
Introducción a C++ y Arduino

Introducción a C++ con Qt

- Componentes de un programa:
 - Librerías.
 - Variables globales.
 - Variables locales.
 - Tipos de datos.
 - ...
- Estructuras de programación:
 - Secuenciales.
 - Selección.
 - Control.

Introducción a C++ con Qt

- Componentes de un programa:
 - Librerías.
 - Variables globales
 - Variables locales.
 - Tipos de datos.
 - ...
- Estructuras de programación:
 - Secuenciales.
 - Selección.
 - Control.
- Ejercicios de práctica:
 - Impresión de los múltiplos del 4 de forma ascendente.
 - Cómputo del promedio de notas del curso.
 - Cálculo del punto de intersección entre dos rectas.

ARDUINO



¿Por qué usar Arduino?

- **Barato:** La versión menos cara del modulo Arduino puede ser ensamblada a mano, e incluso los módulos de Arduino preensamblados cuestan menos de 50\$.
- **Multiplataforma:** El software de Arduino se ejecuta en sistemas operativos Windows, Macintosh OSX y GNU/Linux.
- **Entorno de programación simple y claro:** El entorno de programación de Arduino es fácil de usar para principiantes, pero suficientemente flexible para que usuarios avanzados puedan aprovecharlo también.
- **Código abierto y software extensible:** El software Arduino está publicado como herramientas de código abierto, disponible para extensión por programadores experimentados. El lenguaje puede ser expandido mediante librerías C++.
- **Código abierto y hardware extensible:** El Arduino está basado en microcontroladores ATMEGA8 y ATMEGA168 de Atmel. Los planos para los módulos están publicados bajo licencia Creative Commons, por lo que diseñadores experimentados de circuitos pueden hacer su propia versión del módulo, extendiéndolo y mejorándolo. Incluso usuarios relativamente inexpertos pueden construir la versión de la placa del módulo para entender como funciona y ahorrar dinero.

Resumen características técnicas

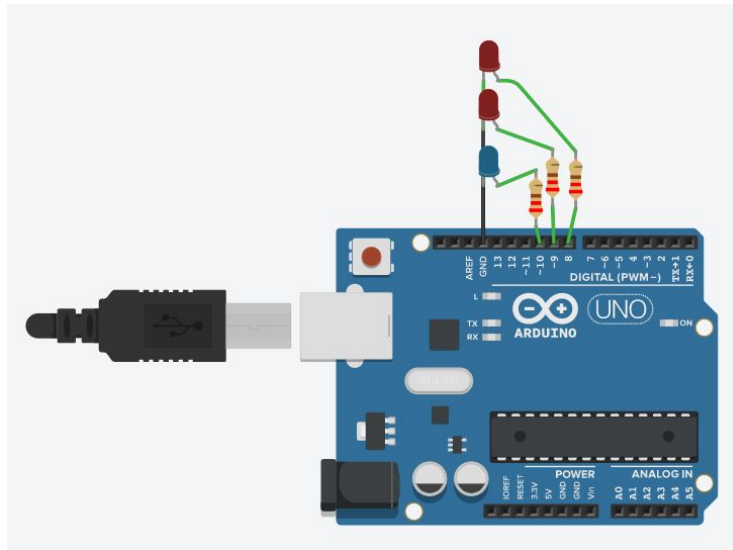
Microcontrolador	Atmega328
Voltaje de operación	5V
Voltaje de entrada (Recomendado)	7 – 12V
Voltaje de entrada (Límite)	6 – 20V
Pines para entrada- salida digital.	14 (6 pueden usarse como salida de PWM)
Pines de entrada analógica.	6
Corriente continua por pin IO	40 mA
Corriente continua en el pin 3.3V	50 mA
Memoria Flash	32 KB (0,5 KB ocupados por el bootloader)
SRAM	2 KB
EEPROM	1 KB
Frecuencia de reloj	16 MHz

Código de ejemplo Tinkercad

```
1 int contador = 0;
2 void setup()
3 {
4   pinMode(13, OUTPUT);
5   Serial.begin(9600);
6 }
7
8 void loop()
9 {
10   Serial.println("Este programa despliega por consola los multiplos del 4 menores a 100");
11   for (int var = 4; var < 100; var +=4) {
12     if ( var % 12 == 0 ){
13       Serial.println(var);
14       digitalWrite(13, HIGH);
15     }
16     delay(1000);
17     digitalWrite(13, LOW);
18   }
19 }
```

Ejemplos de funcionamiento

- Uso del puerto serial.
- Encendido y apagado de LEDS usando estructuras de control, secuenciales e iterativas.



Tarea

- Imprimir el nombre en el display LCD. Y responder:
 - ¿Qué se debe tener en cuenta para la construcción física?
 - ¿Qué se debe incluir en los programas cuando se trabaja con el LCD?

Referencias

- Tutorial completo Arduino. [En línea]
<https://programarfacil.com/blog/arduino-blog/curso-de-arduino/>
- Página oficial arduino. [En línea]
<https://www.arduino.cc/reference/en/>