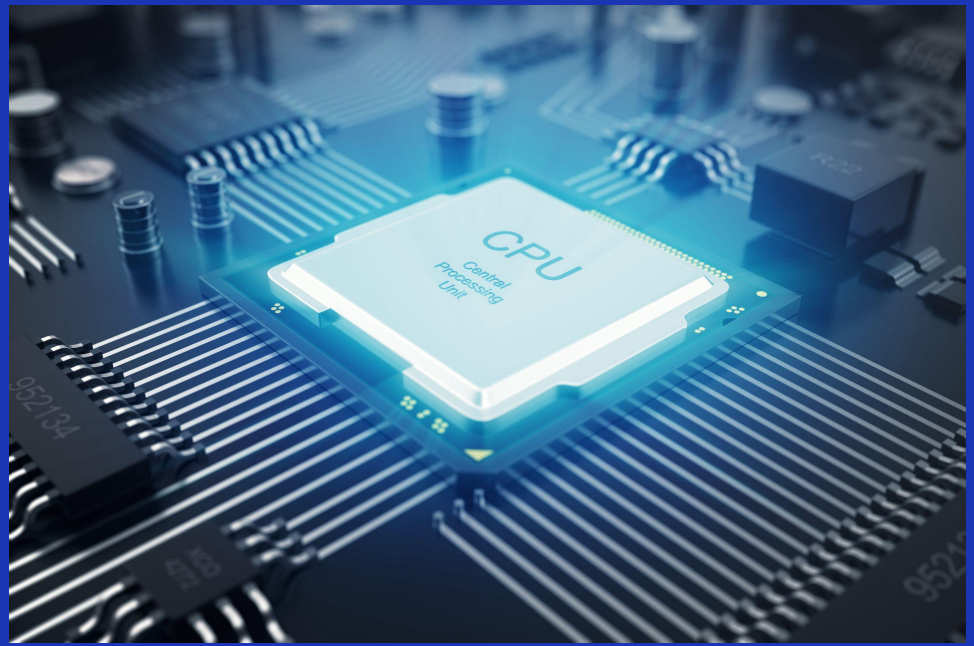
CPU Implementation Project



Crawford James & Quinn Ramsay

The ALU

Input Design:

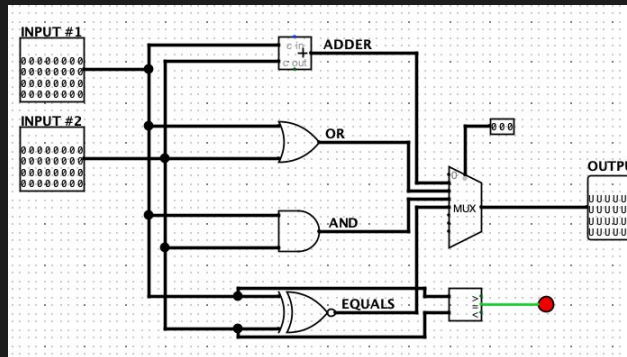
- Created two 32-bit input pins to visualize effectiveness and verify outputs.

Implementation:

- Connected input pins to an arithmetic or gate.
- Linked the gates output to the inputs of a multiplexer for selection of arithmetic.
- Connected the multiplexer to a 32-bit output pin.
- Used a 3-bit selector pin to control the multiplexer.

Equality Operation:

- Used an XNOR gate to send an output to the multiplexer for processing.
- Connected original inputs to a comparator.
- Linked the comparator to an LED output to indicate equality.



The Register File

Input

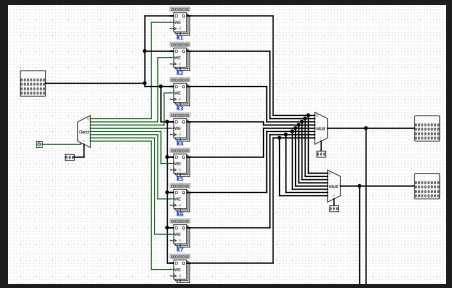
- 8 register banks with 8 bit wide paths for each.
- Decoder used to determine which register to write to based on the address input.
- Two different read addresses to access data simultaneously.

Implementation

- Each register receives data through a sharded bus controlled by the decoder and write enabler.
- Multiplexer to select the output from the correct register.
- Two multiplexers help with simultaneous outputs from the register file.

Output

- Both selected register outputs are connected to 8-bit output pins.
- Ensure correct read/write operation through control signals.



Testing The Program

Testing Setup:

- Assigned separate 32-bit input and output pins for the Register File and ALU to visualize issues.

Independent Testing:

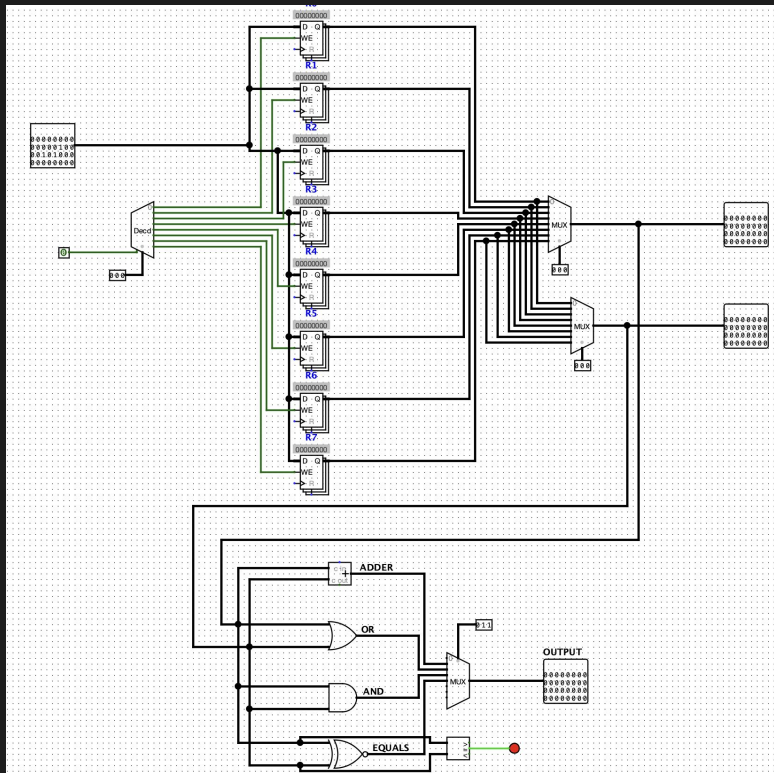
- Built and tested each component separately to confirm functionality.

Integration Testing:

- Combined both components and verified output accuracy.

Verification Process:

- Input random binary numbers and tested different values.
- Ensured outputs remained consistent across all cases.



Problems We Encountered

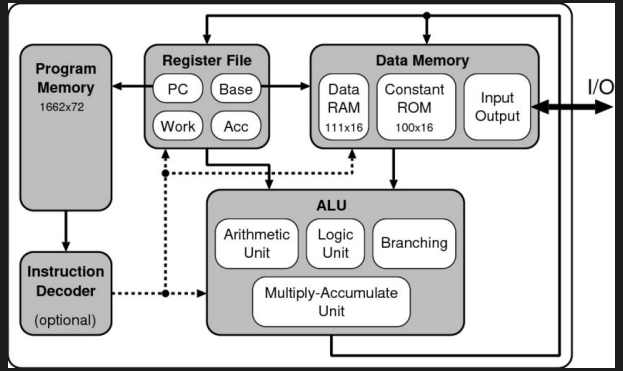
ALU:

- Designing the equality operation was the most difficult part.
- Required trial and error to determine the correct arithmetic function or logic gate.

Register File

- Figuring out how we wanted to use the registers. Specifically addressing the correct register for read and write.
- Making sure that writing only happens at the right time given.

Real World Application



Role of ALU and Register File:

- Essential components of modern CPUs for fast and efficient data processing.

ALU in Real-World CPUs:

- CPUs use multiple ALUs in parallel for complex tasks like graphics processing and floating-point math.

Function of the Register File:

- Stores data for quick access, reducing the need to retrieve it from slower memory.

Performance Optimizations:

- Techniques like pipelining and out-of-order execution enhance speed and efficiency.

Impact on Technology:

- Improves performance in devices ranging from gaming PCs to smartphones and embedded systems.