



**ARQUITECTURA DEL COMPUTADOR**

**Presentado por:**

**ANA MARÍA GARCÍA POSSO  
ANDRES FELIPE DELGADO CHAMORRO  
DAVID ALEJANDRO SANCHEZ ARIAS**

**PONTIFICIA UNIVERSIDAD JAVERIANA**

**Presentado a:**

**JOSE OLIDEN SANCHEZ RAMOS**

**VALLE DEL CAUCA (CALI)**

**2018**

**ARQUITECTURA DEL COMPUTADOR  
PROYECTO DE SEMESTRE**

## INTRODUCCION

En el presente documento se pretende presentar el reporte final del proyecto del curso arquitectura del computador. Tomando en cuenta las correcciones recibidas en la entrega anterior y la investigación y aportes del grupo de trabajo, a continuación, se listan los puntos más importantes que se tratan en el documento.

- Visión de cómo se ve el proyecto finalizado.
- Objetivos cumplidos del proyecto.
- Datapath MIPS multiciclo completo.
- Software final
- Conclusiones

### Visión de cómo se ve el proyecto finalizado.

#### Declaración de entradas y salidas:

##### Entradas:

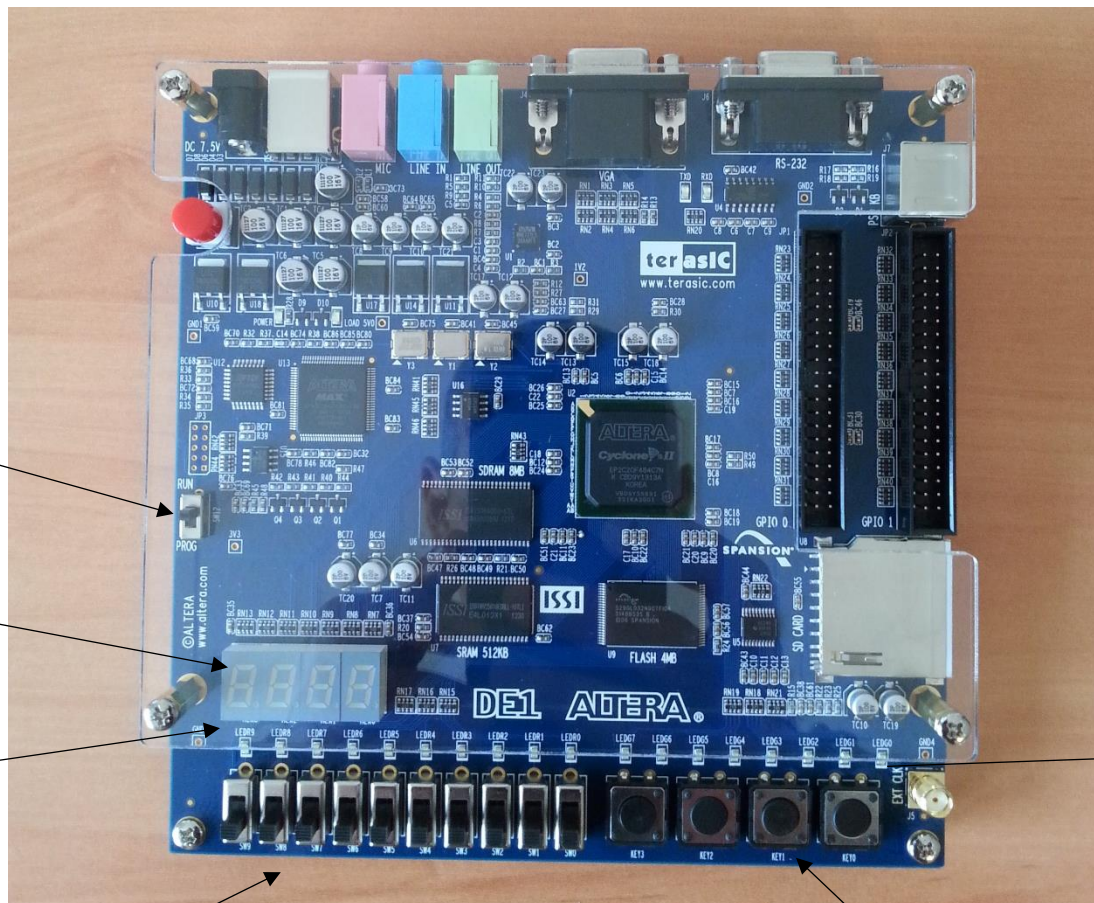
- **Switches:** El sistema utiliza cinco switches de la FPGA como entrada del sistema para programar las sesiones de entrenamiento, los circuitos y el número de descansos que el usuario desea realizar durante su sesión de ejercicio, también un switch que le permitirá al usuario pasar de modo programación a modo ejecución.
- **Pulsadores:** El sistema utiliza tres pulsadores para realizar la parte de la programación de las sesiones de ejercicio y los descansos. Un pulsador se utiliza para sumar tiempo de 10 minutos en 10 minutos a la duración de las sesiones, de igual manera el segundo pulsador realizara lo mismo, pero con la duración de los descansos, el tercer botón se emplea para dar aceptar la programación de las sesiones.

##### Salidas:

- **Leds:** El sistema emplea los leds verdes para que el usuario pueda visualizar el número de sesión en la que se encuentra, si se encuentra en modo programación se prendera un led determinado y si esta en modo ejecución se apagara el led que indica que esta en modo programación y se encenderá otro que indicara que el usuario se encuentra en modo ejecución del programa. Y el ultimo led verde lo emplea para indicarle al usuario si está en un circuito o en un descanso. El sistema emplea los leds rojos para indicarle al usuario el número de circuito en el que se encuentra y el número de descanso en el que se encuentra.

- **Siete segmentos:** El sistema utiliza los siete segmentos para mostrarle al usuario el tiempo en minutos y segundos de la sesión o el descanso en el que se encuentra.

Diagrama visualización final del sistema.



Switch  
run/prog

7 Segmentos

Leds rojos

Switches

Leds verdes

Pulsadores

## DESCRIPCIÓN GENERAL DEL SISTEMA A ENTREGAR:

El sistema construido como proyecto final del curso arquitectura de computadores tiene dos enfoques, un enfoque en cuanto al hardware que se llevó a cabo con ayuda de la herramienta FPGA o matriz de puertas programables y un enfoque de software basado en la programación a bajo nivel en MIPS. El puente entre estos dos enfoques fue el objetivo principal del proyecto.

El sistema es un dispositivo programable enfocado en la organización de un entrenamiento físico basado en sesiones divididas en circuitos con intervalos de descanso. El dispositivo debe generar avisos (visuales o sonoros) durante su ejecución los cuales fueron representados mediante los leds de la FPGA.

### Objetivos Generales cumplidos:

- **Elaboración de una versión propia del procesador MIPS:** Este es uno de los objetivos más importantes que se cumplió pues a partir de la correcta construcción de nuestra versión de MIPS se pudo ejecutar de manera correcta el software que se implementó. El proceso de construcción de dio gracias a los temas vistos en clase y a la investigación propia.
- **Aprender a usar la herramienta Quartus:** Este objetivo se cumplió ya que la mitad del proyecto dependió de este, aprender a manejar esta herramienta para poder programar el procesador de mips para el proyecto fue de bastante importancia para el desarrollo de este. Gracias a los laboratorios se pudo tener un acercamiento a esta herramienta y a cómo usarla para resolver el proyecto.
- **Implementación de MIPS multiciclo:** Uno de los objetivos importantes trazados fue el de la implementación del datapath de MIPS multiciclo, que gracias a los laboratorios y a la investigación propia del grupo lo pudimos lograr este importante objetivo.
- **Desarrollo software:** Uno de los objetivos que se alcanzó es el desarrollo de un software que contempla todas las posibilidades de programación del usuario , mas no las de ejecución,

## Datapath MIPS multiciclo.

En cuanto a hardware el equipo trabajó por etapas, ya que desarrollar un procesador como el de MIPS en una herramienta como quartus puede resultar dispendioso, y mucho más si no se prueba en la FPGA cada elemento que se codifique para comprobar su correcto funcionamiento.

A continuación se enlistan los logros del equipo paso a paso en el semestre.

- Implementación y prueba satisfactoria en FPGA de un ALU.
- Implementación y prueba satisfactoria en FPGA de un multiplexor.
- Implementación y prueba satisfactoria en FPGA de un registro.
- Unificación de ALU, multiplexor y registros más prueba satisfactoria en FPGA.
- Implementación Register File y prueba satisfactoria en simulador de Quartus.
- Implementación Memoria de instrucciones y prueba satisfactoria en simulador de Quartus.
- Implementación shift-left y prueba satisfactoria en simulador de Quartus.
- Implementación sign-extend y prueba satisfactoria en simulador de Quartus.
- Prueba satisfactoria camino de datos sin señales de control en FPGA.

Con lo anterior se completa un objetivo del proyecto.

Los siguientes logros del equipo fueron

- Implementación y prueba satisfactoria en FPGA de la máquina de estados
- Unión máquina de estados y camino de datos con prueba satisfactoria en simulador Quartus.

Una vez llegados a este punto el equipo tenía claro que iba a imaginar instrucciones y datos de 16 bits. A continuación se ilustra la estructura de las instrucciones:

### Tipo R:

OPCODE	Rs	Rt	Rd
[ 15 - 12 ]	[ 11 - 8 ]	[ 7 - 4 ]	[ 3 - 0 ]

### Lw y Sw:

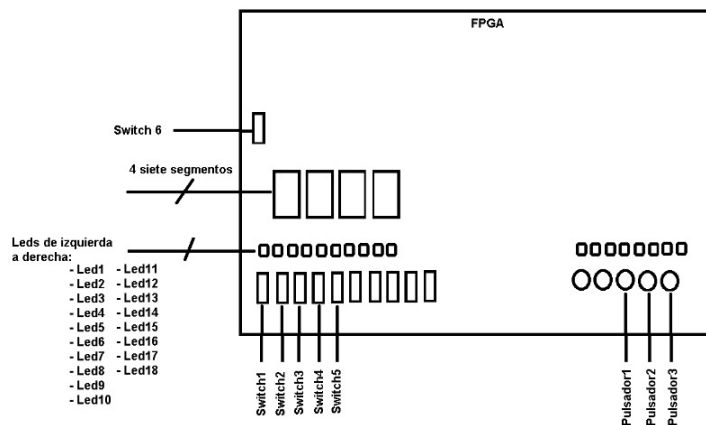
OPCODE	Rd	ADDRESS
[ 15 - 12 ]	[ 11 - 8 ]	[ 7 - 0 ]

## Jump:

OPCODE	ADDRESS
[ 15 - 12]	[ 11 - 0 ]

OPCODES	
Lw	0011
Sw	0101
Beq	0010
Jump	0001
Suma	0111
Resta	1001
And	1010
Or	1011

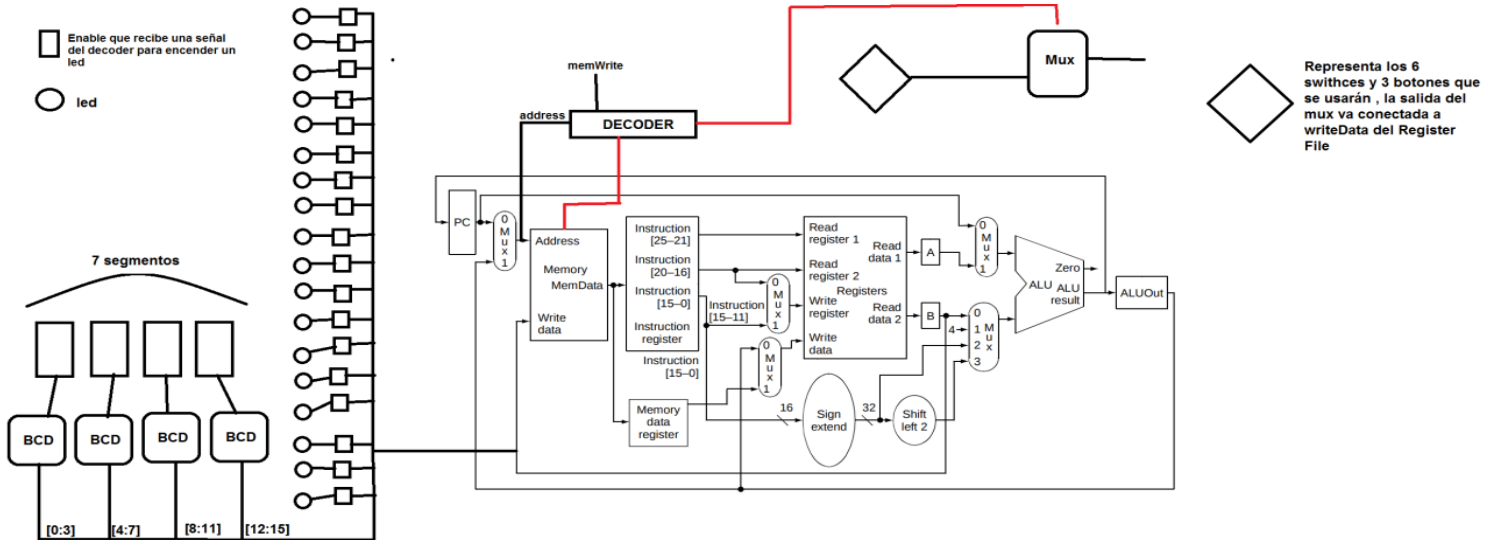
Finalmente el equipo necesitaba realizar una modificación extra en el mult ciclo , la cual era añadir un Decoder para poder trabajar con Los inputs y outputs brindados por la tarjeta de la FPGA. A continuación se ilustran los dispositivos I/O usados:



Dispositivo	Utilidad	Ubicación en memoria
Led1	Circuito 1 ejecutándose	M[ 32500 ]
Led2	Circuito 2 ejecutándose	M[ 32499 ]
Led3	Circuito 3 ejecutándose	M[ 32498 ]
Led4	Circuito 4 ejecutándose	M[ 32497 ]
Led5	Circuito 5 ejecutándose	M[ 32496 ]
Led6	Descanso 1 ejecutándose	M[ 32495 ]
Led7	Descanso 2 ejecutándose	M[ 32494 ]

Led8	Descanso 3 ejecutándose	M[32493]
Led9	Descanso 4 ejecutándose	M[32492]
Led10	Descanso 5 ejecutándose	M[32491]
Led11	Cicuito :On,Descanso: Off	M[32490]
Led12	Modo Programación	M[32489]
Led13	Modo ejecución	M[32488]
Led14	Sesión 1 ejecutándose	M[32487]
Led15	Sesión 2 ejecutándose	M[32486]
Led16	Sesión 3 ejecutándose	M[32485]
Led17	Sesión 4 ejecutándose	M[32484]
Led18	Sesión 5 ejecutándose	M[32483]
Switch1	ejecutar/programar sesión 1	M[32482]
Switch2	ejecutar/programar sesión 1	M[32481]
Switch3	ejecutar/programar sesión 1	M[32480]
Switch4	ejecutar/programar sesión 1	M[32479]
Switch5	ejecutar/programar sesión 1	M[32478]
Switch6	modoProg:Off,modRun;On	M[32477]
Btn1	Aceptar(fin programación circuito/descanso)	M[32476]
Btn2	Incremento 10 minutos	M[32475]
Btn3	Incremento 10 segundos	M[32474]
7 segmentos	Mostrar tiempo ejecución	M[32473]

Con estas asignaciones (las cuales son muy importantes para el correcto entendimiento del software desarrollado para el proyecto) el equipo generó el diagrama con las nuevas modificaciones del multiciclo que se presenta aquí:



Estas últimas modificaciones no pudieron ser probadas con resultados exitosos lo cual imposibilitó la unión de hardware y software.

## Software final.

En cuanto al software final desarrollado para el proyecto se entrega información relevante y código junto a este documento, aclarar que el software fue probado y presenta correctitud en el estado de programación del usuario, pero a falta de tiempo el equipo no pudo probar por completo la correctitud del software en estado de ejecución por lo cual no se incluye.

En el documento adjunto a este se profundiza más en los recursos utilizados en el código, cada registro está creado específicamente para el proyecto.

El código puede pasarse a Binario para luego ingresarlo al procesador creado mediante un programa desarrollado en el lenguaje de programación Python que contiene un diccionario personalizado con todas las direcciones y representaciones binarias de los registros del proyecto.



## CONCLUSIONES

- Debido a que nos encaminamos a una solución muy complicada para el tema de entradas y salidas del dispositivo de entrenamiento con sesiones de ejercicio y descansos como lo fue el uso de un puerto serial que es un dispositivo complicado de usar, perdimos tiempo al darnos cuenta de que este camino para el proyecto era difícil de lograr y replantear el cómo íbamos a solucionar el tema de entradas y salidas para el proyecto nos dimos cuenta que la mejor opción para el proyecto era tomar los componentes de la FPGA como los dispositivos de entrada y salida proyecto.
- Las herramientas como el lenguaje maquina MIPS y el lenguaje de especificación VHDL son muy útiles para realizar sistemas y circuitos complejos, ya que en el caso de VHDL podemos hacer circuitos digitales y hacer pruebas con este, simular el sistema, y esto es muy útil ya que si deseamos realizar un sistema complejo es mejor mirar cómo se comportaría la solución que estamos implementando, así podemos evitar dañar componentes o no encaminarnos por una solución poco viable.
- Podemos concluir que, aunque no se lograron cumplir varios de los objetivos planteados en un principio en el proyecto al final el equipo logró que software y hardware se relacionarán lo cual en anteriores entregas fue imposible por falta de claridad de ideas.
- Gracias a los temas vistos en clase y en los laboratorios para la elaboración del proyecto, se puede ver la gran aplicación de lenguajes de máquina y de bajo nivel como MIPS y ARM y del lenguaje de especificación vhdl, para la elaboración de procesadores y circuitos digitales. Varios de los procesadores de los dispositivos actuales se basan en lenguajes de bajo nivel como MIPS y ARM, como ejemplo vemos los procesadores de los dispositivos móviles que la mayor parte de estos están diseñados con la arquitectura de ARM.