# Taller 2: Percepción

Introducción a la Robótica Movil

$1^{er}$ cuatrimestre de 2018

A continuación se presentan los ejercicios del taller. Recuerde que puede acceder a las diapositivas de la clase. Asimismo, lea por completo el enunciado, dado que al final del mismo hay aclaraciones importantes que aplican a ambos ejercicios.

## Ejercicio 1: Lectura de *bumper*

Se desea implementar un sistema de protección ante colisiones para un robot móvil. Para ello, el robot posee un sensor tipo *bumper* que se activa al colisionar con un objeto. Para evaluar la factibilidad de la idea, se pide:

1. Armar un circuito en el simulador que incluya un bumper, es decir un pulsador y un Arduino que sea capaz de leerlo
2. Escriba el código para detectar cuándo el robot colisiona con un objeto utilizando interrupciones.
3. ¿Hay otra configuración de la interrupción para la cual el algoritmo siga funcionando? ¿Cuál/Cuales?

## Ejercicio 2: Lectura de sonar

Luego de evaluar el sistema de evasión de colisiones los desarrolladores se dieron cuenta que el bumper solo informaba de colisiones que ya se produjeron. Por esta razón, se decidió incorporar un sensor de ultrasonido que es capaz de medir la distancia a un objeto al frente del robot. Para esta nueva versión del sistema de sensado se pide:

1. Armar el circuito de lectura de un sonar mediante un Arduino Uno.
2. Escribir el código de lectura del sonar. Para ello se deben realizar mediciones en forma continua e informar la distancia registrada por el sensor. Se pide implementar dos versiones:
   (a) Mediante polling
   (b) Mediante interrupciones

### Manejo del sonar

**Polling:**

1. Envío un pulso: `digitalWrite` (pin como `OUTPUT`)
2. Me quedo haciendo *polling* del pin (pin como `INPUT`), mido la diferencia de tiempo entre la subida y la bajada
3. Convierto el valor a distancia

**Interrupciones:**

1. Envío un pulso: `digitalWrite` (pin como `OUTPUT`)
2. Ante la interrupción me guardo el tiempo actual, y mido la diferencia.
3. Ciclo principal: espera a que se dé la segunda interrupción y hacer la conversión a distancia.
   → manejar bien los estados!

### Aclaraciones

El pulso de *trigger* debe tener un ancho de aproximadamente 10 $\mu$s. Para generar el mismo, utilice la función `delayMicroseconds(us)` que genera un delay en la ejecución del tiempo indicado en microsegundos.

Para el cálculo del ancho del pulso de retorno del sonar, utilizar la función `micros()` que devuelve los microsegundos que transcurrieron desde que el Arduino comenzó a ejecutar el programa. El factor de conversión de una diferencia de tiempo $\Delta t$ en $\mu$s a un valor de distancia $d$ en m es

$$d = 0.0001722118895088517 \cdot \Delta t$$

Para procesar interrupciones, utilizar la función

```
attachInterrupt(digitalPinToInterrupt(pin), ISR, mode)
```

que asigna la rutina de atención de interrupciones `ISR` (por *Interrupt Service Routine*) con un evento del `pin` indicado. El parámetro `mode` determina qué evento se espera detectar: cuando hay un valor *low* (`LOW`), cuando el valor cambia (`CHANGE`), cuando el valor pasa de *low* a *high* (`RISING`) o al revés (`FALLING`). Ver documentación más detallada en el siguiente link.

**Importante:** recordar que la RAI ejecuta en cualquier momento, efectivamente interrumpiendo la ejecución del ciclo principal. En principio el compilador no sabe qué variables son afectadas (escritas) en la RAI y es necesario avisarle que no asuma nada sobre cuándo una variable es o no modificada en el ciclo principal (dado que podría introducir alguna optimización y tomar una decisión incorrecta). Para ello, anteponer el *keyword* `volatile` a la declaración de este tipo de variables. Por ejemplo: `volatile int mi_variable`.

## Ejercicio 3: Juntando Todo (Opcional)

Para hacer el sistema más robusto, se desea implementar un dispositivo que pueda medir la distancia a un objeto, pero que también cuente con la capacidad de dar una alerta si el robot colisionó con un obstáculo. Se pide implementar este sistema de sensado en el simulador y desarrollar el código necesario para este dispositivo.