República Bolivariana de Venezuela

Universidad de Carabobo

Facultad de Ingeniería

Escuela de Ingeniería Eléctrica y de Telecomunicaciones

Departamento de Sistemas y Automática

Cátedra de Lógica Digital

**Práctica # 5 – Sesión #02**

**Objetivo: Implementar circuitos secuenciales utilizando Flip-Flops, VHDL y la tarjeta de desarrollo BASYS2**

Sección #05 de Laboratorio

**Integrantes:**

Carlos Hernández

C.I.: 25.829.471

Gianfranco Gasbarri

C.I.: 26.654.860

Fecha de entrega: 4/06/19

**Pre-Laboratorio**

1. **Actividad 1:** Ver **Anexo 1**
2. **Actividad 2:** Ver **Anexo 3**
3. **Actividad 3:** Ver **Anexo 3**

**Laboratorio**

Ver **Anexo 4**

**Conclusiones**

Si deseáramos implementar los flip-flops diseñados en esta práctica utilizando compuertas lógicas, debemos hacer los circuitos tal como se muestra en el **Anexo 5.1** para el flip-flop JK y el **Anexo 5.2** para el flip-flop D. Podemos observar que los mismos están construidos sobre los biestables SR, los cuales fueron estudiados en las clases de teoría. Con el lenguaje VHDL, encontramos la facilidad de poder definir dichos componentes simplemente escribiendo su función lógica cuando un evento de flanco de reloj sea detectado, tal como se puede ver en sus respectivas definiciones en el Pre-Laboratorio.

En la práctica anterior, se realizó un circuito secuencial pero a través del diagrama de estados que brinda la interfaz de desarrollo Active-HDL. A través de dicha herramienta, se puede definir el comportamiento del circuito mediante un único paso que es el diseño del diagrama de estados (el cual ya está contenido en un diseño completo de estas unidades de control), con lo cual resulta más práctico y sencillo de implementar que utilizando flip-flops, tal como se hizo en la presente práctica. Se ahorra la elaboración de una tabla de flujo, la obtención de las funciones lógicas y el diseño del circuito.

La dinámica de cambio de estados del circuito depende de varias señales: en primer lugar, cuando se activa la señal **Inicio**, el sistema pasa del estado inicial a la etapa de Llenado. El sistema seguirá llenando hasta que el sensor de nivel **N1** indique que se debe empezar a mezclar y dejar de llenar. En esta etapa, similar a la etapa anterior se seguirá realizando la actividad hasta que el sensor **N2** indique lo contrario. Sin embargo, existe un sensor de temperatura **T** que al activarse, el sistema deberá detenerse y regresar al estado inicial, independientemente del valor de **N2** (de acuerdo a las consideraciones tomadas por nosotros).

Una vez superada la etapa de Mezclado, el sistema pasará a una última etapa de Vaciado que existirá hasta que en el sensor de nivel **N3** indique que el nivel de la mezcla está por debajo del mismo (**N3 = 0)** y se habrá completado un ciclo del sistema volviendo al estado inicial.

Las salidas del sistema **V1, V2, V3 y M** son salidas de tipo Moore puesto que únicamente dependen del estado en el que se encuentra el sistema, en cambio la salida **A** es de tipo Mealy ya que la misma se activa durante una transición de un estado a otro, y por lo tanto, dependen de entradas del sistema (**T y N3** para ser más específicos).

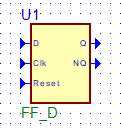
Los diseños implementados en la práctica fueron muy similares a los de otros circuitos secuenciales desarrollados en clase. Los mismos requirieron componentes construidos previamente como lo fueron **ClkDiv**, para dividir la frecuencia del reloj y así poder apreciar los cambios de estado y las salidas; y **MUX41** para las entradas de los flip-flops.

ANEXOS

**Anexo 1**

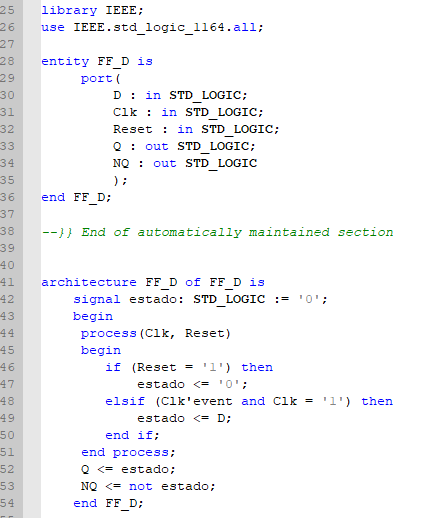
**Anexo 1.1**

**Símbolo del componente**

****

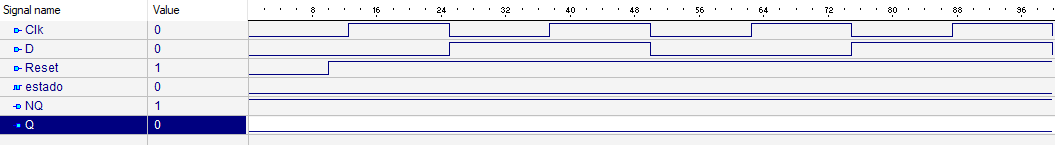
**Anexo 1.2**

**Código VHDL**

****

**Anexo 1.3**

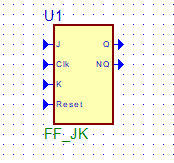
**Simulación**

****

**Anexo 2**

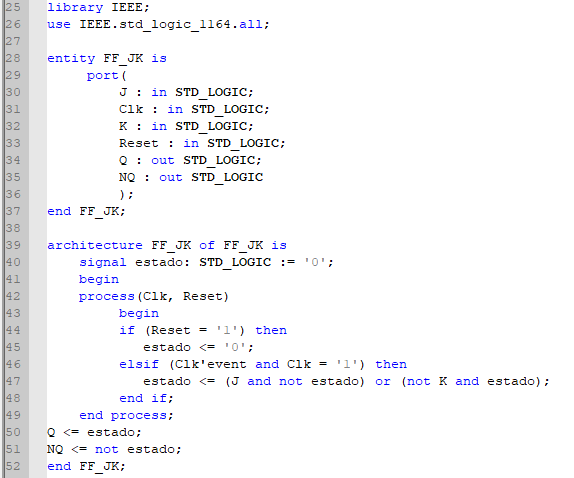
**Anexo 2.1**

**Símbolo del componente**

****

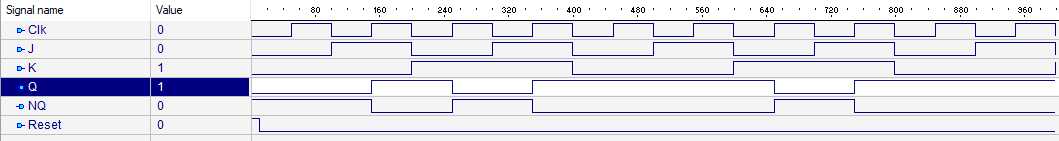
**Anexo 2.2**

**Código VHDL**

****

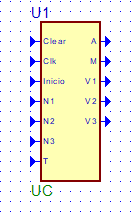
**Anexo 2.3**

**Simulación**

****

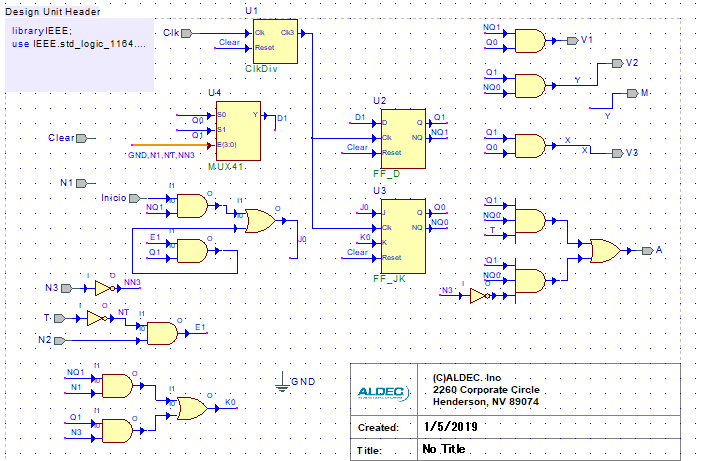
**Anexo 3**

**Anexo 3.1**

**Símbolo del componente**

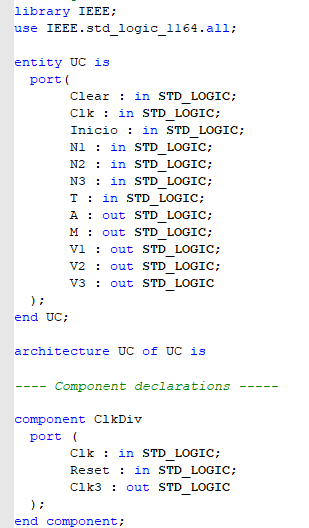
**Anexo 2.2**

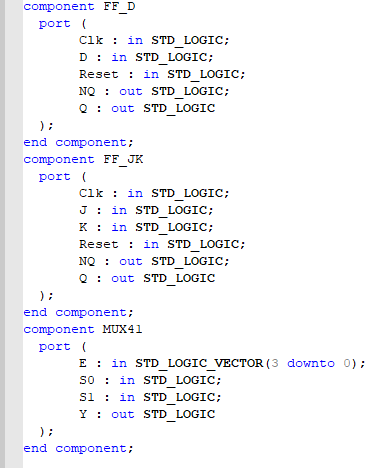
**Diseño**

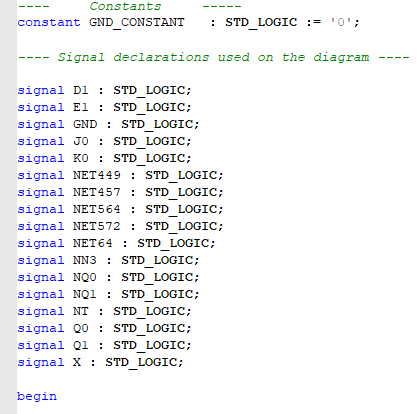
****

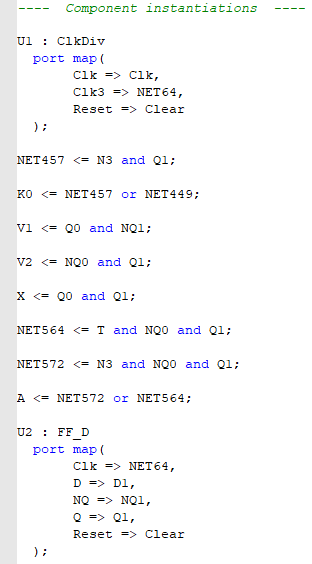
**Anexo 2.3**

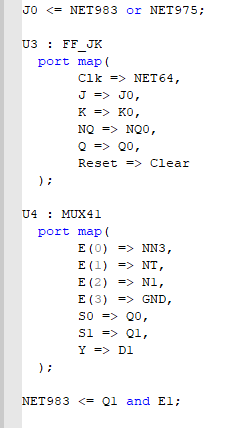
**Código VHDL**

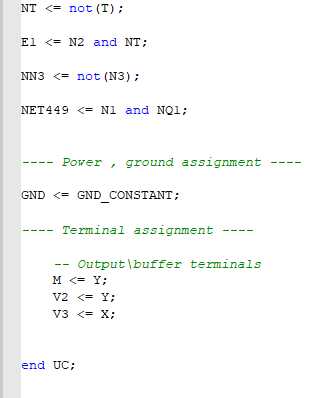
****

****

****

****

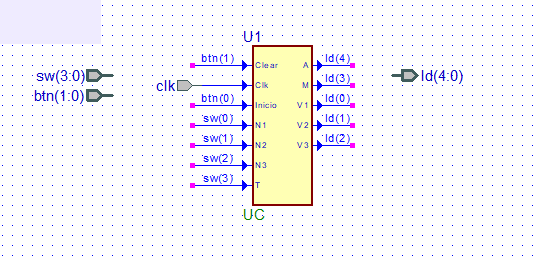
****

****

**Anexo 4**

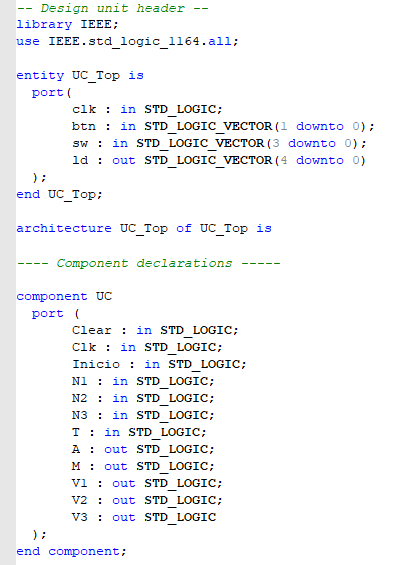
**Anexo 4.1**

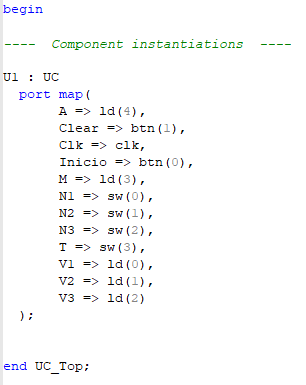
**Diseño del alto nivel**

****

**Anexo 4.2**

**Código VHDL**

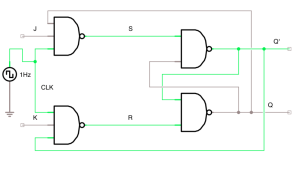
****

****

**Anexo 5**

**Anexo 5.1**

**Flip-flop JK**



**Anexo 5.2**

**Flip-flop D**

