

Trabajo Fin de Grado

Diseño electrónico de consolas deportivas basadas
en pantallas táctiles

Electronic design of sport consoles based on a
touch screen

Autor/es

Jorge Elías Martín García

Director/es

Javier Lasobras Bernad
Rafael Alonso Esteban

ESCUELA DE INGENIERÍA Y ARQUITECTURA
2016



DECLARACIÓN DE AUTORÍA Y ORIGINALIDAD

(Este documento debe acompañar al Trabajo Fin de Grado (TFG)/Trabajo Fin de Máster (TFM) cuando sea depositado para su evaluación).

TRABAJOS DE FIN DE GRADO / FIN DE MÁSTER

D./D^a. _____,

con nº de DNI _____ en aplicación de lo dispuesto en el art.

14 (Derechos de autor) del Acuerdo de 11 de septiembre de 2014, del Consejo de Gobierno, por el que se aprueba el Reglamento de los TFG y TFM de la Universidad de Zaragoza,

Declaro que el presente Trabajo de Fin de (Grado/Máster)
_____, (Título del Trabajo)

es de mi autoría y es original, no habiéndose utilizado fuente sin ser citada debidamente.

Zaragoza, _____

Fdo: _____

Diseño electrónico de consolas deportivas basadas en pantallas táctiles

En la actualidad, cualquier evento deportivo cuenta con un sistema de consolas y marcador electrónico que controla y ofrece los resultados deportivos de forma sencilla y fiable. Estos sistemas electrónicos han adquirido una gran importancia dentro del mundo deportivo y la evolución tecnológica de los mismos es constante, adaptándose a las nuevas tecnologías. En este trabajo se presenta la situación actual de los sistemas electrónicos de control y se desarrolla una evolución tecnológica del teclado y el display de las consolas electrónicas deportivas de control.

Actualmente, Mondo dispone de marcadores electrónicos específicos para cada uno de los deportes, y marcadores multideporte basados en pantallas led pensados para mostrar la información acorde con el evento deportivo que se está celebrando. Sin embargo, solamente se dispone de una única consola deportiva, cuya función es la introducción de los datos del evento deportivo a mostrar en los distintos marcadores. En esta consola el teclado es fijo, con una distribución de los botones diseñada inicialmente para un único deporte, baloncesto, lo que hace su uso poco intuitivo en otros deportes.

En el presente trabajo se desarrolla una pantalla táctil que sustituye el teclado actual de las consolas deportivas, permitiendo utilizarla en todos los eventos deportivos de forma intuitiva. Este desarrollo permite visualizar diferentes layouts según el evento deportivo seleccionado, dotando a la consola deportiva de gran flexibilidad y versatilidad.

El trabajo se distribuye de la siguiente forma; primero se describe el sistema actual de consolas-marcador con el que cuenta hoy en día Mondo de manera que se entienda mejor su funcionamiento. Posteriormente se detalla y motiva la solución tecnológica elegida para sustituir los teclados actuales, seleccionando el dispositivo disponible en el mercado que mejor se adapta a las necesidades. A continuación, se programa y testea para comprobar su viabilidad mediante el desarrollo de un software que simule el sistema real. Finalmente se implementa un prototipo funcional, sustituyendo el teclado actual por la pantalla táctil y finalmente se analizan las modificaciones hardware que se deberán realizarse en un futuro desarrollo en la consola deportiva actual para su implementación en un producto final.

CAP. 1	SISTEMAS DE CONTROL DE EVENTOS DEPORTIVOS UTILIZADOS EN LA	
	ACTUALIDAD	4
1.1.	Introducción	5
1.2.	Descripción del sistema de control utilizado por Mondo	6
1.3.	Principales cambios a realizar en los sistemas actuales	8
CAP. 2	INTRODUCCIÓN DE LA TECNOLOGÍA TÁCTIL EN LOS SISTEMAS	
	EXISTENTES HOY EN DÍA	11
2.1.	Introducción, enfoque y objetivos	12
2.2.	Tipos de pantallas táctiles y compatibilidad con el sistema actual	13
2.3.	Módulo de pantalla TFT FPEP104SA4-00R	15
2.4.	Limitaciones del módulo.....	20
CAP. 3	PROGRAMACIÓN DEL MÓDULO TFT, ANÁLISIS DE LAS COMUNICACIONES	
	Y SIMULACIÓN DEL SISTEMA MEDIANTE ORDENADOR	21
3.1.	Introducción	22
3.2.	Diseño gráfico y configuración del módulo de pantalla	22
3.3.	Control a través del PC del módulo de pantalla	28
3.4.	Software de simulación del marcador.....	32
3.5.	Simulación del sistema de marcador deportivo	33
CAP. 4	INTEGRACIÓN DE LA PANTALLA TFT EN LAS CONSOLAS ACTUALES	34
4.1.	Adaptación del Hardware.....	35
4.2.	Programación del microcontrolador de la consola central	37
CAP. 5	CONCLUSIONES Y LÍNEAS FUTURAS	39
5.1.	Conclusiones y líneas futuras.....	40
ANEXO A	PROTOCOLO DE LA EUROLIGA DE ENVÍO DE DATOS AL MARCADOR	41
ANEXO B	ESQUEMA DEL CÓDIGO DEL PROGRAMA.....	47

Capitulo 1

Sistemas de control de eventos deportivos utilizados en la actualidad

1.1. Introducción

En los últimos años los sistemas electrónicos han adquirido una gran importancia dentro del mundo deportivo. En cualquier evento deportivo de hoy en día se cuenta con un marcador electrónico donde se visualizan los resultados y estadísticas del partido. Previamente al uso de los marcadores dentro del ámbito deportivo, ya existían otros sistemas con los cuales visualizar y gestionar los datos del evento deportivo, pero estos sistemas permitían mostrar mucha menos información que los actuales, además de tener el inconveniente de ser controlados manualmente. Gracias a la evolución de la tecnología y las comunicaciones, ahora se cuenta con marcadores electrónicos capaces de mostrar una gran cantidad de información de un evento, los cuales pueden ser controlados a distancia, ya sea mediante comunicación por cable o inalámbrica, mediante unos dispositivos denominados consolas electrónicas deportivas, manejadas a su vez por una persona.

Una de las empresas que actualmente desarrolla y suministra este tipo de equipamientos, tanto marcadores electrónicos como consolas electrónicas, es Mondo Ibérica, empresa española creada en Zaragoza en 1988 dedicada a la fabricación de equipamiento deportivo filial de la multinacional Italiana Mondo, relacionada con el mundo del deporte desde su fundación en 1948.

En lo referente a electrónica deportiva, un marcador es un dispositivo electrónico formado por dígitos luminosos compuestos por diodos LED de muy alta luminosidad y gran angularidad, lo que permite que se vean desde cualquier punto de una instalación deportiva. Los diferentes dígitos que configuran el marcador se fabrican mediante placas de circuito impreso donde se alojan los leds que definen los segmentos. Los dígitos están en su parte anterior protegidos mediante metacrilatos. En estos marcadores se muestran todos los datos de relevancia del evento deportivo. Existen distintos modelos de marcadores para cada especialidad deportiva.

Por otro lado, las consolas deportivas son dispositivos electrónicos utilizados para el control de los marcadores electrónicos mediante comunicación por cable o inalámbrica. Estas consolas están compuestas por un teclado para introducir los datos y unos displays de 7 segmentos para visualizar los eventos introducidos. Éstas no se han

adaptado físicamente a lo largo del tiempo para cada especialidad deportiva, sino que cuentan con un teclado y displays con una distribución fija y pensada para gestionar inicialmente un solo deporte, baloncesto. El hecho de que la consola sea igual independientemente del evento deportivo, siendo el marcador diferente para cada uno de ellos, hace que la distribución actual del teclado sea poco intuitiva para el control del evento. En este primer capítulo se ve de forma resumida como son tanto las consolas como el marcador que Mondo usa hoy en día para posteriormente ver la alternativa a las primeras de manera que se resuelva la problemática en cuanto a ergonomía sin importar el deporte a gestionar.

1.2. Descripción del sistema de control utilizado por Mondo

Un sistema de control de un evento deportivo consta básicamente de uno o varios marcadores electrónicos, una o varias consolas de control y el conexionado entre todos los dispositivos. A continuación, se explica de manera individual los distintos dispositivos del sistema completo, el cual se ve en la Fig. 1.

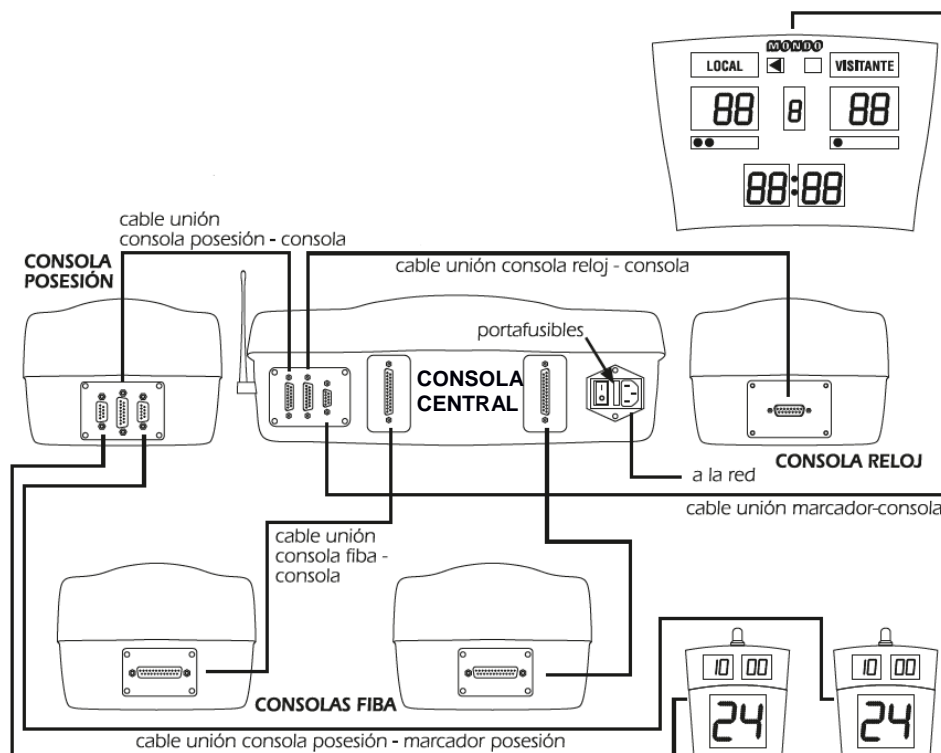


Figura 1. Sistema completo de consolas y marcador en el caso de baloncesto de alta competición



una precisión de centésimas de segundo. La consola de tiempo dispone, como la consola central, de un teclado fijo de membrana con 11 botones, con los que configurar las distintas variables del partido. Una vez pulsado un botón, se envía un dato por un pin de entrada/salida, el cual genera una interrupción en la consola central. A parte del teclado de membrana, la consola cuenta además con un pulsador mecánico rojo de inicio/parada, como se observa en la Fig. 4. Para la visualización cuenta con 5 displays de 7 segmentos, 4 para el tiempo y 1 para el periodo. En la parte trasera de la consola se ve el conector con el cual es conectada a la consola central, Fig. 5.



Figura 4. Parte frontal de la consola de tiempos



Figura 5. Parte posterior de la consola de tiempos

- Consolas auxiliares FIBA. Con unas características muy parecidas a la consola de tiempo, sirven para el control de forma individual de las variables en caso de alta competición.
- Consolas de posesión. Consola encargada de la gestión del tiempo de posesión en baloncesto.
- Marcador electrónico. Dispositivo utilizado para mostrar la información deportiva. Un modelo básico estándar se muestra en la Fig. 6, se trata de un marcador electrónico multideporte formado por cuatro módulos (uno central, uno inferior y dos laterales). Este marcador proporciona una clara y completa información de todos los parámetros relevantes del juego. Los dígitos luminosos están formados por diodos LED de muy alta luminosidad y gran angularidad, lo que permite que se vean desde cualquier punto de la instalación. Los diferentes

dígitos que configuran el marcador se fabrican mediante placas de circuito impreso donde se alojan los leds que definen los segmentos. Los displays están en su parte anterior protegidos mediante metacrilatos. El marcador se comunica con la consola central mediante el protocolo de comunicación serie RS-485, el cual se trata de un sistema de transmisión diferencial que permite el envío de datos a grandes distancias sin interferencias y además la conexión multipunto por lo que varios marcadores pueden recibir simultáneamente datos de una única consola central. La conexión física entre ellos se lleva a cabo mediante un cable apantallado, de 100 metros de longitud, con el objetivo de evitar el acople de ruido eléctrico. También existe la opción de una comunicación inalámbrica, a través de un enlace de radiofrecuencia en banda de uso común (433 MHz), y con un alcance de hasta 150 metros en visibilidad directa. Además de este marcador existen otros pensados específicamente para otros deportes.



Figura 6. Marcador electrónico multideporte

1.3. Principales cambios a realizar en los sistemas actuales

Como ya se ha comentado anteriormente, Mondo cuenta con distintos marcadores según el evento deportivo de manera que su visualización sea adecuada con las reglas del mismo, sin embargo, sólo cuenta con una consola central con una distribución de teclado y displays fija para el control de todos ellos, así pues, los cambios a realizar se centrarán sobre esta consola. Los componentes más importantes de

la consola, y los cuales son los causantes de su poca flexibilidad al cambiar entre deportes, son tanto el teclado de membrana como los displays de 7 segmentos y la pantalla LCD, su importancia reside en que suponen la conexión entre el mundo real y el sistema consolas-marcador y por tanto su sustitución conlleva el mayor cambio en el sistema, con lo que resulta de interés hablar, aunque sea de forma general, algo más sobre ellos.

El teclado de membrana se compone básicamente de dos circuitos separados por una capa con orificios en la posición donde se encuentran los botones como podemos ver en la Fig. 7. Cuando uno de los botones es pulsado, los dos circuitos se unen a través del orificio produciéndose conexión eléctrica y quedando registrado el lugar de la pulsación. Las consolas cuentan con este tipo de teclado debido a su gran durabilidad, bajo coste y porque además dan sensación real de pulsación al ser presionados. Su inconveniente reside en que se trata de un dispositivo el cual sigue una distribución fija y por tanto se debería diseñar uno para cada uno de los deportes y montarlos sobre diferentes consolas cada una especialmente diseñada para dicho deporte, todo esto con el alto coste que supone.

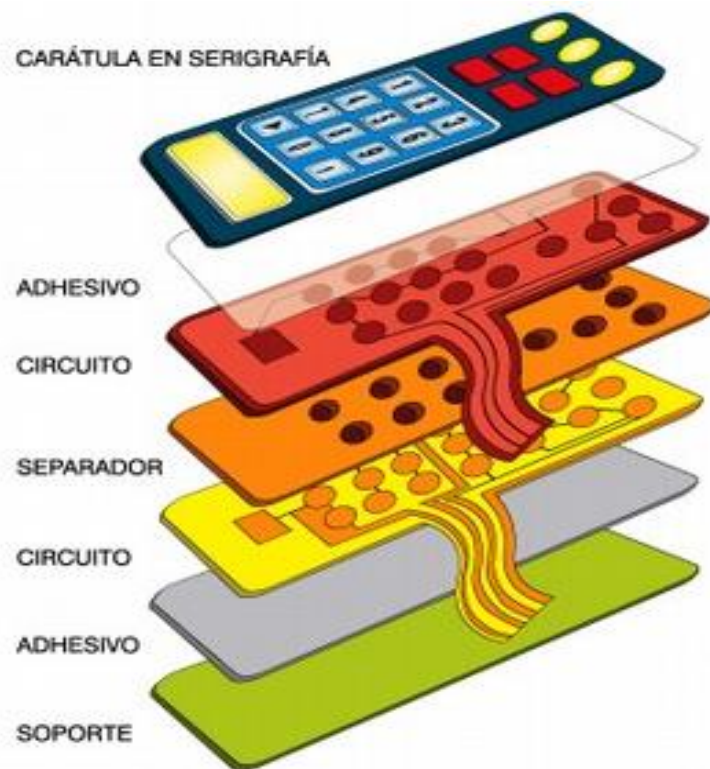


Figura 7. Diferentes capas del teclado de membrana usado en las consolas actuales

Los displays de 7 segmentos, como el de la Fig. 8, constan, como su nombre indica de 7 segmentos en forma de línea que se pueden encender y apagar de manera individual, pudiéndose representar en ellos los números del 0 al 9 además de otros caracteres. En el caso de la consola los displays están multiplexados a través de un driver con el cual se conectan al microcontrolador, ya que, si la conexión se hiciera de manera directa, no habría pines suficientes para la conexión de todos ellos. Puesto que en estos displays la información a mostrar es muy limitada, la consola cuenta con una pantalla LCD como la de la Fig. 9, la cual se puede configurar para mostrar cualquier carácter y que se usa, como se ha dicho anteriormente, para mostrar el nombre del deporte el cual se va a controlar, su inconveniente es que, aunque permite mostrar cualquier carácter su tamaño no permite mostrar mucha información.



Figura 8. Visualizador de 7 segmentos



Figura 9. Pantalla LCD usada para la visualización del deporte

Así pues, una vez conocidos estos componentes, nos damos cuenta de la poca flexibilidad que estos otorgan a la consola. En los siguientes capítulos se propone una solución capaz de hacer frente a este problema de manera económica y sencilla para el usuario a través de la sustitución por una pantalla táctil de todos los componentes nombrados en esta sección. Esta pantalla deberá hacer por tanto el trabajo tanto del teclado de membrana como el de los displays, haciendo así además prescindibles componentes como el chip de teclado y el driver para los displays.

Capitulo 2

Introducción de la tecnología táctil en los sistemas existentes en la actualidad

2.1. Introducción, enfoque y objetivos

En este capítulo se estudiarán las posibles alternativas para la sustitución del teclado y los displays por una pantalla táctil programable, para hacer frente al problema de la poca flexibilidad a la hora de gestionar diferentes deportes de las consolas de Mondo de hoy en día. El capítulo consta de dos partes; en la primera de ellas se estudian las pantallas táctiles existentes hoy en día en el mercado, estudiando su validez dentro del proyecto, y posteriormente se detalla en profundidad el dispositivo elegido explicando los motivos por los cuales se ha seleccionado esta alternativa.

La sustitución del teclado y displays por una pantalla táctil es una modificación del sistema de control muy importante, por lo que el desarrollo de este proyecto debe realizarse en varias fases. En primer lugar, se realizará una simulación del sistema con la utilización de la pantalla táctil a utilizar, simulando con un PC tanto la gestión de la consola de control como el marcador electrónico. En segundo lugar, y una vez validada la solución, se llevará a cabo la modificación física de la consola para comprobar su funcionamiento una vez incluida la pantalla dentro del sistema real.

Por tanto, de acuerdo con el enfoque dado al proyecto, los objetivos a cumplir serán, una vez seleccionada la pantalla:

- En primer lugar, la programación de la pantalla táctil de manera que sea capaz modificar su configuración en función del evento a gestionar, y haga de manera correcta tanto su trabajo como teclado como de dispositivo de visualización, ambos de una forma más cómoda para el usuario.
- En segundo lugar, puesto que la pantalla actuará como un periférico dentro de la consola, la conexión de la misma a un ordenador de manera que podamos observar que es capaz de comunicarse de forma correcta con él, tanto al enviar como al recibir datos.
- Por último, la sustitución del teclado y displays por la pantalla táctil de manera que esto afecte lo menos posible a los demás componentes que lleva la consola.

2.2. Tipos de pantallas táctiles y compatibilidad con el sistema actual

Hoy en día, existen gran cantidad de tipos de pantallas táctiles, dependiendo de su tecnología, tamaño, peso, comunicación, software etc. Por tanto, el primer paso a seguir para la elección de la pantalla que se utilizará en el proyecto, es definir los requisitos necesarios para saber que dispositivo es el que mejor se adapta. La principal función de la pantalla es actuar como periférico dentro del sistema, actuando como teclado y mostrando información, por lo que deberá ir comunicarse con el microcontrolador de la consola, que hace la función de control. Por lo tanto, la pantalla seleccionada deberá incluir un protocolo de comunicaciones. Por otro lado, dado que debe realizar las funciones de teclado y display, la pantalla debe ser lo suficientemente grande para poder ser manejada de manera cómoda pero inferior al tamaño de la parte frontal de la consola actual para poder adaptarla. El último aspecto a definir es cuál de las tecnologías táctiles más comunes se adapta mejor a nuestra aplicación, por lo que a continuación se habla de las más importantes existentes hoy en día y sus pros y contras dentro del ámbito del proyecto:

- Pantallas táctiles resistivas. Las pantallas que cuentan con esta tecnología, están formadas por dos capas transparentes y flexibles, recubiertas por un material resistivo, la capa superior recubierta en su superficie inferior y la capa inferior recubierta en su parte superior, ambas capas separadas por aire. A una de las capas se le aplica una tensión y cuando la capa superior es presionada toca la inferior haciendo que la corriente pase. Gracias a una trama de líneas horizontales y verticales situadas en las capas que actúan como divisor resistivo se conoce la posición donde se ha pulsado, como podemos ver en la Fig. 10. Sus ventajas para el proyecto residen en que pueden ser pulsadas con cualquier objeto que ejerza presión en la capa superior, en su bajo precio, en su alta resistencia a los parámetros ambientales y su bajo consumo. Su mayor contra es que la nitidez de la imagen es más baja que con otras tecnologías.

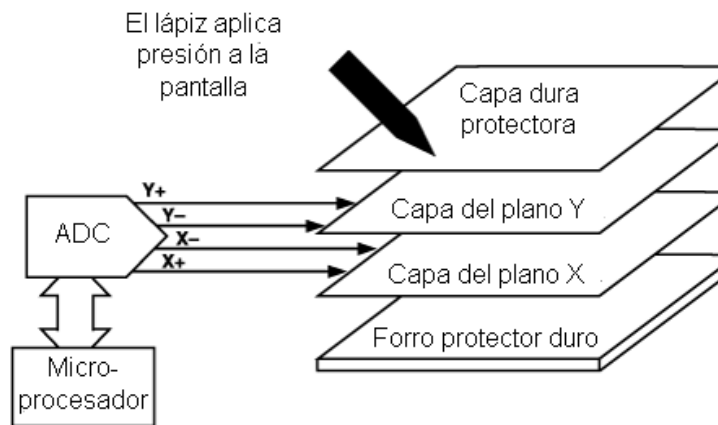


Figura 10. Pantalla táctil con tecnología resistiva

- Pantallas táctiles capacitivas. Se basan en que el cuerpo humano es un conductor eléctrico. Estas pantallas están formadas por un panel aislante recubierto de un conductor transparente, cuando dicho panel es presionado con el dedo se produce un cambio en el campo electrostático que puede ser medido como un cambio de capacitancia, el cual lee un procesador y computa la posición como vemos en la Fig. 11. Sus mayores ventajas son la muy alta nitidez y repetitividad que ofrece, sobretodo esta primera respecto a las pantallas resistivas mientras que su mayor desventaja reside en su alto coste.

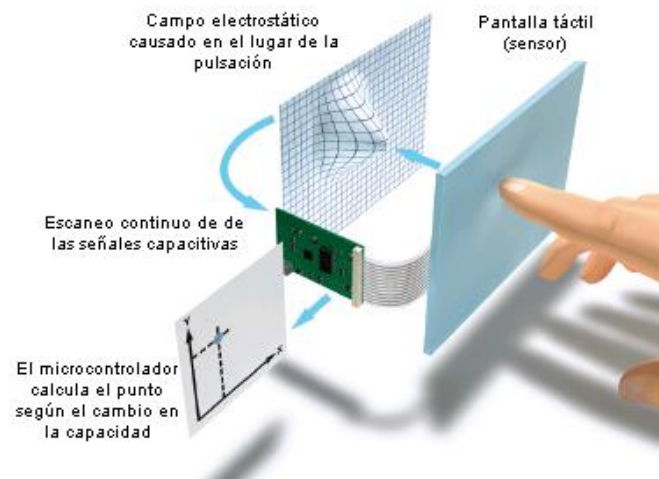


Figura 11. Pantalla táctil con tecnología capacitiva

- SAW (Onda acústica de superficie). Se basan en ondas sonoras las cuales viajan por la superficie de un material elástico, las pantallas están formadas por un panel de cristal con emisores de ultrasonidos los cuales son recibidos al otro lado de la pantalla. Cuando se pulsa una parte de dicho panel, parte de la onda es absorbida y por tanto no recibida al otro lado de la pantalla. Gracias a esto los sensores pueden calcular la posición donde se pulsó. Al igual que las pantallas capacitivas tienen una alta nitidez de imagen además de una alta vida útil, pero es mucho más sensible al ambiente, tanto en lo que a partículas sólidas como líquidos se refiere, además su precio también es mayor al de las resistivas.

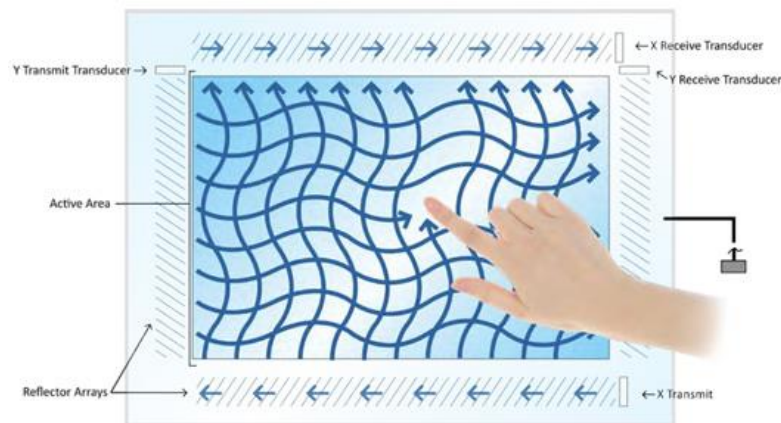


Figura 12. Pantalla táctil con tecnología SAW

De las tres características a observar para la elección de la pantalla táctil, las cuales son el tamaño, la disponibilidad de un módulo de comunicaciones y la tecnología táctil, ya se ha especificado como deben ser las dos primeras al comienzo de este punto, mientras que, en cuanto a tecnología táctil se refiere, la que más ventajas tiene para el proyecto es la resistiva puesto que su único aspecto negativo es que ofrece poca nitidez, pero esto no supone un gran inconveniente para el proyecto al no tener que ser mostradas imágenes o videos de gran calidad y además es significativamente más barata que las otras dos tecnologías.

2.3. Módulo de pantalla TFT FP104SIA24-00R

El módulo de pantalla TFT FP104SIA24-00R fabricado por *FDT corporation*, cuenta con una pantalla táctil resistiva de 10.4'' como la de la Fig. 13, integra un protocolo de comunicación serie RS-232 y además por la forma de su carcasa está preparado para su integración dentro de otro sistema. La elección de este módulo se debe principalmente a que en el mercado no existen muchas pantallas con el tamaño de 10.4'' el cual se adapta bien a la consola actual.



Figura 13. Módulo de pantalla TFT FP104SIA24-00R de 10.4 pulgadas

A parte de las características generales, las características a tener en cuenta en el proyecto a lo que visualización se refiere son: en primer lugar que la pantalla tiene un formato de display de 800 x 600 (RGB), lo cual indica una resolución de 800 x 600 píxeles y visualización a color RGB. Esta resolución limita la calidad de imágenes o iconos a utilizar ya que aunque se usen imágenes de gran calidad estas se observarán a la misma resolución que la de la pantalla; la segunda característica es el ángulo de visión de la pantalla, en este caso es diferente dependiendo de la dirección desde la que se observe, estos se observan en la Fig. 14, donde vemos que el ángulo tiene unos valores típicos bastante pequeños en todas las direcciones con lo que la visualización de la pantalla se hará dificultosa si no se mira a ella de manera perpendicular; la última

característica es que la pantalla cuenta con retroiluminación de manera que el brillo puede ser ajustado en función de las condiciones externas.

Parámetro	Símbolo	Condición	Mínimo	Típico	Unidades
Ángulo de visión	Horizontal	Izquierda	60	70	Grados
		Derecha	60	70	Grados
	Vertical	Arriba	40	50	Grados
		Abajo	50	60	Grados

Figura 14. Ángulo de visión para una ratio de contraste menor a 10

Siendo la ratio de contraste la luminosidad producida por el color más brillante y la producida por el menos brillante, es decir, la luminosidad del color blanco dividida entre la luminosidad del color negro emitidos por la pantalla.

Respecto al protocolo de comunicaciones serie RS-232, este tiene una tasa máxima de 921600 baudios por segundos la cual puede ser configurada durante la programación de la pantalla. El módulo trabaja entre las tensiones de 0 y 3.3V lo cual habrá que tener en cuenta posteriormente al conectarlo en la consola central.

Finalmente, el módulo cuenta con su propio Software de programación de alto nivel llamado *Arranger+*, el cual como veremos en la parte de implementación está pensado para la inclusión de botones, iconos y la alternancia entre diferentes pantallas. Además de permitirnos modificar aspectos del hardware como son el brillo de la pantalla o la velocidad de transmisión.

2.5. Limitaciones

Se pueden diferenciar dos tipos de limitaciones debidas a la pantalla, las producidas por el hardware y las producidas por el software.

Las limitaciones producidas por el Hardware tienen relación con la velocidad de respuesta y la comodidad a la hora de manejar la pantalla, esto se debe a que al tratarse de un módulo con tecnología resistiva, la presión a ejercer sobre la pantalla para

que la pulsación quede registrada es mayor a la presión a realizar sobre una pantalla capacitiva, que es la usada por ejemplo en telefonía móvil. Además el módulo no dispone de vibración por lo que se puede llegar a la confusión entre si realmente la pantalla ha recibido nuestra pulsación o no, sin embargo, el módulo incorpora un pequeño buzzer el cual puede ser configurado para emitir sonido de 10 a 1000 ms cuando se pulsa un botón, de manera que obtengamos algún tipo de respuesta de la pantalla cada vez que un botón ha sido realmente pulsado. La otra limitación en lo que a hardware se refiere es el ángulo de visión, y es que éste sólo permite ver la pantalla de manera correcta si la persona la observa de manera perpendicular a la misma, al igual que antes, esto no supone una desventaja real para el proyecto ya que la pantalla estará en una posición fija en la consola perpendicular a nuestra vista.

En cuanto a las limitaciones producidas por el software, éstas residen en que al incorporar un lenguaje de alto nivel el grado de modificación es muy bajo, permitiendo tan solo, en lo que a este trabajo concierne, enviar comandos por medio de la UART al ordenador cuando se pulsa un botón y cargar iconos desde la memoria interna e ir alternando entre pantallas, por lo que toda la programación debe realizarse en el software de simulación de la consola o en el microcontrolador de la consola en el caso real. Puesto que la pantalla sólo debe actuar como periférico de la consola y todo el control se lleva a cabo dentro del microcontrolador, de nuevo no resulta una limitación real.

Capitulo 3

Programación del módulo TFT, análisis de las comunicaciones y simulación del sistema mediante ordenador

3.1. Introducción

Con los objetivos del trabajo ya fijados y la pantalla TFT táctil seleccionada, en este capítulo se desarrolla la programación de la pantalla táctil y el software del ordenador con el cual se controlará. Estos dos dispositivos formarán el sistema de test necesario para probar la viabilidad del sistema.

Por un lado, la programación de la pantalla TFT táctil consistirá en el diseño gráfico, con el Software *Arranger+*, de las pantallas que se visualizarán en ella. Y por otro lado, en el ordenador se realizará un programa específico, en *Visual Studio*, que permitirá la comunicación entre los dos dispositivos.

El diseño gráfico de la pantalla y la programación en el ordenador se realizan de forma simultánea. Trabajar de manera simultánea facilita la detección de errores, permitiendo además añadir nuevas funcionalidades no especificadas en un primer momento.

3.2. Diseño gráfico y configuración del módulo de pantalla

El diseño gráfico de la pantalla TFT táctil se realizará mediante *Arranger+*, programa de desarrollo de interfaz gráfica de usuario con la capacidad de integrar comandos UART. Con *Arranger+* podremos crear distintas pantallas entre las cuales alternar; crear botones que generen comandos UART para comunicarnos con el ordenador y por último cargar imágenes o iconos en las distintas pantallas.

Antes de explicar cómo se ha creado la interfaz gráfica es necesario hablar de los comandos UART que es capaz de procesar la pantalla TFT táctil, ya que gracias a ellos podremos controlarla a través del ordenador. Con estos comandos el ordenador podrá saber qué orden se le ha dado desde la pantalla TFT táctil y a su vez responder en consecuencia. Los comandos que se envían y reciben en la pantalla tienen la siguiente forma:

- Código de inicio: de 0 a 4 bytes, se mantiene por defecto a 1 byte (0xAF).
- Código de longitud de datos: 1 byte fijo (Longitud comando + longitud dato).
- Orden: 1 byte fijo (0x00 a 0xFA), dice a la pantalla como interpretar el dato.
- Dato: de 0 a 254 bytes, en nuestro caso se usarán como máximo 8 bytes.
- Código de final: de 0 a 4 bytes, se mantiene por defecto a 1 byte (0xEF).

Además, para que los datos sean leídos y enviados de manera que ambos dispositivos se entiendan deberemos configurar la UART de la pantalla TFT táctil con los mismos parámetros que el puerto serie del ordenador. Para ello Arranger+ nos permite modificar algunos de esos parámetros:

- Tasa de baudios: de 4800 a 921600 baudios por segundos
- Byte de inicio de trama y byte de fin de trama
- Número de bits por carácter: fijo a 8 bits
- Bits de parada: fijo a 1 bit
- Paridad: fijo a sin paridad

Una vez analizada la UART el siguiente paso es explicar la creación del interfaz gráfico y para entender la forma en la cual se ha realizado la programación es necesario conocer cómo funciona Arranger+. El programa organiza los elementos que se verán en la pantalla en tres capas diferentes, las cuales tienen un orden de prioridad de visualización. Las tres capas son la de elementos, la de lienzo y la de fondo. La capa con mayor prioridad es la de elementos donde se situarán los botones, después la de lienzo en la cual se visualizarán los iconos y finalmente el fondo. La capa de elementos es fija, es decir, una vez cargado el programa y encendida la pantalla no se podrá añadir ni borrar ningún botón u otro elemento situado en ella. La siguiente capa es la de lienzo donde podremos añadir y borrar iconos incluso en tiempo de ejecución. Por último, los elementos en la capa de fondo son fijos y se sitúan siempre detrás de los demás iconos o botones.

Una vez conocido el modo en el cual se visualizan los elementos en la pantalla, el primer paso a realizar es la creación de una pantalla principal desde donde podamos acceder a las distintas pantallas de cada uno de los deportes. *Arranger+* permite asignar un fondo a cada una de las pantallas, como vemos en la Fig.15 y después colocar y personalizar cada uno de los botones que irán en ella.

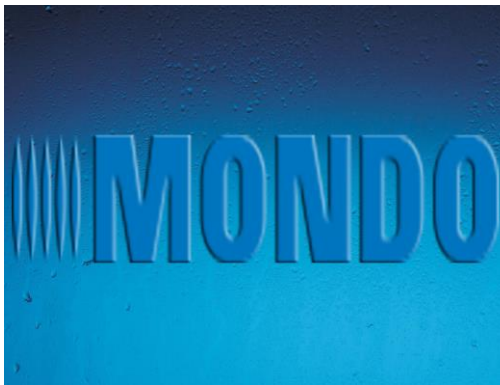


Figura 15. Fondo de pantalla de la pantalla principal

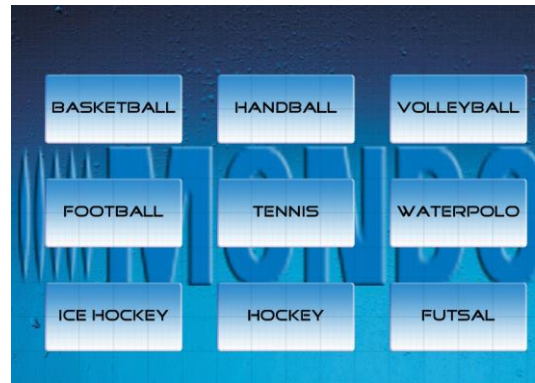


Figura 16. Aspecto final de la pantalla principal de la TFT

Como se ve en la Fig.16, en los botones se pueden cargar imágenes para identificarlos, una para su estado sin pulsar y otra para su estado pulsado, de manera que se vea de forma visual que la pulsación se ha llevado a cabo. Estas imágenes al formar parte del botón se sitúan en la capa de elementos por lo que permanecerán siempre visibles por encima de cualquier otro elemento de la pantalla. Así como los botones tienen siempre forma cuadrada o rectangular, la imagen situada en él puede tener cualquier forma, de manera que se vea que el botón es cuadrado, redondo o incluso transparente. En el caso de ser transparente, las imágenes situadas en capas inferiores se verán a través del botón. Todas las imágenes han sido creadas específicamente para el trabajo mediante la herramienta de diseño *Adobe Photoshop*.

Una vez creados se debe configurar cada uno de los botones de manera individual, esto incluye su posición en la pantalla y otros parámetros que el software permite modificar y que podemos ver en las siguientes tablas:

PARÁMETRO	OPCIÓN 1	OPCIÓN 2	OPCIÓN 3
Acción	Cambiar pantalla	Ninguna	
Activación	Al pulsar	Al soltar	
Área de activación	Toda el área	Área visible	
Grafismo sin presionar	Imagen a seleccionar	Sin imagen	
Grafismo presionado	Imagen a seleccionar	Sin imagen	
Código UART	De 0000 a FFFF		
Envío del código	Al pulsar	Al soltar	En intervalos
Intervalo de respuesta	De 20 ms a 5120 ms		

Tabla 1. Parámetros de configuración del botón, donde las opciones en negrita se refieren a parámetros comunes a todos los botones

PARÁMETRO	SIGNIFICADO
Acción	Configura el botón de manera que mediante su pulsación se cambie de pantalla o no.
Activación	Configura el botón de manera que la acción asignada a él se realice al pulsarlo o al soltarlo.
Área de activación	En el caso en el que la forma de la imagen no coincida con la forma del botón, permite configurar si el área efectiva de pulsación es la del botón o la de la imagen asignada a él. En el caso en el que la forma de la imagen sea rectangular y por tanto coincidente con la forma del botón la opción elegida será indiferente.
Grafismo sin presionar	Imagen asignada al botón cuando este no está pulsado.
Grafismo presionado	Imagen asignada al botón cuando este está pulsado.
Código UART	Código que envía la UART de la pantalla TFT táctil al ordenador de manera que este sepa que botón ha sido pulsado, distintos botones pueden tener el mismo código UART, pero un mismo botón no puede tener más de un código UART asignado.
Envío del código	Configura el botón de manera que el envío del código asignado a él se realice al pulsarlo o al soltarlo.
Intervalo de respuesta	En el caso en el que el envío del código se realice por intervalos, fija el intervalo en el cual se realizará cada uno de dichos envíos mientras el botón permanezca pulsado.

Tabla 2. Explicación de cada uno de los parámetros de configuración del botón

En la pantalla principal la acción de cada uno de los botones es tanto la de cambiar pantalla como la de enviar el código UART asignado a la pantalla seleccionada de manera que el ordenador sepa que deporte es el seleccionado. En general, cada uno de los botones de cualquier pantalla tiene su propio comando UART salvo los de vuelta

a la pantalla principal, situados en cada una de las pantallas de deporte, ya que todos tienen la misma función.

Una vez visto cómo funcionan los botones y la pantalla principal, se debe configurar cada una de las pantallas específicas para cada uno de los deportes. En el trabajo, la pantalla configurada de forma completa será la del **Baloncesto**, mientras que en las otras se incluirá un esquema de botones de manera que se vea que la pantalla TFT táctil es realmente capaz de alternar entre diferentes pantallas y por tanto válida para el proyecto final.

El control del baloncesto se compone de dos pantallas, una principal dentro del propio deporte y otra con la cual se configuran los dorsales de los jugadores, la primera sigue una configuración como la mostrada en las Fig. 17 y 18



Figura 17. Pantalla de baloncesto dentro del programa Arranger+



Figura 18. Pantalla de baloncesto siendo vista en la pantalla TFT táctil

Como vemos, las dos imágenes coinciden en algunos puntos como pueden ser los botones de puntuación o faltas mientras que difieren en puntos como las zonas de local y visitante o el rectángulo rojo debajo del título “basketball”. Estas diferencias se deben a 3 características principales, que son:

- Botones con imagen fija. Estos botones se ven de igual manera en el software y en la pantalla TFT táctil, los botones que siguen esta configuración tienen pulsación simple y sirven para incrementar o disminuir un dato, parar o poner en marcha el tiempo o cambiar de

pantalla. En las Fig. 19 y 20 podemos ver el cambio que se realiza en la imagen asignada a uno de estos botones para dar sensación de pulsación.



Figura 19: Botón sin pulsar

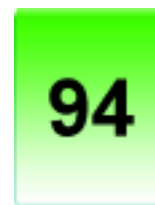


Figura 20: Botón pulsado

- Botones sin imagen. las partes que en la Fig. 17 están representadas como un rectángulo con una cruz roja son botones los cuales no tienen ninguna imagen asignada y por tanto transparentes. El hacerlos transparentes nos permitirá hacerlos funcionar con dos características que *Arranger+* no permite realizar. La primera característica será que esos botones tendrán retención y el hecho de estar activados o no conllevará una acción diferente ante la recepción del mismo código UART en el ordenador. La segunda característica será que su imagen no deberá ser siempre la misma. Para que estas dos características sean visibles en la pantalla TFT táctil una vez sea conectada al ordenador, se trabajará en la capa de lienzo. En esta capa podremos cargar y borrar en tiempo de ejecución los iconos que harán de imagen del botón. Si desde la pantalla TFT táctil se envía un código UART que implique la modificación de un botón, el ordenador responderá con otro código UART que indique la posición y el icono a cargar o borrar.



*Figura 21: Icono del dorsal
94 sin activar*



*Figura 22: Icono del dorsal
94 activado*

- Zona de visualización. En la Fig.17 el rectángulo rojo es el espacio asignado para la función de visualización dentro de la pantalla, en él se cargan y borran exclusivamente iconos que dan información a quien maneja la pantalla, en el software no se visualiza nada puesto que el

contenido es variable mientras que en la pantalla TFT táctil, Fig. 18, se observa la configuración inicial o de reset con todos los datos a 0.

Arranger+ cuenta con una librería donde se guardan los iconos que luego se controlarán desde el ordenador y una vez se carga la interfaz gráfica en la pantalla TFT táctil, la librería queda grabada en la memoria flash. Cada uno de los iconos tiene un identificador diferente que va de 0000 a FFFF y esto permite cargarlos de manera individual en función del botón que haya sido pulsado. *Arranger+* no permite modificar la escala de los iconos en tiempo de ejecución por lo que tendremos que crear, por ejemplo, diferentes números para la puntuación global que para el conteo de faltas o tiempos muertos al estar estos en una escala diferente en la pantalla.

La segunda pantalla, a la cual se accede desde el botón **Dorsales** de la pantalla de baloncesto, nos permite modificar cada uno de los dorsales locales y visitantes para después cargarlos en la pantalla principal, en la Fig. 23 vemos la pantalla como se ve en *Arranger+* y en la Fig. 24 como se ve en la pantalla TFT táctil.

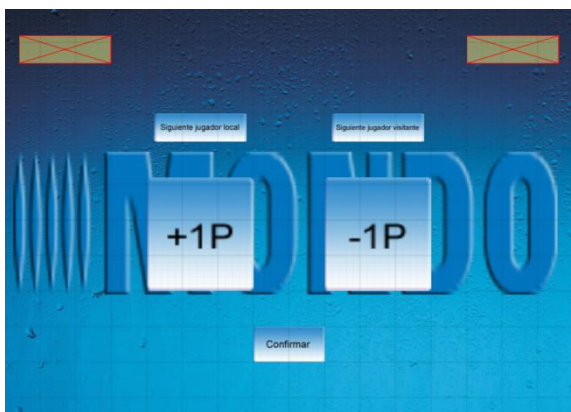


Figura 23. Pantalla de configuración de dorsales en Arranger+

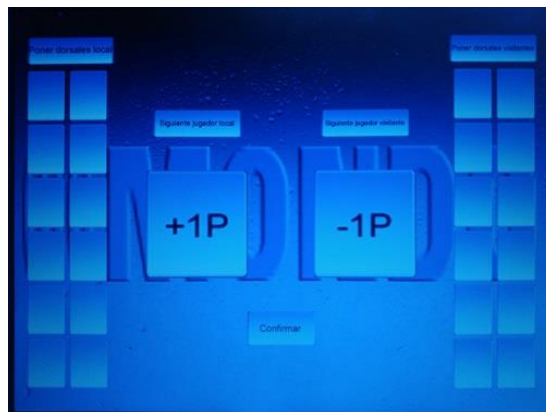


Figura 24. Pantalla de configuración de dorsales en la pantalla TFT táctil

Igual que en la pantalla principal de baloncesto, tenemos dos botones transparentes situados en la parte superior los cuales tienen retención y que permiten cambiar entre seleccionar jugadores locales y visitantes. Además hay 24 posiciones, 12 locales y 12 visitantes, en las cuales se visualizan los iconos con el dorsal del jugador y se configuran con los botones centrales, una vez puestos todos los dorsales de manera correcta se vuelve a la pantalla principal de baloncesto mediante el botón confirmar.

El último paso es la carga del programa en la pantalla. Esto se lleva a cabo mediante la carga de un archivo .BIN el cual la pantalla lee a través de una tarjeta externa microSD. Dicho archivo se genera dentro de Arranger+, se introduce en la microSD y posteriormente la tarjeta en la pantalla. Una vez cargados los datos se retira la tarjeta y se reinicia la pantalla la cual queda configurada de forma correcta.

3.3. Control a través del PC del módulo de pantalla

El control de la pantalla TFT táctil por ordenador se realizará a través de un programa específico. Este programa será creado mediante el lenguaje de programación *Visual Basic* dentro de la herramienta de desarrollo integrado para sistemas operativos Windows *Microsoft Visual Studio*.

La función del programa será, por un lado, la de recibir el comando UART que le llegue desde la pantalla, procesarlo y mandar un comando de vuelta a la pantalla. Y, por otro lado, enviar un comando al software que simula el marcador electrónico de manera que se modifique de acuerdo con el botón pulsado en la pantalla. Para esto el programa tendrá que leer y escribir el puerto serie con el que se comunicará con la pantalla y escribir en otro puerto serie la información correspondiente al marcador.

Para crear el programa el primer paso es diseñar una interfaz gráfica. En esta interfaz podremos seleccionar el puerto de comunicación con la pantalla TFT táctil y el puerto de comunicación con el marcador. Para seleccionarlos, se genera una lista en función de los puertos que estén abiertos en el momento en el que se lanza el programa y se elige clicando en el que se quiera usar. Con el objetivo de ver que los comandos se reciben y envían de forma correcta la interfaz tiene dos cuadros de texto. Para ver si nos encontramos conectados o desconectados la interfaz tiene una etiqueta situada debajo del cuadro de texto superior donde se indica el estado. Y por último tenemos tres botones, uno para conectarnos y otro desconectarnos del puerto, el botón CONECTAR se usa para empezar a leer el puerto sólo una vez que hayamos seleccionado que puertos queremos leer y escribir. El botón DESCONECTAR se usa para evitar generar excepciones a la hora de desconectarnos del programa, y por último el botón ENVIAR nos permite enviar comandos, o bien a la pantalla o al Software que simula el marcador.

Además, aunque no sean visibles en la interfaz, el programa también incluye dos módulos de conexión a puertos serie y un módulo de reloj. La interfaz del programa la vemos en la Fig. 25.

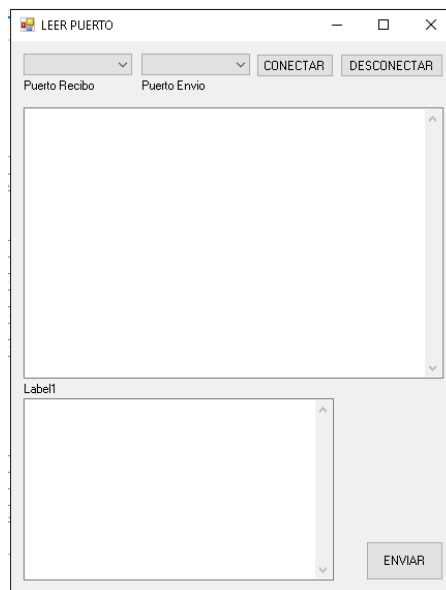


Figura 25. Interfaz gráfica del programa de control

Una vez explicada la interfaz, veremos a continuación los 4 bloques más importantes del programa de control, con los cuales se gestionan los datos recibidos desde la pantalla TFT táctil:

- Adquisición de datos. La trama de datos que envía la pantalla al ordenador cuando se pulsa un botón tiene 6 bytes, así que cuando llega un dato nuevo al puerto de entrada, el programa lee los 6 últimos bytes presentes en él y los guarda en una variable. Como todos los comandos que llegan al ordenador son generados por botones, los bytes de código de inicio (0), código de longitud de datos (1), orden (2) y código final (5) son iguales. Y los bytes (3) y (4) referentes al dato, codifican que botón ha sido pulsado. Para asegurarnos de que los bytes que estamos leyendo son los correctos se realiza un procesamiento del comando completo, comparando cada byte con el valor esperado. En el caso de no coincidir los valores leídos del puerto y los valores fijos, se mira en qué posición ha aparecido cada uno de los valores fijos y se mueven los valores variables a la posición 4 y 5. Esto se

ve de manera más clara en los siguientes ejemplos los cuales se explican de manera individual a continuación:

	byte(0)	byte(1)	byte(2)	byte(3)	byte(4)	byte(5)
DATO ESPERADO	175	3	225	valor 1	valor 2	239

DATO INCORRECTO 1	valor 1	valor 2	239	0	0	0
DATO MODIFICADO	valor 1	valor 2	239	valor 1	valor 2	0

DATO INCORRECTO 2	3	225	valor 1	valor 2	239	0
DATO MODIFICADO	3	225	0	valor 1	valor 2	0

DATO INCORRECTO 3	175	3	225	0	0	0
DATO MODIFICADO	-	-	-	-	-	-

El programa compara los bytes (0) y (5) del dato leído con los valores esperados 175 y 239, si ambos valores coinciden el dato no necesita procesarse. Si los valores no coinciden, el dato pasa por una serie de comparaciones, hasta encontrar en qué posición se encuentran el valor 1 y 2. En el 2º caso el dato esta movido tres posiciones a la izquierda y eso se sabe porque el valor 239 aparece en la 3ª posición y los siguientes son ceros. Este desplazamiento hace que los valores del byte (3) y el byte (4) aparezcan en distinta posición así que el programa copia el byte (0) en el byte (3) y el byte (1) en el byte (4). El caso 3 es similar al 2, pero los datos están desplazados solo una posición a la izquierda por lo que el procesado se lleva a cabo de la misma manera. Sin embargo, en el caso 4 solo llegan al puerto los tres primeros bytes seguidos de ceros. En este caso los valores variables no aparecen haciendo imposible su procesado por lo que este tipo de datos se desecha. Existen otros casos similares al 2 y al 3 dependiendo de la cantidad de posiciones que se desplaza el dato, todos ellos procesables de la misma manera.

- Análisis del dato y acción a realizar. Una vez tenemos el dato de manera que los bytes (3) y (4) sean los correspondientes al botón pulsado debemos actuar en consecuencia a dicha pulsación. Esto se consigue mediante la comparación del dato con todas las combinaciones posibles de byte (3) y byte (4) además de con otras variables que pueden estar activadas en función de si se han pulsado anteriormente otros botones o del devenir del partido. Igual que antes esto se ve de manera más clara con un ejemplo; cuando un jugador anota una canasta de 2 puntos, tanto la anotación de su equipo como la suya propia deben incrementarse y esta información debe verse tanto en la pantalla como en el marcador electrónico. El programa actúa de la siguiente manera:

- En la pantalla TFT táctil se selecciona el jugador que ha anotado y se envía un comando al ordenador. El programa mira si otro jugador estaba seleccionado en cuyo caso pone su icono a azul y pone el icono del jugador seleccionado a verde.
- En la pantalla se pulsa el botón de +2 puntos por lo que se envía el código del botón al ordenador y el programa lo lee.
- Los bytes (3) y (4) del dato procesado corresponden a los valores 0 y 18 respectivamente por lo que se entra en la rutina que suma dos puntos.
- Se mira si el botón **SONAR**, el cual hace que haya una breve vibración cada vez que se pulsa un botón, está activado o no y se actúa en consecuencia.
- Se mira si el jugador seleccionado es válido y después si es local o visitante.
- Se modifica su puntuación personal y se borran los iconos anteriores de la posición en la cual se muestran las estadísticas individuales y se cargan los nuevos iconos correspondientes en la pantalla, los cuales indican a que equipo pertenece, su número y su puntuación actual. Todo esto habiendo previamente comparado si sus faltas indican que no está expulsado, a fin de evitar errores.
- Se modifica la puntuación del equipo y al igual que antes se borran los iconos viejos y cargan los nuevos en la posición

- correspondiente de la pantalla, además se envía junto con la posición individual al software del marcador electrónico.
- Se borra el dato completo para evitar fallos
-
- Envío de datos. El programa está intercambiando datos continuamente tanto con la pantalla TFT táctil como con el software del marcador electrónico, cada uno recibe y lee los datos de una manera diferente por lo que estos se deben enviar en consecuencia.
 - Datos enviados a la pantalla TFT táctil. Los comandos que se envían a la pantalla TFT táctil son las instrucciones de borrado y carga de iconos en función de la propia pulsación realizada y la instrucción opcional de que vibre al ser pulsada. Estas tramas son de 12, 10 y 5 bytes respectivamente, las cuales siguen la forma explicada en el apartado 3.2. Para su envío, el dato se guarda en un búfer de bytes el cual se carga posteriormente en el puerto serie.
 - Datos enviados al marcador electrónico. Los datos que se envían al software que simula el marcador electrónico tienen una longitud de 4 bytes: el byte(0) es un código de inicio, el byte(1) le dice al programa como interpretar el dato, el byte(2) es el dato y el byte(3) es un código de finalización. Al tener todos los datos la misma longitud el envío se realiza en una rutina a parte en la cual, una vez se tiene el dato completo a enviar, se escribe en cuatro pasos en el registro del puerto, enviándose en cada paso un byte. Tanto el byte de dirección como el byte de dato vienen dados en función del protocolo de la Euroliga, mostrado en el Anexo X.
 - Rutina de reloj. Como la pantalla y el marcador deben mostrar el tiempo del partido, pero ninguno de ellos cuenta con un reloj, este se genera mediante un módulo en el programa. El módulo genera una interrupción cada 100 milisegundos para entrar en una rutina independiente donde se computa el tiempo transcurrido y se envían de forma simultanea los datos a ambos dispositivos. El software tiene en cuenta además si el tiempo del cuarto actual es inferior a un minuto, en cuyo caso se cambia la visualización a segundos y decenas de centésimas tanto en la pantalla como en el marcador.

3.4. Software de simulación del marcador

Por último, antes de pasar a ver el montaje completo del sistema de simulación, es necesario explicar cómo funciona el software proporcionado por Mondo, llamado *MONDOVIDEO 4.2*, y que simula el marcador electrónico real. Al abrir el software este permite, a través de su interfaz, seleccionar el puerto por el cual le van a llegar los datos, seleccionar la configuración que se va a utilizar en cuanto a información mostrada, y elegir la resolución y el deporte como vemos en la Fig.26.



Figura 26. Interfaz principal de MONDOVIDEO 4.2, la cual muestra en verde los puertos disponibles en el momento de la ejecución

En el caso del baloncesto, se pueden seleccionar a parte de distintas resoluciones, tres tipos de marcador: uno en el cual se ve solo la información básica, otro más completo con las estadísticas individuales y un tercero que también incluye los nombres de los jugadores. En nuestro caso elegiremos la 2ª configuración, la cual corresponde con la opción seleccionada en la figura anterior y es la que mejor se adecua a como se ha programado tanto la pantalla TFT táctil como el programa de control. La forma la cual se visualiza está representada en la Fig. 27.



Figura 27. Marcador generado por el software MONDOVIDEO 4.2 que muestra las estadísticas individuales de cada jugador además de las estadísticas generales del partido

En cuanto a lo que adquisición de datos se refiere, el Software lee todos los datos del registro del puerto de entrada y una vez tiene todos va cogiendo tramas de 4 bytes las cuales separa según encuentra el carácter de inicio y de final. Dichos datos, de manera similar a como ocurre con el programa de control de la pantalla TFT táctil, se comparan según su dirección y dato (byte (1) y byte (2)) y se actúa sobre el grafismo de la Fig. 27 en consecuencia.

3.5. Simulación del sistema de marcador deportivo

Para el sistema de simulación se usarán la pantalla TFT táctil, un ordenador con el programa de control y el software de simulación del marcador y los cables y adaptadores necesarios para la conexión de todos los dispositivos. La pantalla TFT táctil se conectará a un puerto serie USB del ordenador mediante un cable adaptador como el de la Fig. 28 de RS-232 a USB.



Figura 28. Parte trasera de la pantalla TFT táctil de donde sale el cable RS-232 junto con el adaptador usado para su conexión al puerto serie del ordenador

La conexión entre el programa de control y el software que simula el marcador electrónico dentro del mismo ordenador, se lleva a cabo de la misma forma mediante dos adaptadores USB a RS232, conectando cada extremo del cable a un puerto serie del ordenador distinto, como en la Fig. 29 y Fig. 30.



Figura 29. Cable RS-232 con dos adaptadores RS-232 a USB para der conectado al ordenador

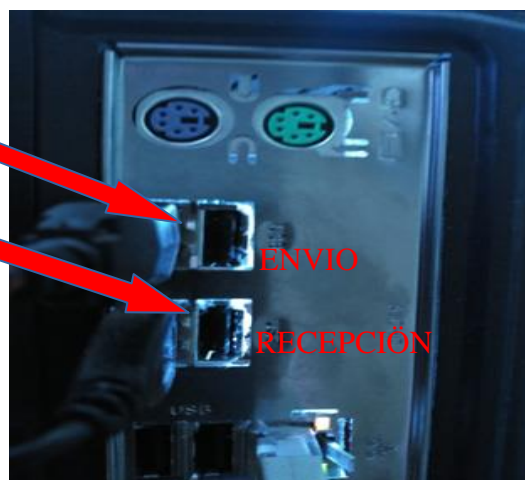


Figura 30. Puerto de salida del programa de control y puerto de entrada de MONDOVIDEO 4.2

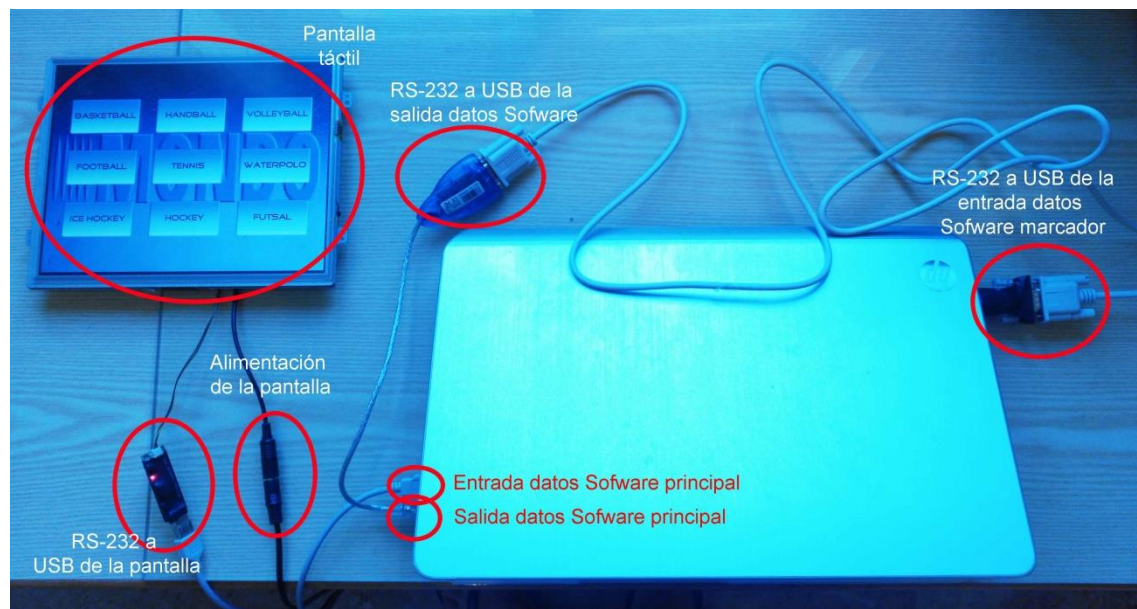


Figura 31. Montaje físico del sistema de simulación completo

Una vez enchufada la pantalla, y habiendo cargado previamente la interfaz gráfica, se conecta a uno de los puertos USB de ordenador, y se comprueba que nombre se le ha asignado, siendo nuestro caso el COM 5. Una vez sabemos esto, conectamos el cable de la Fig. 28 a otros dos puertos USB del ordenador para que el programa de control y MONDOVIDEO 4.2 puedan comunicarse. Con todo conectado físicamente iniciamos ambos programas e introducimos en las pestañas por qué puerto entrarán y saldrán los datos al programa de control. Una vez hecho esto seleccionamos el puerto por el cual recibirá los datos el software del marcador electrónico, elegimos la configuración 2 y marcamos la pestaña de *importar dorsales desde la consola* en la ventana de control que aparece, Fig. 32.

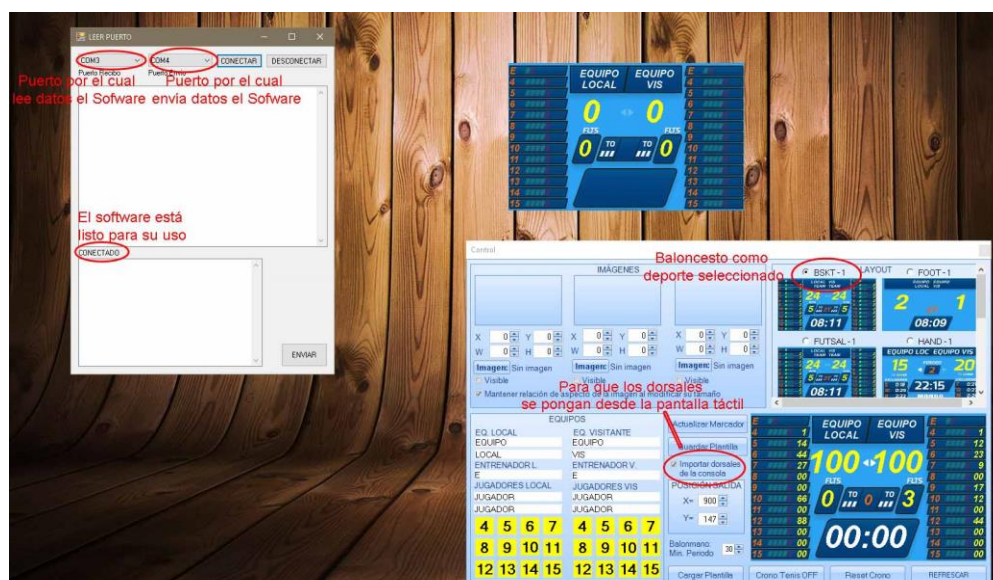


Figura 32. Aspecto del ordenador usado para el control de la pantalla y marcador con ambos programas cargados y listos para su funcionamiento

Con todo esto hecho la pantalla TFT táctil está lista para controlar el marcador electrónico, por lo que se selecciona el baloncesto en la pantalla principal de la pantalla TFT y se introducen los dorsales de cada equipo, cuando estos están configurados se pulsa el botón de confirmar y con esto se envía un comando de reset al software del marcador el cual carga todos los dorsales tal y como se especifica en el anexo X. Con esto hecho la pantalla TFT táctil queda lista para el manejo del marcador electrónico durante el partido.

Una vez completado el sistema de simulación completo se ha llegado a una serie de conclusiones. En primer lugar, los dos dispositivos son capaces de comunicarse entre sí y el sistema completo funciona por lo que los dos primeros objetivos establecidos en el punto 2.1 se han cumplido. En segundo lugar, se ha confirmado que el uso de la pantalla TFT táctil es intuitivo tanto en la introducción de datos al marcador como en la visualización de datos. Y en tercer lugar, la alternativa es viable por lo que, en el siguiente capítulo, se explicará la modificación del sistema real para su funcionamiento con la pantalla TFT táctil.

Capitulo 4

Integración de la pantalla TFT en las consolas actuales

4.1. Adaptación del Hardware

Una vez analizado el funcionamiento del sistema de simulación, el siguiente paso es hacer funcionar el sistema sustituyendo el programa de control del ordenador por la consola de Mondo. Para ello se realizará un prototipo a partir de una de las consolas actuales. Para la realización de este prototipo primero se retirará el teclado de membrana y se colocará la pantalla TFT táctil en su lugar. En segundo lugar se modificará la placa de la consola de tal forma que se pueda conectar la UART de la pantalla TFT táctil a la UART del microcontrolador. Y por último, habrá que conectar la consola al ordenador de manera que se comuniquen con el software de simulación del marcador MONDOVIDEO 4.2. Como ya se dijo en el apartado 1.2 la consola se comunica con el marcador real mediante RS-485 por lo que para conectar la consola al ordenador será necesario un adaptador de RS-485 a RS-232 como el de la Fig.33 y a su vez un adaptador RS-232 a USB.

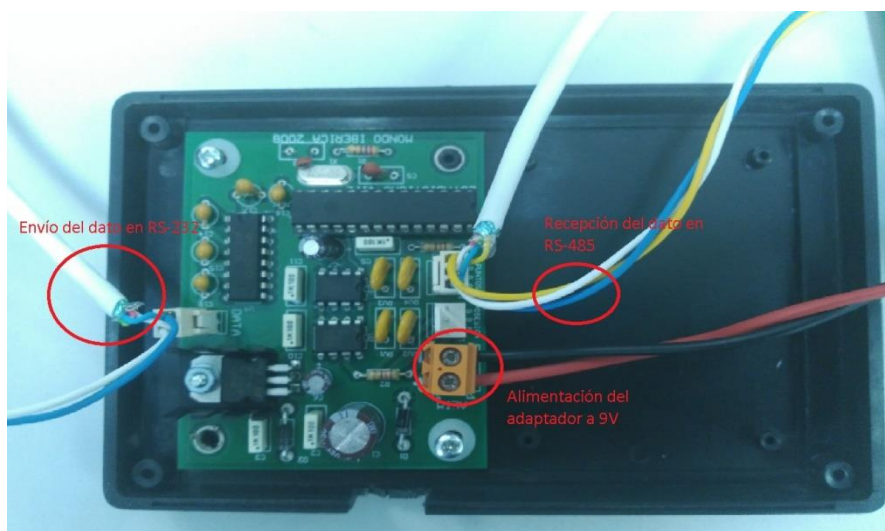


Figura 33. Adaptador de RS-485 a RS-232 para el envío de datos de la consola de Mondo al software de simulación del marcador electrónico en el ordenador

Para la conexión de la pantalla TFT táctil al microcontrolador, se deberá tener en cuenta que la UART de la TFT trabaja de 0 a 3.3 voltios y la UART del microcontrolador trabaja de 0 a 5 voltios. Por ello, la conexión entre ambas se hará a través de un adaptador de tensión bidireccional, basado en transistores MOSFET BSS138, como el de la Fig. 34.



Figura 34. Adaptador bidireccional de tensión para la conexión entre la pantalla TFT táctil y el microcontrolador de la consola de Mondo

El adaptador deberá ser alimentado a 5 voltios y a 3.3 voltios, por ser estas las tensiones de cada una de las UART. Ambas tensiones se obtendrán de la placa de la consola de Mondo, la tensión de 5 voltios directamente y la tensión de 3.3 voltios a través de un regulador de tensión LM3904. Esta decisión se debe a que la pantalla TFT táctil se alimenta a 5 voltios aunque su UART trabaje hasta 3.3 voltios, y esta tensión no está accesible, por lo que resulta mejor obtenerla de la placa de Mondo. De la misma manera, los dos pines de masa del adaptador irán conectados a la masa de la placa de Mondo. El esquema del circuito de adaptación de tensión se muestra en la Fig. 35 y su realización en la Fig. 36.

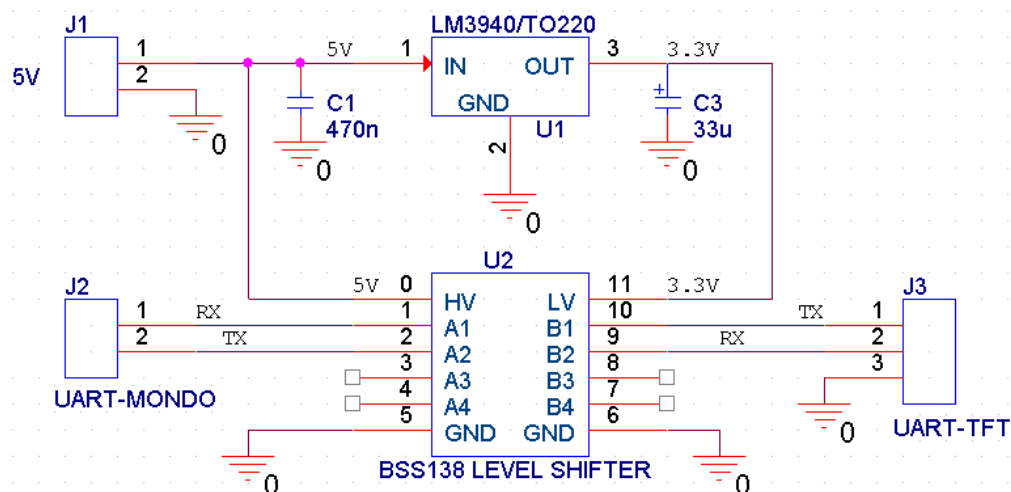


Figura 35. Esquemático del circuito de adaptación de niveles de tensión entre UART

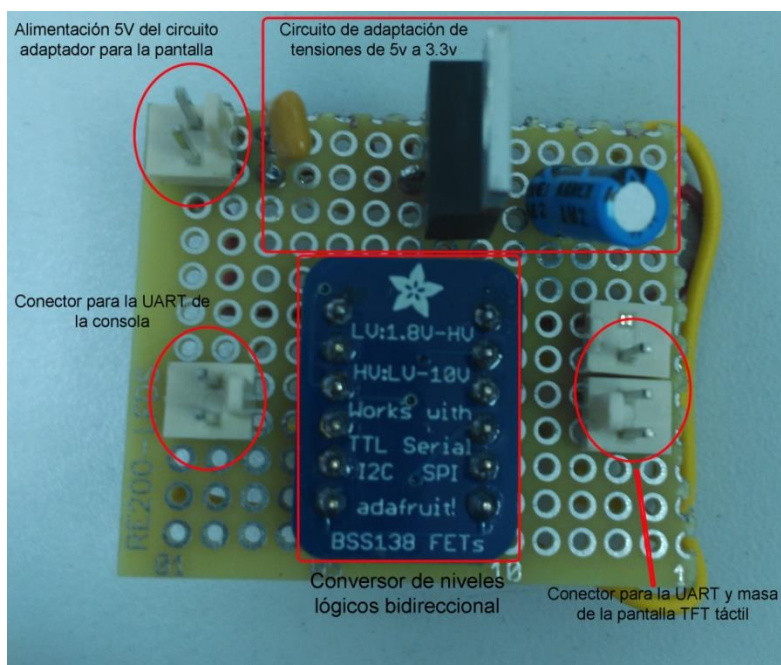


Figura 36. Realización del circuito de adaptación de niveles de tensión entre UART

Con el circuito de adaptación hecho y las dos UART accesibles para su conexión, el montaje queda como se ve en la Fig. 37. Donde el cable azul y blanco van conectados respectivamente a los pines 17 y 18 que corresponden al TX y RX de la UART del microcontrolador. Y por otro lado van conectados a través del adaptador de tensión a los cables azul y blanco correspondientes al RX y TX de la UART de la pantalla TFT táctil.

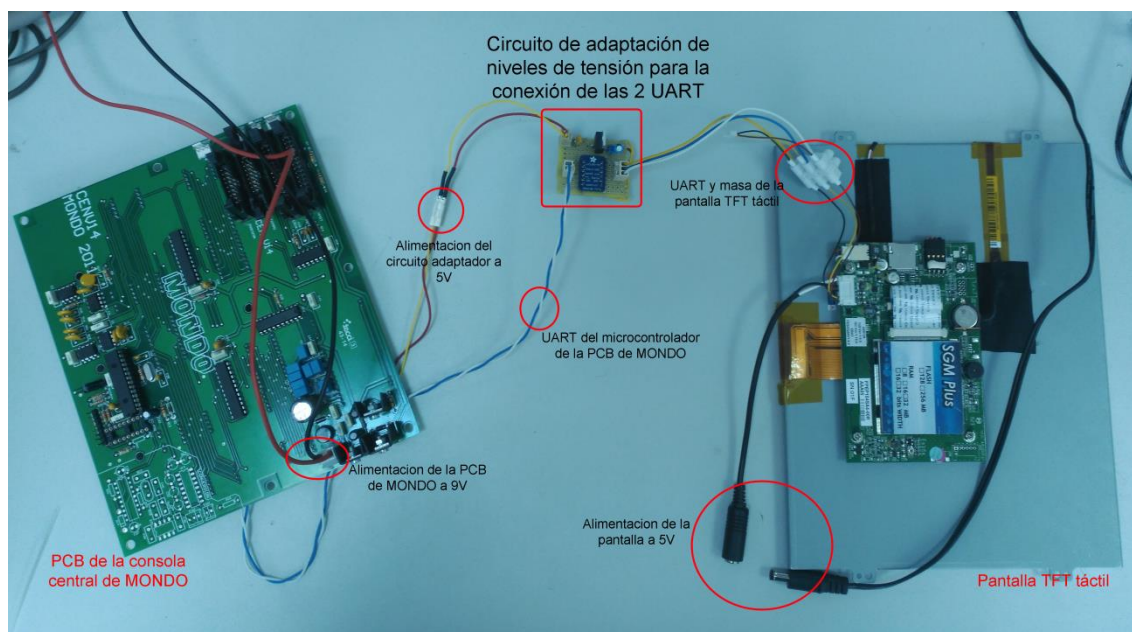


Figura 27. Montaje completo del sistema, con la pantalla TFT, placa de la consola central de MONDO y el circuito adaptador de las 2 UART

La utilización de los pines 17 y 18 para la pantalla TFT táctil, inutiliza el teclado de membrana ya que el chip de teclado va conectado a ellos en la consola actual.

4.2. Programación del microcontrolador de la consola central

Una vez adaptado el hardware de la consola, es necesario programar el microcontrolador de la consola de forma que pueda gestionar los comandos que le llegarán de la pantalla TFT táctil a través de la UART. Para ello, de igual manera que se hizo en el ordenador, se programará el microcontrolador para que al llegar un dato, lo procese para verificar que es correcto. Una vez verificado, enviará los comandos UART necesarios para la modificación de la pantalla TFT táctil y los datos al software de simulación MONDOVIDEO 4.2.

La adquisición de datos, se gestionará mediante una interrupción generada por la UART. Dentro de la interrupción se comprobará que el primer byte corresponde con el inicio de trama y que el último corresponde con el fin de trama, si ambos son correctos el dato se procesará.

Para el procesado, de manera similar a como se hizo con el ordenador, se buscará que combinación de byte(3) y byte(4) (correspondientes al botón pulsado) es la que se ha leído por la UART. Y una vez encontrada se mandará la información a la pantalla táctil y al ordenador.

El envío de datos desde el microcontrolador es distinto si se trata de datos de vuelta a la pantalla TFT táctil o datos al software del ordenador. En el primer caso, el dato de vuelta se enviará a través de la UART, por lo que se cargarán los bytes uno a uno del registro de salida de la UART hasta que la trama completa llegue a la pantalla. Tanto la recepción del dato, su envío de vuelta a la pantalla y el procesado que se realiza se pueden ver en el Anexo B. Por otro lado, para el envío de datos al software de simulación del marcador electrónico se utilizará una función ya programada por Mondo en el microcontrolador y no incluida en el anexo por motivos de confidencialidad.

Capítulo 5

Conclusiones y líneas futuras

5.1. Conclusiones y líneas futuras

El principal objetivo del trabajo era, en primer lugar, estudiar la posible sustitución del teclado por una pantalla táctil en la consola de control de Mondo. Y una vez vista su viabilidad, realizar un prototipo con la pantalla y consola central para comprobar su correcto funcionamiento.

Como la primera parte del trabajo se completó de manera satisfactoria, ha sido posible la realización del prototipo utilizando la pantalla táctil y la placa de la consola actual de MONDO.

El primer paso a realizar en un futuro será completar el diseño de la interfaz gráfica de la pantalla de manera que se configuren todos los deportes y la comunicación se realice con el marcador electrónico real.

El siguiente paso será el realizar un nuevo diseño de la placa de circuito impreso de la consola, de manera que ya no haya componentes no necesarios como el chip de teclado o los displays de 7 segmentos. Y se realizará su sustitución por el circuito adaptador para la conexión directa de la pantalla a la propia placa.

Por otra parte, puesto que el sistema de consolas incorpora una consola específica para el tiempo, el control del mismo sólo se realizó en el sistema de simulación a modo de prueba. Siendo el siguiente paso a realizar en un futuro, cambiar también el teclado de esta consola de manera que se controle a través de una pantalla táctil de menor tamaño para no modificar el tamaño actual de esta consola.

Anexo A: Protocolo de la Euroliga de envío de datos al marcador

Acciones posibles – Cuando debe enviarse y tramas a enviar

1. **Actualizar marcador.** Al pulsar un botón que se debería llamar “Actualizar”, para enviar toda la información al marcador o pantalla led. Esto incluye dorsales, faltas y puntos de cada jugador local y visitante, faltas de entrenador local y visitante, periodo, tiempos muertos de local y visitante, puntuación de equipo local y visitante, faltas de equipo local y visitante
 - a. El envío de faltas personales de jugadores y entrenadores debe ser con Cx y Dx
2. **Reset marcador.** Solo se envía al inicio del partido
 - a. Envío de comando inicio E1 0D
 - b. Envío de comando reset 33 55
 - c. Actualizar marcador
3. **Cambio de periodo**
 - a. Envío de faltas de equipo local y visitante
 - b. Envío de periodo
4. **Cambio de puntuación**
 - a. Envío de puntuación de jugador que se ha modificado
 - b. Envío de puntuación de equipo que se ha modificado
5. **Cambio de número de faltas de un jugador**
 - a. Envío de faltas personales del jugador o entrenador que comete la falta, debe ser con 4x y 5x
 - b. Envío faltas equipo local o visitante
6. **Cambio de número de faltas de un equipo**
 - a. Envío faltas equipo local o visitante
7. **Cambio en tiempo muerto local o visitante**
 - a. Tiempos muertos totales de equipo local o visitante
8. **Modificación de flechas de local o visitante**
 - a. Primero se apagan las dos flechas
 - b. Se enciende la flecha correspondiente o no se enciende ninguna

1er byte	2º byte	Significado
A4	5x	x puntos, unidades del local
A5	5x	x puntos, decenas del local
A6	5x	x puntos, centenas del local
A7	5x	x puntos, unidades del visitante
A8	5x	x puntos, decenas del visitante
A9	5x	x puntos, centenas del visitante
AA	5x	Periodo x
AB	5x	x faltas de equipo local
AC	5x	x faltas de equipo visitante
8x (x>3)	yz	yz (BCD) puntos individuales de jugador local
9x (x>3)	yz	yz (BCD) puntos individuales de jugador visitante
Cx (x≥3)	5y	y faltas personales del jugador x local
Dx (x≥3)	5y	y faltas personales del jugador x visitante
4x (x≥3)	5y	Falta personal y del jugador local x
5x (x≥3)	5y	Falta personal y del jugador visitante x
C0	5x	Tiempos muertos equipo local
D0	5x	Tiempos muertos equipo visitante
E0	51	Enciende flecha del equipo visitante
E0	52	Enciende flecha del equipo local
E0	53	Apaga flechas local y visitante
33	55	Reset marcador
0x (x>3)	yz	Establece yz (BCD) como dorsal de camiseta del jugador local x
1x (x>3)	yz	Establece yz (BCD) como dorsal de camiseta del jugador visitante x
E1	0D	Comando inicio de recepción al actualizar marcador

Tabla de valores x del 1er byte:	
Entrenador	3
Jugador 1	4
Jugador 2	5
Jugador 3	6
Jugador 4	7
Jugador 5	8
Jugador 6	9
Jugador 7	A
Jugador 8	B
Jugador 9	C
Jugador 10	D
Jugador 11	E
Jugador 12	F

Anexo B: Esquema del código del programa

En el siguiente anexo se ve un resumen del código implementado en C y posteriormente compilado y cargado en el microcontrolador incluido en la placa de la consola de Mondo. El objetivo de este anexo es entender mejor cómo funciona la comunicación entre dispositivos por lo que el código que resulta repetitivo no ha sido incluido sino solo un resumen del mismo. El anexo está dividido en 3 secciones: recepción, envío y procesado del dato.

RECEPCIÓN DEL DATO DE LA PANTALLA MEDIANTE INTERRUPCIÓN DE LA UART

```
void Interrupciones_isr(void) {  
    if (PIR1bits.RCIF == 1) {  
        // -----//  
        // -----//  
        //  Interrupción proveniente de la pantalla TFT táctil  //  
        // -----//  
        // -----//  
        PIE1bits.RCIE = 0;  
        boton[contador_uart] = RCREG;  
  
        if (contador_uart == 0 && boton[contador_uart] == 0xAF) {  
            dato_bueno = 1;  
        }  
        else if (contador_uart == 5 && boton[contador_uart] == 0xEF) {  
            procesado = 1;  
            contador_uart = 0;  
            dato_bueno = 0;  
        }  
        else if (contador_uart == 5 && boton[contador_uart] != 0xEF) {  
            contador_uart = 0;  
            dato_bueno = 0;  
        }  
  
        if (dato_bueno == 1) contador_uart++;  
  
        PIR1bits.RCIF = 0;  
        PIE1bits.RCIE = 1;  
    }  
}
```

ENVÍO DEL DATO A LA PANTALLA POR LA UART

```
void enviar_comandos_TFT_corto (unsigned char ch1 , unsigned char ch2,unsigned char ch3) {

    //Comando de sonido del buzzer para la pantalla TFT táctil
    TXREG = 0xAF;
    while (TXSTAbits.TRMT == 0);
    TXREG=ch1;
    while (TXSTAbits.TRMT == 0);
    TXREG=ch2;
    while (TXSTAbits.TRMT == 0);
    TXREG=ch3;
    while (TXSTAbits.TRMT == 0);
    TXREG=0xEF;
    while (TXSTAbits.TRMT == 0);

}

void enviar_comandos_TFT_medio (unsigned char ch1 , unsigned char ch2,unsigned char ch3,
unsigned char ch4, unsigned char ch5, unsigned char ch6, unsigned char ch7, unsigned char ch8)
{ //Comandos de carga de iconos para la pantalla TFT táctil
}

void enviar_comandos_TFT_largo (unsigned char ch1 , unsigned char ch2,unsigned char ch3, unsigned
char ch4, unsigned char ch5, unsigned char ch6, unsigned char ch7, unsigned char ch8, unsigned char
ch9, unsigned char ch10) { //Comandos de borrado de iconos para la pantalla TFT táctil
}
```

FUNCIONES PARA EL PROCESADO DEL BOTÓN PULSADO

```
void procesado_boton (void) {

    if (boton[3] == 0 && boton[4] == 1) { //Botón pantalla principal }

    //-----//
    //  PUESTA Y SELECCIÓN DE DORSALES  //
    //-----//
    if (boton[3] == 0 && boton[4] == 36) { //Botón dorsales de la pantalla principal de baloncesto }
    if (boton[3] == 0 && boton[4] == 222) { //Poner dorsales equipo local }
    if (boton[3] == 0 && boton[4] == 223) { //Poner dorsales equipo visitante }
    if (boton[3] == 0 && boton[4] == 220) { //Siguiete dorsal local
        if (local == 1) {
            if (numero_jugador_local_TFT == 1) {}
            if (numero_jugador_local_TFT == 2) {}
            if (numero_jugador_local_TFT == 3) {}
            if (numero_jugador_local_TFT == 4) {}
            if (numero_jugador_local_TFT == 5) {}
            if (numero_jugador_local_TFT == 6) {}
            if (numero_jugador_local_TFT == 7) {}
            if (numero_jugador_local_TFT == 8) {}
            if (numero_jugador_local_TFT == 9) {}
            if (numero_jugador_local_TFT == 10) {}
            if (numero_jugador_local_TFT == 11) {}
            if (numero_jugador_local_TFT == 12) {}
        }
    }
    if (boton[3] == 0 && boton[4] == 221) { //Siguiete dorsal visitante
        if (local == 2) {
            if (numero_jugador_visitante_TFT == 0) {}
        }
    }
}
```

```

if (numero_jugador_visitante_TFT == 1) {}
if (numero_jugador_visitante_TFT == 2) {}
if (numero_jugador_visitante_TFT == 3) {}
if (numero_jugador_visitante_TFT == 4) {}
if (numero_jugador_visitante_TFT == 5) {}
if (numero_jugador_visitante_TFT == 6) {}
if (numero_jugador_visitante_TFT == 7) {}
if (numero_jugador_visitante_TFT == 8) {}
if (numero_jugador_visitante_TFT == 9) {}
if (numero_jugador_visitante_TFT == 10) {}
if (numero_jugador_visitante_TFT == 11) {}
if (numero_jugador_visitante_TFT == 12) {}

}
}
if (boton[3] == 0 && boton[4] == 209) { //Incrementar en 1 el dorsal
//----- //
// SE PONEN LOS DORSALES LOCALES //
//----- //
if (local == 1) {
if (numero_jugador_local_TFT == 0) {}
if (numero_jugador_local_TFT == 1) {}
if (numero_jugador_local_TFT == 2) {}
if (numero_jugador_local_TFT == 3) {}
if (numero_jugador_local_TFT == 4) {}
if (numero_jugador_local_TFT == 5) {}
if (numero_jugador_local_TFT == 6) {}
if (numero_jugador_local_TFT == 7) {}
if (numero_jugador_local_TFT == 8) {}
if (numero_jugador_local_TFT == 9) {}
if (numero_jugador_local_TFT == 10) {}
if (numero_jugador_local_TFT == 11) {}
if (numero_jugador_local_TFT == 12) {}
}

//----- //
// SE PONEN LOS DORSALES VISITANTES //
//----- //
if (local == 2) {
if (numero_jugador_visitante_TFT == 0) {}
if (numero_jugador_visitante_TFT == 1) {}
if (numero_jugador_visitante_TFT == 2) {}
if (numero_jugador_visitante_TFT == 3) {}
if (numero_jugador_visitante_TFT == 4) {}
if (numero_jugador_visitante_TFT == 5) {}
if (numero_jugador_visitante_TFT == 6) {}
if (numero_jugador_visitante_TFT == 7) {}
if (numero_jugador_visitante_TFT == 8) {}
if (numero_jugador_visitante_TFT == 9) {}
if (numero_jugador_visitante_TFT == 10) {}
if (numero_jugador_visitante_TFT == 11) {}
if (numero_jugador_visitante_TFT == 12) {}
}

}
}
if (boton[3] == 0 && boton[4] == 210) { //Decrementar en 1 el dorsal }
if (boton[3] == 0 && boton[4] == 211) { //Botón de la pantalla de dorsales }

//----- //
//SELECCIÓN DE JUGADOR/ENTRENADOR //
//----- //
if (boton[3] == 0 && boton[4] == 131) { //Entrenador local }
if (boton[3] == 0 && boton[4] == 132) { //Jugador local 1 }
if (boton[3] == 0 && boton[4] == 133) { //Jugador local 2 }
if (boton[3] == 0 && boton[4] == 134) { //Jugador local 3 }
if (boton[3] == 0 && boton[4] == 135) { //Jugador local 4 }
if (boton[3] == 0 && boton[4] == 136) { //Jugador local 5 }
if (boton[3] == 0 && boton[4] == 137) { //Jugador local 6 }

```

```

if (boton[3] == 0 && boton[4] == 138) { //Jugador local 7 }
if (boton[3] == 0 && boton[4] == 139) { //Jugador local 8 }
if (boton[3] == 0 && boton[4] == 140) { //Jugador local 9 }
if (boton[3] == 0 && boton[4] == 141) { //Jugador local 10 }
if (boton[3] == 0 && boton[4] == 142) { //Jugador local 11 }
if (boton[3] == 0 && boton[4] == 143) { //Jugador local 12 }
if (boton[3] == 0 && boton[4] == 147) { //Entrenador visitante }
if (boton[3] == 0 && boton[4] == 148) { //Jugador visitante 1 }
if (boton[3] == 0 && boton[4] == 149) { //Jugador visitante 2 }
if (boton[3] == 0 && boton[4] == 150) { //Jugador visitante 3 }
if (boton[3] == 0 && boton[4] == 151) { //Jugador visitante 4 }
if (boton[3] == 0 && boton[4] == 152) { //Jugador visitante 5 }
if (boton[3] == 0 && boton[4] == 153) { //Jugador visitante 6 }
if (boton[3] == 0 && boton[4] == 154) { //Jugador visitante 7 }
if (boton[3] == 0 && boton[4] == 155) { //Jugador visitante 8 }
if (boton[3] == 0 && boton[4] == 156) { //Jugador visitante 9 }
if (boton[3] == 0 && boton[4] == 157) { //Jugador visitante 10 }
if (boton[3] == 0 && boton[4] == 158) { //Jugador visitante 11 }
if (boton[3] == 0 && boton[4] == 159) { //Jugador visitante 12 }

//-----//
//MODIFICACIÓN DE LAS PUNTUACIONES //
//-----//
if (boton[3] == 0 && boton[4] == 16) { //Sumar un punto }
if (boton[3] == 0 && boton[4] == 18) { //Sumar dos puntos }
if (boton[3] == 0 && boton[4] == 20) { //Sumar tres puntos }
if (boton[3] == 0 && boton[4] == 17) { //Restar un punto }
if (boton[3] == 0 && boton[4] == 19) { //Restar dos puntos }
if (boton[3] == 0 && boton[4] == 21) { //Restar tres puntos }

//-----//
// MODIFICACIÓN DE LAS FALTAS //
//-----//
if (boton[3] == 0 && boton[4] == 22) { //Sumar una falta }
if (boton[3] == 0 && boton[4] == 23) { //Restar una falta }

//-----//
// MODIFICACIÓN DE LAS FLECHAS //
//-----//
if (boton[3] == 0 && boton[4] == 32) { //Pone o quita la flecha local }
if (boton[3] == 0 && boton[4] == 35) { //Pone o quita la flecha visitante }

//-----//
// MODIFICACIÓN DE LOS TM //
//-----//
if (boton[3] == 0 && boton[4] == 24) { //Sumar un TM local }
if (boton[3] == 0 && boton[4] == 25) { //Restar un TM local }
if (boton[3] == 0 && boton[4] == 33) { //Sumar un TM visitante }
if (boton[3] == 0 && boton[4] == 34) { //Restar un TM visitante }

//-----//
// MODIFICACIÓN DE LOS PERIODOS //
//-----//
if (boton[3] == 0 && boton[4] == 50) { //Incrementar el periodo }
if (boton[3] == 0 && boton[4] == 49) { //Decrementar el periodo }

//-----//
// SONIDO DE LA PANTALLA //
//-----//
if (boton[3] == 0 && boton[4] == 52) { //Poner o no sonido a las teclas }

//-----//
// BOTONES ADICIONALES DE CONTROL //
//-----//
if (boton[3] == 0 && boton[4] == 37) { //Botón ENTER de la pantalla de baloncesto }
if (boton[3] == 0 && boton[4] == 38) { //Resetear marcador }

}

```

```

void PonerDorsalesAzules (void) {

    if (boton[4] == 211 || boton[4] == 37 || boton[4] == 1) {

        enviar_comandos_TFT_medio(0x7,0x72,0x0,0x14,0x0,0x32,0x0,230);
        enviar_comandos_TFT_medio(0x7,0x72,0x0,0x14,0x0,0x5A,0x0,DORSAL_LOCAL[4]);
        enviar_comandos_TFT_medio(0x7,0x72,0x0,0x5A,0x0,0x5A,0x0,DORSAL_LOCAL[5]);
        enviar_comandos_TFT_medio(0x7,0x72,0x0,0x14,0x0,0xAA,0x0,DORSAL_LOCAL[6]);
        enviar_comandos_TFT_medio(0x7,0x72,0x0,0x5A,0x0,0xAA,0x0,DORSAL_LOCAL[7]);
        enviar_comandos_TFT_medio(0x7,0x72,0x0,0x14,0x0,0xFA,0x0,DORSAL_LOCAL[8]);
        enviar_comandos_TFT_medio(0x7,0x72,0x0,0x5A,0x0,0xFA,0x0,DORSAL_LOCAL[9]);
        enviar_comandos_TFT_medio(0x7,0x72,0x0,0x14,0x1,0x4A,0x0,DORSAL_LOCAL[10]);
        enviar_comandos_TFT_medio(0x7,0x72,0x0,0x5A,0x1,0x4A,0x0,DORSAL_LOCAL[11]);
        enviar_comandos_TFT_medio(0x7,0x72,0x0,0x14,0x1,0x9A,0x0,DORSAL_LOCAL[12]);
        enviar_comandos_TFT_medio(0x7,0x72,0x0,0x5A,0x1,0x9A,0x0,DORSAL_LOCAL[13]);
        enviar_comandos_TFT_medio(0x7,0x72,0x0,0x14,0x1,0xEA,0x0,DORSAL_LOCAL[14]);
        enviar_comandos_TFT_medio(0x7,0x72,0x0,0x5A,0x1,0xEA,0x0,DORSAL_LOCAL[15]);
        enviar_comandos_TFT_medio(0x7,0x72,0x2,0x8A,0x0,0x32,0x0,230);
        enviar_comandos_TFT_medio(0x7,0x72,0x2,0x8A,0x0,0x5A,0x0,DORSAL_VIS[4]);
        enviar_comandos_TFT_medio(0x7,0x72,0x2,0xD0,0x0,0x5A,0x0,DORSAL_VIS[5]);
        enviar_comandos_TFT_medio(0x7,0x72,0x2,0x8A,0x0,0xAA,0x0,DORSAL_VIS[6]);
        enviar_comandos_TFT_medio(0x7,0x72,0x2,0xD0,0x0,0xAA,0x0,DORSAL_VIS[7]);
        enviar_comandos_TFT_medio(0x7,0x72,0x2,0x8A,0x0,0xFA,0x0,DORSAL_VIS[8]);
        enviar_comandos_TFT_medio(0x7,0x72,0x2,0xD0,0x0,0xFA,0x0,DORSAL_VIS[9]);
        enviar_comandos_TFT_medio(0x7,0x72,0x2,0x8A,0x1,0x4A,0x0,DORSAL_VIS[10]);
        enviar_comandos_TFT_medio(0x7,0x72,0x2,0xD0,0x1,0x4A,0x0,DORSAL_VIS[11]);
        enviar_comandos_TFT_medio(0x7,0x72,0x2,0x8A,0x1,0x9A,0x0,DORSAL_VIS[12]);
        enviar_comandos_TFT_medio(0x7,0x72,0x2,0xD0,0x1,0x9A,0x0,DORSAL_VIS[13]);
        enviar_comandos_TFT_medio(0x7,0x72,0x2,0x8A,0x1,0xEA,0x0,DORSAL_VIS[14]);
        enviar_comandos_TFT_medio(0x7,0x72,0x2,0xD0,0x1,0xEA,0x0,DORSAL_VIS[15]);

    }

    else if (jugador_entrenador_seleccionado == 0)
        enviar_comandos_TFT_medio(0x7,0x72,0x0,0x14,0x0,0x32,0x0,230);
    else if (jugador_entrenador_seleccionado == 1)
        enviar_comandos_TFT_medio(0x7,0x72,0x0,0x14,0x0,0x5A,0x0,DORSAL_LOCAL[4]);
    else if (jugador_entrenador_seleccionado == 2)
        enviar_comandos_TFT_medio(0x7,0x72,0x0,0x5A,0x0,0x5A,0x0,DORSAL_LOCAL[5]);
    else if (jugador_entrenador_seleccionado == 3)
        enviar_comandos_TFT_medio(0x7,0x72,0x0,0x14,0x0,0xAA,0x0,DORSAL_LOCAL[6]);

}

unsigned char ConvertirDorsal (unsigned char dato) {

    unsigned char DorsalReal;
    if (dato == 10) DorsalReal = 0;
    else if (dato == 216) DorsalReal = 255;
    else DorsalReal = (dato-12)/2;
    return DorsalReal;

}

void Sonido (void) {

    enviar_comandos_TFT_corto(0x2,0x18,0x1);

}

```

void Actualizacion (void) {

```
//-----//
//          DORSALES          //
//-----//

    for ( j=4; j<16;j++) { //Dorsales locales
        aux1 = ConvertirDorsal(DORSAL_LOCAL[j]/10);
        aux2 = ConvertirDorsal(DORSAL_LOCAL[j]%10);
        aux3 = 16*aux1+aux2;
        envia_485(j);
        envia_485(aux3);
    }
    for ( j=4; j<16;j++) { //Dorsales visitantes
        aux1 = ConvertirDorsal(DORSAL_VIS[j]/10);
        aux2 = ConvertirDorsal(DORSAL_VIS[j]%10);
        aux3 = 16*aux1+aux2;
        envia_485(16+j);
        envia_485(aux3);
    }
}
```

```
//-----//
//          FALTAS          //
//-----//

    for ( j=0; j<13;j++) { //Faltas locales
        envia_485(195+j);
        envia5x_485(informacion_jugador[3][j]);
    }
    for ( j=13; j<26;j++) { //Faltas visitantes
        envia_485(198+j);
        envia5x_485(informacion_jugador[3][j]);
    }

    envia_485(0xAB);
    envia5x_485(FALTAS_LOCAL);
    envia_485(0xAC);
    envia5x_485(FALTAS_VIS);
```

```
//-----//
//          PUNTOS          //
//-----//

    for ( j=1; j<13;j++) { //Puntos locales
        envia_485(131+j);
        envia5x_485(informacion_jugador[2][j]);
    }
    for ( j=14; j<26;j++) { //Puntos visitantes
        envia_485(184+j);
        envia5x_485(informacion_jugador[2][j]);
    }

    envia_485(0xA6);
    envia5x_485((PUNTOS_LOCAL/100));
    envia_485(0xA5);
    envia5x_485((PUNTOS_LOCAL%100)/10);
    envia_485(0xA4);
    envia5x_485((PUNTOS_LOCAL%100)%10);
    envia_485(0xA9);
    envia5x_485((PUNTOS_VIS/100));
    envia_485(0xA8);
    envia5x_485((PUNTOS_VIS%100)/10);
    envia_485(0xA7);
    envia5x_485((PUNTOS_VIS%100)%10);
}
```

}

```

void Configurar_pantalla_principal (void) {

    enviar_comandos_TFT_largo(0x9,0x82,0x1,0x68,0x0,0x6E,0x1,0xC2,0x0,0xC8);
    enviar_comandos_TFT_largo(0x9,0x82,0x2,0x42,0x0,0x64,0x2,0x66,0x0,0x9E);
    enviar_comandos_TFT_medio(0x7,0x72,0x2,0x43,0x0,0x64,0x0,(PUNTOS_VIS%100)%10);
    enviar_comandos_TFT_largo(0x9,0x82,0x2,0x1E,0x0,0x64,0x2,0x42,0x0,0x9E);
    enviar_comandos_TFT_medio(0x7,0x72,0x2,0x1F,0x0,0x64,0x0,(PUNTOS_VIS%100)/10);
    enviar_comandos_TFT_largo(0x9,0x82,0x1,0xFA,0x0,0x64,0x2,0x1F,0x0,0x9E);
    enviar_comandos_TFT_medio(0x7,0x72,0x1,0xFB,0x0,0x64,0x0,PUNTOS_VIS/100);
    enviar_comandos_TFT_largo(0x9,0x82,0x1,0xB4,0x0,0x64,0x1,0xD8,0x0,0x9E);
    enviar_comandos_TFT_medio(0x7,0x72,0x1,0xB5,0x0,0x64,0x0,(PUNTOS_LOCAL%100)%10);
    enviar_comandos_TFT_largo(0x9,0x82,0x1,0x90,0x0,0x64,0x1,0xB4,0x0,0x9E);
    enviar_comandos_TFT_medio(0x7,0x72,0x1,0x91,0x0,0x64,0x0,(PUNTOS_LOCAL%100));
    enviar_comandos_TFT_largo(0x9,0x82,0x1,0x6C,0x0,0x64,0x1,0x90,0x0,0x9E);
    enviar_comandos_TFT_medio(0x7,0x72,0x1,0x6D,0x0,0x64,0x0,PUNTOS_LOCAL/100);
    enviar_comandos_TFT_largo(0x9,0x82,0x1,0xD4,0x0,0xA0,0x1,0xE7,0x0,0xBE);
    enviar_comandos_TFT_medio(0x7,0x72,0x1,0xD6,0x0,0xA0,0x0,periodo);
    enviar_comandos_TFT_largo(0x9,0x82,0x1,0xEA,0x0,0xA0,0x1,0xFD,0x0,0xBE);
    enviar_comandos_TFT_medio(0x7,0x72,0x1,0xEB,0x0,0xA0,0x0,242);
    enviar_comandos_TFT_largo(0x9,0x82,0x1,0x72,0x0,0xB4,0x1,0x83,0x0,0xD2);
    enviar_comandos_TFT_medio(0x7,0x72,0x1,0x72,0x0,0xB4,0x0,tiempos_local);
    enviar_comandos_TFT_largo(0x9,0x82,0x1,0x86,0x0,0xB4,0x1,0x97,0x0,0xD2);
    enviar_comandos_TFT_medio(0x7,0x72,0x1,0x86,0x0,0xB4,0x0,242);
    enviar_comandos_TFT_largo(0x9,0x82,0x1,0x9A,0x0,0xB4,0x1,0xAB,0x0,0xD2);
    enviar_comandos_TFT_medio(0x7,0x72,0x1,0x9A,0x0,0xB4,0x0,241);
    enviar_comandos_TFT_largo(0x9,0x82,0x2,0x29,0x0,0xB4,0x2,0x3A,0x0,0xD2);
    enviar_comandos_TFT_medio(0x7,0x72,0x2,0x29,0x0,0xB4,0x0,tiempos_visitante);
    enviar_comandos_TFT_largo(0x9,0x82,0x2,0x3D,0x0,0xB4,0x2,0x4E,0x0,0xD2);
    enviar_comandos_TFT_medio(0x7,0x72,0x2,0x3D,0x0,0xB4,0x0,242);
    enviar_comandos_TFT_largo(0x9,0x82,0x2,0x51,0x0,0xB4,0x2,0x62,0x0,0xD2);
    enviar_comandos_TFT_medio(0x7,0x72,0x2,0x51,0x0,0xB4,0x0,241);

}

```

```

void modificar_puntos_TFT (unsigned char sumaresta, unsigned char ppuntos, unsigned char eequipo,
unsigned char ijugador) { //Actualiza la información referente a la puntuación
}

```

```

void modificar_faltas_TFT (unsigned char sumaresta, unsigned char ffaltas, unsigned char eequipo,
unsigned char ijugador) { //Actualiza la información referente a las faltas
}

```