

# Aspectos básicos de los analizadores lógicos para diseño digital

Por Ai-Lee Kuan, Agilent Technologies

Acerca de la autora  
Ai-Lee Kuan dio sus primeros pasos profesionales en Agilent Technologies como ingeniera de pruebas y analizadores lógicos. Amplió su actividad al negocio de los analizadores lógicos como especialista en aplicaciones y actualmente trabaja en varias aplicaciones de analizadores lógicos como directora de producto.

*Necesita ver simultáneamente las entradas y salidas de un contador de 16 bits para determinar un error de temporización y sólo dispone de un osciloscopio de dos canales. ¿Cómo puede verlas todas? Acaba de desarrollar diagramas de temporización para una placa llena de circuitos digitales. ¿Cómo los verifica? ¿Qué puede emplear para capturarlos y analizarlos?*

Si no dispone de la herramienta adecuada, solucionar este tipo de problemas puede resultar muy laborioso. La mejor solución para los problemas descritos es un analizador lógico. Este artículo realiza un breve repaso de los aspectos básicos de los analizadores lógicos y da una idea clara sobre qué puede hacer un analizador lógico.

## ¿Osciloscopio o analizador lógico?

Muchos ingenieros, cuando tienen que elegir entre un osciloscopio y un analizador lógico, optan por el primero. Sin embargo, los osciloscopios tienen una utilidad limitada en algunas aplicaciones. Según cuál sea la tarea, un analizador lógico puede proporcionar información más útil.

### Cuándo utilizar un osciloscopio

- Cuando necesite ver pequeñas excursiones de tensión en la señal.
- Cuando necesite una alta precisión del intervalo temporal.

### Cuándo utilizar un analizador lógico

- Cuando necesite ver muchas señales al mismo tiempo.
- Cuando necesite ver las señales de su sistema de la misma forma que el hardware.
- Cuando necesite disparar en un patrón de valores altos y bajos en varias líneas y ver el resultado.

Un analizador lógico reacciona igual que un circuito lógico cuando una señal del sistema cruza un umbral específico. Reconoce si la señal es alta o baja. También puede disparar en patrones de valores altos y bajos de estas señales.

En general, utilice un analizador lógico cuando necesite ver más líneas de las que puede mostrar el osciloscopio. Los analizadores lógicos resultan particularmente útiles para examinar relaciones temporales o datos de un bus, por ejemplo, los buses de dirección, datos o control de un microprocesador. Pueden decodificar la información de los buses del microprocesador y presentarla en un formato claro.

El analizador lógico es la herramienta adecuada cuando ya ha superado la etapa paramétrica del diseño, le interesan las relaciones de temporización entre muchas señales y necesita disparar en patrones de valores altos y bajos lógicos.

## ¿Qué es un analizador lógico?

La mayoría de los analizadores lógicos son en realidad dos analizadores en uno. El primer componente es un analizador de tiempos y el segundo, un analizador de estados.

## Aspectos básicos del analizador de tiempos

El analizador de tiempos muestra información con el mismo formato general que un osciloscopio, donde el eje horizontal representa el tiempo y

el eje vertical, la amplitud de tensión. Dado que las formas de onda de ambos instrumentos dependen del tiempo, se dice que la pantalla está en el dominio del tiempo.

## Selección del método de muestreo adecuado

Un analizador de tiempos es igual que un osciloscopio de digitalización con un bit de resolución vertical. Con un bit de resolución sólo puede mostrar dos estados: alto o bajo. Sólo considera un umbral de tensión definido por el usuario. Si la señal se sitúa por encima del umbral cuando se realiza el muestreo, el analizador la mostrará como 1 o alta. Cualquier señal muestreada que se encuentre debajo del umbral se mostrará como 0 o baja. Desde estos puntos de muestreo se genera una lista de unos y ceros que representa una imagen de un bit de la forma de onda de entrada. La lista se almacena en memoria y también se emplea para reconstruir una imagen de un bit de la forma de onda de entrada, tal como se aprecia en la figura 1.

El analizador de tiempos tiende a cuadrar todo, lo que parece limitar su utilidad. Si necesita verificar relaciones de temporización entre varios cientos de líneas viéndolas todas a la vez, un analizador de tiempos es la opción ideal.

Recuerde que cada punto de muestreo utiliza una ubicación de memoria. De este modo, cuanto más alta sea la resolución (velocidad de muestreo más rápida), más corta será la ventana de adquisición.

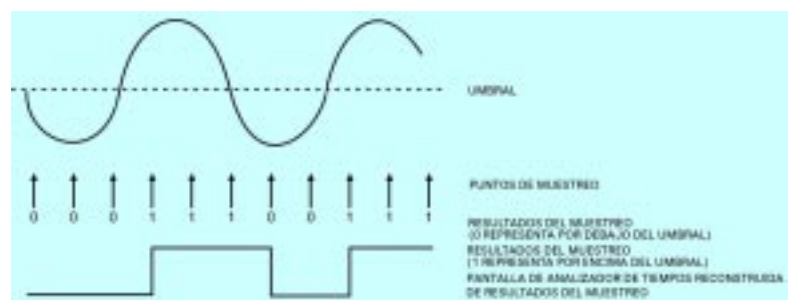
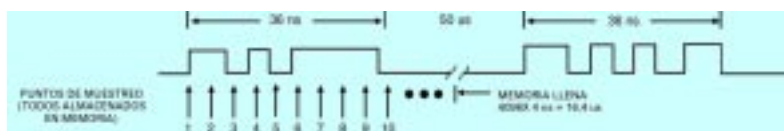


Figura 1. Puntos de muestreo del analizador de tiempos.

### Muestreo transicional

Cuando capturamos datos en una línea de entrada con ráfagas de datos, como se ilustra en la figura 2, debemos ajustar la velocidad de muestreo a una resolución elevada (por ejemplo, 4 ns) para capturar los



pulsos rápidos al principio. Esto significa que un analizador de tiempos con una memoria de 4 K (4.096 muestras) dejaría de adquirir datos al cabo de 16,4  $\mu$ s y no sería posible capturar la segunda ráfaga de datos.

En nuestro trabajo de depuración habitual muestreamos y almacenamos datos durante mucho tiempo cuando no hay actividad. Esto supone un consumo de la memoria del analizador lógico sin obtener información adicional. Este problema se puede evitar si sabemos cuándo se producen las transiciones y si son positivas o negativas. Esta información es la base del análisis de tiempos transicional, que utiliza la memoria de manera eficiente.

Para implementar el análisis de tiempos transicional podríamos utilizar un “detector de transiciones” en la entrada del analizador de tiempos, junto con un contador. El analizador de tiempos sólo almacenará ahora las muestras que vayan precedidas de una transición, junto con el tiempo transcurrido desde la última transición. Con este método empleamos sólo dos ubicaciones de memoria por transición y ninguna memoria si no hay actividad en la entrada.

En nuestro ejemplo, podemos capturar la segunda ráfaga y también la tercera, cuarta y quinta ráfagas, según el número de pulsos por ráfaga que estén presentes. Al mismo tiempo, podemos mantener altas resoluciones de temporización de hasta 4 ns (figura 3).

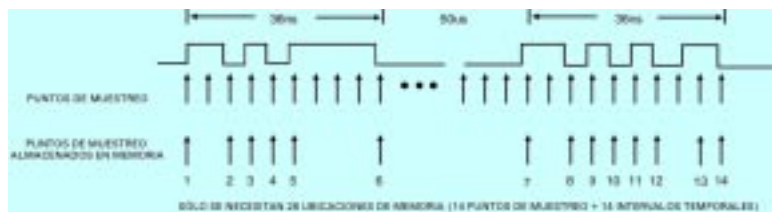


Figura 3. Muestreo con un detector de transiciones.

### Captura de glitches

Los glitches tienen la molesta costumbre de aparecer en los momentos más inoportunos con los resultados más desastrosos.

Un analizador de tiempos muestrea los datos entrantes y realiza un seguimiento de todas las transiciones que se producen entre muestras, por lo que puede reconocer inmediatamente un glitch. En el caso de un analizador, un glitch se define como cualquier transición que cruce el umbral lógico más de una vez entre muestras. Para reconocer un glitch, “enseñamos” al analizador a realizar un seguimiento de todas las transiciones múltiples y mostrarlas como glitches.

Si bien la función de visualización de glitches resulta útil, también puede ser útil disponer de la capacidad de disparar ante un glitch y mostrar los datos que le han precedido. Esto puede ayudarnos a determinar qué ha causado el glitch. Esta función también permite al analizador capturar datos sólo cuando los queremos, cuando se ha producido el glitch.

Pensemos en un ejemplo en el que tenemos un sistema que se bloquea periódicamente porque aparece un glitch en una de las líneas. Dado que se produce con poca frecuencia, el almacenamiento continuo de datos (caso de que dispusiéramos de suficiente capacidad de almacenamiento) genera

ría una cantidad increíble de información que habría que procesar. Otra alternativa sería utilizar un analizador sin función de disparo por glitch y ponerse delante de la máquina pulsando el botón de ejecución esperando a que aparezca el glitch.

### Disparo del analizador de tiempos

Un analizador lógico captura datos continuamente y detiene la adquisición cuando encuentra el punto de traza.

De este modo, un analizador lógico puede mostrar información anterior al punto de traza, lo que se conoce como tiempo negativo, así como información posterior a dicho punto.

### Disparo de patrón

Configurar especificaciones de traza en un analizador de tiempos es algo distinto que configurar el nivel de disparo y pendiente en un osciloscopio. Muchos analizadores disparan ante un patrón de valores altos y bajos en líneas de entrada.

Para facilitar las cosas a algunos usuarios, el punto de disparo de la mayoría de los analizadores puede establecerse en numeración binaria (unos y ceros), hexadecimal, octal, ASCII o decimal. El uso de numeración hexadecimal para el punto de disparo resulta especialmente útil cuando se están examinando buses con una anchura de 4, 8, 16, 24 o 32 bits. Imaginemos lo engorroso que sería definir una especificación para un bus de 24 bits en binario.

Figura 2. Muestreo a alta resolución.

**Disparo de flanco**

Ajustar el botón del nivel de disparo en un osciloscopio equivale a definir el nivel de un comparador de tensión que indica al osciloscopio que dispare cuando la tensión de entrada cruza ese nivel. Un analizador de tiempos funciona prácticamente igual en disparo de flanco, salvo que el nivel de disparo está predefinido en un umbral lógico.

Si bien muchos dispositivos lógicos dependen del nivel, las señales de reloj y control de estos dispositivos son a menudo sensibles al flanco. El disparo de flanco permite iniciar la captura de datos cuando el dispositivo se temporiza.

Puede indicar al analizador que capture datos cuando se produzca el flanco de reloj (subiendo o bajando) y que capture todas las salidas del registro de desplazamiento. Naturalmente, en este caso deberíamos retrasar el punto de traza para atender el retardo de propagación a través del registro de desplazamiento.

**Aspectos básicos del analizador de estados**

Si nunca ha utilizado un analizador de estados, puede pensar que se trata de un instrumento extremadamente complejo cuyo manejo requiere una gran inversión de tiempo. La realidad es que para muchos diseñadores de hardware el analizador de estados es una herramienta muy valiosa.

**Cuándo utilizar un analizador de estados**

Un «estado» de un circuito lógico es una muestra de un bus o una línea cuando sus datos son válidos.

Pongamos el caso de un sencillo biestable D. Los datos en la entrada D no serán válidos hasta

que aparezca un flanco de reloj positivo. De este modo, un estado para el biestable es cuando se produce el flanco de reloj positivo.

Imaginemos ahora que tenemos ocho de estos biestables en paralelo. Los ocho están conectados a la misma señal de reloj. Cuando se produce una transición positiva en la línea de reloj, los ocho capturarán datos en sus entradas D. Una vez más, se produce un estado cada vez que hay una transición positiva en la línea de reloj. Estas ocho líneas son similares a un bus de microprocesador.

Si conectamos un analizador de estados a estas ocho líneas y le indicamos que recoja datos cuando hay una transición positiva en la línea de reloj, el analizador lo hará. El analizador de estados no capturará toda la actividad de las entradas a menos que el reloj esté en un valor alto.

El analizador de tiempos tiene un reloj interno para controlar el muestreo, por lo que muestrea asincrónicamente el sistema sometido a prueba. Un analizador de estados muestrea síncronamente el sistema, ya que toma su reloj de muestreo del sistema.

Un analizador de estados suele mostrar los datos en formato de

lista, mientras que el analizador de tiempos los presenta como un diagrama de formas de onda.

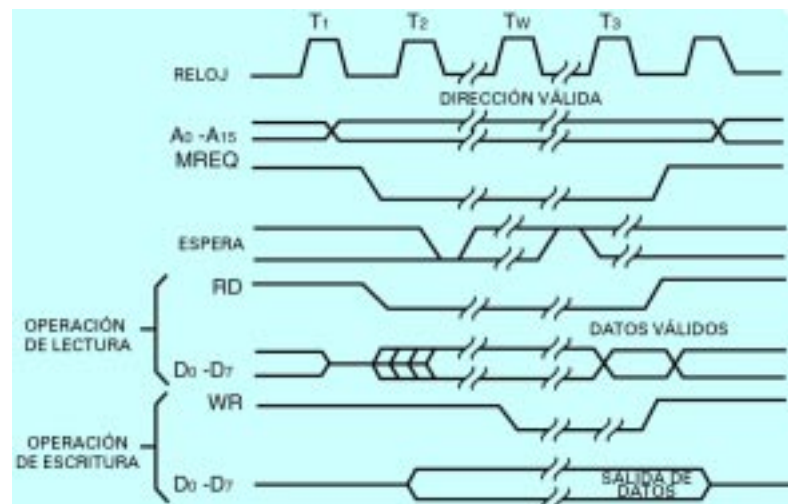
**Funcionamiento de los relojes**

En el analizador de tiempos, el muestreo está bajo la dirección de un único reloj interno. Esto facilita considerablemente las cosas. Sin embargo, en el mundo de los microprocesadores, un sistema puede tener varios «relojes».

Supongamos por un momento que queremos disparar en una dirección específica de la RAM y ver qué datos hay almacenados en ella. Asumiremos que el sistema utiliza un Zilog Z80.

Para capturar direcciones del Z80 con nuestro analizador de estados deberemos capturarlas cuando la línea MREQ esté en un valor bajo. Pero, para capturar datos, necesitaremos que el analizador muestree cuando la línea WR (ciclo de escritura) o RD (ciclo de lectura) esté en un valor bajo. Algunos microprocesadores multiplexan datos y dirección en las mismas líneas. El analizador debe ser capaz de temporizar información de las mismas líneas, pero de relojes diferentes.

Figura 4. Forma de onda de temporización RAM.



En un ciclo de lectura o escritura, el Z80 sitúa primero una dirección en el bus de dirección. A continuación, ejecuta una MREQ, mostrando que la dirección es válida para una lectura o escritura de memoria. Por último, ejecuta la línea RD o WR, según estemos realizando una lectura o una escritura. La línea WR se ejecuta únicamente cuando los datos del bus sean válidos.

Por tanto, un analizador de tiempos actúa como un demultiplexor para capturar una dirección en el momento adecuado y, a continuación, capturar los datos que se producen en las mismas líneas.

### Disparo del analizador de estados

Al igual que un analizador de tiempos, un analizador de estados tiene la capacidad para cualificar los datos que queremos almacenar. Si estamos buscando un patrón específico de valores altos y bajos en el bus de dirección, podemos indicar al analizador que comience a almacenar cuando encuentre el patrón y que siga almacenando hasta que la memoria del analizador esté llena.

La información podrá mostrarse en formato hexadecimal o binario. Puede resultar más útil decodificar de hexadecimal a código de ensamblaje. En el caso de un procesador, caracteres hexadecimales específicos forman una instrucción. La mayoría de los fabricantes de analizadores han diseñado paquetes de software llamados desensambladores o ensambladores inversos. La función de estos paquetes es traducir los códigos hexadecimales a código de ensamblaje para facilitar su lectura.

### Funcionamiento de los niveles de secuencia

Los analizadores de estados tienen "niveles de secuencia" que facilitan el disparo y el almacenamiento. Los niveles de secuencia permiten cualificar el almacenamiento de datos de forma más precisa que con un único punto de disparo. Esto significa que se pueden limitar los datos con precisión sin almacenar información que no necesita. Los niveles de secuencia suelen ser similares a estos:

```
1 find xxxx
else on xxxx go to level x 2 then find xxxx
else on xxxx go to level x 3 trigger on xxxx
```

### El almacenamiento selectivo ahorra memoria y tiempo

El almacenamiento selectivo significa simplemente almacenar sólo parte de un todo mayor. Por ejemplo, supongamos que tenemos una rutina de ensamblaje que calcula el cuadrado de un determinado número. Si la rutina no calcula el cuadrado correctamente, podemos indicar al analizador de estados que capture esa rutina. Para ello, indicamos en primer lugar al analizador que busque el comienzo de la rutina. Cuando encuentre la dirección inicial, le indicamos que busque la dirección final mientras almacena todo entretanto. Cuando se llega al final de la rutina, ordenamos al analizador que deje de almacenar.

### Cómo conectar con el sistema de destino

Hasta ahora hemos hablado de algunas de las diferencias existentes entre osciloscopios y analizadores de tiempos y estados. Antes de que estemos listos para aplicar estas nuevas herramientas, deberíamos tratar otro asunto: el sistema de sondeo.

Una sonda de analizador lógico está diseñada para permitir la conexión de un gran número de canales al sistema de destino fácilmente comprometiendo la precisión de amplitud de la señal sometida a prueba. Tradicionalmente, los analizadores lógicos empleaban vástagos de sonda activos que tenían circuitos integrados de detección de señales para una capacitancia de ocho canales, consiguiendo un total de 16 pF por canal.

### Soluciones de sondas

Las conexiones físicas con los sistemas digitales para de-

Sample Number	Software Address	MPC821/860 PowerQUICC Inverse Assembler
-29	FFFF 3190	addi r11,r11,0x0001
-25	FFFF 3194	addia r12,r0,0x0000
-21	FFFF 3198	stw r11,0x41b0(r12)
-20	0000 4180	mem write 0x00
-19	0000 4181	mem write 0x00
-18	0000 4182	mem write 0x0b
-17	0000 4183	mem write 0x6c
-13	FFFF 319c	addia r12,r0,0x0000
-9	FFFF 31a0	lwr r3,0x41b0(r12)
-8	0000 4180	mem read 0x00
-7	0000 4181	mem read 0x00
-6	0000 4182	mem read 0x0b
-5	0000 4183	mem read 0x6c
-1	FFFF 31a4	bl update display
3	FFFF 3184	wfepc r0,r0
7	FFFF 3186	or r11,r1,r1
11	FFFF 318c	stws r1,0xffff8(r1)
15	FFFF 31c0	bl .text+4A08
19	FFFF 6A08	stw r28,0xffff0(r11)
23	FFFF 6A0c	stw r29,0xffff4(r11)
27	FFFF 6A10	stw r30,0xffff8(r11)

Figura 5. Códigos hexadecimales traducidos a código de ensamblaje.



puración deben ser fiables y cómodas para proporcionar datos precisos al analizador lógico con una intrusión mínima en el sistema de destino que se está depurando.

Una solución de sondas común es la sonda pasiva con 16 canales por cable. Cada canal termina en ambos extremos con 100 kohm y 8 pF. La sonda pasiva se compara mejor eléctricamente con la sonda de osciloscopio. La ventaja del sistema de sondas pasivas, además de su pequeño tamaño y alta fiabilidad, es que permite determinar la sonda justo en el punto de conexión al sistema de destino. De este modo se evita la capacitancia parásita adicional debida a los cables de vástagos activos de mayor tamaño al circuito sometido a prueba. Por eso, el circuito sometido a prueba sólo “ve” capacitancia de carga de 8 pF en lugar de los 16 pF de los sistemas de sondas anteriores.

### Sonda de análisis y otros accesorios

La conexión de un analizador de estados a un sistema de microprocesadores requiere cierto esfuerzo en términos de conexión mecánica y selección de reloj. Recuerde que debemos temporizar el analizador de estados cuando los datos o las direcciones del bus son válidos.

Con algunos microprocesadores puede ser necesario utilizar circuitos externos para decodificar varias señales a fin de obtener el reloj para el analizador de estados. Una sonda de análisis proporciona no sólo una conexión mecánica rápida, fiable y correcta al sistema de destino, sino también la adaptación eléctrica necesaria, como temporización y demultiplexión para capturar correctamente el funcionamiento del sistema.

### Resumen

Este artículo explica qué es y qué hace un analizador lógico. Dado que la mayoría de los analizadores se componen de dos partes principales (analizadores de tiempos y de estados), los hemos tratado por separado. Juntos forman una potente herramienta para el diseñador digital. El analizador de tiempos es más adecuado para estructuras de tipo bus o aplicaciones en las que se tratan muchas líneas. También tiene capacidad para disparar en patrones entre las líneas o en glitches. Un analizador de estados suele considerarse a menudo como una herramienta de software. En realidad, también tiene muchos usos en el dominio del hardware. Puesto que toma su reloj del sistema sometido a prueba, puede servir para capturar datos cuando el sistema los ve, en el reloj del sistema. Ahora que dispone de estos conocimientos básicos puede utilizar un analizador lógico con confianza para depurar sus diseños digitales.

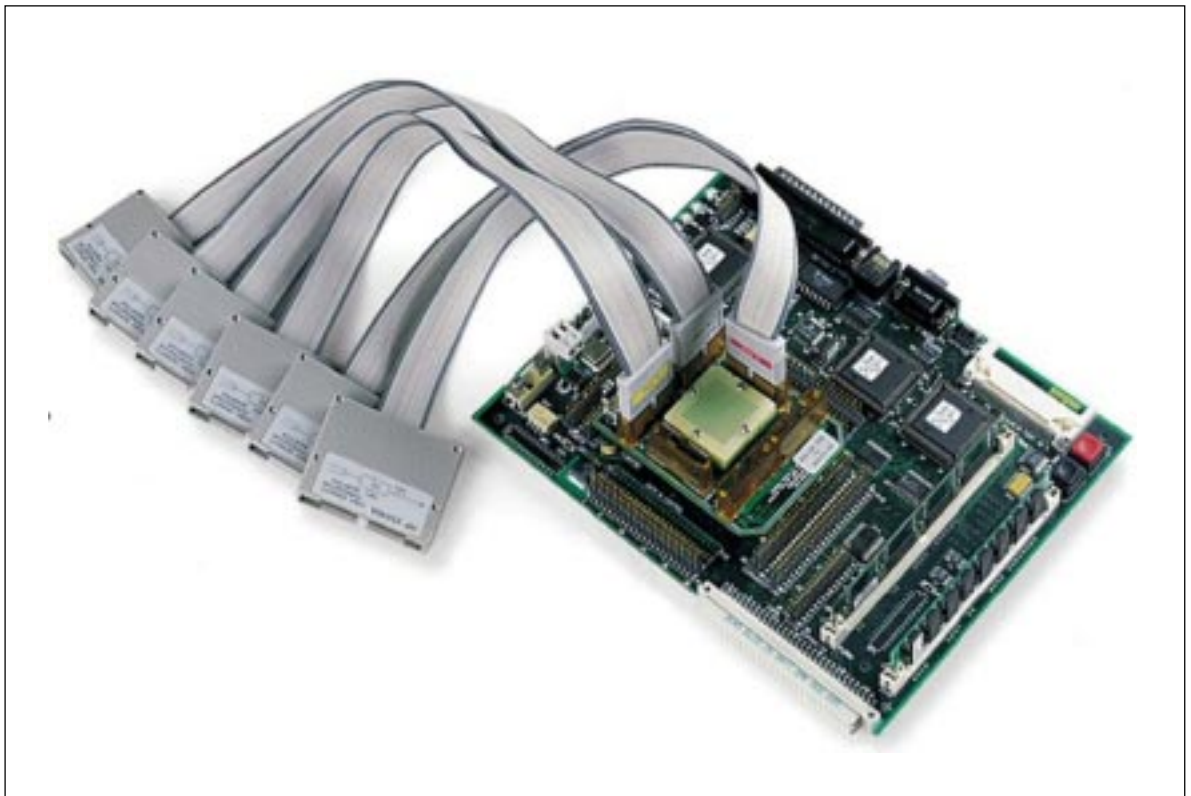


Figura 6. Una sonda de análisis