

## MPEG/H.261 VIDEO DECODER

### PRELIMINARY DATA

- Real-time decompression of MPEG video and CCITT H.261 bitstream at up to 10 Mbits/sec
- Supports real time decoding of pictures of 352 x 288 pixels at 30 fps
- Programmable picture format
- Asynchronous 16 bit YUV/YV (4:2:2 format) pixel port for video output
- Selectable on-chip YUV to RGB convertor
- Video output compatible with PAL and NTSC format display
- Direct addressing of 512 Kbytes up to 4 Mbytes DRAM memory with on-chip memory controller
- Standard 8/16 bit microprocessor interface
- Dedicated serial port for compressed data
- Video synchronisation from external timing generator
- On-chip start code detector
- PQFP144 package

### DESCRIPTION

The STi3240 supports the video decoding schemes of the ISO/MPEG future video standard and CCITT/H.261 recommendation at video rate. The decoder consists of a chip set providing an optimised implementation of a decoderOnly function. The chip set can decode MPEG and H.261 video bitstream in full using a microprocessor.

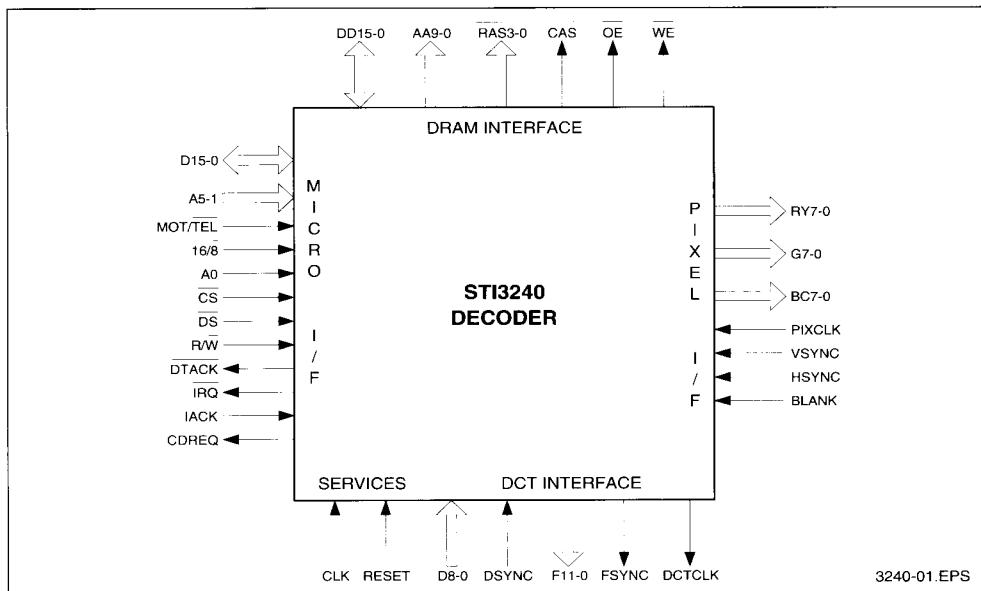
The architecture, which is built up around hardwired operators, has been selected for the ease of further integration of the DCT onto the chip. The hardwired architecture also offers high speed operation as well as an optimised silicon implementation which gives an optimised and low cost decoding scheme. Flexibility is provided through a set of registers available to the user to modify, for example, display parameters.

All temporary storage required by the algorithm such as the blocker buffer, frame buffer and bit buffer is merged into a single low cost local memory realised with conventional Drams. Up to 32 Mbits can be directly addressed, memory control including refresh being integrated onPchip. A minimum configuration consists of 4 Mbits of memory.

The devices require minimal support from an external microprocessor. The microprocessor has to initialise the STi3240 and must process the headers for higher layers of the standard (picture, group of pictures, sequence, etc.). For this purpose a start code detector is provided on-chip.



## PIN CONNECTIONS



## PIN LIST

## DRAM INTERFACE

DD[15:0]	I/O	16 bit bidirectional data port
AA[9:0]	O	10 bit address bus
notRAS[3:0]	O	row address strobes (one per bank)
notCAS,notOE, notWE	O	control strobes (common to all banks)

## PIXEL PORT

RY[7:0]	O	8 bit R/luminance port
G[7:0]	O	8 bit G
BC[7:0]	O	8 bit B/chrominance port
PIXCLK	I	pixel clock
VSYNC	I	frame synchronisation input
HSYNC	I	line synchronisation input
BORDER	I	valid pixel input

## MICROPROCESSOR INTERFACE

D[15:0]	I/O	16 bit bidirectional data bus
A[5:1]	I	8 bit address bus
MOT/notTEL	I	microprocessor mode selection
16/not8	I	data format selection
A[0]	I	address LSB for 8 bit mode

## MICROPROCESSOR INTERFACE

notCS	I	chip select
notDS	I	data strobe
R/notW	I	read/write selection
notDTACK	O	data acknowledge
notIRQ	O	interrupt request
notIACK	I	interrupt acknowledge
notCDREQ	O	compressed data request

## DCT INTERFACE

D[8:0]	I	9 bit pixel bus
DSYNC	I	pixel synchronisation signal
F[11:0]	O	12 bit coefficient bus
FSYNC	O	coefficient synchronisation signal
DCTCLK	O	DCT clock

## SERVICES

CLK	I	primary clock
RESET	I	reset

## I. OVERVIEW

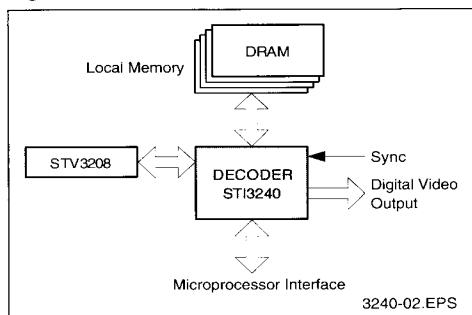
The STi3240 is designed to address a wide range of applications dealing with MPEG and H.261. The different operating modes available are:

- MPEG - variable picture format
- H.261 CIF
- H.261 QCIF

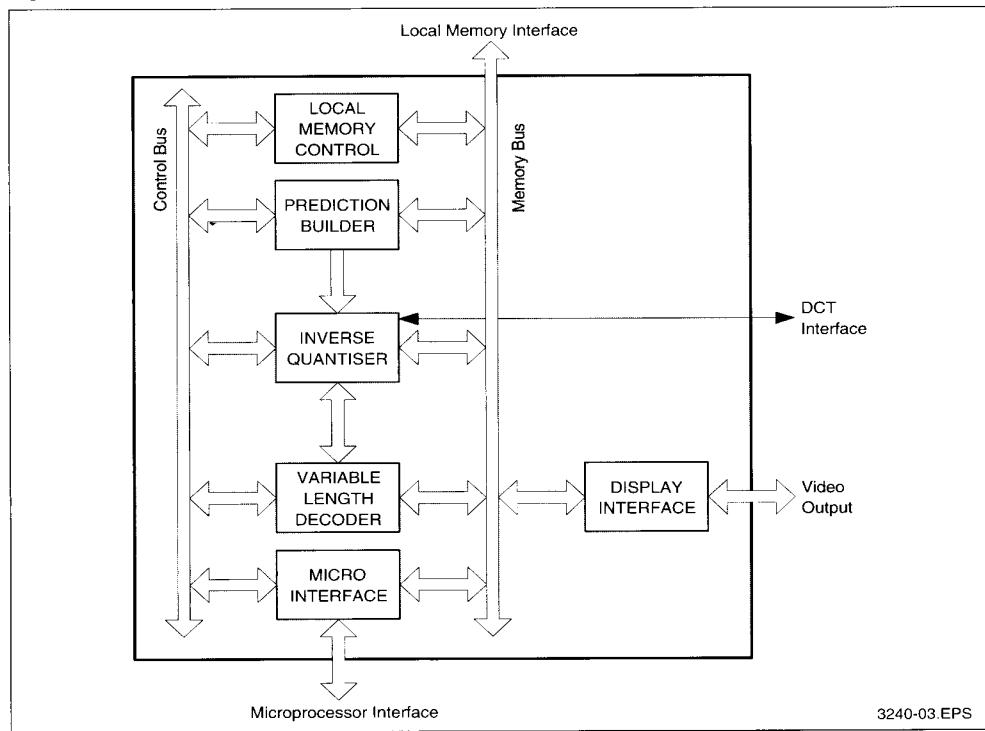
The decoder system (Figure 1) is made of two chips: a DCT processor, the STV3208 and a decoder chip, the STi3240. The STV3208 currently manufactured by SGS-THOMSON, is a 8 x 8 forward and inverse Discrete Cosine Transform processor (DCT) with on chip zig-zag scan of coefficients (see STV3208 data sheet). The STi3240 integrates all the remaining functions necessary to implement a MPEG or H.261 video de-

coder ( see section 4 ) as well as a local memory controller, and a display controller (Figure 2).

**Figure 1: Decoder System**



**Figure 2 : STi3240 Block Diagram**



## II. CPU INTERFACE

The CPU interface is used for accessing the internal registers of the STi3240. It can also be used for

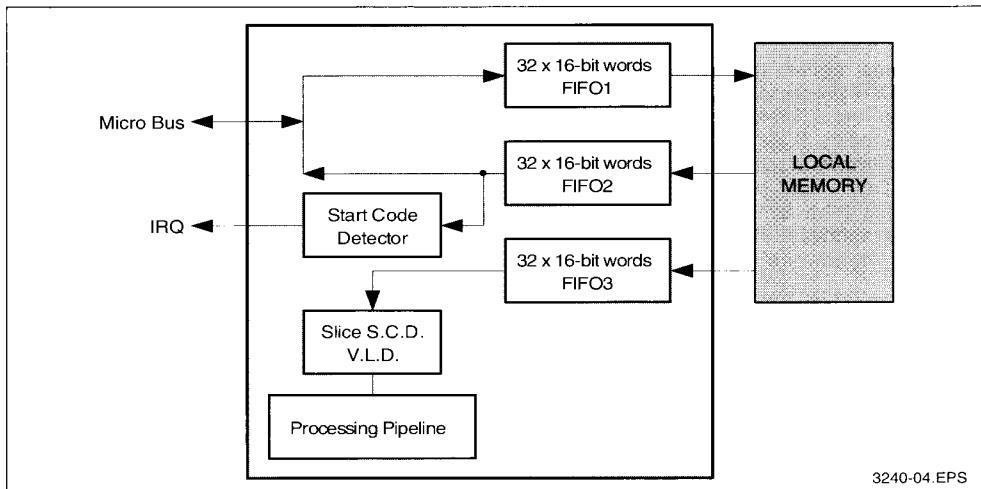
inputting the compressed data stream. It can be configured to be either the Intel or Motorola style of interface and either 8 or 16 bits wide.

### III. COMPRESSED DATA INTERFACE

The compressed data interface is organised around three FIFOs (16-bit width, 32 words depth) see Figure 3. Each FIFO is used to adapt the rate

between the local memory space used for bit buffer storage and three specific functions: respectively the bit stream input, the Start Code Detection and the Variable Length Decoder.

Figure 3 : Compressed Data Interface



3240-04.EPS

#### III.1. BIT STREAM INPUT

The compressed bit stream is loaded from the microprocessor interface.

##### III.1.1. CPU port

In this case the bit stream is loaded in words of 8 or 16 bits depending on the selected micro port format. The consecutive compressed data bits are considered to be stored from the MSB to the LSB of the incoming words. In 16-bit format the FIFO address may be 00 or 01 (A0 value is "don't care"). In the case where the 8-bit micro port format is selected, all the bit stream writes are made at address 01 (A0 = 1). The first byte to be written is the most significant byte and is stored in an internal temporary 8-bit register. The following byte is the least significant byte; when written, both MSbyte and LSbyte are transferred as a single 16-bit word into FIFO 1.

Note: The CDREQ pin could be used in the microprocessor mode to control a DMA transfer. In this case the pin value is updated on each new word write.

##### III.1.2. Bit Buffer

The words that have been written into FIFO1 from

one or the other port are transferred into the external local memory in an area reserved for the bit stream buffer. When a word is extracted from the FIFO for transfer to the memory, the SRREQ pin is set indicating that a new word could be written in the FIFO.

The size of the bit buffer content is accessible to the microprocessor as a multiple of 128 bits in the bit buffer status register. The bit buffer always starts at address 00000 of the local memory space. Its maximum size is specified in a register.

Two interrupts may be generated depending on the bit buffer content: one bit buffer nearly full and one bit buffer nearly empty. The threshold above which the nearly full interrupt should be activated is specified in a register. The nearly empty interrupt is generated when the bit buffer contains less than 256 bits.

#### III.2. START CODE DETECTION

##### III.2.1 Start Code Detector

Start code detection is synchronised with an internal DSYNC signal. This signal is a decoder synchronisation signal derived from the Vertical Sync signal (VSYNC) of the display interface (see sec-

tion V). If the frames are displayed only one time, DSYNC is activated on each VSYNC. If the frames are displayed N times (N ranging from 1 to 7) then DSYNC is activated on one VSYNC over N.

On DSYNC occurrence, the bit buffer content is automatically scanned: the bits are copied in 16-bit word format from the bit buffer to FIFO2 which output is connected to a Start Code Detector (SCD): all the compressed data coding the images are skipped until a start code is detected. When a start code i.e. the value 00 00 00 01 XX in hexadecimal is found, the SCD generates an interrupt to the microprocessor and the automatic bit buffer scanning is stopped until the next DSYNC. Interrupts are not generated on slice or GOB start codes as they are handled by the STi3240.

### III.2.2. Headers Handling

When the micro receives the start code detection interrupt, it can have access to the value XX of the start code into the header code register.

The micro must then read the bit stream (into FIFO 2) and analyse all the header information until it reaches a slice or GOB header. As the start code is not word aligned, the position of the value XX into the 16-bit word FIFO (header data FIFO at address 01) is also accessible. The header information will be used by the micro to control the system and programme the decoder registers (see section VII). When the micro has programmed all the chip registers and considers that the bit buffer is full enough to start decoding, it will enable the decoder by setting the enable bit in decoder control register. The real decoding will start on the next DSYNC. In a continuous flow the headers' detection will always

be one picture in advance with regard to the decoding process. This means that the micro has one frame time to analyse the headers and take the right control decisions.

### III.2.3. Frame Skipping

When the micro reads the header of the current picture detected after a DSYNC it may decide to skip the decoding of that frame. In this case it needs to access to the header's information of the next frame. This is made possible by restarting the Start Code Detection until the next header is reached. The SCD restart is initiated by a write into the header restart register.

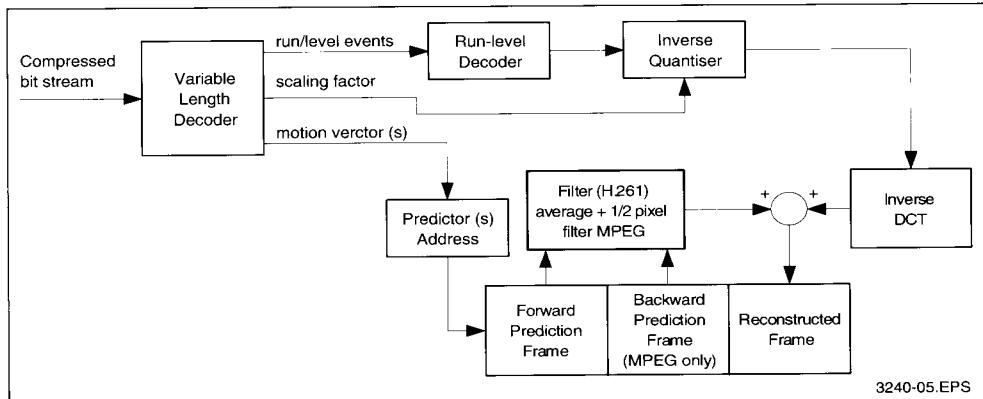
### III.3. VARIABLE LENGTH DECODER

If the decoder has been enabled by the microprocessor, on the next DSYNC occurrence it will start to extract the bits from the bit buffer, transfer them into FIFO 3 and decode them. In order to bypass all the headers' bits that have already been exploited by the micro, a second Start Code Detector is implemented on FIFO 3 output: it bypasses all the bits until a Slice (MPEG) or GOB (H261) start code is found. From that point all the picture bits that have been Huffman encoded are sent to the Variable Length Decoder and to the rest of the decoding pipeline described in the following chapter.

## IV. PIPELINE

The pipeline includes the major functional blocks to implement the MPEG and H.261 decoding schemes (see Figure 4).

Figure 4 : Internal Pipeline



3240-05.EPS

## V. VIDEO INTERFACE

The output format is either YUV 4:2:2 or RGB. The 4:2:2 format is obtained from the 4:1:1 decoded format by a simple duplication of the chroma lines. The YUV to RGB transform is obtained through the CCIR 601 recommended formulae:

$$R = Y + 1.370 \times (Cr - 128)$$

$$G = Y - 0.698 \times (Cr - 128) - 0.336 \times (Cb - 128)$$

$$B = Y + 1.730 \times (Cb - 128)$$

CIF to CCIR 601 format conversion is supported through duplication of frames and pixels. Because the decoded frames generally have a lower resolution than the displayed frame, it is possible to display a frame several times during the time that another one is being decoded. If it is required to do some post processing on the decompressed pixels this will introduce a delay on the pixels. In order to re-synchronise the pixels with the 3 sync signals, the STi3240 provides horizontal offset registers.

Figure 5 shows the display model of the STi3240. The display model is quite simple: offset registers allow to display the decoded pictures anywhere on the screen. Offsets are relative to the active edge of the sync signals. Outside the display region or whenever the blank is active, the border colour is output.

Frames can be displayed several times, pixels can

be displayed during one or two consecutive PIXCLK cycles. Lines cannot be duplicated. This simplified scheme still allows to interface easily with two types of display: computer-type progressive displays and TV-type interlaced displays. A video window can be displayed anywhere on a computer display by setting appropriately the offset registers and can be inserted amid other windows by using colour keying and the BLANK signal. Interlacing is obtained by displaying the same picture twice as the odd and even field and by duplicating pixels horizontally. In the process the image is scaled by a factor 2.

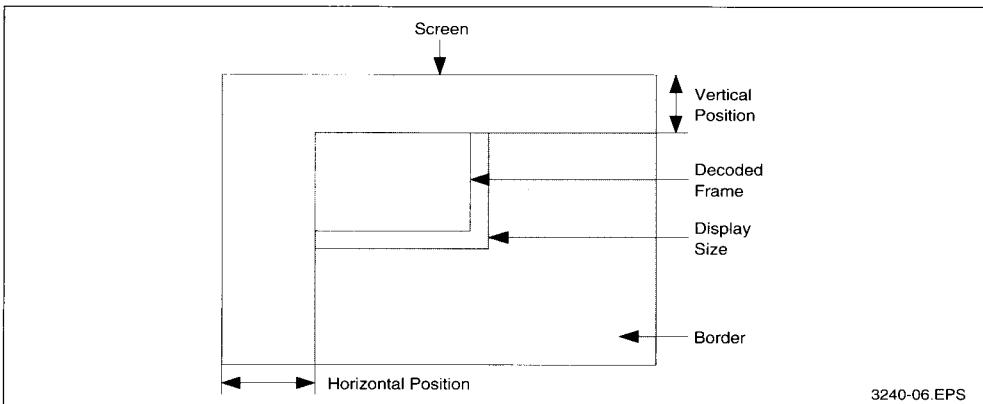
The actual bandwidth available for display at the memory interface is about 10 Mpixels/s. The bandwidth available at the pixel interface is the same except in pixel duplication mode, where it is twice this value. So pixel rates higher than 10MHz (ie CCIR 601) are realised only in this mode.

The display interface of the STi3240 is composed of:

- three 8-bit output busses (RGB or YUV)
- an input pixel clock (PIXCLK),
- two input syncs (VSYNC and HSYNC),
- one input blanking signal (BLANK).

The maximum operating frequency of the pixel clock is 20MHz.

**Figure 5 : Display Model**



## V.1. REGISTER DESCRIPTION

There are several registers controlling the display, but the pixel interface hosts only 6 of them. The display offset registers hold the X and Y offsets in terms of the number of PIXCLK cycles and display lines respectively. The offset is counted from the active edge of the related blank signal. These registers have to be set before operation. The horizontal offset cannot be less than  $42 \times 3 = 126$  primary clock cycles, which is about 3.155s for a primary clock rate of 40MHz. The system must work out how many PIXCLK cycles are necessary to respect this constraint. Once the offset is reached the pixel interface will start delivering video data even if the HSYNC signal is still inactive.

The display size registers hold the X and Y displayed picture dimensions incremented by the respective offsets. The size of the displayed picture should never be larger than the size of the decoded picture.

The display control registers hold the border colour, the sync signal polarities and the pixel duplication bit. The border colour is output by the pixel interface whenever one of the following condition is met, as illustrated in Figure 6.

- (1) the BLANK signal is active
- (2) the horizontal or vertical display offsets are not reached
- (3) the current pixel position is outside the display window
- (4) the force border bit is set

In RGB mode, the contents of the display control registers is an RGB value and is displayed as such

ie their contents are not considered a YUV value and passed through the dematrixing unit. After reset, the value is 0.

The polarity of the sync signals (HSYNC, VSYNC, BLANK) are programmable.

When not duplicated, successive pixels will be output as:... (Y1,U1) (Y2,V1) (Y3,U3) (Y4,V3)...

If duplicated, the same pixels will be output as:... (Y1,U1) (Y1,V1) (Y2,U1) (Y2,V1) (Y3,U3) (Y3,V3)...

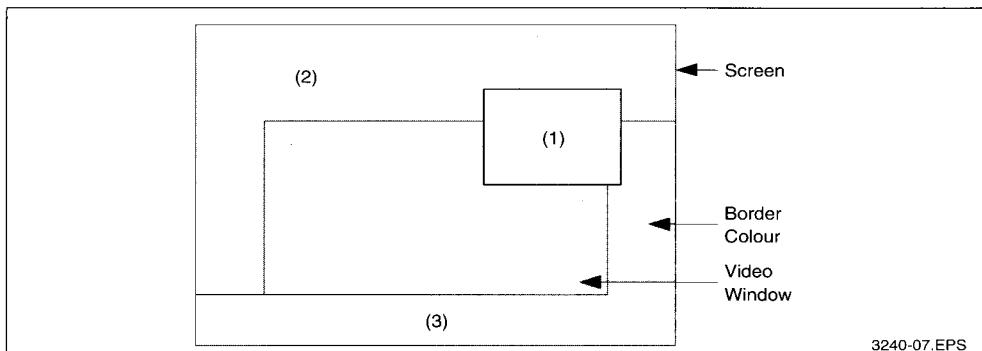
## VI. MEMORY INTERFACE

With a local clock frequency of 40Mhz, the DRAMs to be used are of type -12 with a minimum page mode cycle time of 75ns.

The local memory is organised as 16 bit words and up to 4 banks can be controlled. The minimum configuration is 512 Kbytes and the maximum is 4 Mbytes. The bank selection is done by the RAS signals. The memory size, depending on DRAM type, is:

DRAM type	Memory Size	Bank
256K x 4 or 256K x 16	256K x 16	1
	512K x 16	2
	768K x 16	3
	1M x 16	4
1M x 4	1M x 16	1
	2M x 16	2
	512K x 8	1
512K x 8	1M x 16	2
	1.5M x 16	3
	2M x 16	4

Figure 6: Window Positioning



## VII. THEORY OF OPERATION

Before decoding, a set of registers has to be initialised. These configuration registers are:

- set interrupt masks
- set memory configuration (8 or 9 address bits, number of banks, period of refresh cycles)
- set bit buffer size
- set display registers (display offset, size, colour of border)
- select MPEG, H.261 CIF or H.261 QCIF

The compressed bit stream is then loaded into the bit buffer starting from address \$000000. The transfer can be done either through the dedicated serial port or through the microprocessor interface. An internal FIFO adapts the rate on the compressed data port to the local memory bandwidth (refer to the performance section). The bit buffer fullness is always available to the microprocessor in multiples of 128 bits (see register list).

After each new VSync the STI3240 scans the bit buffer contents until a start code is detected (except slice start codes which will be handled by the processing pipeline). The detection of a start code is reported to the microprocessor by an interrupt. Note that the bit buffer is scanned but not emptied. The microprocessor must then read the bit buffer contents following the detected start code to analyse what kind of header occurred and to extract the relevant information from it. Extraction of the bit buffer information must be done by the microprocessor until it finishes analysing a picture header. The micro must programme the decoding parameters of the chip such as picture size and quantisation matrix (from sequence header), the picture type, the motion vector range and precision, the frame location to be used for backward prediction and for forward prediction, the frame location in which the reconstruction must be done, the frame location to display and the number of times it must be displayed. From that point on the STI3240 is able to work alone and to handle slice and macro block information. If enabled by the micro then the decoding process will effectively start on the next VSync occurrence.

The STI3240 extracts all the header or non video bits from the bit buffer until a slice start code is detected. The compressed picture bits are then processed in the variable length decoder, run level decoder, inverse quantizer, inverse DCT and post adder. The reconstructed blocks are written in the frame location that was specified by the microprocessor.

It is possible to reconstruct the compressed picture

in the same location as the displayed picture. An internal mechanism ensures that the decoding process will not start on a row of macroblocks until the display has finished displaying the row in order to avoid over-writing data. If the frame is to be displayed several times then the decoding process will only be enabled during the last display time.

### VII.1. MICROPROCESSOR TASKS FOR THE CONTROL OF THE STI3240

The bit stream is input in a 16-bit word format into the STI3240. The only microprocessor task in the main programme is to initialise the transfer of data from the storage media into the STI3240, for example through a DMA controller.

Upon detection of a start code, the STI3240 generates an interrupt to the microprocessor which must then analyse the incoming bit stream. A read of the STI3240 internal FIFO enables the microprocessor to find out what start code caused the interrupt. Depending on the start code value, the microprocessor will have to look at the following bits by reading the STI3240 FIFO and then programme the STI3240 or set system parameters:

- Sequence start code: the sequence header mainly allows the microprocessor to preset the system display and to initialise the chip with the correct picture size and quantisation tables. This is the most time consuming header to manage but is also the header that appears least often (one movie may contain only one sequence).
- Group Of Picture start code: the microprocessor verifies that the incoming group of pictures can be correctly decoded.
- Picture start code: the microprocessor must extract the temporal reference and type of the incoming picture and tell the STI3240 what type of picture is to be decoded, into which local RAM location it must be decoded, where the forward and backward prediction frames are located (if needed), the start location of the next frame to display and the number of times it must be displayed. The motion vectors' range is also extracted and set in the STI3240. After this the STI3240 is able to run alone until the next picture start code.

As an example, managing the video headers with a 68000 microprocessor running at 12.5MHz takes less than 5% of its total time. This is in the case where there is one new sequence per picture which is unrealistic. If a more reasonable case is considered where a sequence is composed of one group of pictures and a GOP is composed of 12 pictures,

then the average 68000 load is less than 1% of its time. The delay for computing the headers compared to the number of incoming bits with a bit stream rate of 1.856 Mbit/s represents roughly 500 bits i.e. less than the internal STi3240 FIFO of 512 bits. This example shows that the control of the STi3240 can be done by the central CPU of the global system without any need for an additional microprocessor and that the STi3240 control is not a high priority task.

## VII.2. TIMING EXAMPLE

Figure 7 shows an example of how the different operations involved from bit stream input to decoded frames display are linked. The example is an MPEG one, based around a Group Of Pictures structure of seven pictures organised in the order: I0 B2 P3 B4 B5 P6. The bidirectional frames may have predictors in their respective previous and next I or P frames. Therefore, such a GOP is encoded in the following order: I0 P3 B1 B2 P6 B4 B5 as B1, for instance, cannot be reconstructed until P3 is decoded.

As soon as the bit stream is read into the bit buffer, the Start Code Detector will operate and it will detect a header which will be reported to the micro by an interrupt. After this first interrupt, the micro has a lot of time to analyse the initial headers of the bit stream as the bit buffer is not yet full enough to enable the decoder (enabling the decoder just after

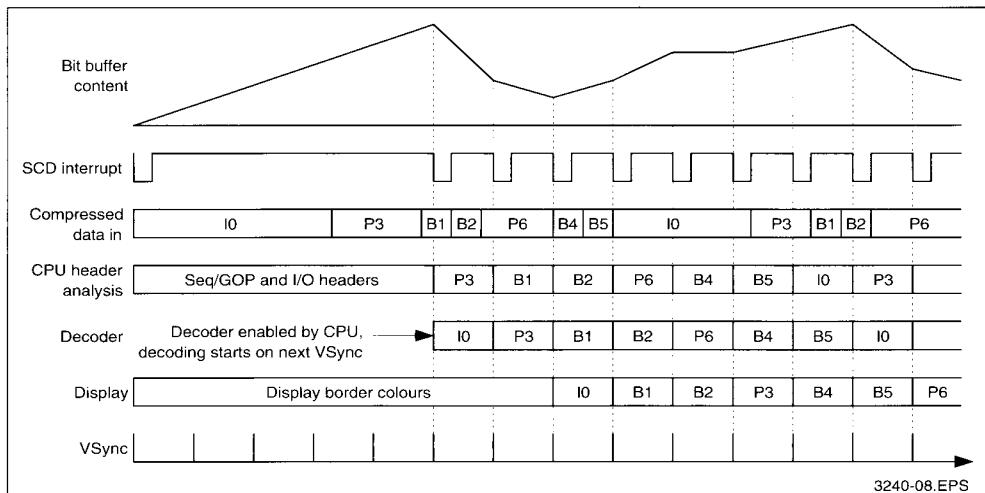
the first interrupt may lead to the bit buffer becoming empty). The micro can take advantage of the time left to initialise the decoder registers.

When the bit buffer is full enough, the micro can enable the decoder which then starts on the next VSync. This will empty the bit buffer while the bit stream input is still going on. The bit buffer content will then vary depending on the number of bits extracted when decoding a frame compared with the number of bits input during the frame (this will depend on the compressed bit rate). Generally I frames are bigger than P frames which are themselves bigger than B frames.

The decoding process takes, on average, less than one frameUs duration. After each VSync an interrupt generated by the SCD will indicate a new picture header to the micro which then has one frame duration to set the relevant decoder registers.

The display of the first I0 picture cannot start on the VSync just after it has been decoded, as it is followed by a predictive picture that is not displayed immediately. Therefore the display starts 2 VSync periods after the start of decoding. This is the absolute maximum delay. In H.261 applications the display can start with only one frame delay as there are no bidirectional frames. Before starting the framesUs display it is possible to display the border colour by forcing bit in a register. This avoids displaying frames which have not been updated.

Figure 7 : Timing Example



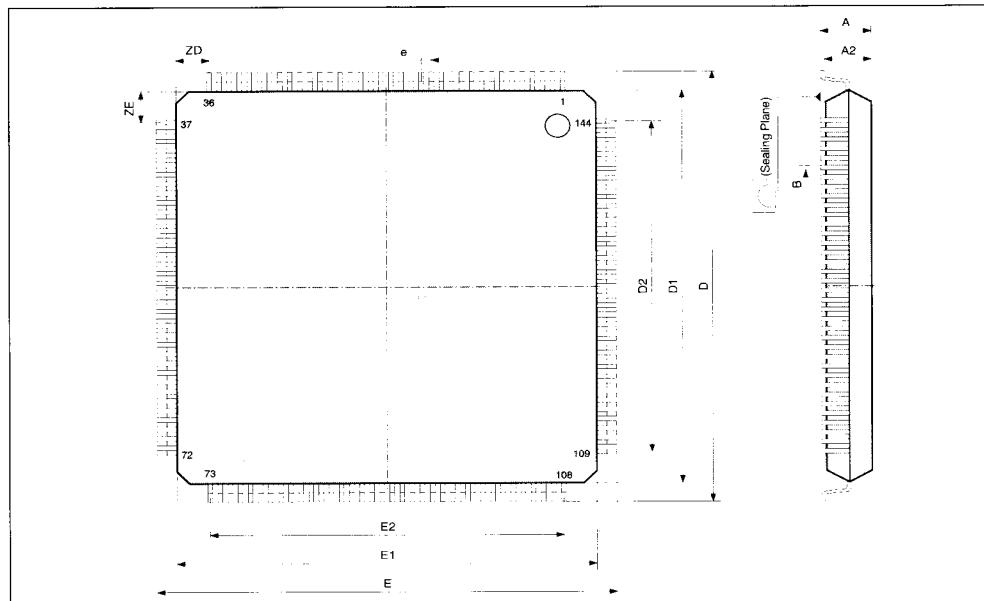
**VIII. REGISTERS**

The STi3240 has 256 addresses:

Address(hex)	Type	Description
<b>COMPRESSED DATA REGISTERS</b>		
00	W	Compressed Data Fifo
01	R	Header Data Fifo
02	R	Header Start Code
<b>STATUS &amp; CONFIGURATION REGISTERS</b>		
03	R/W	Configuration
04	R	Status
05	R/W	Control
06	R/W	Interrupt Vector
07	R/W	Interrupt Mask
08	R	Interrupt Status
09	R/W	Restart Header Scan
<b>INSTRUCTION REGISTERS</b>		
0A	W	Decoding Instruction
0B	W	Display Instruction
<b>FRAME POINTERS</b>		
0C	R/W	Displayed Frame Pointer
0D	R/W	Reconstructed Frame Pointer
0E	R/W	Forward Frame Pointer
0F	R/W	Backward Frame Pointer
<b>BIT BUFFER REGISTERS</b>		
10	R/W	Bit Buffer Size
11	R/W	Bit Buffer Status
19	R/W	Bit Buffer Threshold
<b>DECODED FRAME SIZE</b>		
12	R/W	frame width in macroblock
13	R/W	Number of macroblock per frame
<b>DISPLAY CONTROL REGISTERS</b>		
14, 15	R/W	display offset
16, 17	R/W	display size
1A, 1B	R/W	border colour and control bits
<b>QUANTIZATION WEIGHTING TABLE</b>		
1C	W	Intra Weighting Table Transfer Initialisation
1D	W	Non Intra Weighting Table Transfer Initialisation
1E	W	Weighting Table Write
1F	W	test configuration register

Note: all other addresses are reserved.

**PACKAGE MECHANICAL DATA**  
144 PINS - PLASTIC QUAD FLAT PACK



Dim.	mm			inches		
	Min	Typ	Max	Min	Typ	Max
A			3.92			.160
A2	3.17	3.42	3.67	0.125	0.134	.144
B						
D	30.95	31.20	31.45	1.219	1.228	1.238
D1	27.90	28.00	28.10	1.098	1.102	1.106
D2		22.75			0.896	
e		0.65			0.026	
E	30.95	31.20	31.45	1.219	1.228	1.238
E1	27.90	28.00	28.10	1.098	1.102	1.106
E2		22.75			0.896	
ZD		2.63			0.104	
ZE		2.63			0.104	

Information furnished is believed to be accurate and reliable. However, SGS-THOMSON Microelectronics assumes no responsibility for the consequences of use of such information nor for any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of SGS-THOMSON Microelectronics. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all informations previously supplied. SGS-THOMSON Microelectronics products are not authorized for use as critical components in life support devices or system without express written approval of SGS-THOMSON Microelectronics.

© 1992 SGS THOMSON Microelectronics – Printed in Italy – All Rights Reserved

SGS-THOMSON Microelectronics GROUP OF COMPANIES

Australia - Brazil - France - Germany - Hong Kong - Italy - Japan - Korea - Malaysia - M  
Singapore - Spain - Sweden - Switzerland - Taiwan - United King

031600 ✓ \_ R