

# Sensores y actuadores



[https://commons.wikimedia.org/wiki/File:Arduino\\_Uno\\_-\\_R3.jpg](https://commons.wikimedia.org/wiki/File:Arduino_Uno_-_R3.jpg)

Práctica 2 – Actividades (v1.4.5 septiembre 2022)

## Software para robots

Cristian González García

[gonzalezcristian@uniovi.es](mailto:gonzalezcristian@uniovi.es)

# Índice

Actividades principales .....	2
Control de acceso.....	2
(Abrir2.1) Abrir puerta (0,15 puntos).....	2
(Cerrar2.2) Cerrar puerta (0,10 puntos) .....	3
(Contraseña2.3) Contraseña (0,15 puntos).....	3
(Auto2.4) Cierre automático seguro usando ultrasonidos (0,15 puntos) .....	4
(Contador2.5) Apertura y cierre automático con contador de entradas (0,25 puntos) .....	4
(Semáforo2.6) Configuración de tiempo y semáforo (0,25 puntos).....	6
(Contador2.7) Contador de personas en el interior (0,15 puntos).....	6
Ampliaciones.....	7
(Juego2.8) Juego de rapidez (1 punto).....	7
(Calculadora2.9) Calculadora (0,3 puntos) .....	7
(Juego2.10) Escondite inglés (0,5 puntos) .....	8
(Juego2.11) Ampliación escondite inglés (0,2 puntos) .....	8
(Mando2.12) Mando de consola (0,2) .....	8

El total de las actividades principales tienen un valor de 1,2 puntos y 2,2 puntos en ampliaciones dentro del bloque 1.

## Actividades principales

### Control de acceso

La idea principal de estos seis ejercicios es comenzar construyendo un sistema simple de acceso, para, en cada ejercicio, hacerlo más complejo mediante la ampliación de este con nuevas funcionalidades para hacerlo más seguro o aplicable a diferentes espacios reales:

- Abrir y cerrar puertas.
- Contraseña.
- Cierre automático de la puerta de forma segura para que no aplaste a nadie.
- Apertura automática al detectar a alguien.
- Contador de entradas.
- Configuración del tiempo de puerta abierta.
- Semáforo.

Este sistema será un dispositivo que se compondrá de hardware y software. Permitirá abrir una puerta automática, como podría ser la de un ascensor, tienda o centro comercial. Para lograr esto se utilizarán muchos de los componentes presentados en la teoría de esta práctica.

Para simular la puerta automática se utilizarán dos LEDs:

- Un LED rojo encendido para abierto/abriéndose/no pasar ahora.
- Un LED verde encendido para cerrado/cerrándose/se puede pasar.

Los ejercicios obligatorios son incrementales, así que, deberéis tenerlos implementados en orden si queréis ir haciendo el siguiente.

Si resultara más fácil, se pueden montar todos los ejercicios de control de acceso juntos y grabar un solo vídeo mostrando todos los ejercicios. Si se opta por esto, asegurarse de que se muestran en orden, todos los ejercicios.

Lista de reproducción en YouTube:

<https://www.youtube.com/watch?v=07iYgehBpTo&list=PLpe5dTl2xCy8CNbYdZPkCwvCwpLV4lbnW>

#### (Abrir2.1) Abrir puerta (0,15 puntos)

*Material necesario: 1 LED rojo, 1 LED verde, 2 resistencias y 1 teclado.*

**La puerta se abre** cuando el usuario pulsa el **botón A** del teclado y **permanece abierta durante 5 segundos**, haciendo que un LED de color rojo esté encendido. Pasado ese tiempo se cierra. En otros casos, el LED verde estará iluminando para indicar que se puede pasar.

Si el usuario vuelve a pulsar el botón A mientras la puerta está abierta esa pulsación se ignora.

Si el usuario pulsa cualquier otra tecla, no debe realizar ninguna otra acción.

**Demostrado pulsando varias teclas de cada fila del teclado para que se pueda valorar el ejercicio completo.**

Vídeo del ejercicio: <https://youtu.be/07iYgehBpTo>

(Cerrar2.2) Cerrar puerta (0,10 puntos)

*Material necesario: 1 LED rojo, 1 LED verde, 2 resistencias y 1 teclado.*

Además de abrirse la puerta al pulsar el botón **A**, cuando el usuario pulse el **botón C** del teclado, **la puerta debe cerrarse**, siempre y **cuando** en ese momento **este abierta**, e indistintamente de que aún no hayan pasado los 5 segundos de tiempo de apertura.

El proceso de cerrar la puerta debe ser inmediato una vez se pulsa el botón.

**NOTA:** en este apartado, la espera de 5 segundos no se puede realizar con un delay(5000), ya que debemos continuar procesando pulsaciones de teclado, es decir, ejecutando interacciones de la función loop para poder detectar la orden de cierre de puerta. Una opción puede ser guardar el milisegundo en el que se pulso el botón de abrir la puerta y compararlo con el tiempo actual (para saber si han pasado los 5 segundos).

Vídeo del ejercicio: <https://youtu.be/XHRyKikYAcQ>

(Contraseña2.3) Contraseña (0,15 puntos)

*Material necesario: 1 LED rojo, 1 LED verde, 2 resistencias y 1 teclado.*

Ahora añadiremos seguridad a nuestra puerta, requiriendo contraseña para entrar. La puerta se deberá abrir cuando el usuario introduzca en el teclado una **contraseña: 9876**. La contraseña se comprobará después de que el usuario introduzca 4 pulsaciones de teclado. **Ya no puede abrirse pulsando la tecla A**, pero **sí se cerrará con la tecla C**.

Si el usuario introduce la contraseña de forma correcta, el LED verde debe mantenerse apagado durante 1 segundo. Tras este segundo, la puerta se abrirá durante 5 segundos, lo que implica que el LED rojo se encenderá durante ese tiempo. Durante este periodo, no se podrán introducir contraseñas, y si se introduce algo, excepto la tecla C, se obviará.

**Si el usuario introduce la contraseña de forma incorrecta**, el LED verde deberá parpadear 3 veces en 3 segundos y quedarse encendido, pues la puerta seguirá cerrada. Durante el periodo de parpadeos no se podrán introducir contraseñas ni hará caso a ninguna tecla.

El sistema debe funcionar de forma continua. Es decir, una vez el usuario introduzca una contraseña el sistema abra la puerta y la cierre el sistema o el usuario, el sistema deberá estar listo para continuar su funcionamiento y recibir más contraseñas.

Vídeo del ejercicio: <https://youtu.be/3CbpgwQHaYs>

#### (Auto2.4) Cierre automático seguro usando ultrasonidos (0,15 puntos)

*Material necesario: 1 LED rojo, 1 LED verde, 2 resistencias, 1 teclado y 1 sensor de ultrasonidos.*

Ahora, añadiremos un sistema de seguridad de cara a las personas para que la puerta no se cierre cuando ellos están pasando y así no haga daño a la gente y no les llegue a tocar o asustar. Por eso, **ampliaremos la puerta añadiendo un sensor de ultrasonidos**. La puerta seguirá abriéndose con la misma contraseña de antes, y se cerrará tras 5 segundos o al pulsar la tecla C. Los LEDs funcionarán igual que en el ejercicio anterior.

Supongamos que el usuario está esperando para pasar por la puerta, o está entrando, cuando el sensor de ultrasonidos detecta una distancia muy baja, menor de 10 cm.

Colocaremos un sensor de **ultrasonidos** en la parte de la entrada de la puerta. Mientras este sensor este detectando un objeto a menos de **10 centímetros, la puerta no deberá cerrarse, aunque hayan pasado los 5 segundos**. Si se da este caso, significará que alguien está esperando para entrar, o está entrando, y la puerta deberá reiniciar su contador **añadiendo otros 5 segundos**. El sensor de ultrasonidos podrá estar funcionando continuamente, o bien, solo cuando se necesite, es decir, cuando se vaya a cerrar la puerta ya sea por la tecla C o por el tiempo.

En el caso de que se quiera cerrar con la tecla C, **si alguien está a menos de 10 cm, la puerta no se deberá cerrar**.

NOTA: el vídeo deberá contener todas estas posibilidades.

Vídeo del ejercicio: <https://youtu.be/YOWtwO4H20c>

#### (Contador2.5) Apertura y cierre automático con contador de entradas (0,25 puntos)

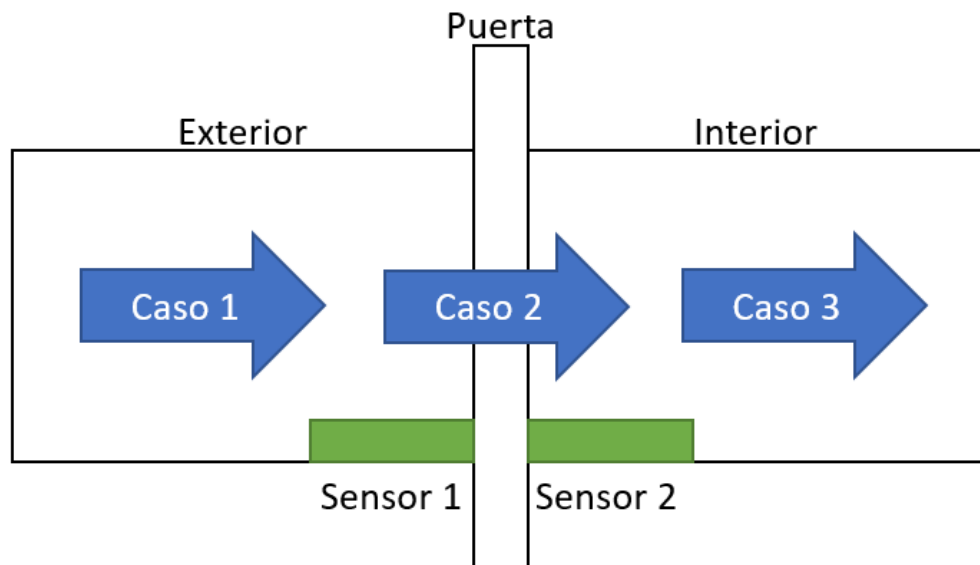
*Material necesario: 1 LED rojo, 1 LED verde, 2 resistencias, 1 teclado, 1 sensor de ultrasonidos, 1 fotorresistor y 1 pantalla de segmentos.*

Ahora incorporaremos un contador de entradas para aquellos sitios que necesiten de esta estadística. Para esto, colocaremos un sensor de ultrasonidos en la parte de entrada de la puerta y otro sensor en la parte interior (el sensor IR de proximidad o el de luz, o cualquier otro que sirva para detectar presencia). Tener dos sensores nos servirá para detectar cuando un usuario ha entrado o ha salido. Además, añadiremos la pantalla de segmentos para visualizar el número de entradas.

Si utilizamos el sensor de luz, supondremos que está en el suelo y que el usuario tiene que pasar por encima del sensor. Luego, detectaremos esta situación cuando el sensor detecte un nivel de luz bajo.

Para entrar, cuando el usuario introduzca la contraseña, la puerta deberá abrirse y cerrarse tras 5 segundos, o bien, no cerrarse si el sensor de ultrasonidos detecta una presencia (Caso 1). Esto es exactamente igual que en los ejercicios anteriores. En cambio, **si el sensor interior detecta algo y el exterior nada (Caso 3), entonces, la puerta deberá cerrarse**, pues esto

significará que la persona ya ha entrado. Si ambos sensores detectan algo, la puerta se mantendrá abierta, pues puede ser que esté pasando un objeto demasiado grande, como una maleta o un carrito, o la persona y/o el objeto esté entre ambos sensores, es decir, en medio de la puerta (Caso 2). La puerta tiene que seguir pudiéndose cerrar con C, por motivos de seguridad y eficacia.



*Ilustración 1 Casos de entrada*

Para que se considere que un usuario ha **entrado**, este debería ser detectado primero por el sensor exterior y luego por el sensor interior. El número de usuarios que han entrado se guardará en una variable global y deberá ser mostrado por la pantalla de segmentos. Esta variable debe contabilizar solo las entradas a lo largo de todo el ejercicio, sin tener en cuenta si han salido o no. Esto es importante si el sistema se incorpora en una tienda o en un centro comercial y se quiere saber el número de clientes que han entrado. Por ello, el valor que se mostrará en la pantalla será el total de entradas que han ido sucediéndose desde que se inició el sistema.

En cambio, si el primer sensor en detectar algo es el interior, entonces, la puerta se abrirá automáticamente y se cerrará una vez el sensor interior, que es el de luz, no detecte nada. En cambio, si el sensor exterior, que es el de ultrasonidos, detecte algo y el sensor interior no detecta nada, la puerta se mantendrá abierta, pues puede haber algo atravesando la puerta todavía. Si ambos sensores están detectando algo, la puerta seguirá abierta, pues significa que algo está ocupando todo el espacio. Si ningún sensor detecta nada, la puerta se cerrará, pues significará que la persona y/o el objeto ha salido.

Hay que mostrar el funcionamiento de todo correctamente. Tened en cuenta que tenéis que mostrar el teclado funcionando de nuevo, pues en este nuevo caso, debería de añadir 1 entrada nueva y esta se visualizará en la pantalla. Si falta algo en el vídeo, se restarán puntos.

Vídeo del ejercicio: <https://youtu.be/wY92m3iLSiY>

### (Semáforo2.6) Configuración de tiempo y semáforo (0,25 puntos)

*Material necesario: 1 LED RGB, 1 teclado, 1 sensor de ultrasonidos, 1 fotorresistor y 1 pantalla de segmentos.*

En este caso, añadiremos un color más a los LEDs para mostrar si la persona está entrando o saliendo y así facilitar el tránsito, y cambiando los LEDs por un LED RGB para solo tener una «bombilla».

La puerta debe funcionar como hasta el ejercicio 2.6. La diferencia es que hay que incluirle la **siguiente posibilidad de configuración**:

1. Definir el número de segundos que permanece la puerta abierta como máximo utilizando el teclado siguiendo la siguiente combinación de teclas: **Botón \* + Números + Botón \***: el número de segundos introducido por el administrador será el tiempo máximo que la puerta permanecerá abierta.

Además, **hay que cambiar los 2 LEDs actuales por un LED RGB** y así utilizar sus tres colores de la siguiente manera para crear un **semáforo**:

1. Cuando un usuario esté **entrando**, que es el equivalente a la **puerta está abierta** o abriéndose, **pero siendo el primer sensor en funcionar el de ultrasonidos**, el LED debe estar encendido en **rojo**. El proceso de entrada comienza cuando el sensor exterior detecta un usuario, sigue cuando ambos sensores están activos, y finaliza cuando el sensor interior lo detecta.
2. Cuando un usuario esté **saliendo**, que es el equivalente a la **puerta está abierta** o abriéndose, **pero siendo el primer sensor en funcionar el de fotorresistor o IR**, el LED debe estar encendido en **azul**. El proceso de salida comienza cuando el sensor interior detecta un usuario y finaliza cuando el sensor exterior lo detecta.
3. Cuando no se está registrando ningún acceso el LED se debe encender en color **verde**. Es decir, puede cruzar la puerta cualquiera, esté en el interior o exterior. Este debe de funcionar exactamente igual que antes de forma que de los avisos de contraseña correcta e incorrecta.

### (Contador2.7) Contador de personas en el interior (0,15 puntos)

*Material necesario: 1 LED RGB, 1 teclado, 1 sensor de ultrasonidos, 1 fotorresistor y 1 pantalla de segmentos.*

Ahora, debido a la pandemia, hay restricciones de acceso a los sitios en función del sitio y del número de personas. Por ello, ampliaremos el trabajo para que el contador no solo lleva la estadística general de entrada, sino también el número de personas totales que están en el interior.

En la pantalla de segmentos se mostrarán ambos valores, primero el total del día y después el actual de personas en el interior.

## Ampliaciones

### (Juego2.8) Juego de rapidez (1 punto)

*Material necesario: 1 o 2 teclados y 1 pantalla de segmentos.*

De 2 a 4 jugadores.

Se deben utilizar 1 o 2 teclados (turnos o a la vez) y pantalla de segmentos. Si usan dos teclados, muy posiblemente se necesitará utilizar un Arduino Mega.

Mostrar una serie de operaciones (sumas, restas, series, etc.) en la consola. Cuando terminen de aparecer, el jugador que introduzca el resultado correcto más rápidamente gana esa ronda. En la pantalla de segmentos se mostrará el marcador entre ambos jugadores: usar los dos primeros dígitos para el primer jugador y los dos siguientes para el segundo jugador. Si fueran 4 jugadores usar un dígito para cada jugador y que jueguen por turnos.

Hay que tener en cuenta que debe tener varias opciones: iniciar partida, número de rondas, avisar cuando va a empezar a la ronda y cuando termina de mostrar números (usando pitidos, una segunda pantalla auxiliar, LEDs, la combinación u otros), pausar el juego, reiniciarlo, finalizarlo, etc. El menú se mostrará en la consola del IDE, pero la entrada será utilizando el teclado del Arduino.

Cuantas más opciones y posibilidades de juego se hayan implementadas, más nota dentro del punto que vale se otorgará. Si se hiciera bastante, puede incluso darse más de 1 punto.

- 1 punto: 4 jugadores + operaciones aleatorias seguidas y no hardcodeadas + menú de opciones y entrada por teclado.
- -0,25: solo con 2 jugadores.
- -0,25: con operaciones hardcodeadas.
- -0,25: operaciones no seguidas.
- -0,25: menú simple (sin configuración del juego, números jugadores, rondas, dificultad de operaciones, etc.).

### (Calculadora2.9) Calculadora (0,3 puntos)

*Material necesario: 1 teclado y 1 pantalla de segmentos.*

Hacer una calculadora utilizando el teclado. Se pueden utilizar las letras como operaciones (+, -, \*, /, =), así que tendréis que decir letra es cuál.

Además, hay que utilizar la pantalla de segmentos para ir mostrando los números que se introducen.

La calculadora también debe soportar también el borrado de un dígito o de toda la operación. Pueden ser dos teclas diferentes, una combinación, etc.:

- 5 borrar/borraTodo = 0
- 51 borrar = 5
- 51 cualquierOperación borrarTodo = 0



- 51 cualquierOperación 13 borrarTodo = 0
- 51 cualquierOperación 13 borrar = 51 cualquierOperación 1

### (Juego2.10) Escondite inglés (0,5 puntos)

*Material necesario: 1 LED RGB, 1 sensor de ultrasonidos y 1 pantalla de segmentos.*

Se requerirá tenerlo implementado para el opcional 2.10

El juego típico consiste en que una persona, quien se la queda, va diciendo en voz alta y de espaldas lo siguiente: «1, 2, 3, escondite inglés» mientras los participantes se acercan a él desde una distancia dada. Después se gira y mira a ver si algún participante está moviéndose cuando terminó de hablar. En este caso, el participante que se haya movido mientras no cantaba la canción vuelve a comenzar.

La pantalla irá marcando una cuenta, aleatoria y que puede variar de velocidad. Cuando esta llegue a 0, el sensor de ultrasonidos deberá de comprobar durante unos segundos si algo se ha movido desde que la cuenta ha terminado y hasta que empiece la siguiente. Los segundos que se pasa comprobando esto tienen que ser aleatorios y en el vídeo deberéis decir cuántos son. Si detecta que algo se mueve, encenderá el LED de color rojo y el participante volverá a la salida. Si el participante llega hasta el sensor de ultrasonidos, o muy cerca de él, por ejemplo, una distancia menor de 5 cm, mientras este está cantando (claramente, en varias iteraciones), entonces ganará y el LED mostrará un estallido de colores.

El juego se reiniciará tras perder la partida tras varios segundos.

### (Juego2.11) Ampliación escondite inglés (0,2 puntos)

*Material necesario: 1 LED RGB, 1 sensor de ultrasonidos, 1 pantalla de segmentos y 1 altavoz.*

Requiere tener implementado el ejercicio opcional 2.9

Añadir al juego 2.9 un altavoz que vaya pitando para marcar la rapidez de la cuenta. De esta manera, se simulará a la persona que va cantando.

### (Mando2.12) Mando de consola (0,2)

*Material necesario: 2 LEDs rojo, 1 joystick, 4 botones.*

En este ejercicio hay que simular un mando de videoconsola. Para ello, tendremos que imprimir por pantalla los datos que vayamos recibiendo del mando, y/o encendiendo el LED, según el siguiente esquema:

- 1 joystick será la cruceta y tendrá 3 posiciones diferentes de velocidad por dirección. Cuanto más cercano al punto de reposo, más lento.
- 1 botón del joystick, será X.
- 2 LEDs rojos: indica si es el jugador 1, 2 o 3. Será aleatorio al encender el mando.
- 1 botón de encendido/apagado del mando.
- 1 botón de pausa.

- 2 botones: A y B.