

**LABORATORIO DE SISTEMAS DIGITALES II  
INFORME DE PROYECTO**

**Sistema de Registro y Alarma de Resistencia a los  
Antimicrobianos (RAM) en los pacientes en  
Hospitales Públicos del Ecuador.**

**ESCUELA SUPERIOR POLITECNICA DEL  
LITORAL  
FIEC**

**INTEGRANTES GRUPO 4:**

**ADRIANA PONCE**

**RICARDO MUÑOZ OCHOA**

**PROFESORAS:**

**PARALLEO 102-ALLISON CONSTANTINE**

**PARALELO 1-SARA RÍOS**

**2DO TÉRMINO 2021-2022**

## INTRODUCCIÓN

La temática de este proyecto es la búsqueda del número X que más se repite, es decir, las frecuencias absolutas máximas, aplicado en Salud Pública, con afán de procesar una base de datos de que contengan información de interés para registrar indicadores de las poblaciones más propensas a la Resistencia a los Antimicrobianos (RAM), lo cual hace a los pacientes resistentes a los medicamentos que los protegen de infecciones a causa de virus como el E.coli.

## ANTECEDENTES

El objetivo principal del proyecto es prevenir el consumo excesivo de antimicrobianos en los hospitales públicos del Ecuador, en la última investigación médica realizada en el hospital de Quito, se evidenció en los pacientes una resistencia a las bacterias E.coli y K muy elevada, incluso llegando al 79%. Siendo estas causantes de infecciones intestinales, problemas respiratorios e infección de torrente sanguíneo. [1] [2]

Paralelo al manejo del contagio por COVID-19, el consumo de antimicrobianos incrementó y esto influencia al sistema inmunológico del paciente a ser tolerante al medicamento conforme entre más en contacto con este. Por ello, la revista científica INSPIP, financió una investigación este 27 de mayo, dado al riesgo que implica el RAM. [1]

En Estados Unidos también actualmente, a raíz del consumo de antimicrobianos por pandemia, se quiso aumentar la eficiencia de la vigilancia a la administración de estos, cuando son necesarios o que casos presentan futura inmunidad al medicamento. Un artículo publicado en NPJ Digital Medicine explica como usando otros artículos, sobre aumento de resistencia del PHS (fármaco estándar para eliminar bacterias), fueron armando una base de datos de la cual hicieron pasar dato a dato por bloques de: Identificación, Presentación, Selección e Inclusión, donde gestionaban los casos con las enfermedades objetivas: historial médico, dosis administradas, grupo familiar, etc. Centrándose en la frecuencia del antimicrobiano. [5]

No obstante, Latinoamérica no se queda atrás, frente al impacto del COVID-19, tratado también con antimicrobianos, se financio un estudio en la Universidad Antonio José Camacho, Colombia, en 2020, donde ilustraban la implementación de los FPGA para contrarrestar años de mala administración en historial médicos, equipos médicos y fármacos. Principalmente, se usaron sistemas digitales para mapear las características, síntomas y zonas de concentración del en ese entonces totalmente desconocido virus. [4]

## OBJETIVOS

- Diseñar un sistema digital que cumplan las funciones de procesamiento de datos leído de una memoria.
- Estructurar una secuencia o instrucciones en un controlador MSS que administra la dinámica del proyecto
- Obtener una cadena indicadora de 6 bits en base a la información de 19 bits de muestras clínicas sobre RAM.

## DESCRIPCIÓN DEL PROBLEMA

Se diseñará una base de datos basada en la información sobre los estudios clínicos, realizados en periodos de 2018-2021, de las personas ecuatorianas en cuyo historial médico se reflejó un alto consumo de antimicrobianos, y, por tanto, alta resistencia a estos [1]. Preparando 100 muestras que recopilan:

| Resultados de cultivos | perfiles de RAM (sexo, edad, ciudad, código ATC, gravedad) | tipo de muestra obtenida |

Formatos basados en los artículos publicados por INSPIP. Estructurando una "cadena cultivos-sexo-edad-gravedad" en lenguaje binario, siendo una etiqueta para separar a la población según ciudad, sexo, rangos de edad (Ej: niñez, adolescencia, adultez), y mayor gravedad, es decir, mayor resistencia a fármacos contra las bacterias E.coli y K. De ese modo obtener indicadores de riesgo, basados en las frecuencias (repeticiones) de los casos a considerar, antes de administrar más fármaco. Así actualizándose según cada nuevo ingreso y cambios en los factores. [3]

La finalidad de este proyecto es cargar las 100 muestras de la ROM a la RAM, luego leer su información desde la RAM y en base a ella diseñar indicadores de mayor frecuencia de síntomas de RAM (resistencia antibióticos microbianos). La cadena se presenta de la siguiente manera:

Resul.cultivos	sexo	edad	ciudad	códigoATC	gravedad	muestra		
0-Ecoli	0-M	1 a más de 64 años	Quito	000-A	00-Leves	00-sangre		
1-K	1-F		Cuenca	001-B	01-Moderados	01-orina		
				010-C	10-Graves	10-heces		
				011-D	11-Mortales			
				100-G				
				101-H				
				110-J				
				111-K				
1bit	1bit	7 bits	3bits	3 bits	2 bits	3 bits		

Siendo una palabra de 20 bits Se cogerá esta información y se crea una nueva cadena para organizar las frecuencias (repeticiones) de la misma dentro de las 100 muestras.

| Resul.cultivos | gravedad | NvEdad | Sexo |

Donde NvEdad corresponde a los rangos de edad: niñez, adolescencia, adultez y vejez. Con esta nueva cadena como referencia se tiene como objetivo cernir las 100 muestras de manera que se obtengan hasta 64 frecuencias posibles, las cuales se guardaran para su posterior evaluación cada 8 filas, de la máxima frecuencia.

Este intervalo de 8 filas se debe a que los indicadores serán una guía para determinar al mayor afectado según la bacteria y nivel de gravedad, cerniendo entre la población de su sexo y rangos de edad. Por lo que, si vemos las posibilidades conjuntas entre sexo y edad, tenemos 8 casos. Por ejemplo, de las frecuencias totales de la población con RAM en bacteria E.coli de gravedad moderada, se escogerán las frecuencias máximas de entre sus 8 filas.

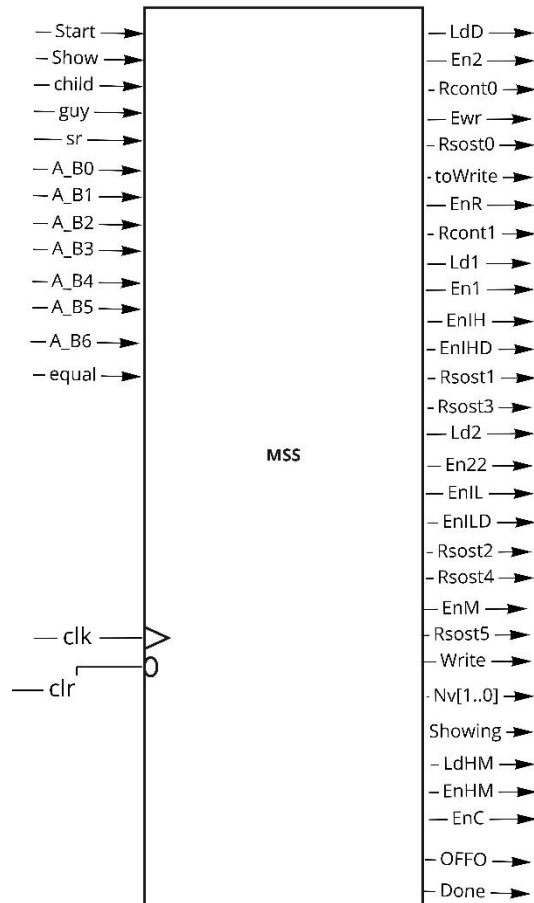
Cuando se encuentre una frecuencia máxima, o frecuencias en caso de varias coincidencias, se escribirá en la RAM un Indicador. Finalmente, se procederá a mostrar median un LCD decimal el nuevo código simplificado que refleje la información del indicador.

Bacteria	gravedad	NvEdad	Sexo
----------	----------	--------	------

0-Ecoli	0-Leves	0-Niño	0-M
1-K	1-Moderado	1-Joven	1-F
	2-Graves	2-Adulto	
	3-Mortales	3-Mayor	

## DO Û CRIÚ CIÓN E RAZÓN DE SOLUCIÓN

### Diseño de la solución:



Esta solución fue escogida debido a que se buscaba una eficiencia en el guardado de información para la búsqueda de frecuencias máximas de los casos revisados, es decir, se quería velar por la seguridad e integridad de los bits que se procesarían. Entonces, en vez de usar registros adicionales o celdas para almacenar las frecuencias, se optó por la conmutación entre lectura y escritura de la RAM, para aprovechar sus celdas sobrantes, ya que, las 100 muestras solo ocupaban casi la mitad de capacidad de memoria y así se decidió repartir las restantes 128 direcciones en 64 para almacenar las repeticiones que se registren y los indicadores resultantes.

Y así, como en un microprocesador básico, ahorrar la mayor cantidad de contadores y registros posibles para múltiples tareas, ya que, este sistema, lee datos de RAM para crear nuevas direcciones y luego en ellas carga datos, los incrementa en +1 y los vuelve a escribir, para luego volver a leerlos y escribir en otra nueva dirección. Este diseño nos permitió ahorrar dispositivos o elementos lógicos en la FPGA. Claro que, sufre el defecto que una sobrecarga de muestras o una frecuencia de trabajo muy grande puedan causar un cruce no deseado entre señales de lectura y escritura.

**Etapa A Carga de Muestras:** En esta etapa se generan direcciones para poder cargar las 100 muestras contenidos en la memoria ROM a la RAM. Señales importantes durante este proceso son:

- Direcciones de Muestras-Direcciones binarias de 0 a 99 para obtener datos de ROM
- Muestras-Cadenas de 20 bits con la información de las muestras
- Señales desde Etapa A-son bits provenientes de un comparador que avisa que se han cargado las 100 muestras
- Señales a Etapa A-son el conjunto de habilitadores En, cargas Ld y reseteadores R.
- Lectura-Señal que habilita la ROM
- Escribir-Señal que habilita escritura en la RAM

**Etapa B Lectura de Muestras:** En esta etapa se generan direcciones para poder obtener las 100 muestras cargadas en la memoria RAM. Señales importantes durante este proceso son:

- Direcciones de Datos-Direcciones binarias de 0 a 99 para obtener datos de RAM
- Datos-Cadenas de 20 bits con la información de los Datos en la RAM
- Señales desde Etapa B-son bits provenientes de un comparador que avisa que se han leído las 100 muestras
- Señales a Etapa B-son el conjunto de habilitadores En, cargas Ld y reseteadores R.

**Etapa C Registro de Repeticiones:** En esta etapa se cuentan las repeticiones en base a las muestras que llegan desde la memoria RAM y se guardan en un Registro. Señales importantes durante este proceso son:

- Datos-Cadenas de 20 bits con la información de los Datos en la RAM
- Repeticiones-Los valores de las repeticiones según cada secuencia específica
- Señales desde Etapa C-son bits provenientes de unos comparadores que ayudan a crear los rangos de edad
- Señales a Etapa C-son el conjunto de habilitadores En, cargas Ld, reseteadores R y bits que determinan los rangos de edad.

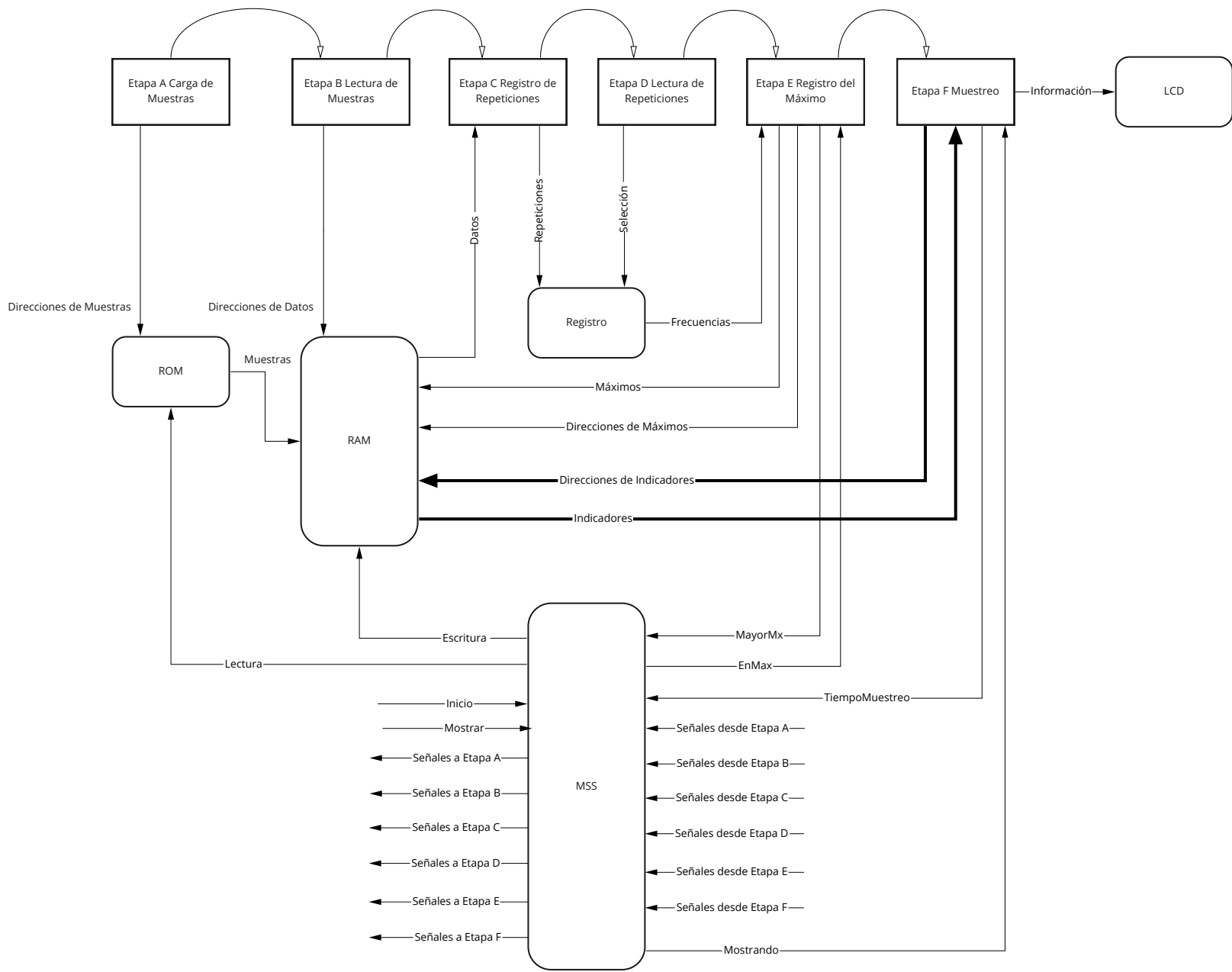
**Etapa D Lectura de Repeticiones:** En esta etapa se revisan las repeticiones/frecuencias desde el registro. Señales importantes durante este proceso son:

- Selección-Numero o dirección de la repetición a revisar.
- Frecuencias-Los valores de las frecuencias
- Señales desde Etapa D-son bits provenientes de un comparador que avisa que se han leído las 64 repeticiones
- Señales a Etapa D-son el conjunto de habilitadores En, cargas Ld y reseteadores

**Etapa E Registro del Máximo:** En esta etapa se comparan las frecuencias con un valor máximo para hallar los numero que más se repiten. Señales importantes durante este proceso son:

- Frecuencias-Los valores de las frecuencias
- Máximos-Frecuencias Máximas cada 8 filas.

DIAGRAMA DE BLOQUES DE LA SOLUCIÓN



- Direcciones de Máximos -Direcciones para escribir las frecuencias máximas en RAM
- MayorMx-Indica que existe un valor máximo
- EnMax-Habilita el registro de un máximo
- Escribir-Señal que habilita escritura en la RAM
- Señales desde Etapa E-s son bits provenientes de un comparador que avisa que se han leído las 64 repeticiones
- Señales a Etapa E-son el conjunto de habilitadores En, cargas Ld y reseteadores

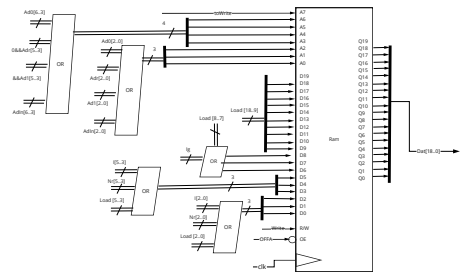
**Etapas F Muestreo:** En esta etapa se muestran los indicadores escritos en la RAM. Señales importantes durante este proceso son:

- Indicadores-Los valores procesados finales en la RAM
- Direcciones de Indicadores-Posiciones de los Indicadores en la RAM
- Mostrando -Habilitador para mostrar en LCD
- Tiempo de muestreo-Indicador de que se ha mostrado el tiempo determinado
- Información-Datos codificados para mostrar la información de los indicadores en formato decimal
- Señales desde Etapa F-s son bits provenientes de un comparador que avisa que se han revisados todos los indicadores
- Señales a Etapa F-son el conjunto de habilitadores En, cargas Ld y reseteadores

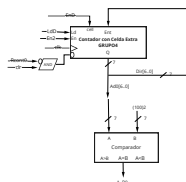


# PARTICIÓN FUNCIONAL DE SOLUCIÓN

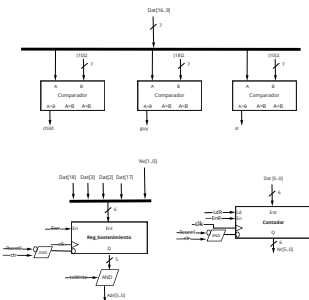
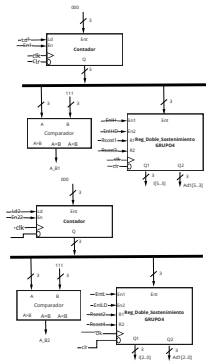
Memorias



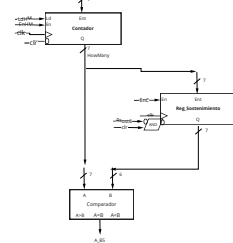
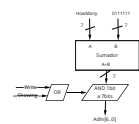
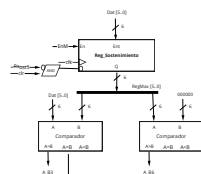
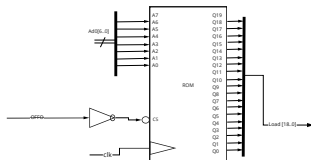
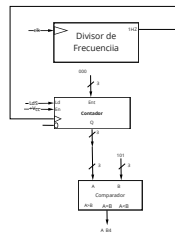
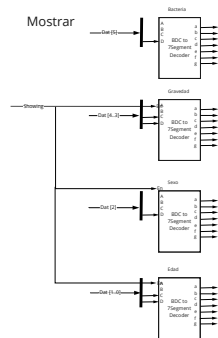
Carga y Lectura de Muestras



Lectura de Repeticiones y Escritura de Máximo



Mostrar



### Listado de señales de la Solución

Ojo: todas son señales con lógica positiva, lo que indican o activan es cuando son HIGH, en el caso de un solo bit.

#### Entradas externas a la MSS:

- Start-Inicia el análisis de muestras
- Show-Muestra por LCD

#### Salidas externas de la MSS:

- Done-Indica que se ha terminado la operación

#### Entradas internas a la MSS:

- A\_B0-Indica que se han revisado las 100 muestras
- A\_B1-Indica que se han revisado las 64 repeticiones
- A\_B2-Indica que se han revisado 8 filas en las repeticiones.
- A\_B3-Indica si el número de repeticiones actual es mayor al máximo registrado.
- A\_B4-indica que han pasado 5 segundos.
- A\_B5-Indica cuando se han revisado todos los indicadores
- A\_B6-Confirma la presencia de un numero de repeticiones distinto de 0
- equal-indica si hay más de un indicador posible con la misma frecuencia absoluta
- child-indica cuando la edad no es de un niño.
- guy-indica cuando la edad no es de un adolescente.
- sr-indica cuando la edad no es de un adulto.

#### Salidas internas de la MSS:

- Nv-Nivel o Rango de Edad
- LdD-Carga en celda extra del contador para carga y lectura de las 100 muestras.
- En2-Habilitador del contador para carga y lectura de las 100 muestras.
- Rcont0-Reseatador del contador para carga y lectura de las 100 muestras.
- Ewr-Habilitador del Registro de nuevas direcciones para el numero de repeticiones
- Rsost0-Reseatador del Registro de nuevas direcciones
- toWrite-Habilitador de direcciones para escribir
- EnR-Habilitador de contador de Repeticiones
- Rcont1-Reseatador del contador de repeticiones
- Ld1-Cargador de Contador para Indicadores del bit 5 al bit 3
- En1-Habilitador del Contador de Indicadores del bit 5 al bit 3
- EnIH-Habilitador para carga Doble Registro para Ad1 del bit 5 al bit 3

- EnIHD-Habilitador para carga Doble Registro para I del bit 5 al bit 3
- Rsost1-Reseatador del Doble Registro para Ad1 del bit 5 al bit 3
- Rsost3-Reseatador del Doble Registro para I del bit 5 al bit 3
- Ld2-Cargador de Contador para Indicadores del bit 2 al bit 0
- En22-Habilitador del Contador de Indicadores del bit 2 al bit 0
- EnIL-Habilitador para carga Doble Registro para Ad1 del bit 2 al bit 0
- EnILD-Habilitador para carga Doble Registro para I del bit 2 al bit 0
- Rsost2-Reseatador del Doble Registro para Ad1 del bit 2 al bit 0
- Rsost4-Reseatador del Doble Registro para I del bit 2 al bit 0
- EnM-Habilitador de Registro de Repetición Máxima
- Rsost5-Reseatador del Registro de Repetición Máxima
- Write-escritura en RAM
- Showing-habilitador para muestra de la información de indicadores por LCD
- LdHM-Carga encerrada del contador para registrar número de indicadores
- EnHM-Habilitador del contador para registrar número de indicadores posibles
- EnC-Habilitador para cargar en registro la cantidad de indicadores

#### **Señales internas del Sistema:**

- Dat-Salida de datos de 18 bits de la RAM
- ADr-Direcciones para escribir las repeticiones de cada posible caso específico
- Nr-Número de Repeticiones a cargar en la RAM
- Ad1-Direcciones para Lectura de las repeticiones en RAM
- I-Información de indicador
- Load-Datos de ROM
- RegMax-Cantidad de repetición máxima
- HowMany-Cantidad de Indicadores

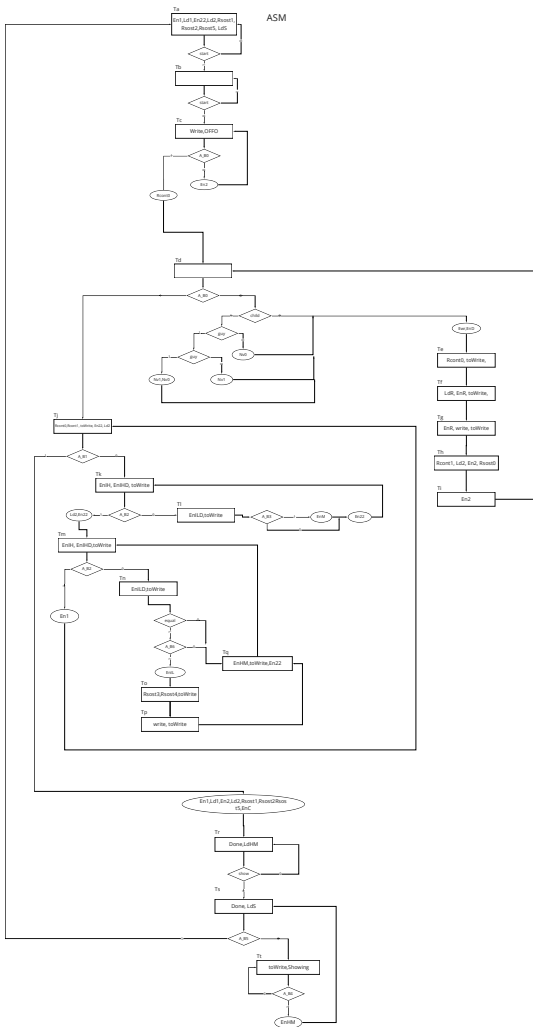
Para esta solución, al presionar y soltar Start, el proceso para determinar los números con más repeticiones. Máximas frecuencias, usa las propias direcciones de la RAM para escribir las cantidades. Así un contador para la carga de Datos en la ROM y para la lectura de la RAM, pero este contador no es ordinario. Específico para este proyecto está un contador con celda extra, la cual almacena la cantidad de la salida Q como reserva, es decir, la mantiene en espera en la entrada Ent aun pese a que se ha borrado de la salida Q, en pocas palabras, es un contador que carga el valor de una salida ya borrada.

Esta celda extra es usada para el proceso de escritura de las repeticiones, dado que como esta acción se realiza simultánea, en la RAM hay un proceso intercalado de lectura y escritura. Así como en el caso anterior los datos de las muestras forman una nueva cadena, y esta será la

dirección de la RAM donde se almacenara la repetición, entonces, se lee una muestra, la nueva cadena se carga como dirección esperándose para una segunda fase de lectura pero esta vez una lectura desde la dirección 10000000, de estas se cargan las cantidad almacenadas a un contador y este la incrementa en 1, y luego de pasa a un modo de escritura en esa misma dirección cargando ahora el valor aumentado.

Al finalizar el proceso se tiene 64 direcciones llenas con las repeticiones, ahora se procede a la revisión de estos para buscar la frecuencia máxima, mediante la comparación de las cantidades. Y Así, como en el proceso previo, se lee y se escribe, solo que la escritura aguarda al momento en que se encuentra con una frecuencia la máxima y esta se compara con las 8 filas para encontrar iguales, es decir, las 8 filas se revisan 2 veces: un recorrido para hallar el máximo, y otro para hallar todos los máximos repetidos, y en el momento de coincidir se prepara la información de esa cadena para escribir en las nuevas direcciones desde 11111111, que puede ser 8 o más indicaras dependiendo de que tantas frecuencias máxima repetidos existan. Al terminar activa la alerta Done.

Durante el proceso de la búsqueda y escritura de los máximos, se utilizan dos nuevos, para este proyecto, Registros de Doble sostenimiento, que tenían dos salidas Q1 y Q2, habilitadas con En1 y En2 respectivamente, y reseteadas por R1 y R2 de igual manera. Y cada vez que se escribían se documentaba la cantidad de indicadores en el contador con señal HowMany, para luego cargar esa cantidad en un Registro, encerrar el contador de HowMany y utilizarlos para revisar los indicadores para presentarlos con LCD decimales. Una vez que se active Show.



## DIAGRAMA ASM DEL CONTROLADOR

**Estado Ta:** Se resetean/enceran la mayoría de registro y contadores que se usan en los últimos procesos, además, es el estado inicial que espera que se presione Start.

**Estado Tb:** Es el antirebote de Start.

**Estado Tc:** Habilita la ROM para lectura y a la RAM para escritura, habilitando al contador de direcciones Ad0 de muestras con En2 para obtener las salidas de ROM y escribirlas en la RAM. Este proceso se mantiene mientras A\_B0 no avise que se han llegado a las 100, para pasar al estado Tj. Al terminar, se resetea al contador,

**Estado Td:** Estado de lectura de Datos en la RAM, aquí se vuelve a usar el contador de direcciones Ad0, para revisar las muestras, se evalúan las señales child, guy y sr de los comparadores para determinar los valores de Nv: 00 niñez, 01 adolescencia, 10 adultez, 11 vejez. Para luego guardar la dirección actual en la celda extra del contador con END, y guardar la nueva cadena en Adr formando con el registro habilitando Ewr.

**Estado Te:** Se resetea el contador con Rcont0 y se cambia a las direcciones desde 10000000 con toWrite.

**Estado Tf:** Se mantiene a toWrite, y se carga la repetición que se obtiene a la salida de la RAM con la dirección sostenida por habilitar Ewr previamente, usando LdR y EnR.

**Estado Tg:** Se aumenta en 1 la cantidad de repetición obtenido en Tf, y al mismo tiempo se sobrescribe la nueva cantidad Nr en la misma dirección Adr, al habilitar con Write.

**Estado Th:** Se encera el contador usado para las repeticiones con Rcont1 y el Registro usado para las direcciones de las repeticiones en la RAM con Rsost0 y se carga la dirección almacenada en la celda extra del contador de direcciones de muestras activando En2 y Ld2.

**Estado Ti:** Se habilita el contador de direcciones de muestras con En2 para seguir con la siguiente dirección, regresando a Td.

En los siguientes estados se mantiene habilitado toWrite.

**Estado Tj:** Se resetean los contadores de direcciones de muestras y de repeticiones con Rcont0 y Rcont1. También se encera un contador para revisar cada 8 filas de las 64 repeticiones con En22 y Ld2. Se habilitan las direcciones con toWrite. Además, se evalúa A\_B1, que indica cuando se revisen las 64 repeticiones, mientras no sea 1 se pasa al estado Tk, sino, encera los contadores de las 64 repeticiones y las 8 filas, las salidas del doble registro que corresponden a los datos de los indicadores con Rsost1 y Rsost2, y se carga la cantidad de indicadores en un registro habilitado por EnC, y finalmente al estado Tr.

**Estado Tk:** Se habilitan las salidas del doble registro de los primeros 3 bits de las direcciones para leer las repeticiones guardadas en la RAM Ad1 y la información de los indicadores que se iban a escribir en la RAM I, con EnIH, EnIHD. Se evalúa A\_B2, que indica cuando se han revisado 8 filas, mientras no se lleguen a las 8, se pasa al estado TI, y cuando si, carga con 0 el contador de 8 filas con Ld2 y En22 para pasar al estado Tm.

**Estado TI:** Se habilita la salida del doble registro de los últimos 3 bits de las direcciones de las repeticiones en la RAM Ad1, con EnILD. Se evalúa A\_B3 que indica cuando la cantidad sacada de la RAM es superior a la salida del Registro de Máxima frecuencia RgMax, cuando ocurra esto se carga esa nueva cantidad con EnM en el Registro de Máxima frecuencia, luego se habilita la siguiente fila con En22, regresando a Tk.

**Estado Tm:** Se vuelven a habilitar las salidas del doble registro con EnIH y EnIHD. Se vuelve a evaluar A\_B2, mientras no se llegue a las 8 filas se pasa al estado Tn, y al llegar se incrementa el contador de las 64 direcciones con En1 a Tj.

**Estado Tn:** Se habilita la salida del doble registro de los últimos 3 bits de las direcciones de las repeticiones en la RAM Ad1, con EnILD. Se evalúa equal, que indica coincidencias con la frecuencia máxima, y el dato es mayor a 0, se registra los últimos 3 bits de la información de los indicadores I con EnIL, para pasar al estado To, si no hay coincidencias pasa al estado Tq.

**Estado To:** Se resetean las direcciones de las repeticiones de la RAM Ad1 con Rsost3 y 4, pasando al estado Tp.

**Estado Tp:** Se escribe en la RAM con write, pasando al estado Tq.

**Estado Tq:** Se habilitado el contador de indicadores HowMany con EnHM y se avanza a la siguiente fila con En22, pasando al estado Tm.

Aquí se deja de tener necesariamente habilitado toWrite.

**Estado Tr:** Se activa la señal Done de que termina el proceso y se encera el contador de HowMany con LdHM, aquí se espera que show sea presionado/activado, para pasar al estado Ts.

**Estado Ts:** Mantiene Done activa y encera al contador para muestreo con LdS, evalúa A\_B5 que indica si se han revisado todos los indicadores, pasando al estado Ta, mientras no pase, ira al estado Tt.

**Estado Tt:** Se habilita toWrite para pasar a otro nivel de direcciones, se activa Showing para activar los decodificadores de BCD a 7 segmentos decimal, para presentar en los LCD el código decimal. Y evalúa a A\_B4, que le avisara cuando hayan pasado 5 segundos de muestreo, cuando ocurra habilita con EnHM al siguiente indicador, sino se mantiene en Tt.

Report

Report not available

Groups Report

Tasks

Early Pin Planning

Early Pin Planning...

Run I/O Assignment

Export Pin Assignment

Pin Finder...

Highlight Pins

I/O Banks

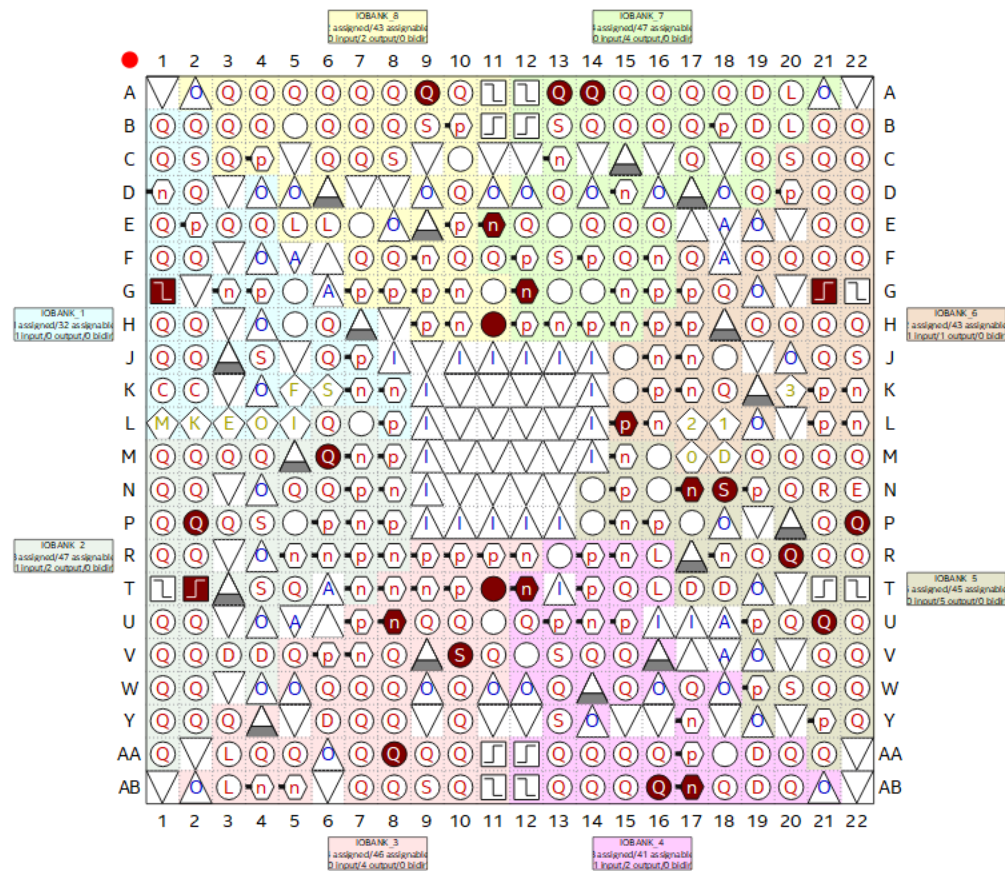
VREF Groups

Edges

Clock Pins

Clock

PLL/DLL Input

Top View - Wire Bond  
Cyclone IV E - EP4CE15F23C6

Pin Legend

Symbol	Pin Type
○	User I/O
●	User assign...
●	Fitter assign...
●	Unbonded ...
●	Reserved pin
○	Other confi...
○	DEV_OE
○	DEV_CLR
○	DIFF_n
○	DIFF_p
○	DQ
○	DQS
○	CLK_n
○	CLK_p
○	Other PLL
○	Other dual ...
○	MSEL0
○	MSEL1
○	MSEL2
○	MSEL3
○	CONF_DONE
○	nCE
○	nCONFIG
○	TDI
○	TCK
○	TMS
○	TDO

Named: \* Edit: 3.0-V LVTTTL

Filter: Pins: all

Node Name	Direction	Location	I/O Bank	VREF Group	Fitter Location	I/O Standard	Reserved	Current Strength	Slew Rate	Differential Pair	ict Pre
Dat[0]	Output	PIN_H11	8	B8_N0	PIN_H11	3.0-V LVTTTL		8mA (default)	2 (default)		
Dat[4]	Output	PIN_A9	8	B8_N0	PIN_A9	3.0-V LVTTTL		8mA (default)	2 (default)		
Dat[9]	Output	PIN_G12	7	B7_N1	PIN_G12	3.0-V LVTTTL		8mA (default)	2 (default)		
Dat[1]	Output	PIN_E11	7	B7_N1	PIN_E11	3.0-V LVTTTL		8mA (default)	2 (default)		



## ASIGNACIÓN DE PINES

Report

Report not available

Groups

Report

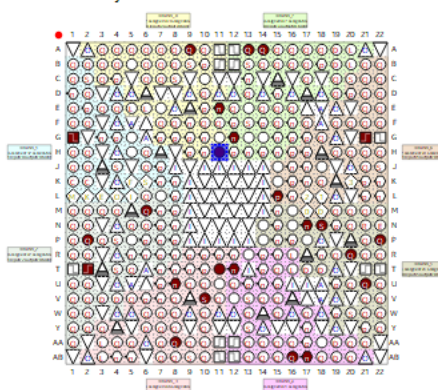
Tasks

Early Pin Planning

Early Pin Planning...

Run I/O Assignment /

Export Pin Assignment

Top View - Wire Bond  
Cyclone IV E - EP4CE15F23C6

Pin Legend

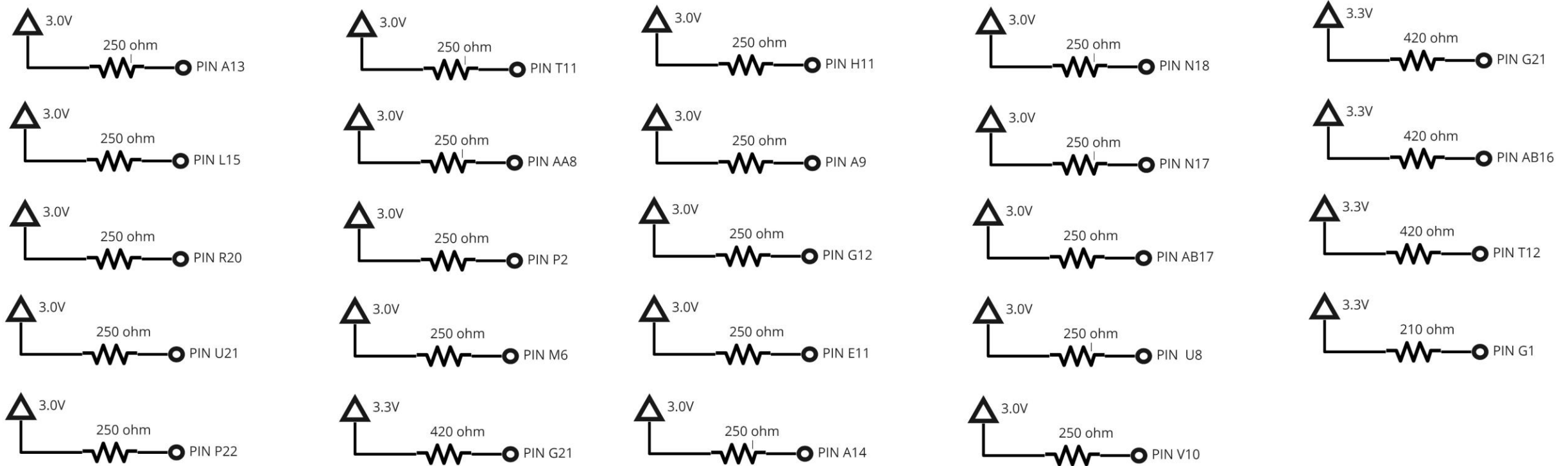
Symbol	Pin Type
	User I/O
	User assign...
	Fitter assign...
	Unbonded ...
	Reserved pin
	Other confi...
	DEV_OE
	DEV_CLR
	DIFF_n
	DIFF_p
	DQ

Named: \* Edit: 2 (default)

Filter: Pins: all

Node Name	Direction	Location	I/O Bank	VREF Group	Fitter Location	I/O Standard	Reserved	Current Strength	Slew Rate	Differential Pair	ict Pre
Dat[0]	Output	PIN_H11	8	B8_N0	PIN_H11	3.0-V LVTTTL		12ma	2 (default)		
Dat[4]	Output	PIN_A9	8	B8_N0	PIN_A9	3.0-V LVTTTL		12ma	2 (default)		
Dat[9]	Output	PIN_G12	7	B7_N1	PIN_G12	3.0-V LVTTTL		12ma	2 (default)		
Dat[1]	Output	PIN_E11	7	B7_N1	PIN_E11	3.0-V LVTTTL		12ma	2 (default)		
Dat[5]	Output	PIN_A14	7	B7_N1	PIN_A14	3.0-V LVTTTL		12ma	2 (default)		
Dat[2]	Output	PIN_A13	7	B7_N1	PIN_A13	3.0-V LVTTTL		12ma	2 (default)		
Dat[3]	Output	PIN_L15	6	B6_N1	PIN_L15	3.0-V LVTTTL		12ma	2 (default)		
start	Input	PIN_G21	6	B6_N1	PIN_G21	3.3-V LVTTTL		8ma			
Dat[15]	Output	PIN_R20	5	B5_N1	PIN_R20	3.0-V LVTTTL		12ma	2 (default)		
Dat[6]	Output	PIN_U21	5	B5_N0	PIN_U21	3.0-V LVTTTL		12ma	2 (default)		
Dat[8]	Output	PIN_P22	5	B5_N0	PIN_P22	3.0-V LVTTTL		12ma	2 (default)		
Dat[11]	Output	PIN_N18	5	B5_N0	PIN_N18	3.0-V LVTTTL		12ma	2 (default)		
Dat[13]	Output	PIN_N17	5	B5_N0	PIN_N17	3.0-V LVTTTL		12ma	2 (default)		
Dat[12]	Output	PIN_AB17	4	B4_N1	PIN_AB17	3.0-V LVTTTL		12ma	2 (default)		
done	Output	PIN_T12	4	B4_N1	PIN_T12	3.3-V LVTTTL		8ma	2 (default)		
show	Input	PIN_AB16	4	B4_N1	PIN_AB16	3.3-V LVTTTL		8ma			
Dat[16]	Output	PIN_U8	3	B3_N1	PIN_U8	3.0-V LVTTTL		12ma	2 (default)		
Dat[18]	Output	PIN_V10	3	B3_N0	PIN_V10	3.0-V LVTTTL		12ma	2 (default)		
Dat[17]	Output	PIN_T11	3	B3_N0	PIN_T11	3.0-V LVTTTL		12ma	2 (default)		
Dat[7]	Output	PIN_AA8	3	B3_N0	PIN_AA8	3.0-V LVTTTL		12ma	2 (default)		
Dat[14]	Output	PIN_P2	2	B2_N0	PIN_P2	3.0-V LVTTTL		12ma	2 (default)		
Dat[10]	Output	PIN_M6	2	B2_N0	PIN_M6	3.0-V LVTTTL		12ma	2 (default)		
resetsn	Input	PIN_T2	2	B2_N0	PIN_T2	3.3-V LVTTTL		8ma			
clk	Input	PIN_G1	1	B1_N1	PIN_G1	3.3-V LVTTTL		16ma			

## ESQUEMÁTICO DE PINES



Las entradas y salidas operan a 8mA

Los datos de la RAM operan a 12mA

La frecuencia de reloj opera a 16mA

## APLICACIONES A FUTURO

Las posibles aplicaciones o mejoras al diseño utilizado serían la remodelación de ciertas etapas, como los AND y OR de vectores lógicos, que se reemplacen por multiplexores de vectores lógicos de hasta 19 bits, también, la aminoración de contadores y registros al asimilar más la estructura de un microprocesador. Es más, el diseño de una ALU o incluso el uso de una ALU convencional podría facilitar ciertos procesos como el aumento en repeticiones para obtener las frecuencias.

Con estas modificaciones, este sistema no solo se limitaría a buscar las frecuencias máximas, dado que, esta ya implementado para ordenamiento de datos, debió a la etapa donde desmonta la cadena original y forma unas nuevas como direcciones de almacenamiento de las frecuencias/repeticiones, se podría aprovechar esto para enumerar y categorizar cadenas, para su posterior ordenamiento según la especificación que se le dé.

## CONCLUSIONES

- Se consiguió efectivamente el procesamiento de datos transferidos de memoria ROM a RAM, logrando estructuras nuevas cadenas que permitieron identificar las frecuencias o repeticiones a evaluar.
- Se logró diseñar una MSS capaz de administrar los procesos conmutados de lectura y escritura con la RAM, aunque, cabe recalcar, se necesitaron subastados de espera para que las señales se asienten durante cada proceso.
- Se determinaron indicadores de alto riesgo de resistencia microbiana de tamaño de 6 bits al lograr desfragmentar las muestras de 19bits.

## RECOMENDACIONES

- Al momento de diseñar una secuencia de lectura y escritura, es preferible darle tiempo a que las direcciones de la memoria se estabilicen, una buena práctica es usar estados como retardo.
- Para el procesamiento de datos es preferible usar registros auxiliares en caso de que la información sea fugaz y así no se perderán datos.

## Referencias Bibliográficas

- [1] Torres. D. A. T. (2021). Indicadores de resistencia antimicrobiana en la unidad de cuidados intensivos en un hospital de Quito, Ecuador Revista científica INSPILIP V, Número 2, Guayaquil, Ecuador. Recuperado de: [https://www.inspilip.gob.ec/wp-content/uploads/2021/08/Indicadores-de-resistencia-antimicrobiana\\_DOI.pdf](https://www.inspilip.gob.ec/wp-content/uploads/2021/08/Indicadores-de-resistencia-antimicrobiana_DOI.pdf)
- [2] Torres, D. A. T. (2019). Indicadores de resistencia antimicrobiana en pacientes de la unidad de cuidados intensivos en un hospital al norte de Quito 2018. Universidad Central del Ecuador, Quito, Ecuador. Recuperado de: <http://www.dspace.uce.edu.ec/bitstream/25000/20135/1/T-UC-0008-CQU-195.pdf>
- [3] Rev Cubana Farm vol.48 no.1 Ciudad de la Habana. (2014). Perfil de reacciones adversas a medicamentos notificadas. Universidad Nacional del Chaco Austral. Chaco, Argentina. Recuperado de: [http://scielo.sld.cu/scielo.php?script=sci\\_arttext&pid=S0034-75152014000100010](http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S0034-75152014000100010)

- [4] Realpe Muñoz P. Cortés Carvajal J. (2020). Tendencias Tecnológicas basadas en FPGA: Un enfoque hacia el COVID-19. Universidad Antonio José Camacho, Colombia. Recuperado de: <https://repositorio.uniajc.edu.co/bitstream/handle/uniajc/118/REVISTA%20ACTITUD%20%20NOVIEMBRE%202020%20baja-3-14.pdf?sequence=1&isAllowed=y>
- [5] Hossein Abad Z. S., Kline A., Sultana M., Noaen M., Nurmambetova E., Lucini F., Al-Jefri M. & Lee J. (2021). Digital public health surveillance: A systematic scoping review. NPJ Digital Medicine. Recuperado de: <https://www.nature.com/articles/s41746-021-00407-6>