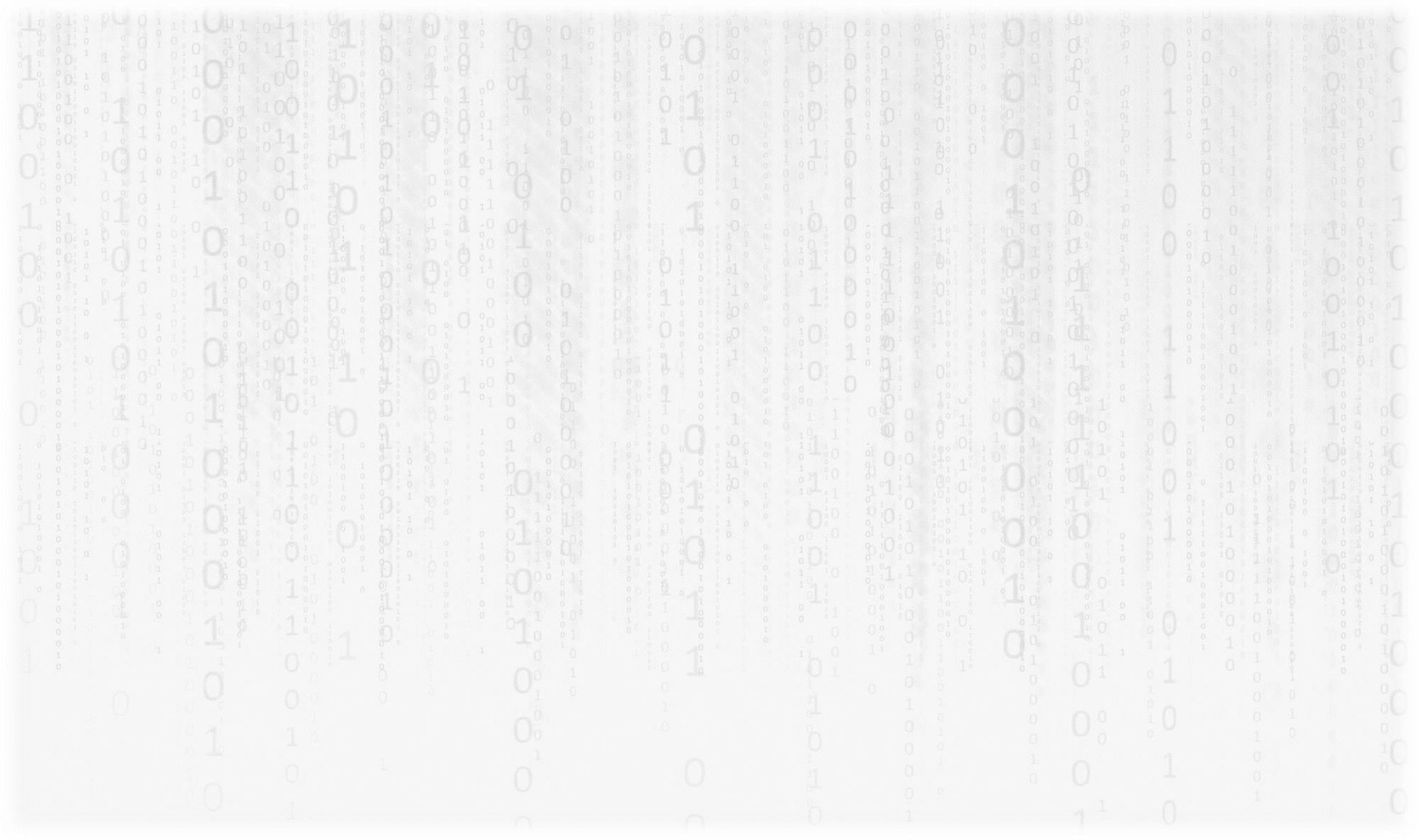
Assembly Language

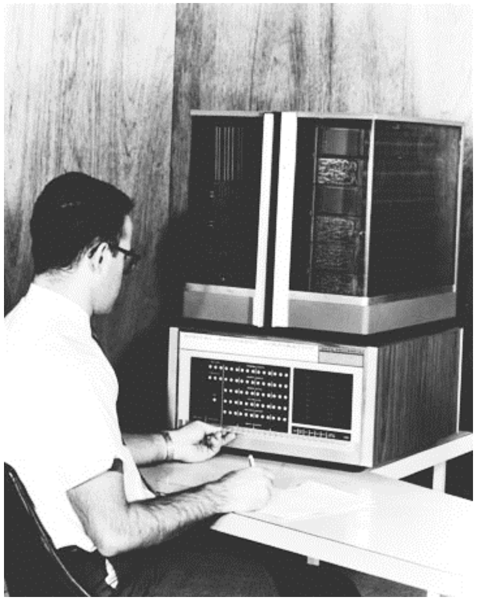
Telling a computer what to do



Computer Evolution

The rapid development of increasingly capable computers was largely driven by improved silicon fabrication techniques (integrated circuits).

The Electronic Computer



1970s

1980s

1990s

2000s

Arduino

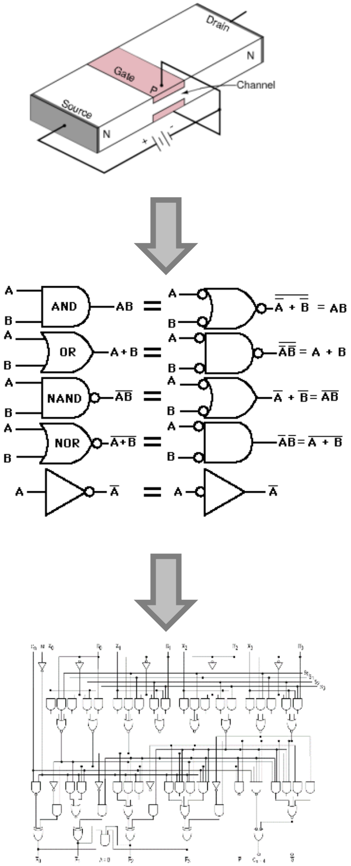
Rasberry Pi

Computer Components (Anatomy)

Modern computers are built from a hierarchical arrangement of devices, the lowest level of which is a complimentary pair of transistor switches (CMOS).

Definition

Micro

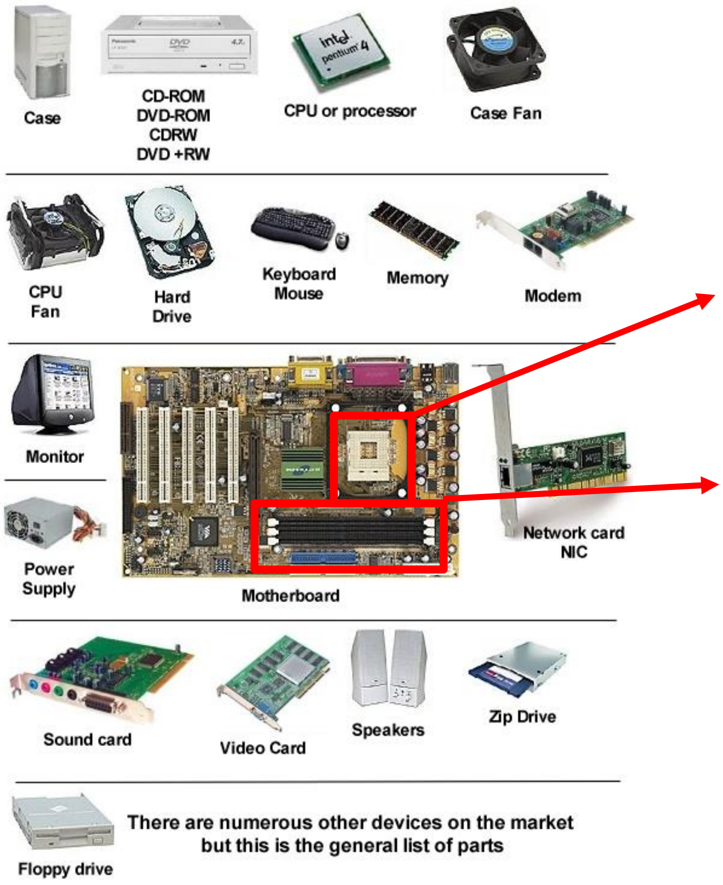

Drain
N
Gate
P
Channel
Source
N
AND
OR
NAND
NOR
AB
A+B
$\overline{AB}$
$\overline{A+B}$
$\overline{\overline{A}+\overline{B}} = AB$
$\overline{\overline{A}\,\overline{B}} = A + B$
$\overline{A} + \overline{B} = \overline{AB}$
$\overline{A}\,\overline{B} = \overline{A+B}$
$\overline{A}$

Transistors

Logic Gates

Circuits

Macro



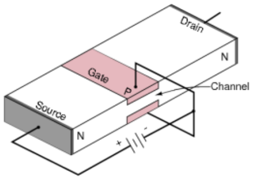
There are numerous other devices on the market but this is the general list of parts

CPU

RAM

Transistors

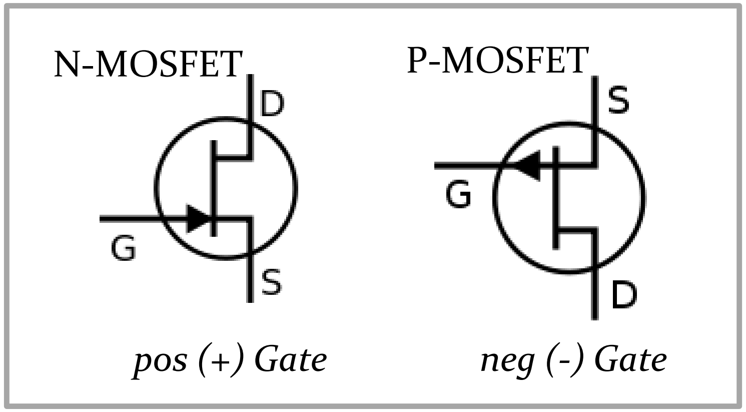
A semiconductor (solid-state) triode.



Definition

awesome

...Don’t get me started...

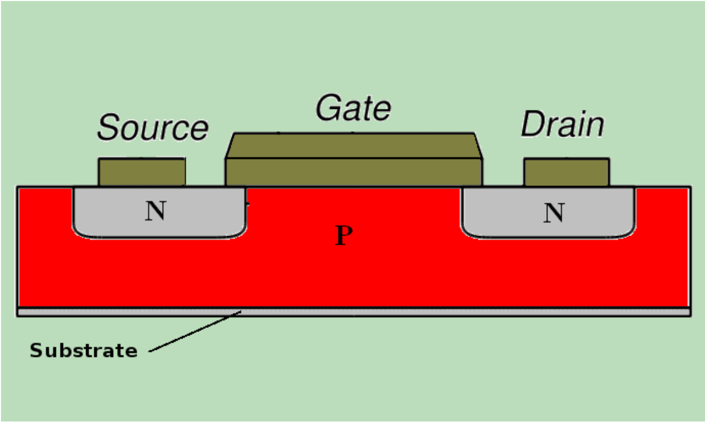


N-MOSFET

P-MOSFET

pos (+) Gate

neg (-) Gate

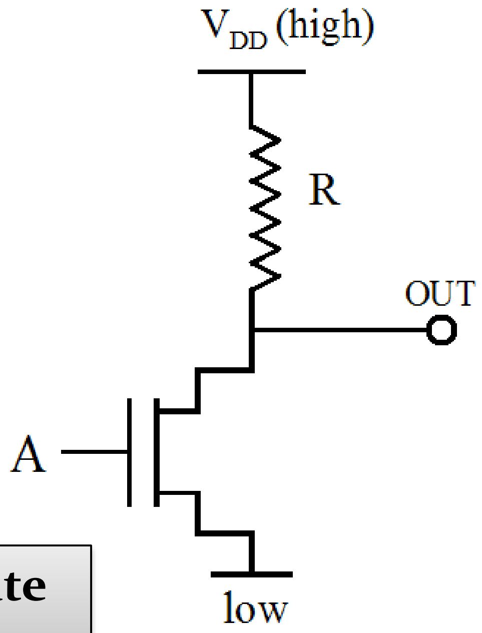


Logic Gates

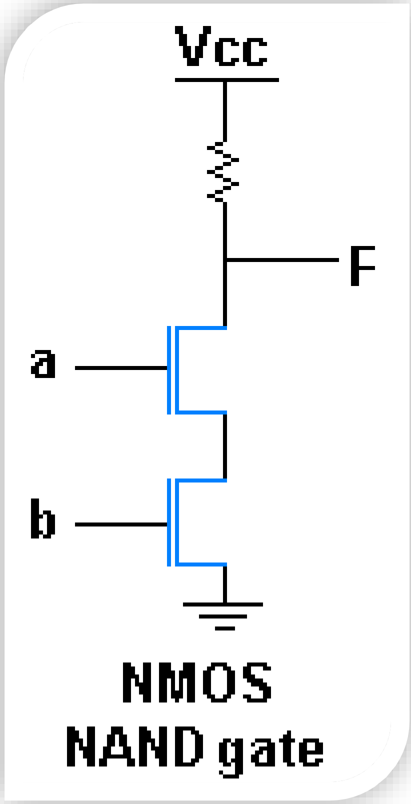
The fundamental units of computation.

Definition

NOT
AND
OR
XOR



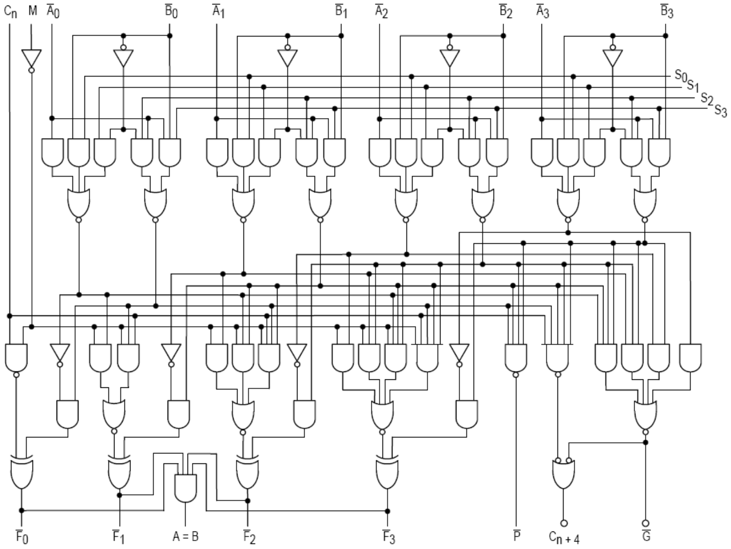
NMOS "NOT" Gate (Inverter)



Circuits

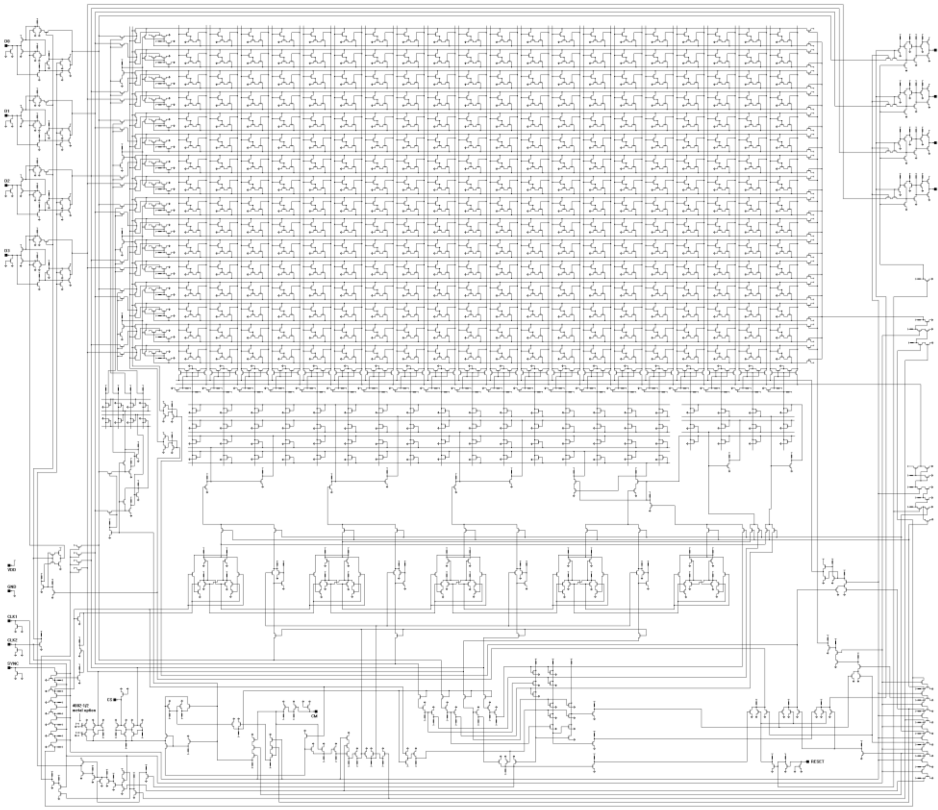
Integrated combinations of logic gates that’s store data, process data, and move data around.

Definition



The 74181 4-bit ALU

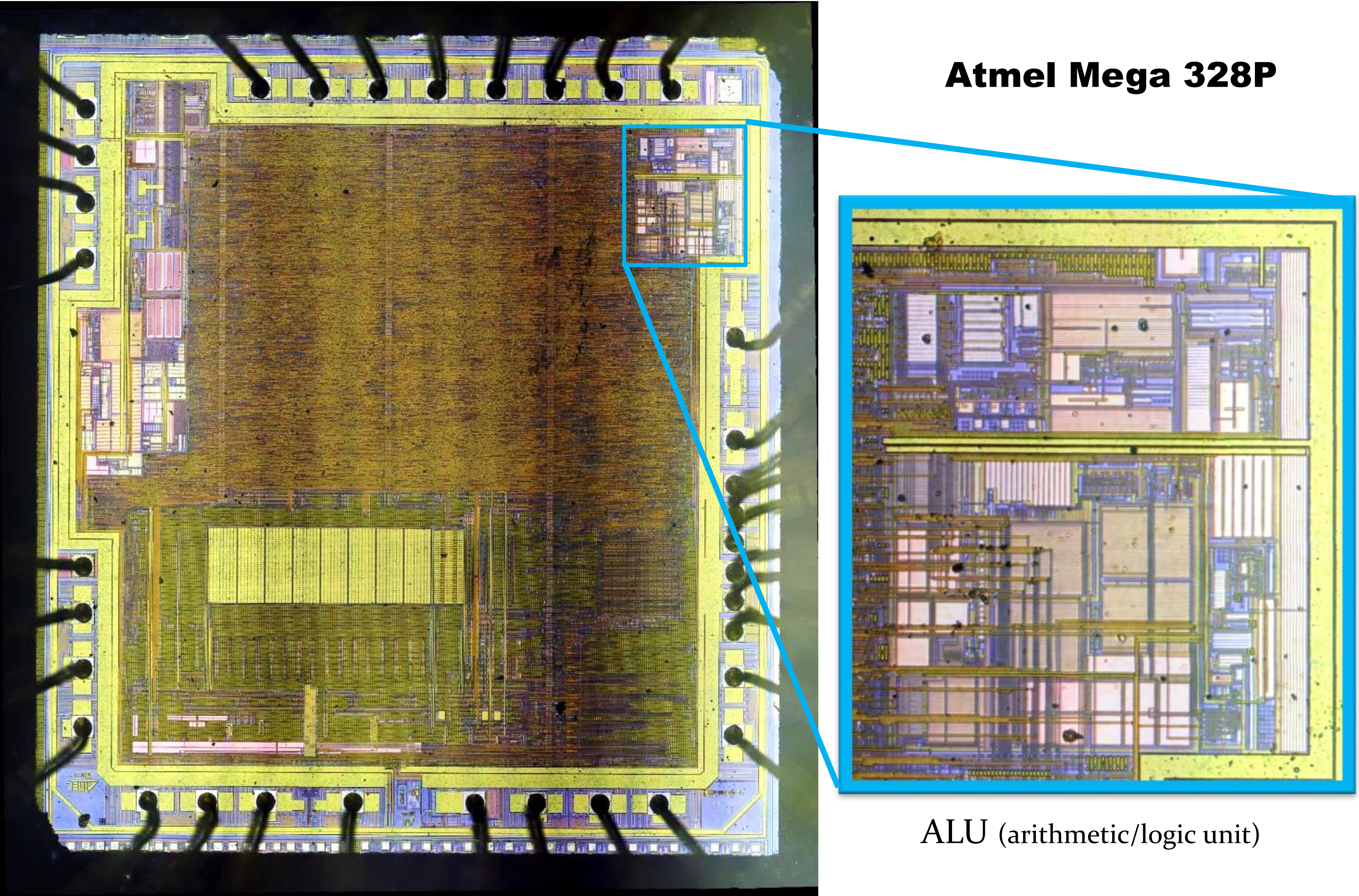
4002 320-bit RAM



An ALU performs a specific logic function of 4 input bits based on a “control input” (instruction)

RAM can read or write bits values (0 or 1) at a specific location (address)

Circuits in Silicon



Atmel Mega 328P

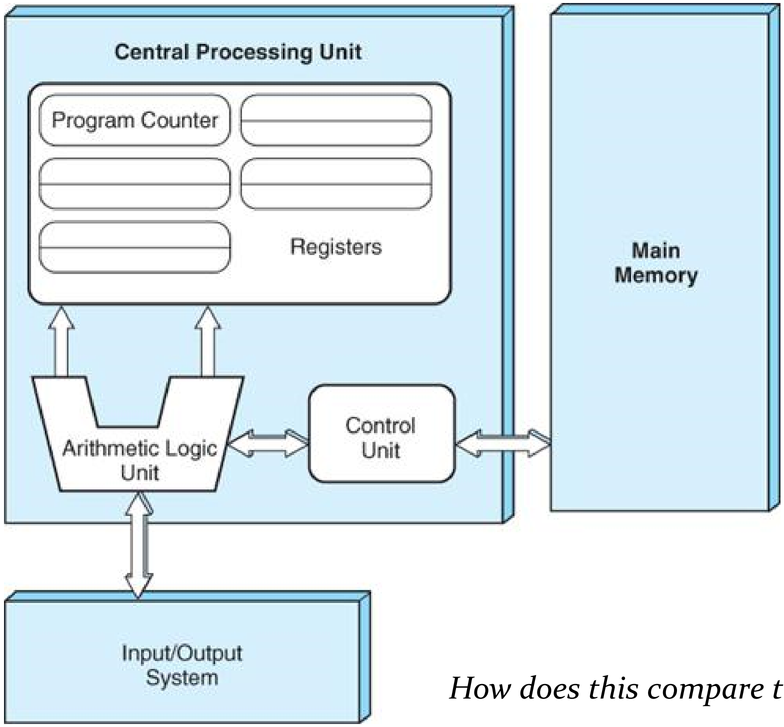
ALU (arithmetic/logic unit)

Von Neumann Architecture

John realized that "Data" and a "Program" do not require distinct substrates…they can be stored in the same hardware, system memory (often RAM) and then processed by a separate CPU.

Definition

John von Neumann

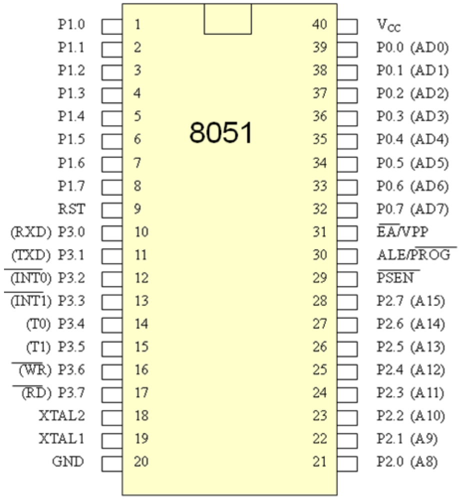


How does this compare to the brain?

Microcontrollers (µC, MCU)

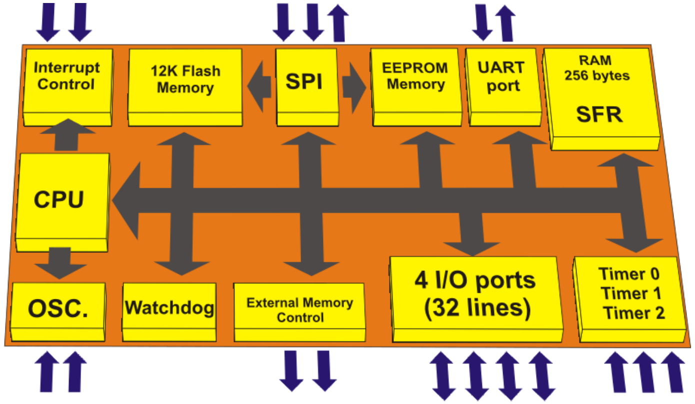
Definition

Microcontrollers, as the name suggests, are small computers containing a CPU and RAM. Unlike a PC, they lack an operating system (OS). They normally come on a board (PCB) integrated with ADCs, DACs, peripheral contacts, and a communication bus (serial, USB, etc.)



8051

Name	Pin	Pin	Name
P1.0	1	40	V_{CC}
P1.1	2	39	P0.0 (AD0)
P1.2	3	38	P0.1 (AD1)
P1.3	4	37	P0.2 (AD2)
P1.4	5	36	P0.3 (AD3)
P1.5	6	35	P0.4 (AD4)
P1.6	7	34	P0.5 (AD5)
P1.7	8	33	P0.6 (AD6)
RST	9	32	P0.7 (AD7)
(RXD) P3.0	10	31	$\overline{\text{EA}}$/VPP
(TXD) P3.1	11	30	ALE/$\overline{\text{PROG}}$
($\overline{\text{INT0}}$) P3.2	12	29	$\overline{\text{PSEN}}$
($\overline{\text{INT1}}$) P3.3	13	28	P2.7 (A15)
(T0) P3.4	14	27	P2.6 (A14)
(T1) P3.5	15	26	P2.5 (A13)
($\overline{\text{WR}}$) P3.6	16	25	P2.4 (A12)
($\overline{\text{RD}}$) P3.7	17	24	P2.3 (A11)
XTAL2	18	23	P2.2 (A10)
XTAL1	19	22	P2.1 (A9)
GND	20	21	P2.0 (A8)



Interrupt Control

12K Flash Memory

SPI

EEPROM Memory

UART port

RAM 256 bytes

SFR

CPU

OSC.

Watchdog

External Memory Control

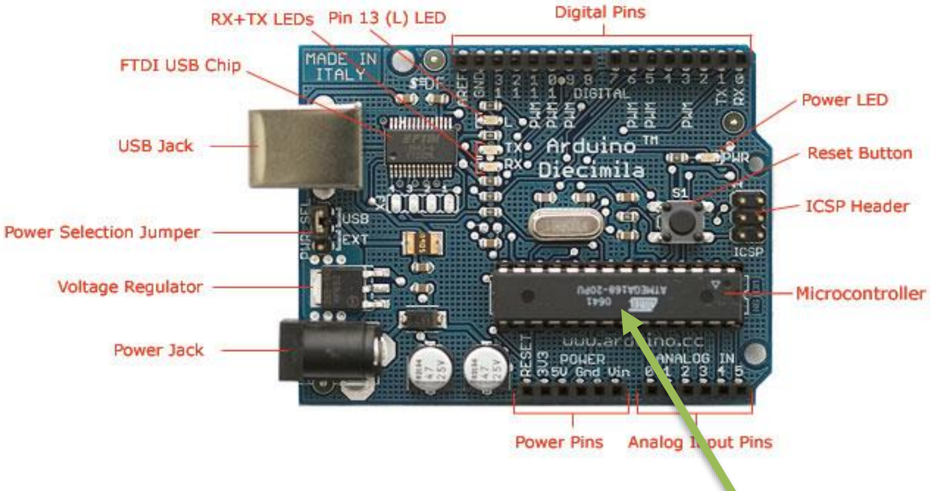
4 I/O ports (32 lines)

Timer 0
Timer 1
Timer 2

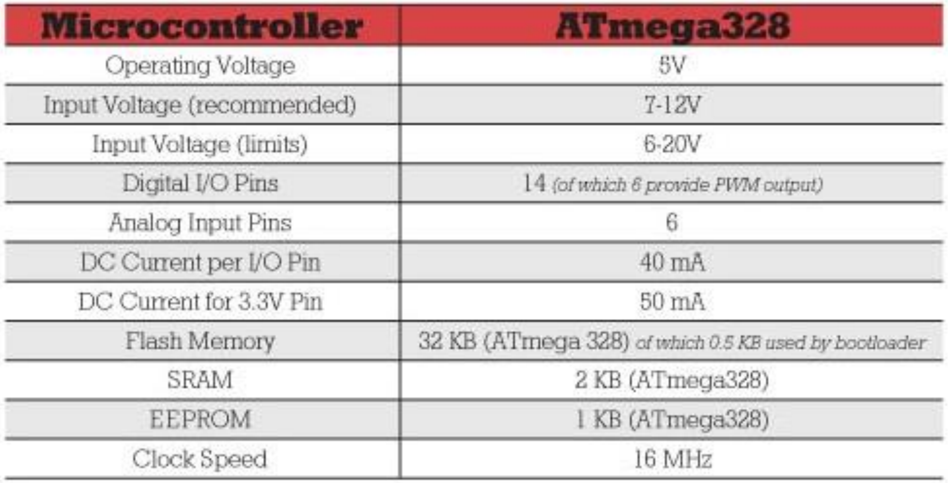
Arduino

Definition

Arduino is an open-source, open-hardware platform that integrates a slow (16 MHz) microcontroller, ADCs, digital ports, PWMs, counters, voltage regulators, USB-to-Serial converter, and useful connectors (headers) onto a single board.



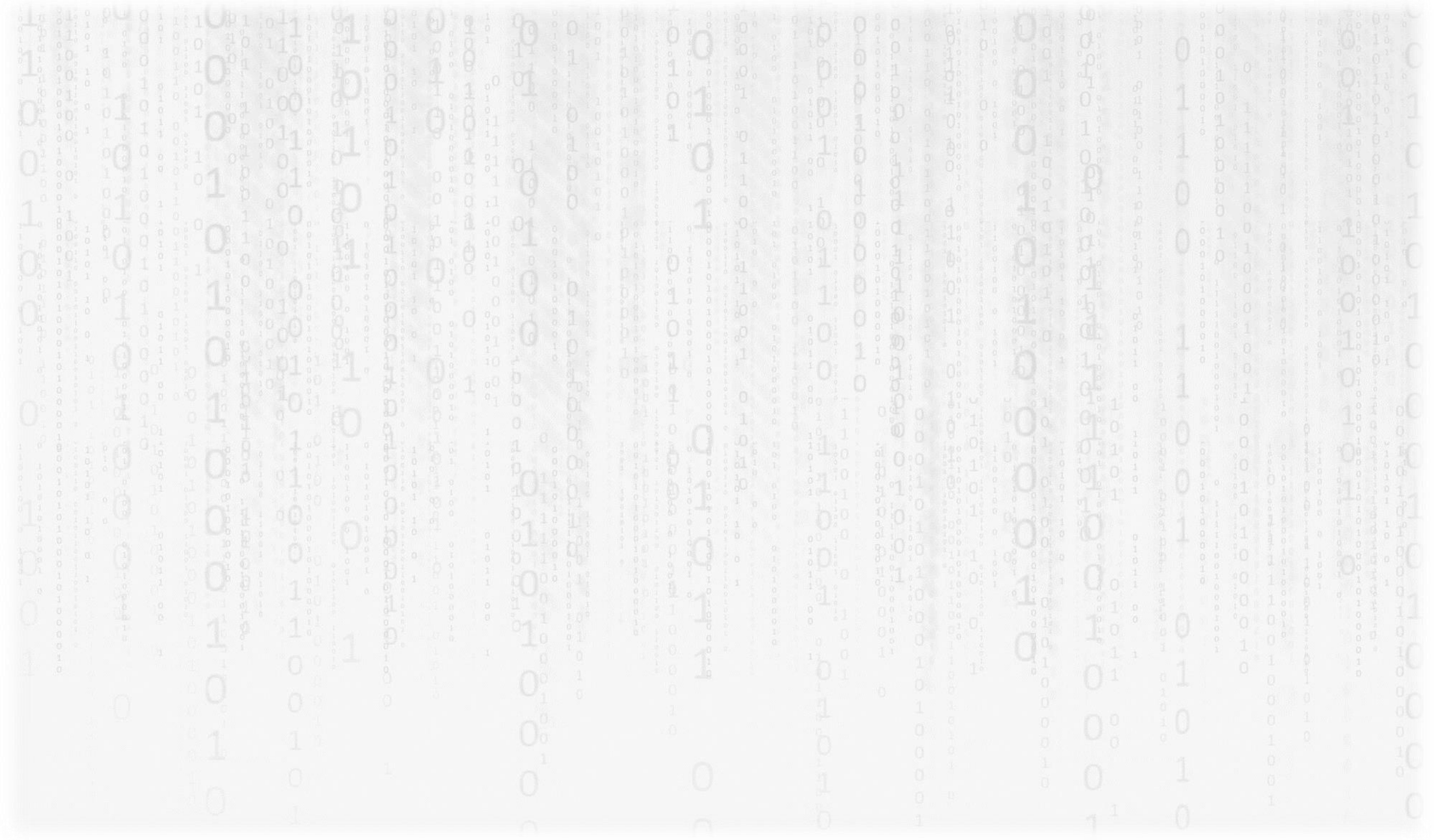
Photograph by SparkFun Electronics. Used under the Creative Commons Attribution Share-Alike 3.0 license.



Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	14 *(of which 6 provide PWM output)*
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega 328) *of which 0.5 KB used by bootloader*
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Clock Speed	16 MHz

Atmel Mega 328P

How do you tell a (binary) computer what to do?



Binary Representation

The bits 0 and 1 can be used to represent any number, as can the digits 0 through 9, when combined appropriately. The number of distinct values that can be represented depends on the number of bits used.

Definition

"Let's count in Binary!"

0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
...
33

#Values = $2^{\#Bits}$

1-bit:	2
8-bits:	256
16-bits:	65,536
32-bits:	4,294,967,296 (~4 billion)
64-bits:	18,446,744,073,709,551,616 (~18 quintillion)
128-bits:	340,282,366,920,938,463,463,374,607,431,768,211,456 (~340 undecillion)

Binary Data Types

Definition Conventions for translating a set of binary values into another format (integer, floating point, text, colour, etc.). The number and range of distinct values in the new format is constrained by the bit-depth of the binary representation (e.g. 8-bit text = 256 possible letters).

Data Types:

- **Boolean** – True vs. False (1-bit)
- **Integers** - *signed/unsigned*

  8-bit, 16-bit, 32-bit, 64-bit

  Note: Range depends on signed/unisgned:

  8-bit signed: 0 to 256

  8-bit unsigned: -127 to 127
- **Floating Point** (with decimal point)

  32 and 64-bit

  (Single) *(Double)* precision
- **Strings/Characters**
  - (ASCII/UNICODE)

- Others?

Binary Memory Block

```
001010011010100010110111010001011011000101011101010010010
010001010100101010100000101010111101100101010100101010010
000010010110100010100001011010100010111001011010100000101
010100010101010101001010001010100100010101001010101010011
111010101010001011101010010100100101000001001001001010001
010101001010101010010101010010101001010101001000100101101
```

ASCII Text File:	01000100011000010111010001100001	=	“Data”
16-bit Integers:	01000100011000010111010001100001	=	17505 29793
32-bit Integer:	01000100011000010111010001100001	=	1147237473
32-bit Float:	01000100011000010111010001100001	=	901.81842041015625

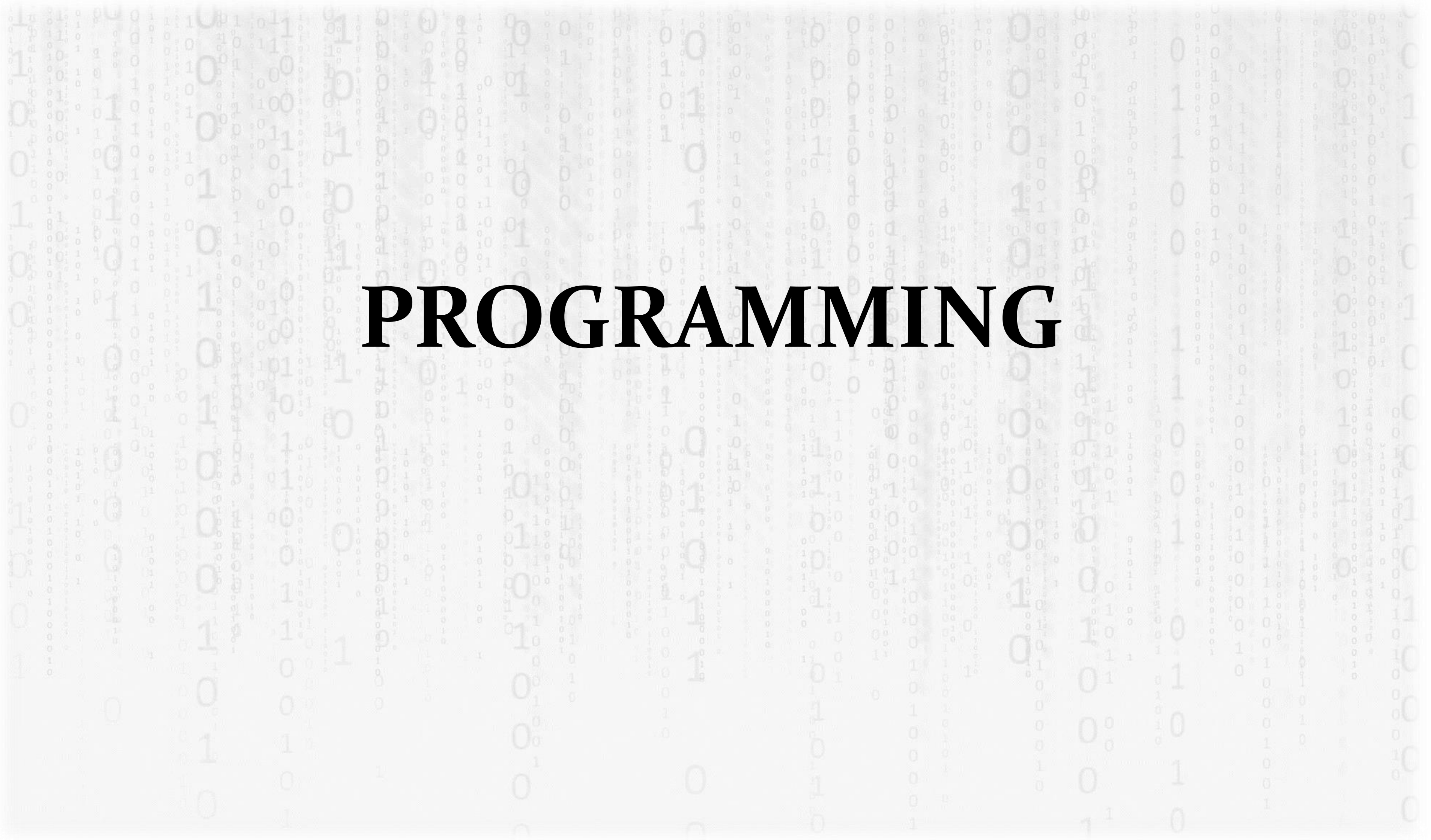
Note

Hexadecimal Representation

Base 16: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F

(each Hex numeral is equivalent to 4-bits, two Hex is a byte,…convenient)

How do you tell a (binary) computer what to do?

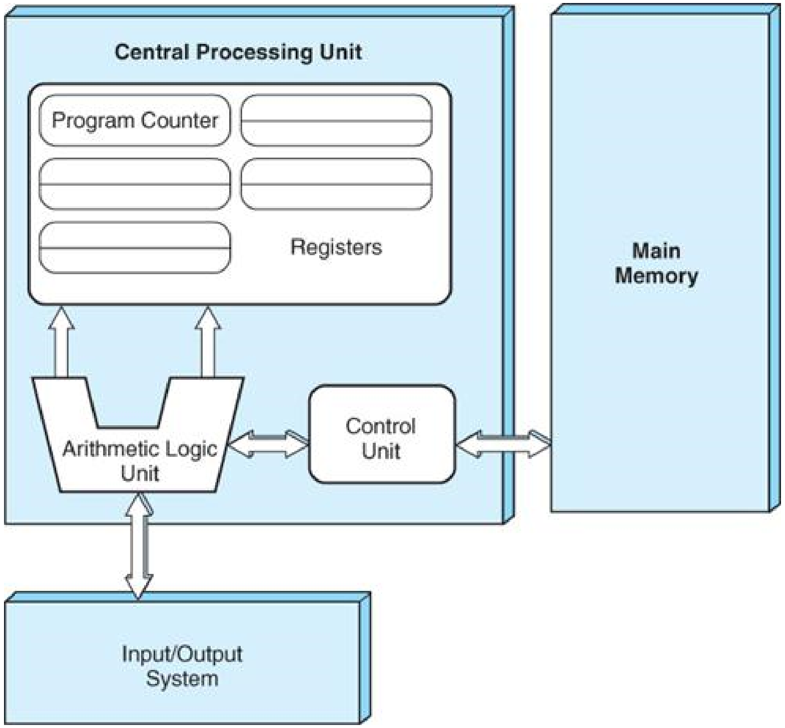


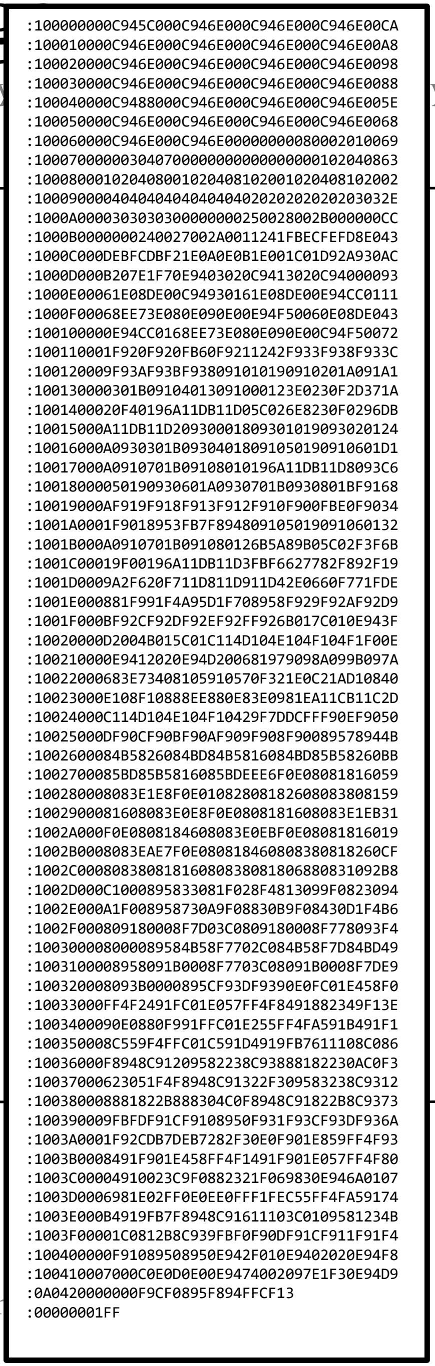
PROGRAMMING

Assembly Languag

The lowest (sensible) level that you can specify your computer.

Definition





```
:100000000C945C000C946E000C946E000C946E00CA
:100010000C946E000C946E000C946E000C946E00A8
:100020000C946E000C946E000C946E000C946E0098
:100030000C946E000C946E000C946E000C946E0088
:100040000C9488000C946E000C946E000C946E005E
:100050000C946E000C946E000C946E000C946E0068
:100060000C946E000C946E00000000000800201069
:1000700000030407000000000000000000102040863
:100080000102040801020408102001020408102002
:100090000404040404040404020202020202203032E
:1000A0000303030300000000250028002B000000CC
:1000B000000000240027002A0011241FBECFEFD8E043
:1000C000DEBFCDBF21E0A0E0B1E001C01D92A930AC
:1000D000B207E1F70E9403020C9413020C94000093
:1000E00061E08DE00C94930161E08DE00E94CC0111
:1000F00068EE73E080E090E00E94F50060E08DE043
:100100000E94CC0168EE73E080E090E00C94F50072
:100110001F920F920FB60F9211242F933F938F933C
:100120009F93AF93BF9380910101909102001A091A1
:100130000301B0910401309100123E0230F2D371A
:1001400020F40196A11DB11D05C026E8230F0296DB
:10015000A11DB11D209300018093010190930201 24
:10016000A0930301B09304018091050190910601D1
:10017000A0910701B09108010196A11DB11D8093C6
:1001800005019093060 1A0930701B0930801BF9168
:10019000AF919F918F913F912F910F900FBE0F9034
:1001A0001F9018953FB7F894809105019091060132
:1001B000A0910701B091080126B5A89B05C02F3F6B
:1001C00019F00196A11DB11D3FBF6627782F892F19
:1001D0009A2F620F711D811D911D42E0660F771FDE
:1001E000881F991F4A95D1F708958F929F92AF92D9
:1001F000BF92CF92DF92EF92FF926B017C010E943F
:10020000D2004B015C01C114D104E104F104F1F00E
:100210000E9412020E94D200681979098A099B097A
:10022000683E7340810591057 0F321E0C21AD10840
:10023000E108F10888EE880E83E0981EA11CB11C2D
:10024000C114D104E104F10429F7DDCFFF90EF9050
:10025000DF90CF90BF90AF909F908F90089578944B
:1002600084B5826084BD84B5816084BD85B58260BB
:1002700085BD85B5816085BDEEE6F0E08081816059
:1002800008083E1E8F0E0108280818260808380815 9
:10029000816080 83E0E8F0E0808181608083E1EB31
:1002A000F0E0808184608083E0EBF0E08081816019
:1002B0008083EAE7F0E08081846080838081 8260CF
:1002C000808380818160808380818068808310 92B8
:1002D000C1000895833081F028F4813099F0823094
:1002E000A1F008958730A9F08830B9F08430D1F4B6
:1002F000809180008F7D03C0809180008F778093F4
:100300008000089584B58F7702C084B58F7D84BD49
:10031000 08958091B0008F7703C08091B0008F7DE9
:100320008093B0000895CF93DF9390E0FC01E458F0
:10033000FF4F2491FC01E057FF4F8491882349F13E
:1003400090E0880F991FFC01E255FF4FA591B491F1
:100350008C559F4FFC01C591D4919FB7611108C086
:10036000F8948C91209582238C93888182230AC0F3
:10037000 0623051F4F8948C91322F309583238C9312
:100380008881822B888304C0F8948C91822B8C9373
:100390009FBFDF91CF9108950F931F93CF93DF936A
:1003A0001F92CDB7DEB7282F30E0F901E859FF4F93
:1003B0008491F901E458FF4F1491F901E057FF4F80
:1003C00004910023C9F0882321F069830E946A0107
:1003D0006981E02FF0E0EE0FFF1FEC55FF4FA59174
:1003E000B4919FB7F8948C91611103C0109581234B
:1003F00001C0812B8C939FBF0F90DF91CF911F91F4
:100400000F91089508950E942F010E9402020E94F8
:1004100007000C0E0D0E00E9474002097E1F30E94D9
:0A0420000000F9CF0895F894FFCF13
:00000001FF
```

```
.ORG      0x0000
RJMP      main

main:
    LDI       r16, 0xFF
    OUT       0x04, r16

loop:
    SBI       0x05, 5
    RCALL     delay
    CBI       0x05, 5
    RCALL     delay
    RJMP      loop
delay:
    LDI  r16, 61

    outer_loop:
    LDI  r24, low(0)
    LDI  r25, high(0)

    delay_loop:
    ADIW      r24, 1
    BRNE      delay_loop
    DEC       r16
    BRNE      outer_loop

    RET
```