

Proyecto final

Dak-G



Materia: Arquitectura de computadoras.

Equipo: Alan Guzman, Kevin Sanchez, Diego Quintero, Gabriel Alcaraz.

Fase 2

Cronograma de actividades

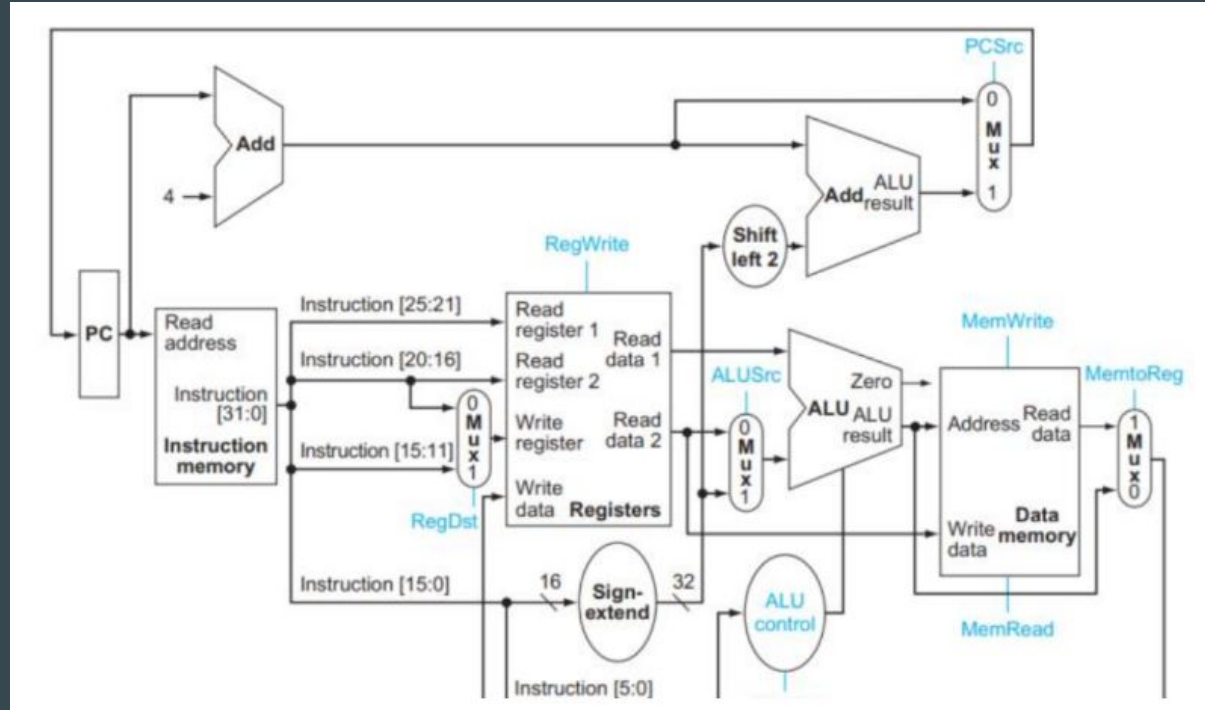
Integrante	Rol	Cosas por hacer	Fecha de Entrega
Alan	Administración	-Administrar que se cumplan con las fechas de entrega. -Hacer presentación describiendo los procesos , fallas y contratiempos que se tuvieron.	Miércoles: Verificar que se hayan llegado a las metas para poder empezar la presentación. Viernes o sábado: Aventajar la presentación para corregir errores y que no se nos junte el trabajo. Domingo: Pulir detalles finales y corregir presentacion (de ser necesario).
Diego	Verilog	- Crear los módulos nuevos para	Miercoles: Creación de los módulos. Viernes: Implementación de las funciones para los módulos. Domingo: Pulir detalles y corregir errores
Kevin	Python	- Crear en python el decodificador para instrucciones tipo I	Miércoles: Instrucciones I. Viernes: Tener un avance del editor de texto en python que pueda escribir datos. Domingo: Hacer correcciones del editor de texto.
Gabriel	Documentación	-Hacer investigación de little endian y big endian	Miércoles: Investigación Domingo: Reporte

Objetivos

General

El objetivo principal es implementar un “single datapath de MIPS de 32 bits para ejecutar instrucciones tipo R.

Con nuevos módulos que complementaran el datapath realizado en la fase anterior para lograr más puertas para que sea mas específico su funcionamiento



Objetivos específicos

- Agregar módulos en verilog al proyecto de la fase 1.. Módulos como PC, add, Memoria de instrucciones, mux5_bits_2a1, mux6_bits_2a1, shift_left_2, sign_extend.
- Crear TestBench para algunos de los módulos antes mencionados.
- Desarrollar una presentación, en un archivo README.md, en el repositorio remoto en el cual se estarán subiendo los archivos y carpetas que se estarán trabajando a lo largo del proyecto.
- Desarrollar un decodificador que sea capaz de leer archivos para conectarlos con el proyecto verilog.

Entregables

Entregables [Reporte].pdf

- Elementos y características generales del procesador MIPS de 32 bits,
- Set de instrucciones y definición de set de instrucciones que podrán ejecutarse en su diseño,
- Describir el desarrollo de los elementos implementados en el punto anterior.

*Hay que leer la información y describirla con sus propias palabras.

Código en Verilog Archivos Verilog –

PC.v, Sumador.v, Memoria_de_instrucciones.v, mux5_bits_2a1.v, mux6_bits_2a1.v, shift_left_2.v, sign_extend.v, tb_MemoriaDeInstrucciones.v, tb_Multiplexor_6_bits_2_a_1.v, tb_PC.v, tb_Sign_Extend.v, tb_Sumador.v

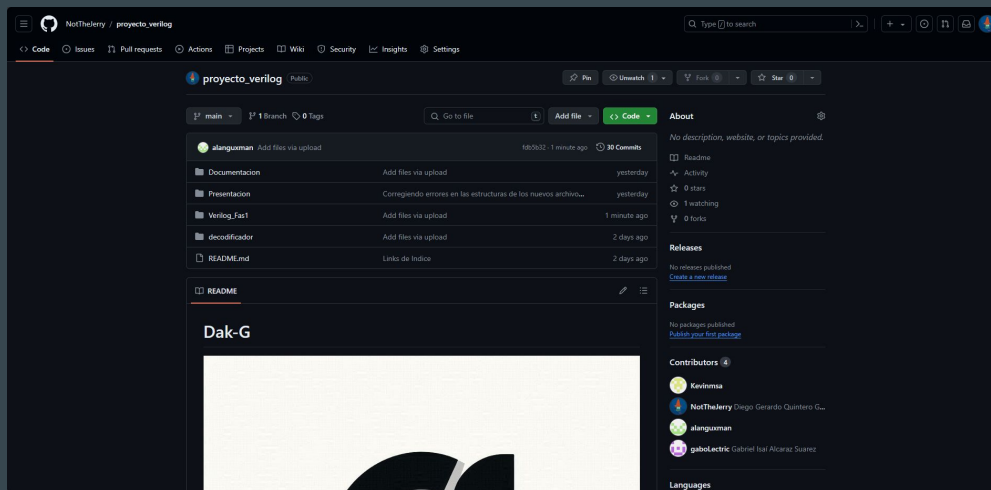
El reporte debe de presentarse con la simulación donde los resultados deben visualizarse en formato (ó radix) decimal.

*Nota: Los nombres de los archivos “.v”, módulos y elementos de entrada y salida son opcionales, los aquí presentados son ejemplos.

Github

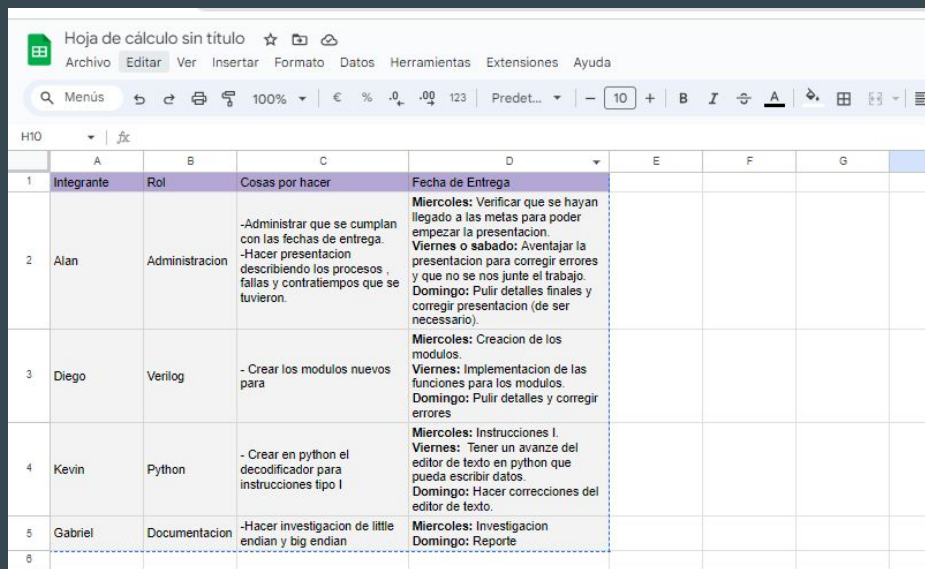
Link al repositorio remoto de nuestro proyecto

https://github.com/NotTheJerry/proyecto_verilog



Trabajo realizado por: Alan - Management

- Presentación de PowerPoint, donde se describen los roles y el cronograma de la realización de actividades durante la semana..
- Revisión y apoyo durante la semana en avances del proyecto.



	A	B	C	D	E	F	G
1	Integrante	Rol	Cosas por hacer	Fecha de Entrega			
2	Alan	Administracion	-Administrar que se cumplan con las fechas de entrega. -Hacer presentacion describiendo los procesos , fallas y contratiempos que se tuvieron.	Miercoles: Verificar que se hayan llegado a las metas para poder empezar la presentacion. Viernes o sabado: Avenatar la presentacion para corregir errores y que no se nos junte el trabajo. Domingo: Pulir detalles finales y corregir presentacion (de ser necesario).			
3	Diego	Verilog	- Crear los modulos nuevos para	Miercoles: Creacion de los modulos. Viernes: Implementacion de las funciones para los modulos. Domingo: Pulir detalles y corregir errores			
4	Kevin	Python	- Crear en python el decodificador para instrucciones tipo I	Miercoles: Instrucciones I. Viernes: Tener un avance del editor de texto en python que pueda escribir datos. Domingo: Hacer correcciones del editor de texto.			
5	Gabriel	Documentation	-Hacer investigacion de little endian y big endian	Miercoles: Investigacion Domingo: Reporte			
6							

Trabajo realizado por: Diego - Verilog

- Se realizaron los módulos de los nuevos componentes como:
Multiplexor de 5 bits, PC, Sumador de 32 bits, Shift left 2, SignExtend, Unidad de Control (se agregaron las nuevas salidas).
- Se hizo el testbench de cada uno de los nuevos módulos para que se mostrara de forma correcta el funcionamiento de cada nuevo módulo.
- Y se interconectaron todos los módulos creados en esta etapa junto con los de las anteriores para que de esta manera quedará el esquema establecido para esta etapa.

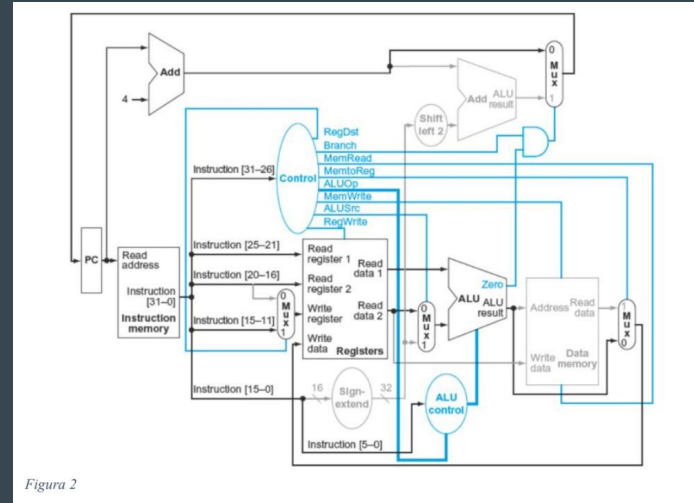


Figura 2

Trabajo realizado por: Kevin - Decodificador (Python)

- Abrir un archivo de ensamblador, mostrarlo en la ventana y modificarlo
- Guardar el archivo modificado
- Convertir el archivo a binario
- Generar el archivo binario para cargar la memoria de instrucciones
- Generar el archivo para inicializar el banco de registros y memoria de datos

Trabajo realizado por: Gabriel - Documentación

- Investigación de endianness (Big endian y little endian), Single Cycle Data Path
- Investigación tipos de Arquitectura Von Newman y Harvard
- Investigación instrucciones tipo R
- Definición de las instrucciones para nuestro proyecto
- Guía de Usuario para el decodificador de fase 1.2

Nuevos roles para la fase 2

- Documentación - Alan Guzman
- Decodificador - Diego Quintero
- Verilog - Kevin Sanchez
- Management - Gabriel Alcaraz