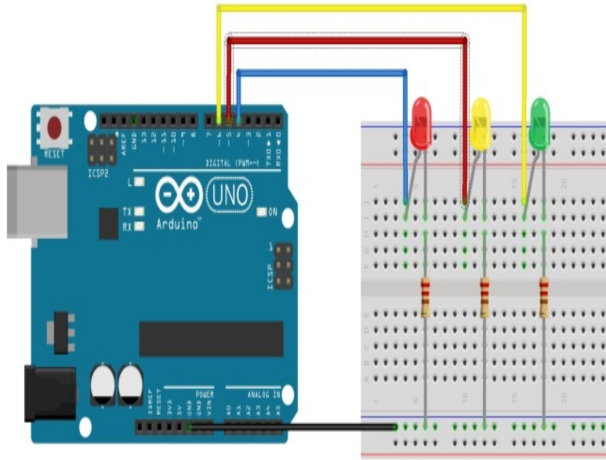
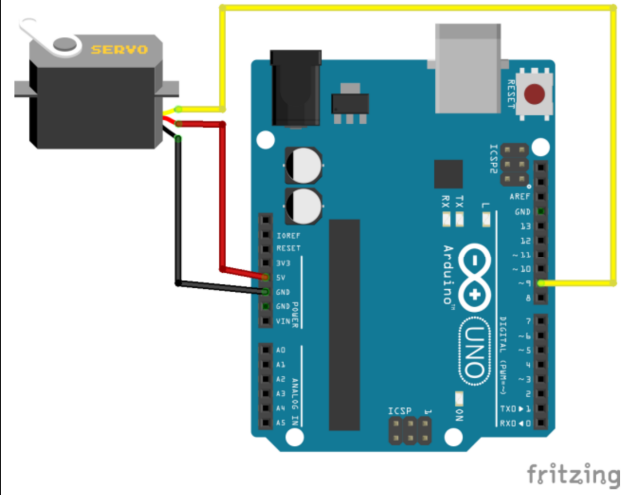


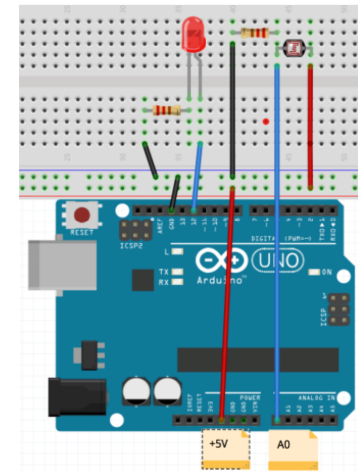
SEMÁFORO



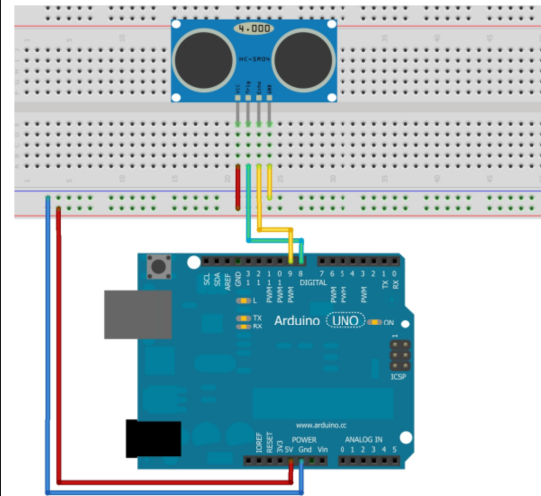
SERVOMOTOR



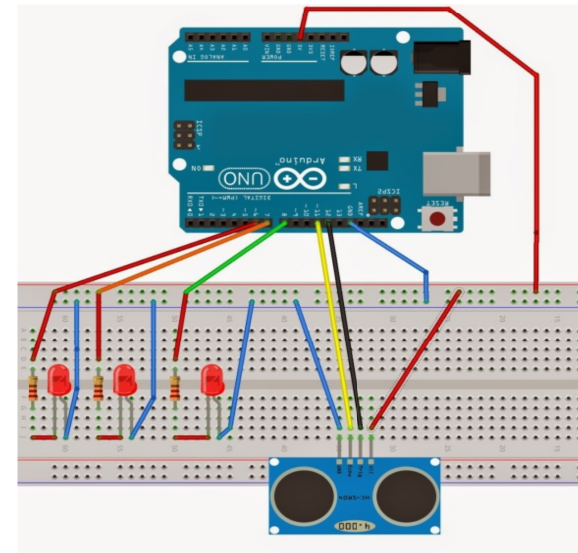
LED y LDR



ULTRASONIDOS



SEMÁFORO Y ULTRASONIDOS



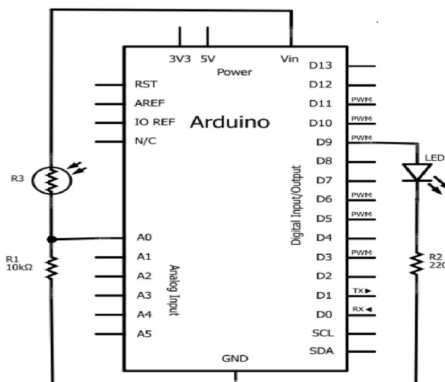
www.arduino.cc/reference/en

LED - SALIDAS DIGITALES //digitalWrite

Para facilitar las conexiones usaremos siempre los mismos Pines:

| | | |
|--|--|---|
| LED verde en Pin=3 (cable verde) LED amarillo en Pin=4 (cable amarillo) LED rojo en Pin=5 (cable rojo) | Ultrasonidos Trigger en Pin=6 Ultrasonidos Echo en Pin=7 5V (cable rojo) GND (cable marrón) | Servo en Pin=9 (cable amarillo) Vin (cable rojo) GND (cable marrón) |
|--|--|---|

Baudios (bits/s)

| | | |
|--|--|--|
| <pre>//led //bloques //instrucciones void setup() { Serial.begin(9600); pinMode (3, OUTPUT); } void loop() { digitalWrite(3, HIGH); }</pre> | <pre>//ledParpadeo //delay void setup() { Serial.begin(9600); pinMode (3, OUTPUT); } void loop() { digitalWrite(3, HIGH); delay(1000); digitalWrite(3, LOW); delay(1000); }</pre> | <pre>//ledParpadeoVariables int led=3; void setup() { Serial.begin(9600); pinMode (led, OUTPUT); } void loop() { digitalWrite(led, HIGH); delay(1000); digitalWrite(led, LOW); delay(1000); }</pre> |
| <pre>//Semáforo = Semáforo LED RGB int verde = 3; int amarillo = 4; int rojo = 5; void setup() { Serial.begin(9600); pinMode(verde, OUTPUT); pinMode(amarillo, OUTPUT); pinMode(rojo, OUTPUT); } void loop() { digitalWrite(verde, HIGH); delay(1000); digitalWrite(verde, LOW); digitalWrite(amarillo, HIGH); delay (1000); digitalWrite(amarillo, LOW); digitalWrite(rojo, HIGH); delay(1000); digitalWrite(rojo, LOW); }</pre> | <pre>//LDR i LED int led=3; void setup() { Serial.begin(9600); pinMode(led,OUTPUT); } void loop() { int luz=analogRead(A0); if (luz<600) digitalWrite(led, HIGH); else digitalWrite (led, LOW); Serial.println(luz); } //Se ajusta el nivel de luz al que se quiere que se encienda el led cambiando el valor 600 por uno menor si hay poca luz o mayor si hay mucha luz</pre> |  <p>Esquema del montaje de LDR y LED: La LDR lleva una resistencia pull-up (eleva una tensión de entrada o salida del circuito mientras está en reposo. Esto evita que se hagan lecturas erróneas si este pin no tiene nada conectado o no está recibiendo una señal. El IED lleva una resistencia limitadora de intensidad</p> |

SALIDAS ANALÓGICAS //analogWrite

Ver apuntes arduino Nivel Pardillo/ salidas analógicas (PWM)

| //ledIntensidad1 //for... | //ledIntensidad2 //while... |
|--|---|
| <pre>int led = 3; int brillo=0; void setup() { Serial.begin(9600); pinMode(led, OUTPUT); } void loop() { for (brillo = 0; brillo < 255; brillo=brillo+1) { analogWrite(led, brillo); delay(30); } }</pre> | <pre>int led = 3; int brillo = 0; void setup() { Serial.begin(9600); pinMode(led, OUTPUT); } void loop() { while (brillo == 0 brillo < 255) { brillo = brillo + 1; analogWrite(led, brillo); delay(30); } }</pre> |

| //ledIntensidad3 //if... | //ledIntensidad4 //if....else if... //el mejor |
|--|---|
| <pre>int led = 3; int brillo = 0; void setup() { Serial.begin(9600); pinMode(led, OUTPUT); } void loop() { if (brillo == 0 brillo < 255) { brillo = brillo + 1; analogWrite(led, brillo); delay(30); } }</pre> | <pre>int led = 3; int brillo = 0; int incremento = 1; void setup() { Serial.begin(9600); pinMode(led, OUTPUT); } void loop() { if (brillo == 0) { incremento = 1; } else if (brillo == 255) { incremento = -1; } brillo = brillo + incremento; analogWrite(led, brillo); delay(20); }</pre> |

Probar:

Brillo>=0

brillo>-1

brillo=0

Brillo<=255

brillo<255

brillo=255

ULTRASONIDOS

Long: ver tutorial "Manual+progamación+arduino"

<http://www.prometec.net/sensor-distancia/>

<https://www.luisllamas.es/medir-distancia-con-arduino-y-sensor-de-ultrasonidos-hc-sr04/>

<http://elcajondeardu.blogspot.com.es/2014/03/tutorial-sensor-ultrasonidos-hc-sr04.html>

```
//ultrasonidosLed

int trigger = 6;
int echo = 7;
int led = 3;
long tiempo;
long distancia ;

void setup()
{ Serial.begin (9600);
  pinMode(trigger, OUTPUT);
  pinMode(echo, INPUT);
  pinMode(led, OUTPUT);
}
void loop() {
  digitalWrite(trigger, LOW);           // Nos aseguramos de que el trigger está desactivado
  delayMicroseconds(2);                // Para asegurarnos de que el trigger esta LOW
  digitalWrite(trigger, HIGH);         // Activamos el pulso de salida
  delayMicroseconds(10);               // Esperamos 10µs. El pulso sigue active este tiempo
  digitalWrite(trigger, LOW);         // Cortamos el pulso
  tiempo = pulseIn(echo, HIGH) ;       // Se detecta echo
  distancia = tiempo / 2 * 1 / 29.2 ;  //Ver NOTA

  Serial.print(" cm= ");
  Serial.println(distancia);           // imprime la distancia en el Monitor Serie (opcional)

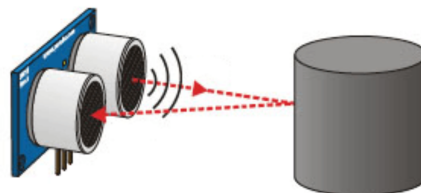
  if ( distancia < 20)
    digitalWrite ( led , HIGH) ;
  else
    digitalWrite( led , LOW) ;
  delay (500) ;
}
```

NOTA

$$343 \frac{m}{s} \cdot 100 \frac{cm}{m} \cdot \frac{1}{1000000} \frac{s}{\mu s} = \frac{1}{29.2} \frac{cm}{\mu s}$$

La velocidad del sonido es $v=343$ m/s. Si cambiamos las unidades a $cm/\mu s$ nos queda que el sonido recorre 1 cm en un tiempo de 29,2 μs

El ultrasonidos mide el tiempo que el pulso emitido por el Trigger tarda en ser recibido por el Echo, por lo que la distancia recorrida por el pulso es el doble de la que queremos medir. Por tanto, hemos de dividir el tiempo entre dos para hallar la distancia real desde el objeto al ultrasonidos..



$$distancia = \frac{tiempo}{2} * \frac{1 cm}{29,2 \mu s}$$

```

// UltrasonidosLedRojoVerde

int trigger = 10;
int echo = 11;
int ledVerde = 3;
int ledRojo = 5;
long tiempo;
long distancia ;

void setup()
{ Serial.begin (9600);
  pinMode(trigger, OUTPUT);
  pinMode(echo, INPUT);
  pinMode(ledVerde, OUTPUT);
  pinMode(ledRojo, OUTPUT);
}

void loop() {
  digitalWrite(trigger, LOW);    // Nos aseguramos de que el trigger está desactivado
  delayMicroseconds(2);        // Para asegurarnos de que el trigger esta LOW
  digitalWrite(trigger, HIGH);  // Activamos el pulso de salida
  delayMicroseconds(10);       // Esperamos 10µs. El pulso sigue active este tiempo
  digitalWrite(trigger, LOW);  // Cortamos el pulso y a esperar el echo
  tiempo = pulseIn(echo, HIGH) ;
  distancia = tiempo / 2 * 1 / 29.1 ;

  Serial.print(" cm= ");
  Serial.println(distancia); // imprime la distancia en el Monitor Serie

  if ( distancia <= 20){
    digitalWrite(ledVerde , HIGH);
    digitalWrite (ledRojo , LOW);
  }

  else {
    digitalWrite( ledVerde , LOW) ;
    digitalWrite( ledRojo , HIGH) ;

    delay (500) ;
  }
}

```

```
// UltrasonidosLedRojoAmarilloVerde
```

```
int verd=3;  
int groc=4;  
int vermell=5;  
int trigger=6;  
int echo=7;  
long temps;  
long distancia;
```

```
void setup() {  
  Serial.begin(9600);  
  pinMode(trigger,OUTPUT);  
  pinMode(echo,INPUT);  
  pinMode(verd,OUTPUT);  
  pinMode(groc,OUTPUT);  
  pinMode(vermell,OUTPUT);  
}
```

```
void loop() {  
  digitalWrite(trigger,LOW);  
  delayMicroseconds(2);  
  digitalWrite(trigger,HIGH);  
  delayMicroseconds(10);  
  digitalWrite(trigger,LOW);  
  temps=pulseIn(echo,HIGH);  
  distancia=temps/2*1/29.2;
```

```
  Serial.print("cm=");  
  Serial.println(distancia);
```

```
  if(distancia<=10){  
    digitalWrite(groc,LOW);  
    digitalWrite(verd,LOW);  
    delay(50);  
    digitalWrite(vermell,HIGH);  
  }
```

```
  else if(distancia>10 & distancia<=20){  
    digitalWrite(vermell,LOW);  
    digitalWrite(verd,LOW);  
    delay(50);  
    digitalWrite(groc,HIGH);  
  }
```

```
  else if(distancia>20){  
    digitalWrite(groc,LOW);  
    digitalWrite(vermell,LOW);  
    delay(50);  
    digitalWrite(verd,HIGH);  
  }
```

```
}
```

SERVOMOTOR

| | |
|---|--|
| <pre>//servolento //Se mueve lentamente hacia los 180°. La vuelta es rápida #include<Servo.h> Servo ser1; int grados=0; void setup() { ser1.attach(9,600,2400); } void loop() { grados=grados+1; ser1.write (grados); delay(20); if (grados==180) grados=0; delay(20); }</pre> | <pre>//Barrera //Se mueve lentamente en los dos sentidos #include <Servo.h> #define IZQUIERDA 0 #define DERECHA 1 //constante Servo ser1; int grados = 0; int sentido = IZQUIERDA; void setup() { ser1.attach(9, 800, 1800); } void moverse() { if (sentido == IZQUIERDA) { grados = grados + 1; } else { grados = grados - 1; } ser1.write(grados); delay(50); } void cambiarSentido() { //declaración de una función if (grados == 90) { delay(5000); sentido = DERECHA; //asignación } if (grados == 0) { delay(5000); sentido = IZQUIERDA; } } void loop() { moverse(); //llamada a función cambiarSentido(); delay(20); }</pre> |
|---|--|

NOTA: ver funcionamiento del servo, de las salidas analógicas i de los PWM en “Apuntes arduino Nivel Pardillo”/4.salidas analógicas.

Nuestro servo emite pulsos de 600 μ s hasta 2400 μ s

| | |
|--|---|
| <p>90% de 5V o de 2400 μs= 4,5 V y 2160 μs 80% de 5V o de 2400 μs= 4 V y 1920 μs 70% de 5V o de 2400 μs= 3,5 V y 1680 μs 60% de 5V o de 2400 μs= 3 V y 1440 μs 50% de 5V o de 2400 μs= 2,5 V y 1200 μs</p> | <p>40% de 5V o de 2400 μs= 2 V y 960 μs 30% de 5V o de 2400 μs= 1,5 V y 720 μs 20% de 5V o de 2400 μs= 1 V y 480 μs 10% de 5V o de 2400 μs= 0,5 V y 240 μs</p> |
|--|---|

```

//ultrasonidosServo

#include <Servo.h>
Servo ser1;
int trigger = 6;
int echo = 7;
int led = 3;
long tiempo;
long distancia ;

void setup()
{
  Serial.begin (9600);
  pinMode(trigger, OUTPUT);
  pinMode(echo, INPUT);
  pinMode(led, OUTPUT);
  ser1.attach(9, 1000, 2000);
}

void loop() {
  digitalWrite(trigger, LOW);    // Nos aseguramos de que el trigger está desactivado
  delayMicroseconds(2);         // Para asegurarnos de que el trigger esta LOW
  digitalWrite(trigger, HIGH);  // Activamos el pulso de salida
  delayMicroseconds(10);        // Esperamos 10µs. El pulso sigue active este tiempo
  digitalWrite(trigger, LOW);   // Cortamos el pulso y a esperar el echo
  tiempo = pulseIn(echo, HIGH) ;
  distancia = tiempo / 2 * 1 / 29.1 ;

  Serial.print(" cm= ");
  Serial.println(distancia); // imprime la distancia en el Monitor Serie

  if ( distancia < 20){
    digitalWrite ( led , HIGH) ;
    ser1.write(90);
  }

  else{
    digitalWrite( led , LOW) ;
    ser1.write(0);
  }

  delay (300) ;
}

```

https://www.youtube.com/results?search_query=codigofacilito+arduino

<http://www.prometec.net/indice-tutoriales/>

<https://www.luisllamas.es/tutoriales-de-arduino/>

<http://elcajondeardu.blogspot.com.es/p/tutoriales.html>

[Arduino Ayuda](#)