



Implementación en FPGA de un sistema totalmente digital de recuperación de reloj

Daniel Cárdenas,* Germán Arévalo**
e-mail: dcardenas@ups.edu.ec, garevalo@ups.edu.ec

Resumen

La recuperación de la señal de reloj y de datos CDR es un subsistema importante de cada dispositivo de comunicaciones dado que el receptor debe recuperar la información exacta del reloj del transmisor, misma que está usualmente codificada dentro de la trama de datos entrante. Se han desarrollado algunas técnicas analógicas para CDR basadas en la teoría de PLLs empleando un VCO externo. Sin embargo, en algunas ocasiones la conexión del núcleo digital (FPGA, DSP) con los componentes externos puede resultar complicada. De este modo, el núcleo digital es también utilizado para solventar la tarea de la recuperación de la señal de reloj mediante técnicas totalmente digitales, sin el uso de un VCO externo. El presente artículo describe un subsistema totalmente digital de recuperación de señal de reloj, implementado en un FPGA.

Palabras clave: Recuperación de la señal de reloj y datos CDR, FPGA, DSP, Sincronización, Recuperación del tiempo.

Abstract

Clock and data recovery CDR is an important subsystem of every communication device since the receiver must recover the exact transmitter's clock information usually coded into the incoming stream. Some analogue techniques for CDR have been developed based on PLL theory employing an external VCO. However, sometimes external components could be cumbersome when interfacing them with the digital core (FPGA, DSP) already present in the device. Thus, the digital core is also used to carry out the timing recovery task by all-digital techniques i.e. without an external VCO. This article will describe an all digital timing recovery subsystem using digital techniques implemented on a FPGA.

Keywords: recovering the clock signal and data CDR, FPGA, DSP, synchronization, recovery time.

* PhD en Electrónica, Máster en comunicaciones ópticas, Ingeniero Electrónico, Profesor de la Carrera de Ingeniería Electrónica -UPS- sede Quito
** Master en Comunicaciones ópticas, Ingeniero Electrónico, Director de la Carrera de Ingeniería Electrónica -UPS- sede Quito

Recibido: 12-Agosto-2011; Aprobado: 4-noviembre-2011

Forma sugerida para citar: Cárdenas López, Daniel. (2011). "Implementación en FPGA de un sistema totalmente digital de recuperación de reloj". INGENIUS. N° 6, (julio/diciembre). pp. 19-28 .ISSN: 1390-650X

1. Introducción

La recuperación de la señal de reloj y de los datos es un elemento clave en un receptor de comunicaciones. Dependiendo de las características del transceptor y de todo el sistema de comunicaciones, se pueden implementar distintas soluciones al problema de la correcta recuperación de la señal de reloj y de los datos a partir de la trama de información entrante. En sistemas digitales la solución «tradicional» usa un oscilador controlado por voltaje (VCO) para controlar al muestreador del receptor. Otra solución emplea procesamiento digital sobre la señal con el objeto de recuperar la información correcta a partir de los datos de modo que no requiere un VCO. El presente artículo describe brevemente este último método y muestra, como un ejemplo, una implementación en hardware empleando un arreglo de compuertas programables en campo «FPGA» (Field Programmable Gate Array).

2. Descripción de la recuperación de la señal de reloj

La figura 1 muestra un típico sistema de comunicaciones en banda base PAM donde los bits de información b_k son aplicados a un codificador de línea que los convierte en una secuencia de símbolos a_k . Esta secuencia ingresa en el filtro de transmisión $G_T(\omega)$ y luego se envía a través del canal $C(\omega)$ mismo que distorsiona la señal transmitida e inserta ruido. En el receptor la señal es filtrada por $G_R(\omega)$ con el objeto de extraer las componentes de ruido de la banda correspondiente a la señal y reducir el efecto de la interferencia ISI. La señal a la salida del receptor es

$$y(t; \varepsilon) = \sum_m a_m g(t - mT - \varepsilon T) + n(t)$$

Ecuación 1

Donde $g(t)$ es la señal de pulsos en banda base dada por la función de transferencia $G(\omega)$ (Ecuación 2), $n(t)$ es el ruido aditivo, T es el periodo de un símbolo (transmisor) y εT es el **tiempo de retardo fraccional** (desconocido) entre el transmisor y el receptor, $|\varepsilon| < 1/2$. Los símbolos \hat{a}_k se estiman en base a estas muestras. Estos finalmente se decodifican para dar la secuencia de bits b_k .

$$G(\omega) = G_T(\omega)C(\omega)G_R(\omega)$$

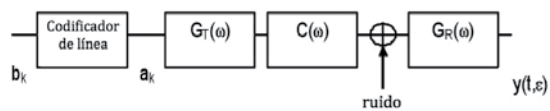
Ecuación 2. Función total de transferencia $G(\omega)$ 

Fig. 1. Sistema básico PAM para comunicaciones en banda base

El receptor no conoce con anticipación los instantes óptimos de muestreo $\{kT + \varepsilon T\}$, de modo que este debe incorporar un circuito para la recuperación de temporización, es decir, un sincronizador de reloj o sincronizador de símbolos, que estime el retardo fraccional ε a partir de la señal recibida.

Dependiendo de su principio de operación se distinguen dos categorías principales de sincronizadores de reloj: sincronizadores de rastreo de error (de bucle hacia atrás o *feed-back*) y sincronizadores de bucle hacia adelante (*feed-forward*) [1].

A. Sincronizador *feedforward*

La figura 2 muestra la arquitectura básica de un sincronizador *feedforward*. Su componente principal es el detector de señal de reloj mismo que calcula directamente el valor del retardo fraccional ε a partir de los datos entrantes. Se

estima el mismo a través de la media de las mediciones tomadas y se lo envía como una señal de control hacia un generador de señal de referencia. La señal de reloj generada finalmente es empleada por el muestreador de datos. [1, 2].



Fig. 2. Sincronizador *feedforward*

B. Sincronizador *feedback*

El componente principal del sincronizador de *feedback* es el detector de error de sincronización, el cual compara los datos PAM entrantes con la señal de referencia, como se muestra en la figura 3. Su salida da el signo y la magnitud del error de sincronización $e = \varepsilon - \hat{\varepsilon}$. El error de sincronización filtrado es utilizado para controlar el muestreador de datos. De modo que los sincronizadores *feedback* usan el mismo principio de un PLL clásico [1, 2].

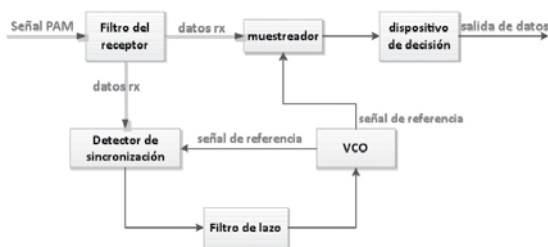


Fig. 3. Sincronizador *feedback*

La principal diferencia entre los dos tipos de sincronizadores es ahora evidente. Los sincronizadores *feedback* minimizan el error en la señal, la señal de referencia es usada para corregirse a sí misma gracias al lazo cerrado; el sin-

cronizador *feedforward* estima la sincronización directamente a partir de la señal entrante y genera la señal de referencia por lo cual no se necesita una retroalimentación (un *feedback*).

Junto a la clasificación previa, se pueden realizar otras. Si el sincronizador utiliza la decisión del receptor al respecto de los símbolos transmitidos para estimar la sincronización, entonces este se denomina de «decisión directa» de otro modo se denomina «no basado en datos». El sincronizador puede a su vez trabajar tanto en tiempo continuo como en tiempo discreto.

3. Arquitecturas de hardware CDR

Se analizan dos opciones: el sincronizador híbrido, que se implementa parcialmente en el dominio digital y parcialmente en el dominio analógico; y el sincronizador digital que opera totalmente en tiempo discreto. Aún cuando la implementación de su *hardware* es diferente, ambos tienen una arquitectura equivalente a la de un sincronizador *feedback* por lo que la teoría para el cálculo de los parámetros del lazo es exactamente la misma.

A. Arquitectura totalmente digital

La figura 4 muestra la arquitectura de un sistema totalmente digital de recuperación de sincronización. El convertidor A/D opera con un oscilador independiente que tiene una frecuencia nominal idéntica a la del convertidor D/A usado en el transmisor. Sin embargo, la relación entre la tasa de los símbolos reales y el reloj de muestreo independiente (de tasa fija) nunca es racional (cambia en el tiempo), es decir, la frecuencia de muestreo y la velocidad de transmisión de los datos son inconmensurables; de modo que el muestreo es asincrónico respecto de los datos entrantes.





Fig. 4. Arquitectura digital

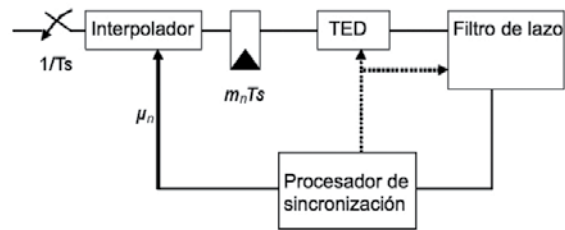
Dado que el reloj de muestreo es un reloj independiente, la sincronización de datos ocurre a través de la interpolación de datos variable en el tiempo con el objeto de «crear las muestras» que hubieran sido obtenidas si el muestreo original hubiera sido con los símbolos.

Luego del interpolador, los datos son enviados al detector de error de sincronización y luego a un filtro de lazo de retroalimentación. La señal de error filtrada controla a un NCO, mismo que cierra el lazo. La salida del NCO da los parámetros correctos para la interpolación.

Esta arquitectura será descrita en más detalla en la sección siguiente.

4. Arquitectura de la recuperación digital de una señal de reloj

La arquitectura de una recuperación digital de señal de reloj se muestra en la figura 5. T_s es el periodo de muestreo asincrónico del convertidor A/D que es inconmensurable con el periodo de los símbolos T . Nótese que aún la más pequeña diferencia entre la señal de reloj del transmisor y del receptor resulta luego de algún tiempo en deslizamientos de ciclos.

Fig. 5. Recuperación digital de la señal de reloj - sincronizador *feedback*

Se deben obtener muestras $y(nT + \hat{\epsilon}T)$, con n enteros, a una tasa de transmisión de símbolos $1/T$ a partir de muestras tomadas a $1/T_s$. De modo que la escala de tiempo del transmisor (definida por T) debe ser expresada en términos de la escala de tiempo del receptor (definida por T_s). La estimación del retardo fraccional de tiempo ϵ es la primera operación importante de la recuperación de la señal de reloj.

Ecuación 3

$$\begin{aligned} nT + \hat{\epsilon}T &= T_s \left[n \frac{T}{T_s} + \hat{\epsilon} \frac{T}{T_s} \right] \\ &= T_s \left[L_{\text{int}} \left(n \frac{T}{T_s} + \hat{\epsilon} \frac{T}{T_s} \right) + \hat{\mu}_n \right] \end{aligned}$$

De aquí que, $y(nT + \hat{\epsilon}T) = y(m_n + \hat{\mu}_n)T_s$

Donde $m_n = L_{\text{int}}(x)$ retorna el entero más grande que sea menor o igual a x , y $\hat{\mu}_n$ es la diferencia entre un instante de muestreo en el receptor y el correspondiente instante de muestra óptima en el transmisor; el índice m_n se denomina el *punto base* y el valor $\hat{\mu}_n$ es la estimación del *retardo fraccional*. Estos conceptos se ilustran en la figura 6.

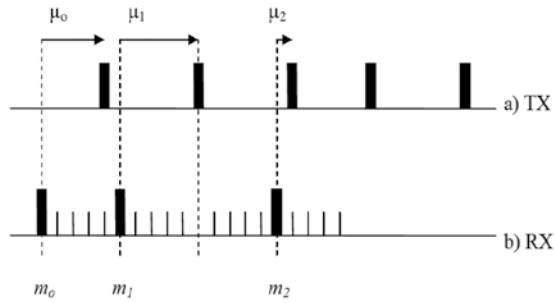


Fig. 6. Escala de tiempo de a) El transmisor y b) El receptor

La figura 6 y la ecuación 3 muestran que el instante correcto de muestreo $nT + \varepsilon T$ puede ser interpolado a partir de un conjunto de muestras definidas por el punto base $m_n T_s$ y que puede ser calculada la diferencia fraccional estimada $\hat{u}_n T_s$ entre el punto base y la nueva muestra. Nótese que el desplazamiento $\hat{u}_n T_s$ es variante aún cuando εT es constante.

La ecuación 3 es la más importante en un sistema totalmente digital de recuperación de señal de reloj. Los parámetros de sincronización (\hat{u}_n, m_n) se calculan una vez que se ha estimado el retardo fraccional de tiempo ε . La segunda función más importante de la recuperación totalmente digital de señal de reloj comprende dos operaciones: la *decimación*, dada por el índice del punto base, y la *interpolación* dada por el retardo fraccional; los valores del punto base y del retardo fraccional se calculan a través del bloque estimador de sincronización en la figura 5. El interpolador variante en el tiempo usa estos valores para calcular el instante óptimo de muestreo. Las siguientes secciones explican en mayor detalle la interpolación, el control de decimación y la estimación de retardo fraccional así como la implementación adoptada para ellos.

A. Detector del error de sincronización (TED)

El detector del error de sincronización TED opera de modo similar al detector de fase en

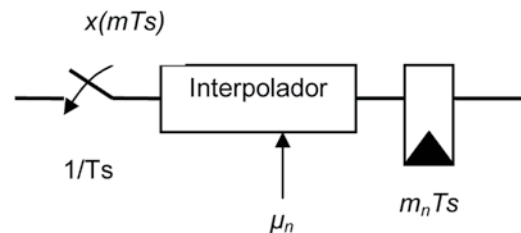
un PLL analógico, es decir, este da la información del error basándose en la diferencia de fase entre la señal entrante y el reloj de referencia.

Existen muchos algoritmos empleados para implementar digitalmente un detector de error de sincronización, dependiendo del factor de sobremuestreo o del formato de modulación [1, 3, 4]. El *hardware* disponible para este prototipo, especialmente el ADC, permite obtener como máximo dos muestras/símbolo; por otra parte, sería mejor si la implementación consigue un buen balance entre complejidad y desempeño, de aquí que se ha seleccionado el detector de error propuesto por Gardner [5, 6].

B. Interpolador digital

La tarea del interpolador es calcular las muestras óptimas $y(nT + \varepsilon T)$ a partir de un grupo de muestras recibidas $x(mT_s)$ como se describe en la ecuación 3. La figura 7 muestra que la interpolación es básicamente un proceso de filtrado variante en el tiempo, dado que T and T_s son inconmensurables.

Fig. 7. Filtro digital interpolador



El filtro interpolador tiene una respuesta impulsiva ideal de la forma $si(x)$ dada por la ecuación 4. Se puede pensar en un filtro FIR con taps infinitos cuyos valores dependen del retardo fraccional μ .

La figura 8 muestra la respuesta del filtro digital cuando $\mu = 0,2$; nótese como la respuesta



varía en función de la gráfica de una repuesta continua centrada en cero.

$$h_n(\mu) = h_1(nT_s, \mu T_s) = \text{sinc} \left[\frac{\pi}{T_s} (nT_s + \mu T_s) \right]$$

Ecuación 4. Respuesta impulsiva de un interpolador ideal

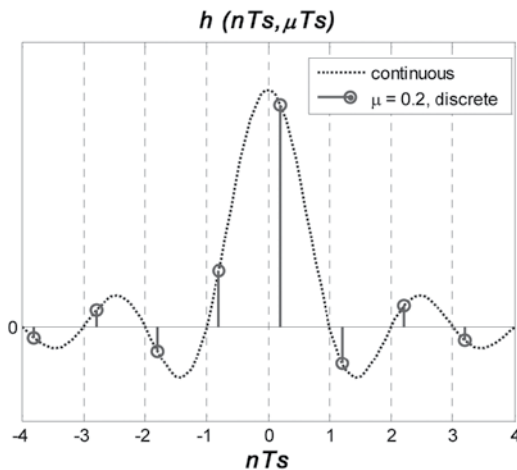


Fig. 8. Respuesta impulsiva de un filtro interpolador ideal

En una implementación práctica, el interpolador puede ser aproximado a un filtro FIR de orden finito (ecuación 5).

$$H(e^{j\omega T_s}, \mu) = \sum_{n=-I_1}^{I_2} h_n(\mu) e^{-j\omega T_s n}$$

Ecuación 5

donde I_1 e I_2 definen respectivamente la menor y la mayor de las muestras de la respuesta impulsiva alrededor del punto central.

La salida del filtro está dada por una combinación lineal de las muestras de la señal ($I_2 + I_1 + 1$) tomadas alrededor del punto de base m_k . Esto conduce a la ecuación 6, misma que es la ecuación fundamental en la interpolación digital. Con el objeto de obtener un único punto de referencia (de base), deberán existir un número par de muestras y la interpolación deberá ser realizada en el intervalo central.

$$y(m_k T_s + \mu T_s) = \sum_{n=-I_1}^{I_2} x[(m_k - n)T_s] h_n(\mu)$$

Ecuación 6. Interpolación digital

Como se mencionó antes, los coeficientes $h_n(\mu)$ del filtro no son fijos y varían de acuerdo a μ . Esto requiere que se limiten los posibles valores de μ y, para cada uno de ellos se debe calcular y almacenar en memoria sus respectivos coeficientes. Los problemas aparecen con la discretización del error en μ , y por la gran complejidad en una implementación en *hardware* real. Una solución común es aproximar cada coeficiente a través de un polinomio en μ .

C. Control del interpolador y NCO

El bloque de control del interpolador calcula el punto base m_n y el retardo fraccional \hat{u}_n basándose en el error de sincronización filtrado.

Los detectores de error producen una señal de error a una tasa $1/T$ usando muestras $kT_1 = kT/M_1$ con M_1 interos (en este caso con $M_1 = 2$ muestras/símbolo). De este modo, por cada muestra el punto base m_k y el retardo fraccional μ_k tienen que ser calculados con el propósito de obtener la muestra interpolada y. La Ecuación 3 ahora se convierte en:

$$y(kT_1 + \varepsilon T_1) = y(L_{int} [kT_1 + \varepsilon_1 T_1] T_s + \mu_k T_s) = y(m_k T_s +$$

Ecuación 7

La figura 9 muestra un diagrama detallado de la arquitectura total de un recuperador digital de señal de reloj. Note que el filtro del lazo decima la salida del TED (a T_s) antes del proceso de filtrado, de modo que la señal de error filtrada se actualiza a una tasa igual a la de los símbolos transmitidos. Esto es una decimación controlada por MI sobre un punto base

$m_k (k = nM_I)$, sujeta al interpolador. Aparte de eso, dado que el interpolador trabaja con M_I muestras por cada símbolo entrante (nominalmente), solamente una de ellas debe ser pasada al resto del subsistema como un valor válido de datos recobrados; esto significa otro proceso de decimación (por M_I) también sujeto al interpolador. Es así que la salida del bloque digital de recuperación de señal de reloj se actualiza a la misma tasa que la de los símbolos.

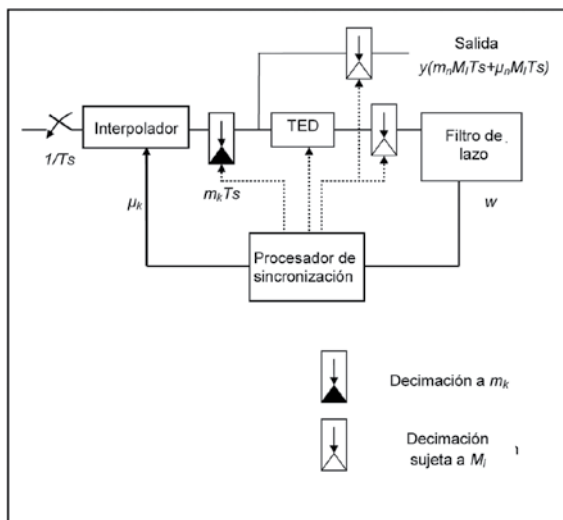


Fig. 9. Arquitectura de la recuperación digital de señal de reloj

La señal de error filtrada w constituye la palabra de control del procesador de sincronización, mismo que calcula el punto base y el retardo fraccional.

Como se puede ver, el procesador de sincronización controla cada bloque en el subsistema digital de recuperación de la señal de reloj en cada ciclo kT_s . Esta tarea se resume en lo siguiente:

Cálculo del punto base m_k o *decimación*, esto involucra el hecho de que el procesador de sincronización selecciona las muestras correctas que se envían a través del interpolador. Si el reloj del receptor es más rápido que la tasa de los bits entrantes, en algún punto una «mues-

tra extra» es tomada por el ADC; el procesador de muestreo no pasa la muestra «extra» al interpolador dado que esta no es útil. Nótese que también el resto los bloques en el subsistema tampoco deben operar con esta muestra. Por otro lado, si el reloj del receptor es más lento que la tasa de los bits entrantes, en cierto punto se pierde una muestra; en este caso no existe decimación para el interpolador, todas las muestras son válidas. Dado que el sistema trabaja con sobremuestreo Mx (M muestras/símbolo), estas decimaciones no se afectan, pues siempre se toma una de cada M muestras.

El procesador de sincronización calcula también el *retardo fraccional* μ_k , de modo que se selecciona la respuesta impulsiva correcta del interpolador.

El procesador de sincronización puede ser llevado a cabo por medio de un NCO [7]. El registro del NCO se calcula iterativamente por:

Ecuación 8

$$n(m_k) = [n(m_k - 1) + w(m_k - 1)] \text{ mod } 1$$

Y el retardo fraccional se estima mediante:

$$\mu_k \approx T_I/T_S n(m_k)$$

Ecuación 9

El prototipo trabaja con dos muestras/símbolo en el transmisor y en el receptor, de modo que nominalmente $T_I/T_S = 1$. Es así que el contenido del NCO es el retardo fraccional.

En lugar de calcular explícitamente m_k , el sobreflujo y subflujo del registro del NCO indica si el reloj del receptor es rápido o lento respectivamente.



D. Filtro de lazo

El procesador basado en un NCO permite considerar la recuperación digital de la señal de reloj como un PLL equivalente que opera a la misma frecuencia de los símbolos. Por lo tanto, el análisis de lazo puede ser llevado a cabo utilizando la teoría clásica de un PLL [8]. La misma consideración aplica para el diseño de un CDR híbrido. El filtro analógico de lazo $F(s)$ debe ser transformado al dominio digital $F(z)$ para poder ser implementado en el FPGA. El diseño considera la transformación bilinear, misma que mapea el lado izquierdo entero del plano- s con el círculo unitario completo del plano- z . De modo que cualquier transformación estable en el tiempo continuo s es mapeada a una transformación z estable en el tiempo discreto. La transformación bilinear se logra mediante la siguiente ecuación.

$$F(s) \rightarrow F(z), \quad s = \frac{2}{Ts} \frac{1-z^{-1}}{1+z^{-1}}$$

Ecuación 10

donde T_s es el periodo de muestreo.

5. Implementación en FPGA y resultados

La figura 10 ilustra la implementación de la solución totalmente digital. El sistema ha sido satisfactoriamente simulado y se ha desarrollado una primera prueba independiente.

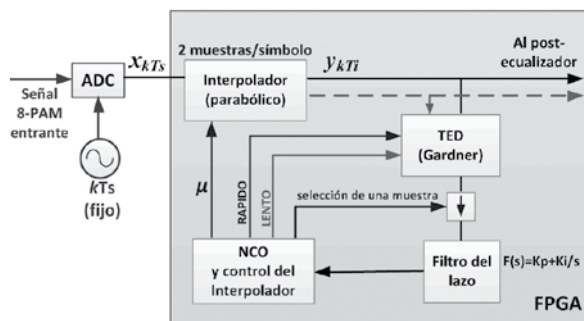


Fig. 10. Arquitectura totalmente digital de la recuperación de señal de reloj

Debe notarse que la interfaz de este módulo totalmente digital con el resto de los subsistemas implica el manejo de la decimación controlada de las muestras interpoladas. El bloque de la recuperación totalmente digital de señal de reloj trabaja con dos muestras por cada símbolo en la entrada y, la teoría dice que debe ejecutarse una decimación por 2 para sacar datos a una tasa equivalente de símbolos. En este caso sin embargo, el ecualizador que sigue a la salida requiere también dos muestras (trabaja a tasa de muestreo); por lo tanto, no se debe realizar una decimación por 2. Se presenta un problema cuando el reloj del receptor es más lento que el del transmisor; usualmente la única muestra obtenida es pasada a la salida, pero en este caso se debe crear dos muestras y en un solo ciclo de reloj pasarlas a la siguiente etapa.

La recuperación totalmente digital de señal de reloj ha sido probada en simulaciones empleando aritmética finita y se han obtenido resultados similares al observarse pruebas independientes desarrolladas en la arquitectura implementada en un FPGA.

La figura 11 ilustra la situación en la cual el reloj del receptor es más rápido que el del transmisor; en este caso una bandera (FLAG RX FAST) indica esta situación y se controla el resto de bloques en la estructura de modo que la muestra extra no se considere en los cálculos. La parte inferior de la Figura muestra el incremento del retardo fraccional en tiempo. Se puede observar que la bandera se activa cuando el retardo fraccional completa un ciclo desde el máximo (1) al mínimo valor (0).

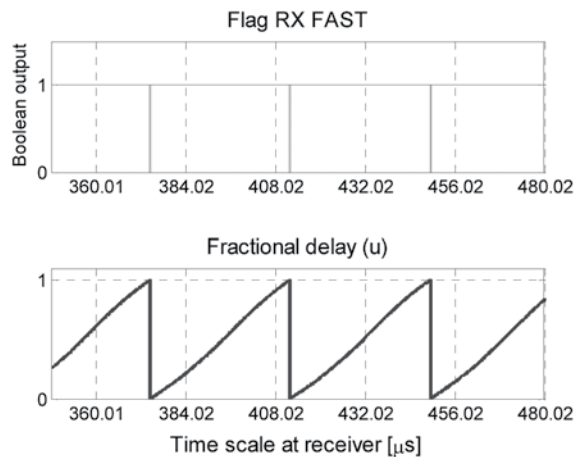


Fig. 11. Recuperación de reloj totalmente digital en el cual el reloj del receptor es más rápido que el reloj del transmisor, la bandera (flag) indica que una muestra no debe ser considerada. Frecuencia nominal del reloj del receptor = 83,33 MHz.

Aún cuando aquí se muestra una pantalla tomada de una simulación, el mismo comportamiento se ha observado en un osciloscopio digital en una prueba real con la primera versión en *hardware* del sistema. Sin embargo, la interfaz de este bloque con el resto de subsistemas está aún en fase de diseño.

6. Conclusiones

Un sistema totalmente digital de recuperación de la señal de reloj de una trama de datos solu-

ciona la complejidad de la interfaz del núcleo digital con elementos analógicos.

Sobre la conclusión anterior y lo expuesto en el artículo, se entiende el porqué sistemas como estos son los más implementados en dispositivos modernos de comunicaciones, tales como los teléfonos celulares.

El sistema propuestos en el presente estudio ha demostrado buenos resultados en las fases de simulación y diseño piloto en FPGA.

7. Agradecimientos

Los autores desean agradecer el apoyo económico y las fructíferas discusiones con el equipo de investigadores del PhotonLab en el ISMB, Turín, Italia.

Referencias bibliográficas

- [1] Heinrich M., Moeneclaey M., Fechtel S. 1998. **Digital Communications Receivers: synchronization, channel estimation and signal processing**. Vol. 2. John Wiley & Sons.
- [2] Mengali U., D' Andrea A. 1997. **Synchronization Techniques for Digital Receivers**. New York: Plenum Press.
- [3] Mueller K, Müller M. 1976. **Timing Recovery in Digital Synchronous Data Receivers**. *IEEE Trans. Commun.* May. COM-24: p. 516-531.



- [4] Oerder M., Meyr H. 1988. **Digital Filter and Square Timing Recovery.** *IEEE Trans. Commun.* May. COM-36: p. 605-612.
- [5] Gardner, F.M. 1986. **A BPSK/QPSK timing-error detector for sampled receivers.** *IEEE Transactions on Communications.* CM-34(5): p. 423.
- [6] Oerder M., Meyr H. 1987. **Derivation of Gardner's Timing Error Detector from the Maximum Likelihood Principle.** *IEEE Trans. Commun.* June. COM-35: p. 684-685.
- [7] Gardner, F .M. 1993. **Interpolation in digital modems. I. Fundamentals.** *IEEE Transactions on Communications.* 41(3): p. 501.
- [8] Gardner, F. M. 2005. 3rd ed. **Phaselock Techniques.** New York: Wiley.

