

# ***DESCRIPCIÓN DEL SISTEMA: DIAGRAMA DE BLOQUES***

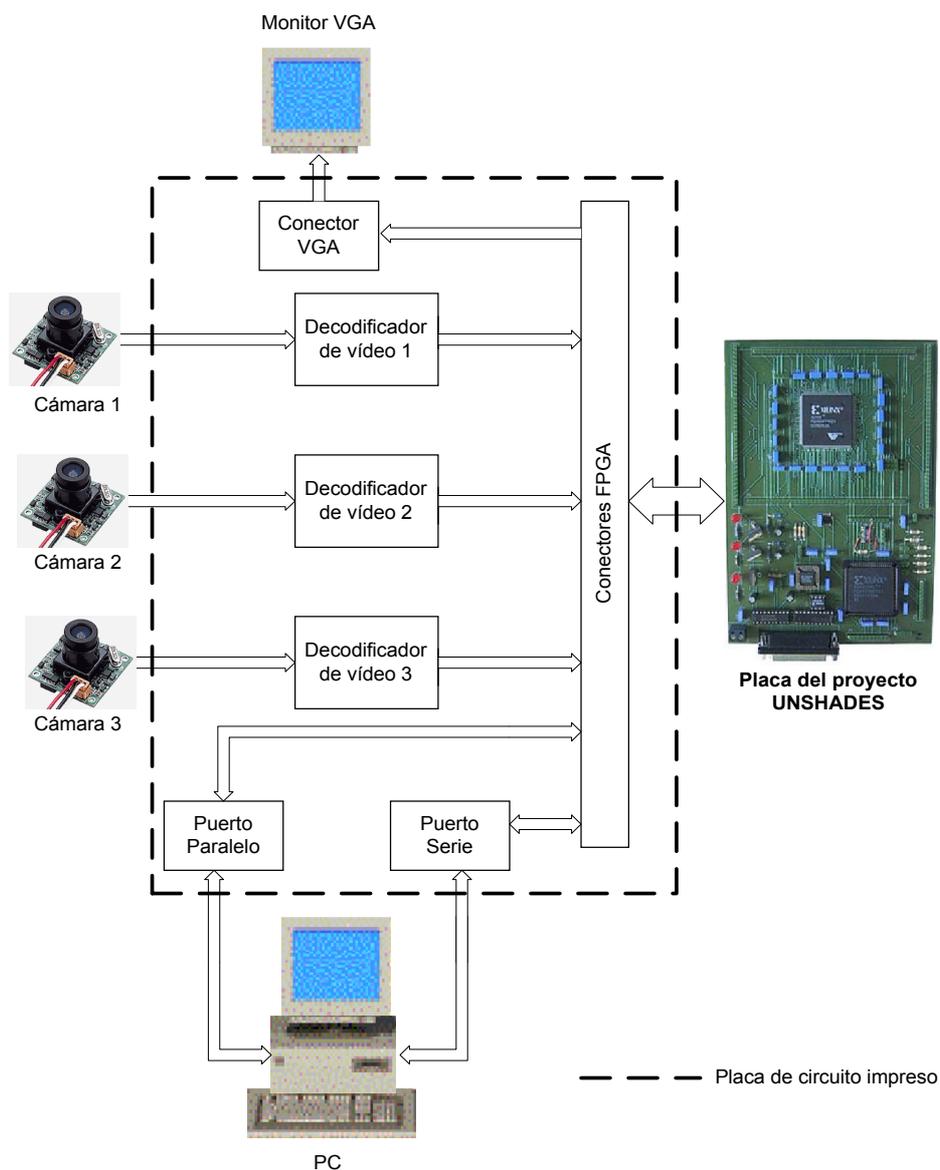
## **4.1. INTRODUCCIÓN**

El sistema de adquisición de datos se compone básicamente de:

- Tres módulos de cámara en Blanco y Negro sub-miniatura en Blanco/Negro TVCCD-30M de MONACOR. Son cámaras de pequeño tamaño que proporcionan señales analógicas de vídeo compuesto (CVBS) en blanco y negro.
- Placa de circuito impreso:
  - Tres decodificadores de vídeo SAA7113H de Philips Semiconductors. (<http://www-us.semiconductors.philips.com/pip/SAA7113H>). Se emplean para digitalizar las señales de vídeo procedentes de las cámaras. Estos procesadores de vídeo junto con el resto de componentes electrónicos forman parte de la placa de circuito impreso o PCB que se va a diseñar en el proyecto, y que se denominará placa del sistema de adquisición o simplemente placa de adquisición.



- Dos conectores peine que permiten la comunicación entre la placa del sistema de adquisición y la FPGA del proyecto UNSHADES. Estos conectores dotarán de un mejor apoyo a la placa de adquisición.
- Un puerto serie y un puerto paralelo, a través de los cuales se comunicará la placa del sistema de adquisición con un PC.
- Un convertidor D/A para la obtención de imágenes VGA de forma directa y de este modo, poder utilizar un monitor VGA como supervisor del sistema.



**Figura 4.1:** Esquema del sistema de adquisición de datos



La placa de circuito impreso diseñada se emplea como interfaz entre la FPGA de UNSHADES-1 y las cámaras de vídeo, un monitor VGA y un ordenador (puertos serie y paralelo). Las imágenes analógicas captadas por las cámaras son digitalizadas y enviadas a la FPGA para su procesamiento. Una vez obtenidos los resultados, es decir, las coordenadas del punto luminoso, se pueden representar en un monitor VGA, eligiendo la perspectiva de una de las tres cámaras, además se transfieren a un ordenador donde, mediante un determinado software, se representarán en 3D. Estos datos podrán ser utilizados también para determinar posibles errores entre la trayectoria real que sigue el punto luminoso y la trayectoria deseada o programada y esta información se emplearía como realimentación de, por ejemplo, un robot de soldadura. Todo ello en tiempo real.

## 4.2. MÓDULO DE CÁMARA SUB-MINIATURA TVCCD-30M

El módulo de cámara sub-miniatura supone el primer elemento de nuestro sistema. Dicho elemento, proporciona las señales de vídeo que serán procesadas en la FPGA Virtex 300 de la tarjeta del proyecto UNSHADES-1, que habrá de programar para ello.

El modelo de cámara TVCCD-30M sólo ofrece imágenes de vídeo en blanco y negro. Así pues, tras analizar el funcionamiento de la videocámara, se realizará una breve descripción de este tipo de señales de vídeo en las que sólo se codifica la *luminancia*.



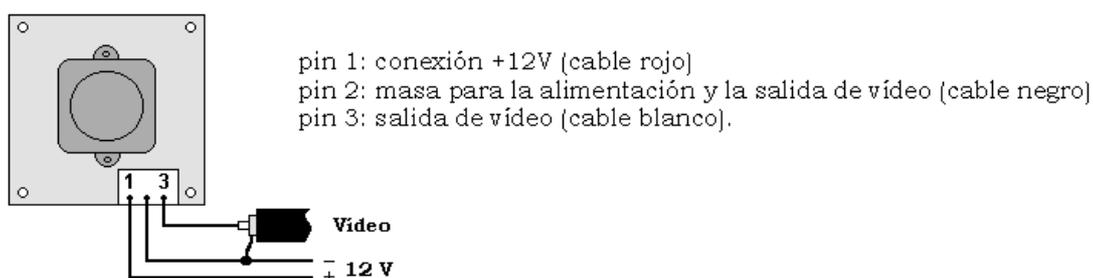
**Figura 4.2:** Cámara TVCCD-30M

## 4.2.1. DESCRIPCIÓN Y CARACTERÍSTICAS DEL MÓDULO DE CÁMARA SUB-MINIATURA TVCCD-30M

Se trata de una pequeña cámara concebida especialmente para su utilización en centrales de vigilancia y de alarma. Sus pequeñas dimensiones permiten utilizarla también como espía de puerta o para vigilancia oculta. Posee una salida de vídeo estándar y la tensión de funcionamiento es de 12VDC.

### 4.2.1.1. CONEXIONES

Su conexión se realiza mediante una toma de 3 polos:



*Figura 4.3: Esquema de conexiones de la videocámara*

### Alimentación

Se debe conectar una tensión estabilizada a 12V al pin 1 (+12V) y al pin 2 (masa). Es importante respetar la polaridad. El consumo es de 110mA aproximadamente.

### Salida vídeo con cable 75 $\Omega$

Se conecta el alma del cable al pin 3 y el blindaje de un cable coaxial de 75 $\Omega$  se conecta a masa (pin 2).



#### 4.2.1.2. CARACTERÍSTICAS TÉCNICAS

Sistema:.....	chip CCD 8.5 mm (1/3").
Sincronización:.....	hor. 15625 Hz, vert. 50 Hz.
Número de puntos <sup>12</sup> :.....	hor. 512 x vert. 582.
Resolución <sup>13</sup> :.....	380 líneas.
Objetivo:.....	1 : 2.0/3.6 mm (92.5°).
Iluminación mínima:.....	0.4 Lux.
Relación señal/ruido:.....	> 45 dB.
Salida vídeo:.....	1 Vcc, 75Ω.
Temperatura de funcionamiento:.....	0-40°C.
Alimentación:.....	12VDC, 110 mA, externa.
Dimensiones (L x A x P):.....	32 x 32 x 28 mm.
Peso:.....	27 g.

#### 4.2.2. FUNCIONAMIENTO DE LA VIDEOCÁMARA

Para comprobar el funcionamiento de la cámara se analizarán las señales de vídeo que proporciona, de este modo, se podrán observar de manera práctica algunas de las características como son: modulación positiva y negativa, exploración de la imagen, sincronismo horizontal y vertical, etc. Con ayuda de un osciloscopio se identificarán cada una de las partes que forman la señal de vídeo que proporciona la cámara.

---

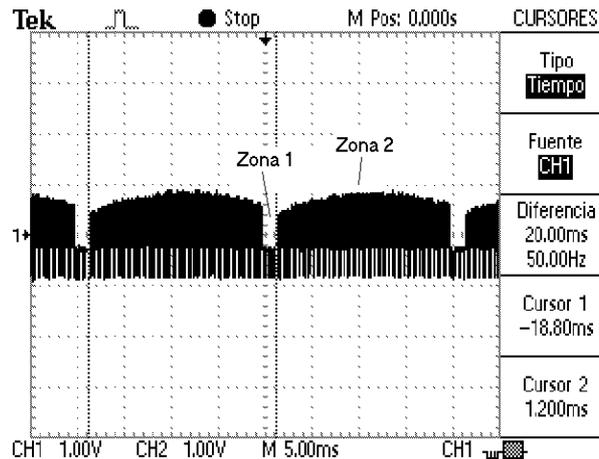
<sup>12</sup> El dato relativo al número de puntos hace referencia al número de muestras de la imagen que el chip CCD es capaz de proporcionar.

<sup>13</sup> El dato de la resolución en número de líneas se corresponde con el número de líneas de vídeo activa que se pueden obtener con esta cámara.



### 4.2.2.1. SINCRONISMO VERTICAL

Para identificar los pulsos de sincronismo vertical, se utilizará una escala de tiempos elevada para tener una visión global de la señal de vídeo. El aspecto de la señal cuando el objetivo está tapado es el siguiente:



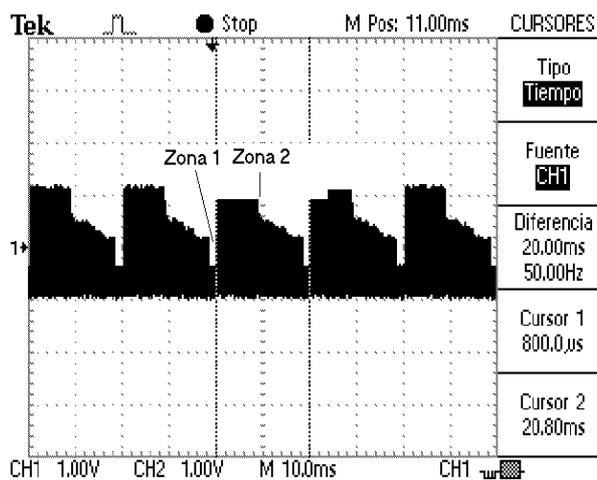
**Figura 4.4:** Sincronismo vertical (de campo) de la cámara con el objetivo tapado

Aparecen una serie de intervalos de forma periódica en los cuales la señal tiene una amplitud menor que en el resto. De hecho, podemos distinguir dos zonas en cada periodo:

- zona 1: donde la amplitud de la señal es inferior y cuya duración es relativamente pequeña.
- Zona 2: donde la amplitud de la señal es superior a la de la zona 1 y cuya duración es mucho mayor.

Si dejamos de tapar el objetivo de la cámara y permitimos que capte la imagen de la sala del laboratorio, la señal que se obtiene en el osciloscopio es la siguiente:





**Figura 4.5:** Sincronismo vertical (de campo) sin tapar el objetivo

Podemos observar que la zona 1 no varía de amplitud, aunque sí lo hace la zona 2, la cual, adquiere una forma irregular que se sigue repitiendo de manera periódica. Esto significa que, la zona que hemos identificado como zona 1 es la que corresponde al *blanking* o *borrado vertical* que se lleva a cabo durante el retroceso que sufre el haz de electrones tras la exploración de uno de los campos (par o impar) que compone la imagen.

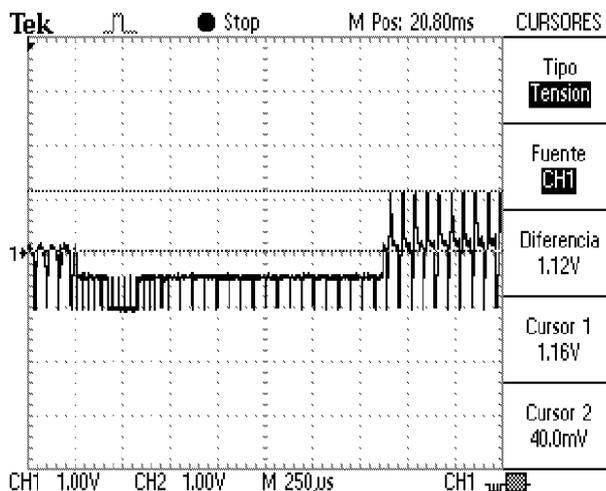
Este hecho, se puede confirmar sin más que medir la duración del periodo de las zonas 1 y 2. Dicho periodo es exactamente 20 ms, lo cual significa que la frecuencia de repetición de estas 2 zonas es de 50 Hz, frecuencia que coincide con el dato de sincronización vertical especificado para la cámara.

Por otra parte, para que el ojo humano no perciba parpadeo, el número mínimo de imágenes por segundo debe ser de 25 imágenes por segundo, esto significa que con una sincronización vertical de 25 Hz es suficiente para tener una imagen de calidad deseada. Sin embargo, la cámara proporciona una frecuencia de sincronización vertical de 50 Hz, esto se debe a que la cámara utiliza **exploración entrelazada**, pues al tener que explorar cada imagen completa dos veces (una para el campo par y otra para el impar), para conseguir las 25 imágenes por segundo se necesita una frecuencia de exploración de campo de valor doble al caso de exploración sencilla (donde hay un solo campo).



También se puede observar que la cámara utiliza *modulación positiva*. Recordando las características de una señal de vídeo en blanco y negro, las zonas de sincronismo estaban marcadas en la señal utilizando amplitudes superiores a la del nivel de negro, en el caso de modulación negativa, e inferiores a la del nivel de negro, en el caso de modulación positiva. El nivel de negro en nuestra señal, es el nivel de amplitud que se mantenía más o menos constante en la zona 2 mientras el objetivo de la cámara permanecía tapado (*figura 4.4*), y se observa que, durante el blanking vertical, la amplitud es menor que el nivel de negro, por tanto, la cámara utiliza *modulación positiva*.

Al medir la duración de la zona de blanking vertical (zona 1), resulta un valor de 1.6 ms. El resto del tiempo se emplea para explorar el campo correspondiente de la imagen. En la figura siguiente se puede observar la zona de blanking vertical ampliada.



**Figura 4.6:** Ampliación de la zona de blanking vertical.

Al principio de la figura aparecen los últimos pulsos de sincronismo horizontal correspondientes a la exploración de las últimas líneas del campo anterior. Después, se observan una serie de impulsos (siempre negativos, por debajo del negro por ser modulación positiva) periódicamente cada  $32 \mu\text{s}$  y de muy corta duración, dichos impulsos aparecen tanto antes como después del pulso dentado de sincronismo vertical y son los llamados *igualadores*. Entre las dos zonas de pulsos igualadores, se distingue una zona donde aparecen una serie de pulsos negativos de gran duración pero pequeña separación, estos pulsos son los que conforman el pulso dentado de sincronismo



vertical. Tras la siguiente zona de pulsos igualadores aparece una nueva donde se generan varios impulsos de sincronismo horizontal. Dichos impulsos, son generados por el emisor (cámara), para que el receptor (monitor o aparato de TV), no pierda el sincronismo horizontal.

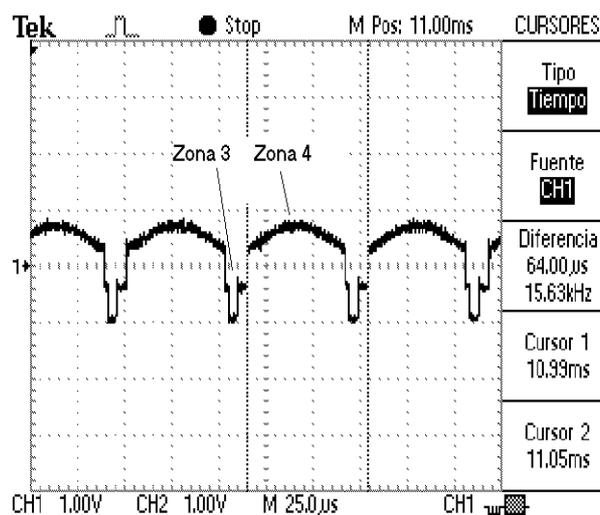
De este modo queda identificada la parte de sincronismo vertical de la señal de vídeo generada por la cámara. A continuación se realizará la identificación del sincronismo horizontal.

#### 4.2.2.2. SINCRONISMO HORIZONTAL

Ampliando la zona identificada anteriormente como zona 2, se puede ver que aparece de nuevo una forma de onda periódica constituida por dos nuevas regiones:

- Una primera región donde se produce un pulso negativo de corta duración ( zona 3).
- Una segunda región donde, si se mantiene el objetivo de la cámara tapado, la amplitud es más o menos constante pero mayor que en la región anterior y de mayor duración ( zona 4).

La figura siguiente muestra la sincronización horizontal de la cámara con el objetivo tapado:

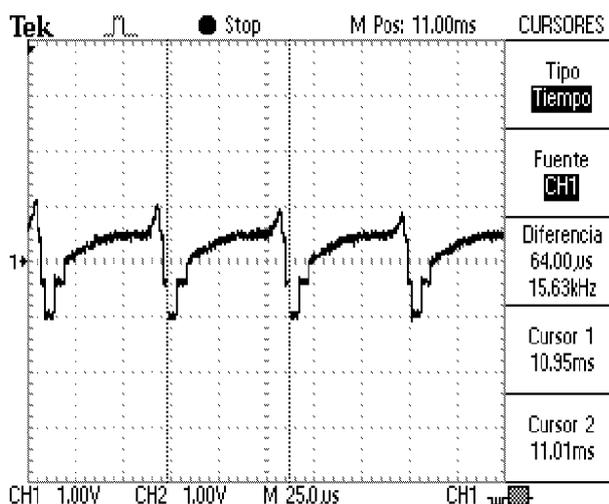


*Figura 4.7: Sincronismo horizontal (de línea) con el objetivo tapado*

La duración del periodo integrado por ambas zonas es de 64  $\mu\text{s}$ . Recordando que el tiempo de *blanking* o *borrado horizontal*, correspondiente a la exploración de una línea y retroceso horizontal del haz, es de 64  $\mu\text{s}$ , se puede concluir que:

- La zona 3 es un pulso de sincronismo horizontal que corresponde al *borrado* o *blanking horizontal* que se realiza durante el retroceso de línea (paso de fin de línea al comienzo de la siguiente línea). La duración de este pulso es de unos 5  $\mu\text{s}$  verificándose el hecho de que dicho pulso dura aproximadamente el 8% del tiempo de exploración y retroceso de una línea (el 8% de 64 $\mu\text{s}$  es 5.12 $\mu\text{s}$ ).
- La zona 4 es la correspondiente a la exploración de la línea más la región final del sincronismo horizontal.

Estas zonas son fácilmente identificables si se destapa el objetivo de la cámara (figura 4.8). En ese caso, la única región que cambia de amplitud, es la de la zona 4, siendo tal región la correspondiente a la exploración de la línea.



**Figura 4.8:** Sincronización horizontal (de línea) con el objetivo destapado

Analizando más detalladamente la zona de blanking horizontal (zona 3) se observa que el pulso de sincronismo horizontal comienza con cierto retraso con respecto al pulso de borrado y su duración es menor que la de éste. Como ya se explicó, dicho pulso debe comenzar con un retraso del 2% del tiempo total de exploración y retroceso de una línea, y terminar un 6% de dicho tiempo antes de comenzar con la exploración de la siguiente línea. Midiendo, se obtiene que el tiempo de retraso inicial es de algo



menos de  $2\mu\text{s}$  (2% de  $64\mu\text{s}$  es  $1.28\mu\text{s}$ ), y el tiempo desde el fin del pulso hasta el comienzo de la exploración de la siguiente línea es de unos  $5\mu\text{s}$  (el 6% de  $64\mu\text{s}$  es  $3.84\mu\text{s}$ ). Así pues, el tiempo total de blanking horizontal es de unos  $12\mu\text{s}$ , que corresponde aproximadamente a algo más del 16% del tiempo total de exploración y retroceso horizontal de línea.

Según los datos de la cámara, el sincronismo horizontal es de  $15625\text{Hz}$ . Calculando la frecuencia de sincronismo horizontal a partir del valor del periodo medido con el osciloscopio se tiene:

$$(1/64\mu\text{s}) = 15625 \text{ Hz}$$

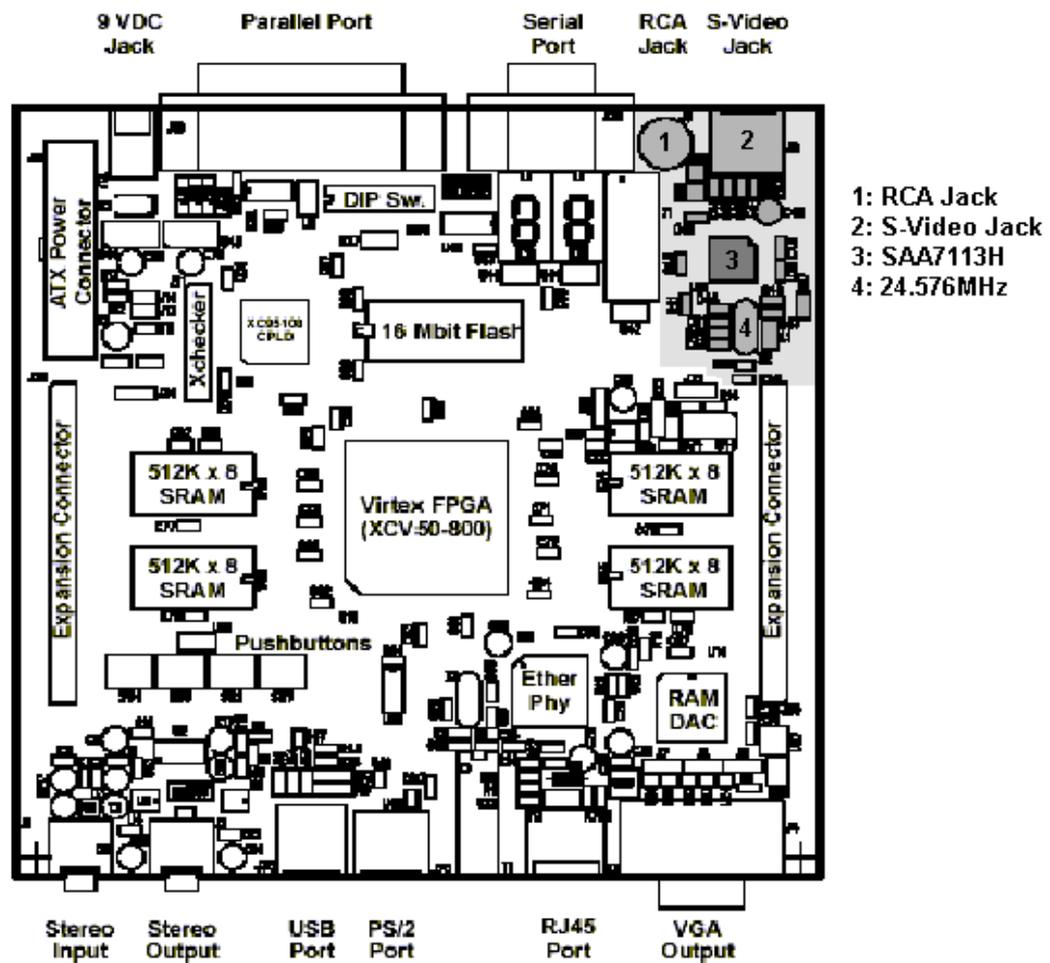
tal y como cabía esperar.

### **4.3. PLACA DE CIRCUITO IMPRESO DEL SISTEMA DE ADQUISICIÓN DE DATOS**

El diseño de la placa del sistema de adquisición se basa en la tarjeta XSV de *XESS Corporation*. Se trata de un sistema electrónico comercial diseñado para el desarrollo de aplicaciones sobre la FPGA de altas prestaciones Virtex de XILINX. Permite el uso de Virtex de distintas capacidades: desde la Virtex 50, con más de 57K puertas de capacidad, hasta la Virtex 800, cuya capacidad supera las 800.000 puertas. La tarjeta XSV, además, dispone de un hardware que permite el procesado de señales de audio y vídeo así como la conexión a numerosos estándares de comunicación. En concreto, el hardware que incorpora la XSV permite digitalizar señales de vídeo PAL, SECAM o NTSC con una resolución de hasta 9 bits.

A continuación se muestra el esquema de la placa XSV, donde se ha señalado en color gris el circuito que se encarga de digitalizar las señales de vídeo.





*Figura 4.9: Esquema de la placa XSV*

La siguiente figura muestra el esquemático del circuito anterior. El decodificador de video SAA7113H será el componente principal del diseño desarrollado en el proyecto. Su funcionamiento se explicará más adelante.



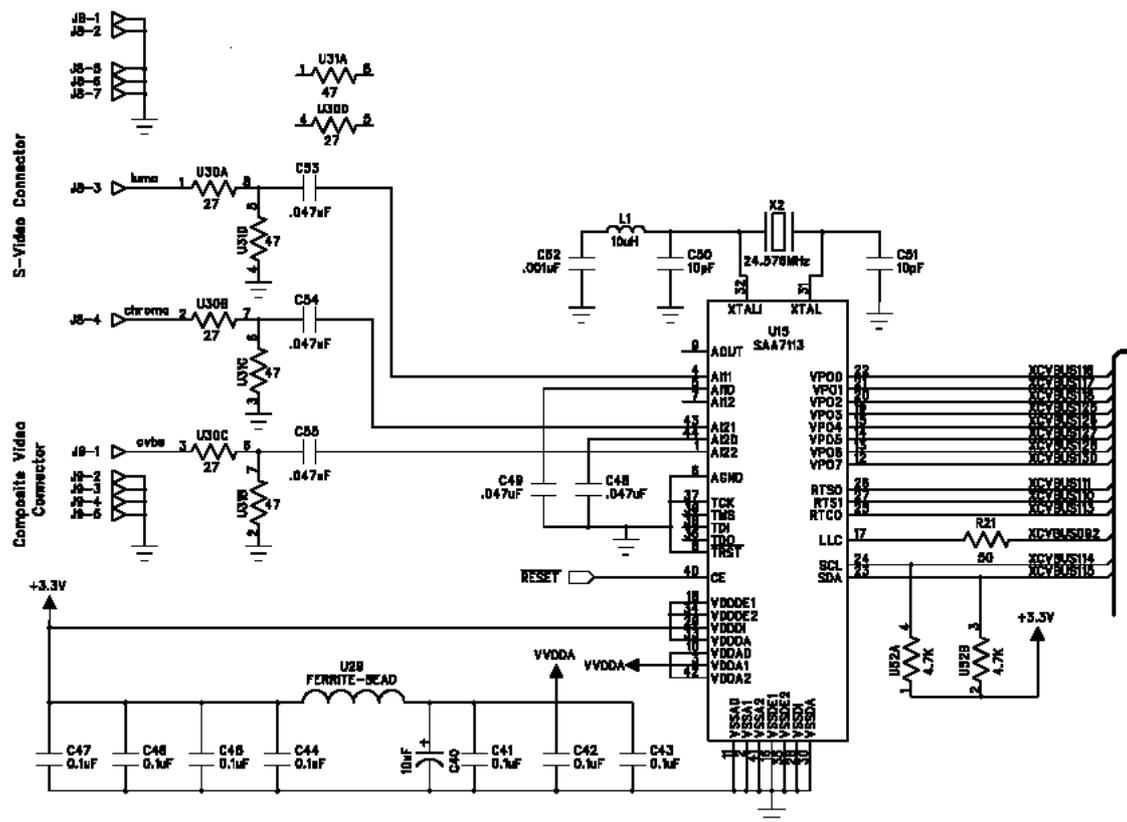


Figura 4.10: Esquemático del circuito digitalizador de señales de vídeo

Debido a la utilización de tres módulos de cámara, la placa del sistema de adquisición consta, entre otras cosas, de tres bloques como el anterior.

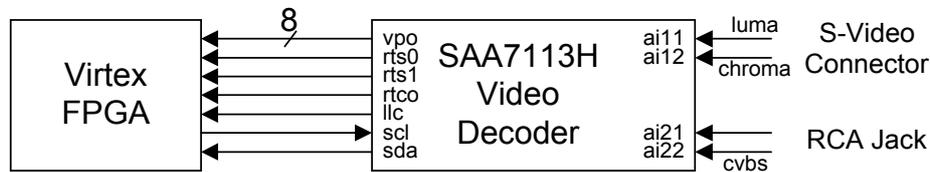
### 4.3.1. DECODIFICADORES DE VÍDEO

Los módulos de cámara sub-miniatura proporcionan *señales analógicas* de vídeo que tendrán que ser procesadas por nuestro sistema. Sin embargo, en la FPGA Virtex del proyecto UNSHADES se va a implementar un sistema completamente *digital*, de ahí la necesidad de utilizar algún componente intermedio que sirva de interfaz entre las cámara y la FPGA. Dicho componente es el **procesador de vídeo SAA7113H**.

Su función es la de proporcionar a la FPGA los datos digitalizados de vídeo a partir de la entrada analógica que proviene de la cámara. Este chip procesador es el



segundo de los componentes que entra en juego en el funcionamiento de nuestro sistema y se encuentra conectado a la FPGA para que ésta pueda controlarlo y configurarlo.



**Figura 4.11:** Conexión de pines comunes con el decodificador de vídeo

#### 4.3.1.1. CARACTERÍSTICAS PRINCIPALES

Las características más importantes que presenta el decodificador de vídeo SAA7113H de Philips se enumeran a continuación:

- Dispone de 4 entradas analógicas de vídeo compuesto (CVBS) que pueden ser usadas también como 2 entradas de vídeo Y/C (componentes de luminancia y crominancia separadas). También cabe la posibilidad de usar 2 entradas para CVBS y las otras dos como 1 única entrada de vídeo Y/C.
- Posibilidad de elección entre programación de la ganancia estática o control automático de ganancia para cada uno de los canales CVBS o Y/C.
- Posee 2 filtros analógicos internos *anti – aliasing*.
- 2 DACs de 9 bits proporcionan los datos digitalizados CVBS o Y/C en el puerto paralelo VPO controlado mediante el *bus I<sup>2</sup>C*.
- PLL digital para detección de sincronismos horizontal y vertical.
- Detección automática de frecuencia de campo: 50 ó 60 Hz. Detección automática de estándar PAL o NTSC.
- Control *I<sup>2</sup>C*, soporta el estándar *fast – mode I<sup>2</sup>C* con una tasa de bits de hasta 400 Kbits/s.
- Decodifica 3 estándares diferentes de vídeo: PAL, SECAM y NTSC.

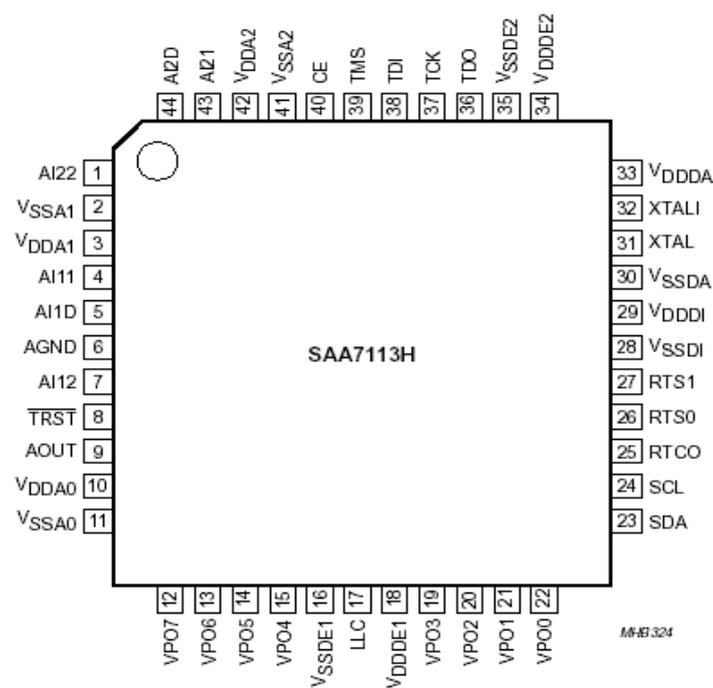
Este procesador de vídeo de 9 bits se compone de los siguientes elementos:



- un circuito de preprocesamiento analógico de 2 canales que incluye selección de fuente.
- Filtro *anti – aliasing* y CAD (convertidor analógico a digital).
- Control automático de ganancia y de *clamping*.
- Un circuito de generación de reloj.
- Un decodificador digital multi–estándar (PAL BGHI, PAL M, PAL N, combinación PAL N, NTSC M, NTSC – Japón, NTSC N y SECAM).
- Un circuito de control de brillo, contraste y saturación.
- Un circuito troceador de datos VBI (*Vertical Blanking Interval*).
- Un circuito *bypass* de datos VBI.

#### 4.3.1.1.1. Descripción de pines

El encapsulado del SAA7113H, que se muestra a continuación, consta de una serie de terminales, de los cuales se dará una breve descripción para comprender un poco mejor las posibilidades del sistema.



**Figura 4.12:** Encapsulado del SAA7113H



**Alimentación de 3.3V:**

<b>Número</b>	<b>Nombre</b>
<i>3</i>	<i>VDDA1</i>
<i>10</i>	<i>VDDA0</i>
<i>18</i>	<i>VDDDE1</i>
<i>29</i>	<i>VDDDI</i>
<i>33</i>	<i>VDDDA</i>
<i>34</i>	<i>VDDDE2</i>
<i>42</i>	<i>VDDA2</i>

**Conexiones a tierra:**

<b>Número</b>	<b>Nombre</b>
<i>2</i>	<i>VSSA1</i>
<i>11</i>	<i>VSSA0</i>
<i>16</i>	<i>VSSDE1</i>
<i>28</i>	<i>VSSDI</i>
<i>30</i>	<i>VSSDA</i>
<i>35</i>	<i>VSSDE2</i>
<i>41</i>	<i>VSSA2</i>

**Bus de datos de salida:**

<b>Número</b>	<b>Nombre</b>
<i>12</i>	<i>VPO7</i>
<i>13</i>	<i>VPO6</i>
<i>14</i>	<i>VPO5</i>
<i>15</i>	<i>VPO4</i>
<i>19</i>	<i>VPO3</i>
<i>20</i>	<i>VPO2</i>
<i>21</i>	<i>VPO1</i>
<i>22</i>	<i>VPO0</i>



**Bus I<sup>2</sup>C:**

Número	Nombre
23	SDA
24	SCL

**Reloj de salida:**

Número	Nombre
17	LLC

**Reloj de entrada:**

Número	Nombre
31	XTAL
32	XTALI

**Control adicional:**

Número	Nombre
25	RTCO
26	RTS0
27	RTS1

**Chip Enable:**

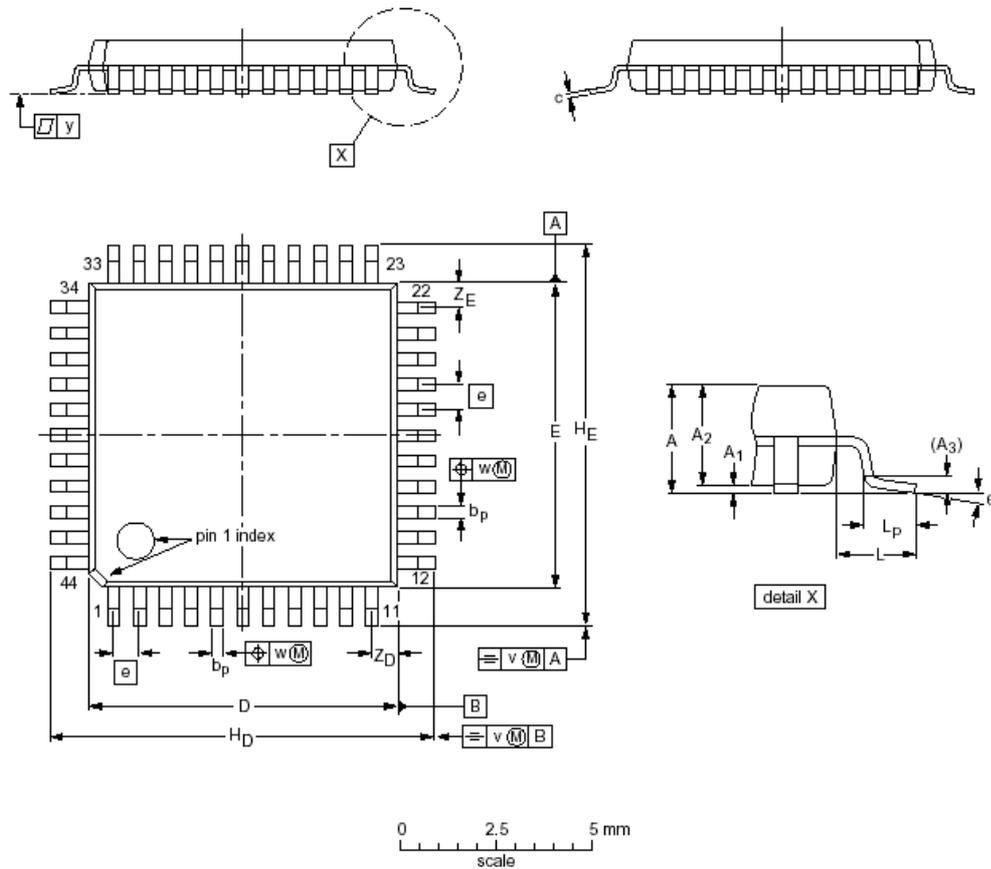
Número	Nombre
40	CE

**4.3.1.1.2. Características mecánicas**

El decodificador de vídeo SAA7113H es un dispositivo de *montaje superficial* cuyo encapsulado es el **QFP44** (*Quad Flat Plastic Package*). Es decir, un encapsulado de plástico, plano y cuadrado, con **44 pines** de 1.3 mm de longitud . Sus dimensiones son: 10 x 10 x 1.75 mm.

En la siguiente figura se muestran con más detalle todas las dimensiones y características del dispositivo.





DIMENSIONS (mm are the original dimensions)

UNIT	A <sub>max.</sub>	A <sub>1</sub>	A <sub>2</sub>	A <sub>3</sub>	b <sub>p</sub>	c	D <sup>(1)</sup>	E <sup>(1)</sup>	e	H <sub>D</sub>	H <sub>E</sub>	L	L <sub>p</sub>	v	w	y	Z <sub>D</sub> <sup>(1)</sup>	Z <sub>E</sub> <sup>(1)</sup>	ϕ
mm	2.10	0.25 0.05	1.85 1.65	0.25	0.40 0.20	0.25 0.14	10.1 9.9	10.1 9.9	0.8	12.9 12.3	12.9 12.3	1.3	0.95 0.55	0.15	0.15	0.1	1.2 0.8	1.2 0.8	10° 0°

*Figura 4.13: Dimensiones del encapsulado del SAA7113H*

#### 4.3.1.2. CONEXIONES CON LA FPGA DE UNSHADES-1

La tarjeta de adquisición de datos, y más concretamente los decodificadores de vídeo, actúan como un interfaz entre las cámaras y la FPGA de la placa UNSHADES-1, tal y como se ha explicado en el apartado 1 del presente capítulo. Por tanto, existe una conexión directa entre los decodificadores y la FPGA Virtex 300.



En primer lugar, la FPGA se encarga de **configurar** los decodificadores para que funcionen del modo más adecuado a la aplicación que se quiera realizar. En nuestro caso, tratar señales de vídeo CVBS. Para llevar a cabo la configuración, se utiliza el **protocolo I<sup>2</sup>C** para la comunicación entre la FPGA y cada decodificador, a través de las líneas **SDA** y **SCL**.

Una vez configurados, los decodificadores comienzan a proporcionar datos digitales de las imágenes procedentes de las cámaras. Dicha información aparece en el bus de salida **VPO** con una frecuencia de 27MHz. Será la línea de reloj de salida **LLC**, la que con su flanco de subida, indique la validez del dato en el bus VPO. La frecuencia de dicho reloj es también de 27MHz.

Otras señales de conexión entre los decodificadores y la FPGA son **RTS0** y **RTS1**, que hacen referencia al **sincronismo horizontal** (principio y fin de línea de vídeo activo) y al **sincronismo vertical** (principio y fin de campo) respectivamente. Dichas líneas se denominarán **HREF** y **VREF**.

A continuación se describirán con más detalle cada una de estas líneas.

#### 4.3.1.2.1. Control I<sup>2</sup>C (SDA, SCL)

Todo el funcionamiento del dispositivo se controla mediante bus I<sup>2</sup>C, es por ello que será necesario comprender el funcionamiento de este bus para poder utilizar este dispositivo. Dicho sistema de control sólo requiere de dos líneas:

- **SDA**, línea bidireccional de datos serie.
- **SCL**, línea serie de reloj.

Ambas líneas se encuentran conectadas a la FPGA de la tarjeta del proyecto UNSHADES-1.



#### 4.3.1.2.1.1. Control del procesador de vídeo: el protocolo I<sup>2</sup>C

Todo el control y configuración del procesador de vídeo se realiza mediante la utilización de un protocolo de comunicación entre circuitos integrados denominado I<sup>2</sup>C. Será necesario implementar un circuito en la FPGA del proyecto UNSHADES-1 que permita la comunicación con el procesador de vídeo utilizando este protocolo.

##### 4.3.1.2.1.1.1. Historia del bus I<sup>2</sup>C

El bus I<sup>2</sup>C fue desarrollado a principios de los años 80 por *Philips Semiconductors*. Su propósito era proporcionar una forma sencilla de conectar una CPU a una serie de chips periféricos en un sistema de televisión.

Los sistemas normales utilizan buses de anchura muy elevada para conseguir este objetivo. Esto da lugar a la necesidad de utilizar numerosas pistas de cobre en las placas PCB de circuitos impresos y una gran cantidad de lógica adicional para poder interconectarlo todo. El uso de muchas líneas de control hace al sistema más susceptible a las perturbaciones electromagnéticas y a las descargas electrostáticas. La necesidad de evitar todos estos problemas desembocó en el desarrollo de un bus de comunicaciones de dos hilos denominado bus I<sup>2</sup>C.

I<sup>2</sup>C son las siglas de Inter-IC bus. Su nombre explica literalmente su propósito: proporcionar un enlace de comunicación entre circuitos integrados.

Hoy en día, el empleo del bus I<sup>2</sup>C se extiende más allá de los equipos de audio y vídeo, y ha llegado a convertirse en un sistema estándar aceptado por la industria.

##### 4.3.1.2.1.1.2. Descripción general

Previamente a la descripción funcional del bus, se van a definir una serie de conceptos necesarios para la correcta comprensión del funcionamiento de este sistema:

- Se define como *transmisor* a aquel dispositivo que transmite señales hacia el bus I<sup>2</sup>C.
- Análogamente, se define como *receptor* a aquel dispositivo que recibe señales desde el bus.



- Al dispositivo que controla la transferencia de señales en las líneas se le denomina *maestro*.
- Y por último, aquel dispositivo que es controlado por el maestro se le llama *esclavo*.

El *maestro* puede transmitir o recibir señales hacia o desde un *esclavo*, o bien, controlar la transferencia de señales entre dos *esclavos* donde uno actuaría de transmisor y otro de receptor. Incluso es posible usar varios *maestros* junto a varios *esclavos* para formar lo que se denomina un sistema *multimaster*. Cuando hay dos o más maestros que intentan controlar el bus simultáneamente existe un procedimiento de arbitraje que decide qué *maestro* tiene prioridad para adoptar el control del bus.

El máximo número de dispositivos conectados al bus depende de la máxima capacidad permitida en las líneas, 400pF, y del límite máximo de direcciones que se establece en 16K.

La simplicidad del sistema I<sup>2</sup>C se debe primeramente al empleo de dos líneas bidireccionales:

- **SDA**, *Serial Data Line*.
- **SCL**, *Serial Clock Line*.

#### 4.3.1.2.1.2. Descripción funcional

Para operar (transmitir o recibir) con un *esclavo* a través del bus I<sup>2</sup>C, sólo es necesario realizar seis operaciones:

- 1º) Generar la condición de **START**.
- 2º) Enviar los 7 bits correspondientes a la dirección del *esclavo* con el que deseamos operar.
- 3º) Enviar un octavo bit que indicará si se realiza una operación de lectura o de escritura sobre el *esclavo*.
- 4º) Esperar la recepción de un bit de acuse de recibo (**ACK**).
- 5º) Enviar el los datos de 8 en 8 bits.
- 6º) Generar la condición de **STOP**.



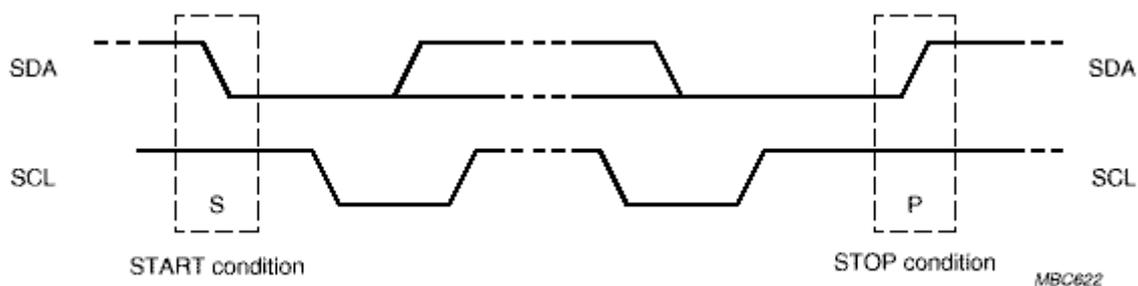
#### 4.3.1.2.1.2.1. Condiciones de START y STOP

Se dice que el bus está en reposo cuando tanto la línea SDA como la SCL se encuentran a nivel alto. Esto ocurre ningún dispositivo actúa sobre al bus, encontrándose éste en alta impedancia a través de un pull-up. Para conseguir el control del bus, un *maestro* debe generar una condición de **START**, del mismo modo que para liberar las líneas, el *maestro* deberá generar una condición de **STOP**.

La condición de START se identifica con una transición de nivel alto a nivel bajo en la línea SDA mientras que la línea SCL se encuentra a nivel alto.

La condición de STOP se identifica con una transición de nivel bajo a nivel alto en SDA mientras SCL está a nivel alto.

Esto puede verse gráficamente en la figura siguiente:



**Figura 4.14:** Condiciones de START y STOP

Siempre es el *maestro* quien genera las condiciones de START y STOP.

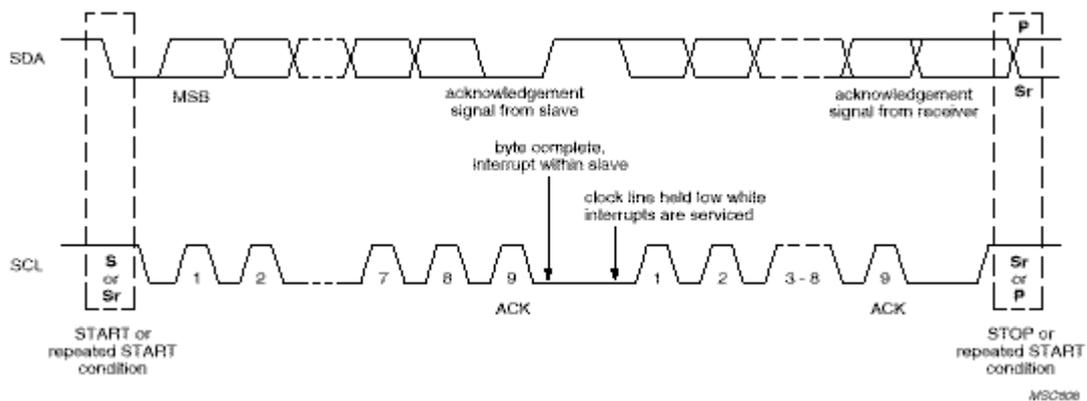
Tras la condición de START, el bus pasa a estar en estado activo y únicamente es liberado cuando se genera una condición de STOP.

#### 4.3.1.2.1.2.2. Transferencia de datos

Después de la condición de START, se genera un bit de datos durante cada pulso de reloj SCL. Dicho bit debe permanecer estable durante todo el tiempo en que SCL permanezca a '1' y la línea de datos SDA sólo podrá cambiar cuando SCL esté a '0'.

Normalmente, cada transferencia de datos se realiza con 8 bits de datos y uno de asentimiento (formato de byte con asentimiento).



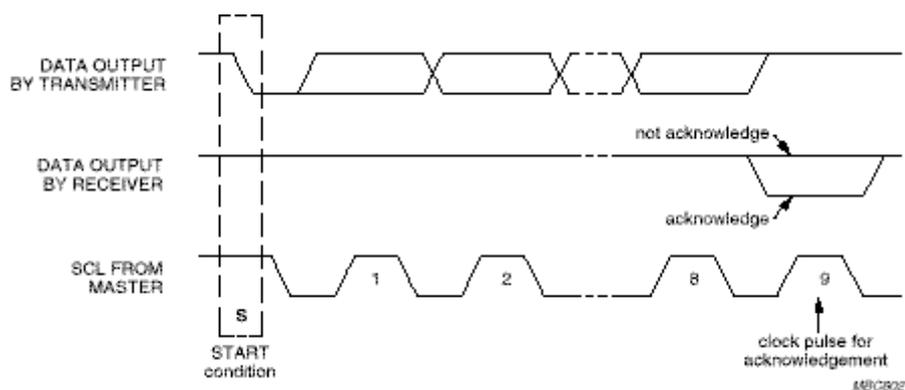


**Figura 4.15:** Ejemplo de transferencia

#### 4.3.1.2.1.2.3. Asentimiento

Todos y cada uno de los bytes enviados deben ser asentidos. El noveno pulso en SCL tras la condición de START se denomina *pulso de asentimiento*. Durante dicho pulso, el transmisor libera la línea de datos (es decir, pasa a nivel alto), y si no se detectó ningún error, será el receptor el que fuerce un '0' en SDA durante el periodo en que SCL permanece a '1'.

Si el *esclavo* no puede asentir, la línea SDA permanecerá a '1' y el *maestro* podrá generar entonces una condición de STOP para abortar la transferencia.



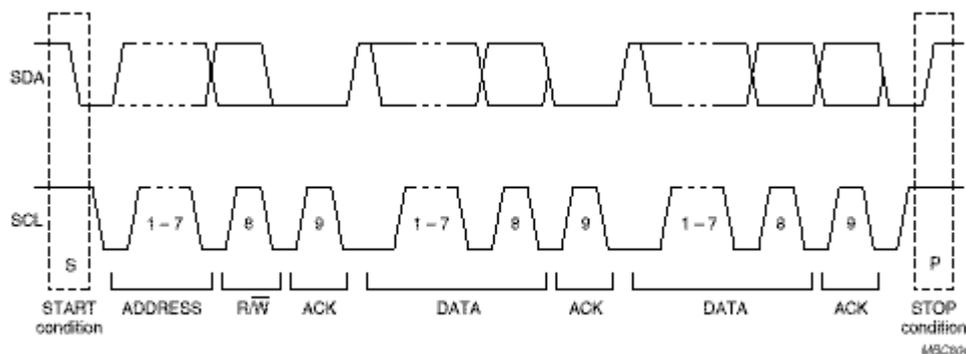
**Figura 4.16:** Asentimiento (ACK)

#### 4.3.1.2.1.2.4. Direccionamiento

El primer byte que se envía tras la condición de START, contiene en sus 7 bits más significativos la dirección del dispositivo I<sup>2</sup>C con el que se desea comunicar, y en el bit menos significativo se codifica el tipo de operación que se va a realizar. Un '0' en el bit menos significativo del primer byte enviado, indica que el *maestro* escribirá información en el dispositivo que posee la dirección seleccionada. Por el contrario, un '1' en dicho bit, indica que el *maestro* leerá datos del *esclavo*.

Algunas direcciones están reservadas para un uso futuro, esto es lo que ocurre con las combinaciones de bits de la forma 1111XXX y 0000XXX. La dirección **00000000** se utiliza como dirección de *llamada general* para todos los dispositivos conectados al bus.

En la figura siguiente se muestra una transferencia I<sup>2</sup>C completa:



**Figura 4.17:** Transferencia de datos completa

#### 4.3.1.2.1.2.5. Arbitraje

El arbitraje del bus sólo tiene sentido en los sistemas *multimaster*. El procedimiento de arbitraje ocurre en la línea SDA mientras SCL está en estado alto y se produce tanto en direcciones como en datos.

Si tenemos varios dispositivos que pueden actuar como *maestros* y más de uno desea acceder al bus, se lleva a cabo un procedimiento de arbitraje. Si un *maestro* transmite un '1' y otro transmite un '0', el *maestro* que fuerza el '0' conseguirá el



control del bus y el otro *maestro* lo liberará. El cambio de *maestro* a *esclavo* se produce de forma inmediata.

#### 4.3.1.2.1.2.6. Formatos

Se soportan tres formatos de transferencia de datos:

- Un *maestro* transmisor escribe en un *esclavo* receptor. No hay cambio en la dirección.
- Un *maestro* inmediatamente después de enviar el byte de dirección.
- Formato combinado con múltiples transferencias de lectura o escritura.

#### 4.3.1.2.1.3. El bus I<sup>2</sup>C del procesador de vídeo

Una vez analizado y comprendido el funcionamiento del protocolo I<sup>2</sup>C pasaremos a ver la manera en que se aplica este protocolo en el caso específico de este dispositivo.

##### 4.3.1.2.1.3.1. Introducción

El procesador de vídeo dispone de 99 *registros internos* que controlan su funcionamiento, algunos de ellos sólo pueden ser leídos y otros están reservados aunque la mayoría pueden ser leídos y programados por el usuario. Todo el funcionamiento del dispositivo se basa en la información con que se programen tales registros. Dicha programación se realiza a través del bus I<sup>2</sup>C.

La FPGA deberá encargarse de la programación de los registros internos del procesador de vídeo para su adecuado funcionamiento. El hecho de que el control de todo el sistema lo llevará a cabo el circuito implementado en la FPGA permite realizar una serie de simplificaciones sobre la parte de control I<sup>2</sup>C que habrá que implementar sobre la FPGA:

- En primer lugar, no será necesario ejercer ningún tipo de arbitraje ya que nuestro sistema tiene un único dispositivo maestro: la FPGA Virtex 300 de la placa del proyecto UNSHADES-1.



- En segundo lugar, el sistema tiene **tres** dispositivos *esclavos*: los procesadores de vídeo.
- Por último, la FPGA sólo tendrá que realizar *operaciones de escritura* mediante el I<sup>2</sup>C sobre los registros internos de los decodificadores de vídeo, y estas operaciones las llevará a cabo una sola vez.

Por estas tres razones podremos implementar un circuito de control I<sup>2</sup>C más sencillo, sin necesidad de tener que implementar el circuito que se ajuste al protocolo completo.

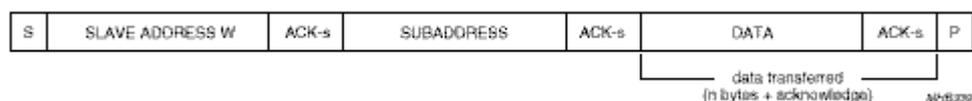
#### 4.3.1.2.1.3.2. Fast Mode I<sup>2</sup>C

El bus I<sup>2</sup>C básico soporta velocidades de transferencia de datos de hasta 100 Kbits/s y un direccionamiento de 7 bits. Sin embargo, la mejora de los sistemas ha permitido incrementar la velocidad de transferencia de datos a través del bus dando lugar al Fast Mode I<sup>2</sup>C. Esta nueva especificación soporta tasas de bit de hasta 400 Kbits/s sin perder la compatibilidad con el modo estándar.

Una de las características del procesador de vídeo que se utilizarán para digitalizar las señales procedentes de las cámaras, es que soporta la especificación *Fast Mode* del bus I<sup>2</sup>C.

#### 4.3.1.2.1.3.3. Formato del bus I<sup>2</sup>C del procesador de vídeo

La figura siguiente muestra el formato I<sup>2</sup>C del procedimiento de escritura en un registro del procesador de vídeo:



**Figura 4.18:** Procedimiento de escritura

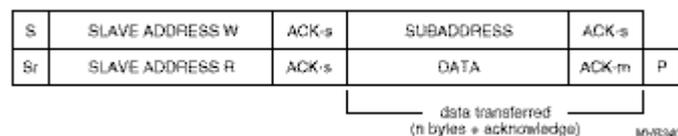


Según esta figura, la FPGA deberá comenzar el procedimiento de escritura en el procesador de vídeo generando una condición de START. Seguidamente deberá enviar la dirección de esclavo I<sup>2</sup>C del procesador. Dicha dirección se encuentra codificada en los siete bits más significativos del byte, siendo el bit menos significativo, bit R/W, el que indica que se trata de una operación de escritura. La *dirección de esclavo* de este dispositivo es la **4AH** (la ‘H’ indica que se trata de un valor hexadecimal) por defecto aunque puede ser cambiada colocando una resistencia entre el pin RTS0 y tierra. El bit menos significativo del primer byte será un ‘0’ (4AH = 0100 1000) indicando que se trata de una operación de **escritura**.

A continuación, la FPGA debería recibir el ACK del byte enviado. Si es así, enviará inmediatamente como primer byte de datos, la subdirección del registro interno que desea programar. De nuevo, deberá recibir un asentimiento y, por último, enviará el dato a escribir sobre el registro indicado. Tras recibir el asentimiento del último dato, la FPGA generará la condición de STOP.

Como puede observarse, se respeta el protocolo I<sup>2</sup>C tal como lo hemos explicado anteriormente.

La figura siguiente muestra el procedimiento de lectura:



**Figura 4.19:** Procedimiento de lectura (formato combinado)

En este caso, se utiliza un formato combinado. Las dos primeras transferencias de datos son similares a las realizadas en el procedimiento de escritura, sin embargo, tras enviar el segundo byte (el que indica la subdirección del registro sobre el que deseamos realizar la operación de lectura), la FPGA deberá generar una nueva condición de START llamada condición de *START repetida*. A continuación, el maestro, es decir, la FPGA, deberá enviar la dirección de lectura del esclavo que, por defecto, es la 4BH (= 0100 1011). El último bit a ‘1’ indica que se va a realizar una operación de lectura. Después de haber realizado la transferencia correcta de este byte,



la FPGA pasará a ser el receptor y esperará a recibir el dato procedente del registro sobre el que se está realizando la operación de lectura. Tras recibir el último bit, la FPGA deberá enviar un ACK al procesador de vídeo y, posteriormente, generará la condición de STOP.

En la siguiente tabla se muestra el significado de la simbología usada en las figuras anteriores y también se da una lista de los registros internos que posee el procesador de vídeo.

CODE	DESCRIPTION	
S	START condition	
Sr	repeated START condition	
Slave address W	0100 1010 (= 4AH, default) or 0100 1000 (= 48H, if pin RTS0 strapped to ground via a 3.3 k $\Omega$ resistor)	
Slave address R	0100 1011 (= 4BH, default) or 0100 1001 (= 49H, if pin RTS0 strapped to ground via a 3.3 k $\Omega$ resistor)	
ACK-s	acknowledge generated by the slave	
ACK-m	acknowledge generated by the master	
Subaddress	subaddress byte; see Table 24	
Data	data byte; see Table 24; note 2	
P	STOP condition	
X = LSB slave address	read/write control bit; X = 0, order to write (the circuit is slave receiver); X = 1, order to read (the circuit is slave transmitter)	
Subaddresses	00H chip version	read only
	01H to 05H front-end part	read and write
	06H to 13H decoder part	read and write
	14H reserved	–
	15H to 17H decoder part	read and write
	18H to 1EH reserved	–
	1FH video decoder status byte	read only
	20H to 3FH reserved	–
	40H to 60H general purpose data slicer	read and write
	60H to 62H general purpose data slicer status	read only
	63H to FFH reserved	–

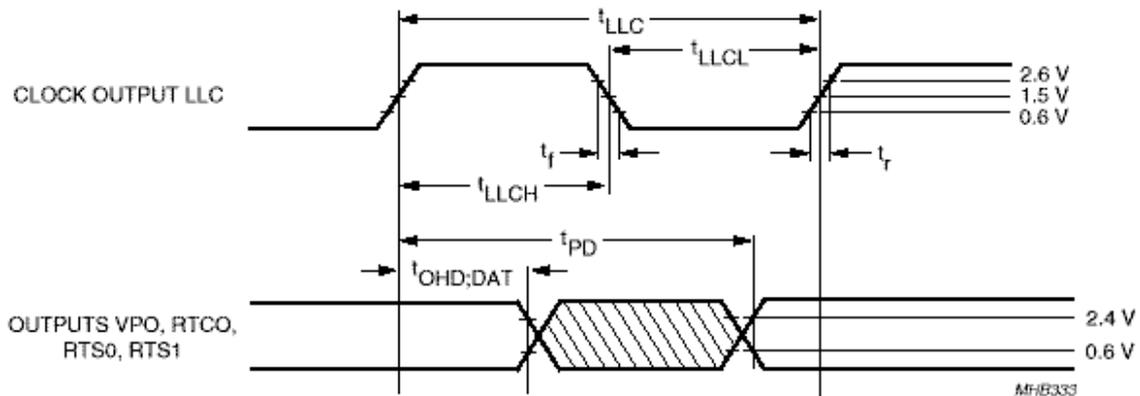
**Tabla 4.1:** Significado de las abreviaturas usadas y registros internos del decodificador

En el *datasheet* del dispositivo que se proporciona al final de esta memoria pueden verse con más detalle los registros internos del procesador de vídeo así como los nombres de los bits y la explicación de cada una de sus funciones (páginas 49 a 72).



#### 4.3.1.2.2. Bus de datos (VPO) y reloj de salida (LLC)

El decodificador se basa en el principio de decodificación *Line Locked Clock (LLC)*, esto es, los datos decodificados aparecerán en el bus de salida VPO con cada flanco de la señal LLC, cuya frecuencia es de 27 MHz y se encuentra en conexión directa con la FPGA.



**Figura 4.20:** Temporización de las salidas respecto a LLC

El puerto digital de salida (VPO7-VPO0) es un puerto de 8 bits que puede proporcionar 16 tipos de datos en tres formatos diferentes, que se pueden seleccionar con los registros de control LCR2 a LCR24 (registros de subdirecciones 41H a 57H).

Los tres formatos de salida posibles son:

- SLICED, es un formato que soporta varios estándares ya definidos.
- YUV 4:2:2, es un formato que proporciona las componentes de luminancia (Y) y crominancia (U, V). Genera 720 píxeles activos por línea.
- RAW, en este caso sólo se proporcionan valores sobremuestreados de luminancia de una señal CVBS. La tasa de datos proporcionada es de 27 MHz. Se puede programar el rango horizontal mediante los registros HSB (Horizontal Sync Begin), HSS (Horizontal Sync Stop) y HDEL1 a HDEL 0 en los registros internos de subdirecciones 06H, 07H y 10H.

El formato de datos RAW será el que nosotros usemos. Este formato es el más adecuado debido a que las cámaras proporcionan señales de vídeo CVBS en las que sólo se codifica la luminancia por ser señales de vídeo en blanco y negro (como ya se explicó con anterioridad).

Los registros LCR2 a LCR24 controlan el tipo de datos para cada una de las líneas que componen el campo de la imagen. El registro LCR2 se refiere a la línea 2, el LCR3 a la línea 3, y así sucesivamente para todas las líneas del campo hasta la número 23. El tipo de datos programado en LCR24 es válido para todo el resto de las líneas del campo, por tanto, sólo las primeras pueden ser programadas con tipos de datos diferentes. No obstante, mediante los bits VOFF8 a VOFF0 se puede variar la relación entre los registros LCR y las líneas del campo a las que se refieren. El valor recomendado es 07H para fuentes de 50 Hz (como es el caso de nuestras cámaras).

El bus VPO puede contener bien los 8 bits más significativos de los convertidores analógico-digital (CADs) o bien los 8 menos significativos dependiendo del valor programado en el bit ADLSB del registro de subdirección 13H.

La configuración de la entrada analógica se realiza mediante la programación de los bits MODE3 a MODE0 (subdirección 02H). En el caso de que el modo programado sea CVBS sólo se proporciona la salida de uno de los CAD, por el contrario, si se programa un modo YC, el bus VPO transporta los datos de salida de ambos CADs multiplexados.

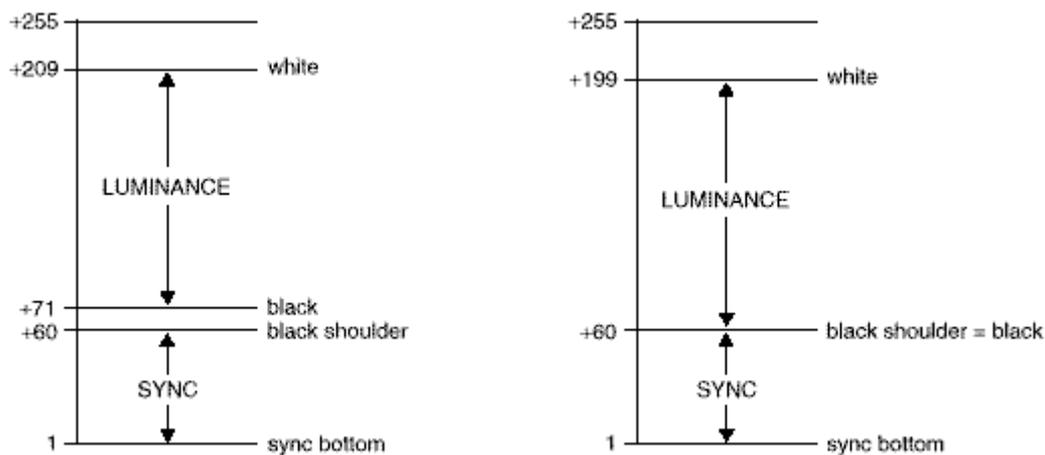
Por tanto, según lo visto hasta ahora, el formato de datos RAW parece ser el más apropiado en nuestro caso por proporcionar como datos de salida, valores sobremuestreados de luminancia de la señal de entrada.

#### **4.3.1.2.2.1. Formato de datos RAW**

Para conseguir una tasa de datos RAW de 27 MHz, la señal CVBS digitalizada es sobremuestreada después de la conversión AD. La supresión de las componentes de frecuencia laterales no deseadas de la señal muestreada se realiza mediante el uso de un filtro de interpolación.



En cuanto a los niveles de salida que se obtienen en el bus VPO podemos verlos en la figura siguiente:



**Figura 4.21:** Niveles RAW con pedestal (izquierda) y sin pedestal (derecha)

El formato de datos RAW es el que se muestra a continuación:

BLANKING PERIOD			TIMING REFERENCE CODE					OVERSAMPLED CVBS SAMPLES							TIMING REFERENCE CODE			BLANKING PERIOD				
...	80	10	FF	00	00	SAV	Y0	Y1	Y2	Y3	Y4	Y5	...	Yn-1	Yn	FF	00	00	EAV	80	10	...

**Table 16** Explanation to Table 15

NAME	EXPLANATION
SAV	start of raw sample range; see Tables 5 to 7
Yi	oversampled raw sample stream (CVBS signal), $n = 0, 1, 2, 3$ to $n$ ; $n$ is programmable via HSB and HSS; see Sections 15.2.7 and 15.2.8
EAV	end of raw sample range; see Tables 5 to 7

**Figura 4.22:** Formato de datos RAW

Yi representan los bytes que contienen los niveles correspondientes a las muestras RAW sobremuestreadas (señal CVBS). El número de muestras es programable mediante los registros HSB y HSS.



SAV y EAV son códigos de referencia temporal que definen el comienzo y final respectivamente de las regiones de datos válidas. El formato de estos bytes es el que se muestra en la página 27 del *datasheet*<sup>14</sup> del procesador de vídeo.

El bit número 6, bit F, es el bit de identificación de campo, mientras que el bit 5, bit V, es el bit de identificación de blanking vertical. Ambos bits se usan para la sincronización vertical de la imagen: el primero para identificar el campo de la imagen que está siendo explorado, y el segundo para diferenciar la región de vídeo activo (exploración) de la de borrado o blanking vertical (retroceso vertical).

A continuación aparece el bit H que identifica si el byte representa el comienzo de la región de datos válida o, por el contrario, el final de la misma. La generación del bit H y, consecuentemente, la temporización de SAV/EAV depende del formato de datos seleccionado. H permanecerá a '0' durante la región de datos activos. Para todos los formatos de datos, excepto para los datos RAW, la longitud de la región activa de datos es 1440 x LLC. Para el caso concreto de datos RAW, la longitud de la región activa de datos es programable.

Durante el periodo de blanking horizontal entre EAV y SAV, se transmite la secuencia '-80-10-80-10-...' que corresponde al código de secuencia blanking ITU.

La posición del bit F es constante de acuerdo a la recomendación ITU 656.

El bit V puede ser generado de cuatro formas diferentes según los valores programados en los bits OFTS0, OFTS1 y VRLN(subdirección 10H), y LCR2 a LCR24 (subdirecciones 41H a 57H).

En nuestro caso, se tienen **625 líneas** con sincronización vertical de **50 Hz** que es lo que proporciona la cámara, por tanto, los valores de los bits F y V son los expresados en la tabla correspondiente de la página 28 del *datasheet*. F y V cambian sincronamente con el código EAV.

Podríamos pensar en utilizar estos formatos de secuencia a la hora de sincronizarnos con los datos de imagen proporcionados por el procesador de vídeo, sin embargo, para ello, la sincronización con la línea LLC debe ser exacta. En nuestro caso, el reloj maestro que controlará el funcionamiento de nuestro circuito no podrá ser programado a 27MHz pues, el reloj DALLAS 1075 incluido en la tarjeta UNSHADES

---

<sup>14</sup> Se adjunta en el Anexo.



sólo puede ser programado dividiendo por números enteros la frecuencia base de 100MHz. Esto nos va a plantear un problema de sincronización, cuya repercusión explicaremos más adelante.

#### 4.3.1.2.3. Sincronismo horizontal y vertical (RTS0, RTS1)

Estos dos pines son pines I/O multifuncionales controlados vía I<sup>2</sup>C mediante la programación de los bits RTSE03 a RTSE00 y RTSE13 a RTSE10, situados en la subdirección 12H.

El terminal RTS0 puede ser conectado a tierra a través de una resistencia de 3.3K $\Omega$  para cambiar la dirección del dispositivo cuando actúa de esclavo I<sup>2</sup>C. La dirección por defecto es 4AH para escritura y 4BH para lectura.

El terminal RTS1 es un pin I/O. Cuando está configurado como salida, su funcionamiento es similar al de RTS0. Pero si lo configuramos como entrada, podremos controlar el estado de alta impedancia del bus VPO. Para usarlo como entrada, habremos de configurarlo como DOT (Data Output to 3-state) escribiendo “0000” en el registro de subdirección 12H (bits D7 a D4). Además, tendremos que escribir un ‘1’ en el bit OEYC (Output Enable YUV data, subdirección 11H, bit D3) para permitir el control del triestado de VPO mediante RTS1.

Estos dos pines tendrán un papel muy destacado en nuestro sistema, pues son capaces de proporcionar información sobre el sincronismo de la imagen. De hecho, programaremos el procesador de vídeo de forma que a través de estos pines obtengamos:

- por RTS0: los pulsos de sincronización horizontal (HREF), cuyos flancos de subida indican el inicio de una nueva línea de vídeo activo.
- Por RTS1: los pulsos de sincronismo vertical (VREF), cuyos flancos de subida indican el inicio de un nuevo campo.

De esta manera, sincronizaremos nuestro circuito con los datos de imagen proporcionados por la cámara sin tener que identificar ninguna secuencia y , por tanto, sin tener que sincronizarnos con LLC.

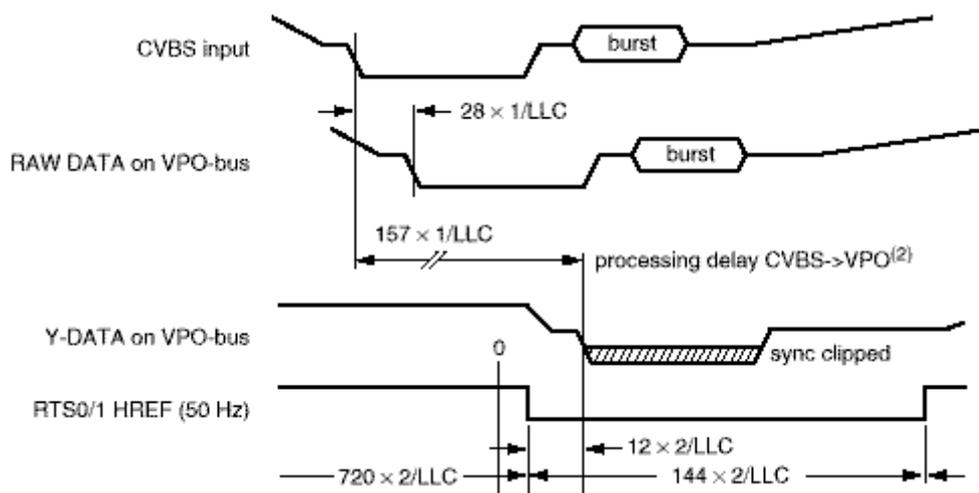


#### 4.3.1.2.3.1. Temporización horizontal

Es la que corresponde al blanking y exploración de una línea de vídeo.

Los registros internos de control del procesador de vídeo asociados a las salidas RTS0 y RTS1 pueden programarse de forma que cualquiera de tales salidas genere una señal **HREF** que indica la presencia de datos válidos en el bus VPO. El flanco positivo de la señal HREF marca el comienzo de una nueva línea de vídeo activo. La señal permanece a nivel bajo durante el periodo de blanking horizontal. La anchura de este pulso negativo durante el periodo de blanking depende del tipo de datos seleccionado por los registros de control LCR2 a LCR24. En el caso concreto de datos **RAW**, el tiempo que permanece en alto la señal HREF, puede programarse en múltiplos de  $LLC/8$ , siendo  $LLC$  la frecuencia de salida de datos a través del bus VPO y cuyo valor nominal es 27 MHz.. Para ello se utilizan los registros HSB (Horizontal Sinc Begin) y HSS (Horizontal Sinc Stop). Además, se puede realizar un ajuste fino de la posición del pulso HREF a través del uso de los bits HDEL1 y HDEL0 situados en el registro de subdirección 10H.

Todo esto puede verse gráficamente en la figura siguiente:



*Figura 4.23: Temporización del sincronismo horizontal (línea)*



Al comienzo de dicha figura, aparece una ampliación del periodo de blanking horizontal. También se muestra el retardo que sufre la señal CVBS desde la entrada analógica hasta que comienza a aparecer en el bus VPO. Dicho retardo se debe a toda la etapa de procesamiento que la señal debe atravesar.

También pueden programarse las salidas RTS0 y RTS1 para generar una señal denominada PLIN que indica la presencia de una secuencia PAL/SECAM.

Si recordamos, la duración de una línea de vídeo en el estándar de 50 Hz y 625 líneas, como es el caso de la señal de vídeo generada por nuestras cámaras, era de  $64\mu\text{s}$ . Dentro de ese intervalo podíamos distinguir dos partes diferentes: un primer periodo donde la señal de vídeo posee un pulso negativo por debajo del nivel de negro, esto es, el periodo de blanking horizontal, y un segundo periodo donde se encuentra la información correspondiente a la línea de vídeo que está siendo explorada, esto es, el periodo de exploración de línea. En el caso de usar el formato de datos YUV, se generan 720 píxeles activos por cada línea de imagen. Los datos de dichos píxeles se codifican en 1.440 bytes, siendo dicho número de bytes, un número fijo que no puede ser programado. Según se aprecia en la figura, el tiempo que permanece la señal HREF a nivel bajo es  $144 \times 2/LLC$ , es decir,  $10.66\mu\text{s}$  que corresponden a al blanking horizontal de la señal en formato YUV. Los otros  $53.33\mu\text{s}$  corresponden a la zona de vídeo activo, y es en esos  $53.33\mu\text{s}$  cuando se generan los 1.440 bytes. Si dividimos los  $53.33\mu\text{s}$  por los 1.440 bytes, obtendremos el tiempo que permanece un byte en el puerto VPO, dicho tiempo es  $37.037\text{ns}$ , y si invertimos dicho tiempo obtendremos la frecuencia a la que aparecen los datos en el bus VPO, es decir 27 MHz, que coincide, como cabía esperar con LLC.

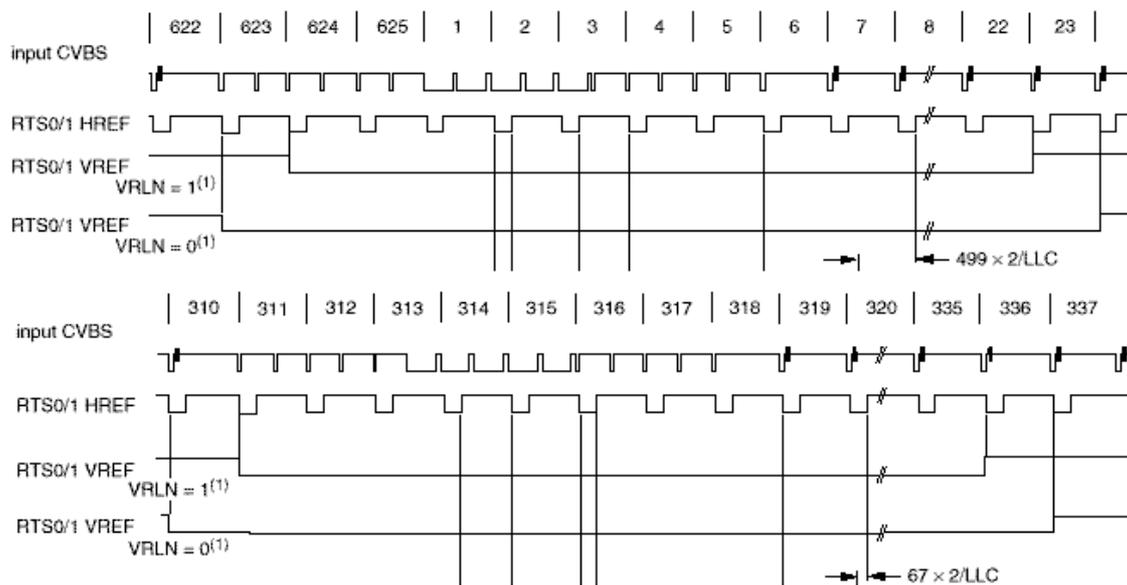
Sin embargo, en el caso de muestras RAW, el número de bytes por línea de vídeo activo es programable, y esto se consigue programando la duración del pulso HREF. De esta forma podremos conseguir un número variable de muestras por cada línea de vídeo, en función de nuestras posibilidades.



#### 4.3.1.2.3.2. Temporización vertical

En la página siguiente se muestra el conjunto de señales asociadas a la temporización vertical de una señal de vídeo con sincronismo vertical de 50 Hz:

Como puede verse en dicha figura, la primera de las señales mostradas corresponde al periodo de blanking vertical de una señal de vídeo compuesto similar a la proporcionada por los módulos de cámara sub-miniatura que vamos a utilizar. A continuación, aparece la señal HREF explicada más arriba y, posteriormente, aparece una señal llamada **VREF** cuya naturaleza y función son similares a las de HREF pero aplicado al caso de la sincronización vertical. Durante el periodo de blanking vertical, la señal VREF permanece a nivel bajo indicando con su flanco de subida, el comienzo de un nuevo campo de la imagen de vídeo. Es posible elegir entre dos duraciones diferentes para el pulso negativo de VREF programando el bit VRLN del registro de subdirección 10H. Esta señal VREF se puede obtener en los pines RTS0 y RTS1 programando adecuadamente los bits RTSE de los registros de control.



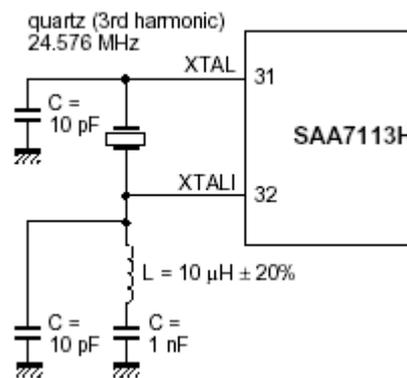
*Figura 4.24: Temporización del sincronismo vertical (campo)*



### 4.3.1.3. OTRAS CONEXIONES

#### 4.3.1.3.1. Reloj de entrada (XTALI, XTAL)

Para el correcto funcionamiento del decodificador de vídeo hay que proporcionarle una señal de reloj de **24.576MHz** como entrada a través del pin nº 32: **XTALI**. Dicha señal puede provenir de un oscilador de cristal o de un reloj externo. En el primer caso, el circuito es el siguiente:

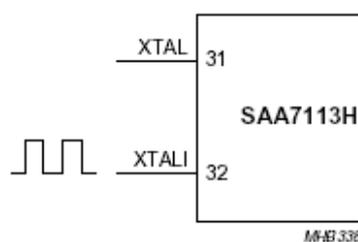


*Figura 4.25: Oscilador de cristal*

donde:

- XTALI: terminal de entrada del oscilador de cuarzo.
- XTAL: segundo terminal del oscilador de cuarzo.

Si se emplea una señal de reloj externa, el circuito es el siguiente:



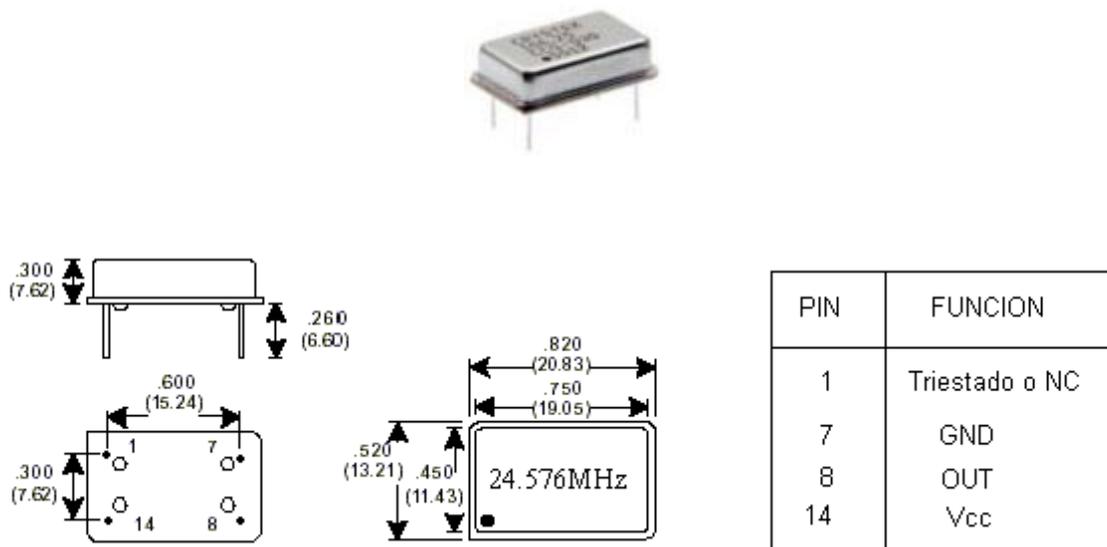
*Figura 4.26: Reloj externo*

Donde:

- XTALI: terminal de entrada del reloj externo.
- XTAL: no se conecta.

En nuestro proyecto se utilizan tres osciladores de cristal de **24.576MHz** de frecuencia nominal<sup>15</sup>, uno para cada decodificador de vídeo. Las principales características de los osciladores empleados son las siguientes:

- *Frecuencia:* 24.576 MHz.
- *Tensión de entrada:*  $3.3V \pm 0.3V$ .
- *Corriente de entrada:* 40mA.
- *Temperatura de funcionamiento:* (Std.  $0^{\circ}C-70^{\circ}C$ ) ó  $-40^{\circ}C-85^{\circ}C$ .
- *Dimensiones*<sup>16</sup>:



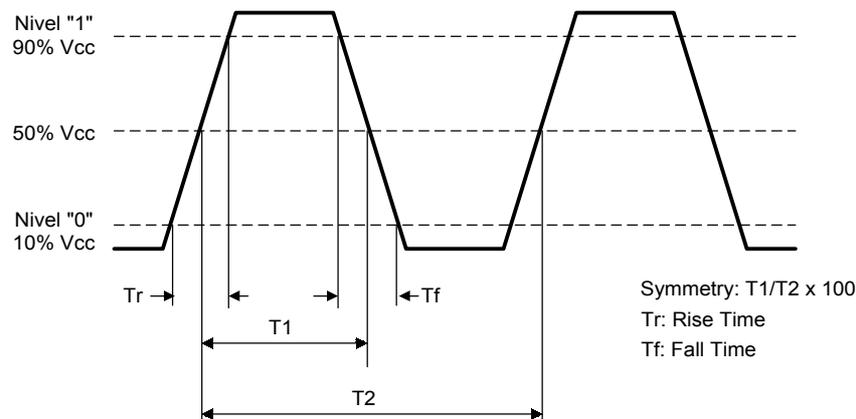
*Figura 4.27: Dimensiones*

<sup>15</sup> Hay que señalar que a pesar de ser ésta una frecuencia muy común en circuitos de vídeo, resultó bastante complicado adquirir estos componentes.

<sup>16</sup> Las dimensiones que aparecen en la figura presentan el siguiente formato: pulgadas(mm)



- *Forma de onda a la salida:* HCMOS  $T_r, T_f = 6\text{ns}$ .



**Figura 4.28:** Forma de onda a la salida

- *Tensión de salida a nivel alto:*  $V_{oh} = 90\%V_{cc}$  (como mínimo).
- *Tensión de salida a nivel bajo:*  $V_{ol} = 10\%V_{cc}$  (como máximo).

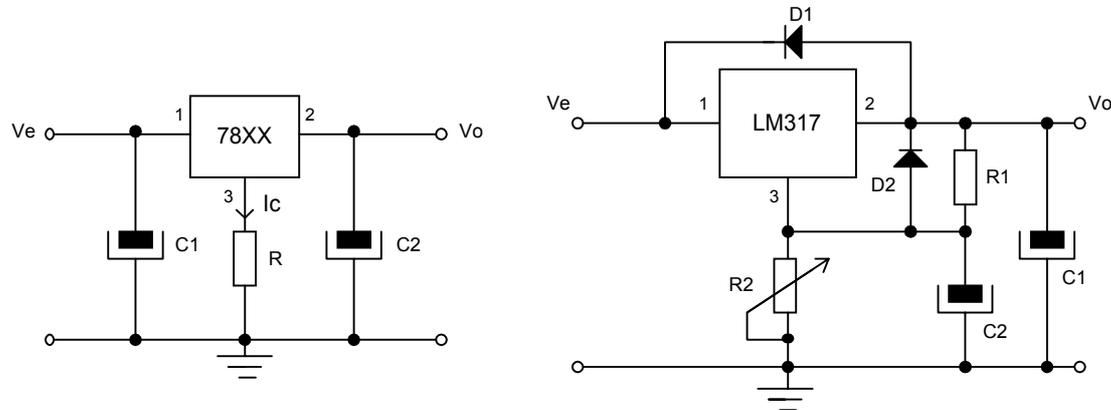
### 4.3.2. ALIMENTACIÓN DEL CIRCUITO

La placa de circuito impreso diseñada en el proyecto se alimentará a 12V continuos, al igual que las cámaras de vídeo. Para obtener a partir de esta fuente de alimentación las tensiones necesarias para el circuito de adquisición se emplearán dos reguladores de tensión, uno *fijo* (7805) y otro *variable* (LM713). Los 5V que se obtienen con el 7805 sirven para alimentar el circuito integrado MAX232A, que hace de interfaz con el puerto serie. Por otro lado, los decodificadores de vídeo precisan de 3.3V para su correcto funcionamiento.

Los dos circuitos que se ven en la *figura 4.29* llevan a la entrada dos condensadores en serie para filtrar la frecuencia de 50Hz, condensador pequeño, y el otro para filtrar rizados en la señal, si este rizado no existe como ocurre cuando la alimentación es continua puede disminuirse o incluso eliminarse, aunque dejarlo tan sólo repercutirá en costes.



También el hecho de tener en serie los dos reguladores de tensión podría hacernos pensar que no es necesario poner condensadores de desacoplo a la salida del 7805 y a la entrada del LM713. Sin embargo, dado que la pista es bastante larga, por exigencias mecánicas, conviene mantener ambos.



*Figura 4.29: Esquemas de los reguladores de tensión*

Para el esquema de la izquierda se toma  $R = 0$ ,  $C1$  y  $C2$  del orden de unos  $100\mu\text{F}$  y en paralelo a  $C1$  colocamos un condensador de  $100\text{nF}$  para rechazar el rizado de  $50\text{Hz}$  de la alimentación, en realidad no se requieren condensadores tan elevados, ni tampoco un condensador para rechazar los  $50\text{Hz}$  si pretendemos alimentar al circuito con una fuente continua, pero tampoco están de más. Para el esquema de la derecha también se considera un condensador de desacoplo a la entrada de  $100\mu\text{F}$ , aunque nuevamente, y aquí con mayor motivo, dado que se alimenta a partir del regulador de  $5\text{V}$ , no es necesario o al menos no se requiere un condensador muy grande.  $R1$  se tomó igual a  $220\Omega$ ,  $R2$  se sustituyó por un potenciómetro de valor:  $330\Omega$ ,  $C2 = 10\mu\text{F}$ , y  $C1 = 100\mu\text{F}$  con un condensador en paralelo de  $100\text{nF}$ , al igual que a la entrada.

### 4.3.3. PUERTO SERIE: MAX232

A la hora de establecer una comunicación entre nuestro diseño y un PC a través del puerto serie es necesario llevar a cabo una conversión de niveles de tensión, para



hacer compatibles las tensiones utilizadas en TTL/CMOS con las empleadas en el protocolo RS-232. De esto se encarga el circuito integrado MAX232A de MAXIM.

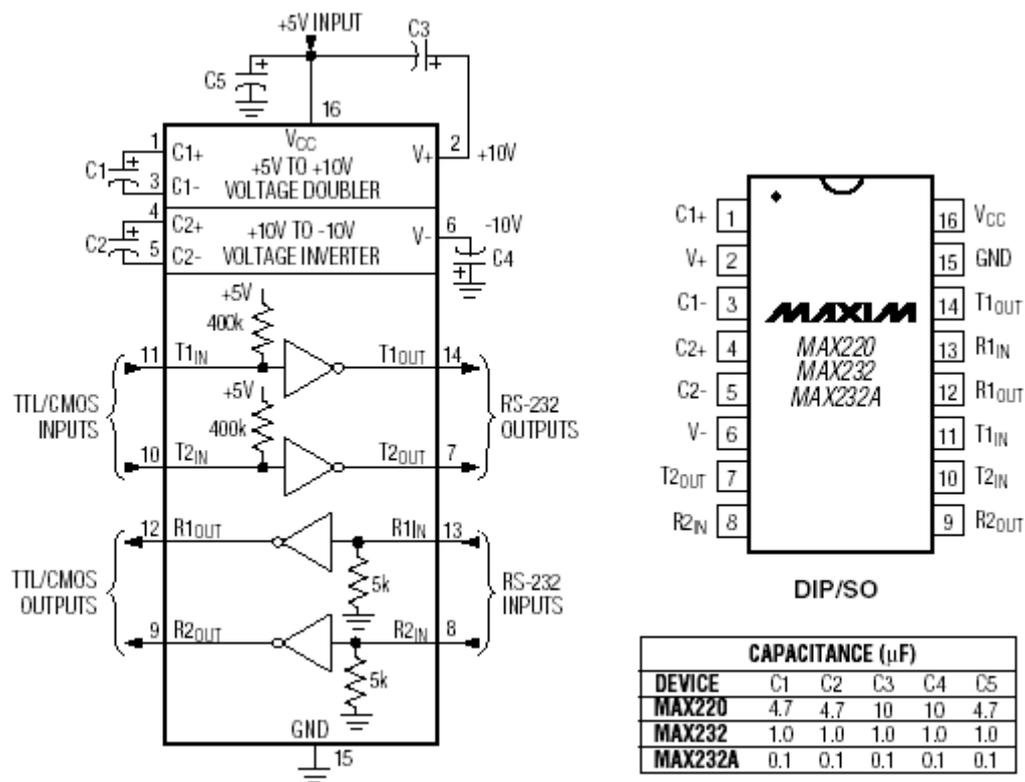


Figure 4.30: MAX220/MAX232/MAX232A Configuración de pines y circuito típico de operación

	Alimentación (V)	Nºdrivers/RX RS-232	NºCaps. Ext.	ValorCaps. ( $\mu\text{F}$ )	Velocidad (Kbps)
MAX232A	+5	2/2	4	0.1	200
MAX232	+5	2/2	4	1.0	120(64)

Tabla 4.2: Comparación entre MAX232A y MAX232

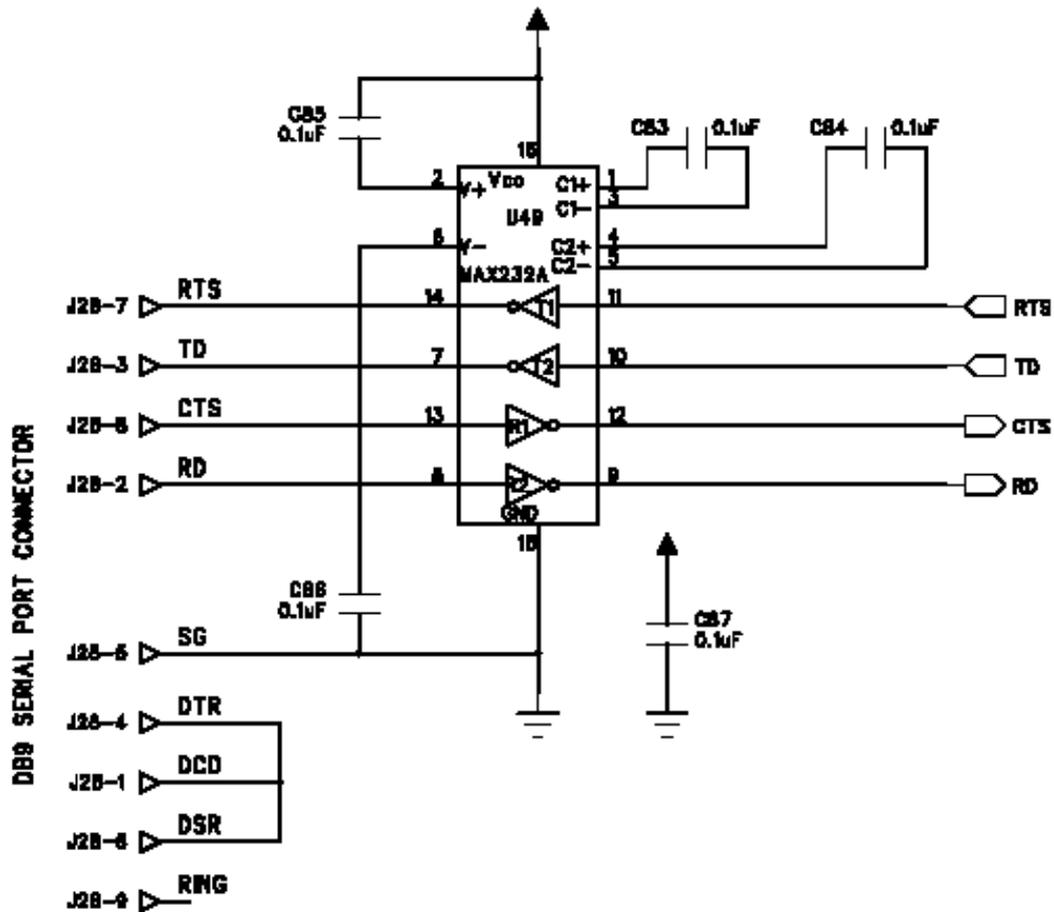


Figura 4.31: Esquemático empleado en el diseño del PCB

Aunque en la tarjeta XSV de XESS Corporation aparece el circuito integrado MAX232A, en nuestro diseño se ha empleado el circuito **MAX232**, más económico y capaz de cumplir con los requisitos del sistema de adquisición.

La principal diferencia entre ambos integrados está en la *velocidad de transferencia*: 200 Kbps del MAX232A frente a los 120 Kbps del MAX232. En nuestro diseño, la velocidad del puerto serie no es determinante ya que es el puerto paralelo el que, debido a su mayor velocidad, se empleará para transferir los datos al PC. Por otra parte, los valores de los condensadores de desacoplo utilizados serán de  $1.0\mu\text{F}$  (MAX232), en vez de  $0.1\mu\text{F}$  (MAX232A).



#### 4.3.4. CONVERTIDOR D/A: GENERADOR DE IMÁGENES VGA DE FORMA DIRECTA

Para enviar la información sobre el color de un punto a dibujar en la pantalla de un monitor VGA existen tres señales: rojo (*Red*), verde (*Green*) y azul (*Blue*). Cada una de estas 3 señales maneja un cañón de electrones que dibuja a uno de estos colores primarios sobre la pantalla del monitor. Niveles analógicos situados entre 0 (negro) y 0.7 V (máximo brillo) en cada una de estas tres líneas, informan al monitor sobre la intensidad de cada uno de estos tres colores primarios para definir el color del píxel que se dibuja sobre la pantalla.

En nuestro caso, la FPGA será la encargada de suministrar los niveles de color que deberán tener cada uno de los píxeles dibujados en la pantalla del monitor. Ahora bien, la FPGA sólo suministrará información digital, hemos de tener algún dispositivo que actúe de interfaz de conversión entre los datos digitales y los niveles analógicos de color que deben suministrarse al monitor. Para ello, en la placa del sistema de adquisición de datos se dispone, para cada una de las tres líneas de color, de un convertidor D/A formado por un par de resistencias (estructura R-2R).

Cada una de las resistencias del convertidor es atacada por una línea digital de salida de la FPGA, de forma que, se tienen dos bits por cada una de las tres líneas de color R, G y B. En total, seis bits controlarán el color de cada uno de los píxeles enviados hacia el monitor, y los cuatro posibles niveles a la entrada de cada convertidor D/A serán combinados por el monitor para crear un píxel con un color de entre  $4 \times 4 \times 4 = 64$  posibles colores diferentes. Es decir, los 6 bits definen un color de una paleta de 64 posibles colores.

Además existirán dos señales procedentes de la FPGA que se utilizarán para indicar a los circuitos de deflexión del monitor los instantes en que se ha de comenzar o detener dicha deflexión, dibujando cada línea de lado a lado de la pantalla, y formando cada *frame* a partir de la generación de líneas desde arriba hacia abajo. Serán señales de sincronismo horizontal *hsync* y vertical *vsync*.



A continuación se analiza cómo han de generarse las señales que controlan al monitor.

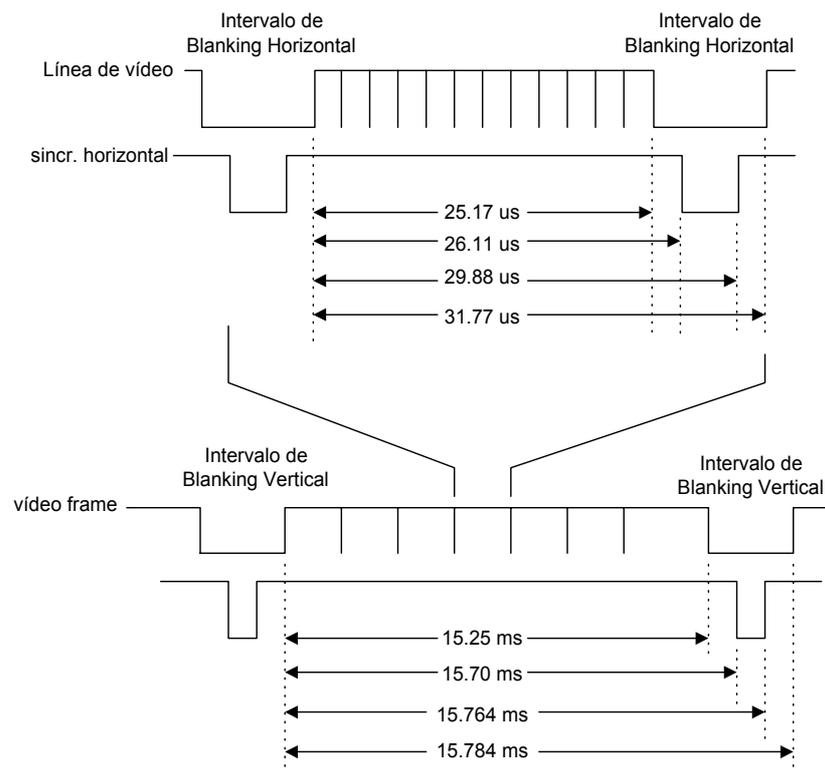
#### 4.3.4.1. TEMPORIZACIÓN DE LAS SEÑALES VGA

La información que aportan las 3 señales anteriores sobre el color de un punto a dibujar en la pantalla del monitor no es suficiente. Será necesario disponer de algún tipo de información más acerca de los píxeles que componen una línea completa en la pantalla, y del número de líneas que compondrán la imagen completa.

Un *frame* de una imagen VGA está compuesta de 480 líneas cada una de las cuales está integrada por 640 píxeles. Para dibujar un *frame* existen una serie de circuitos de deflexión en el monitor que son capaces de desviar el flujo de electrones procedentes de los cañones tanto de lado a lado como de abajo a arriba en la pantalla. Estos circuitos de deflexión requieren de una serie de señales de sincronización para indicar los instantes en que se ha de comenzar o detener la deflexión, dibujando cada línea de lado a lado de la pantalla, y formando cada *frame* a partir de la generación de líneas desde arriba hacia abajo. En la figura siguiente se muestra la temporización que debe ser respetada en la generación de las señales de sincronización para estos circuitos de deflexión.

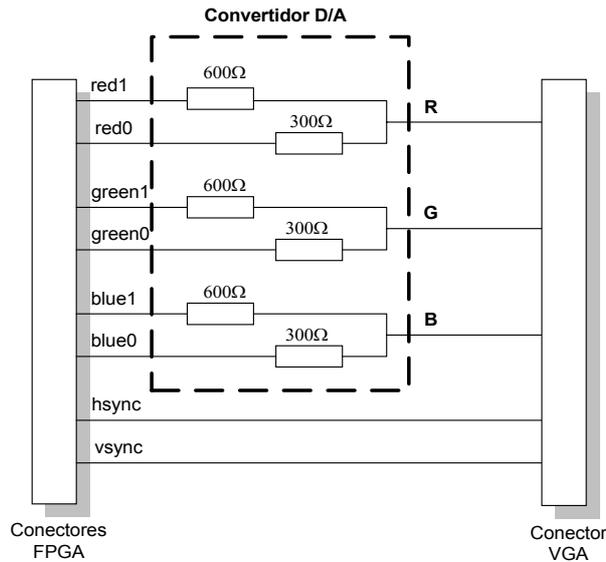
Los pulsos negativos en la señal de sincronismo horizontal marcan el comienzo y final de una línea y aseguran que el monitor dibuje los píxeles entre los flancos izquierdo y derecho de la zona visible de la pantalla. Los píxeles son enviados realmente dentro de una ventana de 25.17  $\mu$ s de duración. La señal de sincronismo horizontal cae a nivel bajo un mínimo de 0.94  $\mu$ s después del último píxel y permanece en dicho estado durante 3.77  $\mu$ s más. La siguiente línea no puede empezar a generarse hasta 1.89  $\mu$ s después de la generación del pulso en la señal de sincronismo horizontal. Así pues, una sola línea dura 25.17  $\mu$ s dentro de un intervalo de 31.77  $\mu$ s. Los 6.6  $\mu$ s restantes de cada línea son para el *blanking horizontal* es decir definen la zona de la pantalla que permanecerá en negro.





**Figura 4.32:** Temporización VGA

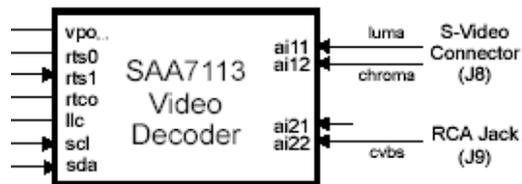
De forma análoga, los pulsos negativos en la señal de sincronismo vertical marcan el comienzo y el final de un *frame* formado a partir de un conjunto de líneas y aseguran que el monitor dibuje las líneas entre los flancos superior e inferior de la zona visible de la pantalla del monitor. Las líneas que componen el *frame* son enviadas al monitor dentro de una ventana de duración 15.25 ms. La señal de sincronismo vertical cae a nivel bajo un mínimo de 0.45 ms después de la última línea, y permanece a dicho nivel durante 64  $\mu\text{s}$ . La primera línea del siguiente *frame* no puede comenzar a generarse hasta transcurridos al menos 1.02 ms después de que el pulso de sincronismo vertical finalice. Así pues, un único *frame* ocupa 15.25 ms dentro de un intervalo de 16.784 ms. Los otros 1.534 ms restantes se usan para el *blanking vertical*, durante el cual, la pantalla permanece en negro.



**Figura 4.33:** Esquema CD/A de estructura R-2R

### 4.3.5. CONECTORES DE VÍDEO

Cada decodificador de vídeo puede recibir la señal de vídeo a través de un conector de vídeo *RCA jack* o a través de un *S-video conector*.



**Figura 4.34:** Entradas de video al decodificador

En nuestro proyecto, se conectan las cámaras a los RCA jack, ya que se trabaja con señales CVBS.

#### RCA Jack:

Conector RF coaxial recto (*straight round*) cuya foto y dimensiones se muestran a continuación:





