

Arduino 101

Presented to PCR NMRA

Earl Girbovan & Seth Neumann

January – February 2018

What can we do to help you get started?

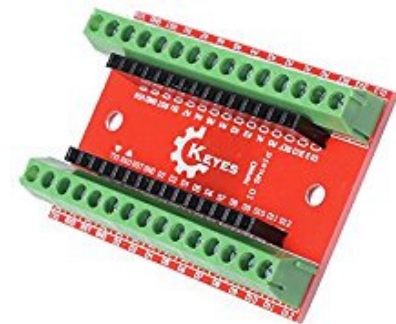
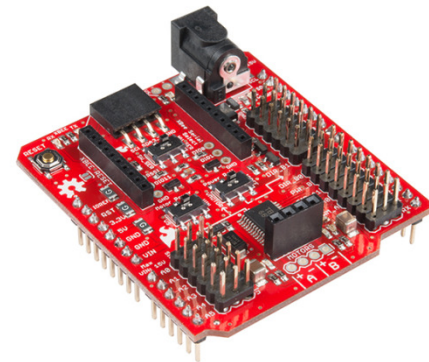
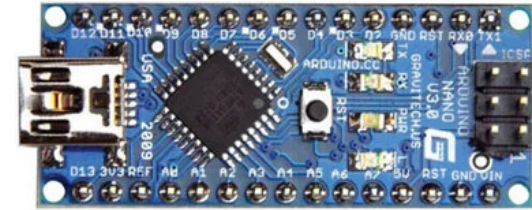
- Introduce the Arduino integrated development environment (IDE)
- Show you how to select the proper board and COM port, and download a program
- Basic programming introduction
- Provide sample programs you can reuse or build upon
- Minimal theory, maximum hands-on

What is Arduino?

- Programmable controller
- Analog and Digital I/O
- Program written in simple language, compiled, and downloaded into the Arduino
- Arduino board then runs the program, computer connection no longer needed
- Don't worry, Arduino was designed in Italy for use by kinetic artists. If they can do it, so can you!

Hardware

- Many boards out there, basic functionality the same
- Terminal blocks and headers allow expansion and connections
- Boards can be powered from a 9V wall wart or stepped down 12V layout power

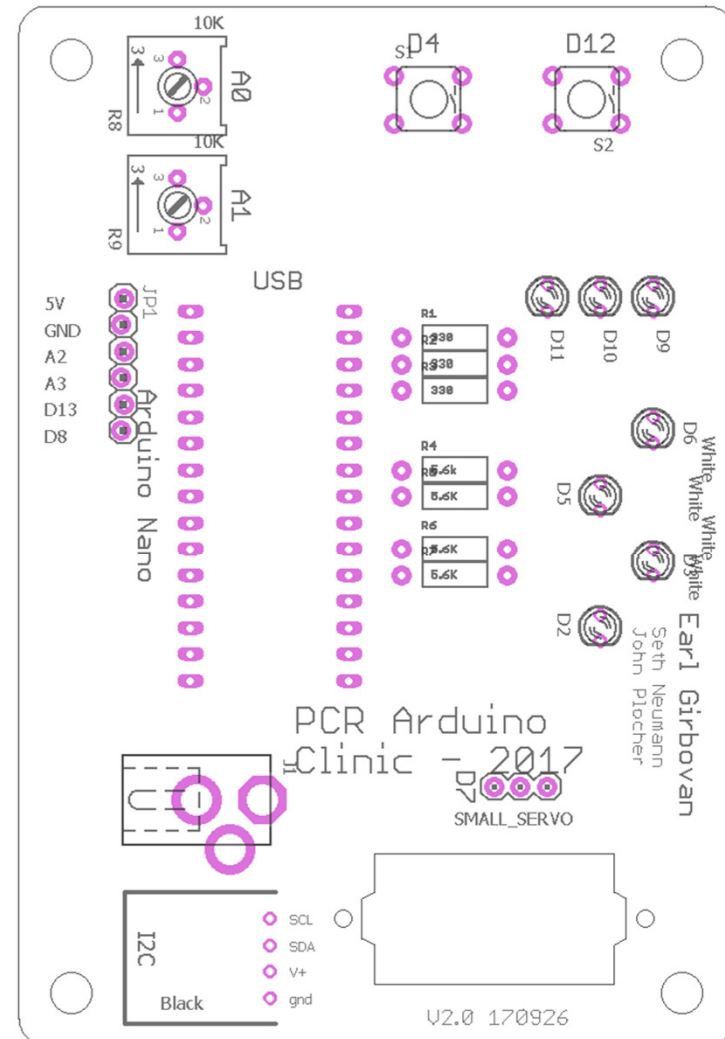


Where to Get It All

- Amazon
- Chinese suppliers like Banggood.com
- eBay sellers
- Jameco in Belmont
- DigiKey, Newark, Mouser
- Modern Devices
- Sparkfun
- Adafruit

Clinic Circuit Board

- Arduino Nano based PCB
- Nano has:
 - 13 Digital I/Os
 - 8 Analog inputs
- Clinic PCB adds an assortment of LEDs, pots, pushbuttons and a servomotor
- Jack for power
- Unused pins brought out to a terminal block
- Schematic is in your workbook
- Most components can be interfaced directly to the Arduino, just need a current limiting resistor for the LEDs



Writing Software

- Often no need to write from scratch. Find a program that's similar and modify it. Many programs available on the web.
- [instructables.com](https://www.instructables.com)
- Compile often as you write, it's easier to find mistakes
- Test code in small modules
- Comment your code

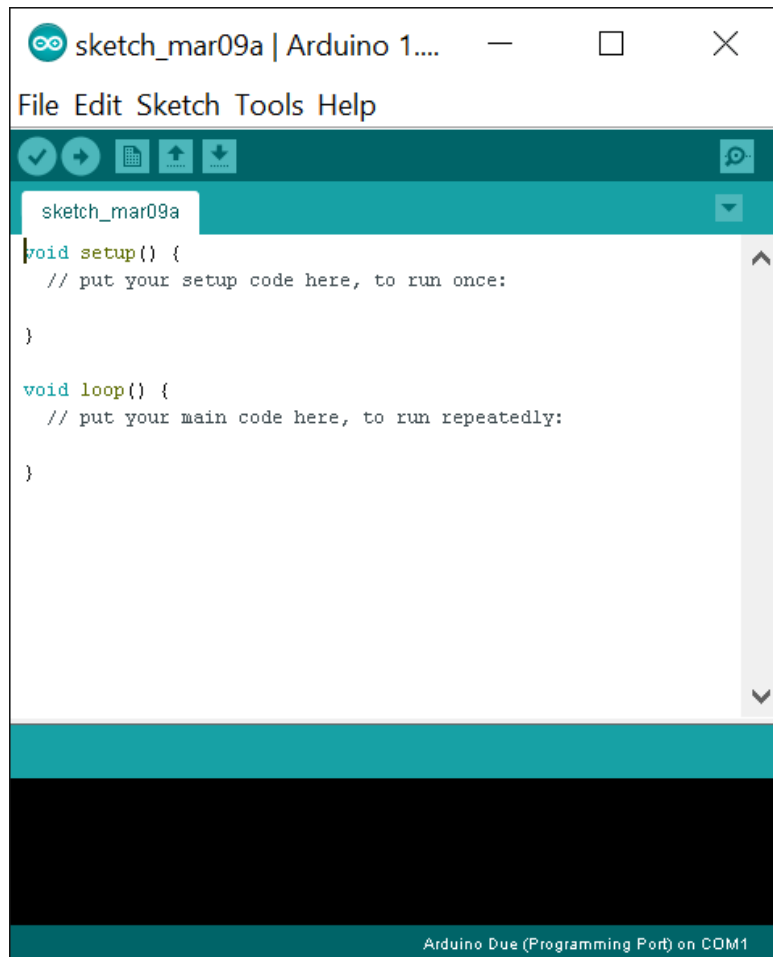
Syntax & The Compiler

This isn't Burger King, you can't have it your way

- Syntax – the 'grammar' the compiler expects
- Most lines terminate with a semicolon
- Variables need to be defined prior to use
- Capitalization and punctuation matter a lot!
- Compiler error messages don't always point at the error. Could be much earlier in the code.
- Make sure your brackets are balanced {}
- Did I mention comment your code?

Arduino IDE

Integrated Development Environment



Most Commonly Used Functions:

- **Sketch – Verify/Compile**, allows you to check your work as you go
- **Tools – Board**, selects the proper Arduino for the compiler. For this clinic, Nano w/ATmega328.
- **Tools – COM Port**, port the Arduino is connected to for downloading code
- **File – Upload**, compiles and uploads your code to the board. The right arrow key in the bar below it performs the same function.

Sample Code – Blink an LED

```
/* Blink
```

```
This program blinks an LED on and off
```

```
In the TOOLS BOARD tab in the Arduino IDE, ensure 'Arduino Nano w/ATmega328' is selected
```

```
*/
```

```
int LED2 = 2; //D2 output port
```

Goes to the LED on your board labeled D2

```
void setup()
```

```
{
```

```
  pinMode( LED2, OUTPUT );
```

```
}
```

Makes pin 2 an output

```
void loop()
```

```
{
```

```
  digitalWrite( LED2, HIGH );
```

```
  delay( 1000);
```

```
  digitalWrite( LED2, LOW );
```

```
  delay ( 1000 );
```

```
}
```

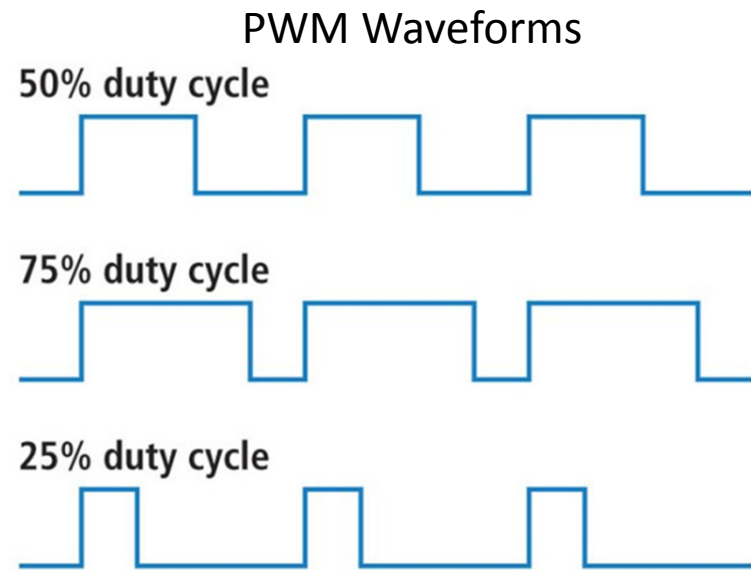
Turn LED2 off (HIGH)

Turn LED2 on (LOW)

PWM & Random (x,y)

- PWM = Pulse Width Modulation

- Varying duty cycle, can be used to vary LED brightness
- Also used to drive servomotor



- Random function, returns a random number between min and max values

Fire Effect

- `/* LED Fire Effect`
- `Uses two amber and one red LED to simulate fire`
- `From the instructables.com web site, slightly modified`
- `*/`

- `int ledPin1 = 9;`
- `int ledPin2 = 10;`
- `int ledPin3 = 11;`

- `void setup()`
- `{`
- `pinMode(ledPin1, OUTPUT);`
- `pinMode(ledPin2, OUTPUT);`
- `pinMode(ledPin3, OUTPUT);`
- `}`

- `void loop()`
- `{`
- `analogWrite(ledPin1, random(135, 255));`
- `analogWrite(ledPin2, random(135, 255));`
- `analogWrite(ledPin3, random(135, 255));`
- `delay(random(1, 100));`
- `}`

Note analogWrite:
The number is
 $x/256$ so it's % of
full brightness

Delay in
Milliseconds

Servo Sweep

```
• /*  
• Servo Sweep  
  
• This program slowly sweeps a servo back and forth from 0 to a presettable maximum number of degrees.  
• */  
  
• #include <Servo.h>  
• Servo Servo1;  
  
• int pos;  
• int MaxPos = 60;  
• int DelayTime = 90;  
  
• void setup()  
• {  
• //tell the Arduino & the servo driver that the servo is connected to pin 7  
• Servo1.attach(7);  
• }  
  
• void loop()  
• {  
• //sweep the servo back and forth  
• for( pos = 1; pos <=MaxPos; pos++) //sweep from 0 to MaxPos in 1 degree steps  
• {  
• Servo1.write(pos);  
• delay(DelayTime);  
• }  
• delay( 1000 );  
•  
• //now sweep it back  
• for( pos = MaxPos; pos >=1; pos--) //sweep from MaxPos to 0 in 1 degree steps  
• {  
• Servo1.write(pos);  
• delay(DelayTime);  
• }  
• delay( 1000 );  
• }
```

Important Programming note:
Use of the servo library disables
analogWrite() (PWM)
functionality on pins 9 and 10,
whether or not there is a Servo
on those pins.

Summary

- Shown how to type in, compile and download Arduino code
- Introduced some of the built-in Arduino functions
- Provided sample programs
- Next Clinic, Arduino 102:
 - Working with inputs
 - Higher level software functions
 - Tying it together

Arduino 102

Presented to PCR NMRA

Earl Girbovan & Seth Neumann

January – February 2018

Arduino 102 Builds on 101

- More complicated programs
- More realistic effects, with random start and run times
- Interfacing with the real world: testing inputs and manipulating outputs (i/o)
- Introduce more Arduino built-in functions

Higher Level Software

- while loop

```
while( X comparison Y )  
  {  
    do something  
  }
```

- Comparison operators include:

> < >= <= == !=

– Note that '=' and '==' are different

- if / else statement

WHILE Loop: Turning the fire off and on

```
• /* LED Fire Effect
•   Uses two amber and one red LED to simulate fire
•   From the instructables.com web site
•
•   Included a WHILE loop, so that the effect turns on and off.
• */

• int ledPin1 = 9;
• int ledPin2 = 10;
• int ledPin3 = 11;
• int OnTime;
• int OffTime;
• int i;

• void setup()
• {
•   pinMode(ledPin1, OUTPUT);
•   pinMode(ledPin2, OUTPUT);
•   pinMode(ledPin3, OUTPUT);
• }

• void loop()
• {
•   OnTime = random( 1000, 5000 );
•   while( OnTime > 0 )
•   {
•     analogWrite(ledPin1, random(120)+135);
•     analogWrite(ledPin2, random(120)+135);
•     analogWrite(ledPin3, random(120)+135);
•     i = random(1, 100);
•     delay( i );
•     OnTime = OnTime - i;
•   }

•   // off time, ensure the LEDs are turned off
•   analogWrite(ledPin1, 255 );
•   analogWrite(ledPin2, 255 );
•   analogWrite(ledPin3, 255 );
•   OffTime = random ( 1000, 3000 );
•   delay ( OffTime );
• }
```

Gives you control over when it stops

Debugging Your Program or, Errors You're Going to Make

- Capitalization
- Spelling
- Inconsistent variable names
- Check the syntax: == is different from =
- Did you terminate the last item with “;” or is there one that doesn't belong?
- Are the {} balanced?

Tying it Together

Simplified Grade Crossing Control

- Combines the BLINK and SERVO SWEEP routines
- Need the while loop to run the two routines simultaneously
- Uses the pushbutton to actuate the sequence

Program to Read Inputs & Set Outputs

- Look at the pots on A0, A1 as a percentage of brightness or servo deflection
 - Read value
 - Adjust as required (re-scale or map to log value)
 - Write to LED, Servo

Hardware options for your projects

- Additional copies of project board
- Use a nano with connectors
- Pro-mini
- Uno
- Mega

Additional Resources

- Banggood sensor kit and output options
- Arduino Development Kits with breadboard
- Sample code on Arduini Yahoo Group, join!
- Google it! Lots of examples.
- Animation projects/videos on YouTube:
 - Geoff Bunza
 - Laurie McLean

Something to Consider

- Multi-function programs can get complicated quickly, and are time-intensive to write and debug.
- Arduinos are inexpensive
- Dedicate an Arduino to each function
- Hardest part then is wiring it all up, not writing and debugging code

Arduino 102 Summary

- Introduced more software input and output functions and provided sample programs
- Introduced more complex software commands
- Shown the most common programming errors and what to watch out for
- Hopefully, inspired you to use Arduinos somewhere on your layout