

1-Introducción

-Este apunte pretende ser una guía para aquellos alumnos que se inician en la programación de microcontroladores con arquitectura ARM, utilizando como plataforma de hardware el kit de desarrollo diseñado especialmente para la cátedra de Técnicas Digitales II.

El mismo posee:

-LPC2105.

-Regulación lineal de tensión (5V, 3.3V y 1.8V), con sus correspondientes 3 LED's de encendido, protección contra transitorios de tensión y cortocircuitos.

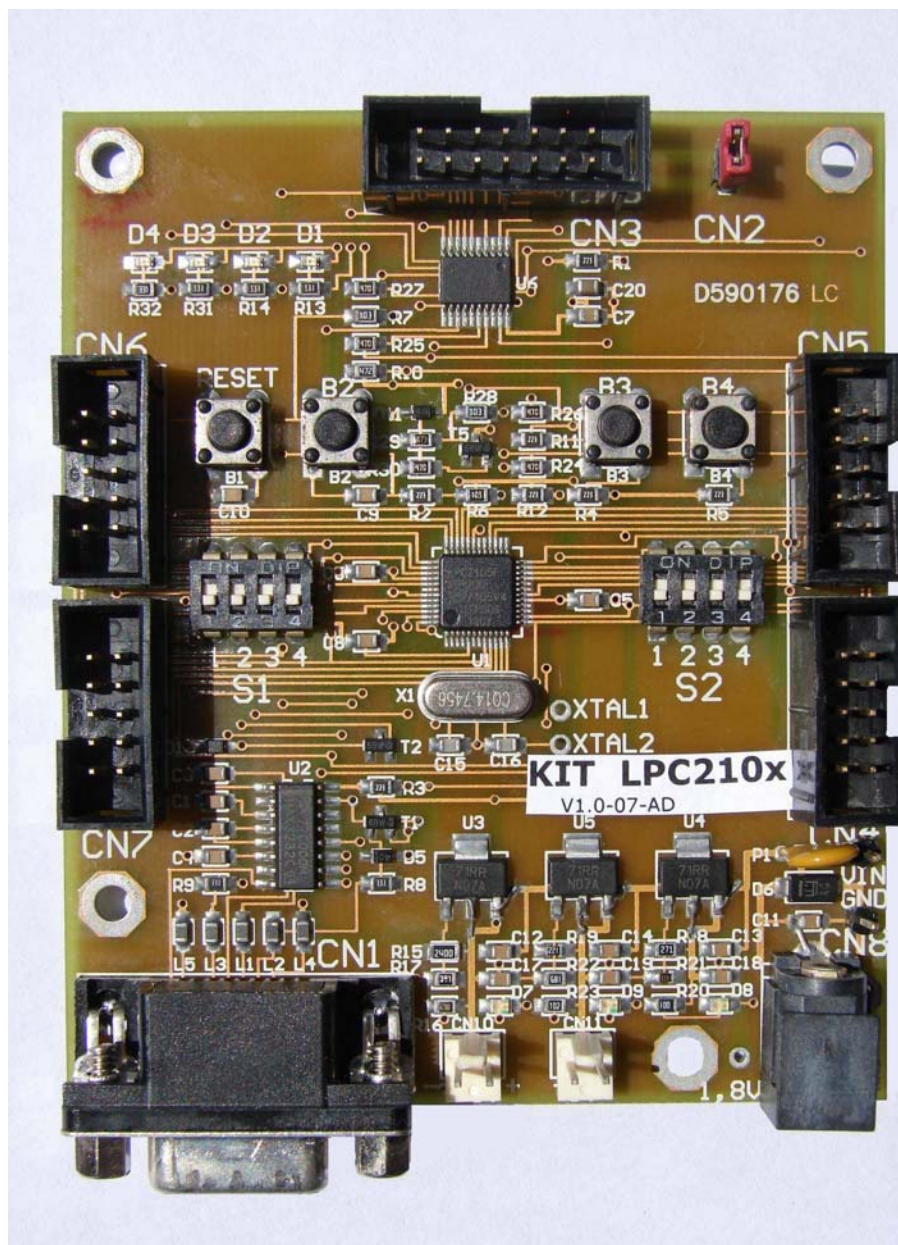
- JTAG onboard para grabación y debugging.

- Driver TTL-EIA232 con salida a DB9.

- Pulsador de reset.

- 4 LED's y 3 pulsadores para uso gral.

- 4 conectores IDC en donde se encuentran presente todos los pines del LPC2105.



2-Digramas esquemáticos eléctricos

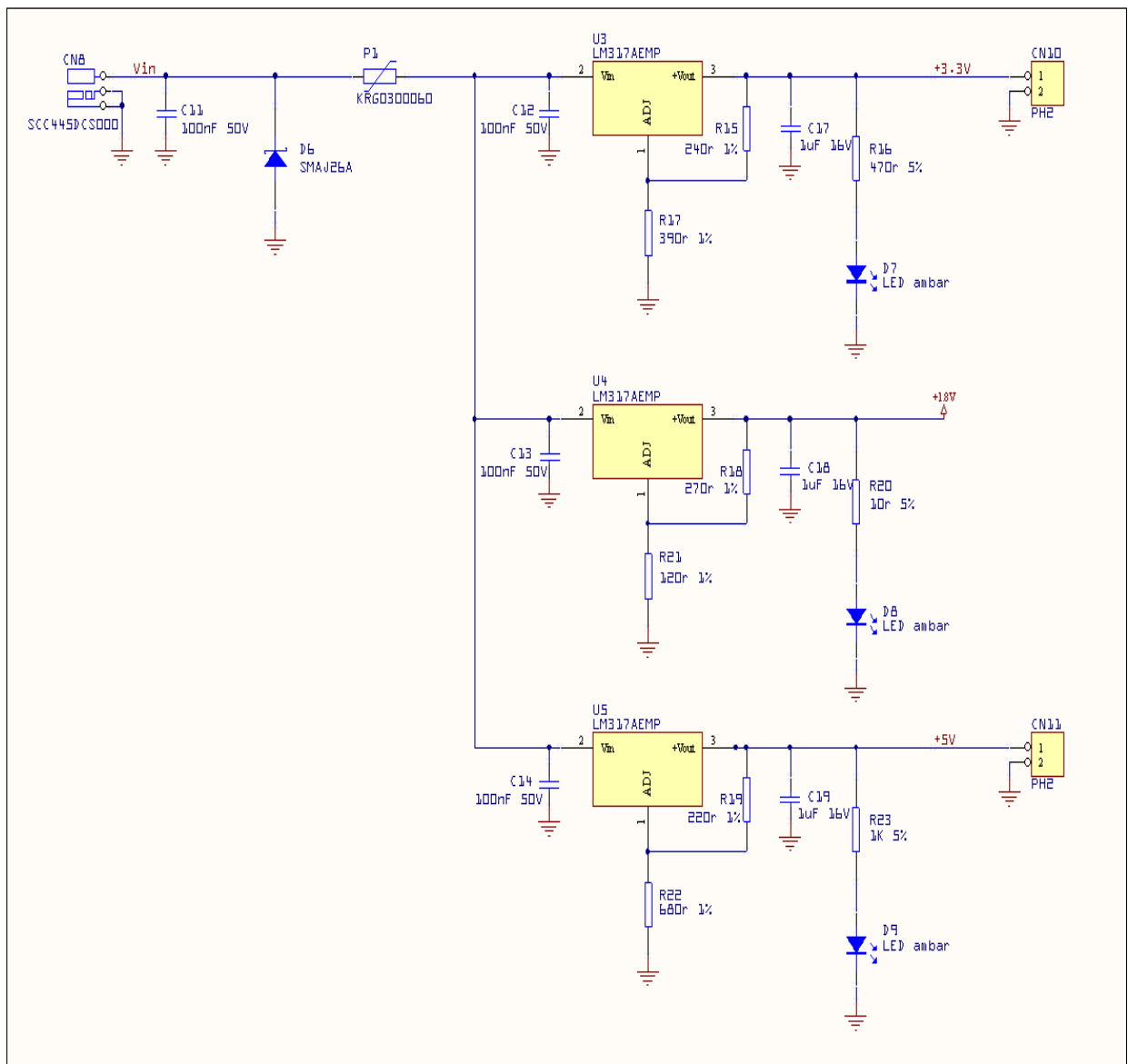
2.1 Fuente de alimentación.

La entrada de alimentación se hace a través de un jack de 3.5mm de diámetro, con valores de tensión en un rango de 7 a 24V C.C., aunque lo óptimo es 9V por una cuestión de disipación de energía en calor sobre los reguladores lineales.

D6 protege de los transitorios de tensión y el polyswitch P1 evita la circulación de corriente por encima de los 300mA.

Cuenta con tensión regulada de 1.8V para el core, 3.3V para los periféricos y 5V para alimentar algún dispositivo externo, ya que este microcontrolador es 5V tolerante en sus entradas.

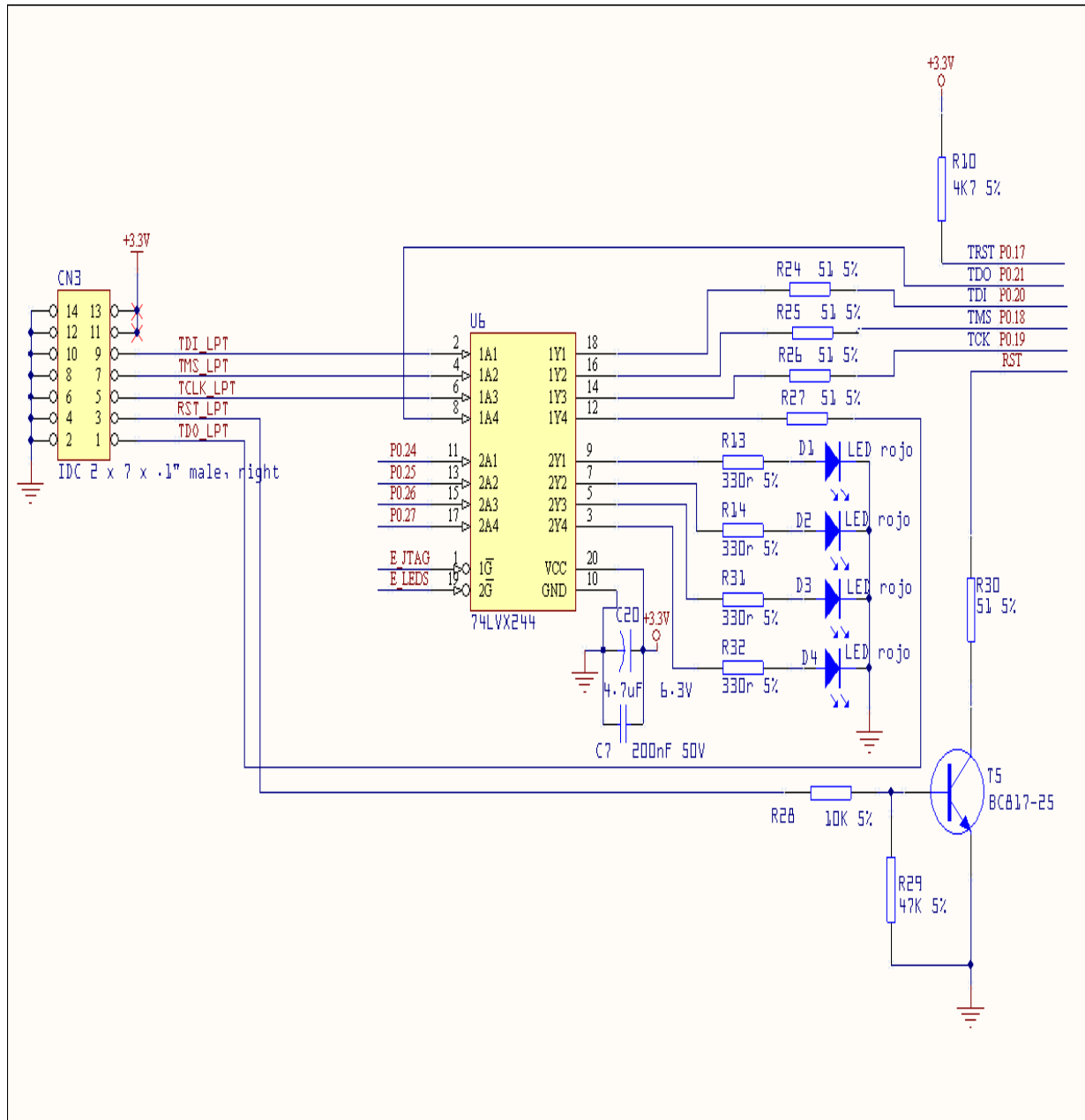
3.3V y 5V están disponibles en CN10 y CN11 respectivamente.



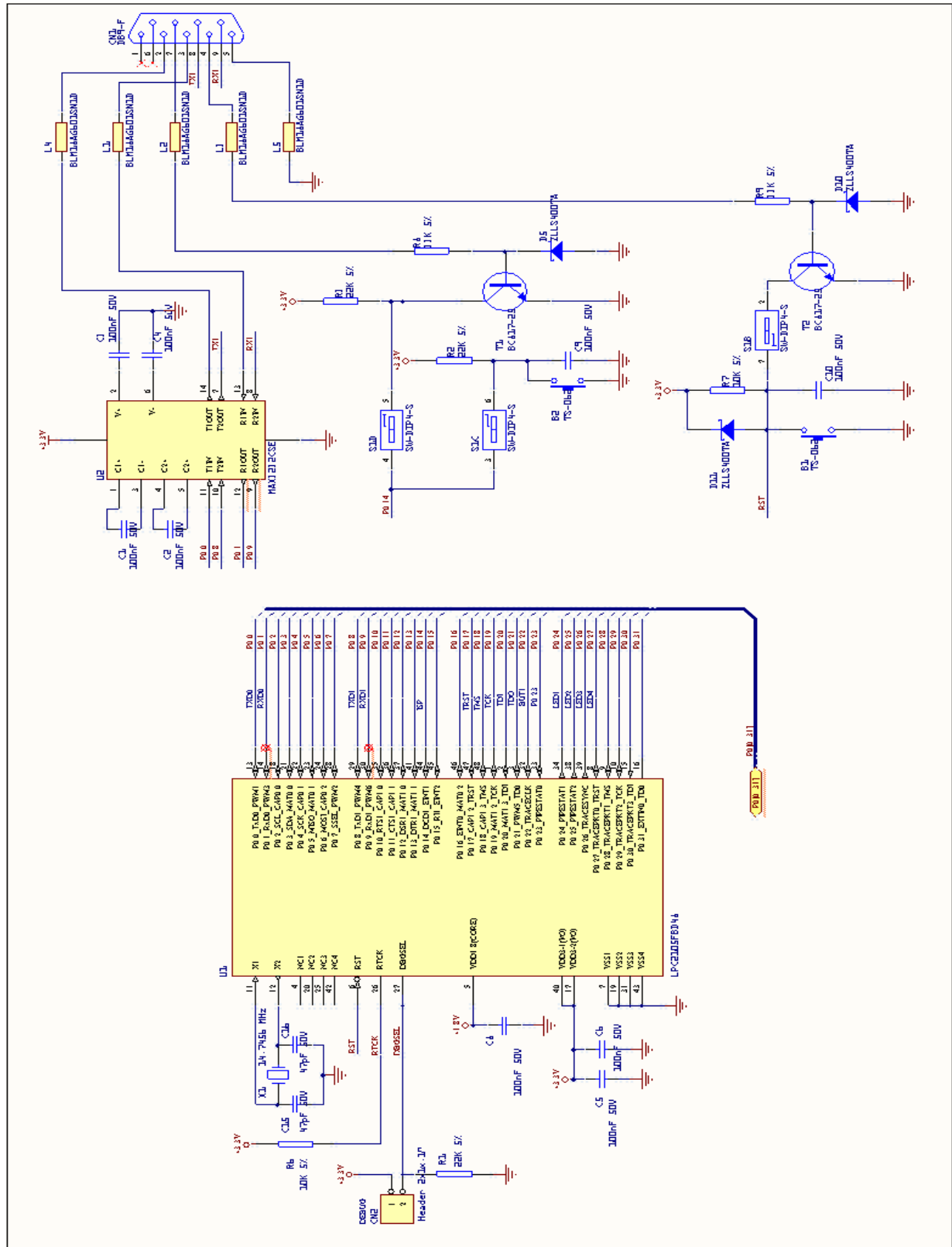
2.2 Interface JTAG y LED's.

La interface JTAG la implementé sobre el mismo impreso, debido a la simplicidad, confiabilidad y popularidad de la interface Wiggler. El conector CN3 por lo tanto, tiene la distribución de señales standard de este tipo de interface.

Ya que el buffer transparente U6, está dividido en mitades, con habilitaciones independientes, utilice la mitad sobrante del JTAG para colocar 4 LED's rojos conectados a 4 líneas de GPIO.



2.3 CPU + Serie.



La CPU está formada por el LPC2105FB48 con encapsulado LQFP de 48 pines, cristal externo de 14.7456 Mhz y el conector CN2 donde se coloca un jumper cuando se quiere entrar en el modo debug.

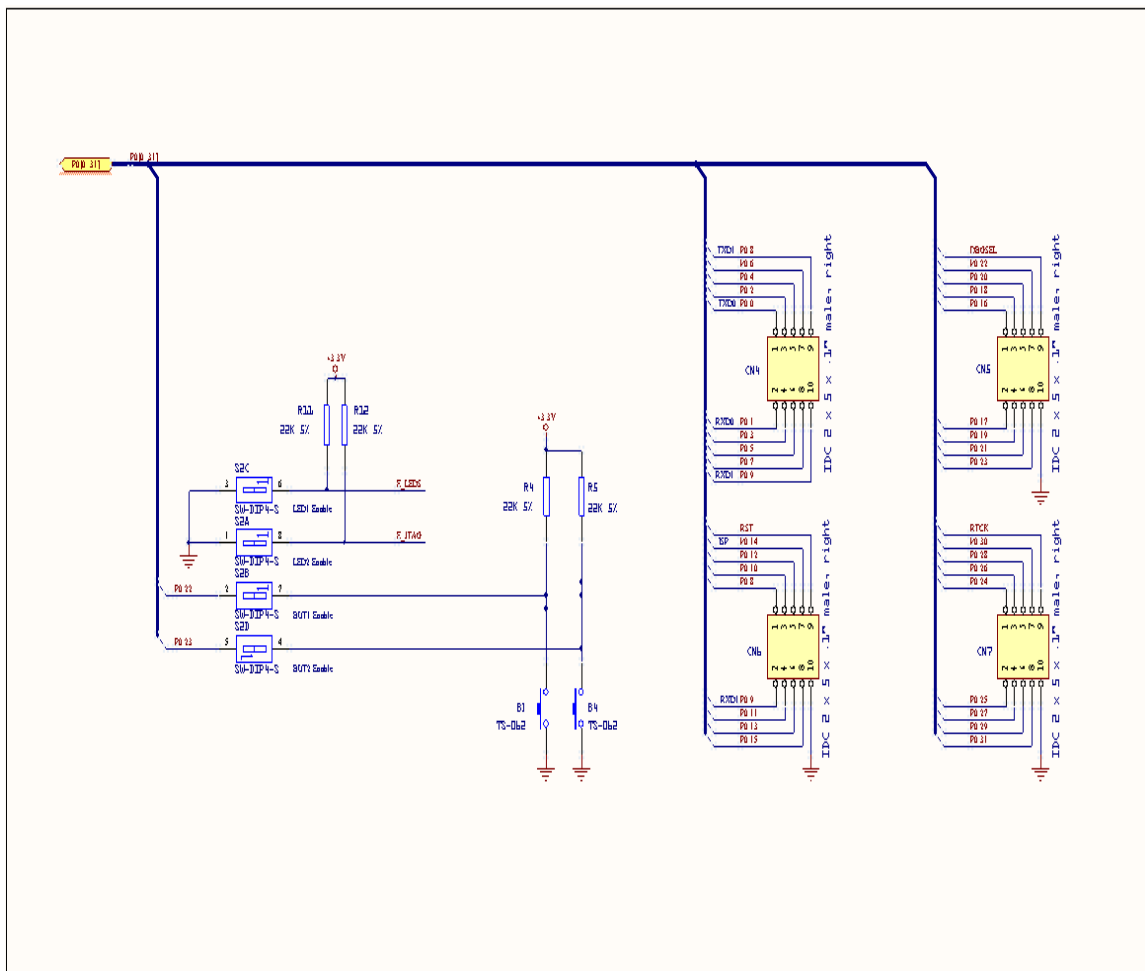
El driver U2, permite manejar sobre el conector CN1 (DB9-F) las señales TxD0, RxD0, TxD1 y RxD1 con niveles EIA-232 correspondientes a las 2 UART que posee el LPC2105.

El dip switch S1D deja llegar previa adaptación de niveles, la señal DCD1 desde el DB9-F hasta el LPC2105, necesaria por el protocolo de descarga de los programas por línea serie, de forma similar ocurre con S1B y la señal de reset.

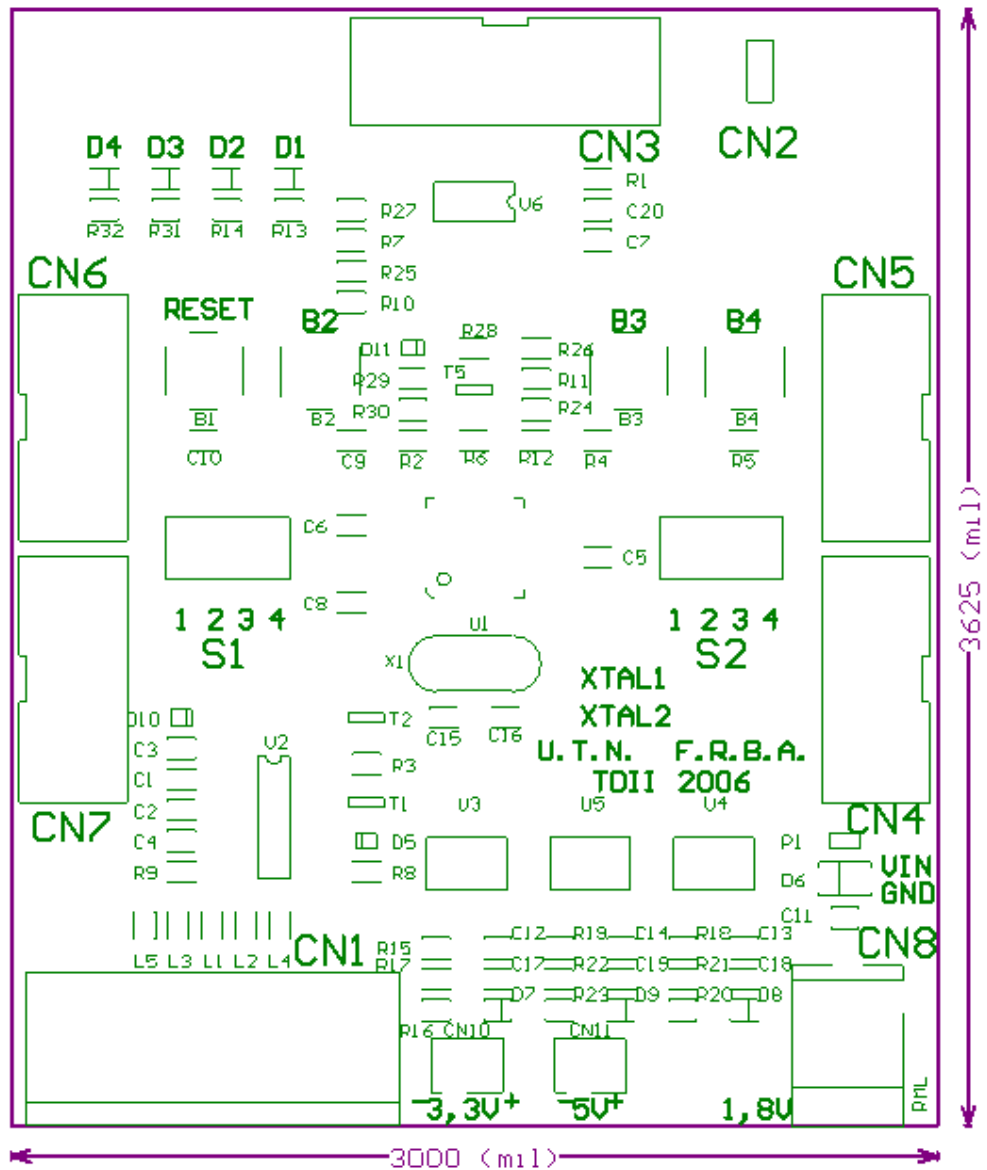
S1C permite colocar un pulsador sobre la línea DCD1.

El pulsador B1 actúa como reset manual.

2.4 Entrada / Salida.



1	Quantity	Components	Comment	Pattern	Description
2	4	B1, B2, B3, B4	TS-062	TS-062	Tact switch
3	13	C1, C2, C3, C4, C5, C6, C8, C9, C10, C11, C12, C13, C14	100nF 50V	C_0805	Capacitor
4	2	C15, C16	47pF 50V	C_0805	Capacitor
5	3	C17, C18, C19	1uF 16V	C_0805	Capacitor
6	1	C20	4.7uF 6.3V	C_0805	Capacitor
7	1	C7	200nF 50V	C_0805	Capacitor
8	1	CN1	DB9-F	DB09_F_8.1	Connector
9	2	CN10, CN11	PH2	B_2B-PH	Connector
10	1	CN2	Header 2x1x.1"	,JMP_102_0.1"	Connector
11	1	CN3	IDC 2 x 7 x .1 male right	IDC-14	Connector
12	4	CN4, CN5, CN6, CN7	IDC 2 x 5 x .1 male right	IDC-10	Connector
13	1	CN8	SCC445DCS000	DC JACK 2.5MM	Connector
14	4	D1, D2, D3, D4	LED rojo	D_0805	LED
15	3	D5, D10, D11	ZLLS400TA	SOD_323	Schottky Diode
16	1	D6	SMAJ26A	D_SMA	Schottky Diode
17	3	D7, D8, D9	LED ambar	D_0805	LED
18	5	L1, L2, L3, L4, L5	BLM18AG601SN1D	B_0805	Ferrite bead
19	1	P1	KRG0300060	AXIAL0.2	Polyswitch
20	7	R1, R2, R3, R4, R5, R11, R12	22K 5%	R_0805	Resistor
21	1	R10	4K7 5%	R_0805	Resistor
22	4	R13, R14, R31, R32	330r 5%	R_0805	Resistor
23	1	R15	240r 1%	R_0805	Resistor
24	1	R16	470r 5%	R_0805	Resistor
25	1	R17	390r 1%	R_0805	Resistor
26	1	R18	270r 1%	R_0805	Resistor
27	1	R19	220r 1%	R_0805	Resistor
28	1	R20	10r 5%	R_0805	Resistor
29	1	R21	120r 1%	R_0805	Resistor
30	1	R22	680r 1%	R_0805	Resistor
31	1	R23	1K 5%	R_0805	Resistor
32	5	R24, R25, R26, R27, R30	51 5%	R_0805	Resistor
33	1	R29	47K 5%	R_0805	Resistor
34	3	R6, R7, R28	10K 5%	R_0805	Resistor
35	2	R8, R9	33K 5%	R_0805	Resistor
36	2	S1, S2	SW-DIP4-S	DIP_SW_8	Dip Switch 4x1
37	3	T1, T2, T5	BC817-25	SOT_23	NPN Transistor
38	1	U1	LPC2105FBD48	LQFP_48	ARM7 microcontroller
39	1	U2	MAX3232CSE	SO_16	Dual RS232 transceivers
40	3	U3, U4, U5	LM317AEMP	SOT_223_4L	LDO adj.
41	1	U6	74LVX244	TSSOP_20	Octal buffer
42	1	X1	14.7456 MHz	AXIAL0.2_XTAL	Crystal



2-Core

El kit de desarrollo posee un ARM 7 TDMI-S de NXP (ex Philips) de la familia LPC210x, más específicamente el LPC2105, cuyas principales características son:

- De última generación, ya que sale al mercado en el año 2004.
- 60 Mhz de frecuencia de operación interna, a través de un PLL (lazo enclavado en fase), que permite aumentar la frecuencia de operación interna a partir del cristal externo.
- Soporta modos de operación de 16/32 bits.
- 128 Kbytes de memoria flash.
- 32 Kbytes de memoria SRAM.
- Al ser TDMI cuenta con modo T:thumb, soporta D: debug interface, tiene M: Multiplier interface e I: interrupciones de alta velocidad.
- Múltiples interfaces series: 2 UARTs(16C550), I2C y SPI de alta velocidad (400Kbits/s).
- 2 Timers de 32 bits con 7 canales de captura/comparación.
- 32 líneas de input/output de propósito general, con 5V tolerantes.
- 100000 ciclos de borrado/escritura de flash.
- RTC (reloj de tiempo real).
- Debugging mediante JTAG.

2-Periféricos

Todos los pines de I/O tienen múltiples funciones, las cuales se seleccionan mediante un multiplexor llamado **Pin Selector Block** utilizando los registros PINSEL0 y PINSEL1.

Luego de un reset, todos los pines quedan configurados como entrada / salida de propósito general (PINSEL0 = PINSEL1 = 0x00000000).

2.1 Input / output de propósito general (GPIO).

Dichos pines son controlados por 4 registros: IODIR, IOSET, IOCLR, IOPIN.

Dichos registros son de 32 bits, ya que la capacidad de la CPU lo permite y además este microcontrolador tiene 32 pines de uso general.

Los bit del 0 al 31 de los registros se corresponden con los pines P0.0 a P0.31.

IODIR: establece la dirección de los pines, colocando un "0" en el bit correspondiente al pin lo coloca como entrada, caso contrario, con "1" queda como salida. Luego de un reset, todos los pines quedan configurados como entrada (IODIR = 0x00000000).

IOSET: Fuerza el pin correspondiente al estado lógico "1".

IOCLR: Fuerza el pin correspondiente al estado lógico "0".

IOPIN: Se utiliza para leer el estado lógico de los pines.

Ejemplo 1: Encender un LED en P0.31 solo cuando se encuentra presionado un pulsador en P0.1.

```
#include <LPC210x.H>
int main (void)
{
    PINSEL0 = 0x00000000;    //Los pines de P0.0 a P0.15 son E/S
    PINSEL1 = 0x00000000;    //Los pines de P0.16 a P0.31 son E/S
    //IOCLR = 0xFFFFFFFF;    //Se limpia el registro
    IODIR = 0x0F000000;    //Pines P0.27 a P0.24 como salidas
    while(1)
    {
        if (!(IOPIN & 0x00400000))    //Según el estado de P0.22 (B3):
            IOSET = 0x0F000000;    //si está presionado B3, se encienden los 4 LED's
        else
            IOCLR = 0x0F000000;    //si está presionado B3, se apagan los 4 LED's
    }
}
```