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 # 3 – Sesión Única**

**Objetivo: Implementar circuitos combinacionales aritméticos utilizando 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: 26/02/19

**Pre-Laboratorio**

1. **Actividad 1: Sumador Completo (Full Adder) en LogicWorks**

Ver **Anexo 1**

1. **Actividad 2: Algoritmo de la resta de 2 números binarios con complemento a 2**

Para restar 2 números enteros binarios con aritmética de complemento, se debe seguir los siguientes pasos:

* Se debe establecer un orden de magnitud para ambos operandos. El mismo puede adaptarse a cierto problema planteado o simplemente al operando con mayor cantidad de bits.
* Se obtiene el complemento a 2 del sustraendo (B) con tanta cantidad de bits como indique el orden de magnitud. En caso de ser necesario, completar con 0´s en las cifras más significativas del operando A para adecuarse al orden de magnitud.
* Realizar la suma del operando A con el complemento a 2 de B.
* Una vez hecha la suma, se deberá tomar una de las siguientes alternativas de acuerdo al acarrea por fuera del orden de magnitud:
  + Si el acarreo es igual a 1, la resta es positiva y la magnitud de la misma se leerá directamente.
  + Si el acarreo es 0, la resta es negativa y para leer la magnitud del resultado de la misma se deberá obtener su complemento a 2

1. **Actividad 3: Componente Comparador4 con VHDL**

Ver **Anexo 2**

**Post-Laboratorio**

1. **Tabla de resultados de las pruebas del sumador completo:** Ver **Anexo 1.5**
2. **Diseño y programa VHDL del sumador completo, diseño y programa VHDL del sumador de 4 bits, simulación y tabla de resultados:** Ver **Anexo 3**
3. **Diagrama de bloque y programa VHDL del comparador de magnitud, simulación y tabla de resultados:** Ver **Anexo 2**
4. **Diagrama de bloque y programa VHDL de la unidad aritmética; diseño y programa VHDL de alto nivel, tabla de resultados:**

**ANEXOS**

**Anexo 1**

**Anexo 1.1**

**Diagrama de caja negra**

**A**

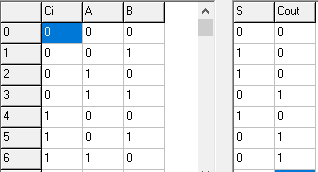
**S**

**B**

**Full Adder**

**Anexo 1.2**

**Tabla de la verdad**

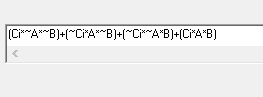


**Anexo 1.3**

**Expresiones lógicas**

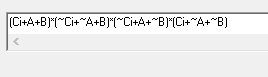
**Anexo 1.3.1**

**SOP Simplificada**



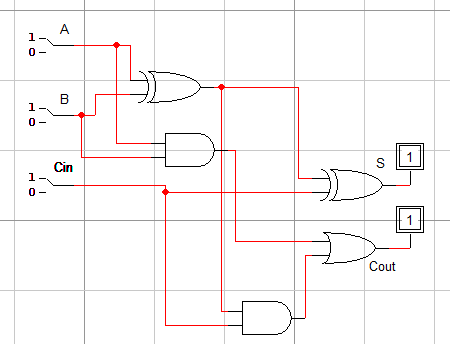
**Anexo 1.3.2**

**POS Simplificada**



**Anexo 1.4**

**Diseño**



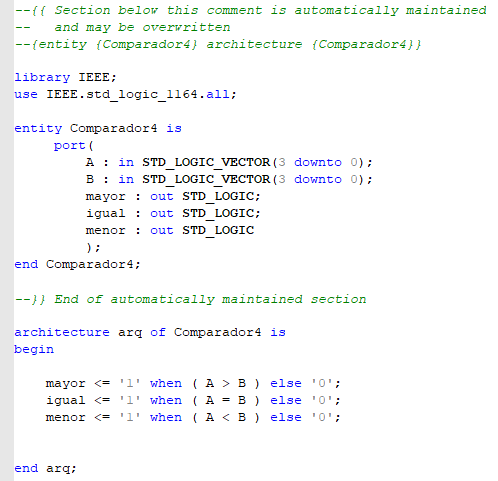
**Anexo 1.5**

**Tabla de resultados**

**Anexo 2**

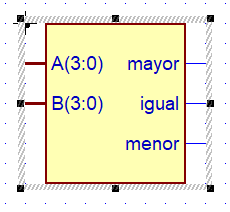
**Anexo 2.1**

**Código VHDL**



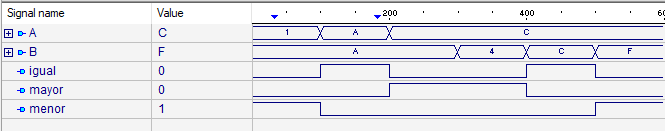
**Anexo 2.2**

**Diagrama de bloques**



**Anexo 2.3**

**Simulación**



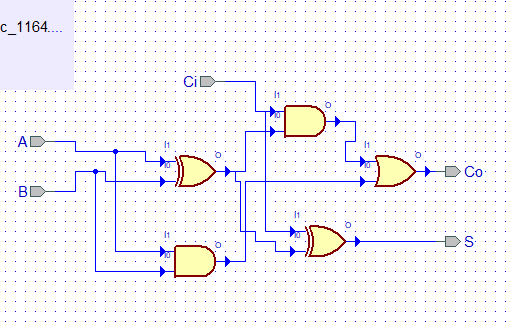
**Anexo 2.4**

**Tabla de resultados**

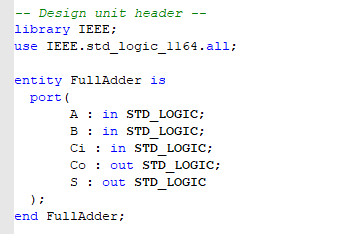
**Anexo 3**

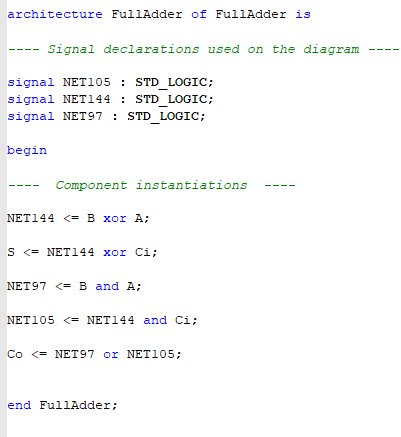
**Anexo 3.1**

**Diseño del sumador completo**



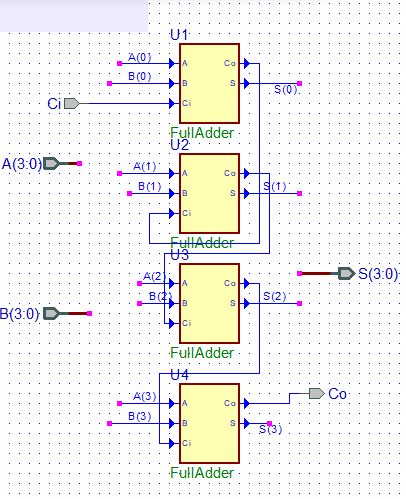
**Anexo 3.2**

**Código VHDL del sumador completo**



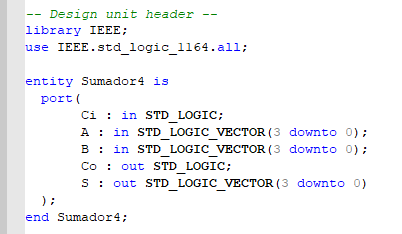
**Anexo 3.3**

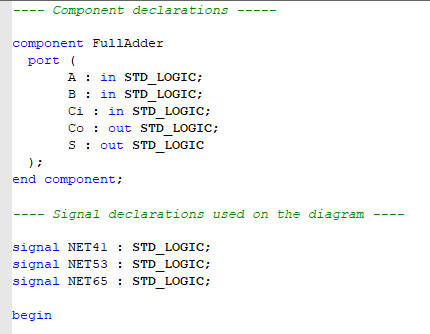
**Diseño del sumador de 4 bits**

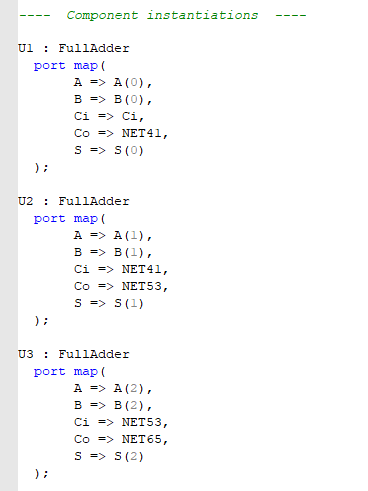


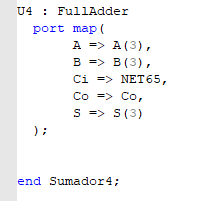
**Anexo 3.4**

**Código VHDL del sumador de 4 bits**

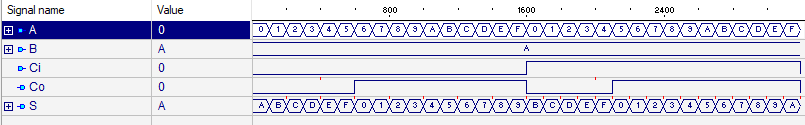








**Anexo 3.5**

**Simulación**