



Escola Universitària  
Politécnica de Mataró

**Enginyeria Tècnica Industrial: Especialitat Electrònica Industrial**

**Mòdul Microcontrolador Universal**

**A – Memòria descriptiva**

**JOSEP PITARCH PAREDES  
JULIÁN HERRILLO TELLO  
PRIMAVERA 2009**



## **RESUM**

Aquest projecte tracta sobre el disseny i construcció d'un mòdul microcontrolador universal que es pugui adaptar a diferents tipus d'aplicacions, ja siguin en un senzill control local o en un complex sistema de control distribuït.

Es pretén desenvolupar un producte amb el màxim de funcionalitat i connectivitat possibles tenint en compte el cost i les dimensions d'aquest mòdul. Algunes de les característiques més importants són: E/S digital i analògica, sortides PWM, connectivitat RS232, RS485 i Ethernet.

La realització d'aquest projecte s'ha dividit en diferents etapes:

Estudi de les solucions que ens ofereix el mercat en aquest tipus de producte per la realització del disseny funcional del mòdul.

Disseny electrònic i muntatge del mòdul microcontrolador universal, incloent totes les funcionalitats que s'han definit per al mòdul.

Realització d'aplicacions pràctiques per demostrar les diferents funcionalitats que ens proporciona el mòdul, com la de servidor web o control PID.



## **RESUMEN**

Este proyecto trata sobre el diseño y construcción de un módulo microcontrolador universal que se pueda adaptar a diferentes tipos de aplicaciones, ya sea en un sencillo control local o en un complejo sistema de control distribuido.

Se pretende desarrollar un producto con el máximo de funcionalidad y conectividad posibles teniendo en cuenta el coste y las dimensiones de este módulo. Algunas de las características más importantes son: E/S digital y analógica, salidas PWM, conectividad RS232, RS485 y Ethernet.

La realización de este proyecto se ha dividido en diferentes etapas:

Estudio de las soluciones que nos ofrece el mercado en este tipo de producto para la realización del diseño funcional del módulo.

Diseño electrónico y montaje del módulo microcontrolador universal, incluyendo todas las funcionalidades que se han definido para el módulo.

Realización de aplicaciones prácticas para demostrar las diferentes funcionalidades que nos proporciona el módulo, como la de servidor web o control PID.



## **ABSTRACT**

This project treats on the design and construction of a universal microcontroller module that it could adapt to different application types, either in a simple local control or in a complex distributed control system.

It is tried to develop a product with the maximum of functionality and connectivity possible bearing in mind the cost and the module dimensions. Some of the most important characteristics are: digital and analog I/O, PWM outputs, connectivity RS232, RS485 and Ethernet.

The carrying out of this project has been divided in different stages:

Study of the solutions that the market offers us in this type of product for the performance of the module functional design.

Electronic design and assembly of the universal microcontroller module, including all the functionalities that have been defined for the module.

Accomplishment of the practical applications to demonstrate the different functionalities that the module provides to us, as that of the web server or PID control.





# ÍNDIX

1. OBJECTE DEL PROJECTE .....	1
2. ABAST DEL PROJECTE .....	3
3. JUSTIFICACIÓ DE LES SOLUCIONS .....	5
4. ESPECIFICACIONS TÈCNIQUES .....	7
5. SOLUCIÓ TECNOLÒGICA .....	9
5.1. Introducció .....	9
5.2. Microcontrolador .....	10
5.2.1. Criteris de selecció .....	10
5.2.2. El microcontrolador PIC18F97J60 .....	12
5.2. Entrades i sortides digitals .....	16
5.3. Entrades analògiques .....	19
5.4. Sortides analògiques .....	21
5.5. Sortides PWM .....	25
5.6. Connectivitat RS232 .....	28
5.6. Connectivitat RS485 .....	30
5.7. Connectivitat Ethernet .....	33
5.7.1. Ethernet .....	33
5.7.2. Controlador d'Ethernet .....	34
5.7.3. Piles de protocols .....	37
5.7.4. Pila TCP/IP de Microchip .....	41
6. MUNTATGE DEL PROTOTIP .....	47
6.1. Soldadura del microcontrolador PIC18F97J60 .....	48
6.1.1. Tècnica de muntatge superficial (SMT) .....	48
6.1.2. Procés de soldadura per refusió .....	50
7. APLICACIONS PRÀCTIQUES .....	55
7.1. Eines de desenvolupament .....	55
7.2. Servidor web .....	57
7.2.2. Servidor HTTP de Microchip .....	58
7.2.3. Generació dinàmica de pàgines HTTP .....	60
7.2.4. Exemple de servidor web embedded .....	62
7.3. Exemples d'aplicacions .....	67

## II *Índex*

7.3.1. Configuració de l'estat de les sortides digitals.....	69
7.3.2. Control PID .....	71
8. CONCLUSIONS I MILLORES.....	81
9. BIBLIOGRAFIA.....	85

## ÍNDIX DE FIGURES

Fig. 5.1. Diagrama de blocs de mòdul microcontrolador universal .....	9
Fig. 5.2. Microcontrolador PIC18F97J60.....	12
Fig. 5.3. Arquitectura interna del microcontrolador PIC18F97J60.....	15
Fig. 5.4. Esquema de la protecció de les entrades digitals (mòdul exterior).....	16
Fig. 5.5. Diagrama de blocs del convertidor A/D.....	19
Fig. 5.6. Limitació de tensió de les entrades analògiques .....	21
Fig. 5.7. Configuració dels pins del AD7302.....	22
Fig. 5.8. Diagrama de blocs funcional del AD7302.....	22
Fig. 5.9. Connexionat del DAC AD7302 en el mòdul .....	24
Fig 5.10. Sortida PWM.....	25
Fig. 5.11. Exemple d'aplicació en pont complet .....	27
Fig. 5.12. Configuració del ST232CN.....	29
Fig. 5.13. Connexionat del ST232CN en el mòdul .....	30
Fig. 5.14. Configuració de pins del MAX491 i circuit típic d'operació.....	31
Fig. 5.15. Connexionat del MAX491 en el mòdul .....	32
Fig. 5.16. Diagrama de blocs del mòdul d'Ethernet.....	34
Fig. 5.17. Components externs necessaris per el mòdul d'Ethernet.....	36
Fig. 5.18. Esquema del connector RJ-45 utilitzat.....	37
Fig. 5.19. Capes del Model OSI .....	38
Fig. 5.20. Pila TCP/IP i protocols que utilitza.....	39
Fig. 5.21. Comparació entre el model OSI i la pila TCP/IP .....	40
Fig. 5.22. Comparació de la pila TCP/IP de Microchip amb el Model TCP/IP.....	43
Fig. 6.1. Vista de la cara superior del prototip .....	48
Fig. 6.2. Vista de la cara inferior del prototip .....	48
Fig. 6.3. Adaptador SDM RE471 .....	49
Fig. 6.4. Mides del microcontrolador PIC18F97J60 .....	50
Fig. 6.5. Soldadura per refusió .....	51
Fig. 6.6. Corba de temperatura per al procés de refusió.....	53
Fig. 6.7. Forn de soldadura per refusió.....	53
Fig. 6.8. PIC18F97J60 soldat a l'adaptador RE471 .....	54
Fig. 7.1. Pantalla principal del MPLAB IDE amb el compilador C.....	56

Fig. 7.2. MPLAB ICD3 .....	56
Fig. 7.3. Pantalla principal del Microchip MPFS Generator.....	59
Fig. 7.4. Pantalla del servidor web d'exemple .....	67
Fig. 7.5. Pantalla del software de configuració d'aplicacions.....	68
Fig. 7.6. Ordinograma configuració sortides.....	70
Fig. 7.7. Ordinograma configuració PID.....	73
Fig. 7.8. Simulador PCS 327 .....	76
Fig. 7.9. Diagrama de blocs del procés simulat (FdT) .....	77
Fig. 7.10. Diagrama de blocs d'un sistema de control realimentat .....	77
Fig. 7.11. Diagrama de blocs del sistema.....	78
Fig. 7.12. Ordinograma rutina control PID .....	79

## ÍNDIX DE TAULES

Taula 5.1. Nivells de tensió d'entrada dels ports d'E/S del PIC18F97J60.....	17
Taula 5.2. Nivells de corrent de sortida dels ports d'E/S del PIC18F97J60 .....	17
Taula 5.3. Tensions de sortida del DAC AD7302.....	23
Taula 5.4. Definicions de la configuració de la pila TCP/IP.....	44
Taula 5.5. Mòduls de la pila TCP/IP .....	46



## **1. OBJECTE DEL PROJECTE**

L'objecte del projecte és el disseny i construcció d'un mòdul microcontrolador universal que ofereixi una forta component de funcionalitat i es pugui adaptar a diferents tipus d'aplicacions.

El mòdul ha d'incloure funcionalitats de E/S digital, E/S analògica, sortides de control PWM, i connectivitat RS232, RS485 i Ethernet.

Es podrà connectar amb altres mòduls per a la implementació de sistemes d'intel·ligència distribuïda, i seran governats des d'un PC que permetrà implementar funcions de càrrega de programes i de supervisió i monitorització d'aplicacions.

El mòdul ha de ser accessible a través d'Internet (embedded Internet) i oferir la possibilitat de funcionar com a servidor web, per tal de controlar l'aplicació des d'un ordinador ubicat a qualsevol punt on tinguem connexió a Internet.

El conjunt de funcionalitat i característiques d'aquest mòdul fan que sigui un controlador molt versàtil capaç de treballar en un ampli ventall d'aplicacions, des de control de màquines simples fins a complexos sistemes de control distribuïts, com poden ser el control de processos industrials.

Un altre aspecte important d'aquest projecte, és la realització d'aquest mòdul microcontrolador universal amb un baix cost que permeti realitzar el màxim de funcions, així com unes dimensions el més petites que sigui possible.

La relació funcionalitat – cost – dimensions és avui en dia un aspecte molt important a l'hora de dissenyar dispositius electrònics aplicats a qualsevol camp.





## 2. ABAST DEL PROJECTE

L'abast d'aquest projecte es centra en el desenvolupament d'un mòdul microcontrolador universal bàsic, que formaria part d'una família de productes amb diferents evolucions en quant a tipus de prestacions que oferirien.

Per tant, els objectius a complir en la realització d'aquest projecte són el disseny i realització d'un mòdul bàsic, amb les característiques explicades en el capítol anterior, però tenint en compte les limitacions fonamentals amb les que ens trobem en un projecte final de carrera, destacant el temps per a la seva realització.

D'aquesta manera, es preveu que el mòdul microcontrolador incorpori una sèrie de funcionalitats que permetin implementar sistemes de control, ja sigui local o distribuït, així com la possibilitat de realitzar la funció de servidor web. En referència a l'entrada/sortida, ha d'incloure entrada/sortida digital i analògica, i sortides de control PWM (Pulse Width Modulation). En el tema de la connectivitat, el mòdul ha d'incloure un port de comunicacions RS232 per connexió amb un PC, un port de comunicacions RS485 per poder crear una xarxa de controladors i un port de comunicacions Ethernet per la connexió del mòdul a Internet.

Pel que tracta del disseny funcional, es realitzarà tenint en compte les diferents solucions que ens ofereix el mercat en aquest tipus de controladors, d'aquesta manera s'intentarà realitzar el màxim de funcions necessàries per aquest tipus de controlador amb un cost mínim.

Una vegada realitzat el disseny funcional, s'estudiarà la manera d'implementar el mòdul microcontrolador universal amb els diferents components que el conformaran, es realitzarà el disseny electrònic del mòdul i s'especificarà totes les seves característiques tècniques.

Es construiran dos prototips per tal d'implementar la simulació d'un sistema de control senzill entre aquests dos mòduls. El muntatge es realitzarà en Wire-Rap, i cada una de les plaques contindrà els elements necessaris per a la realització completa del sistema, circuits

#### 4 *Abast del projecte*

adaptadors de senyals, drivers i receivers per la transmissió sèrie, components per permetre la connectivitat Ethernet, així com el propi microcontrolador.

### 3. JUSTIFICACIÓ DE LES SOLUCIONS

Abans de decidir les característiques i funcionalitats que finalment inclourà el mòdul microcontrolador universal, es va fer un estudi de les solucions que ofereix el mercat per aquest tipus de productes.

La majoria de controladors que els seus fabricants indiquen que són universals i es poden adaptar a diferents tipus d'aplicacions, tenen certes limitacions en quan a la quantitat de funcionalitats que ens ofereixen.

La majoria tenen un controlador bàsic amb una sèrie de funcionalitats que es bastant similar independentment del fabricant:

- Entrades i sortides digitals. Diferent nombre d'E/S digitals depenent de la versió del controlador, ens podem trobar amb controladors que tenen poques E/S fins a tenir una quantitat molt gran en les versions superiors, aquestes quantitats poden anar de 4 fins a 32 E/S digitals. Alguns fabricants donen la possibilitat d'afegir mòduls per augmentar la quantitat d'E/S digitals.
- Entrades i sortides analògiques. Hi ha controladors que en la seva versió més bàsica no ofereix cap E/S analògica, i d'altres que si que en tenen, però el nombre d'aquest tipus d'E/S és bastant petit per a tots els casos, entre 2 i 4 en total. La majoria de controladors que no ofereixen E/S analògica, donen opció a afegir un mòdul que dóna la possibilitat de disposar d'aquesta funcionalitat.
- Connectivitat RS-232 i RS-485. Tots els controladors estudiats disposen tant de connectivitat RS-232 com RS-485 incorporada en el mateix mòdul bàsic.
- Connectivitat Ethernet. Un nombre considerable de controladors que ofereixen aquest tipus de funcionalitat (hi ha alguns que no la contemplen), ho fan com a opcional, i per tant s'ha d'afegir un mòdul extern per disposar d'aquest tipus de connectivitat.

Altres tipus de funcionalitats que s'han trobat, com per exemple el control PID, tampoc l'ofereixen tots els controladors que s'han estudiat. Per altra banda, també s'ha trobat alguns que disposen funcionalitats molt interessants, com la possibilitat de transmetre dades sense cables, via RF o GSM/GPRS, però amb versions superiors o afegint mòduls.

Un altre tipus de producte que s'ha estudiat a l'hora de decidir el tipus de funcionalitats a incloure en el mòdul microcontrolador universal ha estat els autòmats programables de gamma baixa-mitjana dels grans fabricants d'automatismes. Aquest tipus de producte ofereix moltes possibilitats, ja sigui en la seva versió bàsica o afegint mòduls d'ampliació, però el cost d'aquest tipus de controladors és molt alt en comparació al tipus de producte que es vol dissenyar en aquest projecte.

## 4. ESPECIFICACIONS TÈCNIQUES

Les característiques tècniques del mòdul microcontrolador universal són les següents:

- Rang d'alimentació d'entrada al mòdul de 6 - 26 V DC
- Memòria de programa FLASH de 128 kB (per emmagatzemar aplicacions, pila TCP/IP i pàgines web)
- Memòria RAM de 3808 bytes
- Controlador d'Ethernet compatible amb la norma IEEE 802.3, 10Base-T
- 8 entrades digitals. Tensió màxima d'entrada 5,5 V
- 8 sortides digitals. Tensió de sortida 3,3 V. Corrent màxim de sortida 25 mA
- 4 entrades analògiques. 0 – 3,3 V , resolució de 10 bits
- 2 sortides analògiques. 0 – 5 V , resolució de 8 bits
- 1 sortida de control PWM per ondulador trifàsic
- 4 sortides de control PWM simples, combinant-les es pot controlar un chopper per alimentar un motor de CC.
- Mode de treball com a controlador PID, amb una sortida i dues entrades analògiques dedicades
- Connexió de programació RS232, connector DB-9
- Connectivitat amb altres mòduls via RS485, connector de regleta
- Connectivitat Ethernet, connector RJ45
- Servidor web per monitorització i control d'aplicacions
- Watch Dog Timer
- 2 modes de baix consum (Idle: CPU OFF, perifèrics ON ; Sleep: CPU OFF, perifèrics OFF)

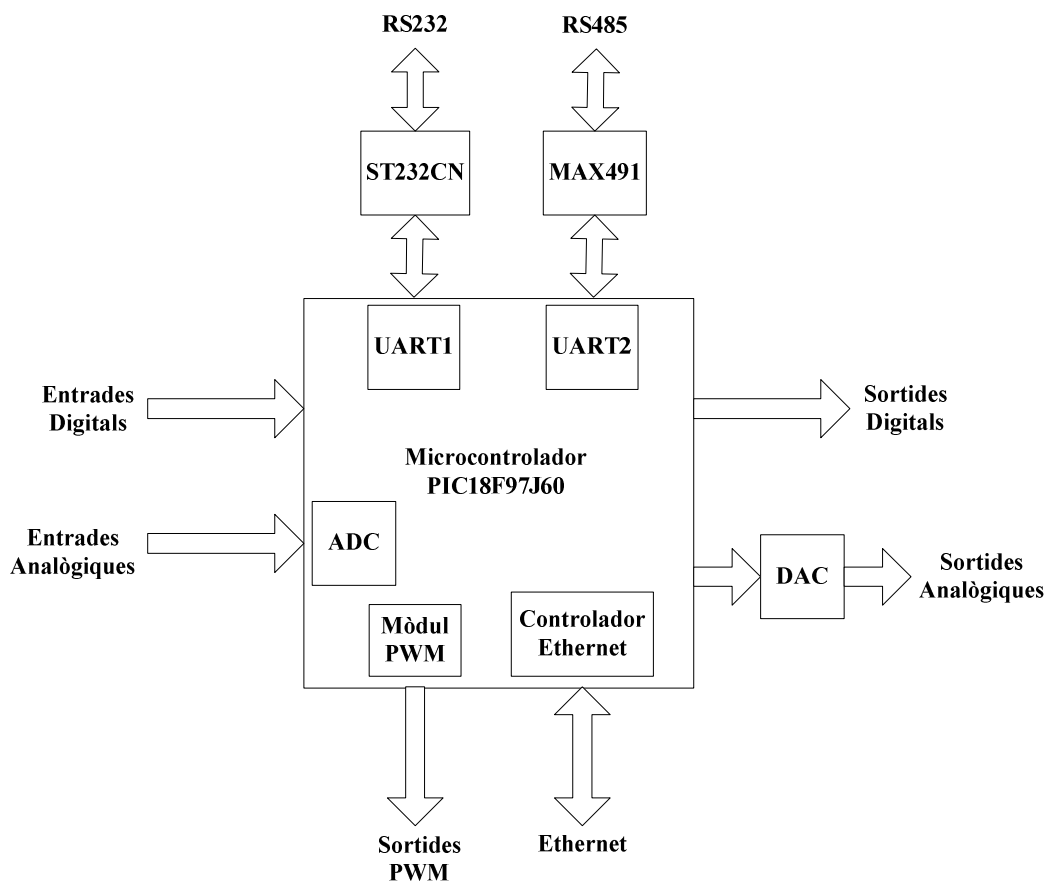


## 5. SOLUCIÓ TECNOLÒGICA

### 5.1. Introducció

En aquest capítol s'expliquen les solucions adoptades per dotar al mòdul microcontrolador universal de totes les funcionalitats previstes explicades anteriorment, així com complir amb les especificacions tècniques. També s'ha intentat incloure la informació necessària de cada element que conforma el mòdul per entendre el seu funcionament. Al CD que s'adjunta amb aquest document es pot trobar més informació sobre aquests elements, amb tots els data sheets i notes d'aplicació utilitzats en la realització del projecte.

La següent figura ens mostra el diagrama de blocs funcional del mòdul microcontrolador universal, on es poden diferenciar els elements principals que el formen.



**Fig. 5.1. Diagrama de blocs de mòdul microcontrolador universal**  
(Font: Elaboració pròpia)

## 5.2. Microcontrolador

### 5.2.1. Criteris de selecció

Els criteris de selecció que s'han tingut en compte per l'elecció del microcontrolador són:

- **Processament de dades:** Entenem per processament de dades la capacitat del microcontrolador per realitzar les operacions de baix nivell amb la màxima rapidesa i eficiència possible. El mòdul microcontrolador universal inicialment està pensat per gestionar tasques de control genèriques, sense grans restriccions temporals ni en la resolució de les dades tractades. Per tant, ens decantarem per un microcontrolador de 8 bits. Per altra banda, al no treballar amb restriccions temporals fortes, la velocitat de processament no serà una característica decisiva en l'elecció del microcontrolador, però serà necessària una freqüència de treball superior als 10 MHz per tal de permetre una compatibilitat mínima amb la tecnologia Ethernet 10Base-T.
- **Capacitat de Memòria:** Entenem per capacitat de memòria la quantitat de dades que el nostre sistema pot emmagatzemar. S'ha de tenir en compte tant la memòria per emmagatzemar dades de manera permanent (no volàtil) i per emmagatzemar dades de manera temporal (volàtil). El microcontrolador ha de tenir la suficient capacitat per emmagatzemar una pàgina web senzilla, el codi de programa necessari per suportar-la (pila TCP/IP), capacitat per gestionar els paquets rebuts i enviats, i finalment capacitat per emmagatzemar el programa de l'aplicació general del mòdul.
- **Tipus de memòria:** En el mercat disposem de microcontroladors amb diferents tipus de memòria segons les necessitats. En el nostre cas, és necessari disposar d'una memòria fàcilment reprogramable per poder realitzar proves, per tant, utilitzarem memòries del tipus EEPROM o FLASH que són borrarables elèctricament. A més, tenen la capacitat de ser reprogramables "en circuit", es a dir, que no cal extreure el circuit integrat de la placa per tal de realitzar modificacions.



- **Entrada/Sortida:** Considerem els requeriments d'entrada/sortida com la capacitat per interactuar amb els diferents sensors i actuadors que poden formar un sistema que ha de ser capaç de gestionar el mòdul microcontrolador universal. En aquest aspecte, és important que el microcontrolador incorpori diferents tipus d'entrada/sortida per tal de minimitzar la utilització de circuits integrats externs, que augmentarien el tamany del mòdul i el seu cost. Un altre aspecte a tenir en compte és el nombre de ports dedicats a l'entrada/sortida, per tal de implementar totes les funcionalitats previstes per al mòdul amb un únic microcontrolador.
- **Comunicacions:** Considerem els requeriments de comunicacions com la capacitat per interactuar amb els diferents elements del mòdul, així com capacitat per interactuar amb altres mòduls o elements externs al mòdul, com pot ser un PC. És necessari que el microcontrolador disposi de suficients ports de comunicació per poder comunicar-se tal com s'ha previst en el disseny funcional del mòdul.
- **Consum:** Encara que aquest és un aspecte important per al mòdul microcontrolador universal, no estipularem cap tipus de consum mínim ni màxim sinó que ens adaptarem als requeriments dels microcontroladors i dels altres elements seleccionats. De totes maneres, és una característica a tenir en compte especialment en el cas d'utilitzar el mòdul com un sistema de control mòbil, i per tant que haguessin d'estar alimentats per bateries auxiliars. Inicialment està pensat per estar ubicat en un lloc fix, on es disposi d'una font d'energia per alimentar al mòdul, però s'ha de preveure la possible alimentació amb bateries. D'aquesta manera, és important que el microcontrolador incorpori modes de funcionament en baix consum.
- **Facilitat de desenvolupament:** Entenem per facilitat de desenvolupament la disponibilitat d'elements que ens permetin configurar el microcontrolador a través d'algun llenguatge de programació que requereixi poc esforç al programador per tal de desenvolupar el codi font i que sigui fàcilment modulable.

Existeixen diferents llenguatges amb els quals podem escriure el nostre codi, però a poder ser ens decantarem per utilitzar un llenguatge d'alt nivell, com pot ser el C, en lloc del llenguatge ensamblador en el qual es basen aquest tipus de dispositius.

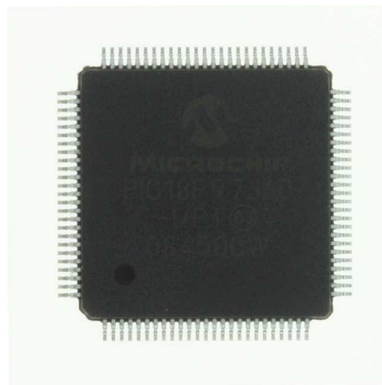
D'aquesta manera, el desenvolupament podrà ser més ràpid, amb un menor cost i oferint una modularitat efectiva. Cal destacar la complexitat del codi dels mòduls de la pila TCP/IP per descartar llenguatges de programació.

Caldrà a més, que el microcontrolador elegit disposi d'algun compilador eficient. Cal comentar també que el fet de no programar directament amb llenguatge màquina o assemblador resta eficiència al codi que finalment serà generat pel compilador.

### 5.2.2. El microcontrolador PIC18F97J60

Tenint en compte els punts descrits en el anterior apartat, s'ha optat com a primera opció la selecció d'un microcontrolador del fabricant Microchip. Un altre raó per la que ens hem decantat per Microchip és que es tracta d'un fabricant amb una llarga tradició dins del mercat dels microcontroladors, i especialment en els de 8 bits. A més, adjunta molta informació útil en els seus documents de suport accessibles des de la seva pàgina web, i existeixen una gran quantitat d'eines software gratuïtes per tal de programar els seus dispositius, ja sigui utilitzant llenguatges d'alt nivell com pot ser el C o amb llenguatge assemblador.

El microcontrolador utilitzat en el disseny del mòdul és el PIC18F97J60 del fabricant Microchip.



**Fig. 5.2. Microcontrolador PIC18F97J60**

**(Font: Microchip)**

Les principals raons per les quals s'ha escollit aquest microcontrolador són les següents:

- Aquest microcontrolador té incorporat un controlador d'Ethernet compatible amb l'estàndard IEEE 802.3. Això ens permet reduir el nombre de components externs per implementar el mòdul microcontrolador universal, simplificant les connexions i reduint l'espai necessari i el cost.
- Incorpora una gran quantitat de ports d'entrada/sortida, així com diferents funcionalitats per hardware. En total aquest microcontrolador té 70 pins d'entrada/sortida, i disposa de dos ports sèrie EUSART, dos ports sèrie MSSP (SPI o I2C), convertidor analògic-digital i mòdul PWM entre d'altres.
- Capacitat de memòria de programa Flash i memòria de dades RAM relativament gran, que ens permetrà implementar la pila TCP/IP, un servidor web senzill i el software de l'aplicació sense utilitzar memòria externa. En concret la capacitat és de 128 kbytes per a la memòria de programa i 3808 per a la de dades.

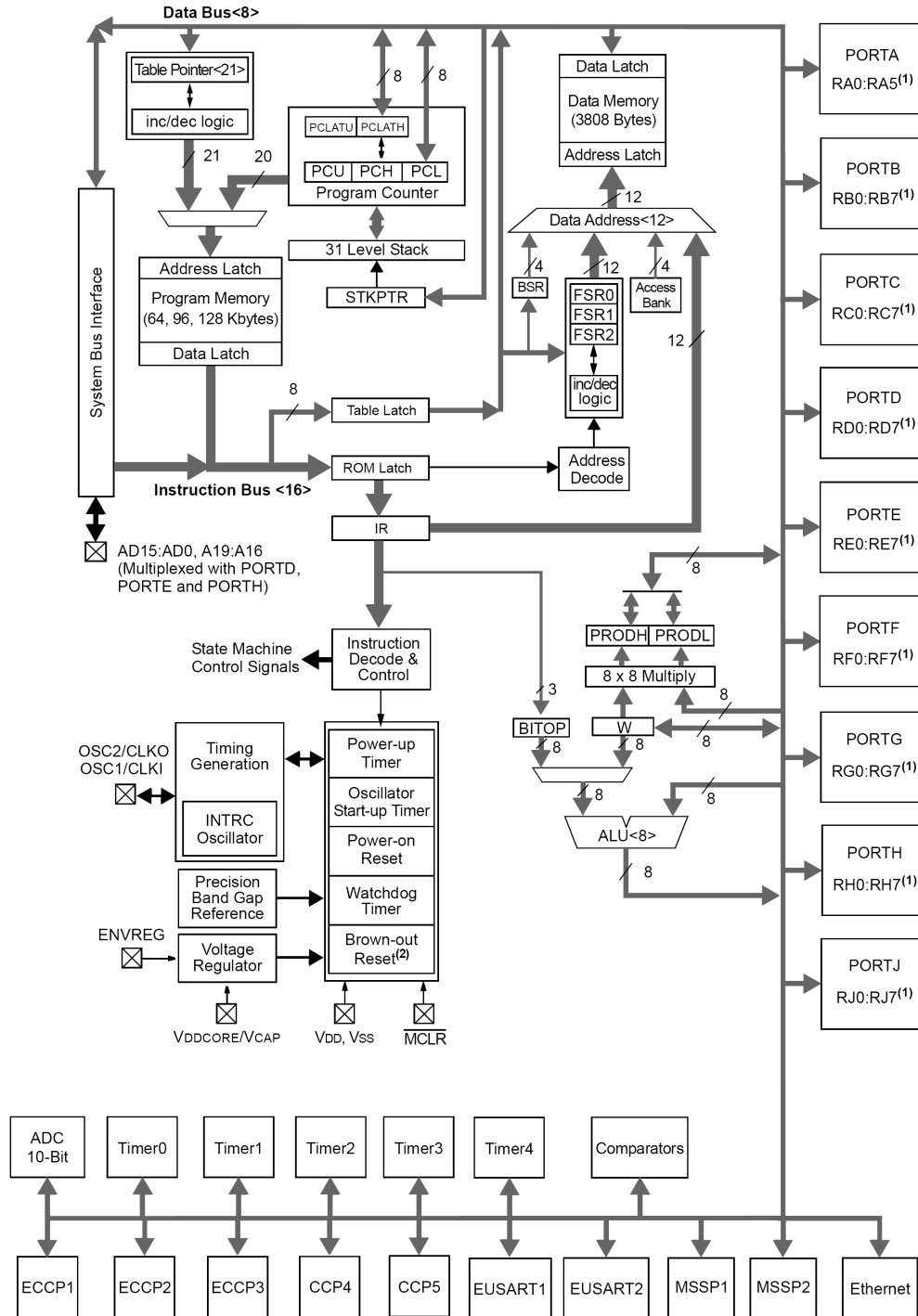
L'únic inconvenient amb el que ens hem trobat al utilitzar el PIC18F97J60 en el muntatge del prototip, és que només es troba disponible en el tipus d'encapsulat TQFP, amb unes mides molt reduïdes i de soldadura superficial. Per realitzar el muntatge del prototip, haurem de soldar el microcontrolador a un adaptador amb un forn de soldadura SMD. Aquest procés s'explica amb més detall al capítol 6.

També cal destacar, que això que per nosaltres és ara un inconvenient, en cas de fabricació en sèrie es convertiria en una avantatge, ja que al ser un encapsulat de dimensions tan reduïdes, implicaria unes dimensions del mòdul microcontrolador universal força més petites que les que tindrà el prototip.

Les principals característiques del microcontrolador PIC18F97J60 són les següents:

- Freqüència màxima d'operació: 41,667 MHz
- Memòria de programa FLASH: 128 kBytes
- Memòria de dades RAM: 3808 Bytes
- 29 fonts d'interrupció
- 9 ports d'E/S i 70 pins d'E/S
- 5 timers
- 5 mòduls PWM / Comparació / Captura
- 2 ports de comunicació sèrie USART
- 2 ports de comunicació sèrie MSSP (SPI o I2C)
- Controlador d'Ethernet (10Base-T)
- Convertidor analògic-digital de 10 bits amb 16 entrades
- Joc de 75 instruccions

Arquitectura interna del microcontrolador PIC18F97J60 a nivell de diagrama de blocs:



- Note** 1: See Table 1-6 for I/O port pin descriptions.  
 2: BOR functionality is provided when the on-board voltage regulator is enabled.

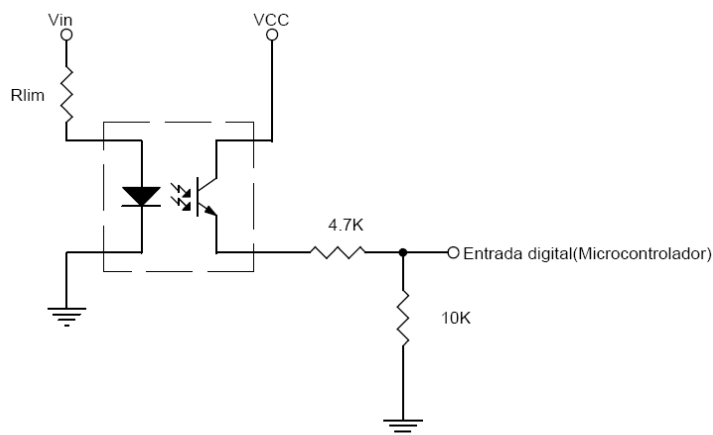
**Fig. 5.3. Arquitectura interna del microcontrolador PIC18F97J60**  
**(Font: Data Sheet PIC18F97J60)**

## 5.2. Entrades i sortides digitals

El mòdul microcontrolador universal disposa d'un total de 8 entrades digitals i 8 sortides digitals. Per al prototip no s'ha contemplat afegir protecció a les entrades digitals, per tant els nivells de tensió que accepten les entrades són 0 i 5 V.

Per a la utilització del mòdul en aplicacions industrials, on els valors de la tensió d'entrada siguin diferents o s'hagin d'aïllar elèctricament les entrades digitals, s'adaptaria un petit mòdul exterior amb els components necessaris per la protecció d'aquestes entrades.

La protecció de les entrades en aquest mòdul s'implementaria amb optoacobladors, ja que d'aquesta manera les entrades digitals del microcontrolador estarien aïllades elèctricament i protegides de l'entrada aplicada al mòdul. La tensió d'entrada per la qual s'han d'activar les entrades del mòdul quedaria fixada per el valor de la resistència limitadora de corrent del diode LED de l'optoacoblador, i la tensió aplicada a l'entrada del microcontrolador quedaria fixada per aquest mòdul exterior i independent de l'entrada externa. La següent figura mostra com es realitzaria aquesta protecció de les entrades digitals:



**Fig. 5.4. Esquema de la protecció de les entrades digitals (mòdul exterior)**

**(Font: Elaboració pròpia)**

A l'hora de seleccionar els ports del microcontrolador PIC18F97J60 per a la implementació de les entrades i sortides digitals, s'han tingut en compte les característiques dels ports d'entrada i sortida.

Els nivell de tensió d'entrada que accepta tenen dos possibles valors depenent dels pins dels ports d'entrada/sortida del microcontrolador.

Port or Pin	Tolerated Input	Description
PORTA<5,3:0>	VDD	Only VDD input levels tolerated.
PORTF<6:1> <sup>(1)</sup>		
PORTH<7:4> <sup>(2)</sup>		
PORTA<4>	5.5V	Tolerates input levels above VDD, useful for most standard logic.
PORTB<7:0>		
PORTC<7:0>		
PORTD<7:0> <sup>(1)</sup>		
PORTE<7:0>		
PORTF<7>		
PORTG<7:0> <sup>(1)</sup>		
PORTH<3:0> <sup>(2)</sup>		
PORTJ<7:0> <sup>(2)</sup>		

**Taula 5.1. Nivells de tensió d'entrada dels ports d'E/S del PIC18F97J60**

(Font: Data Sheet PIC18F97J60)

Per altra banda, totes les sortides digitals d'aquest microcontrolador ens proporcionen una tensió de 3,3 V, però la seva capacitat de subministrar corrent també varia segons els diferents ports d'entrada/sortida del microcontrolador.

Port	Drive	Description
PORTA <sup>(1)</sup>	Minimum	Intended for indication.
PORTF <sup>(2)</sup>		
PORTG <sup>(2)</sup>		
PORTH <sup>(3)</sup>		
PORTD <sup>(2)</sup>	Medium	Sufficient drive levels for external memory interfacing, as well as indication.
PORTE		
PORTJ <sup>(3)</sup>		
PORTB	High	Suitable for direct LED drive levels.
PORTC		

**Taula 5.2. Nivells de corrent de sortida dels ports d'E/S del PIC18F97J60**

(Font: Data Sheet PIC18F97J60)

En concret, el corrents de sortida que ens poden proporcionar cada pin dels diferents ports d'aquest microcontrolador són els següents:

- PORTA, PORTF, PORTG, POTH : 2 mA
- PORTD, PORTE, PORTJ : 8 mA
- PORTB, PORTC : 25 mA

Els ports B i C ens proporcionen un corrent de sortida suficient per connectar directament i activar càrregues com LEDs o relés d'estat sòlid de baixa potència.

Aquests nivells de tensió que accepten els ports d'entrada i el corrent que poden proporcionar els ports de sortida del microcontrolador ens imposen unes certes restriccions en el disseny del mòdul per combinar les diferents funcions que ha d'implementar.

Per a les entrades digitals, s'ha optat per utilitzar el PORTJ, que accepta un nivell de tensió d'entrada fins a 5,5 V.

Per a les sortides digitals s'ha tingut que combinar pins de diferents ports, ja que en el PORTB dos pins s'utilitzen per programació i depuració (ICD, In Circuit Debugger), i en el PORTC dos pins per al mòdul EUSART1, per la implementació de la connectivitat RS232, i també s'utilitzen pins per a les sortides PWM. D'aquesta manera, els pins que s'utilitzen per implementar la sortida digital són els següents:

- PORTB (RB0 – RB5)
- PORTC (RC4 i RC5)

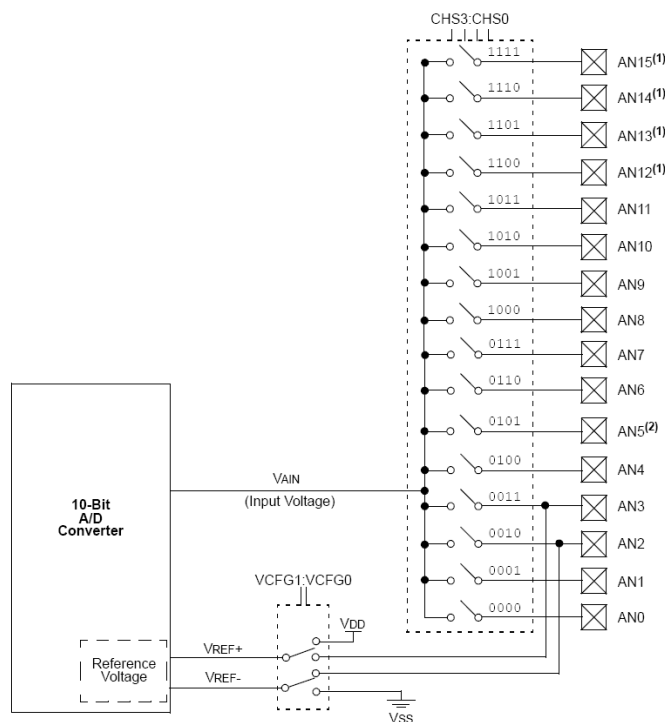
El mòdul exterior que s'ha comentat anteriorment també incorporaria elements per poder subministrar més corrent si necessitem actuar sobre una càrrega que necessita més potència per poder ser governada. Aquesta adaptació de les sortides es podria realitzar amb un circuit integrat ULN2003A (darlington array) amb capacitat de subministrar fins a 0,5 A i relés per actuar sobre les càrregues.



### 5.3. Entrades analògiques

El mòdul microcontrolador universal disposa d'un total de quatre entrades analògiques, amb un rang d'entrada de 0 – 3,3 V amb una resolució de 10 bits.

El microcontrolador PIC18F07J60 té incorporat un mòdul convertidor analògic-digital que té 16 canals analògics multiplexats a un Sample and Hold i a un convertidor analògic digital amb una resolució de 10 bits.



**Note** 1: Channels AN15 through AN12 are not available on 64-pin devices.  
2: Channel AN5 is implemented on 100-pin devices only.

**Fig. 5.5. Diagrama de blocs del convertidor A/D**  
(Font: Data Sheet PIC18F97J60)

La tensió de referència analògica es pot seleccionar per software entre dues possibilitats:

- La tensió d'alimentació positiva i negativa del microcontrolador (AVDD i AVSS)
- Els nivells de tensió en els pins RA3/AN3/VREF+ i RA2/AN2/VREF-

La sortida del Sample and Hold és l'entrada al convertidor, que genera un resultat per aproximacions successives que es pot llegir en els registres ADRESH i ADRESL del microcontrolador.

Les possibles configuracions i habilitacions del convertidor es realitzen a través dels següents registres:

- ADCON0. Controla el funcionament del convertidor A/D, en la conversió actual.
- ADCON1. Configura els diferents pins del convertidor (entrades analògiques o digitals) i la tensió de referència.
- ADCON2. Configura la font de rellotge del convertidor, el temps d'adquisició i el format del resultat (justificació dreta o esquerra).

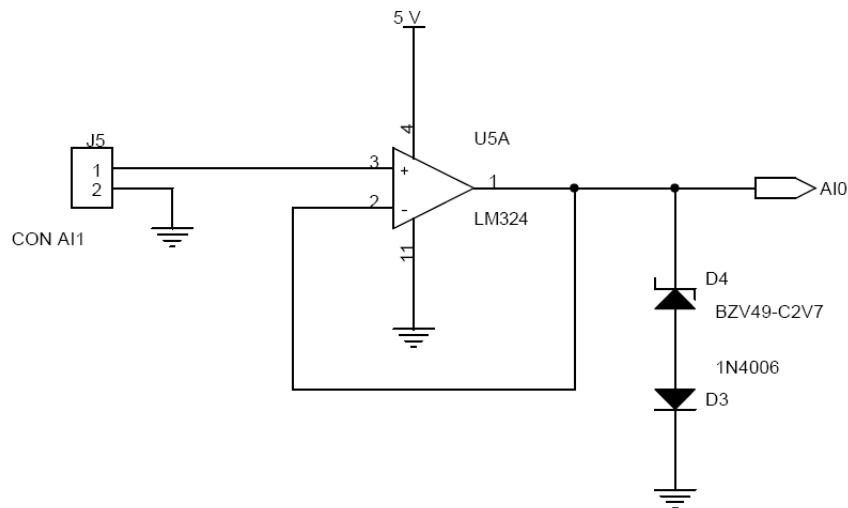
Per les diferents funcionalitats que ha d'incloure el mòdul microcontrolador universal, que imposen certes restriccions a l'hora d'assignar els pins del PIC18F97J60, les quatre entrades analògiques es connectaran als pins AN5 – AN8.

### **Acondicionament de les entrades analògiques**

Per protegir les entrades analògiques del microcontrolador s'implementa un circuit limitador de tensió, on posteriorment es connecta l'entrada analògica del microcontrolador.

El circuit limitador de tensió s'ha calculat de manera que no superi els nivells de tensió que toleren les entrades analògiques del microcontrolador.

Els voltatges de les entrades analògiques del microcontrolador venen determinats per les tensions de referència del convertidor A/D. En aquest muntatge s'ha optat per utilitzar les tensions d'alimentació positiva i negativa del microcontrolador (AVDD i AVSS), que representen el rang més ampli possible, de 0 a 3,3 V.



**Fig. 5.6. Limitació de tensió de les entrades analògiques**  
(Font: Elaboració pròpia)

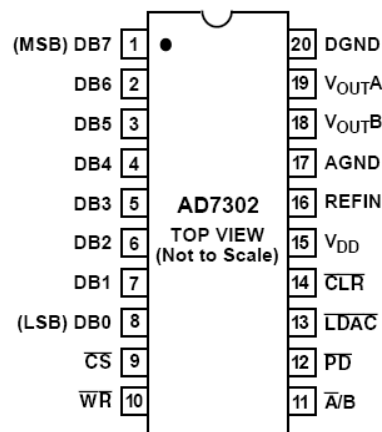
A la figura anterior, AI0 és l'entrada analògica del microcontrolador. Els amplificadors operacionals utilitzats són de baixa potència, i la seva tensió d'alimentació és de 5 V.

Per a la utilització del mòdul en aplicacions industrials, on els valors de la tensió d'entrada analògica segueixen uns estàndards, s'adaptaria un petit mòdul exterior amb els components necessaris per adaptar els nivells de tensió de les entrades al rang d'entrada del microcontrolador de 0 a 3,3 V. Normalment, els sensors que s'utilitzen a la indústria (temperatura, pressió, etc.) donen una senyal de tensió o corrent normalitzada de 0 – 10 V o 4 – 20 mA.

## 5.4. Sortides analògiques

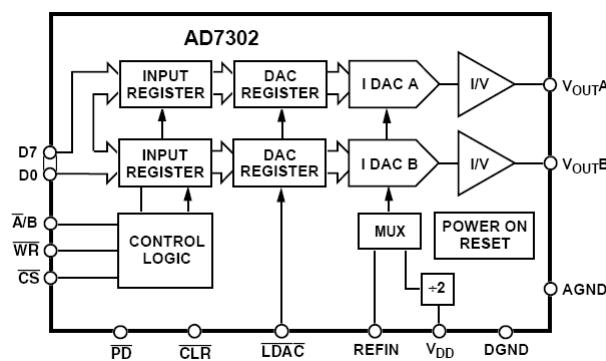
### Convertidor digital – analògic AD7302

El AD7302 és un convertidor digital analògic (DAC) amb una resolució de 8 bits amb doble sortida fabricat per ANALOG DEVICES, ideal per a connexions amb microprocessadors. Incorpora buffers amplificadors de sortida que permeten la implementació de dues sortides analògiques amb un sol circuit integrat.



**Fig. 5.7. Configuració dels pins del AD7302**  
(Font: Data Sheet AD7302)

El rang d'alimentacions d'aquest DAC va des dels 2,7 V fins als 5,5 V, compatible amb les tensions d'alimentació dels diferents elements del mòdul. El baix consum que presenta aquest DAC el fa ideal per dispositius alimentats amb bateria: 10 mW i 15 mW amb alimentacions de 3,3 V i 5 V respectivament, reduint-se considerablement en mode de baix consum.



**Fig. 5.8. Diagrama de blocs funcional del AD7302**  
(Font: Data Sheet AD7302)

Disposa de vuit entrades en paral·lel de dades digitals que es connecten a un registre d'entrada previ al DAC dependent de les senyals de control Chip Select ( $\overline{CS}$ ), Write Input ( $\overline{WR}$ ) i DAC Select ( $\overline{A/B}$ ), aquest últim selecciona l'escriptura de la dada al DAC A o al DAC B.

L'entrada  $\overline{PD}$  s'utilitza per posar al DAC en mode de baix consum.

L'entrada Load DAC Logic Input ( $\overline{\text{LDAC}}$ ) carrega els DACs amb els valors dels seus registres. Si  $\overline{\text{LDAC}}$  està permanentment a nivell baix, els DACs es carreguen amb els flancs del senyal  $\overline{\text{WR}}$ .

L'entrada Asynchronous Clear Input ( $\overline{\text{CLR}}$ ) carrega els registres dels DACs amb zeros, posant a zero la tensió de les dues sortides.

L'entrada REFIN (External Reference Input) s'utilitza com l'entrada de referència per als DACs, que pot anar des de 1 V fins a  $V_{DD}/2$ . Si REFIN es connecta a  $V_{DD}$ , es selecciona la referència interna de  $V_{DD}/2$  V.

La tensió de sortida que ens proporciona aquest DAC en funció de l'entrada digital de vuit bits introduïda es mostra en la següent taula:

<b>Entrada digital (MSB ... LSB)</b>	<b>Sortida analògica</b>	<b>Sortida analògica amb <math>V_{REF}=2.5</math> V</b>
1111 1111	$2 \times 255/256 \times V_{REF}$ V	4,98046875 V
1111 1110	$2 \times 254/256 \times V_{REF}$ V	4,9609375 V
1000 0001	$2 \times 129/256 \times V_{REF}$ V	2,519553125 V
1000 0000	$V_{REF}$ V	2,5 V
0111 1111	$2 \times 127/256 \times V_{REF}$ V	2,48046875 V
0000 0001	$2 \times V_{REF}/256$ V	0,01953125 V
0000 0000	0 V	0 V

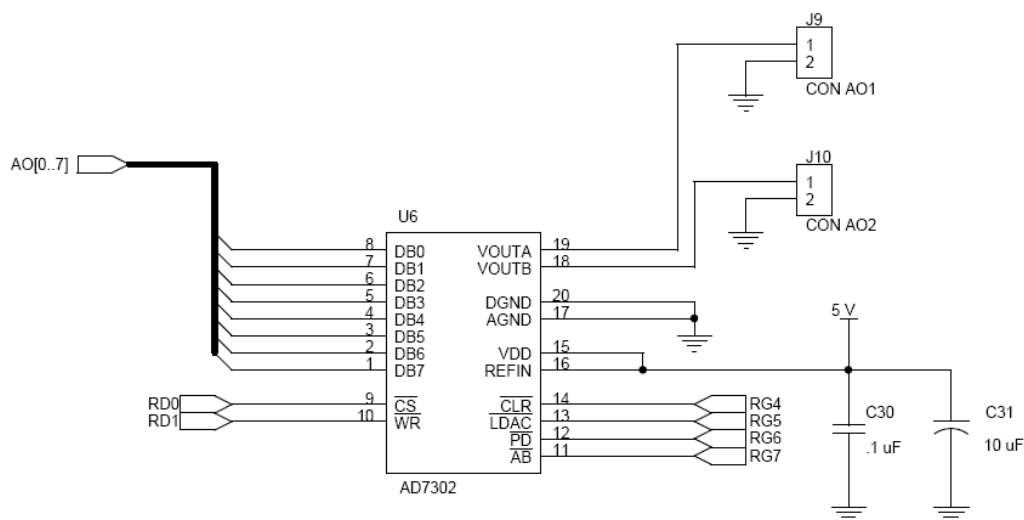
**Taula 5.3. Tensions de sortida del DAC AD7302**

(Font: Data Sheet AD7302)

En el muntatge del prototip utilitzarem una tensió d'alimentació per al convertidor digital-analògic AD7302 de 5 V, i l'entrada de referència externa la connectarem directament a  $V_{DD}$ , per tant la referència interna serà de 2,5 V i la tensió de cada una de les sortides analògiques correspondrà als valors d'aquesta taula.

Per realitzar les connexions entre el DAC AD7302 i el microcontrolador PIC18F97J60 per tal d'implementar les sortides digitals s'han utilitzat els següents pins del microcontrolador:

- PORTH: 8 línies connectades a les entrades digitals del DAC
- PORTD (RD0 i RD1): senyals de control del DAC  $\overline{CS}$  i  $\overline{WR}$
- PORTG (RG4 – RG7): senyals de control del DAC  $\overline{CLR}$ ,  $\overline{LDAC}$ ,  $\overline{PD}$  i  $\overline{A/B}$



**Fig. 5.9. Connexionat del DAC AD7302 en el mòdul**  
(Font: Elaboració pròpia)

### Acondicionament de les sortides analògiques

Com s'ha vist anteriorment, el convertidor digital – analògic AD7302 proporciona una tensió analògica de sortida proporcional a la tensió de referència utilitzada. En el cas del mòdul microcontrolador universal, s'utilitza una tensió de referència de 2,5 V, per tant el rang de sortida serà de 0 – 5 V.

En el cas d'utilitzar el mòdul en aplicacions industrials, pot ser que es necessiti un rang més ampli per realitzar accions de control. Alguns d'aquests rangs són de 0 – 10 V,  $\pm 10$  V o 4 – 20 mA. Per realitzar aquesta adaptació de senyals, com en els altres casos, s'adaptaria un petit mòdul exterior que realitzaria aquesta funció. En el cas d'adaptar la sortida a nivells de tensió més amplis es podria realitzar amb uns quants components passius i

amplificadors operacionals, o amb circuits integrats que realitzen aquestes funcions, així com disposar d'una font de tensió d'alimentació de 10 V.

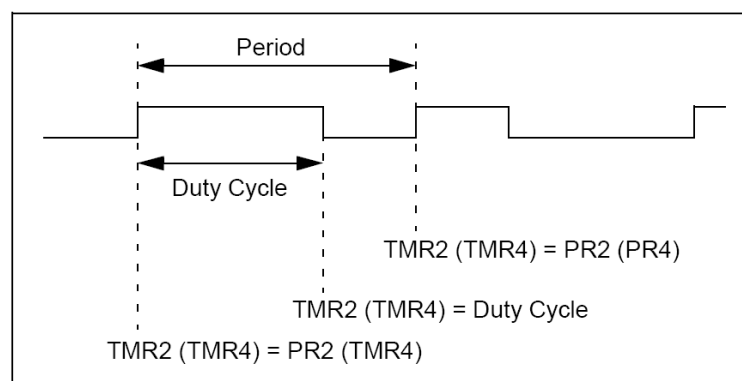
## 5.5. Sortides PWM

El microcontrolador PIC18F97J60 disposa de fins a 5 mòduls de Captura/Comparació/PWM (CCP). En el mòdul microcontrolador universal s'utilitzen tres d'aquests mòduls per implementar els dos tipus de sortides PWM dels quals disposa: PWM per control de motors de corrent continu i PWM per control d'onduladors trifàsics.

En concret s'utilitzen 3 sortides del mòdul 1 i 3 sortides del mòdul 2 per implementar les sortides PWM per al control d'onduladors trifàsics i 4 sortides del mòdul 3 per al control de motors de corrent continu. Cal destacar que en el cas del control de motors de corrent continu, els mòduls disposen de funcionalitats que permeten una fàcil configuració, mentre que en el cas trifàsic, s'hauria de programar per software.

Dels 5 mòduls CCP que conté el PIC18F97J60, dos d'ells són estàndards, i els altres tres són una versió millorada que els anteriors (ECCP). Aquests últims permeten realitzar aquestes funcions específiques que ja estan cablejades per hardware.

Quan aquests mòduls treballen en mode PWM, els pins d'aquests mòduls produeixen una sortida PWM amb una resolució de 10 bits. Per tal de generar les sortides PWM, aquest mòduls treballen amb el Timer2 o Timer4.



**Fig 5.10. Sortida PWM**  
(Font: Data Sheet PIC18F97J60)

Per determinar el període de la sortida PWM s'utilitza el registre PR2 (o PR4). Aquest període es calcula amb la següent expressió:

$$T_{\text{PWM}} = [(PR2)+1] \cdot 4 \cdot T_{\text{OSC}} \cdot (\text{valor prescaler TMR2})$$

El duty cycle de la sortida PWM (fins a 10 bits de resolució) s'especifica amb el registre CCPRxL i els bits 5 i 4 del registre CCPxCON.

Per calcular el duty cycle de les sortides PWM s'utilitza la següent expressió:

$$\text{PWM Duty Cycle} = (\text{CCPRxL:CCPxCPN}<5:4>) \cdot T_{\text{OSC}} \cdot (\text{valor prescaler TMRx})$$

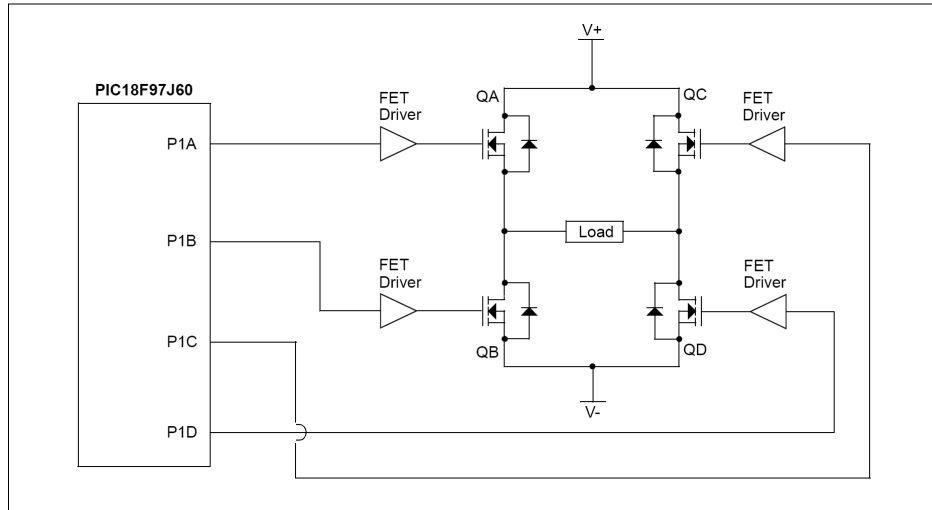
Per configurar un mòdul CCP per treballar com a mòdul PWM s'han de seguir els següents passos:

- Fixar el període de la sortida PWM escrivint al registre PR2 (o PR4).
- Fixar el duty cycle escrivint al registre CCPRxL i als bits 4 i 5 del registre CCPxCON
- Configurar els pins de les sortides PWM esborrant el corresponents bits TRIS dels ports de sortida.
- Fixar el valor del prescaler TMR2 (o TMR4), i posteriorment habilitar el Timer2 (o Timer4).
- Configurar el mòdul CCPx corresponent per treballar en mode PWM.

Com s'ha comentat abans, tres dels mòduls disposen de característiques diferents a la dels mòduls estàndard, proporcionant funcionalitats per hardware, sense tenir que resoldre un algoritme concret.

Algunes de les funcions que tenen incorporades aquests tres mòduls ECCP són configuracions amb 1, 2 o 4 canals de sortida, polaritat seleccionable, control de banda morta, apagada automàtica en cas de mal funcionament, i diferents possibilitats de configurar les sortides, com per exemple en pont complet.





**Fig. 5.11. Exemple d'aplicació en pont complet**

**(Font: Data Sheet PIC18F97J60)**

Per implementar la sortida PWM trifàsica s'utilitzen els següents mòduls i sortides d'aquests:

Mòdul ECCP1: sortides PWM A, B i C.

Mòdul ECCP2: sortides PWM A, B i C.

Per implementar la sortida PWM per control de motors de corrent continu s'utilitza el següent mòdul i sortides d'aquest:

Mòdul ECCP3: sortides PWM A, B, C i D.

## 5.6. Connectivitat RS232

Per realitzar la comunicació entre el mòdul microcontrolador universal i un PC s'utilitzarà l'estàndard RS232.

Aquest estàndard, elaborat per la EIA (Electronics Industries Alliance), defineix la interfase mecànica, els pins, els senyals i els protocols que ha de complir aquesta comunicació sèrie.

En la versió més simple només s'utilitzen dos bits per a la comunicació sèrie, el senyal de transmissió (TX) i el de recepció (RX).

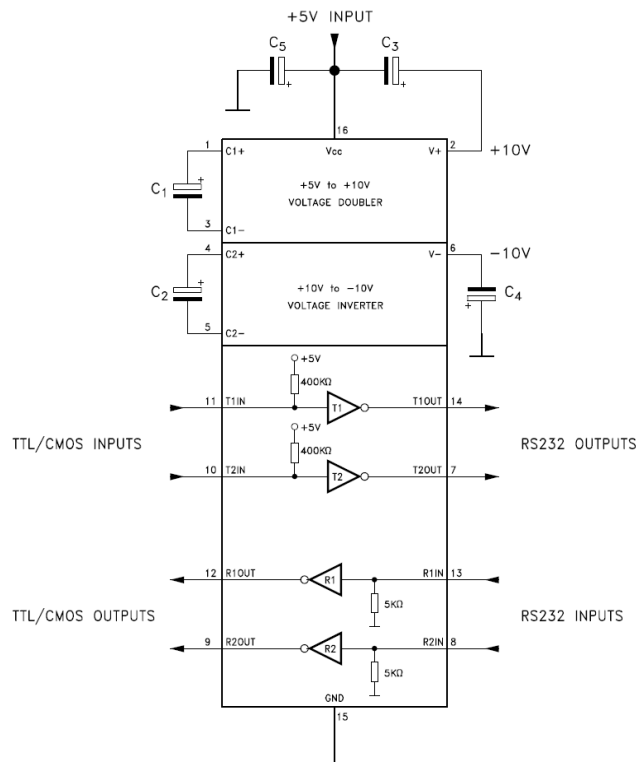
La norma RS232 compleix amb els següents nivells de tensió:

- Un '1' lògic és un voltatge comprés entre -5 i -15 V en el transmissor i entre -3 i -25 V en el receptor.
- Un '0' lògic és un voltatge comprés entre +5 i +15 V en el transmissor i entre +3 i +25 V en el receptor.

Els mòduls UART del PIC18F9J60 treballen amb nivells de tensió digital, de manera que s'han d'adaptar als de l'estàndard RS232.

Per fer la conversió dels nivells de tensió del mòdul UART als de la norma RS232 s'utilitza el circuit integrat ST232CN.

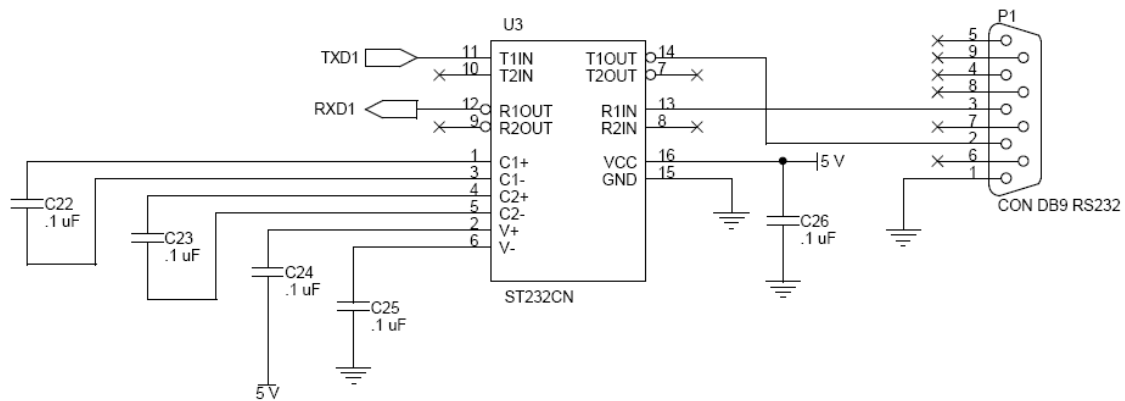
Aquest circuit integrat soluciona els problemes de nivells de tensió quan es tenen que enviar senyals digitals sobre un línia RS232. Es tracta d'una solució adequada per aplicacions on no es disposa de fonts de  $\pm 12$  V. El ST232CN només necessita per al seu funcionament una única font de +5 V i cinc condensadors de 0.1  $\mu$ F. Internament disposa d'un elevador de tensió que converteix la tensió de +5 V a una de doble polaritat de  $\pm 10$  V. En el mercat es disposa d'una gran varietat de circuits integrats que compleixen amb la norma RS232 i que són similars al ST232CN.



**Fig. 5.12. Configuració del ST232CN**  
(Font: Data Sheet ST232CN)

A la figura 5.12 es poden observar els cinc condensadors necessaris per al correcte funcionament d'aquest circuit i la configuració interna del ST232CN.

Aquest circuit disposa de dos drivers i dos receivers, encara que per la implementació del port RS232 en el mòdul microcontrolador universal només utilitzarem un driver i un receiver, així com la configuració més simple de la norma RS232, amb els senyals de transmissió i recepció.



**Fig. 5.13. Connexionat del ST232CN en el mòdul  
(Font: Elaboració pròpia)**

A la figura 5.13 es pot observar el connexionat del ST232CN en el mòdul microcontrolador universal, on TXD1 és la sortida TX i RXD1 és l'entrada RX del mòdul UART1 del PIC18F97J60. La sortida del driver T1 es connecta al pin 2 del connector DB9 i l'entrada del receiver 1 es connecta al pin 3 del connector DB9, seguint amb la norma RS232. Els condensadors de 0.1  $\mu\text{F}$  es connecten seguint les recomanacions del fabricant.

## 5.6. Connectivitat RS485

Per poder implementar una xarxa de mòduls i permetre la comunicació entre ells, s'utilitzarà l'estàndard RS485.

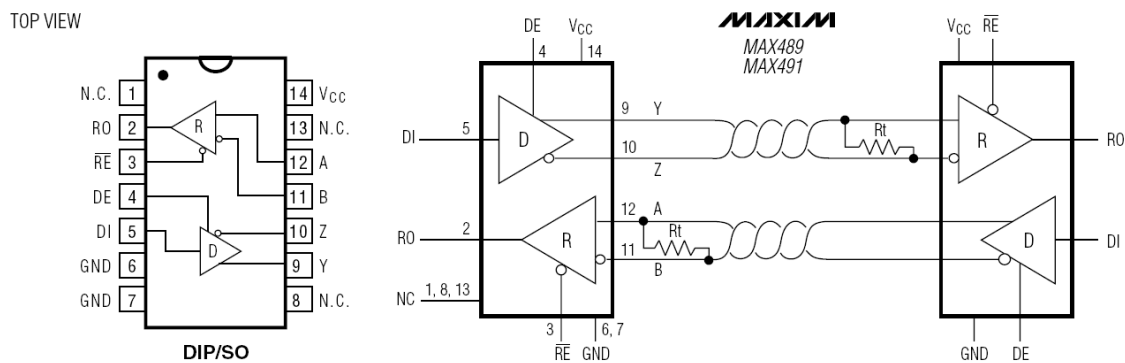
La norma RS485, elaborada per la EIA (Electronics Industries Alliance), és un sistema de transmissió sèrie que utilitza senyals diferencials, per el que és necessari utilitzar dos cables per cada línia.

Aquest sistema permet la connexió de múltiples transmissors i receptors. Depenent de la distància, velocitat de transmissió i els circuits integrats que s'utilitzin, es poden connectar fins a 32 nodes amb un simple parell de cables.

La distància entre un receptor i un emissor amb RS485 pot arribar fins als 1200 metres. La transmissió en llargues distàncies es possible gràcies a la transmissió diferencial.

Per fer la conversió dels nivells de tensió del mòdul UART del PIC18F97J60 als senyals de tipus diferencial que s'utilitzen en un bus RS485, s'utilitza el circuit integrat MAX491.

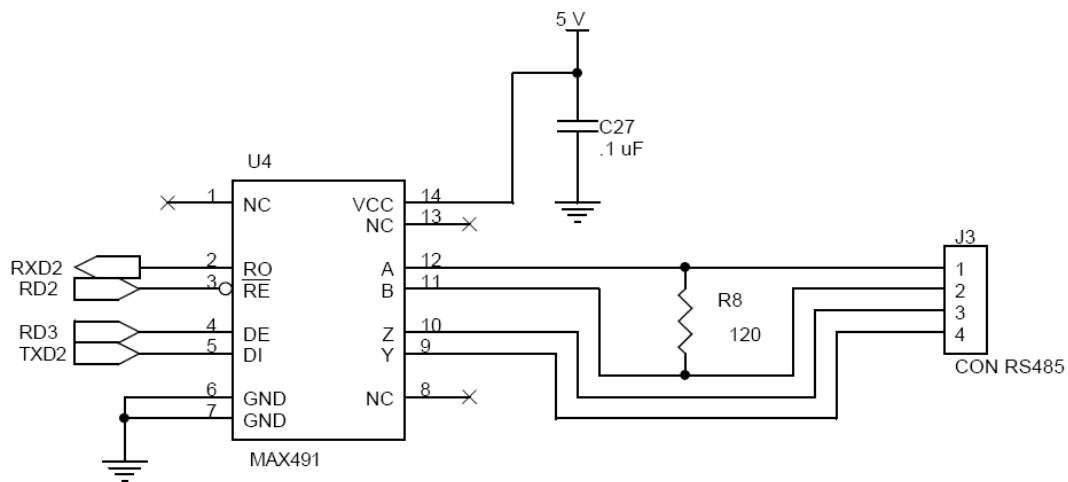
Aquest circuit integrat soluciona els problemes de nivells de tensió quan es tenen que enviar senyals digitals sobre una línia RS485. El MAX491 disposa d'un driver per la transmissió de dades i un receiver per la recepció de dades, i senyals de control que permeten activar o desactivar el driver o el receiver. Aquesta última característica permet connectar més drivers i receivers en una mateixa línia, implementant la connexió de múltiples transmissors i receptors.



**Fig. 5.14. Configuració de pins del MAX491 i circuit típic d'operació**  
(Font: Datasheet MAX491)

A la figura 5.14 es pot observar el circuit típic d'operació del MAX491, amb configuració full-duplex entre dos dispositius, on cada un pot ser transmissor i receptor alhora.

La resistència  $R_t$  que apareix en aquesta configuració és la resistència de terminació de línia. La funció d'aquesta resistència és evitar el fenomen de reflexió, que ens pot retornar el senyal enviat per un parell trenat. Aquesta resistència de terminació de línia s'ha d'instal·lar en els receivers del primer i últim dispositiu de la línia. El valor per aquesta resistència és de  $120 \Omega$ .



**Fig. 5.15. Connexionat del MAX491 en el mòdul**  
(Font: Elaboració pròpia)

A la figura 5.15 es pot observar el connexionat del MAX491 en el mòdul microcontrolador universal, on TXD2 és la sortida TX i RXD2 és l'entrada RX del mòdul UART2 del PIC18F97J60. Els senyals RD2 i RD3 són dos pins del PORTD del microcontrolador que s'utilitzen com senyals de control per activar o desactivar el driver o el receiver. Els senyals diferencials d'entrada i sortida, A-B i Z-Y, es porten a un connector de regleta per la connexió d'un bus extern al mòdul microcontrolador universal, per tal d'implementar una xarxa de controladors.

## **5.7. Connectivitat Ethernet**

### **5.7.1. Ethernet**

Ethernet és un estàndard que defineix un mètode d'accés múltiple orientat a sistemes de comunicacions per el transport de dades digitals a través de xarxes locals, que compren el nivell físic i el nivell d'enllaç del model de referència OSI. Ethernet defineix les característiques de cablejat, tipus de dades i trames de dades que es transmeten en una xarxa d'àrea local, i que segueixen l'estàndard IEEE 802.3.

La xarxa Ethernet va ser desenvolupada per la companyia Xerox. Degut al gran èxit d'aquesta xarxa, Xerox, DEC (Digital Equipment Corporation) i Intel van desenvolupar una norma per la xarxa Ethernet de 10 Mbit/s, que posteriorment va ser utilitzat com la base per definir l'estàndard IEEE 802.3. Aquest estàndard, no només defineix les especificacions d'Ethernet, si no que descriu tota una família de xarxes d'àrea local entre 1 Mbit/s i 10 Gbit/s a l'actualitat, amb futures ampliacions previstes.

Tots els equips d'una xarxa Ethernet estan connectats a la mateixa línia de transmissió i la comunicació es realitza mitjançant la utilització del protocol denominat CSMA/CD (Carrier Sense Multiple Access with Collision Detect). En aquest protocol, els equips verifiquen l'estat de la línia abans de transmetre, si hi ha col·lisió els equips interrompen la comunicació i esperen un període de temps aleatori per tornar a transmetre.

### 5.7.2. Controlador d'Ethernet

El controlador d'Ethernet que incorpora el microcontrolador PIC18F97J60 proporciona una solució completa per a la connectivitat a Internet, incloent els mòduls Media Access Control (MAC) i Physical Layer transceiver (PHY). Amb només dos transformadors i uns quants components passius, es pot connectar aquest microcontrolador directament a una xarxa Ethernet.

El mòdul Ethernet del microcontrolador compleix la norma IEEE 802.3 que correspon a les especificacions per connectivitat 10-BaseT amb cable de parell trenat.

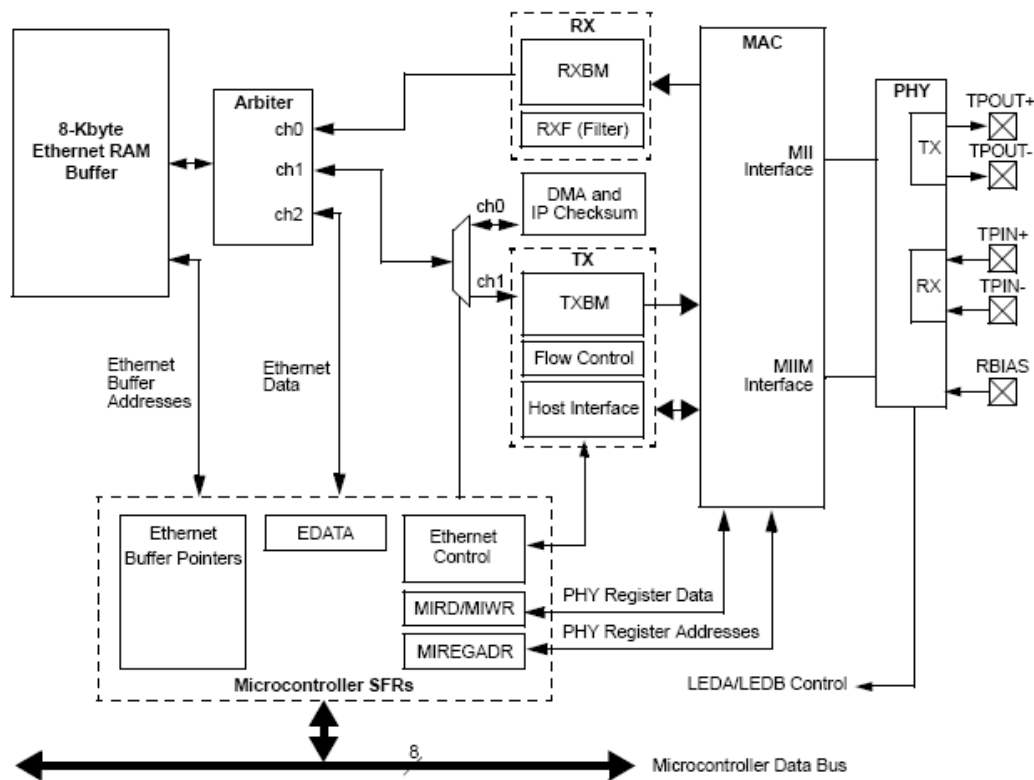


Fig. 5.16. Diagrama de blocs del mòdul d'Ethernet

(Font: Data Sheet PIC18F97J60)



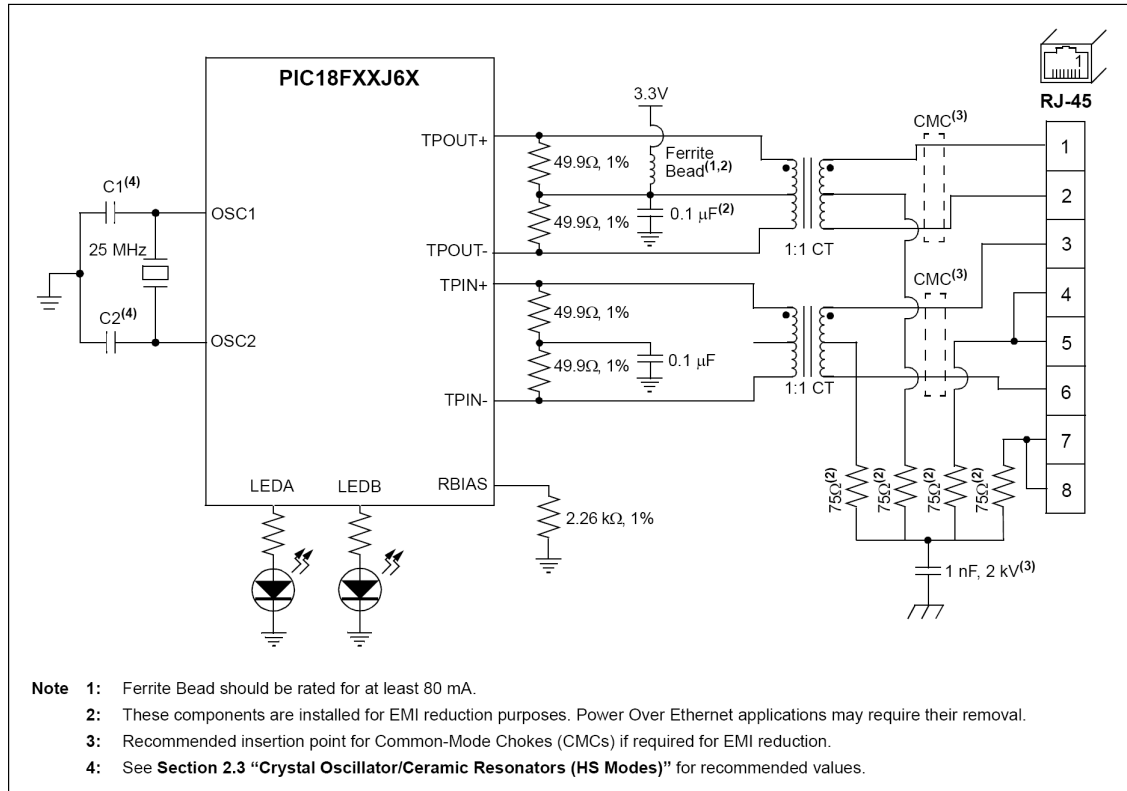
El mòdul d'Ethernet està format per els següents elements principals:

- Mòdul transceiver PHY. Aquest mòdul s'encarrega de codificar i descodificar les dades analògiques que estan presents en el parell trenat i les envia cap a la xarxa o les rep d'aquesta.
- Mòdul MAC. Aquest mòdul permet implementar l'estàndard IEEE 802.3, empaquetant les dades a transmetre complint amb aquesta norma, i transformant els paquets que arriben en dades per al microcontrolador. Proporciona una gestió d'interfase per controlar el mòdul PHY.
- Memòria RAM de 8 Kbytes. El mòdul d'Ethernet incorpora una memòria RAM de 8 Kbytes independent utilitzada com a buffer per emmagatzemar temporalment els paquets de dades rebuts o a enviar.
- Arbitrer. S'encarrega de controlar l'accés a la memòria RAM quan les sol·licituds són realitzades per la CPU del microcontrolador, per el mòdul DMA i per els blocs de transmissió i recepció RX i TX.
- Registres de Funció Específica (SFRs). S'encarrega de controlar el mòdul d'Ethernet i d'indicar l'estat de les comunicacions.

### **Requeriments del mòdul d'Ethernet:**

El mòdul Ethernet està dissenyat per treballar a 25 MHz, que ha de proporcionar la font de rellotge principal del microcontrolador, ja sigui amb un cristall amb freqüència de ressonància de 25 MHz connectat entre els pins OSC1 i OSC2 o amb una font de rellotge externa connectada al pin OSC1 (entrada de rellotge).

Per tal de completar l'interfase amb Ethernet, el mòdul necessita una sèrie de components estàndards extres. A la següent figura es mostren aquests components.



**Fig. 5.17. Components externs necessaris per el mòdul d'Ethernet**

(Font: Data Sheet PIC18F97J60)

El circuit analògic intern del mòdul PHY requereix la instal·lació d'una resistència externa de 2.26 kΩ des de la sortida RBIAS fins a massa. Aquesta resistència és necessària ja que influeix en l'amplitud del senyal diferencial TPOUT+/-.

També és necessari muntar en els pins TPIN+/TPIN- i TPOUT+/TPOUT- uns transformadors 1:1 amb derivació central, que aïllen el circuit de la xarxa, eviten reflexions i adapten la capacitat del cable, permetent una comunicació més neta.

Juntament amb aquests transformadors, s'han de incorporar els components passius que ens recomana Microchip i que podem observar a l'esquema, com les diferents resistències, condensadors i una 'ferrite bead', aquest últim component s'utilitza per eliminar soroll d'alta freqüència.

A la pràctica ens trobem que molts fabricants incorporen en els seus connectors RJ-45 la majoria d'aquests components. Utilitzant aquest tipus de connectors ens podem estalviar la tasca de buscar i comprar aquests transformadors i l'espai necessari per el seu muntatge.

El connector RJ-45 utilitzat per al muntatge dels dos prototips incorpora els transformadors, les resistències de 75  $\Omega$ , el condensador de 1000 pF i els dos LEDs indicadors de l'estat de la comunicació per el mòdul Ethernet.

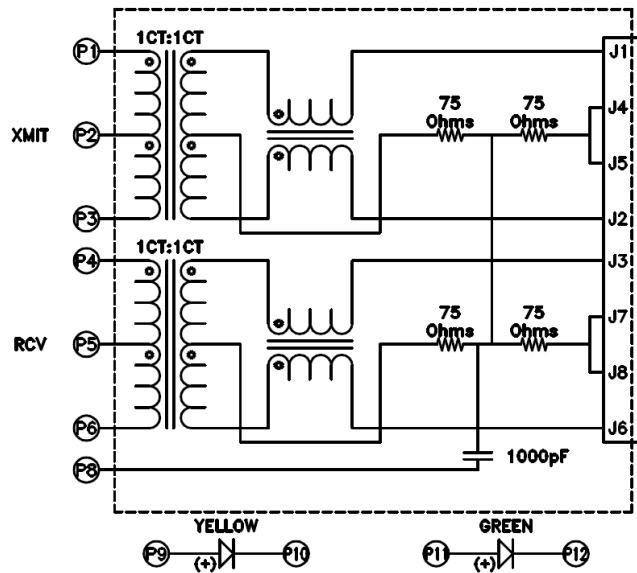


Fig. 5.18. Esquema del connector RJ-45 utilitzat  
(Font: Data Sheet WE-LAN RJ45)

### 5.7.3. Piles de protocols

Una pila de protocols és una arquitectura de protocols que està organitzada en diferents capes, cadascuna lligada a una tasca determinada. A cada capa de la pila s'utilitza un protocol que es comunica amb el respectiu nivell del dispositiu de destí. En una pila de protocols, cada capa té la funció de proporcionar serveis a la capa immediatament superior.

#### 5.7.3.1. Model OSI

A l'any 1984 la ISO (International Standards Organization) va publicar el Model OSI (Open Systems Interconnect), amb la finalitat de establir un model de referència de piles de protocols. En aquest model OSI, queden definides les diferents capes que ha de tenir una pila de protocols. L'objectiu del Model OSI era crear un estàndard per a la organització de protocols en les comunicacions en xarxa.

Els objectius d'aquest model eren:

- Establir un estàndard que permetés la comunicació entre dispositius de diferents fabricants
- Definir les funcions específiques de cada nivell de la pila, així com donar una guia per a la implementació de la comunicació entre capes
- Detallar el conjunt de protocols que s'hauria d'utilitzar a cada capa

Un dels punts a destacar de la pila OSI és que cada nivell actua com una unitat independent. Això vol dir que (teòricament), es podria substituir el protocol que s'utilitza a una certa capa per un altre protocol sense que els nivells superior e inferior a aquesta capa en surtin afectats.

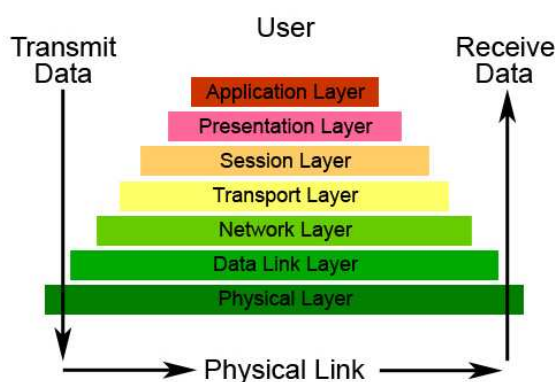


Fig. 5.19. Capes del Model OSI

El model OSI ve definit per 7 nivells:

- Aplicació: Interfície que comunica les fonts i destins de dades amb el sistema de comunicació.
- Presentació: Adapta el format de les dades i la seva codificació per a que es pugui transmetre entre sistemes diferents.
- Sessió: Gestiona el diàleg entre dispositius (inici, final, etc.)
- Transport: Controla la transmissió extrem a extrem, gestiona els errors, controla el flux de dades, etc.
- Xarxa: Estableix la ruta a seguir per els paquets

- Enllaç: S'encarrega del lliurament de dades entre nodes d'un enllaç de xarxa. Assegura la fiabilitat del mitjà de transmissió, realitza controls d'errors, retransmissions, etc.
- Físic: S'encarrega de passar els bits al mitjà de transmissió (cable de coure, aeri, fibra, etc.)

Aquest model es va realitzar, com ja s'ha comentat, per a definir un estàndard en quant a piles de protocols, però en el moment de la seva creació, ja existien altres piles molt utilitzades, com TCP/IP.

### 5.7.3.2. Pila TCP/IP

Encara que el model OSI sigui l'estàndard, abans que aquest model sortís a la llum, es va desenvolupar una família de protocols que ja realitzava les operacions descrites al model OSI. Aquesta família era la família de protocols TCP/IP, que va començar a desenvolupar-se a principis dels anys 70. Actualment és la més estesa, sobretot degut a que és la que s'utilitza a Internet.

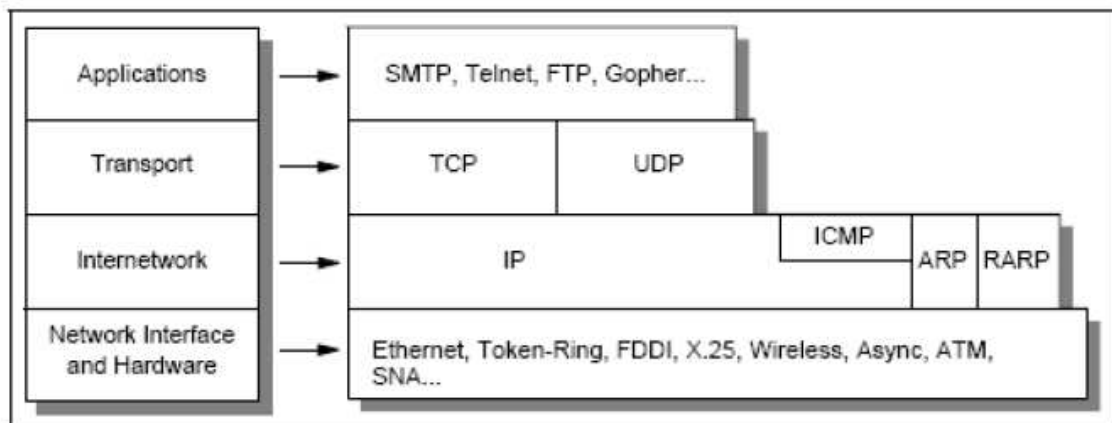


Fig. 5.20. Pila TCP/IP i protocols que utilitza

A diferència del model OSI, la pila TCP/IP només té quatre capes:

- Network Interface and Hardware Layer (Nivell d'enllaç): Aquesta capa, també anomenada Link-Layer, especifica com són transportades les dades a través dels

mitjans físics (cable, aire, etc.). Un exemple de protocol d'enllaç pot ser Ethernet, un dels més utilitzats a xarxes d'àrea local.

- **Internet Layer (Nivell de xarxa):** Aquesta és una de les capes més importants de la pila, ja que és la que permet enviar dades entre un origen i un destí. També coneguda com Internet Layer, té com a protocol principal IP.
- **Transport Layer (Nivell de transport):** Els protocols d'aquesta capa estan orientats a donar fiabilitat a la capa superior i assegurar que les dades arriben correctament. El protocol més conegut és el TCP, que permet realitzar una transmissió de dades totalment fiable.
- **Application Layer (Nivell d'aplicació):** Com en el cas del model OSI, aquest nivell conté els protocols necessaris per a la comunicació amb programes que utilitzin la pila per a enviar o rebre dades. Aquests protocols poden ser FTP, HTTP o Telnet, entre altres. Són els encarregats de passar les dades que arriben de la capa de transport a un format conegut per l'aplicació que espera les dades. Un clar exemple seria un navegador de Internet (aplicació), que espera paquets que viatgin a sobre del protocol HTTP (protocol de la capa d'aplicació).

Com ja hem dit, la pila TCP/IP realitza les mateixes operacions que el model OSI, però TCP/IP només té quatre capes i OSI en té set. Això vol dir que hi ha capes de TCP/IP que realitzen les funcions de dues o més del model OSI.

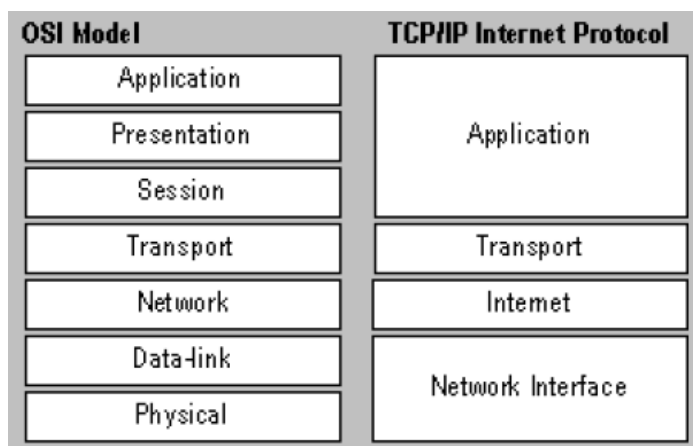


Fig. 5.21. Comparació entre el model OSI i la pila TCP/IP

A la figura 5.21 podem observar la distribució de la pila TCP/IP respecte el model OSI. Veiem que la capa d'aplicació TCP/IP conté les capes d'aplicació, de presentació i de sessió d'OSI. Això és degut a que els protocols que implementa aquesta capa de TCP/IP estan dissenyats per a poder realitzar totes aquestes funcions, sense tenir que utilitzar un protocol a cada capa. Les capes de transport i xarxa són equivalents en els dos models, i el nivell de interfície de xarxa de la pila TCP/IP inclou el nivell d'enllaç i el nivell físic de OSI.

#### **5.7.4. Pila TCP/IP de Microchip**

La casa Microchip ofereix una pila TCP/IP optimitzada per a diferents famílies de microcontroladors, com per exemple els PIC18, PIC24 i la família de microcontroladors dsPIC. Aquesta pila és un conjunt de programes que proveeixen els serveis TCP/IP a l'aplicació principal del microcontrolador. La pila segueix un disseny modular i està escrita en llenguatge de programació C. En conjunció amb el controlador d'Ethernet, permet reduir els temps i costos de desenvolupament d'aplicacions que requereixen treballar a través d'una xarxa d'àrea local o Internet.

Per tal d'implementar aquesta pila TCP/IP de Microchip en el microcontrolador, l'usuari no ha de tenir coneixements molt específics de la pila TCP/IP. Amb uns coneixements bàsics i configurant una sèrie de paràmetres, aquesta pila ja pot ser totalment operativa i preparada per instal·lar en el microcontrolador.

La pila TCP/IP de Microchip està basada en el model de referència TCP/IP, dividida en múltiples capes, on l'objectiu de cada capa és subministrar serveis a les capes superiors. D'aquesta manera, cada capa ha d'ocupar-se exclusivament del seu nivell immediatament inferior, a qui sol·licita serveis, i del nivell immediatament superior, a qui proporciona els resultats.

Per especificacions, moltes de les capes de la pila TCP/IP són 'inteligents', en el sentit de que no només actuen quan es sol·licita un servei, si no que també quan es produeixen esdeveniments com el temps d'espera o l'arribada d'un nou paquet.

Un sistema amb molta memòria de dades i de programa pot incorporar i gestionar amb facilitat una pila de protocols com la TCP/IP. Un sistema operatiu multitasca pot facilitar encara més aquesta tasca i per tant, pot fer una implementació modular. Aquesta gestió es complica quan un sistema utilitza només un microcontrolador de 8 bits, amb limitacions en quant a la capacitat de memòria de dades i de programa. A més, sense accés a un sistema operatiu multitasca, s'ha de gestionar la pila de manera independent de la aplicació principal.

### **Capes de la pila**

Com en el Model de Referència TCP/IP, la pila TCP/IP de Microchip està dividida en varies capes. El codi que implementa cada capa resideix en un arxiu font separat, mentre que els serveis i APIs (Application Programming Interfaces) estan definides a través d'arxius header.

A diferència del Model de Referència TCP/IP, moltes de les capes a la pila TCP/IP de Microchip accedeixen directament a una o més capes que no estan directament per sota d'aquestes. Els mòduls que implementen les diferents capes de la pila treballen de tal manera que en la capçalera de cada codi es comprova si es necessiten els serveis de l'actual capa, per tal de fer un 'bypass' si no és així. Un mòdul molt important en la pila TCP/IP de Microchip i que permet treballar d'aquesta manera és el 'StackTask', que gestiona el funcionament de la pila i de tots els seus mòduls.

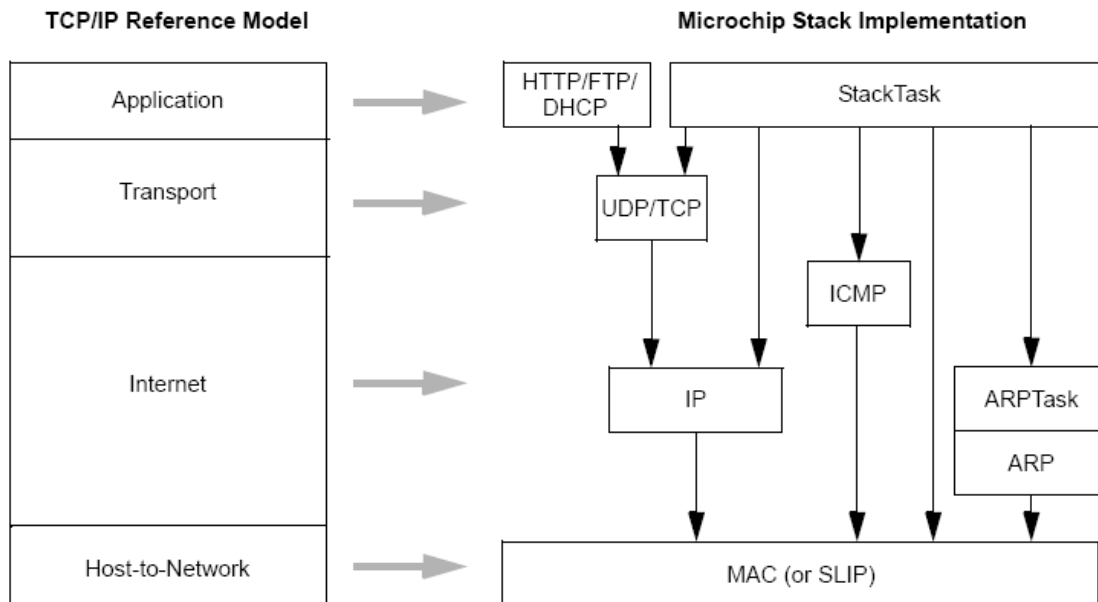
Com s'ha mencionat anteriorment, la pila TCP/IP és una pila 'intelligent'. Algunes de les seves capes han de poder dur a terme algunes operacions de manera asíncrona. Per fer-ho, i mantenir-se relativament independent de l'aplicació principal que utilitza els serveis d'aquesta pila, la pila TCP/IP de Microchip utilitza la tècnica anomenada de multitasca cooperativa, on cada tasca realitza el seu treball i retorna el control de tal manera que la següent tasca pot realitzar el seu treball.

Nomalment la multitasca cooperativa és implementada per el sistema operatiu, o per la pròpia aplicació principal. La pila TCP/IP de Microchip està dissenyada per ser independent de qualsevol sistema operatiu i implementa el seu propi sistema de multitasca



cooperativa. D'aquesta manera, pot ser utilitzada en qualsevol sistema, ja sigui mitjançant un sistema operatiu multitasca o no. No obstant, una aplicació que utilitza la pila TCP/IP de Microchip també ha d'utilitzar un mètode de multitasca cooperativa pròpia. Això es fa dividint el seu treball en múltiples capes, o organitzant el seu treball principal en una màquina d'estats finits (FSM) i dividint els treballs grans en varis treballs més petits.

La pila TCP/IP de Microchip no implementa tots el mòduls que normalment estan presents en el Model de Referència TCP/IP, però en cas de necessitar d'algun mòdul per alguna aplicació en concret es pot afegir o modificar algun existent, si és necessari.



**Fig. 5.22. Comparació de la pila TCP/IP de Microchip amb el Model de Referència TCP/IP**  
(Font: Data Sheet PIC18F97J60)

La multitasca cooperativa permet que l'aplicació principal realitzi les seves pròpies tasques sense tenir que gestionar també la pila TCP/IP. Per aconseguir aquesta funcionalitat, totes les aplicacions que utilitzin la pila TCP/IP de Microchip també han de ser escrites de manera de multitasca cooperativa. A més del disseny de la multitasca cooperativa, també s'han de conèixer alguns detalls de configuració bàsics per al correcte funcionament de la pila TCP/IP.

Per facilitar el procés de configuració, la pila utilitza la directiva ‘define’ del compilador C18. Per habilitar, deshabilitar o establir un paràmetre particular, es poden canviar una o més d’aquestes definicions. La majoria estan definides en el arxiu header “StackTsk.h”. Alguns d’aquests ‘defines’ que estan definits en altres arxius són mostrats amb els seus noms d’arxius corresponents. Una vegada s’ha modificat l’arxiu, s’ha de tornar a compilar el projecte complet per incloure els canvis. Els diferents ‘defines’ es mostren en la següent llista:

Define	Values	Used By	Purpose
CLOCK_FREQ (compiler.h)	Oscillator Frequency (Hz)	Tick.c	Define system oscillator frequency to determine tick counter value
TICKS_PER_SECONDS	10-255	Tick.c	To calculate a second
TICK_PRESCALE_VALUE	2, 4, 8, 16, 32, 64, 128, 256	Tick.c	To determine tick counter value
MPFS_USE_PGRM	N/A	MP File System (MPFS.c)	Uncomment this if program memory will be used for MPFS storage
MPFS_USE_EEPROM	N/A	MPFS.c	Uncomment this if external serial EEPROM will be used for MPFS storage
MPFS_RESERVE_BLOCK	0-255	MPFS.c	Number of bytes to reserve before MPFS storage starts
EEPROM_CONTROL	External Data EEPROM Control Code	MPFS.c	To address external data EEPROM
STACK_USE_ICMP	N/A	StackTsk.c	Comment this if ICMP is not required
STACK_USE_SLIP	N/A	SLIP.c	Comment this if SLIP is not required
STACK_USE_IP_GLEANING	N/A	StackTsk.c	Comment this if IP Gleaning is not required
STACK_USE_DHCP	N/A	DHCP.c, StackTsk.c	Comment this if DHCP is not required
STACK_USE_FTP_SERVER	N/A	FTP.c	Comment this if FTP Server is not required
STACK_USE_TCP	N/A	TCP.c, StackTsk.c	Comment this if TCP module is not required. This module will be auto- matically enabled if there is at least one high-level module requiring TCP.
STACK_USE_UDP	N/A	UDP.c, StackTsk.c	Comment this if UDP module is not required. This module will be auto- matically enabled if there is at least one high-level module requiring UDP.
STACK_CLIENT_MODE	N/A	ARP.c, TCP.c	Client related code will be enabled
TCP_NO_WAIT_FOR_ACK	N/A	TCP.c	TCP will wait for ACK before transmitting next packet
MY_DEFAULT_IP_ADDR_BYTE? MY_DEFAULT_MASK_BYTE? MY_DEFAULT_GATE_BYTE? MY_DEFAULT_MAC_BYTE?	0-255	User Application	Define default IP, MAC, gateway and subnet mask values. Default values are: 10.10.5.15 for IP address 00:04:163:00:00:00 for MAC 10.10.5.15 for gateway 255.255.255.0 for subnet mask

**Taula 5.4. Definicions de la configuració de la pila TCP/IP**

(Font: Data Sheet PIC18F97J60)

Define	Values	Used By	Purpose
MY_IP_BYTE? MY_MASK_BYTE? MY_GATE_BYTE? MY_MAC_BYTE?	0-255	MAC.c, ARP.c, DHCP.c, User Application	Actual IP, MAC, gateway and subnet mask values as saved/defined by application. If DHCP is enabled, these values reflect current DHCP server assigned configuration.
MAX_SOCKETS	1-253	TCP.c	To define the total number of sockets supported (limited by available RAM). Compile-time check is done to make sure that enough sockets are available for selected TCP applications.
MAX_UDP_SOCKETS	1-254	UDP.c	To define total number of sockets supported (limited by available RAM). Compile-time check is done to make sure that enough sockets are available for selected UDP applications.
MAC_TX_BUFFER_SIZE	201-1500	TCP.c, MAC.c	To define individual transmit buffer size
MAC_TX_BUFFER_COUNT	1-255	MAC.c	To define total number of transmit buffers. This number is limited by available MAC buffer size.
MAX_HTTP_CONNECTIONS	1-255	HTTP.c	To define maximum number of HTTP connections allowed at any time
MPFS_WRITE_PAGE_SIZE (MPFS.h)	1-255	MPFS.c	To define writable page size for current MPFS storage media
FTP_USER_NAME_LEN (FTP.h)	1-31	FTP.c	To define maximum length of FTP user name string
MAX_HTTP_ARGS (HTTP.c)	1-31	HTTP.c	To define maximum number of HTML form fields including HTML form name
MAX_HTML_CMD_LEN (HTTP.c)	1-128	HTTP.c	To define maximum length of HTML form URL string

**Taula 5.4. Definicions de la configuració de la pila TCP/IP (continuació)**

**(Font: Data Sheet PIC18F97J60)**

A la següent taula es mostren tots els mòduls necessaris per implementar la pila TCP/IP de Microchip, així com els arxius requerits per cada un dels mòduls.

Module	Files Required	Purpose
MAC	MAC.c Delay.c	Media Access Layer
SLIP	SLIP.c	Media Access Layer for SLIP
ARP	ARP.c ARPTsk.c MAC.c or SLIP.c Helpers.c	Address Resolution Protocol
IP	IP.c MAC.c or SLIP.c Helpers.c	Internet Protocol
ICMP	ICMP.c StackTsk.c IP.c MAC.c or SLIP.c Helpers.c	Internet Control Message Protocol
TCP	StackTsk.c TCP.c IP.c MAC.c or SLIP.c Helpers.c Tick.c	Transmission Control Protocol
UDP	StackTsk.c UDP.c IP.c MAC.c or SLIP.c Helpers.c	User Datagram Protocol
Stack Manager	StackTsk.c TCP.c IP.c ICMP.c ARPTsk.c ARP.c MAC.c or SLIP.c Tick.c Helpers.c	Stack Manager ("StackTask"), which coordinates the other Microchip TCP/IP Stack modules
HTTP Server	HTTP.c TCP.c IP.c MAC.c or SLIP.c Helpers.c Tick.c MPFS.c XEEPROM.c <sup>(1)</sup>	HyperText Transfer Protocol Server
DHCP Client	DHCP.c UDP.c IP.c MAC.c Helpers.c Tick.c	Dynamic Host Configuration Protocol
IP Gleaning	StackTsk.c ARP.c ARPTsk.c ICMP.c MAC.c or SLIP.c	To configure node IP address only.
FTP Server	FTP.c TCP.c IP.c MAC.c or SLIP.c	File Transfer Protocol Server.

**Note 1:** Required only if the external serial EEPROM for MPFS Storage option (MPFS\_USE\_EEPROM definition) is enabled. If selected, the corresponding MPFS image file must be included. (Refer to "MPFS Image Builder" (page 84) for additional information.)

**Taula 5.5. Mòduls de la pila TCP/IP**

## 6. MUNTATGE DEL PROTOTIP

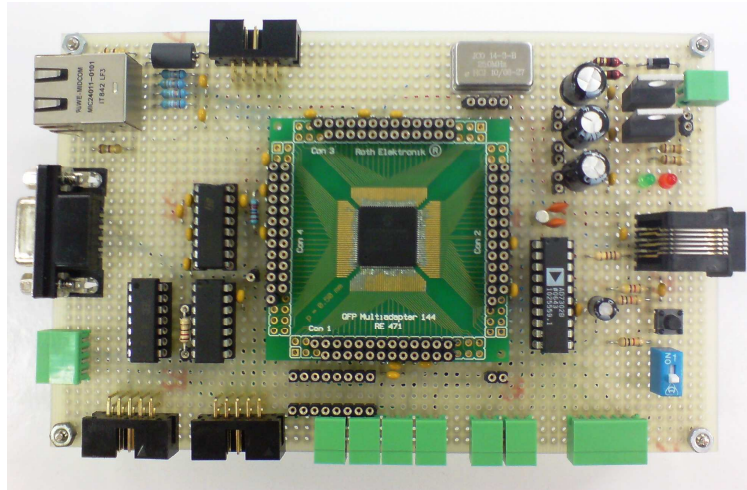
El muntatge del prototip, incloent tots els components que s'han descrit al capítol 5 per dotar al mòdul microcontrolador de totes les funcionalitats previstes, s'ha realitzat sobre un placa perforada de baquelita utilitzant cable wire-rap.

S'ha escollit aquest tipus de muntatge per la facilitat amb la que es poden realitzar modificacions, que a l'hora de muntar un prototip resulta molt important.

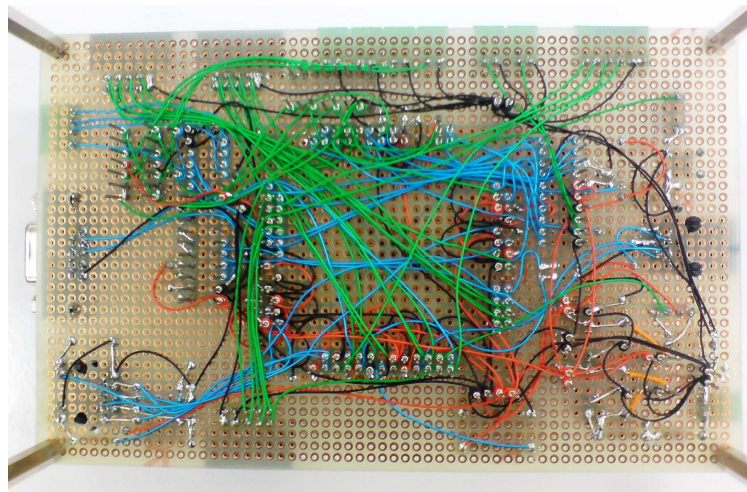
S'han combinat les connexions entre components i circuits integrats amb cable wire-rap i amb soldadura manual amb estany. En el cas de la soldadura del microcontrolador PIC18F97J60, s'ha tingut que utilitzar un forn de refusió. Aquest últim procés de muntatge s'explica més endavant amb detall.

El muntatge s'ha realitzat de manera modular, dividint-lo en diferents parts, ja que degut a la gran quantitat de connexions a realitzar existeix un risc important de falles, i efectuant el muntatge d'aquesta manera la seva detecció resulta molt senzilla.

El muntatge s'ha dividit en les següents parts: alimentació, entrades i sortides digitals, entrades i sortides analògiques, sortides PWM, connectivitat RS232 i RS485, i finalment la connectivitat Ethernet. Al finalitzar cada una de les diferents parts del muntatge s'han efectuat proves de funcionament hardware, per verificar el muntatge i el correcte funcionament dels diferents elements que componen el mòdul microcontrolador universal.



**Fig. 6.1. Vista de la cara superior del prototip**  
(Font: Elaboració pròpia)



**Fig. 6.2. Vista de la cara inferior del prototip**  
(Font: Elaboració pròpia)

## **6.1. Soldadura del microcontrolador PIC18F97J60**

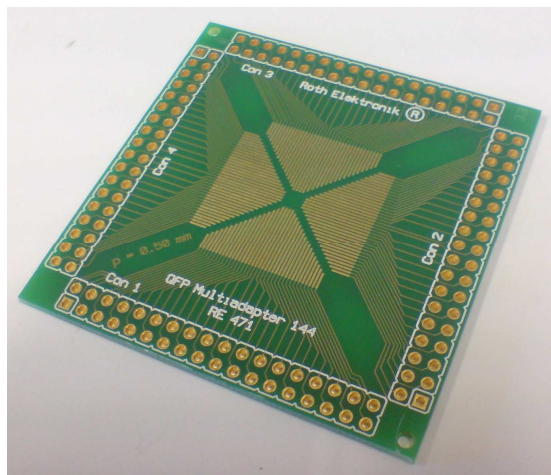
### **6.1.1. Tècnica de muntatge superficial (SMT)**

La tècnica de muntatge superficial (SMT) utilitza components electrònics on les seves connexions es recolzen sobre la placa de circuit imprès, sense orifici. Aquests tipus de components són els denominats SMD (Surface Mounted Device). Un avantatge que presenta aquest tipus de muntatge, a part de la disminució de tamany dels components, es

que es pot equipar la placa de circuits impresos de manera completament automàtica, fet que fa que actualment gairebé tots els muntatges electrònics es realitzin amb aquest tipus de muntatge. Les màquines automàtiques de muntar components aconseguixen un rendiment molt alt, augmentant la producció de circuits electrònics.

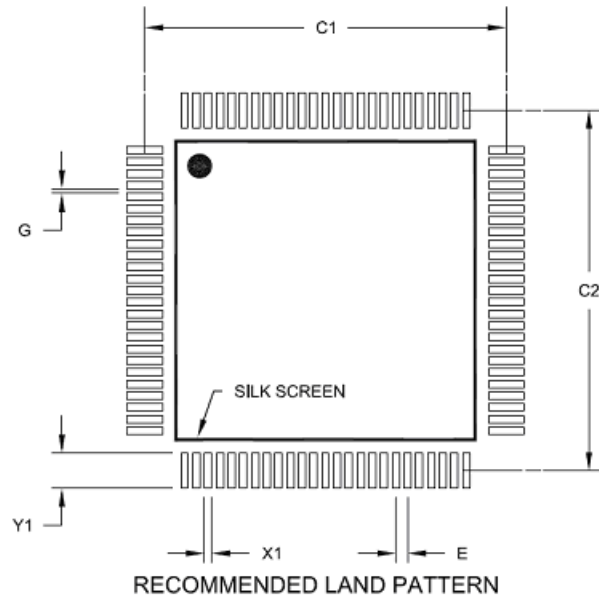
En el nostre cas, el microcontrolador PIC18F97J60 utilitzat en la construcció del prototipus del mòdul microcontrolador universal té un acabat d'encapsulat TQFP de 100 pins, de muntatge superficial. No existeix cap altre tipus d'encapsulat per aquest microcontrolador. Aquest fet suposa un problema afegit a l'hora de la realització del prototip, ja que la soldadura s'ha de realitzar amb un forn de soldadura per refusió, sense tenir experiència en aquest tipus de muntatge.

Per realitzar aquest muntatge també cal un circuit imprès que ens adapti les connexions del microcontrolador a unes altres on les distàncies entre connexions siguin més grans per facilitar el muntatge del prototip. La solució adoptada per adaptar aquestes connexions és utilitzar un adaptador universal per diferents tipus d'encapsulats i nombre de pins. Aquest adaptador és el RE471 de la casa Roth Elektronik, que és compatible amb els encapsulats LQFP, PQFP i TQFP amb un nombre de pins des de 24 fins a 144.



**Fig. 6.3. Adaptador SDM RE471**  
(Font: Elaboració pròpia)

Aquest adaptador disposa de dues cares, amb diferents distàncies entre els centres dels 'pads', 0.5 i 0.65 mm. En el cas del microcontrolador PIC18F97J60, la distància entre el centre dels pins és de 0.5 mm.



Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Contact Pitch	E	0.50 BSC		
Contact Pad Spacing	C1		15.40	
Contact Pad Spacing	C2		15.40	
Contact Pad Width (X100)	X1			0.30
Contact Pad Length (X100)	Y1			1.50
Distance Between Pads	G	0.20		

## Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2110A

**Fig. 6.4. Mides del microcontrolador PIC18F97J60**

(Font: Data Sheet PIC18F97J60)

### 6.1.2. Procés de soldadura per refusió

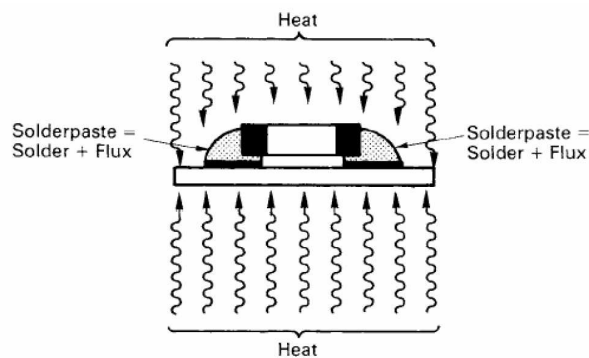
El procés de soldadura amb el forn de refusió o reflux és un mètode industrial clàssic a l'hora de soldar components SMD. Amb aquest mètode es poden soldar una gran varietat de components de mides reduïdes, que seria molt complicat realitzar-ho soldant manualment.

En la soldadura per refusió, s'escalfa tota la tarja de circuit imprès i tots els components alhora.



Per realitzar la soldadura per refusió s'utilitza pasta de soldadura en comptes de l'estany de soldadura estàndard. Aquesta pasta de soldadura és una barreja de partícules molt fines d'estany barrejades amb un líquid.

La pasta de soldadura es col·loca primer a la placa de circuit imprès i posteriorment es col·loquen tots els components. Després s'ha de col·locar la placa amb els components en un forn de soldadura per refusió, on s'escalfa el conjunt fins el punt en que l'estany es fon amb les capes de metall de l'entorn.



**Fig. 6.5. Soldadura per refusió**

El procés de soldadura per refusió està format per varies etapes, on es segueix una trajectòria de temperatura concreta per obtenir un resultat òptim.

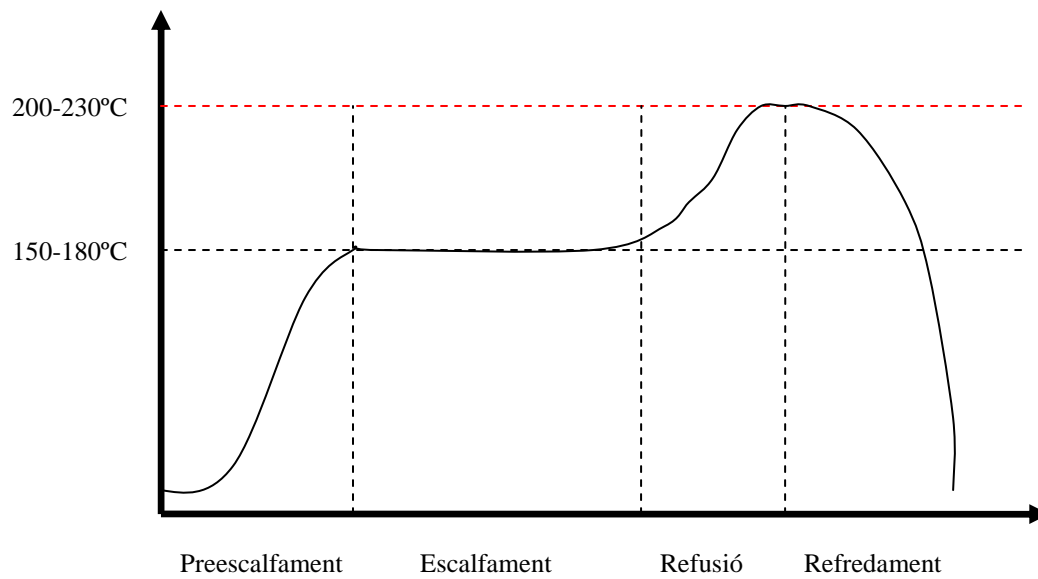
Les diferents etapes són les següents:

- **Preescalfament.** En aquesta etapa es produeix un augment de temperatura gradual per escalfar la placa i els components des de la temperatura ambient fins als 120 -150°C, amb un increment de temperatura de 1°C/s fins a 4°C/s. En aquesta etapa s'evapora la humitat, s'eliminen tensions internes i gasos residuals de la PCB. Si es selecciona un període comprès entre 1 i 5 minuts per aquesta etapa, el preescalfament també serveix com a transició gradual a la següent fase de temperatura. El temps òptim depèn del tamany de la placa i del nombre de components.
- **Fase d'escalfament (o soaking).** L'escalfament activa la liquació del flux contingut a l'aliatge de l'estany, el que permet eliminar capes d'òxid en els components i la preparació

per la soldadura, ja que ens situem a una temperatura que està a prop del punt de fusió de les partícules d'estany. La temperatura d'aquesta fase depèn del tipus d'aliatge de l'estany. En aquest pas s'ha d'incrementar la temperatura a un valor entre 150 i 180°C i mantenir-la durant 60 segons com a mínim.

- Fase de soldadura o refusió. Durant aquesta fase les partícules d'estany es fonen amb les peces de metall adjacents, produint-se la soldadura. La temperatura en aquesta etapa s'ha de situar entre els 200 i 230°C, depenent de la pasta de soldar. El temps d'aquesta etapa no ha de superar els 60 segons per tal de no modificar l'estructura molecular de la pasta de soldar, normalment aquest temps ha d'estar entre els 10 i 30 segons. El procés està marcat per canvis químics i físics substancials de l'estany, els quals determinen en gran mesura l'èxit del procés complet de soldadura. Cal remarcar que aquesta és l'etapa més crítica, ja que s'arriba al valor més alt de temperatura de tot el procés, per el que els components es poden danyar fàcilment si la temperatura i/o el temps han estat seleccionat de manera incorrecta.

- Fase de refredament. Després d'arribar a la temperatura màxima on es produeix la soldadura, s'ha d'entrar en aquesta etapa on es deixa refredar el conjunt de tarja i components. Aquest refredament no s'ha de produir d'una manera molt ràpida, per tal d'evitar grans diferències de temperatura entre els components i la targeta de circuit imprès, i evitar ruptures o deformacions. S'ha d'anar amb precaució a l'hora de retirar el circuit amb els components, vigilant que la temperatura del conjunt hagi disminuït el suficient per tal de no moure cap component quan la pasta encara no s'ha solidificat i evitar el risc de possibles cremades.

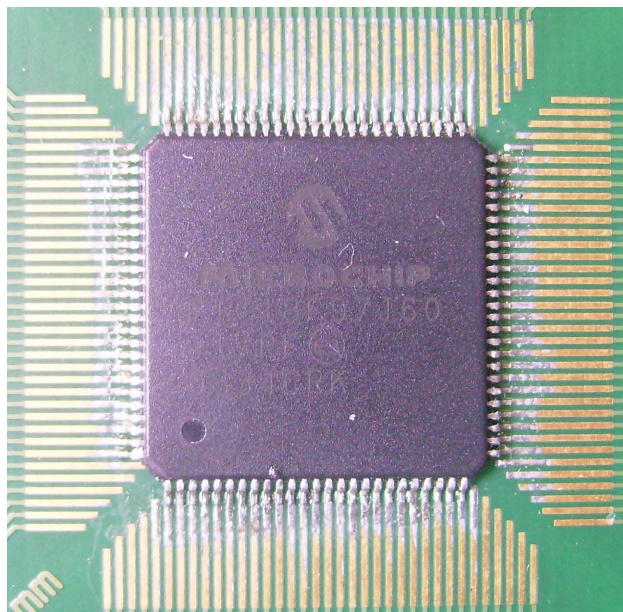


**Fig. 6.6. Corba de temperatura per al procés de refusió**  
(Font: Elaboració pròpia)

Per soldar el microcontrador PIC18F97J60 a l'adaptador s'ha intentat no arribar als valors màxims recomanats per al procés de soldadura per refusió, per tal d'assegurar que no es danyi l'estructura interna del microcontrolador. Les temperatures que s'han utilitzat són 160°C per la fase d'escalfament i 210°C per la fase de refusió, amb uns temps de 90 i 20 segons respectivament.



**Fig. 6.7. Forn de soldadura per refusió**  
(Font: Elaboració pròpia)



**Fig. 6.8. PIC18F97J60 soldat a l'adaptador RE471**

**(Font: Elaboració pròpia)**

## 7. APLICACIONS PRÀCTIQUES

### 7.1. Eines de desenvolupament

Per tal de dur a terme totes les proves que s'han realitzat per verificar el funcionament del mòdul microcontrolador universal, així com la realització de les aplicacions pràctiques, s'han utilitzat una sèrie d'eines de desenvolupament (hardware i software) que es descriuen a continuació:

#### **MPLAB IDE v8.20**

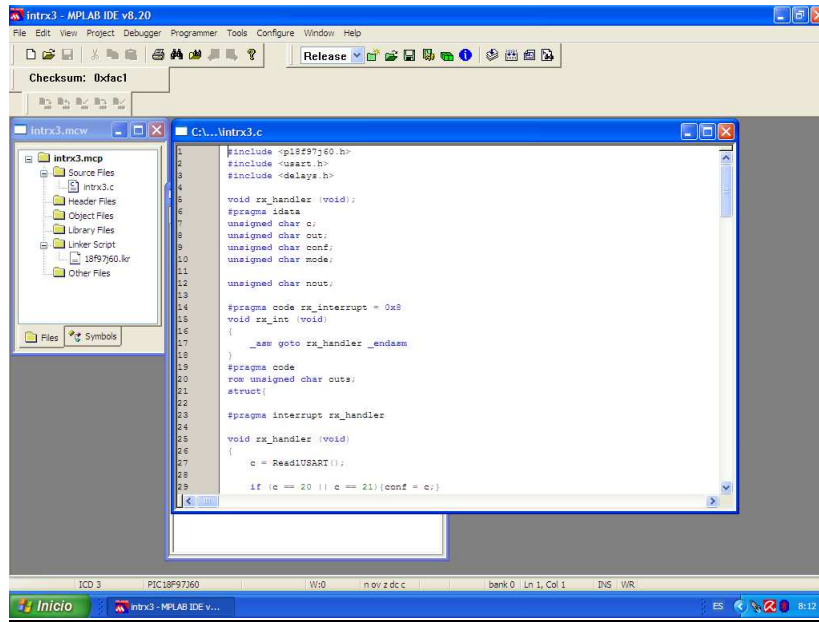
MPLAB és un editor IDE (Entorn de Desenvolupament Integrat) gratuït, destinat a productes de la casa Microchip, que s'executa sobre sistemes operatius Windows. És una eina que ens permet desenvolupar aplicacions per microcontroladors i controladors de senyal digital.

S'anomena Entorn de Desenvolupament Integrat, o IDE, ja proporciona un únic entorn integrat per desenvolupar codi per als microcontroladors. MLAB IDE ens permet editar, compilar, simular, depurar i programar (aquestes dues últimes funcions amb el hardware adequat).

#### **MPLAB C18 v3.30**

MPLAB C18 és un compilador de llenguatge de programació C per microcontroladors de la família PIC18 de Microchip. Aquest software s'ha d'instalar juntament amb el MPLAB IDE per poder generar codi per als microcontroladors en aquest llenguatge.

La versió utilitzada durant el desenvolupament del projecte és una versió de demostració totalment funcional durant 60 dies, ja que aquest software no és gratuït com en el cas del MPLAB IDE.



**Fig. 7.1. Pantalla principal del MPLAB IDE amb el compilador C**  
(Font: Elaboració pròpia)

### **MPLAB ICD 3**

El MPLAB ICD 3 es un depurador ‘en circuit’ que es controla per un PC amb sistema operatiu Windows i el software de Microchip MPLAD IDE (versió 8.15 o superior). El MPLAB ICD3 és compatible amb la majoria de microcontroladors i controladors de senyal digital amb memòria flash de la casa Microchip. El depurador suporta en temps real la visió de variables i registres, i es poden obtenir “breakpoints” en memòria de programa. El MPLAB ICD3 també s'utilitza per programar els microcontroladors. Aquesta eina es connecta al PC mitjançant el port USB, i no necessita cap alimentació externa. Per connectar el ICD3 amb el microcontrolador s'ha d'utilitzar un connector RJ11.



**Fig. 7.2. MPLAB ICD3**  
(Font: Microchip)

## 7.2. Servidor web

Per mostrar la capacitat del mòdul microcontrolador universal de treballar com a servidor web, amb diferents possibilitats com la supervisió, monitorització i control de processos, s'ha creat una pagina web que ens mostra l'estat del mòdul i ens permet realitzar algunes accions. A continuació s'expliquen alguns aspectes teòrics, que són importants a l'hora d'entendre el funcionament del microcontrolador com a servidor web, així com facilitar la tasca de realització de la pàgina web.

### 7.2.1. Common Gateway Interface (CGI)

La interfície d'entrada comú o CGI (Common Gateway Interface) és un dels primers mètodes de programació per generar contingut dinàmic a les pàgines web.

Quan el World Wide Web va iniciar el seu funcionament tal com el coneixem avui en dia, només es podia accedir a textos, imatges i enllaços. La introducció de plugins en els primers navegadors va permetre una major interacció entre l'usuari i el servidor, però estava limitat per la velocitat i la necessitat de tenir que descarregar i instal·lar cada plugin que es necessites, per el que es van començar a desenvolupar per aplicacions més específiques.

El CGI és un mètode per a la transmissió d'informació cap a un compilador instal·lat en el servidor. La seva funció principal es la d'afegir una major interacció als documents web que per mitjà del HTML es presenten de manera estàtica.

Aquesta tecnologia té l'avantatge d'executar-se en el servidor quan l'usuari ho sol·licita, el que fa que sigui dependent del servidor i no de l'ordinador des del qual es vol accedir.

Els programes que treballen amb el CGI poden estar compilats en diferents llenguatges de programació. Entre els llenguatges més habituals podem trobar Perl, C, C++, Java o Visual Basic.

### **7.2.2. Servidor HTTP de Microchip**

El servidor HTTP està implementat en forma de tasca cooperativa que coexistirà amb la pila TCP/IP de Microchip i l'aplicació principal.

El servidor HTTP no implementa totes les funcionalitats pròpies HTTP estàndards, si no que és un servidor mínim enfocat a sistemes embedded, amb la possibilitat d'afegir noves funcionalitats quan sigui necessari.

El servidor HTTP incorpora les següents funcionalitats:

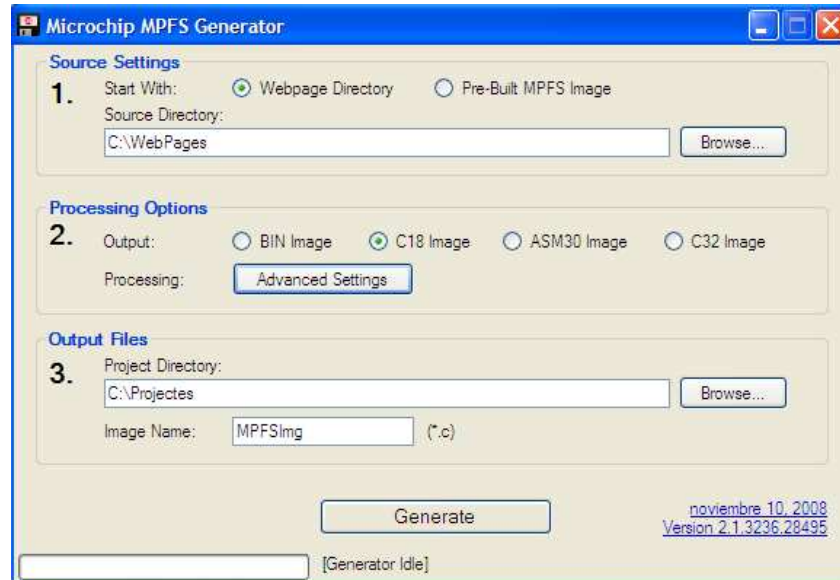
- Suport per a múltiples connexions HTTP.
- Utilització d'un sistema d'arxiu simple (MPFS, Microchip File System) per l'emmagatzemament del codi HTML de la pàgina web, ja sigui en memòria interna o externa.
- Incorpora un programa basat en PC per la generació d'imatges MPFS des d'un directori determinat.
- Suport per al mètode "GET" de HTTP.
- Suport de CGI (Common Gateway Interface) modificada per invocar funcions predefinides des d'un buscador remot.
- Suport per la generació de pàgines web amb contingut dinàmic.

Per a que la pàgina web creada en llenguatge HTML sigui operativa en el microcontrolador, s'ha de convertir a un format que el microcontrolador interpreti correctament i pugui ser emmagatzemada a memòria, tant interna com externa. Per a la pila TCP/IP de Microchip i el seu servidor HTTP, el format al que s'ha de convertir la pàgina web és el MPFS (Microchip File System).

Per realitzar aquesta conversió, Microchip facilita un aplicatiu, Microchip MPFS Generator, que realitza aquesta funció. Aquest programa incorpora varies opcions a l'hora de generar l'arxiu, però la principal fa referència a la memòria on es guardarà la pagina web. Si volem guardar-la a la memòria interna del microcontrolador, l'arxiu a generar ha



de ser en C, i si volem guardar-la en una memòria externa l'arxiu a generar ha de ser binari (extensió \*.bin).



**Fig. 7.3. Pantalla principal del Microchip MPFS Generator**  
(Font: Elaboració pròpia)

El servidor HTTP utilitza l'arxiu "index.htm" com la seva pàgina web per defecte. Si un client remot (buscador) accedeix al servidor HTTP només per la seva adreça IP o nom de domini, la pàgina web que es servirà per defecte serà "index.htm".

Això requereix que totes les aplicacions incloguin un arxiu anomenat "index.htm" com a part de la seva imatge MPFS.

Si és necessari, el nom d'aquest arxiu pot ser canviat modificant la definició HTTP\_DEFAULT\_FILE\_STRING a l'arxiu "http.c".

Un aspecte important a tenir en compte a l'hora de la realització de la pagina web en llenguatge HTML és que cap dels noms de la pàgina web contingui alguns dels següents caràcters, per tal de realitzar correctament la conversió a la imatge MPFS: ' " < > # % [ { } ] \ ^ ~.

Si algun arxiu conté qualsevol d'aquests caràcters, la corresponent pàgina web serà inaccessible sense donar cap avís previ.

Un altre característica del servidor HTTP de Microchip és el tipus d'arxius que suporta. Per defecte, el servidor HTTP de Microchip pot treballar amb els següents tipus d'arxius: “.txt”, “.htm”, “.gif”, “.jpg”, “.cla” i “.wav”. Si es vol treballar amb algun altre tipus d'arxiu, es poden afegir modificant dins de l'arxiu de la pila TCP/IP ‘http.c’.

### **7.2.3. Generació dinàmica de pàgines HTTP**

El servidor http pot modificar dinàmicament pàgines i substituir informació en temps real, com per exemple l'estat de les entrades i sortides.

Per incorporar aquesta informació en temps real, el corresponent arxiu CGI ha de contenir una cadena de text ‘%xx’, on el caràcter ‘%’ és un codi de control i ‘xx’ representa un identificador de variables de dos dígitos.

El rang de valor que pot agafar aquest indicador és 00-99. Quan el servidor HTTP troba aquest text, elimina el caràcter ‘%’ i crida a la funció *HTTPGetVar*. Si el caràcter ‘%’ s’ha de mostrar a la pàgina web, ha d’anar precedit per un altre caràcter ‘%’. Per exemple, per mostrar el valor “23%” a la pàgina web, s’hauria d’escriure “23%%”.

#### **La funció HTTPGetVar**

Aquesta funció és una crida des de HTTP. Quan el servidor HTTP troba una cadena de caràcters ‘%xx’ en una pàgina CGI que aquesta està servint, aquest crida a la funció *HTTPGetVar*.

Aquesta funció és implementada per l’aplicació principal i s’utilitza per transferir l’estat de les variables de l’aplicació al servidor web.

La sintaxis d’aquesta funció és la següent:

```
WORD HTTPGetVar (BYTE var, WORD ref, BYTE *val)
```

Els paràmeres de la funció són els següents:

- *var* (entrada) : Identificador de la variable de la qual s'ha de retornar el seu estat.
- *ref* (entrada) : Aquest valor de referència indica si aquesta ha estat la primera crida a la funció. Després de la primera crida, el valor és mantingut per l'aplicació principal. En el cas de que per actualitzar el valor de la variable es requereix més d'un byte, l'aplicació principal pot utilitzar *ref* com a punter.
- *val* (sortida) : És el byte de dades que s'envia al servidor web.

Com s'ha comentat, el valor que retorna aquesta funció no té per que ser només un byte. Aquesta funció pot tornar una variable que sigui una imatge JPEG, per exemple. En aquest cas, el valor de *ref* s'utilitza com a punter i ha de ser gestionat a l'aplicació principal. A la primera crida a la funció, *ref* ha de ser `HTTP_START_OF_VAR`, que indica l'inici de la transferència dels diferents bytes. A les següents crides s'utilitza *ref* com a punter, i per finalitzar *ref* ha de ser `HTTP_END_OF_VAR`. Tot això ha de ser gestionat per l'aplicació principal, i per tant s'han d'escriure les condicions necessàries per realitzar-ho correctament.

### **La funció `HTTPExecCmd`**

La funció `HTTPGetVar` actualitza els valors del servidor web a partir de les variables del microcontrolador. Per realitzar la comunicació en l'altre sentit i modificar l'estat de les variables de microcontrolador a partir de modificacions que es realitzen a la pàgina web, el servidor HTTP de Microchip utilitza una versió modificada del CGI.

El client HTTP pot invocar una funció i rebre resultats en forma de pàgina web. Per fer-ho, s'utilitza el mètode GET del llenguatge HTML. Quan un cercador remot executa un mètode GET amb més d'un paràmetre, el servidor HTTP ho analitza i crida a l'aplicació principal amb el codi del mètode actual i els seus paràmetres. Per utilitzar correctament aquest mètode, l'aplicació principal ha d'implementar una crida a una funció amb un codi adequat. Aquesta funció que es troba a l'aplicació principal és la funció `HTTPExecCmd`.

Aquesta funció és una crida des de HTTP. Quan el servidor HTTP rep un mètode GET amb més d'un paràmetre, es crida a la funció. Aquesta funció, implementada per l'aplicació principal, ha de tractar el codi rebut i prendre les accions apropiades, com per exemple la actualització de l'estat de les sortides.

La sintaxis d'aquesta funció és la següent:

```
void HTTPExecCmd (BYTE **argv, BYTE argc)
```

Els paràmetres de la funció són els següents:

- *argv* (entrada) : Llista d'arguments. El primer argument enviar (*argv[0]*) representa l'acció a realitzar, mentre que la resta d'arguments (*argv[1..n]*) són paràmetres de comando.
- *argc* (entrada) : Aquí s'ha d'especificar el nombre total de paràmetres, incloent el de l'acció a realitzar.

#### **7.2.4. Exemple de servidor web embedded**

Una vegada definit com treballa el servidor HTTP de Microchip i la forma de generar contingut dinàmic a les pàgines web, es mostra com s'han utilitzat les funcions de generació dinàmica de pàgines HTTP anteriorment descrites a l'aplicació i el servidor web embedded creat com exemple per demostrar la capacitat que té el mòdul microcontrolador universal de treballar com a servidor web per supervisar, monitoritzar i controlar processos.

Cal destacar que el servidor web creat és un exemple on només es mostra una part del seu potencial. En aquest servidor web creat es mostra l'estat de les entrades i sortides, tant digitals com analògiques, i es dóna la possibilitat de modificar l'estat de les sortides.

A continuació es mostra el codi de l'arxiu CGI que està vinculat a la pàgina web creada. Es poden observar els identificadors de variables que criden a la funció *HTTPGetVar*.

```

<style type="text/css">
<!--
-->
</style>
<p>ENTRADES ANAL&ograve;GIQUES:</p>
<table width="228" border="1">
  <tr>
    <td width="50"><div align="center"><b>AIN1</b></div></td>
    <td width="50"><div align="center"><b>AIN2</b></div></td>
    <td width="50"><div align="center"><b>AIN3</b></div></td>
    <td width="50"><div align="center"><b>AIN4</b></div></td>
  </tr>
  <tr>
    <td><div align="center">%02</div></td>
    <td><div align="center">%03</div></td>
    <td><div align="center">%24</div></td>
    <td><div align="center">%25</div></td>
  </tr>
</table>
<p><font face="Arial, Helvetica, sans-serif"></font>ENTRADES DIGITALS:</p>
<table width="400" border="1">
  <tr>
    <th scope="col"><b>IN1</b></th>
    <th scope="col"><b>IN2</b></th>
    <th scope="col"><b>IN3</b></th>
    <th scope="col"><b>IN4</b></th>
    <th scope="col"><b>IN5</b></th>
    <th scope="col"><b>IN6</b></th>
    <th scope="col"><b>IN7</b></th>
    <th scope="col"><b>IN8</b></th>
  </tr>
  <tr>
    <td><div align="center">%04</div></td>
    <td><div align="center">%0D</div></td>
    <td><div align="center">%0E</div></td>
    <td><div align="center">%0F</div></td>
    <td><div align="center">%20</div></td>
    <td><div align="center">%21</div></td>
    <td><div align="center">%22</div></td>
    <td><div align="center">%23</div></td>
  </tr>
</table>
<p>SORTIDES DIGITALS:</p>
<table width="400" border="1">
  <tr>
    <th scope="col"><b>OUT1</b></th>
    <th scope="col"><b>OUT2</b></th>
    <th scope="col"><b>OUT3</b></th>
    <th scope="col"><b>OUT4</b></th>
    <th scope="col"><b>OUT5</b></th>
    <th scope="col"><b>OUT6</b></th>
    <th scope="col"><b>OUT7</b></th>
    <th scope="col"><b>OUT8</b></th>
  </tr>
  <tr>
    <td><div align="center">%00</div></td>
    <td><div align="center">%01</div></td>
    <td><div align="center">%10</div></td>
    <td><div align="center">%11</div></td>
    <td><div align="center">%12</div></td>
    <td><div align="center">%13</div></td>
    <td><div align="center">%14</div></td>
    <td><div align="center">%15</div></td>
  </tr>
</table>
<p>&nbsp;</p>

```

Exemple d'arxiu CGI

Tots els identificadors que apareixen en aquest arxiu CGI es corresponen amb una variable del programa que s'executa en el microcontrolador.

A continuació es mostra una part del codi de l'arxiu html, on es crida a la funció *HTTPExecCmd* per modificar l'estat de les sortides del microcontrolador.

```
<table cellpadding="3">
<form>
<tr>
<td>
    <b>Accions:</b>
</td>
</tr>
<tr>
<td>Actuar sobre les sortides digitals:</td>
<td>
<input type="button" value="OUT1"
onClick="GetServerFile('0?0=OUT1','')"></input>
</td>
<td>
<input type="button" value="OUT2"
onClick="GetServerFile('0?1=OUT2','')"></input>
</td>
<td>
<input type="button" value="OUT3"
onClick="GetServerFile('0?2=OUT3','')"></input>
</td>
<td>
<input type="button" value="OUT4"
onClick="GetServerFile('0?3=OUT4','')"></input>
</td>
<td>
<input type="button" value="OUT5"
onClick="GetServerFile('0?4=OUT5','')"></input>
</td>
<td>
<input type="button" value="OUT6"
onClick="GetServerFile('0?5=OUT6','')"></input>
</td>
<td>
<input type="button" value="OUT7"
onClick="GetServerFile('0?6=OUT7','')"></input>
</td>
<td>
<input type="button" value="OUT8"
onClick="GetServerFile('0?7=OUT8','')"></input>
</td>
</tr>
```

**Part de l'arxiu html corresponent a la modificació de l'estat de les sortides digitals**

Finalment, es mostra una part de l'arxiu C de l'aplicació principal del microcontrolador on es gestiona la generació dinàmica de la pàgina web. Aquí estan declarades les variables que es criden des de la pàgina web, així com les diferents funcions i l'acció que es realitza per a cada cas.

```

#define CGI_CMD_DIGOUT      (0)
#define DOUT1                (0x00) // Sortides digitals
#define DOUT2                (0x01)
#define DOUT3                (0x10)
#define DOUT4                (0x11)
#define DOUT5                (0x12)
#define DOUT6                (0x13)
#define DOUT7                (0x14)
#define DOUT8                (0x15)
#define ANAIN1              (0x02) // Entrades analògiques
#define ANAIN2              (0x03)
#define ANAIN3              (0x24)
#define ANAIN4              (0x25)
#define DIN1                (0x04) // Entrades digitals
#define DIN2                (0x0D)
#define DIN3                (0x0E)
#define DIN4                (0x0F)
#define DIN5                (0x20)
#define DIN6                (0x21)
#define DIN7                (0x22)
#define DIN8                (0x23)

extern BYTE AN0String[8];
extern BYTE AN1String[8];
extern BYTE AN2String[8];
extern BYTE AN3String[8];

void HTTPExecCmd(BYTE** argv, BYTE argc)
{
    BYTE command;
    BYTE var;

    command = argv[0][0] - '0';

    switch(command)
    {
        case CGI_CMD_DIGOUT: // Acció=0
            //Identificació de paràmetres
            var = argv[1][0] - '0';

            switch(var)
            {
                case CMD_OUT1:
                    LATBbits.LATB0 ^= 1; // Canvia l'estat de la sortida 1
                    break;
                case CMD_OUT2:
                    LATBbits.LATB1 ^= 1; // Canvia l'estat de la sortida 2
                    break;

                (...)

                case CMD_OUT8:
                    LATCbits.LATC5 ^= 1; // Canvia l'estat de la sortida 8
                    break;
            }
    }
}

```

**Part de l'arxiu C de l'aplicació principal del microcontrolador**

```

WORD HTTPGetVar(BYTE var, WORD ref, BYTE* val)
{
    // Identificació de variables, es torna l'estat de la variable
    solicitada
    switch(var)
    {
        case DOUT1:
            *val = LATBbits.LATB0 ? '1':'0'; // Torna l'estat de la sortida 1
            break;
        case DOUT2:
            *val = LATBbits.LATB1 ? '1':'0'; // Torna l'estat de la sortida 2
            break;

            (...)

        case DOUT8:
            *val = LATCbits.LATC5 ? '1':'0'; // Torna l'estat de la sortida 8
            break;

        case ANAIN1:
            *val = AN0String[(BYTE)ref];
            if(AN0String[(BYTE)ref] == '\0')
                return HTTP_END_OF_VAR;
            else if(AN0String[(BYTE)++ref] == '\0' )
                return HTTP_END_OF_VAR;
            return ref; // Torna l'estat de la sortida analògica 1

            (...)

        case ANAIN4:
            *val = AN3String[(BYTE)ref];
            if(AN3String[(BYTE)ref] == '\0')
                return HTTP_END_OF_VAR;
            else if(AN3String[(BYTE)++ref] == '\0' )
                return HTTP_END_OF_VAR;
            return ref; // Torna l'estat de la sortida analògica 4

        case DIN1:
            *val = PORTJbits.RJ0 ? '1':'0'; // Torna l'estat de l'entrada 1
            break;
        case DIN2:
            *val = PORTJbits.RJ1 ? '1':'0'; // Torna l'estat de l'entrada 2
            break;

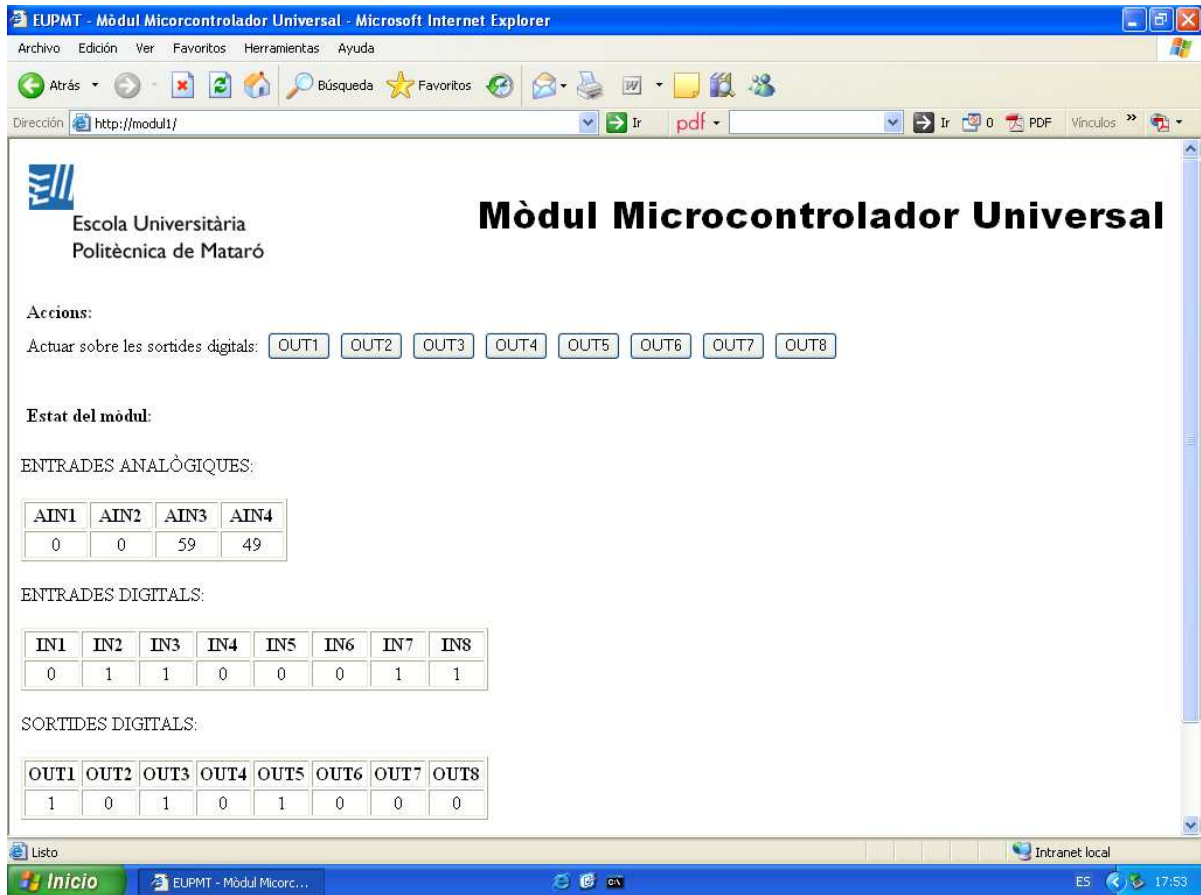
            (...)

        case DIN8:
            *val = PORTJbits.RJ7 ? '1':'0'; // Torna l'estat de l'entrada 8
            break;
    }
}

```

**Part de l'arxiu C de l'aplicació principal del microcontrolador (continuació)**





**Fig. 7.4. Pantalla del servidor web d'exemple**  
(Font: Elaboració pròpia)

### 7.3. Exemples d'aplicacions

En aquest apartat es mostra la capacitat que té el mòdul microcontrolador universal de ser configurat per treballar en diferents aplicacions.

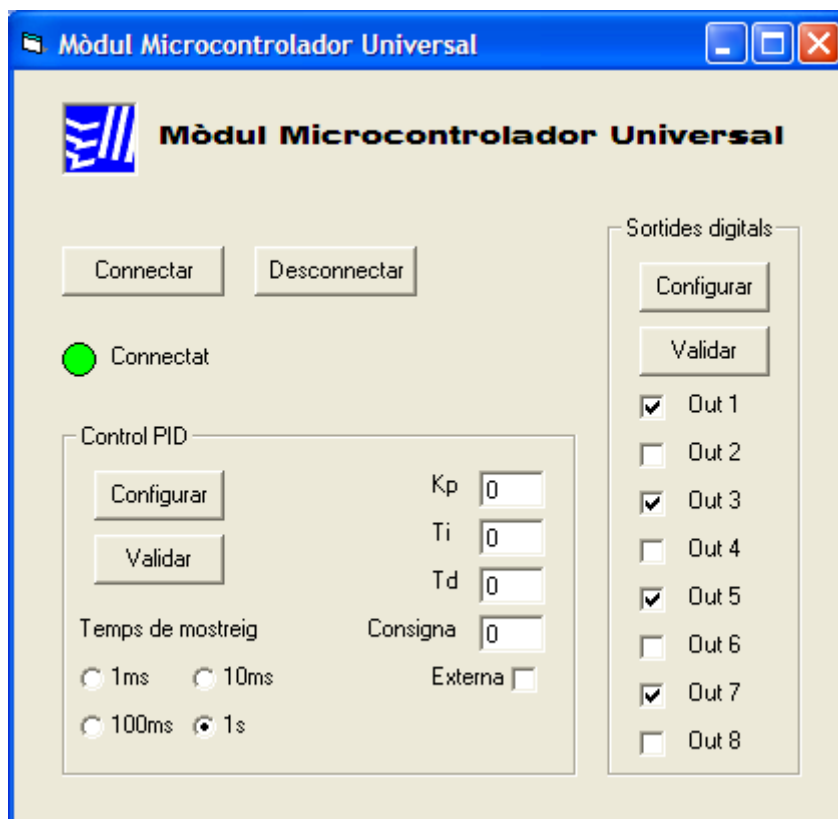
La configuració de les diferents aplicacions es realitza a partir d'un software que s'executa en un PC, connectat via RS232 amb el mòdul microcontrolador universal. Aquest software s'ha implementat amb llenguatge de programació Visual Basic.

Per poder configurar el mòdul microcontrolador universal, aquest disposa d'un switch que segons la seva posició determina l'estat d'aquest. Si el switch en posició ON, el mòdul microcontrolador universal entra en el mode de programació, on espera les configuracions corresponents, i si està en posició OFF, el mòdul entra en el mode de funcionament. Quan

s'està en el mode de funcionament, el mòdul microcontrolador universal executa la última aplicació configurada.

Per indicar l'estat del mòdul, aquest disposa d'un diode LED de color verd. Si estem en mode de funcionament, el LED està apagat.

Si estem en mode de programació, el mòdul pot estar en dos possibles estats. El primer estat possible és que des del software de configuració s'ha connectat amb el mòdul, llavors el LED s'encén. Una vegada s'ha realitzat la connexió, des del software s'ha d'escollir el tipus de configuració a realitzar: configuració de l'estat de les sortides o control PID. Quan s'ha escollit el tipus de configuració a realitzar, el LED s'apaga i el mòdul està a l'espera de rebre els paràmetres per a la configuració del mode de treball. Una vegada s'han introduït els paràmetres en el software i s'han enviat al mòdul, el LED es torna a encendre, fins que canviem l'estat del switch i entrem en mode funcionament.



**Fig. 7.5. Pantalla del software de configuració d'aplicacions**

(Font: Elaboració pròpia)

Per tal realitzar la comunicació entre el mòdul microcontrolador universal i el PC, mitjançant aquest software de configuració, s'ha dissenyat un senzill protocol.

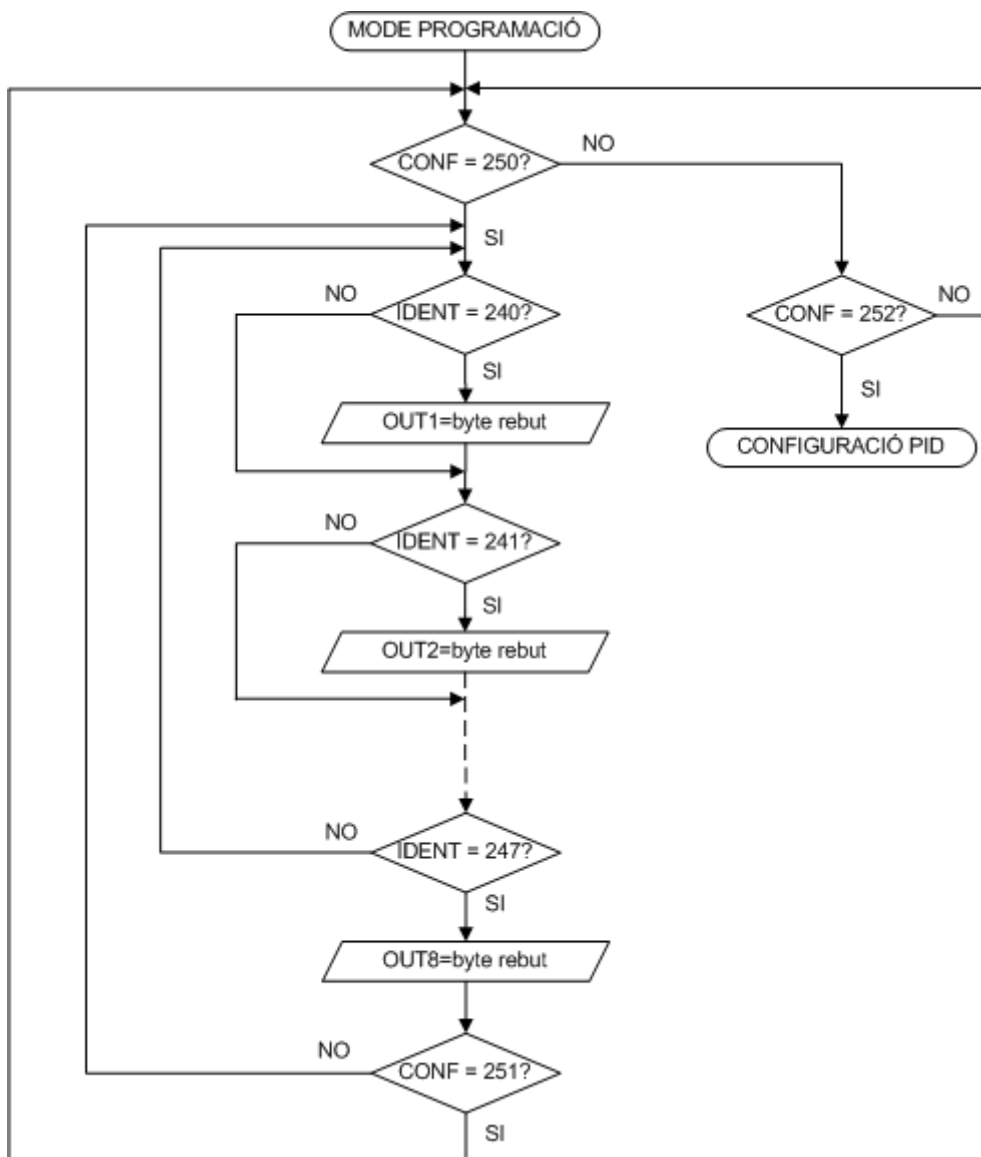
Els diferents botons del software de configuració realitzen les següents accions:

- Connectar: Configura i obre el port sèrie del PC per la comunicació amb el mòdul.
- Desconnectar: Tanca el port sèrie del PC.
- Configurar Sortides Digitals: envia al mòdul un byte amb valor 250, per indicar mode de configuració de sortides digitals.
- Validar Sortides Digitals: envia al mòdul un byte amb valor 251, i a continuació envia un byte amb l'identificador del número de sortida a configurar i el seu valor (0 o 1) per a cada una de les 8 sortides. El valor dels identificadors van del 240 al 247 per a les sortides 1 a 8, respectivament.
- Configurar Control PID: envia al mòdul un byte amb valor 252, per indicar mode de configuració del control PID.
- Validar Control PID: envia al mòdul un byte amb valor 253, i a continuació envia un byte amb l'identificador del paràmetre que s'envia i el seu valor, per a cada un dels paràmetres del controlador PID, la consigna (si és interna) i el temps de mostreig. El valor dels identificadors són 230 per Kp, 231 per Ti, 232 per Td, 233 per consigna externa o interna, 234 per la consigna i 235 per al temps de mostreig.

### **7.3.1. Configuració de l'estat de les sortides digitals**

Per realitzar qualsevol configuració, el switch de programació del mòdul microcontrolador universal ha d'estar en ON, ja que si no, no guardarà las dades que li enviem. Una vegada que indiquem el mode de configuració de sortides digitals amb el software de configuració, seleccionem l'estat que volem per a cada una de les sortides i validem la configuració, s'envien les dades que s'han descrit anteriorment al mòdul microcontrolador universal per el port RS232.

A continuació es mostra l'ordinograma que mostra el protocol que s'ha implementat en el microcontrolador per aquesta configuració. Una vegada realitzada aquesta transferència de dades, on el microcontrolador rep tota la informació necessària per configurar l'estat de les sortides, si es canvia l'estat del switch al mode de funcionament s'activaran les sortides seleccionades. Cal comentar que la interrupció generada quan es rep una dada per RS232 discrimina el tipus de dada que rep, diferenciant entre mode de configuració, identificadors i paràmetres de configuració (en aquest cas l'estat de les sortides).



**Fig. 7.6. Ordinograma configuració sortides**  
(Font: Elaboració pròpia)

### 7.3.2. Control PID

El software de configuració també permet configurar el mòdul microcontrolador universal per treballar com a controlador PID, permetent escollir els paràmetres del PID  $K_p$ ,  $T_i$  i  $T_d$ , i el temps de mostreig entre quatre diferents valors: 1 ms, 10 ms, 100 ms i 1 s.

La fórmula d'un controlador PID discretitzada es pot definir com:

$$u(k)=u(k-1)+K_p \cdot \left[ 1 + \frac{T_m}{2T_i} + \frac{T_d}{T_m} \right] \cdot e(k) + K_p \cdot \left[ \frac{T_m}{2T_i} \cdot \frac{2T_d}{T_m} - 1 \right] \cdot e(k-1) + \frac{K_p \cdot T_d}{T_m} \cdot e(k-2)$$

On:

**$T_m$**  és el temps de mostreig.

**$K_p$ ,  $T_i$  i  $T_d$**  són els paràmetres de sintonia del controlador.

**$e(k)$ ,  $e(k-1)$  i  $e(k-2)$**  són la mostra actual, mostra anterior i mostra dues vegades anterior respectivament.

**$u(k)$  i  $u(k-1)$**  són el resultat actual i anterior de l'acció de control respectivament.

Una vegada definits els paràmetres de sintonia i el temps de mostreig del controlador, tenim els següents valors constants:

$$A = K_p \cdot \left[ 1 + \frac{T_m}{2T_i} + \frac{T_d}{T_m} \right]$$

$$B = K_p \cdot \left[ \frac{T_m}{2T_i} \cdot \frac{2T_d}{T_m} - 1 \right]$$

$$C = \frac{K_p \cdot T_d}{T_m}$$

D'aquesta manera, el mòdul microcontrolador universal haurà d'executar permanentment el càlcul de l'equació d'un controlador PID definida com:

$$u(k)=u(k-1)+A \cdot e(k)+B \cdot e(k-1)+C \cdot e(k-2)$$

Per seleccionar els paràmetres de sintonia del PID, el temps de mostreig, indicar si la consigna és externa o interna, i el valor de la consigna en cas de ser externa, el protocol de comunicació entre el software de configuració i el microcontrolador és similar que en el cas de la configuració de les sortides digitals, però variant els valors de configuració, validació i identificadors de paràmetres. Una altra diferència és que hi ha valors que poden tenir decimals, com els paràmetres  $K_p$ ,  $T_i$  i  $T_d$  o el valor de la consigna, per tant s'ha de passar aquest valor amb decimals (s'ha fixat dos dígitos per la part sencera i dos per els decimals) a 4 bytes per permetre la comunicació entre el PC i el microcontrolador. Una vegada el microcontrolador ha rebut els bytes corresponents a cada paràmetre, ha de tornar a convertir-los en un nombre decimal (amb variables tipus float). A continuació es mostra l'ordinograma corresponent.

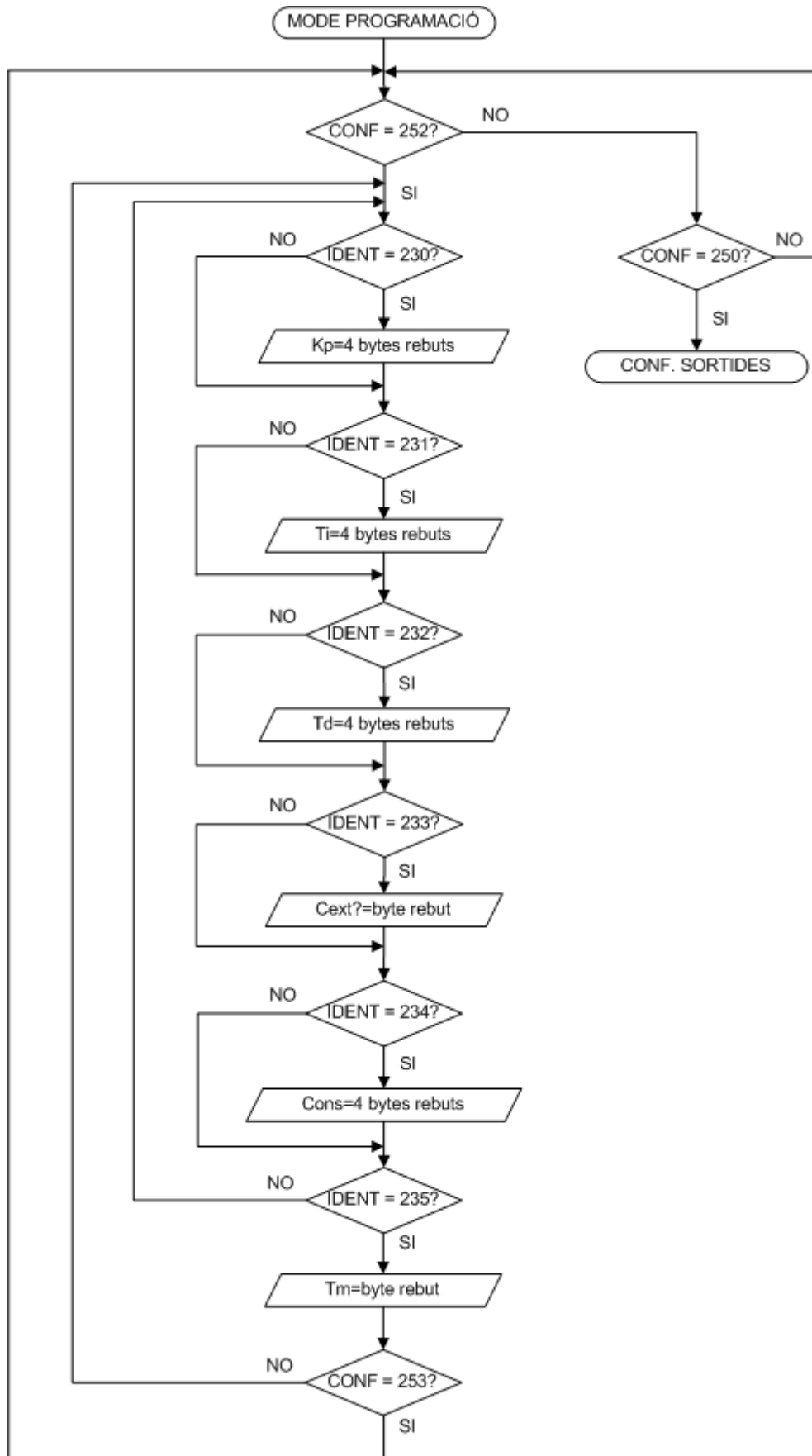


Fig. 7.7. Ordinograma configuració PID

El càlcul de l'equació i l'acció de control per part del mòdul microcontrolador universal s'ha de realitzar amb una freqüència que ve determinada per el temps de mostreig.

Quan el mòdul microcontrolador universal hagi estat configurat per treballar com a controlador PID, es calculen els valors de les constants A, B i C amb els paràmetres  $K_p$ ,  $T_i$ ,  $T_d$  i el temps de mostreig.

Una vegada s'entra en el mode de funcionament, el microcontrolador entrarà en un bucle on no es farà res, i només s'espera al moment en el que ha de realitzar el càlcul de l'equació de control.

Per indicar al microcontrolador cada quant ha de realitzar el càlcul i efectuar l'acció de control s'utilitzarà el Timer0 del mateix.

### **Timer0**

El Timer0 del microcontrolador PIC18F97J60 té les següents característiques:

- Mode d'operació com temporitzador o comptador, ambdós en format de 16 o 8 bits.
- Prescaler de 8 bits
- Configuració del generador de rellotge com extern o intern
- Flanc d'increment per el rellotge extern
- Interrupció per desbordament

Registres que utilitza el Timer0:

TMR0L (Timer0 Register Low Byte)

TMR0H (Timer0 Register High Byte)

INTCON (Interrupt Control) → TMR0IF (Timer0 Interrupt Flag)

T0CON (Timer0 Control)

La interrupció del Timer0 es genera quan es produeix un overflow en el seu registre, de FFh a 00h en mode de 8 bits i de FFFFh a 0000h en mode de 16 bits. Quan es produeix



aquest overflow s'activa el bit de flag d'interrupció del timer0 (TMR0)IF. La interrupció pot ser emmascarada posant a 0 el bit TMR0IE (INTCON<5>). Abans de tornar a habilitar la interrupció, el bit TMR0IF s'ha de posar a 0 per software a la rutina de servei d'interrupció.

Com s'ha comentat anteriorment, el temps de mostreig del controlador PID es pot escollir entre quatre valors: 1 ms, 10 ms, 100 ms i 1 s.

Per cada un d'aquests quatre possibles valors de temps de mostreig, s'haurà de configurar el prescaler i el valor carregat al timer, per tal de definir el període del mateix.

Si treballem amb el timer0 utilitzant la font de rellotge interna i en format de 16 bits, el període del timer ve determinar per la següent expressió:

$$T_{\text{timer0}} = \frac{4 \cdot \text{valor prescaler} \cdot (65535 - \text{valor carregat al timer})}{f_{\text{osc}}}$$

Per cada un dels diferents períodes de mostreig s'han escollit els següents valors per al valor del prescaler i el valor carregat al timer:

Tm = 1 ms : valor prescaler = 1, valor carregat al timer = 59285

Tm = 10 ms : valor prescaler = 1, valor carregat al timer = 3035

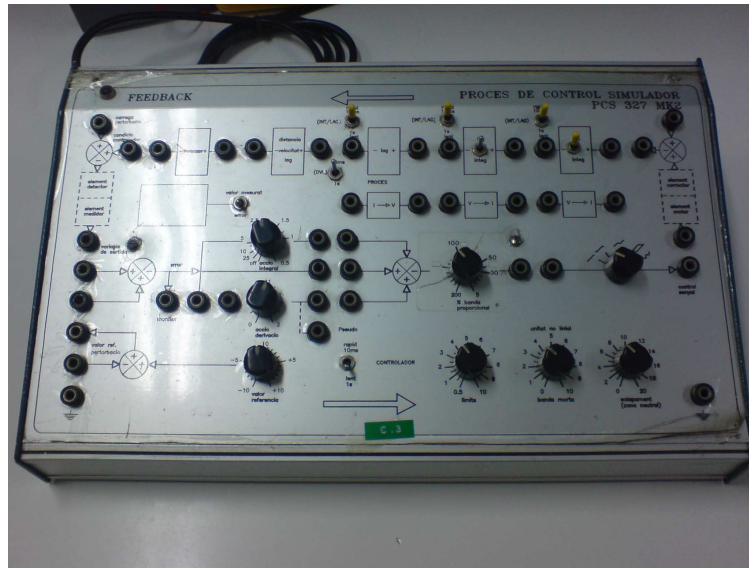
Tm = 100 ms : valor prescaler = 128, valor carregat al timer = 60652

Tm = 1 s : valor prescaler = 128, valor carregat al timer = 16707

Per realitzar les proves de funcionament del control PID, s'ha optat utilitzar el simulador analògic de sistemes de control PCS 327.

**El simulador PCS 327**

El simulador PCS 327 està format per un conjunt de circuits electrònics realitzats amb amplificadors operacionals, que es poden interconnectar des de l'exterior del mòdul per tal de poder realitzar sistemes de diferent ordre i diferent constant de temps.



**Fig. 7.8. Simulador PCS 327**  
(Font: Elaboració pròpia)

El rang de la tensió dels senyals de totes les entrades i sortides és de 10 V, tant negatius com positius.

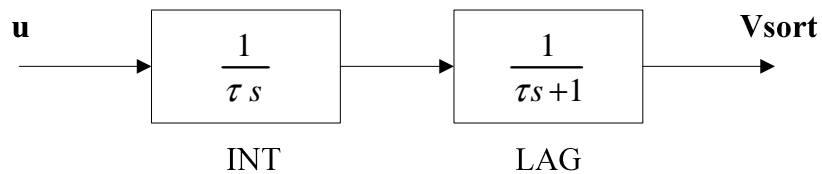
Per realitzar el control PID amb el mòdul microcontrolador universal, només necessitem el procés simulat del simulador PCS 327.

El procés està format pels següents elements:

- 1 LAG simple amb una constant de temps de 10 ms o 1 s.
- 2 LAGs amb una constant de temps de 10 ms o 1s, ambdós disposen d'interruptor per tal de convertir-los en integradors, segons la necessitat.
- 1 LAG “distance-velocity” amb un retard de 10 ms o 1 s.
- 1 inversor

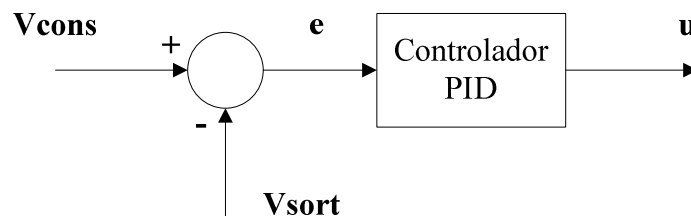
La selecció del valor de les constants de temps, tant pels lags com pels integradors es realitza mitjançant un interruptor, el qual actua de manera simultània en els tres blocs inicialment descrits. La posició FAST correspon a 10 ms i la posició SLOW correspon a 1 s. Cal tenir en compte que cada element produeix una inversió de polaritat. D'aquesta manera, per una entrada al procés positiva, s'obindrà una sortida sense canvi de polaritat, si el nombre d'elements connectats és parell, però la sortida serà invertida si aquest nombre d'elements és senar.

Si agafem del simulador PCS 327 dos LAGs, un d'ells configurat com a integrador i l'altre com a LAG, tindrem un sistema equivalent a un motor actuant com a seguidor de posició.



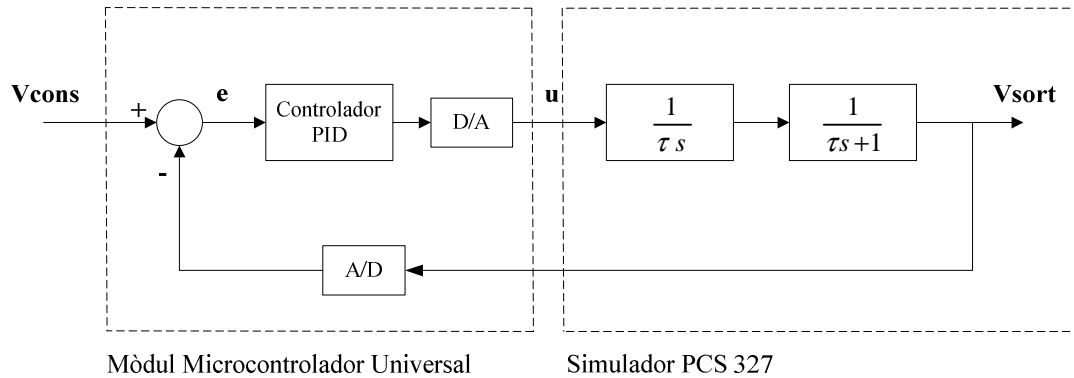
**Fig. 7.9. Diagrama de blocs del procés simulat (FdT)**  
(Font: Elaboració pròpia)

En un sistema de control realimentat, el que fem és comparar el valor de la variable a controlar ( $V_{sort}$ ) amb un valor de consigna,  $V_{cons}$ . En funció del senyal d'error que obtenim ( $e$ ), a través d'un controlador generem la tensió d'entrada per al sistema a controlar ( $u$ ).



**Fig. 7.10. Diagrama de blocs d'un sistema de control realimentat**  
(Font: Elaboració pròpia)

El mòdul microcontrolador universal realitzarà la funció del controlador, en concret un controlador PID.



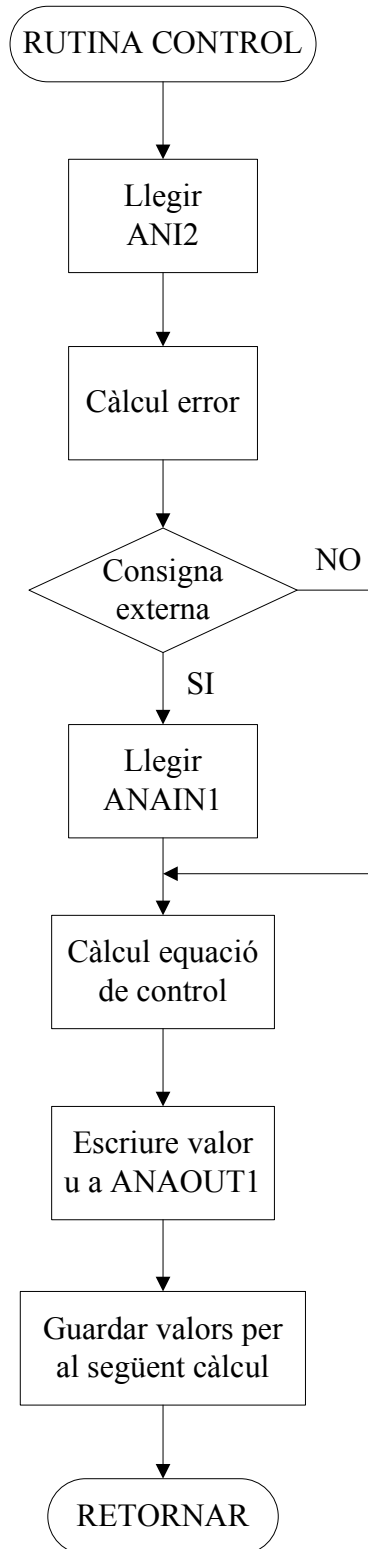
**Fig. 7.11. Diagrama de blocs del sistema**  
(Font: Elaboració pròpia)

També ens farà falta adaptar el rang de les tensions analògiques del mòdul microcontrolador universal amb el del simulador PCS 327.

El rang de les entrades analògiques del mòdul va dels 0 als 3.3 V, i el de les sortides analògiques dels 0 als 5 V. Com s'ha comentat anteriorment, el rang de la tensió dels senyals de totes les entrades i sortides del simulador PCS 327 és de 10 V, tant negatius com positius. Per tant, s'haurà de realitzar un canvi d'escala a l'entrada analògica del mòdul de  $-10 \div +10\text{V}$  a  $0 \div 3.3\text{ V}$ , i un altre canvi d'escala a la sortida analògica del mòdul de  $0 \div 5\text{ V}$  a  $-10 \div +10\text{V}$ .

Per fer aquestes adaptacions de tensió, es realitzaran uns circuits amb amplificadors operacionals i resistències que resolguin aquests canvis d'escala. Aquests circuits seran exteriors al mòdul, i es muntaran en un protoboard.

L'algoritme que s'utilitza a la rutina de servei d'interrupció per realitzar el control PID quan el Timer0 genera una interrupció (període de mostreig) és el següent:



**Fig. 7.12. Ordinograma rutina control PID**  
(Font: Elaboració pròpia)



## 8. CONCLUSIONS I MILLORES

L'objectiu principal d'aquest projecte ha estat el disseny i construcció d'un mòdul microcontrolador universal. Aquesta feina de disseny s'ha començat des de zero, analitzant controladors que hi ha al mercat i decidint com tenia que ser el nostre.

Despres de haver passat per les diferents etapes del projecte, finalment s'ha aconseguit construir un producte que compleix amb les expectatives que s'havien marcat al principi.

El mòdul microcontrolador universal realitza totes les funcions previstes inicialment, amb algun petit problema com la integració de totes les funcions, ja que al no haver disposat de temps per desenvolupar una pila TCP/IP pròpia, es depenia de la pila subministrada per Microchip, que ens implica certes restriccions a l'hora d'integrar les aplicacions. També s'han tingut dificultats, com per exemple la soldadura del microcontrolador. Això ha permès conèixer mètodes alternatius per a la soldadura de components SMD, així com experiència en aquests mètodes.

S'ha aconseguit connectar el mòdul a una xarxa Ethernet, així com crear una pàgina web que actua com a servidor informant de l'estat del mòdul i permetent realitzar certes accions.

També s'ha aconseguit desenvolupar un senzill software per configurar aplicacions en el mòdul microcontrolador universal amb un PC via RS232, i treballar amb els dos prototips construïts connectats via RS485.

En aquest projecte s'ha pogut comprovar la capacitat que en proporciona un microcontrolador per integrar-se en una xarxa Ethernet i les possibilitats que ens ofereix, ja que les aplicacions realitzades en aquest projecte són una petita mostra de tot el potencial que tenen aquests tipus de microcontroladors. Amb temps, coneixements i dedicació es poden desenvolupar aplicacions que semblen reservades a dispositius molt més cars.

En referència a les millores per al mòdul microcontrolador universal, després d'aquests mesos de treball han aparegut moltes idees. Una millora molt clara és la realització d'una pila TCP/IP pròpia, utilitzant els mòduls necessaris i gestionant els recursos del microcontrolador convenients.

Un altre futura millora, és en l'apartat de la connectivitat, sens dubte un aspecte clau actualment, i més en els sistemes embedded. Aquesta millora consisteix en la incorporació de la tecnologia de localització GPS i connectivitat Internet GPRS/UMTS, que permetrien treballar al mòdul microcontrolador universal en aplicacions mòbils. En el mercat trobem mòduls de petites dimensions que incorporen les dues funcions. Aquesta millora incorporaria molt valor al mòdul microcontrolador universal.

En relació amb la millora de connectivitat, ens apareix la necessitat d'alimentar el mòdul microcontrolador universal amb bateries, per poder realitzar tasques de control mòbil.

En les proves realitzades durant aquest projecte, la connectivitat Ethernet s'ha realitzat en una xarxa local. Un altre millora seria la ampliació per accedir des de qualsevol punt amb connexió a Internet. Aquesta millora seria relativament senzilla de realitzar, però ens apareixen nous reptes com la millora en termes de seguretat i encriptació de dades.

Un altre futura millora per al mòdul microcontrolador universal seria l'ampliació o creació d'un nou software que permeti configurar i programar el mòdul per treballar en diferents aplicacions en un entorn PC, sense tenir que utilitzar eines específiques per programar el microcontrolador. En aquest projecte s'ha creat un senzill software amb la possibilitat de configurar l'estat de les sortides digitals i la configuració per treballar com a controlador PID, però es podria ampliar el tipus d'aplicacions utilitzant llibreries per facilitar la configuració de diferents modes de treball.

Totes aquestes millores que s'han comentat, afegirien valor al mòdul microcontrolador universal dissenyat en aquest projecte, convertint-lo en un producte molt interessant i amb possibilitats reals de tenir sortida en el mercat.



Com a experiència personal, la realització d'aquest projecte final de carrera ha permès ampliar coneixements adquirits durant els estudis i adquirir de nous, com en el disseny de sistemes electrònics, funcionament de microcontroladors, comunicacions industrials, disseny d'aplicacions amb microcontroladors i en llenguatge de programació C i html, entre altres.



## 9. BIBLIOGRAFIA

- [1] Jeremy Bentham, *TCP/IP Lean Web Servers for Embedded Systems*. Ed. CMPBooks, 2002
  
- [2] Joan Domingo Peña, Juan Gámiz Caro, Antoni Grau i Saldes, Herminio Martínez García, *Comunicaciones en el entorno industrial*. Ed. UOC, 2003
  
- [3] José M<sup>a</sup> Angulo Usategui, Ignacio Angulo Martínez, *Microcontroladores PIC. Diseño práctico de aplicaciones. Primera parte*. Tercera edición. Mc Graw Hill 2003
  
- [4] Ramón Pallás Areny, *Adquisición y distribución de señales*. Ed. Marcombo, 1993
  
- [5] Mar Roselló Canal, *Aplicaciones electromagnéticas y electrónicas*. P08/89018/00542. Ed. UOC
  
- [6] Jordi Gonzalez Ribelles, *Sistema de control con microcontroladores PIC16C71*. EUPMT 1997
  
- [7] Enric Navarro López, *Desarrollo de aplicaciones internet embedded con el kit PICDEM.NET*. EUPMT 2004
  
- [8] Microchip Technology Inc. *PIC18F97J60 Family Data Sheet*. DS39762D. 2008
  
- [9] Microchip Technology Inc. AN833. *The Microchip TCP/IP Stack*. DS00833C. 2008
  
- [10] Microchip Technology Inc. AN1120. *Ethernet Theory of Operation*. DS01120A. 2008
  
- [11] Microchip Technology Inc. *MPLAB IDE User's Guide with MPLAB Editor and MPLAB SIM Simulator*. DS51519C. 2009

[12] Microchip Technology Inc. *MPLAB C18 C Compiler Getting Started*. DS51295F. 2005

[13] Microchip Technology Inc. *MPLAB C18 C Compiler User's Guide*. DS51288J. 2005

[14] Microchip Technology Inc. *MPLAB C18 C Compiler Libraries*. DS51297F. 2005

[15] “Horno de Refusión de Precisión SMT”. *Revista Elektor*. Octubre 2008. nº340

[16] “Técnicas de Refusión”. *Revista Elektor*. Diciembre 2007. nº330

[17] <http://www.microchip.com>, Fabricant dels microcontroladors PIC

[18] <http://es.rs-online.com/web>, RS-Amidata, distribuïdor de productes electrònics a nivell mundial

[19] <http://es.farnell.com>, Farnell, distribuïdor de productes electrònics a nivell mundial



Escola Universitària  
Politécnica de Mataró

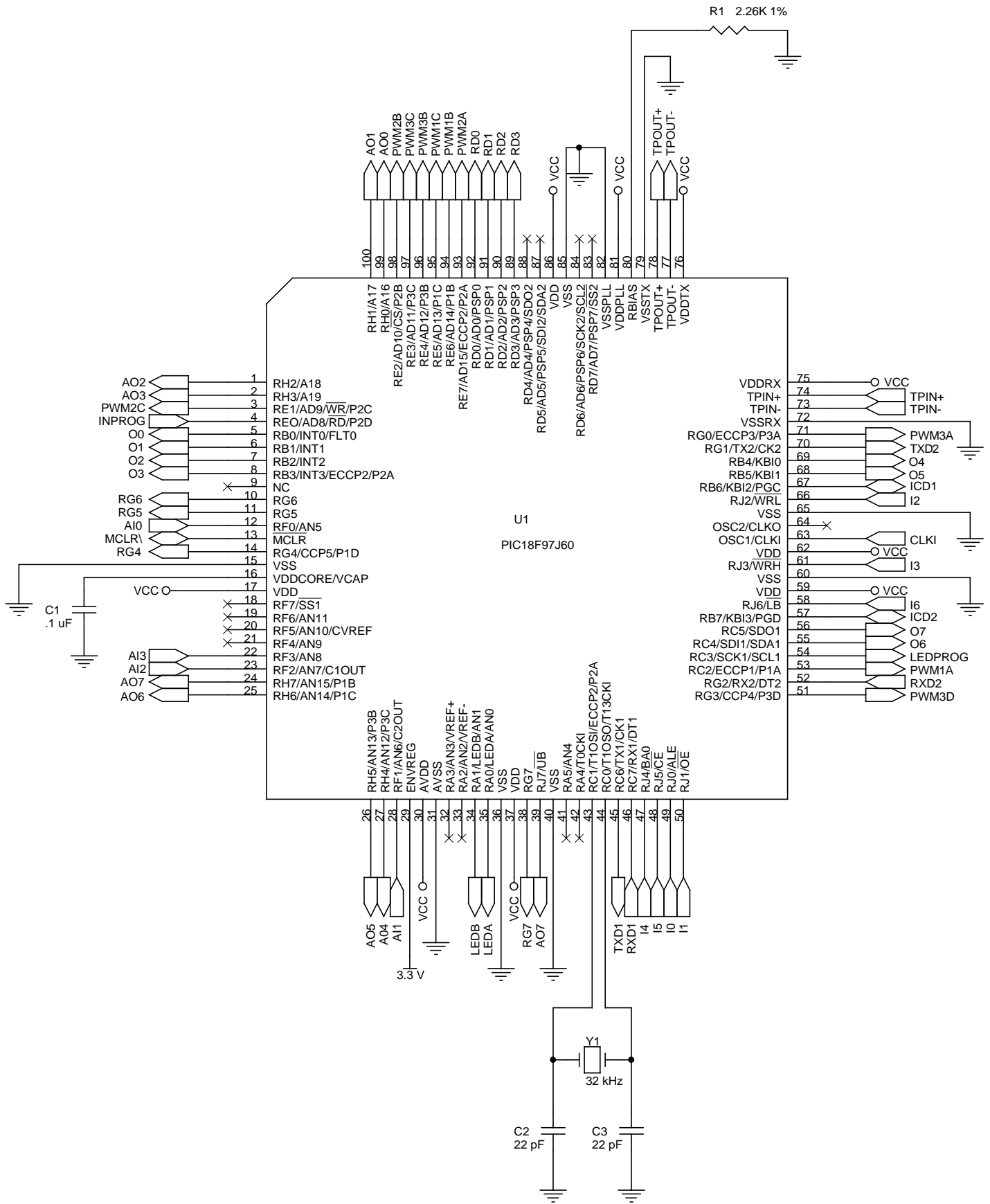
**Enginyeria Tècnica Industrial: Especialitat Electrònica Industrial**

**Mòdul Microcontrolador Universal**

**B - Esquemes**

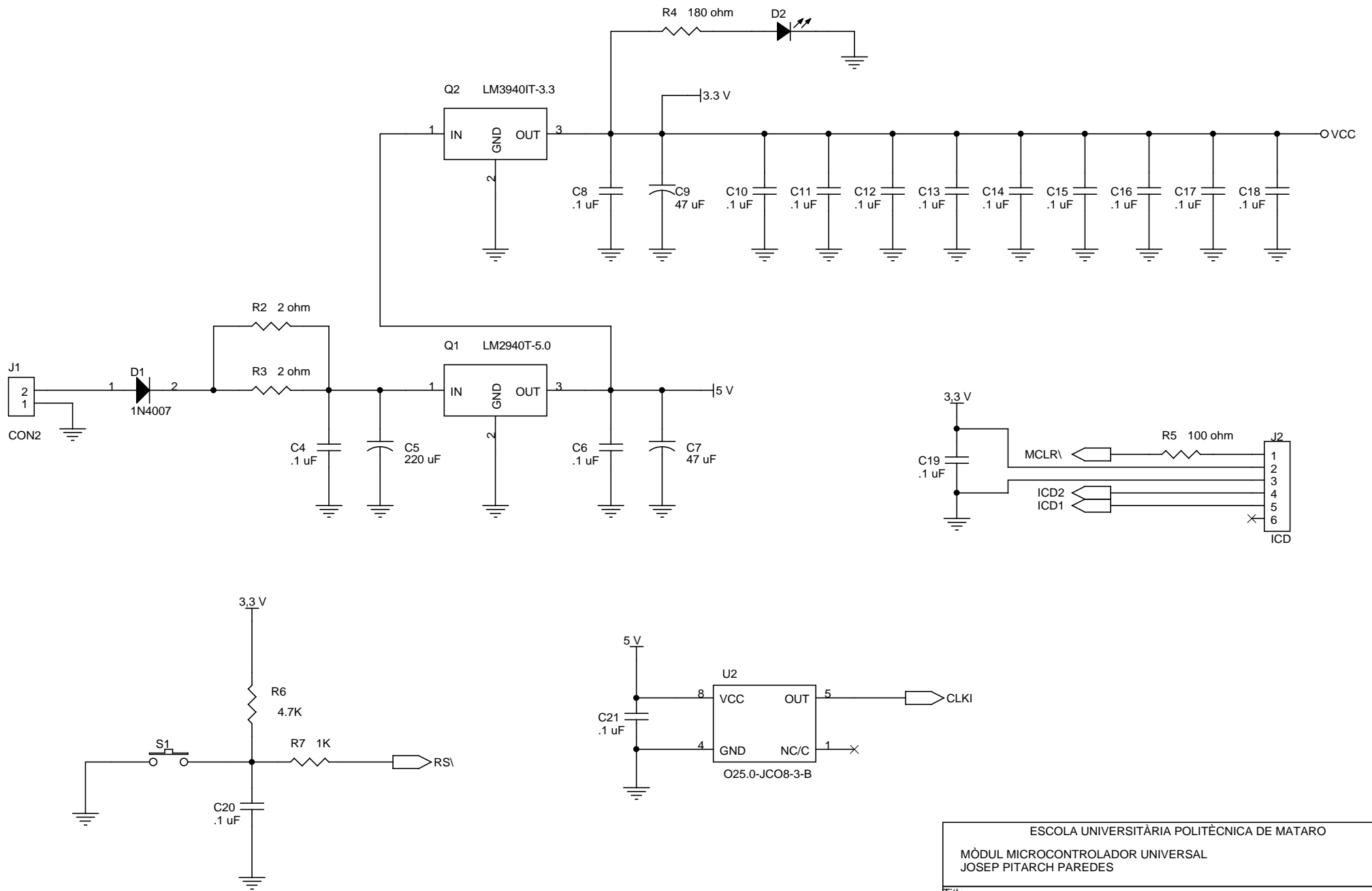
**JOSEP PITARCH PAREDES  
JULIÁN HERRILLO TELLO  
PRIMAVERA 2009**





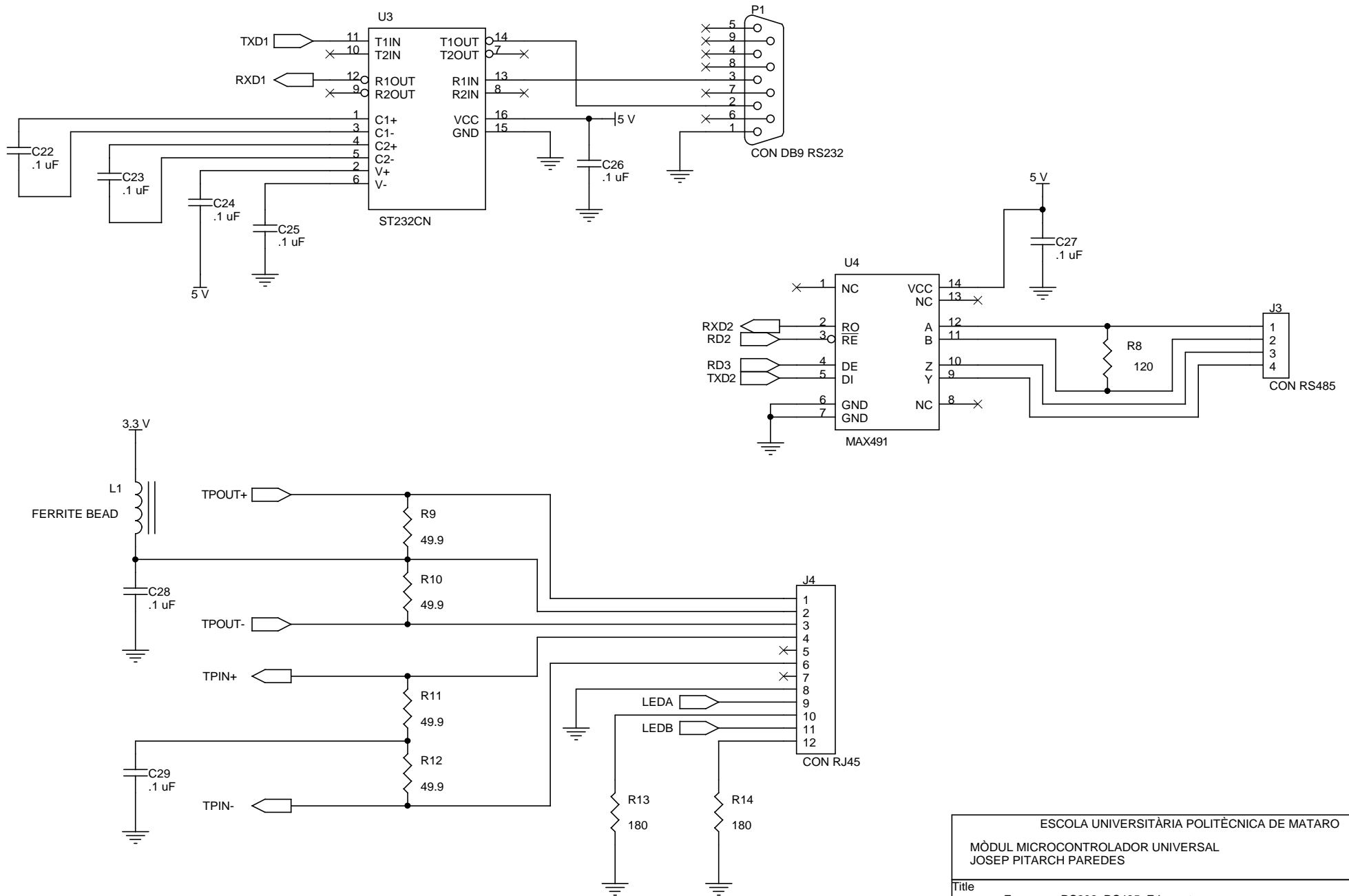




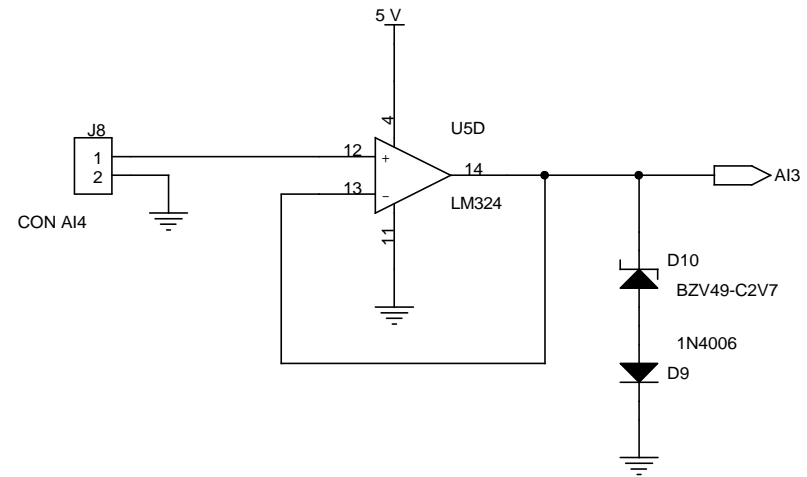
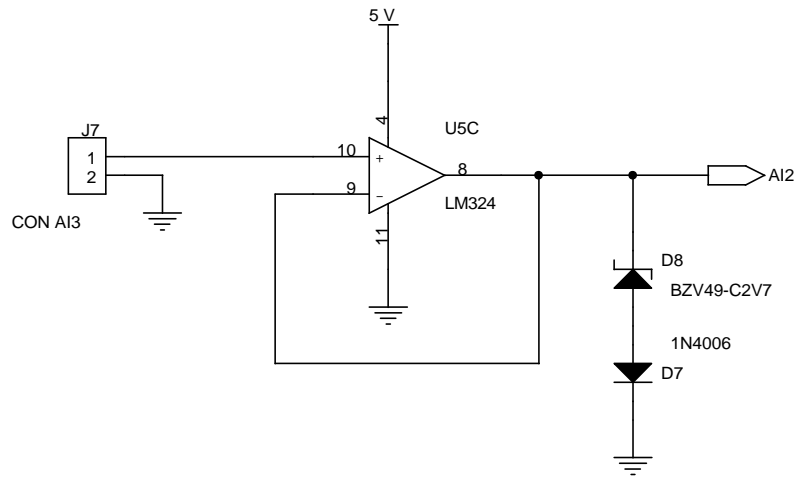
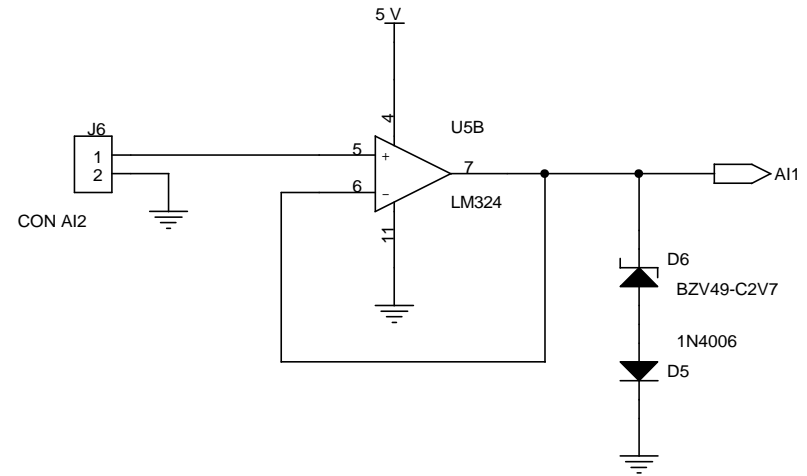
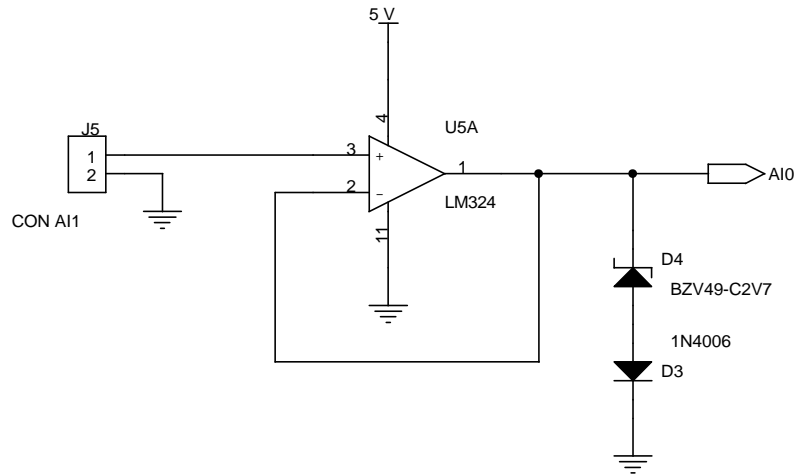


ESCOLA UNIVERSITÀRIA POLITÈCNICA DE MATARO MÒDUL MICROCONTROLADOR UNIVERSAL JOSEP PITARCH PAREDES		
Title		
Esquema: Alimentació, reset, ICD i oscil·lador		
Size	Document Number	Rev
A4	B-2	1
Date:	Wednesday, June 24, 2009	Sheet 2 of 5



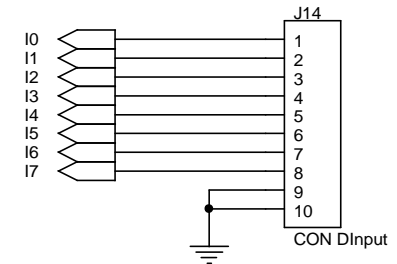
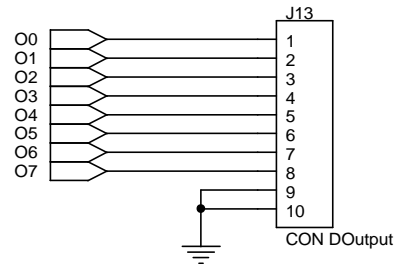
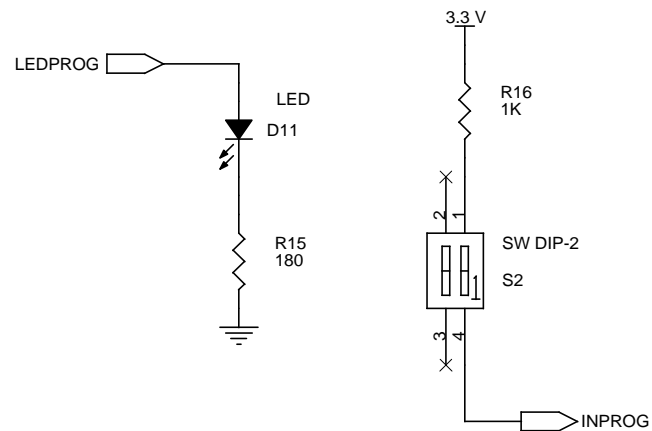
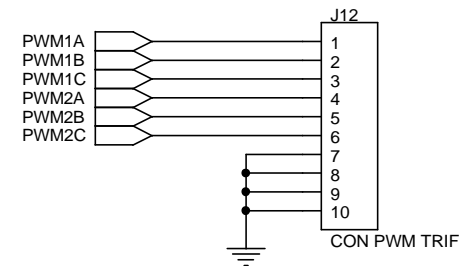
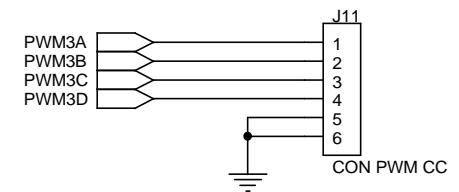
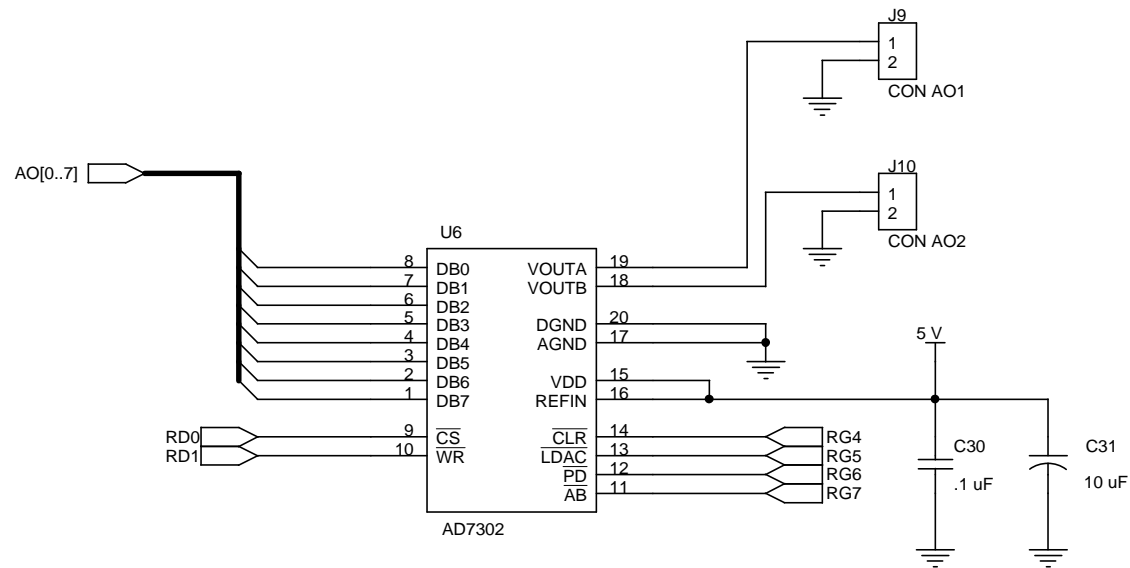






ESCOLA UNIVERSITÀRIA POLITÈCNICA DE MATARO		
MÒDUL MICROCONTROLADOR UNIVERSAL JOSEP PITARCH PAREDES		
Title		
Esquema: Entrades analògiques		
Size	Document Number	Rev
A4	B-4	1
Date:	Wednesday, June 24, 2009	Sheet 4 of 5











Escola Universitària  
Politécnica de Mataró

**Enginyeria Tècnica Industrial: Especialitat Electrònica Industrial**

**Mòdul Microcontrolador Universal**

**C – Pressupost**

**JOSEP PITARCH PAREDES  
JULIÁN HERRILLO TELLO  
PRIMAVERA 2009**



# ÍNDEX

1. COST DEL PROTOTIP .....	1
1.1. Cost del material .....	1
1.2. Costos d'enginyeria .....	3
1.3. Amortització Instrumental .....	3
1.4. Cost de fabricació del prototip.....	4
2. PREU DE VENTA EN EL MERCAT .....	5
2.1. Estudi econòmic per la fabricació de 10000 unitats .....	5
2.2. Estudi del preu de venda en el mercat .....	6



# 1. COST DEL PROTOTIP

## 1.1. Cost del material

Cost del material per al muntatge d'un prototip:

Descripció	Quantitat	Preu unitat	Total
<b>Circuits integrats</b>			
PIC18F97J60	1	7,62 €	7,62 €
DAC AD7302	1	6,05 €	6,05 €
ST232CN	1	0,50 €	0,50 €
MAX491	1	2,70 €	2,70 €
OP AMP LM324	1	0,13 €	0,13 €
Regulador LM2940T-5.0	1	1,59 €	1,59 €
Regulador LM3940IT-3.3	1	1,81 €	1,81 €
<b>Oscil·ladors i cristalls de quars</b>			
Oscil·lador 25 MHz	1	2,89 €	2,89 €
Cristall quars 32 kHz	1	0,32 €	0,32 €
<b>Condensadors</b>			
Condensador 22 pF	2	0,02 €	0,04 €
Condensador 47 uF	2	0,04 €	0,07 €
Condensador 220 uF	1	0,46 €	0,46 €
Condensador 10 uF	1	0,03 €	0,03 €
Condensador 0.1 uF	26	0,03 €	0,73 €
<b>Diodes i LEDs</b>			
Diode LED vermell	1	0,16 €	0,16 €
Diode LED verd	1	0,16 €	0,16 €
Diode 1N 4007	1	0,02 €	0,02 €
Diode 06	4	0,02 €	0,08 €
Diode Zener 2V7	4	0,05 €	0,20 €
<b>Resistències</b>			
Resistència 2,26 k $\Omega$ 1/4 W	1	0,07 €	0,07 €
Resistència 100 $\Omega$ 1/4 W	1	0,04 €	0,04 €
Resistència 4,7 k $\Omega$ 1/4 W	1	0,04 €	0,04 €
Resistència 1 k $\Omega$ 1/4 W	8	0,04 €	0,32 €
Resistència 120 $\Omega$ 1/4 W	1	0,04 €	0,04 €
Resistència 49,9 $\Omega$ 1/4 W	4	0,07 €	0,28 €
Resistència 180 $\Omega$ 1/4 W	2	0,04 €	0,08 €

<b>Connectors</b>			
Connector DB9	1	0,58 €	0,58 €
Connector WE-LAN RJ45	1	6,70 €	6,70 €
Connector RJ11 6p	1	0,36 €	0,36 €
Con.acolzat 2p femella	7	1,13 €	7,91 €
Con.acolzat 4p femella	1	1,91 €	1,91 €
Con.acolzat 6p femella	1	2,70 €	2,70 €
Con.acolzat 2p mascle	7	0,30 €	2,10 €
Con.acolzat 4p mascle	1	0,58 €	0,58 €
Con.acolzat 6p mascle	1	0,84 €	0,84 €
Con.cinta plana 10p femella	3	0,19 €	0,57 €
Con.cinta plana 10p mascle	3	0,19 €	0,57 €
<b>PCBs</b>			
ADAPTADOR SMD RE471	1	24,94 €	24,94 €
Placa WR Europea 160x100	1	3,60 €	3,60 €
<b>Polsadors</b>			
Polsador miniatura CI	1	0,09 €	0,09 €
<b>Sòcols</b>			
Sòcol WR DIP 14	2	0,73 €	1,46 €
Sòcol WR DIP 16	1	1,05 €	1,05 €
Sòcol WR DIP 20	1	1,64 €	1,64 €
<b>Varis</b>			
Ferrite bead	1	0,25 €	0,25 €
Cable WR	12	0,32 €	3,79 €
Connexions C.I. WR	110	0,038	4,18 €
<b>Cost del material</b>			<b>92,25 €</b>

## 1.2. Costos d'enginyeria

Concepte	Hores	Preu/hora	Total
Estudi i documentació	50	60 €	3.000 €
Disseny	50	60 €	3.000 €
Muntatge i posada a punt	200	30 €	6.000 €
Redacció memòria	50	30 €	1.500 €

<b>Total</b>	<b>13.500 €</b>
--------------	-----------------

## 1.3. Amortització Instrumental

- Instrumentació informàtica:

Equip utilitzat	Hores d'utilització	Preu/hora	Total
Ordinador	200	0,50 €	100 €
Software MPLAB IDE 8.2	130	0,00 €	0 €
Software ORCAD	20	2,00 €	40 €
Microsoft Office	50	1,00 €	50 €

<b>Total</b>	<b>190 €</b>
--------------	--------------

- Instrumentació electrònica:

Equip utilitzat	Hores d'utilització	Preu/hora	Total
Oscil·loscopi	20	3,00 €	60 €
Multímetre	10	0,50 €	5 €
Font d'alimentació	100	0,50 €	50 €
Equip de soldadura	10	10,00 €	100 €

<b>Total</b>	<b>215 €</b>
--------------	--------------

### 1.4. Cost de fabricació del prototip

Costos de material	92,25 €
Costos d'enginyeria	13.500 €
Costos d'amortització	405 €
<b>Total</b>	<b>13.997,25 €</b>



## 2. PREU DE VENTA EN EL MERCAT

### 2.1. Estudi econòmic per la fabricació de 10000 unitats

Abans de realitzar l'estudi econòmic per la fabricació de 10000 unitats, cal fer algunes consideracions:

- Els costos de fabricació són els corresponents al muntatge en sèrie del mòdul microcontrolador universal. Aquest muntatge es subcontractaria a una empresa externa especialitzada per tal de minimitzar costos de fabricació.

<b>Costos de fabricació per una unitat</b>	<b>12 €</b>
--	-------------

- Alguns dels components utilitzats en la construcció dels prototips no es corresponen amb els components que s'utilitzarien en el producte final, com per exemple diferents tipus d'encapsulats, ja que tot el muntatge es realitzaria amb SMT.
- L'adaptador SDM RE471 no s'utilitzaria en el muntatge final, per tant ens estalviem el cost més important que ens trobem en la llista de materials de 24,94 €. Per altra banda, s'hauria d'afegir el cost d'una PCB feta a mida per el mòdul microcontrolador universal i una carcassa de plàstic per al producte. Si tenim en compte aquestes consideracions, podem considerar que el cost del material quedaria pràcticament igual per al muntatge del mòdul.
- Al realitzar un estudi econòmic per la fabricació de 10000 unitats, considerem un descompte del 25% per la compra del material en grans quantitats.

<b>Costos</b>		<b>Total</b>
Cost total del material	$(92,47-23,06)*10000$	691.875 €
Cost de fabricació	$12*10000$	120.000 €
Cost total per 10000 unitats		811.875 €
<b>Cost per unitat</b>		<b>81,19 €</b>

El cost obtingut per unitat és de 81,35 €, però a l'hora de posar el producte al mercat s'ha d'afegir el cost de fabricació del prototip, que conté els costos d'enginyeria i amortitzacions.

## 2.2. Estudi del preu de venda en el mercat

<b>Concepte</b>	<b>Total</b>
Cost total per 10000 unitats	811.875 €
Cost del prototip	13.997,25 €
Total	825.872 €
Preu unitari	82,59 €
<b>Preu amb un benefici del 30%</b>	<b>107,37 €</b>



Escola Universitària  
Politécnica de Mataró

**Enginyeria Tècnica Industrial: Especialitat Electrònica Industrial**

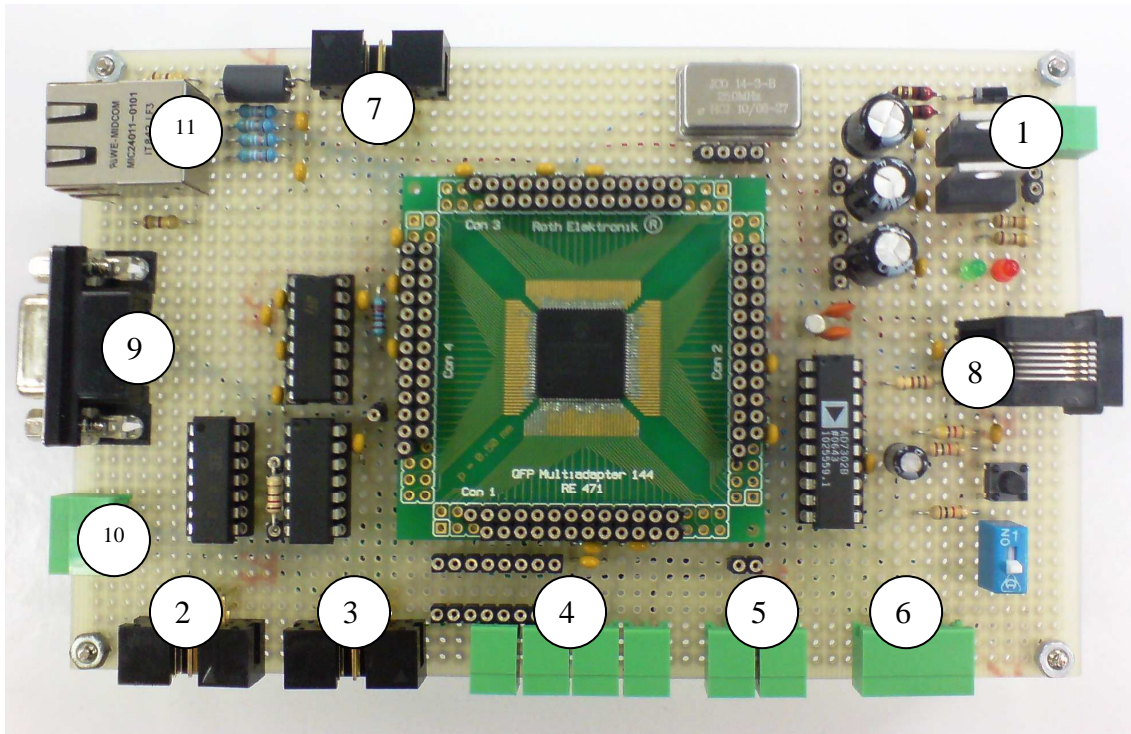
**Mòdul Microcontrolador Universal**

**D – Annexos**

**JOSEP PITARCH PAREDES  
JULIÁN HERRILLO TELLO  
PRIMAVERA 2009**



## ANNEX 1: DISTRIBUCIÓ I CONFIGURACIÓ DE CONNECTORS



- 1) Alimentació: pin 1 V-, pin 2 V+
- 2) Entrades digitals: pins 1 – 8 entrades 1 – 8 respectivament, pins 9 i 10 massa
- 3) Sortides digitals: pins 1 – 8 sortides 1 – 8 respectivament, pins 9 i 10 massa
- 4) Entrades analògiques (d'esquerra a dreta de la 1 fins a la 4): pin 1 V+, pin 2 massa
- 5) Sortides analògiques (d'esquerra a dreta de la 1 fins a la 2): pin 1 V+, pin 2 massa
- 6) Sortides PWM (control motors CC): pins 1 – 4 sortides PWM A, B, C i D respectivament, pins 5 i 6 massa
- 7) Sortides PWM (trifàsic): pins 1 – 6 sortides PWM 1A, 1B, 1C, 2A, 2B i 2C respectivament, pins 7 – 10 massa
- 8) Connector ICD
- 9) Connector RS232 (DB9)
- 10) Connector RS485: pins 1 i 2 senyals diferencials d'entrada A i B, i pins 3 i 4 senyals diferencials de sortida Z i Y
- 11) Connector Ethernet (RJ45)



## **ANNEX 2: CONTINGUT DEL CD-ROM**

- Documentació del projecte (memòria i article resum)
- Data Sheets dels components del prototip
- Manuals MPLAB IDE i MPLAB C18
- Notes d'aplicació
- Codi HTML del servidor web
- Arxius Visual Basic (software configuració)
- Codi font C (aplicacions microcontrolador)
- Utilitats de Microchip