

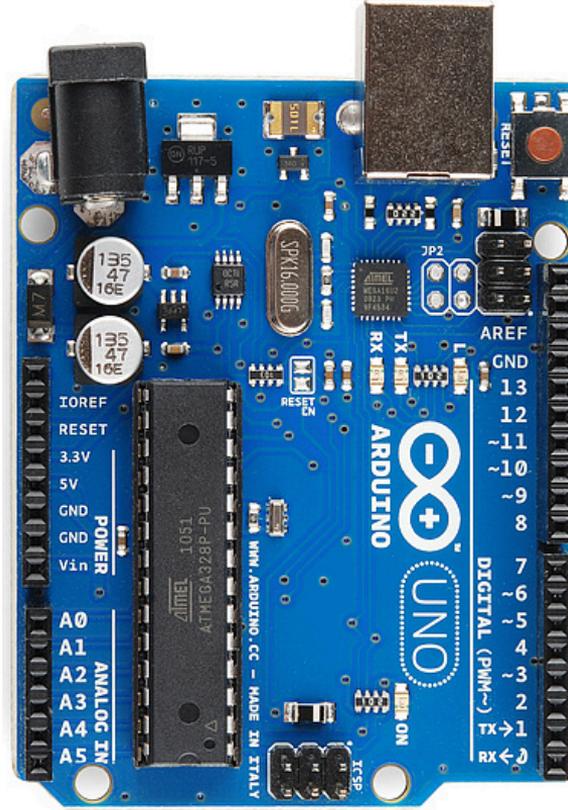
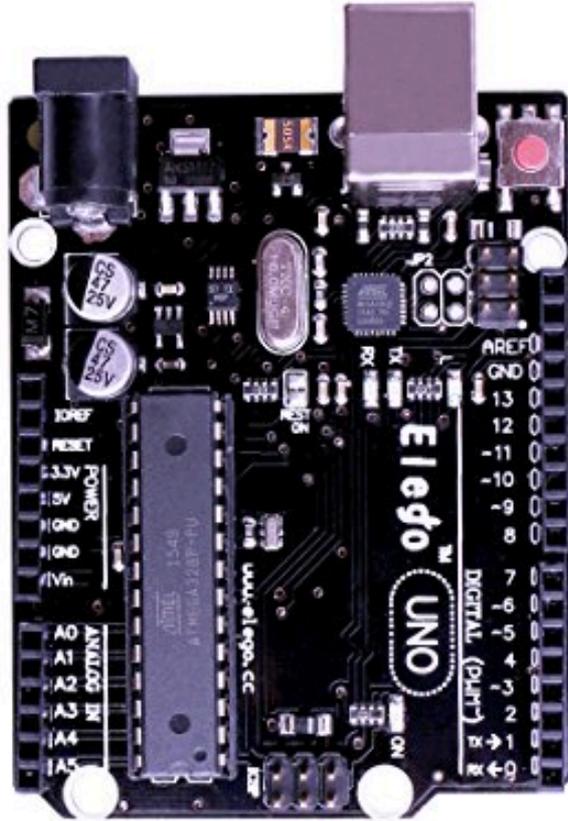
# Arduino Programming - An Introduction

---

EE 189L (Space Vehicles LLC)

September 13, 2016

# Elegoo and Arduino UNO



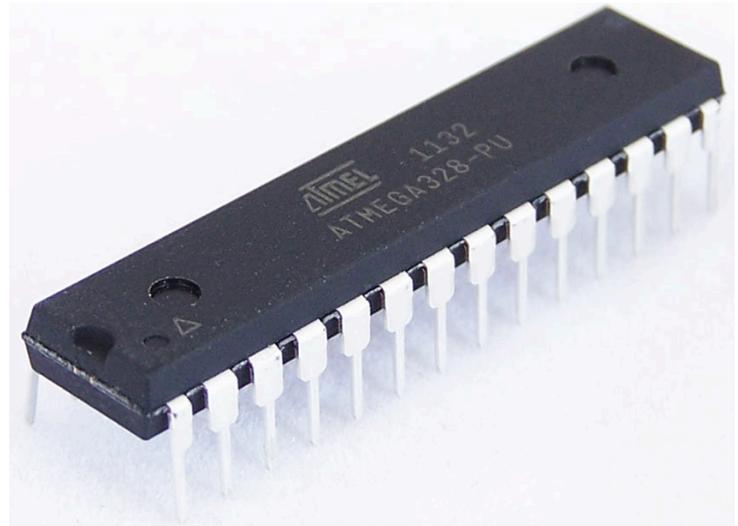
# ATmega328/P 28-pin Microcontroller

14 Digital I/O Pins (6 PWM outputs)

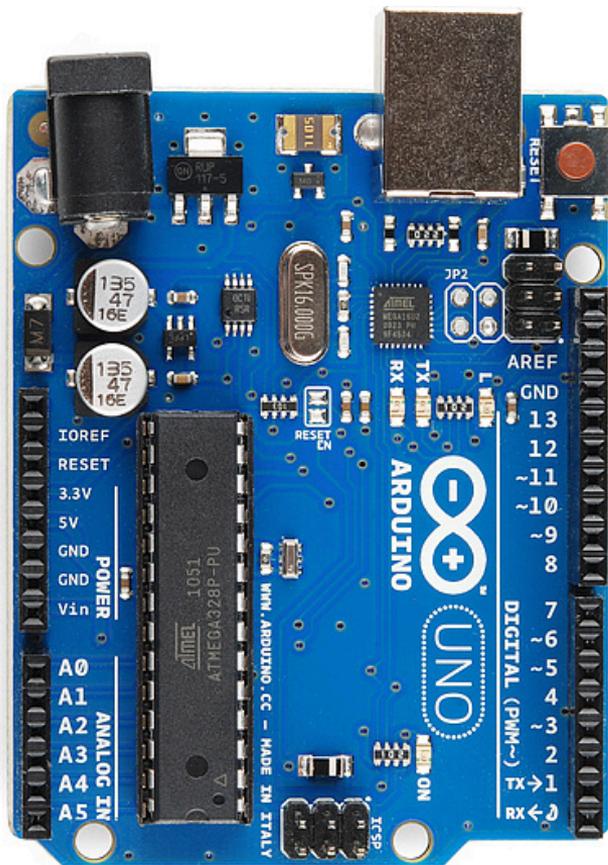
8 Analog Inputs

32kB Flash Memory

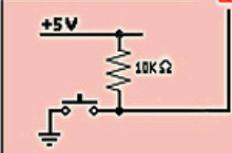
16Mhz Clock Speed



# ATmega328/P 28-pin Microcontroller

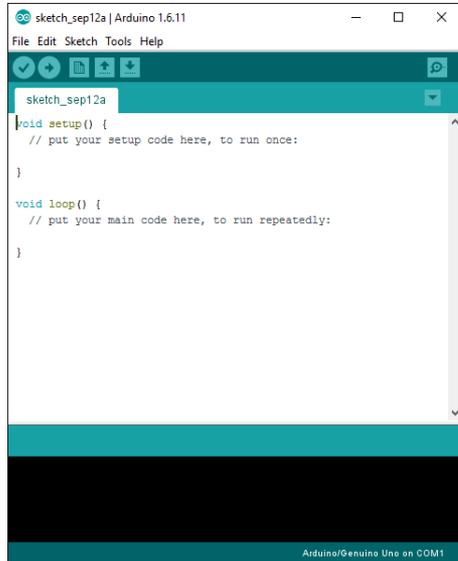


Port Information	"DDRC" - Direction Register "PORTC" - Output Register "PINC" - Input Register MSB A5 A4 A3 A2 A1 A0 LSB B00111111 PORT C	"DDRB" - Direction Register "PORTB" - Output Register "PINB" - Input Register MSB 13 12 11 10 9 8 LSB B00111111 PORT B												
Port Type	ANALOG						DIGITAL							
Feature	A/D 10 B:1	A/D 10 B:1	A/D 10 B:1	A/D 10 B:1	A/D 10 B:1	A/D 10 B:1	GND	AREF	AVCC	SCK	MISO	PWM	PWM	PWM
Arduino Pin	A5	A4	A3	A2	A1	A0				13	12	11	10	9
Pin Number	28	27	26	25	24	23	22	21	20	19	18	17	16	15
PKG. DIP 28														
ATMEGA 168-20P 16KiB Flash 512B EEPROM 1024B RAM	1	2	3	4	5	6	7	8	9	10	11	12	13	14
ATMEGA 328-20P 32KiB Flash 1024B EEPROM 2048B RAM	RESET	0 RX	1 TX	2 FWM	3	4	VCC	GND	XTAL OSC	10	5 PWM	6 PWM	7	8
		DIGITAL							0-20MHZ		DIGITAL			
		b <sup>0</sup>	b <sup>1</sup>	b <sup>2</sup>	b <sup>3</sup>	b <sup>4</sup>	+5V				b <sup>5</sup>	b <sup>6</sup>	b <sup>7</sup>	b <sup>0</sup>
		"DDR D" - Direction Register "PORT D" - Output Register "PIND" - Input Register						MSB 7 6 5 4 3 2 1 0 LSB B11111111 PORT D						

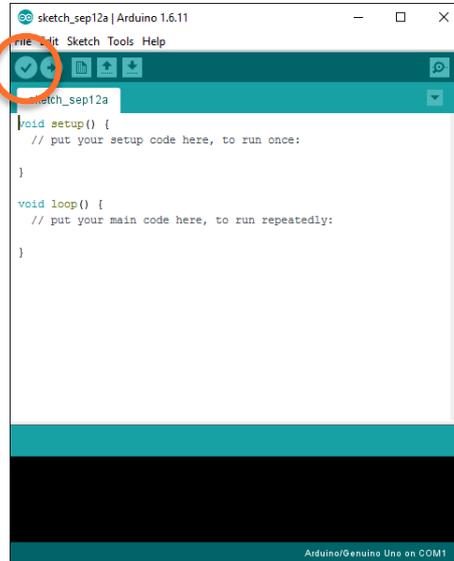


# Programming the UNO via Arduino IDE (Sketch)

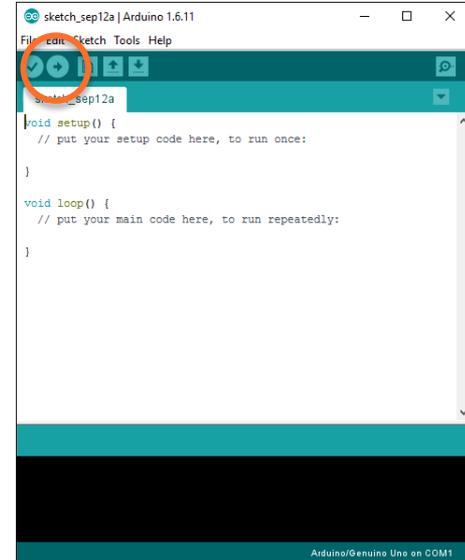
Editor – write program



Verify – check for errors



Upload – compile & send



- ✔ Verify - Checks your code for errors compiling it.
- ➔ Upload - Compiles your code and uploads it to the configured board (UNO)



# Arduino IDE

Remember to check port (COM3?) and board (Uno) in setup

Commands end with semicolon; function declarations and definitions do not

# Program Structure

```
// Remember to use comments, tabs and spaces for readability and  
// references
```

```
void setup() // commands in here executed once at start  
{  
    // tab commands within functions, loops, etc.  
}
```

```
void loop() // commands in here executed continuously (top to bottom)  
{  
    // tab commands within functions, loops, etc.  
}
```

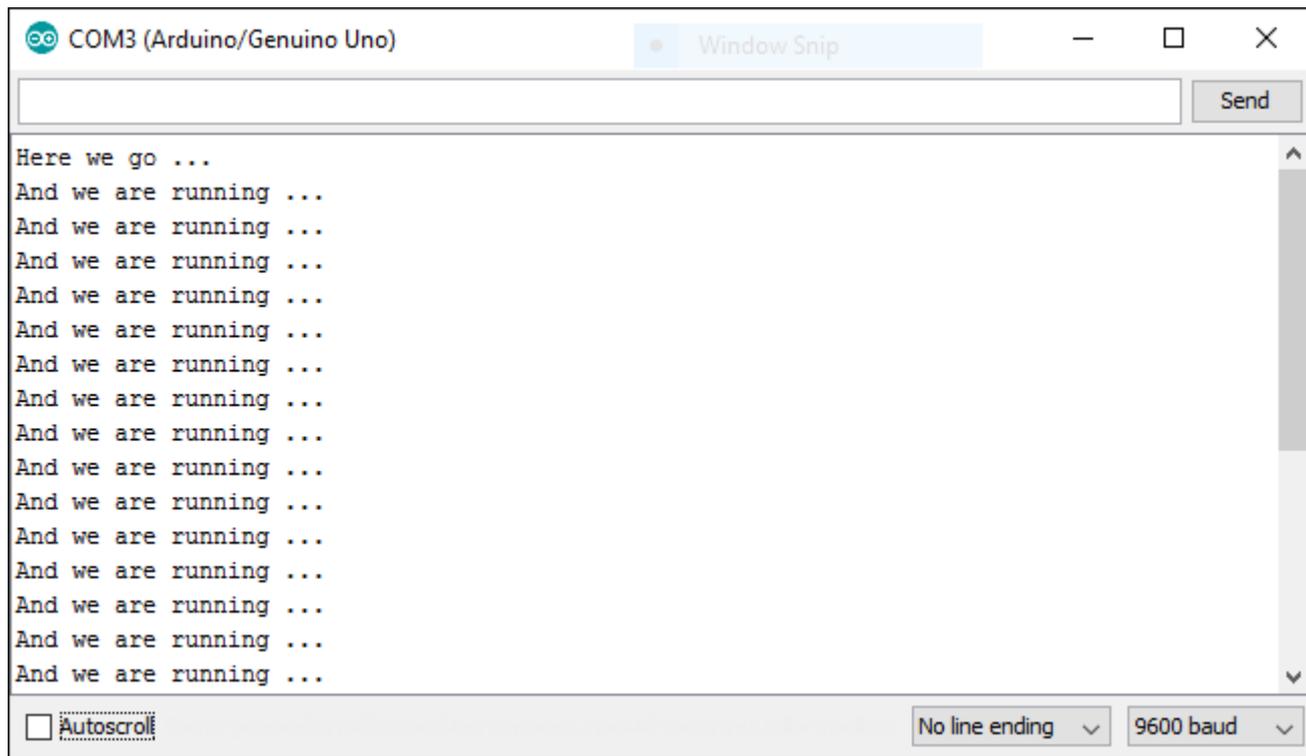
# Program Structure - display message

```
// Demonstrate Structure of Arduino Program and Serial Communication
// K. Wedeward, 09/11/2015
// Two functions (setup and loop) are required

void setup() // commands in here executed once at start
{
    Serial.begin(9600); // Set data rate for communication
                        // at 9600 bits per second
    Serial.println("Here we go ..."); // display a message once
}

void loop() // commands in here executed continuously (top to bottom)
{
    Serial.println("And we are running ..."); // display message
    delay(1000); // wait 1000ms = 1s before continuing
}
```

# Program Structure – display messages



# Digital Output (digitalWrite)

```
// Program to demonstrate digital output (write)

void setup() // commands in here executed once at start
{
    pinMode(9, OUTPUT); // Set digital pin 9 to an output
}

void loop() // commands in here executed continuously (top to bottom)
{
    digitalWrite(9, LOW); // set pin 9 LOW (LED off)
    delay(1000); // wait 1000ms (1 sec) with LED off
    digitalWrite(9, HIGH); // set pin 9 HIGH (LED on)
    delay(1000); // wait 1000ms (1 sec) with LED on
}
```

# PWM Digital Output (analogWrite on ~)

```
// Program to demonstrate PWM output (analogWrite)

void setup() // commands in here executed once at start
{
    pinMode(9, OUTPUT); // Set digital pin 9 to an output for PWM
}

void loop() // commands in here executed continuously (top to bottom)
{
    analogWrite(9, 5); // set pin 9 to on (5/255) x 100%
    delay(1000); // wait 1000ms (1 sec) with LED dim
    analogWrite(9, 255); // set pin 9 to on (255/255) x 100%
    delay(1000); // wait 1000ms (1 sec) with LED bright
}
```

# Today's goal

Write five programs to learn about programming, commands and oscilloscope

1. Example that blinks built-in LED on digital pin 13
2. Uses serial print and monitor to display a message once, and then another over and over
3. Turn an external LED (with resistor) on and off every second using digital port
  - use oscilloscope to measure accuracy of time specified by delay(1000)
4. Vary brightness of external LED (with resistor) using PWM
5. Program something unique (heartbeat pattern, morse code, ...)
  - Use oscilloscope to measure duty cycle of PWM