ESTUDIO DEL SISTEMA DE ADQUISICION DE DATOS KEITHLEY 500A Y APLICACIONES

Ing. Fausto Vasco Moncayo
Título: Ingeniero en Electrónica y Control, obtenido en Abril de 1991 en la Escuela Politécnica Nacional.
Cargo actual: Ingeniero en el departamento técnico de la compañía ***InteleQ*** **ć.**

Ing. Patricio Burbano R.
Título: MSc. Systems and Control, obtenido el año de 1984 en la Universidad de Manchester, Inglaterra.
Cargo actual: Profesor principal a tiempo completo en la Escuela politécnica Nacional.

RESUMEN.- En el presente trabajo se hace un estudio del sistema de adquisición de datos y control KEITHLEY 500A y sus aplicaciones a los sistemas de control en tiempo real.

En principio se realiza un estudio de técnicas de adquisición y salida de datos análogos y digitales, en los dos esquemas de trabajo que tiene el Quick500 (software diseñado para el aprovechamiento de la estación KEITHLEY 500A), el foreground o trabajo dedicado del computador y el background o trabajo en fondo.

Este estudio se complementa con el aprovechamiento de capacidades adicionales del equipo como graficación, temporizadores, análisis de datos, etc., y mediante la aplicación de algoritmos de control adecuados, obtener sistemas de control en tiempo real.

ABSTRACT.- In the present job we will study data acquisition and the control system KEITHLEY 500A and it´s aplication on real time control systems.

First we will realize the study of acquisition and output of analog and digital data technics, inside the two work schemes which Quick500 (software designed for utilization of KEITHLEY 500A workstation) has, foreground and background.

This study is complemented with the use of aditionals features of the unit like real time graphics, timers, data analysis, etc., and trought the aplication of the acquired control procedures, we will obtaine real time control systems.

1.- INTRODUCCION.- En nuestro medio el control en tiempo real es un campo relativamente nuevo, por lo que se planteó la necesidad de crear una biblioteca de programas y rutinas, que en base al equipo con que se cuenta permitan realizar tareas como adquisición de datos, salida de datos, graficacion en tiempo real, procesamiento de información, etc., orientados a posibilitar realizar control de sistemas en tiempo real. El sistema de adquisición de datos y control KEITHLEY 500A se encuentra acoplado a un computador IBM PS/2 modelo 60 y su utilización conjunta implica el uso de dos paquetes de software, el Quick500, diseñado para el aprovechamiento de la estación KEITHLEY 500A y el QuickBASIC 4.5, el mismo que proporciona el medio ambiente para el Quick500, y permite la edición y prueba de los programas que utilizan las diferentes tareas que es posible realizar con el equipo antes mencionado, para finalmente poder ser compilado para futuras aplicaciones.

La programación se puede realizar en dos esquemas diferentes de trabajo; foreground o trabajo dedicado y background o trabajo de fondo, tanto para un solo canal de entrada y/o salida (unicanal) como para sistemas de varias entradas y/o salidas (multicanal), y en nuestro caso en particular está orientada a facilitar el uso del equipo aportando una guía para los usuarios estudiantes y profesores.

1.1.- LA ESTACION KEITHLEY 500A.- En lo referente al hardware, la estación KEITHLEY 500A está constituida por una tarjeta maestra, la cual acepta módulos de aplicación, los mismos que se instalan en los 10 slots de expansión con que cuenta dicha tarjeta matriz. Es compatible con más de 25 módulos diferentes especializados en acondicionamiento de la señal y en actuación.

Algunos de los módulos que se puede mencionar son: entradas análogas y conversión A/D, salidas análogas y conversión D/A, entradas y salidas digitales, entradas de corriente normalizada, amplificadores de corriente, entradas de termocuplas, strain gages, contadores de pulsos, controladores de motores a pasos, etc..

Para el trabajo que desarrollamos se contó con solo tres módulos: el AMM1 Módulo maestro de medición análoga, el AOM1 Módulo de salida análoga y el DOM1 Módulo de salidas digitales, los cuales se describirán en forma breve más adelante.

El KEITHLEY 500A está construida de acuerdo a la arquitectura de memoria de un computador IBM PC, PS/2, 80386 y computadores 100% compatibles y está enlazada al computador, mediante una interfaz que contiene un bus especializado de extensión, el cual reside en el computador "anfitrión".

El conjunto de la estación de adquisición de datos y control KEITHLEY 500A y el computador constituyen una unidad independiente de

adquisición de datos y control en tiempo real, como se muestra en la figura 1.

En lo referente al software, se requiere que el computador sea IBM o 100% compatible, caso contrario ni siquiera es posible la instalación del Quick500. Se debe contar con un DOS 3.0 o mayor (DOS 3.3 en el caso de un PS/2) y el compilador debe ser el QuickBASIC 4.0 o mayores.

En cuanto a la memoria, el computador debe contar con al menos 640Kb. para poder correr la aplicación, el programa desarrollado y tener memoria suficiente para datos, cuya cantidad es seteable por el usuario. (1Mb. de memoria es lo recomendable.). Además para tener las ventajas gráficas del Quick500 se debe contar con una tarjeta adaptadora de gráficos IBM CGA, Hércules color, etc..

El coprocesador matemático no es necesario, sin embargo provee de mayor velocidad de proceso en la ejecución de los comandos del Quick500. El Quick500 soporta diferentes relojes desde 4.77 MHz. hasta 12 MHz. con estados de espera 1 o 0.

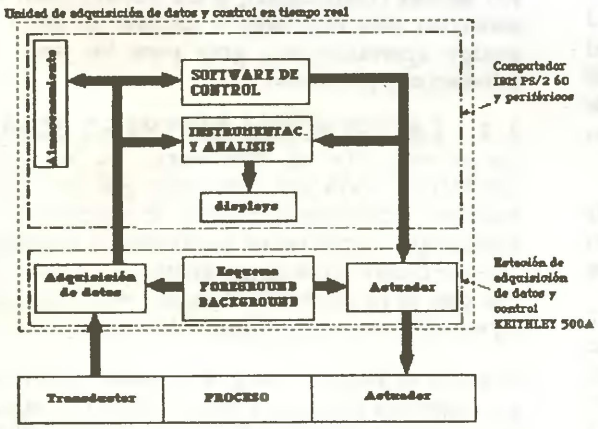

Figura 1.- Unidad de adquisición y procesamiento de datos y control en tiempo real.

1.2.- LOS MODULOS DE EXPANSION.-

Como antes se indicó, se cuenta con tres módulos, los cuales se describen en forma breve a continuación:

AMM1 (módulo maestro de medición análoga).- Indispensable en cualquier sistema de adquisición de datos, realiza la adquisición a 32 KHz. con una resolución de 12 bits. Provee de 8 canales de entradas análogas con ganancia local unitaria, pero con una ganancia global (afecta a todos los canales) seleccionable por software de x1, x2, x5 y x10. Luego de su acondicionamiento las señales son enrutadas a la sección del conversor A/D para su respectiva conversión.

Para la conversión A/D, el AMM1 utiliza un conversor de 12 bits de aproximaciones sucesivas que asegura velocidad y exactitud en la medición y la conversión. El tiempo máximo que puede tomar una conversión puede ser de 25 useg., y una medición por muestreador retenedor puede tomar 5 useg.. Se cuenta además con cinco rangos diferentes de conversión A/D (tres bipolares y dos unipolares) que se pueden seleccionar mediante dip-switches en la tarjeta.

AOM1 (módulo de salida análogo).- Ofrece una resolución de 12 bits y provee de cinco canales de salida análoga. Cada canal tiene un conversor D/A independiente con salida seleccionable mediante dip-switches en los rangos 0 a 10V, 0 a 5V, +/-10V, +/-5V y +/-2.5V (los mismos rangos que el AMM1).

Este módulo tiene la característica de soportar los dos niveles de congelamiento de datos en el conversor D/A y permite sincronizar la actualización de todos los canales de salida análoga.

DOM1 (módulo de salida digital).- Provee de 16 canales de salida digital, aislados ópticamente entre si y a tierra. Estos canales configurados como salidas de voltaje TTL, pueden sacar incluso 28V, o una salida de corriente.

Los 16 canales se agrupan en dos puertos de 8 canales cada uno, donde cada puerto es tratado como un byte. Dicho agrupamiento permite acceso simultáneo a 8 canales y habilita la comunicación con elementos TTL de 4 y 8 bits.

1.3.- QUICK500, CARACTERISTICAS .-

El Quick500 debe trabajar en el medio ambiente que le proporciona el QuickBASIC, por lo que sus comandos deben cumplir con algunas convenciones de forma de poder ser reconocidos como tales. Así se tiene que todos los comandos deben comenzar con la palabra CALL, sin ninguna excepción, por ejemplo CALL INIT.

Todos los comandos siguen dicho formato o sintaxis, pero algunos comandos permiten incluir información adicional acerca de su función mediante parámetros, los cuales deben ser enlistados de la siguiente manera: la lista debe estar encerrada entre paréntesis, además todos los parámetros deben estar presentes (ninguno es opcional o es asumido) y se deben separar con espacios o con comas y pueden ser constantes o variables.

Parámetros mal ubicados o mal mecanografiados causan resultados impredecibles y probables fallas en el sistema. Además existen ciertas palabras reservadas en el Quick500 tales como "PAGEO", "BT", "WBT", "WGO", etc., que no pueden ser utilizadas a no ser para el motivo para el que fueron predefinidas.

INICIALIZACION DEL PROGRAMA.- Existen dos comandos que ejecutan diferentes tareas de inicialización del sistema de adquisición de datos, computador y memoria reservada para datos, estos son INIT y SOFTINIT.

SOFTINIT es indispensable y obligatorio en cualquier programa y debe ser ubicado antes que cualquier otro comando del Quick500. Su labor es inicializar variables y parámetros internos que pueden ser seteados por el compilador del Quick500 para operar apropiadamente.

INIT es opcional como cualquier otro comando y su labor es inicializar algunos elementos del sistema y resetear el hardware de adquisición de datos, así lleva todos los canales de salida análoga o digital a cero, borra toda la memoria para datos, por lo que todo dato almacenado antes de la ejecución de este comando es perdido. Con frecuencia se lo utiliza al inicio del programa, únicamente después que el comando SOFTINIT.

BACKGROUND/FOREGROUND.- El concepto de background/foreground es muy fácil de entender si se examina la tarjeta de interfaz entre el KEITHLEY 500A y el computador. Esta tarjeta contiene un generador de interrupciones que trabaja junto con las interrupciones no mascarables del computador. La frecuencia de dichas interrupciones puede ser especificada mediante el comando INTON (INterrupt ON).

CALL INTON (frec%, ut$)

Donde ut$ indica las unidades de tiempo (seg., mseg., etc.) mientras que frec%, es un valor que indica el número de unidades de tiempo que dura la interrupción.

Con cada interrupción el procesador salta desde foreground, a ejecutar tareas de background. Al ser completadas estas tareas, el control es devuelto al foreground. Este traspaso del control se realiza a intervalos constantes con el fin de maximizar la exactitud en las tareas de adquisición de datos.

Tanto foreground como background pueden actuar independientemente, además la programación de ambos es muy flexible, pero también se puede conseguir que ambos modos de trabajo actúen de manera conjunta y complementaria mediante comandos de foreground que disparan la ejecución de otros en background, o comandos que requieren de información de otros, etc.

Un gráfico que muestra la actividad conjunta de los dos modos de trabajo se muestra en la figura 2.

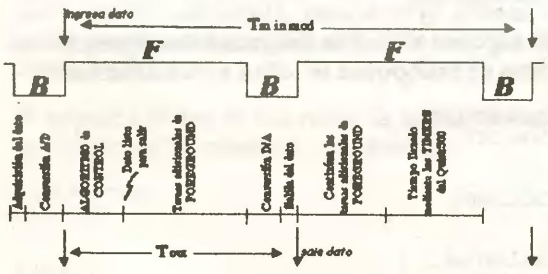

Figura 2.- Periódos de muestreo y tareas de backgound y foreground.

2.- ADQUISICION Y SALIDA DE DATOS.- La adquisición y salida de datos constituyen las tareas fundamentales para realizar control en tiempo real, por lo que se dedica una sección al estudio de estas tareas básicas. Esta revisión se la divide de acuerdo a los tres módulos con que se cuenta en el equipo.

2.1.- ADQUISICION DE DATOS MEDIANTE CANALES ANALOGOS.- El sistema de adquisición de datos KEITHLEY 500A tiene 8 canales de entradas análogas ubicadas en el módulo AMM1 el cual se describió anteriormente. Dichos canales reciben voltajes normalizados entre -10V y +10V.

La adquisición de datos puede realizarse tanto en background como en foreground, y en cada caso se puede tener adquisición unicanal y multicanal.

ADQUISICION EN FOREGROUND.- La adquisición de datos en foreground tiene como aplicación práctica el realizar instrumentación al sensar determinadas variables debido a limitantes que se mencionan a continuación.

En el caso de la adquisición de datos en foreground, no es posible determinar los periodos de muestreo de los canales de entrada, lo cual es fundamental para un sistema de control. Tampoco se puede realizar análisis de los datos puesto que en foreground no es posible definir y crear arreglos de memoria reservada para datos. No está por demás indicar que siendo la rutina más simple, tiene un enorme valor didáctico que introduce al usuario en el mundo del tiempo real.

COMANDOS ESPECIFICOS DEL QUICK500.- Cada tarea necesita de la ejecución de uno o varios comandos del Quick500, para el caso de la adquisición de datos en foreground se necesita un solo comando tanto para el caso univariable como para el caso multivariable.

CALL READVAR (READARRAY)

PROPOSITO: Rutina de foreground que muestrea entradas digitales, análogas y de frecuencia. El comando READVAR lee un canal simplemente y coloca el valor sensado en una variable BASIC. (READARRAY lee varios canales a la vez y los valores leidos se ubican en un arreglo de BASIC).

FORMATO:

CALL READVAR (ion$, iotypo%, rango%, va!, enf%, tm$)

CALL READARRAY (ion$, iotypo%, rango%, va()!, enf%, tm$)

Los parámetros indican lo siguiente: **ion$**, es el o los canales de entrada que se están utilizando; **iotypo%** indica si el canal leido es análogo, digital o de frecuencia; **rango%** solo se utiliza en el caso que el canal sea de frecuencia; **va!** es la variable de BASIC donde se almacena el valor medido., **enf%** determina las unidades en que se especificará el

valor leido y **tm$** es el modo de disparo.

El esquema básico de una rutina de adquisición de datos en foreground se bosqueja a continuación:

```
CALL SOFTINIT
CALL INIT
:
:
WHILE
    :
    :
    CALL READVAR(.....)
    :
    :
WEND
:
:
END
```

ADQUISICION EN BACKGROUND.- La adquisición de datos en background es también una rutina por demás sencilla y de gran utilidad, ya que al utilizar interrupciones no mascarables es capaz de fijar el período de muestreo de los canales al tiempo que crea arreglos de memoria dedicada para datos, la misma que permite realizar análisis de los datos adquiridos ya sea de inmediato o posteriormente. Otra ventaja es que al tiempo que se realiza la adquisición de datos también se ejecutan tareas de foreground de manera paralela.

COMANDOS ESPECIFICOS DEL QUICK500.- La adquisición de datos en background ya exige de la presencia de varios comandos del Quick500, los mismos que se describen ligeramente a continuanción.

CALL ANIN

PROPOSITO: Rutina que realiza el muestreo de hasta 64 canales de entrada análogos. El comando ANIN toma medidas a intervalos constantes de tiempo de un número determinado de canales y crea un área del Quick500, en la que se guardan los valores adquiridos. Puede ejecutarse por si solo o ser disparado por otro comando del Quick500.

FORMATO:

CALL ANIN (arn$, sn!, ionl$, bintv%, cy%, tm$, bfn$)

Los parámetros que se enlistan determinan: **arn$**, el nombre del arreglo de memoria para datos creado por el comando ANIN, **sn!** es la longitud de este arreglo, **ionl$**, es el listado de canales que almacenarán sus datos en este arreglo, **bintv%** es la frecuencia con que se muestrea los canales, se mide en número de interrupciones, **cy%** indica el número de veces que se debe repetir la tarea, **tm$** es el modo de disparo y **bfn$** es el nombre de la tarea con la que otras tareas (comandos) se pueden referir a esta tarea.

CALL ARGETVALF/ARGETVALI

PROPOSITO: Permite accesar a un simple valor guardado en un arreglo Quick500, especificando su ubicación en profundidad y anchura, y expresarlo como una variable BASIC. ARGETVALF y ARGETVALI ejecutan funciones similares. La única diferencia es que ARGETVALI enlista solo valores binarios en una variable entera, en tanto que ARGETVALF guarda este valor en una variable real y puede expresarse en el tipo de unidades más conveniente.

FORMATO:

CALL ARGETVALF (arn$, dop!, wid%, ion$, va!, euf%)

CALL ARGETVALI (arn$, dop!, wid%, ion$, va%)

Los parámetros que se enlistan determinan: **arn$**, el nombre del arreglo de memoria del cual toma los datos este comando, **dop!** es la longitud de este arreglo, **ion$**, es el listado de canales que almacenarán sus datos en este arreglo, en su lugar se puede indicar solo el número de canales con **wid%**, **va!** es la variable en la que coloca el dato y **euf%** determina las unidades de dicho dato.

CALL INTON

PROPOSITO: Habilita las interrupciones, a la vez que setea la frecuencia de las mismas de acuerdo a un valor especificado por el usuario. Las tareas de backgroud que generalmente son seteadas con anterioridad, no se ejecutan hasta que el comando INTON sea ejecutado.

FORMATO:

CALL INTON (frec%, ut$)

Los parámetros indican **ut$** las unidades de tiempo y **freq%** el número de unidades de tiempo entre muestreo y muestreo.

CALL INTOFF

PROPOSITO: Deshabilita las interrupciones que fueron habilitadas por el comando INTON. Este comando no borra o limpia las tareas de background, es decir que si existe un nuevo INTON se volverán a ejecutar estas tareas, aunque se puede perder el sincronisno, ya que todos los timers (en software y hardware) son deshabilitados.

FORMATO:

CALL INTOFF

El esquema básico de una rutina de adquisición de datos en background se indica a continuación:

```
CALL SOFTINIT
CALL INIT
:
:
CALL ANIN(.....)
::
CALL INTON(.....)
    WHILE
        :
        :
        CALL ARGETVALF(.......)
        :
```

```
    WEND
CALL INTOFF
:
:
END
```

2.2.- SALIDA DE DATOS MEDIANTE CANALES ANALOGOS O DIGITALES.-

La salida de datos análogos en el KEITHLEY 500A se realiza a través de los 5 canales de salida análoga del módulo AOM1 el cual se describió anteriormente. Dichos canales entregan voltajes normalizados entre -10V y +10V.

En cuanto a las salidas digitales estas se realizan a través de los 16 canales digitales del módulo DOM1 de manera individual o agrupados en puertos de 8 bits (canales). Dichos canales se encuentran seteados de manera que saquen niveles TTL compatibles.

La salida de datos puede realizarse tanto en background como en foreground, y en cada caso se puede tener adquisición unicanal y multicanal al igual que en la adquisición análoga de datos..

SALIDA DE DATOS EN FOREGROUND.- La salida de datos en foreground prácticamente no tiene aplicación práctica debido a las mismas limitantes que se mencionaron en el caso de la adquisición de datos en foreground.

COMANDOS ESPECIFICOS DEL QUICK500.- Como se indicó antes cada tarea necesita de la ejecución de uno o varios comandos del Quick500, en el caso de la salida de datos en foreground se necesita un solo comando tanto para el caso univariable como para el caso multivariable.

CALL WRITEVAR (WRITEARRAY)

PROPOSITO: Comando que escribe un valor (tomado de una variable BASIC de fácil acceso) especificado anteriormente en un canal de salida. Este canal puede ser análogo, digital, de frecuencia o de pulsos. Si se trata de un canal análogo, se puede sacar los valores en binario o en unidades de ingeniería.

FORMATO:

WRITEVAR (ion$, iotype%, va!, ouf%, tm$)

En adelante los parámetros no se detallan (salvo en los casos de necesidad), pero a estas alturas el lector puede intuir el significado de la mayoria de comandos.

El esquema básico de una rutina de salida de datos en foreground se muestra a continuación:

```
CALL SOFTINIT
CALL INIT
:
:
WHILE
    :
    :
    CALL WRITEVAR(.....)
    :
    :
WEND
:
END
```

SALIDA DE DATOS EN BACKGROUND.- La salida de datos en background es una rutina sencilla y de gran utilidad, ya que al utilizar interrupciones no mascarables es capaz de fijar el periodo de muestreo de los canales al tiempo que crea arreglos de memoria dedicada para datos, la misma que permite realizar análisis de los datos adquiridos ya sea de inmediato o posteriormente. Otra ventaja es que al tiempo que se realiza la salida de datos también se ejecutan tareas de foreground de manera paralela.

COMANDOS ESPECIFICOS DEL QUICK500.- La salida de datos en background ya exige de la presencia de varios comandos del Quick500, algunos de los cuales son generales para cualquier tarea de background y que ya fueron descritos anteriormente, los demás se detallan a continuación.

CALL ANOUT

PROPOSITO: Rutina que toma valores de un arreglo previamente creado y lo saca a través de canales de salida análogos. El comando ANOUT saca valores a intervalos constantes de tiempo de un número determinado de canales pero no crea un área del Quick500, en la que se guardan los valores adquiridos, este arreglo debe ser creado de antemano por ARMAKE, ANIN, etc. Puede ejecutarse por si solo o ser disparado por otro comando del Quick500.

FORMATO:

CALL ANOUT (arn$, ioml$, bintv%, cy%, tm$, bfn$)

CALL DIGOUT(arn$, ioml$, bintv%, cy%, tm$, bfn$)

CALL ARMAKE

PROPOSITO: Crea un arreglo de memoria con el formato Quick500. Dicho arreglo puede ser de cuatro tipos: de bit, byte, palabra o un arreglo real. Todo arreglo tiene dos dimensiones ancho y profundidad. El ancho se relaciona con el número de canales que pueden ser accesados y la profundidad es el número de valores trabajados por cada canal.

FORMATO: CALL ARMAKE(arn$, dep!, wid%, ioml$)

CALL ARPUTVALF/ARPUTVALI

PROPOSITO: Toma valores de una variable BASIC y los guarda en una arreglo Quick500. ARPUTVALF y ARPUTVALI ejecutan funciones similares. La única diferencia es que ARPUTVALI enlista solo valores binarios en una variable entera, en tanto que ARPUTVALF guarda este valor en una variable real y puede espresarse en el tipo de unidades más conveniente.

FORMATO:

CALL ARPUTVALF (arm$, dep!, wid%, ion$, va!, enf%)

CALL ARPUTVALI (arm$, dep!, wid%, ion$, va%)

El esquema básico de una rutina de salida de datos en background se describe a continuación:

```
CALL SOFTINIT
CALL INIT
:
CALL ANOUT(......)
::
CALL INTON(......)
    WHILE
        :
        CALL ARPUTVALF(.......)
        :
    WEND
CALL INTOFF
:
END
```

3.- OPERACION MULTITAREA.- Una vez que se ha descrito el software para la realización de las tareas fundamentales como son la adquisición y salida de datos, se describirá, de manera breve las opciones adicionales que tiene el Quick500 en el manejo del KEITHLEY 500A, opciones que se realizan en foreground, mientras en background se realiza la adquisición y salida de datos.

3.1.- INTERRUPCIONES Y PERIODO DE MUESTREO.- Si bien el período de muestreo se basa y depende de las interrupciones, se debe hacer una diferenciación, pues se trata de dos conceptos distintos del Quick500. Las interrupciones tienen su propio período, el cual es impuesto únicamente por el comando CALL INTON y sus parámetros, mientras que el período de muestreo, tiene como base el período de las interrupciones, pero se fija mediante un parámetro adicional, el **bintv%** (intervalo de background) de los comandos de entrada y salida de datos en background como ANIN, ANOUT, DIGOUT.

INTERRUPCIONES.- Como se ha anotado antes, el período de las interrupciones es fijado únicamente por el comando INTON, el mismo que al habilitar las interrupciones, setea también el período con que estas se presentan. Dicho comando crea un reloj base, el mismo que va a entregar el mando de manera alternativa a uno de los dos esquemas de trabajo del Quick500, tal como se muestra en la figura 3.

Una vez que las tareas de background han sido terminadas, se deshabilita las interrupciones mediante el comando CALL INTOFF.

PERIODO DE MUESTREO.- El muestreo de los canales de adquisición o de salida de datos no se realiza necesariamente cada interrupción, sino más bien cada cierto número de ellas, número que es impuesto por el parámetro **bintv%** presente en los comandos de adquisición y/o salida de datos en background, lo cual deja la opción de determinar un período de muestreo diferente para cada tarea de background, con la base de un único período de interrupciones. Esto cobra gran importancia en control, puesto que el período de muestreo que se impone, suele ser muy superior al tiempo que se demora entre la adquisición del dato hasta la obtención del resultado que debe ser devuelto a la planta, dato que en teoría debería ser entregado de inmediato (sin retardo), de allí que mientras menor sea el retardo, menor será el error cometido. Si bien se tiene diferentes períodos de muestreo para cada tarea, es conveniente que en cada nuevo muestreo de entrada (siempre mayor o igual que el de salida), se repita un ciclo, lo cual trae la necesidad de sincronizar las tareas, lo que se consigue teniendo multiplicidad exacta entre los períodos de muestreo de entrada y de salida, tal como se muestra en la figura 2, donde se observa que todo el tiempo que sobra desde el final de las tareas de foreground, hasta la nueva interrupción es llenado con los timer (del Quick500), con lo cual se asegura tener utilizado todo el tiempo de máquina lo que garantiza que todas las actividades de foreground se realicen una vez por cada dato adquirido, es decir, que también se sincroniza las tareas de foreground y background.

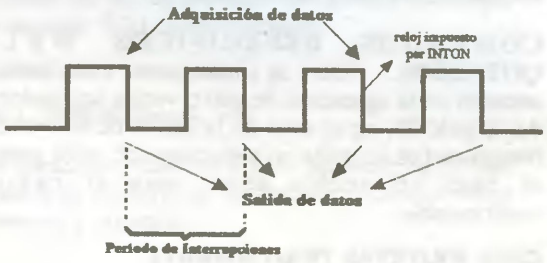

Figura 3.- Periodo de interrupciones.

3.2.-DISPAROS.- Los disparos constituyen una herramienta muy utilizada en la adquisición de datos y sistemas de control. Provee medios para, en condiciones externas o internas, obtener una respuesta programada. Se describen 4 tipos de disparo que se pueden utilizar en el Quick500, cada uno de los cuales tiene ventajas y desventajas, pero utilizados adecuadamente ofrecen un soporte completo para todas las necesidades.

a.- DISPARO DESDE BASIC.- El propósito es suspender la ejecución de ciertos comandos hasta que se cumpla una condición deseada. Utiliza el "WHILE.....WEND" del BASIC. Entre las ventajas que presenta se menciona la facilidad de usar sentencias de BASIC o Quick500 como expresión condicional y para adquirir información; pero la desventaja es que el lazo en el cual se prueba la condición para el disparo es muy lento y la operación de foreground es suspendida hasta comprobarse la condición.

b.- DISPARO DE FOREGROUND.- Suspende la ejecución del BASIC o el Quick500 subsecuente a un comando de foreground, hasta que una condición deseada sea verdadera. Para su

ejecución se llama al comando de disparo del Quick500; la condición deseada se incluye en sus parámetros, por ejemplo el comando SCHMITRIG chequeará si una entrada es mayor a 10 voltios

```
CALL SCHMITRIG("ANLG0", 10!, 10!, "above",.......)
```

Una vez que la condición se ha cumplido, se ejecutan las subsecuentes sentencias que pueden ser de una instrumentación de alarma por ejemplo.

```
PRINT "ALARMA"
BEEP
```

Los dos comandos que se pueden utilizar en disparo de foreground son SCHMITRIG y DIGINTRIG (no disponible). Estre sus ventajas se puede mencionar su velocidad, dado que el lazo de prueba está hecho en assembler, el comando o sentencia que se ejecuta al cumplirse la condición puede ser de BASIC o del Quick500. Las desventajas son sus limitaciones, dada la capacidad de los comandos de disparo de foreground y la paralización de las actividades de este esquema de trabajo hasta que la condición sea cumplida.

c.- DISPARO DE BACKGROUND.- El propósito de un disparo desde background es vigilar la entrega de una condición sin detener el proceso del foreground. Cuando se cumple la condición, la tarea es activada o disparada y la tarea de disparo se desactiva. Para realizar un disparo desde el background sólo se necesita cumplir con dos pasos, primero llamar un comando del Quick500 con el parámetro "modo de diparo" en "bt" (background trigger), éste setea la información acerca del disparo. Un segundo paso es llamar un comando con el parámetro "modo de disparo" en "wbt" (wait a background trigger). Estos dos comandos pueden quedar separados por varias líneas de programa, pero que no contengan un "bt" adicional. Los camandos con modo de disparo "bt" y "wbt" se comunican en conjunto al background. La condición de disparo se chequea cada interrupción, la misma que al cumplirse dispara la tarea "wbt" mientras que la tarea "bt" se auto desactiva.

Las interrupciones no necesariamente deben estar activadas cuando comandos con "bt" y "wbt" son llamados, sin embargo si deben activarse para ejecutars estos comandos. Este tipo de disparo obtiene una respuesta sumamente rápida, en el orden de los useg., ya que las tareas que esperan disparo "wbt", son ejecutadas inmediatamente después de las tareas "bt", siguiendo una secuencia de background.

d.- DISPARO POR GONOW.- El comando GONOW provee el medio para disparar hasta 16 tareas de background simultáneamente. GONOW puede ser autodisparado en background o en foreground. Se implementa en dos pasos: 1º se setea 1-16 comandos de background con el modo de disparo "wgo" (wait a gonow). El 2º paso es llamar al comando GONOW, si es en foreground será ejecutado inmediatamente y si es llamado en background, puede esperar disparo de background "wbt", donde será ejecutado por un comando que contenga "bt".

Las ventajas de este modo de disparo son importantes: el poder disparar hasta 16 tareas simultáneamente y la ventaja del comando GONOW de poder ser considerado como foreground o como background. La única desventaja es que cuando es disparado por otro tipo de disparo, asume sus desventajas.

3.3.- GRAFICOS EN TIEMPO REAL CON QUICK500.- Graficar en tiempo real con los comandos del Quick500 representa una gran ventaja ya que permite planificar en un solo comando muchos parámetros en torno al gráfico, tales como canal o canales a graficarse, el arreglo de memoria donde se encuentran los datos, los colores del gráfico, tipo de pantalla, valores máximos y mínimos, escalas, resolución, unidades, etc..

Sin embargo existe una desventaja de peso que no permite utilizar con eficacia los comandos de graficación en tiempo real del Quick500, en condiciones normales toma demasiado tiempo graficar los datos, demorando muchas veces un tiempo mayor al período de muestreo, lo que provoca pérdida de datos en la graficación ya que se necesita de un RTMDS software (Monitor de tiempo real y sistema directo de almacenamiento) y de una tarjeta aceleradora de gráficos RTM, la misma que se instala entre el monitor CRT y el adaptador de gráficos existente, ocupando uno de los slots de expansión del computador.

El equipo con que se cuenta carece de dicha tarjeta aceleradora de gráficos en tiempo real, por lo que a ciertas frecuencias ya resulta impráctico graficar utilizando los comandos del Quick500, resultando mejor utilizar rutinas en BASIC para la graficación en tiempo real toda vez que se puede conseguir tiempos de graficación menores y mejor resolución en pantalla.

3.4.- MANEJO DE ARREGLOS DE MEMORIA PARA DATOS.- Los arreglos de memoria para datos son herramientas indispensables para el trabajo con la estación KEITHLEY 500A, cuando se trabaja en background. Estos arreglos que son creados por los comandos ANIN o ARMAKE sirven para almacenar los datos que se han adquirido o que se sacan a través de los canales análogos o digitales de entrada o salida.

Así, cuando se ha adquirido datos, es importante ubicar estos datos en un arreglo de memoria; de igual manera cuando se tiene salida de datos en tiempo real se debe tener las señales de salida almacenadas en un arreglo de este tipo. También se puede obtener información del arreglo como profundidad, anchura, último dato accesado.

Pero además es importante poder guardar estos datos en disco para posteriores análisis o para obtención de resultados gráficos en papel mediante la utilización de hojas electrónicas por ejemplo y por lo tanto también poder recuperar datos

guardados en memoria de disco para su utilización en procesos de tiempo real.

3.5.- PROCESAMIENTO DE DATOS.- Cuando se han recopilado datos y estos son almacenados en arreglos de memoria para datos en el formato Quick500, se los puede procesar para obtener informaciones adicionales. Así, se puede obtener la derivada de una señal adquirida en un punto determinado, la integral de un grupo de datos, media aritmética, desviación estándar, ecuación de una recta aproximada mediante regresión lineal, etc..

Estas son las posibilidades que han sido consideradas en el Quick500, sin embargo mediante rutinas que se pueden desarrollar en BASIC se puede obtener mayor información. En realidad el tipo de análisis que es posible realizar con el Quick500 es más bien básico, sin embargo se po-dría contar con software especializado en análisis de datos adquiridos en tiempo real como el ASYSTANT, ASYS, etc., con los cuales el análisis puede ir mucho más alla, analizando las señales en el tiempo y en frecuencia.

4.- APLICACIONES.- El presente trabajo centra su atención en el estudio de las características y potencialidades de la estación KEITHLEY 500A, desarrollando rutinas que facilitan el uso del equipo en tareas de instrumentación, procesamiento y control. Sin embargo para dar una idea más objetiva se incluyen aplicaciones, las cuales se han escogido de forma que el interés se fije en la utilización del KEITHLEY 500A en sistemas de control en tiempo real. No se hace énfasis en el algoritmo, puesto que no es su desarrollo el objetivo del presente trabajo.

4.1.- CONTROLADOR PID DISCRETO.- Se parte de un controlador PID continuo, que responde a la siguiente ecuación:

$$u = Kp * e + Ki * \int e * dt + Kd * de/dt$$

en la cual se tienen los siguientes parámetros:

u señal de control
e error (diferencia :referencia menos salida)
Kp constante proporcional
Kd constante derivativa
Ki constante integral

Al sacar la transformada de Laplace de la anterior expresión, y luego la transformada Z se tiene:

$$\frac{u(z)}{e(z)} = Kp + Ki * \frac{T*(z+1)}{2*(z-1)} + \frac{Kd}{T} * (1 - z^{-1})$$

En la cual T es el período de muestreo. Si a la expresión anterior se le saca la transformada Z inversa, se obtiene la siguiente ecuación de diferencias:

$$u(k) = u(k-1) + b1*e(k) + b2*e(k-1) + b3*e(k-2)$$

$$b_1 = Kp + Ki\, T/2 + Kd/T \;;\quad b_2 = -Kp + Ki\, T/2 - 2*Kd/T$$

$$b_3 = Kd/T$$

u(k) y e(k) son los valores actuales de la ley de control (salida del controlador) y el error (entrada del controlador). **u(k-1) y e(k-1)** son los valores anteriores de las mismas señales. **e(k-2)** es el error existente dos periodos de nuestreo antes del actual. Un diagrama de bloques que describe el control mediante un PID discreto se presenta en la figura 4.

Los resultados obtenidos al aplicar un controlador PID sobre el circuito mostrado en la figura 5, se muestran en las figuras 6 y 7.

4.2.- REALIMENTACION DE ESTADO.- Los métodos convencionales como el lugar geométrico de la raiz y la respuesta de frecuencia son bastante útiles para tratar sistemas de una entrada y una salida, en cambio cuando se tienen plantas de multiples entradas y salidas es conveniente utlizar las técnicas del control moderno (dominio del tiempo), modelando el sistema en variables de estado como lo muestra la figura 8.

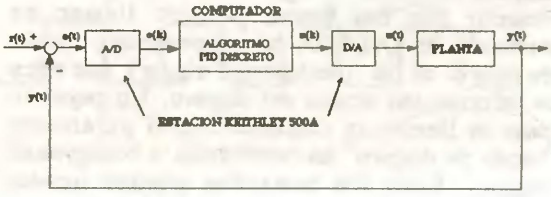

Figura 4.- Diagrama de bloques: Controlador PID discreto.

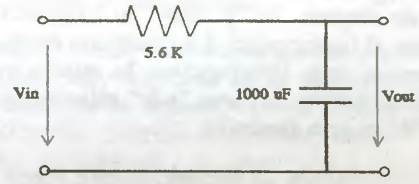

Figura 5.- Esquema del circuito R-C utilizado.

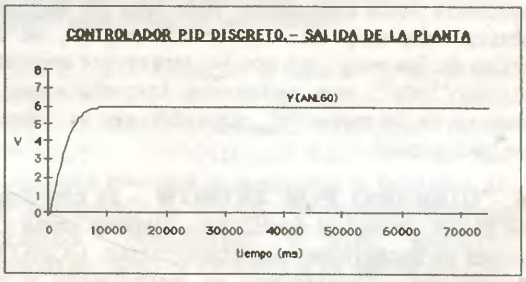

Figura 6.- Salida de la planta

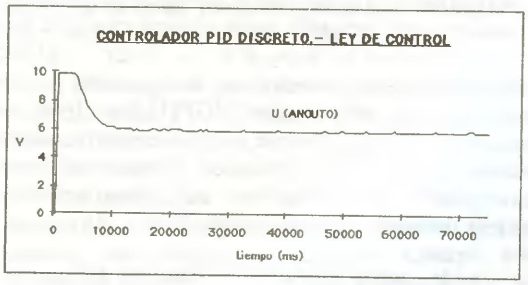

Figura 7.- Ley de control del sistema

Para comprobar la validez del algoritmo desarrolado se aplicó sobre el circuito de segundo orden mostrado en la figura 9, obteniéndose los resultados mostrados en la figura 10.

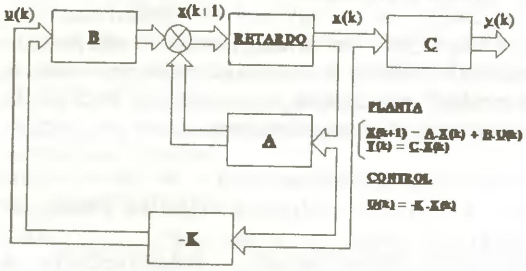

Figura 8.- Realimentación de estado discreta.

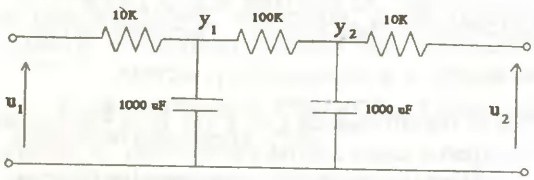

Figura 9.- Circuito R-C de segundo orden utilizado.

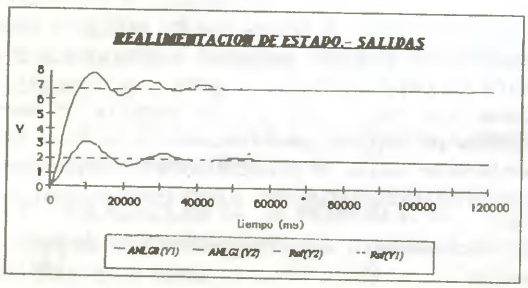

Figura 10.- Realim. de estado; salidas y referencias.

4.3.- CONTROL ADAPTIVO CON REFERENCIA A MODELO.- El control adaptivo con referencia a modelo, tiene como objetivos los dos problemas de control, el seguimiento y la regulación. Seguimiento cuando se cambia la referencia del modelo y la regulación cuando el sistema sufre alguna perturbación o cuando se cambian los parámetros de la planta.

El control adaptivo con referencia a modelo responde al diagrama de bloques general que se presenta en la figura 11, en el cual cada instante se actualiza la ley de control en base a identificación de los parámetros de la planta (en función de los parámetros de entrada y salida) y al modelo de referencia. Con el método escogido se obvia el cálculo de los parámetros de la planta y directamente se calcula los parámetros del control lo que se conoce como método directo o implícito.

Las pruebas del controlador adaptivo con referencia a modelo se realizaron sobre el circuito R-C como el que se muestra en la figura 12, al cual se le cambió la referencia y también se cambió los parámetros de la planta, incluyendo y posteriormente retirando una resistencia de 5.6K en paralelo con el condensador de la planta mostrada en la figura 12. La salida de la planta frente a este controlador se muestra en la figura 13.

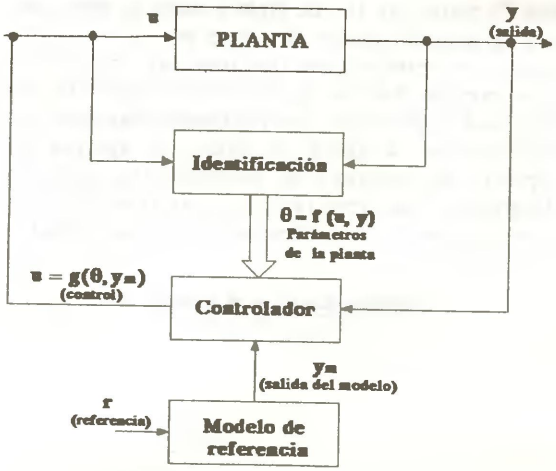

Figura 11.- Diagrama de bloques.- Control Adaptivo con referencia a modelo.

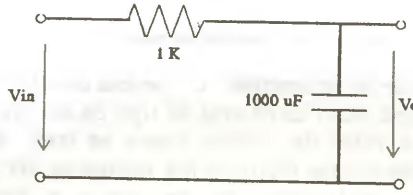

Figura 12.- Circuito R-C utilizado para la aplicación del algoritmo MRAS.

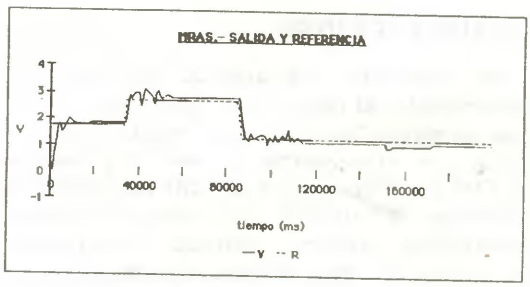

Figura 13.- Control adaptivo con referencia a modelo; salida

4.5.- TRANSFORMADA RAPIDA DE FOURIER.- El objetivo de estudiar la transformada rápida de Fourier en este trabajo, es abrir el campo para el análisis en frecuencia de datos adquiridos mediante la estación KEITHLEY 500A, y como ejemplo de este objetivo se ha

implementado el algoritmo que obtiene el espectro de densidad de potencia de una onda adquirida a través de los canales análogos de entrada.

Para probar la validez del algoritmo desarrollado se obtiene la transformada discreta de Fourier de la siguiente señal:

y = sen (4Hz.) + sen (12Hz.) + 2*rnd - 1

Es decir, contiene dos frecuencias predominantes, y además está contaminada de ruido. Esta señal es generada en un canal análogo de salida mediante una rutina desarrollada para generar ondas de este tipo, siendo almacenada en disco para su futuro análisis.

Para la obtención de la transformada se trabaja con 32 datos, por lo que dichas ondas se obtienen con el mismo número de datos por ciclo con el objeto de tener mayor precisión al obtener la transformada discreta de Fourier y el espectro de densidad de potencia. Los resultados obtenidos se muestran en la figura 14, donde se aprecia el espectro de densidad de potencia que permite determinar las frecuencias predominantes, enmascaradas por el ruido que contamina la señal.

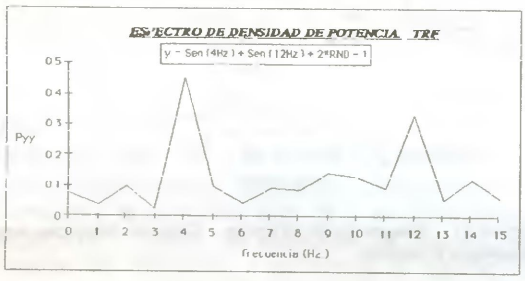

Figura 14.- Espectro de densidad de potencia

Además de las aplicaciones que se han detellado, se trabajo con éxito en control de tipo on-off, ya sea en dos niveles de voltaje como en tres. Para probar las salidas digitales del equipo se hizo un pequeño controlador de un motor a pasos, controlando funciones como velocidad, sentido de giro, duración del giro, etc..

5.- CONCLUSIONES.-

Se ha cumplido con uno de los objetivos fundamentales al iniciar este trabajo que es el de crear un manual explicativo de la utilización del sistema de adquisición de datos y control KEITHLEY 500A, tanto en software como en hardware, detallando los requerimientos, necesidades, alcance, utilidad, versatilidad, limitaciones, etc. de la estación antes mencionada.

El usuario potencial del sistema de adquisición de datos y control KEITHLEY 500A cuenta con una serie de rutinas modelo para realizar control en tiempo real, con el consiguiente ahorro de tiempo en lo que se refiere a familiarización con el equipo, escritura y prueba de las rutinas básicas de adquisición y salida de datos, procesamiento, instrumentación, etc..

Los resultados que arrojan los programas desarrollados son satisfactorios, toda vez que cumplen con lo que de ellos se esperaba, es decir, realizar instrumentación y control en tiempo real con la utilización de la estación KEITHLEY 500A, la misma que se ha probado como una herramienta valiosa para la realización de instrumentación inteligente, procesamiento de información, y aplicar todo ésto al control digital.

Se puede notar claramente que se ha puesto especial énfasis en el esquema de trabajo del background, ya que permite la realización de control en tiempo real al tener la opción de fijar períodos de muestreo, permite realizar análisis de ondas adquiridas o calculadas ya que cuenta con la facilidad de crear arreglos de memoria para datos, permite realizar instrumentación, etc., todo esto de manera simultánea.

El esquema de background trabaja con interrupciones no mascarables, es decir de máxima prioridad, por lo que las tareas de background nunca pueden dejar de realizarse.

Debido a la frecuencia con que se adquiere datos no es posible realizar todas las tareas de foreground, esto no tiene mucha importancia cuando se trata de una simple instrumentación, sin embargo, si se trata de un algoritmo de control o de instrumentación inteligente (con alarmas por ejemplo) si es importante. Debido a esto se recomienda fijar períodos mínimos de muestreo de acuerdo a la extensión del programa.

Una de las virtudes del Quick500, es la posibilidad de exportar datos a hojas electrónicas, programas de análisis, bases de datos, etc. para los diferentes usos que se puede dar a los datos adquiridos en tiempo real. Para la obtención de los resultados del presente trabajo de tesis se ha utilizado esta opción para graficar los datos adquiridos o de salida en papel. Resulta de gran importancia el contar con esta opción ya que permite obtener reportes que se presentan a manera de informe, ya que la observación en pantalla sirve más bien para efectos de instrumentación, o para fines didácticos.

El procesamiento de información que se puede realizar con el Quick500 es bastante elemental, ya que es un paquete de tipo general. Existen paquetes diseñados exclusivamente para el procesamiento de datos, como es el caso del "ASYSTANT", programa que permite entre otras cosas obtener la transformada rápida de Fourier, espectro de potencia, inversión de matrices, obtener determinantes, convoluciones, resolución de ecuaciones de diferencias, operaciones aritméticas básicas, resolución de polinomios, generar ondas, obtener información estadística, graficación, etc., por lo que se recomienda que se trate de adquirir dichos paquetes.

Por esta razón, se ha desarrollado el algoritmo de cálculo de la transformada rápida de Fourier y el espectro de densidad de potencia, para tratar de

suplir la falta de paquetes poderosos de software como es el "ASYSTANT", cuyo costo es elevado.

Se ha desarrollado una serie de rutinas básicas de control como PID discreto, realimentación de estado, controles ON-OFF, a manera de ejemplo o modelo, con la intención de dejar una puerta abierta para que en un futuro mediato, se desarrolle otros algoritmos de control y estos puedan ser implementados en tiempo real utilizando la estación KEITHLEY 500A.

Este es quizá el primer trabajo de este tipo que se realiza en la Facultad de Ingeniería Eléctrica, por lo que se espera que se constituya en el arranque para futuras aplicaciones del control en tiempo real.

No está por demás indicar que el objetivo del presente trabajo de tesis no es el optimizar el control de sistemas sino más bien, utilizar el sistema de adquisición de datos y control KEITHLEY 500A en aplicaciones básicas de control que puedan servir de punto de partida para aplicaciones futuras.

6.- REFERENCIAS:

1.- **KEITHLEY**, "Quick500 Data Acquisition and Control Software", 1988.

2.- **OGATA KATSUHIKO**, "Ingeniería de Control Moderna", Prentice-Hall International, 1973.

3.- **BURR-BROWN CORPORATION**, Data acquisition and control, 1986.

4.- **MICROSOFT**, "Microsoft Quick Basic 4.0", Microsoft Corpration, 1987.

5.- **KUO BENJAMIN**, "Digital Control Systems", Holt-Saunders International Editions, 1981.

6.- **ASTROM KARL**, "Adaptive Control", Addison Wesley, 1989

7.- **FRANKLIN G. & POWELL D.**, "Digital control of dynamic systems", Addison Wesley, 1981.

8.- **ORTIZ, HUGO.**- "Control adaptivo con modelo de referencia para sistemas discretos",1990. Tesis de Grado EPN.

BIOGRAFIAS.-

Ing. PATRICIO BURBANO R.- Nació en Quito. Se groduó de Ingeniero en Electrónica y Telecomunicaciones en la E.P.N. el año de 1974.

Realizó sus estudios de post-grado en el U.M.I.S.T. en la universidad de Manchester en Inglaterra en 1974.

Actualmente es profesor principal de la E.P.N. y director del proyecto de investigación CONUEP-EPN 87-01. "Control de procesos en tiempo real".

Ing. FAUSTO A. VASCO M.- Nació en Quito el 10 de Mayo de 1966. Obtuvo el bachillerato en FF.MM. el año de 1984 en el colegio "La Salle".

Se graduó de Ingeniero en Electrónica y Control en Abril de 1991 en la E.P.N.

Actualmente presta sus servicios al Departamento de Servicio y Soporte de la Compañía ***IntelcQ***.