

## 4 – Dispositivos de E/S

*Dispositivos, puertos y registros de E/S*

*Dispositivos de E/S paralelo*

*Conversión A/D. Pulse Width Modulation*

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

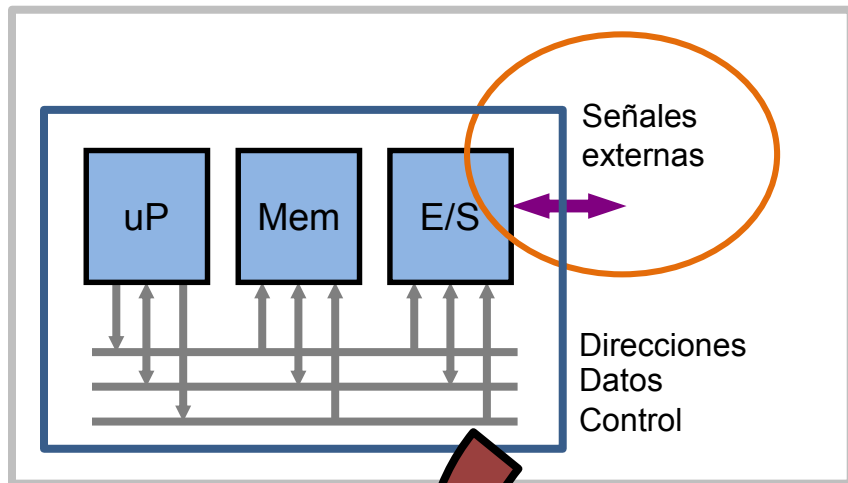
---

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

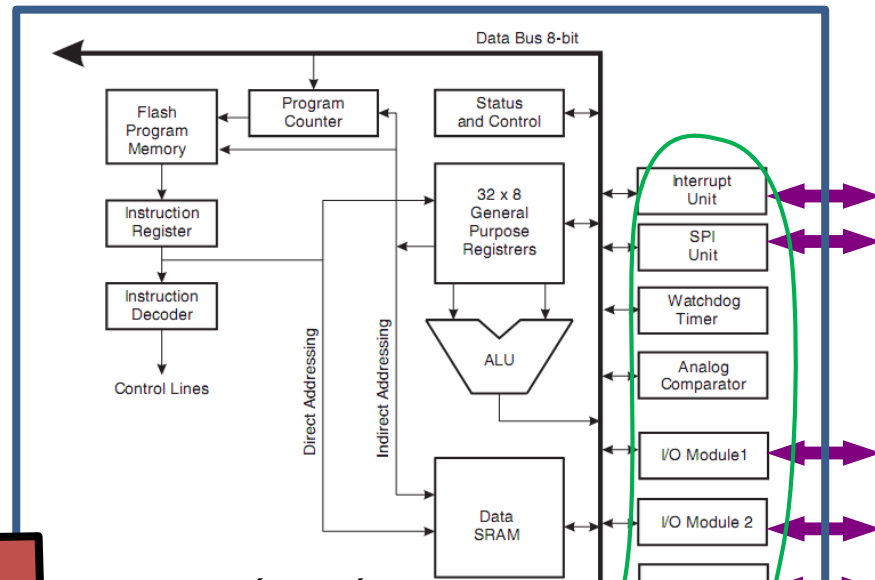


# Dispositivos de Entrada/Salida

- Los dispositivos de E/S permiten leer y escribir las señales externas



En el caso de la familia AVR de Atmel, la funcionalidad y el número de dispositivos de E/S depende del modelo de MCU en particular (la CPU es la misma para toda la familia AVR)



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

---

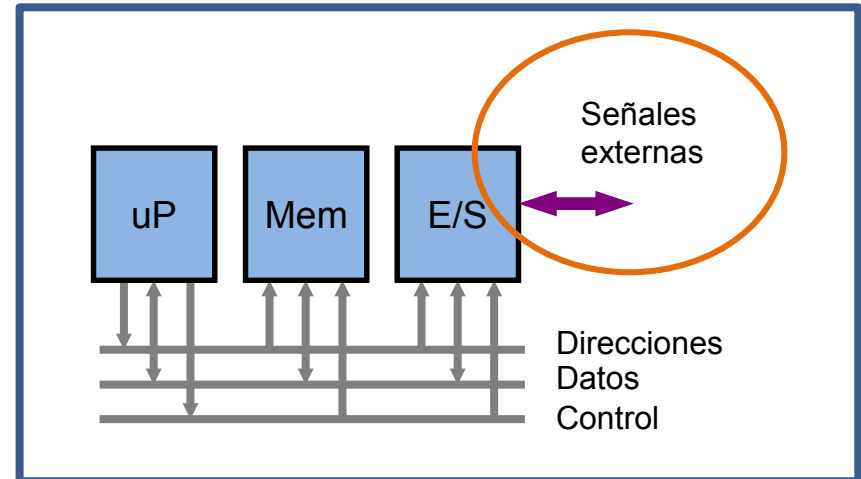
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

Cartagena99

# Puertos y registros de Entrada/Salida

## Puerto de E/S

- ❑ Conjunto de señales externas con una funcionalidad común y una circuitería (dispositivo de E/S) que permite leerlas y/o controlarlas.
- ❑ Diferentes tipos: E/S paralelo, E/S serie, E/S analógica, ...
- ❑ Los puertos de E/S pueden tener asociadas diferentes funcionalidades (según se configuren)



## Registros de E/S

- ❑ Registros asociados a los puertos de E/S. Cada puerto de E/S puede tener asociados uno o varios registros que permiten controlar el funcionamiento del puerto.
- ❑ Los registros de E/S se acceden mediante instrucciones de lectura/escritura de forma análoga a como se accede a las posiciones de memoria.

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# Ejemplo: Puertos de E/S en el AtMega168

- Los puertos de E/S en los AVR pueden operar en varios modos (como en casi todos los uC)

## Puertos de E/S

### Puerto B (PB)

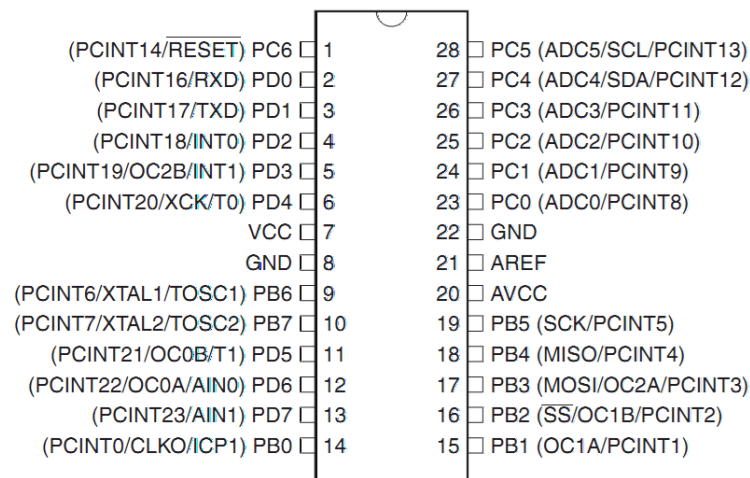
- Entrada/Salida paralelo (se verá)
- Generación de señales PWM (se verá)
- Otros (SPI, interrupciones, capturas..)

### Puerto C (PC)

- Entrada/Salida paralelo
- Convertidor A/D (se verá)
- Otros (Interrupciones)

### Puerto D (PD)

- Entrada/Salida paralelo
- Puerto serie



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

---

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

Cartagena99

# Puertos de E/S paralelo

## Puertos de E/S paralelo

- Puerto de E/S en el que se controlan (leen o escriben) varias señales externas al mismo tiempo.
- Pueden ser de entrada, salida, o configurables por el programa.
- Son los puertos más comunes.
- El tamaño (nº de bits) del puerto suele coincidir con el de la palabra del uP (8 bits, en el AVR)

## Registros asociados

- Registros de datos:** registro conectado a las señales de E/S.
  - Si un bit del puerto está configurado como entrada un nivel de tensión alto (Vcc) en su pin correspondiente se leerá como un '1', y un nivel bajo (Gnd) se leerá como un '0'.
  - Si un bit está configurado como salida, un '1' fijará un valor alto (Vcc) en su pin correspondiente, y un '0' pondrá un valor bajo (Gnd).
  - Escribir sobre un bit configurado como entrada no tiene ningún efecto.
  - En algunos uC, los registros de datos de entrada (para leer) y de salida (para escribir), son distintos (como en el caso de los AVR que usamos)

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

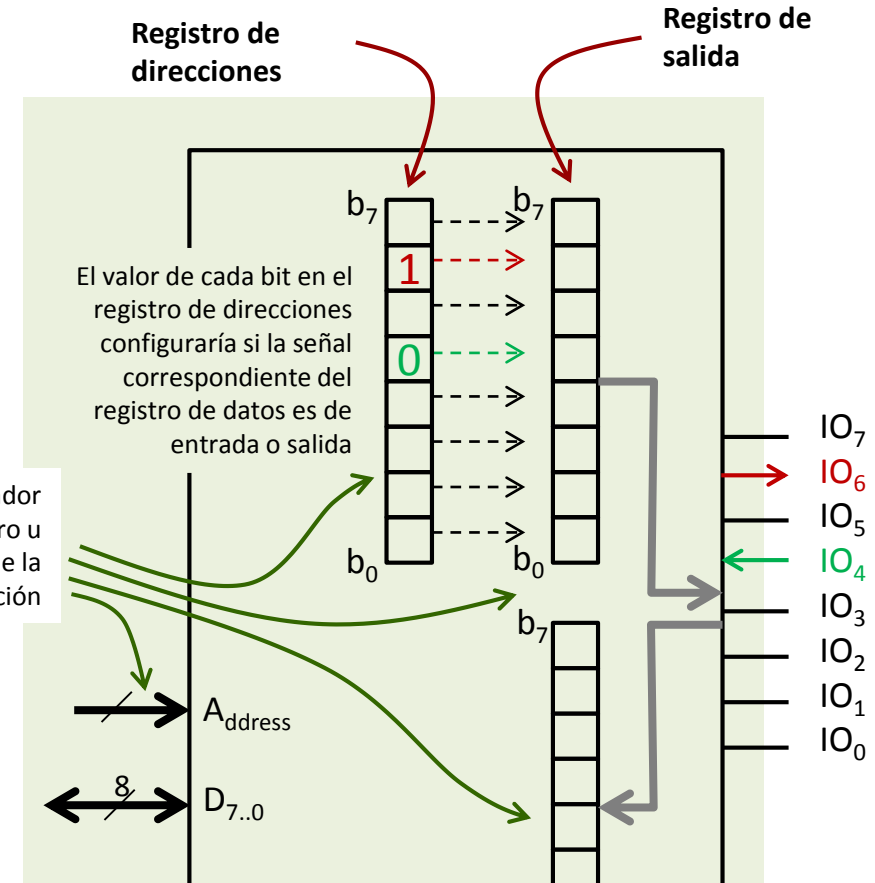
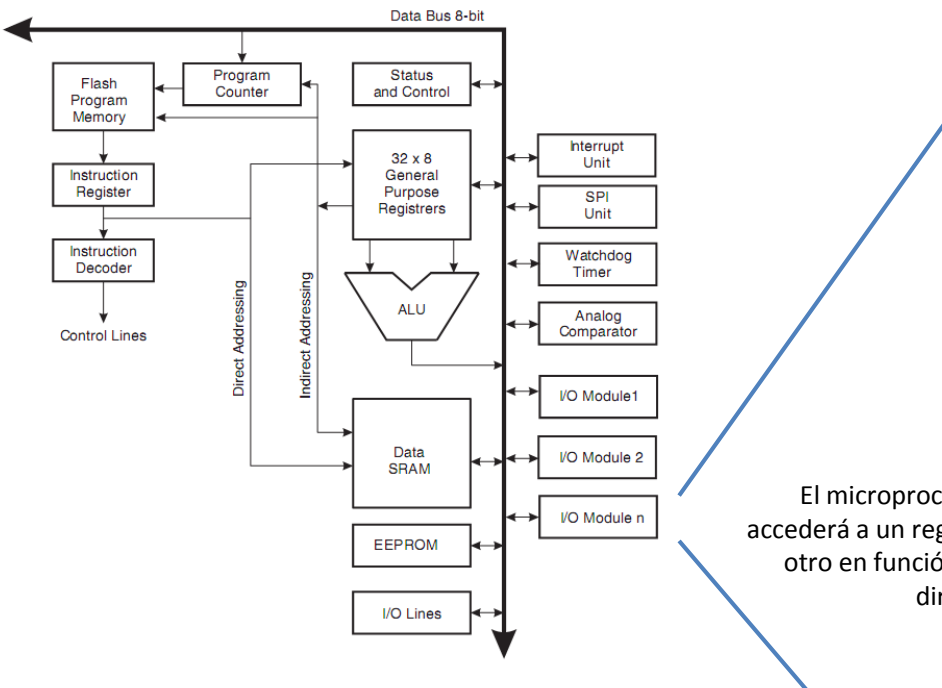
- - -

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70



# Puertos de E/S paralelo en el AtMega168

- Dispone de tres puertos de E/S paralelo (B, C y D)



El microprocesador accederá a un registro u otro en función de la dirección

- ❑ **Registro de entrada:** registro de lectura de puerto
  - Un bit se lee como un '1' si hay un '1' (Vcc) en su pin correspondiente
  - Un bit se lee como un '0' si hay un '0' (Gnd) en su pin correspondiente
- ❑ **Registro de direcciones** (Data Direction Register):

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
 LLAMA O ENVÍA WHATSAPP: 689 45 44 70

---

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
 CALL OR WHATSAPP: 689 45 44 70

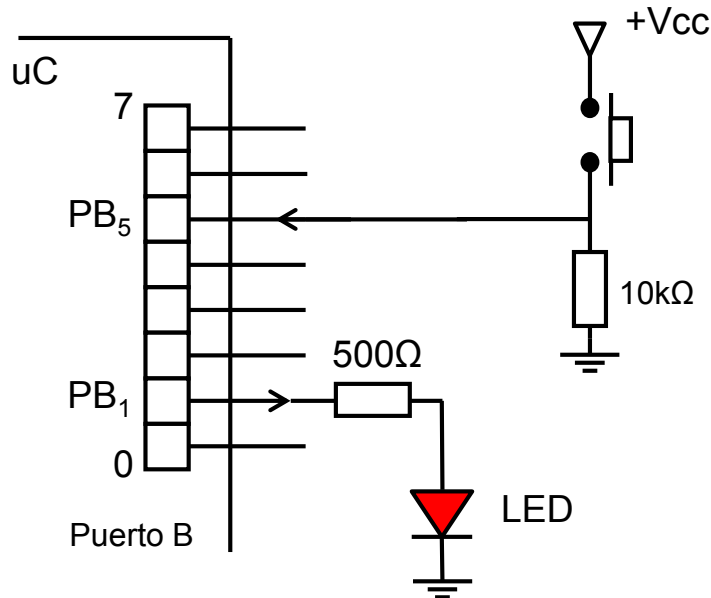
Cartagena99

Registro de entrada



# Ejemplo: Encender y apagar un LED con un pulsador

- ¿Cómo podríamos hacer un sistema que encienda las luces cuando anochece?



```
unsigned char boton;  
unsigned char ledState = 0;  
  
// Primero planteo la funcionalidad, luego ya veremos  
// cómo accedo al hardware  
int main() {  
    // Inicialización  
    InicializaES();  
  
    while (true) { // Bucle infinito  
        boton = leerBoton(); // leo el estado del botón  
        if (boton != 0) { // boton pulsado → actuar  
            if (ledState == 0) {  
                setLed(1);  
                ledState = 1;  
            } else {  
                setLed(0);  
                ledState = 0;  
            }  
        }  
    }  
}
```

Planteamiento de alto nivel

- Configurar las señales de entrada y salida
- Leer el bit del puerto que me interesa

1

?? Si comprobásemos el funcionamiento del sistema, se observaría que mientras se mantiene el botón pulsado el led luciría levemente. Por otro lado, veríamos que el led se enciende y apaga de forma

3

Cartagena99

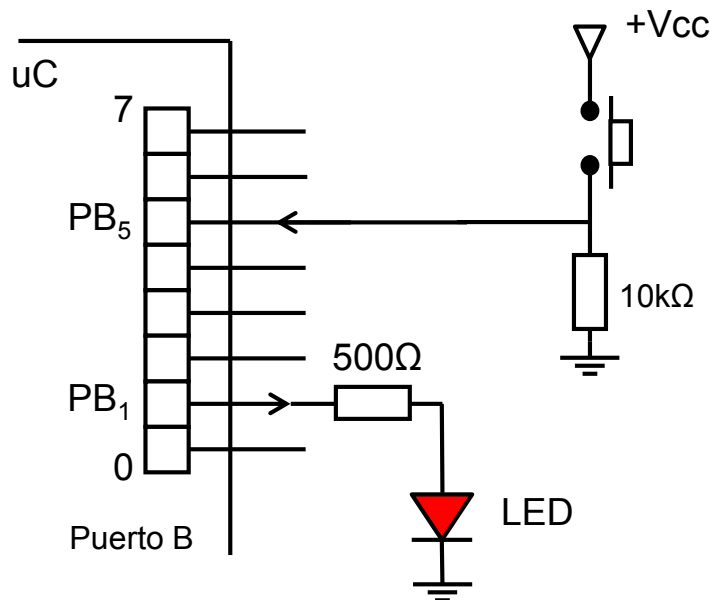
CLASES PARTICULARES, TUTORIAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

---

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

# Ejemplo: Encender y apagar un LED con un pulsador

- Implementando las funciones de E/S, (acceso al hardware)



```
unsigned char boton;  
unsigned char ledState = 0;  
  
// Primero planteo la funcionalidad, luego ya veremos  
// cómo accedo al hardware  
int main() {  
    // Inicialización  
    InicializaES();  
  
    while (true) { // Bucle infinito  
        boton = leerBoton(); // leo el estado del botón  
        if (boton != 0) { // boton pulsado → actuar  
            if (ledState == 0) {  
                setLed(1);  
                ledState = 1;  
            } else {  
                setLed(0);  
                ledState = 0;  
            }  
        }  
    }  
}
```

```
void InicializaES() {  
    // Configura el pin 1 el puerto B para que sea salida  
}
```

```
void setLed(char state) {  
    // Modifica el valor del pin 1 del puerto B  
    // en función de 'state'
```

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

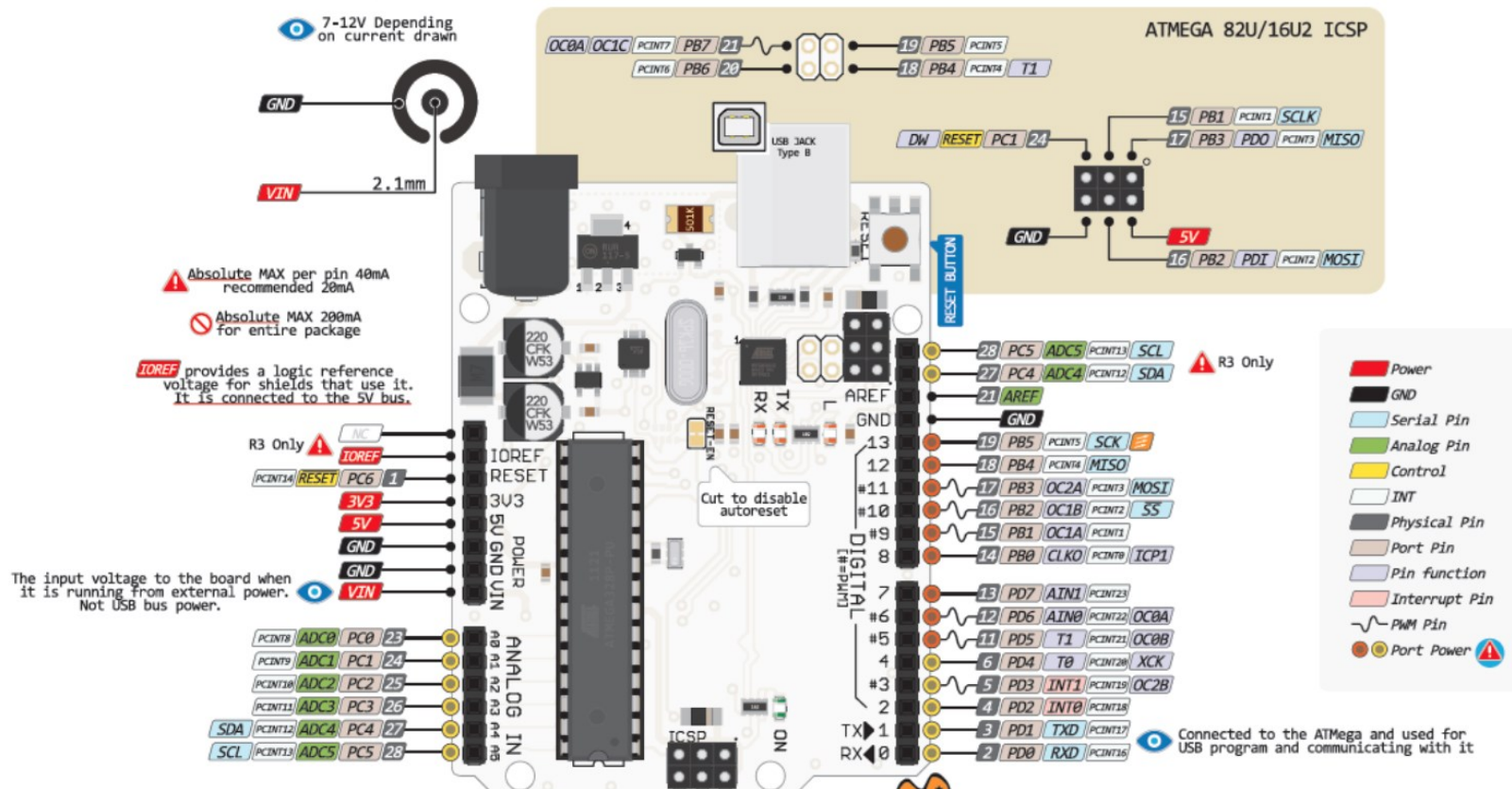
---

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70



# Ejemplo: Encender y apagar un LED con un pulsador

## UNO PINOUT



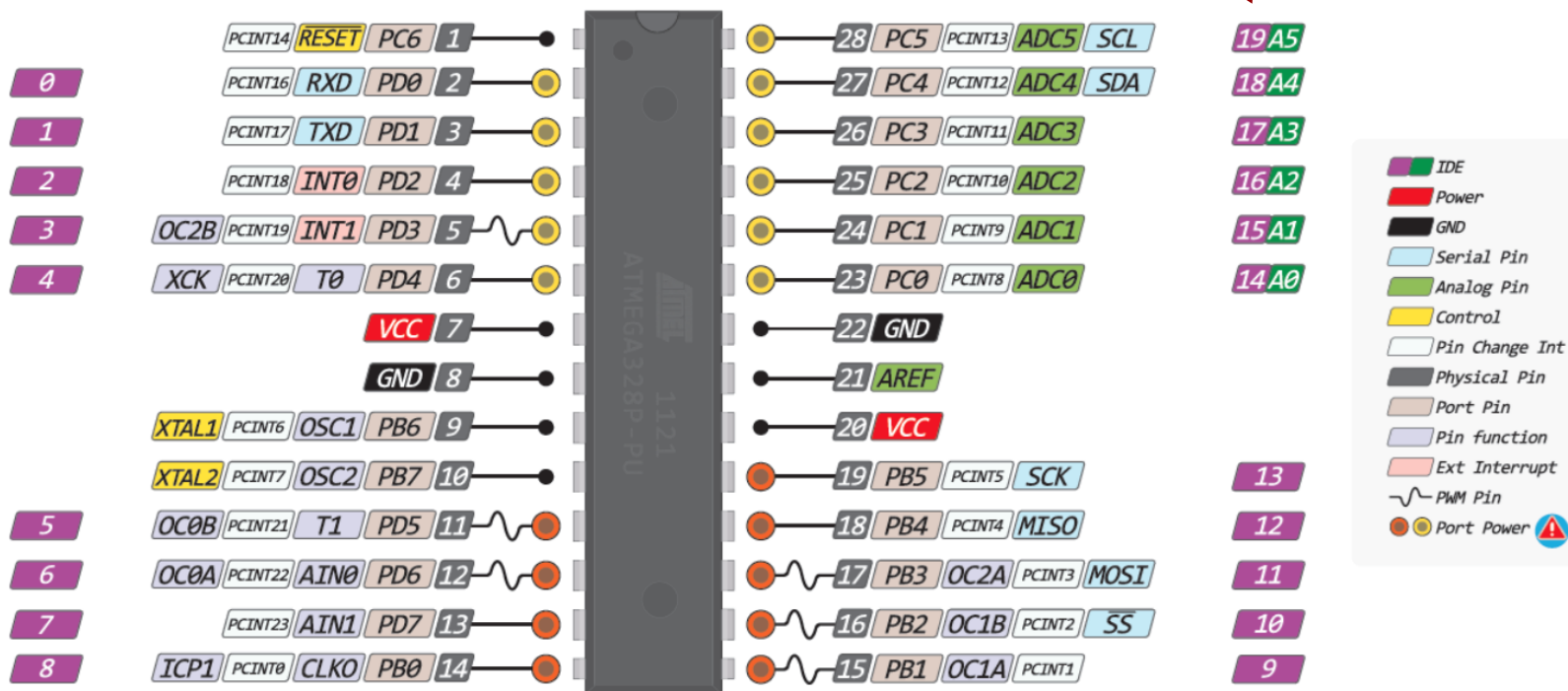
CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

Cartagena99

# Ejemplo: Encender y apagar un LED con un pulsador

## ATMEGA328 PINOUT



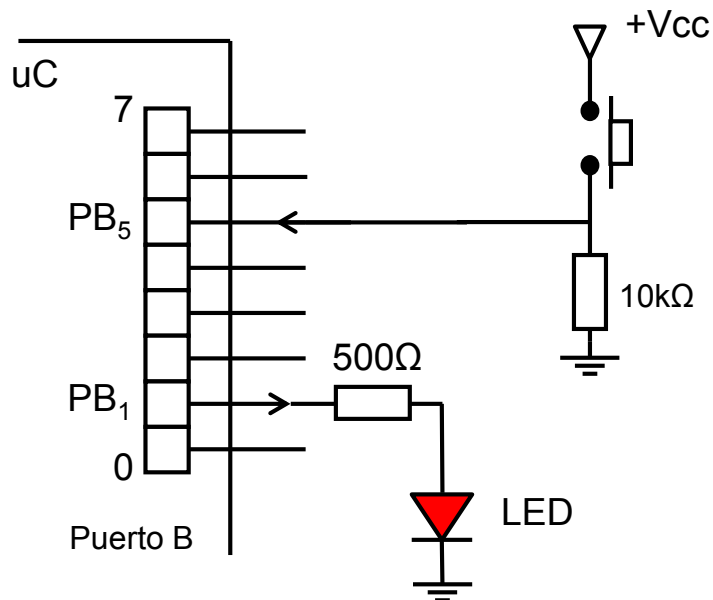
CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
 LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
 CALL OR WHATSAPP: 689 45 44 70

Cartagena99

# Ejemplo: Encender y apagar un LED con un pulsador

- Lo mismo con la **API** de Arduino



```
unsigned char boton;
unsigned char ledState = 0;

void setup() { // esta función sólo se ejecuta una vez
  pinMode( 9, OUTPUT ); // Configura los pines
  pinMode( 13, INPUT );
}

void loop() { // Esta función se ejecuta continuamente
  boton = digitalRead(13);
  if (boton != 0) { // boton pulsado → actuar
    if (ledState == 0) {
      digitalWrite(9, HIGH);
      ledState = 1;
    } else {
      digitalWrite(9, LOW);
      ledState = 0;
    }
  }
}

//////////////////////////////////// esto no lo haríamos nosotros, ya que
int main() { // el entorno de arduino incluye la
  setup(); // función main() automáticamente, así
  while (true) { // que nosotros solo tenemos
    loop(); // que hacer las funciones de setup(),
  }
}
```

Nota: En la placa de Arduino, el pin 9 de la placa se corresponde con el pin PB1 del Atmega, y el pin 13 de la placa se

# Cartagena99

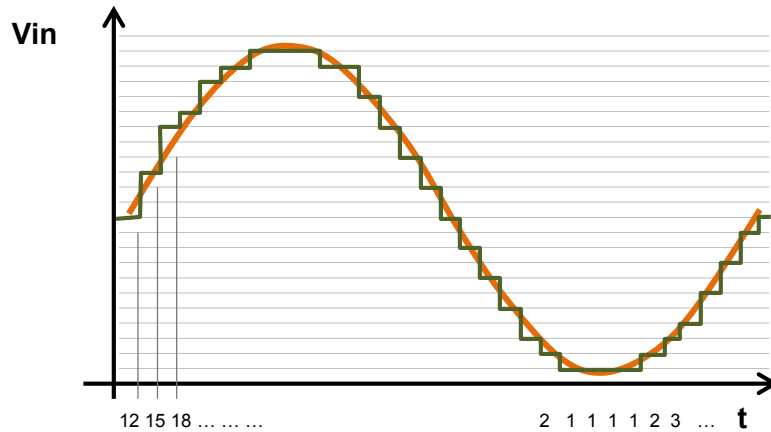
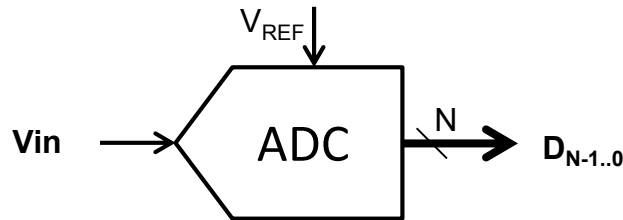
CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

---

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

# Convertidor Analógico-Digital (ADC)

- Digitaliza una señal continua (analógica) en  $2^N$  niveles.



(PCINT14/RESET) PC6	1	28	PC5 (ADC5/SCL/PCINT13)
(PCINT16/RXD) PD0	2	27	PC4 (ADC4/SDA/PCINT12)
(PCINT17/TXD) PD1	3	26	PC3 (ADC3/PCINT11)
(PCINT18/INT0) PD2	4	25	PC2 (ADC2/PCINT10)
(PCINT19/OC2B/INT1) PD3	5	24	PC1 (ADC1/PCINT9)
(PCINT20/XCK/T0) PD4	6	23	PC0 (ADC0/PCINT8)
VCC	7	22	GND
GND	8	21	AREF
(PCINT6/XTAL1/TOSC1) PB6	9	20	AVCC
(PCINT7/XTAL2/TOSC2) PB7	10	19	PB5 (SCK/PCINT5)
(PCINT21/OC0B/T1) PD5	11	18	PB4 (MISO/PCINT4)
(PCINT22/OC0A/AIN0) PD6	12	17	PB3 (MOSI/OC2A/PCINT3)
(PCINT23/AIN1) PD7	13	16	PB2 (SS/OC1B/PCINT2)
(PCINT0/CLKO/ICP1) PB0	14	15	PB1 (OC1A/PCINT1)

- Características generales:

$$\text{Resolución } q = \frac{V_{REF}}{2^N}$$

Cartagena99

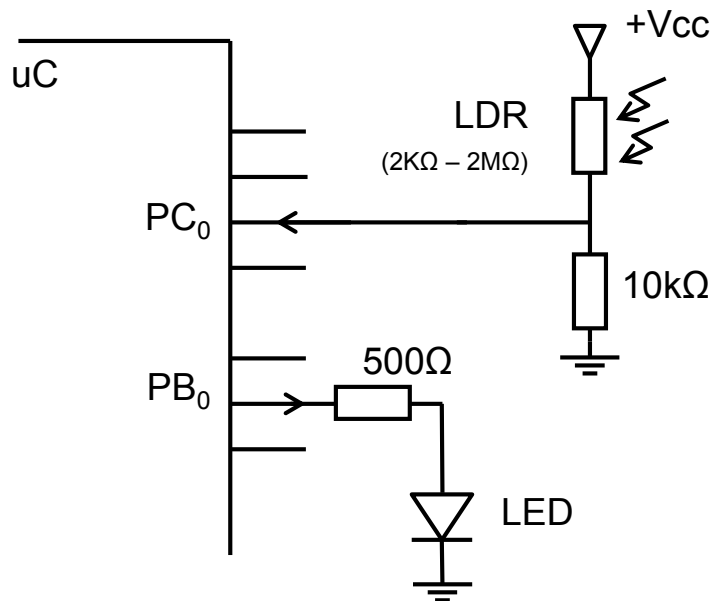
CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

---

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# Ejemplo: Controlar un LED con la luz ambiente

- ¿Cómo podríamos hacer un sistema que encienda las luces cuando anochece?



```
...
int analogVal;
unsigned char led = 0;
...
configurar PB0 como salida (*);
configurar ADC para leer el valor analógico en PC0 (*);
...

while (true) { // bucle infinito
  analogVal = valor devuelto por el ADC (0 - 1023) (*);
  if (analogVal < 100) { // 100 sería un valor obtenido
                        // al anochecer, experimentalmente
    encender led (*);
  } else {
    apagar led (*);
  }
}

... (*) se verá luego cómo se haría usando Arduino
```

- 1 Configurar el convertidor A/D y la señal de salida
- 2 Leer el valor del convertidor A/D
- 3 Decidir en función del valor

?? Al comprobar el funcionamiento del sistema, se observa que al anochecer y al amanecer la luz vibra (se enciende y apaga de forma irregular) durante un rato.

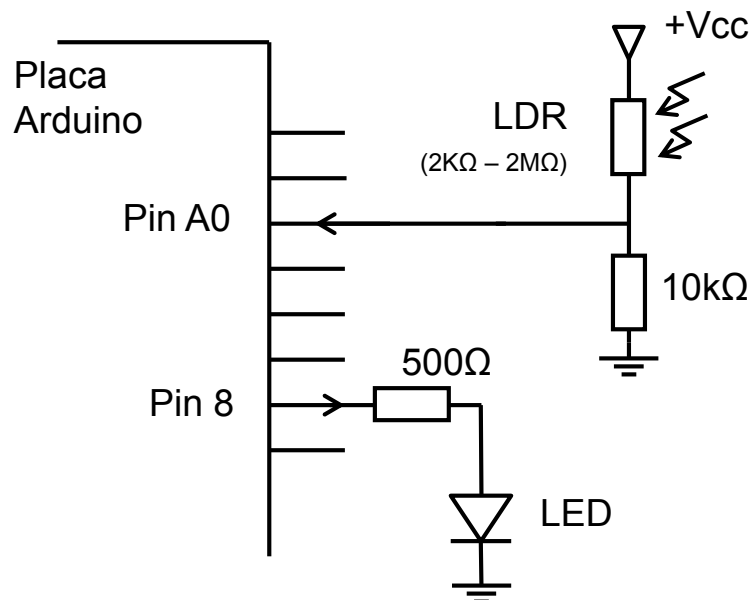
CLASES PARTICULARES, TUTORIAS TECNICAS ONLINE  
LLAMA O ENVIA WHATSAPP: 689 45 44 70

---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

Cartagena99

# Ejemplo: Controlar un LED con la luz ambiente

- ¿Cómo podríamos hacer un sistema que encienda las luces cuando anochece utilizando la placa y la API\* de Arduino?



```
void setup() { // esta función sólo se ejecuta una vez
  pinMode( 8, OUTPUT ); // Configura los pines
  pinMode( A0, INPUT ); // Puede no hacer falta, ya que
                        // tras el reset las E/S suelen
                        // estar como entradas
}

void loop() { // Esta función se ejecuta continuamente

  int analogVal = analogRead( A0 ); // valores de 0 a 1023

  if (analogVal < 100) { // 100 sería un valor obtenido
                        // al anochecer experimentalmente
    digitalWrite( 8, HIGH );
  } else {
    digitalWrite( 8, LOW );
  }
}
```

Nota: En la placa de Arduino, el pin A0 de la placa se corresponde con el pin PC0 del AtMega, y el pin 8 de la placa se

Cartagena99

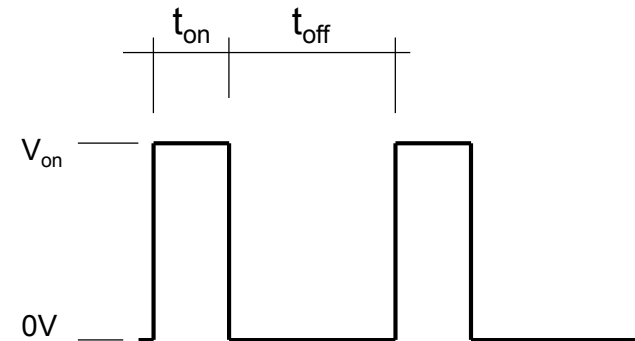
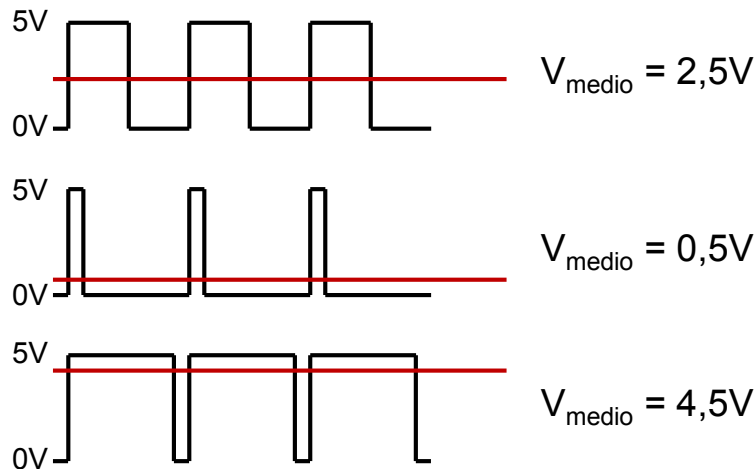
CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

---

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

# Modulación por ancho de pulso (PWM)

- Para generar una señal «analógica», la mayoría de los uC no incluyen un convertidor digital-analógico (DAC), sino que utilizan el valor medio de una señal digital periódica con ancho de pulso variable



$$V_{\text{medio}} = V_{\text{on}} * t_{\text{on}} / (t_{\text{on}} + t_{\text{off}}) = V_{\text{on}} * d$$

$$d \text{ (duty cycle)} = t_{\text{on}} / (t_{\text{on}} + t_{\text{off}})$$

% del tiempo que la señal esta encendida



- Se usa muy a menudo para el control de motores y luces, o en generación de audio (o a veces también como método de comunicación, p.ej: radiocontrol).

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

---

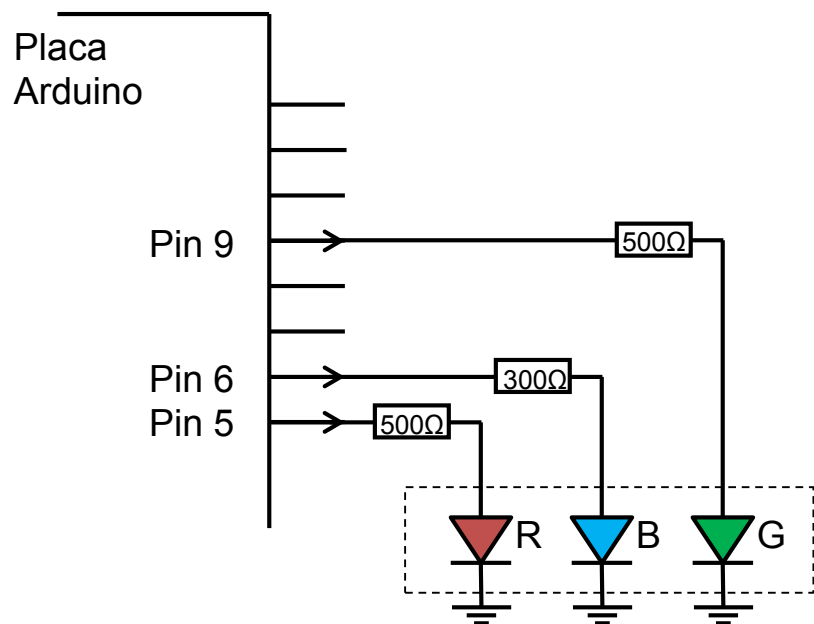
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

Cartagena99



# Ejemplo: Controlar un LED RGB

- Un led RGB está compuesto de tres leds de colores rojo (R), verde (G) y azul (B). El color final depende de la intensidad de cada uno



```
#define LED_R 5
#define LED_G 9
#define LED_B 6

void setup() {
  pinMode(LED_R, OUTPUT);
  pinMode(LED_G, OUTPUT);
  pinMode(LED_B, OUTPUT);
}

void setColor(int r, int g, int b) {
  analogWrite(LED_R, r); // Valores de 0 a 255
  analogWrite(LED_G, g);
  analogWrite(LED_B, b);
}

// cambia el color de forma progresiva (solo la
// componente del rojo)
int inc = 1;
int r = 0;
int loop() {
  setColor(r, 200, 100);
  r = r + inc;
  if (r == 255) inc = -1;
```

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

---


ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

Cartagena99

pinos 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11 de la placa), y su frecuencia es fija (490Hz aprox)



# Otros dispositivos de E/S típicos

Tipo	Sirven para	Ejemplos (estándares)
E/S serie	Comunicaciones	UART, USART, USB, I2C, FireWire, CAN, etc.
Temporizadores y contadores	Control (en general) Datado (en particular)	
Perro guardián (watchdog)	Control de la ejecución correcta de un programa	
Controladores específicos	Discos duros, disquetes, etc.	SCSI, IDE, SATA, VGA, ...



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

---

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

Cartagena99