

Electrónica Básica

Clase 8

INTRODUCCIÓN AL ARDUINO
ESPECIFICACIONES DEL ARDUINO
TIPOS DE VARIABLES EN EL ARDUINO
OPERADORES TÍPICOS EN EL ARDUINO
ARQUITECTURA DE UN PROGRAMA EN ARDUINO
COMANDOS MAS USADOS
FUNCIONES IF Y ELSE, SWITCH

Arduino

► ¿Qué es?

- Plataforma de electrónica abierta.
- Permite la creación de prototipos electrónicos.
- Basada en software de código abierto.
- Fácil de usar.
- Permiten realizar secuencias y operaciones matemáticas.
- Poseen alta velocidad de procesamiento.
- Sirven para automatizar.

► Tipos de Arduino



Arduino Uno



Arduino Leonardo



Arduino Due



Arduino Yún



Arduino Micro



Arduino Robot



Arduino Esplora



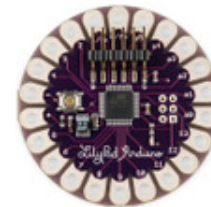
Arduino Mega 2560



Arduino Ethernet



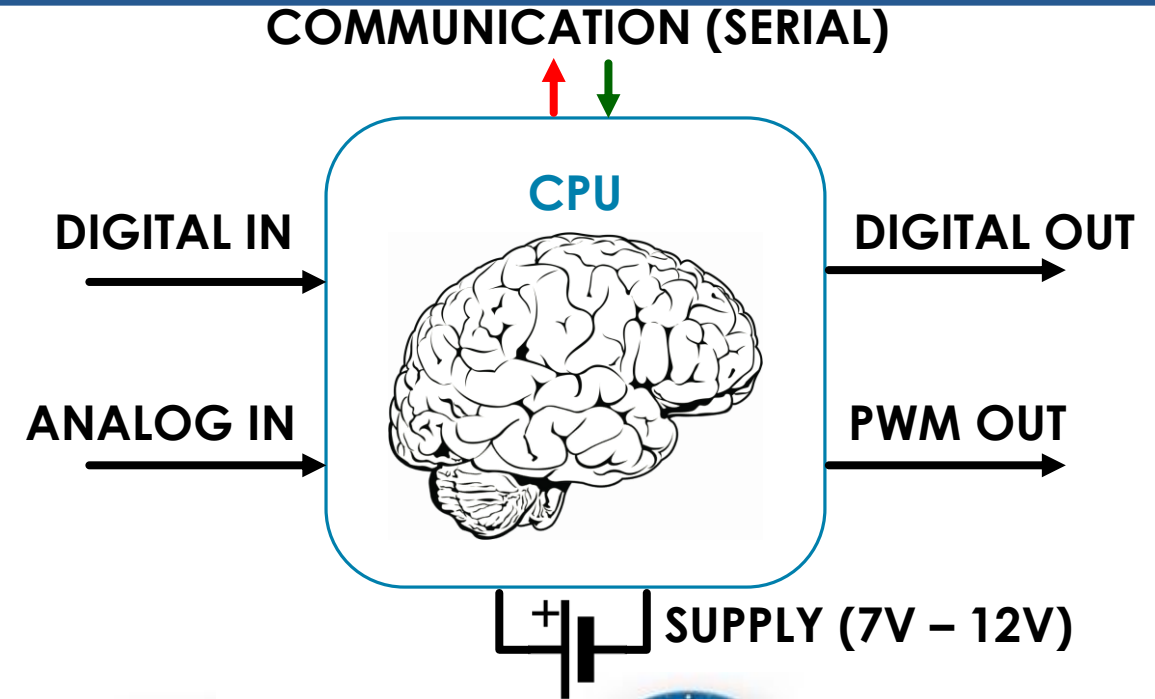
Arduino Mini



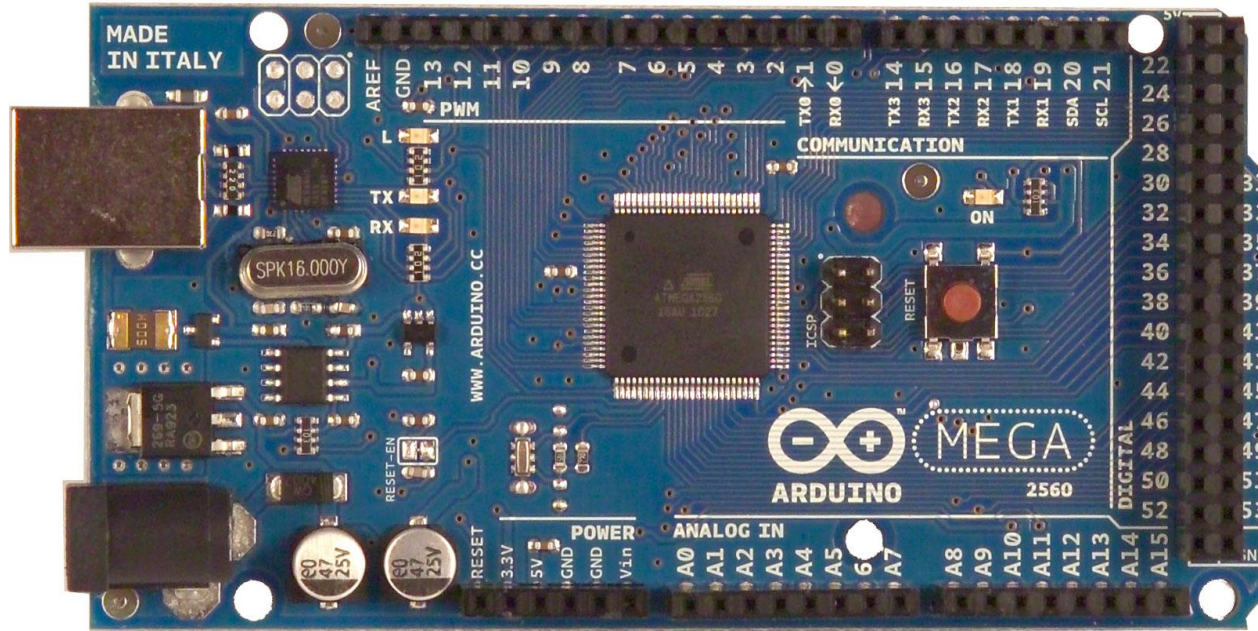
LilyPad Arduino



Arduino Nano



Arduino mega 2560



- Basado en el microcontrolador **ATMEGA 2560**.
- **Alimentación:** Por conector Jack desde **7V hasta 12V**. También se puede alimentar por USB pero para bajos consumos.
- **Corriente Máxima por pin de I/O digital:** 40 mA.
- **Corriente Máxima para el pin de 3.3V:** 50 mA.
- **Entradas y salidas digitales:** 54 pines (**0V ó 5V**).
- **Entradas Análogas:** 16 pines (desde **0V hasta 5V** de voltaje análogo).
- **Salidas por PWM:** 15 pines (están contemplados dentro de los 54 de I/O).
- **Velocidad del oscilador:** 16 MHz.
- **Comunicación:** Conexión USB CDC (Serial).

Tipos de variables básicas para programación

NOMBRE	SINTAXIS	TAMAÑO	RANGO		EJEMPLO
			SIN SIGNO	CON SIGNO	
Booleano	boolean	1 bit	false True	N/A	boolean estado = false;
Caracter ¹	char unsigned char	8 bits (1 byte)	0 a 255	-128 a 127	char micaracter = 'A'; char micaracter = 65; Ambos son equivalentes
Byte	byte	8 bits (1 byte)	0 a 255	N/A	byte hola = B00000111; byte hola = 7; B indica que se escribirá en notación binaria. El B00000111 es igual a 7 en decimal.
Entero	int unsigned int	16 bit (2 bytes)	0 a 65535	-32768 a 32767	int ledPin = 13;
Doble	long unsigned long	32 bit	0 a 4,294,967,295	-2,147,483,648 a 2,147,483,647	long numero = 20000;
Flotante ²	float	32 bit	N/A	-3.4028235E+38 a 3.4028235E+38	float temperatura = 88.5;

¹Consultar tabla ASCII para ver correspondencia (<http://www.asciitable.com/index/asciifull.gif>)

²Consultar documentación de ARDUINO para mayor información (http://arduino.cc/en/Reference/Float#.UxOT7_I5Njl)

EQUIVALENCIAS

word	unsigned int
short	int

Operadores típicos

	SIMBOLO	DESCRIPCIÓN
ARITMÉTICOS	=	Asignación
	+	Suma
	-	Resta
	*	Multiplicación
	/	División
	%	Módulo (Residuo)
COMPARACIÓN	==	Igualdad: $x == y$ es equivalente a: x es igual a y ?
	!=	Desigualdad: $x != y$ es equivalente a: x es distinto a y ?
	<	Menor que
	>	Mayor que
	<=	Menor o igual
	>=	Mayor o igual
BOOLEANOS	&&	AND
		OR
	!	Negación (NOT)
ACUMULADORES	++	Incremento: $y = x ++$ es equivalente a: $y = x + 1$
	--	Decremento: $y = x --$ es equivalente a: $y = x - 1$
	+=	Suma y acumulación: $y += x$ es equivalente a: $y = y + x$
	-=	Resta y acumulación: $y -= x$ es equivalente a: $y = y - x$
	*=	Multiplicación y acumulación: $y *= x$ es equivalente a: $y = y * x$
	/=	División y acumulación: $y /= x$ es equivalente a: $y = y / x$

Arquitectura de un programa en arduino

Declaración de librerías (Ej: #include <SFEMP3Shield.h>)

Definición de pines (Ej: #define ledPin 13)

Declaración de constantes (Ej: const int numSensores = 6;)

Declaración de variables (Ej: float temperatura = 0;)

Declaración de subrutinas o funciones:

Ejemplo subrutina:

```
void leer() //Ejemplo de una subrutina que lee el valor análogo de 0 a 1023 y lo convierte de 0
{
    //a 100 grados guardándolo en la variable flotante temperatura.
    y = analogRead(1); //Lectura análoga del pin A1.
    temperatura = (float)y*100/1023; //Conversión a flotante y en grados celcius
}
```

Ejemplo función:

```
int sumar(int x, int y) //Ejemplo de una función que suma dos números "x" y "y".
{
    return x + y;
}
```

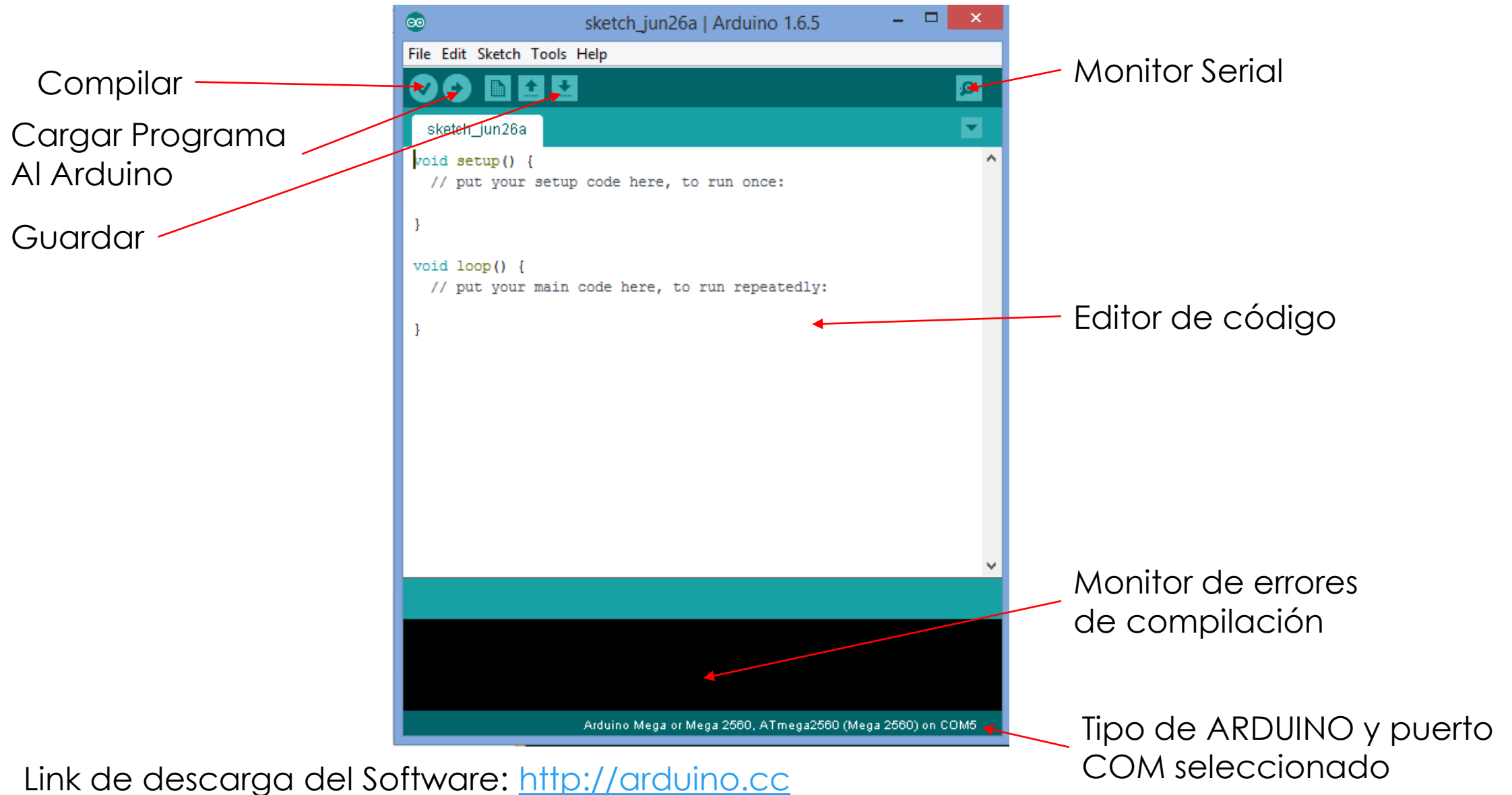
Configuración de puertos y limpieza de puertos:

```
void setup()
{
    //Configuración: Aquí se define que pines son de entrada, salida y las comunicaciones a //usar con la instrucción "pinMode(PIN,OUTPUT o
    INPUT);" sin comillas. Para las //comunicaciones se utiliza la instrucción "Serial.begin(BAUDIOS);" sin comillas.
    //Limpieza: por seguridad, es correcto limpiar las salidas a utilizar con el fin de que estén //apagadas al comienzo del programa. Se utiliza la
    instrucción "digitalWrite(PIN,LOW);" sin //comillas.
}
```

Ciclo infinito (Programa Principal):

```
void loop()
{
    //Programa principal
}
```


Entorno de programación de Arduino



Comandos mas usados

➤ **pinMode**

- Configura el pin especificado como entrada o salida.
- Sintaxis: **pinMode**(pin, mode);
 - Pin: El # del PIN que se desea configurar según el Arduino
 - Mode: Determina si el pin es entrada ó salida. Recibe **INPUT** ó **OUTPUT**.

➤ **digitalWrite**

- Escribe a un pin de salida digital un valor ALTO (5V) ó un BAJO (0V).
- Sintaxis: **digitalWrite**(pin, value);
 - Pin: El # del PIN que se escribirle un valor.
 - Value: **HIGH** ó **LOW**.

➤ **digitalRead**

- Lee y devuelve el valor (value) en que se encuentra un pin de entrada digital.
- Sintaxis: **digitalRead**(pin)
 - Pin: El # del PIN de entrada que se desea leer su valor.
 - Devuelve **HIGH** ó **LOW** dependiendo del valor en que se encuentre el pin de entrada leído.

➤ **delay**

- Pausa el programa por un determinado tiempo (en milisegundos).
- No es muy recomendable utilizarla debido a que frena del todo el programa y luego despues del tiempo continua ejecutandose.
- Sintaxis: **delay**(ms);
 - ms: El numero de milisegundos que se desea pausar el programa (tipo **unsigned long**).

Comandos mas usados - Ejemplo

- Ejemplo: En el PIN 13 hay un LED (L1). Titile el LED ½ Segundo ON y ½ Segundo OFF.

```
//Definición de pines de I/O
#define L1 13 //L1 conectado en el pin 13

void setup()
{
    //Definición de que pin es entrada y que es salida
    pinMode(L1, OUTPUT); //L1 como salida

    //Limpieza de salidas
    digitalWrite(L1, LOW); //Apago L1
}

void loop()
{
    digitalWrite(L1, HIGH); //Prendo L1
    delay(500); //Retardo de 1/2 segundo (500 ms)
    digitalWrite(L1, LOW); //Apago L1
    delay(500); //Retardo de 1/2 segundo (500 ms)
}
```

Función if

- Utilizada en conjunto con un operador de comparación.
- Prueba si una condición se cumple y en caso de cumplirse ejecuta determinadas acciones, luego continua con el programa.
- Sintaxis:

```
if (condición)
{
    //Hago algo aqui
}
else if (otra condición)
{
    //Hago algo aqui si hay otra condicion
}
else
{
    //Hago algo aqui si no se cumple nada de lo anterior
}
```

- Ejemplo con pines digitales de entrada

```
if (digitalRead(pin) == HIGH)
{
    //Hago algo aqui si el pin esta en ALTO
}
```

- Ejemplo con variables internas

```
if (temperatura > 25)
{
    //Hago algo aqui si temperatura es mayor
    a 25 grados
}
```

Función if - Ejemplo

- Ejemplo: en el PIN 2 hay un suiche (SW) y en el PIN 13 hay un LED (L1). Prenda el LED si el suiche esta encendido, de lo contrario, apague el LED

```
//Definición de pines de I/O
#define SW 2 //SW conectado en el pin 2
#define L1 13 //L1 conectado en el pin 13

void setup()
{
    //Definición de que pin es entrada y que es salida
    pinMode(SW, INPUT); //SW como entrada
    pinMode(L1, OUTPUT); //L1 como salida

    //Limpieza de salidas
    digitalWrite(L1, LOW); //Apago L1
}

void loop()
{
    if (digitalRead(SW) == HIGH) //Pregunto si SW esta en HIGH
    {
        digitalWrite(L1, HIGH); //En caso afirmativo, enciendo L1
    }
    else //De resto
    {
        digitalWrite(L1, LOW); //De resto apago L1
    }
}
```

Función switch

- ▶ Permite realizar diferentes acciones dependiendo de una variable que puede tener varias posibilidades.
- ▶ Es como hacer varios if por la misma variable pero diferentes valores (Switch ahorra mas memoria);
- ▶ Cada case es el posible valor que puede tomar la variable y este case se finaliza con break;
- ▶ La variable por la que se pregunta en lo posible debe ser de tipo entero.
- ▶ Sintaxis:

```
switch (var)
{
    case 0:
        //Hago algo aquí si var es igual a cero
        break;
    case 1:
        //Hago algo aquí si var es igual a uno
        break;
    case 2:
        //Hago algo aquí si var es igual a dos
        break;
}
```

- ▶ Es posible también preguntar por etiquetas (label) predefinidas al comienzo de un programa con la instrucción #define.

```
switch (var)
{
    case label1:
        //Hago algo aquí si var es igual a la etiqueta label1
        break;
    case label2:
        //Hago algo aquí si var es igual a la etiqueta label2
        break;
}
```

Ejemplo final

➔ Escriba un programa que controle un semáforo y una vez se active el fotosensor (sensor) el semáforo se desactive y quede solo el Led Rojo intermitente.

```
//Definición de pines de I/O
#define LedRed 24 //LedRed en el pin digital 24
#define LedYellow 23 //LedYellow en el pin digital 23
#define LedGreen 22 //LedGreen en el pin digital 22
#define sensor 30 //sensor en el pin digital 30

void setup()
{
    //Definición de que es entrada y que es salida
    pinMode(LedRed, OUTPUT); //LedRed como salida
    pinMode(LedYellow, OUTPUT); //LedYellow como salida
    pinMode(LedGreen, OUTPUT); //LedGreen como salida
    pinMode(sensor, INPUT); //sensor como entrada

    //Limpieza de salidas
    digitalWrite(LedRed, LOW); //Apago LedRed
    digitalWrite(LedYellow, LOW); //Apago LedYellow
    digitalWrite(LedGreen, LOW); //Apago LedGreen
}

void loop()
{
    if (digitalRead(sensor) == LOW) //Si el sensor esta apagado
    {
        //Secuencia semaforo normal
        digitalWrite(LedRed, HIGH);
        digitalWrite(LedYellow, LOW);
        digitalWrite(LedGreen, LOW);
        delay(5000); //Retardo de 5 segundos

        digitalWrite(LedRed, LOW);
```

```
        digitalWrite(LedYellow, LOW);
        digitalWrite(LedGreen, HIGH);
        delay(2500); //Retardo de 2.5 segundos

        digitalWrite(LedRed, LOW);
        digitalWrite(LedYellow, HIGH);
        digitalWrite(LedGreen, LOW);
        delay(1000); //Retardo de 1 segundo
    }
    else //De resto
    {
        //Apago todos los LED
        digitalWrite(LedRed, LOW);
        digitalWrite(LedYellow, LOW);
        digitalWrite(LedGreen, LOW);
        delay(5000); //Retardo de 5 segundos

        //Prendo el LedRed
        digitalWrite(LedRed, HIGH);
        digitalWrite(LedYellow, LOW);
        digitalWrite(LedGreen, LOW);
        delay(5000); //Retardo de 5 segundos
    }
}
```

MUCHAS GRACIAS