CLASS 3

DIGITAL I/O, ANALOG OUTPUT

PLAN FOR CLASS 3

Review

- Blog review
- AMA, Class 2, partial quiz review
- Look ahead - plan for class 5 (Jeff out)

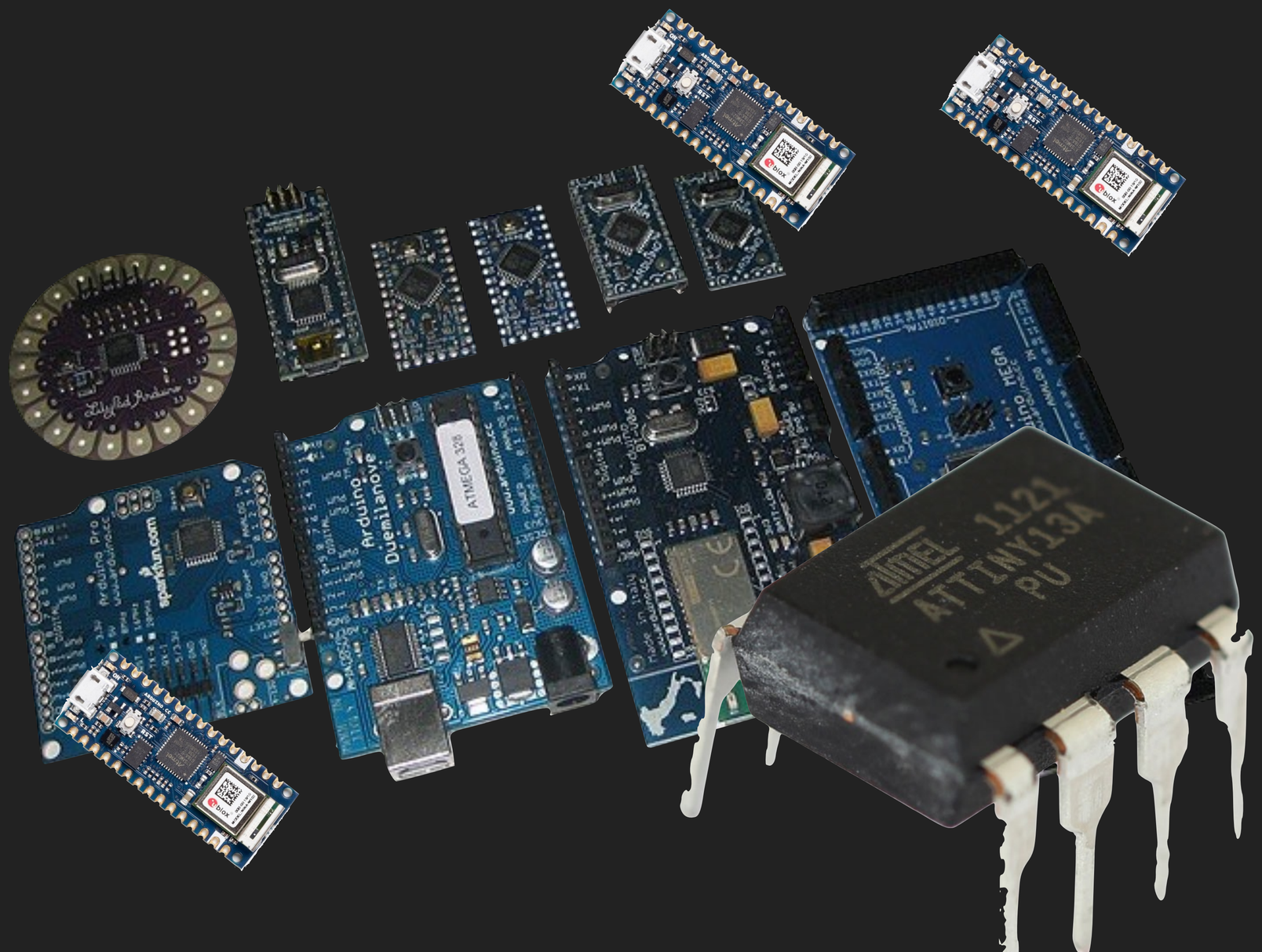
New

- Microcontrollers
- Sensors
- Programming terms and environment
- Digital Input and Output
- Analog Input

Bonus (if time) soldering demo

MICROCONTROLLERS CAN BE LOTS OF DIFFERENT THINGS

(Reminder)



SENSORS

convert something in the world (smell, light, mass, motion, etc.) into something the microcontroller can read

- Voltage (usually)
  - digital = two states (1-bit) above or below a threshold
  - analog = many states (2+ bits) mapped to many levels
- Digital data (covered later)

PROGRAMMING TERMS AND ENVIRONMENT

An **IDE (Integrated Development Environment)** combines everything you need:

- text editor, compiler, libraries, uploader
- Arduino IDE has tools for specifying board, adding libraries, finding examples

Programming

- C / C++
- **typed** language (a big difference from JS)
- generally, since we're "closer" to the machine, we need to be aware a bit more how it works (e.g. bits and bytes)

LEARNING ~~A~~ Any LANGUAGE

Learn the **syntax** and **reserved words**

- case matters, semicolons matter, etc.

Learn how **data is handled**

- Variables, types

Learn how to **organize code**

- Functions (objects)
- Order of operations

Understand “scope”

Learn **flow control**

- `for, if/then, while`

Learn operators

- = , ==, ! , && , ||

LEARNING A Any LANGUAGE

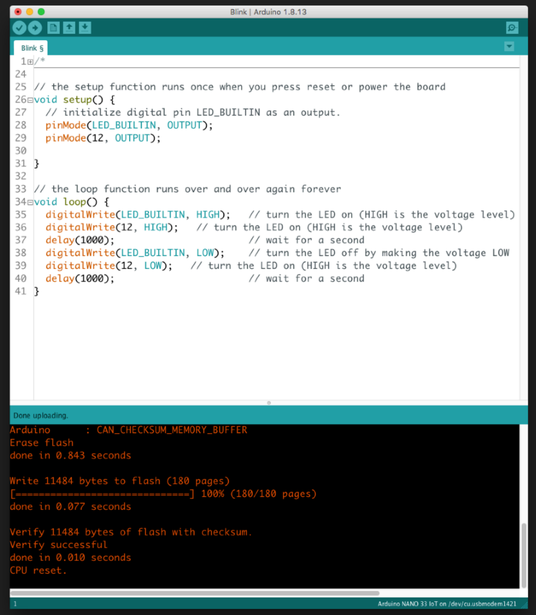
Type code in!!!

You'll learn faster than cut-n-paste

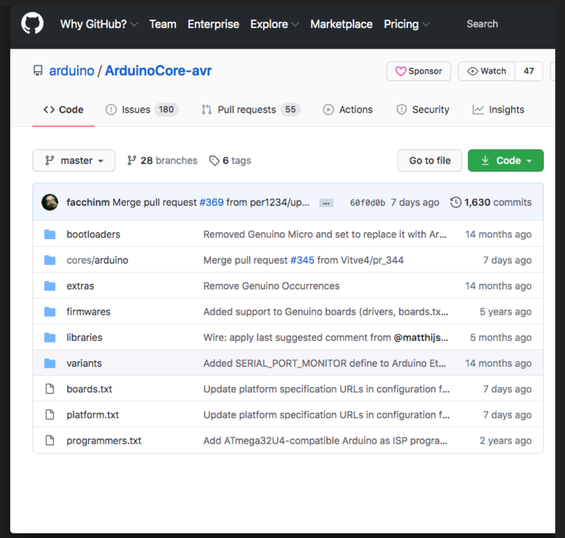
PCOMP ENVIRONMENT

Most development environments will have similar elements

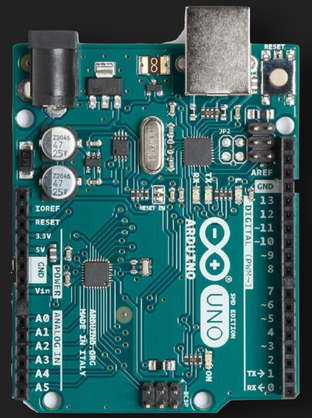
I.D.E.

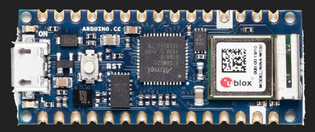


LIBRARIES

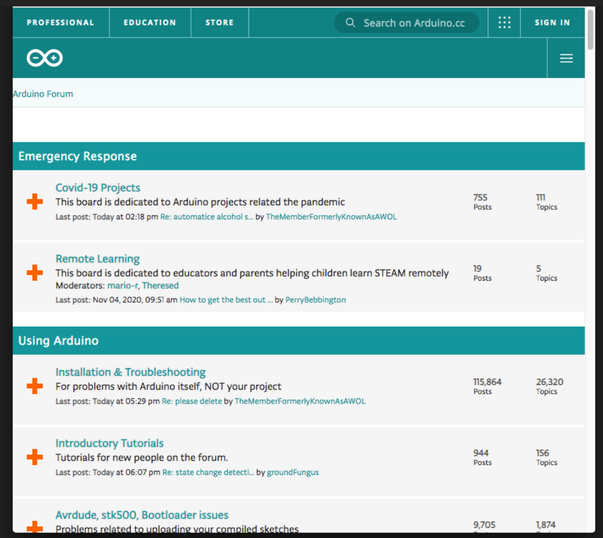


HARDWARE





SUPPORT



IDE

Compile + upload to board

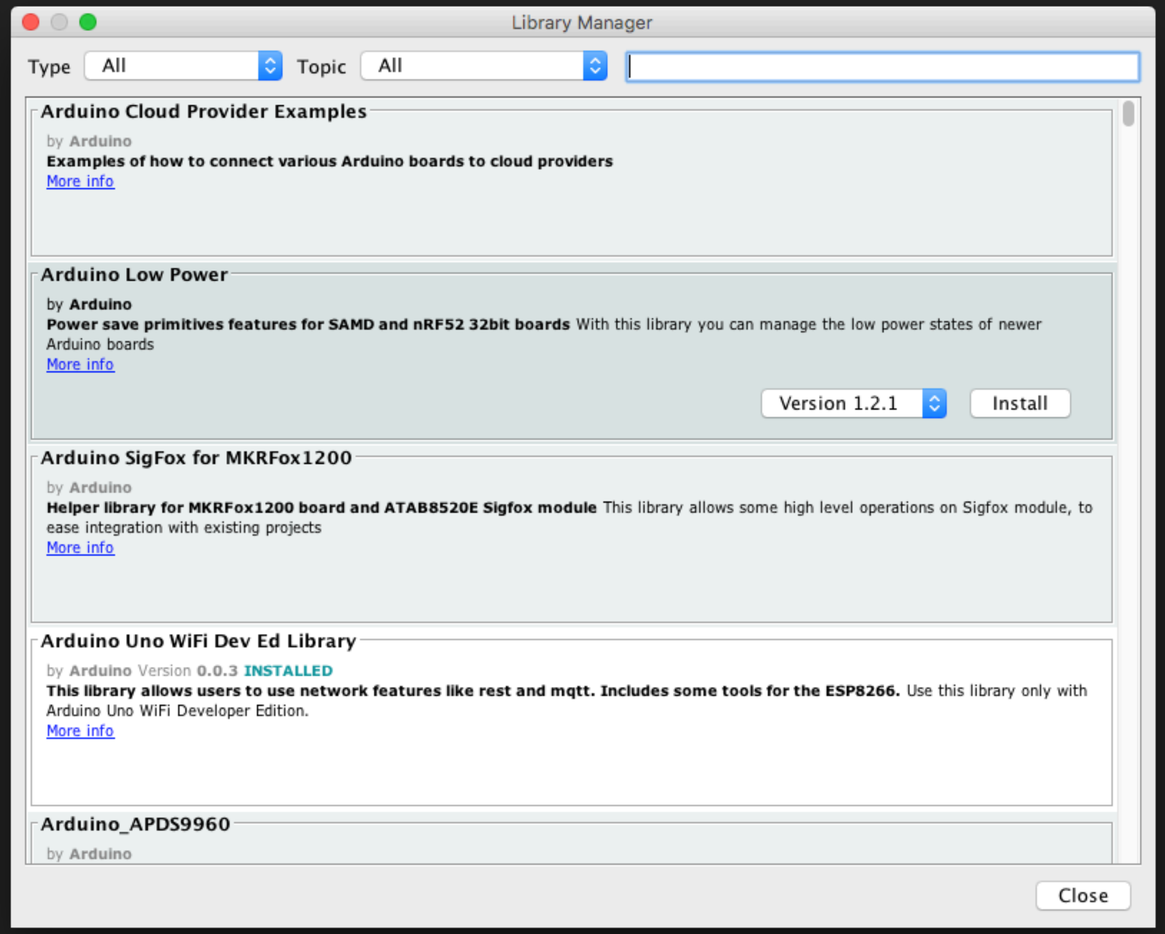
Code

Stats and status

Integrated Development Environment links your code with core libraries, compiler, and uploading tool chains.

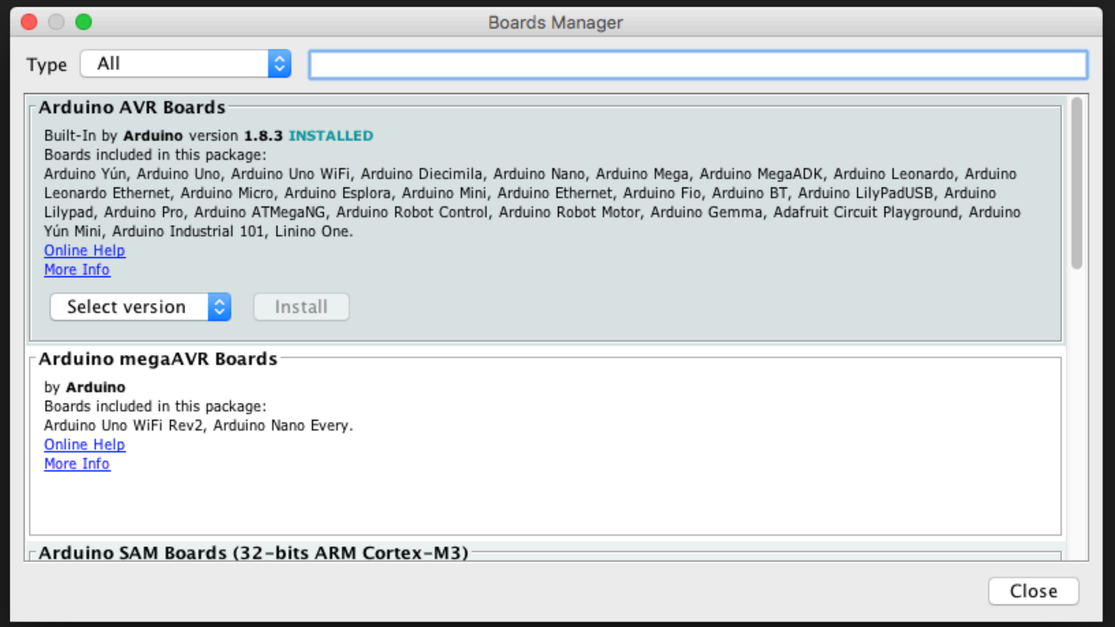
IDE

Library Manager*



Install software modules for additional functionality

Board Manager*



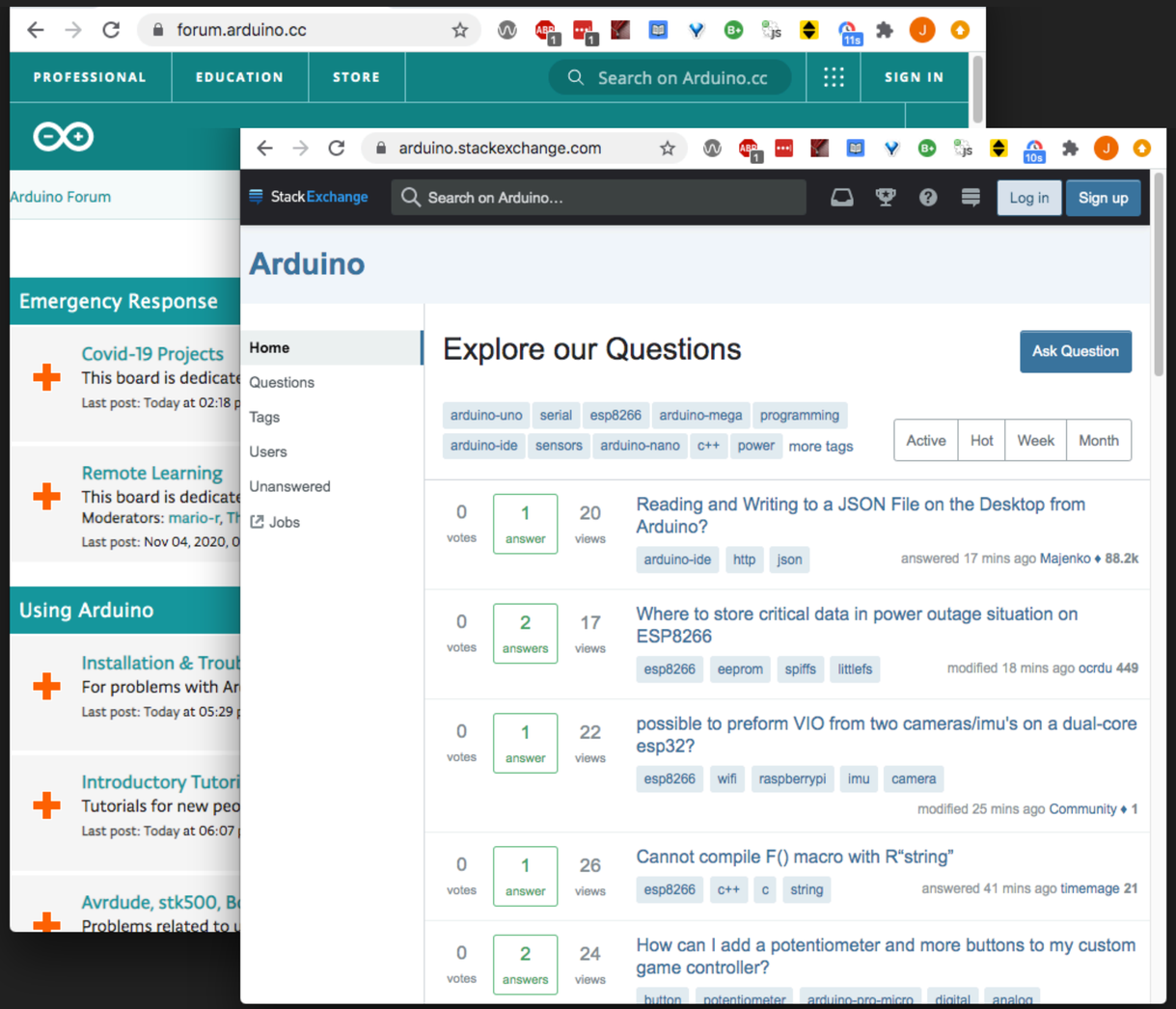
* IN SIDEBAR FOR ARDUINO IDE 2.0

Install hardware definitions to talk to other microcontrollers

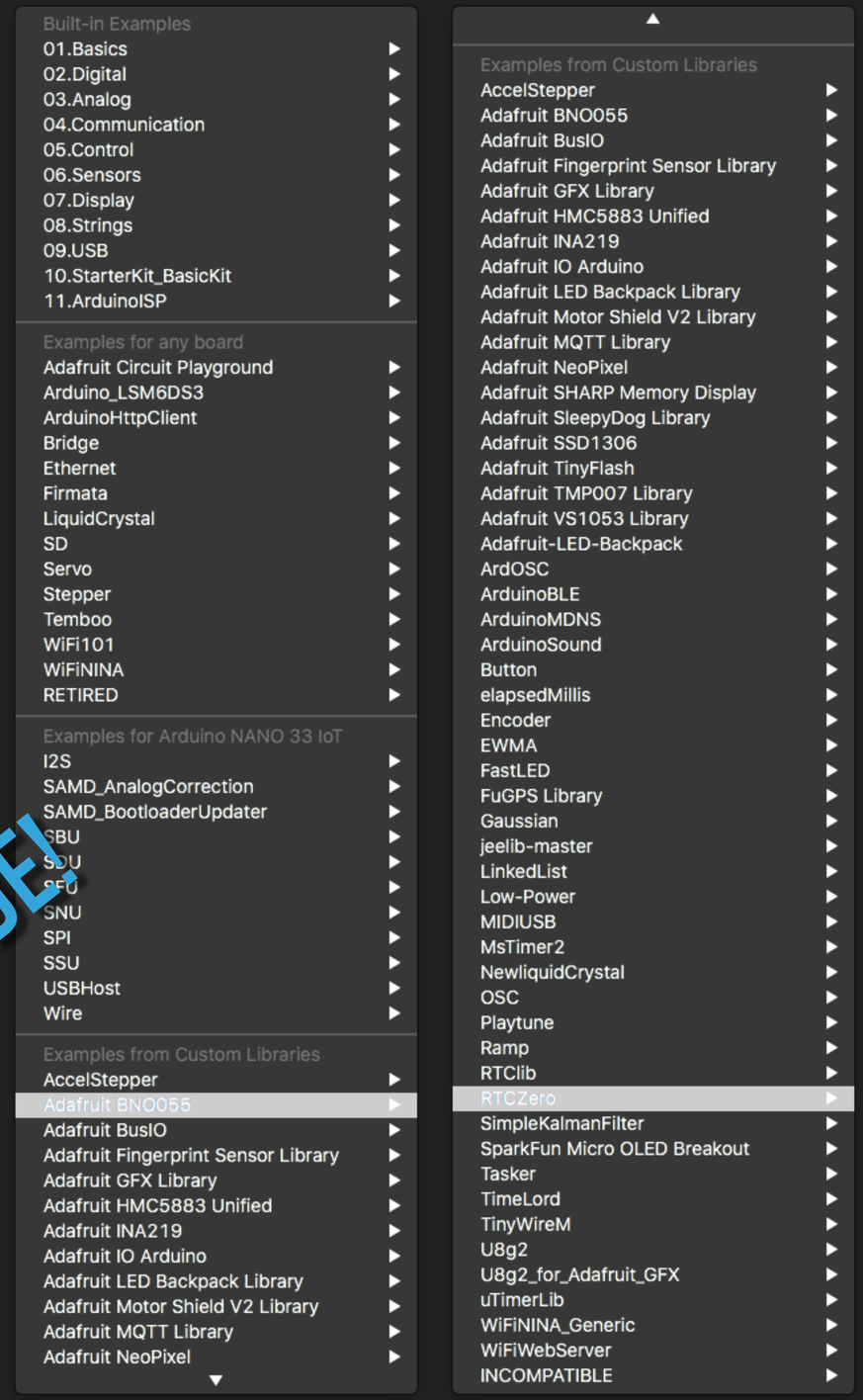
SUPPORT

In addition to PCOMP syllabus

Forums (lots!)



Examples (lots!)



IN IDE!

SUPPORT

New 2.0 IDE adds:

- Code completion (Yay!)
- In-circuit debugging (to be explored…)
- More serial plotting features

BINARY

~~BINARY~~ DECIMAL

Review

"Place-value" number systems

You have some number of symbols (e.g. '0'-'9')

You assemble those symbols to represent a value. The place of the symbol determines its contribution to the total

"Thousands" place — 5723 — "Ones" place

"Hundreds" place

"Tens" place

=

$5 * 1000 + 7 * 100 + 2 * 10 + 3 * 1$

=

$5 * 10^3 + 7 * 10^2 + 2 * 10^1 + 3 * 10^0$

In other words, the "place value" of the symbol is:

the number of possible symbols raised to the power of it's place in the string of symbols

BINARY

1 COIN, 2 STATES:

Heads

Tails

BINARY

2 COIN, 4 STATES:

HH

HT

TH

TT

BINARY

Each additional coin doubles the number of possible states.

With 3 coins there are 8 states:

Previous states, plus Heads

Previous states, plus Tails

BINARY

Put another way, the number of states is:

$$2^{\text{number of coins}}$$

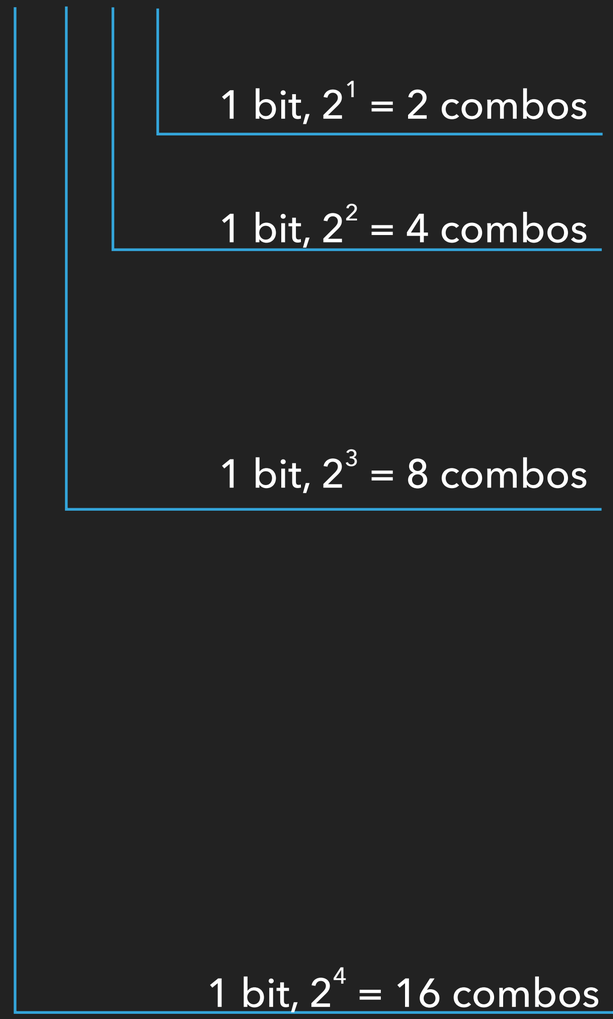
Previous states, plus Heads

Previous states, plus Tails

BINARY

Instead of coins, computers use bits, but the idea is the same.

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



BINARY

Microcontrollers like the Nano often have 8-bit PWM output resolution*:

$$8 \text{ bits}, 2^8 = 256 \text{ combos} = [0 \ldots 255]$$

…and 10-bit analog input resolution*:

$$10 \text{ bits}, 2^{10} = 1024 \text{ combos} = [0 \ldots 1023]$$

* by default

BOOLEAN / BINARY LOGIC

1 == HIGH == TRUE

0 == LOW == FALSE

HOW A MICROCONTROLLER TOUCHES THE WORLD