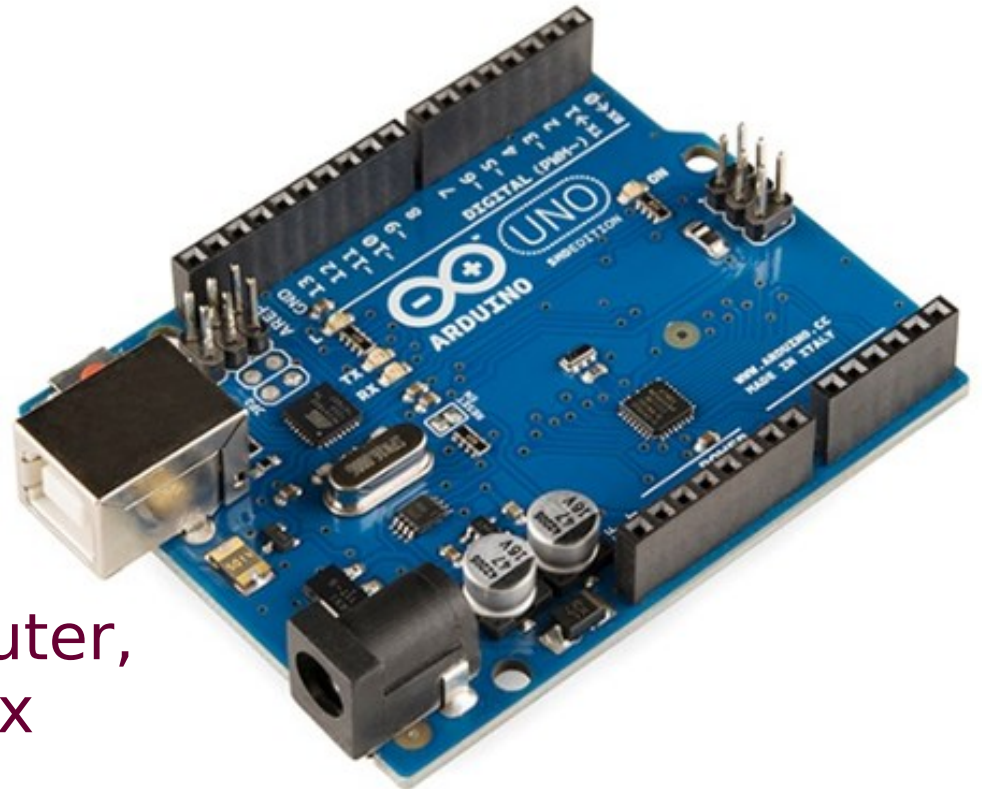


Arduino

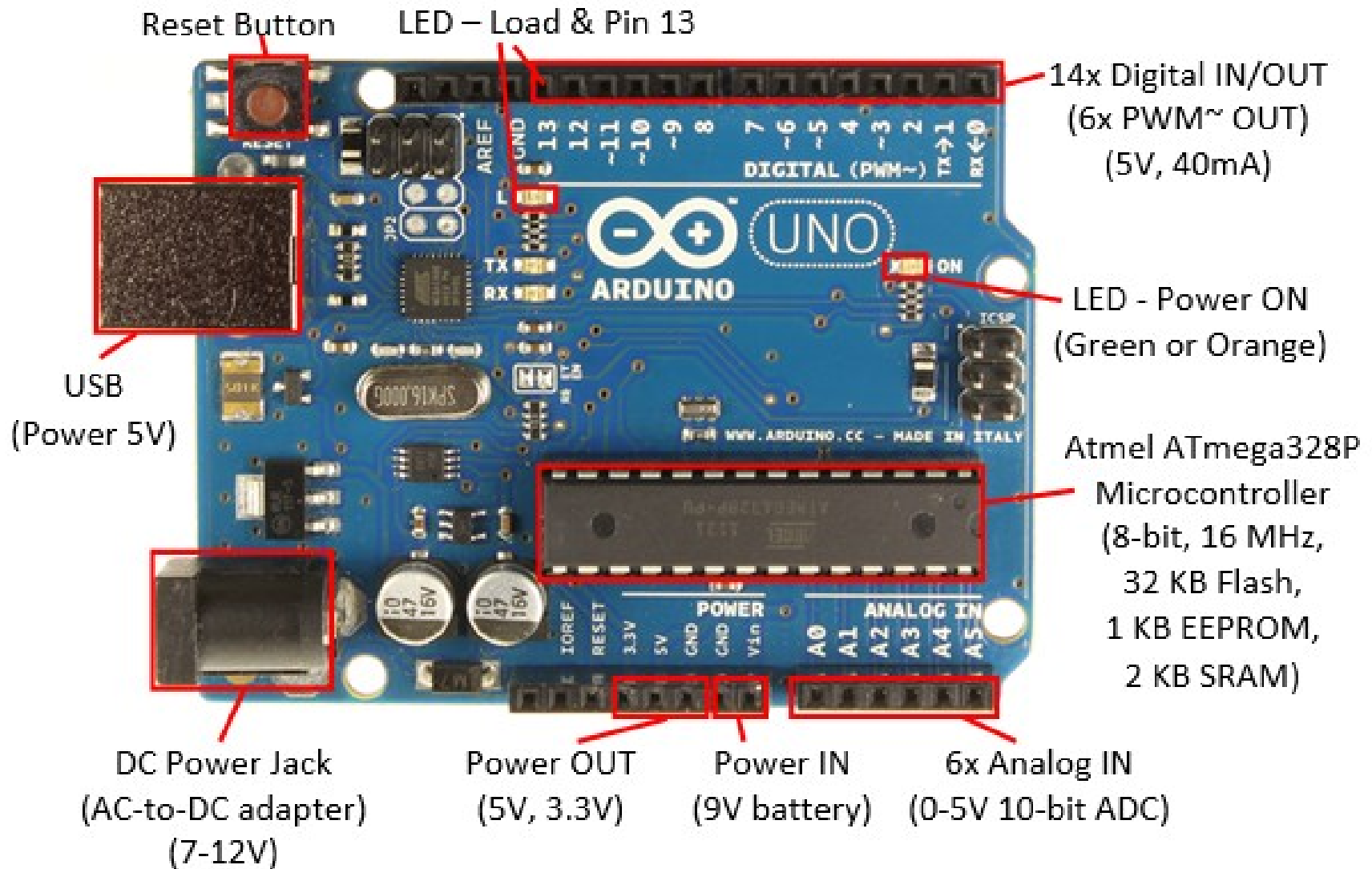
A short look at the Arduino Uno

What is Arduino?

- A microcontroller board
 - ATmega328P is an 8-bit microcontroller CPU
 - 32KBytes of Flash memory, 2KBytes of RAM
 - 14 digital I/O pins
 - 6 analog Input pins
- Not a computer
 - Raspberry Pi is a computer, running a flavor of Linux or Windows



What it looks like...



Ramped Blinking

- Nested loop blinks once, then twice, then three times, then four times... up to fifteen times
 - one blink is 100ms on, 100ms off
 - ½ second between groups of blinks
- After 15, the whole process repeats

```
// Simple blink demo

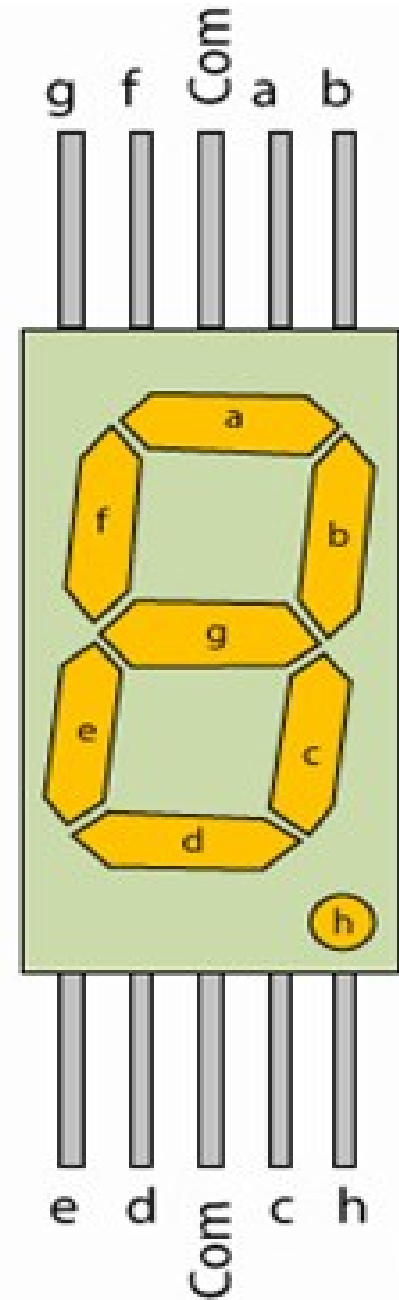
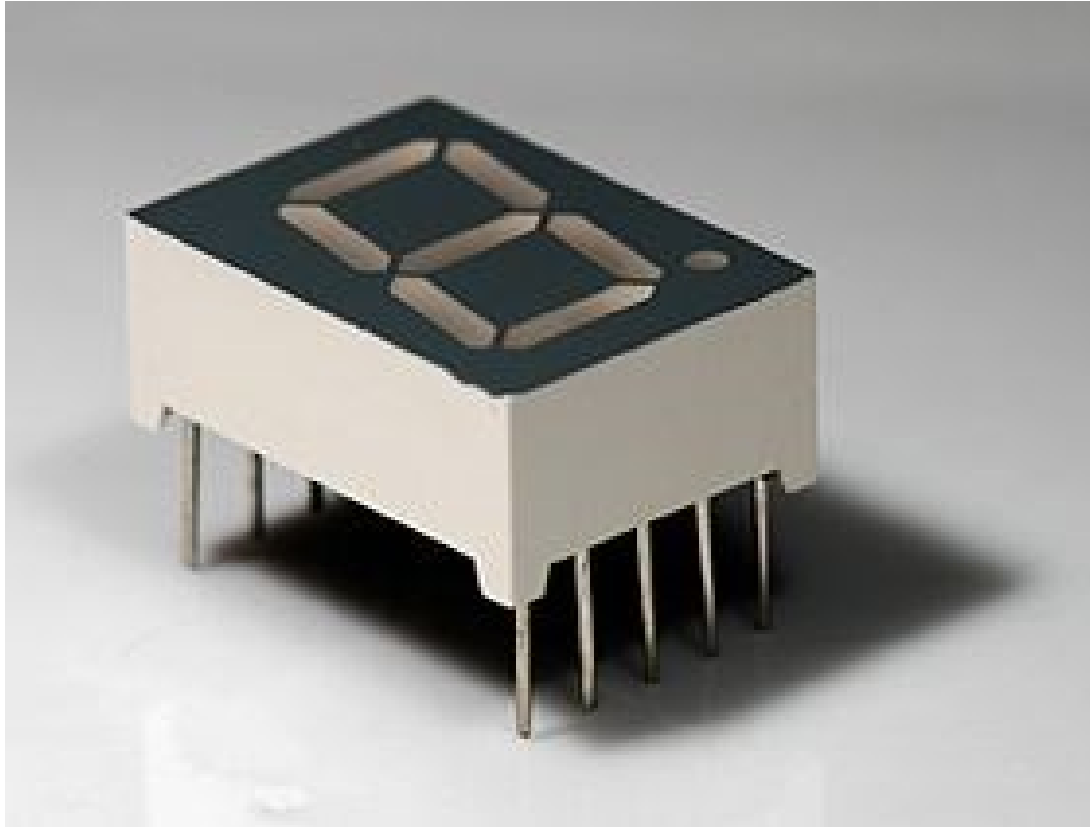
#define FAST 100
#define SLOW 500

int led = 13;

void setup() {
  pinMode(led, OUTPUT);
}

void loop() {
  unsigned i, j;
  for (i = 1; i < 15; i++) {
    for (j = 1; j <= i; j++) {
      digitalWrite(led, HIGH);
      delay(FAST);
      digitalWrite(led, LOW);
      delay(FAST);
    }
    delay(SLOW);
  }
}
```

The 7-segment display




```

// Driver for 7-segment display
// 2014-07-16 -bob,mon.
/*
  Compiled Storage:
  2732 no routines
  3190 sequentialSegments()
  3638 sequentialDigits()
  3622 sequentialBits()
  3784 sequentialBits(), bit extractions done inline instead of beforehand
*/

//Output-pins mapped to display segments a,b,c,d,e,f,g,h :
const int displaySegment[] = {3,2,5,10,11,8,9,4};

void setup() {
  int i;
  Serial.begin(9600);
  for (i = 0; i < 8; i++) {
    pinMode(displaySegment[i], OUTPUT);
  }
}

#define segA (!b2 & !b0) | (!b3 & b1) | (b2 & b1) | (!b3 & b2 & b0) | (b3 & !b2 & !b1)
#define segB (!b3 & !b2) | (!b2 & !b0) | (!b3 & !b1 & !b0) | (!b3 & b1 & b0) | (b3 & !b1 & b0)
#define segC (b3 & !b2) | (!b3 & b2) | (!b1 & b0) | (!b3 & !b1) | (!b3 & b0)
#define segD (!b2 & !b1 & !b0) | (!b2 & b1 & b0) | (!b3 & b1 & !b0) | (b2 & !b1 & b0) | (b2 & b1 & !b0) |
(b3 & b2 & !b1)
#define segE (!b2 & !b0) | (b1 & !b0) | (b3 & b2) | (b3 & b1)
#define segF (!b3 & !b1 & !b0) | (!b3 & b2 & !b1) | (b2 & b1 & !b0) | (b3 & b1) | (b3 & !b2)
#define segG (!b2 & b1 & b0) | (b1 & !b0) | (b2 & !b1) | (b3)
#define segH LOW

void sequentialCount(void) {
  int i;
  bool b0, b1, b2, b3;
  //for (i = 15; i >= 0; i--){
  for (i = 0; i < 16; i++) {

    b0 = (i & 0x01); // These instructions result in
    b1 = ((i & 0x02)>>1); // slightly smaller code than bitRead().
    b2 = ((i & 0x04)>>2);
    b3 = ((i & 0x08)>>3);
  }
}

```

```

b1 = ((i & 0x02)>>1); // slightly smaller code than bitRead().
b2 = ((i & 0x04)>>2);
b3 = ((i & 0x08)>>3);
/*
b0 = bitRead(i, 0);
b1 = bitRead(i, 1);
b2 = bitRead(i, 2);
b3 = bitRead(i, 3);
*/
Serial.print(b3);Serial.print(b2);Serial.print(b1);Serial.println(b0);

digitalWrite(displaySegment[0], segA );
digitalWrite(displaySegment[1], segB );
digitalWrite(displaySegment[2], segC );
digitalWrite(displaySegment[3], segD );
digitalWrite(displaySegment[4], segE );
digitalWrite(displaySegment[5], segF );
digitalWrite(displaySegment[6], segG );
digitalWrite(displaySegment[7], segH );
delay(489);
digitalWrite(displaySegment[7], HIGH);
delay(10);
}
for (i = 0; i < 7; i++)
  digitalWrite(displaySegment[i], LOW);
digitalWrite(displaySegment[7], HIGH);
//delay(1000);
}

void loop() {
  Serial.print("starting at ");
  Serial.println(millis());

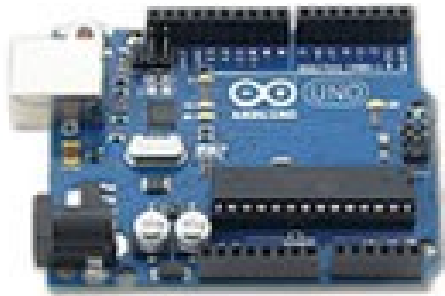
  sequentialCount();

  Serial.print("? ");
  while (!Serial.available())
    {}
  byte junk = Serial.read();
  Serial.print("Got ==>");
  Serial.println(junk);
}

```


A simple robot

The Arduino
brain...



...plugs into the
BOE Shield...



...which mounts on the robot chassis,
to make a BOE Shield-Bot