



**Universitat Autònoma
de Barcelona**

**Monitoratge Remot
d'Esdeveniments per al Control
i Supervisió utilitzant "System on
chip" i "w-USB"**

Memòria del Projecte

D'Enginyeria Tècnica en

Informàtica de Sistemes

Realitzat per

"Alberto Carrete Geijo"

i dirigit per

"Raúl Aragonés Ortíz"

Escola Universitària d'Informàtica
Sabadell, Juny de 2010

El sota signant, Raúl Aragonés Ortíz,
professor de l'Escola Universitària d'Informàtica de la UAB,

CERTIFICA:

Que el treball al que correspon la present memòria
ha estat realitzat sota la seva supervisió
per en **Alberto Carrete Geijo**.

I per a que consti firma la present.
Sabadell, Juny de 2010

Signat: **Raúl Aragonés Ortíz**

Presentació

Projecte:

Monitoratge Remot d'Esdeveniments per al Control i Supervisió utilitzant "System on chip" i "w-USB"

Autor:

Alberto Carrete Geijo

Director:

Raúl Aragonés Ortíz

Departament:

Departament de Microelectrònica i Sistemes electrònics

Resum de la memòria:

Aquesta memòria descriu el procés de desenvolupament del projecte de fi de carrera "Monitoratge Remot d'Esdeveniments per al Control i Supervisió utilitzant System on chip y w-USB".

Es tracta d'un conjunt de hardware, "PSoC" (Programmable System on Chip), i un software, C#, mitjançant els quals s'automatitza la gestió de comandes a les taules d'un restaurant.

A cada taula, trobem uns aparells anomenat "WaiterClient", a través del qual els clients sol·liciten l'atenció d'un cambrer. Aquest hardware té una pantalla on es mostrarà informació i un conjunt de polsadors per demanar.

Per una altra banda, trobem un altre aparell, "WaiterServidor", encarregat de rebre els senyals enviats per wireless des dels "WaiterClients" que hi ha a cada taula. Un cop rebudes, les transmet a un ordinador central per cable sèrie RS-232.

El programa que s'utilitza per el processament de dades enviades des de els clients als cambrers, serà el "Waiter 1.0" i serà l'encarregat de mostrar l'estat de les taules i així permetre una gestió més ràpida d'aquestes.

Material lliurat:

Juntament amb la memòria s'adjunta un CD-ROM amb el codi font de les aplicacions, aquesta memòria en format digital i els arxius del programa.

ÍNDEX DE CONTINGUTS

CAPÍTOL I: INTRODUCCIÓ

1.1 MOTIVACIONS PERSONALS.....	9
1.2 OBJECTIU DEL PROJECTE.....	9
1.3 DESCRIPCIÓ DEL PROJECTE.....	10
1.4 ESTRUCTURA DE LA MEMÒRIA.....	11
1.5 METODOLOGIA DE DESENVOLUPAMENT.....	12

CAPÍTOL II: ESTUDI DE VIAVILITAT

2.1 INTRODUCCIÓ	14
2.2 DESCRIPCIÓ DE LA SITUACIÓ A TRACTAR	14
2.3 FUNCIONAMENT DEL SISTEMA.....	14
2.4 PERFIL D'USUARI.....	15
2.5 OBJECTIUS.....	15
2.6 DESCRIPCIÓ GENERAL DEL SISTEMA.....	15
2.7 ESTAT DE L'ART	17
2.7.1 SISTEMA DE CONTROL.....	17
2.7.2 SISTEMA DE POLSADORS	19
2.7.3 SISTEMA DE COMUNICACIONS	19
2.7.3.1 Comunicacions entre unitats emissores i unitat receptora.....	20
2.7.3.2 Comunicacions entre unitat receptora i ordinador	21
2.7.4 SISTEMA DE VISUALITZACIÓ	21
2.7.4.1 Visualització a les unitat emissores.....	21
2.7.4.2 Visualització a l'ordinador.....	22
2.7.5 SISTEMA DE PROCESSAMENT	22
2.8 RECURSOS	22
2.8.1 RECURSOS HUMANS	23
2.8.2 RECURSOS HARDWARE	23
2.8.3 RECURSOS SOFTWARE.....	24
2.9 PRESSUPOST	25
2.10 AVALUACIÓ DE RISCOS.....	25
2.11 PLANIFICACIÓ DEL PROJECTE.....	25
2.12 CONCLUSIONS.....	26

3.1 INTRODUCCIÓ	28
3.2 INTERCONNEXIÓ.....	29

CAPÍTOL III: DISSENY DEL SISTEMA

3.3 COMUNICACIÓ	30
3.3.1 COMUNICACIÓ ENTRE UNITATS	30
3.3.2 COMUNICACIÓ ENTRE UNITAT I ORDINADOR	32
3.4 VISUALITZACIÓ	33
3.4.1 VISUALITZACIÓ A LES UNITATS	34
3.5 CONSUM	36
CAPÍTOL IV: PROGRAMACIO DEL SISTEMA	
4.1 INTRODUCCIÓ	38
4.2 PROGRAMACIÓ DE LA UNITAT EMISSORA.....	39
4.3 PROGRAMACIÓ DE LA UNITAT EMISSORA.....	42
4.2 PROGRAMACIÓ DE LA UNITAT DE GESTIÓ	44
4.2.1 DESCRIPCIÓ DEL MENÚ PRINCIPAL.....	44
4.2.2 BOTONS ACTIUS	46
4.2.3 CONFIGURACIÓ MANUAL.....	48
4.2.4 ETAPES DE PROCÉS I CASOS D'ÚS	51
4.2.5 BASE DE DADES.....	56
CAPÍTOL V: PROVES AL SISTEMA CONFIGURAT	
5.1 INTRODUCCIÓ	58
5.2 PROVES AMB LES PLAQUES	58
5.3 PROVES ENTRE PLACA I PC.....	59
5.4 PROVES DE PROCESSAMENT	59
5.5 PROVES DE LA BASE DE DADES	60
5.6 PROVES DEL SISTEMA COMPLET	61
CAPÍTOL VI: MANUAL DEL SISTEMA	
6.1 INTRODUCCIÓ	63
6.2 MANUAL DE LA UNITAT EMISSORA I LA UNITAT RECEPTORA.....	63
6.3 MANUAL DE LA UNITAT DE GESTIÓ.....	64
CAPÍTOL VII: FUTURES AMPLIACIONS, ESTUDI DE MERCAT I CONCLUSIONS FINALS	
7.1 AMPLIACIONS.....	67
7.2 ESTUDI DE MERCAT I D'AMORTITZACIÓ	67
7.3 CONCLUSIONS	69

ÍNDIX D'IMATGES

Imatge 1 - Microprocessadors	17
Imatge 2 - Polsador	19
Imatge 3 - Mòdul de radio	20
Imatge 4 - Pantalla LCD	22
Imatge 5 - Pantalla ordinador	22
Imatge 6 - Contingut kit "PSoC"	23
Imatge 7 - Software de desenvolupament	24
Imatge 8 - Unitat emissora	28
Imatge 9 - Unitat receptora	28
Imatge 10 - Unitat de gestió	28
Imatge 11 - Mòdul de radio CYRF6936	30
Imatge 12 - Unitat receptora amb cable sèrie	32
Imatge 13 - Missatge estat	34
Imatge 14 - Missatge error	34
Imatge 15 - Missatge confirmació	34
Imatge 16 - LED	35
Imatge 17 - Missatge del software de gestió	35
Imatge 18 - Descripció del Menú Principal	44
Imatge 19 - Botons Connectar i Disconnectar	46
Imatge 20 - Missatge de diàleg de sortir del programa	46
Imatge 21 - Botons Sortir i minimitzar	47
Imatge 22 - Boto Opcions i requadre de configuració	47
Imatge 23 - Missatge de diàleg aplicar els canvis	47
Imatge 24 - Botons configuració manual ratolí	49
Imatge 25 - Botons configuració manual tàctil	50
Imatge 26 - Missatge d'esdeveniment	50
Imatge 27 - Proves visuals	58
Imatge 28 - Proves amb l'hiperterminal	59
Imatge 29 - Proves de processament	60
Imatge 30 - Proves de base de dades	60
Imatge 31 - Descripció unitat emissora/receptora	63
Imatge 32 - Descripció unitat de gestió	64
Imatge 33 - Taula esperant primer plat	64
Imatge 34 - Missatge de esdeveniment	65
Imatge 35 - Taula demana un extra	65
Imatge 36 - Taula demana el compte total	65

ÍNDIX DE FIGURES

Figura 1 - Esquema sistema PDA	10
Figura 2 - Esquema del sistema a desenvolupar	11
Figura 3 - Esquema desenvolupament en cascada	12
Figura 4 - Unitat emissora	16
Figura 5 - Unitat receptora	16
Figura 6 - Unitat de gestió	17
Figura 7 - Esquema kit "PSoC"	18
Figura 8- Esquema cable sèrie	21
Figura 9 - Diagrama de Gantt	26
Figura 10 - Esquema de interconnexió del sistema	29
Figura 11 - Esquema interconnexió "PSoC"	29
Figura 12 - Esquema ACK	32
Figura 13 - Esquema d'estats unitat emissora	39
Figura 14 - Esquema de comunicació a través del bloc SPI	41
Figura 15 - Esquema d'estats unitat receptora	42
Figura 16 - Estats dels botons actius en qualsevol moment	45
Figura 17 - Estats dels botons de configuració manual	48
Figura 18 - Esquema d'estats unitat de gestió	51
Figura 19 - Diagrama de casos d'ús	55
Figura 20 - Taula de la base de dades	56
Figura 21 - Plànol restaurant amb el sistema instal·lat	61
Figura 22 - Unitat emissora amb estructura externa	67

ÍNDIX DE TAULES

Taula 1- Comparativa de microcontroladors	18
Taula 2 - Pressupost	25
Taula 3 - Distribució de les etapes	26
Taula 4 - Estats unitat emissora	39
Taula 5 - Estats unitat receptora	43
Taula 6 - Estats unitat de gestió	52
Taula 7 - Costos plaques per peces	68
Taula 8 - Costos d'implementació del sistema	68

CAPÍTOL I

INTRODUCCIÓ

1.1 Motivacions personals

La motivació principal d'aquest projecte és la d'aprofundir en la programació de dispositius de hardware "PSoC". Des del primer cop que vaig sentir a parlar d'aquesta tecnologia em va atreure la facilitat de programació i la gran quantitat d'usos que se li poden donar.

Aquest tipus de programació no és gaire complicat si ja tens una base lògica de desenvolupament i el tipus de tecnologia que es pot fer té una gran quantitat de possibilitats en el món real.

1.2 Objectiu del projecte

L'objectiu d'aquest projecte és crear un sistema de gestió de les peticions de comandes i de les comunicacions a un restaurant. La idea és posar a cada taula, un aparell que integra uns botons i una pantalla LCD per a que els clients interactuïn amb el dispositiu. Així doncs, aquests clients podran fer peticions als cambrers per mitjà d'aquests "intèrfons" digitals. Al altre costat, els cambrers podran gestionar les taules i les seves comandes amb un software a un ordinador que va llegint els aparells disposats a cada taula a temps real.

Per això és necessària la comunicació sense fils entre els diferents hardwares de tal manera que es pugui posar un aparell a cada taula i que aquest es comuniqui amb un aparell central encarregat de rebre les senyales de les taules. A més a més, aquest últim aparell receptor, serà l'encarregat de transmetre la informació al ordinador per a que el software la gestioni. Aquesta informació s'utilitzarà per controlar l'estat de les taules i fer les diferents funcions d'un restaurant.

Així doncs, i a mode d'exemple, si un client vol una altra ampolla de vi i prem el botó de extra, aquesta informació apareixerà a la pantalla de l'ordinador en forma d'avís. Un cambrer anirà a servir al client en qüestió.

Actualment existeixen uns productes que es fan servir per a controlar els restaurants. Aquests tipus de productes es basen en hardware i software orientats a la gestió del compte i de la comanda de cada taula sense arribar a gestionar les comunicacions. El que fan és que, els cambrers porten uns dispositius PDA que serveixen per apuntar el que cada taula demana. Després aquesta informació s'envia a l'ordinador per a la gestió.

També hi ha altres tecnologies que t'avisen quan els plats estan llestos per mitja de aparells sonors i visuals però aquests aparells encara estan per desenvolupar i no són comuns al nostre país.

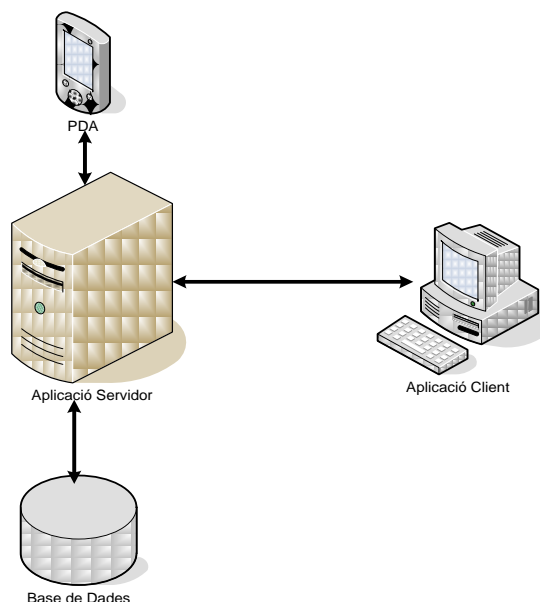


Figura 1 – Esquema sistema PDA

La versió desenvolupada en aquest projecte no està destinada a fer la competència a aquests productes, sinó més aviat a complementar-la donat que controla la comunicació i l'estat de les taules del restaurant. A més a més, la comunicació sense fils i el control per pantalla són un valor afegit per poder reduir el temps d'espera i millorar el tracte amb els clients. Com ha resultat d'això, el número de cambrers i el temps de servei es veuen reduïts. Això suposa més comoditat i rapidesa i menys despeses.

Per altra banda, aquest projecte es podria complementar perfectament amb els aparells i softwares descrits amb anterioritat per fer una automatització total de la gestió en general d'un restaurant, cosa molt interessant.

1.3 Descripció del projecte

El nostre sistema consta de dues parts: la part de hardware i la part de software. Per la part de hardware es tracta d'uns aparells disposats a cada taula d'un restaurant. Per la part del software, el restaurant tindrà un programa, anomenat Waiter 1.0, que gestionarà tota la informació de peticions dels seus clients i millorarà la gestió de les taules.

De la part de hardware es pot destacar, a nivell d'usuari, la seva estructura. Consta de 3 pulsadors. El primer serveix per demanar. Quan un client arriba, el cambrer els disposa a una taula i els dona la carta. Quan ja sap que vol demanar pot prémer el primer pulsador. Sempre que vulgui el següent plat ha de prémer el primer pulsador. Si és al segon plat i prem aquest primer pulsador, serà reconegut com que vol cafè o postra. El segon pulsador té la funció de demanar un complement extra com pot ser una altra beguda per exemple. L'últim pulsador, serveix per demanar el compte total de la taula.

Les accions anteriors són processades pel sistema que hi ha a cada taula i s'envia a un aparell preparat per a rebre la informació en forma de paquet de dades via wireless. Quan es rep un paquet de dades, la unitat encarregada de rebre els senyals de les

diferents taules, el torna a enviar, processat, i ara cap a l'ordinador a través del port sèrie via protocol RS-232, donat que la placa no té interfície USB.

El programa del ordinador és l'encarregat de rebre la informació de les taules per gestionar-la. A més a més, una pantalla (d'ordinador o de computador) ens mostrarà si una taula és al primer, al segon plat o al postra i cafè. També mostra les taules que es troben esperant, ordenades per orde de temps. Quan un client ha pres algun polsador, un avís cridarà l'atenció dels cambrers i s'actualitzarà el monitor amb l'estat d'aquella taula. Quan s'ha portat el que el client ha demanat, es valida l'estat de la taula a la base de dades.

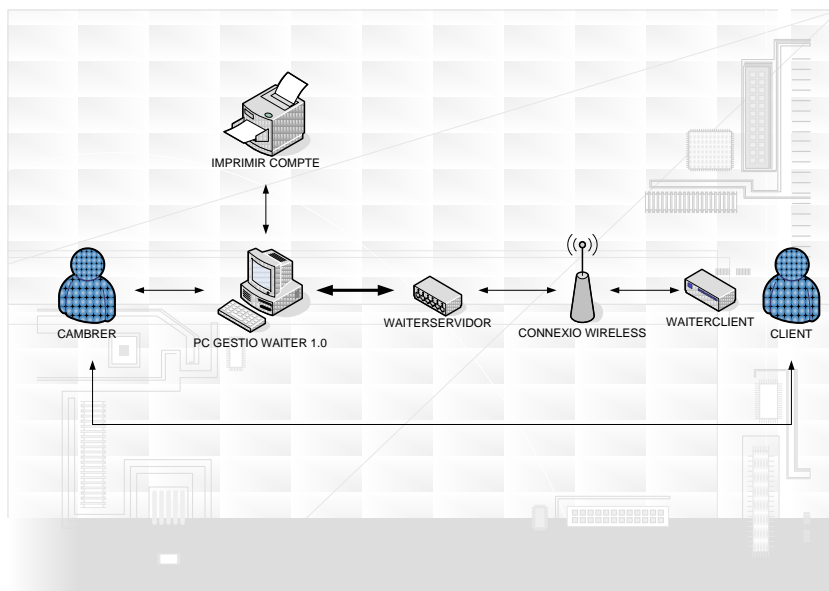


Figura 2 – Esquema del sistema a desenvolupar

1.4 Estructura de la memòria

Aquesta memòria està estructurada en 8 capítols, cadascun d'ells subdividit en apartats.

El primer capítol és una introducció on s'especifiquen els objectius principals del projecte i una descripció d'aquest. A més a més, també és descriu com s'ha desenvolupat i quines han sigut les motivacions personals per tal de dur-lo a terme.

El segon capítol es basa en l'estudi de viabilitat. Es descriu la situació en la que es basa el projecte, el funcionament del sistema final i els objectius que intenta aconseguir. La descripció general s'expressa amb detall de cada part del sistema. A continuació s'exposen els recursos necessaris i es detallen el cost i els riscos que poden esdevenir. Per finalitzar es fa un estudi de la duració i la planificació del projecte i es conclou amb la solució de si el projecte es viable o no.

El tercer capítol exposa detalladament el funcionament de cadascun dels elements que componen el sistema, especificant la interconnexió entre ells. Es fa una menció al consum, donat que el sistema que es pretén realitzar funciona amb bateries.

El quart capítol descriu la programació dels diferents mòduls que componen el sistema. Aquí es detalla la funcionalitat interna de cada programa com també la funcionalitat de cada acció possible.

El cinquè capítol explica les proves que s'han realitzat així com els resultats obtinguts.

El sisè capítol és un manual del sistema. Aquí s'explica com el usuari (ja siguin cambrers o clients del restaurant) han de fer servir els aparells per a la seva correcta utilització.

El setè capítol ens descriu les futures millores o ampliacions del sistema. També hi ha un estudi de mercat sobre el impacte del sistema al mon real. Per últim es comenta les conclusions a les que s'ha arribat després de haver realitzat el projecte.

Per finalitzar, s'afegeix un capítol amb la bibliografia que s'han fet servir.

1.5 Metodologia de desenvolupament

La metodologia de desenvolupament que s'ha seguit per a la realització d'aquest projecte ha sigut en cascada, es a dir, dividint-lo en etapes consecutives. Es comença realitzant l'estudi de viabilitat que ens determina si el projecte es pot dur a terme. A continuació es fa un anàlisi de requisits necessaris. Més tard es fa la divisió del projecte en les diferents etapes de desenvolupament seguint un cert marc d'anàlisi, disseny, codificació i proves. Cada vegada que una etapa es finalitzada es torna a començar el cicle. Quan totes les etapes han sigut finalitzades es comença a fer la unió del bloc i s'inicia la documentació ha realitzar.

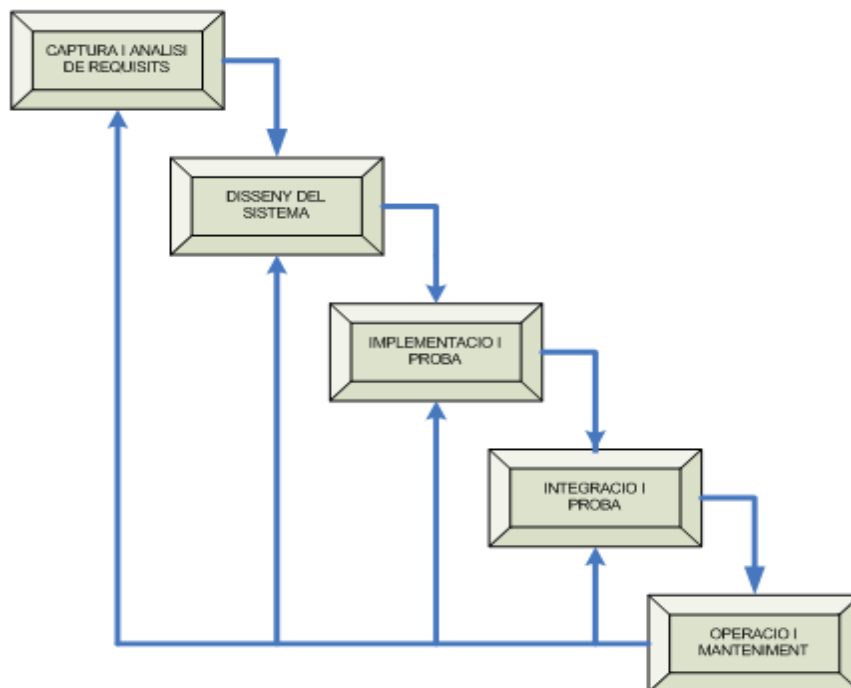


Figura 3 - Esquema desenvolupament en cascada

CAPÍTOL II

ESTUDI DE VIABILITAT

2.1 Introducció

L'estudi de viabilitat ha de servir per determinar si aquest projecte és realitzable o no. Això en el cas d'una empresa es extremadament important donat que determinarà si es segueix endavant amb les següents etapes o es desestima la idea. Un mal estudi de viabilitat pot suposar grans pèrdues econòmiques.

Inicialment es fa una descripció de la situació a tractar, explicant el funcionament esperat del sistema i a quin tipus d'usuari va dirigit. A continuació s'exposen els objectius que es pretén aconseguir amb aquest projecte. Seguidament es dona una visió general del sistema i es fa una exposició de l'estat de l'art dels components necessaris. En aquest estudi s'exposen les diferents alternatives que existeixen per a cada subsistema, quina s'ha escollit i perquè.

Després d'això s'enumeren i descriuen els recursos necessaris per a la creació del sistema i s'elabora un pressupost que pugui ajudar a determinar la viabilitat del projecte.

Finalment s'avaluen els riscos associats a aquest i es descriu la planificació que s'haurà de seguir. Aquesta planificació consisteix en definir les etapes en les que es dividirà el projecte que s'hauran d'anar acomplint per finalitzar el projecte en els terminis establerts.

2.2 Descripció de la situació a tractar

L'estudi de viabilitat següent tracta de crear un sistema que permeti visualitzar, per una part, l'estat de les taules d'un restaurant, i per l'altre millorar el servei d'atenció al client. Per això és necessari un programa senzill d'utilitzar però potent i uns aparells útils però petits que siguin autònoms.

Per fer la comunicació entre les taules i el sistema, aquesta ha de ser sense fils, de tal forma que cada client es pugi comunicar de manera independent amb el servei. Aquesta comunicació ha de minimitzar els possibles errors de comunicació alhora que permeti una futura ampliació.

2.3 Funcionament del sistema

El funcionament normal del sistema no serà gaire complicat. Per la part del hardware dels clients, l'aparell mostrarà un missatge amb el número de taula i restarà a la espera de que s'actuï un dels 3 polsadors. Les accions dels polsadors són la de demanar un extra, demanar el següent plat o demanar el compte total de la taula.

Per part de la gestió del restaurant, un aparell autònom serà l'encarregat de rebre la informació sense necessitat de interactuar amb ell. El programa encarregat de mostrar la informació serà senzill d'utilitzar. Es tractarà d'un programa on totes les opcions són visibles i amb el nom de la funció que fan.

Per a més informació d'ús es pot consultar el manual del sistema inclòs en aquesta memòria.

2.4 Perfil d'usuari

El perfil d'usuari del sistema es divideix en dos. Per una part els cambrers que hauran de saber fer servir tot el sistema i per altra part els clients del restaurant que són usuaris de caire general. Els clients rebran informació per part dels cambrers sobre l'ús del sistema encara que és molt intuïtiu i fàcil de controlar.

Qualsevol persona mínimament experimentada amb equips electrònics, com poden ser televisor, mini cadenes, ordinadors, entre altres, ha de poder fer el manteniment. Per això s'ha de dissenyar de tal manera que sigui senzill i en cas de errors imprevistos, corregir-los eficaçment.

2.5 Objectius

L'objectiu principal d'aquest projecte és reduir el temps de servei i d'ocupació de les taules en un restaurant. Per aconseguir això es crea un sistema que sigui capaç de comunicar les peticions dels clients i transmetre-les a un ordinador per mostrar-les per pantalla i cridar així l'atenció dels cambrers.

A més, el sistema ha de complir aquests altres objectius generals:

- Intuïtiu: s'ha de poder utilitzar sense complicacions.
- Barats: el preu final ha de ser econòmic i per això els components no poden ser cars.
- Bonic: ha de quedar bé al restaurant.
- Autònom: les unitats de les taules han de ser independents de la corrent elèctrica.

2.6 Descripció general del sistema

El sistema que es pretén crear consta de tres parts. Les unitats de captura són les que reben els valors dels pulsadors, processen aquesta informació i la visualitzen per pantalla a més d'enviar-les a la unitat receptora, que rebrà els valors i serà l'encarregada de transmetre els valors rebuts a un ordinador per gestionar-los amb un programa, que serà la unitat de gestió i així poder visualitzar-los per pantalla.

Aquests sistemes tindran un **sistema de control** i un sistema de processament de dades cada un, que s'encarregarà de fer els passos adients per al bon funcionament del sistema.

Vist amb detall, cada un d'aquests sistemes es divideix en subsistemes encarregats de realitzar una tasca concreta.

Així doncs, la unitat de captura de valors es descompon en:

- **Sistema de visualització** per mostrar informació al LCD.
- **Sistema de polsadors** que permeten capturar les accions per tractar-les.
- **Sistema de comunicació** que és l'encarregat de transmetre els valors.
- **Sistema de processament** per tal de processar els valors en informació i adequar-la per enviar o visualitzar.

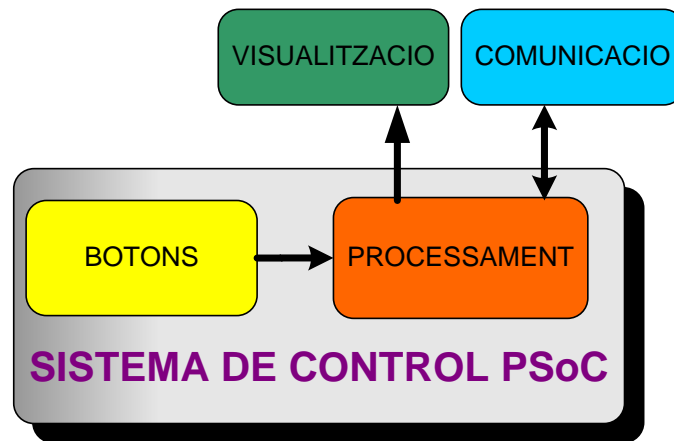


Figura 4 – Unitat emissora

La unitat receptora de valors, unitat receptora, constarà de:

- **Sistema de comunicació** que és l'encarregada de rebre els valors i transmetre'ls al ordinador. Encara que les comunicacions són per wireless i per cable sèrie, és un sol sistema l'encarregat de la comunicació.
- **Sistema de processament** per tal de processar els valors rebuts per a tornar-los a enviar per un altre canal.

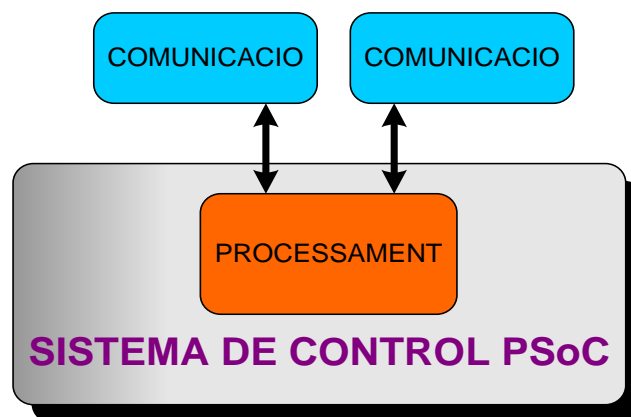


Figura 5 – Unitat receptora

La unitat de gestió que és controlada per software a un PC, constarà de:

- **Sistema de visualització** per mostrar informació per el monitor.
- **Sistema de processament** per tal de processar les dades rebudes i gestionar-les adequadament.
- **Sistema de comunicació** encarregat de rebre i transmetre els valors.

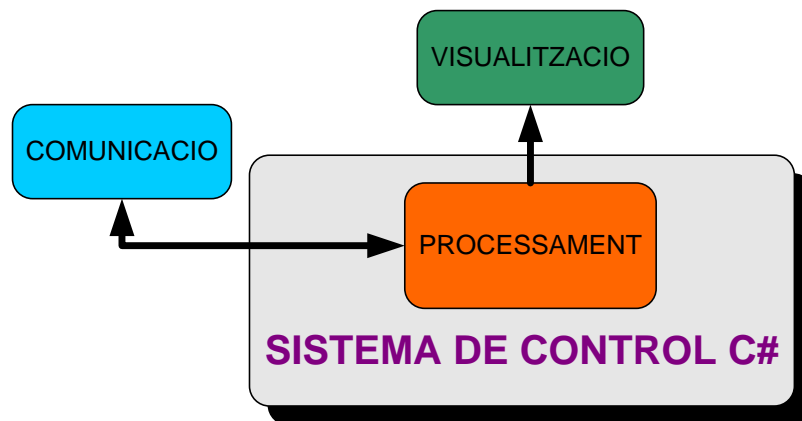


Figura 6 – Unitat de gestió

2.7 Estat de l'art

En aquests apartat s'exposa l'estat de l'art de cadascun dels 5 subsistemes. S'ha de dir que encara que aquestos sistemes necessitaran uns elements concrets per a la seva tasca, han de ser capaços de comunicar-se entre ells, i per tant, han de tenir unes característiques en comú.

2.7.1 SISTEMA DE CONTROL

Per a controlar el sistema s'ha escollit utilitzar un microcontrolador. Aquestos són sistemes complets, integrats en un sol xip i dissenyats per fer una tasca concreta. Per altra banda, aquest tipus de tecnologia permet tornar a programar, i això vols dir que es pot modificar si la tasca que controla varia.



Imatge 1 – Microprocessadors

Per aquest projecte es necessita un microcontrolador que tingui un canal de sortida per la comunicació, i un canal d'entrada per rebre els senyals dels botons. No serà necessari gran velocitat ja que les dades no tenen una variació molt ràpida en el temps. Tampoc farà falta gaire memòria ja que les dades s'enregistraran a una base de dades.

A part, també serà necessari buscar un sistema que permeti transmetre les dades entre els dispositius a través del aire. El cost també serà un aspecte a tenir en compte.

Buscant entre els diferents fabricants s'ha pogut observar que aquestos disposen d'unes plaques d'avaluació, amb característiques similars, que contenen tot el necessari per tal de realitzar un projecte senzill. Això simplifica molt el disseny del sistema ja que tenen els components inclosos dins del kit d'avaluació. La taula següent mostra els diferents kits amb les seves característiques:

	memòria	velocitat	E/S	simplicitat	sense fils	preu
Motora HCS12	1 k	normal	16	baixa	No	348 €
Cypress "PSoC"	8 k	normal	24	alta	Si	165 €
Parallax Propeller	32 k	normal	32	alta	No	89 €

Taula 1- Comparativa de microcontroladors

Encara que més o menys tots compleixen els requisits, s'ha escollit el "PSoC" (Programmable System on Chip) de Cypress, **model CY7C60323** que està integrat al kit **CY3630** ja que és l'únic que té connexió sense fils i que permet utilitzar el protocol WirelessUSB. Aquest disposa d'un sistema que integra un microcontrolador compatible amb els requisits necessaris, de la mateixa manera que es pot dissenyar, programar i provar.

Internament es divideix en set sistemes:

- El microprocessador de 8 bits.
- La memòria RAM.
- La memòria Flash.
- El sistema d'entrada/sortida.
- El sistema de comunicacions.
- El sistema digital.
- El sistema analògic.

Aquests dos últims estan preparats per contenir els components programats per fer una funció determinada, com pot ser les comunicacions per wireless o amplificadors, entre d'altres.

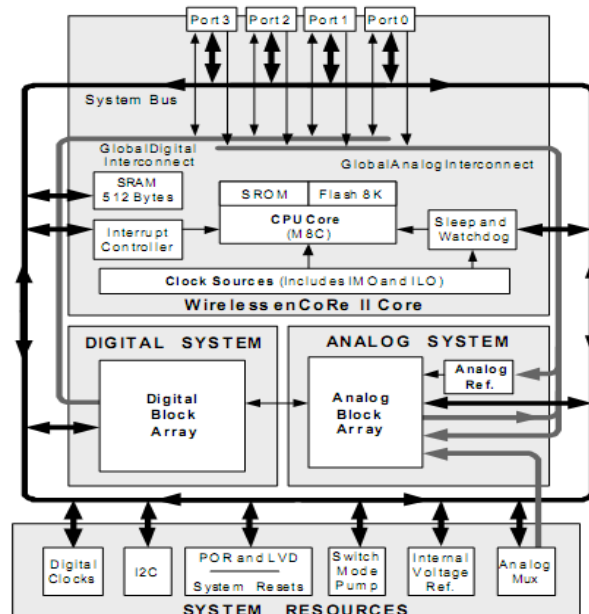
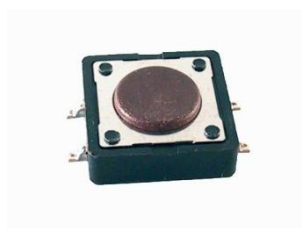


Figura 7 – Esquema kit "PSoC"

El microprocessador central, anomenat M8C, de 8 bits, té una arquitectura “Harvard” on les instruccions i les dades s’emmagatzemen en memòries físicament separades. Opera a velocitats de rellotge de fins a 24MHz i dona un rendiment màxim de 4 MIPS (Milions d’Instruccions Per Segon).

2.7.2 SISTEMA DE POLSADORS

Des dels inicis dels temps, l’home ha intentat conèixer el seu entorn i interactuar amb ell per tal d’adaptar-se. Al llarg dels anys, gracies a la tecnologia, ha anat desenvolupant aparells per fer-ho, cada vegada més automatitzats. Un dispositiu per a controlar aquests aparells han sigut els polsadors.



Imatge 2 - Polsador

Són elements de comandament, els contactes dels quals, només tenen una posició estable (posició de repòs). Aquests elements es caracteritzen perquè quan deixa d’actuar la pressió que provoca el canvi d’estat del contacte, aquest automàticament torna a la seva posició de repòs, és a dir, només executa un dels estats mentre es prem.

Segons la posició del contacte en la seva posició de repòs, podem parlar de contactes normalment oberts (NO) o contactes normalment tancats (NT):

- Normalment obert (NO). Connecta el circuit quan es prem. El polsador normalment obert, en el moment que es prem, fa contacte amb els dos borns, de manera que el corrent pot circular, mentre que, quan es deixa anar, es perd el contacte i queda tallat el pas del corrent.
- Normalment tancat (NT). Desconnecta el circuit quan es prem. És a dir, el polsador normalment tancat funciona a l’inrevés que l’anterior.

El kit del “PSoC” porta ja integrat dos polsadors del tipus Normalment obert. També porta un altre programat amb la funció de reiniciar del tipus Normalment tancat.

2.7.3 SISTEMA DE COMUNICACIONS

El sistema de comunicacions d’aquest projecte és l’encarregat de transmetre els diferents paquets de dades amb la informació per a ser processada. Aquest sistema es divideix en dues parts:

2.7.3.1 Comunicacions entre unitats emissores i unitat receptora

Un dels objectius d'aquest projecte és que les unitats emissores es comuniquin sense fils amb la unitat receptora. Per tant, no és possible fer servir un sistema de comunicacions per cable elèctric. Una opció seria mitjançant infrarojos però això suposaria que els dispositius haurien d'estar un a la vista de l'altre. Això seria un impediment molt gran ja que hi hauran molts obstacles entre les unitats com poden ser, parets, portes, cortines, etc. Així doncs, descartant aquest tipus de comunicació tan sols queda un sistema: la radiofreqüència.

El sistema de control que s'ha triat inclou un sistema de comunicacions sense fils Wireless-USB. Aquest neix de la premissa que no es necessita un protocol pesat i complex per a comunicacions punt a punt i, multipunt a punt, com ara teclats o ratolins.



Imatge 3 – Mòdul de radio

Opera a una freqüència oberta de 2.4 GHz i es pot fer separació en 79 canals d'1 MHz d'amplada per tal de evitar les interferències. També es fa fer servir l'expansió de l'espectre per seqüència directa, que és una tècnica on cada bit de dades es codifica com una seqüència d'uns i zeros (anomenada codi Pseudo Noise, PN) coneguda per l'emissor i per el receptor. El sistema Wireless-USB permet seleccionar diferents modes de comunicació de dades depenent de les necessitats de cada cas. Depenent del número de bits codificats a cada símbol i del número d'aquests s'aconseguirà més o menys velocitat i seguretat.

Per assegurar la integritat de les dades s'afegeix a cada paquet transmès un codi de 16 bits de comprovació de redundància cíclica. A més, la llavor per generar aquest codi es diferent per a cada parella emissor/receptor de tal manera que s'evita que un paquet sigui confirmat per un aparell que faci servir el mateix canal i codi PN. Per altra banda, quan el receptor rep un paquet correctament envia automàticament un codi de confirmació a l'emissor. Si aquest últim no rep aquest codi sabrà que el paquet no s'ha rebut correctament i el podrà reenviar.

En resum, s'utilitzarà el Wireless-USB ja que el sistema de control que s'ha triat inclou aquest sistema de comunicació i per tant no serà necessari afegir cap component. Aquest permet comunicacions amb distàncies de 50 metres i té un consum de corrent molt petit, que permet la seva utilització amb bateries.

2.7.3.2 Comunicacions entre unitat receptora i ordinador

Tots els sistemes informàtics disposen de comunicacions amb l'exterior, ja sigui d'entrada o de sortida. Aquesta es realitza a través del que s'anomena ports. Els ordinadors estan equipats amb varis ports que els permeten operar i comunicar-se amb dispositius perifèrics. Alguns d'aquests dispositius són crucials per l'usuari per poder utilitzar l'ordinador, com poden ser la pantalla o el teclat. La finalitat de qualsevol port és la de transferir informació, ja sigui en un o dos sentits.

Com que el sistema de control triat ja incorpora un port sèrie amb l'estàndard RS-232 i la majoria d'ordinadors en tenen (o si no segur que tenen USB), s'ha triat aquesta forma de comunicació entre l'unitat receptora i l'ordinador. Físicament es farà servir un cable amb un connector de 9 pins a cada extrem, essent en un costat del tipus mascle i en l'altre de tipus femella. Aquest sistema permet una comunicació molt senzilla i ràpida.

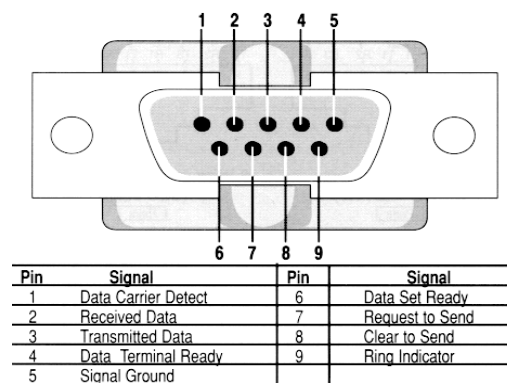


Figura 8- Esquema cable sèrie

Tècnicament funciona enviant els bits com a nivell de voltatge. Un 1 lògic es representa com voltatge negatiu mentre que un 0 lògic com a positiu. La informació s'envia en grups entre 5 i 8 bits, a més dels de sincronització i control d'errors. Aquests bits s'envien a una freqüència predefinida, anomenada índex de baud (Baud Rate).

2.7.4 SISTEMA DE VISUALITZACIÓ

Un dels objectius del projecte és que la informació es mostri a dos llocs: a cada unitat emissora i a l'ordinador. Així doncs cal especificar el sistema de visualització cada cas.

2.7.4.1 Visualització a les unitat emissores

Ja que el kit proporciona una pantalla LCD i que permet mostrar informació alfanumèrica, a més de funcionar amb bateries, s'utilitzarà aquest sistema conegut com pantalla de matrius de punts. Aquestes pantalles funcionen amb cristall líquid i venen acompanyades de tot el necessari per al seu funcionament, a més de permetre visualitzar dues línies de 16 caràcters.



Imatge 4 – Pantalla LCD

2.7.4.2 Visualització a l'ordinador

El sistema de visualització a la pantalla de l'ordinador ha de ser un sistema informàtic. Els requeriments necessaris per tal de visualitzar aquesta informació no són gens elevats donat que el programa simplement haurà de llegir la base de dades amb els valor rebuts. Qualsevol ordinador ha de ser capaç de fer aquesta tasca. De totes formes s'utilitzarà un ordinador actual per tal de visualitzar la informació.

Es selecciona un PC amb Windows donat que és un bon compromís entre preu i facilitat de programació i de gestió. D'aquesta manera l'aplicació de visualització serà suficientment ràpida de programar i donarà bons resultats.



Imatge 5 – Pantalla ordinador

2.7.5 SISTEMA DE PROCESSAMENT

El sistema de processament serà l'encarregat de convertir les dades que es reben a l'escala a la que hi haurà d'estar relacionada. Aquesta transformació la realitza un microprocessador que, executant un programa, realitza les operacions necessàries per transformar aquests valors. Aquestes són generalment operacions matemàtiques senzilles.

El sistema de control que s'ha triat ja incorpora els elements necessaris per fer aquest processament.

2.8 Recursos

Per poder realitzar el projecte seran necessaris un seguit de recursos, classificables en:

- Recursos humans
- Recursos hardware
- Recursos software

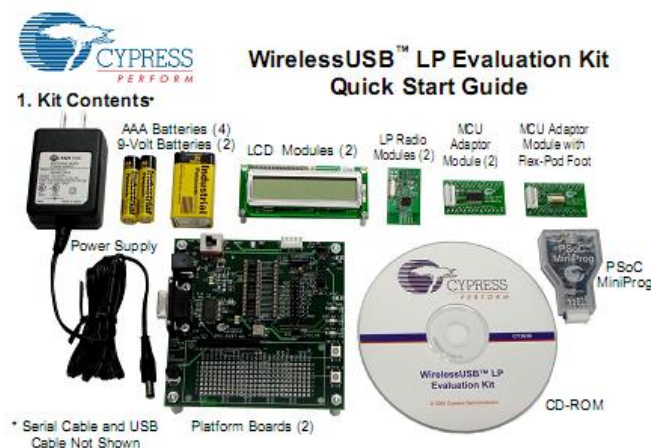
2.8.1 RECURSOS HUMANS

Ja que es tracta d'un projecte final de carrera, els únics recursos humans necessaris són l'alumne i el director de projecte. L'alumne serà l'encarregat de l'estudi, el disseny, la construcció, les proves i la creació de la memòria. El director serà l'encarregat de guiar a l'alumne per poder fer la seva tasca.

2.8.2 RECURSOS HARDWARE

En quant a recursos hardware, tot el necessari ve inclòs en el kit d'avaluació que s'ha escollit (Cypress CY3630). Aquest kit inclou:

- Un parell de microcontroladors CY7C60323-PVXC que seran els encarregats de gestionar el funcionament del sistema.
- Un parell de mòduls de radio CYRF6936 que s'encarregaran de les comunicacions sense fils.
- Dues plaques de circuits integrats amb tot el necessari per a la programació i comunicació dels sistemes.
- Un visor LCD per mostrar informació.
- Dos mòduls de programació per a descarregar el programa als microcontroladors.
- Cables USB per a la connexió dels mòduls amb l'ordinador.
- Piles per al funcionament.



Imatge 6 – Contingut kit “PSoC”

Apart, també serà necessari el següent:

- Cable sèrie per a la comunicació de la unitat receptora amb l'ordinador.
- PC amb Windows amb un port sèrie per les comunicacions i port USB per a la programació.

2.8.3 RECURSOS SOFTWARE

En quant a software o programes necessaris per poder dur a terme aquest projecte, es poden dividir en tres grups: la programació dels microcontroladors, la programació de l'aplicació de gestió i la documentació. Per dur a terme això seran necessaris els següents programes:

- PSoC Programmer
- PSoC Designer
- Visual Estudio C# 2008 Express Edition
- Paquet Office



Imatge 7 – Software de desenvolupament

2.9 Pressupost

Es important realitzar un anàlisi del cost que tindrà el projecte. Aquí s'ha de tenir en compte tant el cost de disseny com el cost del sistema final. Es realitza fent un pressupost on es detalla el cost dels elements. Aquest pressupost és professional.

Nom	Hores	Cost
Kit Cypress per peces		165,00 €
Cable Sèrie RS232		2,50 €
PC Windows		350,00 €
Disseny i implementació (15 €/h)	320 h	4.800,00 €
TOTAL		5.317,50 €

Taula 2 - Pressupost

Com que es tracta d'un projecte de final de carrera, i en aquest cas concret, el cost total és 0 € ja que els materials són adquirits per l'escola i el disseny i la implementació la porten a terme el professor i els alumnes. Així doncs el preu no suposa un risc per el desenvolupament del projecte.

2.10 Avaluació de riscos

Els principals riscos a l'hora de desenvolupar el projecte són:

- Test incorrecte: deficiència operativa.
- Endarreriment de la finalització del projecte i baixa qualitat.
- Equip reduït: endarreriment de la finalització del projecte.
- Eines inadequades de implementació: endarreriment de la finalització del projecte.
- Incompliment legislació: repercussions legals.
- Abandonament del projecte: pèrdues econòmiques.

2.11 Planificació del projecte

El projecte s'ha dividit en 4 parts principals: estudi de viabilitat, disseny del sistema, programació del sistema i documentació. A cada etapa s'ha marcat una fita per tal de poder passar a la següent.

La durada de cada etapa es pot veure a la taula següent:

Tasca	Inici	Duració	Final
Planificació	01/02/10	10 hores	02/02/10
Estudi viabilitat	02/02/10	35 hores	08/02/10
Disseny de l'aplicació	08/02/10	45 dies	12/04/10
Disseny de les plaques	08/02/10	30 dies	22/03/10
Disseny del software	22/03/10	15 dies	12/04/10
Programació de l'aplicació	12/04/10	9,13 dies	23/04/10
Programació de les plaques	12/04/10	5 dies	19/04/10
Programació del software	20/04/10	30 hores	23/04/10
Test i proves	26/04/10	15 hores	27/04/10
Generació documents	28/04/10	50 hores	06/05/10

Taula 3 – Distribució de les etapes

Podem observar que el projecte comença al febrer i que termina aproximadament al maig. Durant el juny hi ha una fase de rectificació general fins a l'entrega i l'exposició.

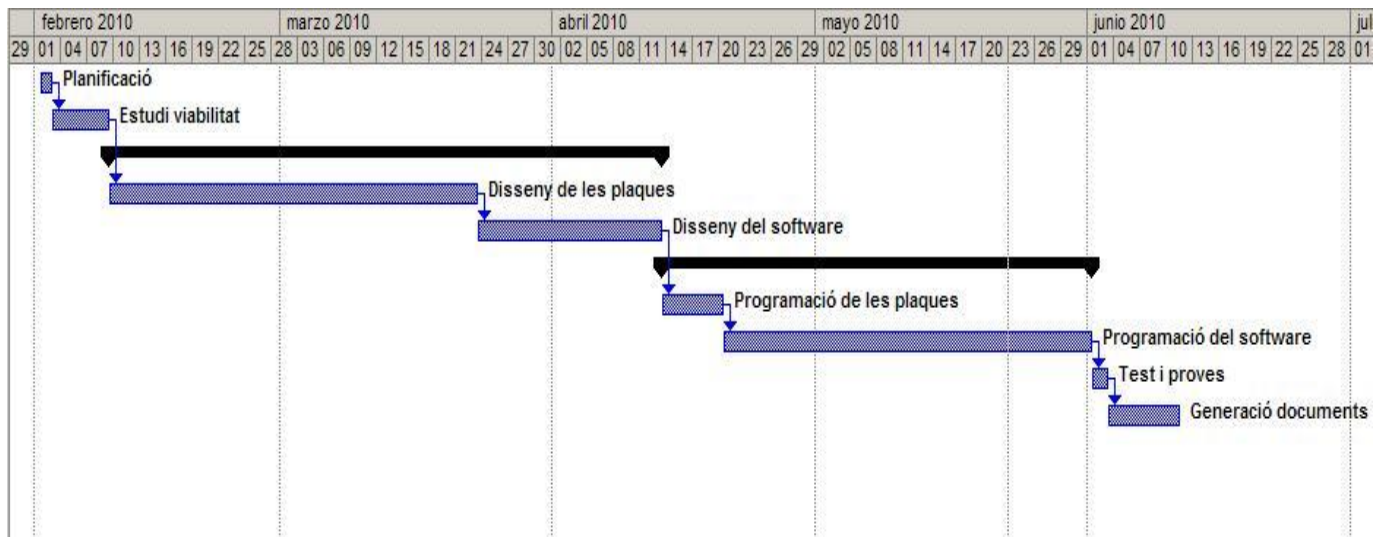


Figura 9 – Diagrama de Gantt

2.12 Conclusions

Un cop realitzat l'estudi de viabilitat del projecte es pot decidir la seva viabilitat. Si es conclou que és viable es durà a terme, i en cas que no ho sigui s'hauran de buscar alternatives i s'haurà d'iniciar un de nou.

Ja que els costos que suposa la realització d'aquest projecte són assumibles i que els riscos que es pot topar en el desenvolupament del projecte són salvables, podem afirmar que el projecte serà viable.

Per tant, es procedeix a la seva realització.

CAPÍTOL III

DISSENY DEL SISTEMA

3.1 Introducció

Com ja s'ha explicat, el sistema es divideix en tres unitats: unitat emissora, unitat receptora i unitat de gestió. La primera unitat conté els pulsadors, el sistema de visualització, el de processament i el de comunicacions. La segona unitat conté dos sistemes de comunicacions: un que comunica amb la primera unitat per wireless i un altre que ho fa amb l'ordinador per cable sèrie i un sistema de processament. L'última unitat té el sistema de comunicacions per rebre les dades, el sistema de processament i el sistema de visualització.

En aquest capítol es farà una descripció de com són les unitats emissores, la unitat receptora i la unitat de gestió, internament. Després s'explicarà detalladament el funcionament de cadascuna.



Imatge 8 – Unitat emissora



Imatge 9 – Unitat receptora



Imatge 10 – Unitat de gestió

3.2 Interconnexió

El següent esquema mostra la connexió entre les tres unitats:

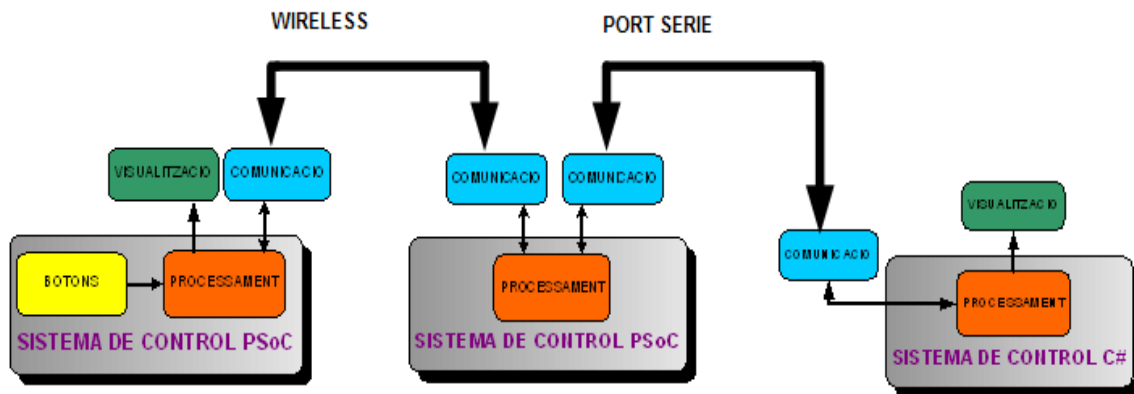


Figura 10 – Esquema de interconnexió del sistema

Pel que fa la unitat emissora, l'esquema ens mostra els polsadors que estan connectats al sistema de processament. Aquest modifica les dades perquè es puguin mostrar per la pantalla LCD amb el sistema de visualització i perquè es puguin enviar amb el sistema de comunicació a la unitat receptora a través de Wireless-USB.

La unitat receptora és més simple. El sistema de comunicació rep les dades per Wireless-USB, on el sistema de processament les transforma per enviar-les per el sistema de comunicació del port sèrie (RS-232) a la unitat de gestió.

Aquesta última unitat rep les dades a través del sistema de comunicació per port sèrie (RS-232). Les transforma amb el sistema de processament i les mostra per la pantalla del ordinador amb el sistema de visualització.

La connexió física dels sistemes interns del xip "PSoC" es realitza a l'hora de dissenyar-lo, amb l'eina PSoC Designer. Aquesta permet incorporar blocs que realitzen les diferents funcions necessàries.

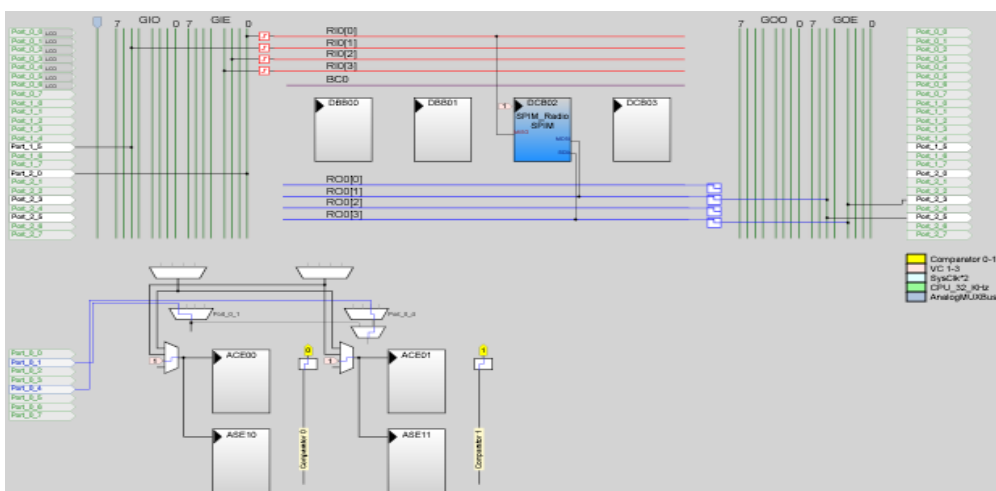


Figura 11 – Esquema interconnexió "PSoC"

En la imatge anterior es mostra la programació de blocs de la unitat emissora. Podem observar el controlador LCD a la esquerra de la imatge distribuït des de el port_0_0 fins al port_0_4. El controlador SPI (connexió wireless) es troba localitzat al centre en color blau.

Pel que fa a la unitat receptora, la configuració seria totalment idèntica exceptuant el controlador LCD.

3.3 Comunicació

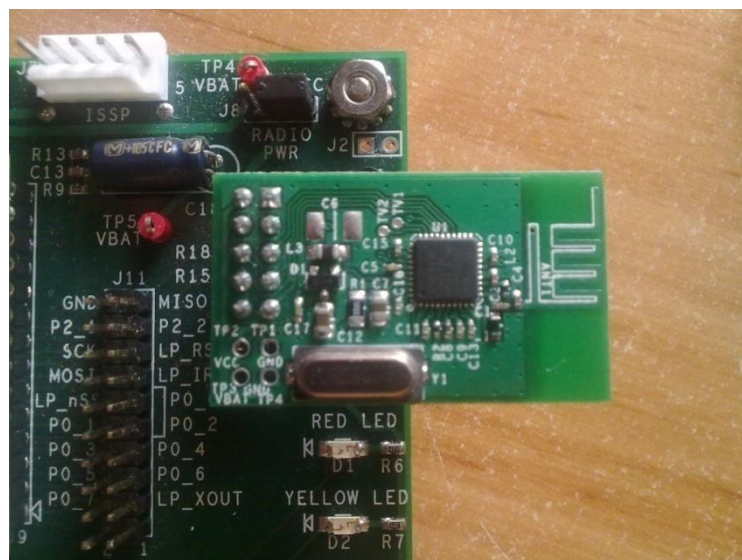
L'enviament d'informació es divideix en dues parts. D'una banda l'enviament entre les dues plaques que es farà sense fils, i per l'altra l'enviament des de la unitat receptora a l'ordinador mitjançant un cable.

Cada part fa servir un sistema diferent: Wireless-USB en la primera i cable sèrie RS-232 en la segona. En aquesta apartat s'explica el funcionament de cada sistema per separat.

3.3.1 COMUNICACIÓ ENTRE UNITATS

Un cop les dades dels polsadors han sigut processades, aquestes han de ser enviades a la unitat receptora per tal que les enviï al ordinador. Per fer-ho, es farà servir el protocol de comunicacions sense fils Wireless-USB.

Físicament es tracta d'un petit microcontrolador (CYRF6936) encarregat de realitzar la transformació d'impulsos elèctrics en ones radioelèctriques (o al revés) que s'envien o es reben a través de l'antena. Està integrat en una petita placa juntament amb l'antena i els connectors per comunicar-se amb el microcontrolador principal.



Imatge 11 – Mòdul de radio CYRF6936

Aquesta comunicació es fa a través d'un sistema estàndard anomenat SPI (Serial Peripheral Interface Bus) que permet la comunicació sèrie entre un dispositiu mestre i un o més esclaus (placa "PSoC" i mòdul de radio). Per fer la comunicació es necessiten quatre canals independents (i per consegüent, quatre pins en cada microcontrolador). El mestre és qui controla el rellotge per sincronitzar les comunicacions a través de la sortida SCK. El senyal SS (Slave Select) serveix per seleccionar amb quin esclau es vols comunicar. Finalment les comunicacions es realitzen a través de dos pins: MOSI (Master In, Slave Out) per enviar dades del mestre a l'esclau i MISO (Master In, Slave out) per enviar dades en sentit contrari. Això permet comunicacions Full Dúplex: es pot enviar i rebre informació simultàniament.

En quant a funcionament, permet diferents modes que variaran la velocitat de la informació enviada. Aquesta pot oscil·lar entre 15,6Kbps fins a 1Mbps. Aquesta velocitat és inversament proporcional a la qualitat de la comunicació. En aquest projecte la velocitat no és un element important.

Com ja s'ha explicat amb anterioritat, s'utilitza el protocol WirelessUSB. El propòsit de les especificacions és preservar el model funcional de USB, desenvolupat en torn a amfitrions (hosts) amb intel·ligència i dispositius senzills. Al mateix temps, ha de ser capaç de funcionar en entorns sense fils i mantenir els nivells de seguretat que proporcionaria un sistema cablejat equivalent. També pretén preservar els nivells de consum del sistema amb cables. Per aconseguir això, s'utilitza un estàndard ja existent que defineixi unes capes de base adequades: un nivell físic i un control d'accés al medi, que és capaç d'obtenir un rendiment com el mencionat. A això s'adhereix un nivell de convergència que unifica ambdós esforços de especificació.

WirelessUSB es defineix com un bus, encara que un bus lògic i no físic. Aquest pot connectar un amfitrió amb un cert nombre de perifèrics. L'amfitrió divideix l'ample de banda disponible entre els dispositius per mitjà d'un mètode de divisió de temps (time-division multiple access). Segueix preservant el model de manipulació segura de dispositius al vol, característica de USB, que se expandeix a una distància de 10 metres.

Per controlar els errors de comunicació s'ha utilitzat un element important: la transacció automàtica que ens indica si el missatge s'ha rebut correctament o no. Així doncs quant la unitat emissora envia un missatge a la unitat receptora, aquesta ha de retornar un missatge de confirmació (ACK). Si no ho envia, la unitat emissora sabrà que no s'ha enviat correctament i tornarà a intentar-ho fins a tres cops. Si en aquest cas no s'ha rebut la confirmació, el missatge a enviar es descarta i s'informa d'error per pantalla i amb els leds. D'aquesta manera es controla si la comunicació entre les diferents unitats és correcta o errònia i així mostrar-lo al usuari.

Pel que fa als tipus de missatges que s'envien, aquests tenen la següent estructura: {'01P'}. Els dos primers números fan referència a la taula que fa la petició. El tercer caràcter pot ser una P, E o C segons la petició sigui següent Plat, un Extra o Compte total. Així doncs, si una unitat emissora envia {'12E'}, s'interpreta com que la taula 12 està demanat un extra.

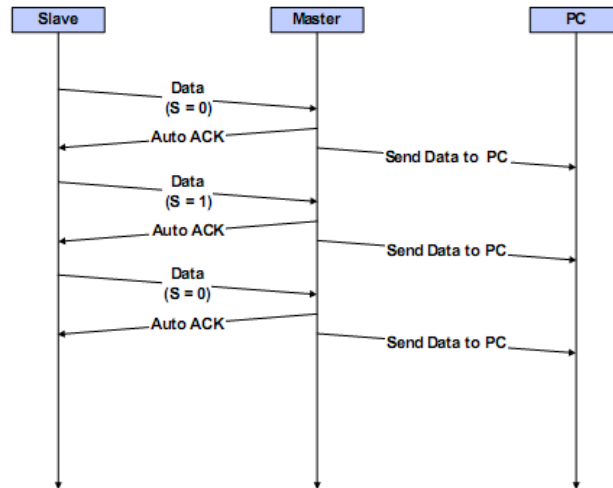


Figura 12 – Esquema ACK

3.3.2 COMUNICACIÓ ENTRE UNITAT I ORDINADOR

Un cop rebuda la informació a la unitat receptora, aquesta s'ha d'enviar a l'ordinador per tal de mostrar-la per pantalla. Això es fa mitjançant un cable sèrie amb el protocol RS-232.



Imatge 12 – Unitat receptora amb cable sèrie

L'ordinador controla el port sèrie a través d'un circuit integrat específic, anomenat UART (Transmissor – Receptor - Asíncron Universal).

Per controlar el port sèrie, la CPU utilitza direccions de port de E/S i línies de interrupció (IRQ). Cada usuari ha de escollir d'acord a les que té lliures o dependent de l'ús que es vagi a fer del port sèrie ja que les IRQ no estan especificades.

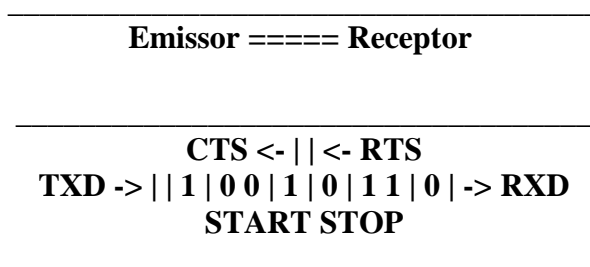
A través dels ports de E/S es poden intercanviar dades, mentre que les IRQ produeixen una interrupció per indicar a la CPU que hi ha hagut un esdeveniment. La CPU ha de respondrà a aquestes interrupcions el més ràpid possible i així agafar la dada abans que la següent dada la sobreescrigui.

El RS-232 pot transmetre les dades en grups de 5,6,7 o 8 bits, a unes velocitats determinades (normalment de 9600 bits per segon). Després de la transmissió de les dades, segueix un bit opcional de paritat (indica si el numero de bits transmesos és parell o imparell, per a detectar errors) i després 1 o 2 bits de STOP.

Una vegada ha començat la transmissió de una dada, els bits han d'arribar un rere l'altre a una velocitat constant i en determinats instants de temps. Per això es diu que RS-232 és asíncron per caràcter i asíncron per bit. Els pins que porten les dades són RXD i TXD. La resta s'encarrega de altres treballs: DTR indica que l'ordinador està encès, DSR que l'aparell connectat està encès, RTS que l'ordinador pot rebre dades (no està ocupat), CTS que l'aparell connectat pot rebre dades i DCD detecta que existeix una comunicació.

L'aparell a connectar amb l'ordinador ha d'utilitzar el mateix protocol sèrie per a permetre la comunicació. Donat que l'estàndard RS-232 no permet indicar en que mode està treballat, és l'usuari qui té que configurar-lo.

Aquest és el diagrama de transmissió d'una dada amb format 8N1. El receptor indica al emissor que pot enviar dades activant la sortida RTS. L'emissor envia un bit START (nivell alt) abans de les dades, i un bit de STOP (nivell baix) al final d'aquestos.



Com que la comunicació entre la unitat receptora i la unitat de gestió es fa per el port sèrie amb el protocol RS-232 es tractarà d'una comunicació senzilla. Així doncs la informació s'enviarà amb el mateix format amb que s'ha rebut, sense modificar-la.

3.4 Visualització

En aquest apartat es descriu els dos tipus de sistemes de visualització del sistema: per una part en una pantalla situada a cada unitat emissora i per l'altra a l'ordinador. Apart, cal comentar els tres tipus de leds que incorpora que també ajuden a entendre millor que està fent el sistema en certs moments.

3.4.1 VISUALITZACIÓ A LES UNITATS

Per mostrar la informació a cada unitat s'han fet servir les pantalles LCD que inclou el kit de Cypress. Aquestes pantalles permeten visualitzar 2 línies d'informació en 16 columnes. Això dona espai suficient per mostrar els missatges necessaris.

La informació que surt per pantalla és pràcticament igual a totes les unitats. La única diferència és que varia el numero de taula que apareix per pantalla.



Imatge 13 – Missatge estat

La primera fila sempre és estàtica amb el nom de l'aplicació i indicant el projecte: PFC Waiter 1.0. La segona fila serveix per mostrar els diferents missatges d'accions. Si la unitat està processant, apareix una barra de progrés amb el temps que trigarà. Segons l'acció pulsada, apareixerà un o altre missatge referent a l'acció que s'està demanant.

Si la comunicació falla amb la unitat de recepció, un missatge ens avisa que no s'ha pogut enviar l'acció i el led vermell s'encén, gràcies al control d'errors.



Imatge 14 – Missatge error

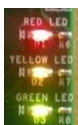
Si la comunicació amb la unitat de recepció és correcta i s'ha pogut enviar el senyal sense cap problema, apareix un missatge de confirmació.



Imatge 15 – Missatge confirmació

En quant als leds, hi ha de tres colors diferents, el verd, el groc i el vermell. A la unitat emissora, el led verd serveixen per a indicar que la comunicació ha sigut enviada correctament a la unitat receptora. En contra, el led vermell ens indica que no s'ha pogut enviar correctament el senyal. Per altra banda, el led groc ens indica que se està processant l'ordre escollida i que esperem uns instants.

A la unitat receptora, el led groc, serveix per indicar que la unitat està encesa i funcionant. Si aquest led estigués apagat o amb poca lluminositat, ens indicaria que s'ha de canviar la bateria. El led verd ens informa que s'ha rebut una senyal correctament. En aquesta unitat, el led vermell no té cap us.



Imatge 16 – LED

3.4.2 VISUALITZACIÓ A L'ORDINADOR

El que fa a la visualització d'informació a l'ordinador, el programa permet veure els estats actuals de cada taula així com les accions que demanen.



Imatge 17 – Missatge del software de gestió

El quadre d'estat, punt 1 a l'esquema, ens mostra l'estat actual de les taules del restaurant. Aquesta informació està dividida en 6 columnes diferents. En aquestes columnes se'ns indica les taules lliures, les que romanen al primer plat, al segon plat, al postre i/o cafè i les que demanen el compte o les que es troben esperant un dels anteriors estats. Per exemple, la taula 7 ha pres el botó següent plat, i està esperant a que l'hi portin o un altre cas és el de la taula 10 que es troba lliure.

Com es pot observar a la part inferior de l'esquema, un missatge informa quant una taula ha enviat una acció (punt 2). Els diferents missatges que es poden rebre són: la taula ha demanat un extra, la taula ha demanat el següent plat, la taula ha demanat el compte. Al cas que es mostra d'exemple la taula ha demanat el següent plat.

3.5 Consum

Al ser aquest un sistema alimentat per bateries s'ha de tenir en compte el consum d'energia de cadascuna de les unitats. Aquestes funcionen amb una pila de 9V, que tenen una capacitat de 550mAh. S'ha mesurat el consum màxim de corrent de cadascuna de les plaques i ha donat aproximadament 11,2mA . Fent el càlcul de la durada màxima s'obté que el sistema pot estar funcionant 49 hores.

$$550 \text{ mAh} / 11,2 \text{ mA} = 49 \text{ hores aproximadament.}$$

CAPÍTOL IV

PROGRAMACIÓ

DEL SISTEMA

4.1 Introducció

La programació d'aquest projecte es divideix en tres parts:

- La programació de la unitat emissora
- La programació de la unitat receptora
- La programació de la unitat de gestió

La unitat emissora i la receptora es programen en llenguatge C mitjançant l'entorn gratuït PSoC Designer què és proporciona amb el kit de Cypress. El codi generat per aquesta aplicació, que s'anomena firmware, s'ha d'introduir al microcontrolador. Això es fa mitjançant el mòdul de programació, anomenat PSoC MiniProg, que es connecta a l'ordinador a través del bus USB i al "PSoC" per mitjà d'un connector específic anomenat ISSP (In System Serial Programmer).

S'ha d'anar amb compte de no confondre la programació de l'aplicació amb la programació del microcontrolador. La primera es refereix a la codificació d'un programa en C que realitzi una acció concreta, mentre que el segon es refereix a la gravació d'aquest programa o firmware (un cop compilat) a l'interior del microcontrolador. Cal dir que aquest microcontrolador pot ser programat (gravat) milers de vegades. Ve a ser com una memòria flash que permet ser reescrita. Això dona una gran facilitat a l'hora de fer modificacions que serien extremadament costoses amb un sistema electrònic amb components discrets.

Pel que fa a la programació de l'aplicació al PC, aquesta es programa en llenguatge visual C# mitjançant l'entorn "Visual C# 2008 Express Edition". Aquest entorn gratuït disposa de totes les eines bàsiques per tal de generar el codi, depurar-lo i publicar-lo, ja sigui a Internet o en un suport físic. Per altra banda, es fa ús d'una base de dades en MySQL, aplicació també gratuïta i que encara que es pot programa amb una aplicació específica de MySQL, la programació es fa a través del C#.

Aquest tipus de programació és diferent a la de les unitats, ja que al ser Visual C# un llenguatge de programació visual, no s'executa en seqüència. L'execució és basa en el que s'anomena "esdeveniments". Per exemple, quan es fa clic en un botó es genera un esdeveniment i s'executa el codi associat a aquest. Per això es diu que no s'executa de manera seqüencial ja que depenent del que faci l'usuari s'executarà un o altre codi. Tots els elements que es mostren a la finestra s'anomenen controls o components. Cada un d'aquests és un objecte que disposa de propietats i mètodes. Per exemple, un botó té les propietats de mida, color, text que es mostra, etc. i els mètodes de fer clic, seleccionar, etc.

4.2 Programació de la unitat emissora

Bàsicament la unitat emissora és una màquina d'estats on cada un d'aquests representa una etapa en el funcionament de l'aplicació. L'esquema d'estats es mostra en el següent diagrama:

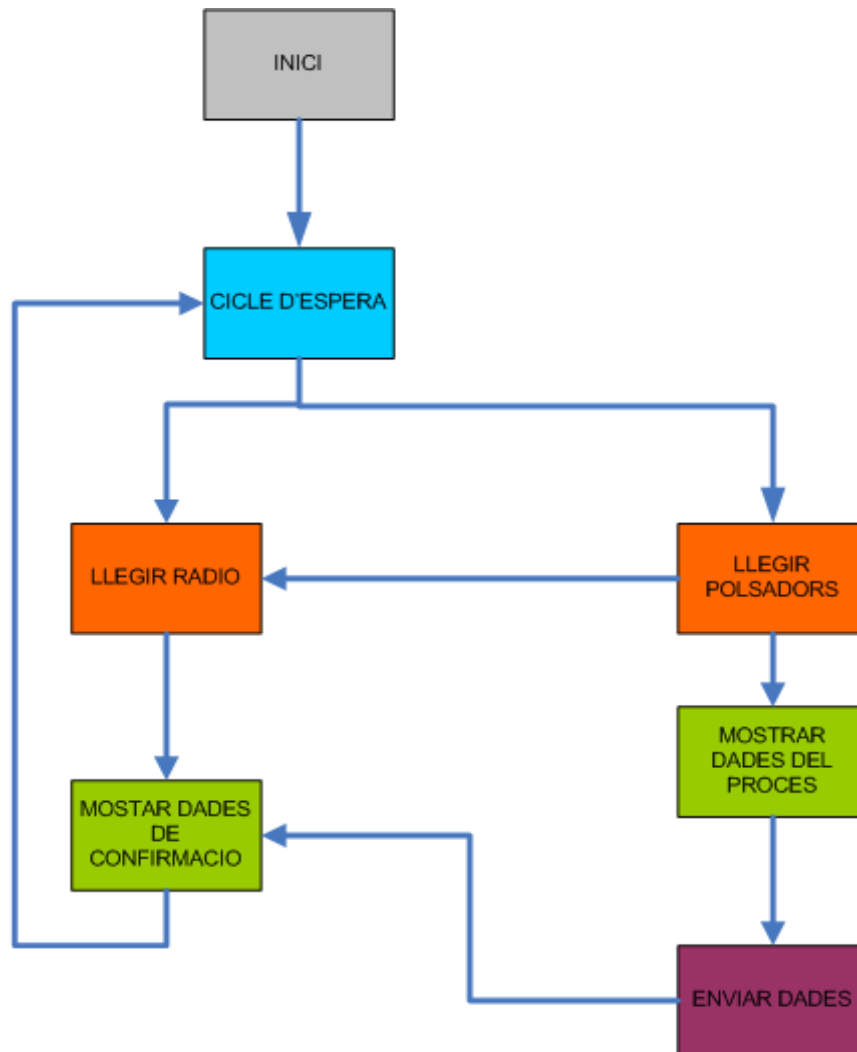


Figura 13 – Esquema d'estats unitat emissora

Estat	Descripció
Inici	Inicialitza les variables i els diferents components.
Cicle d'Espera	Espera a que un polsador s'accioni o es rebi un missatge.
Llegir Radio	Llegeix la comunicació entrant.
Llegir Polsador	Llegeix el polsador que s'ha pres.
Mostrar	Mostra visualment l'acció presa.
Enviar	Envia les dades sol·licitades.
Mostrar Confirmació	Mostra visualment la confirmació o no de la comunicació.

Taula 4 - Estats unitat emissora

La unitat emissora farà sempre aquest cicle fins que s'apagui o s'esgoti la bateria. Internament cada estat realitza les següents operacions:

Estat Inicial:

1. Inicialització de variables i subrutines.
2. Inicialització de components hardware: polsadors, leds, pantalla LCD i el mòdul de radio per les comunicacions.
3. Mostreig del missatge "PFC Waiter 1.0" i "Albert Carrete" per la pantalla LCD. Més tard mostrarà la taula assignada: "MESA - 001"
4. Configura les opcions de la radio per enviar i rebre paquets.

Estat Espera:

1. Es mante en un cicle esperant una interrupció dels polsador o del mòdul de radio conforme s'ha rebut un paquet.

Estat Llegir polsador:

1. Llegeix la rutina de programació del polsador que ha generat la interrupció.
2. Comprova si s'han rebut paquets pel mòdul de radio. Si s'han rebut, primer els processa, si no, continua.
3. Prepara el missatge per enviar corresponent a l'acció sol·licitada.

Estat Mostrar dades:

1. Encén el LED groc de processament de dades.
2. Comprovació de que no es rebí cap paquet. Si es rep es processa, en cas contrari continua amb el següent pas.
3. Mostreig del missatge corresponent a la acció desitjada:
"PIDIENDO PLATO", "GENERANDO CUENTA", "PIDIENDO EXTRA"
4. Mostreig de la barra de progrés.

Estat Enviar dades:

1. Enviament de les dades necessàries segons l'acció demanda.
2. Espera del paquet de confirmació de comunicació.
3. Apagar LED groc de processament.

Estat Mostrar dades:

1. Encesa de LED verd si la comunicació ha sigut confirmada o LED vermell si ha fallat.
2. Mostreig de missatge "PETICION ENVIADA" si la comunicació s'ha realitzat correctament, o en contra, mostreig del missatge "FALLO DE ENVIO" si la comunicació ha fallat.
3. Configuració del sistema per tornar al mode d'espera.
4. Apagada de LED verd o vermell segons la comunicació.

Per fer l'enviament d'informació cap a la unitat receptora es fa servir el mòdul de ràdio. El seu funcionament s'ha explicat anteriorment. Aquest mòdul és un petit circuit "PSoC" connectat al principal a través d'un bus SPI. Quan el programa (anomenat firmware en aquest tipus de sistemes) vol accedir al mòdul de radio, ho fa a través d'un bloc de comunicació SPI que disposa de les funcions i registres necessaris per aquesta comunicació.

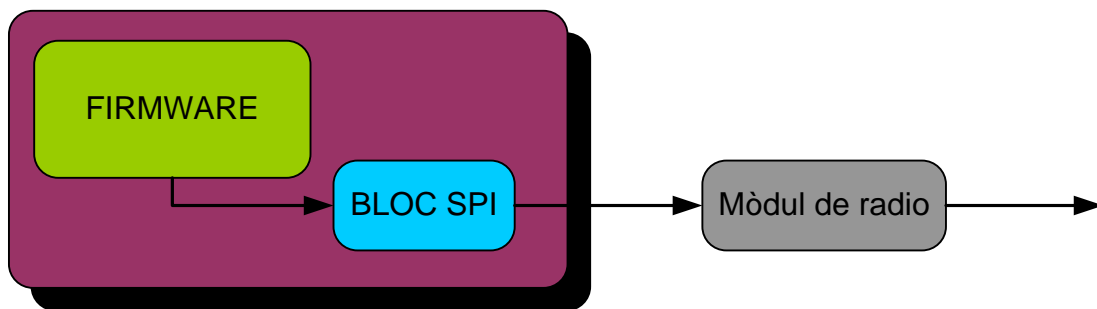


Figura 14 – Esquema de comunicació a través del bloc SPI

4.3 Programació de la unitat emissora

La unitat receptora també és una màquina d'estats. Cada estat representa una etapa en el funcionament de l'aplicació. El diagrama de flux d'aquests estats es mostra a continuació:

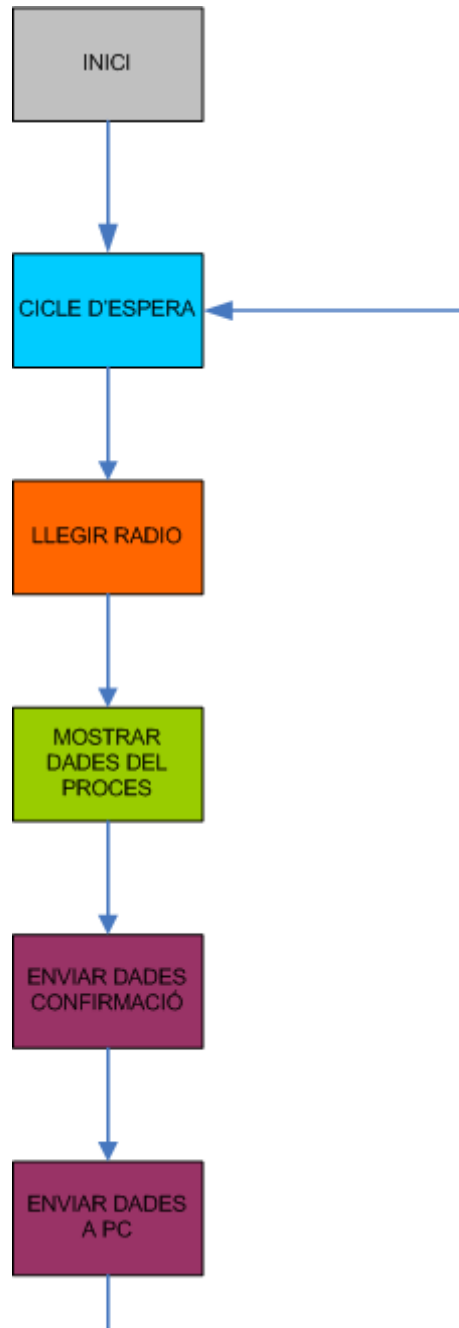


Figura 15 – Esquema d'estats unitat receptora

Estat	Descripció
Inici	Inicialitza les variables i els diferents components.
Cicle d'Espera	Espera a que es rebi un missatge.
Llegir Radio	Llegeix la comunicació entrant.
Mostrar dades	Mostra visualment que s'ha rebut un paquet.
Enviar Confirmació	Envia senyal de confirmació de paquet rebut.
Enviar senyal al PC	Envia les dades rebudes al PC.

Taula 5 - Estats unitat receptora

A diferència de la unitat emissora, aquesta es manté esperant tan sols a rebre senyals per el mòdul de radio. Una vegada rebuda una senyal tots els estats són conseqüència de l'anterior.

Estat Inicial:

1. Inicialització de variables i subrutines.
2. Inicialització de components hardware: leds i el mòdul de radio per les comunicacions.
3. Configura les opcions de la radio per enviar i rebre paquets.

Estat Espera:

1. Encén el LED groc per indicar que el sistema està funcionant.
2. Es mante en un cicle d'espera esperant una interrupció del mòdul de radio conforme s'ha rebut un paquet.

Estat Llegir radio:

1. Comprova i processa el missatge rebut.
2. Prepara el missatge per enviar al PC.

Estat Mostrar dades:

1. Encén el LED verd de senyal rebuda correctament.

Estat Enviar confirmació:

1. Enviament de senyal de confirmació a la unitat emissora corresponent.

Estat Enviar dades a PC:

1. Envia per el port sèrie amb direcció al PC el paquet rebut.
2. Apaga LED verd de rebuda de senyal.
3. Configuració del sistema per tornar al mode d'espera.

4.2 Programació de la unitat de gestió

Pel que fa a l'aplicació encarregada de la gestió de les comunicacions, construïda en C#, es pot dividir en dos parts essencials: el Serial Port i la base de dades (MySQL).

El Serial Port s'encarrega de les comunicacions per el port sèrie i genera un esdeveniment cada cop que es rep un senyal. D'aquesta manera es pot processar la informació rebuda sense haver d'estar constantment comprovant el port. La base de dades és l'encarregada de guardar la informació de les taules que més tard serà processada. Així doncs, cada cop que es necessiti saber o modificar l'estat d'una taula, es consultarà i modificarà la base de dades. Més endavant veurem una explicació més detallada de com funciona cada element.

4.2.1 DESCRIPCIÓ DEL MENÚ PRINCIPAL

La unitat de gestió està programada per un ús fàcil i simple de la informació de les taules. La següent figura mostra el menú principal:



Imatge 18 – Descripció del Menú Principal

Com que es tracta d'una aplicació de Windows, l'estructura d'execució ja no és tan seqüencial com passava al "PSoC". En aquest software, les diferents opcions estan repartides en botons d'acció. Així doncs, hi ha algunes opcions accessibles en qualsevol moment. És el cas dels botons sortir, opcions de número de taula, minimitzar el programa, connectar i desconnectar (aquests dos últims són complementaris, si un estat activat l'altre està desactivat).

Opció 1: Mostra els tres botons que són els encarregats de controlar el programa.

Opció 2: Mostra els requadres de text amb la informació de l'estat de les taules.

Opció 3: Mostra els diferents botons de la configuració manual.

Opció 4: Indica que s'ha rebut un avís d'una taula.

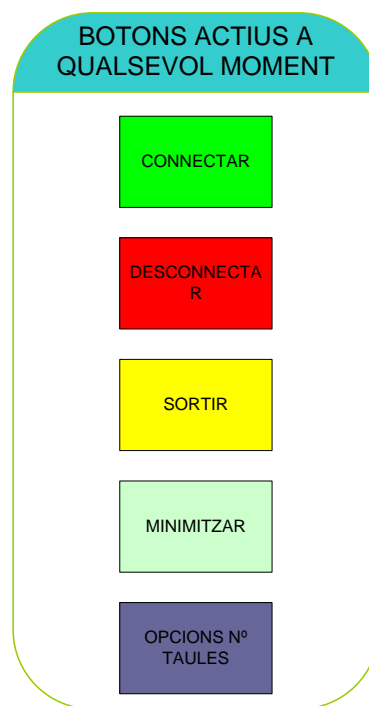


Figura 16 – Estats dels botons actius en qualsevol moment

4.2.2 BOTONS ACTIUS

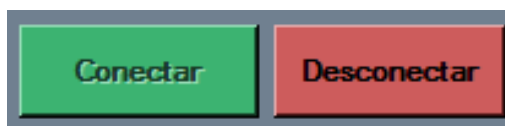
Ara es fa una breu explicació de les diferents funcions i processos que fan internament aquestos botons especials:

Botó CONNECTAR:

- Quan es prem el botó connectar, una vegada ja carregat el programa, aquest internament i sense intervenció de l'usuari es posa a escoltar el port sèrie i així notificar un esdeveniment. També fa una consulta a la base de dades "MySQL" i la mostra als diferents quadres de text destinats a aquesta informació. Per altra banda activa els botons per fer una configuració manual de les taules. Si el botó connectar s'ha pres, es desactiva immediatament i activa el botó desconnectar.

Botó DESCONNECTAR

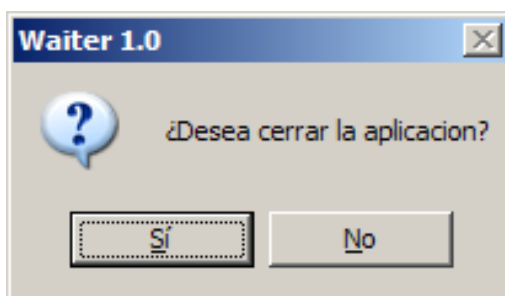
- Igual que el botó connectar, aquest una vegada pres, es desactivarà i només es tornarà a activar quan es pressioni el botó connectar. Aquest botó té la funció de tancar la connexió amb el serial port, de netejar el quadres de text i els avisos generats a més de desactivar la configuració manual de les taules.



Imatge 19 – Botons Connectar i Desconnectar

Botó SORTIR

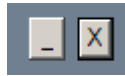
- Aquest botó compleix la opció de tots el programes de sortir. És important dir que té prioritat per damunt de totes les altres opcions. Si s'ha pres, un missatge ens preguntarà si volem sortir. En cas afirmatiu, tanquem el port sèrie i esborrem la base de dades "MySQL". Si la resposta és negativa, el programa continuarà fent el que feia.



Imatge 20 – Missatge de diàleg de sortir del programa

Botó MINIMITZAR

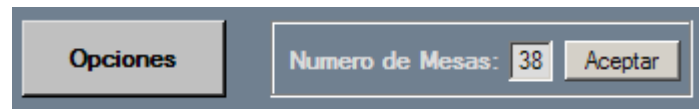
- El botó minimitzar, que es troba amb el símbol “_” al costat de la “X” de sortir del programa, té la simple funció d’amagar la finestra a la barra de Menú d’Inici de Windows. Polsant de nou el requadre del programa a la barra de Menú d’Inici, tornarà a la posició anterior.



Imatge 21 – Botons Sortir i minimitzar

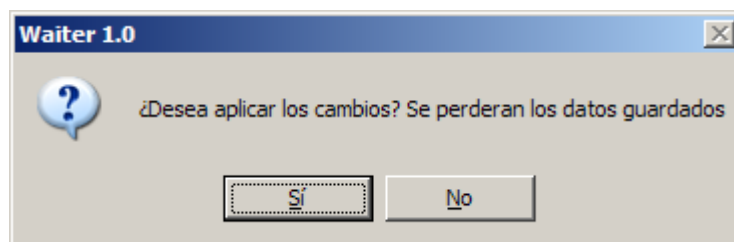
Botó OPCIONS

- La funció del botó opcions, ens mostrar un requadre per seleccionar el numero de taules que volem gestionar i un botó per aplicar els canvis. Si això ja era visible i es prem el botó Opcions, l’amaga.



Imatge 22 – Boto Opcions i requadre de configuració

- Una vegada s’ha introduït el numero de taules al requadre destinat per això i s’ha pres el botó acceptar, el programa et pregunta si vols aplicar el canvis. Si la resposta és afirmativa, internament s’esborra la taula de la base de dades i crear una nova amb els canvis requerits. Com que es tracta d’un botó sempre actiu, si el programa no està connectat i s’apliquen els canvis, no es mostrarà la configuració fins que es connecti. Si diem que no volem aplicar els canvis, es continuarà amb la configuració anterior.



Imatge 23 – Missatge de diàleg aplicar els canvis

4.2.3 CONFIGURACIÓ MANUAL

Ara es fa una descripció de les opcions de la configuració manual. Existeixen dos menús de botons de configuració. Un és estàtic i sempre visible a la dreta del programa (botons tàctils), l'altre menú es crida amb el botó dret del ratolí al seleccionar una taula del requadre d'informació. Els diferents botons activats varien segons l'estat de la taula que es vol modificar.



Figura 17 – Estats dels botons de configuració manual

A continuació es fa una descripció de les accions que fan i els processos que porten a terme aquestos botons de configuració manual. Els botons tàctils com el submenú fan les mateixes funcions. Les diferents opcions possibles són:

Botó Següent plat:

- Aquest botó fa una consulta a la base de dades per veure l'estat actual de la taula. Si es troba lliure, actualitza l'estat a primer plat. Si es troba al primer, actualitza l'estat al segon plat i si es troba al segon plat actualitza a l'estat postra o cafè.

Botó Compte:

- El botó compte té la funció d'actualitzar l'estat de la taula a l'estat compte. Si ja es trobava en aquest estat deixa l'estat a la base de dades tal i com estava.

Botó Enviar missatge:

- Encara que aquesta opció no es pot utilitzar, està programat de cara a una nova versió. La funció d'aquest botó és d'enviar missatges preestablerts des del programa de gestió a una taula en concret. Per fer això necessita connectar per el port sèrie per enviar el missatge estructurat de forma que la unitat emissora ho entengui.

Botó Deseleccionar:

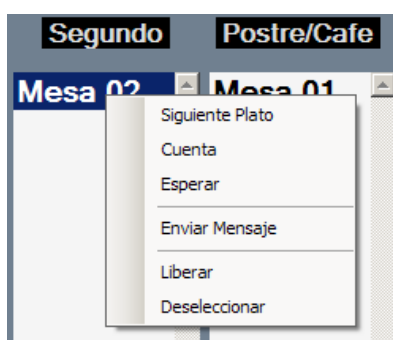
- Aquest botó tan sols desselecciona la taula seleccionada.

Botó Esperant:

- La funció d'aquest botó és indicar que la taula romana esperant el seu estat. Per fer això no es modifica ni es consulta l'estat de la base de dades. La columna d'espera està ordenada per ordre de temps. Tenen més prioritat el que més temps porta esperant.

Botó Alliberar taula:

- Aquest botó modifica directament la base dades a l'estat lliure. No fa falta fer una consulta perquè aquesta ordre té prioritat per damunt de la resta.



Imatge 24 – Botons configuració manual ratolí

Botó Esborrar taula:

- Aquest botó té la funció d'esborrar del requadre d'espera, la taula gestionada. Ja que aquesta informació no es troba a la base de dades, no té que fer ni consulta ni modificació.



Imatge 25 – Botons configuració manual tàtil

Botó Eliminar Avís:

- Aquest botó elimina l'avís generat per un esdeveniment rebut. Aquest botó no té submenú de botó dret de ratolí, però en comptes d'això, fent dos vegades clic amb el ratolí s'esborra automàticament.

ATENCIÓN: MESA 01 PIDE PLATO

Imatge 26 – Missatge d'esdeveniment

4.2.4 ETAPES DE PROCÉS I CASOS D'ÚS

Pel que fa a la execució del programa, la següent figura mostra l'esquema de seqüència que realitza:

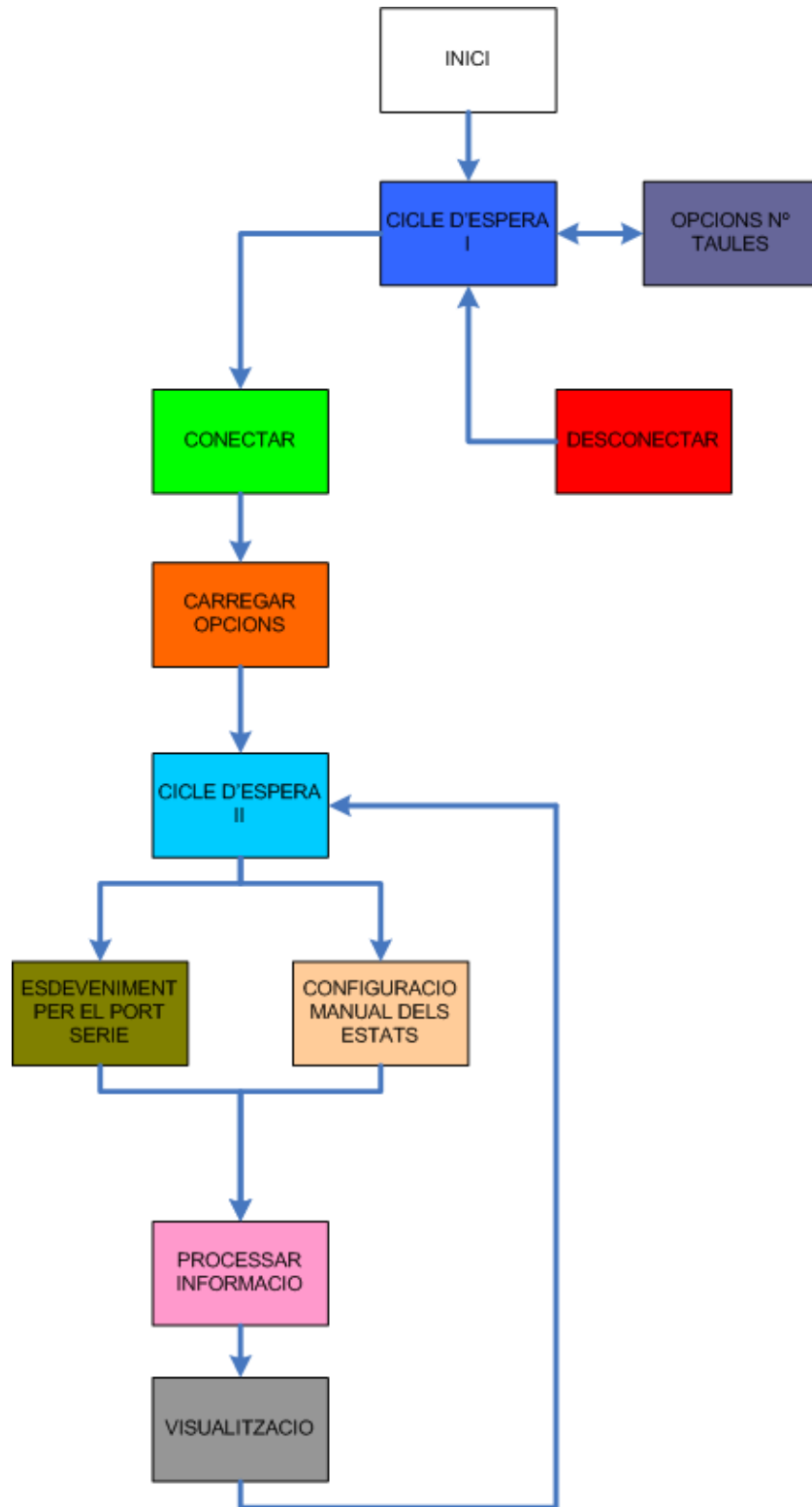


Figura 18 – Esquema d'estats unitat de gestió

A diferència de les altres dues unitats (unitat receptora i unitat emissora), la unitat de gestió té dos cicles d'espera. En el primer es configura la base de dades i resta esperant que l'usuari ordeni una acció. En el segon cicle d'espera, la unitat de gestió resta esperant un esdeveniment per el port sèrie, enviat per la unitat receptora. Si l'usuari ho necessita, també pot configurar manualment l'estat de les taules, o fer ús dels botons que sempre estan a actius.

Estat	Descripció
Inici	Inicialitza les variables i crea la base de dades "MySQL"
Cicle d'Espera I	Es mante esperant ordres de l'usuari.
Connectar	Configura el port sèrie per rebre esdeveniments.
Carregar Opcions	Mostra visualment la base de dades.
Cicle d'Espera II	Es mante esperant un esdeveniment o una acció de l'usuari.
Esdeveniment	Llegeix el port sèrie.
Configuració Manual	Modifica manualment l'estat de les taules
Processament	Processa la nova informació, ja sigui rebuda per port sèrie o per configuració manual
Visualització	Mostra la nova informació.

Taula 6 - Estats unitat de gestió

Estat Inici:

1. Inicialització de variables i tots els components gràfics necessaris.
2. Es crida a la subrutina "Limpiar":
 - a. Desactiva els botons de configuració manual.
 - b. Desactiva el submenú de clic dret amb el ratolí.
 - c. Neteja els requadres d'informació.
3. Es crida a la rutina "MySQL":
 - a. Es comproven les base de dades existents.
 - b. S'esborra si existeix la base de dades antiga.
 - c. Es crea una nova en blanc.
 - i. Es crea la taula.
 - ii. Inserim els registres (10 taules).

Estat Cicle Espera I:

1. Es manté esperant a que l'usuari demani una acció. Les accions possibles són: connectar, opcions, minimitzar o sortir

Estat Connectar:

1. Es connecta el "SerialPort" per escoltar per el port sèrie.
2. Es crida a la subrutina "Mostrar":
 - a. Neteja tots els valors dels requadres d'informació.
 - b. Fa una consulta a la base de dades.
 - c. Segons els valors llegits de la base de dades, distribueix la informació en el requadre corresponent.
3. Es crida a la subrutina "Limpiar" per evitar errors de programació.

Estat Carregar Opcions:

1. Si és visible el menú de configuració, l'oculta, sinó el visualitza.
2. Crida a la subrutina "Limpiar".
3. Si s'ha introduït valors al requadre de número de taula i es fa clic al botó acceptar:
 - a. S'esborra la taula de la base de dades.
 - b. Es crea una taula nova.
 - c. Afegim els registres amb la nova configuració.
 - d. Netegem.
 - e. Oculta el menú d'opcions.

Estat Cicle Espera II:

1. Es manté esperant a que l'usuari demani una acció o es rebi un esdeveniment per el port sèrie.
 - a. L'usuari pot utilitzar la configuració de les taules manuals. Si selecciona una taula, s'activen els botons possibles per configurar.
 - b. Es rep un esdeveniment

Estat Esdeveniment:

1. Es llegeix el port sèrie (subrutina "Recepcion").

Estat Configuració Manual:

1. Si es selecciona una taula s'activen els botons corresponents al estat. També existeix l'opció clic dret del ratolí per desplegar un submenú amb aquests botons. Les opcions possibles són:
 - a. Plat següent, Compte, Enviar missatge (Actualització Waiter 2.0), Deseleccionar, Esperant, Alliberar taula, Esborrar taula, Eliminar avís.

Estat Processament:

1. Es crida a la rutina "Actualizar" per processar les dades rebudes:
 - a. Es processa les dades rebudes: taula d'origen i petició.
 - b. Es consulta la base de dades per veure si la petició de la taula és correcta (subrutina "Consulta").
 - c. Si és correcta, es fa el processament necessari per modificar la base de dades, sinó ho és, la deixa tal i com està.

Estat Visualització:

1. Mostra un missatge d'avís de taula sol·licitant plat, extra o compte, si s'ha rebut un esdeveniment.
2. Es mostren les dades actualitzades generades per el nou esdeveniment (subrutina "Actualizar") o per configuració manual (subrutina "Actualizar Mesas") als requadres corresponents.

Per altra banda, un diagrama de casos d'ús és una tècnica per a la captura de requisits potencials d'un nou sistema o una actualització de software. Cada cas d'ús proporciona un o més escenaris que indiquen com deuria d'interactuar el sistema amb l'usuari per aconseguir un objectiu específic.

Normalment, en els casos d'ús s'evita la implementació de llenguatge tècnic, preferint en el seu lloc un llenguatge més proper a l'usuari final. A continuació es mostra el diagrama de casos d'us del software.

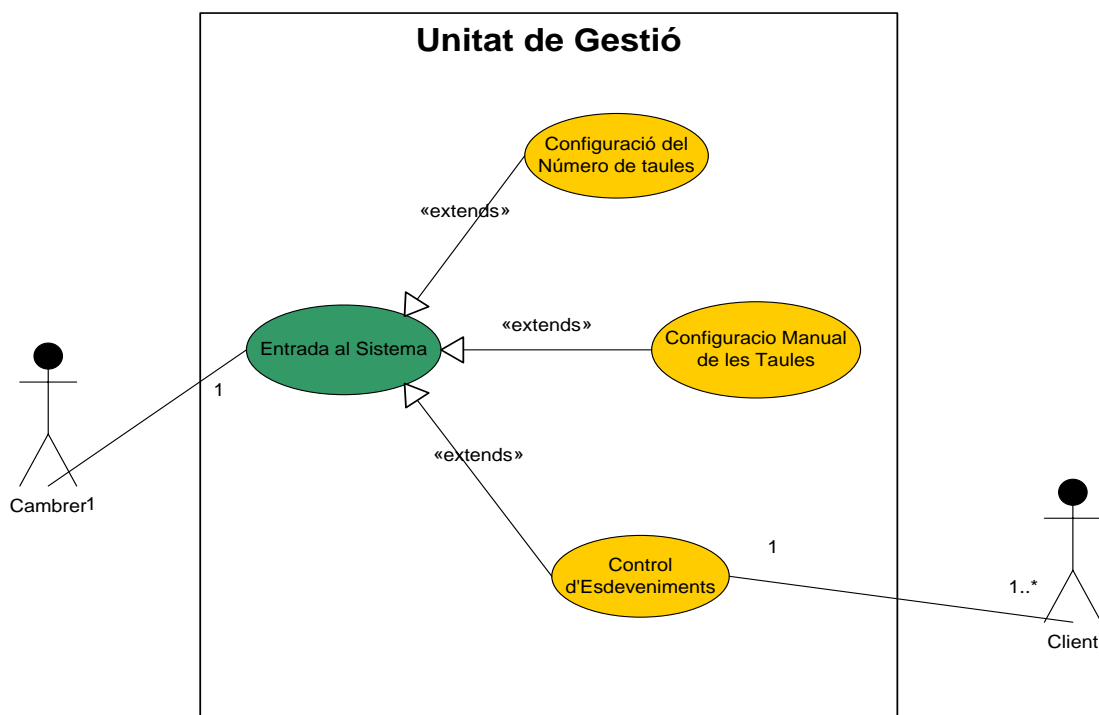


Figura 19 – Diagrama de casos d'ús

Els cambrers són els encarregats d'iniciar el sistema. Una vegada són dins poden interactuar amb ell configurant el número de taules, configurant manualment l'estat de les taules o controlant els esdeveniments.

Els clients són els encarregats de generar els esdeveniments al sistema a través de les unitats emissora i receptora.

4.2.5 BASE DE DADES

Una base de dades és un grup d'arxius relacionats que faciliten un ràpid accés a la informació necessària per prendre de decisions. Un gestor de base de dades és un software molt específic que està destinat a fer-se servir de interfase entre la base de dades i els usuaris o aplicacions que la utilitzen.

La unitat de gestió utilitza com a gestor "MySQL". Aquest software és lliure i per tant gratuït. A més, és molt fàcil de configurar, d'utilitzar i és reconegut per el Visual C#, que permet programar totes les seves funcions de manera interna.

La base de dades que s'utilitza és molt senzilla. Aquesta s'anomena "Waiter". Tan sols disposa d'una taula i de 'n' registres (número de taules per a la gestió). El número de registres es pot configurar automàticament amb el botó opcions.

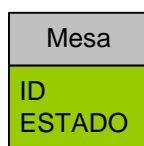


Figura 20 – Taula de la base de dades

Les ordres per manipular la base de dades són:

```
- @"SHOW DATABASES"
```

Ens mostra les bases de dades que hi han configurades.

```
- @"CREATE DATABASE waiter;"
```

Crea una base de dades nova anomenada "waiter".

```
- @"CREATE TABLE mesa(id varchar(2), estado varchar(2));"
```

Crea una tabla a la base de dades activa amb "id" i "estado" com a "Varchar(2)".

```
- @"INSERT INTO mesa(id, estado) VALUES ('" + numero + "', 'LI')"
```

Introdueix a la taula "mesa", als camps "id" i "estat" els valors necessaris.

```
- @"DROP DATABASE waiter;"
```

Eborra la base de dades "waiter".

CAPÍTOL V

PROVES AL

SISTEMA CONFIGURAT

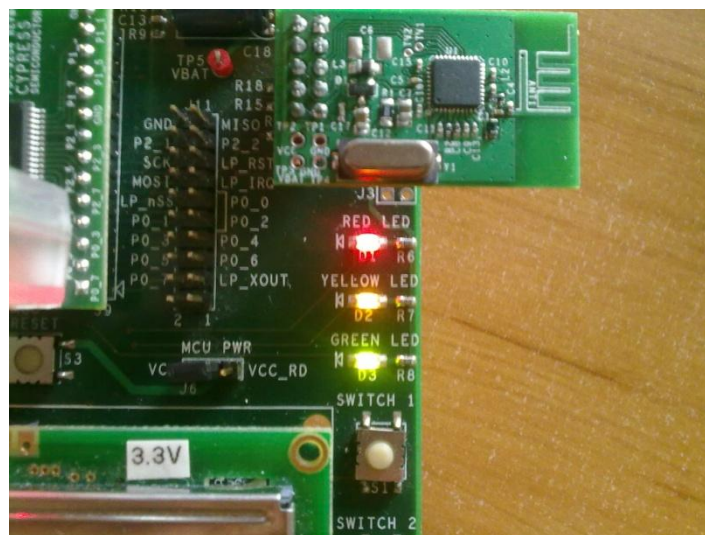
5.1 Introducció

En un projecte com el que s'ha realitzat és molt important fer proves cada vegada que es desenvolupa una part, ja que es treballa amb tecnologia nova i no es té experiència anterior. Sí que s'havia treballat anteriorment amb sistemes de tipus "PSoC" però no s'havia fet cap muntatge tan complex.

La metodologia que s'ha anat fent servir durant la major part del projecte ha sigut la de desenvolupament, prova i error, per anar aprenent com funcionen el sistemes i com es controlen aquests.

5.2 Proves amb les plaques

El primer que es va provar va ser el sistema de visualització a través dels LED i les pantalles LCD. Quan això fa estar funcionant es va començar a provar el sistema de polsadors. Una vegada es polsava un polsador, si estava ben programat sortia un missatge per pantalla i encenia un LED.



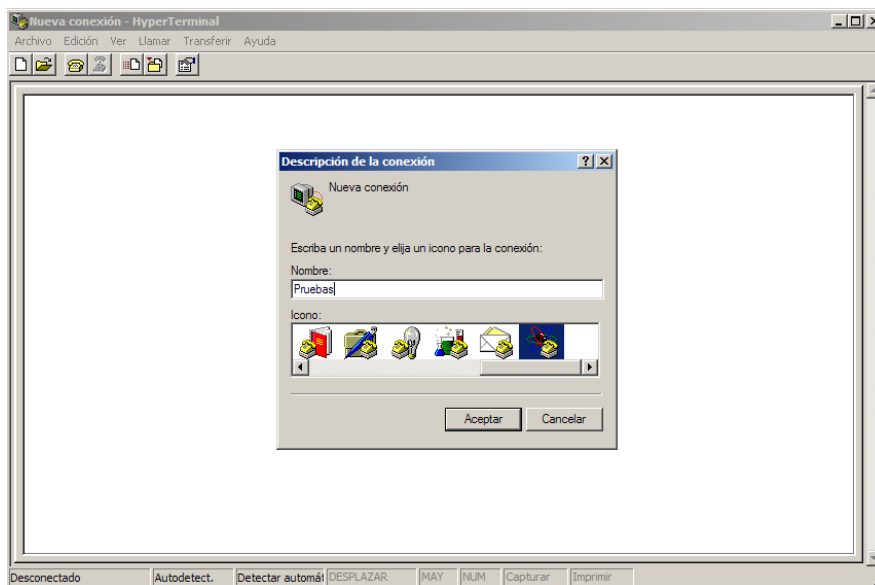
Imatge 27 – Proves visuals

Quant això ja era una realitat, es va continuar amb el desenvolupament i les següents proves de comunicacions entre unitats. Per fer això es va construir un programa que generava valors aleatoris i els enviava a la unitat receptora a intervals d'un segons. A la unitat receptora es visualitzava aquest valor per pantalla. D'aquesta forma es va poder provar les diferents formes de comunicacions i velocitats entre les unitats i comprovar com aquestes influïen en la distància màxima entre les dues unitats.

5.3 Proves entre placa i PC

Quan les comunicacions entre les plaques ja eren una realitat, es va prosseguir amb les proves de comunicacions entre la unitat receptora i la unitat de gestió al PC. Es va començar fent proves amb l'hiperterminal de Windows, és a dir, el missatges que la unitat receptora enviava per el port sèrie, s'havien de mostrar al PC i tornar-los a enviar a la unitat receptora per visualitzar. Un cop fet això es va començar a desenvolupar el programa en C# capaç de llegir el port sèrie, processar les dades i mostra-les per pantalla.

Amb aquestes proves es va descobrir que aquest model de placa no permet llegir del port sèrie, per el qual, es va desestimar la possibilitat d'enviar missatges en direcció a les taules.



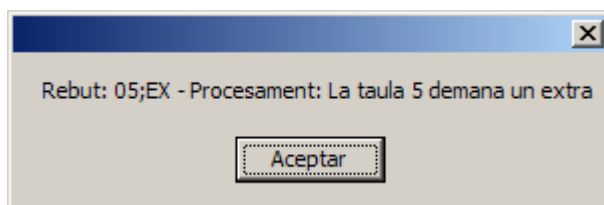
Imatge 28 – Proves amb l'hiperterminal

5.4 Proves de processament

Una de les proves més importants a l'hora de desenvolupar un projecte de qualsevol tipus és comprovar si es processa correctament la informació, i per tant, la sortida és l'esperada.

La millor forma de fer aquestes proves és interrompin el cicle del programa en qualsevol punt i veure les dades amb les que està treballant.

Per fer el processament de les dades rebudes per el port sèrie, es mostrava els paquets que es rebien en missatges de Windows. Una vegada desxifrada la informació es mostrava a una finestra especial per a veure si les dades eren correctes.



Imatge 29 – Proves de processament

5.5 Proves de la base de dades

Per a construir i provar la base de dades es va utilitzar el gestor “MySQL Command Line Client” que permet fer totes les accions d’una base de dades (crear, esborrar, modificar) a través de línia de comandos. Una vegada provat això es va començar a implementar la base de dades al programa en C#. Gràcies a l’ajuda del missatges de Windows i els requadres d’informació del sistema es va anar provant.

```
MySQL Command Line Client
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 1
Server version: 5.1.47-community MySQL Community Server (GPL)

Copyright (c) 2000, 2010, Oracle and/or its affiliates. All rights reserved.
This software comes with ABSOLUTELY NO WARRANTY. This is free software,
and you are welcome to modify and redistribute it under the GPL v2 license

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| test |
+-----+
3 rows in set (0.09 sec)

mysql> use mysql
Database changed
mysql> show tables;
+-----+
| Tables_in_mysql |
+-----+
| columns_priv |
| db |
| event |
| func |
| general_log |
| help_category |
| help_keyword |
| help_relation |
| help_topic |
| host |
| ndb_binlog_index |
| plugin |
| proc |
| procs_priv |
| servers |
| slow_log |
| tables_priv |
| time_zone |
| time_zone_leap_second |
| time_zone_name |
| time_zone_transition |
| time_zone_transition_type |
| user |
+-----+
23 rows in set (0.15 sec)

mysql>
```

Imatge 30 – Proves de base de dades

5.6 Proves del Sistema Complet

Un cop finalitzats i provats tot els subsistemes es va realitzar una prova general per comprovar el correcte funcionament. Aquí es van detectar cert problemes de connexió del hardware. Les plaques s'han utilitzat molt cops per a investigació i ja que la memòria flash dels dispositius no esta massa utilitzada, redueix el seu rendiment en velocitat i bytes disponibles. El director del projecte les va reparar soldant de nou els microcontroladors a les plaques.

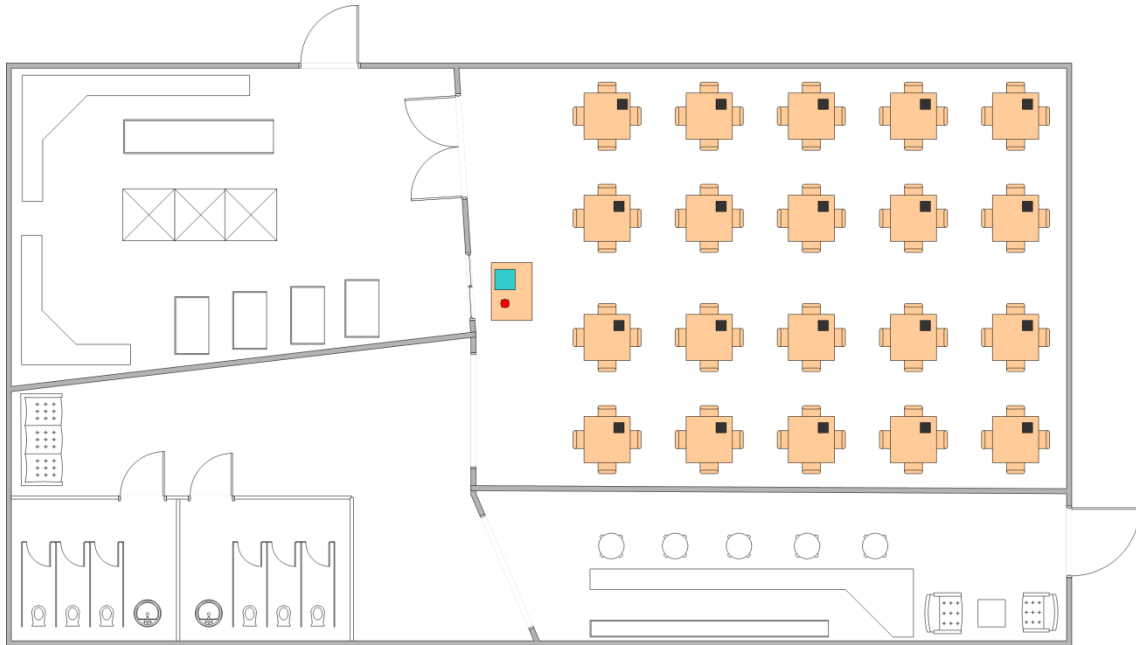
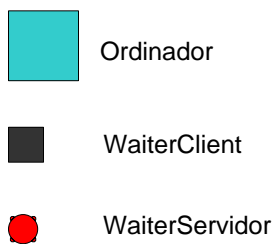


Figura 21 - Plànol restaurant amb el sistema instal·lat



CAPÍTULO VI

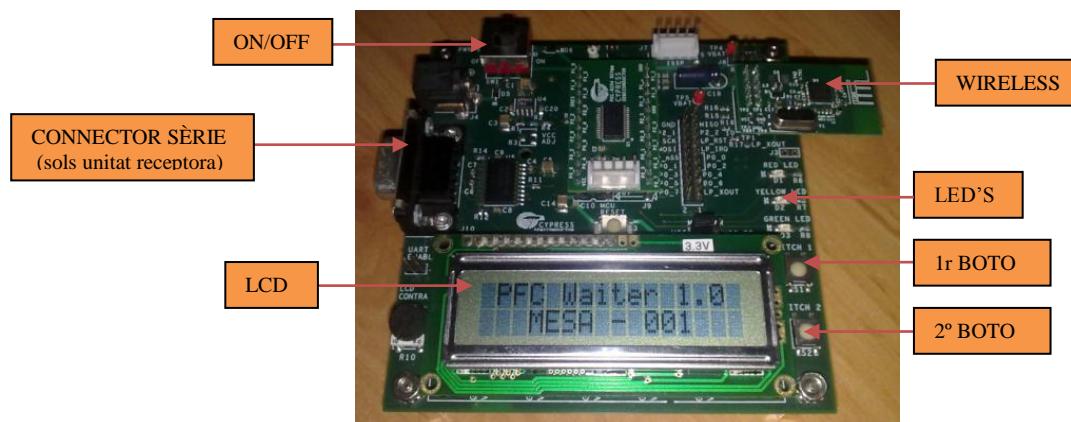
MANUAL DEL SISTEMA

6.1 Introducció

En aquest capítol es pretén orientar als usuaris en la utilització del sistema a través d'exemples reals. Aquest capítol està dividit en dos parts. La primera, manual de la unitat emissora i receptora, ens explica com es fa funcionar i es connecta el sistema creat. La segona part, manual de la unitat de gestió, ens fa una descripció del funcionament del sistema de gestió.

6.2 Manual de la Unitat Emissora i la Unitat Receptora

Per començar, cal connectar els dispositius. La unitat emissora s'ha de connectar amb el PC amb el cable sèrie. Per altra banda, la unitat emissora i receptora, necessiten d'una pila de 9V. Una vegada introduïda, s'encén polsant el boto On/Off a la part superior esquerra de la placa. Si el sistema està funcionant correctament es comprova amb el LED groc encès (en el cas de la unitat receptora) i amb el missatges a la pantalla LCD (en el cas de la unitat emissora). Una vegada que la unitat receptora està connectada i encesa, no necessita de cap altre configuració.



Imatge 31 – Descripció unitat emissora/receptora

Per veure com es fa funcionar correctament la unitat emissora, es fa una breu descripció a través d'un exemple real:

1. Un client entra al restaurant i s'asseu a una taula. El cambrer li porta la carta i li pren nota del primer, segon i del postra i/o cafè.
2. Quant el client vol el primer plat, polsa el primer polsador. El cambrer porta el primer plat. Quant en vol el segon, torna a polsar el primer polsador. En tot moment apareixen els missatges de les accions sol·licitades per la pantalla LCD i els LED s'il·luminen segons si la comunicació és correcta o no.
3. Ara, el client, vol un extra, així que prem el segon polsador i el cambrer va a la taula per prendre nota i portar-li el extra desitjat.
4. El client, que ara vol el cafè, torna a polsar el primer polsador. Ràpidament serà servit.

- Per últim, polsant a l'hora el primer i el segon polsador, es demana el compte total que serà portat a la taula.

6.3 Manual de la Unitat de Gestió

Una vegada s'ha executat el programa de gestió de les taules apareix el menú principal. Polsem el botó Opcions, introduïm el numero de taules a gestionar i acceptem. Ara connectem el programa i romandrà a l'espera.

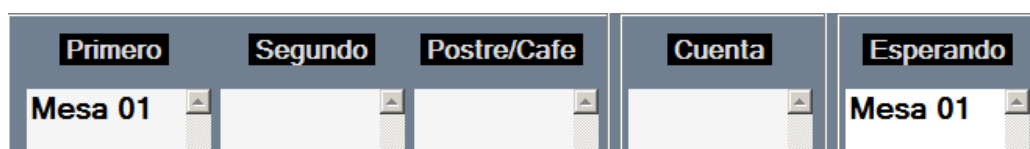


Imatge 32 – Descripció unitat de gestió

Ara és el moment de la configuració manual. Per exemple, un client no vol primer plat. Llavors el seu estat serà segon plat. Selecciónant la taula del client en qüestió i polsant dues vegades el botó Següent Plat, s'actualitzarà el seu estat.

Per altra banda, suposant un exemple estàndard a continuació es descriu, a través del exemple anterior, com s'utilitza aquest programa:

- El client demana el primer plat. Apareix un missatge al requadre de esdeveniments (part inferior del programa) conforme la taula en qüestió vol el primer plat. Aquesta taula romandrà a l'estat primer plat i al requadre d'espera.



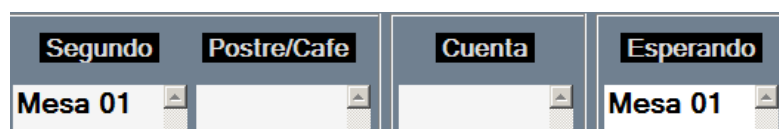
Imatge 33 – Taula esperant primer plat

L'encarregat de gestionar el programa prem el botó eliminar avís per eliminar l'avís generat i una vegada el client ha sigut servit, selecciona la taula i prem el botó esborrar ja que la taula ja no està esperant.

ATENCIÓN: MESA 01 PIDE PLATO

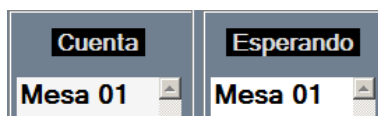
Imatge 34 – Missatge de esdeveniment

2. Quant el client demana el següent plat, el procediment és el mateix que el segon.
3. Ara, el client, demana un extra. Apareix el missatge d'extra i la taula, mantenint l'estat al que es troba, s'afegeix a la cua del requadre d'espera. Es dona l'ordre d'anar a servir la taula. Una vegada servida s'esborra amb el botó esborrar taula.



Imatge 35 – Taula demana un extra

4. Per últim, el client sol·licita el compte. El cambrer li porta per cobrar i s'elimina de la columna d'espera. Una vegada el client ja no és a la taula, es posa la taula a l'estat lliure amb el botó alliberar.



Imatge 36 – Taula demana el compte total

CAPÍTOL VII

FUTURES AMPLIACIONS, ESTUDI DE MERCAT I CONCLUSIONS FINALS

7.1 Ampliacions

L'ampliació més òbvia és la de dotar a les plaques d'una estructura externa per fer el sistema més còmode a l'ull i millorar la integració dins del restaurant. Per altra banda, a l'hora de dotar-la d'una estructura externa, cal també retallar parts de la placa que no serien necessàries per així reduir el cost i l'espai que ocupen.

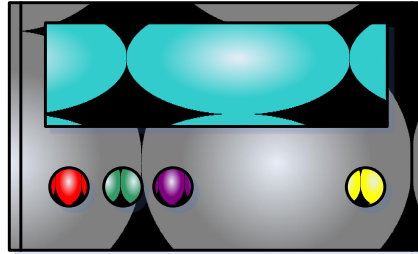


Figura 22 – Unitat emissora amb estructura externa

Per altra banda, l'ampliació més important és la de poder escollir el menú a través dels botons. És a dir, millorar la unitat emissora per a que surtin els plats disponibles i el client, a través de navegació amb l'aparell, pogués escollir el que vol a qualsevol moment (carta digitalitzada). Per la part de programació i de hardware no suposaria gaire cost més, però sí més hores de desenvolupament del sistema. En quant a resultat, per la part del restaurant, augmentaria els beneficis de manera molt elevada.

Fent la combinació d'aquesta última ampliació amb els sistemes actuals de gestió de comandes de les taules (sistemes amb PDA), l'automatització d'un restaurant rondaria el 90%.

7.2 Estudi de Mercat i d'amortització

Un estudi de mercat consisteix en una iniciativa empresarial amb el fi de fer una idea sobre la viabilitat comercial d'una activitat econòmica.

Per altre costat, un estudi d'amortització s'utilitza per denominar un procés financer a través del qual s'extingeix gradualment un deute per mitja de pagaments periòdics que poden ser iguals o diferents.

Suposem una empresa que s'encarrega de construir sistemes de hardware segons les peticions dels clients. Aquesta empresa construeix aparells per peces. Li fem una comanda de construcció per parts de les diferents unitats del nostre sistema (unitat receptora i unitat emissora).

El cost total dels serveis d'aquesta empresa són:

NOM	UNITATS	COST	COST TOTAL
U. Emissora			
LCD	1	30,00 €	30,00 €
RADIO	1	35,00 €	35,00 €
BOTONS	2	5,00 €	10,00 €
ENSSAMBLATGE			15,00 €
TOTAL			75,00 €
U. Receptora			
RADIO	1	35,00 €	35,00 €
SERIE	1	10,00 €	10,00 €
ENSSAMBLATGE			15,00 €
TOTAL			60,00 €

Taula 7 – Costos plaques per peces

Suposem ara un restaurant de 40 taules, ja que el sistema s'aplicarà a restaurant de un volum de taules elevat, de com a mínim aquesta capacitat, o en un restaurant on la seva forma estructural sigui tan irregular i dispersa que no es puguin veure les taules.

S'estableix que cada taula té una unitat emissora i que hi ha una unitat receptora més un PC. Els costos de compra del sistema complet són:

NOM	UNITATS	COST	COST TOTAL
PC	1	300,00 €	300,00 €
Cable sèrie	1	2,50 €	2,50 €
U. Receptora	1	60,00 €	60,00 €
U. Emissora	40	75,00 €	3.000,00 €
Bateries Recarregables	40	2,00 €	80,00 €
TOTAL:			3.442,50 €

Taula 8 – Costos d'implementació del sistema

Així doncs partim d'una inversió inicial de 3442,50 euros per a instal·lar el sistema. Ja que el sistema funciona amb bateries i calculant el cost de funcionament, el sistema ens costa mantenir-lo el següent:

Renovació bateries recarregables:

$$40 \times 2 \text{ €} = 80 \text{ €/any}$$

Consum PC:

$$9 \text{ W} \times 10 \text{ hores} \times 0.083816 \text{ euros/kWh} \text{ durant el dia} \times 320 \text{ dies} = 2,41 \text{ €/any}$$

El manteniment del sistema ens surt per un cost de 82,41 € anuals aproximadament.

Suposem que existeix un menú diari i establím un benefici final sobre aquest menú, tenint en compte els costos de matèries primeres, preparació i sous, de 1,5 euros.

Si una taula és ocupada durant una hora i establím que es serveixen menús durant 6 hores (3 hores migdia i 3 hores nit), els beneficis són:

$$(40 \text{ Taules} \times 6 \text{ hores} \times 1,5 \text{ €}) / 1 \text{ Taula} \cdot \text{hora} = 360 \text{ € nets}$$

Ara considerem el sistema que el restaurant té instal·lat el sistema. Gràcies a aquesta instal·lació aconseguim reduir el temps de servei de les taules, cosa que repercuteix en el temps final que un client ocupa una taula. Això permet atendre més clients en el mateix espai de temps.

Així doncs, suposem que ara el temps mitja d'estada a cada taula és de 50 minuts.

$$50\text{min}/60\text{min}=0,83$$

$$(40 \text{ Taules} \times 6 \text{ hores} \times 1,5 \text{ €}) / 0,83 \text{ Taula} \cdot \text{hora} = 433,7 \text{ €}$$

Podem observar una clara diferència de 73,7 € de benefici per dia. Aquests increment és molt important.

Ja que el cost inicial del sistema oscil·la entre 3442,50 € i tenint en compte que cada dia es recaptin 73,7 € per damunt dels beneficis, es calcula que el sistema serà amortitzat en 48 dies de servei.

Hem pogut observar que a través d'una petita inversió per comprar el sistema podem incrementar en 73,7 €/dia aproximadament el benefici a partir de la setena setmana.

Aquests resultats ens asseguren el risc de la implementació del sistema.

7.3 Conclusions

Una vegada arribat al final del desenvolupament, hem de considerar si els objectius que es buscaven al plantejar el projecte han sigut correctament solucionats i si s'ha aconseguit un producte satisfactori, així com donar una sèrie de valoracions crítiques sobre el que finalment ha aconseguit aportat aquest projecte.

Per començar, es pot afirmar que tots els objectius descrits amb anterioritat durant la memòria s'han portat a terme i que s'han obtingut amb els resultats esperats. La visualització de missatges a les taules, la comunicació sense fils i el programa de gestió automatitzat són objectius que s'han plasmat en el projecte, obtenint un sistema que realitza un objectiu principal, ser capaç de portar la gestió dels estats de les taules d'un restaurant i les comunicacions entre els clients i el servei.

Un dels punts claus, el desenvolupament del programa en C# per a la gestió automatitzada, ha suposat un gran esforç a l'hora de ser implementat degut en part a la

necessitat d'aprendre un nou entorn de programació i una sèrie de noves utilitats noves fins ara. Igualment hi ha que dir, que aquest esforç ha sigut àmpliament recompensat amb els resultats obtinguts, trobant-me capaç de desenvolupar aplicacions totalment distintes, en qualsevol llenguatge de programació.

En aquest projecte buscava aprendre una sèrie de tecnologies noves, ja sigui a nivell d'iteració amb l'usuari, com de programació de hardware i software, cosa que he pogut aconseguir a través de la unió de la programació amb "PSoC" i amb la programació en Visual C#. Cal comentar que gracies a assignatures com Sistemes Digitals II durant el segon curs de carrera ja disposava de nocions de programació de dispositius programables però que va ser a través dels apunts i els consells del director del projecte on vaig poder aconseguir les nocions necessàries per poder dissenyar sistemes basats en "Systems on Chip" com el "PSoC".

Finalment, considero que aquest projecte m'ha aportat molts coneixements informàtics nous, així com habilitat i confiança en buscar informació necessària sobre llenguatges o sistemes que desconeixes i que has de treballar amb ells, cosa molt important en un bon enginyer informàtic.

CAPÍTOL VIII

BIBLIOGRAFIA

Llocs web:

www.cypress.com/psoc

<http://www.microsoft.com/express/windows>

Documents web:

[www.lawebdelprogramador.com/cursos/enlace.php?idp=3869&id=227&texto=C+sharp](http://www.lawebdelprogramador.com/cursos/enlace.php?idp=3869&id=227&texto=C%2Bsharp)

<http://msdn.microsoft.com/es-es/library/z9hsy596%28VS.80%29.aspx>

Documents Tècnics de Cypress:

Wireless-USB LP System Overview.pdf

Wireless-USB LP EVK Quick Start Guide.pdf

Wireless-USB LP EVK User's Guide.pdf

DataSheet_CY7C603xx.pdf

DataSheet_CYRF6936_C.pdf

C Language Compiler.pdf

LCD.pdf

Memòria del projecte realitzada per

**Alberto Carrete Geijo
Sabadell, a 29 de Juny del 2010**