



**Universidad**  
Zaragoza



Escuela de  
Ingeniería y Arquitectura  
**Universidad** Zaragoza

Proyecto final de carrera

Ingeniería Técnica Industrial. Esp: Electrónica Industrial

Curso 2014-2015

# **DRIVER PARA MATRIZ DE LEDS RGBW**

**César Lambán Cosculluela**

Febrero 2014

Director: Luis Porta Royo

Universidad de Zaragoza

# **DRIVER PARA MATRIZ DE LEDS RGBW**

**MEMORIA**



**Índice:**

1-Objeto.....	2
2-Alcance .....	2
3-Antecedentes.....	2
3.1. Antecedentes históricos.....	2
4-Normas y referencias.....	4
4.1 Bibliografía.....	4
4.1.1 Linkografía .....	4
4.2 Programas de cálculo utilizados.....	5
5-Definiciones y abreviaturas .....	5
5.1 Abreviaturas.....	5
5.2 Definiciones.....	6
6-Requisitos de diseño.....	8
6.1 Legislación, reglamentación y normativa aplicables .....	8
6.2 Requisitos de conexión .....	8
6.3 Requisitos de la matriz de leds.....	8
6.4 Requisitos físicos .....	9
7-Análisis de soluciones .....	9
7.1 Alternativa estudiada.....	9
7.2 Solución final adoptada .....	10
8-Resultados finales .....	11
8.1 Diagrama de bloques .....	11
8.2 Matriz de leds .....	11
8.3 Arduino.....	12
8.4 PC .....	14
8.3 Driver .....	15
9-Planificación.....	16
9.1 Etapas .....	16
9.2 Diagrama de Gantt.....	17
10-Orden de prioridad de los documentos básicos.....	18



## 1-Objeto

Se pretende desarrollar un driver de potencia capaz de regular la corriente de una matriz de leds RGBW. El control de este driver, así como la intensidad de la iluminación de los leds de la matriz, se realizará mediante un aplicación de Windows.

## 2-Alcance

El alcance de este proyecto se dirige a las prácticas de la asignatura de "Iluminación y Domótica", en el grado de Ingeniería Eléctrica.

## 3-Antecedentes

### 3.1. Antecedentes históricos

#### LA ELECTRÓNICA DE POTENCIA

La electrónica de potencia tiene sus inicios en el año 1900, con la introducción del rectificador de arco de mercurio. Luego aparecieron, gradualmente, el rectificador de tanque metálico, el rectificador de tubo al alto vacío de rejilla controlada, el ignitrón, el fanotrón y el tiratrón. Estos se aplicaron al control de la energía hasta la década de 1950.

En 1948 se inicia la primera revolución electrónica con la invención del transistor de silicio en los Bell Telephone Laboratories por los señores Bardeen, Grattain y Shockley. Otros de los grandes inventos fue la del transistor de disparo pnpn, que se definió como tiristor o rectificador controlado de silicio (SCR por sus siglas en inglés.) la segunda revolución electrónica fue en 1958 con el desarrollo del tiristor comercial por General Electric Company. Ese fue el principio de la nueva era de



la electrónica de potencia, hasta la fecha se han introducido diversos dispositivos semiconductores de potencia y técnicas de conversión. La revolución de la electrónica de potencia nos está dando la capacidad de dar forma y controlar grandes cantidades de energía con una eficiencia cada mayor.

### CONTROL DMX (Digital Multiplex)

DMX512, abreviado como DMX, es un protocolo electrónico utilizado en Luminotecnia para gestión y control de iluminación espectacular permitiendo la comunicación entre los equipos de control de luces y las propias fuentes de luz.

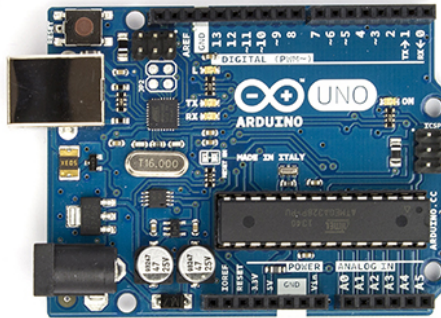
Aparece como la solución al problema de incompatibilidad que existía entre marcas por la utilización de protocolos propios del fabricante, lo cual obligaba a tener un control de manejo específico por cada marca de luces que se tenía.

DMX512 pronto se convirtió en el protocolo preferido no sólo para controladores de enlace y dimmers, sino también para controlar aparatos de iluminación como scanners y cabezas móviles, y dispositivos de efectos especiales como máquinas de humo.

### ARDUINO

Arduino es una plataforma de prototipos electrónica de código abierto (open-source) basada en hardware y software flexibles y fáciles de usar.

Arduino presenta una gran variedad de placas con distintos montajes y posibilidades. Para el desarrollo de este proyecto hemos utilizado la placa Arduino Uno.



**Ilustración 1, placa Arduino UNO**

Arduino recibió una mención honorífica en la sección Digital Communities del Ars Electronica Prix en 2006.

## **4-Normas y referencias**

### **4.1 Bibliografía**

Los libros y documentación escrita que han sido consultados son los siguientes:

- TORRES, Manuel y TORRES, Miguel Ángel. Diseño e ingeniería electrónica asistida con Protel DXP. Editorial Ra-Ma.
- BONO, Antonio y MARTÍN, Bonifacio. Apuntes de microprocesadores e instrumentación electrónica. Año 2011-2012
- FERNÁNDEZ, Vicente. Apuntes de electrónica analógica. Año 2010-2011
- TORRES, Miguel Ángel. Apuntes de oficina técnica. Año 2011-2012

#### **4.1.1 Linkografía**

- Digkey <http://www.digkey.com>
- Rs <http://www.es.rs-online.com>
- Arduino <http://www.arduino.cc>
- Microsoft <http://msdn.microsoft.com>



## 4.2 Programas de cálculo utilizados

Los programas utilizados para el diseño del prototipo han sido:

- ALTIUM DESIGNER 13.1.2: herramienta de diseño asistido, para llevar a cabo el diseño de la PCB.
- AUTOCAD 2007: herramienta de diseño asistido, para la realización de planos.
- ADOBE ACROBAT X: permite generar ficheros en formato PDF.
- VISUAL STUDIO 2012: permite crear una aplicación Windows, como interfaz.

## 5-Definiciones y abreviaturas

### 5.1 Abreviaturas

- Micro: microcontrolador
- USB: Universal Serial BUS (comunicación serie)
- DMX: Digital multiplex
- PCB: Placa de circuito impreso (Printed Circuit Board)
- PWM: Modulación por anchura de pulsos (Pulse Width Modulation)



## 5.2 Definiciones

### DIMMER

Sirve para regular la energía en uno o más puntos de luz, con el fin de variar la intensidad de la luz que emiten.

-Modulación por anchura de pulsos: técnica consistente en la variación del ciclo de trabajo de una señal periódica (cuadrada generalmente) para transmitir información a través de un canal o para controlar la cantidad de energía que se envía a una carga. El ciclo de trabajo (D) es la relación existente entre el tiempo en el que la función está en estado alto (t), con el periodo de la función (T):

$$D = \frac{t}{T}$$

### PCB

Placa o tarjeta de circuito impreso, suele ser una superficie plana de un espesor variable y normalmente de forma rectangular o cuadrada; está construida por un material base o sustrato de tipo laminado rígido o flexible que sirve de soporte físico aislante para la colocación y soldadura de los componentes y el trazado de las pistas conductoras de cobre.

### Restriction of Hazardous Substances (RoHS). (R.D. 208/2005)

Es una directiva de la UE (Unión Europea) (2002/95/CE) en la cual se restringe el uso de seis materiales peligrosos en la fabricación de diversos tipos de equipos eléctricos y electrónicos (obliga a los productores a demostrar que sus productos contienen valores de concentración máximos (VCM) de las siguientes sustancias: plomo, mercurio, cadmio, cromo hexavalente, bifenilos policromados (PBB) y éter de binifenilo policromado (PBCE).





## TERMISTOR

Semiconductor térmicamente sensible cuya resistencia varía con la temperatura. Cada sensor tiene una resistencia nominal designada que varía proporcionalmente con la temperatura de acuerdo a una aproximación alineada. Los termistores tienen un coeficiente de temperatura negativo (NTC) o positivo (PTC). El primero de ellos (es el que contiene nuestra matriz de leds) disminuye su resistencia al aumentar la temperatura, y el segundo presenta mayor resistencia conforme aumenta la temperatura. Los PTC suelen usarse como dispositivos de corriente limitada para protección de circuitos (en lugar de fusibles). Por otro lado, los NTC suelen ser utilizados para la medida de temperatura, siendo su principal aplicación en termostatos digitales y en automóviles para visualizar temperaturas de motor.

La ecuación del valor de la resistencia en función de la temperatura, es la siguiente:

$$R = R_0 e^{\beta \left( \frac{1}{T} - \frac{1}{T_0} \right)}$$

Siendo:

R= valor de la resistencia a la temperatura T (K)

R<sub>0</sub> = Valor de la resistencia a la temperatura de referencia T<sub>0</sub> (K).

β= parámetro beta. Temperatura característica del material (K)

El parámetro beta se obtiene de la siguiente ecuación:

$$\frac{1}{T} = \frac{1}{T_0} + \frac{1}{\beta} \ln \frac{R}{R_0}$$



## **6-Requisitos de diseño**

### **6.1 Legislación, reglamentación y normativa aplicables**

Habría que tener en cuenta que todos los componentes y elementos que forman el driver para la regulación de la matriz de leds, cumplen la normativa RoHS, así se garantiza que el sistema es seguro y no supone un riesgo ni para la salud ni para el medio ambiente.

### **6.2 Requisitos de conexión**

Para la correcta utilización de este producto, es necesario que exista una conexión a una tensión continua y estable de 24 V del driver de potencia.

A su vez, también es necesario un PC con el programa específico que nos aporte la interfaz necesaria para controlar dicho driver, y con una conexión USB a la placa Arduino.

Una placa Arduino Uno con el programa correspondiente, capaz de ejecutar dicho control.

### **6.3 Requisitos de la matriz de leds**

Dado que el objeto del proyecto es el control de una matriz de 12 leds Lumispot K-12, es necesario que nuestro driver entregue las señales adecuadas.



COLOR CANAL	POTENCIA MÁX.	CORRIENTE MÁX. (mA)
Rojo	7.56 W	700
Verde	11.16 W	1000
Azul	11.16 W	1000
Blanco	11.16W	1000

**Tabla 1, valores máximos LumiSpot K2**

## 6.4 Requisitos físicos

Dado que va a ser utilizado en las prácticas de una asignatura, nos interesa que la PCB presente unas medidas lo más reducidas posibles.

## 7-Análisis de soluciones

Una de las partes importantes de este proyecto ha sido la búsqueda del control del driver, y por tanto del regulador MIC3203. Este regulador nos proporciona una salida PWM con la corriente regulada y demandada por la matriz para el control de un canal de la matriz de leds.

### 7.1 Alternativa estudiada

La matriz presenta 4 canales, uno para cada color (rojo, verde, azul y blanco respectivamente). Cada canal supone una señal PWM distinta, con el fin de poder controlar cada uno de los canales de manera independiente. Por tanto, necesitamos poder generar 4 señales PWM independientes, para lo cual barajamos la siguiente alternativa hasta dar con la solución final adoptada: utilización de 2 microcontroladores, con 2 señales PWM cada uno.



Esta alternativa, nos aportaba mayor facilidad de uso que la utilización de un microcontrolador con 4 salidas PWM, y una mayor variedad a la hora de buscar un micro que cumpliera nuestras expectativas.

Uno de esos dos micros (master), tendría 2 funciones, además del control de sus salidas PWM:

- Controlaría las salidas PWM del otro micro (slave)
- Debería ser capaz de procesar el programa necesario de control del driver.

A su vez, deberíamos utilizar una pasarela virtual de puerto serie, con la intención de poder comunicarnos con el PC. Esto sumaría otra función a realizar por el micro (master), llevándonos a desechar esta alternativa dada su complejidad

## **7.2 Solución final adoptada**

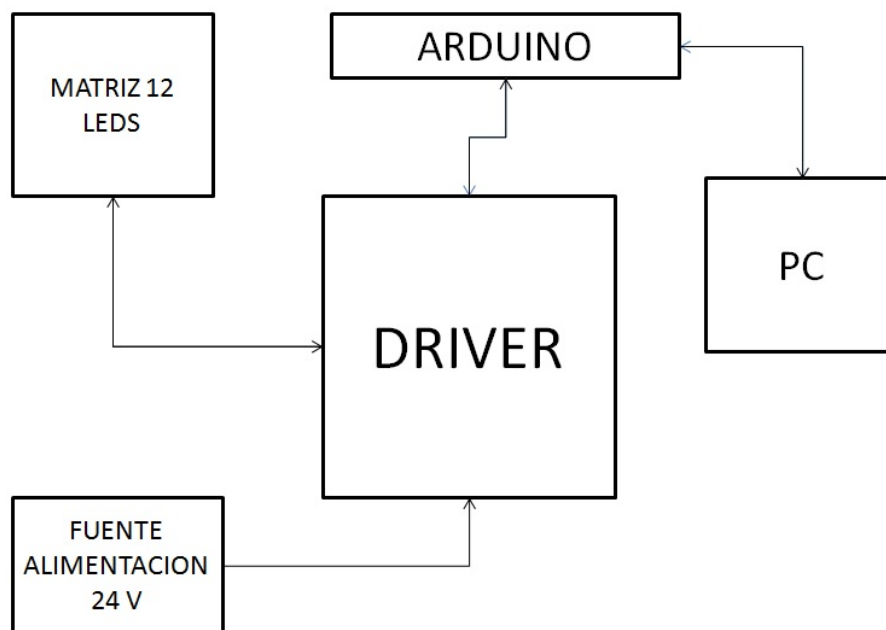
En la búsqueda de información acerca de micros con 4 salidas PWM (lo normal en el mercado es que tengan 2-3 salidas), encontramos información acerca de la plataforma de Arduino.

Esta alternativa nos ofrecía una placa (modelo Arduino UNO) que integraba un micro programable (Atmega328). Tiene 14 pines de salidas/entradas (de los cuales 6 pueden ser usados como salidas PWM), 6 entradas analógicas, conexión USB, un conector ICSP, un conector Jack y un botón de reset. Esta placa nos aportaba la facilidad de conexión al PC mediante el USB, y nos ofrecía la posibilidad de conectarla a una batería o a una fuente de tensión continua (esta última opción para nosotros no resultaba útil, ya que pretendíamos hacer un control desde el PC).



## 8-Resultados finales

### 8.1 Diagrama de bloques



**Ilustración 2, diagrama de bloques**

Para una correcta descripción del producto final, es necesario diferenciar cada uno de los bloques, y explicar por separado.

### 8.2 Matriz de leds

Consiste en una matriz de 12 leds, que contiene 3 leds de cada color (rojo, verde, azul y blanco) dispuestos de una determinada manera. Está diseñada para trabajar a bajas tensiones, y con fuentes de corriente constantes.

Los diodos leds son controlados por señales PWM, y gracias a una resistencia de valor fijo, establece la corriente demandada para cada trío de leds. Los tres diodos



de cada canal, se encuentran en serie, protegidos por un diodo Schottky (colocado en el driver) para evitar su polarización inversa.

Dispone a su vez, de un termistor que nos permite controlar la temperatura de la matriz, pudiendo llegar a regular el consumo y funcionamiento de la misma a través de la temperatura. El termistor adquiere una gran importancia para el control de la temperatura, ya que uno de las principales aplicaciones de este tipo de matrices de leds, es la utilización en iluminaciones ambientales o espectaculares, donde se ven expuestas a muchas horas de funcionamiento. Esto supone un calentamiento de los leds, pudiendo llegar a reducir de manera considerable su vida útil. A fin de mantener una vida útil apropiada, hemos realizado un control de seguridad basado en dos situaciones:

-Temperatura  $>70^{\circ}\text{C}$ : alcanzada esta temperatura, reduciremos a la mitad la intensidad de cada uno de los canales para evitar el sobrecalentamiento.

-Temperatura  $=80^{\circ}\text{C}$ : esta temperatura supone el límite del rango recomendado. Una vez alcanzada, se producirá el apagado de la matriz.



**Ilustración 3, matriz LumiSpot k2**

## 8.3 Arduino

Utilizamos la placa Arduino UNO, con el fin de poder suministrar 4 canales PWM independientes.



Esta placa desempeña dos funciones principales:

- Comunicación entre PC-Driver.
- Adecuación de la señal del termistor.

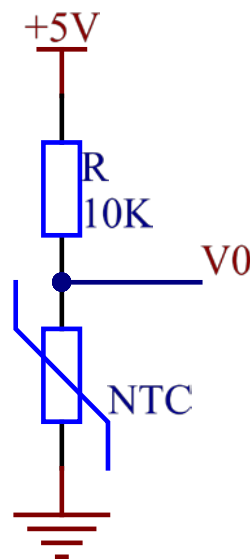
### COMUNICACIÓN PC-DRIVER

Mediante conexión USB enviamos datos desde el PC hasta la placa Arduino UNO.

Cada uno de los canales PWM tiene una resolución de 8 bytes, lo que corresponde a una variación entre 0 y 255. Gracias a esta variación podemos controlar la intensidad de la iluminación de cada canal, estando en 0 apagado, y alcanzando en 255 su máxima intensidad.

### ADECUACIÓN DE LA SEÑAL DEL TERMISTOR

La comunicación PC-DRIVER es bidireccional, ya que a través de Arduino UNO también enviamos datos al PC de la temperatura de la matriz. La matriz presenta un termistor, y para conocer la temperatura es necesario adecuar su variación de resistencia. Para ello, formaremos un divisor resistivo con una resistencia externa de 10K $\Omega$ .



**Ilustración 4, divisor de tensión**



En donde:

$$V_0 = 5V \frac{NTC}{10K + NTC}$$

No conocíamos las características concretas de la NTC que presenta la matriz, por tanto hemos tenido que obtener sus valores de manera experimental. Dicho termistor presenta una resistencia de  $10K\Omega$  a una temperatura de  $25^{\circ}C$ . Posteriormente hicimos otra medida a  $21^{\circ}C$  obteniendo un valor muy cercano a los  $12K\Omega$ .

Dada la estandarización de los valores de las NTC, buscamos una NTC cuyos valores correspondieran a los experimentales.

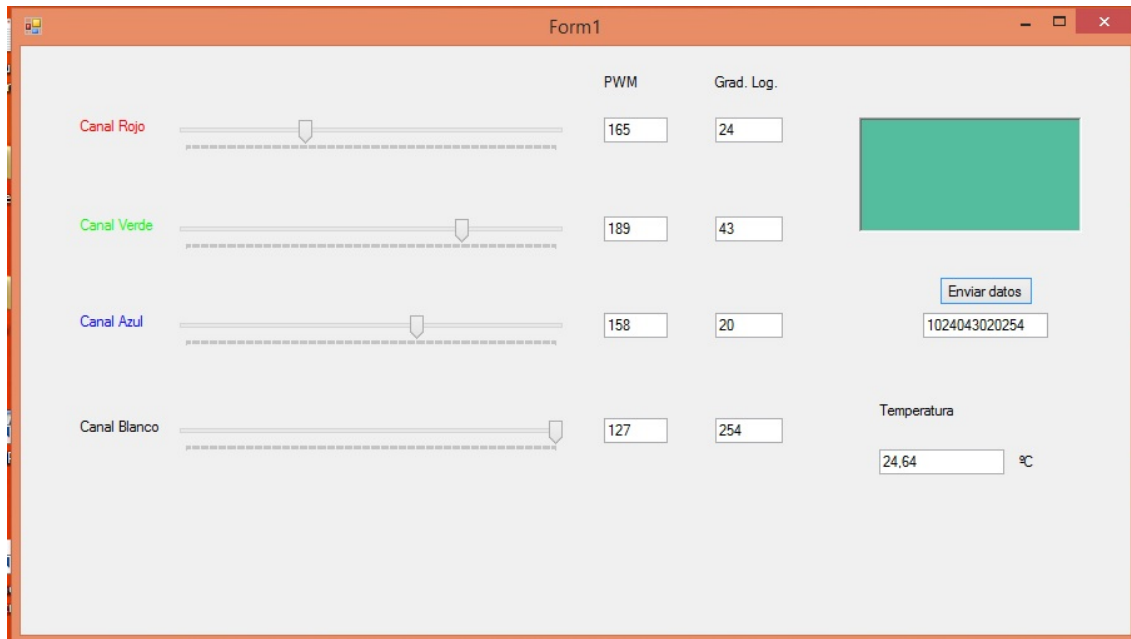
Con todos estos valores, creamos una función en el ARDUINO UNO capaz de obtener la temperatura en función del valor de la resistencia.

## 8.4 PC

Con el fin de poder controlar de manera cómoda e independiente cada uno de los canales, hemos realizado una interfaz con una visualización muy sencilla para su manejo.

Consiste en una aplicación desarrollada a través del programa Visual Studio 2012, cuya ventana de interacción nos permite variar la intensidad de la luminosidad de cada uno de los canales, mostrando en un cuadro el color resultante de la combinación de cada uno de ellos (rojo, verde, azul y blanco). A su vez, aparece la temperatura a la que se encuentra la matriz, para evitar posibles daños debidos al sobrecalentamiento de la misma.





**Ilustración 5, aplicación de Windows**

La aplicación también desarrolla una conversión entre el valor de 0 a 255 de cada canal y su correspondiente valor en una curva de graduación logarítmica con un nivel de potencia en el arco mínimo de 0.1% (tabla norma UNE –EN 60929:2006). Esto nace de la no linealidad en la percepción del ojo de la cantidad de luz. Además de esta conversión, hemos realizado otra operación para aprovechar al máximo la resolución de nuestras salidas PWM (8 bits). Para más detalle, ver el capítulo “Cálculos detallados” del documento Anexos.

### 8.3 Driver

El componente principal del driver es el regulador de corriente MIC3203. Debido a que hay 4 canales (uno para cada color) es necesario establecer un regulador de corriente para cada canal, con el fin de poder controlar de manera independiente la corriente de cada uno de los canales.

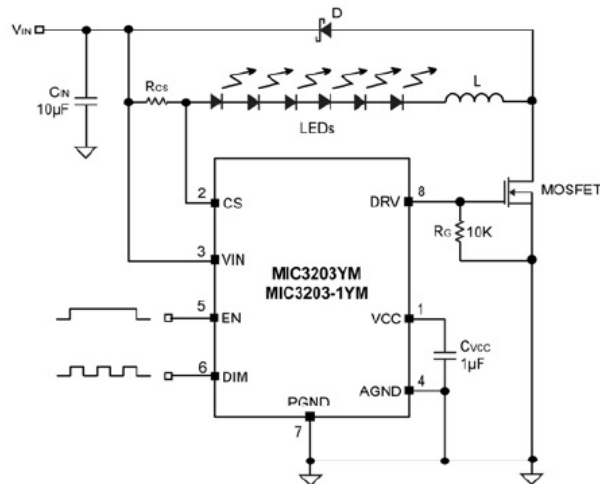


Ilustración 6, MIC3203

El valor de  $R_{cs}$ , depende de la corriente máxima permitida por los leds. Según el fabricante de la matriz, en canal rojo admite una corriente menor a los otros canales, 0.7 A y 1 A respectivamente. Con estos datos, y la tabla facilitada por el fabricante del MIC3203, obtenemos los distintos valores de  $R_{cs}$ .

Para calcular el valor de la  $L$  hemos seguido un procedimiento similar. El fabricante del MIC3203 nos proporciona unas tablas, en las que la elección de la bobina ( $L$ ) depende del valor de  $R_{cs}$ , la corriente máxima admisible de cada canal, y la cantidad de leds que atraviesa. Dado que no existe una tabla con el cálculo realizado para 3 leds, hemos utilizado por aproximación la tabla estudiada con 4 leds. Obteniendo gracias a esta tabla dos valores importantes: el valor de la bobina responsable de establecer un nivel de continua adecuado, y el valor de  $V_{in}$ .

## 9-Planificación

### 9.1 Etapas

- Seguimiento del proyecto
- Búsqueda de información y toma de datos.
- Estudio de posibles soluciones.
- Realización de diseños preliminares.
- Selección y definición del diseño final.
- Búsqueda de componentes.
- Creación prototipo de pruebas.
- Programación software.



- Pruebas prototipo.
- Elaboración documentos básicos.
- Elaboración producto final.

## 9.2 Diagrama de Gantt

Tarea/mes	1	2	3	4	5	6	7	8	9	10	11
Seguimiento del proyecto	■	■	■	■	■	■	■	■	■	■	■
Búsqueda de información y toma de datos	■	■	■	■							
Estudio de posibles soluciones			■	■	■						
Realización de diseños preliminares					■	■	■				
Selección y definición del diseño final							■				
Búsqueda de componentes					■	■	■				
Creación prototipo de pruebas							■	■	■		
Programación software								■	■		
Pruebas prototipo										■	
Elaboración documentos básicos										■	■
Elaboración producto final										■	■



## **10-Orden de prioridad de los documentos básicos**

1. Planos
2. Pliego de condiciones
3. Presupuesto
4. Memoria