

Apéndice

A

## Apéndice A Cronología del desarrollo del sistema

### A.1 Primer diseño

Al inicio, la propuesta fue desarrollar la etapa de sensado y control de un brazo mecánico; de carácter muy sencillo en diseño pues simplemente era conectar las señales provenientes de los transductores a los AO de instrumentación, transferirlas a un multiplexor y de ahí al convertidor análogo-digital. Obviamente con una lógica de control que incluyera un contador binario para la selección de las 9 galgas y un reloj. El diagrama fue el siguiente:

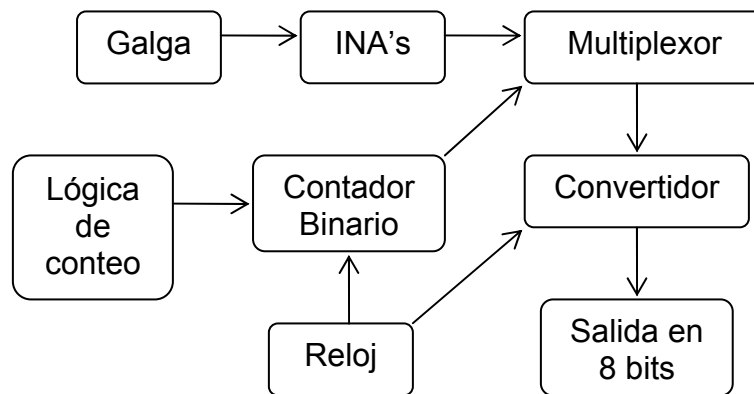


Figura 41 Diagrama a bloques del primer diseño.

Lo complicado en el primer diseño era el acondicionamiento de las galgas y la forma en la que serían colocadas en el guante para poder realizar las mediciones. Por lo que con un diseño tan sencillo no era necesario el uso de circuitos integrados de mayor complejidad. Y aún así el sistema era escalable.

Al continuar con el desarrollo de la tesis, se originaron problemas con el diseñador de la interfaz, ya que la velocidad con la que se capturaban los datos y con la que eran recibidos no coincidía; después de intentar varias opciones sin logro alguno el Dr. Rubén Alejos propuso que las dos partes enviaran los datos a una memoria y con ello se solucionaba el problema, además de separarlas pues no había igualdad del progreso en ambas tesis como se había propuesto al inicio de las tesis.

## A.2 Segundo diseño

Continuando con el nuevo diseño se encontraron otros problemas pues no se habló nada sobre el control del tiempo o tamaño de la memoria y únicamente se quería para almacenar los datos; perdiéndose el concepto original de ejecución de los movimientos en tiempo real y el poder visualizar la correcta medición obtenida de los transductores. El nuevo diseño quedó como se ilustra:

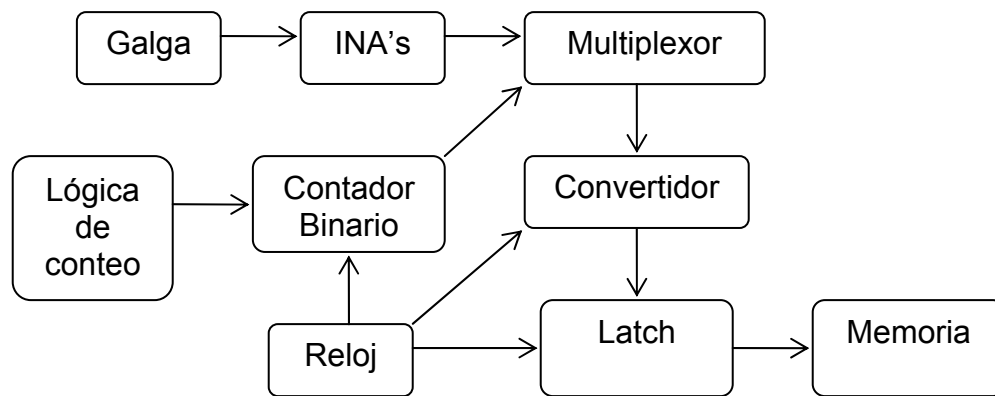


Figura 42 Diagrama a bloques del segundo diseño.

El diseño seguía siendo simple y no necesitaba de circuitería más compleja pero se descartó al no obtener nuevamente coordinación con la parte de la interfaz, efectivamente existían discrepancias en los tiempos corroborando lo dicho en el párrafo anterior.

## A.3 Tercer diseño

Como el usuario de la interfaz supuestamente estaba utilizando un microcontrolador para su sistema y la memoria seleccionada para el almacenamiento de los datos muestreados es de 1Mb la interfaz necesitaba de un microcontrolador extra para poder leerla pues serían 20 señales de la localidad de memoria, 8 señales de datos, 2 para transmisión y recepción, 1 para el reloj y otro último para la señal de reinicio. Además que la transmisión de los datos sería en

forma serial y no en paralelo como se maneja en el sistema, provocando retardos no deseados.

Por ello en el nuevo diseño se ocupó una lógica de dos eventos al utilizar los ciclos de reloj (nivel bajo y nivel alto), recuperándose el concepto original de movimientos en tiempo real y no únicamente el de almacenar los datos y leerlos posteriormente. Así en los ciclos bajos del reloj el control lo tenían el los multiplexores, el ADC y el generador de direcciones interno, para los ciclos altos el convertidor y el generador de direcciones interno se desconectaban para que el tomara posesión del bus el generador de direcciones externo y los datos de la memoria pudiesen ser leídos.

Se superaron varios conflictos pero con ciertas limitaciones por ejemplo:

- No se sabía el momento en que la memoria agotaba su capacidad, provocando que se escribiera constantemente.
- También no había conocimiento del evento de fin de lectura de los datos por lo que se leería indefinidamente.
- La frecuencia utilizada en el reloj interno era fija, estableciéndose un único tiempo de grabación.
- No se contaba con otro tipo de lógica de control para el manejo del sistema.
- Se carecía de indicadores de estado para conocer el comportamiento y cambios del sistema.

El diagrama siguiente muestra la nueva configuración, cabe hacerse notar que la señal de reloj enviada a los generadores de direcciones externos es invertida para que opere en los ciclos altos del reloj y no se tengan los conflictos ya indicados. La interfaz recibiría los datos y generaría las señales de reinicio y reloj externos para controlar el bus externo al momento de leer la memoria.

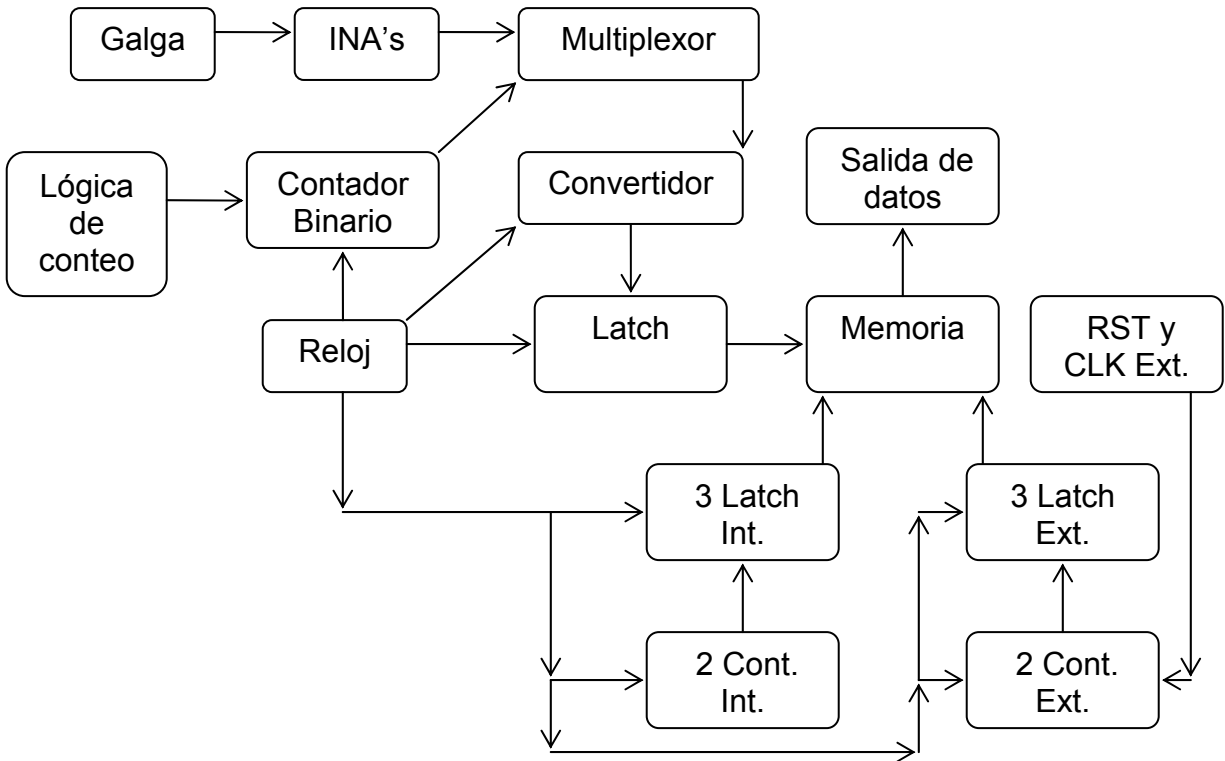


Figura 43 Diagrama a bloques del tercer diseño.

#### A.4 Cuarto diseño

Tomando en cuenta que no era posible visualizar el estado del sistema se implementó el utilizar una matriz de led's para el despliegue de los datos. La matriz es de bajo costo en comparación con un visualizador LCD, así que se utilizó una matriz de 9x8; para los nueve sensores de ocho bits cada uno. Todos tenemos conocimiento de lo que es una matriz y un led por ello no incluyo un diagrama esquemático del diseño. Lo que si hay que hacer notar es que los led's deben de ser colocados de tal manera que se permita unir de cada led una pata para las columnas y otra pata para los renglones. Las primeras son para el despliegue de cada uno de los sensores y las segundas para visualizar el valor binario de ellos.

La lógica y circuitos integrados para el despliegue de los datos en la matriz tanto para la lectura como para la escritura, se comentó en capítulos anteriores.

## A.5 Quinto diseño

Pensando en cómo los datos eran escritos y leídos y en que localidades se encontraban se incluyeron dos líneas más de led's una para observar la localidad de memoria a la que se escribe y otra para la localidad de memoria que es leída. Asimismo varios led's más individualmente para indicar el estado del sistema y botones para controlarlo más eficientemente.

Lo anterior para visualizar y saber el estado del sistema, hacer la calibración de cada sensor más fácil y corroborar el funcionamiento en tiempo real de los movimientos realizados, los datos guardados y los datos leídos. Haciendo más fácil el poder expandir o trabajar en posibles modificaciones del sistema.

De igual manera se añadió un switch para el manejo de distintas velocidades de operación incluyéndose la posibilidad de una velocidad controladora global de ambos sistemas (control e interfaz) proveniente de la interfaz.

El control lógico resultó sencillo al requerir solamente de flip-flops y compuertas lógicas para su implementación, aunque en este momento ya podría haberse utilizado circuitería más compleja para el diseño (microcontroladores) no se utilizaron ya que el sistema se fue modificando poco a poco (como se ha descrito) hasta llegar al punto final cuyo diagrama se muestra en la página 76.

Como ventaja en la utilización del microcontrolador destaca, el uso de menos hardware; pero si se quisiera expandir el sistema se tendría que añadir un multiplexor para las señales provenientes de los sensores, también la comunicación con la memoria sería en forma serial, y en caso de haberse utilizado la programación debería realizarse desde el inicio del proyecto para poder controlar el sistema por completo, en el caso de haber tenido el concepto del diseño final en la propuesta de tesis.