Programming

Section 1: Logic

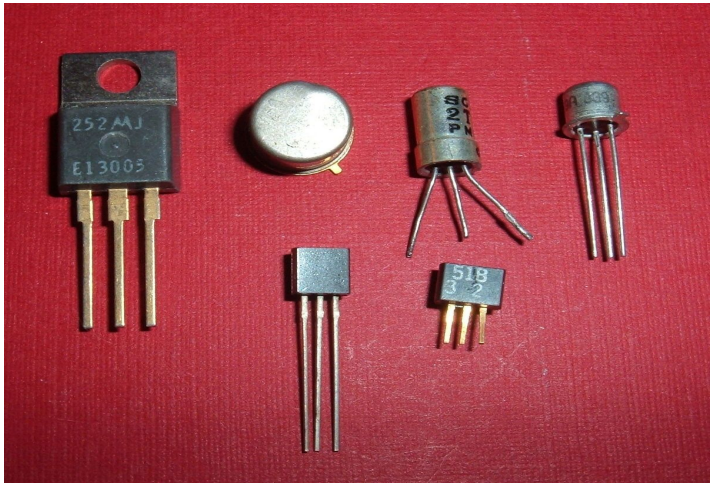
Abstraction

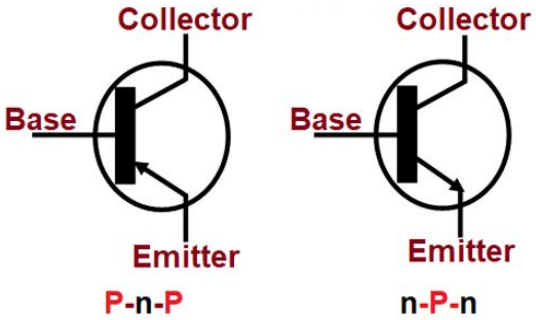
https://en.wikipedia.org/wiki/Abstraction_(computer_science):

In software engineering and computer science, **abstraction** is the process of generalizing concrete details,[1] such as attributes, away from the study of objects and systems to focus attention on details of greater importance.[2] Abstraction is a fundamental concept in computer science and software engineering, especially within the object-oriented programming paradigm.[3] Examples of this include:

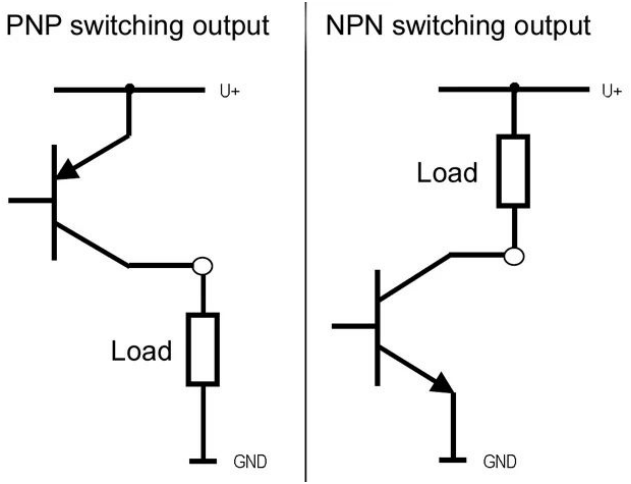
- the usage of abstract data types to separate usage from working representations of data within programs;[4]
- the concept of functions or subroutines which represent a specific way of implementing control flow;
- the process of reorganizing common behavior from groups of non-abstract classes into abstract classes using inheritance and sub-classes, as seen in object-oriented programming languages.

Transistors

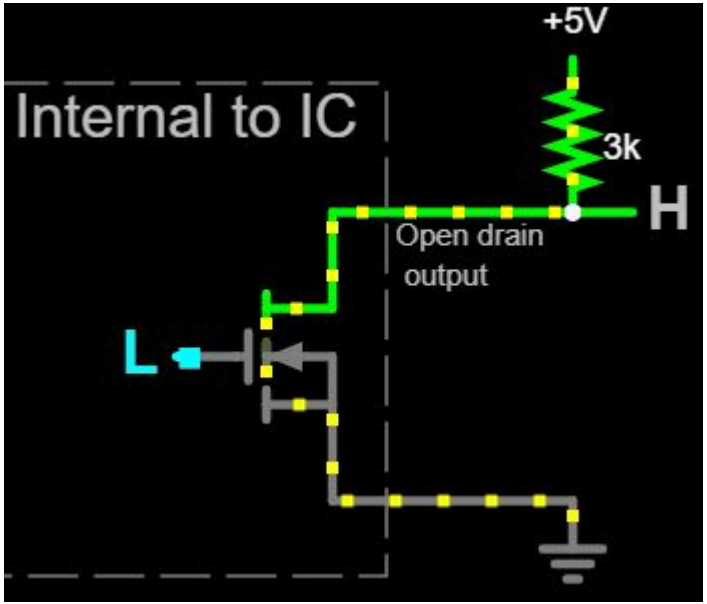




- Basic building block of modern electronics
- Can be thought of as a switch
- The Base controls the “on” and “off” between the Collector and Emitter

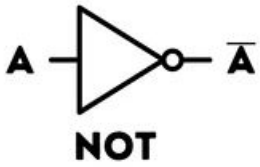


Transistor used as a “switch”



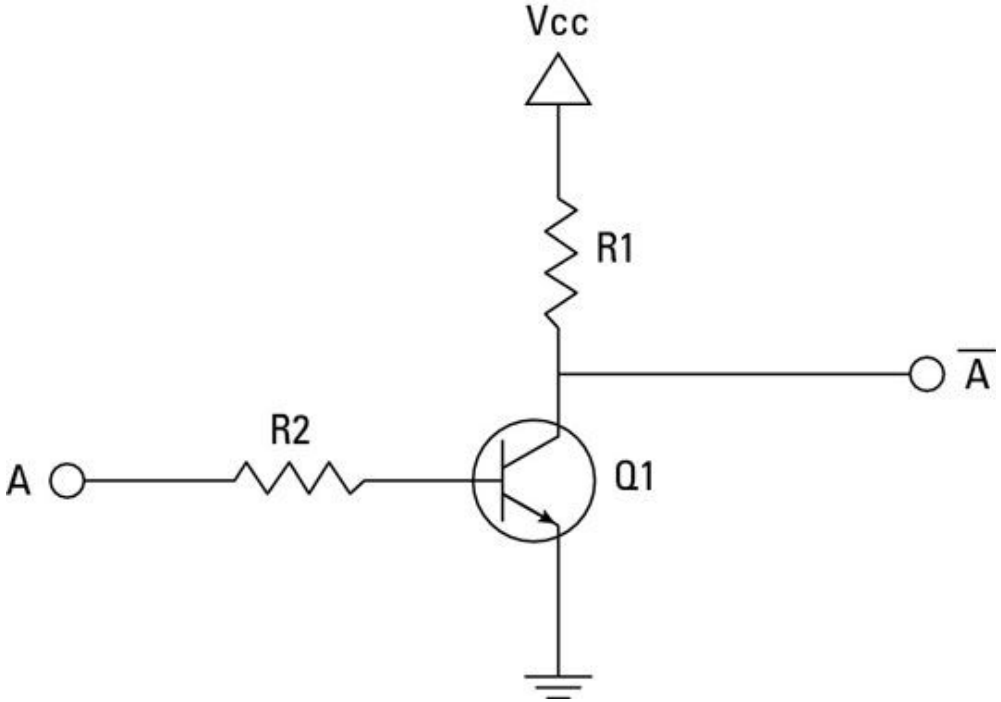
- Switching “on” provides a ground in this scenario. It can be the other way around.
- Using the transistor as a switch allows us to combine multiples to create logic

Not Gate



NOT Gate

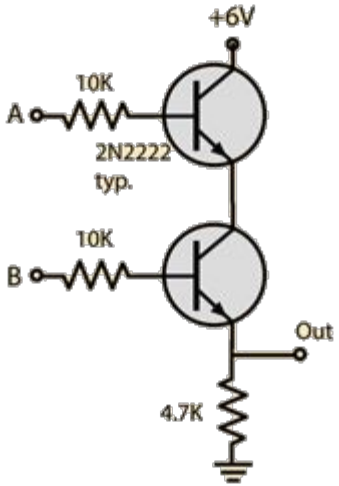
A	$\overline{A}$
0	1
1	0



And Gate

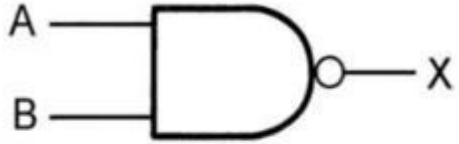
2 - input AND gate

A	B	Output
0	0	0
0	1	0
1	0	0
1	1	1

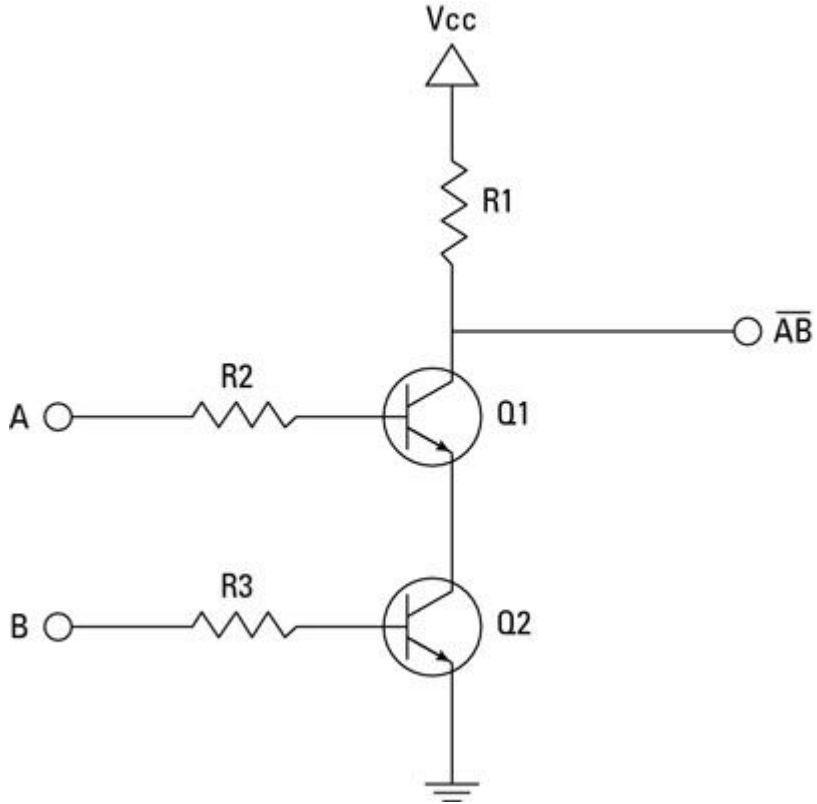


NAND Gate

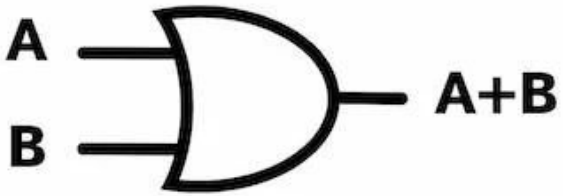
$$X = \overline{A.B}$$



A	B	X
0	0	1
0	1	1
1	0	1
1	1	0

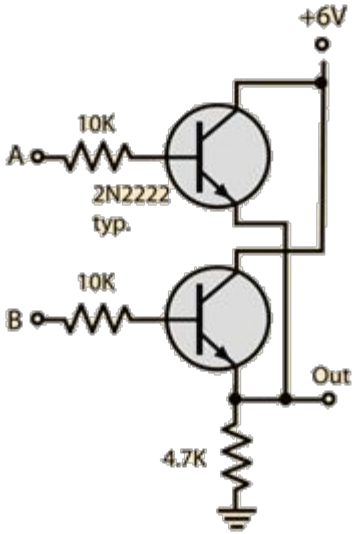


OR Gate

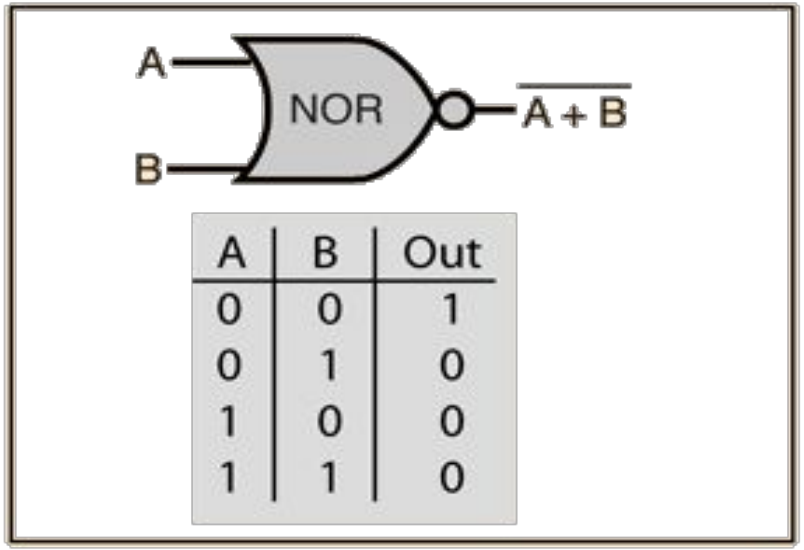


2 input OR gate

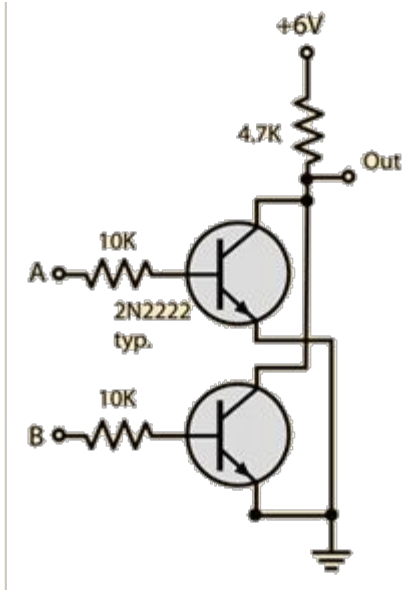
A	B	A+B
0	0	0
0	1	1
1	0	1
1	1	1



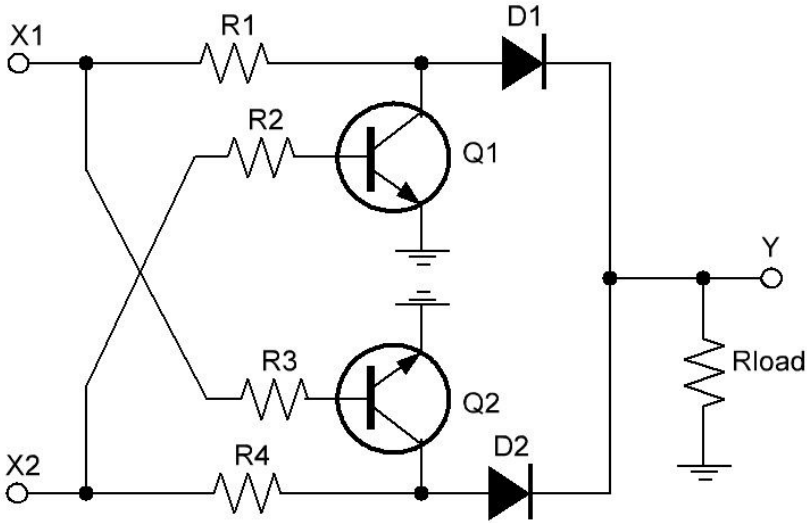
NOR Gate



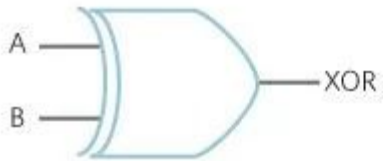
A	B	Out
0	0	1
0	1	0
1	0	0
1	1	0



XOR Gate

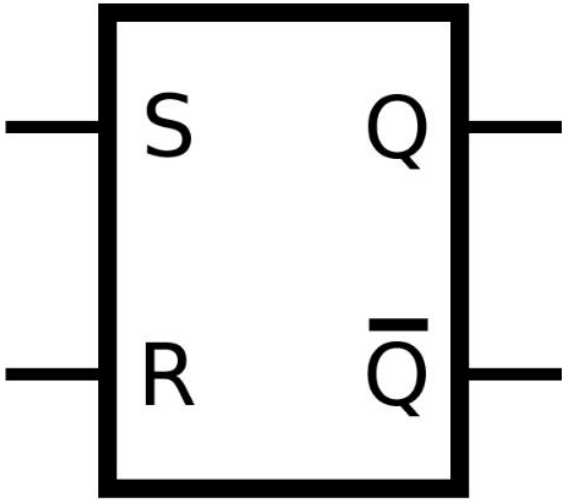


$$X = A \oplus B$$

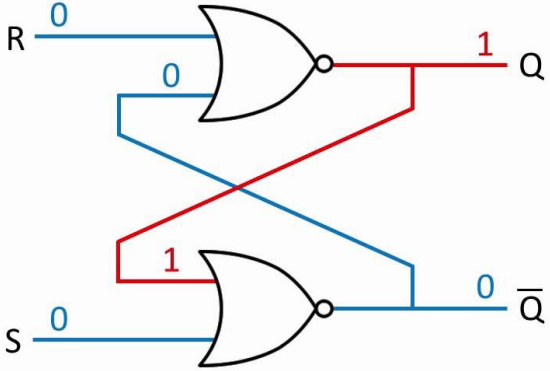


A	B	XOR
0	0	0
0	1	1
1	0	1
1	1	0

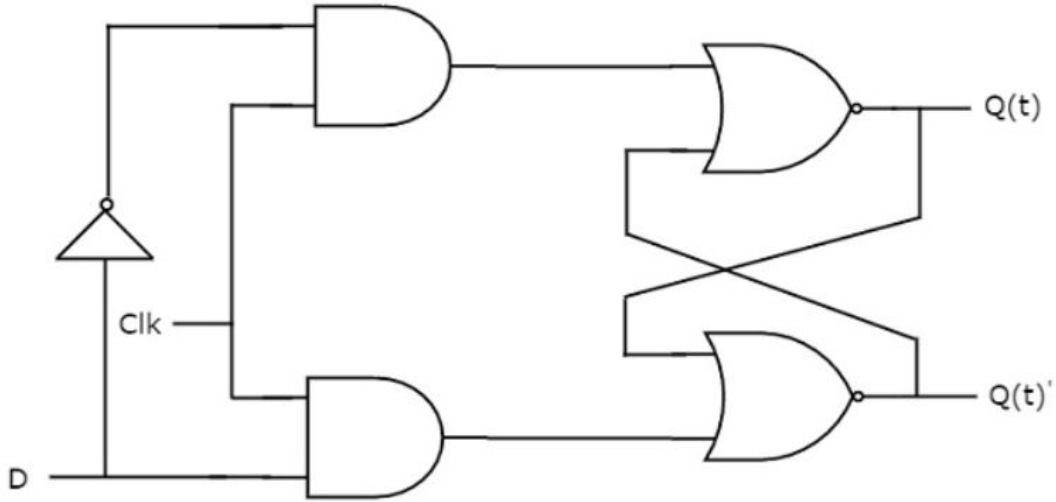
SR (Set-Reset) Latch

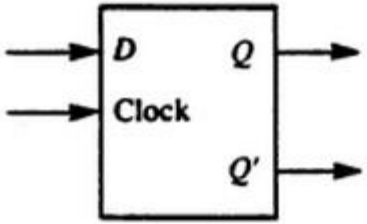


S	R	Q	$\overline{Q}$
0	0	1 0	0 1
0	1	0	1
1	0	1	0
1	1	0	0

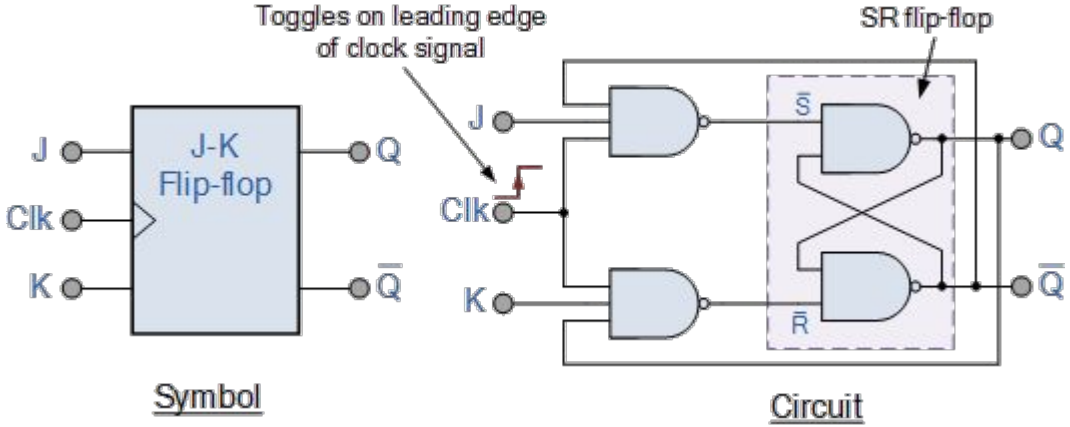


D Latch

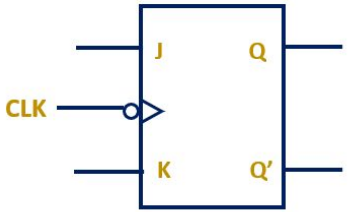




JK Flip Flop



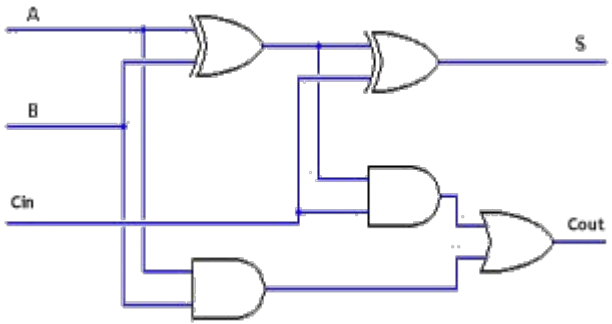
Symbol

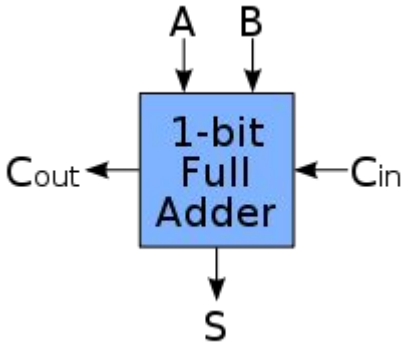


Truth Table

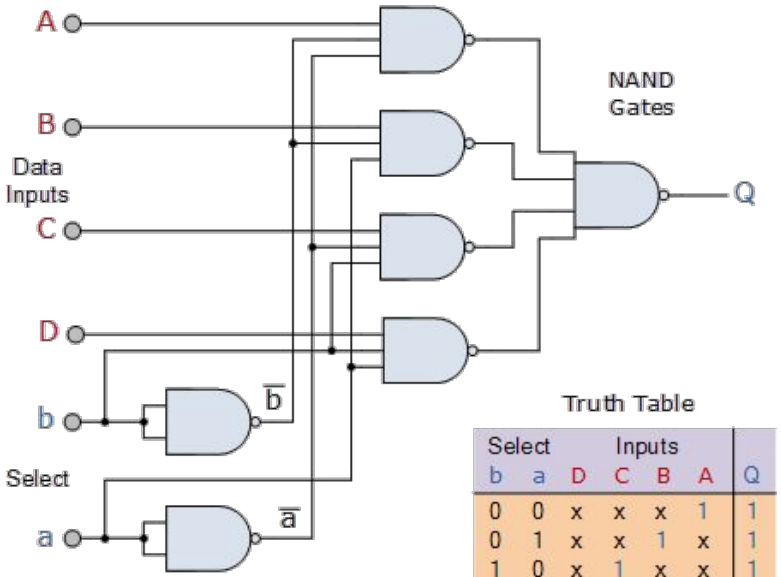
CLK	J	K	Q_{n+1}
↓	0	0	Q_n
↓	0	1	0
↓	1	0	1
↓	1	1	Q_n'

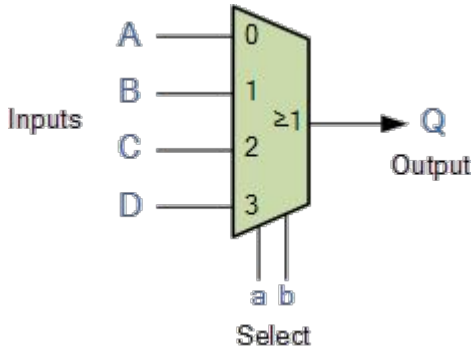
1 Bit full adder

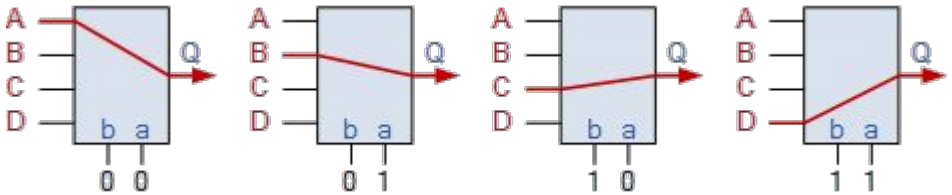




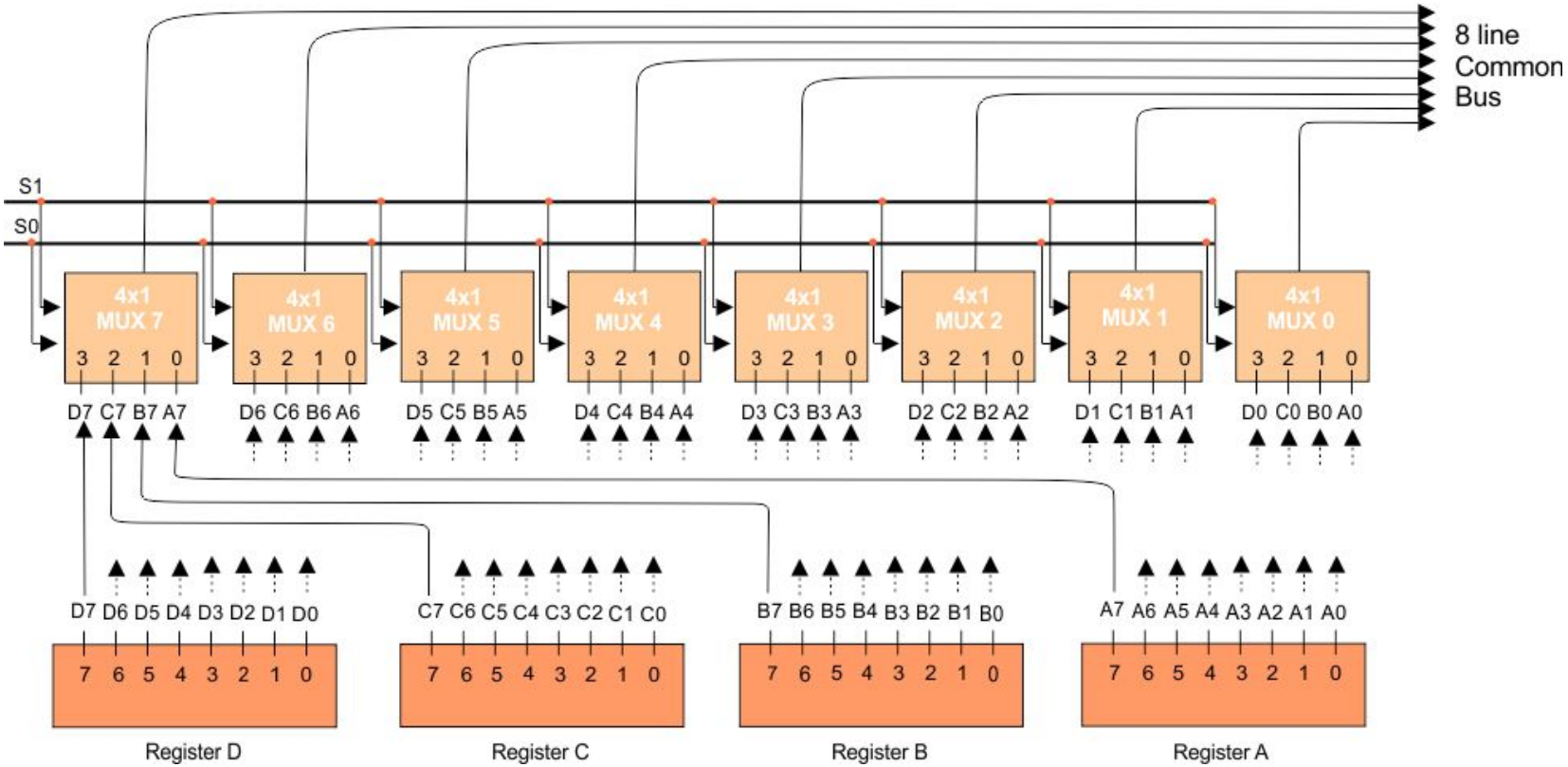
Multiplexer



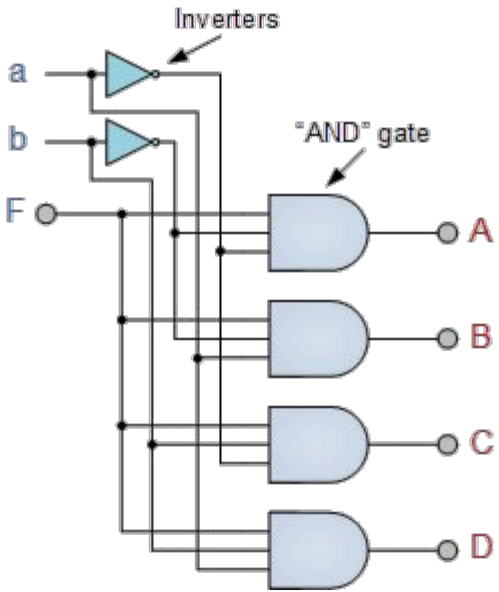


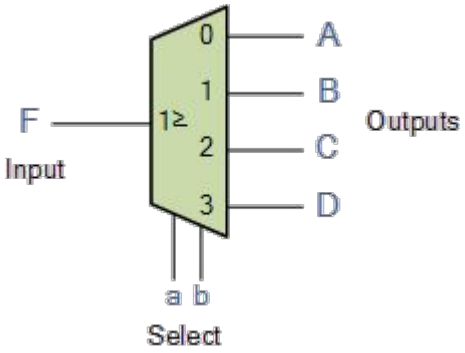


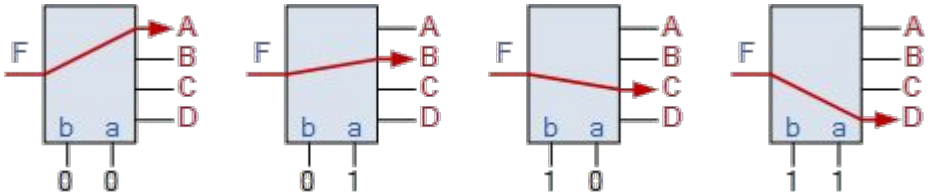
Multiplexer in data bus



Demultiplexer







7400 Series Integrated Circuits

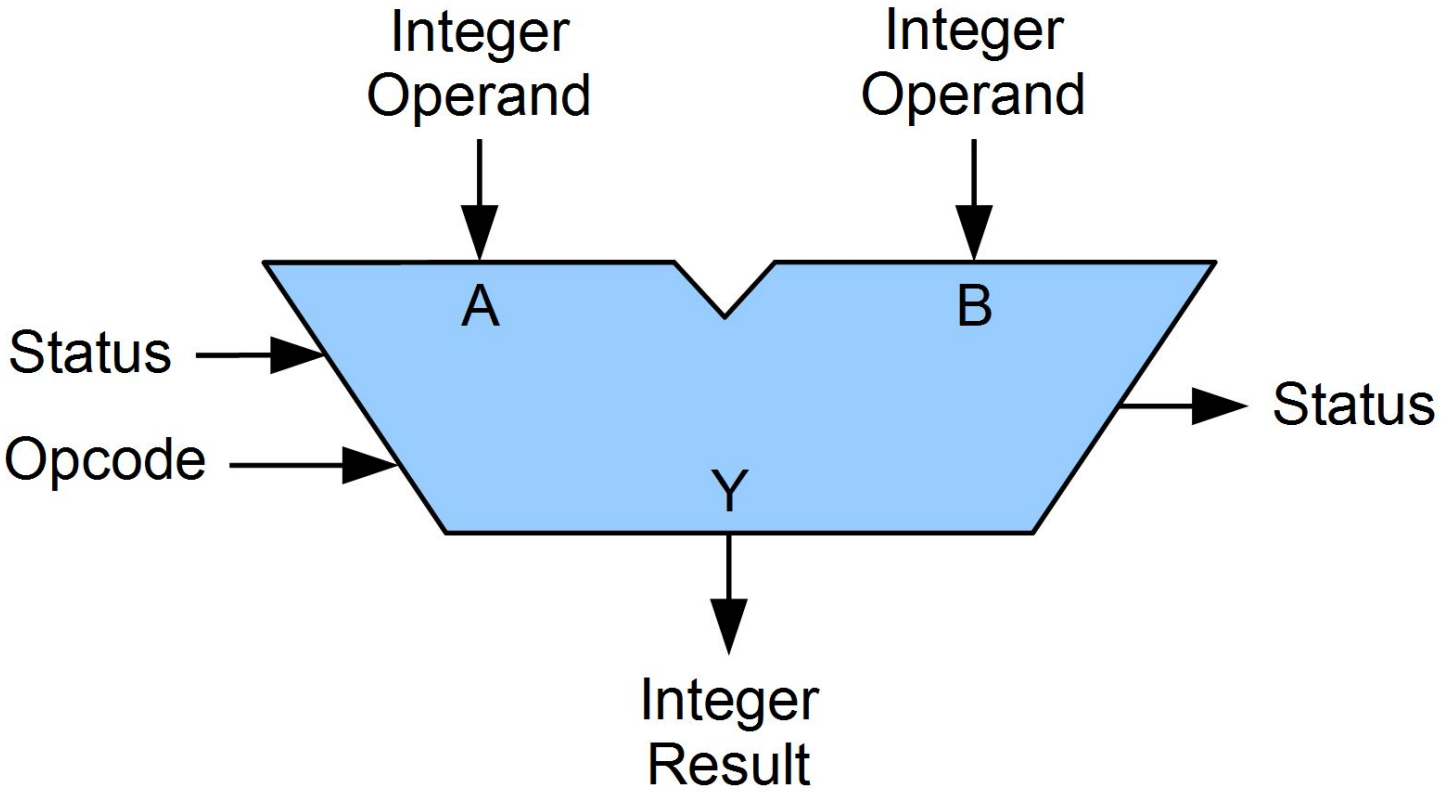
A lot of logic chips are still available today:

https://en.wikipedia.org/wiki/List_of_7400-series_integrated_circuits

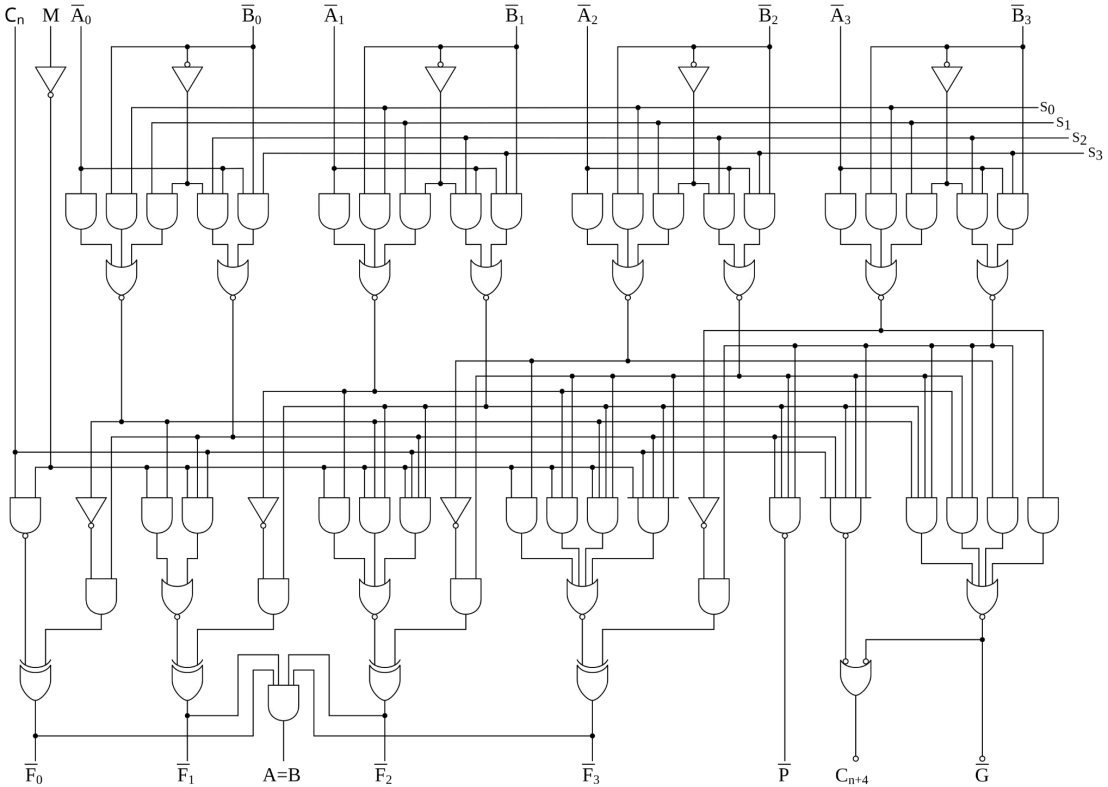
Overview

- Transistors are combined and abstracted into logic symbols
- Logic symbols are combined and abstracted into latch symbols, De/Multiplexers
- All of that is combined to create RAM, memory busses, etc.

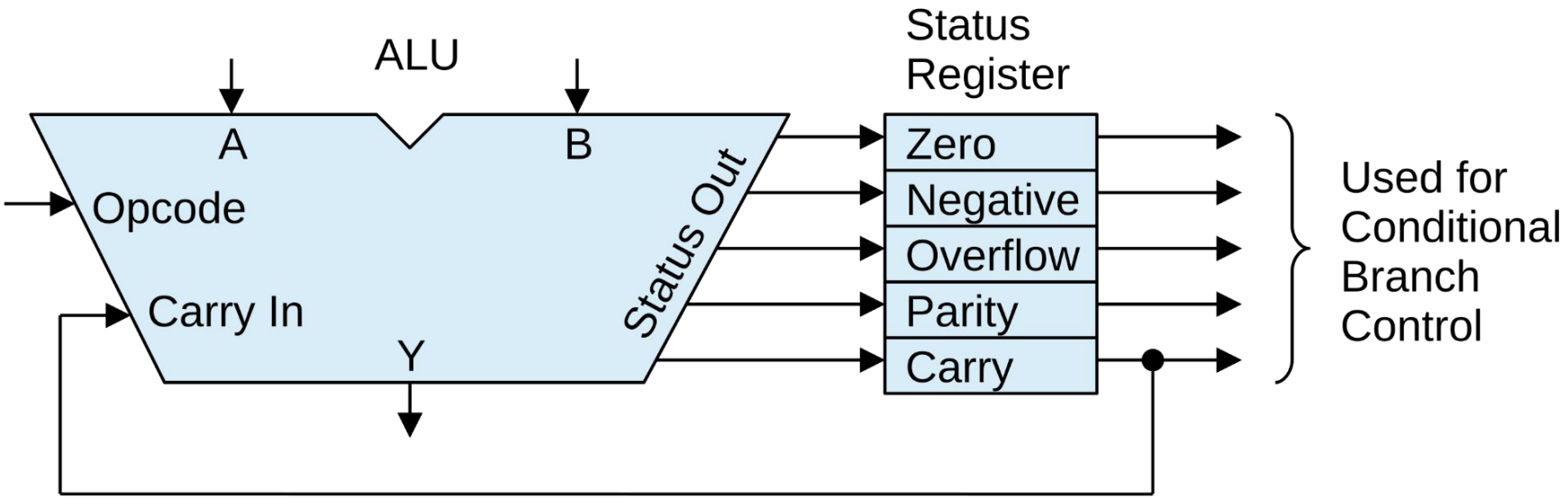
ALU


Integer
Operand
Integer
Operand
A
B
Status
Opcode
Y
Status
Integer
Result

ALU Internal logic



Status Registers



Arduino

- Uno uses ATmega328P 8 bit microcontroller
- https://store-usa.arduino.cc/products/arduino-uno-rev3
- https://ww1.microchip.com/downloads/aemDocuments/documents/MCU08/ProductDocuments/DataSheets/ATmega48A-PA-88A-PA-168A-PA-328-P-DS-DS40002061B.pdf
- https://ww1.microchip.com/downloads/aemDocuments/documents/MCU08/ProductDocuments/ReferenceManuals/AVR-InstructionSet-Manual-DS40002198.pdf

Instruction Cycle