

10. PROBLEMAS Y SOLUCIONES DURANTE EL DESARROLLO DEL TRABAJO

10.1. Problemas en hardware

10.1.1 Fuente de CD a -15V.

El LCD necesita una fuente de voltaje negativo con el fin de manejar el contraste. Construir una fuente negativa a partir de una positiva fue una labor que involucró bastante investigación y pruebas. Al principio, resultaba ser una fuente que tenía muchas irregularidades en su forma de onda, porque mostraba picos y valles. Con la ayuda de un condensador se pudo acondicionar mejor la señal y permitir que el voltaje fuese uniforme para el LCD.

Con el ánimo de minimizar las fallas del circuito de alimentación y garantizar que la potencia que necesita todo el hardware se pueda suministrar, se optó por construir una fuente dual de voltaje de +5 voltios y -12 voltios. Esto para garantizar una forma de onda óptima por completo, al igual que por simplicidad: necesita menos componentes y es menos susceptible a fallas. Finalmente se pensó en algo más simple y menos voluminoso. Se usó un integrado MAX 232, el cual por el pin 6 saca un voltaje de -9 voltios, los necesarios para poder alimentar el pin de contraste de la pantalla. La alimentación del integrado es 9 voltios, los suficientes para que se entregue la corriente necesaria al LCD y no se caliente demasiado. Los 9 voltios se sacan de un adaptador de voltaje.

10.1.2. Protocolo de comunicación I2C.

Las pruebas realizadas al protocolo I2C implementado por software mostraron a través del osciloscopio que el protocolo funcionaba. Aparentemente la señal de *Acknowledge* que emite el receptor, en este caso la memoria, se estaba dando. Cuando se procedió a realizar la prueba final con señales procedentes del fisiógrafo, se encontró que a la memoria se le escribían secuencias de ceros y no el valor de voltaje adquirido a través del ADC. Luego de muchos ensayos y diversas pruebas, se llegó a la conclusión que era el mismo osciloscopio quien bajaba drásticamente el voltaje en el pin correspondiente al bus serial de datos y por eso, el microcontrolador creía que era la memoria quien le daba el *acknowledge*, lo cual no era cierto.

Se plantearon varias soluciones, entre ellas la de manejar el protocolo I2C a través del módulo SPI del microcontrolador. Pero más bien lo que se realizó fue remover la rutina de interrupción de TBM de la implementación por software de protocolo. Esto se puede hacer porque el protocolo I2C no requiere una señal de reloj simétrica para operar, requiere más bien de flancos de subida y bajada

bien definidos. Por lo tanto, los cambios de nivel lógico de reloj se realizaron dentro de la misma rutina y mejoró la velocidad de transferencia. Se dejaron de presentar bloqueos durante la ejecución.

10.2. Problemas en software

10.2.1 Inicialización del LCD:

Para realizar la inicialización del LCD se deben tener en cuenta ciertos parámetros de configuración, los cuales determinan en cierta medida el funcionamiento del LCD al igual que su desempeño según la utilidad en que se va a usar.

Recién adquirido el LCD se procedió a hacer pruebas para que escribiera texto: ésto es lo más básico y su funcionamiento es un indicio de que el LCD opera bien al igual que es una base para optimizar su funcionamiento. Luego, cuando se logró escribir texto se procedió a hacer ensayos para poder dibujar píxel a píxel.

El LCD tenía un tiempo de inicialización promedio de 15 segundos. Esto para aplicaciones médicas no es deseable porque se necesita que el instrumento entre en operación lo más pronto posible. Variando retardos y optimizando el código, se logró inicializar el LCD en 3 segundos.

10.2.2. Velocidad del bus de datos en los microcontroladores:

Inicialmente se realizaron los programas en los microcontroladores usando el bus a 1.228 MHz. Esto permitía que los programas funcionaran, pero se observaba que el programa era muy lento. Se procedió a programar el PLL interno para incrementar la velocidad del bus hasta su máximo de 8 MHz. Esta nueva velocidad arroja mejores resultados respecto a la densidad de puntos en la gráfica, su resolución es mucho mejor.

10.2.3. Velocidad de transferencia de datos en el protocolo I2C:

El protocolo I2C tolera velocidades de transmisión muy altas. Esto es soportado por circuitos integrados que tienen hardware diseñado para este fin. Como la implementación que se realizó en este trabajo fue desarrollada en software, es más lento en ejecución y está sujeto a errores y a bloqueos. EL protocolo I2C implementado en el microcontrolador tiene un límite de velocidad de transferencia de datos. Al tener que leer y escribir tantos datos, el proceso de graficación del espectro de la señal ECG es muy lento, demora 8 minutos.

10.2.4. Desempeño del algoritmo de la Transformada de Hartley Discreta:

La Transformada de Hartley Discreta se halló a través de un método iterativo. Éste método se vale del algoritmo de la Mariposa para hallar valores que al final del proceso permitían obtener un vector

de términos en el dominio de la frecuencia. Éste procedimiento es largo y se involucran bastantes condiciones. El programa se escribió en lenguaje C/C++ orientado a una plataforma embebida usando CodeWarrior versión 3.0. Escribir el programa en lenguaje C es fácil, ya que se disponen de herramientas para manipular los datos fácilmente. Cuando el compilador de CodeWarrior traduce hacia lenguaje assembler, crea un código bastante largo y complejo. Debido a la extensión del programa la ejecución es lenta. Que el programa demore bastante en ejecutar no quiere decir que la solución al problema no sea convergente, sino que el programa debe ejecutar muchas líneas de código para alcanzar su objetivo.

10.3 Problemas varios

10.3.1 Robo de los implementos: El día 2 de Julio de 2004 fueron robados los elementos de hardware del presente trabajo de grado, incluyendo la pantalla de cristal líquido importada. Se realizó la denuncia pertinente de forma inmediata y se procedió a notificar del hecho a las partes interesadas por parte de la universidad. Gracias a fondos familiares se empezó a comprar los elementos necesarios para proseguir con las pruebas de visualización en el LCD.