

UNIVERSIDAD PRIVADA DE TACNA
FACULTAD DE INGENIERÍA
ESCUELA PROFESIONAL DE INGENIERÍA ELECTRÓNICA



TESIS

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA ELECTRÓNICO
DE CONTROL BASADO EN FPGA, QUE OPTIMICE LAS
CONDICIONES CLIMÁTICAS DE UN INVERNADERO PARA EL
CULTIVO DE TOMATE EN LA CIUDAD DE TACNA, 2017”

PARA OPTAR:

TÍTULO PROFESIONAL DE INGENIERO ELECTRÓNICO

PRESENTADO POR:

Bach. Cándido Molanes Miovich

TACNA – PERÚ

2019

UNIVERSIDAD PRIVADA DE TACNA
FACULTAD DE INGENIERÍA
ESCUELA PROFESIONAL DE INGENIERÍA
ELECTRÓNICA

TESIS:

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA ELECTRÓNICO DE CONTROL BASADO EN FPGA, QUE OPTIMICE LAS CONDICIONES CLIMÁTICAS DE UN INVERNADERO PARA EL CULTIVO DE TOMATE EN LA CIUDAD DE TACNA, 2017”

PRESENTADA POR EL BACHILLER: Cándido, Molanes Miovich

PARA OPTAR EL TÍTULO PROFESIONAL DE: INGENIERO EN ELECTRÓNICA

TESIS SUSTENTADA Y APROBADA EL 26 DE NOVIEMBRE DEL 2019; POR EL JURADO REVISOR CONFORMADO POR:

PRESIDENTE



Ing. Heraclio Henry Gómez Del Carpio

SECRETARIO



Ing. Carlos Armando Rodríguez Silva

VOCAL



Ing. María Elena Vildoza Zambrano

ASESOR DE TESIS



Ing. Tito Leoncio Cordova Miranda

TACNA – PERÚ

2019

DECLARACIÓN DE ORIGINALIDAD

TÍTULO DEL TRABAJO QUE SE PRESENTA:

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA ELECTRÓNICO DE CONTROL BASADO EN FPGA, QUE OPTIMICE LAS CONDICIONES CLIMÁTICAS DE UN INVERNADERO PARA EL CULTIVO DE TOMATE EN LA CIUDAD DE TACNA, 2017”

Por medio de la presente certifico que soy autor(a) del trabajo que estoy presentando para posible publicación en la revista Educación de la Universidad Privada de Tacna. Me hago responsable de su contenido, el mismo que es resultado inédito de mi producción intelectual y que no lo he presentado a otra revista, libro o publicación alguna para su evaluación y posterior divulgación.

La información y referencias a publicaciones anteriores se encuentran identificadas correctamente con los créditos correspondientes incluidos en las citas textuales y en las referencias bibliográficas.

Debido a ello, declaro que los materiales empleados en el manuscrito que presento se encuentran libres de derecho de autor, por lo que me responsabilizo de cualquier reclamo que hubiere relacionado con los derechos de propiedad intelectual. Por ese motivo, exoneró a la Universidad Privada de Tacna de responsabilidad alguna.

Si el artículo “Diseño e implementación de un sistema electrónico de control basado en fpga, que optimice las condiciones climáticas de un invernadero para el cultivo de tomate en la ciudad de Tacna, 2017” es evaluado favorablemente y aceptada su publicación, autorizo a que la Universidad Privada de Tacna lo publique, reproduzca, edite, distribuya, exhiba y comunique las veces que considere necesarias tanto en el país como en el extranjero, por medios impresos, electrónicos, CD-ROM, Internet y cualquier otro medio.

Si el artículo “Diseño e implementación de un sistema electrónico de control basado en fpga, que optimice las condiciones climáticas de un invernadero para el cultivo de tomate en la ciudad de Tacna, 2017” es evaluado favorablemente y aceptada su publicación, autorizo a que la Universidad Privada de Tacna lo publique, reproduzca, edite, distribuya, exhiba y comunique las veces que considere necesarias tanto en el país como en el extranjero, por medios impresos, electrónicos, CD-ROM, Internet y cualquier otro medio.

Si el artículo "Diseño e implementación de un sistema electrónico de control basado en fpga, que optimice las condiciones climáticas de un invernadero para el cultivo de tomate en la ciudad de Tacna, 2017" es evaluado favorablemente y aceptada su publicación, autorizo a que la Universidad Privada de Tacna lo publique, reproduzca, edite, distribuya, exhiba y comunique las veces que considere necesarias tanto en el país como en el extranjero, por medios impresos, electrónicos, CD-ROM, Internet y cualquier otro medio.

Tacna, 26 de Noviembre de 2019



Bach. Cándido Molanes Miovich

DEDICATORIA

Le dedico esta tesis a Jehová mi Dios porque sin él no estuviese aquí, realizando todo lo que voy logrando hasta este momento.

AGRADECIMIENTO

Agradezco a mis padres por su apoyo y a terceras personas que no menciono sus nombres, pero que me acompañaron en muchos de mis pasos.

Agradezco a mi universidad, y en especial a sus distintos docentes por su apoyo, comprensión y enseñanza impartida a mi persona.

Y claramente agradezco a mi asesor de tesis, Ing. Tito Cordova.

Muchas gracias a todos por ser parte de muchos de mis logros.

ÍNDICE GENERAL

RESUMEN	1
PALABRAS CLAVES	1
SUMMARY	2
KEY WORDS	2
INTRODUCCIÓN	3
CAPITULO I: PLANTEAMIENTO DEL PROBLEMA	4
1.1 Descripción del problema	4
1.2 Formulación del problema	5
1.2.1 Formulación de los problemas específicos	5
1.3 Justificación e importancia	5
1.4 Objetivos	5
1.4.1 Objetivo general	5
1.4.2 Objetivos específicos.....	5
1.5 Hipótesis general.....	6
1.5.1 Hipótesis específicas.....	6
CAPITULO II: MARCO TEÓRICO	7
2.1 Antecedentes del estudio	7
2.2 Bases teóricas.....	8
2.2.1 El sistema de crecimiento de cultivos en invernadero.....	8
2.2.2 Características generales en la automatización de invernaderos	9
2.2.3 Tipos comunes de invernaderos.....	10
2.2.4 Factores que afectan la producción de tomate	12
2.2.5 Enfoques de control climático en invernaderos.....	14
2.2.6 Investigación y evaluación de sensores.....	19
2.2.6.1 Tecnología de sensores de temperatura.....	20
2.2.6.2 Tecnología de detección de humedad relativa.....	23

2.2.6.3 Tecnología de detección Humedad del suelo	26
2.2.7 Consideraciones para el algoritmo de control	29
2.2.7.1 Control de Humedad del suelo	29
2.2.7.2 Monitoreo de temperatura	30
2.2.7.3 Monitoreo de humedad relativa	30
2.3 Definición de términos	31
2.3.1 Definiciones de control	31
2.3.1.1 Definición de variables de control desde el punto de vista del control clásico	31
2.3.1.2 Definición de variables de control desde el punto de vista del control en un invernadero.	32
2.3.2 Definiciones de agricultura	33
2.3.2.1 Conceptos previos de agronomía	33
2.3.3 Definiciones generales	35
2.3.3.1 Sensores..	35
2.3.3.2 Conceptos previos FPGA nexys 4 DDR	36
CAPITULO III: MARCO METODOLÓGICO	45
3.1 Tipo y diseño de la investigación	45
3.2 Operacionalización de variables	45
3.3 Técnicas e instrumentos para la recolección de datos.....	45
CAPITULO IV: DISEÑO DEL SISTEMA DE CONTROL.....	48
4.1 Características de la ubicación del invernadero.....	48
4.2 Requerimientos climáticos y de riego del invernadero	49
4.3 Descripción del sistema de monitorización y control.....	51
4.4 Descripción de los bloques del diseño propuesto	52
4.4.1 Planta invernadero	52
4.4.2 Sensores	53
4.4.2.1 Selección del sensor de temperatura y humedad relativa.....	53

4.4.2.2 Selección del sensor de humedad del suelo	54
4.4.3 Acondicionamiento de la señal del sensor VH400	55
4.4.4 ADC.....	57
4.4.5 Selección de la plataforma de control	58
4.4.5.1 Microprocesador – Nexys 4 DDR	58
4.4.6 PWM.....	60
4.4.7 Acondicionamiento de la señal del actuador.....	61
4.4.8 Actuador.....	61
4.4.9 Monitor de visualización	62
CAPITULO V: RESULTADOS.....	63
5.1 Sistema físico de control y riego	63
5.2 Resultados del primer escenario	64
5.3 Resultados del segundo escenario.....	64
5.4 Resultados de los recursos de la plataforma nexys 4 ddr	64
CONCLUSIONES	67
RECOMENDACIONES	67
REFERENCIAS BIBLIOGRÁFICAS	69

ÍNDICE DE FIGURAS

Figura 1. “Diagrama conceptual de la producción en invernadero.” Fuente: F. Rodríguez et al., 2015.	9
Figura 2. “Esquema de control en invernadero.” Fuente: A. Pawlowski et al., 2009	10
Figura 3. “Diferentes tipos de termocuplas.” Fuente: thermometricscorp.....	20
Figura 4. “Tipos de detectores de temperatura resistivos.” Fuente: Designworldonline ..	21
Figura 5. “Tipos de termistores.” Fuente: Ussensor.....	22
Figura 6. “Tipos de circuitos integrados de temperatura y humedad relativa.” Fuente: Autodaewoospark.	23
Figura 7. “Tipos de sensores de humedad capacitivos.” Fuente: Indiamart.	24
Figura 8. “Tipo de sensores de humedad resistivos.” Fuente: lthead.....	25
Figura 9. “Tipos de sensores de humedad de conductividad térmica. Fuente: Grandtech.	25
Figura 10. “Sensor con tecnología TDR.” Fuente: labmodules.soilweb.ca	27
Figura 11. “Sensor con tecnología FDR.” Fuente: Experimental-hydrology.net	28
Figura 12. “Sensor de vegetronix modelo VH400.” Fuente: Vegetronix.com	29
Figura 13. “Varios niveles de agua en el contenido del suelo.” Fuente: (Z. Salazar, 2013)	34
Figura 14. “Estructura conceptual de un dispositivo FPGA.” (P. Chu, 2017).....	37
Figura 15. “Distribución de los periféricos en la plataforma nexys 4 ddr. Fuente: Manual de referencia nexys 4 DDR.	40
Figura 16. “Temperatura en la ciudad de Tacna 2014 y 2015.” Fuente: Sehnami.	48
Figura 17. “Humedad relativa en la ciudad de Tacna 2014 y 2015.” Fuente: Sehnami. .	49
Figura 18. “Diagrama de bloques del sistema de control climático y riego propuesto.” Fuente: Elaboración propia.	51
Figura 19. “Prototipo disposición de sensores y actuadores propuesto.” Fuente: Elaboración propia.	52
Figura 20. “Curva de VWC versus voltaje de salida sensor VH400.” Fuente: Vegetronix.com	56
Figura 21. Circuito de acondicionamiento sensor – adc. Fuente: Elaboración propia. ...	57
Figura 22. “Diagrama de bloques conceptual del XADC.” Fuente: (PONG P. CHU, 2017).....	57
Figura 23. “Diagrama de nivel superior del sistema FPro.” Fuente: (PONG P. CHU, 2017).....	59

Figura 24. “Diagrama de bloques del circuito básico de un PWM.” Fuente: (PONG P. CHU, 2017).	60
Figura 25. “Circuito de acondicionamiento microprocesador – servomotor.” Fuente: Elaboración propia.	61
Figura 26. “Servo TowerPro Modelo MG996R.” Fuente: Electronicoscaldas.com	61
<i>Figura 27. “Monitor de visualización en consola.” Fuente: Elaboración propia.....</i>	<i>62</i>
Figura 28. “Sistema físico de control de clima y riego para un invernadero.” Fuente: Elaboración propia.	63

ÍNDICE DE TABLAS

Tabla 1. “Clasificación de los invernaderos.” Fuente: P. Ponce et al.(2015)	12
Tabla 2. “Factores óptimos para la producción de tomate.” Fuente: P. Ponce et al. (2015)	14
Tabla 3. “Diferentes enfoques en el control climático del invernadero.” (I) Fuente: F. Rodríguez et al. (2015).....	16
Tabla 4. “Diferentes enfoques en el control climático del invernadero.” (II) Fuente: (FRANCISCO RODRÍGUEZ et al., 2015)	19
Tabla 5. “Operacionalización de variables.” Fuente: Elaboración propia.	45
Tabla 6. “Comparación teórico-práctica del factor y rango óptimo para el cultivo de tomate.” Fuente: Elaboración propia	51
Tabla 7. “Comparación de sensores industriales de temperatura y humedad relativa.” Fuente: Elaboración propia.	54
Tabla 8. “Comparación de sensores industriales de humedad de suelo.” Fuente: Elaboración propia	55
Tabla 9. “Ecuaciones del voltaje de salida del sensor VH400.” Fuente: Vegetronix.com	56
Tabla 10. “Comparación de microchips comúnmente usados vs microprocesador de software”. Fuente: Elaboración propia.....	58
Tabla 11. “Monitoreo y control de valores para el primer escenario. Fuente: Elaboración propia.”	64
Tabla 12. Monitoreo y control de valores para el segundo escenario. Fuente: Elaboración propia.	65
Tabla 13. “Comparación de los recursos de la plataforma nexys 4 ddr”. Fuente: Elaboración propia	66

ÍNDICE DE ANEXOS

ANEXO A: Matriz de consistencia.....	69
ANEXO B: Código del programa del sistema de control c++ orientado a objetos.....	71
ANEXO C: Código del programa del sistema de control VDHL.....	73
ANEXO D: Hoja de datos transistorP2N2222A	79
ANEXO E: Hoja de datos sensor de temperatura y humedad relativa SHT85.....	85
ANEXO F: Catalogo sensor de humedad del suelo VH400.....	104
ANEXO G: Presupuesto aproximado de costos de estructura, malla, sistema de riego y tecnología para un invernadero en la ciudad de Tacna	107
ANEXO H: Manual de referencia de la plataforma de desarrollo nexys 4 ddr.....	121
ANEXO I: Descripción general de los sistemas empotrados	150

RESUMEN

El propósito de la tesis es diseñar e implementar un sistema de control que optimizará las condiciones climáticas de un invernadero. El sistema está basado en una plataforma de desarrollo FPGA, concretamente la Nexys 4 ddr de digilent.

El enfoque principal de la tesis es la reconfigurabilidad y flexibilidad del hardware a futuro. Para ello se usa un sistema de control que incorpora un microprocesador microblaze de 32 bits el cual recolecta los datos medidos por sensores, y en base a un algoritmo de control desarrollado, mantiene de forma automática los parámetros determinados por los expertos en agronomía. Los parámetros a controlar son: temperatura, humedad relativa y humedad del suelo, deseables en un invernadero de tomate en la ciudad de Tacna. La temperatura y humedad relativa se mantiene en un rango de (24 – 26) °C y (70 y 80) % respectivamente y la humedad del suelo debe llegar hasta la mitad de la raíz de la planta.

Para medir la temperatura y humedad relativa se selecciona el sensor SHT85 con un rango de medición (-40 a 105) °C y (0 a 100) % respectivamente. La medición de la humedad del suelo se realiza mediante el sensor VH400 con dimensiones de (9.3x3.9x0.7) cm que cuenta con una regleta milimetrada. Además, se muestra las variables medidas en una consola de visualización en el monitor de una computadora aprovechando los datos enviados y recepcionados por el puerto uart de la plataforma FPGA.

PALABRAS CLAVES: Invernadero, Microprocesador, Sistema de control y automatización, sensor, hardware, software, reconfigurabilidad, flexibilidad.

SUMMARY

The purpose of the thesis is to design and implement a control system that will optimize the climatic conditions of a greenhouse. The system is based on an FPGA development platform, specifically the Nexys 4 ddr of digilent.

The main focus of the thesis is the reconfigurability and flexibility of the future hardware. For this, a control system incorporating a 32-bit microblaze microprocessor is used, which collects the data measured by sensors, and based on a developed control algorithm, automatically maintains the parameters determined by the agronomy expert. The parameters to control are: temperature, relative humidity and soil moisture, desirable in a tomato greenhouse in the city of Tacna. The temperature and relative humidity are maintained in a range of (24 - 26) ° C and (70 and 80)% respectively and the soil moisture must reach half the root of the plant.

To measure the temperature and relative humidity, the SHT85 sensor is selected with a measuring range (-40 to 105) ° C and (0 to 100)% respectively. The soil moisture measurement is made by the VH400 sensor with dimensions of (9.3x3.9x0.7) cm that has a millimeter strip. In addition, the variables measured in a display console are displayed on a computer monitor taking advantage of the data sent and received by the UART port of the FPGA card.

KEY WORDS: Greenhouse, Microprocessor, Control and automation system, sensor, hardware, software, reconfigurability, flexibility.

INTRODUCCIÓN

Hoy en día, el cultivo de tomate en la ciudad de Tacna se hace comúnmente en el suelo como medio de cultivo, con sistemas de riego precarios, y un clima desfavorable, así atravesando por una baja productividad. Una alternativa eficaz es utilizar un invernadero con un clima y riego controlado. Otro elemento que normalmente se usa es un vivero para que las plantas crezcan hasta un determinado tamaño para luego puedan ser trasplantadas y su posterior cultivo en suelo. Pero poco propietarios de estas tierras usan invernaderos.

Otro factor que no se tiene en cuenta, es la falta de capacitación de los agricultores en el cultivo de tomate, ya que no se cuentan con datos aplicados y de respaldo práctico con orientación a la realidad de Tacna. Los agricultores normalmente fertiirrigan sus plantas tres veces a la semana sin medir el consumo o los fertilizantes que aplican.

La ciudad de Tacna tiene una zona de agricultura que cada día va a aumentando cada vez más, con pocos propietarios de invernaderos, que presentan problemas de baja producción en meses de verano y no cuentan actualmente con la tecnología necesaria, están limitados a programadores de riego de alto costo y poca efectividad.

La presente tesis describe el diseño e implementación de un sistema electrónico de control basado en FPGA con el propósito de optimizar las condiciones climáticas de un invernadero para el cultivo de tomate en la ciudad de Tacna.

La organización de esta tesis de grado es la siguiente:

El capítulo I “Planteamiento del problema”: Examina la problemática agrícola actual en la ciudad de Tacna. Muestra la justificación, objetivos e hipótesis que tiene la tesis.

El capítulo II “Marco teórico”: Señala las investigaciones pasadas y conclusiones a las que se llegó, luego se desglosa los fundamentos de la tesis y un glosario de definiciones.

El capítulo III “Marco metodológico”: Contiene el tipo de investigación de la tesis y un cuestionario a expertos en agricultura de la región de Tacna.

El capítulo IV “Diseño del sistema de monitoreo y control”: Presenta el desarrollo del diseño propuesto y los criterios de selección de elementos de control.

El capítulo V “Resultados”: Evidencia y analiza los resultados, de acuerdo a las pruebas realizadas en un circuito que permite aplicar una lógica de control de tres variables.

CAPITULO I: PLANTEAMIENTO DEL PROBLEMA

1.1 Descripción del problema

Retrocedamos 20 años en el tiempo, la zona de agricultura en Tacna ocupaba una fracción del valle del río Caplina. Actualmente en el año 2017, los terrenos de cultivo van expandiéndose cada vez más, se extienden hasta llegar a terrenos cercanos al litoral. Los mayores problemas que se presentan son la escasez de agua por los cambios climáticos y el uso deficiente del recurso hídrico, por el uso de métodos precarios (riego por surcos, inundación). Un agricultor está acostumbrado a regar y fertilizar manualmente 3 veces a la semana sus plantas, lo que termina en un desperdicio de agua y fertilizantes.

Pocos propietarios de estas tierras usan invernaderos, los cuales utilizan dos métodos distintos: como viveros, para cultivar todo tipo de plantas hasta que alcanzan el estado adecuado para su trasplante, distribución y venta (etapa de semilla a plantín), con un uso de todo el año, y como invernadero para cultivar algunos tipos de plantas hasta alcanzar una producción determinada (etapa de semilla a fruto), con un uso de 9 meses, ya que los 3 meses de verano se prefiere cultivar en el exterior por el costo elevado de los métodos tradicionales para bajar la temperatura. Cabe resaltar que es vital monitorear la temperatura y humedad relativa por el problema de garuas esporádicas durante el año.

Ambos métodos de cultivo interno e externo tienen una tecnología de riego, que está muy limitada a un programador de riego de alto costo el cual es programado por un ingeniero agrónomo para regar en determinados horarios. Dependiendo de la evaporación, temperatura promedio, vientos y otros parámetros extraen mediante estadística cuantos minutos deben regar el cultivo, lo cual no tiene el rendimiento más óptimo.

Otro factor a tener en cuenta, es la falta de capacitación de los agricultores en el cultivo de tomates y otros productos agrícolas, en invernaderos, cuya producción se ve afectada especialmente en épocas de invierno.

De lo expresado anteriormente, el problema que se encontró en los invernaderos en la ciudad Tacna, fue una tecnología limitada, sin capacidad de mejorar los valores de las variables del invernadero como temperatura, humedad relativa, humedad de suelo, especialmente en verano. Surge la interrogante,

¿Con que tecnología, sensores y actuadores se puede lograr obtener valores óptimos de temperatura, humedad relativa y humedad de suelo en un invernadero en la ciudad de Tacna?

1.2 Formulación del problema

De la descripción del problema, se formula las siguientes preguntas:

¿En qué medida el diseño e implementación de un sistema electrónico de control basado en fpga, optimizara las condiciones climáticas de un invernadero en la ciudad de Tacna, en el año 2017?

1.2.1 Formulación de los problemas específicos

¿En qué medida el de control de humedad del suelo optimizará las condiciones climáticas en un invernadero de la ciudad Tacna?

¿En qué medida el control de humedad del suelo influirá en la temperatura y humedad relativa de un invernadero en la ciudad de Tacna?

1.3 Justificación e importancia

Dado que, en el año 2017, la ciudad de Tacna es una zona de agricultura creciente que cuenta con pocos propietarios de invernaderos con problemas como baja producción del tomate debido al cambio climático, escasez de agua, y tecnología de riego deficiente, limitada a un programador de riego de alto costo el cual es programado por un ingeniero agrónomo no siendo lo más óptimo desde el punto de vista del diseño electrónico, es que esta tesis tiene relevancia en diseñar e implementar un sistema electrónico de control para optimizar las condiciones climáticas de un invernadero en la ciudad de Tacna.

1.4 Objetivos

1.4.1 Objetivo general

Diseñar e implementar un sistema electrónico de control basado en la tecnología fpga, que optimice las condiciones climáticas de un invernadero en la ciudad de Tacna.

1.4.2 Objetivos específicos

- Diseñar e implementar un sistema de control de humedad de suelo que optimice las condiciones climáticas de un invernadero en la ciudad de Tacna.
- Diseñar e implementar un sistema de monitoreo que muestre la temperatura y humedad relativa de un invernadero en la ciudad de Tacna.

1.5 Hipótesis general

El diseño e implementación de un sistema electrónico de control si optimizará las condiciones climáticas de un invernadero en la ciudad de Tacna

1.5.1 Hipótesis específicas

El diseño de un sistema de control de humedad del suelo si optimizará las condiciones climáticas de un invernadero.

El diseño de un sistema de monitoreo de temperatura y humedad relativa si tendrá incidencia en un invernadero.

CAPITULO II: MARCO TEÓRICO

2.1 Antecedentes del estudio

En el momento que nos proponemos diseñar la solución de un determinado problema, la consideración previa de la plataforma de desarrollo es de gran importancia, el campo de la agricultura bajo invernadero exige y pone a prueba a dichas plataformas que son necesarias considerarse al comienzo del diseño, logrando así previamente garantizar la robustez del sistema.

Cuando se comenzaron a dar los primeros pasos de uso de tecnología en la agricultura la primera plataforma que se utilizo fue la computadora personal (PC), pero debido al ambiente del invernadero que se caracterizó por ser un ambiente ruidoso con altos niveles de humedad y cambios constantes de temperatura, es que poco a poco dejo de ser la aplicación más apropiada, siendo reemplazada por otras tecnologías. (Zhong-fu sun et al., 2005)

Con el abandono de la plataforma de desarrollo PC, surgen dos nuevas tecnologías, Sistemas basados en microcontroladores o procesadores digitales de señales (DSP) que fueron desarrollados, con resultados prometedores. (J.P. Coelho et al., 2005)

(Ramírez, A., 2014) en la tesis titulada “Diseño de un sistema para control y monitoreo de un invernadero hidropónico” utilizo como plataforma de desarrollo una plataforma arduino uno basada en el microcontrolador ATmega328 creado por Atmel, se llegó a la conclusión de que al implementar un controlador on/off en este microcontrolador se logra controlar y monitorear la temperatura y humedad de un invernadero.

(Lara, K., 2015) en la tesis titulada “Diseño e implementación de un sistema de control microclimático para la preservación de orquídeas endémicas del Perú en invernadero” también utiliza una plataforma arduino uno y se llegó a la conclusión que con este microcontrolador fue capaz de implementar 10 reglas de lógica difusa logrando replicar las tres zonas de vidas de las.

(Rodolfo, E. , 2015) En la tesis titulada “Diseño e implementación de un sistema de control de humedad relativa y absoluta en un invernadero” cambio el uso del microcontrolador ATmega328 por un microcontrolador MC9S08QE32, el cual es producido por Freescale Semiconductor, se llegó a la conclusión que al

implementar un sistema difuso manadami fue capaz no solo de controlar la temperatura y humedad sino de llevar las variables de temperatura y humedad a unos rangos óptimos para el cultivo de tomate bajo invernadero.

Todos tuvieron propósito de controlar y monitorear las variables dentro de un invernadero, el primero para el cultivo hidropónico en México, el segundo para la preservación de orquídeas en el Perú, y el ultimo para cultivos en general en el país de Colombia. Lograron controlar y monitorear los parámetros dentro de lo presentado en sus objetivos, pero no han sido cubierto muchos aspectos por la amplitud y complejidad del tema.

(A. Pawlowski et al., 2016) explica que el desarrollo de sistemas empotrados para aplicaciones particulares ha demostrado ser una solución madura y eficaz que puede diseñarse para resolver diversos problemas complejos en aplicaciones industriales.

Este antecedente ha migrado al campo de los invernaderos. En los últimos años los FPGAs, se han usado con éxito y se ha comprobado que es una solución de alto rendimiento, flexibilidad y bajo costo para aplicaciones en invernaderos (R. Castañeda et al., 2006)

2.2 Bases teóricas

2.2.1 El sistema de crecimiento de cultivos en invernadero

El diagrama de bloques conceptual de la producción en un invernadero se muestra en la figura 1.

Un invernadero tiene diferentes elementos y variables que se interrelacionan e interactúan entre sí dinámicamente, haciendo que el sistema sea complejo por lo que es necesario separarlas e identificarlas. En la figura 1 se divide por colores el sistema invernadero en subsistemas, procesos, variables y nos muestra como es la relación con el cultivo.

Las perturbaciones (clima exterior, plagas y enfermedades) son variables difíciles de predecir, modelar, no pueden ser manipuladas y afectan indirectamente el crecimiento normal de los cultivos, provocando que las variables controladas se desvíen de los valores deseados por el jefe de cultivos en el invernadero. Pero se pueden contabilizar monitoreándolas, para medir el impacto que tienen sobre el sistema.

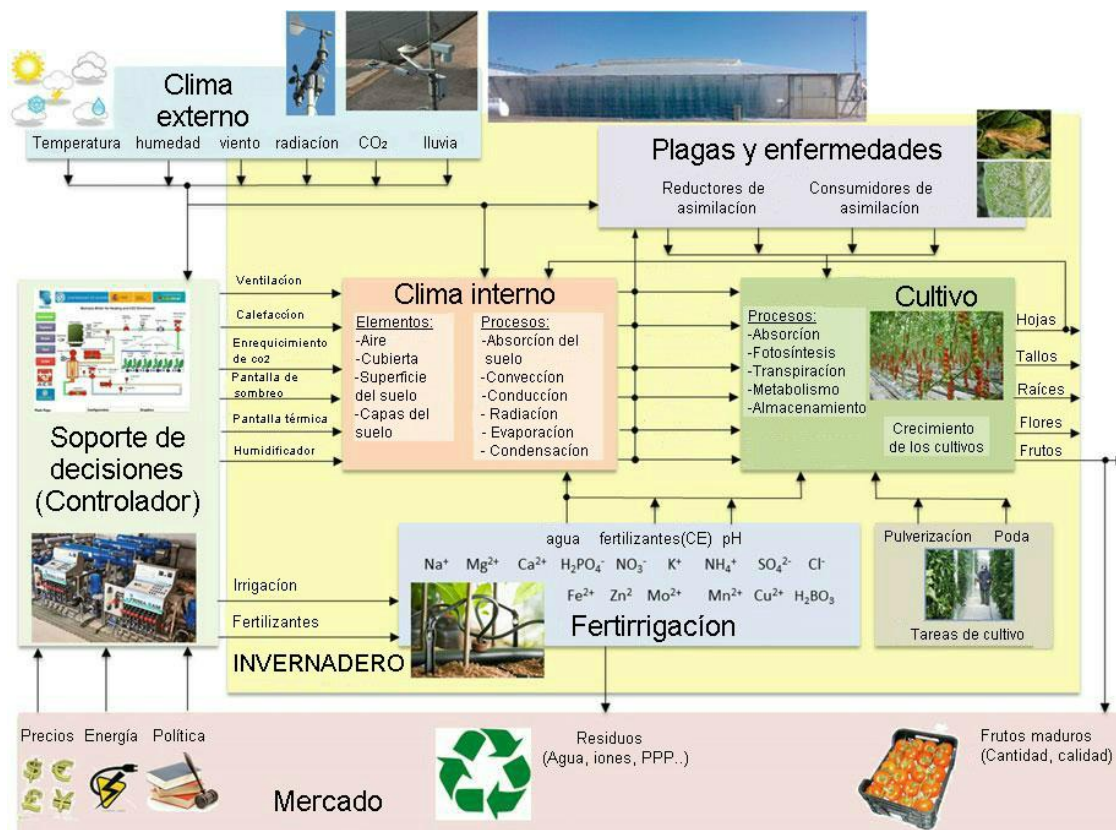


Figura 1. "Diagrama conceptual de la producción en invernadero." Fuente: F. Rodríguez et al., 2015.

Las entradas (Clima interno, Fertirrigación, Tareas de cultivo) al bloque verde cultivo son variables que se pueden pronosticar, manipular y controlar, afectan directamente al crecimiento normal de los cultivos.

Las salidas existen de dos tipos: Del cultivo en producción (Frutos, hojas, flores, tallos, o raíces) y de fertirrigación (agua, iones, PPP) son las variables a controlar.

Las regulaciones de mercado o ambientales son elementos que contribuyen al soporte de decisiones (controlador). (F. Rodríguez et al., 2015)

2.2.2 Características generales en la automatización de invernaderos

Para simplificar el problema de control se debe tener en cuenta que el desarrollo del cultivo está sujeto principalmente a las variables climáticas circundantes, cantidad de agua y fertilizantes que se mezclan y suministran en el riego. Por lo cual es de importancia resaltar que el clima y la fertirrigación son dos subsistemas independientes cada uno con diferentes problemas de control. (A. Pawlowski et al., 2009)

Cuando se analiza estos subsistemas desde el punto de vista del control automático, se deben distinguir las variables que afectan el proceso de crecimiento del cultivo en cada problema de control, estas se muestran en la figura 2 y se pueden agrupar en 3 conjuntos: variables controladas, variables manipuladas y perturbaciones. (A. Pawlowski et al., 2016)

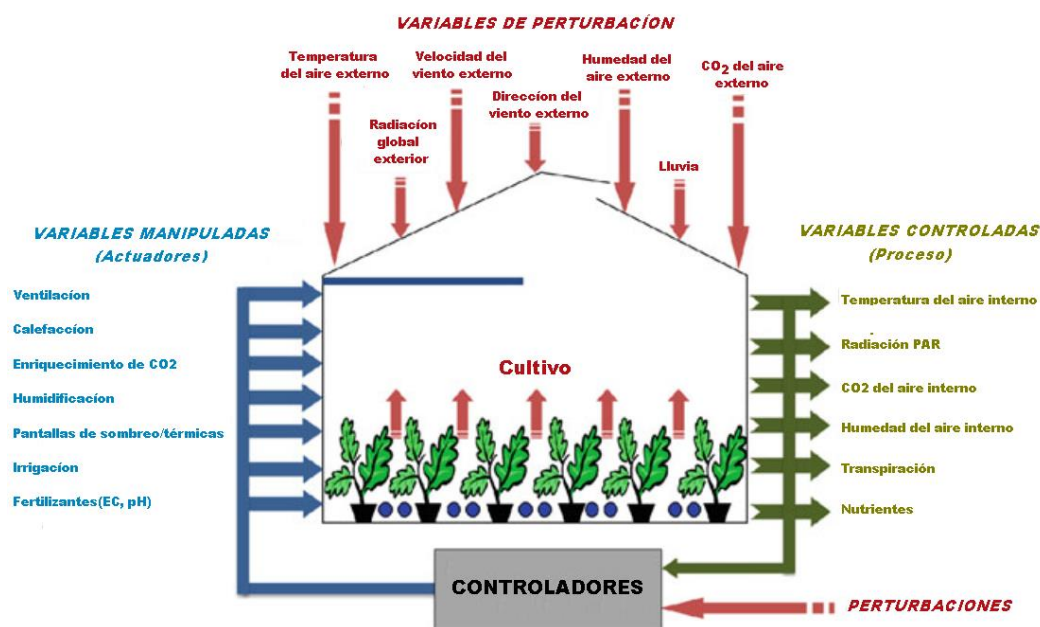


Figura 2. "Esquema de control en invernadero." Fuente: A. Pawlowski et al., 2009

2.2.3 Tipos comunes de invernaderos

Los invernaderos podemos encontrarlos con diferentes perfiles externos y medidas, cada uno adecuado a un tipo de clima que existe en el mundo. Cada clima requiere diferentes perfiles para proporcionar las condiciones climáticas más favorables para el crecimiento de las plantas.

Existen algunas clasificaciones de invernadero que dependen de distintos criterios (por ejemplo, material para la construcción, tipo de material de cubierta, características del techo, etc.). (P. Ponce et al., 2015)

A pesar de ello, para encontrar los más importantes es mejor enumerarlos haciendo caso omiso de algunas características para la clasificación. Entre los tipos más comunes de invernaderos en el mundo se encuentran:

Tipo	Características	
Multipilla de Techo curvo o túnel	Se distingue por una cubierta formada por arcos curvos semicirculares, por una estructura totalmente metálica y por su gran facilidad de adaptación a la producción de cualquier tipo de cultivo y superficie, tanto ornamental como hortícola.	
Capilla o pico estándar	Es un invernadero típico, y es una de las estructuras más antiguas, utilizadas en el cultivo forzado. Se encuentran de cubierta plana a una o dos aguas que utiliza vidrio o plástico .	
Diente de sierra	Este es una variación de los invernaderos capilla, donde ya no existe una cubierta en forma de pico sino que ahora es reemplazada por una forma que se asemeja a los dientes de una sierra.	

Parral o almeriense Su origen es en la provincia de Almería constituido por postes y alambres llamados “parral”, que vienen a ser una modificación de estructura que se utiliza para el cultivo de uvas de mesa, se utiliza en climas poco lluviosos

Venlo o Holandés Invernaderos de vidrio que se utilizan generalmente en el norte de Europa donde el clima es extremo con bajas temperatura.



Tabla 1. “Clasificación de los invernaderos.” Fuente: P. Ponce et al.(2015)

2.2.4 Factores que afectan la producción de tomate

Los principales factores que afectan la producción de tomate en un invernadero son:

- Humedad del suelo
- Temperatura
- Humedad relativa
- Intensidad y longitud luminosa
- Dióxido de carbono (CO₂)
- Control de enfermedades y insectos
- Manejo nutricional durante la vida de la planta de tomate
- Características vegetales varietales
- Manejo para producir frutos de calidad con rendimientos más altos.

Los rangos óptimos para los factores se detallan en la siguiente tabla.

Rangos óptimos

Factor	Rango de crecimiento	Rango optimo
Temperatura del aire	Etapa de germinación hasta plántula	24 – 26 °C
Día soleado	Plántula hasta terminación	25 - 27 °C
Día nublado	Plántula hasta terminación	22 - 24 °C
Temperatura del aire nocturno	Plántula hasta terminación	18 - 20 °C
Humedad relativa	Germinación hasta el crecimiento temprano	75 - 88 %
	Etapa de plántula	70 - 80 %
	vegetativa hasta terminación	60 – 80 %
pH de la solución de nutrientes	Germinación hasta el crecimiento temprano	5.5 – 6.5

Conductividad eléctrica de la solución de nutrientes	Germinación hasta el crecimiento temprano	1.8 – 2.0 (Ds M^{-1})
Día soleado	Plántula hasta terminación	1.5 - 2.0 (Ds M^{-1})
Día nublado	Plántula hasta terminación	2.5 – 4.0 (Ds M^{-1})

Tabla 2. “Factores óptimos para la producción de tomate.” Fuente: P. Ponce et al. (2015)

2.2.5 Enfoques de control climático en invernaderos

Los factores que afectan las diferentes etapas de crecimiento de una planta y que sensores tenemos disponibles para medirlos, expuestos anteriormente, nos da una visión de las variables que vamos a controlar en el invernadero. El desarrollo de este subíndice se centrará en las técnicas de control existentes, en relación al control de humedad de suelo, monitoreo de temperatura y humedad relativa. A continuación, se muestra las distintas estrategias que se emplearon para controlar las variables climáticas en el interior de diferentes invernaderos. La tabla I y II muestra una descripción detallada de cada uno de los controladores utilizados.

Variable controlada	Control de variable	Técnica de control	Referencia
Temperatura	Ventilación natural Calefacción	Control proporcional PI + Antiwindup	[49, 103, 461, 462, 478]
Temperatura	Calefacción	PI (en cascada)	[104]
Temperatura Humedad, CO2	Ventilación natural	Control PID (multiobjetivo)	[192]

	Pulverización, enriquecimiento CO2		
Temperatura	Calefacción	Control PIP	[482, 483]
Temperatura, humedad	Ventilación Calefacción, humidificadores, sombreado	Control de PDF	[6, 389, 390]
Temperatura	Calefacción	Control previo o feedforward	[361, 439, 440]
Temperatura, humedad, radiación PAR, CO2	Ventilación natural Calefacción, sombreado enriquecimiento CO2	Ganancia en programación on/off	[41, 215]
Temperatura	Ventilación natural	Control adaptativo (multitasa)	[13, 43, 362, 407]
Temperatura	Ventilación, Calefacción	Control GPC	[49, 342, 354]
Temperatura	Calefacción	Control GPC	[296]
Temperatura, Humedad, CO2	Calefacción, ventilación por aspersión, enriquecimiento de CO2	Control PID adaptativo no lineal basado en RBFN	[485]
Temperatura, humedad	Ventilación natural, humidificadores	MPC multivariable	[384]
Temperatura, CO2	Ventilación, calefacción, sombreado,	MPC multivariable	[50]

	enriquecimiento de CO2		
Temperatura	Calefacción	PPC descentralizado	[119]
Temperatura	Calefacción	MPC, retroalimentación linealización	[327]
Temperatura	Ventilación, calefacción	MPC (genética y optimización de enjambre de partículas)	[94, 95]
Temperatura, humedad	Calefacción, nebulización, ventilación natural	MPC no lineal	[48, 153, 154, 492]
Temperatura	Calefacción, enriquecimiento de CO2	Red neural no lineal MPC (sistemas expertos)	[471]
Temperatura CO2	Ventilación natural, calefacción, enriquecimiento de CO2	Horizonte de retroceso MPC Control óptimo (Pontryagin), feedforward	[132, 177, 428, 430, 432, 441, 442, 443]
Temperatura CO2	Calefacción, Ventilación natural	Control óptimo, control de la red neuronal feedforward	[7]
CO2	Ventilación, enriquecimiento de CO2	Control óptimo	[255]
Temperatura	Calefacción	Control óptimo (Pontryagin)	[447]
Temperatura	Calefacción, Ventilación natural	Control óptimo (programación lineal, Pontryagin)	[159, 200]

Tabla 3. "Diferentes enfoques en el control climático del invernadero." (I) Fuente: F. Rodríguez et al. (2015)

Variable controlada	Control de variable	Técnica de control	Referencias
Temperatura, CO2	Ventilación natural, enriquecimiento de CO2	Control óptimo (Pontryagin), feedforward	[202]
Temperatura, humedad, CO2	Calefacción, ventilación (fogging)	control óptimo, Backstepping	[34]
Temperatura y humedad	Calefacción, ventilación, empañamiento (fogging)	Linealización feedback-feedforward, filtro extendido de Kalman	[169, 395]
Temperatura y humedad	Ventilación, empañamiento (fogging)	Lineamiento feedforward de retroalimentación, desacoplamiento y feedforward	[42, 158]
Temperatura y humedad	Ventilación natural, calefacción	Linealización exacta	[158]
Temperatura, CO2	Ventilación natural, calefacción enriquecimiento de CO2	Control robusto de QFT	[253, 284]
Temperatura	Calefacción	Control H^∞ PI	[406]
Temperatura, humedad	Calefacción, cubierta, sombreado, humectación	Control robusto H_2	[38]
Temperatura	Ventilación	Control de redes neuronales	[20]
Temperatura, humedad, CO2	Calefacción, obturador, rociador, sombreado, Ventilación natural	Control de redes neuronales	[130, 260, 371, 481]

Humedad	Humidificadores	Control de redes neuronales	[399, 402, 403, 404]
Temperatura y humedad	Ventilación natural, sombreado de calefacción, nebulización	Control híbrido	[165, 259, 280, 336, 337, 362]
Temperatura	Calefacción	Control predictivo híbrido	[242, 280]
Temperatura	Ventilación natural, calefacción	Control basado en eventos	[127, 313, 314, 316, 317, 318, 319, 320, 321, 322]
Temperatura y humedad	Calefacción, Ventilación natural, sombreado, fogging	Control sin modelo	[240]
Temperatura y humedad	Calefacción, humidificadores	Control de lógica difusa	[203]
Temperatura y humedad	Ventilación natural, calefacción, humidificadores	Control de lógica difusa	[238]
Temperatura	Ventilación, calefacción	Control de lógica difusa	[139]
Temperatura, humedad de la iluminancia, CO2	Ventilación natural, calefacción, enriquecimiento de CO2, sombreado, iluminación artificial, fogging	Control de lógica difusa	[76, 77, 79, 89, 122, 123, 139, 148, 168, 195, 205, 209, 225, 238, 247, 258,

274, 288,
290,
291,292,
298, 334,
372, 396,
417, 457,
490]

Tabla 4. "Diferentes enfoques en el control climático del invernadero." (II) Fuente: (FRANCISCO RODRÍGUEZ et al., 2015)

2.2.6 Investigación y evaluación de sensores

Los sensores son otro asunto clave en un invernadero, ya que hay diversos tipos en el mercado. Sin embargo, es necesario identificar el tipo de sensor requerido para cada variable climática. Por lo cual debemos tener cuidado al seleccionar el tipo de sensor, teniendo en cuenta características, ventajas, desventajas, y lo que es más importante, el costo.(A. Garg et al., 2016)

Los sensores tienen dos funciones principales, la primera es monitorear las variables climáticas como: Temperatura, humedad relativa y del suelo. Vigilando con precisión y cuidado los valores medidos por los sensores respectivos. Luego estos datos irán a un algoritmo de control, que, de acuerdo a sus cálculos, tomará acción enviando una señal de corrección a un actuador, si cualquier valor monitoreado se ha desviado de su rango delimitado. La segunda función de los sensores es proveer retroalimentación a los actuadores. Luego el proceso de ajuste durara un cierto periodo hasta que la desviación vuelva al valor delimitado.(L. Feng et al., 2013)

La selección de cualquier sensor empieza con la revisión del estado del arte o antecedentes de aplicación y hoja técnica. Luego en caso falten datos para su adecuada selección, procedemos a una verificación, en un laboratorio hacemos mediciones de las características faltantes con instrumentos electrónicos comerciales o industriales confiables, si cumple con nuestros objetivos planteados ya podríamos ubicar los sensores en el invernadero y realizar una calibración manual, de lo contrario procederíamos a descartar el sensor y elegir otro.

2.2.6.1 Tecnología de sensores de temperatura

La tecnología de medición o detección de temperatura mediante sensores es una de las más utilizadas en el mundo. Nos permite medir la temperatura en diversas aplicaciones comerciales e industriales y nos proporciona protección contra las variaciones excesivas de temperatura. A continuación, se investigan y evalúan 4 familias diferentes de sensores de temperatura. Cada familia de sensores de temperatura tiene sus ventajas y desventajas. Dependiendo de las evaluaciones, un sensor puede ser más adecuado que otro. (VU MINH QUAN, 2011)

Termocupla



Figura 3. "Diferentes tipos de termocuplas." Fuente: *thermometriccorp.*

Ventajas:

- Amplio rango de temperatura (-233 ° C - 2316 ° C)
- Relativamente barato
- Muy preciso
- Deriva mínima a largo plazo
- Tiempo de respuesta rápido.

Desventajas:

- La relación entre la temperatura y la señal de la termocupla no es lineal
- Señal de salida baja (mV)
- Vulnerable a la corrosión
- La calibración de termopares puede ser tediosa y difícil.

Aplicabilidad:

Los termopares son apropiados para medir un amplio rango de temperatura. No es bueno utilizarlos para aplicaciones donde las diferencias de temperatura son pequeñas y que se deben medir con gran precisión. Para tales

aplicaciones, los termistores y los detectores de temperatura de resistencia son más adecuados. No es recomendable usar termocuplas cuando el sitio de medición y el instrumento están lejos (más de 10 a 20 metros de distancia). Las aplicaciones incluyen medición de temperatura para hornos, gases de escape de turbinas, motores diésel, producción de energía, y otros procesos industriales.

Detector de temperatura resistivo (RTD)

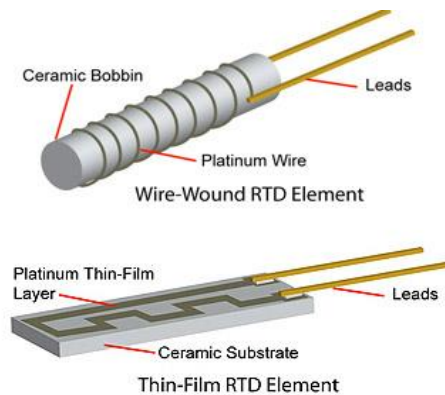


Figura 4. "Tipos de detectores de temperatura resistivos." Fuente: Designworldonline

Ventajas:

- Lineal sobre rango de operación de temperatura amplia
- Relativamente preciso
- Buena estabilidad y repetibilidad a alta temperatura (65-700 ° C)

Desventajas:

- Baja sensibilidad
- Mayor costo en comparación con los termopares
- Vulnerable a golpes y vibraciones

Aplicabilidad:

Los RTD son ampliamente utilizados en muchas aplicaciones industriales, tales como: aire acondicionado, procesamiento de alimentos, producción textil, procesamiento de plásticos, microelectrónica y medición de la temperatura de los gases de escape.

Termistores



Figura 5. "Tipos de termistores." Fuente: Ussensor

Ventajas:

- Altamente sensible
- Bajo costo
- Preciso sobre un pequeño rango de temperatura
- Buena estabilidad

Desventajas:

- Características de resistencia-temperatura no lineales
- Auto-calentamiento
- Rango operativo de temperatura limitada

Aplicabilidad:

Los termistores no son tan precisos ni estables como los RTD, pero son más fáciles de cablear, cuestan menos y casi todos los paneles de automatización los aceptan directamente. Se dividen en subtipos, el primero es un termistor de coeficiente de temperatura negativo (NTC), sus principales aplicaciones son: Termómetros de muy baja temperatura, termostatos digitales, monitoreo de baterías, y dispositivos de protección inmediata. El segundo, es un termistor de coeficiente de temperatura positivo (PTC) y su principal aplicación es como limitador de corriente.

Circuitos integrados (IC) sensores de temperatura



Figura 6. “Tipos de circuitos integrados de temperatura y humedad relativa.”

Fuente: Autodaewoospark.

Ventajas:

- Bajo costo
- Excelente linealidad
- Relativamente preciso
- Tienen un tamaño físico relativamente pequeño

Desventajas:

- Rango de temperatura limitado
- Autocalentamiento
- Frágil

Aplicabilidad:

Los sensores de temperatura IC a menudo se usan en aplicaciones donde la demanda de precisión es baja. Sus principales aplicaciones son calefacción, enfriamiento, detección de condiciones de falla, y otras aplicaciones de control y alarma.

2.2.6.2 Tecnología de detección de humedad relativa

Siguiendo el orden en esta sección se investiga y evalúa las 3 familias de sensores de humedad relativa más relevantes: capacitivo, resistivo y de conductividad térmica.

Sensores de humedad capacitivos (CHS)

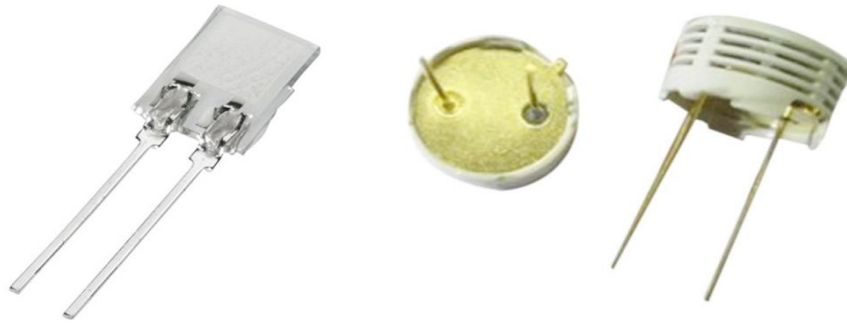


Figura 7. "Tipos de sensores de humedad capacitivos." Fuente: Indiamart.

Ventajas:

- Posibilidad de funcionar en entornos de altas temperaturas (hasta 200 ° C)
- Salida de voltaje lineal cercano
- Amplio rango de HR
- Alta tolerancia a la condensación
- Resistencia razonable a vapores químicos y contaminantes
- Deriva mínima a largo plazo
- Alta precisión
- Pequeño en tamaño y bajo costo

Desventajas:

- Distancia de detección limitada
- La integración del sensor puede ser tediosa y difícil.

Aplicabilidad:

Los sensores de humedad capacitivos son ampliamente utilizados en control/prevención de humedad de casas, sistemas de calefacción, ventilación, aire acondicionado y telemetría del clima. Estos también se usan en oficinas, automóviles humidificadores, museos, espacios industriales e invernaderos y también se usan en estaciones de meteorología.

Sensores de humedad resistivos (RHS)

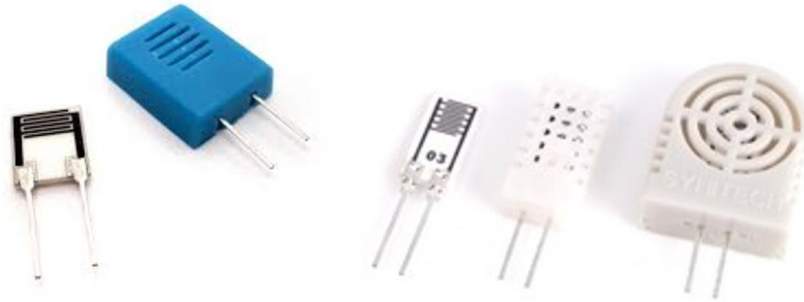


Figura 8. "Tipo de sensores de humedad resistivos." Fuente: Itead.

Ventajas:

- Tiempo de respuesta rápido
- Salida de voltaje lineal cercano
- Alta precisión
- Talla pequeña
- Bajo costo
- Amplio rango de HR

Desventajas:

- Temperatura de funcionamiento más baja en comparación con los CHS
- Sensible a los vapores químicos
- Baja tolerancia a los contaminantes
- Baja tolerancia a la condensación

Aplicabilidad:

Los RHS son adecuados para su uso en productos de control y visualización para aplicaciones industriales, comerciales y residenciales.

Sensores de humedad de conductividad térmica (TCHS)



Figura 9. "Tipos de sensores de humedad de conductividad térmica. Fuente: Grandtech.

Ventajas:

- Muy durable
- Posibilidad de operar en ambientes de altas temperaturas (hasta 600 ° C)
- Excelente inmunidad a muchos contaminantes químicos y físicos
- Alta precisión
- Alta tolerancia a la condensación

Desventajas:

- Responde a cualquier gas que tenga propiedades térmicas diferentes a las del nitrógeno seco
- Costoso

Aplicabilidad:

Los TCHS se usan comúnmente en electrodomésticos como secadoras de ropa, hornos microondas, y inyección de vapor. Las aplicaciones industriales incluyen secado de madera, maquinaria textil de secado, producción farmacéutica y deshidratación de alimentos. (MATHEW G. PELLETIER et al., 2016).

2.2.6.3 Tecnología de detección Humedad del suelo

La detección de humedad es importante en una serie de situaciones diferentes. Por ejemplo, la medición de la humedad del suelo es útil para minimizar la cantidad de agua de riego aplicada para el crecimiento de las plantas y para optimizar el crecimiento de las plantas. Debido a la importancia de conocer el contenido de humedad de los materiales, se han desarrollado diversas técnicas para medirlo. (M. Quan, 2011)

Esta sección sigue la ilación describiendo las tecnologías más relevantes de detección de humedad del suelo, se exhorta el leer los conceptos clave de agronomía en la sección definición de términos.

En esta sección se investiga y evalúa la tecnología de medición de humedad del suelo. Se analizan los sensores basados en el contenido de agua del suelo. Reflectometría de dominio de tiempo (TDR)



Figura 10. "Sensor con tecnología TDR." Fuente: labmodules.soilweb.ca

Ventajas:

- El TDR responde rápidamente a las diversas humedades del suelo.
- Mide la humedad con bastante precisión ($\pm 2\%$) en cualquier tipo de suelo.
- La humedad del suelo desde múltiples profundidades puede obtenerse de una sola sonda.
- Hay poca o ninguna perturbación en el sitio de prueba durante el proceso de prueba

Desventajas:

- Deben calibrarse cuidadosamente para medir con precisión la cantidad de tiempo que demora el pulso
- Este instrumento es más costoso que otros métodos de medición.
- Las aplicaciones de TDR son limitadas debido a los altos costos.
- TDR lee la humedad del suelo solo en las proximidades del sensor.

Aplicabilidad:

TDR se utiliza principalmente en los campos que tienen cultivos minerales y cultivos que se cultivan en suelos orgánicos. Han incluido el maíz dulce, el pimiento verde y los cultivos que se cultivan en suelos arenosos para los cuales se puede usar TDR.

Reflectometría de dominio de frecuencia (FDR)



Figura 11. "Sensor con tecnología FDR." Fuente: *Experimental-hydrology.net*

Ventajas:

- Es muy preciso ($\pm 1\%$) si se calibra adecuadamente.
- A diferencia del TDR, puede usarse con suelos con alta salinidad.
- Con FDR, se pueden realizar mediciones a varias profundidades en el mismo lugar.
- Es costoso en comparación con TDR

Desventajas:

- Requiere calibración específica del suelo.
- En FDR, se debe asegurar un buen contacto entre el suelo y el sensor para evitar la formación de espacios de aire.
- Puede detectar el contenido de humedad solo en las proximidades del sensor.

Aplicabilidad:

FDR se pueden utilizar en medios con suelo minerales, lana de roca, perlita y macetas. En cambio los TDR suelos pesados, especialmente con alta CE (conductividad eléctrica), la atenuación de la señal es grave y no tienen las capacidades de abastecimiento actuales ni un rango dinámico suficiente para proporcionar una medición de calidad. (M. Pelletier et al., 2016)

VH400 Sensor de humedad del suelo

El sensor VH400 mide la humedad del suelo y usa una tecnología FDR. Ayuda en el monitoreo preciso de bajo costo del contenido de agua del suelo. Tiene un tiempo de respuesta rápido, puede tomar lectura en menos de un segundo y es muy sensible a un mayor contenido de agua volumétrica. La sonda de humedad del suelo se inserta en el suelo, preferiblemente en posición horizontal en el nivel de la raíz. Este sensor es pequeño, robusto, resistente al agua y consume menos energía. También es insensible a la salinidad del agua, no se corroe con el tiempo y es sensible incluso a pequeños cambios en el contenido de agua.



Figura 12. "Sensor de vegetronix modelo VH400." Fuente: Vegetronix.com

Aplicabilidad:

Hidrología, Agronomía, Física del suelo, Sistemas de irrigación y riego, Fisiología de las plantas, Fenotipificación, Ecología de raíces, Monitoreo ambiental y Monitoreo de la lluvia.

2.2.7 Consideraciones para el algoritmo de control

2.2.7.1 Control de Humedad del suelo

La medición de humedad del suelo se basa en la medición de contenido volumétrico de agua (vwc) o potencial matricial del suelo. Para fijar un control sobre la humedad del suelo, debemos determinar el contenido de humedad, los sensores utilizados se apoyan en dos principios de la medición de la constante eléctrica de la matriz del suelo, el primero es la reflectometría del dominio del tiempo (TDR), que utiliza una señal electromagnética a lo largo de un tubo

metálico insertado en el suelo, y el segundo la reflectometría del dominio de la frecuencia (FDR), utiliza la capacitancia para medir la constante dieléctrica. (F. Rodríguez et al., 2015).

El exceso de suministro de agua a una planta de tomate produce un crecimiento pobre, menos flores y menor cantidad de fruto. Se necesita mucha agua, pero no un exceso. (PEDRO PONCE et al., 2015).

En sus primeras semanas el consumo es bajo y varía entre 0.249L hasta 0.303 L, cuando está en plena producción de frutos consumirá alrededor de 0.713 hasta 1L de agua por día.(M. Hanping et al., 2017)

(P. Ponce et al., 2015), dicen lo siguiente: Después de ver los resultados obtenidos, se puede inferir que la técnica de control proporcional tiene la ventaja sobre otras estrategias de control, de un sistema simple que exige una baja capacidad computacional. A pesar de esto puede manejar sistemas no lineales como el invernadero presentando un nivel de robustez, por lo que es el tipo más común de controlador visto en el mercado en todo el mundo.

2.2.7.2 Monitoreo de temperatura

Una planta de tomate no soporta el frío ni tampoco el calor, por lo que las temperaturas por debajo de 10°C y por encima de 35°C impedirán y reducirán el desarrollo de frutos.

La planta de tomate tiene un crecimiento normal y un buen desarrollo de frutos cuando su clima es cálido con una temperatura diurna de 21-29,5°C y una nocturna de 18,5-21°C. (P. Ponce et al., 2015).

2.2.7.3 Monitoreo de humedad relativa

La cantidad de vapor de agua en el aire del interior del invernadero es medida en porcentaje por la humedad relativa, esta no es una variable climática que influya directamente al crecimiento del cultivo, aunque su control es de peculiar interés. Pero para evitar la aparición y desarrollo de enfermedades fúngicas es necesario que la humedad relativa se mantenga entre 60 y 80%. (FRANCISCO RODRÍGUEZ et al., 2015).

Cuando decidimos que la humedad relativa está provocando el desarrollo de enfermedades y creando un hábitat favorable para los insectos en el follaje que no podemos controlar, es necesario implementar un control sobre la

humedad relativa que mantenga la planta lo más seco posible. La humedad relativa ideal es del 50%, siendo un rango tolerable del 40 al 70%.

Es bueno recordar que la temperatura y la humedad relativa deben monitorearse en cada invernadero. Además, se debe conocer que la temperatura está inversamente relacionada con la humedad relativa, es decir que, a altas temperaturas, el aire aumenta la capacidad de contener vapor de agua y por lo tanto disminuye la humedad relativa a bajas temperaturas, la humedad relativa aumenta (P. Ponce et al., 2015).

También es de consideración mencionar, que la calefacción y la ventilación simultaneas van a dar o resultar en un aumento del consumo de energía bastante alto por parte del invernadero. (J.C. BAKKER et al., 1995)

2.3 Definición de términos

2.3.1 Definiciones de control

2.3.1.1 Definición de variables de control desde el punto de vista del control clásico

Variable controlada:

Una variable controlada es la cantidad que se mide y controla. (KATSUHIKO OGATA, 2010)

Variable manipulada o señal de control:

Es la cantidad que el controlador utiliza para modificar el valor de la variable controlada. Normalmente, la variable controlada es la salida del sistema. (KATSUHIKO OGATA, 2010)

Perturbaciones:

Señal que tiende a afectar negativamente el valor de salida de un sistema. (KATSUHIKO OGATA, 2010)

Planta:

Una planta puede ser una parte de un equipo, tal vez un conjunto de los elementos de una máquina que funcionan juntos, y cuyo objetivo es efectuar una operación particular. En este libro se llamará planta a cualquier objeto físico que se va a controlar (como un dispositivo mecánico, un horno de calefacción, un reactor químico o una nave espacial). (KATSUHIKO OGATA, 2010)

2.3.1.2 Definición de variables de control desde el punto de vista del control en un invernadero.

Variables controladas:

Son variables que afectan directamente en el crecimiento del cultivo. Se dividen en dos, las variables climáticas como la temperatura interna, la radiación PAR, y la concentración de CO₂. La humedad relativa también debe controlarse porque afecta indirectamente la absorción de CO₂ ayudando al crecimiento de plagas. Y las variables de fertilización que son el suministro de agua, EC y pH. (FRANCISCO RODRÍGUEZ et al., 2015).

Variables de perturbación:

Son variables que afectan a las variables controladas, sin embargo, no pueden ser manipuladas. En este caso, estas variables son las climáticas externas (temperatura, humedad relativa, radiación solar, lluvia, velocidad y dirección del viento), la transpiración del cultivo (que se basa en la etapa del cultivo) y un conjunto de diferentes variables que describen los elementos de invernadero (techo, superficie del suelo, etc.) (FRANCISCO RODRÍGUEZ et al., 2015).

Variables manipuladas:

Son variables utilizadas para compensar o aprovechar el efecto de las perturbaciones y que están asociadas con los actuadores del proceso. Los actuadores climáticos son dependientes de la latitud donde se ubica el invernadero, siendo los más comunes: ventilación natural, sistema de calefacción, sistemas de enriquecimiento de CO₂, sistema de nebulización, pantalla de sombreo o térmicas. En el caso del proceso de fertirrigación, estos sistemas son los utilizados para suministrar agua y fertilizantes. (F. Rodríguez et al., 2015).

Proceso:

Variable a controlar (por ejemplo, temperatura). (FRANCISCO RODRÍGUEZ et al., 2015).

Sistema de medida:

Para medir el valor actual de cada variable (por ejemplo, sensor de temperatura), las perturbaciones medibles y el estado del actuador. (F. Rodríguez et al., 2015).

Controlador:

El sistema que compara el valor de referencia deseado (valor de consigna) con el valor actual de la variable controlada, y luego toma la acción correspondiente basada en esta comparación (por ejemplo, ordenador y software de control para regular las variaciones de temperatura). (FRANCISCO RODRÍGUEZ et al., 2015).

Actuadores:

Dispositivos gobernados por el controlador para mantener la variable controlada dentro de los límites deseados (por ejemplo, ventilación natural, sistema de calefacción). (F. Rodríguez et al., 2015).

2.3.2 Definiciones de agricultura**Invernadero:**

Es un recinto que permite a los propietarios controlar las variables climáticas, nutricionales, bióticas y de manejo cultural que influyen en el crecimiento y desarrollo de los cultivos, de manera que se obtienen condiciones óptimas en las diferentes etapas del crecimiento de los cultivos, permitiendo cultivar fuera de temporada. (F. Rodríguez et al., 2015).

Es un espacio cerrado que crea un ambiente diferente al que se encuentra en el exterior debido al confinamiento del aire y la absorción de radiación solar de onda corta a través de una cubierta de plástico o vidrio. (M. Ghomari et al., 2005)

VWC:

Contenido volumétrico del agua es una medida numérica de la humedad del suelo. (J.C. Bakker et al., 1995)

2.3.2.1 Conceptos previos de agronomía

La tecnología de medición de humedad del suelo también nos sirve para determinar variables importantes, como capacidad de campo, punto de marchitez permanente y equilibrio mínimo. Estas variables dependen de la composición y textura del suelo, las cuales son diferentes en cada tipo de suelo. El procedimiento para determinarlos es poner un sensor en el suelo, luego inundar el campo homogéneamente, hasta saturar el sensor, finalmente determinar los datos importantes, en la gráfica. (ZULEMA VARAS SALAZAR, 2013).

A medida que el agua se infiltra en el suelo, los espacios porosos se llenan de agua y el agua comienza a filtrarse hacia abajo. A medida que este proceso continúa, el suelo alcanza la capacidad de campo, pero la filtración de agua continúa debido a la acción capilar y la gravedad. Cuando el agua del suelo excede la capacidad del campo, el exceso de agua se drena (punto de saturación). La marchitez permanente en la figura 13 indica el punto en el cual las plantas han absorbido toda el agua disponible y se marchitarán de tal forma que no puedan recuperarse.

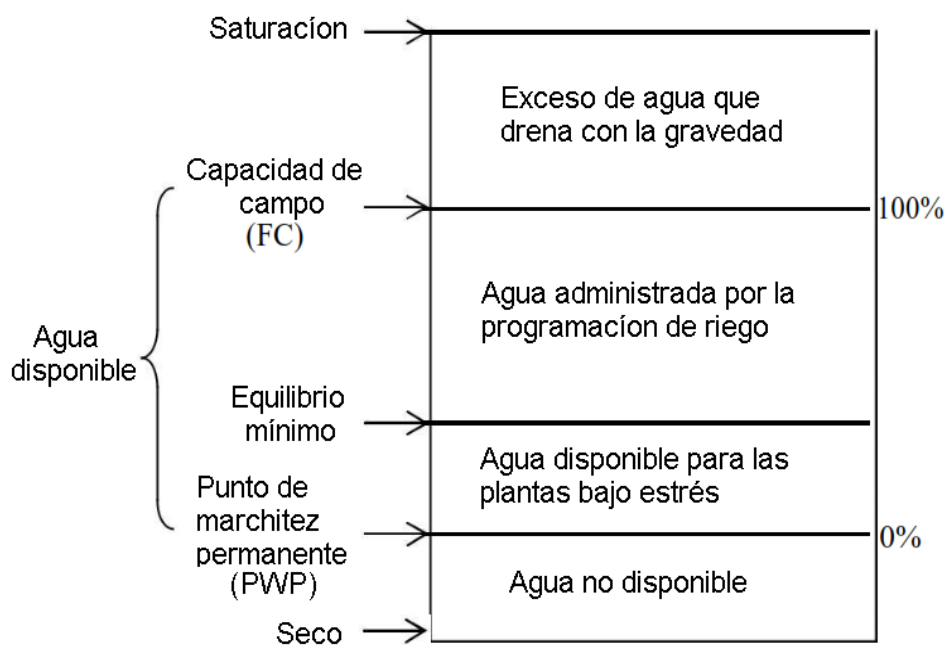


Figura 13. "Varios niveles de agua en el contenido del suelo." Fuente: (Z. Salazar, 2013)

La capacidad disponible de retención de agua en el suelo es diferente para diferentes tipos de suelos. Inicialmente, a medida que el agua del suelo se agota desde la capacidad de campo (100% del agua disponible) hacia el punto de marchitez permanente (0% del agua disponible), el crecimiento de la planta no se ve afectado hasta que el agotamiento alcanza el punto de equilibrio mínimo (también conocido como manejo permisible déficit). El agotamiento del agua del suelo por debajo de este equilibrio mínimo conduce a pérdidas de rendimiento. Por lo tanto, se debe tener cuidado de que el agua en la zona de la raíz del cultivo se mantenga entre el equilibrio mínimo y la capacidad de campo. (ANCHIT GARG et al., 2016).

Reflectometría de dominio de tiempo (TDR)

La reflectometría de dominio de tiempo, inyecta un pulso de energía de radiofrecuencia en una línea de transmisión en el suelo y su velocidad se mide al detectar el pulso reflejado desde el final de la línea. Esta velocidad depende de la constante dieléctrica. Mide el contenido de humedad midiendo el tiempo que demora en regresar el pulso reflejado. (ANCHIT GARG et al., 2016).

Reflectometría de dominio de frecuencia (FDR)

El sensor FDR consiste en un par de anillos de metal que se forman como un condensador y la muestra de suelo actúa como un dieléctrico. La capacitancia del sensor eléctrico es una medida directa del contenido volumétrico del suelo. Su principio es similar al sensor TDR. (A. Garg et al., 2016).

2.3.3 Definiciones generales

2.3.3.1 Sensores

Termocupla

Una termocupla o termopar es un sensor para medir la temperatura formado por unión de dos cables cada uno con un metal diferente, unidos en un extremo. Cuando la unión de los cables se calienta o enfría se genera un voltaje que es proporcional a la temperatura. (M. Quan, 2011).

Detector de temperatura resistivo

Un RTD es un detector de temperatura resistivo, es decir, un sensor de temperatura que correlaciona la resistencia del elemento RTD con la temperatura. La mayoría de los elementos RTD consisten en una longitud de alambre enrollado fino envuelto alrededor de un núcleo de cerámica o vidrio. El elemento suele ser bastante frágil, por lo que a menudo se coloca dentro de una sonda forrada para protegerlo. (VU MINH QUAN, 2011).

Termistores

Un termistor es también un sensor de temperatura resistivo. A diferencia de los RTD, que cambian la resistencia de un modo casi lineal, los termistores presentan un cambio de la resistencia claramente no lineal. Su funcionamiento consiste en la variación proporcional de la resistencia que presenta un semiconductor ante un pequeño cambio de temperatura. (M. Quan, 2011).

Sensores tipo circuito integrado

Los sensores de temperatura del circuito integrado son dispositivos semiconductores que se fabrican de forma similar a otros dispositivos semiconductores como los microcontroladores. En aplicaciones de bajo costo, la mayoría de los sensores indicados anteriormente son costosos o requieren circuitos o componentes adicionales para ser utilizados. Sin embargo, los sensores de temperatura IC son circuitos de detección basados en silicio completados, con salidas analógicas o digitales. (VU MINH QUAN, 2011).

Sensores de humedad capacitivos (CHS)

Los sensores de humedad capacitiva consisten en un material dieléctrico higroscopio colocado entre un par de electrodos. La mayoría de sensores capacitivos cuentan con una constante dieléctrica típica que varía de 2 a 15. Cuando no hay humedad presente en el sensor, tanto esta constante como la geometría del sensor determinan el valor de la capacitancia. (M. Quan, 2011).

Sensores de humedad resistivos (RHS)

Los sensores de humedad resistiva miden los cambios en la impedancia eléctrica de un medio higroscópico como: polímero conductor, sal o sustrato tratado. Los RHS consisten en electrodos de metal ya sea depositados en un sustrato mediante técnicas de fotorresistencia o electrodos enrollados en un cilindro de plástico o vidrio(VU MINH QUAN, 2011).

Sensores de humedad de conductividad térmica (TCHS)

Los sensores de humedad de conductividad térmica (TCHS) miden la humedad absoluta al cuantificar la diferencia entre la conductividad térmica del aire seco y la del aire que contiene vapor de agua. TCHS consiste en dos elementos de termistor de coeficiente de temperatura negativa (NTC) emparejados en un circuito de puente; uno está encapsulado herméticamente en nitrógeno seco y el otro está expuesto al medio ambiente. (M. Quan, 2011).

2.3.3.2 Conceptos previos FPGA nexys 4 DDR

Descripción general de un dispositivo FPGA

Un FPGA (Arreglo de compuertas programables en campo) es un dispositivo lógico que contiene una matriz bidimensional de celdas lógicas genéricas e interruptores programables. La estructura conceptual de un dispositivo FPGA se muestra en la Figura 14. Se puede configurar una celda

lógica (es decir, puede ser programada) para realizar una función simple, y un conmutador programable se puede personalizar para proporcionar interconexiones entre las celdas lógicas. Se puede implementar un diseño personalizado especificando la función de cada celda lógica y configurando selectivamente la conexión de cada interruptor programable.

Una vez que se complete el diseño y la síntesis, podemos usar un cable adaptador simple para descargar la celda lógica deseada y cambiar la configuración al dispositivo FPGA y obtener el circuito personalizado. Dado que este proceso se puede hacer "en el campo" en lugar de "en una instalación de fabricación", el dispositivo se conoce como campo programable. (PONG P. CHU, 2017)

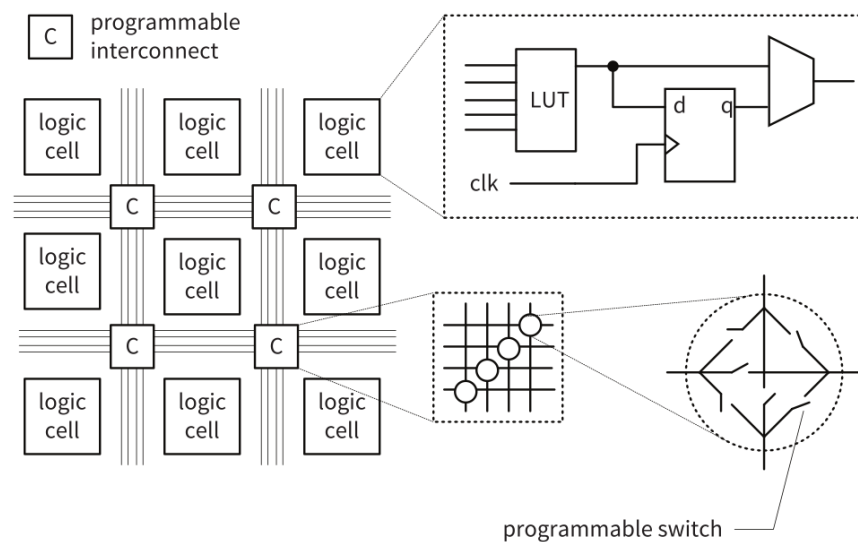


Figura 14. "Estructura conceptual de un dispositivo FPGA." (P. Chu, 2017).

Descripción general de los dispositivos Xilinx Artix-7

La plataforma de desarrollo nexys 4 ddr posee un dispositivo FPGA de la familia Xilinx Artix-7 y a continuación se proporciona una breve descripción de este dispositivo en esta sección. (P. Chu, 2017).

Celda lógica, slice y CLB

El elemento más básico de un dispositivo Artix-7 es la celda lógica (LC). Una celda lógica contiene una LUT, que se puede configurar como una LUT de 6 entradas o como dos LUT de 5 entradas y dos FF. Además, una celda lógica contiene un circuito portador, que se usa para implementar funciones aritméticas, y un circuito multiplexor, que se usa para implementar multiplexores anchos.

Algunos LUT también se pueden configurar como un pequeño módulo SRAM (memoria de acceso aleatorio estático) distribuido o un registro de desplazamiento. Para aumentar la flexibilidad y mejorar el rendimiento, se combinan ocho celdas lógicas junto con una estructura de enrutamiento interno especial.

En términos de Xilinx, cuatro celdas lógicas se agrupan para formar una división, y dos divisiones se agrupan para formar un bloque lógico configurable (CLB). (PONG P. CHU, 2017)

Macroelda

El dispositivo Artix-7 contiene varios tipos de macroceldas. La macroelda MMCM (administrador de reloj en modo mixto) es un núcleo que gestiona el reloj para producir una amplia gama de frecuencias a partir de una sola entrada de oscilador, reducir la desviación del reloj y ajustar el desplazamiento de fase de una señal de reloj. La macroelda BRAM (memoria de acceso aleatorio de bloques) es una SRAM síncrona de doble puerto de 36K bits que puede organizarse en varios tipos de configuraciones. La macroelda DSP (procesamiento de señal digital) se compone de un multiplicador binario de 25 por 18 y un acumulador de 48 bits y está diseñado para admitir algoritmos DSP intensivos en cómputo. Una macro celda IOB (bloque de entrada / salida) está asociada con un pin de E/S físico del dispositivo FPGA. Se puede configurar para admitir una amplia variedad de estándares de señalización de E/S y enlaces de datos en serie de alta velocidad.

El XADC (convertidor analógico a digital Xilinx) contiene dos convertidores de analógico a digital de 12 bits. Además de estos, el dispositivo también puede incluir bloques especiales para los transceptores gigabit ethernet y el bus PCI Express. (P. Chu, 2017)

Dispositivos en la familia Artix-7

Aunque los dispositivos FPGA Artix-7 tienen tipos similares de celdas lógicas y macroceldas, sus densidades difieren. La familia contiene una variedad de dispositivos de varias densidades. Los números de celdas lógicas, BRAM de 36K bits, segmentos DSP y MMCM de los dispositivos se resumen en la Tabla x La placa de desarrollo nexys 4 DDR utilizada en esta tesis contiene un dispositivo Artix XC7A100T. (PONG P. CHU, 2017).

Plataforma Nexys 4 ddr

La plataforma Nexys 4 DDR es una sistema de desarrollo de circuitos digitales completos y listos para usar, basada en la última gama de puertas programables (FPGA) Artix-7™ de Xilinx®. Con una gran FPGA de alta capacidad (Xilinx número de pieza XC7A100T-1CSG324C), generosas memorias externas, y una serie de puertos USB, Ethernet y otros, la Nexys4 DDR puede alojar diseños que van desde circuitos combinacionales introductorios hasta poderosos procesadores empotrados. Tiene varios periféricos integrados, incluyendo un acelerómetro, un sensor de temperatura, un micrófono digital MEMs (sistema micro electromecánico), un amplificador de altavoz y varios dispositivos de E / S, permiten utilizar el Nexys4 DDR para una amplia gama de diseños sin necesidad de otros componentes. La distribución de los periféricos en la plataforma de desarrollo se muestra en la figura 15. (P. Chu, 2017).

Los principales componentes y conectores son los siguientes:

1. Conector de alimentación para fuente de alimentación externa opcional
2. USB JTAG y puerto UART compartidos
3. Dispositivo FPGA Artix XC7A100T
4. Puerto Pmod (JD)
5. Puerto Pmod (JC)
6. Dieciséis LED discretos
7. Dieciséis interruptores deslizantes
8. Sensor de temperatura
9. Pantallas LEDs de siete segmentos de ocho dígitos
10. Puerto Pmod (JB)
11. Cinco interruptores tipo botón
12. Puerto Pmod (JA)
13. Botón de reinicio del soft-core
14. Puerto Pmod con entrada analógica (conectado al XADC)
15. Conector de audio
16. Puerto VGA
17. Conector Ethernet
18. Puerto de host USB (conectado al mouse / teclado USB)

19. Interruptor de encendido

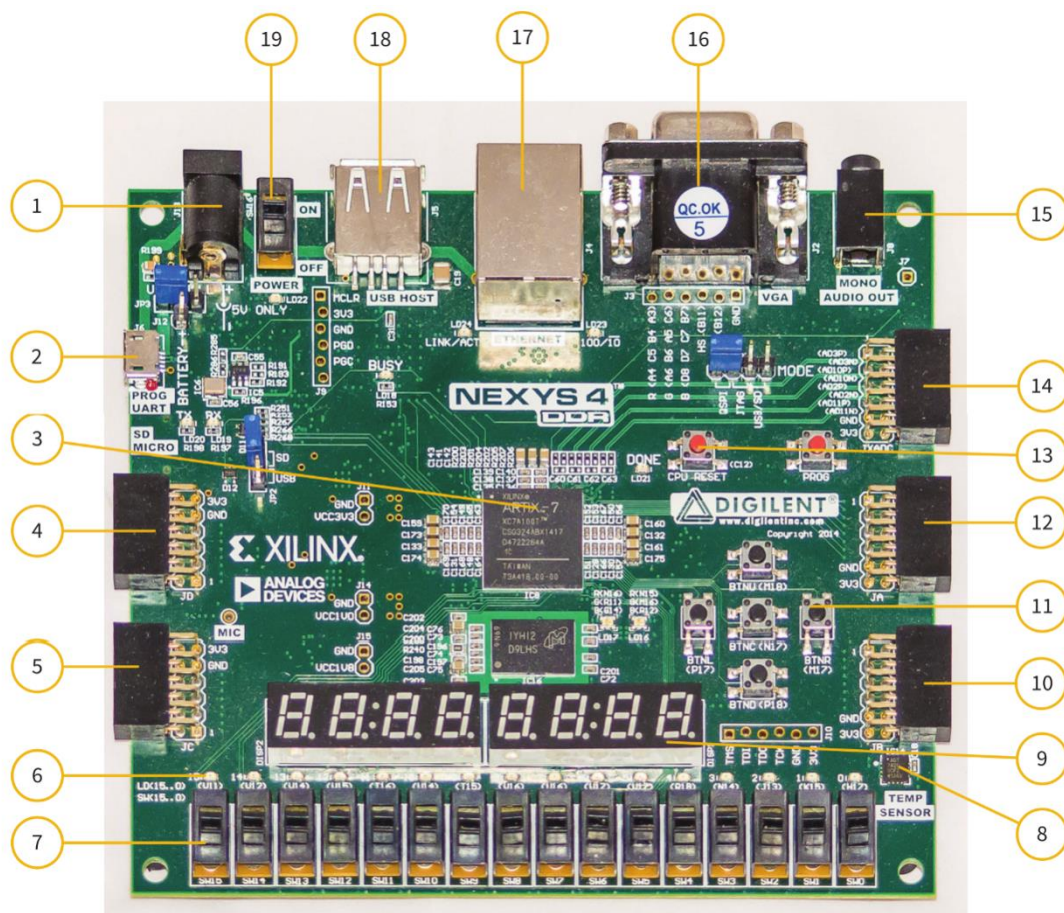


Figura 15. "Distribución de los periféricos en la plataforma nexys 4 ddr. Fuente: Manual de referencia nexys 4 DDR.

Descripción general de los sistemas empotrados

Un sistema empotrado (o un sistema informático empotrado) se puede definir de forma flexible como un sistema informático diseñado para realizar una o algunas tareas específicas. El sistema informático no es el producto final sino una parte dedicada "integrada" de un sistema más grande que a menudo incluye partes electrónicas y mecánicas adicionales.

Por el contrario, un sistema informático de uso general, como una PC (computadora personal), es una plataforma informática general y en sí misma es el producto final. Nos referimos a él como un sistema informático parecido a un escritorio en el libro. Un sistema tipo escritorio está diseñado para ser flexible y para soportar una variedad de necesidades del usuario final. Los programas de

aplicación se desarrollan en función de los recursos disponibles del sistema informático de uso general. (PONG P. CHU, 2017).

SoC Basado en FPGA

Un sistema empotrado de "gama alta" generalmente tiene un procesador y periféricos de E/S simples para realizar tareas generales de administración de la interfaz de usuario y aceleradoras de hardware especiales para manejar operaciones de computación intensiva. Estos componentes se pueden empotrar en un solo circuito integrado, comúnmente conocido como SoC (sistema en un chip). (P. Chu, 2017).

Núcleos IP

En el desarrollo de SoC, los sistemas frecuentemente tienen ciertas funcionalidades comunes y los mismos bloques de construcción se pueden reutilizar en diferentes diseños. Estos componentes se conocen como núcleos de IP (propiedad intelectual) o simplemente como IP. Son algo así como funciones en una biblioteca de software, que se pueden usar en diferentes programas de aplicación. (PONG P. CHU, 2017).

Software Vivado - Xilinx

Vivado Design Suite o suite de diseño vivado es el entorno de desarrollo para construir sistemas de procesadores integrados MicroBlaze (o ARM - vea Zynq) en FPGA Xilinx. reemplazando a Xilinx ISE con características adicionales para el desarrollo del sistema en un chip y síntesis de alto nivel. Las versiones anteriores usaban el paquete de desarrollo EDK (Embedded Development Kit) de Xilinx. (PONG P. CHU, 2017).

Microblaze

Microblaze es un microprocesador por software diseñado para Xilinx (FPGAs). Como procesador de software, MicroBlaze se implementa completamente en el tejido de memoria y lógica de uso general de los FPGA Xilinx. La CPU MicroBlaze es una familia de configuraciones de microprocesador RISC de 32 bits predefinidas y modificables. Los diseñadores de sistemas pueden aprovechar el kit de desarrollo de software Xilinx basado en Eclipse sin costo sin experiencia previa en FPGA para comenzar a desarrollar de inmediato para el procesador MicroBlaze utilizando kits de evaluación seleccionados. El procesador MicroBlaze cumple con los requisitos de muchas aplicaciones

diversas, incluidos los mercados industrial, médico, automotriz, de consumo y de comunicaciones. (PONG P. CHU, 2017).

Soporte de dispositivo

A partir de 2018, Xilinx recomienda Vivado Design Suite para nuevos diseños con Ultrascale, Ultrascale +, Spartan-7, Virtex-7, Kintex -7, Artix -7 y Zynq-7000. Vivado admite dispositivos más nuevos de alta capacidad y acelera el diseño de lógica programable y E / S. Vivado proporciona una integración e implementación más rápida para sistemas programables en dispositivos con tecnología de interconexión de silicio apilada en 3D, sistemas de procesamiento ARM, señal mixta analógica (AMS) y muchos núcleos de propiedad intelectual (IP) de semiconductores. (PONG P. CHU, 2017).

Kit de desarrollo de software xilinx (SDK)

El Kit de desarrollo de software (SDK) de Xilinx® es un entorno de desarrollo integrado (IDE) para el desarrollo de aplicaciones de software integradas destinadas a procesadores empotrados de Xilinx. SDK funciona con diseños de hardware creados con Vivado® Design Suite. SDK se basa en el estándar de código abierto Eclipse. Las características del SDK incluyen:.

- Soporte para Zynq UltraScale + MPSoC, Zynq-7000 SoCs y MicroBlaze
- Editor de código C / C ++ rico en funciones y entorno de compilación
- Basado en Eclipse 4.5.0 y CDT 8.8.0 (a partir de la versión 2016.3)
- Entorno de diseño integrado completo (IDE) que interactúa directamente con el entorno de diseño de hardware integrado de Vivado
- Diseño completo de software y flujos de depuración compatibles, incluidas capacidades de multiprocesador y co-depuración de hardware / software
- Editor, compiladores, herramientas de compilación, gestión de memoria flash e integración de depuración JTAG
- Conjunto completo de bibliotecas y controladores de dispositivos
- FreeRTOS integrado como RTOS disponible para todas las plataformas (PONG P. CHU, 2017).

Plataforma SoC FPRO

Fpro es una plataforma simple SoC (Sistema en un chip) para iniciarse en la metodología poderosa de diseño de hardware. Contiene un subsistema de video y un subsistema de E/S mapeado en memoria con periféricos de uso

general, aceleradores de hardware personalizados y un sintetizador de música. Nuestro estudio todavía se centra en el diseño de hardware, pero dentro del contexto de SoC. Las principales características de la plataforma FPro SoC son las siguientes:

- Sencillo. La plataforma FPro SoC define un protocolo de bus síncrono simple y una estructura de controlador de dispositivo directa. Una vez que se desarrolla un circuito de hardware, se puede convertir a un núcleo de IP agregando un circuito de interfaz simple y un controlador de dispositivo. Entonces, el núcleo se puede incorporar al sistema integrado existente.
- Funcional. La plataforma SoC de FPro proporciona una variedad de periféricos de E/S e interfaces en serie de uso común (UART, SPI e I²C) e incluye controladores de dispositivos en funcionamiento. Se asemeja a una placa de microprocesador de metal de 32 bits y puede implementar proyectos reales diseñados para este tipo de placas.
- Portátil. Excepto por el procesador, los núcleos IP de FPro SoC se desarrollan desde cero en HDL y no utilizan los componentes de propiedad de ningún proveedor. El protocolo de bus y los controladores de dispositivo tampoco están vinculados a ninguna plataforma comercial específica. Por lo tanto, los núcleos IP y los códigos de software son portátiles y pueden reutilizarse para diferentes dispositivos FPGA y placas de prototipos.
- "Compatible con versiones anteriores". Si bien la plataforma FPro SoC es simple, el desarrollo sigue principios y prácticas de diseño rigurosos y comprobados. Estos conocimientos y habilidades se pueden aplicar en el futuro para plataformas comerciales más complejas y proyectos más grandes. De hecho, los núcleos y controladores de IP desarrollados se pueden modificar fácilmente para incorporarlos a los marcos de IP comerciales existentes.
- Divertido. Debido a que el sistema desarrollado es como una placa de microprocesador real, puede incorporar módulos de E/S existentes y desarrollar rápidamente un proyecto de prototipado funcional. Además, esta plataforma puede proporcionar capacidad de aceleración de hardware y, por lo tanto, es más capaz y más flexible que cualquier placa de microprocesador. Esto nos da la oportunidad de desarrollar proyectos

interesantes y desafiantes y hacer que el estudio del hardware sea más "divertido" en lugar de "aprender hardware por el bien del hardware". (PONG P. CHU, 2017).

CAPITULO III: MARCO METODOLÓGICO

3.1 Tipo y diseño de la investigación

Experimental

3.2 Operacionalización de variables

Variable	Definición conceptual	Indicadores
Temperatura	Grado o nivel térmico del invernadero	13-30°C
Humedad relativa	Cantidad de agua en el aire del invernadero en forma de vapor	55%
Humedad del suelo	Cantidad de agua por volumen de tierra que hay en el cultivo	25%

Tabla 5. "Operacionalización de variables." Fuente: Elaboración propia.

3.3 Técnicas e instrumentos para la recolección de datos

Cuestionarios a expertos en la materia

Para recolectar información se realizó visitas a campo con el propósito de conocer a fondo la problemática. Con un cuestionario de preguntas se dialogó con diversos expertos en la materia. A continuación, se muestran las preguntas que se hicieron y las respuestas que se obtuvieron:

Experto A

¿Qué tipo de invernadero se utiliza en la ciudad de Tacna?

Se utiliza un invernadero tipo capilla a dos aguas con una malla antiafida, no hay discusión de querer poner uno de plástico o vidrio por la rentabilidad de bajar la temperatura en verano.

¿Cuáles son los parámetros más importantes y dentro de que rango se deben manejar?

La de temperatura, humedad relativa se usa como referencia para calcular aproximadamente cuanto va a consumir de agua la planta. La incidencia de una tesis respecto a un invernadero en Tacna debe ser en la alimentar la planta cuando lo necesite, antes de controlar la temperatura, humedad relativa, u otros parámetros.

La clave de optimizar y hacer rentable un invernadero en Tacna es el poder monitorear y manejar la fertirrigación. La humedad relativa es un parámetro que se monitorea para las enfermedades de las plantas. Pionsac lleva 10 años produciendo tomate y mantiene su temperatura con una malla con la cual llega hasta los 30°C, los inconvenientes son las esporádicas garuas que existen durante el año, y las caídas o subidas en las temperaturas e humedad relativa. Cualquier cultivo en Tacna sea tomate, pimiento, etc.

La temperatura se va manejar dentro de 15 – 28 °C. La temperatura y humedad relativa en invierno bajo malla están dentro del rango normal. El riego es una pieza clave en invernaderos bajo malla en Tacna. En cuanto a la humedad relativa debe estar entre 70 y 80%.

¿En qué momento se activa el riego, se debería regar todo el tiempo?

Normalmente las plantas se alimentan de 6am - 6pm, y durante ese lapso de tiempo se debe activar las veces que necesite la planta alimentarse, el actuador está preparado para encenderse y apagarse constantemente.

Experto B

¿Cómo es el funcionamiento del algoritmo de control del invernadero y cuál debería ser lo óptimo?

Mediante estadística calculamos cuando se va a regar las plantas de acuerdo a un análisis por tablas del viento, evaporización, temperatura. Después ponemos los horarios de riego en un programado.

¿Qué sistema utilizan para la fertirrigación?

Un programador de riego con el cual se alimenta a las plantas de tomate por tiempo y estadísticas.

Experto C

¿Cuántas tiempo se necesita para cosechar el tomate y que diferencia hay con cultivar en campo abierto?

En un invernadero se puede cosechar de 2 a 3 veces al año y en campo abierto de 1 vez normalmente, pero además la diferencia de rendimiento bajo invernadero es 10 a 12 veces mas que en campo abierto.

CAPITULO IV: DISEÑO DEL SISTEMA DE CONTROL

4.1 Características de la ubicación del invernadero

La ciudad de Tacna presento las temperaturas de la figura 16.

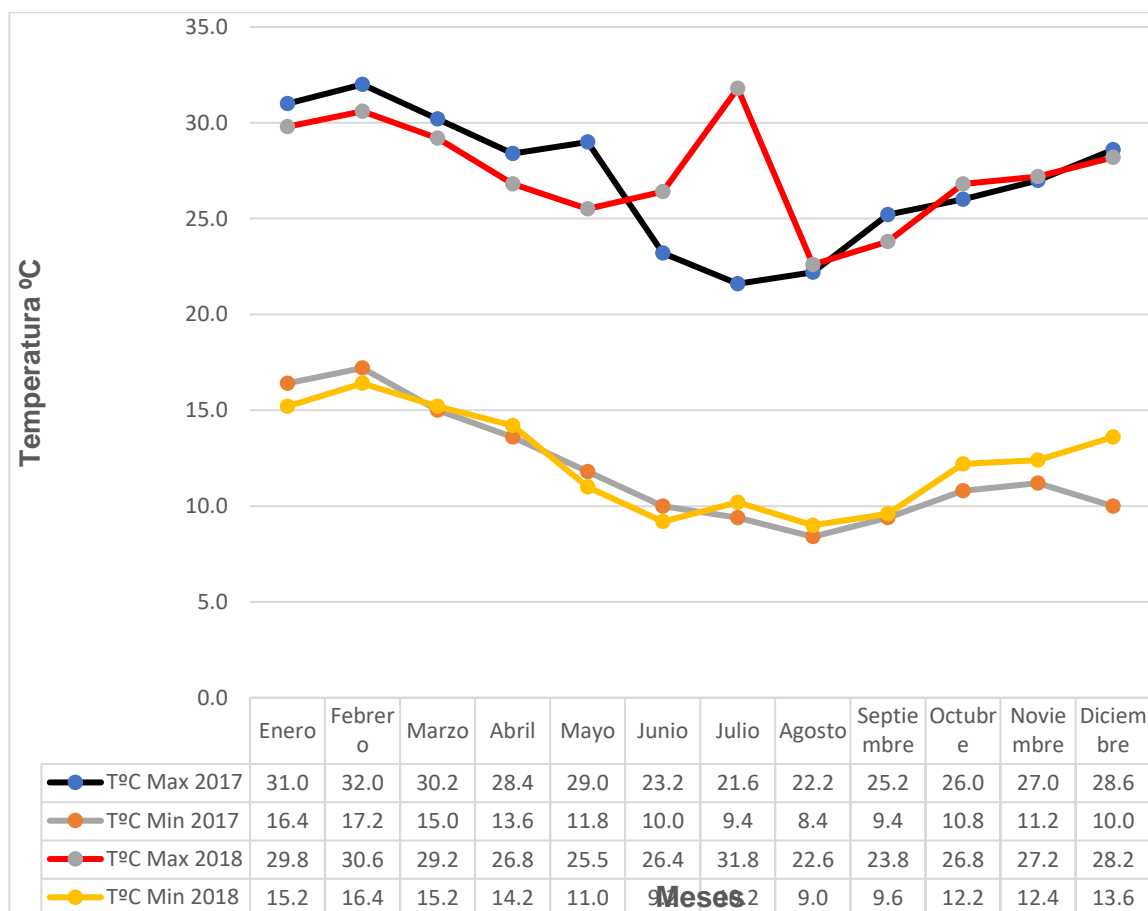


Figura 16. "Temperatura en la ciudad de Tacna 2014 y 2015." Fuente: Sehnami.

Las más altas se dieron en los meses de enero, febrero, y marzo alcanzando un valor promedio mensual para la máxima y mínima de 30.5°C y 15.9°C. Las temperaturas más bajas se presentaron en el mes de agosto, alcanzando un valor promedio mensual para la máxima y mínima de 22.4°C y 9°C.

Las humedades relativas se presentaron como se muestra en la figura 17. La más alta fue en el mes de agosto con un valor de 98.9%, el promedio anual de fue 88%. La más baja se presentó en el mes de febrero con un valor de 48.7%, el promedio anual fue de 66.8%.

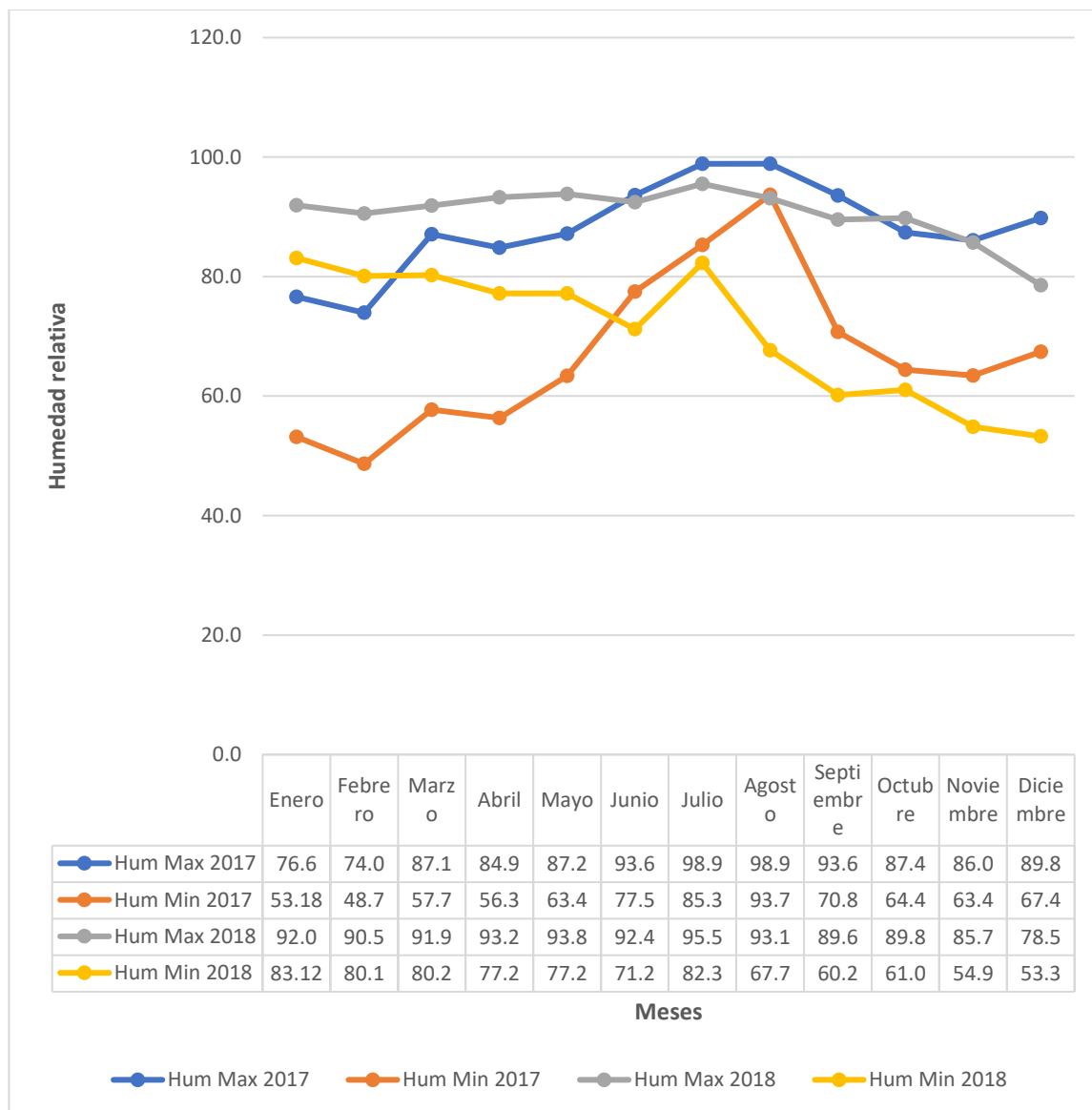


Figura 17. "Humedad relativa en la ciudad de Tacna 2014 y 2015." Fuente: Sehnami.

4.2 Requerimientos climáticos y de riego del invernadero

De acuerdo al análisis e investigación que se realizó, se resumió en un cuadro los factores que afectan la producción de tomate y sus rangos de crecimiento se detallaron en la tabla 6.

Rangos				
Factor	Rango crecimiento	de	Rango normal según cuestionario a expertos	Rango optimo
Temperatura del aire	Etapa germinación plántula	de	15 – 28 °C	24 – 26 °C
Día soleado	Plántula terminación	hasta	15 – 28 °C	25 - 27 °C
Día nublado	Plántula terminación	hasta	15 – 28 °C	22 - 24 °C
Temperatura del aire nocturno	Plántula terminación	hasta	15 – 28 °C	18 - 20 °C
Humedad relativa	Germinación el crecimiento temprano	hasta	65 – 75%	75 - 88 % 70 - 80 %
	Etapa de plántula			60 – 80 %
	vegetativa terminación	hasta		

Humedad del suelo	Plántula	hasta	Regar	hasta	1Litro de agua por
	terminación		llegar	a día	a una
			humedecer	la	conductividad
			mitad de la raíz	eléctrica entre 1.8 –	
				2.0 Ds m ⁻¹	

Tabla 6. “Comparación teórico-práctica del factor y rango óptimo para el cultivo de tomate.” Fuente: Elaboración propia

4.3 Descripción del sistema de monitorización y control

En la figura 18 se mostró el diagrama de bloques del sistema de control climático y de riego propuesto.



Figura 18. “Diagrama de bloques del sistema de control climático y riego propuesto.” Fuente: Elaboración propia.

Estuvo conformado por un controlador, tres sensores, un actuador, y un monitor de visualización en consola en una laptop.

Los sensores midieron la temperatura, humedad relativa y humedad del suelo. Los valores medidos del clima fueron enviados al microprocesador directamente, mientras que los de humedad del suelo pasaron antes por un acondicionamiento de señal.

La monitorización de los valores de los sensores se realizó a través de la medición y muestra de datos en el monitor de visualización, cuando los valores se desviaban de los rangos normales estos eran regulados automáticamente.

La regulación del riego se llevó a cabo mediante la activación del sistema de riego, este se activaba o no en base al algoritmo de control, pero siempre su objetivo principal era mantener el riego en la mitad de la raíz de la planta. Esta señal de control previamente paso por un acondicionamiento de la señal,

Finalmente, el microprocesador se comunicó a través del puerto uart con una laptop que contaba con un monitor de visualización y mostro los parámetros climáticos en tiempo real.

4.4 Descripción de los bloques del diseño propuesto

4.4.1 Planta invernadero

La figura 19 nos mostró una propuesta de cómo debería ser la disposición de sensores y actuadores dentro de un invernadero.

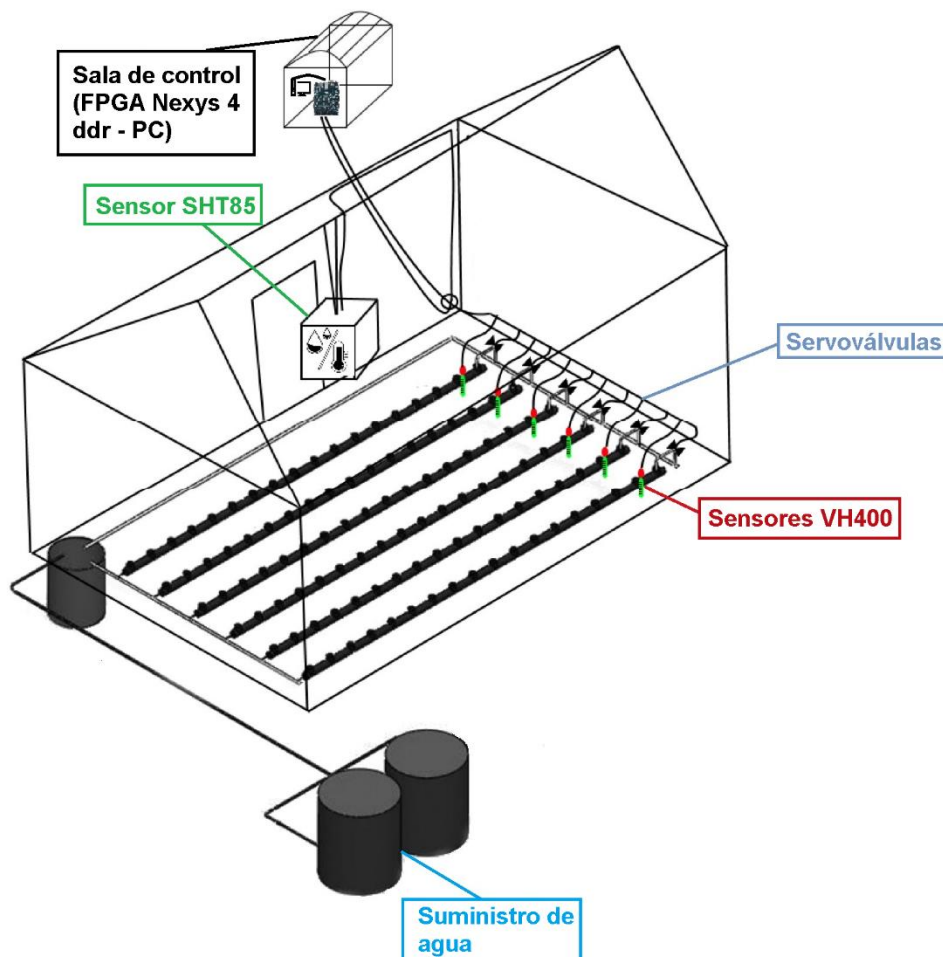


Figura 19. "Prototipo disposición de sensores y actuadores propuesto." Fuente: Elaboración propia.

Los sensores de temperatura están en contacto directo con la humedad y el agua ya que estos están diseñados para ello. El sensor temperatura y humedad relativa se ubicaron a una altura media-baja con respecto a la altura del invernadero, en una caja protectora con el objetivo de protegerlos del entorno hostil al cual están expuestos. Los sensores de humedad de suelo y servoválvulas estuvieron ubicados de forma paralela a las líneas de cultivo.

Los cables de sensores y actuadores todos han sido conectados al microprocesador el cual estuvo ubicado en un área externa al invernadero. Por último, en el modelo se agregó una puerta ubicada en la parte lateral media que permitió el ingreso del personal al invernadero.

4.4.2 Sensores

La elección se realizó conforme a la hoja técnica de los sensores, investigación en libros textos y requerimientos determinados por el experto, en la siguiente subsección se dan los detalles.

4.4.2.1 Selección del sensor de temperatura y humedad relativa

La temperatura y humedad relativa frecuentemente oscilaba entre 15 y 30°C, 60 y 80% respectivamente. La investigación y evaluación de sensores en el capítulo 2 llevó a reducir el número de opciones al seleccionar el elemento de medición más adecuado para el interior del invernadero. La termocupla y el RTD fueron más convenientes para uso industrial, alta temperatura, alta precisión. Dos opciones muy aceptadas fueron usar termistores y circuitos integrado, pero al compararlos en base a los requerimientos, los circuitos integrados fue la opción más adecuada.

El sensor SHT85 de tipo capacitivo fue el seleccionado, los criterios determinantes fueron precisión, linealidad, bajo costo. La resolución no fue relevante para este tipo de aplicación.

	M12FTH4Q	PH1125	SHT85
Rango temperatura	(-40 a 85)°C	(-30 a 80) °C	(-40 a 105)°C
Rango humedad relativa	(0 a 100)%	(10 a 95) %	(0 a 100)%
Tiempo de respuesta		HR: 10 seg	HR: 8 seg T: 2 seg
Precisión	HR: +/- 2% T: +/- 0.4 °C	HR: +/- 2% T: 0.75°C	HR: +/-1.5% T: +/- 0.1°C
Resolución	HR: 0.1% T: 0.1 °C	HR: 0.4% T: 0.2 °C	HR: 0.01% T: 0.01°C
Costo (S/.)	615.00	167.90	S/100

Tabla 7. "Comparación de sensores industriales de temperatura y humedad relativa." Fuente: Elaboración propia.

4.4.2.2 Selección del sensor de humedad del suelo

Los tipos de sensores de humedad de suelo que se tuvo disponibles fueron: FDR, TDR bloques de yeso y sondas de neutrones. Los más utilizados en invernadero fueron FDR y TDR. El sensor seleccionado fue el VH400 que usa tecnología FDR, ya que fue el mejor en estabilidad a largo plazo y bajo costo.

	TDR 100	Decagon EC-5	VH400
Rango medición	0% de vwc hasta saturación	(0 a 100) %	(0 a 100) %
Voltaje alimentación	de (3 a 5)V @ 6 a 10mA	(2.5 a 5)VDC @ 10mA	(3.5V a 20VDC)@ < 13mA
Precisión	+/-3%	+/- 2%	2% (Para 25 °C)
Dimensiones	(7.8x7.5x2.5) cm	(8.9x1.8x0.7) cm	(9.3x3.9x0.7) cm
Extensión máxima del cable		40m	10m
Estabilidad a largo plazo	Media	Buena	buena
Tecnología	TDR	FDR	FDR
Costos (S./)	802.73	633.96	188.00

Tabla 8. "Comparación de sensores industriales de humedad de suelo." Fuente: Elaboración propia

4.4.3 Acondicionamiento de la señal del sensor VH400

El modelo VH400 de la empresa vegetronix nos permitió acceder a un sensor de bajo costo con una precisión del 2% a 25°C, con un rango de medición que abarco desde 0% hasta 100%, de respuesta lineal por tramos y se muestran en la figura 20 y tabla 7.

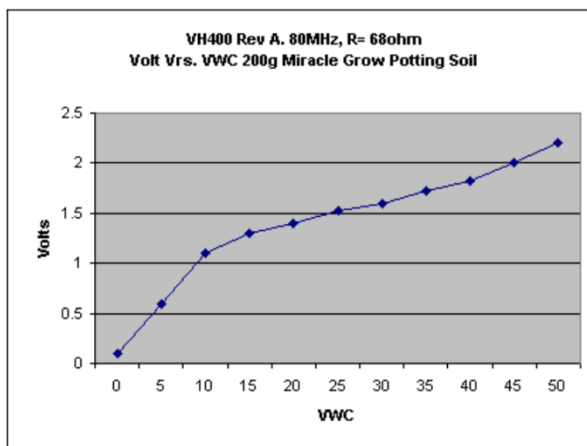


Figura 20. "Curva de VWC versus voltaje de salida sensor VH400." Fuente: Vegetronix.com

Rango de voltaje	Ecuación
0 a 1.1V	$VWC = 10V - 1$
1.1 a 1.3V	$VWC = 25V - 1$
1.3 a 1.8V	$VWC = 48.08 - 47.5$
1.82 a 2.2V	$VWC = 26.32 - 7.89$

Tabla 9. "Ecuaciones del voltaje de salida del sensor VH400." Fuente: Vegetronix.com

Como el sensor tuvo un voltaje con rango salida de 0 a 3V y la configuración del adc que se seleccionó solo admitía una señal de entrada de 0 a 1v, se acondiciono la señal mediante un divisor de tensión, el cual se mostró en la figura 21.

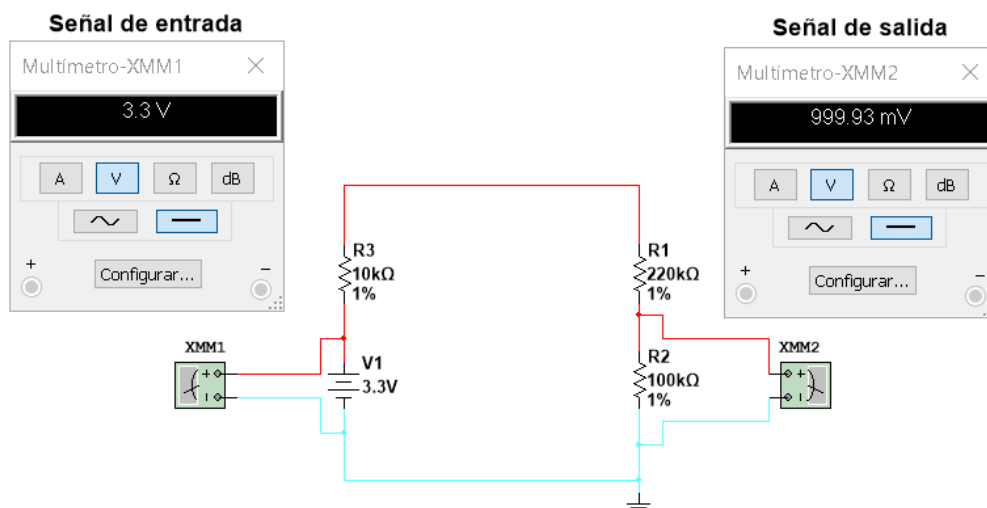


Figura 21. Circuito de acondicionamiento sensor – adc. Fuente: Elaboración propia.

Además, como el sensor tenía una impedancia de 10k por lo cual ponerlo en la entrada del adc se obtenía una imprecisión debido a que la impedancia no era la misma. Se utilizó 2 resistencias de precisión del 1% para crear un divisor de voltaje 1:3. Como la impedancia de entrada era grande al adc (condensadores conmutados), se utilizó una resistencia en serie de 220kohms a una resistencia de 100k a tierra, con eso se obtuvo la reducción de voltaje necesaria sin la imprecisión de lectura del 3%.

4.4.4 ADC

El diagrama de bloques conceptual de una macrocelda XADC se mostró en la figura 22.

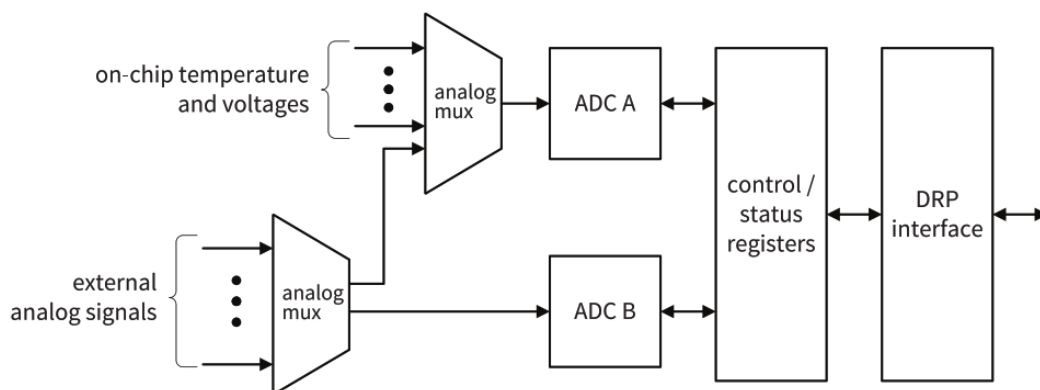


Figura 22. "Diagrama de bloques conceptual del XADC." Fuente: (PONG P. CHU, 2017).

Tuvo 5 partes principales: Dual ADC, sensores y alarmas en chip, multiplexores analógicos, registros de control y registros de estado y una interfaz DRP (puerto con reconfiguración dinámica)

Los componentes clave fueron dos ADCs con una resolución de 12bits y una velocidad de muestreo de 100 MSPS(muestras por segundo). Utilizo una estructura de entrada diferencial, en el cual la diferencia se dio a través de dos pines, estas se utilizaron para la medición, la máxima oscilación de la señal de entrada fue de 1.0V. (PONG P. CHU, 2017).

El tipo de configuración que se utilizó fue el modo unipolar, teniendo un rango de tensión entre 0.0V y +1.0V.

4.4.5 Selección de la plataforma de control

Se seleccionó la plataforma nexys 4 ddr y los criterios principales de selección fueron: hardware reconfigurable y entradas/salidas con capacidad de procesamiento en paralelo. En la tabla 10 se comparó los dos microchips comúnmente usados en comparación a un microprocesador de tipo software.

Características	Microprocesador Microblaze 3.0	Microcontrolador MC9S08QE32	Microcontrolador ATmega328P
Arquitectura	32 BITS	8 BIT	8 BIT
Almacenamiento de código	bloque interno/externo ram	bloque interno ram	bloque interno ram
Espacio de almacenamiento de datos	0 a 4 Gb	32 bytes	32 bytes
Entradas/Salidas	Paralelo	Serial	Serial
Multiplexores hardware	Si	No	No
Hardware reconfigurable	Si	No	No
Uso industrial	Si	No	No

Tabla 10. "Comparación de microchips comúnmente usados vs microprocesador de software". Fuente: Elaboración propia.

4.4.5.1 Microprocesador - FPGA Nexys 4 ddr

Para lograr un diseño electrónico reconfigurable el diseño electrónico consto de tres partes:

La primera parte fue el desarrollo del hardware - SoC (Sistema en chip) el cual fue programado en en el lenguaje VHDL, para su desarrollo se utilizó la suite de herramientas de vivado para importar el microprocesador microblaze 3.0 de 32bits y el sistema FPro del dr. Ponchu, del cual se reutilizo el subsistema mmio (Entradas/salidas de multipropósito), las modificaciones se ubicaron en el anexo C - Código del programa del sistema de control VHDL (Hardware) ,

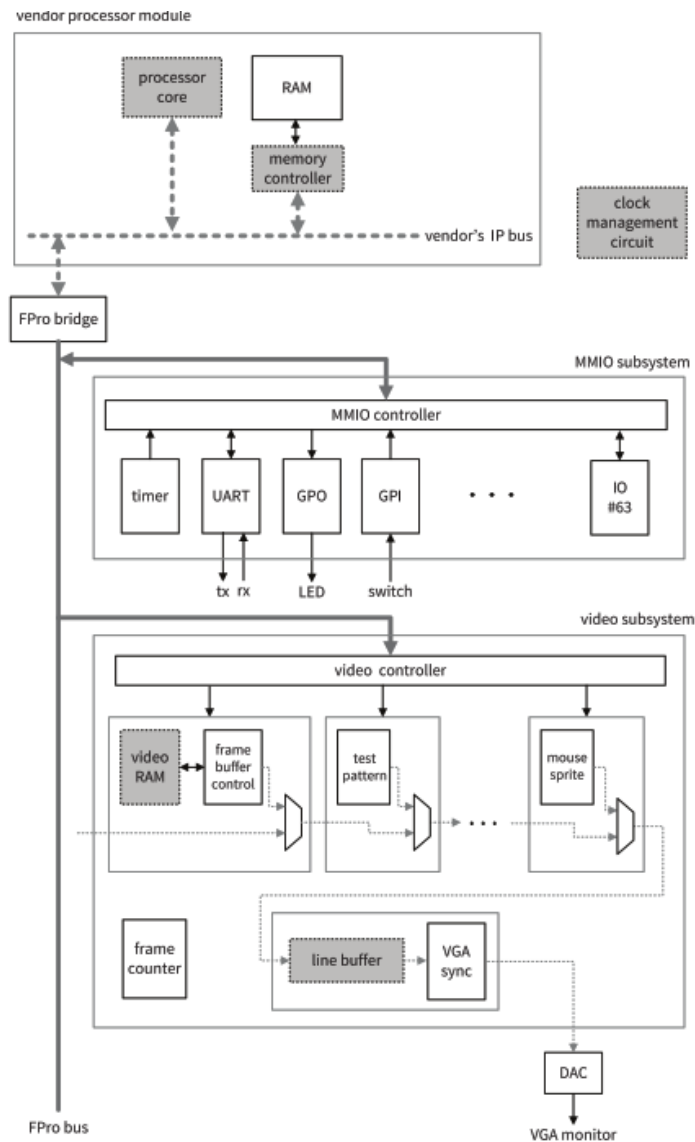


Figura 23. "Diagrama de nivel superior del sistema FPro." Fuente: (PONG P. CHU, 2017).

La segunda parte fue el desarrollo del software en el entorno de desarrollo de xilinx SDK, en el cual se realizó el algoritmo de control en el lenguaje C++ orientado a objetos, el cual se ubicó en el anexo B- Código del programa del sistema de control c++ orientado a objetos (Software).

La tercera parte fue la vinculación del hardware y el software en el entorno de desarrollo vivado. La arquitectura del sistema Fpro se mostró en la figura 23.

En el microprocesador está el algoritmo basado en una lógica de control de tipo proporcional. Este recibirá los datos recolectados por los sensores y en base al algoritmo proporcionará una señal PWM al actuador regulando el ángulo del servomotor.

Finalmente, con el control del ángulo del servomotor se logra regular el riego hacia las plantas en el invernadero, y mantener la temperatura y humedad relativa, humedad de suelo en sus valores óptimos entre 24 y 26°C, 70-80%, respectivamente.

4.4.6 PWM

Un circuito PWM básico consto de un contador binario y un comparador como se muestro en la figura 24.

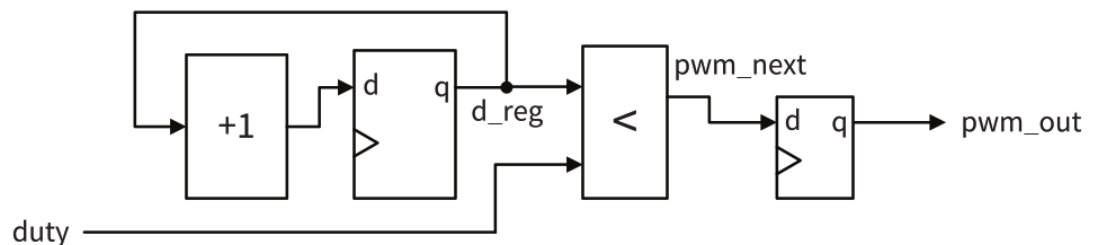


Figura 24. "Diagrama de bloques del circuito básico de un PWM." Fuente: (PONG P. CHU, 2017).

Pero tuvo dos problemas no se pudo controlar la frecuencia de conmutación y tampoco se pudo llegar al 100% del ciclo de trabajo. Y se resolvió al agregar dos componentes más un pre-escalador o contador especial (redujo la frecuencia del fpga a una frecuencia más baja) y un circuito de comparación extendido. (PONG P. CHU, 2017).

El circuito PWM se configuro con una resolución de 8 bits el cual tiene una cantidad de 256 pasos, con el fin de llegar a la mitad de cada raíz de planta.

4.4.7 Acondicionamiento de la señal del actuador

El circuito eléctrico que acondiciono la señal del microprocesador hacia el servo motor se mostró en la figura 25.

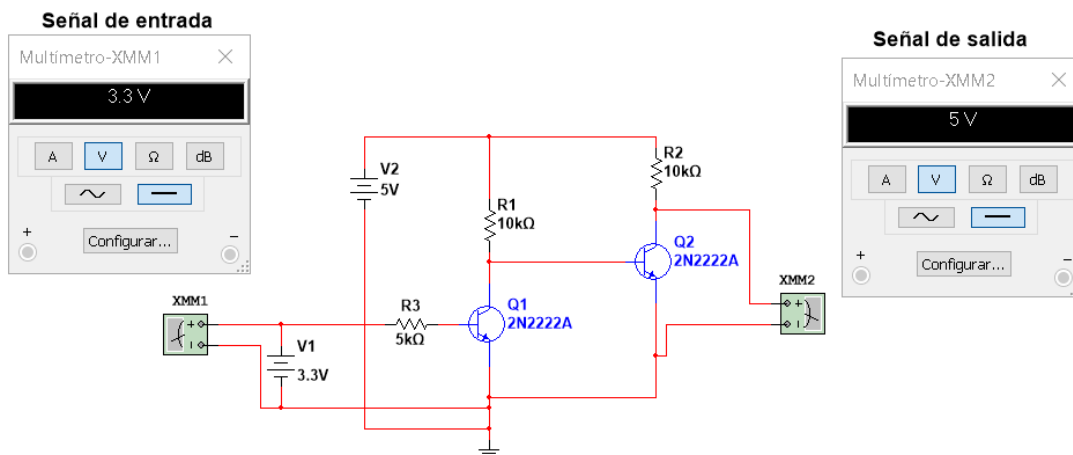


Figura 25. “Circuito de acondicionamiento microprocesador – servomotor.” Fuente: Elaboración propia.

Se utilizó 2 transistores 2N2222A para construir un buffer no inversor y lograr amplificar la señal de 3.3v hasta los 5v mínimos que necesito el servo motor.

4.4.8 Actuador

El actuador que se seleccionó como reemplazo de una servoválvula fue un servomotor de la marca towerpro modelo MG996R, ya que ambo admiten como señal de control una señal pwm se mostró en la figura 26.

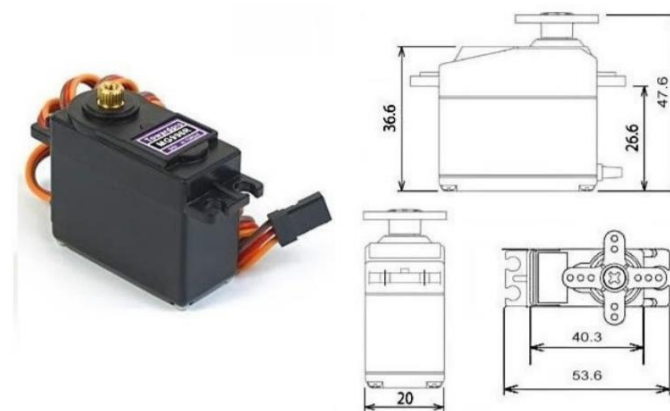


Figura 26. “Servo TowerPro Modelo MG996R.” Fuente: Electronicoscaldas.com

Este servo estándar puede girar aproximadamente 120°, así se pudo lograr regar una cierta cantidad de plantas en la línea de cultivo, con una rotación que dependió del tamaño de la raíz de la planta de tomate.

4.4.9 Monitor de visualización

El tipo de tecnología más usado fue el tipo uart y fue el que se empleó para el diseño propuesto. El monitoreo de las variables climáticas en el invernadero se visualizó a través de una consola en un monitor de visualización en una laptop y se mostró en la figura 27

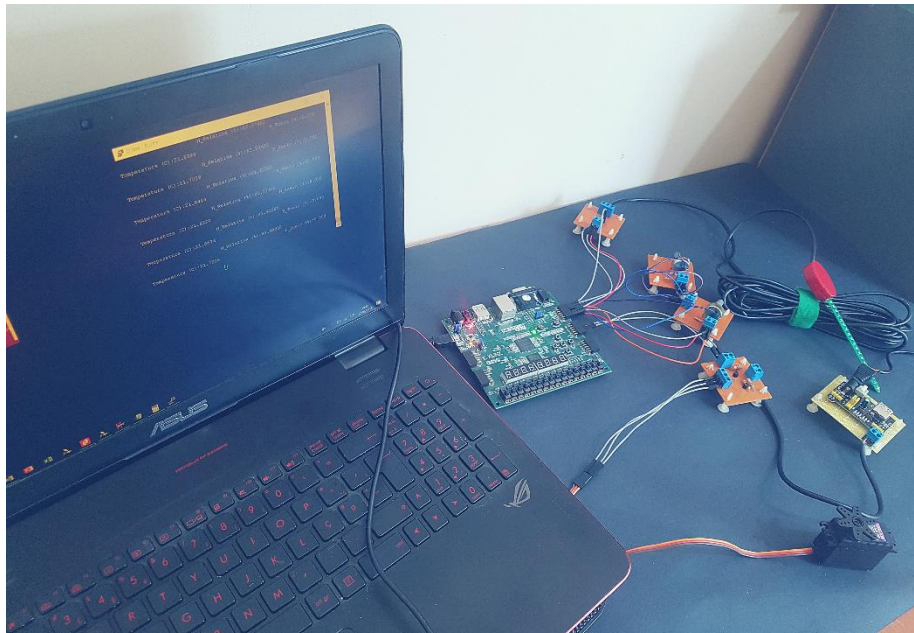
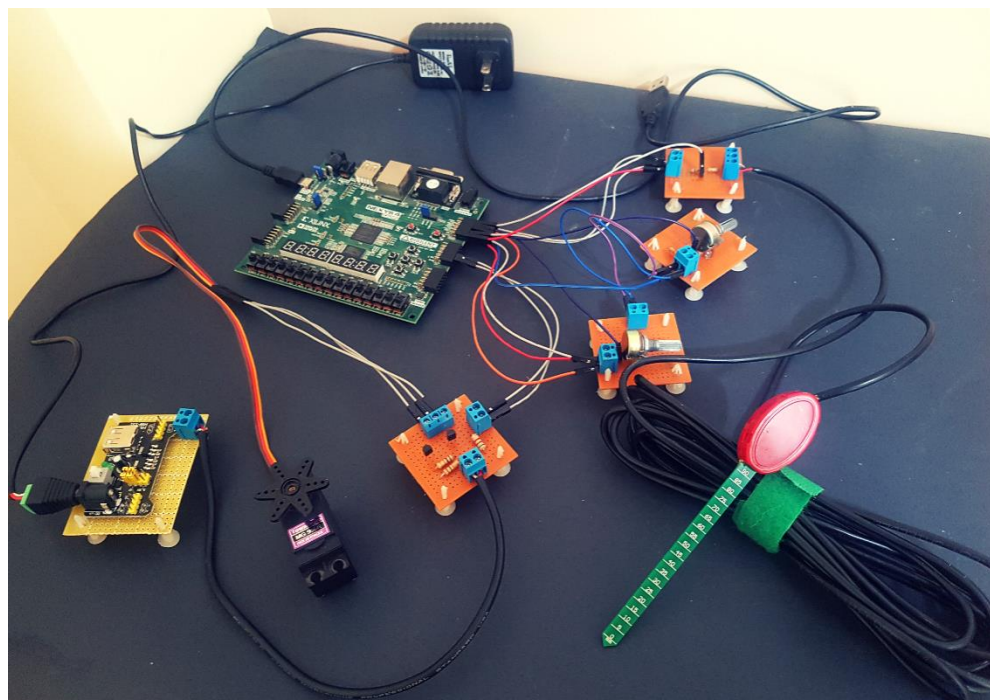


Figura 27. "Monitor de visualización en consola." Fuente: Elaboración propia.

CAPITULO V: ANALISIS Y RESULTADOS

5.1 Sistema físico de control y riego

El sistema físico de control de clima y riego del invernadero se muestra en la figura 28.



*Figura 28. "Sistema físico de control de clima y riego para un invernadero."
Fuente: Elaboración propia.*

Este sistema fue sometido a prueba en función de su capacidad para responder y garantizar a una combinación de reglas aportados por expertos en el ámbito de la supervisión y operación de invernaderos en la ciudad de Tacna.

El propósito fue de probar la confiabilidad de controlar el clima y riego dentro de un invernadero, por lo que se llevó a cabo un experimento que permite evidenciar este motivo.

En vista de que no se contó con una infraestructura y equipo necesario para hacer pruebas dentro de un invernadero. Con el sistema físico creado se simuló las posibles entradas y salidas que podía tener el sistema de control.

El simulador consiste en una plataforma FPGA con 3 entradas para sensores. La salida PWM de la plataforma está conectada al servomotor para representar la servoválvula que se debe controlar en el invernadero.

Se incluyeron dos escenarios diferentes en el experimento.

La Figura 29 y la Figura 30 muestran las respuestas del controlador a cada escenario de caso.

5.2 Resultados del primer escenario

El primer caso se da en la época de verano cuando la temperatura se eleva y la humedad relativa tiende a bajar, y la planta de tomate normalmente consume más agua.

Temperatura	Humedad relativa	Humedad de suelo	Apertura
28.9953	63.4521	0.0000	0%
28.9245	63.5253	0.0000	0%
28.8891	63.5986	0.0000	0%
28.9245	63.57421	0.0000	0%
29.0661	63.5253	0.0000	0%
28.9953	63.4521	23.974	100%

Tabla 11. "Monitoreo y control de valores para el primer escenario. Fuente: Elaboración propia."

Esta temperatura si es mayor a 26°C, la humedad relativa es menor al 70% y la humedad del suelo está por debajo del 40% entonces el actuador tiende a encenderse y apagarse más veces.

Podemos visualizar en la tabla 11 que el sistema responde al primer escenario con éxito, monitoreando cuando una variable sale de su valor de referencia y luego procede a aperturar el servomotor.

5.3 Resultados del segundo escenario

El segundo caso se da en la época de invierno cuando la temperatura baja y la humedad relativa tiende a subir, y la planta de tomate consume menos agua. Si esta temperatura está por debajo de 24°C, la humedad relativa es mayor al 80% y la humedad del suelo es mayor del 50% entonces el actuador tiende a encenderse y apagarse menos veces.

Temperatura	Humedad relativa	Humedad de suelo	Cierre
21.1364	82.0068	5.371	0%
20.9240	82.2509	5.859	0%
20.9594	82.2021	5.981	0%
20.8532	82.1533	5.493	0%
21.0656	82.0556	5.639	0%
20.9948	82.0800	23.779	100%

Tabla 12. Monitoreo y control de valores para el segundo escenario. Fuente: Elaboración propia.

Podemos visualizar en la tabla 12 que el sistema responde al segundo escenario con éxito, monitoreando cuando una variable sale de su valor de referencia y luego procede cerrar el servomotor.

El riego es controlado por la señal pwm la cual nos permite tener una apertura grado a grado teniendo la posibilidad de llegar a la mitad de la raíz de cada planta de tomate a lo largo de las líneas de cultivo.

El resultado obtenido de cada escenario de caso muestra que se tomaron medidas inmediatas para ajustar el nivel de clima en el invernadero de acuerdo con las condiciones dadas al controlador por expertos en la materia. No se detectaron problemas ni demoras en todo el proceso de prueba, lo que indica que el sistema de control desarrollado funcionó satisfactoriamente.

5.4 Resultados de los recursos de la plataforma nexys 4 ddr

La implementación del sistema de control se realizó con la herramienta Vivado 2017.4. Los resultados de los recursos utilizados se mostraron en la tabla 13.

Componentes de la aplicación	Total de recursos de la tarjeta		Recursos utilizados en la PC		Recursos libres	
	Cantidad	%	Cantidad	%	Cantidad	%
BRAMs	132	100	32	23.70	100	76.30
Slices	63400	100	2954	4.66	60446	95.34
Slices Flip Flops	126800	100	2861	2.26	123939	97.74
Pines E/S	210	100	4	1.90	206	98.10

DSPs	240	100	3	1.25	237	98.75
-------------	-----	-----	---	------	-----	-------

Tabla 13. "Comparación de los recursos de la plataforma nexys 4 ddr". Fuente: Elaboración propia

Se pudo observar que no se utilizan todos los recursos indicados de la plataforma Nexys 4 ddr.

Las 4 BRAM están configuradas para constituir una memoria de palabras de 32 bits. El controlador deja libre el 76.30% del espacio y el 98.10% de los pines de la FPGA, lo que permite reservar recursos para futuras mejoras y adaptaciones.

CONCLUSIONES

- El objetivo enfoque principal de la tesis se ha cumplido con éxito. El diseño simulado e implementado es reconfigurable y cuenta disponibilidad de crecimiento a futuro sin necesidad de cambiar toda la placa. Además, ha logrado responder a escenarios típicos en un invernadero, planteados por expertos en la materia en la ciudad de Tacna.
- Se seleccionó sensores de temperatura, humedad relativa y humedad de suelo, así como un actuador que permitió el funcionamiento adecuado del sistema de control.
- Se diseñaron las etapas de acondicionamiento de señal de los sensores y actuadores, y luego se comprobó su correcto funcionamiento en el capítulo V de resultados.
- El monitoreo y control del clima de un invernadero es un área atractiva, pero al igual que toda tesis investigada en este tema se sabe que es amplia, pero este es una buena fuente de referencia para otros tesis o desarrollo de proyectos similares.

RECOMENDACIONES

- En el acondicionamiento de la señal del sensor se recomienda añadir un diodo zener antes del divisor resistivo para proteger todo, ya que el sensor tiene voltajes que podrían dañar algo si no se tiene cuidado.
- Cuando se intente leer un sensor con la plataforma FPGA y se necesite una resolución más alta, se recomienda dimensionarlo tomando en cuenta la hoja de datos del puerto JXADC.
- Se puede mejorar el control teniendo información externa, por lo que se recomienda colocar un sensor de temperatura y humedad relativa afuera del invernadero.
- Se recomienda para mejorar la interacción del sistema de control y el personal calificado, modificar el código fuente para que el valor deseado de las variables pueda ser ingresado mediante el teclado de una PC o un Smartphone.

- En el desarrollo futuro, se recomienda probar el sistema en un entorno de invernadero con suficiente equipo con el objetivo de verificar los resultados obtenidos en el prototipo.
- Se puede implementar un sistema de procesamiento de video del cultivo de tomate para crear un sistema de control mas robusto aprovechando y reutilizando la arquitectura tipo software desarrollada para procesar video del Dr. Ponchu.
- Se pueden utilizar múltiples microprocesadores para obtener lecturas y escrituras en tiempo real de las entradas/salidas del sistema de control.

REFERENCIAS BIBLIOGRÁFICAS

- A. PAWLOWSKI, M. BESCHI, J.L. GUZMÁN, A. VISIOLI, M. BERENGUEL, S. DORMIDO. (2016). *Application of ssod-pi and pi-ssod event-based controllers to greenhouse climatic control*.
- A. RAMIRÉZ. (2014). *Diseño de un sistema para control y monitoreo de un invernadero hidropónico (tesis de pregrado)*. Instituto Politécnico Nacional, Mexico.
- A. GARG, R. GOYAL, P. MUNOTH. (2016). *Application of soil moisture sensors in agriculture: a review*.
- A. PAWLOWSKI, J. SÁNCHEZ, J. GUZMAN, F. RODRÍGUEZ, M. BERENGUEL, S. DORMIDO. (2009). *Simulation of greenhouse climate monitoring and control with wireless sensor network and event-based control*.
- E. RODOLFO. (2015). *Diseño e implementación de un sistema de control de humedad relativa y absoluta en un invernadero (tesis de pregrado)*. Universidad Distrital Francisco Jose de Caldas, Bogota D.C.
- F. RODRÍGUEZ, M. BERENGUEL, J. GUZMÁN, A. RAMÍREZ ARIAS. (2015). *Modeling and control of greenhouse crop growth (Almería)*. España: Springer.
- J.C. BAKKER, G.P.A. BOT, H. CHALLA, N.J. VAN DE BRAAK. (1995). *Greenhouse climate control*. Wageningen: Wageningen Academic Publishers.
- J.P. COELHO, P.B. OLIVEIRA, J. BOAVENTURA CUNHA. (2005). *Greenhouse air temperature predictive control using the particle swarm optimisation algorithm*.
- K. OGATA. (2010). *Ingeniería de control moderna* (Madrid). España: Pearson.
- K. LARA. (2015). *Diseño e implementación de un sistema de control microclimático para la preservación de orquídeas endémicas del Perú en invernadero (tesis de pregrado)*. Pontificia Universidad Católica del Perú, Lima.

- L. FENG, H. LI, Y. ZHI. (2013). *Greenhouse CFD simulation for searching the sensors optimal placements.*
- M. HANPING, I. ULLAH, N. JIHENG, Q. JAVED, A. AZEEM. (2017). *Estimating tomato water consumption by sap flow measurement in response to water stress under greenhouse conditions.*
- M. PELLETIER, R. SCHWARTZ, G. HOLT, J. WANJURA, T. GREEN. (2016). *Frequency domain probe design for high frequency sensing of soil moisture.*
- M. EL GHOUMARI, H. TANTAU, J. SERRANO. (2005). *Non-linear constrained MPC: Real-time implementation of greenhouse air temperature control.*
- P. PONCE, A. MOLINA, P. CEPEDA, E. LUGO, B. MACCLEERY. (2015). *Greenhouse design and control* (México). Ciudad de México: CRC Press.
- P. CHU. (2017). *FPGA Prototyping by VHDL examples 2da* (USA). Hoboken: Wiley.
- R. CASTAÑEDA, E. VENTURA, R. PENICHE, G. HERRERA. (2006). *Fuzzy greenhouse climate control system based on a field programmable gate array.*
- V. QUAN. (2011). *Review of sensors for greenhouse climate monitoring.*
- Z. FU, X. XING, X. JUN. (2005). *Greenhouse climate control techniques in China and the setup of the control system for a chamber.*
- Z. VARAS. (2013). *Diseño de un sistema electrónico para riego de césped de jardín empleando un sensor de humedad de suelos y comunicación inalámbrica.* Pontificia Universidad Católica del Perú, Lima.

ANEXOS

A. Matriz de consistencia

Interrogante	Objetivos	Hipótesis	Variables	Indicadores	Métodos
Problema general	Objetivo general	Hipótesis general	Independiente		Tipo de investigación
¿En qué medida el diseño e implementación de un sistema electrónico de control basado en una plataforma fpga, optimizara las condiciones climáticas de un invernadero en la ciudad de Tacna, en el año 2017?	- Diseñar e implementar un sistema electrónico de control basado en la tecnología fpga, que optimice las condiciones climáticas de un invernadero en la ciudad de Tacna.	El diseño e implementación de un sistema electrónico de control si optimizará las condiciones climáticas de un invernadero en la ciudad de Tacna	Diseño e implementación de un sistema electrónico de control basado en fpga	Prototipo	Experimental
Problemas específicos	Objetivos específicos	Hipótesis específicas	Dependiente		
¿En qué medida el control de humedad del suelo optimizará las	- Diseñar e implementar un sistema de control de humedad de suelo que optimice las condiciones climáticas de un	- El diseño de un sistema de control de humedad del suelo si optimizará las condiciones	Optimización de las condiciones climáticas de un invernadero	Temperatura Humedad relativa	Experimental

<p>condiciones climáticas en un invernadero de la ciudad Tacna?</p> <p>¿En qué medida el control de humedad del suelo influirá en la temperatura y humedad relativa de un invernadero en la ciudad de Tacna?</p>	<p>invernadero en la ciudad de Tacna.</p> <ul style="list-style-type: none"> - Diseñar e implementar un sistema control de temperatura y humedad relativa de un invernadero en la ciudad de Tacna. 	<p>climáticas de un invernadero.</p> <ul style="list-style-type: none"> - El diseño de un sistema control de temperatura y humedad relativa si tendrá incidencia en un invernadero. 		<p>Humedad de suelo</p>	
--	---	--	--	-------------------------	--

B. Código del programa del sistema de control c++ orientado a objetos

```

/* Declaracion de librerias */
#include "chu_init.h"
#include "xadc_core.h"
#include "gpio_cores.h"
#include "i2c_core.h"

//Bandera para que el movimiento del servo
//solo se eiecute una vez y no infinitamente
bool bandera = false;

//Control de temperatura, humedad relativa y riego
//atravès de la humedad del suelo de un invernadero
void control_HS_HR_T(XadcCore *adc_p,PwmCore *pwm_p) {
    double duty;
    double S1_HS,S1_ESC,S2_T,S2_ESC,S3_HR,S3_ESC,VWC;
    int n, i;
    uint16 t raw;

//Lectura de la temperatura
    S2_T = adc_p->read_adc_in(1);
    S2_ESC= ((S2_T*145)-(40));
    sleep_ms(100);
    uart.disp("Temperatura (C):");
    uart.disp(S2_ESC, 4);
    uart.disp("\n\r\|t|t|t|t");
    sleep_ms(200);

//Lectura de la humedad relativa
    S3_HR = adc_p->read_adc_in(2);
    S3_ESC = (S3_HR*100);
    sleep_ms(100);
    uart.disp("H_Relativa (%):");
    uart.disp(S3_ESC, 5);
    uart.disp("\n\r\|t|t|t|t|t|t");
    sleep_ms(200);

    pwm_p->set_freq(50);
    S1_HS = adc_p->read_adc_in(0);
    S1_ESC = (S1_HS*100);
    sleep_ms(100);

    if (S1_HS <= 0.1 && bandera == false ) {
        VWC = 0;
        S1_ESC = 0;
    }
}

```

```

pwm_p->set_freq(50);
S1_HS = adc_p->read_adc_in(0);
S1_ESC = (S1_HS*100);
sleep_ms(100);

if (S1_HS <= 0.1 && bandera == false ) {
    VWC = 0;
    S1_ESC = 0;
}

//Apertura del servo
else if (S1_ESC>=10 && S1_ESC<=45 && S2_ESC>=26 && S3_ESC<=70 && bandera == false ) {
    duty = 0.050;    //-90º
    pwm_p->set_duty(duty, 7);
    sleep_ms(2000);
    duty = 0.110;    //+90º
    pwm_p->set_duty(duty, 7);
    sleep_ms(1000);
    bandera=true;
}

//Cierre del servo
else if (S1_ESC>=50 && S1_ESC<=55 && S2_ESC<24 && S3_ESC>=80 ){
    duty = 0.060; //-90º
    pwm_p->set_duty(duty, 7);
    sleep_ms(1000);
    bandera = false;
}

    uart disp("H_Suelo (%):");
    uart disp(S1_ESC, 3);
    uart disp("\n\r");
    sleep_ms(200);
}

// Instancias de los nucleos
GpoCore led(get_slot_addr(BRIDGE_BASE, S2_LED));
GpiCore sw(get_slot_addr(BRIDGE_BASE, S3_SW));
XadcCore adc(get_slot_addr(BRIDGE_BASE, S5_XDAC));
PwmCore pwm(get_slot_addr(BRIDGE_BASE, S6_PWM));

//funcion principal
int main() {
    while (1) {
        control_HS_HR_T(&adc, &pwm);
    } //fin while
} // fin main

```

C. Código del programa del sistema de control VHDL

```

1  library ieee;
2  use ieee.std_logic_1164.all;
3  use work.chu_io_map.all;
4  entity mcs_top_dlx is
5      generic(BRIDGE_BASE : std_logic_vector(31 downto 0) := x"C0000000");
6      port(
7          clk          : in    std_logic;
8          reset_n     : in    std_logic;
9          -- switches and LEDs
10         sw           : in    std_logic_vector(15 downto 0);
11         led          : out   std_logic_vector(15 downto 0);
12         -- uart
13         rx           : in    std_logic;
14         tx           : out   std_logic;
15         -- xadc
16         adc_p        : in    std_logic_vector(3 downto 0);
17         adc_n        : in    std_logic_vector(3 downto 0);
18         -- rgb leds
19         rgb_led1     : out   std_logic_vector(2 downto 0);
20         rgb_led2     : out   std_logic_vector(2 downto 0);
21         -- buttons
22         btn          : in    std_logic_vector(4 downto 0);
23
24         -- 4-digit 7-seg LEDs
25         an           : out   std_logic_vector(7 downto 0);
26         sseg         : out   std_logic_vector(7 downto 0);
27
28         -- spi accelerator
29         acl_sclk     : out   std_logic;
30         acl_mosi     : out   std_logic;
31         acl_miso     : in    std_logic;
32         acl_ss_n     : out   std_logic;
33
34         -- i2c temperature sensor
35         tmp_i2c_scl  : out   std_logic;
36         tmp_i2c_sda  : inout std_logic;
37
38         -- ps2
39         ps2d         : inout std_logic;
40         ps2c         : inout std_logic;
41
42         -- nexsys 4 audio
43         audio_on     : out   std_logic;
44         audio_pdm    : out   std_logic;
45
46         -- spi sd card
47         sd_sclk      : out   std_logic;
48         sd_mosi      : out   std_logic;
49         sd_miso      : in    std_logic;

```



```

44     sd_ss_n    : out    std_logic;
45     sd_reset  : out    std_logic;
46
47     -- EMOD JA (divided into top row and bottom row
48     ja_top    : out    std_logic_vector(4 downto 1);
49     ja_btm    : out    std_logic_vector(10 downto 7);
50     -- EMOD JB (divided into top row and bottom row
51     jb_top    : inout  std_logic_vector(4 downto 1);
52     jb_btm    : inout  std_logic_vector(10 downto 7);
53     -- EMOD JC (divided into top row and bottom row
54     jc_top    : inout  std_logic_vector(4 downto 1);
55     jc_btm    : inout  std_logic_vector(10 downto 7);
56     -- EMOD JD (divided into top row and bottom row
57     jd_top    : inout  std_logic_vector(4 downto 1);
58     jd_btm    : inout  std_logic_vector(10 downto 7)
59 );
60 end mcs_top_dlx;
61
62 architecture arch of mcs_top_dlx is
63     component cpu
64     port(
65         clk            : in  std_logic;
66
67         reset          : in  std_logic;
68         io_addr_strobe : out  std_logic;
69         io_read_strobe : out  std_logic;
70         io_write_strobe : out  std_logic;
71         io_address     : out  std_logic_vector(31 downto 0);
72         io_byte_enable : out  std_logic_vector(3 downto 0);
73         io_write_data  : out  std_logic_vector(31 downto 0);
74         io_read_data   : in   std_logic_vector(31 downto 0);
75         io_ready       : in   std_logic
76     );
77     end component;
78     signal io_addr_strobe : std_logic;
79     signal io_read_strobe : std_logic;
80     signal io_write_strobe : std_logic;
81     signal io_byte_enable : std_logic_vector(3 downto 0);
82     signal io_address     : std_logic_vector(31 downto 0);
83     signal io_write_data  : std_logic_vector(31 downto 0);
84     signal io_read_data   : std_logic_vector(31 downto 0);
85     signal io_ready       : std_logic;
86     signal mmio_cs        : std_logic;
87     signal mmio_wr        : std_logic;
88     signal mmio_rd        : std_logic;

```

```

88     signal mmio_addr      : std_logic_vector(20 downto 0);
89     signal mmio_wr_data  : std_logic_vector(31 downto 0);
90     signal mmio_rd_data  : std_logic_vector(31 downto 0);
91     -- clk/reset related
92     signal clk_100M      : std_logic;
93     signal reset_sys     : std_logic;
94     -- pwm
95     signal pwm           : std_logic_vector(7 downto 0);
96     -- ddfs/audio pdm
97     signal pdm           : std_logic;
98     signal ddfs_sq_wave  : std_logic;
99     -- dlx system signals
100    signal spil_sclk      : std_logic;
101    signal spil_mosi      : std_logic;
102    signal spil_miso      : std_logic;
103    signal spil_ss        : std_logic_vector(3 downto 0);
104    signal spi2_ss        : std_logic_vector(3 downto 0);
105    signal i2cl_scl       : std_logic;
106    signal i2cl_sda       : std_logic;
107    signal rx2, tx2      : std_logic;
108    signal pwm1           : std_logic_vector(7 downto 0);

109
110    begin
111        -- clock and reset
112        clk_100M          <= clk;           -- 100 MHz external clock
113        reset_sys        <= not reset_n;
114        -- audio
115        audio_pdm        <= pdm;
116        audio_on         <= '1';
117        -- rgb leds
118        rgb_led2         <= pwm(5 downto 3);
119        rgb_led1         <= pwm(2 downto 0);
120    -- sd card
121    -- asserting sd_reset 0 to supply sd card power *see manual
122    sd_reset <='0';
123    sd_ss_n <=spi2_ss(0);
124    -- PMOD JA
125    ja_top(1)           <= ddfs_sq_wave;
126    -- ja_top(2)         <= pdm;
127    ja_top(4 downto 2) <= pwm(7 downto 5);
128    ja_btm(10 downto 7) <= "0000";
129    -- PMOD JB (per pmod spi spec; with additional cs on bottom)

```

```

130     jb_top(1) <= spi2_ss(0);
131     jb_top(2) <= spil_mosi;
132     spil_miso <= jb_top(3);
133     jb_top(4) <= spil_sclk;
134     jb_btm <= spil_ss (3 downto 1) & '0';
135     -- PMOD JC (top per pmod i2c spec; bottom per uart spec)
136     --jc_top(1) <= i2c1_sda;
137     --jc_top(2) <= i2c1_scl;
138     jc_btm(8) <= tx2;
139     rx2 <= jc_btm(9);
140     -- PMOD JD
141     jd_top <= pwml(3 downto 0);
142     jd_btm <= pwml(7 downto 4);
143
144     -- instantiate microBlaze MCS
145     mcs_0 : cpu
146         port map(
147             clk          => clk_100M,
148             reset        => reset_sys,
149             io_addr_strobe => io_addr_strobe,
150             io_read_strobe => io_read_strobe,
151             io_write_strobe => io_write_strobe,
152
153             io_byte_enable => io_byte_enable,
154             io_address      => io_address,
155             io_write_data   => io_write_data,
156             io_read_data    => io_read_data,
157             io_ready        => io_ready
158         );
159     -- instantiate MCS IO bus to FPro bus bridge
160     bridge_unit : entity work.chu_mcs_bridge
161         generic map(BRG_BASE => BRIDGE_BASE)
162         port map(
163             io_addr_strobe => io_addr_strobe,
164             io_read_strobe => io_read_strobe,
165             io_write_strobe => io_write_strobe,
166             io_byte_enable => io_byte_enable,
167             io_address      => io_address,
168             io_write_data   => io_write_data,
169             io_read_data    => io_read_data,
170             io_ready        => io_ready,
171             fp_video_cs     => open,
172             fp_mmio_cs      => mmio_cs,
173             fp_wr           => mmio_wr,
174             fp_rd           => mmio_rd,

```

```

174         fp_addr      => mmio_addr,
175         fp_wr_data   => mmio_wr_data,
176         fp_rd_data   => mmio_rd_data
177     );
178     -- instantiate sampler MMIO subsystem
179     mmio_sys_unit : entity work.mmio_sys_dlx
180     port map(
181         clk           => clk_100M,
182         reset        => reset_sys,
183         mmio_cs      => mmio_cs,
184         mmio_wr      => mmio_wr,
185         mmio_rd      => mmio_rd,
186         mmio_addr    => mmio_addr,
187         mmio_wr_data => mmio_wr_data,
188         mmio_rd_data => mmio_rd_data,
189         sw           => sw,
190         led          => led,
191         rx           => rx,
192         tx           => tx,
193         adc_p        => adc_p,
194         adc_n        => adc_n,
195         pwm          => pwm,
196
197         btn          => btn,
198         --          an           => an,
199         --          sseg        => sseg,
200         tmp_i2c_scl => tmp_i2c_scl,
201         tmp_i2c_sda => tmp_i2c_sda,
202         acl_sclk    => acl_sclk,
203         acl_mosi    => acl_mosi,
204         acl_miso    => acl_miso,
205         acl_ss      => acl_ss_n,
206         ps2d        => ps2d,
207         ps2c        => ps2c,
208         ddfs_sq_wave => ddfs_sq_wave,
209         pdm         => pdm,
210         rx2         => rx2,
211         tx2         => tx2,
212         rx3         => '0',
213         tx3         => open,
214         spil_sclk   => spil_sclk,
215         spil_mosi   => spil_mosi,
216         spil_miso   => spil_miso,
217         spil_ss     => spil_ss,
218         spi2_sclk   => sd_sclk,

```

```
218     spi2_mosi    => sd_mosi,
219     spi2_miso    => sd_miso,
220     spi2_ss      => spi2_ss,
221     i2c1_scl     => jc_top(2),
222     i2c1_sda     => jc_top(1),
223     i2c2_scl     => open,
224     i2c2_sda     => open,
225     ddfs1_sq_wave => open,
226     ddfs1_pdm    => open,
227     ddfs2_sq_wave => open,
228     ddfs2_pdm    => open,
229     pwm1         => pwm1
230   );
231
232   end arch;
233
234
```

D. Hoja de datos transistor P2N2222A

P2N2222A

Amplifier Transistors

NPN Silicon

Features

- These are Pb-Free Devices*

MAXIMUM RATINGS ($T_A = 25^\circ\text{C}$ unless otherwise noted)

Characteristic	Symbol	Value	Unit
Collector - Emitter Voltage	V_{CEO}	40	Vdc
Collector - Base Voltage	V_{CBO}	75	Vdc
Emitter - Base Voltage	V_{EBO}	6.0	Vdc
Collector Current - Continuous	I_C	600	mA dc
Total Device Dissipation @ $T_A = 25^\circ\text{C}$ Derate above 25°C	P_D	625 5.0	mW mW/ $^\circ\text{C}$
Total Device Dissipation @ $T_C = 25^\circ\text{C}$ Derate above 25°C	P_D	1.5 12	W mW/ $^\circ\text{C}$
Operating and Storage Junction Temperature Range	T_J, T_{stg}	-55 to +150	$^\circ\text{C}$

THERMAL CHARACTERISTICS

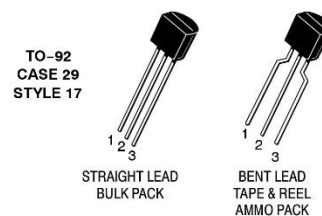
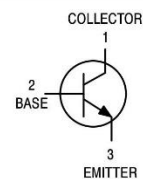
Characteristic	Symbol	Max	Unit
Thermal Resistance, Junction to Ambient	$R_{\theta JA}$	200	$^\circ\text{C/W}$
Thermal Resistance, Junction to Case	$R_{\theta JC}$	83.3	$^\circ\text{C/W}$

Stresses exceeding Maximum Ratings may damage the device. Maximum Ratings are stress ratings only. Functional operation above the Recommended Operating Conditions is not implied. Extended exposure to stresses above the Recommended Operating Conditions may affect device reliability.

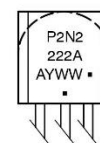


ON Semiconductor®

<http://onsemi.com>



MARKING DIAGRAM



A = Assembly Location
Y = Year
WW = Work Week
• = Pb-Free Package

(Note: Microdot may be in either location)

ORDERING INFORMATION

Device	Package	Shipping†
P2N2222AG	TO-92 (Pb-Free)	5000 Units/Bulk
P2N2222ARL1G	TO-92 (Pb-Free)	2000/Tape & Ammo

†For information on tape and reel specifications, including part orientation and tape sizes, please refer to our Tape and Reel Packaging Specification Brochure, BRD8011/D.

P2N2222A

ELECTRICAL CHARACTERISTICS ($T_A = 25^\circ\text{C}$ unless otherwise noted)

Characteristic	Symbol	Min	Max	Unit
OFF CHARACTERISTICS				
Collector - Emitter Breakdown Voltage ($I_C = 10\text{ mAdc}$, $I_B = 0$)	$V_{(BR)CEO}$	40	-	Vdc
Collector - Base Breakdown Voltage ($I_C = 10\text{ }\mu\text{Adc}$, $I_E = 0$)	$V_{(BR)CBO}$	75	-	Vdc
Emitter - Base Breakdown Voltage ($I_E = 10\text{ }\mu\text{Adc}$, $I_C = 0$)	$V_{(BR)EBO}$	6.0	-	Vdc
Collector Cutoff Current ($V_{CE} = 60\text{ Vdc}$, $V_{EB(off)} = 3.0\text{ Vdc}$)	I_{CEX}	-	10	nAdc
Collector Cutoff Current ($V_{CB} = 60\text{ Vdc}$, $I_E = 0$) ($V_{CB} = 60\text{ Vdc}$, $I_E = 0$, $T_A = 150^\circ\text{C}$)	I_{CBO}	-	0.01 10	μAdc
Emitter Cutoff Current ($V_{EB} = 3.0\text{ Vdc}$, $I_C = 0$)	I_{EBO}	-	10	nAdc
Collector Cutoff Current ($V_{CE} = 10\text{ V}$)	I_{CEO}	-	10	nAdc
Base Cutoff Current ($V_{CE} = 60\text{ Vdc}$, $V_{EB(off)} = 3.0\text{ Vdc}$)	I_{BEX}	-	20	nAdc
ON CHARACTERISTICS				
DC Current Gain ($I_C = 0.1\text{ mAdc}$, $V_{CE} = 10\text{ Vdc}$) ($I_C = 1.0\text{ mAdc}$, $V_{CE} = 10\text{ Vdc}$) ($I_C = 10\text{ mAdc}$, $V_{CE} = 10\text{ Vdc}$) ($I_C = 10\text{ mAdc}$, $V_{CE} = 10\text{ Vdc}$, $T_A = -55^\circ\text{C}$) ($I_C = 150\text{ mAdc}$, $V_{CE} = 10\text{ Vdc}$) (Note 1) ($I_C = 150\text{ mAdc}$, $V_{CE} = 1.0\text{ Vdc}$) (Note 1) ($I_C = 500\text{ mAdc}$, $V_{CE} = 10\text{ Vdc}$) (Note 1)	h_{FE}	35 50 75 35 100 50 40	- - - - 300 - -	-
Collector - Emitter Saturation Voltage (Note 1) ($I_C = 150\text{ mAdc}$, $I_B = 15\text{ mAdc}$) ($I_C = 500\text{ mAdc}$, $I_B = 50\text{ mAdc}$)	$V_{CE(sat)}$	-	0.3 1.0	Vdc
Base - Emitter Saturation Voltage (Note 1) ($I_C = 150\text{ mAdc}$, $I_B = 15\text{ mAdc}$) ($I_C = 500\text{ mAdc}$, $I_B = 50\text{ mAdc}$)	$V_{BE(sat)}$	0.6 -	1.2 2.0	Vdc
SMALL-SIGNAL CHARACTERISTICS				
Current - Gain - Bandwidth Product (Note 2) ($I_C = 20\text{ mAdc}$, $V_{CE} = 20\text{ Vdc}$, $f = 100\text{ MHz}$)C	f_T	300	-	MHz
Output Capacitance ($V_{CB} = 10\text{ Vdc}$, $I_E = 0$, $f = 1.0\text{ MHz}$)	C_{obo}	-	8.0	pF
Input Capacitance ($V_{EB} = 0.5\text{ Vdc}$, $I_C = 0$, $f = 1.0\text{ MHz}$)	C_{ibo}	-	25	pF
Input Impedance ($I_C = 1.0\text{ mAdc}$, $V_{CE} = 10\text{ Vdc}$, $f = 1.0\text{ kHz}$) ($I_C = 10\text{ mAdc}$, $V_{CE} = 10\text{ Vdc}$, $f = 1.0\text{ kHz}$)	h_{ie}	2.0 0.25	8.0 1.25	$k\Omega$
Voltage Feedback Ratio ($I_C = 1.0\text{ mAdc}$, $V_{CE} = 10\text{ Vdc}$, $f = 1.0\text{ kHz}$) ($I_C = 10\text{ mAdc}$, $V_{CE} = 10\text{ Vdc}$, $f = 1.0\text{ kHz}$)	h_{re}	-	8.0 4.0	$\times 10^{-4}$
Small-Signal Current Gain ($I_C = 1.0\text{ mAdc}$, $V_{CE} = 10\text{ Vdc}$, $f = 1.0\text{ kHz}$) ($I_C = 10\text{ mAdc}$, $V_{CE} = 10\text{ Vdc}$, $f = 1.0\text{ kHz}$)	h_{fe}	50 75	300 375	-
Output Admittance ($I_C = 1.0\text{ mAdc}$, $V_{CE} = 10\text{ Vdc}$, $f = 1.0\text{ kHz}$) ($I_C = 10\text{ mAdc}$, $V_{CE} = 10\text{ Vdc}$, $f = 1.0\text{ kHz}$)	h_{oe}	5.0 25	35 200	μMhos
Collector Base Time Constant ($I_E = 20\text{ mAdc}$, $V_{CB} = 20\text{ Vdc}$, $f = 31.8\text{ MHz}$)	$rb'C_c$	-	150	ps
Noise Figure ($I_C = 100\text{ }\mu\text{Adc}$, $V_{CE} = 10\text{ Vdc}$, $R_S = 1.0\text{ k}\Omega$, $f = 1.0\text{ kHz}$)	N_F	-	4.0	dB

1. Pulse Test: Pulse Width $\leq 300\text{ }\mu\text{s}$, Duty Cycle $\leq 2.0\%$.2. f_T is defined as the frequency at which $|h_{fe}|$ extrapolates to unity.

P2N2222A

ELECTRICAL CHARACTERISTICS ($T_A = 25^\circ\text{C}$ unless otherwise noted) (Continued)

Characteristic	Symbol	Min	Max	Unit
SWITCHING CHARACTERISTICS				
Delay Time	t_d	-	10	ns
Rise Time	t_r	-	25	ns
Storage Time	t_s	-	225	ns
Fall Time	t_f	-	60	ns

SWITCHING TIME EQUIVALENT TEST CIRCUITS

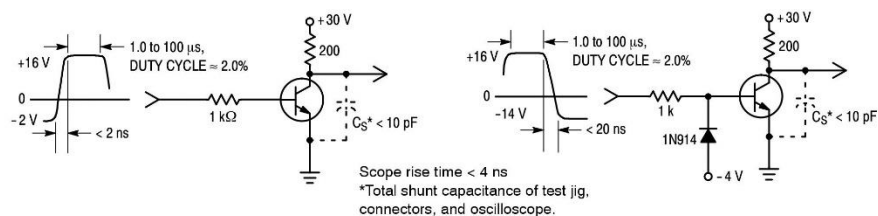


Figure 1. Turn-On Time

Figure 2. Turn-Off Time

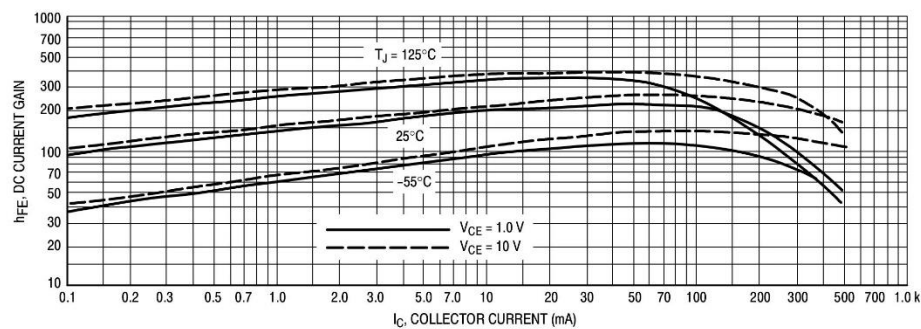


Figure 3. DC Current Gain

P2N2222A

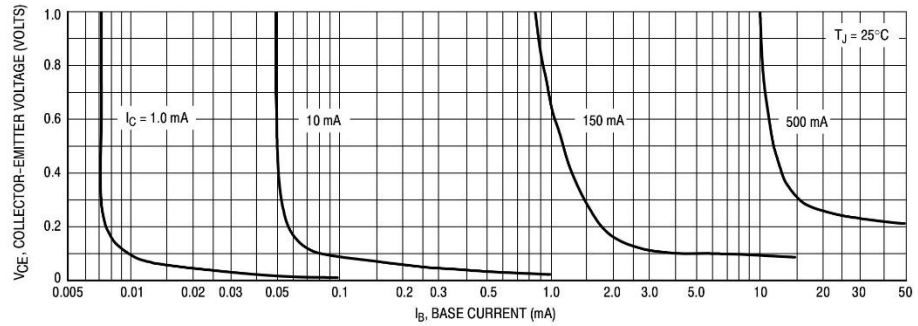


Figure 4. Collector Saturation Region

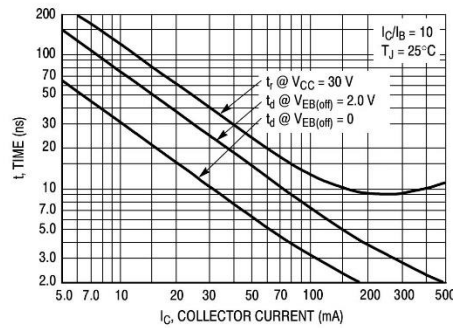


Figure 5. Turn-On Time

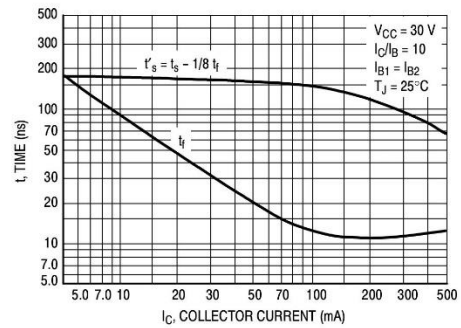


Figure 6. Turn-Off Time

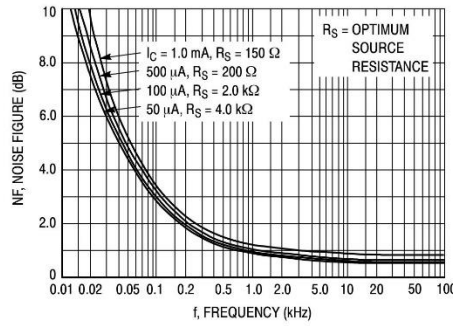


Figure 7. Frequency Effects

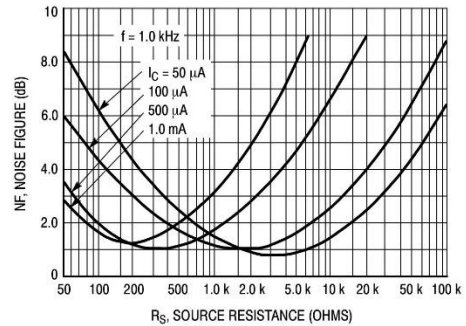


Figure 8. Source Resistance Effects

P2N2222A

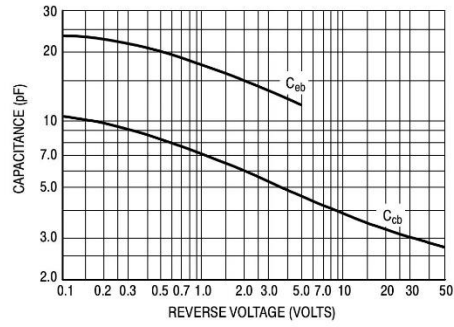


Figure 9. Capacitances

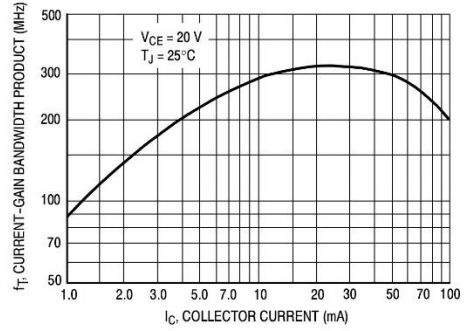


Figure 10. Current-Gain Bandwidth Product

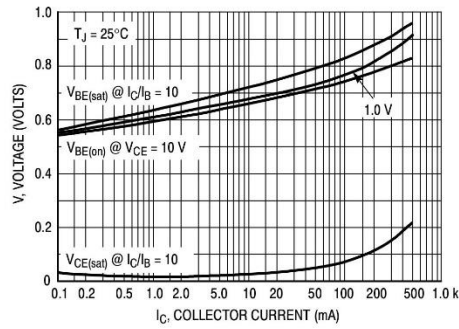


Figure 11. "On" Voltages

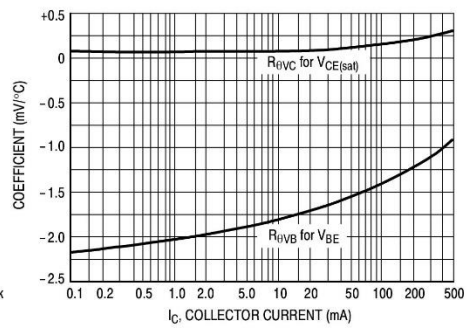
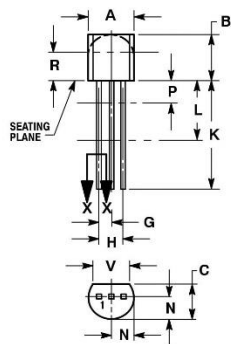


Figure 12. Temperature Coefficients

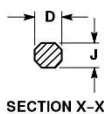
P2N2222A

PACKAGE DIMENSIONS

**TO-92 (TO-226)
CASE 29-11
ISSUE AM**

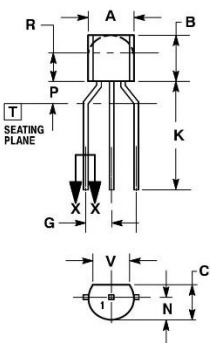


**STRAIGHT LEAD
BULK PACK**

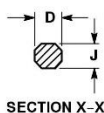


- NOTES:
1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
 2. CONTROLLING DIMENSION: INCH.
 3. CONTOUR OF PACKAGE BEYOND DIMENSION R IS UNCONTROLLED.
 4. LEAD DIMENSION IS UNCONTROLLED IN P AND BEYOND DIMENSION K MINIMUM.

DIM	INCHES		MILLIMETERS	
	MIN	MAX	MIN	MAX
A	0.175	0.206	4.45	5.20
B	0.170	0.210	4.32	5.33
C	0.125	0.165	3.18	4.19
D	0.016	0.021	0.407	0.533
G	0.045	0.055	1.15	1.39
H	0.095	0.105	2.42	2.66
J	0.015	0.020	0.39	0.50
K	0.500	---	12.70	---
L	0.250	---	6.35	---
N	0.080	0.105	2.04	2.66
P	---	0.100	---	2.54
R	0.115	---	2.93	---
V	0.135	---	3.43	---



**BENT LEAD
TAPE & REEL
AMMO PACK**



- NOTES:
1. DIMENSIONING AND TOLERANCING PER ASME Y14.5M, 1994.
 2. CONTROLLING DIMENSION: MILLIMETERS.
 3. CONTOUR OF PACKAGE BEYOND DIMENSION R IS UNCONTROLLED.
 4. LEAD DIMENSION IS UNCONTROLLED IN P AND BEYOND DIMENSION K MINIMUM.

DIM	MILLIMETERS	
	MIN	MAX
A	4.45	5.20
B	4.32	5.33
C	3.18	4.19
D	0.40	0.54
G	2.40	2.80
J	0.39	0.50
K	12.70	---
L	2.04	2.66
N	1.50	4.00
P	---	2.54
R	2.93	---
V	3.43	---

STYLE 17:
PIN 1: COLLECTOR
2: BASE
3: EMITTER

ON Semiconductor and are registered trademarks of Semiconductor Components Industries, LLC (SCILLC). SCILLC owns the rights to a number of patents, trademarks, copyrights, trade secrets, and other intellectual property. A listing of SCILLC's product/patent coverage may be accessed at www.onsemi.com/site/pdf/Patent-Marketing.pdf. SCILLC reserves the right to make changes without further notice to any products herein. SCILLC makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does SCILLC assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation special, consequential or incidental damages. "Typical" parameters which may be provided in SCILLC data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including "Typicals" must be validated for each customer application by customer's technical experts. SCILLC does not convey any license under its patent rights nor the rights of others. SCILLC products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the SCILLC product could create a situation where personal injury or death may occur. Should Buyer purchase or use SCILLC products for any such unintended or unauthorized application, Buyer shall indemnify and hold SCILLC and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that SCILLC was negligent regarding the design or manufacture of the part. SCILLC is an Equal Opportunity/Affirmative Action Employer. This literature is subject to all applicable copyright laws and is not for resale in any manner.

PUBLICATION ORDERING INFORMATION

LITERATURE FULFILLMENT:
Literature Distribution Center for ON Semiconductor
P.O. Box 5168, Denver, Colorado 80217 USA
Phone: 303-675-2175 or 800-344-3860 Toll Free USA/Canada
Fax: 303-675-2176 or 800-344-3867 Toll Free USA/Canada
Email: orderlit@onsemi.com

N. American Technical Support: 800-282-9855 Toll Free USA/Canada
Europe, Middle East and Africa Technical Support: Phone: 421 33 793 2910
Japan Customer Focus Center
Phone: 81-3-5817-1050

ON Semiconductor Website: www.onsemi.com
Order Literature: <http://www.onsemi.com/orderlit>
For additional information, please contact your local Sales Representative

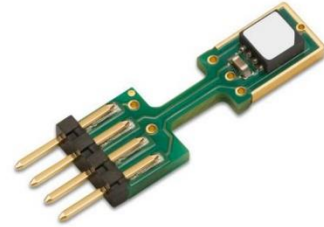
E. Hoja de datos sensor de temperatura y humedad relativa SHT85

SENSIRION
THE SENSOR COMPANY

Datasheet SHT85

Humidity and Temperature Sensor

- High-accuracy RH&T sensor for demanding measurement & test applications
- Typical accuracy of $\pm 1.5\% \text{RH}$ and $\pm 0.1\text{ }^\circ\text{C}$
- Pin-type packaging for easy integration and replacement
- Fully calibrated, linearized, and temperature compensated digital output



Product Summary

SHT85 is Sensirion's best-in-class humidity sensor with pin-type connector for easy integration and replacement. It builds on a highly accurate and long-term stable SHT3x sensor that is at the heart of Sensirion's new humidity and temperature platform. The unique package design allows for the best possible thermal coupling to the environment and decoupling from potential heat sources on the main board. The SHT85 features a PTFE membrane dedicated to protect the sensor opening from liquids and dust according to IP67, without affecting the response time of the RH signal. It thus allows for sensor use under harsh environmental conditions, (such as spray water and high exposure to dust). Final accuracy testing on product level ensures best performance, making the SHT85 the ultimate choice for even the most demanding applications.

Benefits of Sensirion's CMOSens® Technology

- High reliability and long-term stability
- Industry-proven technology with a track record of more than 10 years
- Designed for mass production
- Optimized for lowest cost
- Low signal noise

Content

1 Humidity and Temperature Sensor Specifications	3
2 Electrical Specifications	6
3 Pin Assignment	8
4 Operation and Communication	9
5 Packaging	18
6 Shipping Package	20
7 Quality	21
8 Ordering Information	22
9 Further Information	23
10 Important Notices	24
11 Revision History	25
12 Headquarters and Subsidiaries	25

Preliminary datasheet

1 Humidity and Temperature Sensor Specifications

Relative Humidity

Parameter	Conditions	Value	Units
Accuracy tolerance ¹	Typ.	±1.5	%RH
	Max.	see Figure 1	-
Repeatability ²	Low, typ.	0.21	%RH
	Medium, typ.	0.15	%RH
	High, typ.	0.08	%RH
Resolution	Typ.	0.01	%RH
Hysteresis	At 25°C	±0.8	%RH
Specified range ³	Non-condensing environment ⁴	0 to 100	%RH
Response time ⁵	τ 63%	8 ⁶	s
Long-term drift ⁷	Typ.	<0.25	%RH/y

Table 1: Humidity sensor specifications

Temperature

Parameter	Conditions	Value	Units
Accuracy tolerance ¹	Typ., 20°C to 50 °C	±0.1	°C
	Max.	see Figure 2	-
Repeatability ²	Low, typ.	0.15	°C
	Medium, typ.	0.08	°C
	High, typ.	0.04	°C
Resolution	Typ.	0.01	°C
Operating range	-	-40 to 105 ⁸	°C
Response time ⁹	τ 63%	2	s
Long-term drift	Typ.	<0.03	°C/y

Table 2: Temperature sensor specifications

¹ For definition of typ. and max. accuracy tolerance, please refer to the document "Sensirion Humidity Sensor Specification Statement".

² The stated repeatability is 3 times the standard deviation (3σ) of multiple consecutive measurement values at constant conditions and is a measure for the noise on the physical sensor output.

³ Specified range refers to the range for which the humidity sensor specification is guaranteed.

⁴ Condensation shall be avoided because of risk of corrosion and leak currents on the PCB. For details about recommended humidity and temperature operating range, please refer to Section 1.2.

⁵ Time for achieving 63% of a humidity step function, valid at 25°C and 1 m/s airflow. Humidity response time in the application depends on the design-in of the sensor.

⁶ With activated ART function (see Section 4.7) the response time can be improved by a factor of 2.

⁷ Typical value for operation in normal RH/T operating range. Max. value is < 0.5 %RH/y. Value may be higher in environments with vaporized solvents, out-gassing tapes, adhesives, packaging materials, etc. For more details please refer to Handling Instructions.

⁸ All parts, incl. PCB are rated up to 125°C, except for the connector, which is rated for 105°C.

⁹ Temperature response time depends on heat conductivity of sensor substrate and design-in of sensor in application.

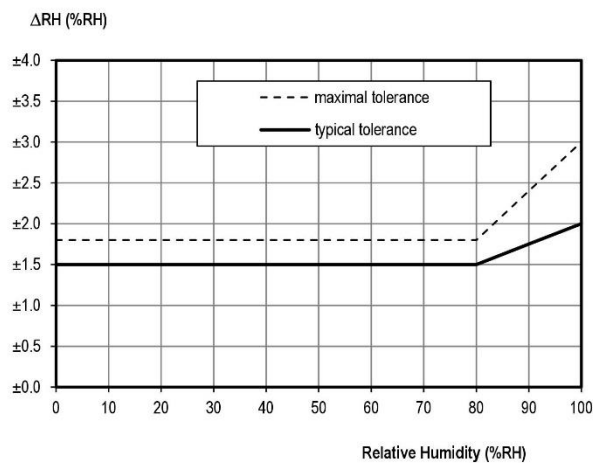


Figure 1: Typical and maximal tolerance for relative humidity in %RH at 25 °C

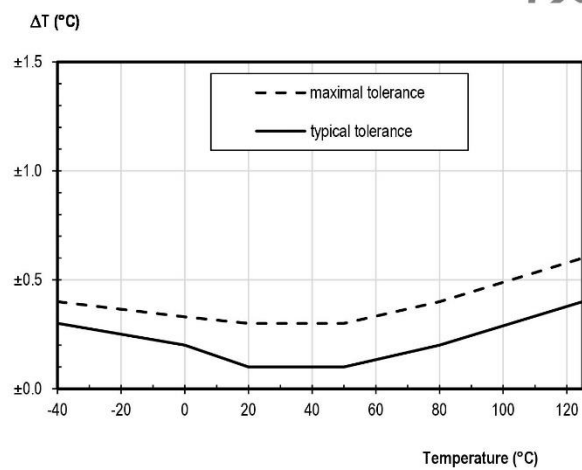


Figure 2: Typical and maximal tolerance for temperature sensor in °C

1.1 RH Accuracy at Various Temperatures

Typical RH accuracy at 25°C is defined in Figure 2. For other temperatures, typical accuracy has been evaluated to be as displayed in Figure 4.

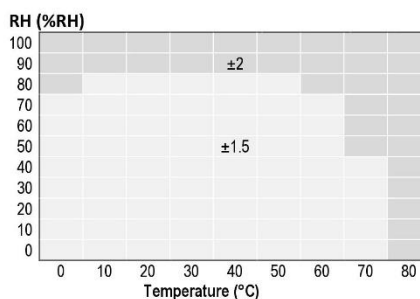


Figure 3: Typical accuracy of relative humidity measurements given in %RH for temperatures 0 – 80°C.

1.2 Recommended Operating Conditions

The sensor shows best performance when operated within recommended normal temperature and humidity range of 5 – 60 °C and 20 – 80 %RH, respectively. Long term exposure to conditions outside normal range, especially at high humidity, may temporarily offset the RH signal (e.g. +3%RH after 60h at >80%RH). After returning into the normal temperature and humidity range, the sensor will slowly come back to calibration state by itself. Prolonged exposure to extreme conditions may accelerate ageing.

To ensure stable operation of the humidity sensor, the conditions described in the document "SHTxx Assembly of SMD Packages", Section "Storage and Handling Instructions" regarding exposure to volatile organic compounds have to be met. Please note as well that this does apply not only to transportation and manufacturing, but also to operation of the SHT85.

Preliminary datasheet

2 Electrical Specifications

2.1 Electrical Characteristics

Parameter	Symbol	Conditions	Min	Typ.	Max	Units	Comments
Supply voltage	V_{DD}		2.15	3.3	5.5	V	-
Power-up/down level	V_{POR}		1.8	2.1	2.15	V	-
Slew rate change of the supply voltage	$V_{DD,slw}$		-	-	20	V/ms	Voltage changes on the VDD line between $V_{DD,min}$ and $V_{DD,max}$ should be slower than the maximum slew rate; faster slew rates may lead to reset;
Supply current	I_{DD}	Idle state (single shot mode) T= 25°C	-	0.2	12.0	μA	Current when sensor is not performing a measurement during single shot mode
		Idle state (single shot mode) T= 125°C	-	-	6.0		
		Idle state (periodic data acquisition mode)	-	45	-	μA	Current when sensor is not performing a measurement during periodic data acquisition mode
		Measurement	-	600	1500	μA	Average current consumption while sensor is measuring ¹⁰
		Average	-	1.7	-	μA	Average current consumption (continuous operation with one measurement per second) ¹⁰
Heater Power	P_{Heater}	Heater running	3.6	-	33	mW	Depending on the supply voltage

Table 3: Electrical specifications, typical values are valid for T=25°C, min. & max. values for T=-40°C ... 125°C.

2.2 Timing Specifications

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Units	Comments
Power-up time	t_{PU}	After hard reset, $V_{DD} \geq V_{POR}$	-	0.5	1.5	ms	Time between V_{DD} reaching V_{PU} and sensor entering idle state
Soft reset time	t_{SR}	After soft reset.	-	0.5	1.5	ms	Time between ACK of soft reset command and sensor entering idle state
Measurement duration	$t_{MEAS,l}$	Low repeatability	-	2.5	4.5	ms	The three repeatability modes differ with respect to measurement duration, noise level and energy consumption.
	$t_{MEAS,m}$	Medium repeatability	-	4.5	6.5	ms	
	$t_{MEAS,h}$	High repeatability	-	12.5	15.5	ms	

Table 4: System timing specifications, valid from -40 °C to 125 °C and VDDmin to VDDmax.

¹⁰ These values can be reduced by using the low power measurement mode, see separate application note.

2.3 Absolute Minimum and Maximum Ratings

Stress levels beyond those listed in Table 5 may cause permanent damage to the device. These are stress ratings only and functional operation of the device at these conditions cannot be guaranteed. Exposure to the absolute maximum rating conditions for extended periods may affect the reliability of the device. Ratings are only tested each at a time.

Parameter	Rating
Supply voltage, V_{DD}	-0.3 to 6 V
Max voltage on pins (pin 1 (SCL); pin 4 (SDA);	-0.3 to $V_{DD}+0$. V
Input current on any pin	± 100 mA
Operating temperature range	-40 to 105 °C
Storage temperature range ¹¹	-40 to 105 °C
ESD HBM (human body model) ¹²	4 kV
ESD CDM (charge device model) ¹³	750 V

Table 5: Absolute maximum ratings.

Preliminary datasheet

¹¹ The recommended storage temperature range is 10-50°C. Please consult the document "SHTxx Handling Instructions" for more information.

¹² According to ANSI/ESDA/JEDEC JS-001-2014; AEC-Q100-002.

¹³ According to ANSI/ESD S5.3.1-2009; AEC-Q100-011.

3 Pin Assignment

The SHT85 comes with a 4-pin-type connector, see Table 6.

Pin	Name	Comments
1	SCL	Serial clock; input only
2	VDD	Supply voltage; input
3	VSS	Ground
4	SDA	Serial data; input / output

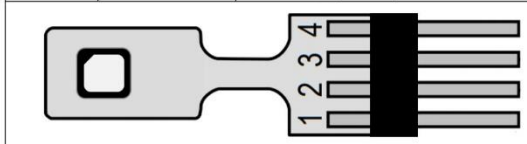


Table 6: SHT85 pin assignment (transparent top view). The die pad is internally connected to VSS.

3.1 Power Pins (VDD, VSS)

The electrical specifications of the SHT85 are shown in Table 3. Decoupling of VDD and VSS by a 100nF capacitor is integrated on the front side of the sensor packaging. See Figure 4 for a typical application circuit.

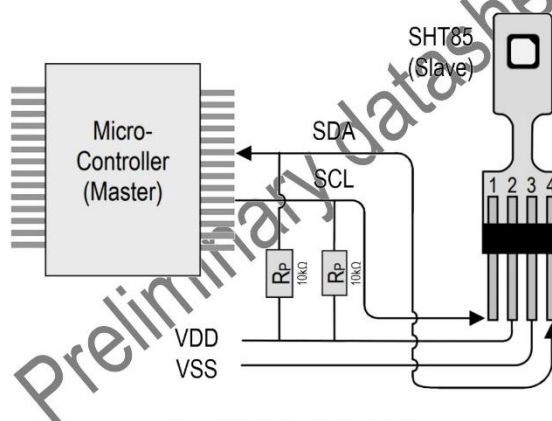


Figure 4: Typical application circuit

3.2 Serial Clock and Serial Data (SCL, SDA)

SCL is used to synchronize the communication between microcontroller and the sensor. The clock frequency can be freely chosen between 0 to 1000 kHz.

The SDA pin is used to transfer data to and from the sensor. Communication with frequencies up to 400 kHz must meet the I2C *Fast Mode*¹⁴ standard. Communication frequencies up to 1 Mhz are supported following the specifications given in Table 19.

¹⁴ http://www.nxp.com/documents/user_manual/UM10204.pdf

4 Operation and Communication

The SHT85 supports I2C fast mode (and frequencies up to 1000 kHz). For detailed information on the I2C protocol, refer to NXP I2C-bus specification¹⁵.

After sending a command to the sensor a minimal waiting time of 1ms is needed before another command can be received by the sensor.

Furthermore, to keep self-heating below 0.1°C, SHT85 should not be active for more than 10% of the time.

All SHT85 commands and data are mapped to a 16-bit address space. Additionally, data and commands are protected with a CRC checksum. This increases communication reliability. The 16 bits commands to the sensor already include a 3 bit CRC checksum. Data sent from and received by the sensor is always succeeded by an 8 bit CRC.

In write direction it is mandatory to transmit the checksum, since the SHT85 only accepts data if it is followed by the correct checksum. In read direction it is left to the master to read and process the checksum.

4.1 Power-Up and Communication Start

The sensor starts powering-up after reaching the power-up threshold voltage V_{POR} specified in Table 3. After reaching this threshold voltage the sensor needs the time t_{PU} to enter idle state. Once the idle state is entered it is ready to receive commands from the master (microcontroller).

Each transmission sequence begins with a START condition (S) and ends with a STOP condition (P) as described in the I2C-bus specification. Whenever the sensor is powered up, but not performing a measurement or communicating, it automatically enters idle state for energy saving. This idle state cannot be controlled by the user.

4.2 Starting a Measurement

A measurement communication sequence consists of a START condition, the I2C write header (7-bit I2C device address plus 0 as the write bit) and a 16-bit measurement command. The proper reception of each byte is indicated by the sensor. It pulls the SDA pin low (ACK bit) after the falling edge of the 8th SCL clock to indicate the reception. A complete measurement cycle is depicted in Table 7.

With the acknowledgement of the measurement command, the SHT85 starts measuring humidity and temperature.

4.3 Measurement Commands for Single Shot Data Acquisition Mode

In this mode one issued measurement command triggers the acquisition of *one data pair*. Each data pair consists of one 16-bit temperature and one 16-bit humidity value (in this order). During transmission each data value is always followed by a CRC checksum, see Section 4.1.

In single shot mode different measurement commands can be selected. The 16-bit commands are shown in Table 7. They differ with respect to repeatability (low, medium and high).

The repeatability setting influences the measurement duration and thus the overall energy consumption of the sensor. This is explained in Section 2.

¹⁵ http://www.nxp.com/documents/user_manual/UM10204.pdf

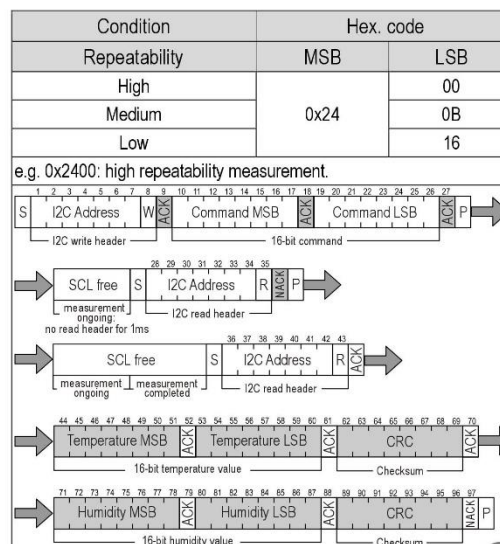


Table 7: Measurement commands in single shot mode. The first "SCL free" block indicates a minimal waiting time of 1ms. (Clear blocks are controlled by the microcontroller, grey blocks by the sensor).

4.4 Readout of Measurement Results for Single Shot Mode

After the sensor has completed the measurement, the master can read the measurement results (pair of RH & T) by sending a START condition followed by an I2C read header.

The sensor responds to a read header with a not acknowledge (NACK), if the measurement is still ongoing and thus no data is present.

If the measurement is completed, the sensor will acknowledge the reception of the read header and send two bytes of data (temperature) followed by one byte CRC checksum and another two bytes of data (relative humidity) followed by one byte CRC checksum. Each byte must be acknowledged by the microcontroller with an ACK condition for the sensor to continue sending data. If the sensor does not receive an ACK from the master after any byte of data, it will not continue sending data.

The sensor will send the temperature value first and then the relative humidity value. After having received the checksum for the humidity value a NACK and stop condition should be sent (see Table 7).

The I2C master can abort the read transfer with a NACK condition after any data byte if it is not interested in subsequent data, e.g. the CRC byte or the second measurement result, in order to save time.

In case the user needs humidity and temperature data but does not want to process CRC data, it is recommended to read the two temperature bytes of data with the CRC byte (without processing the CRC data); after having read the two humidity bytes, the read transfer can be aborted with a with a NACK.

4.5 Measurement Commands for Periodic Data Acquisition Mode

In this mode one issued measurement command yields a *stream of data pairs*. Each data pair consists of one 16-bit temperature and one 16-bit humidity value (in this order).

In periodic mode different measurement commands can be selected. The corresponding 16-bit commands are shown in Table 8. They differ with respect to repeatability (low, medium and high) and data acquisition frequency (0.5, 1, 2, 4 & 10 measurements per second, mps).

The data acquisition frequency and the repeatability setting influences the measurement duration and the current consumption of the sensor. This is explained in Section 2 of this datasheet.

If a measurement command is issued, while the sensor is busy with a measurement (measurement durations see Table 4), it is recommended to issue a break command first (see Section 4.8). Upon reception of the break command the sensor will abort the ongoing measurement and enter the single shot mode.

Condition		Hex. code	
Repeatability	mps	MSB	LSB
High	0.5	0x20	32
Medium			24
Low			2F
High	1	0x21	30
Medium			26
Low			2D
High	2	0x22	36
Medium			20
Low			2B
High	4	0x23	34
Medium			22
Low			29
High	10	0x27	37
Medium			21
Low			2A

e.g. 0x2130: 1 high repeatability mps – measurement per second

Table 8: Measurement commands for periodic data acquisition mode (Clear blocks are controlled by the microcontroller, grey blocks by the sensor). N.B.: At the highest mps setting self-heating of the sensor might occur.

4.6 Readout of Measurement Results for Periodic Mode

Transmission of the measurement data can be initiated through the fetch data command shown in Table 9. If no measurement data is present the I2C read header is responded with a NACK (Bit 9 in Table 9) and the communication stops. After the read out command fetch data has been issued, the data memory is cleared, i.e. no measurement data is present.

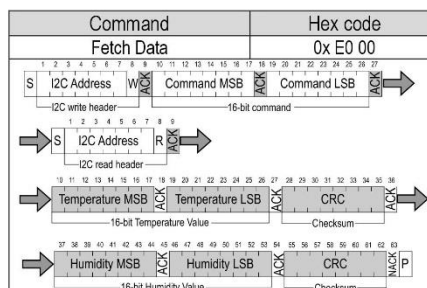


Table 9: Fetch Data command (Clear blocks are controlled by the microcontroller, grey blocks by the sensor).

4.7 ART Command

The ART (accelerated response time) feature can be activated by issuing the command in Table 10. After issuing the ART command the sensor will start acquiring data with a frequency of 4Hz.

The ART command is structurally similar to any other command in Table 8. Hence Section 4.5 applies for starting a measurement, Section 4.6 for reading out data and Section 4.8 for stopping the periodic data acquisition.

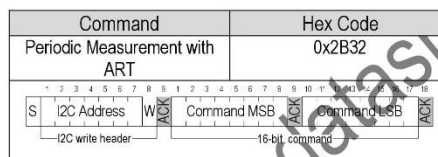


Table 10: Command for a periodic data acquisition with the ART feature (Clear blocks are controlled by the microcontroller, grey blocks by the sensor).

4.8 Break command / Stop Periodic Data Acquisition Mode

The periodic data acquisition mode can be stopped using the break command shown in Table 11. It is recommended to stop the periodic data acquisition prior to sending another command (except Fetch Data command) using the break command. Upon reception of the break command the sensor will abort the ongoing measurement and enter the single shot mode. This takes 1ms.

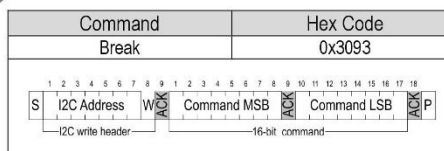


Table 11: Break command (Clear blocks are controlled by the microcontroller, grey blocks by the sensor).

4.9 Reset

A system reset of the SHT85 can be generated externally by issuing a command (soft reset). Additionally, a system reset is generated internally during power-up. During the reset procedure the sensor will not process commands.

Interface Reset

If communication with the device is lost, the following signal sequence will reset the serial interface: While leaving SDA high, toggle SCL nine or more times. This must be followed by a Transmission Start sequence preceding the next command. This sequence resets the interface only. The status register preserves its content.

Soft Reset / Re-Initialization

The SHT85 provides a soft reset mechanism that forces the system into a well-defined state without removing the power supply. When the system is in idle state the soft reset command can be sent to the SHT85. This triggers the sensor to reset its system controller and reloads calibration data from the memory. In order to start the soft reset procedure the command as shown in Table 12 should be sent.

It is worth noting that the sensor reloads calibration data prior to every measurement by default.

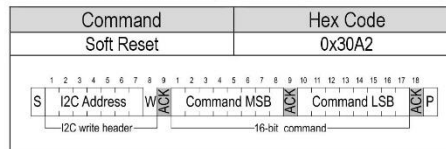


Table 12: Soft reset command (Clear blocks are controlled by the microcontroller, grey blocks by the sensor).

Reset through General Call

Additionally, a reset of the sensor can also be generated using the "general call" mode according to I2C-bus specification¹⁵. It is important to understand that a reset generated in this way is not device specific. All devices on the same I2C bus that support the general call mode will perform a reset. Additionally, this command only works when the sensor is able to process I2C commands. The appropriate command consists of two bytes and is shown in Table 13.

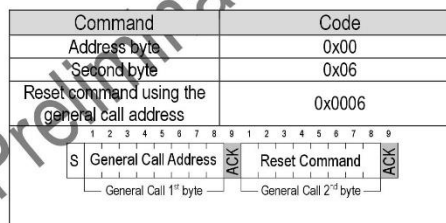


Table 13: Reset through the general call address (Clear blocks are controlled by the microcontroller, grey blocks by the sensor).

Hard Reset

A hard reset is achieved by switching the supply voltage to the VDD Pin off and then on again. In order to prevent powering the sensor over the ESD diodes, the voltage to pins 1 (SCL) and 4 (SDA) also needs to be removed.

4.10 Heater

The SHT85 is equipped with an internal heater, which is meant for plausibility checking only. The temperature increase achieved by the heater depends on various parameters and lies in the range of a few degrees centigrade. It can be switched on and off by command, see table below. The status is listed in the status register. After a reset the heater is disabled (default condition).

Command	Hex Code	
	MSB	LSB
Heater Enable	0x30	6D
Heater Disabled		66

Table 14: Heater command (Clear blocks are controlled by the microcontroller, grey blocks by the sensor).

4.11 Status Register

The status register contains information on the operational status of the heater, the alert mode and on the execution status of the last command and the last write sequence. The command to read out the status register is shown in Table 15 whereas a description of the content can be found in Table 16.

Command	Hex code
Read Out of status register	0x32D

Table 15: Command to read out the status register (Clear blocks are controlled by the microcontroller, grey blocks by the sensor).

Bit	Field description	Default value
15	Alert pending status '0': no pending alerts '1': at least one pending alert	'1'
14	Reserved	'0'
13	Heater status '0': Heater OFF '1': Heater ON	'0'
12	Reserved	'0'
11	RH tracking alert '0': no alert '1': alert	'0'
10	T tracking alert '0': no alert '1': alert	'0'
9:5	Reserved	'xxxxx'
4	System reset detected '0': no reset detected since last 'clear status register' command '1': reset detected (hard reset, soft reset command or supply fail)	'1'
3:2	Reserved	'00'
1	Command status '0': last command executed successfully '1': last command not processed. It was either invalid, failed the integrated command checksum	'0'
0	Write data checksum status '0': checksum of last write transfer was correct '1': checksum of last write transfer failed	'0'

Table 16: Description of the status register.

Clear Status Register

All flags (Bit 15, 11, 10, 4) in the status register can be cleared (set to zero) by sending the command shown in Table 17.

Command	Hex Code
Clear status register	0x 30 41

Table 17: Command to clear the status register (Clear blocks are controlled by the microcontroller, grey blocks by the sensor).

4.12 Checksum Calculation

The 8-bit CRC checksum transmitted after each data word is generated by a CRC algorithm. Its properties are displayed in Table 18. The CRC covers the contents of the two previously transmitted data bytes. To calculate the checksum only these two previously transmitted data bytes are used.

Property	Value
Name	CRC-8
Width	8 bit
Protected data	read and/or write data
Polynomial	0x31 ($x^8 + x^5 + x^4 + 1$)
Initialization	0xFF
Reflect input	False
Reflect output	False
Final XOR	0x00
Examples	CRC (0xBEEF) = 0x92

Table 18: I2C CRC properties.

4.13 Conversion of Signal Output

Measurement data is always transferred as 16-bit values (unsigned integer). These values are already linearized and compensated for temperature and supply voltage effects. Converting those raw values into a physical scale can be achieved using the following formulas.

Relative humidity conversion formula (result in %RH):

$$RH = 100 \cdot \frac{S_{RH}}{2^{16} - 1}$$

Temperature conversion formula (result in °C & °F):

$$T [^{\circ}\text{C}] = -45 + 175 \cdot \frac{S_T}{2^{16} - 1}$$

$$T [^{\circ}\text{F}] = -49 + 315 \cdot \frac{S_T}{2^{16} - 1}$$

S_{RH} and S_T denote the raw sensor output for humidity and temperature, respectively. The formulas work only correctly when S_{RH} and S_T are used in decimal representation.

4.14 Communication Timing

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Units	Comments
SCL clock frequency	f_{SCL}		0	-	1000	kHz	
Hold time (repeated) START condition	$t_{HD,STA}$	After this period, the first clock pulse is generated	0.24	-	-	μs	
LOW period of the SCL clock	t_{LOW}		0.53	-	-	μs	
HIGH period of the SCL clock	t_{HIGH}		0.26	-	-	μs	
SDA hold time	$t_{HD,DAT}$		0	-	250	ns	Transmitting data
			0	-	-	ns	Receiving data
SDA set-up time	$t_{SU,DAT}$		100	-	-	ns	
SCL/SDA rise time	t_R		-	-	300	ns	
SCL/SDA fall time	t_F		-	-	300	ns	
SDA valid time	$t_{VD,DAT}$		-	-	0.9	μs	
Set-up time for a repeated START condition	$t_{SU,STA}$		0.26	-	-	μs	
Set-up time for STOP condition	$t_{SU,STO}$		0.26	-	-	μs	
Capacitive load on bus line	CB		-	-	400	pF	
Low level input voltage	V_{IL}		0	-	$0.3 \times V_{DD}$	V	
High level input voltage	V_{IH}		$0.7 \times V_{DD}$	-	$1 \times V_{DD}$	V	
Low level output voltage	V_{OL}	33 mA sink current	-	-	0.4	V	

Table 19: Timing specifications for I2C communication, valid for $T = -40^{\circ}C$, $125^{\circ}C$ and $V_{DD} = V_{DDmin} \dots V_{DDmax}$. The nomenclature above is according to the I2C Specification (UM10204, Rev.6, April 4, 2014).

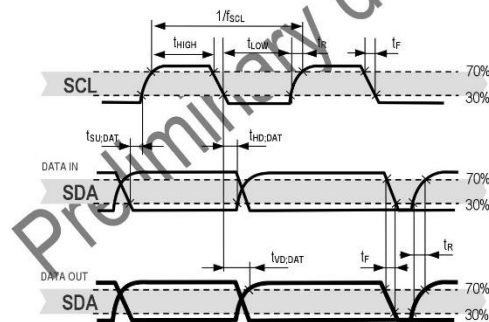


Figure 5: Timing diagram for digital input/output pads. SDA directions are seen from the sensor. Bold SDA lines are controlled by the sensor, plain SDA lines are controlled by the micro-controller. Note that SDA valid read time is triggered by falling edge of preceding toggle.

5 Packaging

The SHT85 is supplied in a single-in-line pin type package. The SHT35-DIS sensor housing consists of an epoxy-based mold compound, see "Datasheet SHT3x-DIS" for more information. The sensor opening of the housing is protected by a PTFE membrane dedicated to protect the sensor opening from liquids and dust according to IP67, see "Datasheet Membrane Option" for more information. The sensor head is connected to the pins by a small bridge to minimize heat conduction and response times. The pins are soldered to the FR4 substrate by lead-free solder paste. The gold plated backside of the sensor head is connected to the VSS pin. A 100nF capacitor is mounted on the front side between VDD and VSS. The device is fully RoHS compliant – thus it is free of Pb, Cd, Hg, Cr(6+), PBB and PBDE. All pins are Au plated to avoid corrosion. They can be soldered or mate with most 1.27 mm (0.05") sockets, for example: Preci-dip / Mill-Max R851-83-004-20-001 or similar. When the sensor is further processed by soldering, it should be ensured that the solder connections between pins and the SHT85 PCB are not melted.

5.1 Traceability

The SHT85 provides a device specific serial number, which can be read-out via the serial interface (I2C), see the command in Table 20. The Serial number allows an unambiguous identification of each individual device.

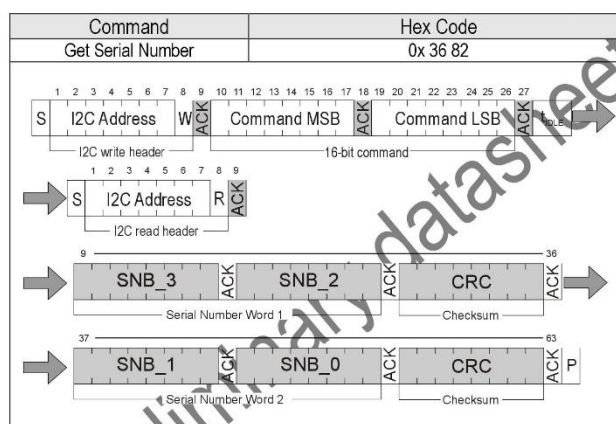


Table 20: Command to read out the Serial Number (Clear blocks are controlled by the microcontroller, grey blocks by the sensor.)

After issuing the measurement command and sending the ACK Bit the sensor needs the time $t_{IDLE} = 0.5\text{ms}$ to respond to the I2C read header with an ACK Bit. Hence it is recommended to wait $t_{IDLE} = 0.5\text{ms}$ before issuing the read header. The Get Serial Number command returns 2 words; every word is followed by a CRC Checksum. Together the 2 words (SNB_3 to SNB_0 in Table 20, SNB_0 is the LSB, whereas SNB_3 is the MSB) constitute a unique serial number with a length of 32 bit. This serial number can be used to identify each sensor individually.

5.2 Package Outline

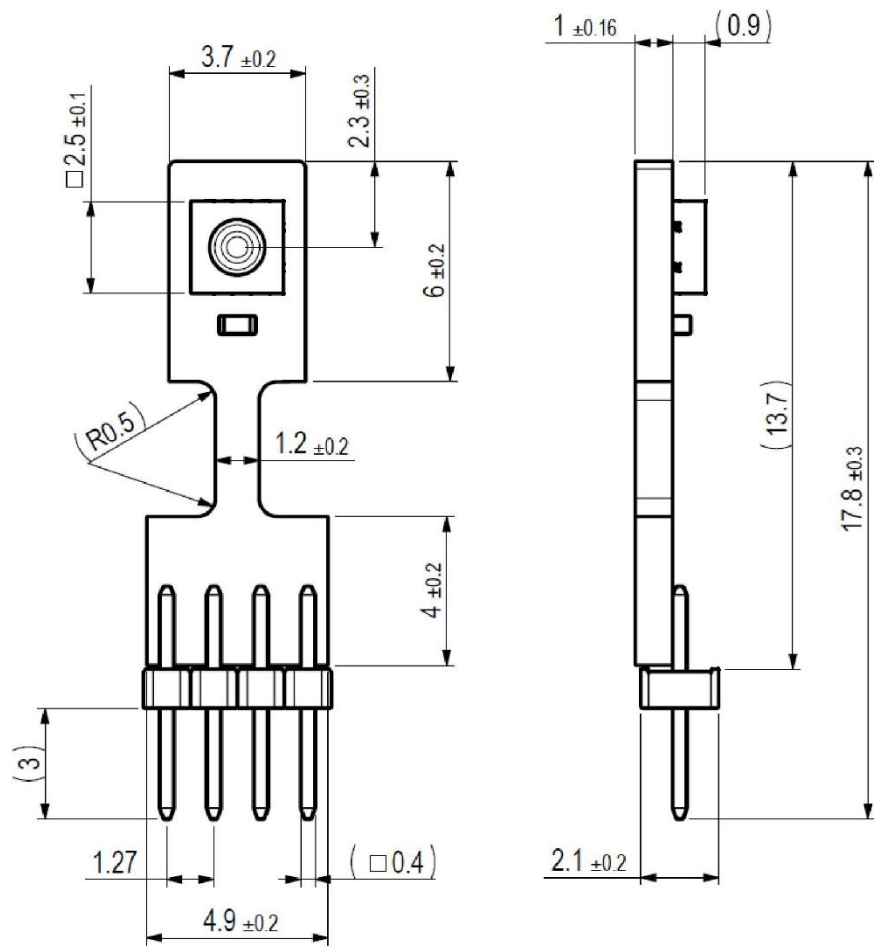


Figure 6: Dimensional drawing of the SHT85 sensor packaging. Dimensions are in mm (1mm = 0.039 inch).

F. Catalogo sensor de humedad del suelo VH400

VH400 Soil Moisture Sensor Probes

1 . Information



Our VH400 series soil moisture sensor probes enable precise low cost monitoring of soil water content.

Because our probe measures the dielectric constant of the soil using transmission line techniques, it is insensitive to water salinity, and will not corrode over time as does conductivity based probes. Our probes are small, rugged, and low power.

Compared to other low cost sensor such as gypsum block sensors, our probes offer a rapid response time. They can be inserted and take an accurate reading in under a second.

Probes come in standard cable lengths of 2 meters, 5 meters and 10 meters.

2 . Soil Moisture Sensor Probe Applications

- Irrigation and sprinkler systems.
- Moisture monitoring of bulk foods.
- Rain and weather monitoring.
- Environmental monitoring.
- Water conservation applications.

3 . Soil Moisture Sensor Probe Features

- Extreme low cost with volume pricing.
- Not conductivity based.
- Insensitive to salinity.
- Probe does not corrode over time.
- Rugged design for long term use.
- Small size.
- Consumes less than 7mA for very low power operation.
- Precise measurement.
- Measures volumetric water content (VWC) or gravimetric water content (GWC).
- Output Voltage is proportional to moisture level.
- Wide supply voltage range.
- Can be buried and is water proof.
- Probe is long and slender for wider use, including smaller potted plants.

4 . Soil Moisture Sensor Probe Ordering Info

ORDER INFO

Part Number	Description
VH400-2M	Soil Moisture Sensor - 2 meter cable
VH400-5M	Soil Moisture Sensor - 5 meter cable
VH400-10M	Soil Moisture Sensor - 10 meter cable

5 . Soil Moisture Sensor Relay Boards

The quickest way to evaluate if the VH400 series is right for you is to also order some of our sensor relay boards. The relay boards accept input from a single sensor and control a solid state or mechanical relay.

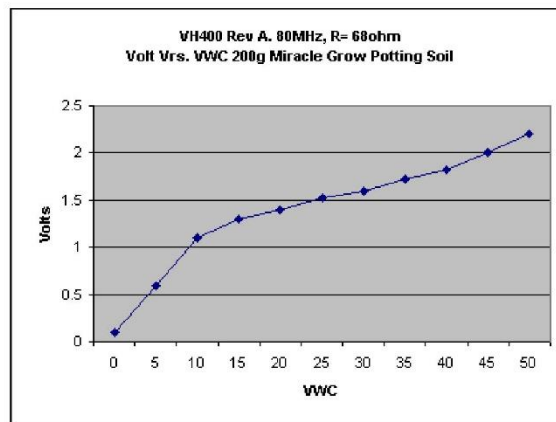
The relay contacts can be configured to close when wet, or open when wet. The dry/wet threshold is easily set by an on board variable resistor, which can be tuned with a small screwdriver. With our relay boards, within minutes you'll be using your VH400 soil moisture probe to control, valves, alarms or home automation systems.

6 . Soil Moisture Sensor Probe Specifications

Power consumption	< 13mA
Supply Voltage	3.5V to 20 VDC.
Dimensions	See drawing below.
Power on to Output stable	400 ms
Output Impedance	10K ohms
Operational Temperature	-40°C to 85°C
Accuracy	2% at 25°C
Output	0 to 3V related to moisture content
Shell Color	Red

6.1.) Figure 1: VH400 Rev A to Rev G: VWC to Voltage Curves

VWC to Voltage curve for container with 200g of Miracle Grow Potting Soil.



6.2) VH400 Piecewise Curve

Most curves can be approximated with linear segments of the form:

$$y = m \cdot x - b,$$

where m is the slope of the line

The VH400's Voltage to VWC curve can be approximated with 4 segments of the form:

$$VWC = m \cdot V - b$$

where V is voltage.

$$m = (VWC2 - VWC1) / (V2 - V1)$$

where V1 and V2 are voltages recorded at the respective VWC levels of VWC1 and VWC2.

After m is determined, the y-axis intercept coefficient b can be found by inserting one of the end points into the equation:

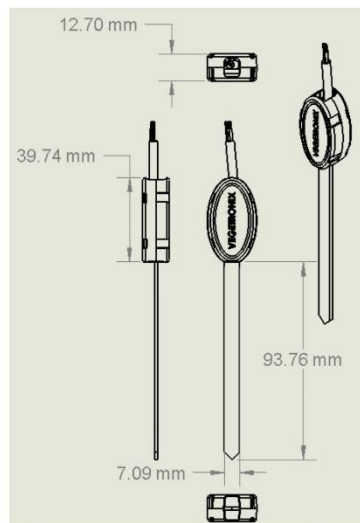
$$b = m \cdot v - VWC$$

Voltage Range	Equation
0 to 1.1V	VWC = 10 * V - 1
1.1V to 1.3V	VWC = 25 * V - 17.5
1.3V to 1.82V	VWC = 48.08 * V - 47.5
1.82V to 2.2V	VWC = 26.32 * V - 7.89

7. Soil Moisture Sensor Probe Wiring Table

Color	Description
Bare	Ground
Red	POWER: 3.5V to 20 VDC.
Black	OUT: (0 to 3V related to moisture content.)

8. VH400 Soil Moisture Sensor Probe Drawing



G. Presupuesto aproximado de costos de estructura, malla, sistema de riego y tecnología para un invernadero en la ciudad de Tacna



COTIZACION N°. CT1-162068

Pag. 1

SEÑORES: UNIVERSIDAD PRIVADA DE TACNA

FECHA: 02/10/2018

REF:

ATENCION:

Es grato el dirigiros a ustedes con la finalidad de alcanzarles nuestra cotización por lo siguiente:

CODIGO	DESCRIPCION	CANT.	UND.	V. VENTA S/	VALOR TOTAL S/
1	plantinera			0.00	0.00
2	208500104 ELECTROBOMBA CONFORTO 1 HP INOX 220V-60Hz // NPXM100GIR	1	u.	920.00	920.00
3	121200300 TERMINAL MIXTO MACHO PVC 32 X 40 X 1"	10	u.	2.00	20.00
4	106100800 CODO 90° PE 32 MM	2	u.	4.00	8.00
5	110200300 FILTRO 1" DISCO 130 MIC ROSCADO AZUD	1	u.	75.00	75.00
6	114000400 MANGUITO PVC MIXTO 32 X 1"	3	u.	5.00	15.00
7	1011094 VALVULA ENLACE R/HEMBRA 1"	4	u.	35.00	140.00
8	222500111 TUBO HID PVC 32 MM C-10	4	Tb.	48.00	192.00
9	109501100 ENLACE R/H PE 32 X 1	3	u.	3.00	9.00
10	206200111 CODO 90° PVC SOLDABLE 32MM	9	u.	4.00	36.00
11	221000111 TEE SOLDABLE 32MM	2	u.	9.00	18.00
12	120400800 TAPON FINAL PE 32 MM	2	u.	4.00	8.00
13	109502000 ENLACE R/M PE 20 X 3/4"	3	u.	3.00	9.00
14	118000900 REDUCCION MACHO/HEMBRA PE 1" X 3/4"	3	u.	2.00	6.00
15	213500209 MANGUERA CIEGA 20MM X 1.30 ESP X 300 MTS	0.50	rol	245.00	122.50
16	115100500 NEBULIZADOR GREEN MIST 30 LPH AZUL	100	u.	12.00	1,200.00
17	120400600 TAPON FINAL PE 20MM	3	u.	2.00	6.00
18	115700700 PEGAMENTO PVC 1/4 WELDALL	1	u.	33.00	33.00
19	205600201 CINTA TEFLON 19MM 50MTS	1	u.	20.00	20.00
20	tablero electrico de proteccion	1	u.	850.00	850.00
21	207800109 DEPOSITO P.E NEGRO X 1500 LTS.	1	u.	1,800.00	1,800.00
22	115500100 PASAMUROS PVC 1"	1	u.	95.00	95.00
23	114000400 MANGUITO PVC MIXTO 32 X 1"	3	u.	5.00	15.00

CONDICIONES DE VENTA: Forma de Pago : CONTADO
 Validez de la Oferta : 0 dias
 Lugar de Entrega :

* Los precios unitarios Incluyen IGV
 Valor Venta S/ 4,743.63
 IGV 18% S/ 853.85
 Precio de Venta S/ 5,597.48

CUENTA CORRIENTE

540-1787227-0-41 SOLES BCP - Tacna
 540-1773486-1-53 DOLARES BCP - Tacna

0011-0232-01-00072868 SOLES BBVA - Tacna
 0011-0232-01-00072876 DOLARES BBVA - Tacna

000-7683944 SOLES SCOTIABANK - TACNA
 000-3574714 DOLARES SCOTIABANK - TACNA

0151-072070 DETRACCIONES BCO. DE LA NACION

Sistemas de riego por goteo, reservorios - geomembranas, invernaderos - casas malla.

TACNA: Av. Ejercito 435, Para Grande Telef. 052-314300, Cel. 952644113, 952659512, Email: jpino@pinosac.com
 ICA: Panamericana Sur Km. 293, Salas Guadalupe. Telef. 056-406008, Cel. 970903305, 990809291, Email: rzegarrapino@pinosac.com
 LIMA: Cel. 958610979, Email: mpino@pinosac.com
 CHICLAYO: Cel. 990809291, Email: rzegarrapino@pinosac.com
 www.pinosac.com

	CRIADO Y LOPEZ PERU SAC	
	TLF: 052-314300 Tacna 056-720078 Ica	www.criadoylopez.com
	TÚNEL KIT IMATEC	
	DEPARTAMENTO COMERCIAL	

CLIENTE:	Universidad Privada de Tacna		
UBICACIÓN OBRA:	Fundo de la UPT		
TLF / E-mail:	electrónica.transmite@gmail.com	FECHA	04/10/18

TÚNEL KIT IMATEC



invernaderos • umbráculos • túneles • mesas • climatización • accesorios • invernaderos • umbráculos • túneles • mesas • climatización

	CRIADO Y LOPEZ PERU SAC	
	TLF: 052-314300 Tacna 056-720078 Ica	www.criadoylopez.com
	TÚNEL KIT IMATEC	
	DEPARTAMENTO COMERCIAL	PAG. 1

ÍNDICE


1. CARACTERÍSTICAS TÉCNICAS
2. ESTRUCTURA Y CUBIERTA
 - 2.1. ESTRUCTURA
 - 2.2. SISTEMA DE SUJECIÓN
 - 2.3. PUERTA Y VENTILACIÓN
 - 2.4. CIMENTACIÓN
 - 2.5. CUBIERTA
 - 2.6. CALIDAD DEL ACERO
 - 2.7. GALVANIZADO
3. PRESUPUESTO
 - 3.1. DESCRIPCIÓN DE LA ESTRUCTURA Y OFERTA ECONÓMICA
 - 3.2. CONCEPTOS NO INCLUIDOS EN LA OFERTA
 - 3.3. FORMA DE PAGO
 - 3.4. PLAZO DE ENTREGA
4. PLANOS
5. FOTOS

	CRIADO Y LOPEZ PERU SAC	
	TLF: 052-314300 Tacna 056-720078 Ica	www.criadoylopez.com
	TÚNEL KIT IMATEC	
	DEPARTAMENTO COMERCIAL	PAG. 2





1.- CARACTERÍSTICAS TÉCNICAS

2.- ESTRUCTURA Y CUBIERTA

2.1.- ESTRUCTURA


- 
Arcos de tubo de diámetro 40 mm y 1,5 mm de espesor, colocados cada 2 m. Anclados sobre piquetas clavadas directamente al suelo.





- 
Tirantes y pendolones de tubo redondo de 32 mm de diámetro y 1,2 mm de espesor. Se coloca un tirante y pendolón cada arco. El tirante es uno de los elementos más importantes en el correcto arriostramiento del invernadero
- 
 El tirante sirve como **soporte de cultivo**, para ello se ha dotado de un refuerzo vertical, el pendolón, que evita el desplazamiento del plano recto del tirante.
- 
 Altura a la cumbre 3 m
- 
 Altura al tirante 2 m.



2.2.- SISTEMA DE SUJECIÓN


- 
5 Correas longitudinales de tubo de 32 mm de diámetro y 1,2 mm de espesor. Permiten amarrar los arcos entre sí de manera que trabajan a tracción y contracción para la correcta sujeción de la estructura.

	CRIADO Y LOPEZ PERU SAC	
	TLF: 052-314300 Tacna 056-720078 Ica	www.criadoylopez.com
	TÚNEL KIT IMATEC	
	DEPARTAMENTO COMERCIAL	PAG. 3

-  **5 Refuerzos frontales** de arco a arco en cada extremo completan el arriostamiento del invernadero.
-  **Tornillería**, todas las uniones están previstas para facilitar el montaje y desmontaje de sus elementos con la mayor facilidad y fiabilidad. Tornillos y tuercas hexagonales DIN-933 y DIN-934 en calidad comercial 5.6, correspondiendo a las designaciones F-7417, según Norma UNE-36-089-72. Resistencia media a la tracción 55 Kg/mm².



2.3.- PUERTA Y VENTILACIÓN

-  **2 Puertas Correderas**, colocadas una en cada frontal de dimensiones 2x2 m. Disponen de un paño fijo que se puede cubrir con plástico o con malla para mejorar la ventilación.



	CRIADO Y LOPEZ PERU SAC	
	TLF: 052-314300 Tacna 056-720078 Ica	www.criadoylopez.com
	TÚNEL KIT IMATEC	
	DEPARTAMENTO COMERCIAL	PAG. 4

- **2 Ventanas frontales abatibles** de dimensiones 2 x 0,6 m. Colocadas en la media luna del invernadero. Incorporan pestillo para un cierre adecuado.

	CRIADO Y LOPEZ PERU SAC	
	TLF: 052-314300 Tacna 056-720078 Ica	www.criadoylopez.com
	TÚNEL KIT IMATEC	
	DEPARTAMENTO COMERCIAL	PAG. 5

2.4.- CIMENTACIÓN

La estructura no requiere cimentación, lo que facilita y agiliza el montaje. El plástico enterrado en 2 zanjas paralelas al invernadero y cubierto de tierra sujeta fuertemente la estructura al suelo.

2.5.- CUBIERTA

Material de cubierta **plástico térmico de 800gg** (0,2 mm).

Garantía: 48 meses en el norte de España y 36 en el sur

Termicidad: >al 86%

Trasmisión luminosa global: > 92%

12 líneas de monofilamento sujetas en los extremos mantienen el plástico tenso y evitan la formación de bolsas de agua.



2.6.- CALIDAD DEL ACERO

Todo el acero que se utiliza en los invernaderos IMA como materia prima es de probada calidad y homogeneidad y ha sido seleccionado en las mejores acerías europeas.

Para la realización de todo tipo de tubos, se parte de fleje comercial, el cual debe poderse plegar, su resistencia máxima no debe pasar de 41-42 Kg/mm². La resistencia media del material empleado es de 35-36 Kg/mm², se puede equiparar con Rst-34.

Hay que tener en cuenta que en todo proceso posterior al perfilado para la conformación de los tubos, el material adquiere una resistencia adicional, que mejora sus características, lo cual se ve en los ensayos mecánicos, pero no recogidos en la norma.

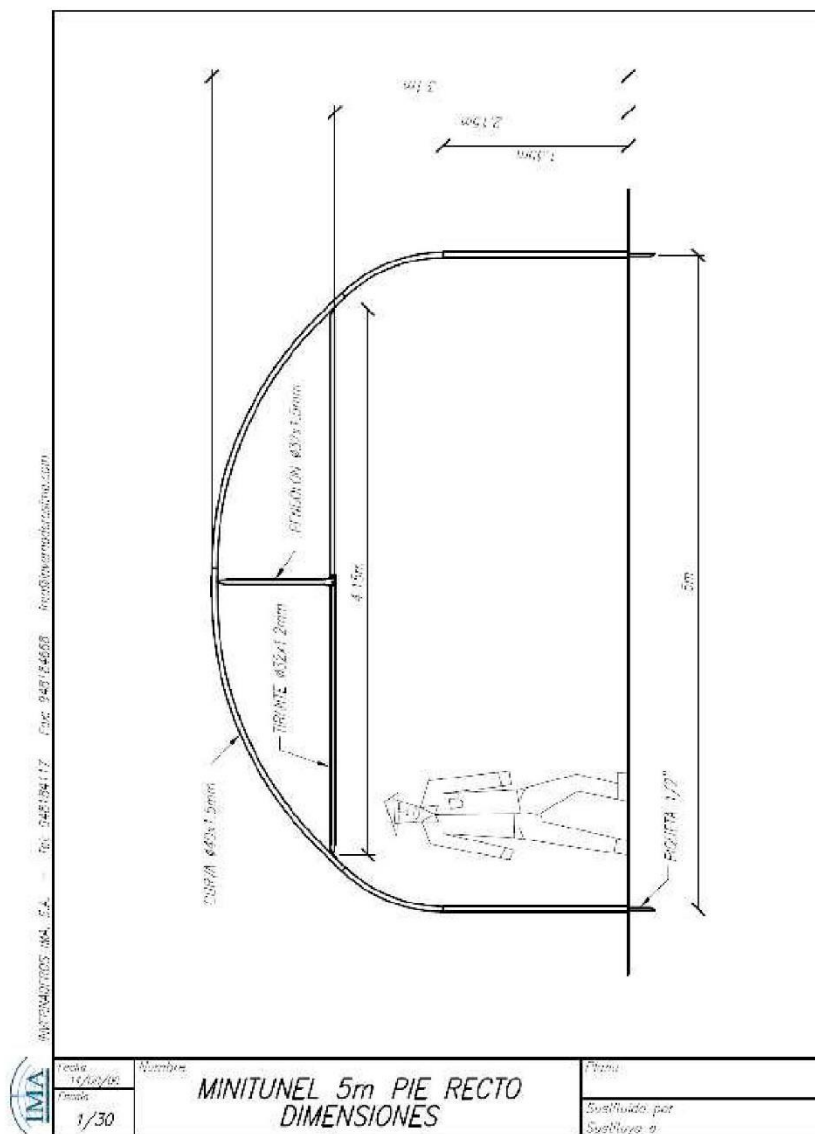
	CRIADO Y LOPEZ PERU SAC	
	TLF: 052-314300 Tacna 056-720078 Ica	www.criadoylopez.com
	TÚNEL KIT IMATEC	
	DEPARTAMENTO COMERCIAL	PAG. 6

2.7.- GALVANIZADO

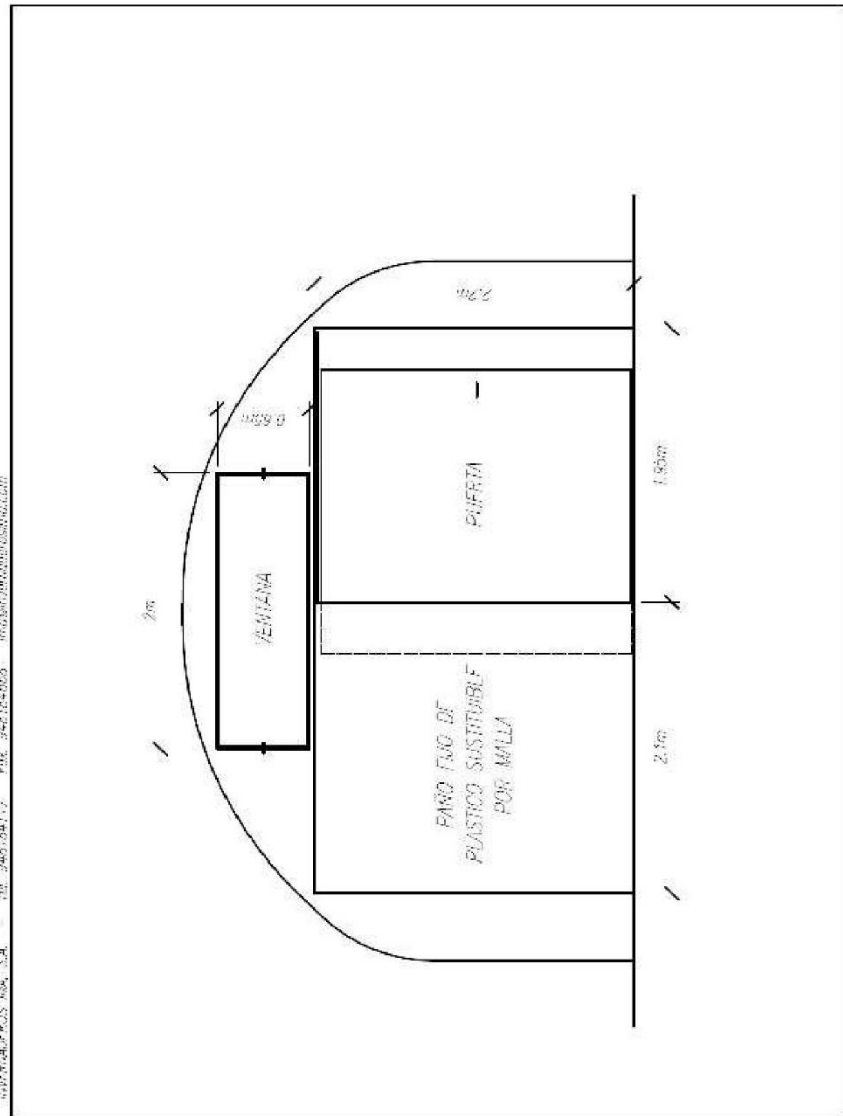
Todo el material con el que se fabrican los invernaderos IMA, es de Acero galvanizado por Inmersión según el procedimiento Sendzimir. Este proceso consiste en la inmersión en un baño de Zinc a una temperatura específica y constante durante todo el proceso, de chapa de bobina, laminada tanto en frío como en caliente y preparada para el galvanizado con un decapado previo, con lo que se consigue una auténtica fusión entre el acero y el zinc. Este proceso garantiza un recubrimiento de zinc según la norma UNE 36130 y la Euro norma 142-79. Según estas especificaciones el material utilizado es Z-275, con 275 g/m². Con esto se evita cualquier proceso de corrosión.


	CRIADO Y LOPEZ PERU SAC	
	TLF: 052-314300 Tacna 056-720078 Ica	www.criadoylopez.com
	TÚNEL KIT IMATEC	
	DEPARTAMENTO COMERCIAL	PAG. 7

3.- PLANOS



	CRIADO Y LOPEZ PERU SAC	
	TLF: 052-314300 Tacna 056-720078 Ica	www.criadoylopez.com
	TÚNEL KIT IMATEC	
	DEPARTAMENTO COMERCIAL	PAG. 8



	Fecha 14/06/02	Nombre	Plano Sustituido por Sustituye a
	Formato 1/30	MINITUNEL 5m PIE RECTO FRONTAL CON PUERTA	

	CRIADO Y LOPEZ PERU SAC	
	TLF: 052-314300 Tacna 056-720078 Ica	www.criadoylopez.com
	TÚNEL KIT IMATEC	
	DEPARTAMENTO COMERCIAL	PAG. 9

4.- FOTOS

Marcaje y piquetas



Ensamblaje arcos, tirantes, pendolones y correas



Estructura



	CRIADO Y LOPEZ PERU SAC	
	TLF: 052-314300 Tacna 056-720078 Ica	www.criadoylopez.com
	TÚNEL KIT IMATEC	
	DEPARTAMENTO COMERCIAL	PAG. 10

Cubierta plástico



Ventana y Puerta



	CRIADO Y LOPEZ PERU SAC	
	TLF: 052-314300 Tacna 056-720078 Ica	www.criadoylopez.com
	TÚNEL KIT IMATEC	
	DEPARTAMENTO COMERCIAL	PAG. 11

5.- PRESUPUESTO ESTRUCTURA Y MALLA

El presupuesto es de s/7000.

6. Presupuesto Final

Pino S.A.C. (Sistema de riego) + IMATEC(Estructura y malla) + Tecnología = **s/13 252.48**

H. Manual de referencia de la plataforma de desarrollo nexys 4 ddr



1300 Henley Court
Pullman, WA 99163
509.334.6306
www.digilentinc.com

Nexys4 DDR™ FPGA Board Reference Manual

Nexys4 DDR rev. C; Revised April 11, 2016

1 Overview

The Nexys4 DDR board is a complete, ready-to-use digital circuit development platform based on the latest Artix-7™ Field Programmable Gate Array (FPGA) from Xilinx®. With its large, high-capacity FPGA (Xilinx part number XC7A100T-1CSG324C), generous external memories, and collection of USB, Ethernet, and other ports, the Nexys4 DDR can host designs ranging from introductory combinational circuits to powerful embedded processors. Several built-in peripherals, including an accelerometer, temperature sensor, MEMs digital microphone, a speaker amplifier, and several I/O devices allow the Nexys4 DDR to be used for a wide range of designs without needing any other components.



The Nexys4 DDR

The Artix-7 FPGA is optimized for high performance logic, and offers more capacity, higher performance, and more resources than earlier designs. Artix-7 100T features include:

- 15,850 logic slices, each with four 6-input LUTs and 8 flip-flops
- 4,860 Kbits of fast block RAM
- Six clock management tiles, each with phase-locked loop (PLL)
- 240 DSP slices
- Internal clock speeds exceeding 450 MHz
- On-chip analog-to-digital converter (XADC)



The Nexys4 DDR also offers an improved collection of ports and peripherals, including:

- | | | |
|-------------------------|---|--|
| • 16 user switches | • 16 user LEDs | • Two 4-digit 7-segment displays |
| • USB-UART Bridge | • Two tri-color LEDs | • Micro SD card connector |
| • 12-bit VGA output | • PWM audio output | • PDM microphone |
| • 3-axis accelerometer | • Temperature sensor | • 10/100 Ethernet PHY |
| • 128MiB DDR2 | • Serial Flash | • Four Pmod ports |
| • Pmod for XADC signals | • Digilent USB-JTAG port for FPGA programming and communication | • USB HID Host for mice, keyboards and memory sticks |

The Nexys4 DDR is compatible with Xilinx's new high-performance Vivado® Design Suite as well as the ISE® toolset, which includes ChipScope™ and EDK. Xilinx offers free WebPACK™ versions of these toolsets, so designs can be implemented at no additional cost. The Nexys4 DDR is not supported by the Digilent Adept Utility.

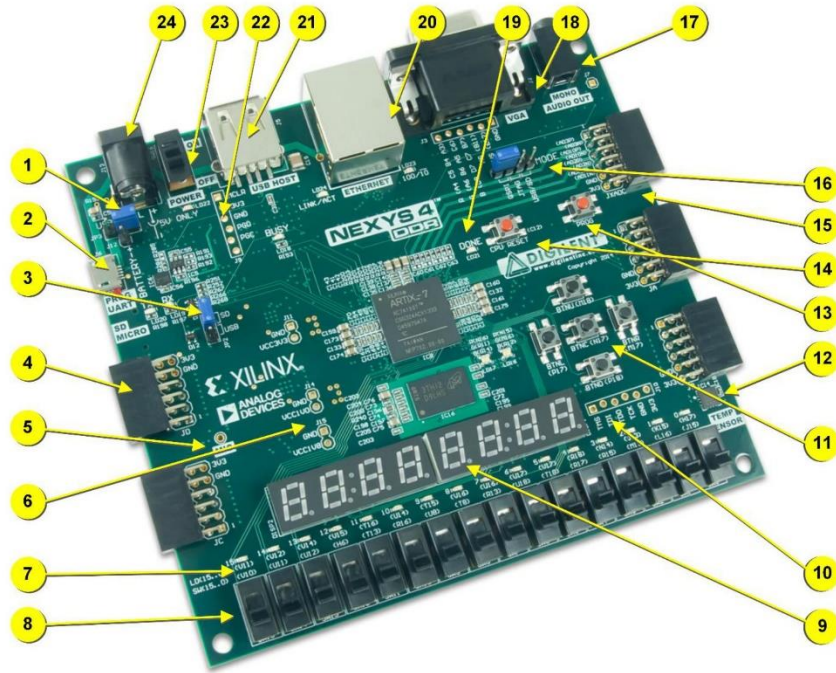


Figure 1. Nexys4 DDR board features.

Callout	Component Description	Callout	Component Description
1	Power select jumper and battery header	13	FPGA configuration reset button
2	Shared UART/ JTAG USB port	14	CPU reset button (for soft cores)
3	External configuration jumper (SD / USB)	15	Analog signal Pmod port (XADC)
4	Pmod port(s)	16	Programming mode jumper
5	Microphone	17	Audio connector
6	Power supply test point(s)	18	VGA connector
7	LEDs (16)	19	FPGA programming done LED
8	Slide switches	20	Ethernet connector
9	Eight digit 7-seg display	21	USB host connector
10	JTAG port for (optional) external cable	22	PIC24 programming port (factory use)
11	Five pushbuttons	23	Power switch
12	Temperature sensor	24	Power jack

A growing collection of board support IP, reference designs, and add-on boards are available on the Digilent website. See the Nexys4 DDR page at www.digilentinc.com for more information.

1.1 Migrating from Nexys4

The Nexys4 DDR is an incremental update to the Nexys4 board. The major improvement is the replacement of the 16 MiB CellularRAM with a 128 MiB DDR2 SDRAM memory. Digilent will provide a VHDL reference module that wraps the complexity of a DDR2 controller and is backwards compatible with the asynchronous SRAM interface of the CellularRAM, with certain limitations. See the Nexys4 DDR page at www.digilentinc.com for updates.

Furthermore, to accommodate the new memory, the pin-out of the FPGA banks has changed as well. The constraints file of existing projects will need to be updated.

The audio output (AUD_PWM) needs to be driven open-drain as opposed to push-pull on the Nexys4.

2 Power Supplies

The Nexys4 DDR board can receive power from the Digilent USB-JTAG port (J6) or from an external power supply. Jumper JP3 (near the power jack) determines which source is used.

All Nexys4 DDR power supplies can be turned on and off by a single logic-level power switch (SW16). A power-good LED (LD22), driven by the “power good” output of the ADP2118 supply, indicates that the supplies are turned on and operating normally. An overview of the Nexys4 DDR power circuit is shown in Figure 2.

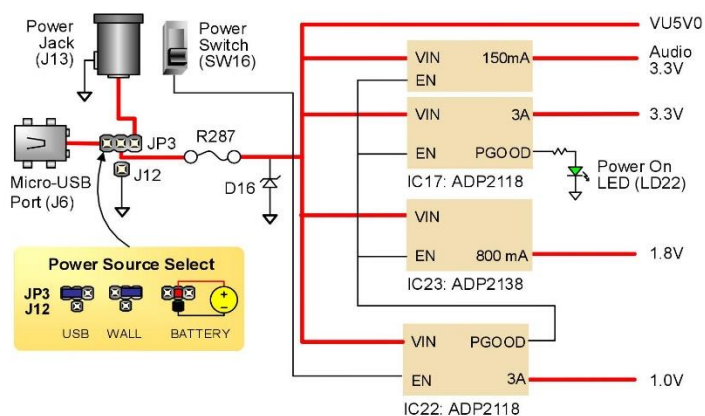


Figure 2. Nexys4 DDR power circuit.

The USB port can deliver enough power for the vast majority of designs. Our out-of-box demo draws ~400mA of current from the 5V input rail. A few demanding applications, including any that drive multiple peripheral boards, might require more power than the USB port can provide. Also, some applications may need to run without being connected to a PC's USB port. In these instances, an external power supply or battery pack can be used.

An external power supply can be used by plugging into to the power jack (JP3) and setting jumper J13 to “wall”. The supply must use a coax, center-positive 2.1mm internal-diameter plug, and deliver 4.5VDC to 5.5VDC and at

least 1A of current (i.e., at least 5W of power). Many suitable supplies can be purchased from Digilent, through Digi-Key, or other catalog vendors.

An external battery pack can be used by connecting the battery's positive terminal to the center pin of JP3 and the negative terminal to the pin labeled J12, directly below JP3. Since the main regulator on the Nexys4 DDR cannot accommodate input voltages over 5.5VDC, an external battery pack must be limited to 5.5VDC. The minimum voltage of the battery pack depends on the application: if the USB Host function (J5) is used, at least 4.6V needs to be provided. In other cases, the minimum voltage is 3.6V.

Voltage regulator circuits from Analog Devices create the required 3.3V, 1.8V, and 1.0V supplies from the main power input. Table 1 provides additional information. Typical currents depend strongly on FPGA configuration and the values provided are typical of medium size/speed designs.

Supply	Circuits	Device	Current (max/typical)
3.3V	FPGA I/O, USB ports, Clocks, RAM I/O, Ethernet, SD slot, Sensors, Flash	IC17: ADP2118	3A/0.1 to 1.5A
1.0V	FPGA Core	IC22: ADP2118	3A/ 0.2 to 1.3A
1.8V	DDR2, FPGA Auxiliary and RAM	IC23: ADP2138	0.8A/ 0.5A

Table 1. Nexys4 DDR power supplies.

2.1 Power protection

The Nexys4 DDR features overcurrent and overvoltage protection on the input power rail. A 3.5A fuse (R287) and a 5V Zener diode (D16) provide a non-resettable protection for other on-board integrated circuits, as displayed in Figure 2. Applying power outside of the specs outlined in this document is not covered by warranty. If this happens, either or both might get permanently damaged. The damaged parts are not user-replaceable.

3 FPGA Configuration

After power-on, the Artix-7 FPGA must be configured (or programmed) before it can perform any functions. You can configure the FPGA in one of four ways:

1. A PC can use the Digilent USB-JTAG circuitry (portJ6, labeled "PROG") to program the FPGA any time the power is on.
2. A file stored in the nonvolatile serial (SPI) flash device can be transferred to the FPGA using the SPI port.
3. A programming file can be transferred to the FPGA from a micro SD card.
4. A programming file can be transferred from a USB memory stick attached to the USB HID port.

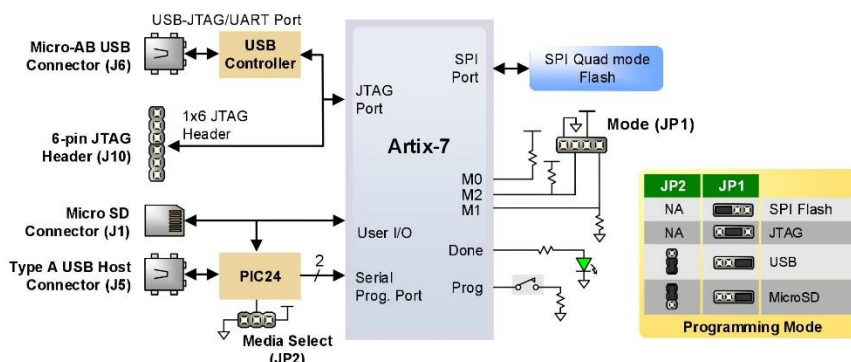


Figure 3. Nexys4 DDR configuration options.

Figure 3 shows the different options available for configuring the FPGA. An on-board “mode” jumper (JP1) and a media selection jumper (JP2) select between the programming modes.

The FPGA configuration data is stored in files called bitstreams that have the .bit file extension. The ISE or Vivado software from Xilinx can create bitstreams from VHDL, Verilog®, or schematic-based source files (in the ISE toolset, EDK is used for MicroBlaze™ embedded processor-based designs).

Bitstreams are stored in SRAM-based memory cells within the FPGA. This data defines the FPGA’s logic functions and circuit connections, and it remains valid until it is erased by removing board power, by pressing the reset button attached to the PROG input, or by writing a new configuration file using the JTAG port.

An Artix-7 100T bitstream is typically 30,606,304 bits and can take a long time to transfer. The time it takes to program the Nexys4 can be decreased by compressing the bitstream before programming, and then allowing the FPGA to decompress the bitstream itself during configuration. Depending on design complexity, compression ratios of 10x can be achieved. Bitstream compression can be enabled within the Xilinx tools (ISE or Vivado) to occur during generation. For instructions on how to do this, consult the Xilinx documentation for the toolset being used.

After being successfully programmed, the FPGA will cause the “DONE” LED to illuminate. Pressing the “PROG” button at any time will reset the configuration memory in the FPGA. After being reset, the FPGA will immediately attempt to reprogram itself from whatever method has been selected by the programming mode jumpers.

The following sections provide greater detail about programming the Nexys4 DDR using the different methods available.

3.1 JTAG Configuration

The Xilinx tools typically communicate with FPGAs using the Test Access Port and Boundary-Scan Architecture, commonly referred to as JTAG. During JTAG programming, a .bit file is transferred from the PC to the FPGA using the onboard Digilent USB-JTAG circuitry (port J6) or an external JTAG programmer, such as the Digilent JTAG-HS2, attached to port J10. You can perform JTAG programming any time after the Nexys4 DDR has been powered on, regardless of what the mode jumper (JP1) is set to. If the FPGA is already configured, then the existing configuration is overwritten with the bitstream being transmitted over JTAG. Setting the mode jumper to the JTAG

setting (seen in Figure 3) is useful to prevent the FPGA from being configured from any other bitstream source until a JTAG programming occurs.

Programming the Nexys4 DDR with an uncompressed bitstream using the on-board USB-JTAG circuitry usually takes around five seconds. JTAG programming can be done using the hardware server in Vivado or the iMPACT tool included with ISE and the Lab Tools version of Vivado. The demonstration project available at www.digilentinc.com gives an in-depth tutorial on how to program your board.

3.2 Quad-SPI Configuration

Since the FPGA on the Nexys4 DDR is volatile, it relies on the Quad-SPI flash memory to store the configuration between power cycles. This configuration mode is called Master SPI. The blank FPGA takes the role of master and reads the configuration file out of the flash device upon power-up. To that effect, a configuration file needs to be downloaded first to the flash. When programming a nonvolatile flash device, a bitstream file is transferred to the flash in a two-step process. First, the FPGA is programmed with a circuit that can program flash devices, and then data is transferred to the flash device via the FPGA circuit (this complexity is hidden from the user by the Xilinx tools). This is called indirect programming. After the flash device has been programmed, it can automatically configure the FPGA at a subsequent power-on or reset event as determined by the mode jumper setting (see Figure 3). Programming files stored in the flash device will remain until they are overwritten, regardless of power-cycle events.

Programming the flash can take as long as four to five minutes, which is mostly due to the lengthy erase process inherent to the memory technology. Once written however, FPGA configuration can be very fast—less than a second. Bitstream compression, SPI bus width, and configuration rate are factors controlled by the Xilinx tools that can affect configuration speed. The Nexys4 DDR supports x1, x2, and x4 bus widths and data rates of up to 50 MHz for Quad-SPI programming.

Quad-SPI programming can be done using the iMPACT tool included with ISE or the Lab Tools version of Vivado.

3.3 USB Host and Micro SD Programming

You can program the FPGA from a pen drive attached to the USB Host port (J5) or a microSD card inserted into J1 by doing the following:

1. Format the storage device (Pen drive or microSD card) with a FAT32 file system.
2. Place a single .bit configuration file in the root directory of the storage device.
3. Attach the storage device to the Nexys4 DDR.
4. Set the JP1 Programming Mode jumper on the Nexys4 DDR to “USB/SD”.
5. Select the desired storage device using JP2.
6. Push the PROG button or power-cycle the Nexys4 DDR.

The FPGA will automatically configure with the .bit file on the selected storage device. Any .bit files that are not built for the proper Artix-7 device will be rejected by the FPGA.

The Auxiliary Function Status, or “BUSY” LED, gives visual feedback on the state of the configuration process when the FPGA is not yet programmed:

- When steadily lit, the auxiliary microcontroller is either booting up or currently reading the configuration medium (microSD or pen drive) and downloading a bitstream to the FPGA.
- A slow pulse means the microcontroller is waiting for a configuration medium to be plugged in.

- In case of an error during configuration, the LED will blink rapidly.

When the FPGA has been successfully configured, the behavior of the LED is application-specific. For example, if a USB keyboard is plugged in, a rapid blink will signal the receipt of an HID input report from the keyboard.

4 Memory

The Nexys4 DDR board contains two external memories: a 1Gib (128MiB) DDR2 SDRAM and a 128Mib (16MiB) non-volatile serial Flash device. The DDR2 modules are integrated on-board and connect to the FPGA using the industry standard interface. The serial Flash is on a dedicated quad-mode (x4) SPI bus. The connections and pin assignments between the FPGA and external memories are shown below.

4.1 DDR2

The Nexys4 DDR includes one Micron MT47H64M16HR-25:H DDR2 memory component, creating a single rank, 16-bit wide interface. It is routed to a 1.8V-powered HR (High Range) FPGA bank with 50 ohm controlled single-ended trace impedance. 50 ohm internal terminations in the FPGA are used to match the trace characteristics. Similarly, on the memory side, on-die terminations (ODT) are used for impedance matching.

For proper operation of the memory, a memory controller and physical layer (PHY) interface needs to be included in the FPGA design. There are two recommended ways to do that, which are outlined below and differ in complexity and design flexibility.

The straightforward way is to use the Diligent-provided DDR-to-SRAM adapter module which instantiates the memory controller and uses an asynchronous SRAM bus for interfacing with user logic. This module provides backward compatibility with projects written for older Nexys-line boards featuring a CellularRAM instead of DDR2. It trades memory bandwidth for simplicity.

More advanced users or those who wish to learn more about DDR SDRAM technology may want to use the Xilinx 7-series memory interface solutions core generated by the MIG (Memory Interface Generator) Wizard. Depending on the tool used (ISE, EDK or Vivado), the MIG Wizard can generate a native FIFO-style or an AXI4 interface to connect to user logic. This workflow allows the customization of several DDR parameters optimized for the particular application. Table 2 below lists the MIG Wizard settings optimized for the Nexys4 DDR.

Setting	Value
Memory type	DDR2 SDRAM
Max. clock period	3000ps (667Mbps data rate)
Recommended clock period (for easy clock generation)	3077ps (650Mbps data rate)
Memory part	MT47H64M16HR-25E
Data width	16
Data mask	Enabled
Chip Select pin	Enabled
Rtt (nominal) – On-die termination	50ohms
Internal Vref	Enabled
Internal termination impedance	50ohms

Table 2. DDR2 settings for the Nexys4 DDR.

Although the FPGA, memory IC, and the board itself are capable of the maximum data rate of 667Mbps, the limitations in the clock generation primitives restrict the clock frequencies that can be generated from the 100 MHz system clock. Thus, for simplicity, the next highest data rate of 650Mbps is recommended.

The MIG Wizard will require the fixed pin-out of the memory signals to be entered and validated before generating the IP core. For your convenience, an importable UCF file is provided on the Digilent website to speed up the process.

For more details on the Xilinx memory interface solutions, refer to the 7 Series FPGAs Memory Interface Solutions User Guide (ug586)¹.

4.2 Quad-SPI Flash

FPGA configuration files can be written to the Quad-SPI Flash (Spansion part number S25FL128S), and mode settings are available to cause the FPGA to automatically read a configuration from this device at power on. An Artix-7 100T configuration file requires just less than four MiB (mebibyte) of memory, leaving about 77% of the flash device available for user data. Or, if the FPGA is getting configured from another source, the whole memory can be used for custom data.

The contents of the memory can be manipulated by issuing certain commands on the SPI bus. The implementation of this protocol is outside the scope of this document. All signals in the SPI bus except SCK are general-purpose user I/O pins after FPGA configuration. SCK is an exception because it remains a dedicated pin even after configuration. Access to this pin is provided through a special FPGA primitive called STARTUPE2.

NOTE: Refer to the manufacturer's data sheets² and Xilinx user guides³ for more information.

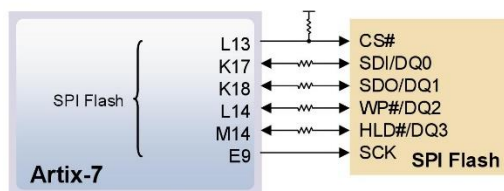


Figure 4. Nexys4 DDR SPI flash pin-out.

5 Ethernet PHY

The Nexys4 DDR board includes an SMSC 10/100 Ethernet PHY (SMSC part number LAN8720A) paired with an RJ-45 Ethernet jack with integrated magnetics. The SMSC PHY uses the RMII interface and supports 10/100 Mb/s. Figure 5 illustrates the pin connections between the Artix-7 and the Ethernet PHY. At power-on reset, the PHY is set to the following defaults:

- RMII mode interface
- Auto-negotiation enabled, advertising all 10/100 mode capable
- PHY address=00001

¹ http://www.xilinx.com/support/documentation/ip_documentation/mig_7series/v2_1/ug586_7Series_MIS.pdf

² http://www.spansion.com/Support/Datasheets/S25FL128S_256S_00.pdf

³ http://www.xilinx.com/support/documentation/user_guides/ug470_7Series_Config.pdf

Two on-board LEDs (LD23 = LED2, LD24 = LED1) connected to the PHY provide link status and data activity feedback. See the PHY datasheet for details.

EDK-based designs can access the PHY using either the `axi_etherlite` (AXI EthernetLite) IP core or the `axi_ethernet` (Tri Mode Ethernet MAC) IP core. A `mii_to_rmii` core (Ethernet PHY MII to Reduced MII) needs to be inserted to convert the MAC interface from MII to RMII. Also, a 50 MHz clock needs to be generated for the `mii_to_rmii` core and the `CLKIN` pin of the external PHY. To account for skew introduced by the `mii_to_rmii` core, generate each clock individually, with the external PHY clock having a 45 degree phase shift relative to the `mii_to_rmii` `Ref_Clk`. An EDK demonstration project that properly uses the Ethernet PHY can be found on the Nexys4 DDR product page at www.digilentinc.com.

ISE designs can use the IP Core Generator wizard to create an Ethernet MAC controller IP core.

NOTE: Refer to the LAN8720A data sheet⁴ for further information.

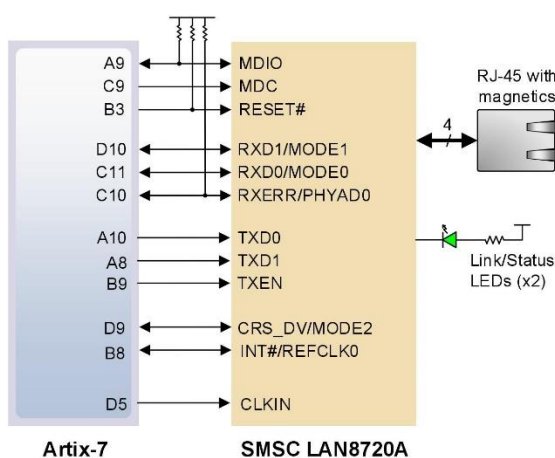


Figure 5. Pin connections between the Artix-7 and the Ethernet PHY.

6 Oscillators/Clocks

The Nexys4 DDR board includes a single 100 MHz crystal oscillator connected to pin E3 (E3 is a MRCC input on bank 35). The input clock can drive MMCMs or PLLs to generate clocks of various frequencies and with known phase relationships that may be needed throughout a design. Some rules restrict which MMCMs and PLLs may be driven by the 100 MHz input clock. For a full description of these rules and of the capabilities of the Artix-7 clocking resources, refer to the “7 Series FPGAs Clocking Resources User Guide” available from Xilinx.

Xilinx offers the Clocking Wizard IP core to help users generate the different clocks required for a specific design. This wizard will properly instantiate the needed MMCMs and PLLs based on the desired frequencies and phase relationships specified by the user. The wizard will then output an easy-to-use wrapper component around these

⁴ <http://ww1.microchip.com/downloads/en/DeviceDoc/8720a.pdf>

clocking resources that can be inserted into the user's design. The clocking wizard can be accessed from within the Project Navigator or Core Generator tools.

7 USB-UART Bridge (Serial Port)

The Nexys4 DDR includes an FTDI FT2232HQ USB-UART bridge (attached to connector J6) that allows you use PC applications to communicate with the board using standard Windows COM port commands. Free USB-COM port drivers, available from www.ftdichip.com under the "Virtual Com Port" or VCP heading, convert USB packets to UART/serial port data. Serial port data is exchanged with the FPGA using a two-wire serial port (TXD/RXD) and optional hardware flow control (RTS/CTS). After the drivers are installed, I/O commands can be used from the PC directed to the COM port to produce serial data traffic on the C4 and D4 FPGA pins.

Two on-board status LEDs provide visual feedback on traffic flowing through the port: the transmit LED (LD20) and the receive LED (LD19). Signal names that imply direction are from the point-of-view of the DTE (Data Terminal Equipment), in this case the PC.

The FT2232HQ is also used as the controller for the Digilent USB-JTAG circuitry, but the USB-UART and USB-JTAG functions behave entirely independent of one another. Programmers interested in using the UART functionality of the FT2232 within their design do not need to worry about the JTAG circuitry interfering with the UART data transfers, and vice-versa. The combination of these two features into a single device allows the Nexys4 DDR to be programmed, communicated with via UART, and powered from a computer attached with a single Micro USB cable.

The connections between the FT2232HQ and the Artix-7 are shown in Figure 6.

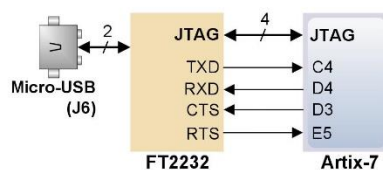


Figure 6. Nexys4 DDR FT2232HQ connections.

8 USB HID Host

The Auxiliary Function microcontroller (Microchip PIC24FJ128) provides the Nexys4 DDR with USB Embedded HID host capability. After power-up, the microcontroller is in configuration mode, either downloading a bitstream to the FPGA, or waiting to be programmed from other sources. Once the FPGA is programmed, the microcontroller switches to application mode, which is USB HID Host in this case. Firmware in the microcontroller can drive a mouse or a keyboard attached to the type A USB connector at J5 labeled "USB Host". Hub support is not currently available, so only a single mouse or a single keyboard can be used. Only keyboards and mice supporting the Boot HID interface are supported. The PIC24 drives several signals into the FPGA – two are used to implement a standard PS/2 interface for communication with a mouse or keyboard, and the others are connected to the FPGA's two-wire serial programming port, so the FPGA can be programmed from a file stored on a USB pen drive or microSD card.

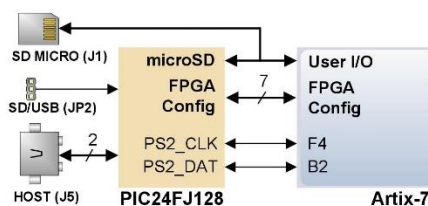


Figure 7. Nexys4 DDR PIC24 connections.

8.1 HID Controller

The Auxiliary Function microcontroller hides the USB HID protocol from the FPGA and emulates an old-style PS/2 bus. The microcontroller behaves just like a PS/2 keyboard or mouse would. This means new designs can re-use existing PS/2 IP cores. Mice and keyboards that use the PS/2 protocol use a two-wire serial bus (clock and data) to communicate with a host. On the Nexys4 DDR, the microcontroller emulates a PS/2 device while the FPGA plays the role of the host. Both the mouse and the keyboard use 11-bit words that include a start bit, data byte (LSB first), odd parity, and stop bit, but the data packets are organized differently, and the keyboard interface allows bi-directional data transfers (so the host device can illuminate state LEDs on the keyboard). Bus timings are shown in Figure 8.

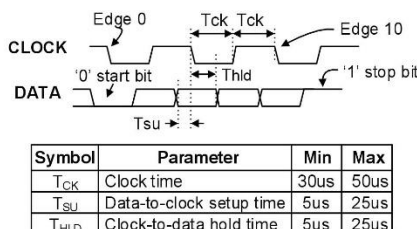


Figure 8. PS/2 device-to-host timing diagram.

The clock and data signals are only driven when data transfers occur; otherwise, they are held in the idle state at high-impedance (open-drain drivers). This requires that when the PS/2 signals are used in a design, internal pull-ups must be enabled in the FPGA on the data and clock pins. The clock signal is normally driven by the device, but may be held low by the host in special cases. The timings define signal requirements for mouse-to-host communications and bi-directional keyboard communications. A PS/2 interface circuit can be implemented in the FPGA to create a keyboard or mouse interface.

When a keyboard or mouse is connected to the Nexys4 DDR, a “self-test passed” command (0xAA) is sent to the host. After this, commands may be issued to the device. Since both the keyboard and the mouse use the same PS/2 port, one can tell the type of device connected using the device ID. This ID can be read by issuing a Read ID command (0xF2). Also, a mouse sends its ID (0x00) right after the “self-test passed” command, which distinguishes it from a keyboard.

8.2 Keyboard

PS/2-style keyboards use scan codes to communicate key press data. Each key is assigned a code that is sent whenever the key is pressed. If the key is held down, the scan code will be sent repeatedly about once every 100ms. When a key is released, an F0 key-up code is sent, followed by the scan code of the released key. If a key can be shifted to produce a new character (like a capital letter), then a shift character is sent in addition to the scan code and the host must determine which ASCII character to use. Some keys, called extended keys, send an E0 ahead of the scan code (and they may send more than one scan code). When an extended key is released, an E0 F0 key-up code is sent, followed by the scan code. Scan codes for most keys are shown in Figure 9.

ESC 76	F1 05	F2 06	F3 04	F4 0C	F5 03	F6 0B	F7 83	F8 0A	F9 01	F10 09	F11 78	F12 07	
~ 0E	1! 16	2@ 1E	3# 26	4\$ 25	5% 2E	6^ 36	7& 3D	8* 3E	9(46	0) 45	-_ 4E	=+ 55	BackSpace ← 66
TAB 0D	Q 15	W 1D	E 24	R 2D	T 2C	Y 35	U 3C	I 43	O 44	P 4D	[{ 54]} 5B	\ 5D
Caps Lock 58	A 1C	S 1B	D 23	F 2B	G 34	H 33	J 3B	K 42	L 4B	:: 4C	'" 52	Enter ↵ 5A	
Shift 12	Z 1Z	X 22	C 21	V 2A	B 32	N 31	M 3A	,< 41	>. 49	/? 4A	↑	Shift 59	
Ctrl 14	Alt 11	Space 29						Alt E0 11	Ctrl E0 14				

Figure 9. Keyboard scan codes.

A host device can also send data to the keyboard. Table 3 shows a list of some common commands a host might send.

The keyboard can send data to the host only when both the data and clock lines are high (or idle). Because the host is the bus master, the keyboard must check to see whether the host is sending data before driving the bus. To facilitate this, the clock line is used as a “clear to send” signal. If the host drives the clock line low, the keyboard must not send any data until the clock is released. The keyboard sends data to the host in 11-bit words that contain a ‘0’ start bit, followed by 8-bits of scan code (LSB first), followed by an odd parity bit, and terminated with a ‘1’ stop bit. The keyboard generates 11 clock transitions (at 20 to 30 KHz) when the data is sent, and data is valid on the falling edge of the clock.

Command	Action
ED	Set Num Lock, Caps Lock, and Scroll Lock LEDs. Keyboard returns FA after receiving ED, then host sends a byte to set LED status: bit 0 sets Scroll Lock, bit 1 sets Num Lock, and bit 2 sets Caps lock. Bits 3 to 7 are ignored.
EE	Echo (test). Keyboard returns EE after receiving EE
F3	Set scan code repeat rate. Keyboard returns F3 on receiving FA, then host sends second byte to set the repeat rate.
FE	Resend. FE directs keyboard to re-send most recent scan code.
FF	Reset. Resets the keyboard.

Table 3. Keyboard commands.

8.3 Mouse

Once entered in stream mode and data reporting is enabled, the mouse outputs a clock and data signal when it is moved; otherwise, these signals remain at logic '1.' Each time the mouse is moved, three 11-bit words are sent from the mouse to the host device, as shown in Figure 10. Each of the 11-bit words contains a '0' start bit, followed by 8 bits of data (LSB first), followed by an odd parity bit, and terminated with a '1' stop bit. Thus, each data transmission contains 33 bits, where bits 0, 11, and 22 are '0' start bits, and bits 11, 21, and 33 are '1' stop bits. The three 8-bit data fields contain movement data, as shown in Figure 10. Data is valid at the falling edge of the clock, and the clock period is 20 to 30 KHz.

The mouse assumes a relative coordinate system wherein moving the mouse to the right generates a positive number in the X field, and moving to the left generates a negative number. Likewise, moving the mouse up generates a positive number in the Y field, and moving down represents a negative number (the XS and YS bits in the status byte are the sign bits – a '1' indicates a negative number). The magnitude of the X and Y numbers represent the rate of mouse movement; the larger the number, the faster the mouse is moving (the XV and YV bits in the status byte are movement overflow indicators. A '1' means overflow has occurred). If the mouse moves continuously, the 33-bit transmissions are repeated every 50ms or so. The L and R fields in the status byte indicate Left and Right button presses (a '1' indicates the button is being pressed).

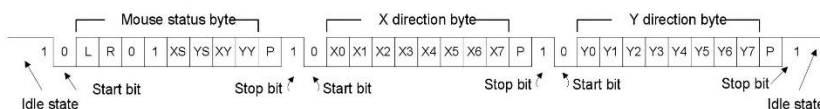


Figure 10. Mouse data format.

The microcontroller also supports Microsoft® IntelliMouse®-type extensions for reporting back a third axis representing the mouse wheel, as shown in Table 4.

Command	Action
EA	Set stream mode. The mouse responds with "acknowledge" (0xFA) then resets its movement counters and enters stream mode.
F4	Enable data reporting. The mouse responds with "acknowledge" (0xFA) then enables data reporting and resets its movement counters. This command only affects behavior in stream mode. Once issued, mouse movement will automatically generate a data packet.
F5	Disable data reporting. The mouse responds with "acknowledge" (0xFA) then disables data reporting and resets its movement counters.
F3	Set mouse sample rate. The mouse responds with "acknowledge" (0xFA) then reads one more byte from the host. This byte is then saved as the new sample rate, and a new "acknowledge" packet is issued.
FE	Resend. FE directs mouse to re-send last packet.
FF	Reset. The mouse responds with "acknowledge" (0xFA) then enters reset mode.

Table 4. Microsoft IntelliMouse-type extensions, commands, and actions.

9 VGA Port

The Nexys4 DDR board uses 14 FPGA signals to create a VGA port with 4 bits-per-color and the two standard sync signals (HS – Horizontal Sync, and VS – Vertical Sync). The color signals use resistor-divider circuits that work in conjunction with the 75-ohm termination resistance of the VGA display to create 16 signal levels each on the red, green, and blue VGA signals. This circuit, shown in Figure 11, produces video color signals that proceed in equal increments between 0V (fully off) and 0.7V (fully on). Using this circuit, 4096 different colors can be displayed, one for each unique 12-bit pattern. A video controller circuit must be created in the FPGA to drive the sync and color signals with the correct timing in order to produce a working display system.

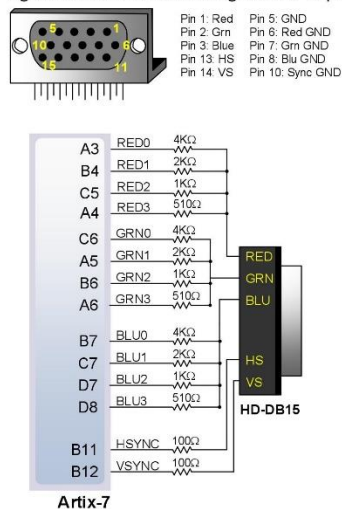


Figure 11. Nexys4 DDR VGA interface.

9.1 VGA System Timing

VGA signal timings are specified, published, copyrighted, and sold by the VESA® organization (www.vesa.org). The following VGA system timing information is provided as an example of how a VGA monitor might be driven in 640 by 480 mode.

NOTE: For more precise information, or for information on other VGA frequencies, refer to documentation available at the VESA website.

CRT-based VGA displays use amplitude-modulated moving electron beams (or cathode rays) to display information on a phosphor-coated screen. LCD displays use an array of switches that can impose a voltage across a small amount of liquid crystal, thereby changing light permittivity through the crystal on a pixel-by-pixel basis. Although the following description is limited to CRT displays, LCD displays have evolved to use the same signal timings as CRT displays (so the “signals” discussion below pertains to both CRTs and LCDs). Color CRT displays use three electron beams (one for red, one for blue, and one for green) to energize the phosphor that coats the inner side of the display end of a cathode ray tube (see Figure 12).

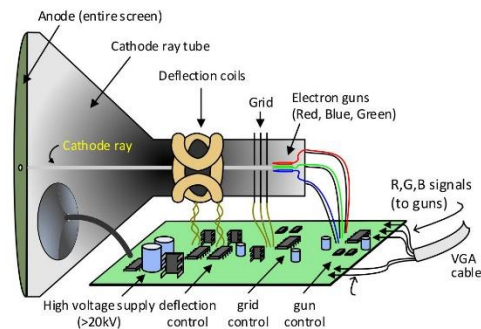


Figure 12. Color CRT display.

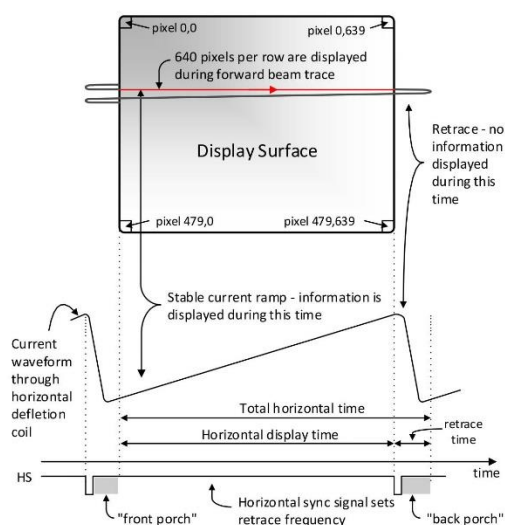
Electron beams emanate from “electron guns,” which are finely-pointed heated cathodes placed in close proximity to a positively charged annular plate called a “grid.” The electrostatic force imposed by the grid pulls rays of energized electrons from the cathodes, and those rays are fed by the current that flows into the cathodes. These particle rays are initially accelerated towards the grid, but they soon fall under the influence of the much larger electrostatic force that results from the entire phosphor-coated display surface of the CRT being charged to 20kV (or more). The rays are focused to a fine beam as they pass through the center of the grids, and then they accelerate to impact on the phosphor-coated display surface. The phosphor surface glows brightly at the impact point, and it continues to glow for several hundred microseconds after the beam is removed. The larger the current fed into the cathode, the brighter the phosphor will glow.

Figure 13. VGA horizontal synchronization.

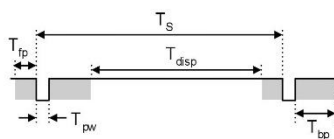
Between the grid and the display surface, the beam passes through the neck of the CRT where two coils of wire produce orthogonal electromagnetic fields. Because cathode rays are composed of charged particles (electrons), they can be deflected by these magnetic fields. Current waveforms are passed through the coils to produce magnetic fields that interact with the cathode rays and cause them to transverse the display surface in a “raster” pattern, horizontally from left to right and vertically from top to bottom, as shown in Figure 13. As the cathode ray moves over the surface of the display, the current sent to the electron guns can be increased or decreased to change the brightness of the display at the cathode ray impact point.

Information is only displayed when the beam is moving in the “forward” direction (left to right and top to bottom), and not during the time the beam is reset back to the left or top edge of the display. Much of the potential display time is therefore lost in “blanking” periods when the beam is reset and stabilized to begin a new horizontal or vertical display pass. The size of the beams, the frequency at which the beam can be traced across the display, and the frequency at which the electron beam can be modulated determine the display resolution.

Modern VGA displays can accommodate different resolutions, and a VGA controller circuit dictates the resolution by producing timing signals to control the raster patterns. The controller must produce synchronizing pulses at 3.3V (or 5V) to set the frequency at which current flows through the deflection coils, and it must ensure that video data is applied to the electron guns at the correct time. Raster video displays define a number of “rows” that corresponds to the number of horizontal passes the cathode makes over the display area, and a number of “columns” that corresponds to an area on each row that is assigned to one “picture element,” or pixel. Typical



information. The VS signal defines the “refresh” frequency of the display, or the frequency at which all information on the display is redrawn. The minimum refresh frequency is a function of the display’s phosphor and electron beam intensity, with practical refresh frequencies falling in the 50Hz to 120Hz range. The number of lines to be displayed at a given refresh frequency defines the horizontal “retrace” frequency. For a 640-pixel by 480-row display using a 25 MHz pixel clock and 60 +/-1Hz refresh, the signal timings shown in Figure 14 can be derived. Timings for sync pulse width and front and back porch intervals (porch intervals are the pre- and post-sync pulse times during which information cannot be displayed) are based on observations taken from actual VGA displays.



Symbol	Parameter	Vertical Sync			Horiz. Sync	
		Time	Clocks	Lines	Time	Clks
T_s	Sync pulse	16.7ms	416,800	521	32 μ s	800
T_{disp}	Display time	15.36ms	384,000	480	25.6 μ s	640
T_{pw}	Pulse width	64 μ s	1,600	2	3.84 μ s	96
T_{fp}	Front porch	320 μ s	8,000	10	640 ns	16
T_{bp}	Back porch	928 μ s	23,200	29	1.92 μ s	48

Figure 14. Signal timings for a 640-pixel by 480 row display using a 25 MHz pixel clock and 60 Hz vertical refresh.

A VGA controller circuit, such as the one diagramed in Figure 15, decodes the output of a horizontal-sync counter driven by the pixel clock to generate HS signal timings. You can use this counter to locate any pixel location on a given row. Likewise, the output of a vertical-sync counter that increments with each HS pulse can be used to generate VS signal timings, and you can use this counter to locate any given row. These two continually running

counters can be used to form an address into video RAM. No time relationship between the onset of the HS pulse and the onset of the VS pulse is specified, so you can arrange the counters to easily form video RAM addresses, or to minimize decoding logic for sync pulse generation.

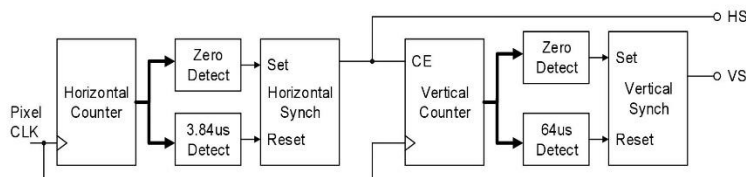


Figure 15. VGA display controller block diagram.

10 Basic I/O

The Nexys4 DDR board includes two tri-color LEDs, sixteen slide switches, six push buttons, sixteen individual LEDs, and an eight-digit seven-segment display, as shown in Figure 16. The pushbuttons and slide switches are connected to the FPGA via series resistors to prevent damage from inadvertent short circuits (a short circuit could occur if an FPGA pin assigned to a pushbutton or slide switch was inadvertently defined as an output). The five pushbuttons arranged in a plus-sign configuration are "momentary" switches that normally generate a low output when they are at rest, and a high output only when they are pressed. The red pushbutton labeled "CPU RESET," on the other hand, generates a high output when at rest and a low output when pressed. The CPU RESET button is intended to be used in EDK designs to reset the processor, but you can also use it as a general purpose pushbutton. Slide switches generate constant high or low inputs depending on their position.

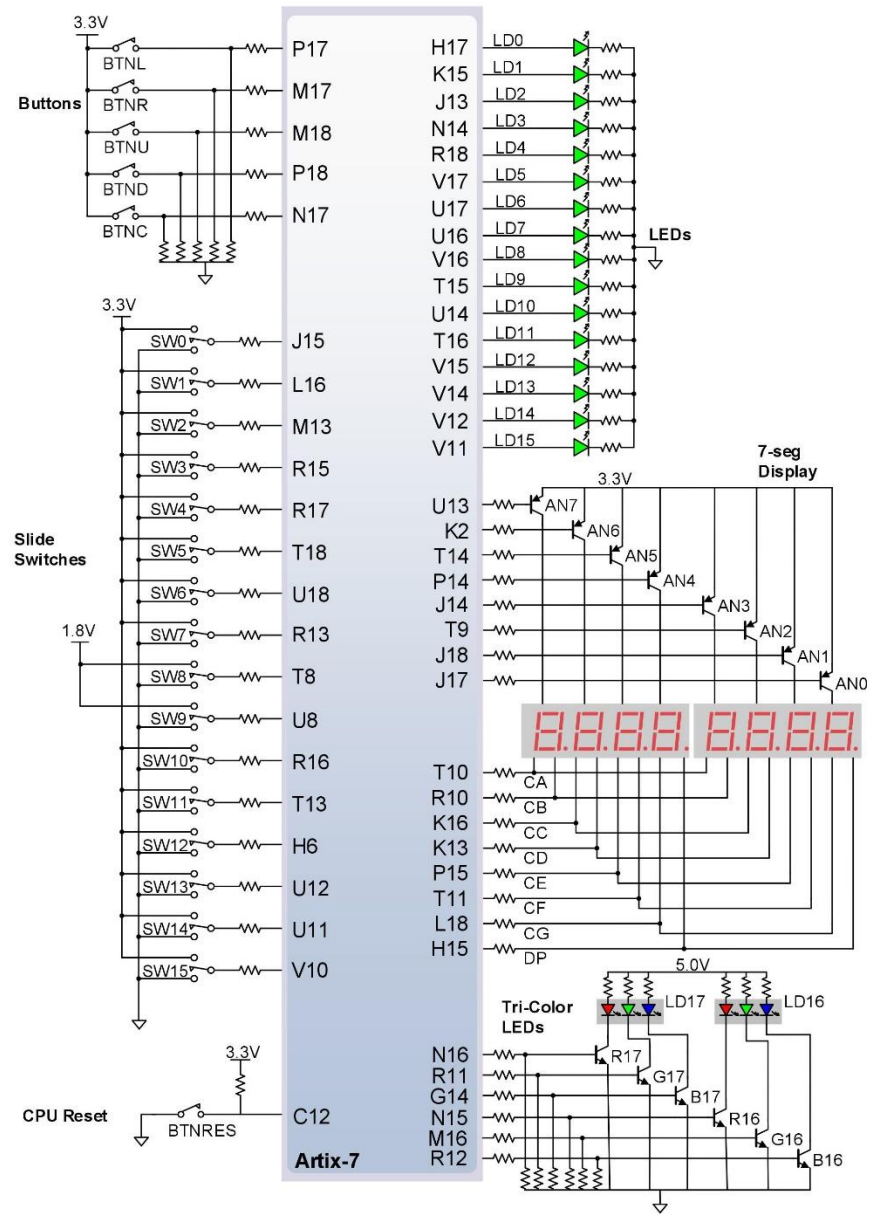


Figure 16. General Purpose I/O devices on the Nexys4 DDR.

The sixteen individual high-efficiency LEDs are anode-connected to the FPGA via 330-ohm resistors, so they will turn on when a logic high voltage is applied to their respective I/O pin. Additional LEDs that are not user-accessible indicate power-on, FPGA programming status, and USB and Ethernet port status.

10.1 Seven-Segment Display

The Nexys4 DDR board contains two four-digit common anode seven-segment LED displays, configured to behave like a single eight-digit display. Each of the eight digits is composed of seven segments arranged in a “figure 8” pattern, with an LED embedded in each segment. Segment LEDs can be individually illuminated, so any one of 128 patterns can be displayed on a digit by illuminating certain LED segments and leaving the others dark, as shown in Figure 17. Of these 128 possible patterns, the ten corresponding to the decimal digits are the most useful.

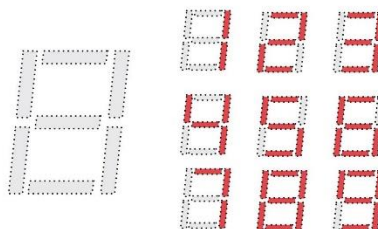


Figure 17. An un-illuminated seven-segment display and nine illumination patterns corresponding to decimal digits.

The anodes of the seven LEDs forming each digit are tied together into one “common anode” circuit node, but the LED cathodes remain separate, as shown in Fig 18. The common anode signals are available as eight “digit enable” input signals to the 8-digit display. The cathodes of similar segments on all four displays are connected into seven circuit nodes labeled CA through CG. For example, the eight “D” cathodes from the eight digits are grouped together into a single circuit node called “CD.” These seven cathode signals are available as inputs to the 8-digit display. This signal connection scheme creates a multiplexed display, where the cathode signals are common to all digits but they can only illuminate the segments of the digit whose corresponding anode signal is asserted.

To illuminate a segment, the anode should be driven high while the cathode is driven low. However, since the Nexys4 DDR uses transistors to drive enough current into the common anode point, the anode enables are inverted. Therefore, both the AN0..7 and the CA..G/DP signals are driven low when active.

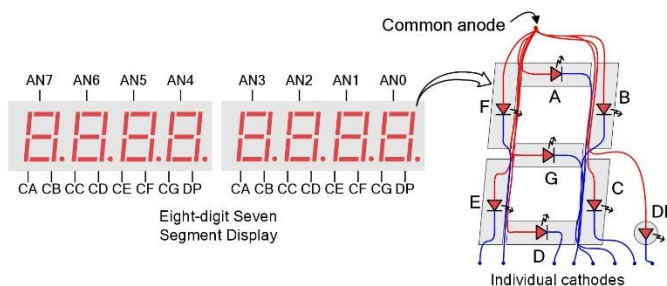


Figure 18. Common anode circuit node.

A scanning display controller circuit can be used to show an eight-digit number on this display. This circuit drives the anode signals and corresponding cathode patterns of each digit in a repeating, continuous succession at an update rate that is faster than the human eye can detect. Each digit is illuminated just one-eighth of the time, but because the eye cannot perceive the darkening of a digit before it is illuminated again, the digit appears continuously illuminated. If the update, or “refresh”, rate is slowed to around 45Hz, a flicker can be noticed in the display.

For each of the four digits to appear bright and continuously illuminated, all eight digits should be driven once every 1 to 16ms, for a refresh frequency of about 1 KHz to 60Hz. For example, in a 62.5Hz refresh scheme, the entire display would be refreshed once every 16ms, and each digit would be illuminated for 1/8 of the refresh cycle, or 2ms. The controller must drive low the cathodes with the correct pattern when the corresponding anode signal is driven high. To illustrate the process, if AN0 is asserted while CB and CC are asserted, then a “1” will be displayed in digit position 1. Then, if AN1 is asserted while CA, CB, and CC are asserted, a “7” will be displayed in digit position 2. If AN0, CB, and CC are driven for 4ms, and then AN1, CA, CB, and CC are driven for 4ms in an endless succession, the display will show “71” in the first two digits. An example timing diagram for a four-digit controller is shown in Figure 19.

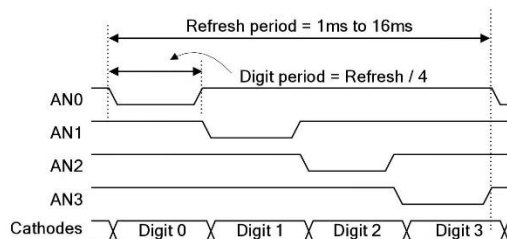


Figure 19. Four digit scanning display controller timing diagram.

10.2 Tri-Color LEDs

The Nexys4 DDR board contains two tri-color LEDs. Each tri-color LED has three input signals that drive the cathodes of three smaller internal LEDs: one red, one blue, and one green. Driving the signal corresponding to one of these colors high will illuminate the internal LED. The input signals are driven by the FPGA through a transistor, which inverts the signals. Therefore, to light up the tri-color LED, the corresponding signals need to be driven high. The tri-color LED will emit a color dependent on the combination of internal LEDs that are currently being illuminated. For example, if the red and blue signals are driven high, and green is driven low, the tri-color LED will emit a purple color.

Note: Digilent strongly recommends the use of pulse-width modulation (PWM) when driving the tri-color LEDs (for information on PWM, see section 15.1 Pulse Density Modulation (PDM)). Driving any of the inputs to a steady logic ‘1’ will result in the LED being illuminated at an uncomfortably bright level. You can avoid this by ensuring that none of the tri-color signals are driven with more than a 50% duty cycle. Using PWM also greatly expands the potential color palette of the tri-color led. Individually adjusting the duty cycle of each color between 50% and 0% causes the different colors to be illuminated at different intensities, allowing virtually any color to be displayed.

11 Pmod Ports

The Pmod ports are arranged in a 2x6 right-angle, and are 100-mil female connectors that mate with standard 2x6 pin headers. Each 12-pin Pmod port provides two 3.3V VCC signals (pins 6 and 12), two Ground signals (pins 5 and 11), and eight logic signals, as shown in Figure 20. The VCC and Ground pins can deliver up to 1A of current. Pmod data signals are not matched pairs, and they are routed using best-available tracks without impedance control or delay matching. Pin assignments for the Pmod I/O connected to the FPGA are shown in Table 5.

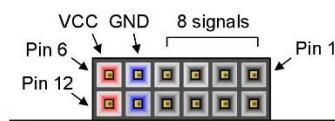


Figure 20. PMOD ports; front view, as loaded on PCB.

Pmod JA	Pmod JB	Pmod JC	Pmod JD	Pmod XDAC
JA1: C17	JB1: D14	JC1: K1	JD1: H4	JXADC1: A13 (AD3P)
JA2: D18	JB2: F16	JC2: F6	JD2: H1	JXADC2: A15 (AD10P)
JA3: E18	JB3: G16	JC3: J2	JD3: G1	JXADC3: B16 (AD2P)
JA4: G17	JB4: H14	JC4: G6	JD4: G3	JXADC4: B18 (AD11P)
JA7: D17	JB7: E16	JC7: E7	JD7: H2	JXADC7: A14 (AD3N)
JA8: E17	JB8: F13	JC8: J3	JD8: G4	JXADC8: A16 (AD10N)
JA9: F18	JB9: G13	JC9: J4	JD9: G2	JXADC9: B17 (AD2N)
JA10: G18	JB10: H16	JC10: E6	JD10: F3	JXADC10: A18 (AD11N)

Table 5. Nexys4 DDR Pmod pin assignments.

Diligent produces a large collection of Pmod accessory boards that can attach to the Pmod expansion connectors to add ready-made functions like A/D's, D/A's, motor drivers, sensors, as well as other functions. See www.digilentinc.com for more information.

11.1 Dual Analog/Digital Pmod

The on-board Pmod expansion connector labeled "JXADC" is wired to the auxiliary analog input pins of the FPGA. Depending on the configuration, this connector can be used to input differential analog signals to the analog-to-digital converter inside of the Artix-7 (XADC). Any or all pairs in the connector can be configured either as analog input or digital input-output.

The Dual Analog/Digital Pmod on the Nexys4 DDR differs from the rest in the routing of its traces. The eight data signals are grouped into four pairs, with the pairs routed closely coupled for better analog noise immunity. Furthermore, each pair has a partially loaded anti-alias filter laid out on the PCB. The filter does not have capacitors C60-C63. In designs where such filters are desired, the capacitors can be manually loaded by the user.

NOTE: The coupled routing and the anti-alias filters might limit the data speeds when used for digital signals.

The XADC core within the Artix-7 is a dual channel 12-bit analog-to-digital converter capable of operating at 1 MSPS. Either channel can be driven by any of the auxiliary analog input pairs connected to the JXADC header. The XADC core is controlled and accessed from a user design via the Dynamic Reconfiguration Port (DRP). The DRP also

provides access to voltage monitors that are present on each of the FPGA's power rails, and a temperature sensor that is internal to the FPGA. For more information on using the XADC core, refer to the Xilinx document titled "7 Series FPGAs and Zynq-7000 All Programmable SoC XADC Dual 12-Bit 1 MSPS Analog-to-Digital Converter."

12 MicroSD Slot

The Nexys4 DDR provides a microSD slot for both FPGA configuration and user access. The on-board Auxiliary Function microcontroller shares the SD card bus with the FPGA. Before the FPGA is configured the microcontroller must have access to the SD card via SPI. Once a bit file is downloaded to the FPGA (from any source), the microcontroller power cycles the SD slot and relinquishes control of the bus. This enables any SD card in the slot to reset its internal state machines and boot up in SD native bus mode. All of the SD pins on the FPGA are wired to support full SD speeds in native interface mode, as shown in Figure 21. The SPI is also available, if needed. Once control over the SD bus is passed from the microcontroller to the FPGA, the SD_RESET signal needs to be actively driven low by the FPGA to power the microSD card slot. For information on implementing an SD card controller, refer to the SD card specification available at www.sdcard.org.

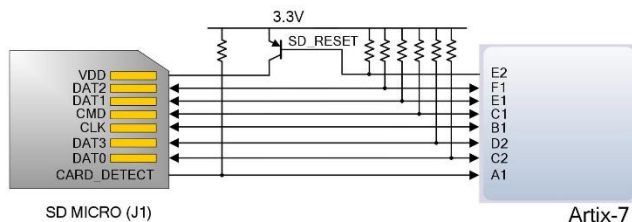


Figure 21. Artix-7 microSD card connector interface (PIC24 connections not shown).

13 Temperature Sensor

The Nexys4 DDR includes an Analog Device ADT7420 temperature sensor. The sensor provides up to 16-bit resolution with a typical accuracy better than 0.25 degrees Celsius. The interface between the temperature sensor and FPGA is shown in Figure 22.



Figure 22. Temperature sensor interface.

13.1 I2C Interface

The ADT7420 chip acts as a slave device using the industry standard I2C communication scheme. To communicate with ADT7420 chip, the I2C master must specify a slave address (0x4B) and a flag indicating whether the communication is a read (1) or a write (0). Once specifications are made for communication, a data transfer takes

place. For ADT7420, the data transfer should consist of the address of the desired device register followed by the data to be written to the specified register. To read from a register, the master must write the desired register address to the ADT7420, then send an I2C restart condition, and send a new read request to the ADT7420. If the master does not generate a restart condition prior to attempting the read, the value written to the address register will be reset to 0x00.

As some registers store 16-bit values as 8-bit register pairs, the ADT7420 will automatically increment the address register of the device when accessing certain registers, such as the temperature registers and the threshold registers. This allows for the master to use a single read or write request to access both the low and high bytes of these registers. A complete listing of registers and their behavior can be found in the ADT7420 datasheet available on the Analog Devices website.

13.2 Open Drain Outputs

The ADT7420 provides two open drain output signals to indicate when pre-set temperature thresholds are reached. If the temperature leaves a range defined by registers TLOW (0x06:0x07) and THIGH (0x04:0x05), the INT pin can be driven low or high based upon the configuration of the device. Similarly, the CT pin can be driven low or high if the temperature exceeds a critical threshold defined in TCRIT (0x08:0x09). Both of these pins need internal FPGA pull-ups when used.

For details on the electrical specifications and configuration of the INT and CT pins, refer to the ADT7420 datasheet.

13.3 Quick Start Operation

When the ADT7420 is powered up, it is in a mode that can be used as a simple temperature sensor without any initial configuration. By default, the device address register points to the temperature MSB register, so a two byte read without specifying a register will read the value of the temperature register from the device. The first byte read back will be the most significant byte (MSB) of the temperature data, and the second will be the least significant byte (LSB) of the data. These two bytes form a two's complement 16-bit integer. If the result is shifted to the right three bits and multiplied by 0.0625, the resulting signed floating point value will be a temperature reading in degrees Celsius.

For information on reading and writing to the other registers of the device, as well as notes on the accuracy of the temperature measurements, refer to the ADT7420 datasheet.

14 Accelerometer

The Nexys4 DDR includes an Analog Device ADXL362 accelerometer. The ADXL362 is a 3-axis MEMS accelerometer that consumes less than 2 μ A at a 100Hz output data rate and 270nA when in motion triggered wake-up mode. Unlike accelerometers that use power duty cycling to achieve low power consumption, the ADXL362 does not alias input signals by under-sampling; it samples the full bandwidth of the sensor at all data rates. The ADXL362 always provides 12-bit output resolution; 8-bit formatted data is also provided for more efficient single-byte transfers when a lower resolution is sufficient. Measurement ranges of ± 2 g, ± 4 g, and ± 8 g are available with a resolution of 1 mg/LSB on the ± 2 g range. The FPGA can talk with the ADXL362 via SPI interface. While the ADXL362 is in Measurement Mode, it continuously measures and stores acceleration data in the X-data, Y-data, and Z-data registers. The interface between the FPGA and accelerometer can be seen in Figure 23.

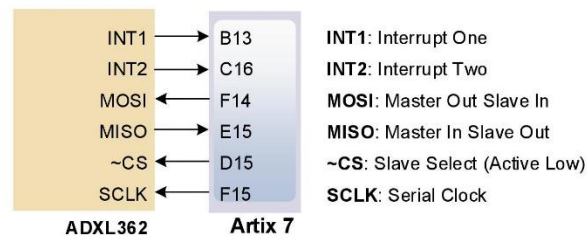


Figure 23. Accelerometer interface.

14.1 SPI Interface

The ADXL362 acts as a slave device using an SPI communication scheme. The recommended SPI clock frequency ranges from 1 MHz to 5 MHz. The SPI operates in SPI mode 0 with CPOL = 0 and CPHA = 0. All communications with the device must specify a register address and a flag that indicate whether the communication is a read or a write. Actual data transfer always follows the register address and communication flag. Device configuration can be performed by writing to the control registers within the accelerometer. Access accelerometer data by reading the device registers.

For a full list of registers, their functionality, and communication specifications, refer to the ADXL362 datasheet⁵.

14.2 Interrupts

Several of the built-in functions of the ADXL362 can trigger interrupts that alert the host processor of certain status conditions. Interrupts can be mapped to either (or both) of two interrupt pins (INT1, INT2). Both of these pins require internal FPGA pull-ups when used. For more details about the interrupts, see the ADXL362 datasheet.

15 Microphone

The Nexys4 DDR board includes an omnidirectional MEMS microphone. The microphone uses an Analog Device ADMP421 chip which has a high signal to noise ratio (SNR) of 61dBa and high sensitivity of -26 dBFS. It also has a flat frequency response ranging from 100Hz to 15 kHz. The digitized audio is output in the pulse density modulated (PDM) format. The component architecture is shown in Figure 24.



Figure 24. Microphone block diagram.

⁵ <http://www.analog.com/adxl362>

15.1 Pulse Density Modulation (PDM)

PDM data connections are becoming more and more popular in portable audio applications, such as cellphones and tablets. With PDM, two channels can be transmitted with only two wires. The frequency of a PDM signal usually falls in the range of 1 MHz to 3 MHz. In a PDM bitstream, a 1 corresponds to a positive pulse and a 0 corresponds to a negative pulse. A run consisting of all '1's would correspond to the maximum positive value and a run of '0's would correspond to the minimum amplitude value. Figure 25 shows how a sine wave is represented in PDM signal.

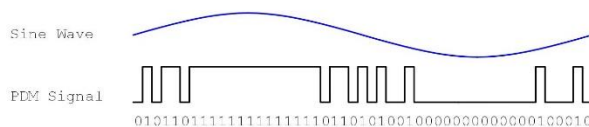


Figure 25. PDM representation of a sine wave.

A PDM signal is generated from an analog signal through a process called delta-sigma modulation. A simple idealized circuit of delta-sigma modulator is shown in Figure 26.

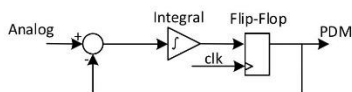


Figure 26. Simple delta-sigma modulator circuit.

Sum	Integrator Out	Flip-flop Output
0.4-0=0.4	0+0.4=0.4	0
0.4-0=0.4	0.4+0.4=0.8	1
0.4-1=-0.6	0.8-0.6=0.2	0
0.4-0=0.4	0.2+0.4=0.6	1
0.4-1=-0.6	0.6-0.6=0	0
0.4-0=0.4	0+0.4=0.4	0
0.4-0=0.4	0.4+0.4=0.8	1
0.4-1=-0.6	0.8-0.6=0.2	0

Table 6. Sigma Delta Modulator with a 0.4V_{dd} input.

To keep things simple, assume that the analog input and digital output have the same voltage range 0~V_{dd}. The input of the flip-flop acts like a comparator (any signal above V_{dd}/2 is considered as '1' and any input below V_{dd}/2 is considered '0'). The input of the integrator circuit is the difference of the input analog signal and the PDM signal of the previous clock cycle. The integrator circuit then integrates both of these inputs, and the output of the integrator circuit is sampled by a D-Flip-flop. Table 6 shows the function of the delta-sigma modulator with an input of 0.4V_{dd}.

Note that the average of the flip-flop output equals the value of the input analog signal. So in order to get the value of analog input, all that is needed is a counter that counts the '1's for a certain period of time.

15.2 Microphone Digital Interface Timing

The clock input of the microphone can range from 1 MHz to 3.3 MHz based on the sampling rate and data precision requirement of the applications. The L/R Select signal must be set to a valid level, depending on which edge of the clock the data bit will be read. A low level on L/RSEL makes data available on the rising edge of the clock, while a high level corresponds to the falling edge of the clock, as shown in Figure 27.

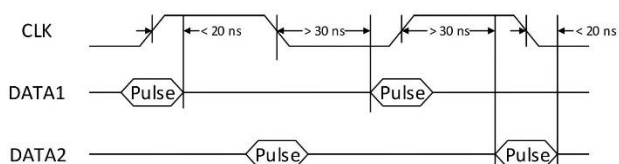


Figure 27. PDM Timing Diagram.

The typical value of the clock frequency is 2.4 MHz. Assuming that the application requires 7-bit precision and 24 KHz, there can be two counters that count 128 samples at 12 KHz, as shown in Figure 28.

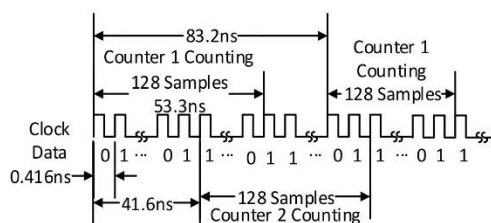


Figure 28. Sampling PDM with two counters.

16 Mono Audio Output

The on-board audio jack (J8) is driven by a Sallen-Key Butterworth Low-pass 4th Order Filter that provides mono audio output. The circuit of the low-pass filter is shown in Figure 29. The input of the filter (AUD_PWM) is connected to the FPGA pin A11. A digital input will typically be a pulse-width modulated (PWM) or pulse density modulated (PDM) open-drain signal produced by the FPGA. The signal needs to be driven low for logic '0' and left in high-impedance for logic '1'. An on-board pull-up resistor to a clean analog 3.3V rail will establish the proper voltage for logic '1'. The low-pass filter on the input will act as a reconstruction filter to convert the pulse-width modulated digital signal into an analog voltage on the audio jack output.

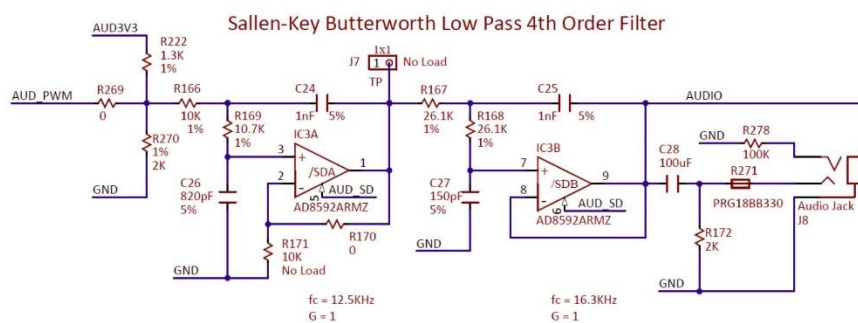


Figure 29. Sallen-Key Butterworth Low-Pass 4th Order Filter.

The frequency response of SK Butterworth Low-Pass Filter is shown in Figure 30. The AC analysis of the circuit is done using NI Multisim 12.0.

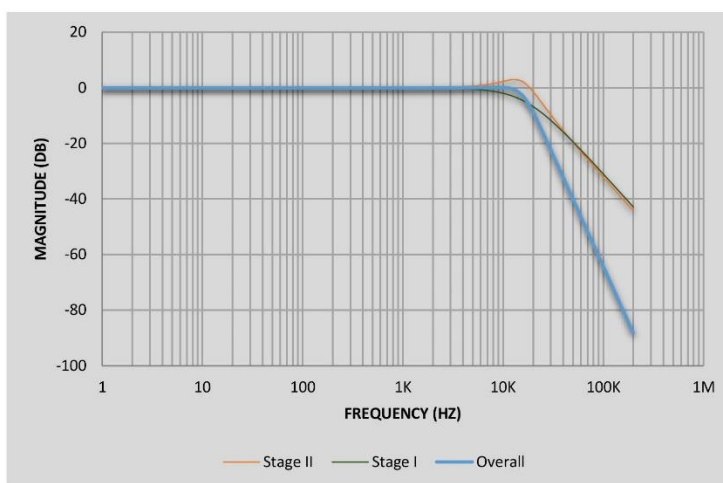


Figure 30. SK Butterworth Low-Pass Filter frequency response.

16.1 Pulse-Width Modulation

A pulse-width modulated (PWM) signal is a chain of pulses at some fixed frequency, with each pulse potentially having a different width. This digital signal can be passed through a simple low-pass filter that integrates the digital waveform to produce an analog voltage proportional to the average pulse-width over some interval (the interval is determined by the 3dB cut-off frequency of the low-pass filter and the pulse frequency). For example, if the pulses are high for an average of 10% of the available pulse period, then an integrator will produce an analog value that is 10% of the V_{dd} voltage. Figure 31 shows a waveform represented as a PWM signal.

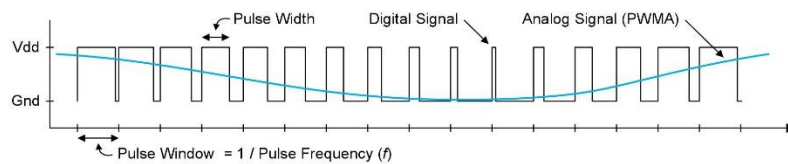


Figure 31. Simple waveform represented as PWM.

The PWM signal must be integrated to define an analog voltage. The low-pass filter 3dB frequency should be an order of magnitude lower than the PWM frequency, so that signal energy at the PWM frequency is filtered from the signal. For example, if an audio signal must contain up to 5 KHz of frequency information, then the PWM frequency should be at least 50 KHz (and preferably even higher). In general, in terms of analog signal fidelity, the higher the PWM frequency, the better. Figure 32 shows a representation of a PWM integrator producing an output voltage by integrating the pulse train. Note the steady-state filter output signal amplitude ratio to V_{dd} is the same as the pulse-width duty cycle (duty cycle is defined as pulse-high time divided by pulse-window time).

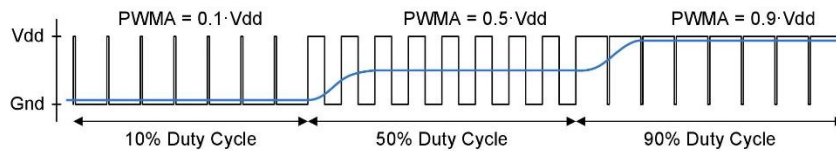


Figure 32. Representation of a PWM integrator producing an output voltage by integrating the pulse train.

17 Built-In Self-Test

A demonstration configuration is loaded into the Quad-SPI flash device on the Nexys4 DDR board during manufacturing. The source code and prebuilt bitstream for this design are available for download from the Diligent website. If the demo configuration is present in the flash and the Nexys4 DDR board is powered on in SPI mode, the demo project will allow basic hardware verification. Here is an overview of how this demo drives the different onboard components:

- The user LEDs are illuminated when the corresponding user switch is placed in the on position.
- The tri-color LEDs are controlled by some of the user buttons. Pressing BTNL, BTNC, or BTNR causes them to illuminate either red, green, or blue, respectively. Pressing BTND causes them to begin cycling through many colors. Repeatedly pressing BTND will turn the two LEDs on or off.
- Pressing BTNU will trigger a 5 second recording from the onboard PDM microphone. This recording is then immediately played back on the mono audio out port. The status of the recording and playback is displayed on the user LEDs. The recording is saved in the DDR2 memory.
- The VGA port displays feedback from the onboard microphone, temperature sensors, accelerometer, RGB LEDs, and USB Mouse.
- Connecting a mouse to the USB-HID Mouse port will allow the pointer on the VGA display to be controlled. Only mice compatible with the Boot Mouse HID interface are supported.
- The seven-segment display will display a moving snake pattern.

All Nexys4 DDR boards are 100% tested during the manufacturing process. If any device on the Nexys4 DDR board fails test or is not responding properly, it is likely that damage occurred during transport or during use. Typical damage includes stressed solder joints and contaminants in switches and buttons resulting in intermittent failures.



Stressed solder joints can be repaired by reheating and reflowing solder and contaminants can be cleaned with off-the-shelf electronics cleaning products. If a board fails test within the warranty period, it will be replaced at no cost. Contact Digilent for more details.

I. Descripción general de los sistemas empotrados

CHAPTER 8

OVERVIEW OF EMBEDDED SOC SYSTEMS

The remainder of the book applies the basic hardware design techniques learned in Part I to develop a simple and functional *embedded SoC (System on a Chip)* that contains a video subsystem and a memory-mapped I/O subsystem with general-purpose peripherals, customized hardware accelerators, and a music synthesizer. Our study is still focusing on the hardware design, but within the context of SoC, and it introduces many important design concepts, such as hardware acceleration, bus interface, and software drivers, along the way. In this chapter, we introduce the concept of an embedded SoC, discuss the development flow, explain the simple SoC framework used in this book, and provide an overview of Parts II, III, and IV.

8.1 EMBEDDED SOC

8.1.1 Overview of embedded systems

An *embedded system* (or *embedded computer system*) can be loosely defined as a computer system designed to perform one or a few specific tasks. The computer system is not the end product but a dedicated “embedded” part of a larger system that often includes additional electronic and mechanical parts. By contrast, a *general-purpose computer system*, such as a PC (personal computer), is a general computing platform and itself is the end product. We refer to it as a *desktop-like computer system* in the book. A desktop-like system is designed to be flexible and

to support a variety of end-user needs. Application programs are developed based on the available resources of the general-purpose computer system.

Embedded systems are used in a wide range of applications and each application has its own specific requirements. On one hand, a “low-end” system, such as a microwave oven, involves only a simple control function and can be implemented by an 8-bit single-chip microcontroller. On the other hand, a “high-end” system, such as a digital camera, is more complex. It performs two major tasks. The first task involves the general “housekeeping” I/O operations, including processing the button and knob activities, generating a menu on an LCD display, and writing image files to the storage device. These operations are more involved than those of a microwave oven and the system requires a more capable processor. The second task is to process the image and perform data compression to reduce the file size. Because of the large number of pixels and the complexity of the compression algorithm, the task requires a significant amount of computation. An embedded processor is usually not powerful enough to handle the computation-intensive operation. A custom digital circuit, sometimes known as a *hardware accelerator*, can be designed to perform this particular task and take the load off the processor.

8.1.2 FPGA-based SoC

A “high-end” embedded system usually has a processor and simple I/O peripherals to perform general user interface and housekeeping tasks and special hardware accelerators to handle computation-intensive operations. These components can be integrated into a single integrated circuit, commonly referred to as an *SoC (system on a chip)*.

As the capacity of FPGA devices continues to grow, the same design methodology can be realized in an FPGA chip. Instead of just realizing the system functionalities by *customized software*, we can incorporate *customized hardware* into the embedded system as well. The FPGA technology allows us to tailor the processor, select only the needed I/O peripherals, create a custom I/O interface, and develop specialized hardware accelerators for computation-intensive tasks. The FPGA embedded system provides a new dimension of flexibility because both the hardware and software can be customized to match specific needs. The methodology of exploiting the trade-offs between hardware and software and developing and integrating them concurrently is referred to as *hardware-software co-design*.

8.1.3 IP cores

In SoC development, systems frequently have certain common functionalities and the same building blocks can be reused in different designs. These components are known as *IP (intellectual property) cores*, or simply as *IPs*. They are somewhat like functions in a software library, which can be used in different application programs. The IP cores can be developed by the device manufacturers, third-party vendors, or the users themselves. Unlike software functions, FPGA vendor’s IP cores are usually tailored for their own proprietary platforms. They are not portable and frequently delivered as “black boxes” (i.e., without HDL source codes). For example, all companies provide FFT (fast Fourier transform) IP cores. While the cores perform similar functions, their interfaces, timing characteristics, and configuration

procedure are different. Therefore, a system must be redesigned or modified if it is re-targeted to a device from a different vendor.

8.2 DEVELOPMENT FLOW OF THE EMBEDDED SOC

The embedded SoC design consists of the following tasks:

- Partition the tasks to software routines and hardware accelerators.
- Design user custom IP cores if needed.
- Develop the hardware.
- Develop the software.
- Implement the hardware and software and perform testing.

These tasks are discussed in the following subsections.

Because of the complexity of modern digital systems, pre-designed IP cores are used extensively in SoC development. Each vendor has its own IP framework, which provides a comprehensive collection of IP cores and supporting software device drivers. The development flow is frequently centered on the IP cores and is shown in Figure 8.1.

8.2.1 Hardware–software partition

Step 1 (labeled 1 in the diagram) is to determine the software–hardware partition. An embedded application usually performs a collection of tasks. In an SoC-based design, a task can be implemented by hardware, software, or both. Based on the performance requirement, complexity, and hardware core availability, we can decide the type of implementation accordingly.

In an ideal scenario, the vendor IP library contains all the needed IP cores for the SoC design. However, in reality, most designs require a certain number of custom IP cores for hardware accelerators and special I/O peripherals. Step 2 is to develop the hardware codes and the corresponding software drivers of these custom IP cores. The details are discussed in Subsection 8.2.5.

8.2.2 Hardware development flow

The left branch represents the hardware design flow. Step 3 is to utilize and integrate the IP cores to construct the system. In Vivado Design Suite, it is done by the IP integrator utility. A user can select IP cores, configure them with the desired characteristics, and connect the cores with a proper interface. IP Integrator will invoke the cores from the library and generate the HDL codes. The top-level HDL file usually resembles the top-level block diagram of an SoC design. During the generation, IP Integrator also produces an auxiliary *hardware platform specification file*, which contains the “definition” of the SoC design, including the processor configuration, memory size and structure, I/O peripheral cores used, memory address mapping, etc.

The top-level HDL file can be treated as a normal HDL file and processed accordingly. Steps 4 and 5 perform synthesis and placement and routing and eventually generate the FPGA configuration file (i.e., the `.bit` file).

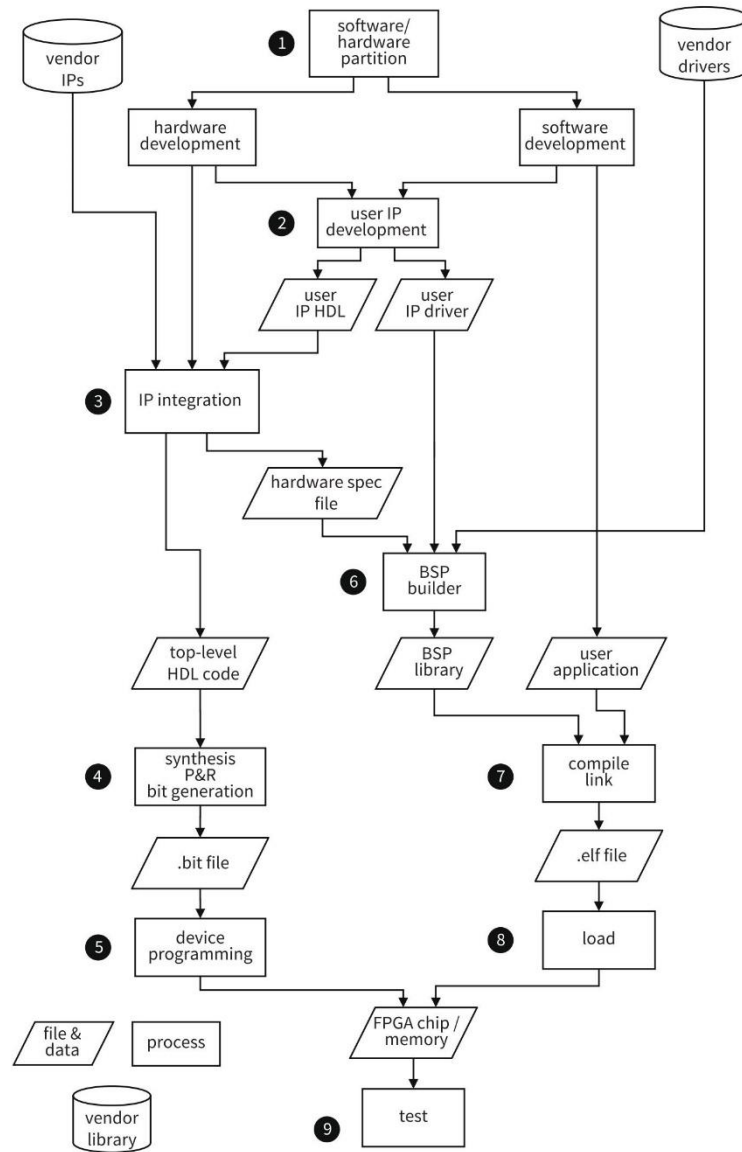


Figure 8.1 IP-centered SoC development flow

8.2.3 Software development flow

The right branch represents the software design flow. A top-level software program usually contains two types of codes. One type is the “system codes,” which are pre-designed and provided with the system. They can be functions from libraries, service routines from the operating system, etc. The other type is the “application codes,” which are developed by the user to perform the custom tasks. These system functions and service routines are called by the application codes.

In an embedded system, a *BSP (board support package)* is a mechanism to encapsulate the system codes. Since an embedded system is designed to perform a specific task, each system has a different memory structure and contains a unique set of I/O peripherals. BSP is a customized collection of device drivers and initialization routines that support a particular system. The term “board” is used since earlier embedded systems were implemented in a printed circuit board rather than a single silicon device.

An important ingredient of a BSP is the *device drivers*. A device driver is a set of routines that operate or control a particular peripheral device. A driver acts as a “translator” between the hardware peripheral and application programs and enables the application programs to access peripheral functions without needing to know precise details. If an IP core is expected to interface with a processor, a device driver should be developed concurrently.

The embedded SoC flow adopts the BSP mechanism. Step 6 is to build the BSP according to the SoC hardware configuration. Recall that, in Step 3, IP integrator generates a hardware platform specification file. The BSP builder utility examines the IP core information, extracts the pre-build device drivers and initialization routines from the vendor software library, and creates the BSP for the specific SoC design.

The application program can invoke the driver routines in the BSP to access the peripheral I/O cores. Step 7 compiles and links the software routines and BSP library and builds the final software image file (i.e., the .elf file).

8.2.4 Physical implementation and test

Physically implementing the system involves two steps. We first download the FPGA configuration file to the FPGA device (i.e., “program” the device), as in Step 5, and then load the software image into processor’s main memory, as in Step 8. The physical system can be tested afterward, as in Step 9.

8.2.5 Custom IP core development

Although FPGA vendors provide a comprehensive collection of pre-designed IP cores, they seldom can cover all the project needs. We usually have to design custom IP cores for special I/O peripherals or less common computation algorithms. The development consists of three tasks:

- Design a custom digital circuit to implement the computation algorithm or special functionality.
- Derive an interface to connect the circuit to the bus or interconnect structure of the vendor’s IP framework.

- Develop a device driver to control the new hardware core and integrate it into vendor's software library.

Note that the latter two activities depend on the FPGA vendor's IP platform. We need to carefully study the platform's interface protocols and driver structure so that the IP core can be integrated into vendor's framework and used in the IP integration utility. Since each vendor has its own proprietary IP platform, the interface and driver are not portable and must be re-designed for each vendor.

8.3 FPRO SOC PLATFORM

8.3.1 Motivations

While the embedded SoC is powerful methodology, it is not the emphasis of this book. First and foremost, this book focuses on the register-transfer level hardware design rather than the system-level analysis and integration. In addition, a commercial IP platform is not ideal for learning introductory hardware design for several reasons:

- A commercial IP platform is quite complex and thus a significant amount of time will be spent on learning to use the tool rather than doing design.
- Most commercial IP cores are provided as black boxes.
- The interface protocol and driver structure are quite complex.
- The IP framework is proprietary. Thus, learning is tied to a particular platform and the developed IP cores are not portable.

In this book, we define a simple SoC platform and call it *FPro SoC*, (which is abbreviated from the book title “FPGA Prototyping” or can be interpreted as “Fun and Professional”). It contains a video subsystem and a memory-mapped I/O subsystem with general-purpose peripherals, customized hardware accelerators, and a music synthesizer. Our study is still focusing on the hardware design but within the SoC context. The main characteristics of the FPro SoC platform are as follows:

- *Simple.* The FPro SoC platform defines a simple synchronous bus protocol and a straightforward device driver structure. Once a hardware circuit is developed, it can be converted to an IP core by adding a simple interface circuit and a device driver. The core then can be incorporated into the existing embedded system.
- *Functional.* FPro SoC platform provides a variety of I/O peripherals and commonly used serial interfaces (UART, SPI, and I²C) and includes working device drivers. It resembles a bare-metal 32-bit microprocessor board and can implement real-world projects targeted for this type of boards.
- *Portable.* Except for the processor, FPro SoC's IP cores are developed from scratch in HDL and do not use any vendor's proprietary components. The bus protocol and device drivers are not tied to any specific commercial platform, either. Thus, the IP cores and software codes are portable and can be reused for different FPGA devices and prototyping boards.
- *“Upward compatible.”* While the FPro SoC platform is simple, the development follows rigorous and proven design principles and practices. These knowledge and skills can be applied in the future for more complicated commercial platforms and larger projects. In fact, the IP cores and drivers devel-

oped can be easily modified to be incorporated into existing commercial IP frameworks.

- *Fun.* Because the developed system is like a real microprocessor board, it can incorporate existing I/O modules and quickly develop a functional prototyping project. In addition, this platform can provide hardware acceleration capability and thus is more capable and more flexible than any microprocessor board. This give us an opportunity to develop interesting and challenging projects and make studying hardware more “fun” rather than “learning hardware for the sake of hardware.”

8.3.2 Platform hardware organization

The top-level diagram of an FPro system is shown in Figure 8.2. It is composed of four major parts:

- Processor module
- FPro bridge and FPro bus
- MMIO (memory mapped I/O) subsystem
- Video subsystem

We only use vendor’s IP cores for the processor, memory controller, line buffer, and clock management circuit, which are shown as dotted gray boxes in the figure, and construct all other cores from scratch.

Processor module The processor module consists of a processor, a memory controller core, and RAM. It is the part that is constructed from the vendor’s IP cores. To be used in the FPro SoC platform, the processor core must exhibit the following characteristics:

- 32-bit-wide data path
- 32-bit memory address space
- *Memory-mapped-I/O* scheme for I/O access

Almost all FPGA-based processors support these features. There is no restriction on types of RAM. It can be FPGA’s internal memory modules or external memory devices. However, since an FPro system resembles an entry-level 32-bit embedded system, we assume that the size of RAM is limited and software is developed in this context.

FPro bridge and FPro bus The processor needs to communicate with other cores. This is done by a *bus* or *interconnect structure* specified in the vendor’s IP platform. The modern interconnect is designed to accommodate a wide variety of communication and data transfer needs and involves complex protocols. For our learning purposes, we define a simple synchronous bus protocol for the two subsystems and call it *FPro bus*. The *FPro bridge* converts vendor’s native bus signals into FPro bus signals. The FPro bus protocol and bridge are discussed in Chapter 10.

MMIO subsystem In the memory-mapped-I/O scheme, the memory and registers of the I/O peripherals are mapped to the same address space. This means that the processor makes no distinction between the memory and I/O peripherals and uses the same read and write instructions to access the I/O peripherals.

The MMIO subsystem provides a framework to accommodate memory-mapped general-purpose and special I/O peripherals as well as hardware accelerators. For

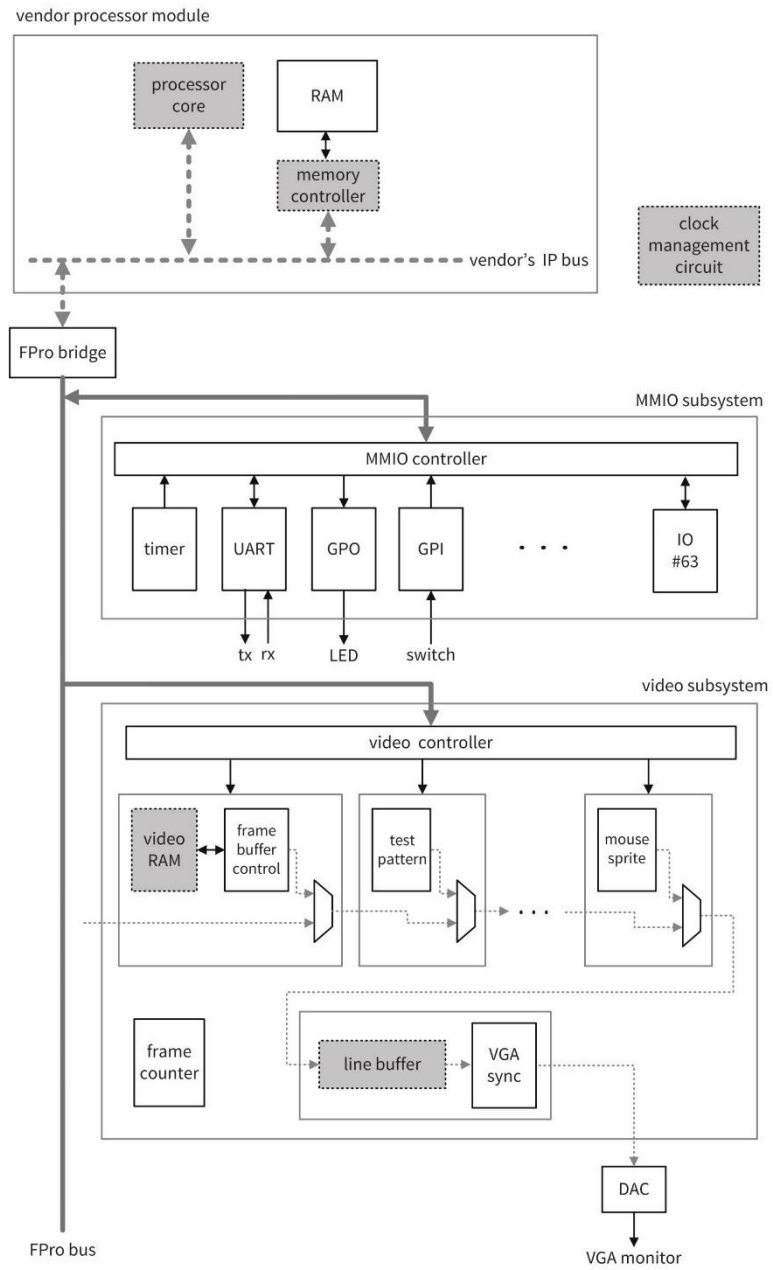


Figure 8.2 Top-level diagram of an FPro system

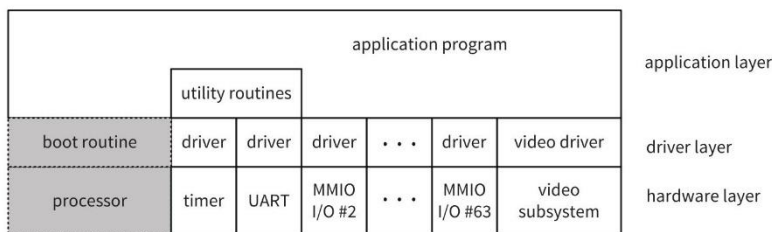


Figure 8.3 Software hierarchy of an FPro SoC system

simplicity, we define a standard *slot interface* that conforms to the FPro bus protocol. The MMIO subsystem consists of a controller to select a specific slot and can accommodate up to 64 instantiated cores. After being “wrapped” with an interface circuit, custom digital logic can be plugged into the FPro platform. About a dozen IP cores are developed and integrated into the MMIO subsystem in the subsequent chapters.

Video subsystem The video subsystem establishes a framework to coordinate the operation of video cores. A video core generates or processes the video data stream. The cores are arranged as a cascading chain. The data stream is pipelined and “blended” through each stage and eventually displayed on a VGA monitor. The video subsystem demonstrates the principles of handling *stream data*, in which data are generated continuously and passed through a chain of components for processing.

8.3.3 Platform software organization

Since the book focuses on hardware design, we use a simple *bare metal* software scheme for the system. A bare metal system contains no operating system. In its simplest form, the processor boots directly into an infinite main loop, which contains functions to check input, perform computation, and write outputs.

The software hierarchy of an FPro system is shown in Figure 8.3. It contains a *hardware layer*, a *driver layer*, and an *application layer*. A *boot routine* is associated with the processor. It first performs the basic initialization process, such as clearing the caches, configuring the stack and heap segments, and initializing the interrupt, and then transfers control to the main program. The codes are obtained from the vendor, as shown in a dotted gray box in the figure. All other device drivers are constructed from scratch.

To facilitate the software development, we develop several simple utility routines that maintain a system time and assist displaying a debug message on the console. The timer core and UART (universal asynchronous receiver and transmitter) core in slots 0 and 1 are used for this purpose, as shown in Figure 8.2. Thus, the two cores should always be instantiated in the first two slots and not be replaced.

Every I/O core in the FPro system is accompanied by a driver. We select C++ for driver development because of its support of data encapsulation. A C++ class will be created for each core.

Except for accessing system time (via a timer core) and sending debugging messages (via a UART core), a class is largely “self-contained” and does not interact with other classes. When a core is attached or removed from an FPro system, the corresponding driver files should be included or deleted from the software projects. In the main application program, an instance will be created for each instantiated IP core and the methods in the class will be used to access and control the core. The “state” of the core, if existing, is kept within the private section of the instantiated object and involves no external variables.

8.3.4 Modified development flow

The original development flow shown in Figure 8.1 needs to be revised to accommodate the FPro SoC platform. While the basic procedure remains unchanged, we need to manually construct the top-level HDL code and manually include the device driver files in our software application. The modified flow is shown in Figure 8.4 and the new paths are highlighted as thick gray dotted lines in the top half. The main changes are as follows:

- In Step 3, only the processor module, which contains a processor core and RAM, is generated via the IP integration utility. We must manually construct the HDL code for the top-level system, which is composed of the instantiation of the previously generated processor module and the MMIO and video subsystems from Step 2.
- In Step 6, since only the processor module configuration is listed in the hardware specification file, only processor-related codes, such as the boot routine, will be included in the BSP library. We must manually examine the IP cores in the top-level HDL file and include the corresponding driver files in the application software project.
- Since the processor module is the same most of the time, Steps 1 and 6 only need to be executed once. The generated HDL files and BSP library can be used in subsequent designs.

8.4 ADAPTATION ON THE DIGILENT NEXYS 4 DDR BOARD

The book uses the Digilent Nexys 4 DDR prototyping board, which is designed around the Xilinx Artix 7 XC7A100T device, for the experiments and projects. Xilinx provides a soft-core processor, known as *MicroBlaze*, as well as a completely “pre-configured” system, known as *MicroBlaze MCS* (for *MicroBlaze Micro Controller System*). We select MCS as the processor module in Figure 8.2.

MicroBlaze is a 32-bit FPGA-based processor with RISC (reduced instruction set computer) architecture. It is highly configurable and can incorporate an optional floating-point unit, instruction and data caches, a memory management unit, etc. MicroBlaze mainly uses the *AXI* (*Advanced eXtensible Interface*) protocols from ARM to interface with other IP cores. Hundreds of IPs from Xilinx and third-party vendors, including memory controllers, I/O peripherals, and various types of hardware accelerators, can be integrated with a MicroBlaze to form an SoC design. The flow in Figure 8.1 is targeted for this type of setting.

MicroBlaze MCS is a *complete computer system* that is composed of a pre-configured MicroBlaze processor, a RAM constructed with FPGA’s internal mem-

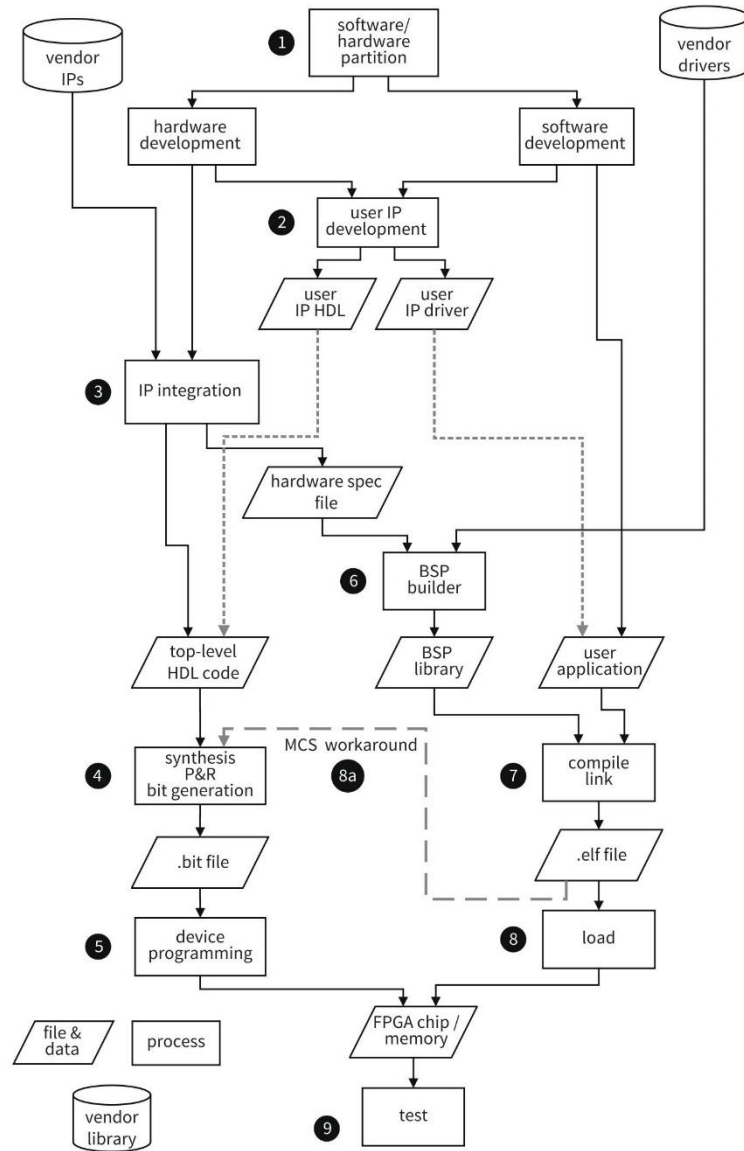


Figure 8.4 FPro SoC development flow

ory, and an I/O module with a standard set of microcontroller peripherals. MicroBlaze MCS provides only a limited degree of configurability. A user can set the size of RAM (between 8 KB and 128 KB) and select a small set of simple I/O peripherals.

Since the focus of the book is hardware design rather than system-level integration, MicroBlaze MCS serves the purpose very well. In addition, many simpler prototyping boards use Xilinx's earlier Spartan devices and must use ISE WebPack for development. At the time of writing, MicroBlaze MCS is free across all Xilinx platforms, including both the ISE WebPack edition and Vivado WebPack edition, but the full-featured MicroBlaze processor is only free for the Vivado WebPack edition. Thus, MicroBlaze MCS can be adopted by more entry-level prototyping boards.

On the down side, support for MicroBlaze MCS is not as comprehensive. The Vivado 2016 edition is used at the time of writing. Step 8 in Figure 8.4 does not function properly. The workaround is to associate the `.elf` file as the "initial values" of FPGA's internal memory and regenerate the configuration file (i.e., `.bit` file). The approach is shown as a thick dashed line in the bottom of Figure 8.4. The revised flow becomes:

- Develop and implement hardware (Steps 1 to 4).
- Develop and implement software (Steps 2 to 7).
- Associate the `.elf` file in the hardware project (Step 8a).
- Regenerate the configuration `.bit` file with the embedded `.elf` file (i.e., repeating Step 4).
- Program the FPGA device and perform testing (Steps 5 and 9).

This flow is less ideal since regenerating the `.bit` file for each software revision is time-consuming and Vivado software must be invoked during software development.

XC7A100T is a fairly large device and its internal memory modules can accommodate 128 KB RAM for MicroBlaze MCS and 350 KB video RAM for a 9-bit VGA frame buffer. Thus, no external memory device is involved.

8.5 PORTABILITY

A main goal of this book is to develop a portable system to learn hardware design and to introduce SoC practice. Because of the proprietary development software, the IP platform, and IP cores, it is difficult to construct a complete device- and board-independent FPGA-based SoC system. The experiments and projects in this book are constructed and tested on a specific board (Digilent Nexys 4 DDR) that contains a specific FPGA device (Xilinx Artix 7 XC7A100T). The following subsections discuss portability issues.

8.5.1 Processor module and bridge

Since the processor module is constructed from the vendor's proprietary IP cores, it potentially introduces several portability issues:

- Processor
- Memory controller and RAM
- Interface and bridge
- Loading and booting of software

The FPro platform basically requires a 32-bit processor core that supports memory-mapped I/O scheme. Almost all FPGA-based processors satisfy this requirement. The internal and external memory sizes and configurations can vary significantly among different FPGA devices and prototyping boards. However, since the RAM inside the processor module only interacts with the processor core, it does not affect the subsystems directly. In summary, although the proprietary and different memory configurations are used in the processor module, they will not cause serious compatibility issues. The simplest way to create the processor modules is to utilize FPGA's internal memory, as in MicroBlaze MCS. However, older and simpler FPGA devices provide less internal memory. MicroBlaze MCS can be configured with smaller RAM. The size of the RAM, of course, sets the limit on the size of application program.

The FPro bus protocol is designed for simple non-burst synchronous read and write transactions. It can be considered as a very small subset of existing full-featured bus interfaces. Designing a bridge is not very difficult.

While compiling and linking the software code follows a similar tool chain, there is no standard procedure to load an `.elf` file (Step 8 in the development flow). The process depends on the device, memory configuration, prototyping board, and software development platform. We need to consult the specific manual or user guide to complete this task.

8.5.2 MMIO subsystem

Since the MMIO subsystem's controller and the attached IP cores are constructed from scratch and use no vendor's proprietary IPs or components, the HDL codes are completely portable. They can be implemented as long as a prototyping board has adequate external peripherals. The only exception is the Artix's built-in ADC (analog to digital converter), known as *XADC*, which is only available for newer Xilinx devices.

Since the Nexys 4 DDR board contains all the needed peripherals, all MMIO IP cores can be implemented and tested without any external component. Some peripherals may not be available on other prototyping boards. However, the external circuitries are quite simple and can be easily implemented on a breadboard. The schematics for these peripherals can be found in the Nexys 4 DDR on-line manual and reconstructed accordingly.

8.5.3 Video subsystem

While the majority of the video subsystem is designed from scratch, three components – clock management circuit, line buffer, and frame buffer – utilize vendor's proprietary IP cores. The clock management circuit and line buffer accommodate the VGA synchronization, whose clock rate is different from system clock rate. The former requires a PLL (phase-locked loop) like macro cell and the latter is based on a dual-clock FIFO buffer macro cell. Although these macro cells are proprietary, they are common and can be found in all FPGA devices. The proper macro cells can be instantiated in HDL code directly. Thus, the clock management circuit and line buffer do not lead to serious portability issues.

The frame buffer tends to be the most troublesome and least portable IP core in the FPro framework. The key part of the frame buffer is a dual-port memory that is

accessed by the processor and frame control. The latter retrieves data from memory and converts the data into a video stream. The buffer requires a substantial amount of RAM and thus should be implemented by external memory devices. This raises several issues:

- FPGA prototyping boards have different types of memory devices and configurations and some simpler boards may have none.
- Except for simple SRAM devices, a sophisticated proprietary memory controller IP core is needed.
- The frame buffer control must interface with the proprietary memory controller and implement the dual-port access control circuit.
- The same external memory device may be used as processor's RAM and frame buffer at the same time. The partition further complicates the interface and configuration.

Thus, it is difficult to construct a portable frame buffer.

To demonstrate the design principle, the book uses FPGA's internal memory for the video memory. 350 KB of internal RAM is allocated for a frame buffer with a 9-bit VGA resolution. This is doable because the Nexys 4 DDR board contain a large XC7A100T device. It cannot be duplicated in boards with smaller devices. One possible alternative is to reduce the color depth from 9 bits to 1 bit.

Some advanced prototyping boards use HDMI port for the video output. Instead of using a DAC to generate the analog signal, the HDMI interface encodes the output from the line buffer, "serializes" the data, and transmits the video signal digitally through three serial lines. Thus, additional circuits must be added to accommodate the new interface.

8.6 ORGANIZATION

The remaining book consists of three parts. The rest of Part II provides an overview of the hardware architecture and the bare metal embedded software development via the construction of the *vanilla FPro system*, which contains a timer core, a UART core, a GPI (general-purpose input) core, and a GPO (general-purpose output) core. The conceptual diagram is shown in Figure 8.5.

Part III shows how to design an array of MMIO cores for the peripherals on the Nexys 4 DDR prototyping board, including a PWM (pulse width modulation) core, a debouncing core, a seven-segment LED core, a Xilinx XADC controller core, an SPI core, an I²C core, a PS2 core, and a music synthesis module with a DDFS (direct digital frequency synthesis) core and an ADSR (attack-decay-sustain-release) envelope core. Part IV discusses the video subsystem framework and covers construction of relevant IP cores.

8.7 BIBLIOGRAPHIC NOTES

Embedded systems encompass a spectrum of design issues. The two books, *Embedded System Design: A Unified Hardware/Software Introduction* by F. Vahid and T. D. Givargis and *Computers as Components: Principles of Embedded Computing System Design, 2nd edition*, by W. Wolf, provide a comprehensive discussion. Software-hardware co-design is an emerging research area. *A Practical Introduction*

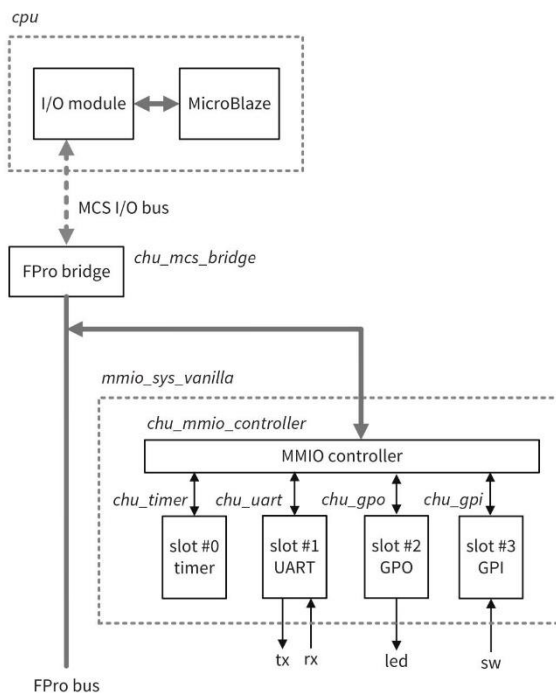


Figure 8.5 Vanilla FPro system.

to Hardware/Software Codesign by P. R. Schaumont addresses the basic concepts and issues of combining hardware and software into a single system.