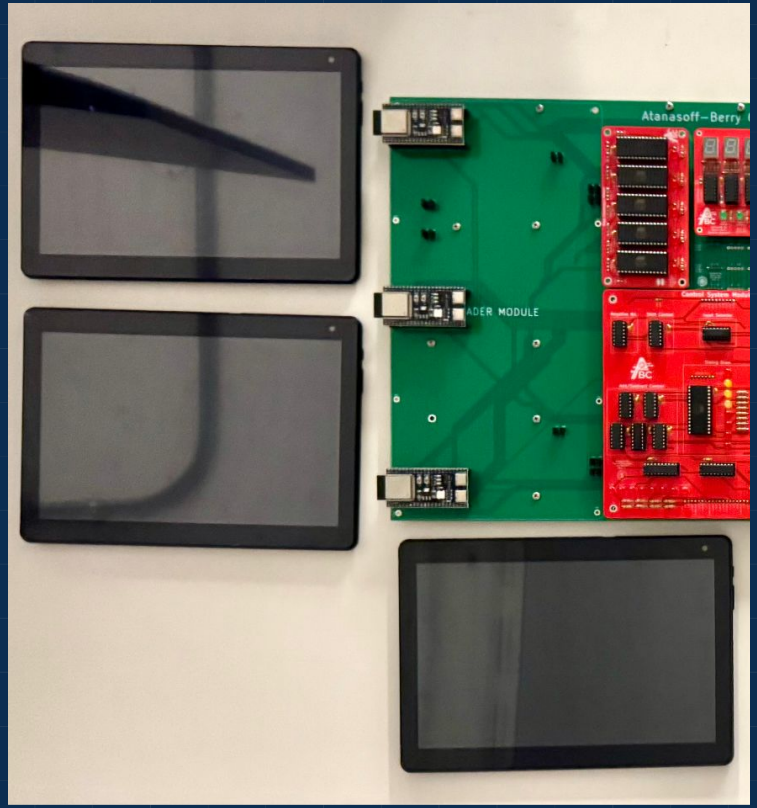


Card Reader/Writer

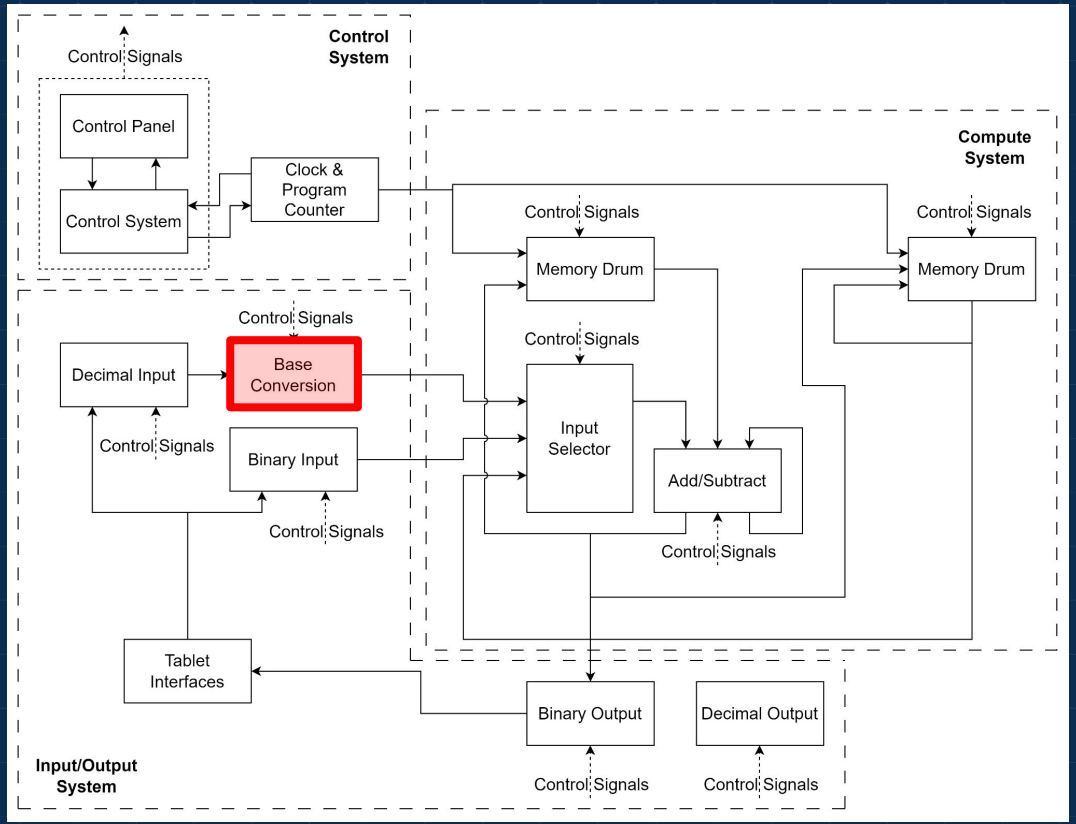


https://jva.cs.iastate.edu/reconstructionoperation.php

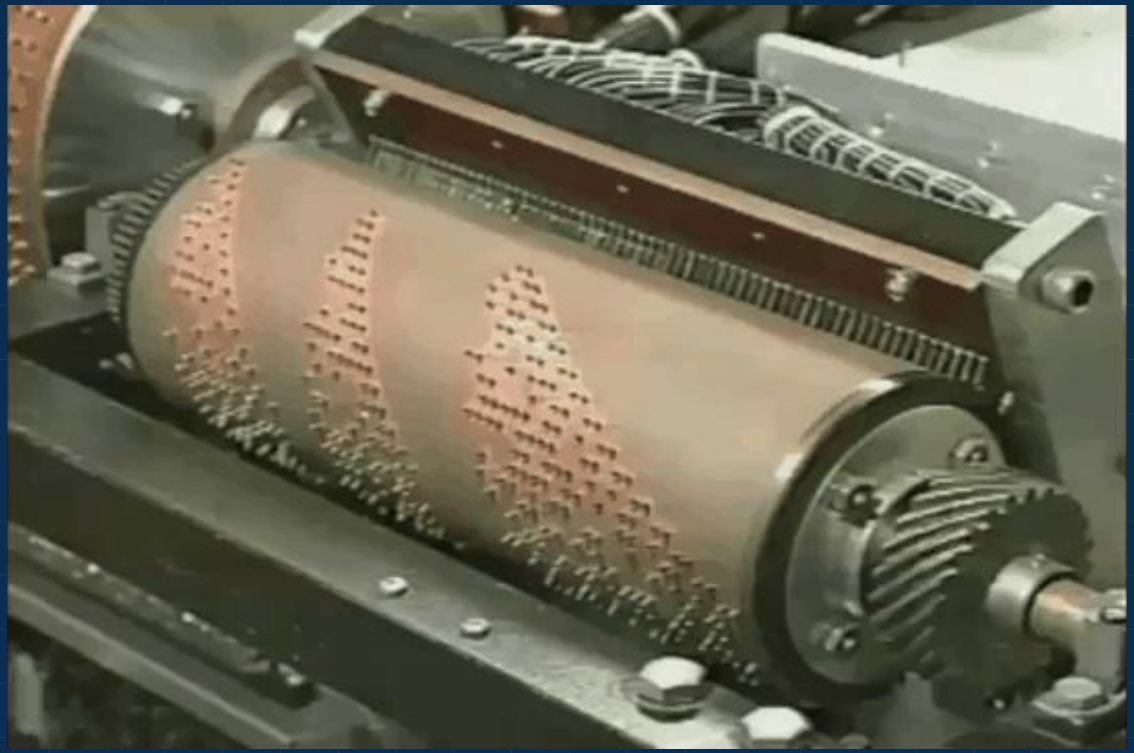
Card Reader/Writer



Base Conversion

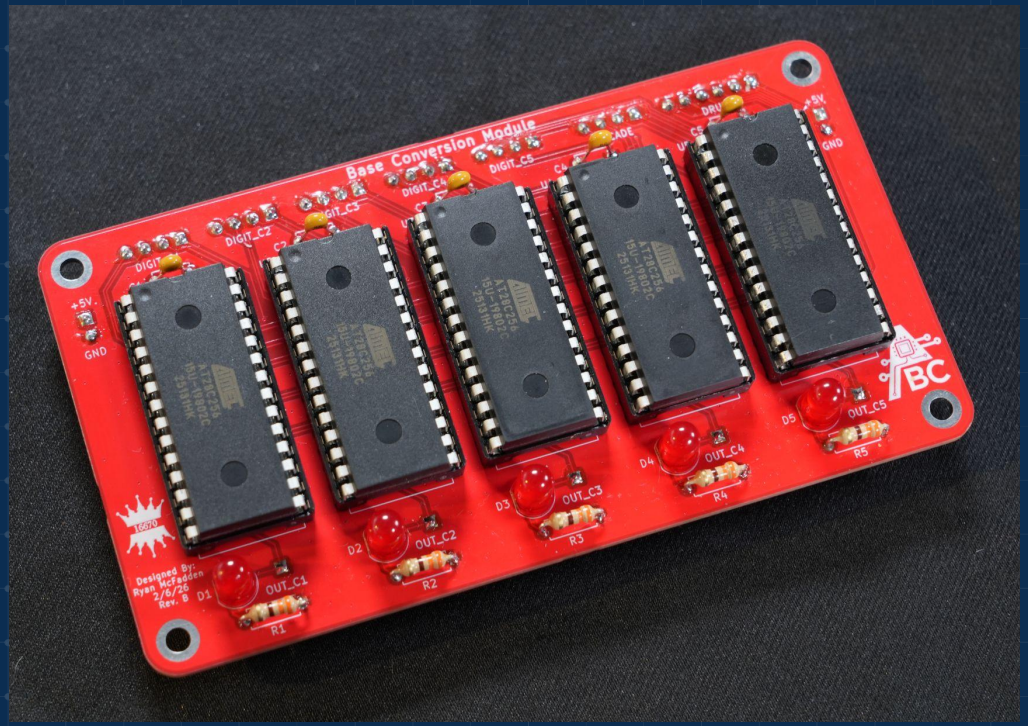


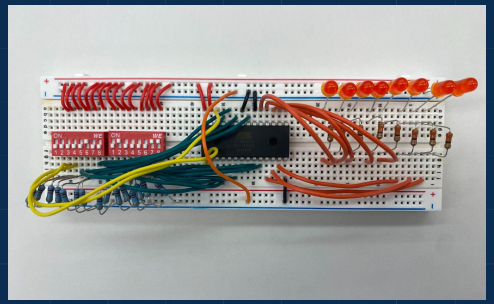
Base Conversion

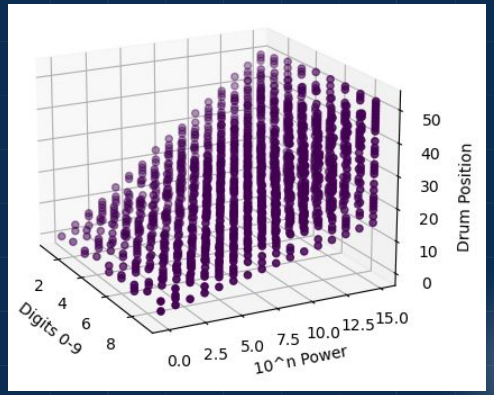


https://jva.cs.iastate.edu/reconstructionoperation.php

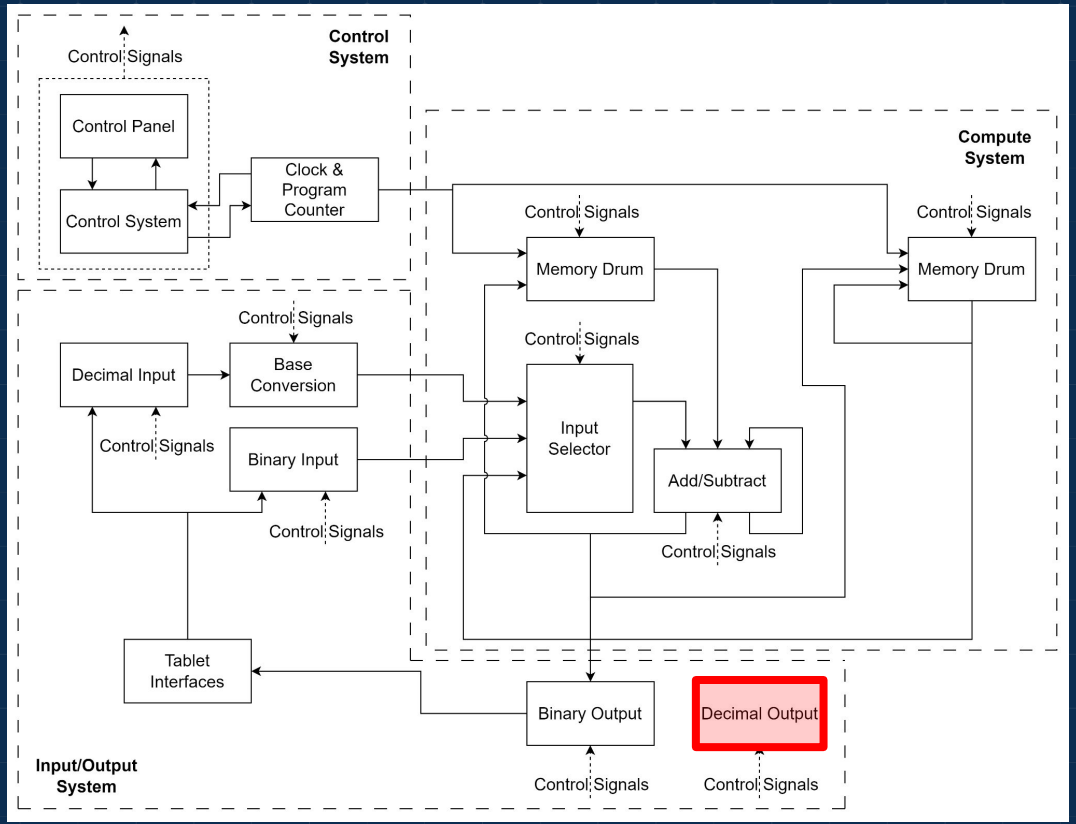
Base Conversion







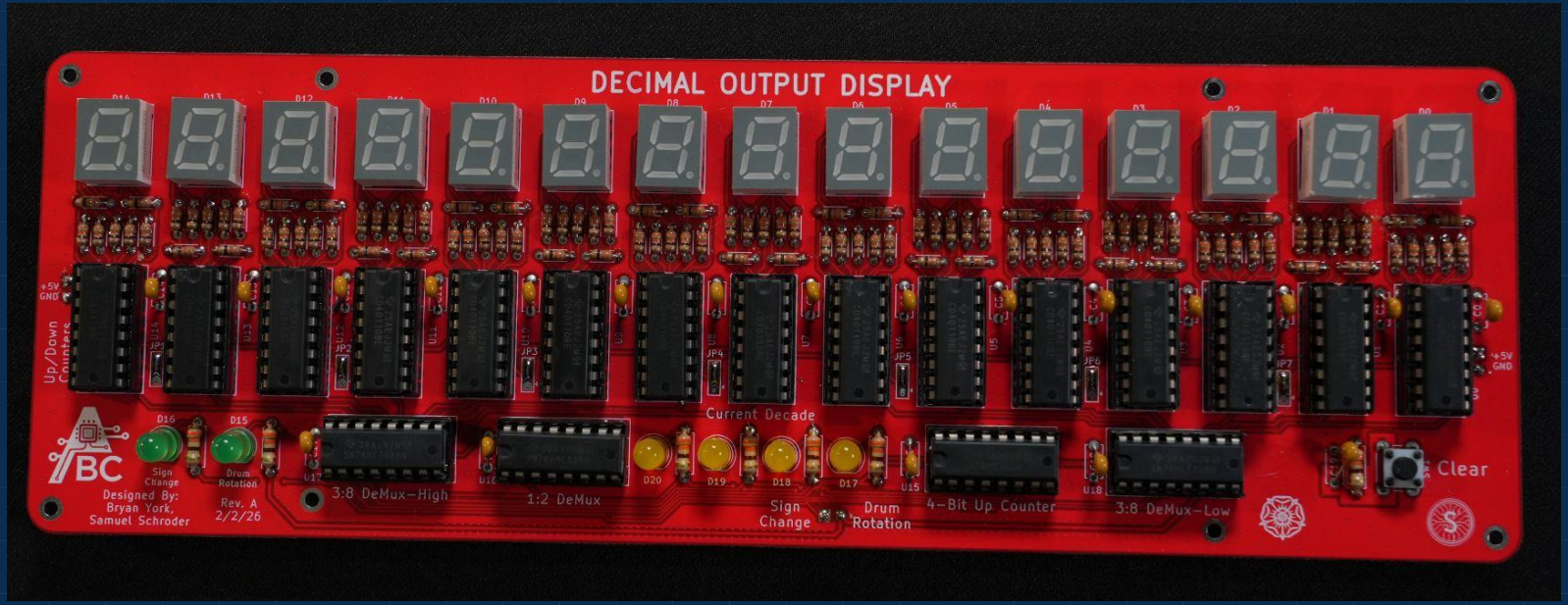
Decimal Output



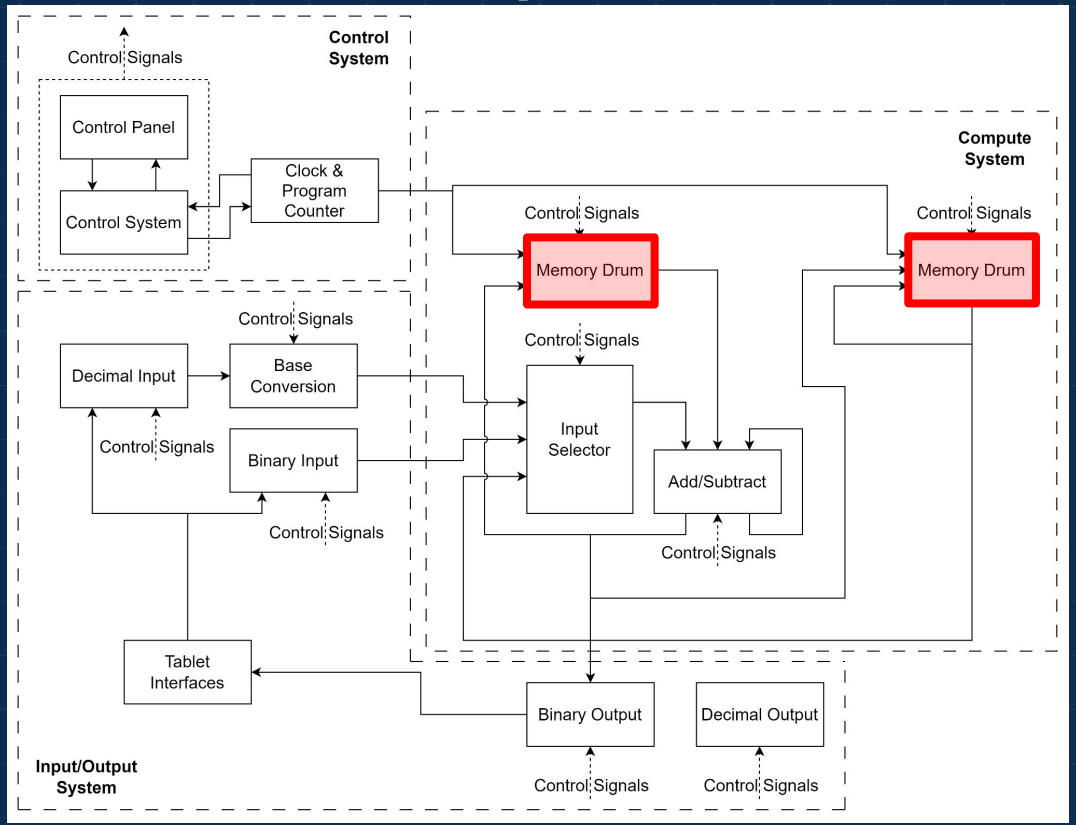
Decimal Output



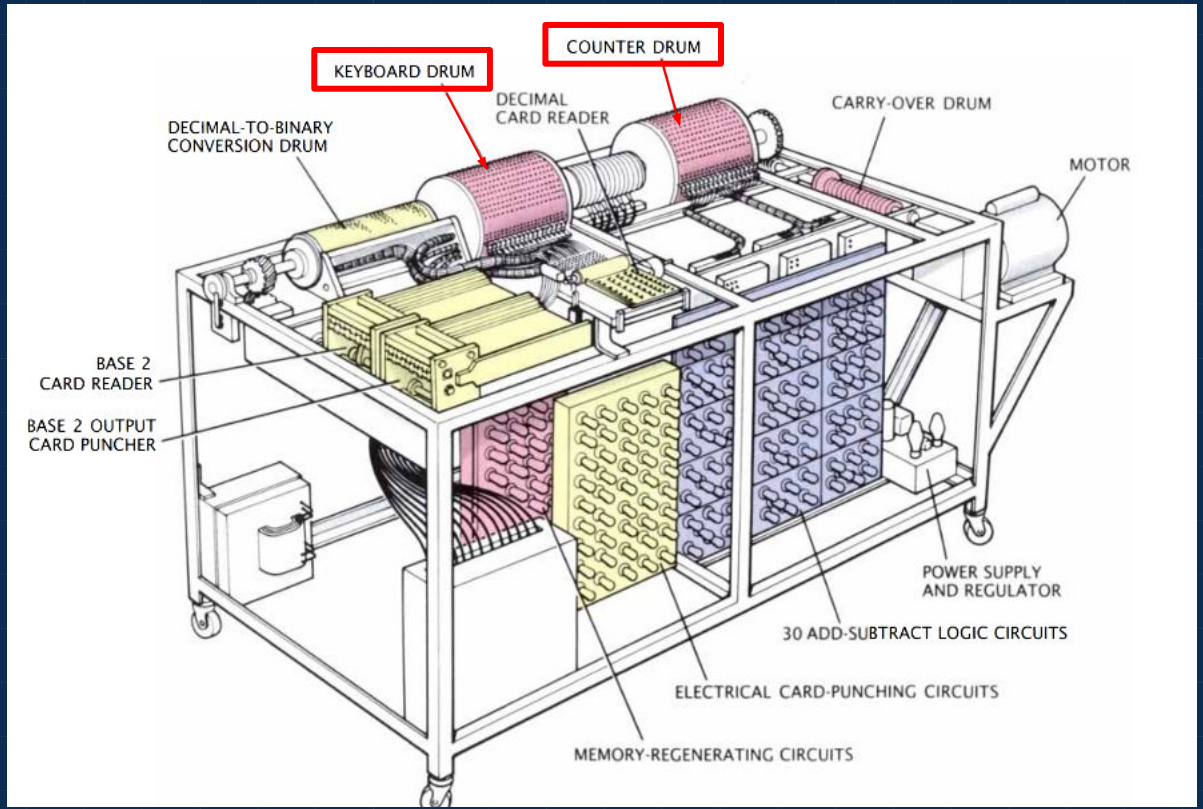
Decimal Output



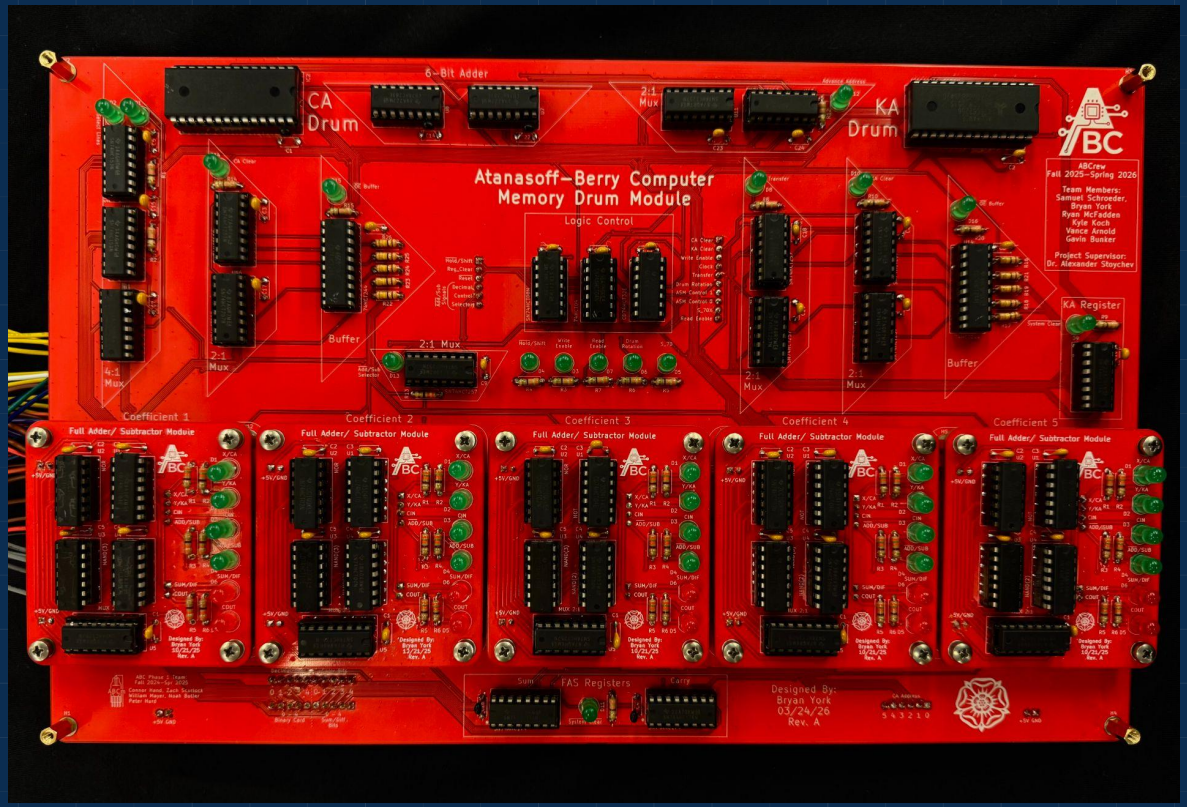
Memory Drums



Memory Drums

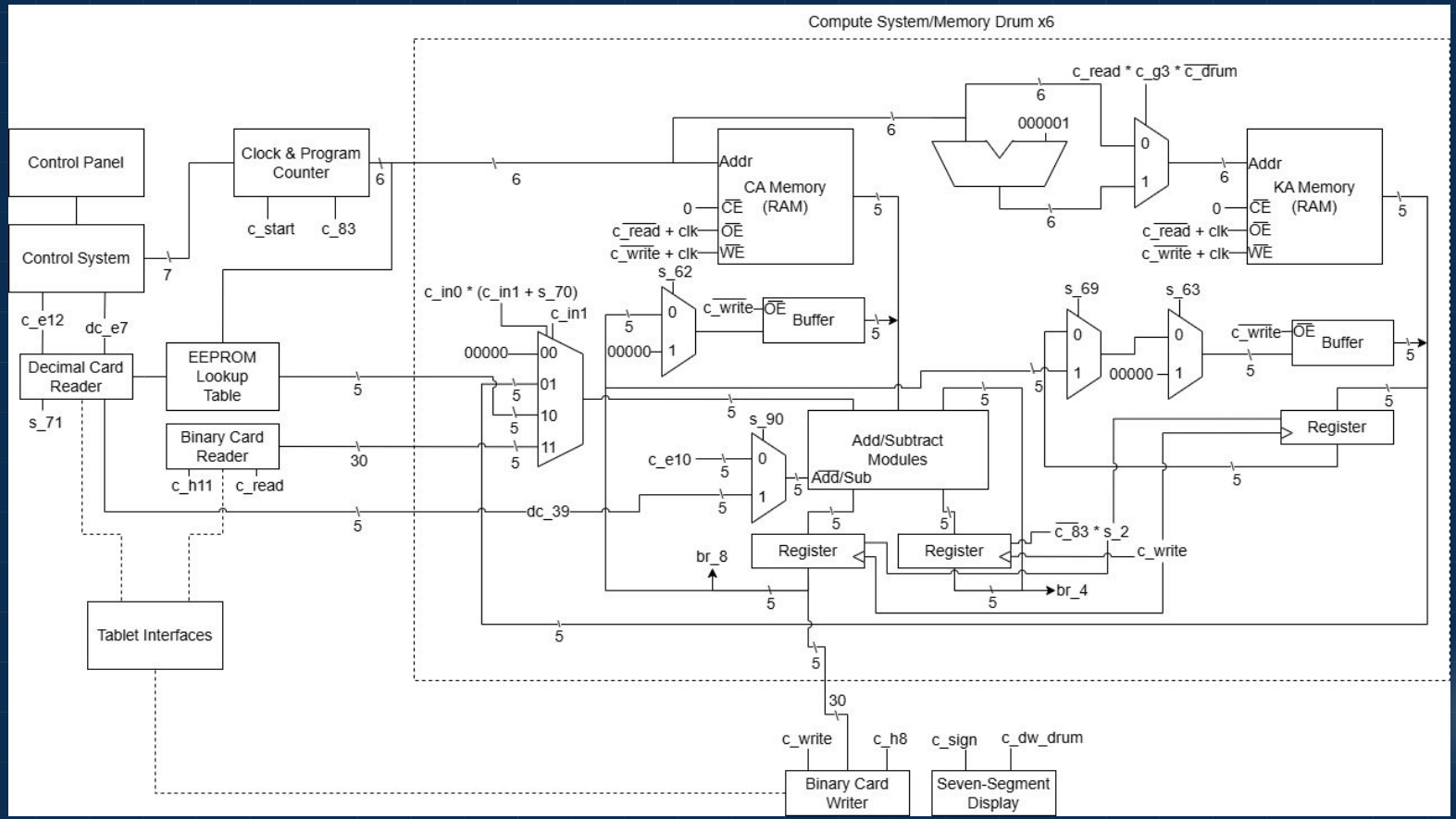


Memory Drums

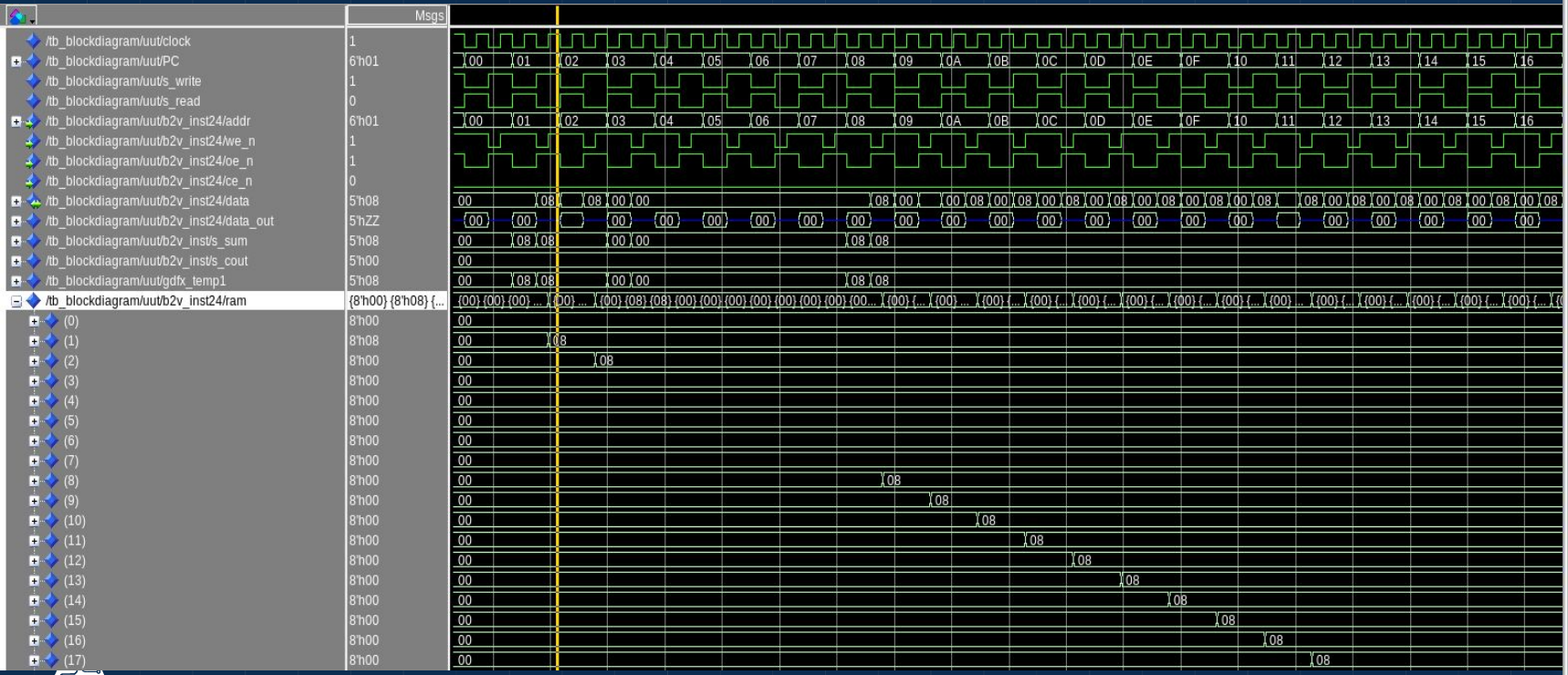


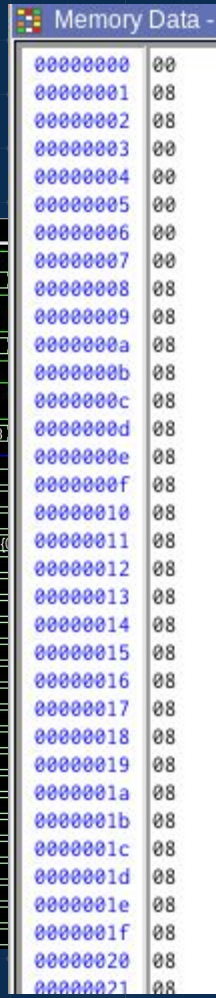
Integration

VHDL Simulation

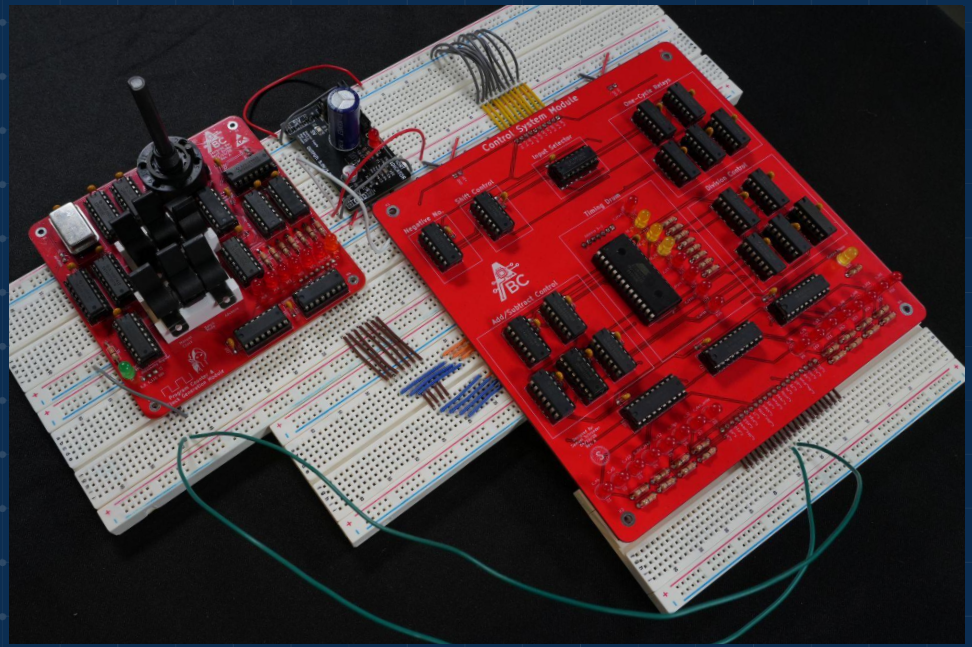


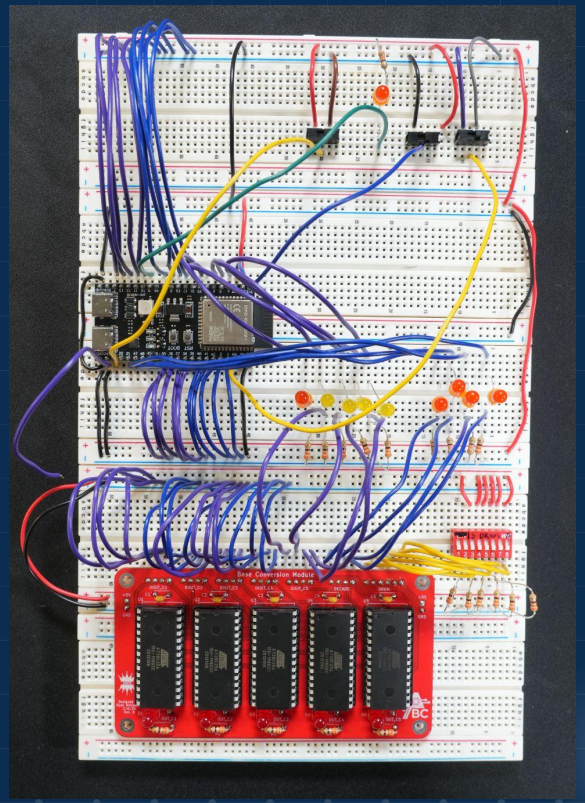
VHDL Simulation



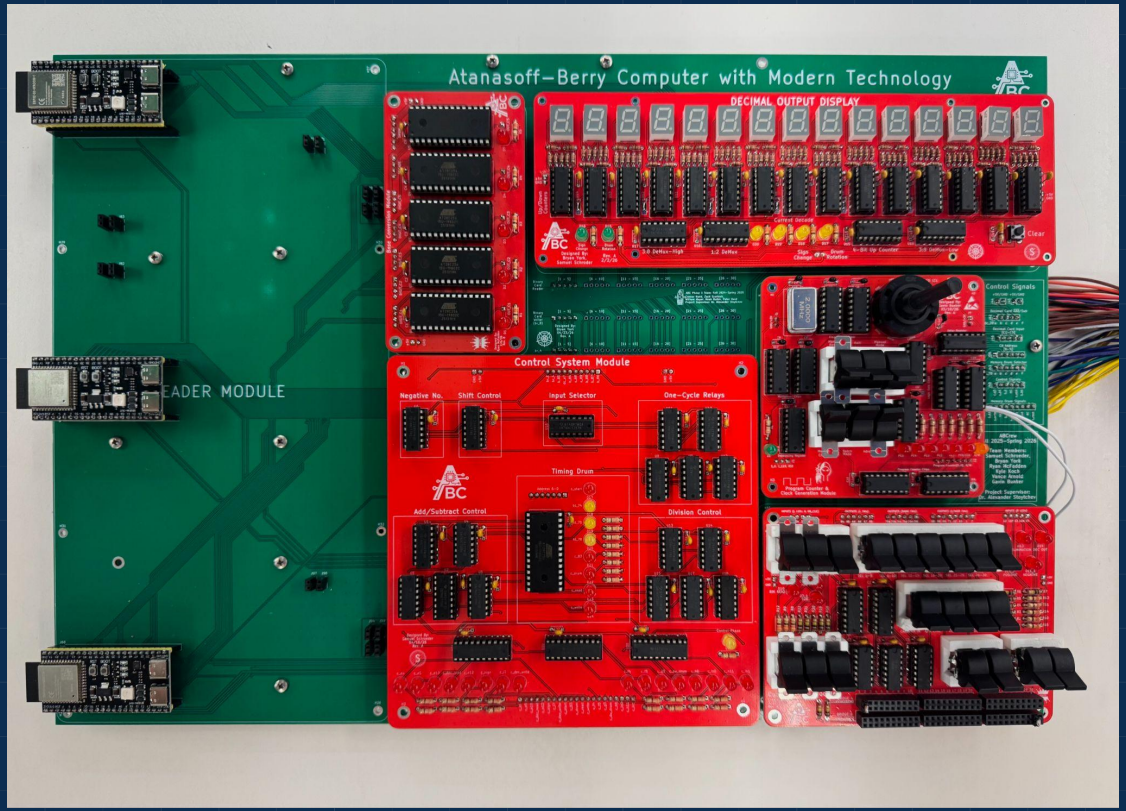


Integration Testing

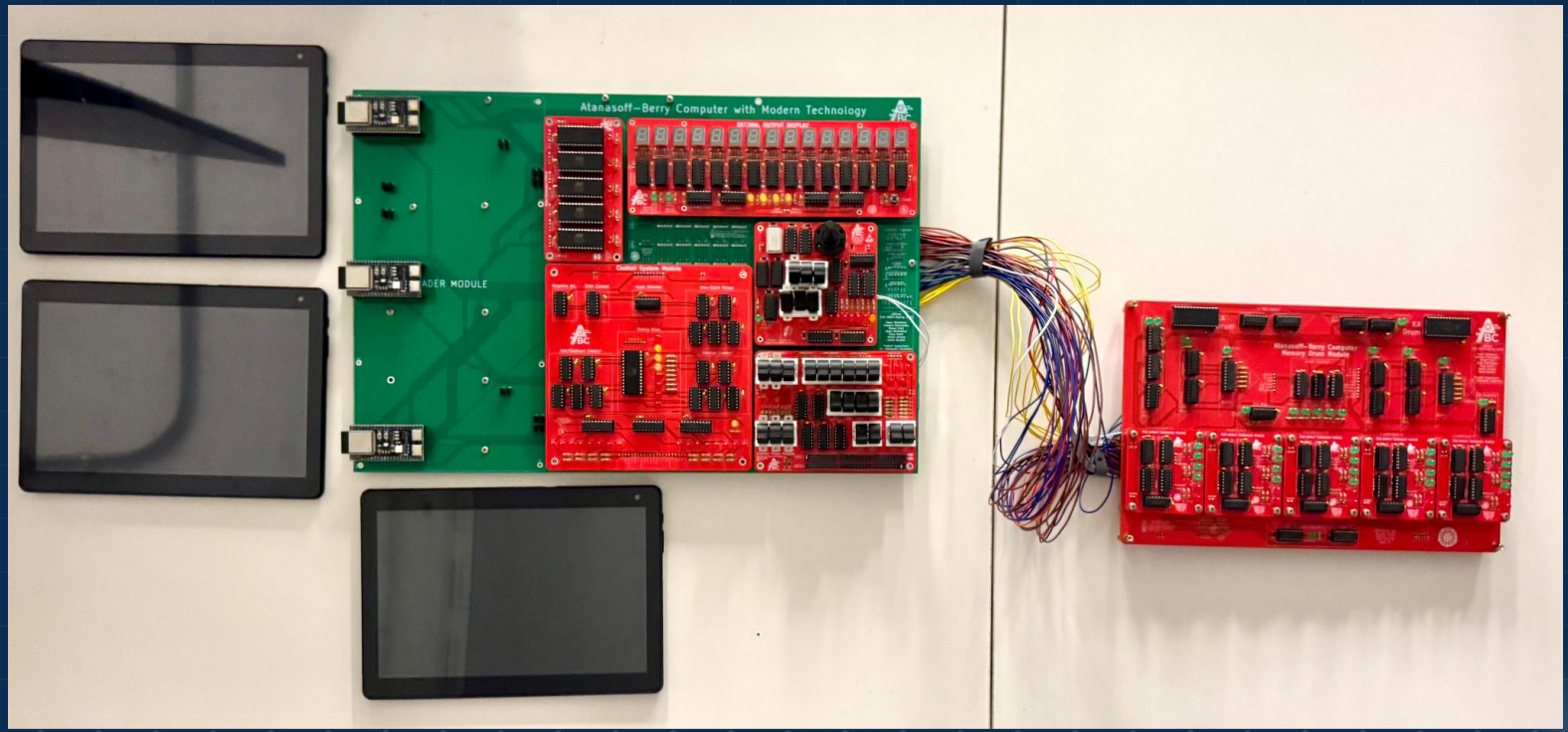




System Assembly & Testing



System Assembly & Testing



Equation to Solve

Equation

x + 2y = 8

2x + 6y = 12

Scale by 100

Decimal Input Cards

100x + 200y = 800

200x + 600y = 1,200

Solve for Y

KA:

CA:

Step 0: Input Cards

KA: 100x + 200y = 800

CA: 200x + 600y = 1,200

Step 1: Subtract KA from CA

KA: 100x + 200y = 800

CA: 100x + 400y = 400

Step 2: Subtract KA from CA

CA: 0x + 200y = -400

Step 3: Output CA to Binary Card



Equation to Solve

Solve for X

KA:

CA:

Step 0: Input Cards

KA: 0x + 200y = -400

CA: 200x + 600y = 1,200

Step 1: Subtract KA from CA

KA: 0x + 200y = -400

CA: 200x + 400y = 1,600

Solve for X

KA: 0x + 200y = -400

CA: 200x + 400y = 1,600

Step 2: Subtract KA from CA

KA: 0x + 200y = -400

CA: 200x + 200y = 2,000

Step 3: Subtract KA from CA

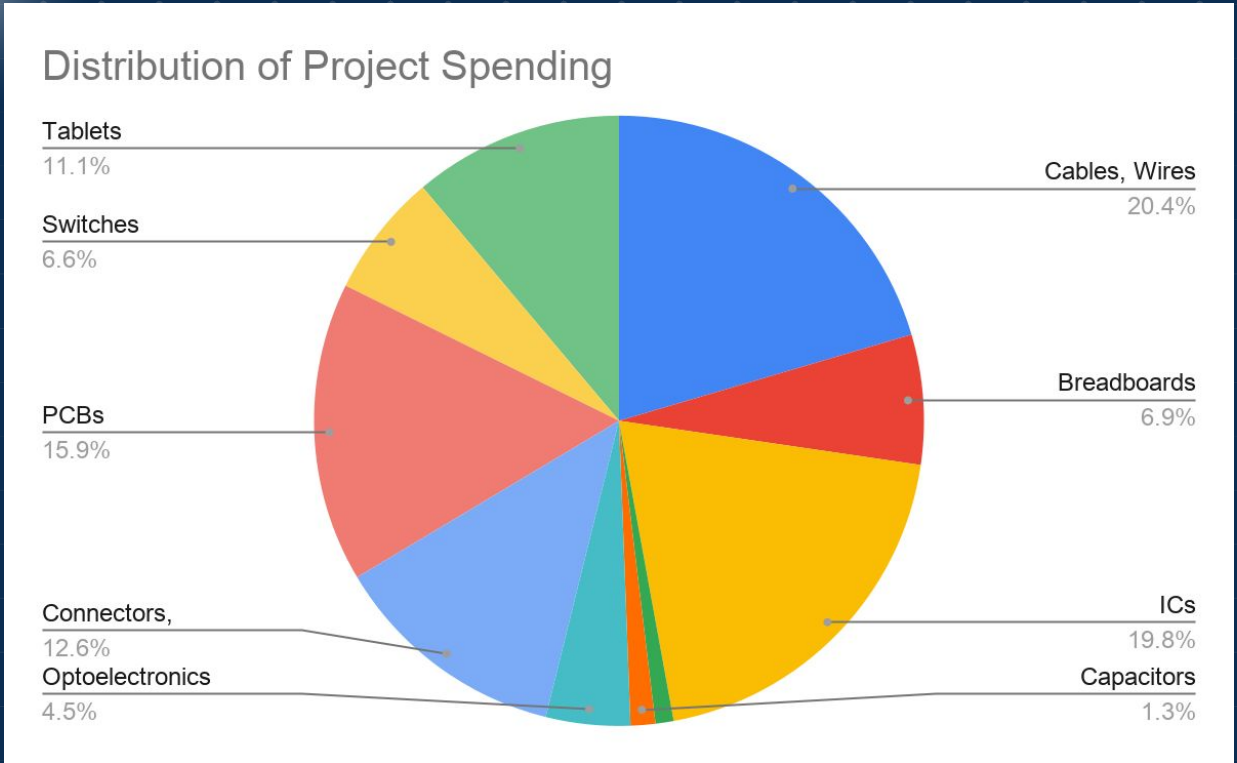
CA: 200x + 0y = 2,400

Step 4: Output CA to Binary Card

Logistics & Conclusion

Budget Overview

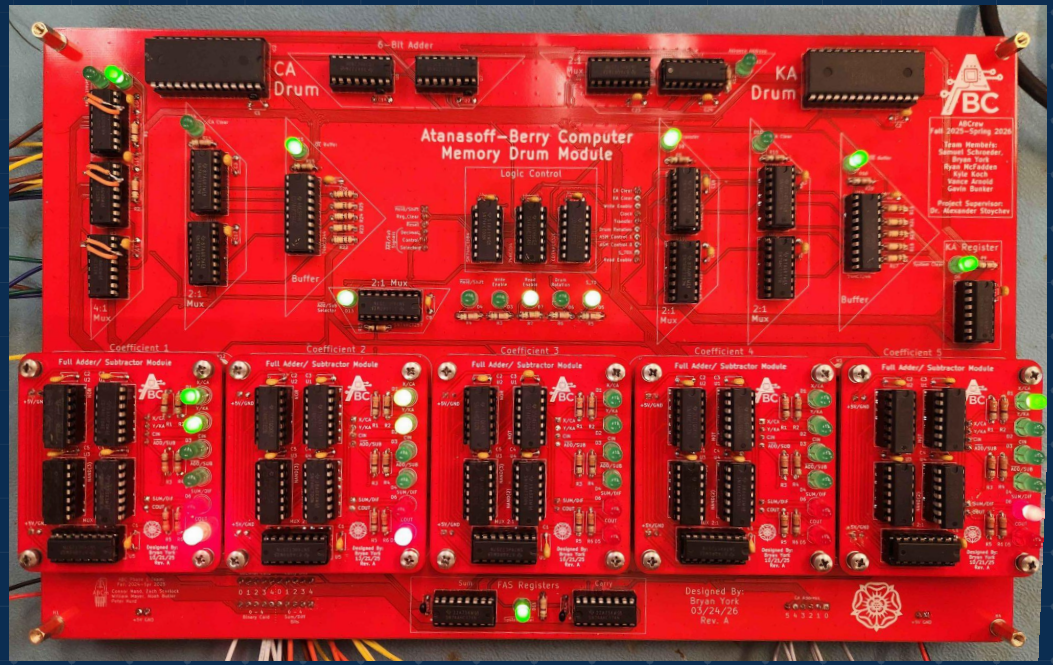
To-Date Spending:
~$1500



Note: Does not include extra costs such as shipping, fees, etc.

Value Provided & Next Steps

- Historical preservation of ABC
- Visualization of the inner workings and connections of a computer
- First Step: Fix defects and make a more polished version
- Second Step: Scale to full system size
- Third Step: Full Automation of Control System



Thank you for your time!

Questions?