

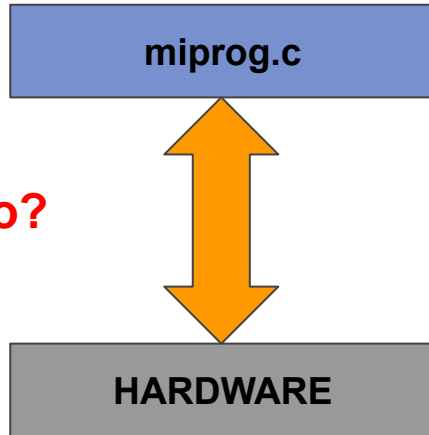
Configuración de Pines - Matriz de multiplexación LPC845 y bloque de registros IOCON

Informática II - R2004

Diagrama en capas



¿Cómo llego?

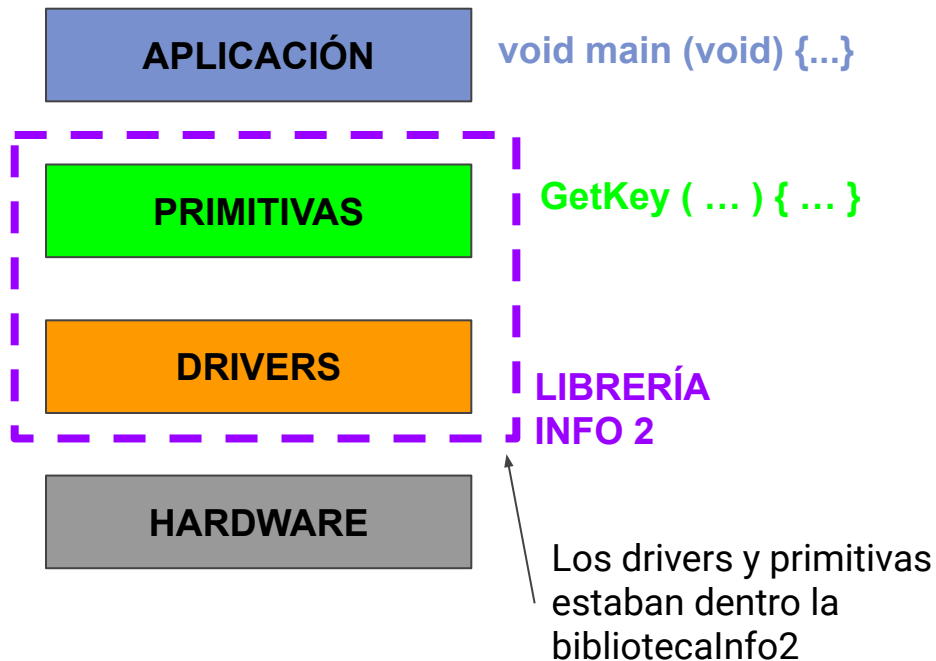


LO TENEMOS QUE HACER NOSOTROS

1. Hojas de datos (me dicen cómo tengo que hacer para tocar el Hardware)
2. Desarrollo de **DRIVERS** (funciones que están diseñadas para manipular el Hardware de un determinado sistema)
3. Desarrollo de **PRIMITIVAS** (funciones de fácil acceso desde la aplicación que ayudan a que no tenga que modificar mi programa cada vez que cambio el Hardware). Todas las funciones que usamos de bibliotecaInfo2 son las primitivas de dicha librería.

Diagrama en capas

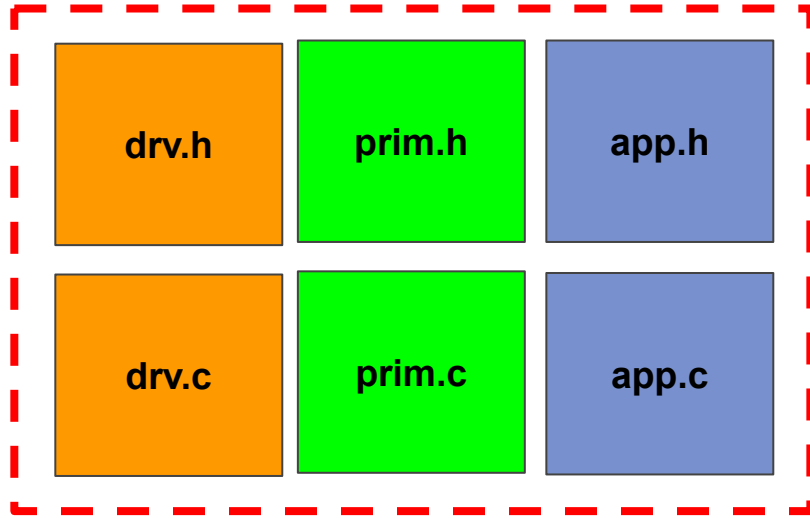
La aplicación ya sabemos realizarla por medio de la estrategia de máquinas de estados. Esta hace uso de las primitivas para acceder al Hardware



El diagrama en capas me permite que el código desarrollado sea portable. Puedo reutilizar los drivers y primitivas en distintas aplicaciones y en caso de tener que cambiar el hardware (usar otro micro) solo reescribo la capa de Drivers.

A partir de ahora nos concentraremos en realizar las primitivas y drivers de los distintos periféricos que dispone el LPC1769 y algunos externos que se utilizan frecuentemente

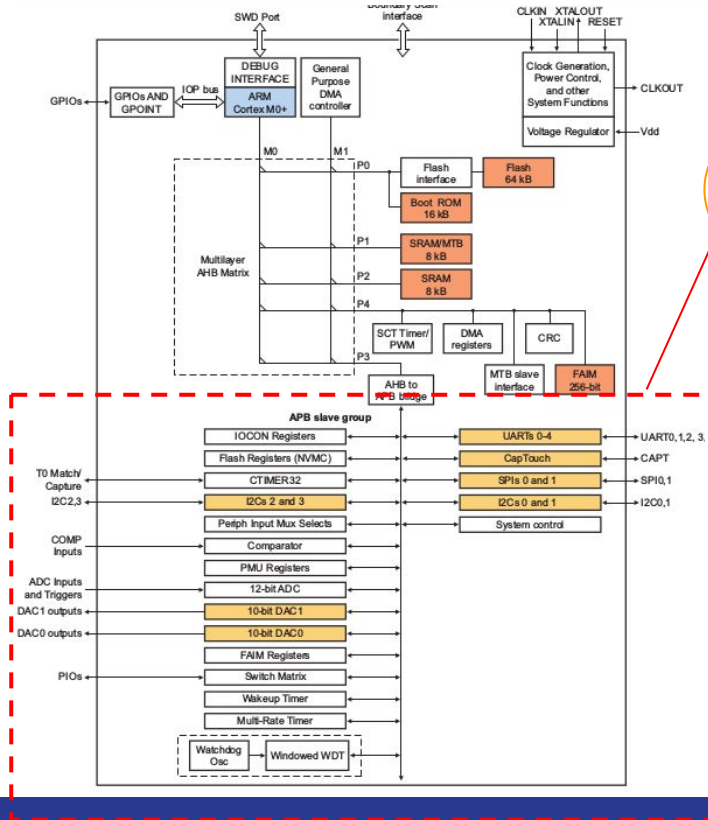
Armado de nuestro Proyecto



PROYECTO

Para cada periférico vamos a armar un .h y .c para los drivers del mismo, otro .h y .c con las primitivas. Esto nos permite mover de un proyecto a otro los archivos necesarios para la aplicación que vayamos a realizar.

¿Qué periféricos tiene mi microcontrolador? y... ¿Cómo hago para comunicarlos con el mundo exterior?

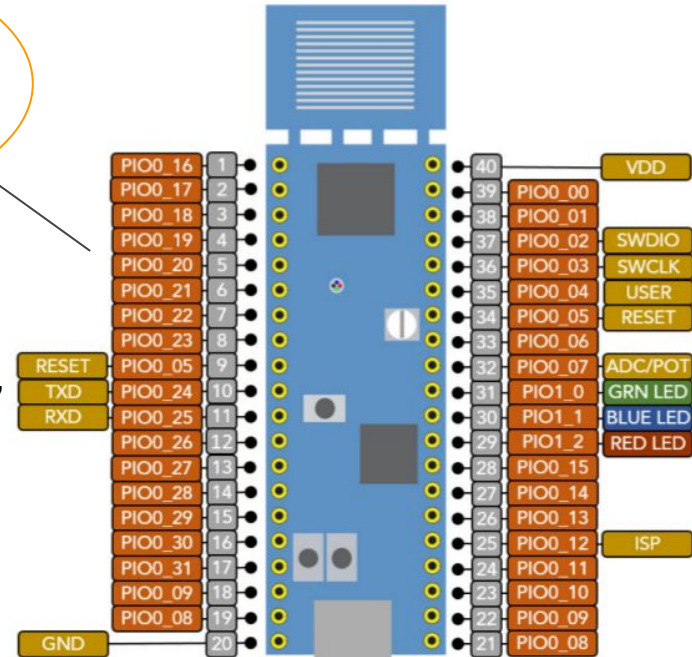


Periféricos disponibles

Solo tengo 35 pines

Se encuentran divididos en 2 **PUERTOS**, el puerto 0, que tiene 32 pines disponibles, y el puerto 1, que solo tiene 3.

Se nombran como $PIO_n.m$ o $Pn.m$, donde n es el número de puerto y m el número de pin



Configuración de pines en el LPC845

El LPC845 posee una gran cantidad de periféricos, y en su presentación LPC845BRK, tenemos 35 pines para ingresar o enviar señales desde o hacia el mundo exterior. Esto significa que, una vez que seleccionemos que periféricos y que funciones necesitamos de los mismos, tendremos que seleccionar también a través de que pin queremos enviar o recibir dichas señales.

No todos los microcontroladores tienen esta particularidad de poder configurar los pines para diferentes funciones, sino que en este caso la funcionalidad viene dada por una matriz de multiplexación, que envía o recibe señales de los periféricos hacia o desde los pines del microcontrolador.

APB peripherals		
31-30	(reserved)	0x4007 FFFF
29	UART4	0x4007 8000
28	UART3	0x4007 4000
27	UART2	0x4007 0000
26	UART1	0x4006 C000
25	UART0	0x4006 8000
24	CapTouch	0x4006 4000
23	SPI1	0x4006 0000
22	SPI0	0x4005 C000
21	I2C1	0x4005 8000
20	I2C0	0x4005 4000
19	(reserved)	0x4005 0000
18	Syscon	0x4004 C000
17	IOCON	0x4004 8000
16	Flash controller	0x4004 4000
15	(reserved)	0x4004 0000
14	CTIMER 0	0x4003 C000
13	I2C3	0x4003 8000
12	I2C2	0x4003 4000
11	Input Multiplexing	0x4003 0000
10	(reserved)	0x4002 C000
9	Analog Comparator	0x4002 8000
8	PMU	0x4002 4000
7	ADC	0x4002 0000
6	DAC1	0x4001 C000
5	DAC0	0x4001 8000
4	PA0M controller	0x4001 4000
3	Switch Matrix	0x4001 0000
2	Wake-up Timer	0x4000 C000
1	Multi-Rate Timer	0x4000 8000
0	Watchdog timer	0x4000 4000

¿Qué tengo que hacer para modificar la función de un pin?

Antes de comenzar el programa, tengo que configurar la funcionalidad de todos los pines de mi placa, de acuerdo al hardware que se encuentre conectado al mismo.

Para configurar los pines, tengo que seguir los siguientes pasos:

- Habilitar el **clock** de la matriz de multiplexación.
- Deshabilitar o habilitar las **funciones especiales** de cada pin.
- Habilitar la **función** que quiera asociar al pin.



I. Habilitar el clock

Todos los periféricos poseen una señal de clock. La misma es necesaria para el funcionamiento de cualquier dispositivo sincrónico, por lo que sin ésta, el dispositivo se encuentra deshabilitado. Para ahorrar energía, la mayoría de los dispositivos amanecen con su señal de clock deshabilitada, y el primer paso para configurarlos es habilitar la misma, y en algunos casos, seleccionar la frecuencia de funcionamiento.

La habilitación del clock para la matriz de multiplexación se hace mediante el registro SYSAHBCLKCTRL0, bit 7.

Más adelante volveremos a usar este registro para habilitar el clock de otros periféricos.

Table 146. System clock control 0 register (SYSAHBCLKCTRL0, address 0x4004 8080) bit description

Bit	Symbol	Value	Description	Reset value
0	SYS		Enables the clock for the AHB, the APB bridge, the Cortex-M0+ core clocks, SYSCON, and the PMU. This bit is read only and always reads as 1.	1
1	ROM		Enables clock for ROM.	1
		0	Disable	
		1	Enable	
2	RAM0_1		Enables clock for SRAM0 and SRAM1.	1
		0	Disable	
		1	Enable	
3			Reserved	0
4	FLASH		Enables clock for flash.	1
		0	Disable	
		1	Enable	
5	I2C0		Enables clock for I2C0.	0
		0	Disable	
		1	Enable	
6	GPIO0		Enables clock for GPIO0 port registers.	0
		0	Disable	
		1	Enable	
7	SWM		Enables clock for switch matrix.	1
		0	Disable	
		1	Enable	
8	SCT		Enables clock for state configurable timer SCTimer/PWM.	0
		0	Disable	
		1	Enable	
9	WKT		Enables clock for self-wake-up timer.	0
		0	Disable	
		1	Enable	
10	MRT		Enables clock for multi-rate timer.	
		0	Disable	
		1	Enable	
11	SPI0		Enables clock for SPI0.	0
		0	Disable	

II. Deshabilitar funciones especiales

Cada pin del microcontrolador tiene 2 funciones asociadas que no pueden moverse a otro pin: La función de GPIO, y una función especial (analógica, relacionada con un TIMER o un clock) que puede habilitarse o deshabilitarse mediante los registros PINENABLE0 y PINENABLE1. Cada uno de los bits de este registro permite habilitar la función especial de cada pin.

Si quiero utilizar el pin como GPIO, o asignarle la función de un periférico, debo deshabilitar la función especial del pin, escribiendo un 0 en el bit del registro asociado.

Table 195. Pin enable register 0 (PINENABLE0, address 0x4000 C1C0) bit description

Bit	Symbol	Value	Description	Reset value
14	ADC_0		ADC_0 function select.	1
		0	ADC_0 enabled on pin PIO0_7.	
		1	ADC_0 disabled.	
15	ADC_1		ADC_1 function select.	1
		0	ADC_1 enabled on pin PIO0_6.	
		1	ADC_1 disabled.	
16	ADC_2		ADC_2 function select.	1
		0	ADC_2 enabled on pin PIO0_14.	
		1	ADC_2 disabled.	
17	ADC_3		ADC_3 function select.	1
		0	ADC_3 enabled on pin PIO0_23.	
		1	ADC_3 disabled.	
18	ADC_4		ADC_4 function select.	1
		0	ADC_4 enabled on pin PIO0_22.	
		1	ADC_4 disabled.	
19	ADC_5		ADC_5 function select.	1
		0	ADC_5 enabled on pin PIO0_21.	
		1	ADC_5 disabled.	
20	ADC_6		ADC_6 function select.	1
		0	ADC_6 enabled on pin PIO0_20.	
		1	ADC_6 disabled.	
21	ADC_7		ADC_7 function select.	1
		0	ADC_7 enabled on pin PIO0_19.	
		1	ADC_7 disabled.	
22	ADC_8		ADC_8 function select.	1
		0	ADC_8 enabled on pin PIO0_18.	
		1	ADC_8 disabled.	
23	ADC_9		ADC_9 function select.	1
		0	ADC_9 enabled on pin PIO0_17.	
		1	ADC_9 disabled.	

III. Colocar la función de un periférico en un pin

Los registros PINASSIGN permiten asignar las diferentes funciones de un periférico a un pin determinado. Un pin puede funcionar como la entrada de más de un periférico, pero solo puede funcionar como la salida de un periférico.

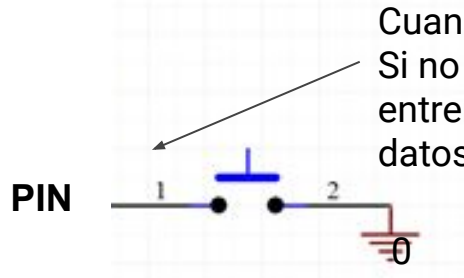
Table 180. Pin assign register 0 (PINASSIGN0, address 0x4000 C000) bit description

Bit	Symbol	Description	Reset value
7:0	U0_TXD_O	U0_TXD function assignment. The value is the pin number to be assigned to this function. The following pins are available: PIO0_0 (= 0) to PIO0_31 (= 0x1F) and from PIO1_0 (= 0x20) to PIO1_21(= 0x35).	0xFF
15:8	U0_RXD_I	U0_RXD function assignment. The value is the pin number to be assigned to this function. The following pins are available: PIO0_0 (= 0) to PIO0_31 (= 0x1F) and from PIO1_0 (= 0x20) to PIO1_21(= 0x35).	0xFF
23:16	U0_RTS_O	U0_RTS function assignment. The value is the pin number to be assigned to this function. The following pins are available: PIO0_0 (= 0) to PIO0_31 (= 0x1F) and from PIO1_0 (= 0x20) to PIO1_21(= 0x35).	0xFF
31:24	U0_CTS_I	U0_CTS function assignment. The value is the pin number to be assigned to this function. The following pins are available: PIO0_0 (= 0) to PIO0_31 (= 0x1F) and from PIO1_0 (= 0x20) to PIO1_21(= 0x35).	0xFF

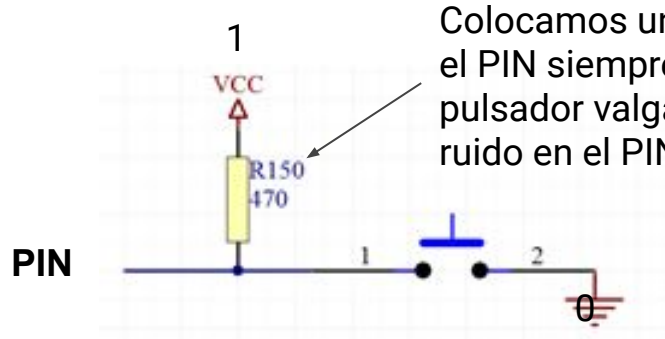
Resistencias de pull-up o pull-down



VCC	GND
1	0



Cuando se presiona el pulsador el PIN vale 0. Si no se presiona está "al aire". Esto genera que entren ruidos a PIN lo que genera la lectura de datos erróneos.



Colocamos una resistencia de pull-up para que el PIN siempre valga 1 y cuando se presione el pulsador valga 0. Evitamos lecturas erróneas y ruido en el PIN.

Modo open drain

Para adaptar la salida del microcontrolador a diferentes niveles de tensión (por ejemplo, para interfacear con dispositivos de 5V), se puede configurar que, cuando la salida esté en estado “alto”, el pin esté “al aire”. Por fuera, se puede conectar la tensión deseada:



En este caso, desactivaría la resistencia de pull-up y pull down y activaría la función open-drain