Logotipo

Descripción generada automáticamente

DISEÑO DIGITAL 2

MEMORIA DEL DISEÑO

**Autores:**

Ricardo Javier Cuenca Cuenca,

Lukas Sylvester Gdanietz de Diego,

Hao Feng Chen Fu

**Curso** 2021-2022.

**Control de versiones : V5.0**

**Control de versiones**

|  |  |  |  |
| --- | --- | --- | --- |
| Versión | Fecha | Autor | Cambios realizados |
| 0.0 | 19/04/2022 | DTE | Inicial |
| 0.0 | 03/05/2022 | G4 | Inicio, donde se realizo la estructura de la memoria |
| 1.0 | 09/05/2022 | G4 | Se realizo la introducción y el objetivo del proyecto |
| 2.0 | 18/05/2022 | G4 | Modificaciones en el apartado 2, apartado 4 y asignaciones de pines |
| 3.0 | 20/05/2022 | G4 | Introducción inicial de especificaciones del apartado 1, introducción del nivel jerárquico del apartado 2, modificación y finalización del apartado 4, modificación de los apartados 5.2, 5.3 y 5.4 |
| 4.0 | 21/05/2022 | G4 | Modificación del apartado 1, apartado 2 y apartado 3 |
| 5.0 | 22/05/2022 | G4 | Revisar |

Tabla de contenido

[1 Especificación del diseño. 4](#_Toc446353295)

[1.1 Introducción 5](#_Toc69038455)

[1.2 Interfaces 5](#_Toc318390073)

[1.2.1 Interfaz con el acelerómetro 5](#_Toc1291863287)

[1.2.2 Interfaz con la barra de displays de 7 segmentos 6](#_Toc1626897415)

[1.2.3 Interfaz con la barra de LEDs 6](#_Toc735004845)

[1.3 Especificaciones 7](#_Toc1424605084)

[2 Diseño jerárquico 7](#_Toc202352561)

[2.2 Bloque controlador del acelerómetro 9](#_Toc592456266)

[2.3 Bloque Calibrador 10](#_Toc34830236)

[2.4 Bloque Axis\_Screen\_X 10](#_Toc1554926358)

[2.5 Bloque Axis\_Screen\_Y 11](#_Toc1388277746)

[3 Diseño detallado 11](#_Toc153682844)

[4 Pruebas de verificación funcional de NIVEL 12](#_Toc1856334946)

[4.1 Test del Máster SPI 13](#_Toc1746019133)

[4.2 Test del controlador 13](#_Toc2096117180)

[4.3 Test del diseño completo 14](#_Toc1071427463)

[5 Verificación del diseño final 15](#_Toc759780084)

[6 Diseño físico 15](#_Toc2134812616)

[6.1 Asignación de pines 16](#_Toc769377540)

[6.2 Restricciones de la síntesis 16](#_Toc1168973222)

[6.3 Recursos utilizados 17](#_Toc1554865563)

[6.4 Frecuencia máxima de reloj 17](#_Toc2073458881)

[7 Bibliografía 17](#_Toc75574290)

# Especificación del diseño.

## Introducción

El Grupo G4 de Diseño Digital II con la guía del profesorado de la Asignatura, y con el conocimiento adquirido, se ha realizado el diseño en VHDL de un acelerómetro que medirá el movimiento de un sensor, en el eje X e Y, este movimiento será mostrado en una barra de leds para el eje **X** y en una barra de Displays que se encuentran en la **XDECA** proporcionada en el laboratorio, las características de esta tarjeta como las del sensor serán dadas con más detalle en los puntos siguientes.

El sensor **LIS2DH12** que está integrado en la Tarjeta **XDECA** llamándose en esta tarjeta como **Gsensor,** será un acelerómetro lineal para tres ejes (XYZ) controlable mediante una interfaz SPI/I2C, este sensor tiene una amplia variedad de funcionalidades que pueden ser usadas, sin embargo, en este diseño se utilizaran solo algunas de las posibilidades que nos da este sensor y su control será hecho mediante **interfaz SPI**.

## Interfaces

El Sistema se interconectara al acelerómetro, consiguiendo realizar medidas de la posición del sensor en los ejes X e Y, mostrándolo en una barra de leds para el eje X y en una barra de displays de 7 segmentos para el eje Y.

El diagrama de bloques general del diseño se muestra en la siguiente figura.

Imagen que contiene Interfaz de usuario gráfica

Descripción generada automáticamente

*Fig. 1. Interconexión del NIVEL al acelerómetro y a los LEDs y displays de 7 segmentos.*

### Interfaz con el acelerómetro

El sistema se comunica con el acelerómetro (**LIS2DH12**) que actuara como esclavo del sistema, con una interfaz SPI. Las señales de esta interfaz son las siguientes:

|  |  |  |
| --- | --- | --- |
| Señal | Dirección | Descripción |
| CS | Salida | Selección de chip, activa a nivel alto |
| SPC - SCK | Salida | Reloj en serie |
| SDI – MISO | Entrada | Entrada de datos en serie |
| SDO-MOSI | Salida | Salida de datos en serie |

### Interfaz con la barra de displays de 7 segmentos

El sistema realiza la visualización de la inclinación en uno de los ejes utilizando una barra de displays de 7 segmentos del tipo cátodo común. La interfaz es la siguiente :

|  |  |  |
| --- | --- | --- |
| Señal | Dirección | Descripción |
| seg[7..0] | salida | seg[0] : segmento g  seg [1] : segmento f  seg [2] : segmento e  seg [3] : segmento d  seg [4] : segmento c  seg [5] : segmento b  seg [6] : segmento a  seg [7] : segmento punto |
| mux\_disp[7:0] | salida | mux\_disp [0] : cátodo del display 0 (LSD)  mux\_disp [1] : cátodo del display 1  mux\_disp [2] : cátodo del display 2  mux\_disp [3] : cátodo del display 3  mux\_disp [4] : cátodo del display 4  mux\_disp [5] : cátodo del display 5  mux\_disp [6] : cátodo del display 6  mux\_disp [7] : cátodo del display 7 |

La interfaz permite iluminar solo un display a la vez. El display se selecciona activando (a nivel bajo) el cátodo correspondiente. El display activo se ilumina de acuerdo al código de 7 segmentos y punto decimal introducido (nivel alto).

### Interfaz con la barra de LEDs

El sistema realiza la visualización de la inclinación en el otro eje utilizando una barra de displays de 7 segmentos:

|  |  |  |
| --- | --- | --- |
| Señal | Dirección | Descripción |
| led\_disp[7..0] | Salida | Se activará en nivel alto, uno por uno los leds dependiendo del grado de inclinación |

## Especificaciones

Las especificaciones funcionales y no funcionales del NIVEL se detallan en el documento [1].

1. Especificaciones funcionales

|  |  |
| --- | --- |
| Ref | Especificación |
| ESP00 | El sistema realiza detecciones de movimiento |
| ESP01 | Iluminación inicial de todos los leds y displays de 7 segmentos que aseguran que se toma un punto de referencia. |
| ESP02 | Iluminación o apagado de los leds o displays de 7 segmentos en función del movimiento respecto al eje X o al eje Y. |
| ESP03 | Iluminación correcta de los leds |

1. Especificaciones no funcionales

|  |  |
| --- | --- |
| Ref | Especificación |
| ESP04 | El sistema se diseñará utilizando VHDL. |
| ESP05 | Se utilizará *ModelSim* como herramienta de simulación y *Quartus Prime* como herramienta para la realización del diseño físico, |
| ESP06 | El sistema se prototipará utilizando una tarjeta DECA-MAX10 del fabricante Arrow [2] |
| ESP07 | Se utilizará como fuente de reloj uno de los osciladores de 50MHz que posee la tarjeta DECA-MAX10. |
| ESP08 | Para prototipar el sistema se conectará a la tarjeta DECA-MAX10 una tarjeta de expansión [3] con 8 displays |

# Diseño jerárquico

Diagrama, Esquemático

Descripción generada automáticamenteEl diagrama de la Fig. 2 representa el primer nivel de la jerarquía del diseño[[1]](#footnote-2):

*Fig. 2. Diagrama de bloques del primer nivel de la jerarquía de NIVEL.*

En los siguientes subapartados se describe la interfaz y la función de cada uno de estos bloques.

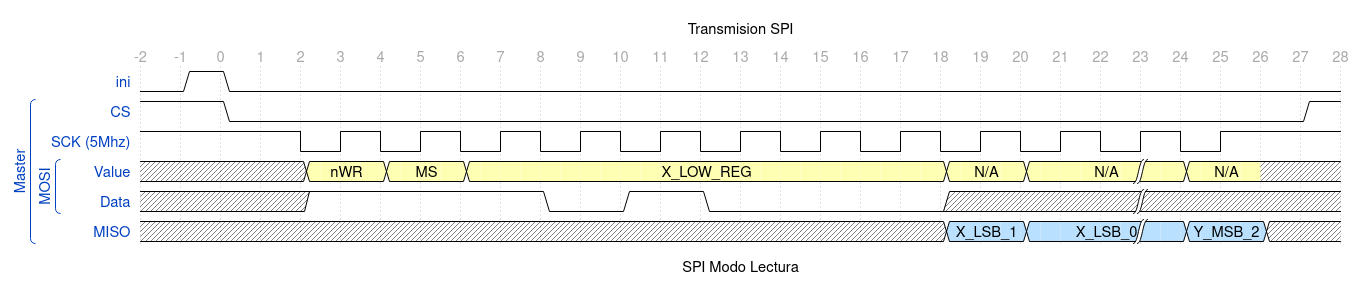
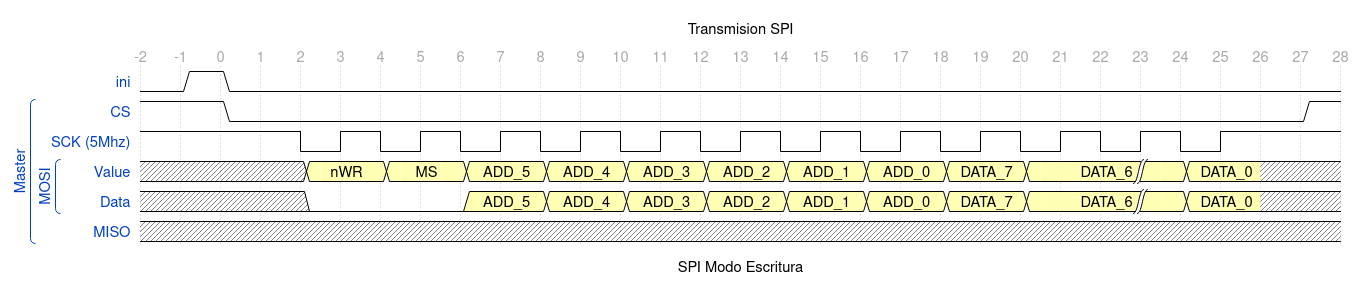
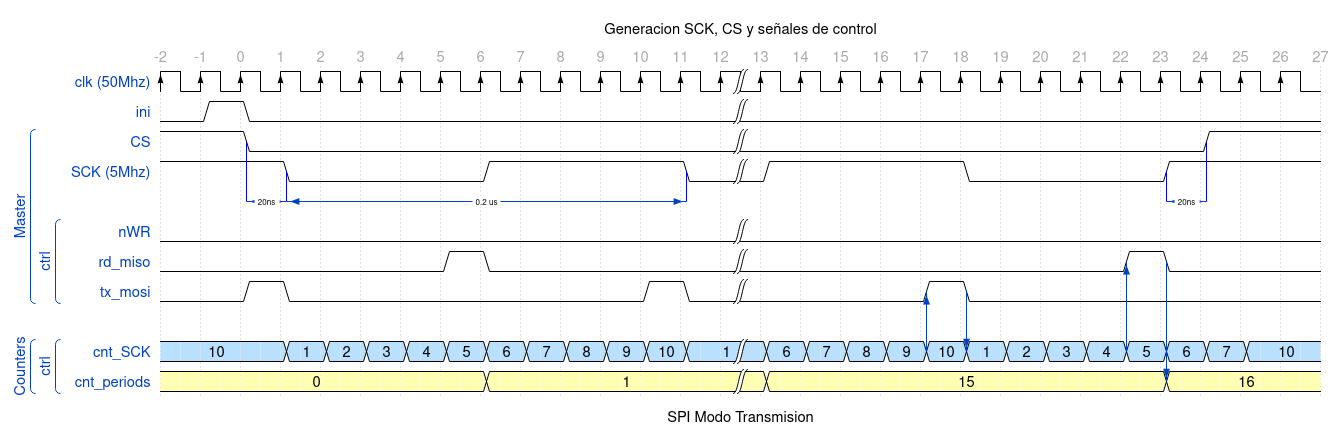
* 1. ***Bloque Máster SPI***

|  |  |  |
| --- | --- | --- |
| Señal | Dirección | Descripción |
| CS | salida | Señal del Chip select |
| SCK | salida | Señal de reloj del interfaz SPI |
| MOSI | salida | Buffer de salida del Máster SPI |
| MISO | entrada | Buffer de entrada del Máster SPI |
| ini | entrada | Señalización del inicio de una transacción |
| dir | entrada | Señal que habilita la lectura o escritura de los datos |
| set\_dato | entrada | Señalización de la escritura del buffer de salida |
| dato\_wr | entrada | Dato que se introduce al periférico SPI |
| dato\_rd | entrada | Dato que se lee del periférico SPI |
| ena\_rd | entrada | Señal de habilitación para lectura |

El bloque está dividido en los siguientes circuitos:

* gen\_SCK: Realiza la generación de señales del reloj serie para la interfaz SPI
* reg\_miso: Registro de salida de desplazamiento de 8 bits
* reg\_mosi: Registro de entrada de 8 bits por el cual se introducen las direcciones y los datos de escritura.

Su funcionamiento se describe en las siguientes figuras:



*Fig. 3. Master SPI señales internas, tiempos y modos de operación.*

## Bloque controlador del acelerómetro

|  |  |  |
| --- | --- | --- |
| Señal | Dirección | Descripción |
| dato\_rd | salida | Dato que se lee del periférico SPI |
| ena\_rd | salida | Señal de habilitación para lectura |
| CS | salida | Señal del Chip select |
| SCK | salida | Señal de reloj del interfaz SPI |
| MOSI | salida | Buffer de salida del Máster SPI |
| MISO | entrada | Buffer de entrada del Máster SPI |

El circuito tiene el siguiente funcionamiento:

* Inicialmente espera 5 ms para luego modificar la configuración de los registros de control CRTL\_REG4 y CTRL\_REG1
* Genera las señales de control que permiten el funcionamiento correcto del Master SPI

## Bloque Calibrador

|  |  |  |
| --- | --- | --- |
| Señal | Dirección | Descripción |
| CS | salida | Señal del Chip select |
| SCK | salida | Señal de reloj del interfaz SPI |
| MOSI | salida | Buffer de salida del Máster SPI |
| MISO | entrada | Buffer de entrada del Máster SPI |
| X\_media | salida | Señal de salida que indica el valor medio del eje X |
| Y\_media | salida | Señal de salida que indica el valor medio del eje Y |
| Muestra\_bias\_rdy | salida | Señal de salida que indica si la muestra está lista para ser utilizada |

El bloque está dividido en los siguientes circuitos:

* Estimador: realiza la media de las acumulaciones de muestras del eje X y del eje Y
* Calc\_offset: Circuito que calculara el offset al que se encuentra el acelerómetro al principio de la ejecución.
* Auxiliar: Servirá para ejecutara dos funciones, que nos permitirán que el sistema funcione correctamente, por una parte ceil\_log, que calculara el menor numero y por otra parte cal\_LEDs que determinara los les que se encenderán

## Bloque Axis\_Screen\_X

|  |  |  |
| --- | --- | --- |
| Señal | Dirección | Descripción |
| X\_media | Entrada | Señal de entrada que indica el valor medio de X |
| Muestra\_bias\_rdy | Entrada | Señal que indica si la muestra está lista para ser utilizada |
| X\_disp | salida | Señal que muestra los displays encendidos |

Este bloque indica cuáles serán los displays de 7 segmentos que se van a iluminar en la tarjeta DECA

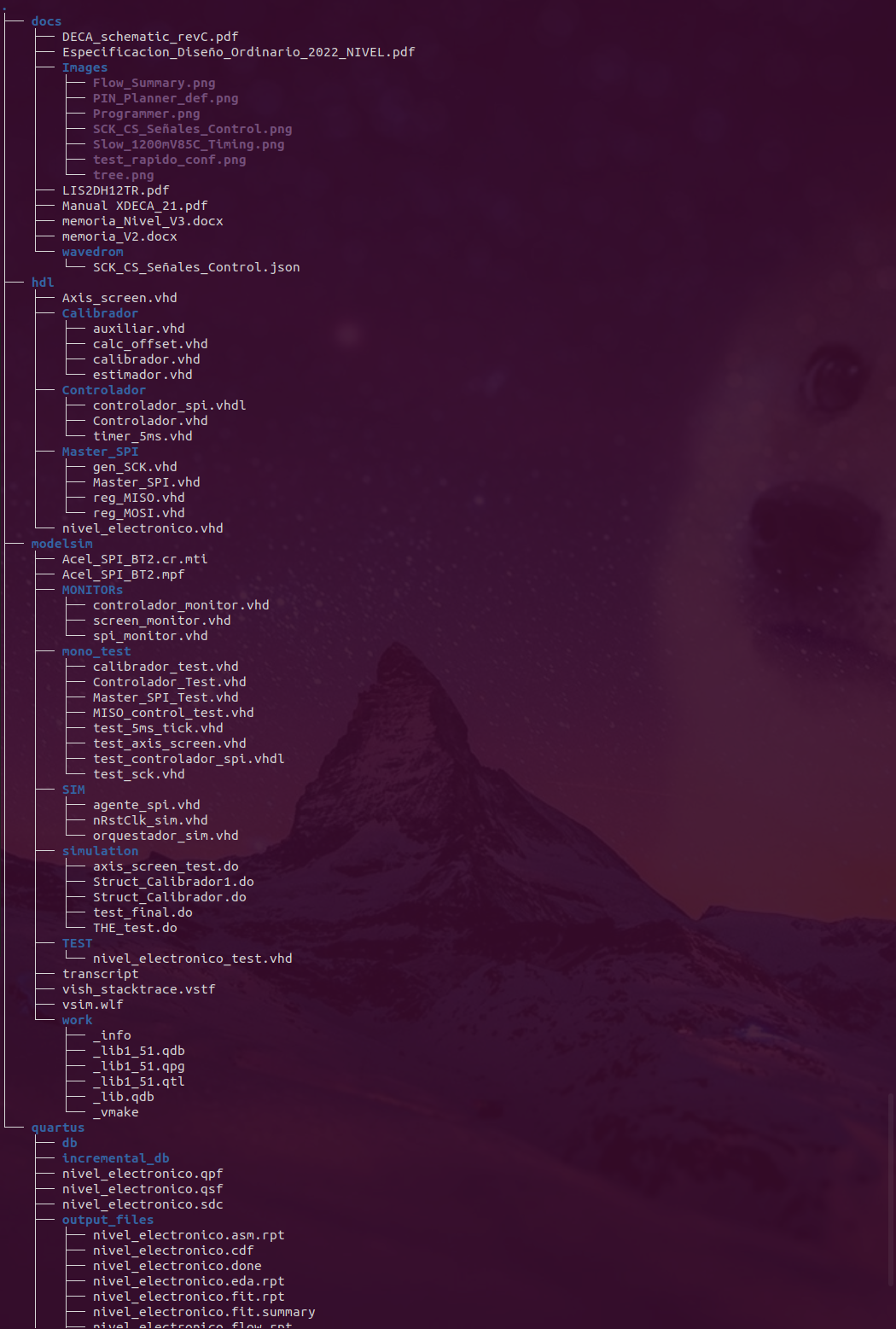
## Bloque Axis\_Screen\_Y

|  |  |  |
| --- | --- | --- |
| Señal | Dirección | Descripción |
| Y\_media | Entrada | Señal de entrada que indica el valor medio de Y |
| Muestra\_bias\_rdy | Entrada | Señal que indica si la muestra está lista para ser utilizada |
| Y\_disp | salida | Señal que muestra los leds encendidos |

Este bloque indica cuáles serán los leds que se van a iluminar en la tarjeta DECA

# Diseño detallado

El proyecto está almacenado en la carpeta BT2, la cual a su vez contiene las carpetas hdl, modelsim y quartus. La carpeta *hdl* contiene los ficheros RTL y estructural del diseño. La carpeta *modelsim* contiene el proyecto de simulación (Acel\_SPI\_BT2.mpf), y los ficheros donde se definen los diferentes testbenches. Finalmente, *quartus* contiene el proyecto para el diseño físico y los ficheros relacionados con éste.



*Fig. 4. Estructura de carpetas del proyecto nivel electronico*

# Pruebas de verificación funcional de NIVEL

El plan de pruebas de NIVEL consistirá en la comprobación hito por hito y circuito por circuito que el circuito funciona correctamente.

El proyecto contiene 2 tipos de pruebas, las primeras de verificacion de bloque, las cuales se pueden encontrar en la ruta “*modelsim*/*mono\_test*” y las segundas de verificacion de sistema complejo que se puede encontrar en *“/modelsim/TEST”.*

## Test del Máster SPI

Es un test específico para el bloque Máster SPI. Consiste en un conjunto de prueba que evalúa el funcionamiento correcto de dicho bloque.

|  |  |  |
| --- | --- | --- |
| Ubicación de los ficheros del test | En /NIVEL/modelsim/mono\_test | |
| Simulación escalada | No | |
| Fichero test | Master\_SPI\_Test.vhd | Testbench |
| Ficheros Adyacentes | Agente\_spi\_original.vhd | sim |
| MISO\_Control\_test.vhd | test |
| Descripción del test | Verifica el funcionamiento de las transacciones de escritura Master SPI (Verificacion visual)  Verifica el funcionamiento de las transacciones de lectura del Master SPI (Verificacion automatica) | |

## Test del controlador

Es un test específico para el bloque Controlador. Consiste en un conjunto de prueba que evalúa el funcionamiento correcto del controlador en conjunto al bloque Máster SPI.

|  |  |  |
| --- | --- | --- |
| Ubicación de los ficheros del test | En /NIVEL/modelsim/mono\_test | |
| Simulación escalada | No | |
| Ficheros | controlador\_test.vhd | Testbench |
| Ficheros Adyacentes | Agente\_spi\_original.vhd | sim |
| Descripción del test | Verifica que se lleva a cabo la configuración inicial de los registros y el funcionamiento de las transacciones de escritura y lectura del Master SPI. (Verificacion visual)  Verifica que los tiempos de accion solicitados en la especificacion, se cumplen (Verificacion automatica) | |

## Test del diseño completo

Es un test específico para el diseño final. Consiste en un conjunto de prueba final que verifica el funcionamiento completo de todo el diseño.

|  |  |  |  |
| --- | --- | --- | --- |
| Ubicación de los ficheros del test | En /NIVEL/modelsim/TEST | | |
| Simulación escalada | Si | | |
| Fichero | Nivel\_electronico\_test.vhd | test | |
| Ficheros Adyacentes | Sim | Agente\_spi.vhd | sim |
| Orquestador\_sim.vhd | sim |
| nRst\_Clk.vhd | sim |
| Monitor | Spi\_monitor.vhd | monitor |
| Controlador\_monitor. | monitor |
| Screen\_monitor.vhd | monitor |
| Descripción de la prueba | Verifica el funcionamiento correcto de todo el diseño mediante los monitores que revisar el Master SPI, el controlador y la representación de los leds y displays de 7 segmentos. (Verificacion Automatica) | | |

El test completo se ocupa de comprobar que siempre se cumplen las siguientes condiciones de cada seccion:

Master SPI

* Comprobar que mientras se produce una transmisión, CS está a nivel bajo.
* Comprobar que la transición de MOSI se produce cuando SCK, comienza la transición a nivel bajo.
* Comprobar que el número de pulsos de SCK se corresponden a la operación a realizar.
* Contrastar el dato transmitido por agente\_spi con el dato recibido en master\_spi.

Controlador

* Comprobar que se respeta el tiempo de inicialización dado por el datasheet.[[2]](#footnote-25856)
* Comprobar que cada lectura se realiza en el momento especificado en la especificación.
* Comprobar que la configuración de spi\_slave se realiza adecuadamente según la especificación.

Sistema completo

* Comprobar que según el ángulo dado por el agente\_spi, una vez pasado el tiempo de operación del sistema, las pantallas devuelven el grado de inclinación correcto.

Para ver la simulacion al completo, de una forma ordenada y concisa, se puede visualizar en el simulador con el fichero “*./modelsim/simulation/THE\_test.do”*

# Verificación del diseño final

Imagen que contiene circuito

Descripción generada automáticamenteComo comprobación final de que el sistema cumple todas las funcionalidades, se ha cargado el programa en la placa, comprobando que las especificaciones y que el modelo que se nos dio, al principio del Bloque II como ejemplo, sea igual que el que se ha cargado en la placa.

En la primera imagen se ha cargado el programa el diseño final.

En la segunda y tercera imagen se comprobó que el sistema mostraba correctamente, las medidas.

Imagen que contiene circuito, computadora

Descripción generada automáticamenteImagen que contiene sostener, tabla

Descripción generada automáticamente

# Diseño físico

En este apartado se documentan los detalles básicos relacionados con el diseño físico del circuito: la asignación de pines, las restricciones de la síntesis y los informes que proporciona Quartus Prime sobre los recursos de la FPGA utilizados y la frecuencia máxima de reloj obtenida.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Pin de la interfaz del diseño** | **Dirección** | **Pin FPGA** | **I/O bank** | **I/O standard** |
| clk | Input | M8 | 2 | 2.5-V |
| rst\_n | Input | H21 | 6 | 1.5-V Schmitt Trigger |
| MISO | Input | D5 | 8 | 1.2-V |
| MOSI | Output | C6 | 8 | 1.2-V |
| CS | Output | E9 | 8 | 1.2-V |
| SCK | Output | B5 | 8 | 1.2-V |
| X\_disp [0] | Output | Y5 | 3 | 3.3-V LVTTL |
| X\_disp [1] | Output | W6 | 3 | 3.3-V LVTTL |
| X\_disp [2] | Output | W8 | 3 | 3.3-V LVTTL |
| X\_disp [3] | Output | AB8 | 3 | 3.3-V LVTTL |
| X\_disp [4] | Output | R11 | 3 | 3.3-V LVTTL |
| X\_disp [5] | Output | AB6 | 3 | 3.3-V LVTTL |
| X\_disp [6] | Output | AA6 | 3 | 3.3-V LVTTL |
| X\_disp [7] | Output | V10 | 3 | 3.3-V LVTTL |
| Y\_disp [0] | Output | C7 | 8 | 1.2-V |
| Y\_disp [1] | Output | C8 | 8 | 1.2-V |
| Y\_disp [2] | Output | A6 | 8 | 1.2-V |
| Y\_disp [3] | Output | B7 | 8 | 1.2-V |
| Y\_disp [4] | Output | C4 | 8 | 1.2-V |
| Y\_disp [5] | Output | A5 | 8 | 1.2-V |
| Y\_disp [6] | Output | B4 | 8 | 1.2-V |
| Y\_disp [7] | Output | C5 | 8 | 1.2-V |
| mux\_disp[0] | Output | Y5 | 3 | 3.3-V LVTTL |
| mux\_disp[1] | Output | W6 | 3 | 3.3-V LVTTL |
| mux\_disp[2] | Output | W8 | 3 | 3.3-V LVTTL |
| mux\_disp[3] | Output | AB8 | 3 | 3.3-V LVTTL |
| mux\_disp[4] | Output | R11 | 3 | 3.3-V LVTTL |
| mux\_disp[5] | Output | AB6 | 3 | 3.3-V LVTTL |

## Asignación de pines

En la siguiente tabla se detalla la asignación de los pines de la interfaz de NIVEL a los pines de la FPGA especificando para cada caso el tipo de pin, el número de pin de la FPGA que se corresponde con el pin del diseño, el banco al que corresponde y el estándar de entrada/salida que utiliza.

## Restricciones de la síntesis

Se ha utilizado el fichero “*./quartus/nivel\_electronico.sdc”*

Se establece como reloj de sistema 50Mhz (20 ns), los puertos de entrada nRst y MISO, y como puertos de salida SCK, MOSI, CS, X\_disp[7:0], Y\_disp[7:0], Y\_disp\_sel[7:0].

## Recursos utilizados

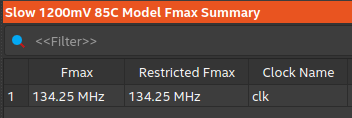
A continuación se muestra la pantalla que genera el compilador (opción *Flow Summary* del *Compilation Report*) :

|  |  |
| --- | --- |
| **Quartus Prime** | **Versión 16.0.0** |
| **Arquitectura** | **MAX 10** |
| **Dispositivo** | **10M50DAF484C6GES** |
| **Número total de elementos lógicos** | **404 / 49,760 ( <1% )** |
| **Número total de registros usados** | **191** |
| **Número total de pines usados** | **30 / 360 ( 8 % )** |
| **Número total de pines virtuales** | **0** |
| **Número total de bits en la memoria** | **660 / 1,677,312 ( < 1 % )** |
| **Número total de PLLs** | **0 / 4 ( 25 % )** |
| **Bloques UFM** | **0 / 1 ( 0 % )** |
| **Bloques ADC** | **0 / 2 ( 0 % )** |

## Frecuencia máxima de reloj

A continuación se muestra la pantalla que genera el compilador (*Fmax Summary*) :

134.25 MHz



# Bibliografía

[1] Especificación del diseño : nivel electrónico (NIVEL) [moodle DD2]

[2] Tarjeta DECA-MAX10 (página web del fabricante). [online] https://www.arrow.com/en/products/deca/arrow-development-tools

[3] Tarjeta XDECA. Manual de usuario. [moodle DD2-documentacion técnica]

1. Todos los bloques tienen un reloj común, clk, y una entrada de reset asíncrono, rst\_n, activa a nivel bajo. Estas señales no se incluyen en las interfaces por simplicidad. [↑](#footnote-ref-2)
2. En la simulacion, este test se puede omitir al poner en “*nivel\_electronico\_test.vhd” “omit\_timing\_monitor”* a *true* [↑](#footnote-ref-25856)