



INAOE

Instrumentación de un arreglo de microbolómetros de 120x160

por

Julisa Verdejo Palacios

Tesis presentada en cumplimiento parcial de los requisitos para
el grado de:

Maestría en Ciencias en la Especialidad de Electrónica

en

Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE)

Agosto, 2024

Santa María Tonantzintla, Puebla

Asesor:

Dr. Mario Moreno Moreno
Departamento de Electrónica INAOE

©INAOE 2024

Todos los derechos reservados

El autor otorga al INAOE el permiso para reproducir y
distribuir copia de esta tesis en su totalidad o en partes
mencionando la fuente.



Agradecimientos

- Esto es un ejemplo de agradecimiento.
- Este es otro agradecimiento.

Índice general

Agradecimientos	I
Índice general	IV
Índice de figuras	V
Índice de tablas	VII
Índice de códigos	IX
Resumen	XI
1. Introducción	1
1.1. Estado del arte	1
1.2. Objetivos	1
1.2.1. Objetivo general	1
1.2.2. Objetivos específicos	1
2. Microbolómetros y descripción de especificaciones	3
3. Sistema de adquisición de datos DAQ	5
3.1. Protocolos de comunicación	5
3.2. Tipos de ADC	5
3.3. Tipos de DAC	5
3.4. PCBs	5
3.5. Ruido	5
4. Diseño de firmware	7
5. Mediciones y resultados	9
6. Conclusiones	11

A. Códigos	13
A.1. Códigos de Verilog	13
A.2. Códigos de Verilog	13
Bibliografía	15

Índice de figuras

2.1. Jitter del reloj.	3
2.2. FSM template.	4

Índice de tablas

2.1. Parámetros recomendados para el conjunto de pruebas del NIST. . . .	4
--	---

Índice de códigos

A.1. Comunicación RS232	13
A.2. Máquina de estados ejemplo.	13

Resumen

Capítulo 1

Introducción

1.1. Estado del arte

Los microbolómetros son bonitos y [1]

Porque los sistemas de adquisición de datos son tan importantes en la ciencia Cuales son las dificultades de realizar uno (depende de las especificaciones, el tiempo y el capital humano) Historia de los sistemas de adquisición de datos Ejemplos de sistemas de adquisición de datos Aplicaciones de sistemas de adquisición de datos Puedes enfocarlo al microbolometro o en general

1.2. Objetivos

1.2.1. Objetivo general

- Diseño de un sistema de adquisición de datos basado en FPGA para la medición de una matriz de pixeles de un microbolómetro.

1.2.2. Objetivos específicos

- Obtención de especificaciones para el sistema de adquisición de datos.
- Selección de ADC y DAC y protocolo de comunicación a partir de las especificaciones.
- Diseño de un firmware en el lenguaje de descripción de hardware (HDL) Verilog para el protocolo de comunicación SPI reconfigurable y robusto.
- Implementación de protocolo SPI para control de un ADC y un DAC de 12 bits.

- Pruebas experimentales y de estrés para verificación de robustez del diseño y mejora.
- Adquisición de los datos utilizando una terminal de usuario basada en UART.

Capítulo 2

Microbolómetros y descripción de especificaciones

En este capítulo se presentan los conceptos necesarios para comprender cómo funcionan las diferentes clases de generadores de números aleatorios, sus fuentes de aleatoriedad, sus principales características y cuales son factibles para implementación en FPGA.

El jitter del reloj en un sistema digital es una desviación del flanco de reloj real con respecto a un flanco de reloj ideal. Una señal de reloj ideal se define mediante la ecuación (2.1), donde $t(n)$ representa el tiempo del periodo n -ésimo de una señal de reloj y T es el periodo de una señal de reloj.

$$t(n) = n \cdot T \quad (2.1)$$

En la práctica, una señal de reloj real no alcanza siempre múltiplos enteros de su periodo ideal, sino que sus flancos fluctúan alrededor de este valor debido al jitter. Esta variación es causada por diversos fenómenos físicos, como el ruido térmico, el ruido de la fuente de alimentación y el ruido electromagnético ambiental, entre otros. La Figura 2.1 muestra cómo se ve una señal de reloj afectada por el jitter.

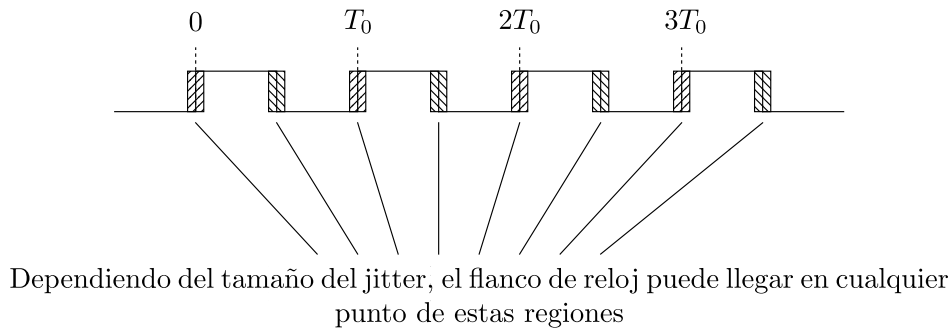


Figura 2.1: Jitter del reloj.

Inputs: START, OPC
 Outputs: EOF, FLAGt

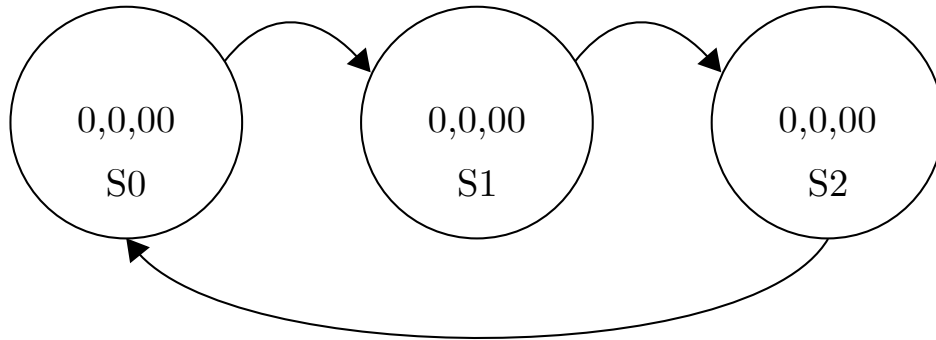


Figura 2.2: FSM template.

Los parámetros recomendados para configurar las pruebas NIST se muestran en la Tabla 2.1:

Tabla 2.1: Parámetros recomendados para el conjunto de pruebas del NIST.

Test	Configuration item	Setting
All tests	Bits per sequence	1000000
All test	Number of sequences (sample size)	1073
Frequency test within a block	Block length	20000
Non-overlapping template test	Template length	10
Overlapping template	Block length	10
Mauler's Universal Statistical test	Test block length L	7
Mauler's Universal Statistical test	Initialization steps	1280
Approximate entropy test	Block length	8
Linear complexity test	Block length	1000
Serial test	Block length	16

Capítulo 3

Sistema de adquisición de datos DAQ

3.1. Protocolos de comunicación

3.2. Tipos de ADC

3.3. Tipos de DAC

3.4. PCBs

3.5. Ruido

En este capítulo se estudiarán los mapas caóticos utilizando como ejemplo el mapa logístico para entender cómo produce el caos y las diferentes técnicas para ver de manera cualitativa este fenómeno.

Capítulo 4

Diseño de firmware

Capítulo 5

Mediciones y resultados

Capítulo 6

Conclusiones

- Parrafo.
- Parrafo.
- Parrafo.

Apéndice A

Códigos

A.1. Códigos de Verilog

Código A.1: Comunicación RS232

```
%% prog04_graficas_multiples
clear; close all; clc;
x = 0:0.1:2*pi;
y1 = sin(x); y2 = cos(x);

subplot(2,1,1);
plot(x,y1,'-sr','DisplayName','f(x) = sin(x)');
axis([min(x) max(x) min(y1)*1.1 max(y1)*1.1]);
grid on; grid minor;
legend('Location','northeast','FontSize',12);
title('f(x) = sin(x)'); xlabel('x'); ylabel('f(x)');

subplot(2,1,2);
plot(x,y2,'-sk','DisplayName','f(x) = cos(x)');
axis([min(x) max(x) min(y2)*1.1 max(y2)*1.1]);
grid on; grid minor;
legend('Location','southeast','FontSize',12);
title('f(x) = cos(x)'); xlabel('x'); ylabel('f(x)');
```

A.2. Códigos de Verilog

Código A.2: Máquina de estados ejemplo.

```
module counter #(
    parameter Width = 8
) (
    input        clk_i,
    input        rst_i,
    output       max_tick_i,
    output [Width-1:0] q_o
);

    reg [Width-1:0] reg_q;
    wire [Width-1:0] sum_d;

    assign sum_d = reg_q + 1;
    assign max_tick_i = (reg_q == 2**Width-1) ? 1'b1 : 1'b0;

    always @(posedge clk_i, posedge rst_i) begin
        if (rst_i)
            reg_q <= 0;
        else
            reg_q <= sum_d;
        end

    assign q_o = reg_q;
endmodule
```

Bibliografía

- [1] J. Hernandez, J. Rangel-Magdaleno, R. Jimenez, M. Moreno, A. Torres, A. Ponce, D. Ferrusca, and J. Castro-Ramos, “An automated v-i acquisition system for microbolometer array with FPGA-based drive,” in *2021 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, IEEE, may 2021.