

# **Práctica 4: SPI en Arduino**

---

Departamento de Automática  
Universidad de Alcalá

# Controlador SPI

- Controlador SPI integrado en los microcontroladores AVR
- Configuración Maestro/Esclavo
- Tasa de bps programable (divisor de frecuencia)
- Interfaz conectada a los pines de salida de Arduino:
  - Arduino Uno: 11 (MOSI), 12 (MISO), 13 (SCK), 10 (SS)
  - Arduino Mega: 51 (MOSI), 50 (MISO), 52 (SCK), 53 (SS)

# Registros SPI: SPCR

- SPI Control Register

SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0
7	6	5	4	3	2	1	0

- **SPIE:** SPI Interrupt Enable (Enabled = 1, Disabled = 0)
- **SPE:** SPI Enable (Enabled = 1, Disabled = 0)
- **MSTR:** Master/Slave Select (Master = 1, Slave = 0)
- **SPR1 & SPR0:** SPI Clock Rate Select
  - (1, 0): osc/64

# Registros SPI: SPSR y SPDR

- SPI Status Register

SPIF	WCOL						SPI2X
7	6	5	4	3	2	1	0

- **SPIF:** SPI Interrupt Flag. Se pone a 1 cuando finaliza una transferencia. Se vuelve a poner a 0 cuando se leen los registros SPSR y SPDR

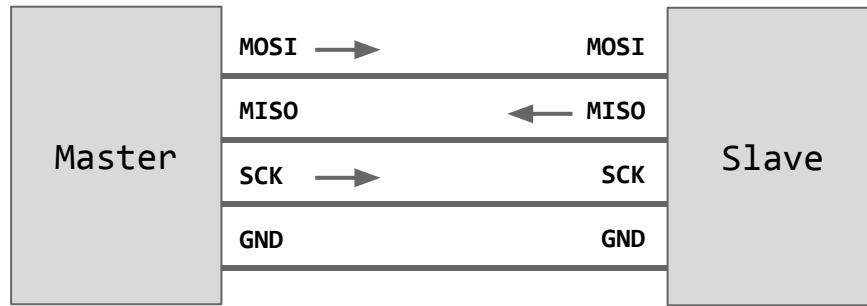
- SPI Data Register

MSB							LSB
7	6	5	4	3	2	1	0

- Registro de datos empleado para envío y recepción

# Conexión entre Arduinos

- Ejercicio propuesto: conexión entre un maestro (Uno) y un esclavo (Mega)



# Conexión entre Arduinos: Maestro

- Inicialización del programa:
  - Configuración de los pines de entrada/salida
  - Configuración de los registros SPI (Master, SPI enabled, Freq = osc/64)
  - Salidas iniciales (SCK → LOW, MOSI → LOW, SS → HIGH)
  - Configuración de la puerta serie
- Bucle principal con un periodo de 1 segundo:
  - Inicio de transmisión (SS → LOW)
  - Envío de datos (Dato → SPDR). El dato es un contador de 8 bits.
  - Espera activa a fin de transmisión (SPSR[SPIF] = 1)
  - Lectura de datos recibidos (SPDR → Dato)
  - Fin de transmisión (SS → HIGH)
  - Impresión del dato recibido por la puerta serie
  - Incremento del contador

# Conexión entre Arduinos: Esclavo

- Inicialización del programa:
  - Configuración de los pines de entrada/salida
  - Configuración de los registros SPI (Slave, SPI enabled, SPI interrupt enabled)
  - Carga del dato inicial (0xFF → SPDR)
- Rutina de atención a la interrupción SPI → ISR (SPI\_STC\_vect) { ... }
  - Lectura del dato recibido (SPDR → Dato)
  - Decremento del dato recibido
  - Preparación del dato para la siguiente transmisión (Dato → SPDR)



© Departamento de Automática. Universidad de Alcalá. Este documento se ha publicado con la licencia Creative Commons Attribution Share-Alike 4.0 (international): <https://creativecommons.org/licenses/by-sa/4.0/>