

But Can It Run Doom? Building a Breadboard Computer

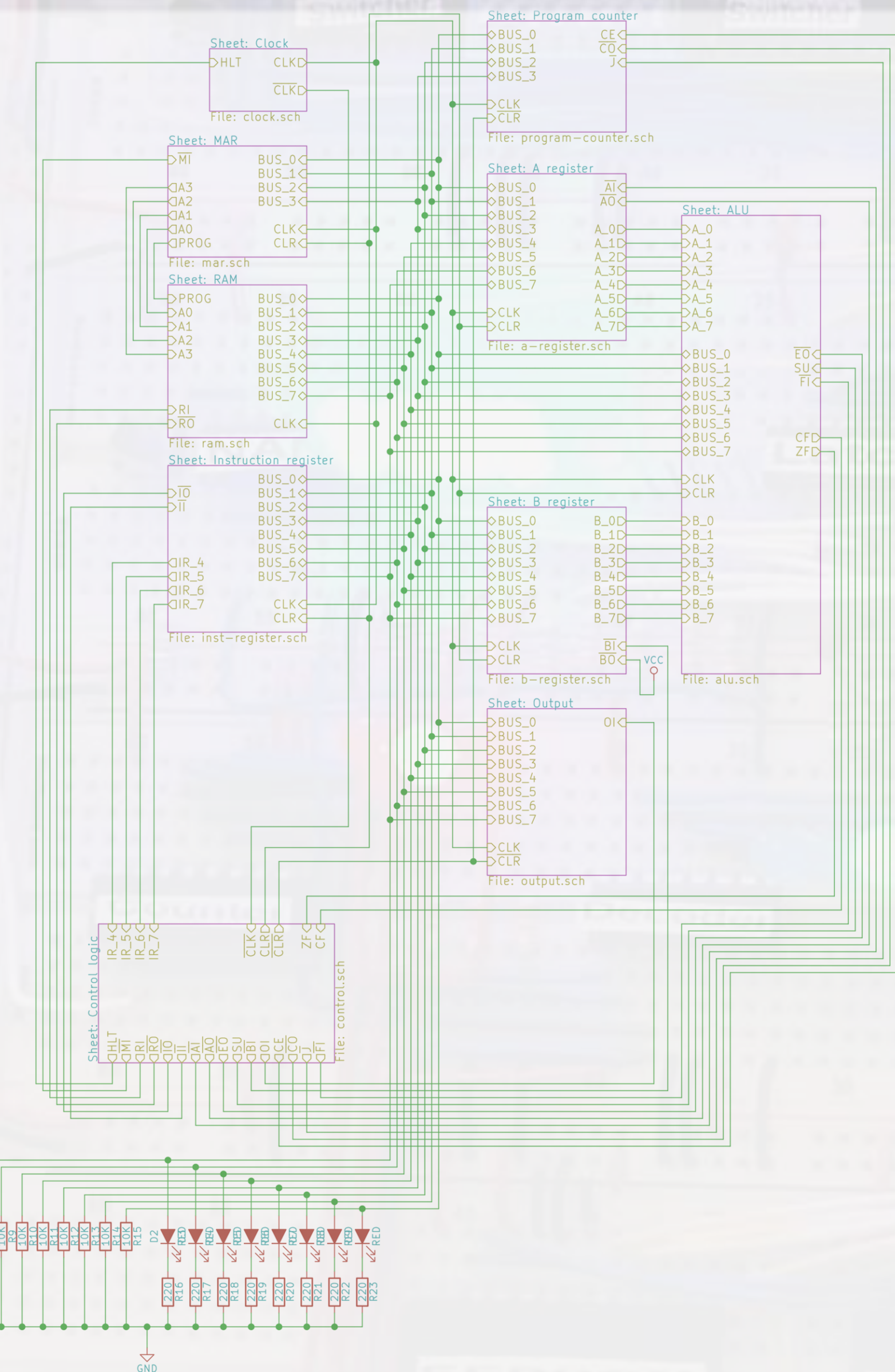
Cooper Sanders

Introduction

I had the opportunity to work with a Volvo engineer, and I was able to shadow his work in the engineering field as I completed an engineering project of my own. I have a basic knowledge of programming, but I've realized recently that I have very little understanding of how processors actually execute commands. In this proof of concept project, I built a fully functional computer from scratch. The computer I built is not remotely comparable to what modern computers can do, but the design process of this project has given me a much deeper understanding of the inner workings of computer processors.

Process

The design of this project is very modular. I was able to build small segments at a time and connect them all together at the end. The website eater.net also provides some fascinating tutorials and schematics, like the one shown below.

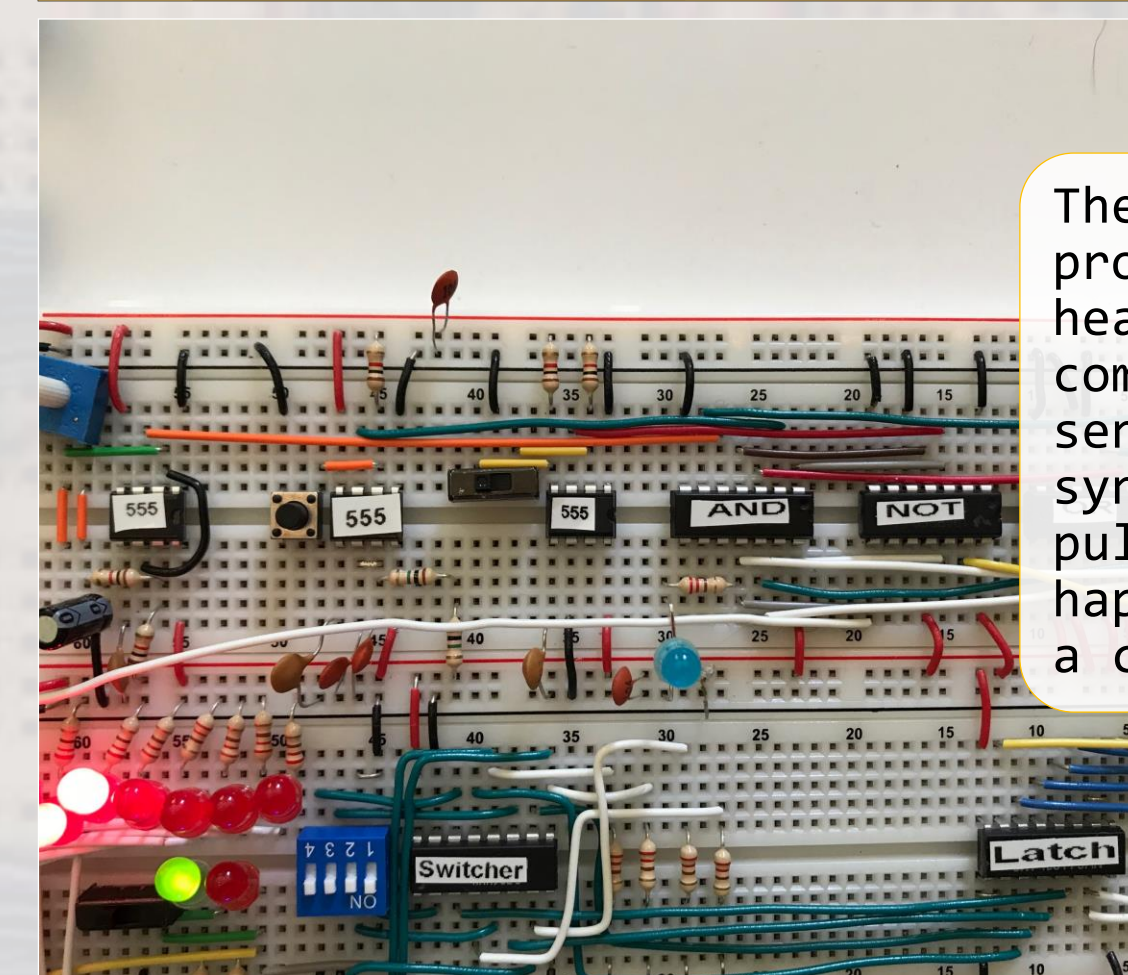


Programming

This computer has no long term program storage, so you must program it every time you turn it on. To program it, put it in programming mode, and use the command table to program binary commands into the RAM. Use the switches on the memory address register to scroll through the memory addresses. There is enough memory for 16 total commands and variables.

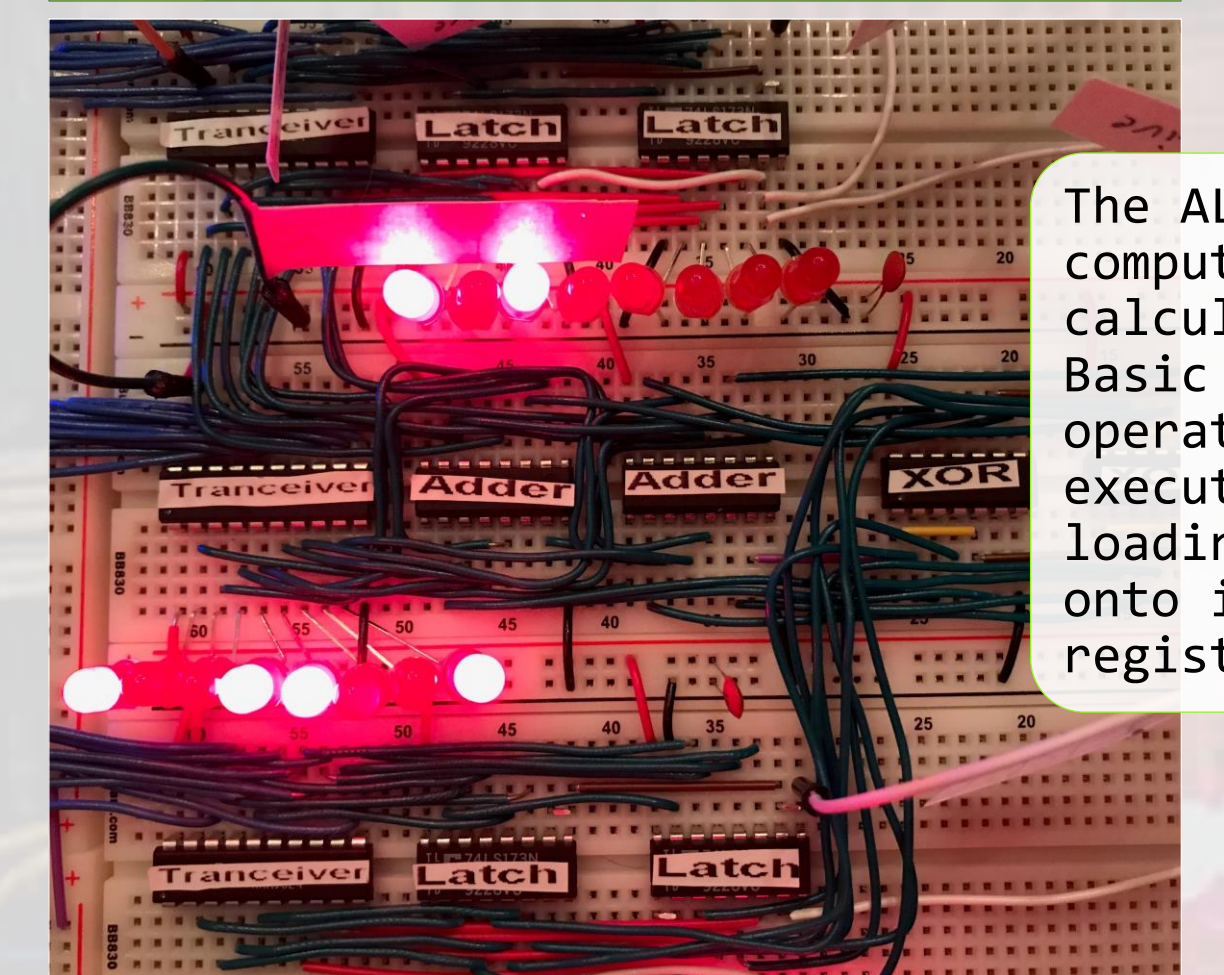
ASSEMBLY	BINARY	DESCRIPTION
NOP	0000	Skips to next line
LDA	0001	Loads address to A
ADD	0010	Adds address to A
SUB	0011	Subtracts address from A
STA	0100	Stores A at address
LDI	0101	Loads value to A
JMP	0110	Jumps to address
JC	0111	Jumps if carry
JZ	1000	Jumps if zero
OUT	1110	Output A
HLT	1111	Stop Clock

Clock Module



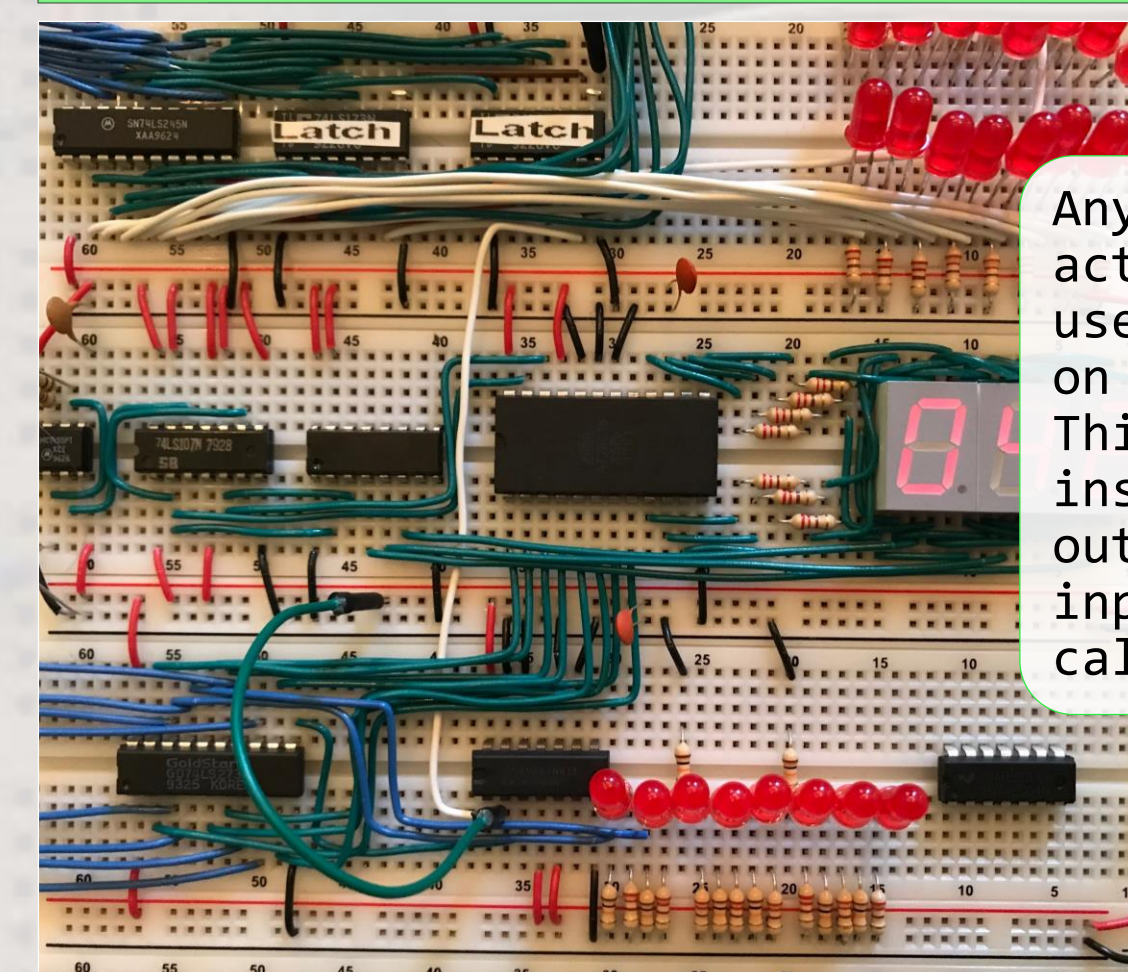
The clock module provides the heartbeat of the computer, sending synchronizing pulses. Nothing happens without a clock signal.

Arithmetic & Logic Unit



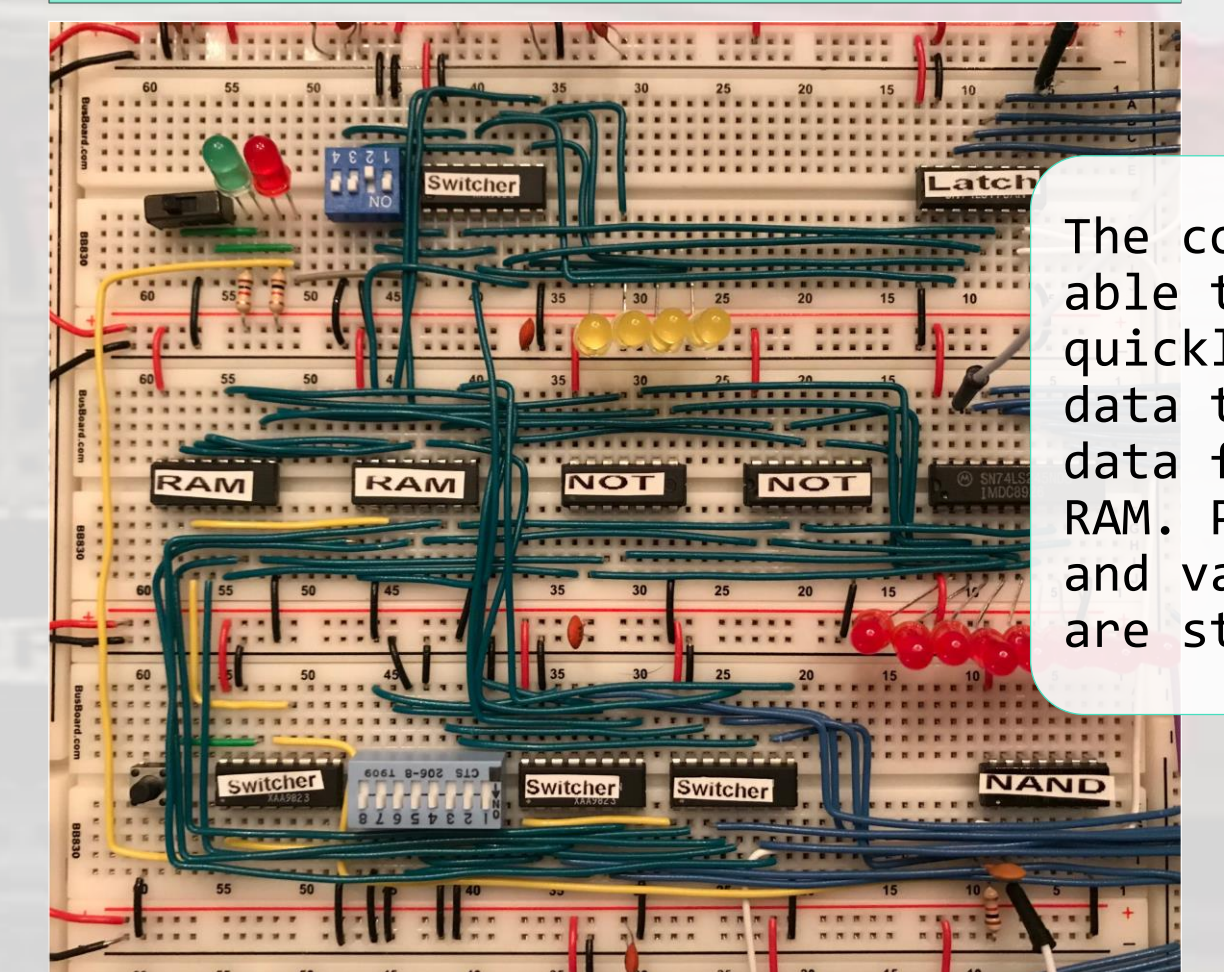
The ALU is the computer's calculator. Basic math operations are executed here by loading inputs onto its registers.

Data Registers



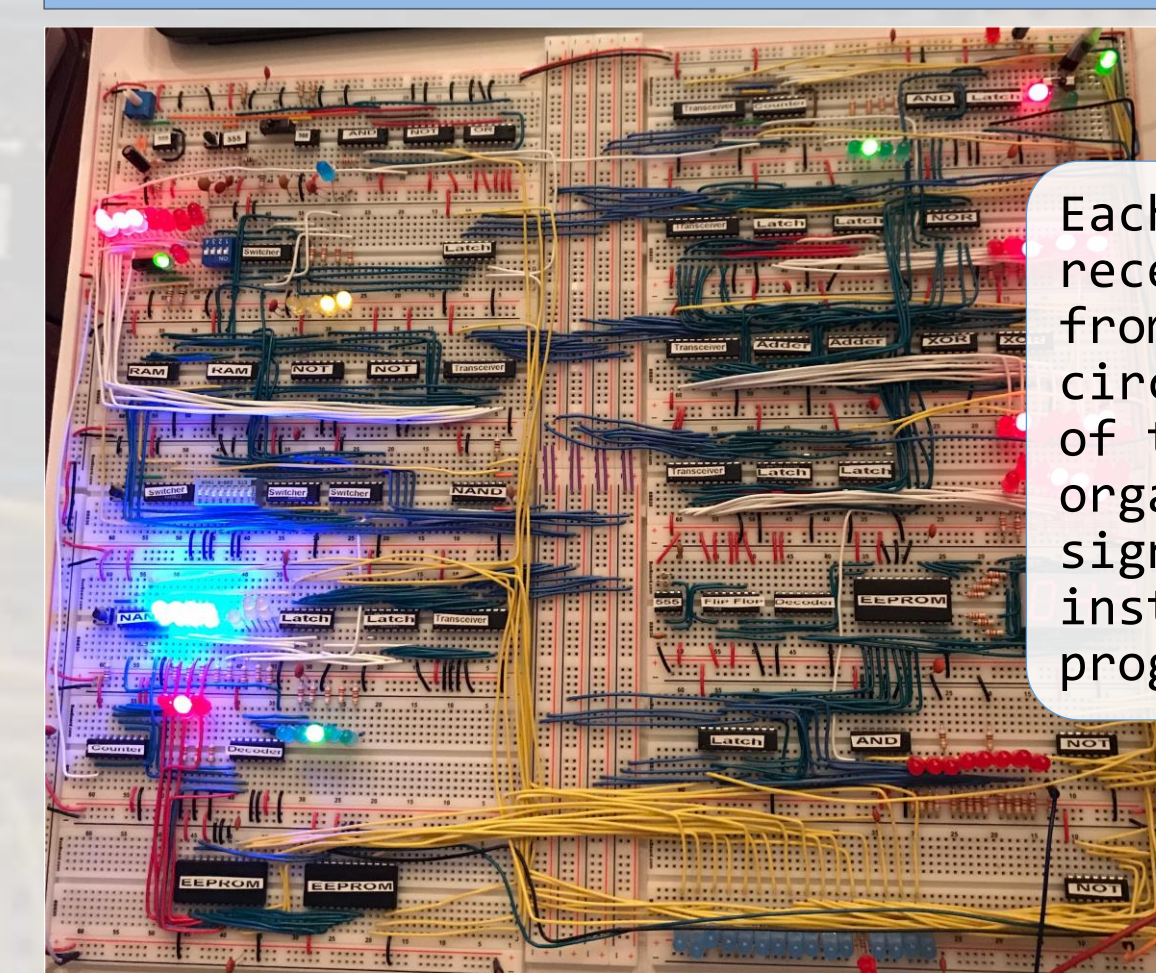
Any value actively being used is stored on a register. This includes instructions, outputs, and inputs for calculations.

Random Access Memory



The computer is able to very quickly store data to, or read data from, the RAM. Programs and variables are stored here.

Control Circuitry



Each module receives signals from the control circuitry. Many of these organized signals form instructions and programs.