

TECNOLÓGICO DE COSTA RICA
INTRODUCCION A LOS SISTEMAS EMBEBIDOS

PROYECTO 2

DISEÑO DE SoC MÍNIMO PARA RELOJ DESPERTADOR

DOCUMENTO DE DISEÑO

REALIZADO POR:

DAVID CORDERO CHAVARRÍA

EMANUEL ESQUIVEL LÓPEZ

RICARDO VALVERDE HERNÁNDEZ

PROFESOR:

LUIS BARBOZA

I SEMESTRE - 2021

CARTAGO, COSTA RICA

Índice

1. Documento de diseño de software	2
1.1. Primera solución	2
1.2. Segunda Solución	4
1.3. Solución elegida.	5
2. Herramientas de ingeniería	6
2.1. Altera Qsys	6
2.2. Eclipse	6
2.3. Makefile	6
3. Requerimientos	7
4. Referencias	10

Documento de diseño de software

1.1. Primera solución

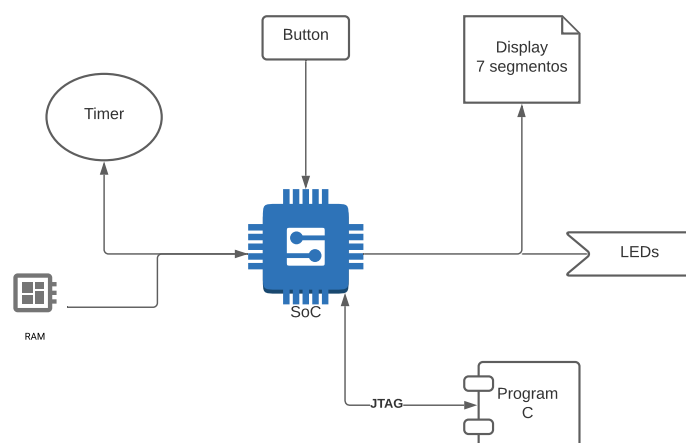


Figura 1: Diagrama de Arquitectura del sistema

En el diagrama de la figura 1 se puede ver como esta estructurado el sistema, como se ve las entradas a este son el programa principal el cual esta en C, este es cargado mediante conexión JTAG, desde una computadora, asi como tenemos de entrada botones los cuales funcionan para setear la alarma, tenemos un timer, el cual se encarga de llevar el control de la hora así como de salidas los leds y display de 7 segmentos.

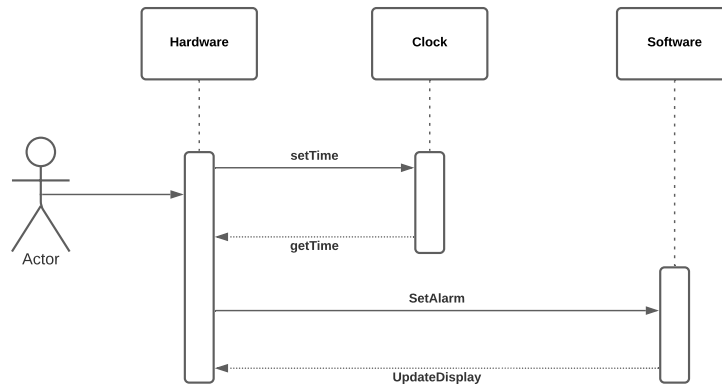


Figura 2: Diagrama de secuencia del sistema

Para este diagrama se puede ver que el usuario interactúa directamente con el hardware, el cual se utiliza para monitorear la alarma y esta interactúa directamente con el software el cual se encarga de mantener el control del tiempo.

1.2. Segunda Solución

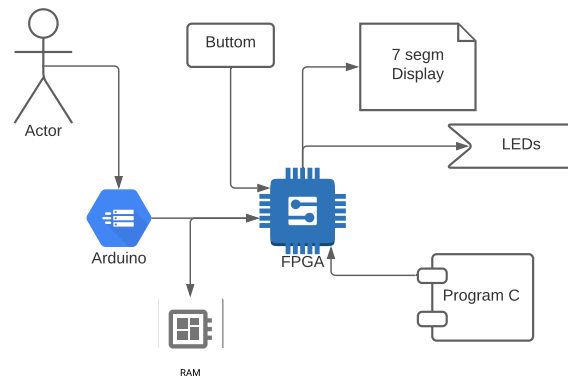


Figura 3: Diagrama de Arquitectura del sistema 2

Como se ve en la figura anterior prácticamente es la misma solución planteada en la parte 1, con la diferencia de que el usuario interactúa con un dispositivo externo, por lo que se eligió un arduino como primera instancia, ya que este es facil de programar y no tiene mucho costo.

Con este dispositivo se plantea la idea de solo setear desde este dispositivo la hora de la alarma así como su activación.

1.3. Solución elegida.

La solución elegida para la entrega final fue la primera, ya que como se vio a nivel de hardware es mucho más rentable el uso de solo la FPGA para el control de la alarma.

Aunque en la especificación se veía la funcionalidad de un dispositivo externo como obligatoria se vio un trabajo más grande a la hora de montar esto junto a la FPGA por lo que en sí se optó por tener un trabajo más completo con la FPGA que agregando más complejidad que podría generar errores.

El sistema con el dispositivo externo también se cambió ya que por situación actual de pandemia se veía más problemático la coordinación de los compañeros en tanto a hardware ya que no todos tenían acceso a este tipo de dispositivos.

Herramientas de ingeniería

2.1. Altera Qsys

Es una herramienta de alto nivel que se basa en la construcción de dispositivos así pudiendo elegir memoria reservada para tal propósito, además de esto permite de manera mas fácil la implementación de SoC mínimo.

2.2. Eclipse

Es un entorno de desarrollo con la posibilidad de generar código en C y así poder entrelazar con el proyecto de quartus Qsys, esto para mejorar la experiencia para desarrollar en C código ejecutable para un SoC.

2.3. Makefile

Básicamente es una receta la cual es generada automáticamente al ser cargado el sistema específico para el lenguaje de programación C en eclipse, por lo que este makefile es específico para esa aplicación dada.

Requerimientos

Numero	Nombre de requerimiento	Código
1	Formato de hora	A-01
2	Configuración de hora	A-02
3	Localización de luces	A-03
4	Memoria en chip	A-04
5	Alarma activa	A-05
6	NIOS II	S-01
7	SoC Minimo	H-01
8	Modulo de temporizador	H-02
9	Hardware externo	H-03

Tabla 1: Requerimientos mínimos del proyecto

Como se ve en la [tabla 2](#) los requerimientos están nombrados y con su respectivo código los cuales veremos a continuación:

- Alarma - 01:
El sistema debe mostrar la hora en formato BCD, de la forma HH:MM:SS para mayor comprension.
- Alarma - 02:
El sistema debe permitir la configuración de la hora mediante switches o botones.
- Alarma - 03:
Debe contar con un sistema de luces para identificar la alarma.
- Alarma - 04:
El sistema debe contar con una memoria en chip ya sea una para datos y otra para instrucciones o unificada.
- Alarma - 05:
El sistema debe permitir la configuración de una alarma de una manera similar al punto anterior,asi tambien se debe notificar al usuario el momento en el que la alarma se activa.
- Software - 01:
El sistema debe implementarse utilizando Nios II.
- Hardware- 01:
El sistema debe implementar un módulo temporizador por hardware para el conteo del tiempo.
- Hardware - 02:
El sistema debe contar únicamente con los componentes necesarios para el cumplimiento de los requerimientos establecidos así para poder cumplir la implementación de SoC mínimo.
- Hardware - 03:
La alarma debe poder ser controlada mediante un sistema externo capaz de controlar la hora de manera remota.

Numero	Nombre de requerimiento	Estado
1	Formato de hora	completado
2	Configuración de hora	completado
3	Localización de luces	completado
4	Memoria en chip	completado
5	Alarma activa	completado
6	NIOS II	completado
7	SoC Minimo	completado
8	Modulo de temporizador	completado
9	Hardware externo	Incompleto

Tabla 2: Requerimientos mínimos del proyecto completados

4

SECTION

Referencias

- [1] S. Pedre, “Sistemas embebidos,” *Laboratorio de Robótica y Sistemas Embebidos, Departamento de computación FCEN UBA*, 2017.
- [2] M. Desnoyers and M. R. Dagenais, “The ltng tracer: A low impact performance and behavior monitor for gnu/linux,” in *OLS (Ottawa Linux Symposium)*, vol. 2006. Citeseer, 2006, pp. 209–224.