SGS-THOMSON MICROELECTRONICS

STi3500A

MPEG-2 / CCIR 601 VIDEO DECODER

- REAL-TIME DECODING OF MPEG-1 AND MPEG-2 VIDEO BITSTREAMS AT CCIR 601 RESOLUTION
- FULLY SUPPORTS MPEG-2 MAIN PROFILE/MAIN LEVEL (MP@ML)
- VIDEO OUTPUT COMPATIBLE WITH CCIR 601
- INTERLACED AND LINE-SEQUENTIAL OUTPUT MODES, 3:2 PULLDOWN SUPPORTED
- HORIZONTAL AND VERTICAL FILTERS FOR PICTURE FORMAT CONVERSION
- ON-SCREEN DISPLAY FUNCTION
- AUTOMATIC MACROBLOCK ERROR CONCEALMENT
- DIRECT ADDRESSING OF UP TO 32MBITS OF DRAM
- STANDARD 8-BIT INTERFACE FOR MICROCONTROLLER AND COMPRESSED DATA INPUT
- MAXIMUM POWER DISSIPATION 1W

APPLICATIONS

- DBS RECEIVER
- DIGITAL TV RECEIVER
- DIGITAL CABLE TV RECEIVER

DESCRIPTION

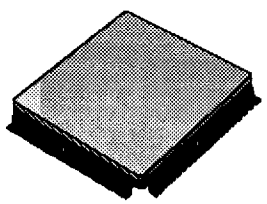
The STi3500A is a real-time video decompression processor supporting the MPEG-1 and MPEG-2 standards at video rates up to 720 x 480 x 60Hz or 720 x 576 x 50Hz. The complete decoding function is realised with the STi3500A, a standard 8-bit microcontroller and a bank of DRAM memory. A typical memory configuration is four 256K x 16 DRAMs.

The STi3500A requires minimal support from an external microcontroller, which is mainly required to initialise the decoder at the start of every picture. To aid the external processing of the upper layers of MPEG bitstream syntax, a start code detector is provided on-chip. In addition registers are provided to allow the tracking of time-stamps.

User-defined bitmaps may be superimposed on the displayed picture through use of the on-screen display function. These bitmaps are written directly into the DRAM memory by the microcontroller.

Picture format conversion for display is performed by a vertical and a horizontal filter (sample rate converter).

Undetected bitstream errors which would cause decoder errors bring into play an error concealment function, replacing the lost data with data from a previous picture.



PQFP144
(Plastic Quad Flat Pack)

ORDER CODE : STi3500ACV

7929237 0072966 87T

CONTENTS

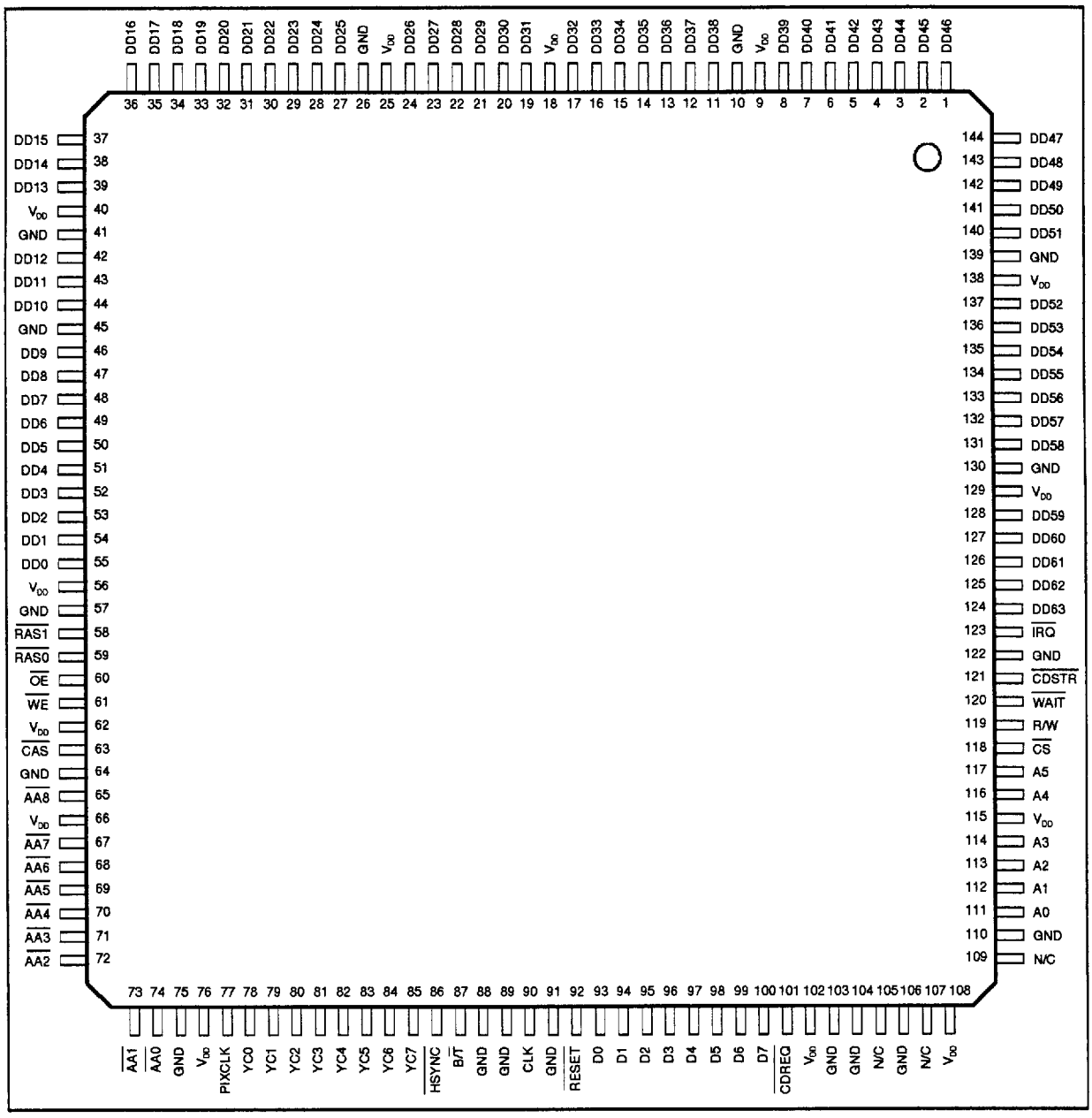
SGS-THOMSON MICROELECTRONICS

7929237 0072967 706

7929237 0072968 642

I - PIN DESCRIPTIONS

I.1 - Pin Connections



SGS-THOMSON MICROELECTRONICS

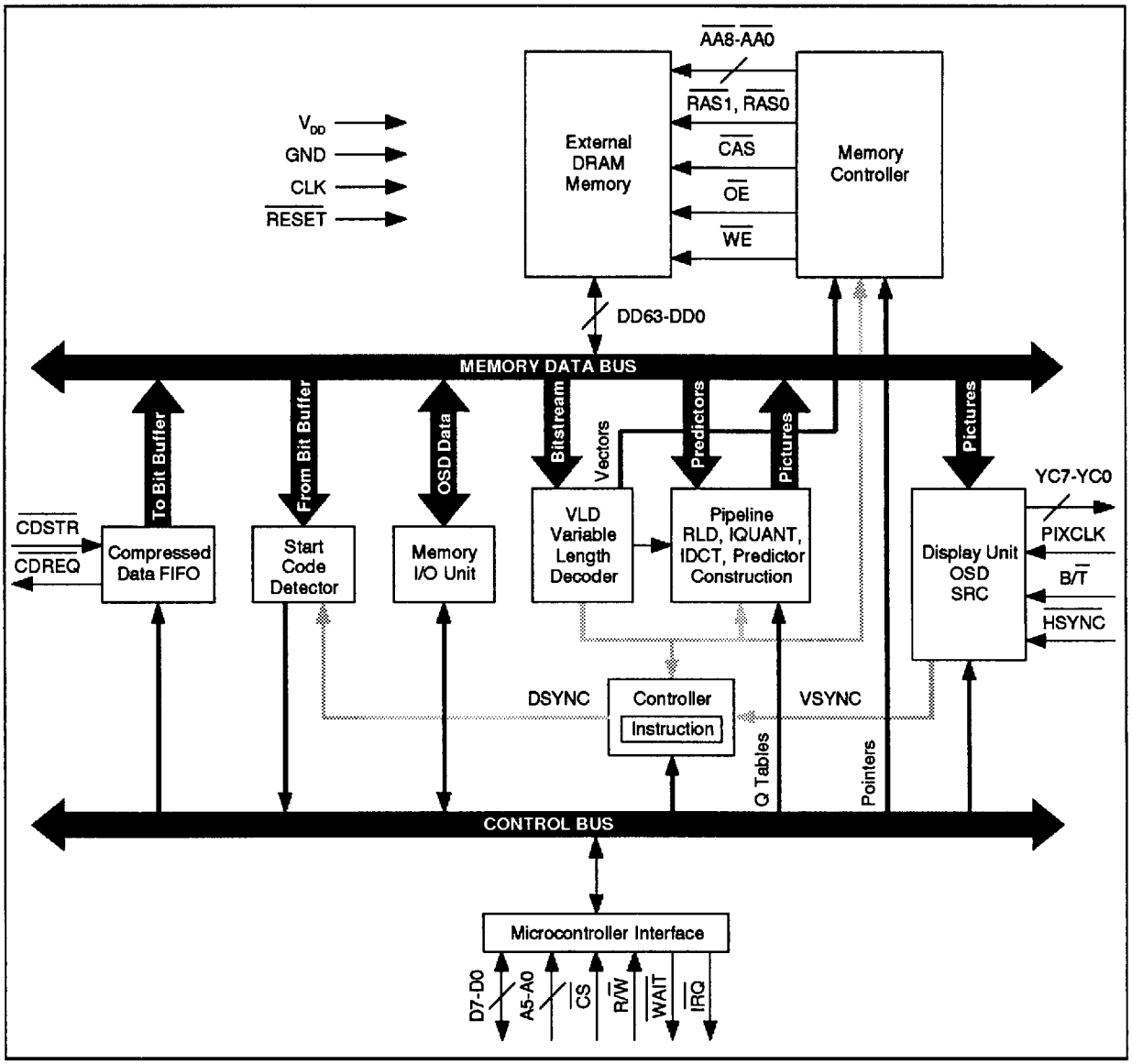
7929237 0072969 589

I.2 - Pin List

Pin N°	Name	Type	Description
SYSTEM SERVICES AND NON-FUNCTIONAL PINS			
9, 18, 25, 40, 56, 62, 66, 76, 102, 115, 129, 138	V_{DD}		Power Supply
10, 26, 41, 45, 57, 64, 75, 91, 103, 122, 130, 139	GND		Ground
90	CLK	In	System Clock
92	$\overline{\text{RESET}}$	In	Master Reset
105, 107,109		N/C	Reserved Pin, Leave Unconnected
108		V_{DD}	Reserved Pin, Connect to V_{DD}
88, 89, 104, 106, 110		GND	Reserved Pins, Connect to Ground
DRAM INTERFACE			
124-128, 131-137, 140-144, 1-8, 11-17, 19-24, 27-39, 42-44, 46-55	DD63 - DD0	In/Out	Bidirectional Data Port
65, 67-74	$\overline{\text{AA8}}$ - $\overline{\text{AA0}}$	Out	Address
58, 59	$\overline{\text{RAS1}}$, $\overline{\text{RAS0}}$	Out	Row Address Strobes for Banks 1 and 0
63	$\overline{\text{CAS}}$	Out	Column Address Strobe
60	$\overline{\text{OE}}$	Out	Output Enable
61	$\overline{\text{WE}}$	Out	Write Enable
VIDEO INTERFACE			
85-78	YC7 - YC0	Out	Multiplexed YC_BC_R Video Port
77	PIXCLK	In	Pel Clock
87	B/$\overline{\text{T}}$	In	Bottom/top Field Selection (Vertical Sync)
86	$\overline{\text{HSYNC}}$	In	Horizontal Sync
MICROCONTROLLER INTERFACE			
100-93	D7 - D0	In/Out	Bidirectional Data Bus
117, 116, 114-111	A5 - A0	In	Address
118	$\overline{\text{CS}}$	In	Chip Select
119	R/$\overline{\text{W}}$	In	Read/Write Selection
120	$\overline{\text{WAIT}}$	Out (3-state)	Data Acknowledge
101	$\overline{\text{CDREQ}}$	Out	Compressed Data Request
121	$\overline{\text{CDSTR}}$	In	Compressed Data Strobe
123	$\overline{\text{IRQ}}$	Out (Open-drain)	Interrupt Request

7929237 0072970 2T0

II - BLOCK DIAGRAM



SGS-THOMSON MICROELECTRONICS

7929237 0072971 137

III - FUNCTIONAL DESCRIPTION

III.1 - STi3500A Architecture

A functional block diagram of the STi3500A is given in Section II. The three external interfaces to the microcontroller, DRAM memory and display are also shown. Together with a minimum of 8 or 16Mbits of DRAM, a microcontroller, and some video post processing, a complete video decoder system can be constructed.

The STi3500A has two internal global buses, a high speed 64-bit memory data bus, and a 16-bit control bus. All data transfers to and from the external DRAM memory pass through the memory bus. This data is organised in packets in order to take advantage of page-mode access to the memory. The memory controller allocates bus bandwidth to the processes requiring data transfers according to a fixed priority scheme. The low bandwidth control bus is the communication path for data passing through the microcontroller interface.

The microcontroller interface has an 8-bit data bus and a 6-bit address bus. This access port has two functions :

- to pass the compressed data into the bit buffer, located in the external DRAM,
- to enable control of the STi3500A by providing interrupts and a path for accessing internal registers.

The DRAM interface includes all of the signals necessary for control of the memory. Refresh is handled automatically by the STi3500A. The memory is used to hold the bit buffer, store decoded pictures, and provide the display buffer. It also holds the user-defined on-screen display (OSD) bitmap and can be used by the microcontroller for private storage of data.

The video interface outputs digital video in 8-bit serial C_BYC_R format under the control of an external clock and synchronization signals.

During the process of decoding, there are four concurrent activities :

- buffering of the incoming bitstream,
- searching for start codes in the bitstream,
- decoding of a picture,
- display of a picture.

For each of these processes, the microcontroller must set up parameters and monitor events communicated by interrupts. The main features of each of these processes are summarized below.

Bitstream Buffering

The STi3500A performs the bitstream buffering needed by the decoder. The size of this buffer, which is located in the DRAM memory, is set up by the user. The bitstream is input through the 8-bit microcontroller data bus. The writing process is asynchronous to all other processes in the STi3500A. The bitstream data passes through a 1Kbit internal FIFO (the compressed data FIFO) before being transferred in packets to the bitstream buffer through the memory data bus. An output signal (and associated interrupt) indicate when this FIFO is full, enabling the use of DMA for bitstream input.

The maximum continuous bitstream input rate is application-dependent. A rate of 15Mbit/s (where 1Mbit/s = 10^6bit/s) is possible when decoding MPEG-2 MP@ML bitstreams. The maximum burst rate, for up to 1Kbit bursts, is 228Mbit/s.

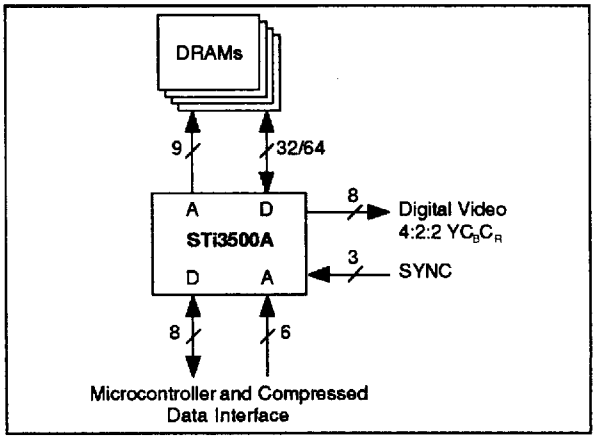
The STi3500A only accepts MPEG video bitstreams ; audio and systems data must not be input.

Start Code Search

The STi3500A is able to decode in its entirety a video bitstream from the slice layer downwards. The user must decode the higher layers (i.e. picture and upwards) in order to extract the information needed for decoding and appropriately set up the STi3500A registers and quantization tables. Since the header information is byte-aligned and requires minimal interpretation, this task represents only a small load on the microcontroller.

The start code detector parses the bitstream stored in the bit buffer and locates start codes corresponding to picture layer and above. When one of these start codes has been found, the start code detector stops and raises an interrupt. The microcontroller is then able to read the header data following the start code. The start code detector starts automatically whenever the decoding of a new picture starts and on user command. In normal operation, start code parsing is performed one picture in advance of decoding.

Figure 1 : Decoder System



7929237 0072972 073

Decoding

The STi3500A is a picture decoder ; it decodes a whole picture and then stops until instructed to decode the next picture present in the video bitstream.

Normally, the decoding of a new picture commences in response to the start of the displaying of a new picture. The registers whose contents can change from picture to picture are double-banked and are updated automatically when decoding starts. The bitstream is read from the bit buffer into the variable-length code decoder (VLD), and picture reconstruction can commence. Any predictors required are fetched from the appropriate area of the external memory, and the reconstructed picture is written back into the area of this memory assigned to the decoded picture. The pipeline contains the following sub-blocks :

- predictor construction and averaging ("filter"),
- run/level decoder (RLD),
- inverse zig-zag reordering,
- inverse quantizer (IQUANT),
- inverse discrete cosine transformation (IDCT).

While a picture is being decoded the start code detector is used to locate the start of next picture header, which the microcontroller then reads in order to set up the double-banked registers for the decoding of the next picture.

Display

The STi3500A is optimized for use with an interlaced display. However, it can also be programmed to produce a non-interlaced (line-sequential) output. The standard video clock rate is 27MHz, which corresponds to a pel rate of 13.5MHz. The active video data output format is compatible with CCIR 656 ; 00 and FF codes are never generated.

In order to match the horizontal size of the decoded picture to the display line length, an 8-tap upsampling filter, or sample-rate converter (SRC), is provided for both luminance and chrominance. A 2-tap vertical filter is provided for reconstruction of chrominance samples for 4:2:2 output, and for vertical luminance interpolation when displaying half-resolution pictures. The vertical filter includes a data delay line of length 720.

An on-screen display (OSD) function allows the user to define a bitmap for each field which can be superimposed on the decoded picture output. An OSD bitmap is defined as a set of rectangular regions of programmable position and size, each of which has a unique palette of 4095 colors. The 4096th "color" is transparency.

OSD data is written into memory areas assigned for this purpose by the user. Reading and writing to and from the memory through the microcontroller interface can take place at any time. A block move feature allows OSD data to be moved from one part of memory to another without microcontroller intervention.

III.2 - Prediction Modes Supported

III.2.1 - Field Pictures and Frame Pictures

Two types of frame sequence are defined in MPEG-2, progressive and interlaced. In a progressive sequence each picture is a frame - separate fields are not defined. In an interlaced sequence each frame has two fields, a top field and a bottom field*. Each frame in an interlaced sequence may be treated either as a single picture, or as a pair of pictures, one for each field. An interlaced sequence can contain a mixture of frame pictures and field pictures. A frame picture in an interlaced sequence may also be progressive. The different possibilities are shown in Figure 2.

Frames are always sampled in the 4:2:0 format in the main and simple profiles of MPEG-2. This pattern is shown in Figure 3.

* The terms odd field and even field are not used in MPEG-2.

Figure 2 : Frame Pictures and Field Pictures

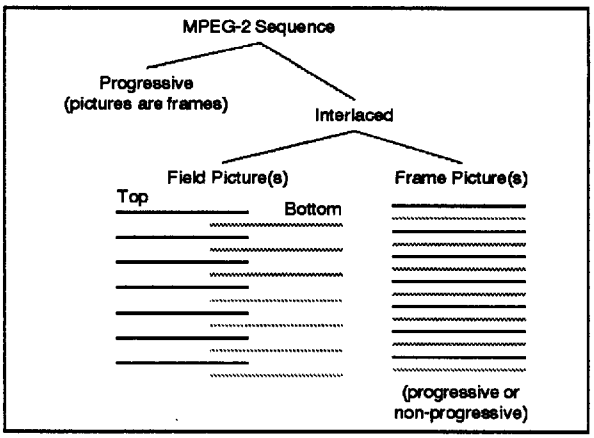
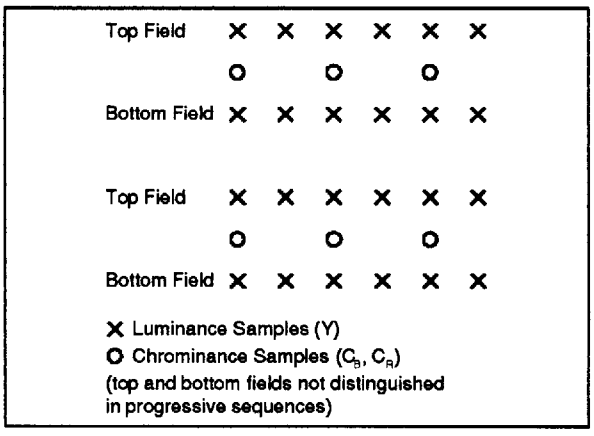


Figure 3 : Frame Sampling Structure



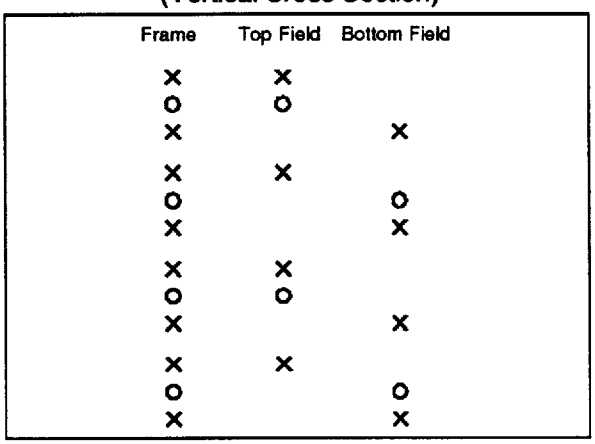
SGS-THOMSON MICROELECTRONICS

7929237 0072973 TOT

It is the responsibility of the encoder to resample the input picture chrominance data to conform with this format. When encoding an interlaced sequence, the encoder may either interpolate each half of the chroma samples using information from a single field, or use all of the chrominance data in the frame. The latter would be the case for a progressive frame, i.e. a frame in which all of the picture data was sampled at the same instant. The decoder must reconstruct the chrominance data for display in the appropriate manner.

When an interlaced frame is coded as two field pictures, the chrominance samples maintain their spatial positions, as shown in Figure 4. Note that the chrominance samples are not midway between luminance samples in the vertical direction.

Figure 4 : Position of Chrominance Samples in Field Pictures (Vertical Cross Section)



The different prediction modes available in MPEG-2 are outlined below. The prediction mode is defined at the macroblock layer ; therefore within a single picture more than one prediction mode may be defined.

III.2.2 - Prediction in Frame Pictures

Macroblocks in frame pictures contain information from both fields. Each macroblock represents a 16 x 16 area of the image. In an interlaced picture macroblocks will contain information from both fields, as can be seen from Figure 3 and Figure 4.

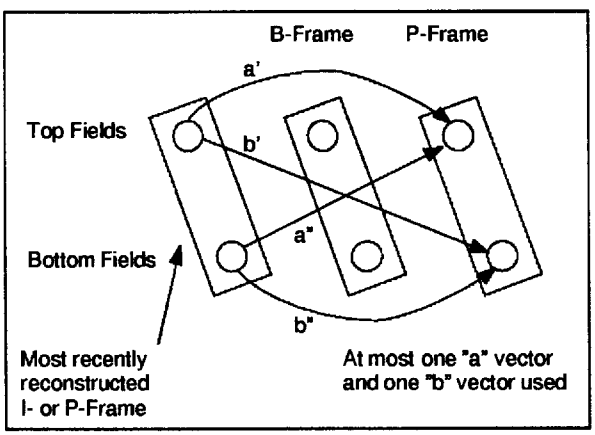
It is sometimes more efficient to predict such a macroblock from other interlaced macroblocks, and sometimes more efficient to predict each field separately. The former corresponds to frame prediction, the latter to field prediction.

Frame prediction in MPEG-2 is identical to the prediction method of MPEG-1. When decoding a P-picture (frame) the predictor macroblocks are taken from the most recently reconstructed I- or P-frame. This reference frame could have been reconstructed as a frame picture or two field pictures. In a B-picture (frame) the predictor macroblocks are taken from either or both of the two most recently reconstructed I- or P-frames. Each of these reference frames could be a frame picture or a pair of field pictures. Not more than one motion vector is necessary to locate the predictors for P-frames, two for B-frames.

With field prediction, the predictors for each field are independent. In a P-frame picture with field prediction, the prediction is from the two fields of the most recently decoded I- or P-frame, as shown in Figure 5. This frame may have been decoded as a frame picture or as two field pictures. One vector is used for each predictor, which could be in a field with the same parity* as the field being predicted, or the opposite. Each predictor contains 16 x 8 luminance samples and two blocks of 8 x 4 chrominance samples, all from the same field.

* The MPEG-2 convention is that the top field has parity zero, and the bottom field parity one.

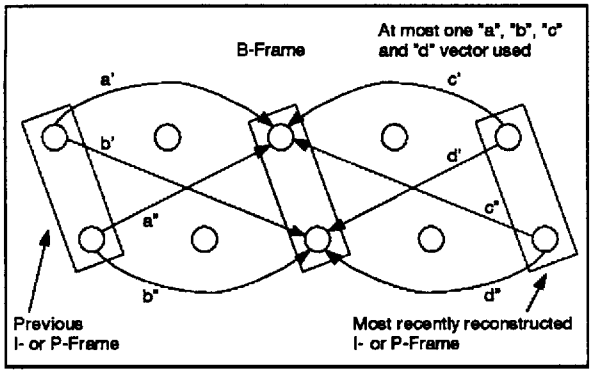
Figure 5 : Field Prediction of P-Frame Picture



7929237 0072974 946

Field prediction of a B-frame picture requires up to 4 vectors, as shown in Figure 6. The predictions for each field are formed by averaging the forward and backward predictors. The frames from which the predictors are taken may have been decoded as a frame picture or as two field pictures.

Figure 6 : Field Prediction of B-Frame Picture



III.2.3 - Prediction in Field Pictures

Macroblocks in field pictures contain information from only one field. A 16 x 16 macroblock represents an area of the image of size 16 x 32, i.e. it is twice as high as it is long since half of the lines belong to the other field. In a sequence of field pictures, the two fields of a frame may have a different picture type. The two simplest prediction modes in field pictures are field prediction and 16 x 8 motion compensation.

Field prediction in field pictures is the analogue of frame prediction in frame pictures. In both cases zero, one or two motion vectors are required for each 16 x 16 predictor. Figure 7 shows the predictor sources for a P-field picture. When the second field is being predicted, one of the most recently reconstructed I- or P-fields may in fact be the first field of the same frame. In the figure it is assumed that the top field is transmitted first, but it is also possible for the bottom field to be transmitted first. In the top case, the reference fields may have been decoded as two field pictures or as a single frame picture, and in the bottom one the earlier reference field may have been decoded as a field picture or as one field of a frame picture.

Figure 8 illustrates field prediction for a B-field picture. Here each half of the prediction is made from one of the two most recently reconstructed I- or P-fields which belong to the same frame. These frames may have been decoded as frame pictures or as pairs of field pictures. The prediction for the bottom field is made from the same reference fields as those used for the top field.

Figure 7 : Field Prediction of P-Field Picture

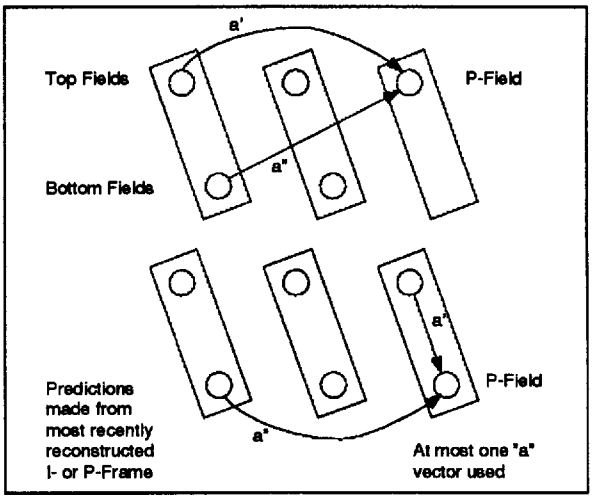
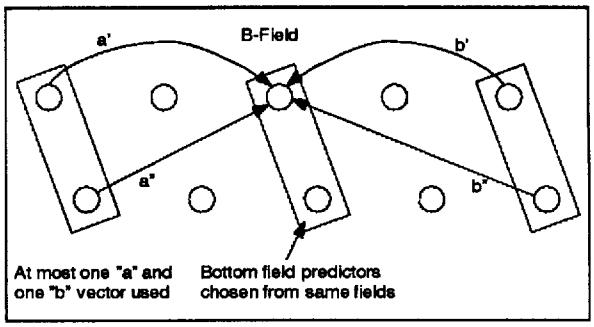
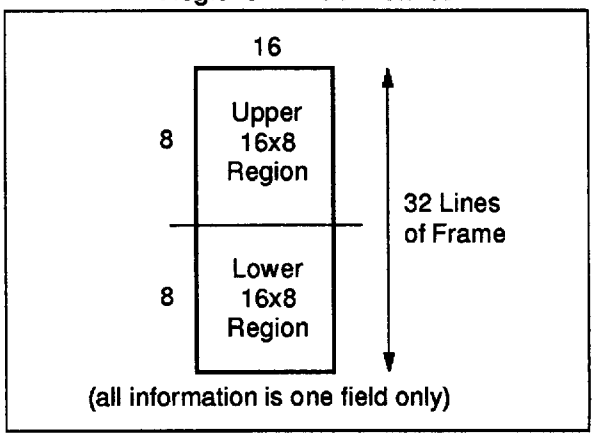


Figure 8 : Field Prediction of B-Field Picture



When using the 16 x 8 motion compensation prediction mode, the upper and lower halves of each macroblock are predicted separately. Each half-macroblock covers a 16 x 16 region of a frame, as shown in Figure 9. Macroblocks now requires zero, two or four vectors since independent vectors are required for each half-macroblock.

Figure 9 : Upper and Lower Macroblock Regions in Field Pictures



SGS-THOMSON MICROELECTRONICS

7929237 0072975 882

16 x 8 prediction in a P-field picture follows the pattern of Figure 7, except that now an upper vector and a lower vector are required. These vectors may or may not reference the same field of the reference frame. Figure 8 also applies to the 16 x 8 case. In a B-field picture up to 4 vectors may be required, a maximum of two for each half-macroblock.

III.2.4 - Dual-prime Prediction

This prediction mode is only used in P-pictures and is applicable to sequences in which there are no B-pictures. Only one vector, plus a small differential vector, is sent with every macroblock. All necessary vectors are derived from these, as shown below for the cases of field pictures and frame pictures.

Figure 10 : Dual-Prime Prediction in Field Pictures

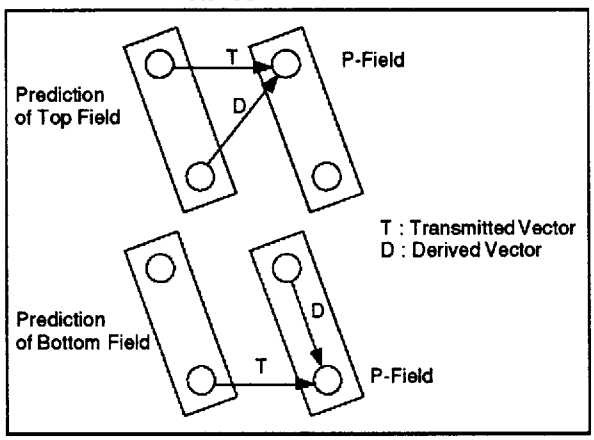
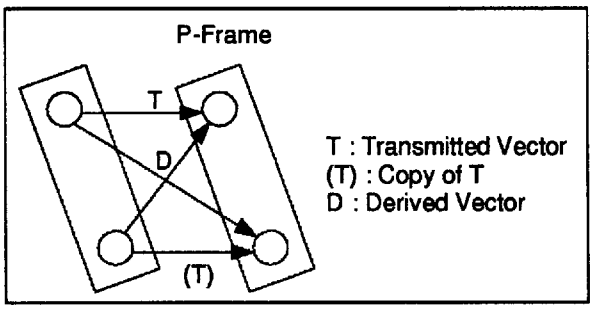


Figure 11 : Dual-Prime Prediction in Frame Pictures



In both cases the predictor used for each field is the average of the two predictors specified by the vectors. (Dual-prime prediction thus retains one of the features of B-picture prediction). In field pictures, the predictors are of the same type as those used in field prediction, i.e. they are of size 16 x 16. In frame pictures, the predictors are the same as those used in field prediction, i.e. they are of size 16 x 8 and contain information from one field only.

III.3 - Other MPEG-2 Modes Supported

In this section the other major differences of MPEG-2 with respect to MPEG-1 are indicated. All are supported by the STi3500A.

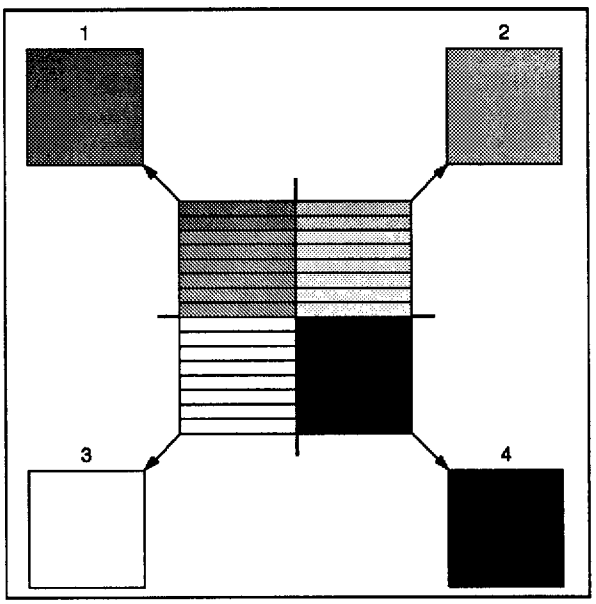
Extended Motion Vector Range

In MPEG-1 the maximum motion vector range is -1024 to 1023 for full-pel resolution, and -512 to 511.5 for half-pel resolution. MPEG-2 specifies a maximum range of -2048 to 2047.5 at half-pel resolution ; full-pel resolution is not defined. MPEG-2 MP@ML specifies a vertical range of -128 to 127.5 and horizontal range of -1024 to 1023.5. The STi3500A supports a maximum range of -1024 to 1023.5 in both dimensions. In field pictures the vertical range is halved.

Frame/Field DCT Coding

The luminance part of macroblock data may be reordered after the IDCT is performed on the blocks. The frame ordering is the same as that used in MPEG-1, as shown in Figure 12.

Figure 12 : Frame DCT Ordering

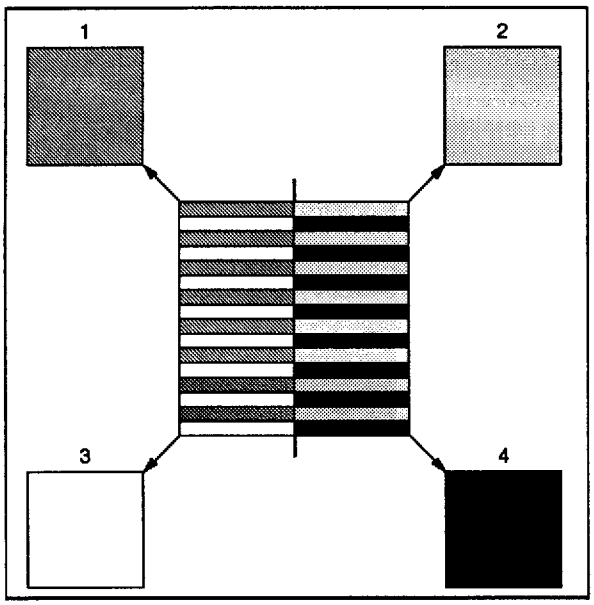


7929237 0072976 719

The field ordering is shown in Figure 13.

Field or frame ordering is selectable macroblock-by-macroblock and is specified in the bitstream.

Figure 13 : Field DCT Ordering



Concealment Motion Vectors

The option exists in MPEG-2 to transmit motion vectors with intra macroblocks. This additional information is intended to be used when data is lost due to errors in the bitstream. In this case the lost macroblocks are reconstructed as if they were skipped macroblocks in a forward predicted field or frame, using motion vectors which have been stored by the decoder in the previous row of macroblocks.

Alternative Zig-Zag Scan

MPEG-2 allows two zig-zag block scanning patterns. Which one is used is specified in the picture layer of the bitstream. The second pattern is intended to maximise the run lengths of blocks containing data from a single field.

Increased DCT Coefficient Precision

MPEG-2 decoders must be able to handle DCT coefficients in the range -2048 to 2047 (12 bits). The range in MPEG-1 is -256 to 255.

Alternative Intra VLCs

The precision to which the DC coefficients of intra-coded blocks are coded can be changed every picture. Values of 8, 9 or 10 bits are allowed in MPEG-2 MP@ML. (The precision is fixed at 8 bits in MPEG-1).

There is also the option in MPEG-2 to select at the picture layer an alternative, more efficient, VLC table for coding the intra DCT coefficients. This selection is independent of the intra DC precision selection.

Non-Linear Quantizer Scale Mapping

Two mappings are defined between the 5-bit quantizer scale code transmitted in the bitstream and the actual quantizer scale. The first mapping is linear, as in MPEG-1. The alternative mapping is non-linear and is designed to give a finer control of quantizer scale when the values are low. Which mapping is used is defined at the picture layer of the bitstream.

Alternative IDCT Mismatch Control

The coefficients at the input to the IDCT are modified in order to avoid certain patterns of data. This operation reduces the mismatch of outputs between different IDCT implementations, specifically in the encoder and decoder. In MPEG-1 the dequantized AC coefficients are forced to be odd before saturation. In MPEG-2 all of the coefficients of a block are summed after the saturation stage. If this sum is even, then the last coefficient is adjusted.

Pan-Scan

Information is transmitted in the bitstream which defines the dimensions and location of the displayable region within the decoded picture. The position of this window can be changed every field. This feature is useful, for example, when displaying part of a 16/9-format coded picture on a 4/3-format display. The MPEG-2 standard does not specify what should be done with this information, since it affects the display and not the decoding process.

III.4 - VSYNC and DSYNC

These are the two principal internal synchronization signals in the STi3500A. They will be referred to many times in the sections that follow.

VSYNC occurs on every transition of the input signal B/$\overline{T}$. VSYNC always starts the display of a new field. It also can potentially start the decoding of a picture, since it is the primary synchronization signal of the pipeline controller (see block diagram).

At the start of a new picture decoding task, the signal DSYNC is generated. DSYNC also launches the search for a start code. Thus the picture decoding and start code detection functions are slaved to the display. (It is also possible for the external controller to initiate start code detection if required).

SGS-THOMSON MICROELECTRONICS

7929237 0072977 655

III.5 - A Note on Conventions

In this data sheet, the following conventions are used when documenting the functions of signals :

I/O signals can either be active high or active low. The former have names without an overbar (i.e. SIGNAL), the latter have an overbar (i.e. $\overline{\text{SIGNAL}}$). Where a signal has two different and mutually exclusive actions, a dual name is used (e.g. COME/$\overline{\text{GO}}$).

Internal signals and variable names (e.g. bits in registers) are always documented as active high.

When the condition indicated by the name of the signal or variable is true, the signal or variable is said to be true, asserted or to have the value 1.

When the condition indicated by the name of the signal or variable is not true, the signal or variable is said to be false, de-asserted or to have the value 0.

When an active high signal is true or asserted, the logic voltage level is high.

When an active low signal is true or asserted, the logic voltage level is low.

When an internal signal or variable is set, it has the value 1. A bit is never "set to 0", but "reset to 0".

When an internal signal or variable is reset, it has the value 0.

Hexadecimal numbers are indicated by appending an "h", e.g. A70h.

III.6 - Summary Specification

Bitstreams Accepted

MPEG-1 video (ISO/IEC 11172-2).

MPEG-2 video (ISO/IEC 13818-2).

MPEG-2 Profiles/Levels Supported

Main Profile @ Main Level (MP@ML).

Main Profile @ Low Level (MP @ LL).

Simple Profile @ Main Level (SP @ ML).

Maximum Picture Size

Width : 4080.

Number of macroblocks : 16,383.

Motion Vector Range

MPEG-1 : -1024 to 1023 (full pel), -512 to 511.5 (half pel) horizontal and vertical.

MPEG-2 : -1024 to 1023.5 horizontal and vertical.

(vertical range must be reduced in 8Mbit memory mode with certain picture sizes).

Compressed Data Input

8-bit asynchronous data port (shared with microcontroller interface).

Peak input rate : 28.5Mbyte/s (228Mbit/s).

Maximum sustained input rate (with 56MHz primary clock) : 100Mbit/s in 16Mbit memory mode, 60Mbit/s in 8Mbit memory mode*.

* In practice, the sustained rate will be constrained by the memory bandwidth required for real-time decoding.

Microcontroller Interface

8-bit data port with fixed length "WAIT" pulse acknowledge.

Single interrupt request pin.

DRAM Interface

External DRAM used for storage of picture buffers, bit buffer and on-screen display definitions.

32 or 64 bit data bus.

Refresh handled by STi3500A.

DRAM directly accessible through microcontroller interface.

Configurations supported : 4Mbits (1 bank), 8Mbits (1 bank), 16Mbits (1 bank), 20Mbits (2 banks), 32Mbits (2 banks).

Start Code Detection

Automatic detection of start codes of picture layer and above to enable microcontroller to access header data.

Counters provided for time-stamp tracking.

Decoding Pipeline

Instruction register set up each picture defines pipeline operation.

Double-buffered quantization matrices enable loading of new tables concurrently with decoding.

Error Concealment

Automatic concealment of errors detected by VLD and decoding pipeline by macroblock copy.

Video Output

8-bit 30MHz multiplexed C_BYC_R port, compatible with CCIR 601 and 656 (00h and FFh values never output).

External pel clock and horizontal/vertical synchronization required.

Interlaced or line-sequential output.

3:2 pulldown operation supported.

Programmable horizontal up-sampling by 8-tap filter.

Vertical chroma reconstruction or luma filtering by 2-tap filter including 720-sample delay line.

7929237 0072978 591

Pan & Scan Vectors

Horizontal : Maximum vector size : 512 pels, resolution : 1/8 pel.

Vertical : Maximum vector size : 508 lines, resolution : 4 lines.

On-Screen Display (OSD)

Bitmap separately definable for each field can be superimposed on final picture output.

OSD defined as rectangular regions, each with unique palette defining 3 colors and transparency or 4 colors.

Number of regions limited only by the memory space allocated to OSD.

Block move facility available for reduction of microcontroller loading.

Primary Clock

56MHz maximum.

Power Dissipation

1W maximum.

Package

144-pin PQFP.

IV - MICROCONTROLLER INTERFACE AND COMPRESSED DATA INPUT

IV.1 - Interface Signals

The combined microcontroller/compressed data interface consists of the following signals :

D7 - D0	Data Port
A5 - A0	Address
$\overline{CS}$	Register Read/Write Strobe
R/$\overline{W}$	Read/write Selection
$\overline{WAIT}$	Acknowledge
$\overline{CDSTR}$	CD Write Strobe
$\overline{CDREQ}$	CD Request
$\overline{IRQ}$	Interrupt Request

Input of compressed data and read/write access to STi3500A registers share the same data port, D7-D0. The signal $\overline{CS}$ requests a register access cycle, while signal $\overline{CDSTR}$ requests a compressed data write cycle.

The multiplexing of data onto D7-D0 and the arbitration of compressed data and register I/O cycle requests must be performed externally.

The microcontroller interface is completed by the signals A5-A0, R/$\overline{W}$ and $\overline{WAIT}$. The delay between the start of a register access and $\overline{WAIT}$ is a fixed interval. $\overline{WAIT}$ will only be needed if the microcontroller has a short bus cycle.

The signal $\overline{CDREQ}$ completes the compressed data transfer handshake.

STi3500A interrupt requests are signalled by $\overline{IRQ}$.

IV.2 - Register Access

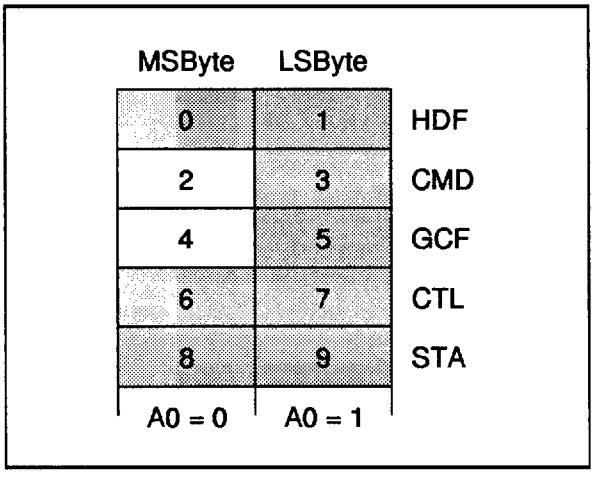
The 6 address bits, A5-A0, select one of the 64 one-byte internal register locations. The function of each register is detailed in section XII, "REGISTERS". Some are read only, some write only, and some read/write. The signal R/$\overline{W}$ defines whether the register access cycle is a read or a write.

A cycle is defined by the assertion of signal $\overline{CS}$. In response to this the signal $\overline{WAIT}$ is always asserted. When $\overline{WAIT}$ is de-asserted, indicating the completion of the read or write operation, $\overline{CS}$ can return to its high state and the cycle ends. The time from the assertion of $\overline{CS}$ and the de-assertion of $\overline{WAIT}$ has a fixed maximum value. During a register read/write cycle (i.e. while $\overline{CS}$ is low), the signal $\overline{CDSTR}$ must be high, since compressed data write and register read/write cycles are mutually exclusive. The timing diagrams of section XIV.2, "Register Read and Write Cycles", define the timing constraints. The minimum time between two successive single-byte read or write cycles is 70ns*. The signal $\overline{WAIT}$ is in its high impedance state when a read/write cycle is not in progress (i.e. while $\overline{CS}$ is high).

The internal registers are organised in 16-bit units, as shown in Figure 14. Some require both byte addresses, some only one. A 16-bit register is mapped into two consecutive addresses.

* This is equal to the sum of the values of tSLWH and tSHSL, defined in section XIV.2, "Register Read and Write Cycles".

Figure 14 : Register Addressing



In general register accesses require two cycles, with the most significant byte being accessed first. The exceptions are noted below.

SGS-THOMSON MICROELECTRONICS

7929237 0072979 428

Writing to a Register

To write to a 16-bit register, the most significant byte (A0 = 0) is written first. This byte is stored in an internal temporary register during the first cycle and is not written to the destination register. In the second cycle, the least significant byte (A0 = 1) is written. In this operation, both bytes, the new byte and the one stored in the first cycle, are written to the destination register. So for example to write to the CTL register, a write must first be made to address 6, followed by a write to address 7.

It is not possible to write to the most significant byte alone, but it is possible in certain cases to write to the least significant byte only. These are where the most significant byte is not defined, or where it is a read-only register. For example a single cycle write to the one-byte MWF register (address 11) is possible because the register at the preceding even address, MRF (address 10), is a read-only register.

The two parts of a register write operation must never be interrupted by another register write operation, since the contents of the internal temporary register could be overwritten before it is used.

Write cycle timing is given in Figure 67.

Reading a Register

To read a 16-bit register, the most significant byte (A0 = 0) is read first. In this cycle the most significant byte is available at the data port, and the least significant byte of the register is stored in an internal temporary register. (This is not the same physical register as that used to store the most significant byte when writing). In the second cycle (A0 = 1), the contents of the temporary register are sent to the data port ; the register addressed is not read again.

It is thus possible to read only the most significant byte of a register. In this case the byte loaded into the temporary register is not read. It is not possible to read the least significant byte alone ; the preceding most significant byte must be read first in order to load the least significant byte into the temporary register.

The two parts of a register read operation must never be interrupted by another register read operation, since the contents of the temporary register will be overwritten before it is read. It is possible to mix the cycles of the read from one register and the write to another. Also, any register read or write cycles can be mixed freely with compressed data write cycles.

Read cycle timing is given in Figure 65.

When it is required to read two consecutive bytes, it is possible to combine these into a single extended cycle. The cycle starts normally with the reading of the most significant (even) byte. When the de-assertion of $\overline{WAIT}$ indicates that the first byte is available, the least significant address bit can be changed. This causes the contents of the temporary register to be switched to the output. $\overline{WAIT}$ will remain high during the read of the second byte. The timing of this operation is given in Figure 66.

IV.3 - Interrupts

The conditions which can cause an interrupt are represented by the bits of the STA register. Any change of state from 0 to 1 of one of these bits (an "event") will cause an interrupt unless it is masked by the corresponding bit of the ITM register being reset. Each event causes a bit of the ITS register to be set, regardless of the state of ITM. Any unmasked bit becoming set in the ITS register causes an interrupt request, indicated by the driving of the open-drain signal $\overline{IRQ}$ into its asserted state.

The interrupt is acknowledged and its source(s) identified by reading the ITS register. The reading of the most significant byte of this register has the effect of clearing the whole register and thereby removing the interrupt request (i.e. $\overline{IRQ}$ returns to its undriven state). The least significant byte can be read from the temporary register in a second cycle. The timing of the removal of the interrupt request is given in section XIV.3, "Interrupt Acknowledge".

Before reading ITS, all interrupts should be masked by clearing the ITM register. This prevents an event occurring during the read cycle from generating a new interrupt. When ITM is restored, events which occurred while the mask was zeroed will now generate an second interrupt request.

IV.4 - Compressed Data Input

The compressed data input to the STi3500A must be either an MPEG-1 or an MPEG-2 video elementary stream.

Compressed data input is placed on the shared data bus, D7 - D0, and is strobed in on the rising edge of the signal $\overline{CDSTR}$. During a compressed write cycle the signal $\overline{CS}$ must be high, since compressed data write and register read/write cycles are mutually exclusive. Compressed data input bytes are written into the 1024-bit Compressed Data (CD) FIFO (see block diagram). Data from this FIFO are transferred in 512-bit bursts to the bit buffer area of the external memory, whose base address is specified by the BBG register. The rate at which these transfers can occur is governed by the number of higher priority requests waiting for service from the memory controller. The de-asser-

7929237 0072980 14T

tion of the signal $\overline{\text{CDREQ}}$ indicates that the CD FIFO is full, but that 3 more bytes can be written. No further compressed data bytes must be written while $\overline{\text{CDREQ}}$ remains de-asserted, since data will be lost.

The timing diagram of section XIV.4, "Compressed Data Write Cycle", defines the timing constraints. The minimum time between two successive write cycles is 35ns (giving an upper limit on port input rate of roughly 228Mbit/s).

Compressed data write cycles can be combined in any sequence with register read/write cycles.

Start codes in the compressed data stream must be byte-aligned.

The average rate of compressed data input over a picture period is one of the decoder performance constraints. It is related to picture size, prediction modes and primary clock frequency. This rate can be exceeded momentarily, provided that the number of bits written during a picture period does not exceed the number used in the performance calculation.

There are three ways of controlling the input of compressed data bytes :
- A. Handshake every byte.
- B. Byte input without handshake.
- C. Burst input without handshake.

In all cases, the average compressed data input rate cannot exceed that used in the STi3500A performance calculation.

A. Handshake every Byte

In this mode, the signal is $\overline{\text{CDREQ}}$ is checked after the writing of every byte. The maximum rate of data transfer is one byte every 55ns, assuming that $\overline{\text{CDSTR}}$ can be asserted again immediately $\overline{\text{CDREQ}}$ becomes high. This is equivalent to a maximum instantaneous rate of roughly 145Mbit/s. If this is greater than the relevant sustained rate given in the next section, some cycles will be delayed due to the CD FIFO being full.

B. Byte Input without Handshake

Provided that the input rate is kept below given limits, it can be guaranteed that the CD FIFO will never become full, thus making unnecessary the checking of CDREQ. The limits are :

Maximum sustained rate (normal memory mode) :

Primary clock frequency x 1.8.

With a primary clock of 56MHz, this rate is 100Mbit/s.

Maximum sustained rate (8Mbit memory mode) :

Primary clock frequency x 1.1.

With a primary clock of 56MHz, this rate is 60Mbit/s.

Maximum sustained rate (normal memory mode, display only) :

Primary clock frequency x 2.3.

With a primary clock of 56MHz, this rate is 128Mbit/s.

Maximum sustained rate (8Mbit memory mode, display only) :

Primary clock frequency x 1.4.

With a primary clock of 56MHz, this rate is 77Mbit/s.

Maximum sustained rate (normal memory mode, no decoding or display) :

Primary clock frequency x 13.

With a primary clock of 56MHz, this rate limited by the maximum CD port input rate of 228Mbit/s.

Maximum sustained rate (8Mbit memory mode, no decoding or display) :

Primary clock frequency x 8.

With a primary clock of 56MHz, this rate limited by the maximum CD port input rate of 228Mbit/s.

C. Burst Input without Handshake

Data may be entered in bursts of up to 1024 bits at the maximum rate (228Mbit/s). The minimum time between bursts is 350 primary clock cycles (6.4µs at 56MHz) in 16-Mbit memory mode, and 590 primary clock cycles (10.7µs) in 8-Mbit memory mode. These are the maximum times which must be allowed for the CD FIFO to be emptied into the bit buffer.

IV.5 - Bit Buffer Control

The amount of data in the bit buffer is available (in units of 2Kbits) by reading the BBL register. When this level is greater than or equal to the value loaded into the BBT register (also defined in units of 2Kbits), the status bit STA.BBF becomes true. This can be used to generate a "bit buffer nearly full" interrupt. When the bit buffer contains no data, the status bit STA.BBE becomes true. This can be used to generate a "bit buffer empty" interrupt.

When the CD FIFO is full, the status bit STA.BFF is true. This bit is thus equivalent to the signal $\overline{\text{CDREQ}}$.

If the bit CTL.PBO is set, then transfer of data from the CD FIFO to the bit buffer is prevented if the bit buffer level is at or above the level defined in the BBT register. If BBT is set to a value equal to the size of the bit buffer, then this automatic mechanism will ensure than overflow never occurs.

SGS-THOMSON MICROELECTRONICS

7929237 0072981 086

IV.6 - Bit Buffer Flushing

If it is required to complete the decoding of a sequence before the arrival of another, for example if the sequence contains a single picture, 63 zero bytes must be written at the end of the first sequence, to ensure that the last packet is always sent from the CD FIFO to the bit buffer.

If in the application it is not possible to do this, a flush can be initiated from the microcontroller by entering a test mode. An Application Note is available on this topic.

V - EXTERNAL MEMORY

V.1 - Memory Interface

The memory interface consists of the following signals :

DD63-DD0	Bidirectional Data Port
AA8-AA0	Address
RAS0	Row Address Strobe for Bank 0
RAS1	Row Address Strobe for Bank 1
CAS	Column Address Strobe
OE	Output Enable
WE	Write Enable

These signals correspond directly the DRAM signals of the same names. Note that the STi3500A address output signals are inverted.

Memories with multiplexed 9-bit row and 9-bit column addresses must be used, except in the second bank when in 20-Mbit mode, in which memories with 8-bit row and column addresses are required.

The memory interface is disabled when bit CTL.EDI is reset (the default state). In this state all of the memory interface signals are in their high impedance state. If it is required to keep the RAS and CAS signals high when the interface is disabled in order to reduce DRAM power consumption, they should be pulled up to V_{DD} with 20kΩ resistors.

Normal (16Mbit) Mode

In the normal mode all 64 bits of the data bus are connected to the memory.

The Figures 15, 16 and 17 illustrate the memory configurations possible in normal mode. Figure 15 shows the configuration in which there is a single bank of 16Mbits. The second row address strobe (RAS1) signal is not used.

Figure 15 : 16Mbit Memory Configuration

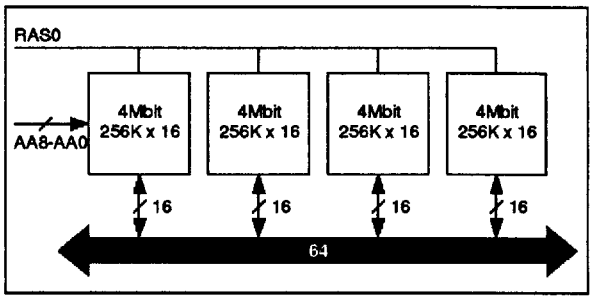
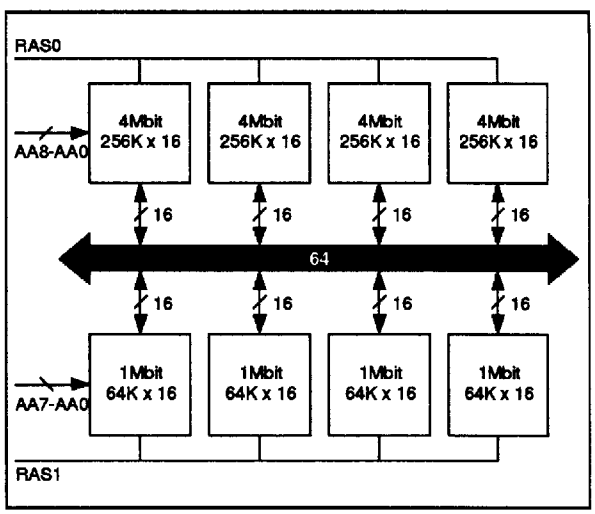


Figure 16 shows the configuration in which an extra 4Mbits is added in the second bank to give a total memory capacity of 20Mbits. The memory space of the second bank is contiguous with that of the first. When this mode is used, bit GCF.M20 must always be set. Address signals AA7 to AA0 are used for the second bank. (It is also possible to use the second 4Mbit bank alone, to give a 4Mbit configuration).

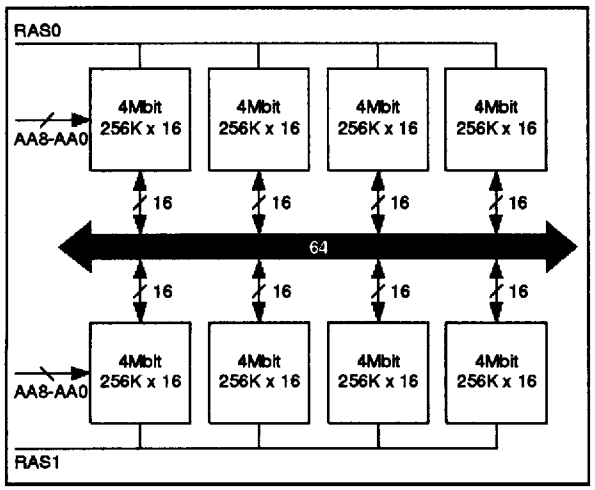
Figure 16 : 20Mbit Memory Configuration



7929237 0072982 T12

Figure 17 shows the configuration in which both banks contain 16Mbits. Bit GCF.M20 must be reset.

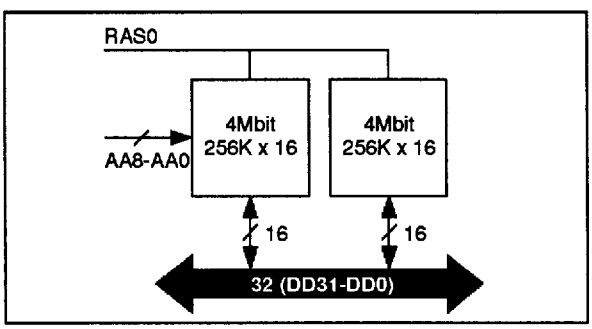
Figure 17 : 32Mbit Memory Configuration



8-Mbit Mode

In 8-Mbit mode only bits DD31 to DD0 are used. Only one memory configuration is possible - a single bank connected as shown in Figure 18. When this mode is used, bit CTL.S8M must always be set.

Figure 18 : 8Mbit Memory Configuration



V.2 - Timing Requirements

The timing parameters for the memory interface are given in section XIV.5, "DRAM Interface". In most instances, these are dependant on the primary clock frequency, which could enable slower memories to be used when the primary clock frequency is lower than the maximum. With a primary clock of 56MHz, "-80" or faster memories are required.

The maximum permissible memory access times are related to the memory interface parameters as follows :

$$t_{AA} = t_{CAL} - t_{DS}\ (read) = 3T - 12ns$$

$$t_{ACP}{}^{*} = 3T + 2ns$$

$$t_{CAC} = t_{CAS} - t_{DS}\ (read) = 2T - 4ns$$

$$t_{OEA} = t_{OCH} - t_{DS}\ (read) = 3T - 5ns$$

$$t_{RAC} = t_{CSH} - t_{DS}\ (read) = 5T - 6ns$$

Where T is the primary clock period.

* Also sometimes referred to as t_{CPA}.

V.3 - Refresh

Memory refresh is handled automatically by the STi3500A memory controller by inserting "CAS before RAS" refresh cycles. The duration of a refresh cycle is 9 primary clock periods. During these cycles RAS0 and RAS1 are driven together, thus refreshing a row from each bank. The refresh period is defined, in units of 24 primary clock periods, by the programming of GCF.RFI[6:0]. For example if 512 memory rows must be refreshed every 8ms and the primary clock is 55MHz, GCF.RFI[6:0] must be loaded with :

8ms/512 x 55MHz/24 = 35 (after rounding down)

SGS-THOMSON MICROELECTRONICS

7929237 0072983 959

V.4 - Memory Mapping Examples

The first example (Figure 19) shows the memory map in an application in which the picture size is 720 x 480. With 3 frame buffers each occupying just less than 4Mbits and an on-screen display (OSD) buffer sufficient to define a bitmap covering the full screen occupying 0.72Mbits, there is sufficient space for a 3.4Mbit bit buffer. It is assumed in the OSD space calculation that there is one OSD region per line. The frame buffer addresses are constrained to start on 128-word boundaries.

The second example (Figure 20) shows the memory map in an application in which the picture size is 720 x 576. With 3 frame buffers each occupying 4.75Mbits there is sufficient space for a 1.75Mbit bit buffer. The frame buffer addresses are constrained to start on 128-word boundaries. At the end of each of the frame buffers, there are 64 unused words (4096bits) which can be used for OSD buffers.

If a larger bit buffer is required, or if space needs to be made for OSD (0.86Mbits are needed to fill a 720 x 576 screen), then a 20Mbit configuration will be necessary*.

If it is required to store more than 3 frame buffers (for example if overwrite mode is not used), then at least 20Mbits (for 720 x 480) or 32Mbits (for 720 x 576) are necessary.

* It is possible to extend the bit buffer off-chip if the burst mode of compressed data input is used. An application note is available on this topic.

Figure 19 : 720 x 480 Frames in 16Mbits

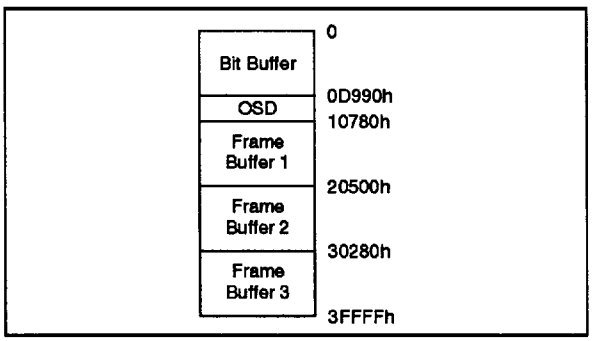
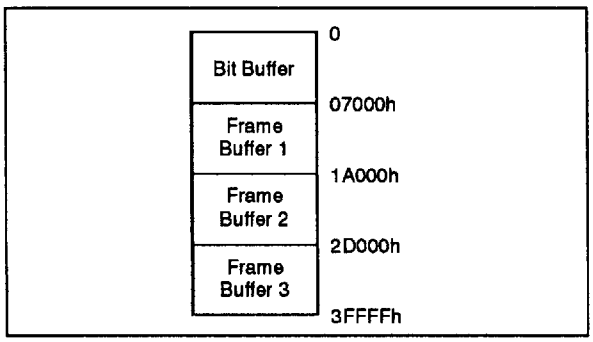


Figure 20 : 720 x 576 Frames in 16Mbits

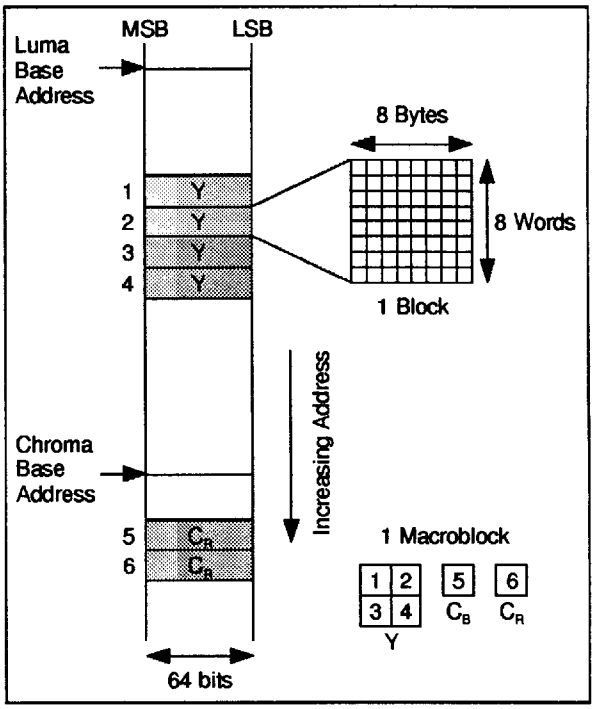


V.5 - Picture Storage Data Structure

Pictures are stored in memory as frames. They can be read and written as either frames or fields. In the latter case, every second line is accessed.

Figure 21 shows how the data for a frame is mapped into memory when the memory interface is operating in normal (16Mbit) mode.

Figure 21 : Storage of a Macroblock



The luminance and chrominance data for a frame are stored separately ; first all of the luminance macroblocks, and then all of the chrominance macroblocks. The base address of the luminance is defined by the frame base address as set up in one of the registers DFP (display frame pointer), RFP (reconstructed frame pointer), FFP (forward frame pointer) or BFP (backward frame pointer). The chrominance base addresses are calculated internally using the contents of the DFS (decoded frame size) register and the bits GCF.DFA[7:0]. For example for the display buffer the luminance start address in normal mode is :

$$32 \times DFP$$

While the chrominance start address is :

$$32 \times (DFP + DFS + 4 \times GCF.DFA[7:0])$$

The size of the chrominance area is half that of the luminance area.

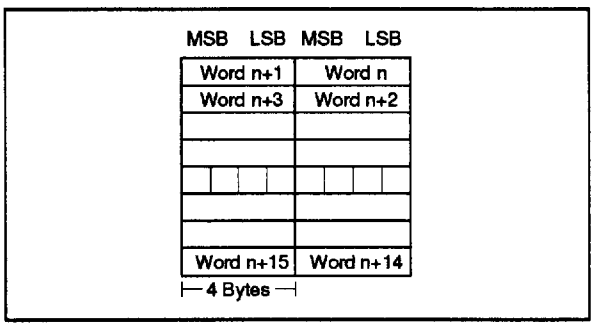
GCF.DFA[7:0] is generally set to zero when operating in normal mode. The use of this offset in 8-Mbit mode is described in section X, "8-Mbit MODE".

7929237 0072984 895

One memory page contains 512 64-bit words. One block requires 8 words. Thus one page can hold the luminance blocks of 16 macroblocks or the chrominance blocks of 32 macroblocks. In order to maximise the efficiency of memory access, macroblocks never cross page boundaries. There is an additional constraint imposed on frame buffers which are to be displayed: their base addresses must be multiples of 128 (i.e. the bottom two bits of the frame pointer register must both be zero).

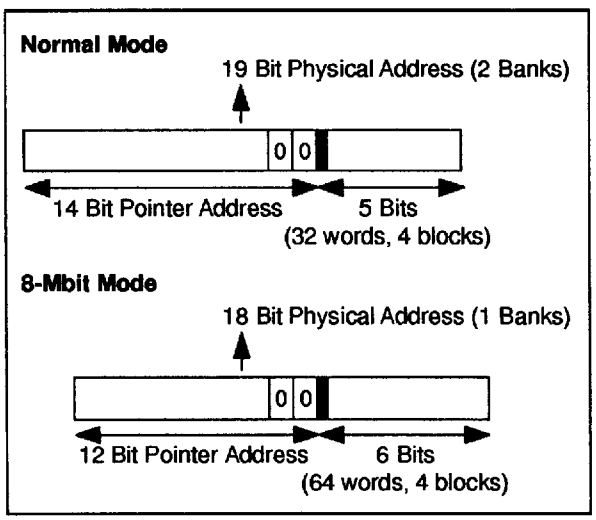
In 8-Mbit mode (in which the external memory data bus width is 32bits), a block is stored as shown in Figure 22. Each word holds 4 8-bit luminance or chrominance values. Macroblock storage order is the same as that shown in Figure 21.

Figure 22 : Physical Storage of a Block in 8Mbit Mode



From the user point of view, the frame pointers are programmed identically in the normal mode and 8-Mbit mode. In both cases, the pointer address units correspond to 4 blocks ; in normal mode, they correspond to 32 memory words, whereas in 8-Mbit mode, they correspond to 64 memory words. Figure 23 illustrates the mapping in the two cases. In both modes the bottom two bits of a frame pointer must be zero if the corresponding buffer is to be displayed. In 8-Mbit mode the top two bits of the 14-bit pointer are not used.

Figure 23 : Pointer to Physical Address Mapping



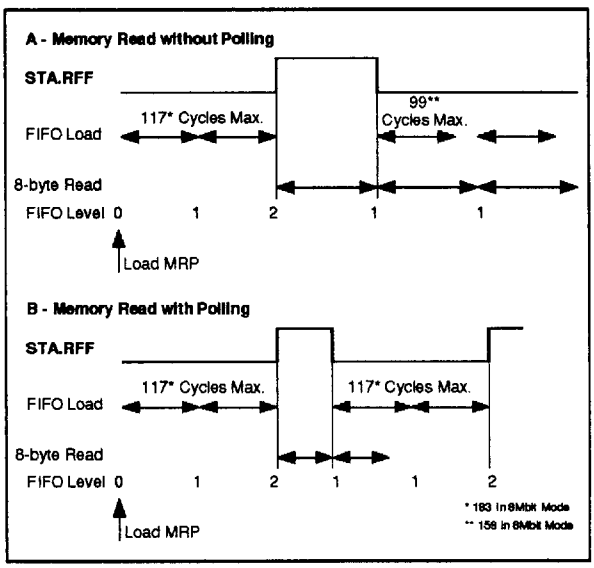
V.6 - Memory Read and Write through the Microcontroller Interface

Words can be read and written directly into the external memory into areas defined by the memory read pointer register, MRP, and the memory write pointer register, MWP, respectively.

Read

Data is transferred from the memory to the microcontroller interface through a 2-word FIFO. When the starting address of the area to be read from is loaded into the MRP register, this FIFO is cleared and 2 memory data words, starting at the address specified, are transferred into it. The status bit STA.RFF indicates when these two words are available in the FIFO. At this point the words can be read byte-by-byte (most significant byte first) from address MRF. When one word has been read, a new word is automatically loaded into the FIFO from the next memory address. After each transfer of a word from the memory to the FIFO, MRP is incremented. The maximum latency between the completion of the reading of a word and the loading of another into the FIFO from the memory is 99 primary clock cycles (1.8ms with a 55MHz clock) in normal mode and 159 clock cycles in 8-Mbit mode (2.9ms). Therefore, if the reading of a word takes longer than this, it is not necessary to check STA.RFF ; there will always be a word available in the FIFO for reading. The timings of the two modes of reading, with and without polling, are given in Figure 24.

Figure 24 : Memory Read Timing



SGS-THOMSON MICROELECTRONICS

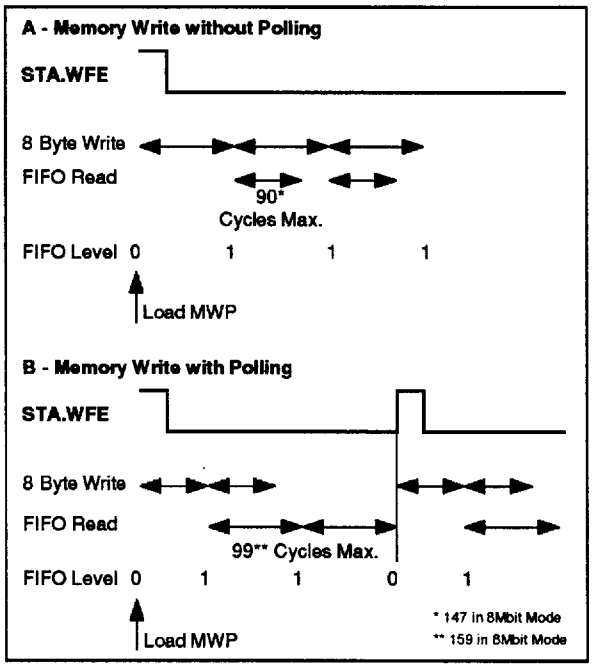
7929237 0072985 721

Write

Data in transferred from the microcontroller interface to the memory through a 2-word FIFO. When the starting address of the area to be written to is loaded into the MWP register, this FIFO is cleared. The status bit STA.WFE indicates when this FIFO is empty. At this point two words can be written byte-by-byte (most significant byte first) to address MWF. When one word has been written, a word is transferred into the memory at the address specified by MWP. After each transfer into the memory MWP is incremented. The maximum latency between the completion of writing of a word and the transfer of a word from the FIFO to the memory is 90 primary clock cycles (1.64µs with a 55MHz clock) in normal mode and 147 cycles in 8-Mbit mode (2.67µs). Therefore, if the writing of a word takes longer than this, it is not necessary to check STA.WFE; there will always be space available in the FIFO when writing.

The timings of the two modes of writing, with and without polling, are given in Figure 25.

Figure 25 : Memory Write Timing



V.7 - Block Move

The block move facility allows blocks of words to be written from one part of the memory to another without the microcontroller having to perform any data read and write operations. The procedure for executing a block move is the following :

- set bit CMD.SBM to enable block move and disable user read/write to memory. Write the base address of the block to be moved to register MRP,
- write the base address of the destination to register MWP,
- set up the BMS register with the number of words to be moved. At the end of the write cycle, the block move will start. STA.BMI will be reset,
- when bit STA.BMI is set, the block move is complete,
- reset bit CMD.SBM. User read/write is now available again.

Memory read/write and block move operations can not be performed simultaneously.

VI - RESETS AND POWER-DOWN MODES

VI.1 - Resets

There are three types of reset :

- a hard reset is generated by asserting pin $\overline{\text{RESET}}$ for a duration of at least 15 primary clock cycles,
- a soft reset is generated by setting and resetting bit CTL.SRS. It must be set for a duration of at least 40 primary clock cycles,
- a pipeline reset is generated by setting and resetting bit CTL.PRS. It must be set for a duration of at least 3 primary clock cycles.

After a hard reset, all circuit activity stops and the registers are forced into the reset states defined in section XII.2, "Register Descriptions". The circuit is put into low power mode (defined in section VI.2, "Power-Down Modes"), and the video and memory interfaces are put into a high impedance state. All data remaining in the external memory is lost. A hard reset would normally be used after power-up and when it is required to place the circuit in low power mode.

The software reset should be implemented as follows :

- configure CTL register MSB depending on application,
- in CTL register LSB enable CLK3 and software reset,
- wait for at least 40 primary clock periods,
- disable CLK3 while leaving SRS still active,
- release SRS while leaving CLK3 still disabled,
- enable CLK3 and enable decoding (CLK.EDC).

After a soft reset, all processes concerning decoding and bit buffer control are reset. Any data remaining in the bit buffer, the compressed data FIFO and the start code detector FIFO are lost. The interrupt unit is reset. All registers maintain their contents and the display process is not disturbed. A soft reset would normally be used when the decoding of the current bitstream must be terminated and it is required to restart on a new sequence.

After a hard or a soft reset, the first task performed by the pipeline when it has been enabled will always be a search for the beginning of a new sequence. The bit buffer data is flushed until the

7929237 0072986 668

first picture start code following a sequence start code is detected by the pipeline, at which time it stops. At this point normal picture decoding behaviour is resumed. After a hard or a soft reset, the first search performed by the start code detector in response to the first DSYNC will always be a search for a sequence start code, after which it stops. After this, the start code detector operates normally.

A pipeline reset terminates the decoding of the current picture. The remaining bits of the picture are flushed from the bit buffer until the next picture start code is detected by the pipeline. At this point normal behaviour is resumed, i.e. the pipeline waits for the next picture decoding instruction. No other part of the circuit is affected by a pipeline reset. A pipeline reset would normally be used as part of a manual error recovery procedure. A pipeline reset has no effect if the decoding pipeline is in its idle state.

VI.2 - Power-Down Modes

Low power mode is entered after a hard reset, or by resetting bits CTL.EVI, CTL.EDI and CTL.ECK. In low power mode, the whole circuit is shut down, and the video and memory and compressed data interfaces are disabled. In low power mode, the registers do not lose their contents (if there was no hard reset) and can be accessed normally. Power dissipation in this mode is the minimum possible.

Reduced power mode is entered by resetting bit CTL.EC2 and keeping bits CTL.EC3, CTL.ECK and CTL.EDI set. In this mode decoding is idle, but access to the external memory through the read and write FIFOs is possible. If the video interface is enabled by setting bit CTL.EVI, then the display interface may be used.

VII - BIT BUFFER AND START CODE DETECTION

VII.1 - Bit Buffer

The mechanism of writing compressed data into the bit buffer through the CD FIFO is detailed in section IV.4, "Compressed Data Input" and section IV.5, "Bit Buffer Control".

As part of the initialization sequence of the decoder, the registers BBG (bit buffer starting address), BBS (bit buffer stop address) and BBT (threshold for generation of BBF interrupt) must be set up.

The level of the bit buffer at any instant can be determined by reading the BBL register.

If it is required to change BBG before starting to decode a new sequence, this should be done before the soft reset is activated.

If the bit CTL.PBO is set (see section IV.5, "Bit Buffer Control"), then BBT defines a "full" rather than "nearly full" level and should be set equal to BBS + 1 - BBG to optimize memory usage.

VII.2 - Start Code Detection

The start code detector operates in parallel with the decoding pipeline. The purpose of this unit is to allow external access to the header data which follows start codes in the input bitstream. Compressed data is read twice from the bit buffer - once into the pipeline, and once into the start code detector through the 128-byte header FIFO. The transfer of data into the header FIFO does not affect the bit buffer level; only the data transfer into the pipeline can reduce the bit buffer level.

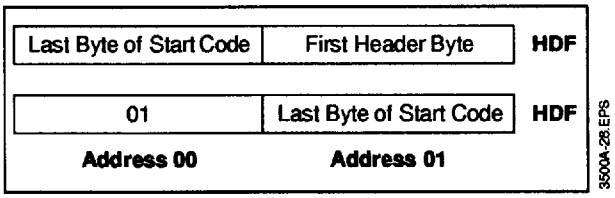
Start code detection is initiated in two ways :

- automatically whenever the internal event DSYNC occurs. DSYNC is derived from VSYNC as described in section VIII.4, "Decoding Task Control". A DSYNC is generated every time the pipeline starts a new picture decoding task,
- manually by writing to the CMD register with bit CMD.HDS set.

When start code detection has been started, data is read continuously from the bit buffer into the header FIFO and parsed by the start code detector, which receives the FIFO output data. When a start code is detected, the data scanning stops and the status bit STA.SCH becomes 1. When a start code has been detected, it can be identified by reading the HDF register.

The start code detector detects all start codes other than the slice start codes, which are "00000101h" through "000001AFh". After detection of a start code the HDF register will be one of the states shown in Figure 26.

Figure 26 : States of HDF after Start Code Detection



SGS-THOMSON MICROELECTRONICS

7929237 0072987 5T4

The first step is to examine the byte at address 00. If this contains "01", then the start code can be identified by reading the byte at address 01. If the first byte is not "01" then it must be the last byte of the start code and the second byte is the first byte of the header data. In both cases subsequent reads from HDF will give access to the header data which follows the start code.

Scanning for start codes will recommence on the next DSYNC or write to CMD.HDS. Whenever a start code has been detected, the HDF register must be read in order for the start code detector to restart correctly.

The first start code search after a hard or soft reset will be a search for a sequence header start code; all other start codes will be ignored. When this start code has been read, all subsequent searches will look for any start codes other than slice start codes.

The two status bits STA.HFE (header FIFO empty) and STA.HFF (header FIFO full) indicate the state of the header FIFO. Reading from HDF must never be performed if STA.HFE is 1. STA.HFF is set whenever the header FIFO contains at least 66 bytes.

VII.3 - Handling of Time-Stamps

The STi3500A is designed to accept only MPEG-1 or MPEG-2 video elementary streams. Time-stamps are not transmitted in this layer, but at the packet layer.

A mechanism is thus provided to enable the association of the time-stamps which are included in video packet headers with the times at which particular pictures are decoded. This is needed because the number of pictures which may be stored in the bit buffer at any instant is unknown, and therefore there is a variable delay between the input of a picture into the bit buffer and its entry into the decoding pipeline.

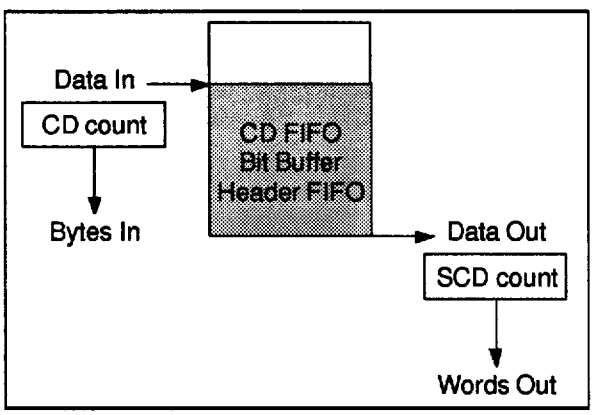
There is a 24-bit counter at the input and at the output of the CD FIFO - bit buffer - header FIFO chain, as shown in Figure 27. Each time a byte is written into the CD FIFO the counter "CDcount" is incremented. Each time a 16-bit word is read from the header FIFO the counter "SCDcount" is incremented. Both of the counters are reset by a hard reset. Both are modulo 2^{24}, i.e. the state following FFFFFF is 000000.

When the first byte of video data from a new packet containing a time-stamp is written into the CD FIFO, CDcount is read. This value is recorded by the microcontroller in a list along with the time-stamp. When a picture start code is detected by the start code detector, SCDcount is read. If this value multiplied by two is greater (modulo 2^{24}) than the last CDcount in the list, then the next picture to be decoded is associated with the time-stamp stored at this position of the list. This time-stamp and CDcount pair is now removed from the list.

Bits CDcount[23:16] are read by first writing "01" to CMD.AVS[1:0] and then reading the least significant byte of CMD. Bits CDcount[15:0] are read by first writing "00" to CMD.AVS[1:0] and then reading CMD.

Bits SCDcount[23:16] are read by first writing "11" to CMD.AVS[1:0] and then reading the least significant byte of CMD. Bits SCDcount[15:0] are read by first writing "10" to CMD.AVS[1:0] and then reading CMD.

Figure 27 : Bit Buffer Read and Write Counters



7929237 0072988 430

VIII - DECODING PIPELINE

VIII.1 - General

The pipeline is that part of the circuit which converts the compressed bitstream data for each picture into a decoded (or reconstructed) picture. These pictures can be frame or field pictures. The operation of the pipeline is controlled picture-by-picture. The decoding of a new picture can potentially start on every VSYNC, but usually the rate of decoding is less than the VSYNC rate.

The pipeline is controlled by the pipeline controller (see Block Diagram). When the pipeline controller starts the decoding pipeline a DSYNC signal is issued. This signal is also sent to the start code detector. When the pipeline has completed its decoding operation, a completion signal is sent to the pipeline controller, which is then able to launch another decoding operation, either immediately or when the next VSYNC occurs. The pipeline controller interprets certain bits of the decoding instruction, which must be set up by the user before the start of each new task. The remaining bits of the instruction define the decoding task itself.

The pipeline receives its compressed data from the bit buffer. This data is first processed by the variable length decoder (VLD) which regenerates the run/level coded DCT coefficients and the motion vectors (if present) for each macroblock. The picture data is reconstructed by passing the run/level data through the inverse quantizer and inverse DCT blocks. This is then added to the predictors which have been fetched from the memory taking into account the macroblock prediction modes and motion vectors. Finally, the decoded picture is written back into the memory, from where it can be accessed by the display unit for output.

The pipeline is also able to skip through picture data for various reasons. The different possibilities are :

- skip to Next Sequence. This occurs unconditionally on the first instruction execution after a hard or soft reset (see section VI.1, "Resets"). Compressed data is skipped until the first picture start code following a sequence start code is found. The pipeline then indicates task completion and waits for a new instruction,
- skip to Next Picture. This occurs either after a pipeline reset (see section VI.1, "Resets"), or when the decoding instruction specifies that one or two pictures should be skipped (see section VIII.4, "Decoding Task Control"). In the first case compressed data is skipped until the next picture start code is found, after which the pipeline indicates task completion and waits for a new instruction. In the second case, after the skipping operation the decoding of the following picture is started immediately,
- skip to Next Slice. This occurs after automatic error concealment (see section VIII.6, "Error Recovery and Missing Macroblock Concealment"). Compressed data is skipped until the next slice start code in the picture is found, after which normal decoding resumes.

Before starting to decode a sequence, certain static parameters must be set up. These are :

- MPEG-1 or MPEG-2 mode selection. Bit CTL.MP2 must be set for an MPEG-2 sequence, reset for an MPEG-1 sequence,
- decoded picture size. Register DFW must be set up with the picture width in macroblocks, and register DFS must be set up with the number of macroblocks in the picture.

Decoding is enabled by setting bit CTL.EDC.

VIII.2 - Quantization Table Loading

The two quantization matrices (intra and non-intra) used by the inverse quantizer must be initialized by the user. There are no built-in quantization matrices. Therefore, they must be loaded either with default matrices or with those extracted from the bitstream by the microcontroller. The quantization tables are double-buffered. This enables one or both tables to be updated without disturbing the decoding task in progress. The STi3500A maintains two bits which record whether one or both of the tables have been modified. A modified table is automatically brought into operation at the start of the next decoding operation, i.e. when the next DSYNC occurs.

After a hard reset, the same pair of tables is always selected. The data previously loaded into the tables is not affected. Other types of reset have no effect on the quantization tables.

The quantization tables are written at address QMW. Bits CMD.QMN and CMD.QMI are used to control access to the tables. The writing procedure is described in the CMD and QMW register descriptions in section XII.2, "Register Descriptions".

VIII.3 - Utilization of Picture Pointers

Before the decoding of each picture the following frame buffer pointers must be set up :

- RFP - reconstructed frame pointer.
- FFP - forward prediction frame pointer.
- BFP - backward prediction frame pointer.

(A fourth pointer, DFP, the displayed frame pointer is described in section IX.3, "Setting up the Display").

RFP defines the memory buffer to which the decoded picture is written. FFP and BFP defined the areas in memory from which the predictors are fetched. How these two latter pointers are used

SGS-THOMSON MICROELECTRONICS

7929237 0072989 377

depends on the prediction mode. The rules are given below.

Note that pictures are always stored as frames, and that to access a field (top or bottom), the starting address of the frame must be defined.

P-Frame-Picture
(Frame, Field or Dual-Prime Prediction)

FFP is set to the address of the predictor frame (in which the two predictor fields lie). BFP is not used.

B-Frame-Picture
(Frame or Field Prediction)

FFP is set to the address of the forward predictor frame (in which the two predictor fields lie). BFP set to the address of the backward predictor frame (in which the two predictor fields lie).

P-Field-Picture
(Field, 16 x 8 or Dual-Prime Prediction)

When decoding either field, FFP is set to the address of the previous decoded I or P frame. BFP is not used.

B-Field-Picture (Field or 16 x 8 Prediction)

FFP is set to the address of frame in which the two forward predictor fields lie. BFP set to the address of the frame in which the two backward predictor fields lie.

I-Pictures

For I-picture decoding, no predictors are necessary, but FFP must be set to the address of the last decoded I or P-picture for use by the automatic error concealment function.

There are additional rules for frame pointer use in 8-Mbit mode. These are given in section X, "8-Mbit MODE".

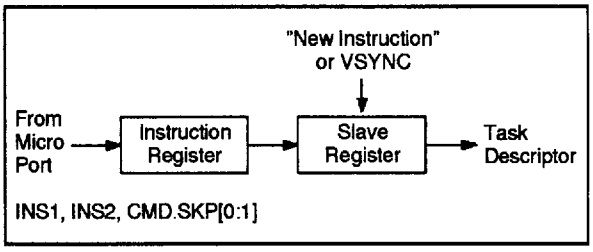
VIII.4 - Decoding Task Control

A task is a single picture decoding operation. A task is specified by the task description or instruction, which is set up before the decoding of each picture. A task commences when the internal signal DSYNC is generated. A task completes (the pipeline becomes idle) when the picture header of the following picture is detected by the pipeline and the picture is entirely reconstructed in the memory. The instruction is double buffered, so that during execution of a decoding task, the instruction for the next task can be set up by the microcontroller. When the next instruction is activated, a DSYNC can be generated, and the next decoding task started. The buffering mechanism is illustrated in Figure 28. Note that some instruction bits are latched by VSYNC, others by a signal from the pipeline controller "new instruction"*.

The Instruction is written into registers INS1 and INS2, and bits CMD.SKP[1:0]. If a new instruction is not written, the task descriptor will be the same as the previous one.

* This is only consequences when a task overruns. See section VIII.5, "Task Overrun".

Figure 28 : Instruction Buffering

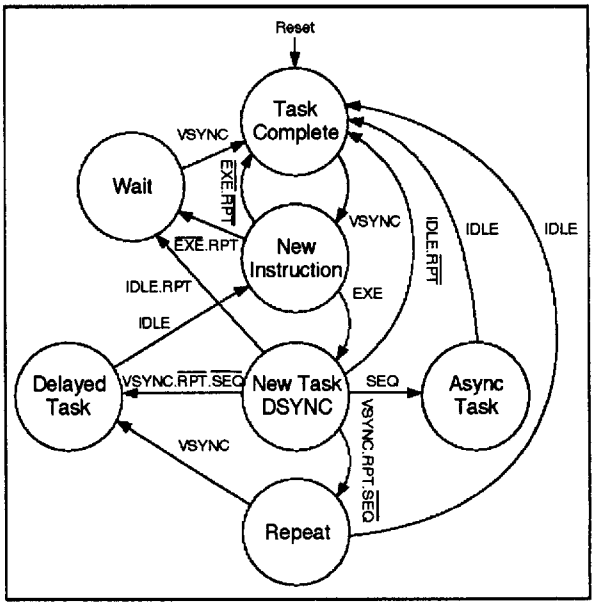


Normally, it is a VSYNC that starts the execution of a new instruction, and thus the generation of DSYNC. If however, a VSYNC occurs before task completion (i.e. before the pipeline becomes idle), the start of the next task will be delayed until the present one is completed. In this way the decoding of a picture can be allowed to extend beyond the nominal period allotted to it, usually one or two VSYNC periods.

Two status bits (and thus interrupts) are associated with pipeline control. STA.PSD indicates the occurrence of a DSYNC. STA.PID indicates that the pipeline is idle.

The operation of the pipeline controller is shown in the state diagram of Figure 29.

Figure 29 : Task Control State Diagram



7929237 0072990 099

The instruction bits which affect state transitions are INS1.EXE, INS1.RPT and INS1.SEQ. The events to which the controller responds are VSYNC, which could be a "VSYNC top" or a "VSYNC bottom" and "IDLE" representing the idle state of the pipeline.

The resting state of the controller is "task complete" which is entered after hard or soft resets or when a decoding task is completed. If a VSYNC occurs while the controller is in this state, the controller moves to state "new instruction". Here new instruction bits are loaded using the mechanism shown in Figure 28.

If the action required is "wait for one VSYNC period", i.e. do not generate a DSYNC and thus do not start the pipeline and start code detector, bits INS1.EXE and INS1.RPT must both be 0. The controller returns to state "task complete" and waits for the next VSYNC. If the action required is "wait for two VSYNC periods", the bit INS1.EXE must be 0 and bit INS1.RPT must be 1. The controller now passes through the states "new instruction" and "wait".

If the pipeline and start code detector are to be started, then bit INS1.EXE must be set. This will cause the controller to enter the state "new task" from which a DSYNC is generated. The pipeline will now execute the operation defined by the task description bits of the instruction (see later), provided that bit CTL.EDC is set. If CTL.EDC is not set, controller operation is not affected, but the task will not be executed by the pipeline and idle will remain reset.

If the time allocated to the task is one VSYNC period, INS1.RPT must be 0. The controller will remain in state "new task" until either the task completes or a VSYNC occurs. In the former case, the controller returns to state "task complete" and waits for the next VSYNC. In the latter case, the task has overrun and VSYNC arrives before IDLE. The state "delayed task" is entered. The controller remains here until IDLE occurs, when the state "new instruction" is immediately entered. Thus the next task is chained immediately to the one just completed.

If the time allocated to the task is two VSYNC periods, INS1.RPT must be 1. The controller will remain in state "new task" until the VSYNC at the end of the first period occurs. It now moves into state "repeat". The controller will remain in this state until either the task completes or the second VSYNC occurs. In the former case, the controller returns to state "task complete" and waits for the next VSYNC. In the latter case, the task has overrun and VSYNC arrives before IDLE. The state "delayed task" is entered. The controller remains here until IDLE occurs, when the state "new instruction" is immediately entered. Thus the next task is chained immediately to the one just completed.

The bit INS1.SEQ has a special function and is only used after a hard or soft reset. Before decoding is enabled after such a reset (bit CTL.EDC is set), bit INS1.SEQ and INS1.EXE must be set. On the occurrence of the reset, the pipeline starts scanning the bitstream until the first picture start code following a sequence start code is found. The controller will enter the state "async task" in which it is not responsive to VSYNCs. When the pipeline has found the start code for which it was searching, IDLE becomes true and the state "task complete" is entered. To prevent the pipeline starting a task immediately on the following VSYNC, an instruction with INS1.EXE set to 0 can be loaded while the controller is in state "async task". (Entry into this state can be detected by monitoring bit STA.PSD).

It is possible to skip one or two pictures by using the bits CMD.SKP[1:0] as defined in the CMD register description (see section XII.2, "Register Descriptions"). A skipping task consists of a skip followed by the decoding of the next picture.

The bit INS1.OVW must be set when the picture is being reconstructed into the same buffer as that from which the displayed picture is being read, i.e. RFP = DFP. Overwrite mode is explained in more detail in section XI.2, "Buffer Sequencing and Overwrite Mode".

The remaining bits of INS1 and INS2 are all set up with picture parameters derived directly from the picture header of the bitstream. These are detailed in the INS1 and INS2 descriptions (see section XII.2, "Register Descriptions").

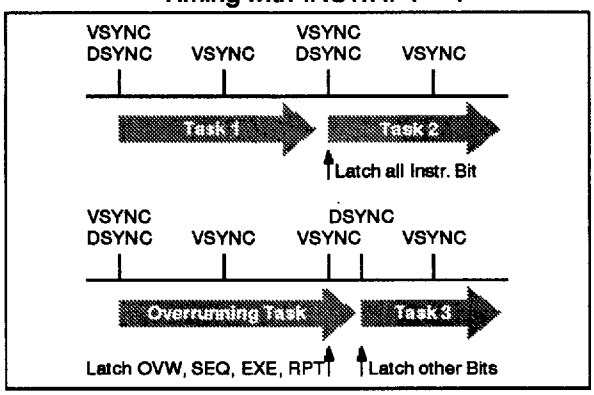
SGS-THOMSON MICROELECTRONICS

7929237 0072991 T25

VIII.5 - Task Overrun

Figure 30 illustrates the timing, with respect to VSYNC, of both normal and overrunning tasks. In the former case, the decoding task is complete before the next (or second when INS1.RPT = 1) VSYNC arrives. At this point all bits of the new instruction are updated. In the case of an overrunning task, the latching of the new instruction bits occurs in two stages; the first when the first (INS1.RPT = 0) or second (INS1.RPT = 1) VSYNC arrives, the second when the task is completed, at which point a DSYNC is also generated (if INS1.EXE = 1). The instruction must thus never be changed between the last VSYNC and the following DSYNC.

Figure 30 : Normal and Overrunning Task Timing with INS1.RPT = 1



VIII.6 - Error Recovery and Missing Macroblock Concealment

There are four levels of error detection and recovery available in the STi3500A:

- Bitstream syntax error detection with the option of automatic missing macroblock concealment.
- Bitstream semantic error detection with the option of automatic concealment or skip to the next picture.
- Pipeline severe error detection.
- User-initiated skip to next sequence using soft reset.

Syntax Error Detection and Concealment

In normal operation of the STi3500A, error concealment must always be enabled, i.e. CTL.DEC should be reset.

If the VLD detects a syntax error in the bitstream, the pipeline will copy macroblocks from the previous picture using the motion vectors reconstructed for the previous row of macroblocks in the current picture, while scanning the bitstream until a slice start code is detected. At this point normal decoding resumes. If the slice in which the error occurred was the last one in the picture, concealment will continue until the end of the picture, at which time the pipeline stops normally (assuming that the following picture start code is intact).

Concealment of macroblocks is carried out by using the vectors of the macroblock immediately above the lost macroblock. The pipeline is able to store one row of such information, for a decoded picture size of up to a maximum of 46 macroblocks*. Two vectors are stored for each macroblock in the row.

* For decoded picture widths of more than this, error concealment will be degraded.

The concealment macroblocks are accessed using the pointers FFP and BFP. Lost Macroblocks in the first row are copied directly from the previous pictures (i.e. as P-macroblocks with zero motion vectors). If an intra picture is coded with concealment motion vectors, these will be used. If not, then the concealment will be a simple copy from the previous picture using zero vectors. Even in intra pictures, the pointer FFP must be set up.

The following rules are used for the fetching of concealment macroblocks :

- **I-pictures :**
  - I-macroblocks without vectors → copy with zero motion,
  - I-Macroblocks with vectors → copy as forward predicted macroblock.
- **P-pictures :**
  - I-macroblocks → as above,
  - P- macroblocks → copy using stored vector,
  - P-field-macroblocks → copy in field mode using both vectors,
  - skipped macroblocks → copy with zero vector,
  - dual-prime macroblocks → same as for normal P-macroblock, since both full vectors are saved.
- **B-pictures :**
  - I-macroblocks → as above,
  - forward macroblocks → as above for P-macroblocks,
  - backward macroblocks → as above for P-macroblocks, but using backward vectors,
  - bidirectional macroblocks → only the forward vectors are stored, concealed as forward macroblock,
  - skipped macroblocks → copy in frame mode using the same mode and vectors as the previous macroblock.

7929237 0072992 961

If an error is detected in the bitstream before it enters the STi3500A, then an error start code can be inserted into the bitstream in order to initiate concealment. However, when doing this there are certain restrictions on the placement of the error start code in order to avoid emulation of other start codes. An Application Note is available on this topic.

Pipeline Error

A pipeline error occurs whenever the pipeline reconstructs more than 64 coefficients in a block. This condition is signalled by bit STA.PER. If bit CTL.EPR is set, a pipeline reset is automatically generated, and STA.PER is reset. If a pipeline reset is generated, the remainder of the picture will not be reconstructed; the data displayed will be that which was already in the buffer from a previously decoded picture.

Severe Error

A severe error occurs whenever the pipeline reconstructs more macroblock than are defined by the decoded picture size, DFS. This condition is signalled by bit STA.SER. Decoding is halted when this error is detected. In order to restart decoding a pipeline reset must be performed.

Soft Reset

The effect of this last resort action is described in

IX - DISPLAY FUNCTIONS

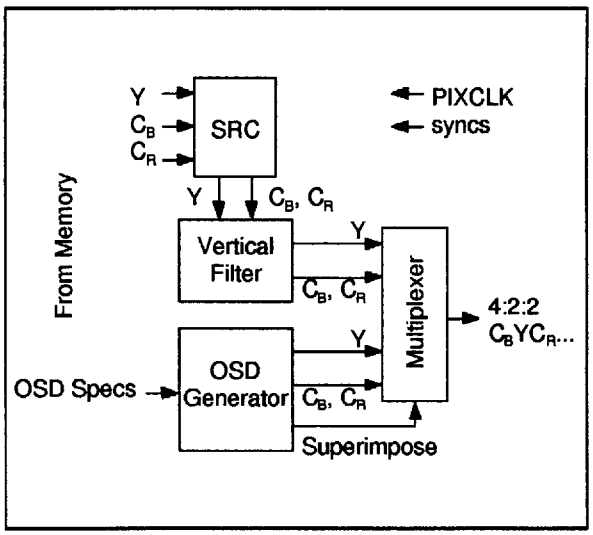
IX.1 - Operation of the Display Unit

The display unit performs the following functions :

- requests and receives from the picture buffer in external memory the decoded picture data for display,
- optionally performs horizontal resampling of both luminance and chrominance data,
- reconstructs vertical data to create 4:2:2 sample format,
- generates on-screen display bitmap for superposition onto picture output.

Figure 31 is a simplified block diagram of the display unit, showing the sequence in which the horizontal and vertical filtering operations occur and where the on-screen display (OSD) data is added.

Figure 31 : Display Unit Block Diagram



The picture data is received from the display frame buffer area of the external memory through three FIFOs (one for luminance and two for chrominance) into the luminance and chrominance horizontal sample-rate converter (SRC). For every line of luminance (Y) samples, an equivalent number of chrominance (C_B, C_R) samples are read from the memory. This chrominance line duplication is done in order to simplify the vertical filtering operation. The lines of a frame can be read out from the memory either in interlaced or line-sequential order. The integer part of the horizontal pan-scan offset set up by the user defines which pel will be the first one of a line to be read from the FIFOs.

The sample rate converter (SRC), an 8-tap filter, has two functions : upsampling of pel data when the displayed line length is greater than the decoded picture width, and implementation of the fractional part of the pan-scan horizontal offset. The outputs of the SRC are upsampled lines each having equal numbers of luminance and chrominance samples. The SRC can be bypassed if desired.

The vertical filter, used for vertical interpolation, is 2-tap filter which includes a 720-sample delay line. The filter can be applied either to chrominance or luminance data, but not both at the same time. There are four different modes for vertical reconstruction of chrominance data for interlaced pictures, and 3 modes for the display of half resolution pictures.

SGS-THOMSON MICROELECTRONICS

7929237 0072993 8T8

The coded specification for the top or bottom field OSD bitmap is received through a FIFO from the external memory into the OSD generator, which creates a bitmap. When there is bitmap data which is not defined as "transparent", the output multiplexer replaces the decoded picture with the OSD data. Four bits are available to define each of the Y, CB and CR values of an OSD region. A special OSD output value is available to define transparency. The OSD bitmap is defined with respect to the display area and is independent of decoded picture size and any pan-scan offset.

IX.2 - Video Interface

The video interface consists of the following signals :

YC7 - YC0	Video Port
PIXCLK	Pel Clock
$\overline{HSYNC}$	Horizontal Sync
B/$\overline{T}$	Bottom/top Field Selection

In response to each rising edge of PIXCLK a new 8-bit data word is available at the video port YC7-YC0. These data words are output in the 4:2:2-format sequence :

$$C_B\ Y\ C_R\ Y\ C_B\ ...$$

The first three words are the co-sited samples of the first pel, while the fourth is the luminance only sample of the second pel. The next three refer to the third pel, and so on. The first data word output in each active line period is always C_B.

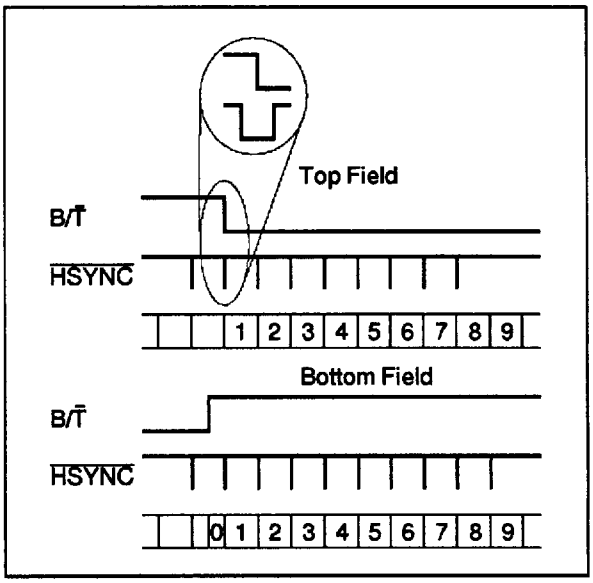
The start of each line is signalled by a falling edge of the $\overline{HSYNC}$ input. Internal pel (i.e. horizontal) counting is started by this event. The internal line counter is incremented by the rising edge of this signal. $\overline{HSYNC}$ must be low for at least 4 PIXCLK cycles, and must return to high before the end of horizontal blanking.

Vertical synchronization is derived from the B/$\overline{T}$ input. This signal is high during the bottom field and low during the top field. The internal vertical synchronization signals "VSYNC top" and "VSYNC bottom" are derived from the transitions of B/$\overline{T}$*. These signals are mapped onto the register bits STA.VST and STA.VSB, and can be used to generate interrupt requests.

* When the term "VSYNC" is used alone, it refers to "VSYNC top" or "VSYNC bottom".

The internal line counter is reset on every transition of B/$\overline{T}$, and incremented by the rising edge of $\overline{HSYNC}$. The state of the line counter at the start of the top and bottom fields is shown in Figure 32. Note that the internal line count restarts every field. In the 525/60 standard, internal line count 1 in the top field corresponds to line 4, and internal line count 1 in the bottom field corresponds to line 267. In the 625/50 standard, internal line count 1 in the top field corresponds to line 1, and internal line count 1 in the bottom field corresponds to line 314.

Figure 32 : Internal Line Numbering



At the beginning of the top field, the falling transition of the signal B/$\overline{T}$ must be sampled by the STi3500A before the rising transition of the first $\overline{HSYNC}$ of this field. The timing constraints of the video interface are given in section XIV.6, "Video Interface Timing". In most applications the value of PIXCLK will be 27MHz, in order for the output to be compatible with the CCIR 601 standard.

The video interface can be disabled in order to reduce power consumption by resetting register bit CTL.EVI. When the video interface is disabled, the outputs are put into their high impedance state and the inputs are disabled. PIXCLK is thus removed from the internal circuitry. After a hard reset, the video interface is in the disabled state.

IX.3 - Setting Up the Display

The DFP register must be set up with the base address of the buffer containing the picture to be displayed. This register is double-buffered ; when a new value is written it is taken into account on the occurrence of a VSYNC. Thus it is possible to write a new value of DFP every field, although it would normally be updated only once per frame.

7929237 0072994 734

The picture stored in the buffer is always treated as a frame by the STi3500A.

The STi3500A has two built-in ways of displaying this data :

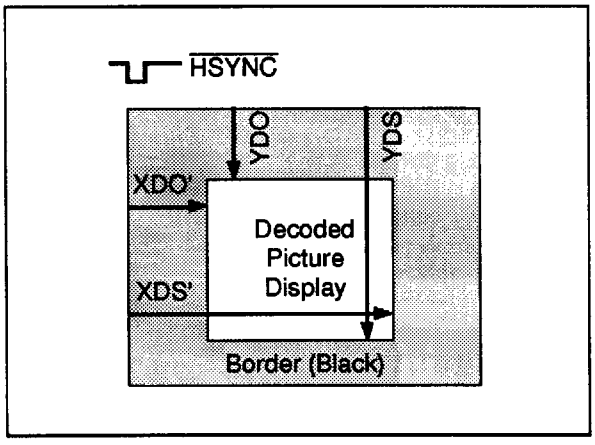
- every second line of this data is read from the buffer. The reading of one of these fields is started on every VSYNC. If B/$\overline{T}$ indicates the top field, the 1st, 3rd, 5th, etc. lines will be read. If B/$\overline{T}$ indicates the bottom field, then the 2nd, 4th, 6th, etc. lines will be read. This mode is selected by resetting bit DCF.VFC[2] (together with bit CTL.HRD if overwrite mode is enabled),
- every line of the picture buffer is read starting on every VSYNC, i.e. scan is line-sequential. This mode would normally be used when displaying a picture which was decoded at half of the resolution of the display, and where it is necessary to use the same picture data for each field. It is selected by setting bit DCF.VFC[2] (together with bit CTL.HRD if overwrite mode is enabled).

To use these standard modes, bit DCF.USR must be reset. Other, non-standard, modes of controlling the display sequence are described in section IX.7, "Displayed Field Sequence Control".

If at any time no display is required, bit DCF.EVD may be reset, in which case a constant black value ($Y = 16$, $C_B = C_R = 128$) is output.

The size and location of the display window is defined by the registers XDO, XDS, YDO and YDS. The values loaded into these registers define the horizontal and vertical boundaries of the displayed picture, as shown in Figure 33. Outside of the picture area a black ($Y = 16$, $C_B = C_R = 128$) border is generated.

Figure 33 : Display Window Positioning



Register YDO is loaded with the number of the last line of the upper border, where lines are numbered in fields as shown in Figure 32. The first active line is therefore defined by :

$$\text{First active line} = YDO + 1$$

The same YDO value serves for both fields ; the uppermost line of the picture display will be in the top field.

Register YDS is loaded with a number defining the last line of the picture display in a field, according to the relation :

$$\text{Last active line} = YDO + (\text{vertical size}/2) = YDS + 129$$

For example, with a 525/60 display, in which the vertical size of the decoded picture is 480 lines, typical values of YDO and YDS could be :

$$YDO = 21$$
$$YDS = YDO + 240 - 129 = 132$$

and with a 625/50 display, in which the vertical size of the decoded picture is 576 lines, typical values of YDO and YDS could be :

$$YDO = 22$$
$$YDS = YDO + 288 - 129 = 181$$

Register XDO defines the number of PIXCLK cycles between the falling edge of the signal HSYNC and the beginning of the picture display, according to the relation :

$$\text{Cycles from } \overline{\text{HSYNC}} \text{ to start of picture} = 2XDO + 40$$

The CCIR 601 standard defines this number to be 264 27MHz clock cycles for a 625/50 display, and 244 for a 525/60 display. The respective values of XDO are thus 112 and 102.

XDO must never be less than :

$$(177 \times f_{pel})/(2 \times f_{primary})$$

where $f_{primary}$ is the STi3500A primary clock frequency and f_{pel} is the pel clock frequency.

If $f_{primary} = 55MHz$ and $f_{pel} = 27MHz$, the minimum value of XDO is 44.

The first picture data in a line, output in the 2XDO + 41st PIXCLK cycle after the falling edge of $\overline{\text{HSYNC}}$, is always a C_B component. Since the external video generation circuitry will usually relate its Y/C phasing to the horizontal synchronization signal, and has no knowledge of the value of XDO, not all values of horizontal offset will be useable ; some will cause incorrect interpretation of the color difference components. In any given system, XDO values will have to be either always odd or always even.

SGS-THOMSON MICROELECTRONICS

7929237 0072995 670

XDS is loaded with a number defining the last active sample in each line, counted in units of PIXCLK cycles from the falling edge of the signal HSYNC, according to the relation :

Last sample of active video = 2XDS + 28

Thus, if L is the number of pels per line of the displayed picture, then :

2XDO + 40 + 2L = 2XDS + 28, and thus
XDS = XDO + L + 6

If L = 720, then XDS = XDO + 726.

The resolution to which the horizontal offset and end values can be defined is equal to two cycles of PIXCLK. In order to position the display window horizontally to a finer precision the DCF.PXD bit is used. When this bit is set, the active video is delayed by one PIXCLK cycle. Since the first active video sample is C_B, the Y/C phasing with respect to the horizontal synchronization signal will change.

IX.4 - Sample Rate Converter and Pan Vector

IX.4.1 - Sample Rate Converter

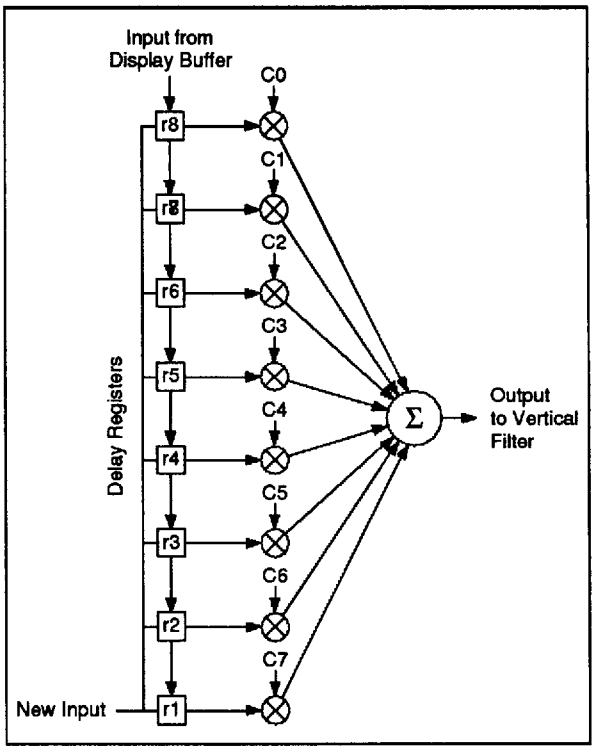
The purpose of the sample rate converter (SRC) is allow upsampling of picture data in order to increase the number of horizontal samples in a line. This is necessary if the horizontal size of the display is greater than the decoded picture width. For example if it is required to display a 720-pel wide 16:9 source image on a 4:3 display also of 720-pel width, then 540 pels selected each source line must be upsampled to 720.

To enable the SRC, bit DCF.DSR must be reset. If this bit is set, the SRC is bypassed and the horizontal resolution of the decoded picture is not changed.

The SRC operates by interpolating samples between those of the decoded picture data read from the display buffer. This is performed by an 8-tap interpolation filter with the structure shown in Figure 34. The filter has three sets of delay registers multiplexed between the Y, CB and CR samples. It has 8 sets of coefficients, each set defining one of 8 sub-pel interpolation positions. For example, for sub-pel position 0, the output is aligned with stored sample "r4", for sub-pel position 1, the output corresponds to an interpolated pel position one eighth of the distance from sample "r4" to sample "r5", and so on.

The number of inputs clocked into the SRC is equal to the number of samples used in each line of the source image, and the number of outputs generated is equal to the number of samples displayed. Thus the rate of generation of outputs will be greater than the input data rate.

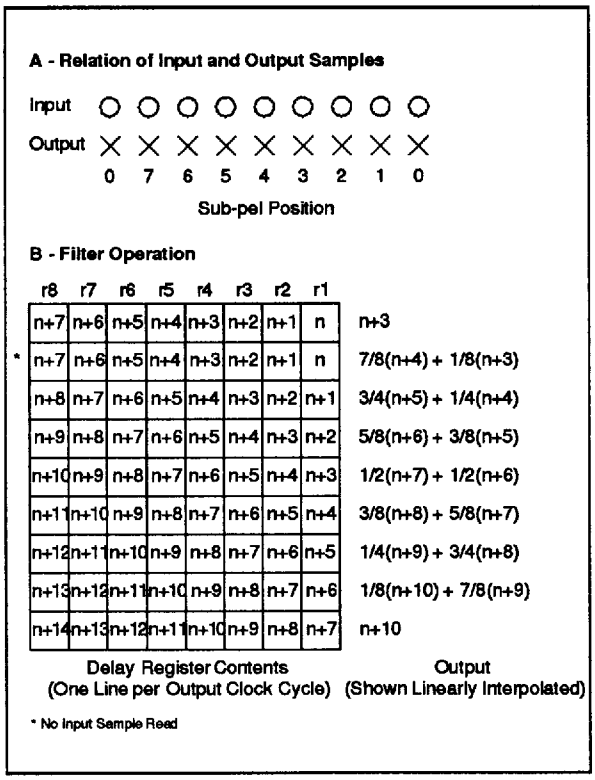
Figure 34 : 8-Tap Interpolation Filter



7929237 0072996 507

The example in Figure 35 illustrates the operation of the sample rate converter when the upsampling ratio is 8:7. For every 8 samples clocked out of the filters, 7 samples are clocked in. To illustrate the interpolation positions, at the right of the figure are shown the outputs which would occur with a simple linear interpolation (i.e.a 2-tap filter). The actual SRC output values are the 8-tap filter outputs with coefficients appropriate to sub-pel positions 0, 7, 6, 5, 4, 3, 2, 1, 0 etc.

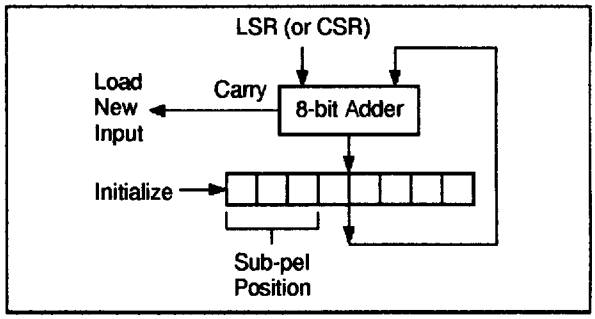
Figure 35 : SRC Example for 8:7 Upsampling



The SRC output is limited to lie within the range [1,254], i.e. the codes 00h and FFh are never output, giving compatibility with CCIR 656.

The SRC upsampling factor is set up in the LSR and CSR registers, which hold the factors for the luminance and chrominance components respectively. The same value must be loaded into both of these registers. The upsampling factor is equal to 256/LSR (or 256/CSR). This value is used to determine both the rate of input of data into the filters and the sequence of sub-pel interpolation positions. The mechanism by which this is achieved is shown in Figure 36.

Figure 36 : Upsampling Filter Control



The LSR value is added into an accumulator register at a rate equal to the filter output rate. A new input is loaded into the filter at beginning of a line and whenever a carry is generated by the adder. The top three bits of the accumulator register are used to select the sub-pel position. For example, with an upsampling factor of 8:7, the LSR value is (256/8) x 7 = 224. The sequence of values in the accumulator register will be (assuming that it is initialized to zero) :

Acc. Register	New Input	Sub-pel Position
0	YES	0
224	NO	7
192	YES	6
160	YES	5
128	YES	4
96	YES	3
64	YES	2
32	YES	1
0	YES	0

The LSR/CSR value thus defines a cycle of sub-pel positions as well as the rate of data input. If a value of less than 32 is loaded into LSR/CSR, i.e. an upsampling ration of greater than 8 is defined, there could be repeated values in the filter output. This may cause unacceptable display artifacts.

At the start of a line, the 3 sets of delay registers r1, r2 and r3 are loaded with the black value (Y = 16, $C_B = C_R = 128$). The first output is thus derived from the inputs stored in registers r4 to r8. At the end of a line, the last eight input samples are stored in registers r1 to r8. The last valid interpolation is between the samples stored in r4 and r5. Correct Interpolation is not possible beyond this except in the case where the next output is in sub-pel position 0. This output is valid since coefficient C0 is zero

SGS-THOMSON MICROELECTRONICS

7929237 0072997 443

for this position and the invalid sample beyond the end of the line is ignored. There is thus no valid interpolation possible between the last four input samples. This is illustrated in Figure 37 in which 544 pels are upsampled to 721, in which the upsampling ratio is 4:3. The LSR and CSR registers would be loaded with the value 192.

The number of valid outputs generated can be calculated as follows :

The ratio of the distances between input and output samples is 256:LSR. Given that the last output sample cannot occupy a position beyond the fourth-last input sample, the following inequality is always true :

$$LSR(N-1) \leq 256(M-4)$$

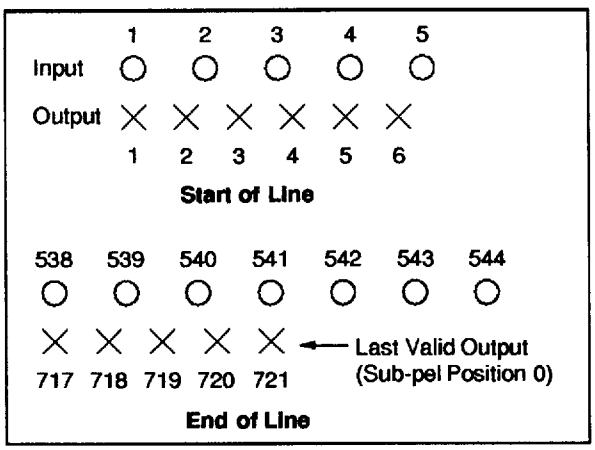
where N is the number of output samples and M is the number of input samples. The value of N is thus given by :

$$N = \lfloor 256(M-4)/LSR + 1 \rfloor$$

where "$\lfloor x \rfloor$" indicates the integer part of x.

The value programmed into the XDS register must be such that all samples beyond the last valid one are masked.

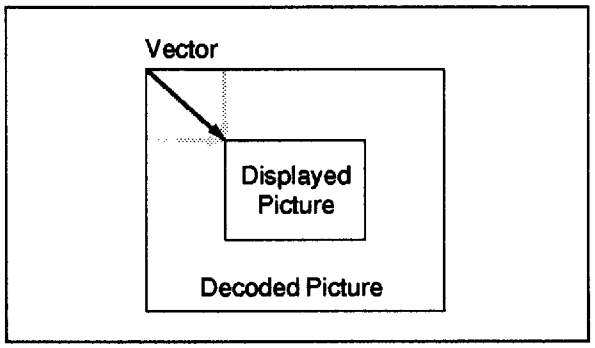
Figure 37 : Upsampling from 544 to 720



IX.4.2 - Pan/Scan Vectors

When the display window has a smaller horizontal dimension than the decoded picture, a vector can be programmed in order to define the starting point of the displayed picture, as shown in Figure 38. This vector defines the point in the decoded picture which corresponds to the top-left-hand corner of the displayed picture. The displayed picture size and location is defined by the numbers programmed in registers XDO, XDS, YDO and YDS.

Figure 38 : Pan/Scan Vector

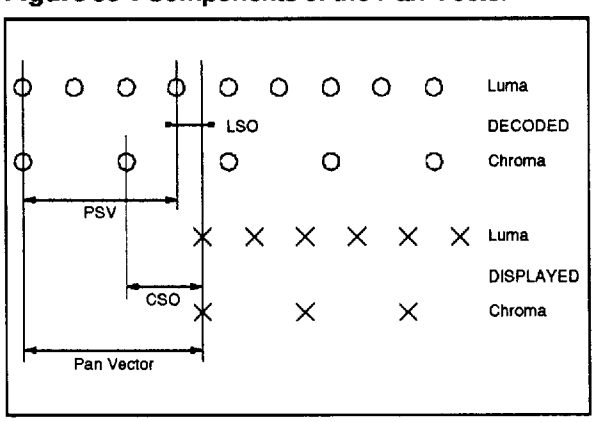


The pan/scan vector is programmed into the PSV register. This register is double-buffered ; when a new value is written it is taken into account on the occurrence of a VSYNC. Thus it is possible to write a new value of PSV every field.

The integer part of the horizontal component of the pan/scan vector is loaded into PSV.H[8:0], and the fractional part defines the contents of the LSO and CSO registers. The relationship between these quantities is illustrated in Figure 39. The numbers loaded into the LSO and CSO registers are used to initialize the luminance and chrominance upsampling control registers (see Figure 36) at the start of every line. LSO is set up directly with the value of the fractional part of the pan/scan vector horizontal component. CSO is set up with half of this number, plus 128 if the integer part is an odd number. The resolution to which the horizontal component can be defined is 1/8 pel.

The vertical component of the pan/scan vector is programmed into PSV.V[7:1], in units of 2 field lines, or 4 frame lines. For example, if PSV.V[7:1] is set to the value 16, then the 32 top lines of every field would not be output to the display.

Figure 39 : Components of the Pan Vector



7929237 0072998 38T

IX.5 - Vertical Filter

The vertical filter performs the reconstruction of the 4:2:2 data output from the decoded (and possibly upsampled) picture data which is stored in 4:2:0 format.

The vertical filter has six modes for the handling of different methods of chrominance reconstruction and for the display of half-resolution pictures. The vertical filter mode is selected by the programming of bits DCF.VFC[2:0].

In order to explain the modes, the following line numbering system will be used :

- For an interlaced frame, the fields stored in the memory have the sampling patterns and line numbers shown in Figure 40.
- For a line-sequential frame, the sampling pattern and line numbering is as shown in Figure 41.

These patterns are preserved by the SRC and are thus the patterns entering the vertical filter. For every luminance line output from the memory, a chrominance line is also output. The sequence in which chrominance lines are output is determined by the vertical filter mode.

Figure 40 : Field Sampling Patterns

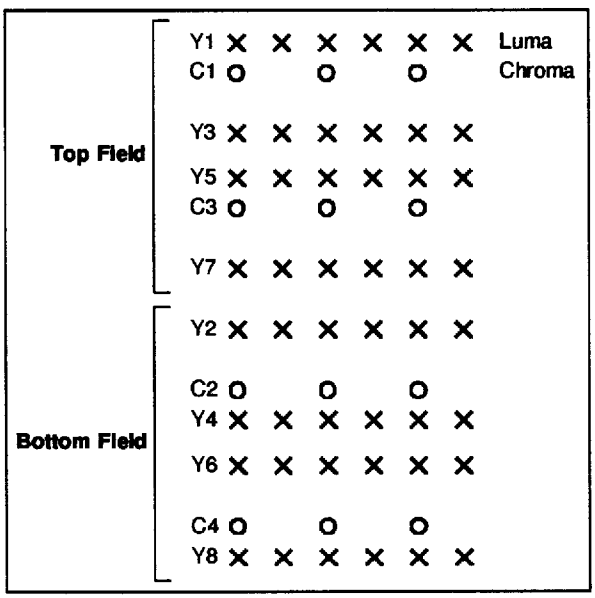
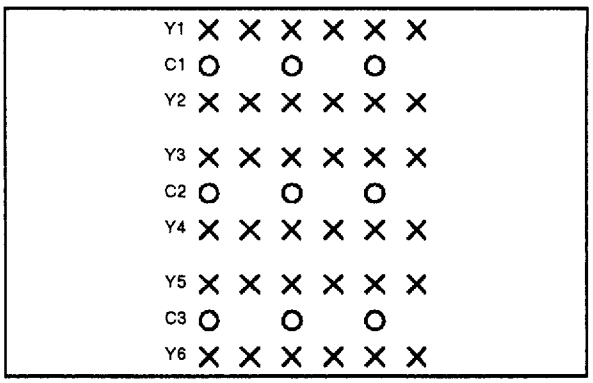


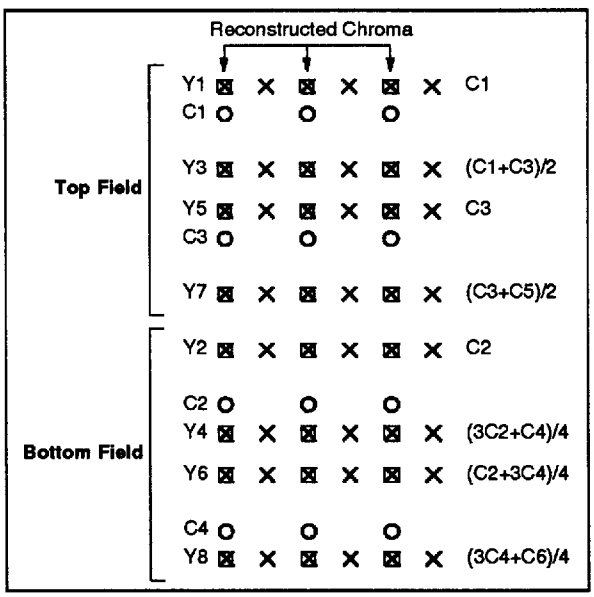
Figure 41 : Frame Sampling Pattern



Mode 0 : Full Resolution, Chrominance Line Repeat with Interpolation

This mode is used when the MPEG-2 picture-layer bitstream variable "progressive_frame" is 0, indicating that the chrominance samples were derived for each field independently, and when interpolation is required. Figure 42 shows how the chrominance is reconstructed in the two fields for 4:2:2 output. The luminance is unaffected by the vertical filter.

Figure 42 : Derivation of Chrominance in Mode 0



SGS-THOMSON MICROELECTRONICS

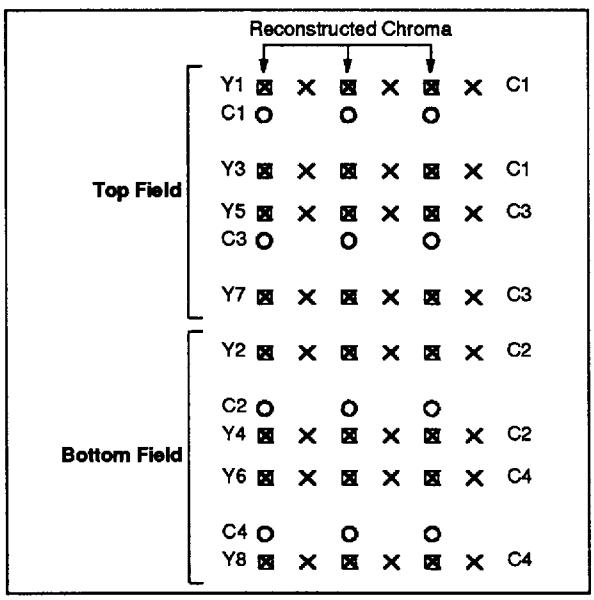
7929237 0072999 216

The chrominance data output with the last line of the top field (line 479 in a 480-line frame, or line 575 in a 576-line frame) is the same as that output with the previous line (line 477 or line 573). The chrominance data output with the last line of the bottom field (line 480 or 576) is the same as the last line of chrominance input.

Mode 1 : Full Resolution, Chrominance Line Repeat

This mode is used when the MPEG-2 picture-layer bitstream variable "progressive_frame" is 0, indicating that the chrominance samples were derived for each field independently, and when interpolation is not required. Figure 43 shows how the chrominance is reconstructed in the two fields for 4:2:2 output. The luminance is unaffected by the vertical filter.

Figure 43 : Derivation of Chrominance in Mode 1

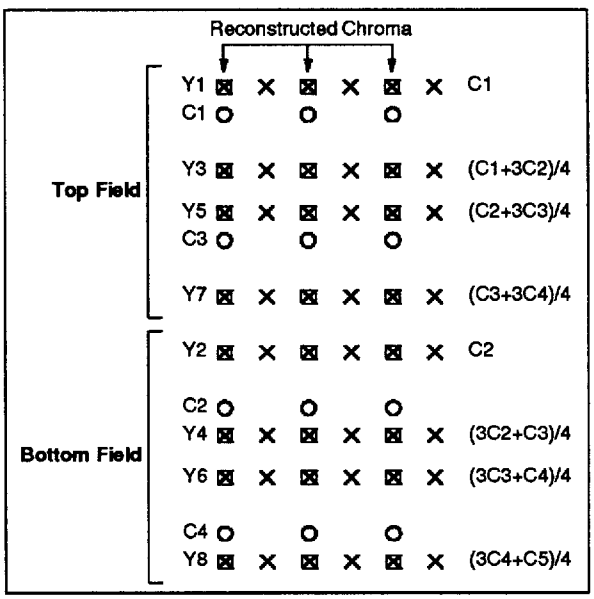


Mode 2 : Full Resolution, Chrominance Field Repeat with Interpolation

This mode is used when the MPEG-2 picture-layer bitstream variable "progressive_frame" is 1, indicating that the chrominance samples in each field were derived from both fields, and when interpolation is required. Figure 44 shows how the chrominance is reconstructed in the two fields for 4:2:2 output. The luminance is unaffected by the vertical filter.

The chrominance for the last line of the bottom field (line 480 in a 480-line frame, or line 576 in a 576-line frame) is equal to the last line of chrominance input.

Figure 44 : Derivation of Chrominance in Mode 2

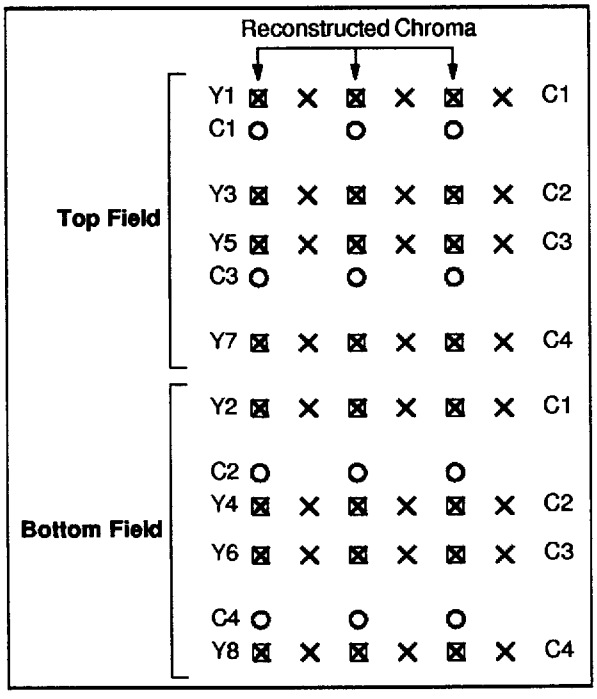


7929237 0073000 636

Mode 3 : Full Resolution, Chrominance Field Repeat

This mode is used when the MPEG-2 picture-layer bitstream variable "progressive_frame" is 1, indicating that the chrominance samples in each field were derived from both fields, and when interpolation is not required. Figure 45 shows how the chrominance is reconstructed in the two fields for 4:2:2 output. The luminance is unaffected by the vertical filter.

Figure 45 : Derivation of Chrominance in Mode 3

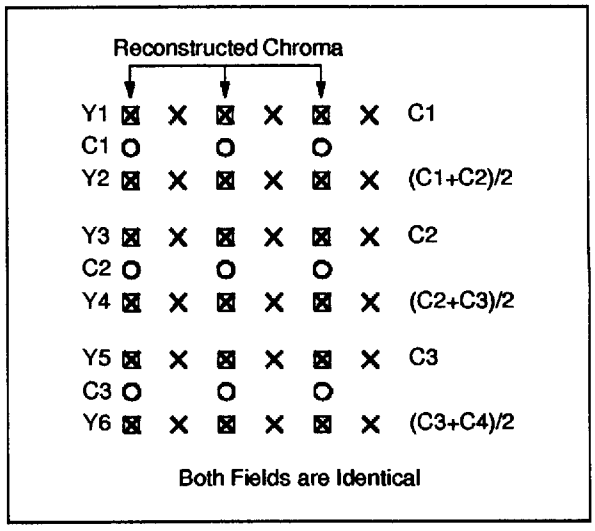


Mode 4 : Half Resolution with Chrominance Interpolation

This mode is for the display of pictures having only half the resolution of the display, when the same picture is displayed in both fields, and when interpolation of chrominance is required. The most common application is the display of MPEG-1 SIF pictures on a CCIR 601 resolution interlaced display. Figure 46 shows how the chrominance is constructed for each identical field. The luminance is unaffected by the vertical filter.

The chrominance output in last line of each field is equal to that output in the second-last line.

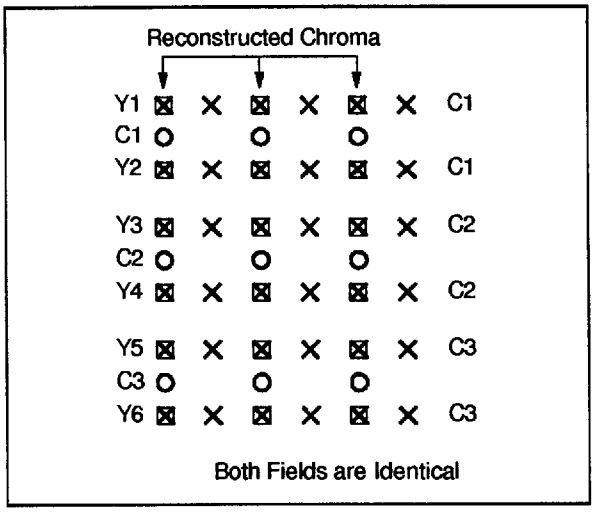
Figure 46 : Derivation of Chrominance in Mode 4



Mode 5 : Half Resolution, Chrominance Repeat

This mode is for the display of pictures having only half the resolution of the display, when the same picture is displayed in both fields, and when interpolation of chrominance is not required. The most common application is the display of MPEG-1 SIF pictures on a CCIR 601 resolution interlaced display. Figure 47 shows how the chrominance is constructed for each identical field. The luminance is unaffected by the vertical filter.

Figure 47 : Derivation of Chrominance in Mode 5



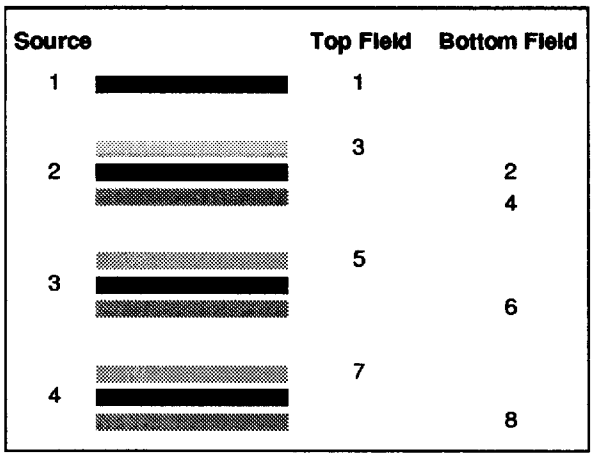
SGS-THOMSON MICROELECTRONICS

7929237 0073001 572

Mode 6 : Half Resolution with Luminance Interpolation

This mode is for the display of pictures having only half the resolution of the display, and when it is required to interpolate two different fields from the luminance lines of the source picture. The most common application is the display of MPEG-1 SIF pictures on a CCIR 601 resolution interlaced display. Figure 48 shows how the two luminance fields are interpolated.

Figure 48 : Luminance Interpolation in Mode 6



3500A-50.EPS

The interpolation of the luminance lines is performed as follows, where the subscript "s" indicates the source picture :

Top Field	Bottom Field
$Y1 = Y1_s$	$Y2 = Y2_s$
$Y3 = (Y1_s + 3Y2_s)/4$	$Y4 = (3Y2_s + Y3_s)/4$
$Y5 = (Y2_s + 3Y3_s)/4$	$Y6 = (3Y3_s + Y4_s)/4$
...	...
$Y2N-1 = (YN-1_s + 3YN_s)/4$	$Y2N = YN_s$

The output chrominance is constructed by duplication. The first line of chrominance is used for lines 1, 2, 3 and 4, the second for line 5, 6, 7 and 8, and so on, as in mode 5.

Vertical Filter Precision

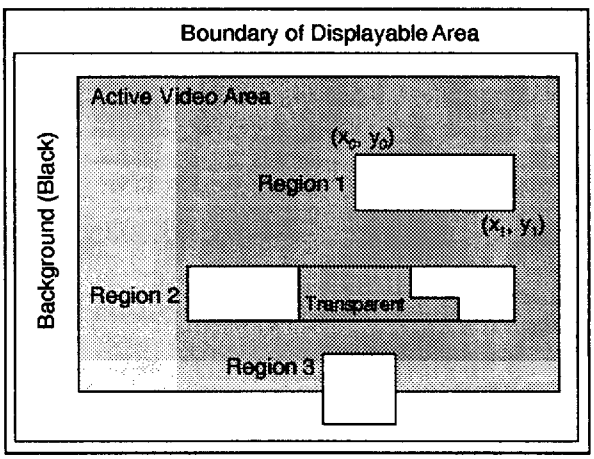
The vertical filter calculation is performed in unsigned arithmetic with full precision and then the 10-bit results are rounded to 8 bits for output. Different rounding rules are used for luminance and chrominance. The rules are as follows :

- for luminance, the results are rounded towards zero, i.e. if the bottom two bits are 00, 01 or 10, the 10-bit number is truncated to 8, while if the bottom two bits are 11, one is added to the truncated 8-bit number,
- for chrominance, the results are rounded towards 128, i.e. if the 10-bit number is larger than 1000000000_2, the rule above is applied, while if the number is less than this, 10_2 is added to the 10-bit number before truncation.

IX.6 - On-Screen Display (OSD)

The OSD function can be used to display a user-defined bitmap over any part of the displayable (i.e. non-blanked) screen, independent of the size and location of the active video area (defined by XDO, XDS, YDO, YDS). This bitmap can be defined independently for each field, and is specified as collections of "OSD regions". A region is a rectangular area specified by its boundaries and by a bitmap defining its contents. Each region has associated with it a palette defining four colors which can be used within that region. If required, one of these colors can be "transparent", allowing the background to show through. Figure 49 shows examples of OSD regions.

Figure 49 : OSD Regions



3500A-51.EPS

OSD is enabled if bit DCF.EOS is set.

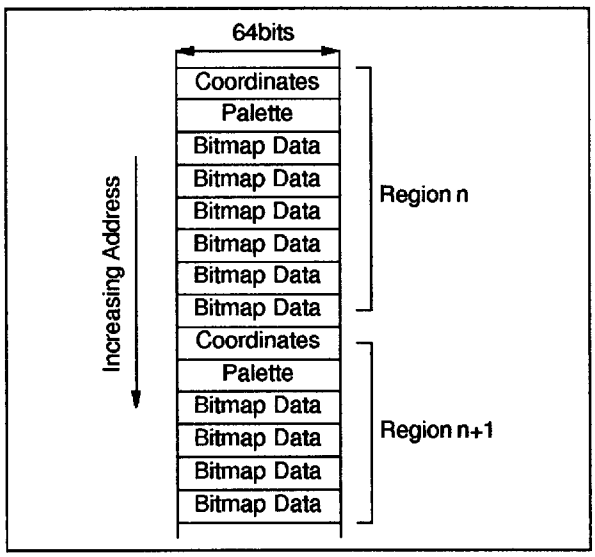
The starting address in memory of the OSD specification for the top field is defined by register OTP, and that for the bottom field is defined by register OBP. OSD specifications are written into the memory using the procedure described in section V.6, "Memory Read and Write through the Microcontroller Interface". OSD specifications can be rapidly moved in memory using the procedure described in section V.7, "Block Move".

7929237 0073002 409

Figure 50 shows how OSD specifications are stored in memory as a sequence of words. The first word defines the boundaries of the region, the second word defines the palette for the region, and the subsequent words define the color of every pel in the region. There must not be unused words between the specifications of any two regions within an OSD specification. A display line cannot be included in more than one OSD region ; in other words, only one OSD region can be active on a line*. Within an OSD specification the region specifications must be stored in order of increasing starting line number. The last word in an OSD specification must define a starting line which is beyond the displayable area. Line numbers are the internal (field) line numbers defined in Figure 32. It is thus possible to share the same OSD specification for both fields of a frame. In this case the OTP and OBP registers would be loaded with the same address. The number of OSD specifications which may be resident in memory at any time is limited only by the amount of memory available.

* This limitation can be partially overcome by the use of tranparency within an OSD region.

Figure 50 : OSD Buffer Format



The format of the OSD region specification words is given in Figure 51.

In the first word, the positions of the left and right edge samples of an OSD region, counted in numbers of PIXCLK cycles from the falling edge of HSYNC, are defined as follows :

$$\text{Left edge position} = 2X_{left} + 9$$
$$\text{Right edge position} = 2X_{right} + 10$$

where X_{left} and X_{right} are the values defined in the first word of the OSD region specification. This is illustrated in Figure 52. X_{left} must always have the same parity as the offset loaded into the XDO register (i.e. both must be even or both must be odd). These constraints ensure that the OSD region data samples are always correctly phased with respect to the active video. The first sample output in an OSD region is always a C_B value.

The top line specified in the first word of an OSD region specification must be greater than or equal to 17.

The second word defines the palette which applies to the OSD region. Four colors can be defined, each chosen from a set of 4096. The 4 bits of Y, C_R and C_B output from the palette define the top 4 bits of an output sample. The bottom four are set to zero. The "color" $Y = C_R = C_B = 0$, in palette index 0 only, defines transparency.

Figure 52 : OSD Region Horizontal Positioning

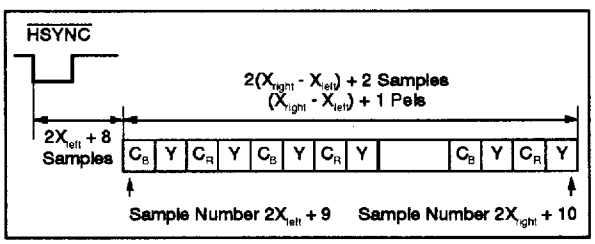
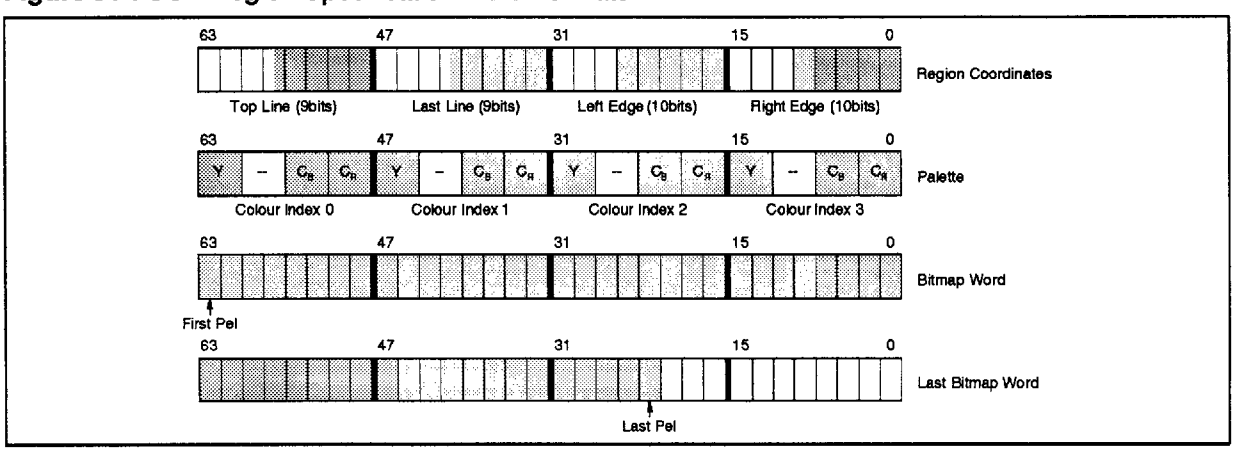


Figure 51 : OSD Region Specification Word Formats



SGS-THOMSON MICROELECTRONICS

7929237 0073003 345

The following words define the bitmap in left to right and top to bottom order. Two bits are used to define the color index (i.e. palette address) of each pel. The first, third, fifth, etc. bitmap bit-pairs are used to reference all three components (Y, C_B and C_R) of the respective pels ; the second, fourth, sixth, etc. reference only the luminance components of the palette*. For this reason the bitmap for a region must define horizontal segments containing a whole number of pel-pairs.

For the same reason, the transition to and from transparency must only occur at points which are an even number of pels from the start of the left-hand edge of an OSD region.

There is an additional constraint in the fact that a horizontal segment must contain a number of pixels which is a multiple of 4 (i.e. $X_{right} - X_{left} + 1$ must be a multiple of 4).

The following table shows the 4-bit OSD palette Y, C_B, C_R values nearest to the standard "color bar" colors.

* In the 4:2:2 format chrominance is only defined for every second pel.

	Y/16	C_B/16	C_R/16
White	1111	1000	1000
Black	0001	1000	1000
Red	0101	0110	1111
Green	1001	0011	0010
Blue	0011	1111	0111
Yellow	1101	0001	1001
Cyan	1011	1010	0001
Magenta	0111	1101	1110

IX.7 - Displayed Field Sequence Control

This function is performed automatically, unless specifically disabled by the user. By default the display field output sequence is governed by the input signal B/$\overline{T}$; the bottom field is output when it is high, and the top field is output when it is low. When bit DCF.USR is set, the user has direct control over which field is to displayed after the occurrence of VSYNC ; the built-in modes are disabled. For example in an interlaced sequence it is possible to program the same field to be displayed twice if desired.

Control over the order in which picture data is read from the picture buffer is given by bits DCF.FLD and DCF.DAM[2:0]. These bits must not be programmed independently of bits DCF.VFC[2:0], which define the vertical filter mode. The allowed combinations are detailed below. The DCF register is double buffered, and new values written are taken into account on the occurrence of a VSYNC. The values of bits DCF.FLD and DCF.DAM[2:0] are not taken in account when bit DCF.USR is reset.

The value of bit DCF.FLD controls which field is to be displayed : 1 for the bottom field, 0 for the top field. The signal B/$\overline{T}$ is thus overridden.

The allowed combinations are given in the following table :

DCF.VFC[2:0]	Field	DCF.FLD	DCF.DAM[2:0]
0	Top	0	6
	Bottom	1	6
1	Top	0	2
	Bottom	1	2
2	Top	0	3
	Bottom	1	7
3	Top	0	3
	Bottom	1	3
4	Both	0	4
5	Both	0	0
6	Top	0	0
	Bottom	1	0

These values must be updated every field.

These bits do not affect the display of OSD. OSD field selection is under the exclusive control of the B/$\overline{T}$ pin.

An application of user display sequence control is when the displayed picture must be frozen. In this case the same field can be output continuously in order to reduce flicker.

X - 8-Mbit MODE

X.1 - General

8-Mbit mode can be used for decoding smaller (up to approximately half-CCIR 601) pictures without any restrictions on prediction modes, or for decoding larger pictures with certain restrictions on prediction modes and motion vector range.

The setting up of 8-Mbit mode is explained in section V.1, "Memory Interface", and the data storage structure in memory is described in section V.5, "Picture Storage Data Structure".

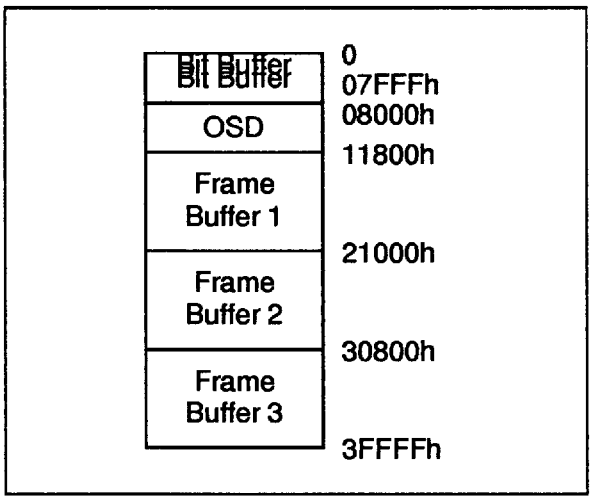
The physical memory address range is the same as that in normal (16-Mbit) mode, i.e. 00000h to 3FFFFh, but each memory word contains 32 bits instead of 64 bits.

7929237 0073004 281

X.2 - Unrestricted Decoding

Decoding can be performed in 8Mbits without any restriction on decoding modes, provided that all of the required picture buffers, the bit buffer and any OSD area required can be fitted into the 8Mbits of memory available. Figure 53 show how 3 frames of size 352 x 480, together with a 1Mbit bit buffer and an OSD buffer area can be placed in 8Mbits of memory. The frame buffer starting addresses are constrained to lie on 4 x 64 = 256 word boundaries (in normal mode, frame buffers lie on 4 x 32 = 128 word boundaries). The pointer values for the frame buffers are 1120, 2112 and 3104.

Figure 53 : 352 x 480 Frames in 8Mbits



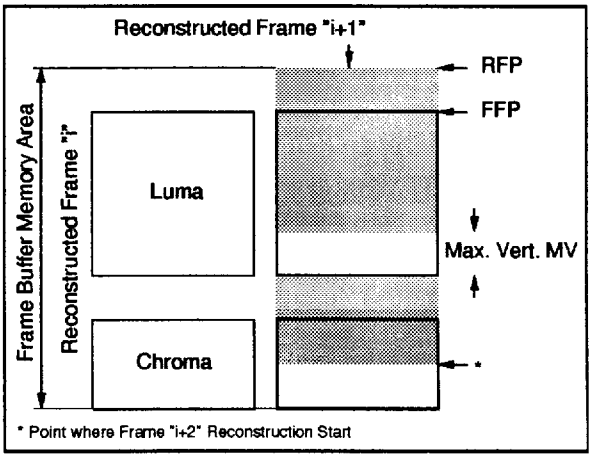
The BFP/FBP register is loaded with the value of the backward prediction frame pointer. The fold-back function is never activated since memory addressing never passes the end of memory, 3FFFh.

X.3 - Memory Folding Principle

A memory folding option is available in 8-Mbit mode. This allows the decoding of forward-predicted pictures of size too large to allow the storage of 2 frames and the bit buffer in 8 Mbits of memory. This folding is invoked automatically in 8-Mbit mode whenever memory addressing passes the end of memory.

Memory is conserved by overwriting the last reconstructed picture with the one currently being decoded, and rotating memory addresses around a predefined frame buffer area. This is illustrated in Figure 54.

Figure 54 : Memory Reuse in 8-Mbit mode



Reconstructed frame "i+1" is written into an area of memory starting at a lower address than the start of the reconstructed picture "i". Note that there must be a gap between the luminance and chrominance storage areas to prevent overwriting of the end of the luminance area of picture "n" with chrominance of picture "n+1". This gap is equal to the difference in the starting addresses of the two frame buffers. The size of this gap also defines the maximum negative vertical motion vector possible.

The size of the gap, expressed in macroblocks, allows the value of GCF.DFA[7:0] to be calculated. An example calculation is given in section XII.2, "Register Descriptions".

In addition, the register BFP/FPB needs to be set up with the start address of the frame buffer area. This is the address to which fold-back occurs when memory addresses exceed the 8Mbit limit. When folding is invoked, BFP must not be used ; backward prediction is thus not allowed.

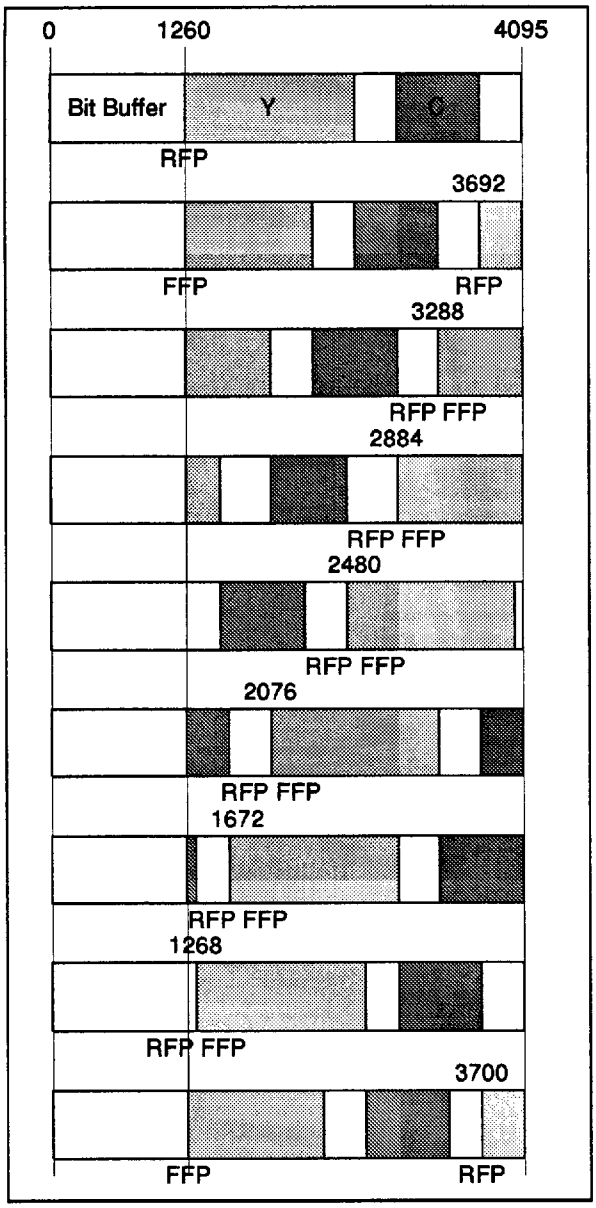
The amount of memory required for the frame buffer area is equal to the size of one frame buffer plus twice the gap defined by DFA.

SGS-THOMSON MICROELECTRONICS

7929237 0073005 118

The example (Figure 55) below illustrates the evolution of memory usage during the decoding of a sequence of pictures of size 720 x 480. The motion vector range is -128 to 127, and the gap required between luma and chroma is thus 404 macroblocks (giving a value of GCF.DFA of 101).

Figure 55 : Buffer Sequencing Example in 8-Mbit Mode



The value of 1260 for the start of the picture buffer area is calculated as follows :

Total amount of memory = 4096 x 64 32-bit words (pointer addresses are in units of 64 32-bit words in 8-Mbit mode).

Memory for one frame buffer = 720 x 480 x 12bits = 2025 x 64 32-bit words.

2 DFA gaps = 2 x 404 x 4 x 64 x 8bits = 808 x 64 32-bit words.

4096 - 2025 - 808 = 1263, the space left for bit buffer and OSD.

However, since pointer addresses must be multiples of 4, 2025 must be rounded up to 2028.

Thus, 4096 - 2028 - 808 = 1260.

In this example the buffer states are shown after reconstruction of the picture into the buffer pointed to by RFP. The forward predictors are fetched from the buffer pointed to by FFP. The pointer address values given are the values loaded into the RFP register. The algorithm used to calculate pointer value is the following :

- First frame :

RFP = first allowed starting address beyond bit buffer

FBP = first allowed starting address beyond bit buffer

- Subsequent frames :

FFP = RFP

RFP = RFP - 4 x GCF.DFA

If RFP < FBP then RFP = RFP + 4096 - FBP

X.4 - Performance in 8-Mbit Mode

STi3500A performance is lower in 8-Mbit mode since the reduced memory data bus width reduces the available data transfer bandwidth.

For picture sizes which allow 3 frame buffers to be stored in the memory, all prediction modes can be supported.

For larger picture sizes, where memory folding is required, only forward prediction is possible. In this case, for CCIR 601-sized pictures, there is a limitation on the number of dual-prime macroblocks which can be decoded in any picture.

7929237 0073006 054

XI - CONTROL OF THE STi3500A

This section explains in outline how the STi3500A must be controlled in order for it to perform real-time decoding. Only the principles are given here ; more detail is available in the STi3500A Application Note.

XI.1 - Initialization

After a power-on hard reset, certain registers must be set up before decoding can start. These concern :

- general configuration : enable interfaces, set memory mode and refresh interval (CTL and GCF),
- bit buffer : set up BBS,
- interrupt unit : read ITS to clear all pending interrupts. Set up ITM,
- memory pointers : set up RFP for first decoded picture,
- display : set up display window (XDO, XDS, YDO, YDS). Set up OSD if required. Disable display.

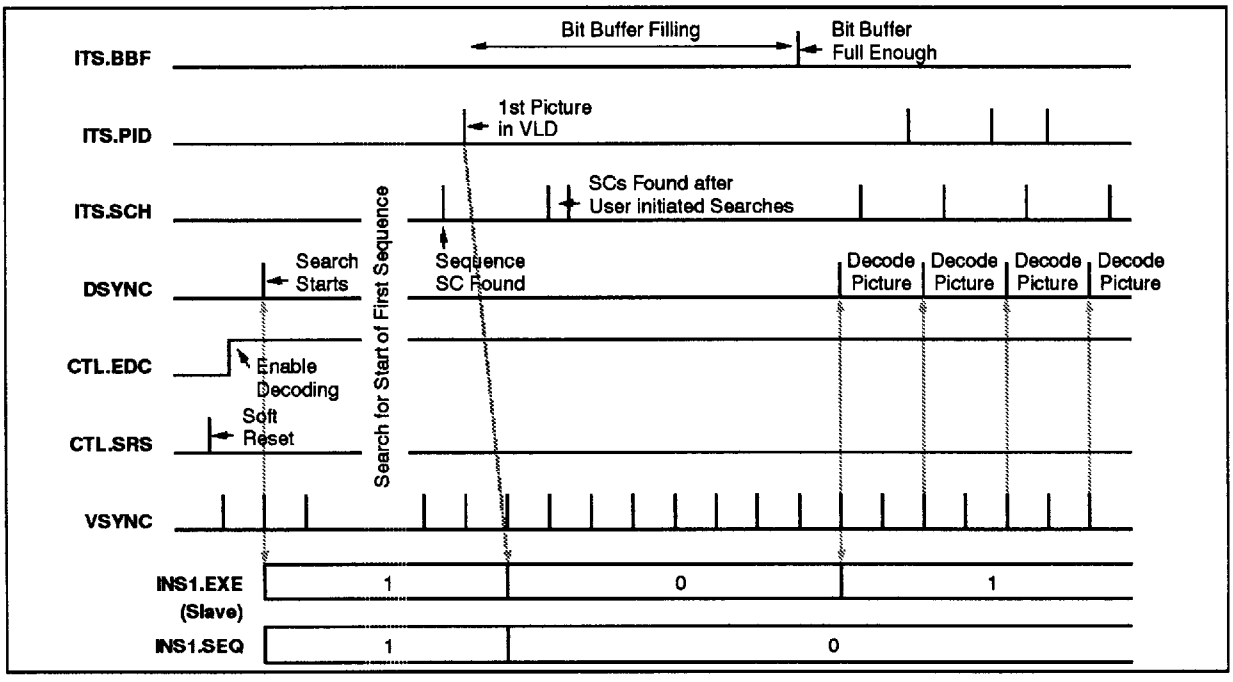
Parameters concerning the format of the decoded picture will be read from the first sequence header.

Figure 56 shows the events which occur during the start-up of decoding the new sequence. This would occur after a hard or a soft reset. (In the diagram a soft reset is shown).

After the reset, the instruction bits INS1.EXE and INS1.SEQ are set, and decoding is enabled by setting bit CTL.EDC. When the next VSYNC occurs, the pipeline starts its task and a DSYNC is generated. The latter event starts the start code detector. When the DSYNC has been detected (indicated by the PSD interrupt), the instruction bits INS1.EXE and INS1.SEQ are written as zero. This will take effect on the next "new instruction" event (see section VIII.4, "Decoding Task Control").

The search for the beginning of the new sequence now takes place - this is the only possible action after a hard or soft reset. When the start code detector has found a sequence start code it stops and an SCH interrupt is generated. When the pipeline has found the first picture header after a sequence header, it too stops and a PID interrupt is generated. On the next VSYNC the new instruction is executed, but since this has INS1.EXE = 0, the pipeline waits. During this searching process the bit buffer does not fill ; all data entering the bit buffer is transferred immediately to the pipeline. However, when the pipeline has found the start of the sequence, the bit buffer starts to fill. During this time the sequence header can be read from the header FIFO and analysed. The decoder set-up can be completed and the quantization tables loaded. The start codes following the first sequence header can be detected by launching additional start code searches (using command CMD.HDS). This will include the first picture start code, following which is the picture header containing the information needed to complete the first decoding instruction. The bit buffer threshold, BBT, can now be calculated from the "vbv_delay" parameter and set up.

Figure 56 : Initialization of a Sequence



SGS-THOMSON MICROELECTRONICS

7929237 0073007 T90

When the BBF interrupt occurs, there is enough data in the bit buffer for decoding to commence. The decoding instruction for the first picture is now loaded, with bit INS1.EXE set. On the next VSYNC, the instruction is executed, and a DSYNC is generated. The correct phasing of the first picture decode with respect to the top and bottom fields can be ensured by monitoring the VST and VSB interrupts before loading the first instruction. A start code detection is launched in response to the DSYNC, enabling the start code for the next picture to be found and analysed. When decoding is complete, a PID (pipeline idle) interrupt is generated.

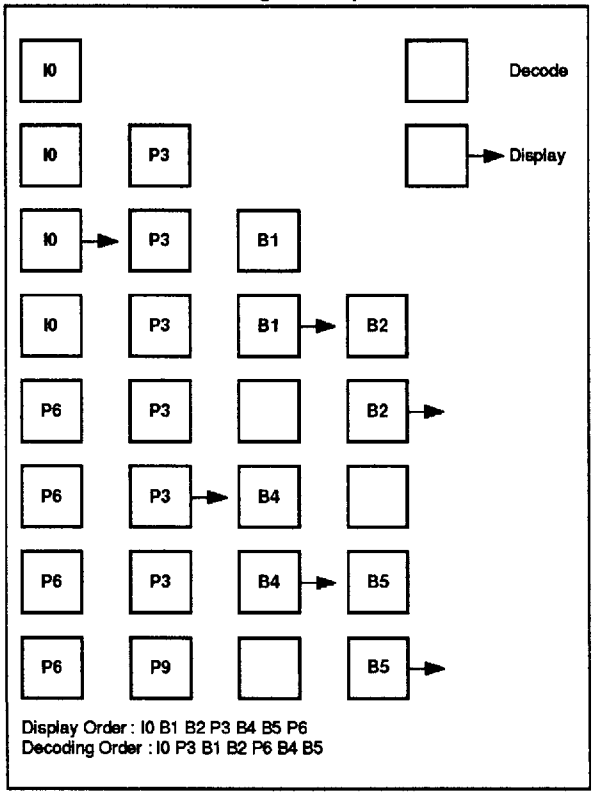
This pattern of events is now repeated until the end of the sequence.

XI.2 - Buffer Sequencing and Overwrite Mode

Before the decoding of each picture, when the picture header has been analysed, the memory addresses pictures buffers from which the predictors are to be fetched, the buffer in which the pictures is reconstructed, and the display buffer must be set up.

The use of the different buffers is described in section VIII.3, "Utilization of Picture Pointers".

Figure 57 : Allocation of 4 Picture Buffers for Decoding of Sequence with M = 3



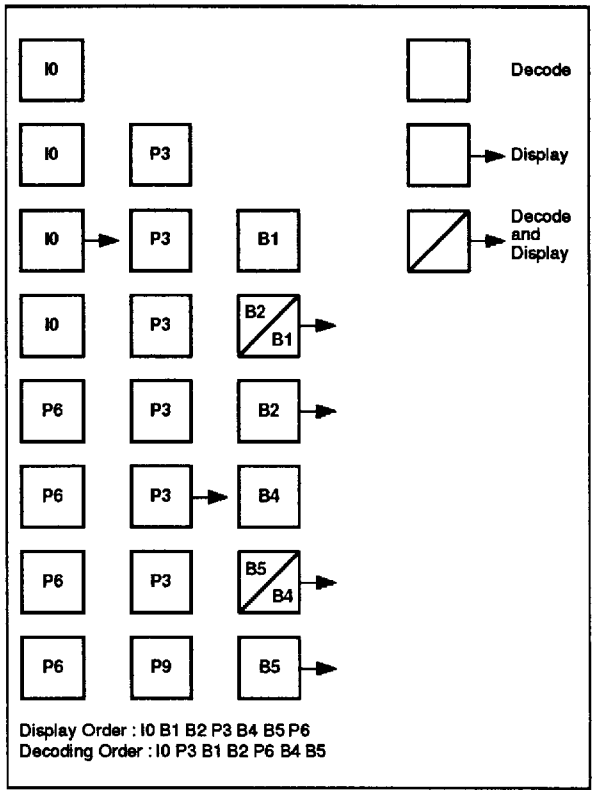
The pointer to the displayed picture, DFP, is updated on every VSYNC. The pointers to the buffers used by the pipeline BFP, FFP and RFP, are updated when a DSYNC occurs, i.e. at the start of decoding.

Figure 57 shows one way of allocating picture buffers for decoding a sequence in which there are 2 B-pictures between I- or P-pictures (M = 3). Four buffers are used, since in the worst case - the decoding of a B-picture while the previously decoded B-picture is being displayed - two buffers are required for the reference pictures, one for the picture currently being decoded, and one for the picture being displayed.

Figure 58 shows how the same sequence can be decoded using only three picture buffers. This is made possible by using the same picture buffer for reconstruction and display (i.e. RFP = DFP) when the displayed picture does not have to be saved, which is the case for B-pictures. The displayed picture is thus overwritten by the picture being constructed.

Overwrite Mode is enabled for a picture decoding operation by setting bit INS1.OVW of the instruction, and setting RFP and DFP to the same value.

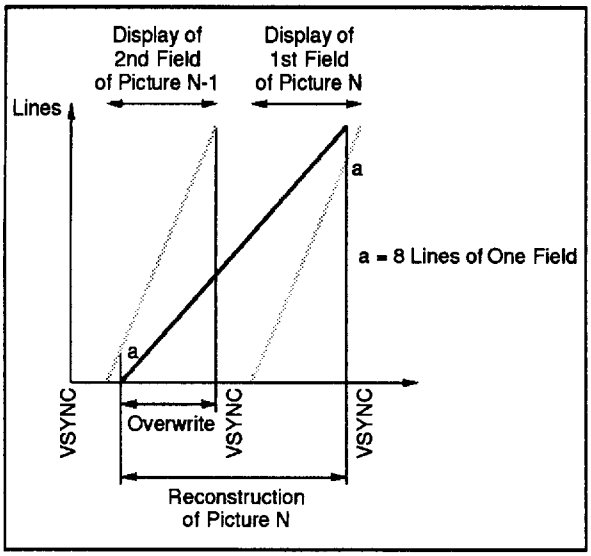
Figure 58 : Allocation of 3 Picture Buffers for Decoding of Sequence with M = 3 with Overwrite Mode Selected



7929237 0073008 927

Figure 59 shows how overwrite mode operates when using the same buffer for reconstruction and display of an interlaced picture.

Figure 59 : Picture Buffer Overwrite



The reconstruction of a picture, which is performed macroblock-by-macroblock, is overlapped with the display of the second field of the previous picture and the display of the first field of the current picture. Overwriting cannot start until 8 lines have been displayed ; this frees one row of macroblocks in the buffer for writing. Reconstruction must be complete 8 lines before the end of the display ; this ensures that the last line of macroblocks has been written. There is an automatic mechanism which ensures that the overwriting of reconstructed data never overtakes the display process.

When decoding a picture with overwrite mode enabled, there is less than a full frame period available for decoding. If the decoding of a picture can not be completed in this time, the end of the display would not be updated, but the decoding sequence would not be disturbed, since reconstruction would still be completed before the next picture decoding starts (see section VIII.4, "Decoding Task Control").

XI.3 - Decoding/Display Synchronization

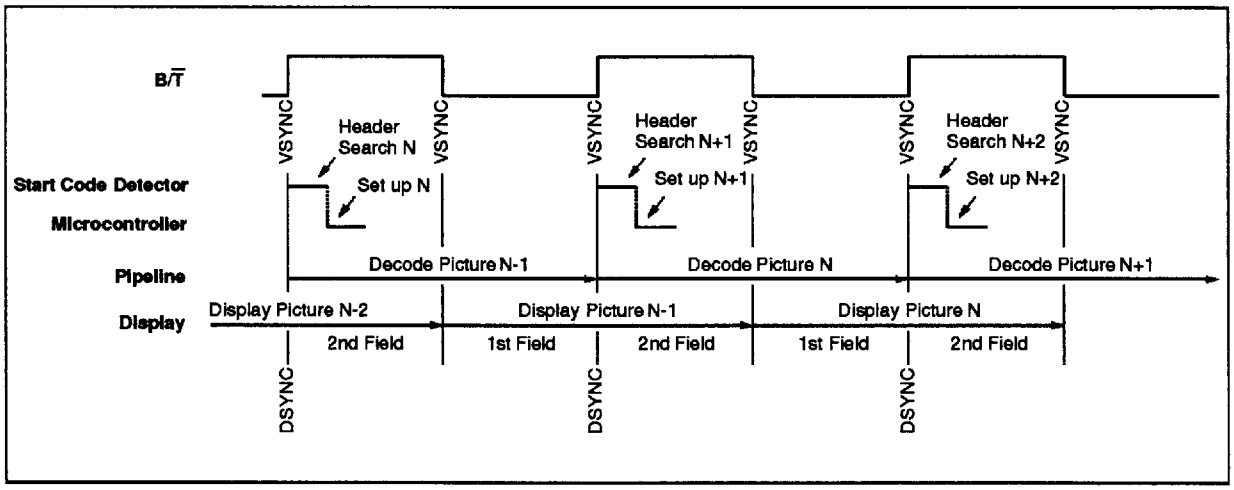
Figure 60 shows how picture decoding and display are synchronized in the most common application of the STi3500A : full resolution pictures, interlaced display with overwrite mode enabled when decoding and displaying B-pictures. It is assumed in this diagram that the decoding and display order is the same. This does not affect the principle of operation.

The bit INS1.RPT is set in every instruction ; this ensures that the decoding task duration is two VSYNC intervals. A DSYNC is generated on every second VSYNC.

While a picture is being decoded the start code detector scans through the same picture data in order to locate the start code of the following picture. When this has been detected the microcontroller can read the header data and set up in advance the parameters which will be taken into account on the next DSYNC.

When decoding half-resolution pictures, the same information must be read from the display in both fields. If overwrite mode is used, then the timing will be the same as that shown in Figure 60. Bit CTL.HRD must be set since overwriting cannot commence until 16 lines have been displayed.

Figure 60 : Decoding/Display Synchronization : Full Resolution, Interlaced Display, Overwrite Mode



SGS-THOMSON MICROELECTRONICS

7929237 0073009 863

XI.4 - 3:2 Pull-Down Operation

For display of a progressive sequence coded at 24 frames/sec on a 30 frames/sec display, pictures must be alternately displayed for periods of 2 and 3 fields. Figure 61 shows how decoding and display are synchronized in this application. (As before it is assumed in this diagram that the decoding and display order is the same).

For one VSYNC period in five, no decoding operation is required. In these periods the pipeline is put into a waiting state by defining an instruction with INS1.EXE = 0 and INS1.RPT = 0.

XI.5 - Control in 8-Mbit Mode

XI.5.1 - Overwrite Mechanism

In 8-Mbit mode display buffer overwrite works in a manner different to that of normal mode. The operation has three phases, illustrated in Figure 62.

Phase 1 :

The STi3500A generates DSYNC (and a PSD interrupt) and latches the new instruction. The reconstruction of the new picture can start into the free memory area whose size is equal to GCF.DFA x 4 macroblocks. Reconstruction will stop when this area is filled. Note that if GFA.DFA = 0, no data are reconstructed during this phase.

During this phase the first field of the previous picture is displayed.

Figure 61 : Decoding/Display Synchronization in 3:2 Pull-Down Mode with Overwrite

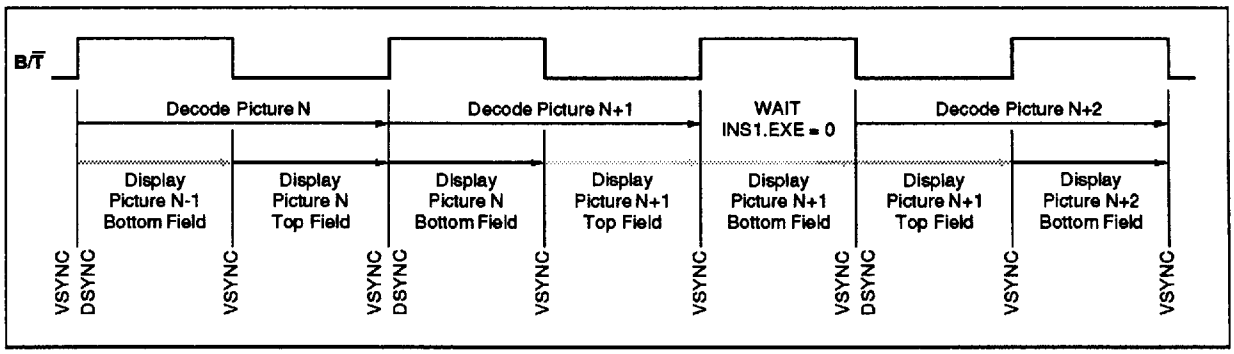
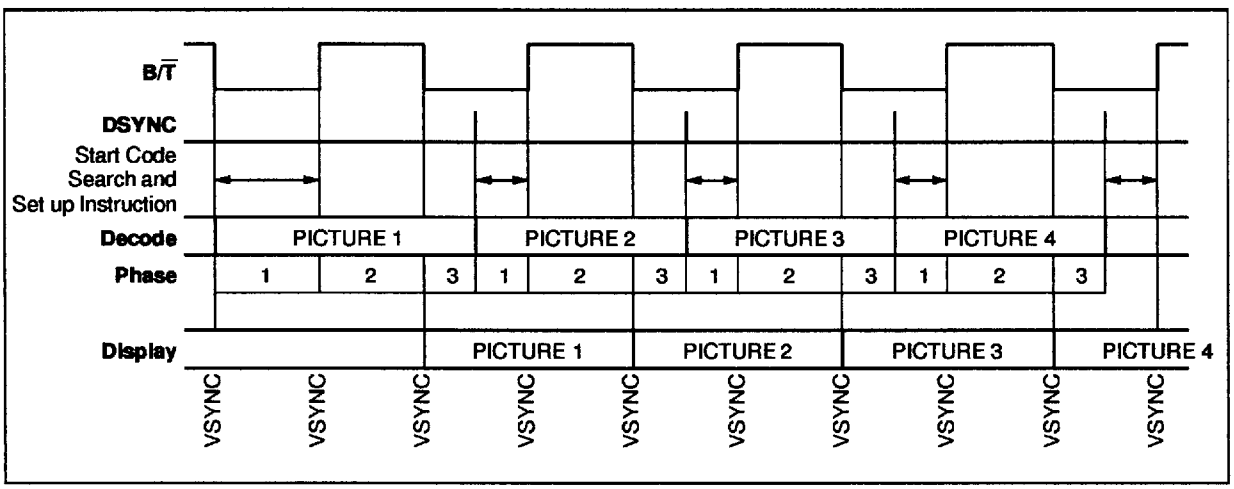


Figure 62 : Decoding/Display Synchronization in 8-Mbit Mode



7929237 0073010 585

Phase 2 :

Reconstruction will continue with the constraint that memory locations still required for the display of the second field of the previous picture will not be overwritten ; the condition :

decode address < display address + GCF.DFA x 4

is satisfied.

At the end of this phase the picture will not be completely decoded.

Phase 3 :

With the overwrite constraint removed, decoding will proceed as quickly as possible up the end of the picture. The end of decoding can occur several milliseconds after VSYNC. At this point a new DSYNC is generated., the next instruction is latched and a new decoding "Phase 1' starts for the next picture. Note that some bits of the instruction (INS1.OVW, INS1.SEQ, INS1.EXE, INS1.RPT and CMD.SKP[1:0]) are latched on VSYNC, while the others are latched on DSYNC which occurs several milliseconds later (cf. section VIII.5, "Task Overrun").

Except for the very first picture, the complete decoding task will take roughly 2 VSYNC periods, as in normal mode, except that decoding is shifted with respect to VSYNC. All tasks are delayed instead of starting immediately.

XI.5.2 - Decoder Control

The consequence of the changes in the overwrite mechanism is that decoding control must be modified ; the decoding process must start two fields before the first display instead of one. If the display is done top-field-first (as in the example of Figure 62), then the decoding task must also start on top fields.

The following points must be noted :

- The instruction can only be computed after the next start code is detected. As start code detection is started automatically by DSYNC, the header will only be available in the middle of the VSYNC period. The time available for the microcontroller to compute the instruction is less than normal. The instruction must be set up before the next VSYNC which leaves less than one and half fields.
- The DFP register is not set up at the same time as before - it must not be changed on DSYNC or on start code detection but only after the following VSYNC (i.e. at the start of decoding Phase 2).

Unrestricted Decoding

In this case the 8Mbits of memory allows enough space for the bit buffer and the frame buffers. The control can be done in two different ways :

- 1. Overwrite mode always off. In this case the control of the STi3500A is exactly as in standard mode.
- 2. Overwrite mode always on. In this case the control must be performed as described in this section.

It is not recommended to switch the overwrite control bit (INS1.OVW) on and off while decoding since the control phase depends on overwrite and may become difficult to manage.

Decoding with Memory Folding

If memory folding is used (because 8Mbits are not enough in which to do unrestricted decoding), overwrite mode must always be activated and control must be done as described in this section.

Note that when DFP is changed in Phase 2, it must be changed to the value of the current RFP, not to the value of the next one which has at this time been pre-computed.

SGS-THOMSON MICROELECTRONICS

7929237 0073011 411

XII - REGISTERS

XII.1 - Register Map

Address (hex)	MSByte	LSByte	Address (hex)
00	HDF (R)		01
02		CMD (RW)	03
04	GCF (R/W)		05
06	CTL (R/W)		07
08	STA (R)		09
0A	ITM (R/W)		0B
0C	ITS (R)		0D
0E	INS1 (R/W)		0F
	INS2 (R/W)		
10	MRF (R)	MWF (W)	11
	BMS (R/W)		
12			13
14	MRP (R/W)		15
16			17
18	MWP (R/W)		19
1A	DFP (R/W)		1B
1C	RFP (R/W)		1D
1E	FFP (R/W)		1F
20	BFP/FBP (R/W)		21
22	BBL (R)		23
24	BBS (R/W)		25
	BBG (R/W)		
26	BBT (R/W)		27
28		DFW (R/W)	29
2A	DFS (R/W)		2B
2C	YDO (R/W)	XDO (R/W)	2D
2E	YDS (R/W)	XDS (R/W)	2F
30	OBP (R/W)		31
32	OTP (R/W)		33
34	LSO (R/W)	LSR (R/W)	35
36	CSO (R/W)	CSR (R/W)	37
38	DCF (R/W)		39
3A	PSV (R/W)		3B
3C	QMW (W)	QMW (W)	3D
3E			3F

7929237 0073012 358

XII.2 - Register Descriptions

Registers are listed in alphabetical order.

All addresses are in hexadecimal.

All unspecified bits of the register map are reserved. Only the value 0 must be written to any of these bits. The values which are read from these bits are undefined.

The reset state is the state existing after a hard reset.

Synchronization

There are two types of register : synchronized and unsynchronized.

Synchronized registers only change value in response to an internal event, either DSYNC or VSYNC, depending on the register. These registers are double-banked ; during the write cycle the new value is loaded into a master register, and on the occurrence of the synchronizing event this value is loaded into a slave register, at which time the new value is available to the circuit. When a synchronized register is read, the value returned is that held in the master register.

Unsynchronized registers change their value immediately they are written to. With the exception of the BBT, BMS, CTL and ITM registers and bit CMD.HDS, the unsynchronized registers are simple latches which are open when the signals $\overline{CS}$ and R/W are both low.

BBG - Start of Bit Buffer

13	0
BBG[13:0]	

Address : 24-25
Type : R/W
Reset State : 0
Synchronization : None

Description

The register holds the starting address of the bit buffer, defined in units of 2Kbits (32 words). If the bit buffer starts at address 0, then this register does not need to be set up, since its reset state is 0. A soft reset must be done immediately after the loading of this register in order for the value to be taken into account. In other words it must only be changed before the first compressed data of a new sequence is input, and never during the decoding of a sequence.

This register shares the same addresses as the BBS register. Bit CMD.BBG selects which of the two registers is accessible at these addresses. When this bit is set, the BBG register is selected.

BBL - Bit Buffer Level

13	0
BBL[13:0]	

Address : 22-23
Type : R
Reset State : 0

Description

This register holds the current level of occupation of the bit buffer, defined in units of 2Kbits (32 words). It can be read at any time for the monitoring of the bit buffer level. When BBL is greater than or equal to the value held in the BBT register, the status bit STA.BBF becomes set. When BBL is zero, the status bit STA.BBE becomes set. These conditions can be used to cause interrupts.

BBS - Bit Buffer Stop Address

13	0
BBS[13:0]	

Address : 24-25
Type : R/W
Reset State : 0
Synchronization : None

Description

This register holds the address of the top of the bit buffer, defined in units of 2Kbits (32 words). The space allocated to the bit buffer starts at the address defined by the BBG register, or, by default, 0. The end address of the bit buffer is :

$$(32 \times BBS) + 31$$

BBS must only be changed before the first compressed data of a new sequence is input, and never during the decoding of a sequence.

This register shares the same addresses as the BBG register. Bit CMD.BBG selects which of the two registers is accessible at these addresses. When this bit is reset, the BBS register is selected.

BBT - Bit Buffer Threshold

13	0
BBT[13:0]	

Address : 26-27
Type : R/W
Reset State : 0
Synchronization : None

Description

This register holds the level of occupancy of the bit buffer, in units of 2Kbits (32 words), which when

SGS-THOMSON MICROELECTRONICS

7929237 0073013 294

reached causes the status bit STA.BBF to become set, i.e. if BBL ≥ BBT, STA.BBF is set.

This threshold would normally be used to generate a "bit buffer nearly full" interrupt.

If the bit CTL.PBO is set, then transfer of data from the CD FIFO to the bit buffer is prevented if the bit buffer level is at or above the level defined in the BBT register. If BBT is set to a value equal to the size of the bit buffer (i.e. BBT = BBS + 1 - BBG), then this automatic mechanism will ensure than overflow never occurs.

BFP/FBP - Backward Frame / Fold-back Pointer

13	0
BFP/FBP[13:0]	

Address : 20-21
Type : R/W
Reset State : 0
Synchronization : DSYNC

Description

This register holds the start address of the backward prediction frame picture buffer, defined in units of 32 words.

If this buffer is also to be used for display, then the bottom two bits, BFP[1] and BFP[0], must be zero. In this case the backward frame buffer is constrained to start at an address which is a multiple of 128 words.

When the STi3500A is operating in 8-Mbit mode (bit CTL.S8M is set), this register has a dual function. If address folding is required, the register must be set up with the value of the starting address in memory, in units of 64 32-bit words, of the frame buffer area. This is the fold-back address. If address folding is not required, i.e. accesses will never go beyond the end of memory, then this register is set up with the backward prediction frame buffer address. Thus, address folding and B-picture decoding are mutually exclusive.

BMS - Block Move Size

15	0
BMS[15:0]	

Address : 10-11
Type : R/W
Reset State : 0
Synchronization : None

Description

This register holds the number of words to be moved in a block move operation. Writing to it starts a block move operation.

This register shares the same addresses as the MRF and MWF registers. Bit CMD.SBM selects whether MRF/MWF or BMS is accessible at these addresses. When this bit is set, the BMS register is selected.

CMD - Command (Write)

9	8	7	6	5	4	3	2	1	0
AVS[1:0]		SBM	BBG	SKP[1:0]		INS	QMN	QMI	HDS

Address : 02-03
Type : W
Reset State : 0
Synchronization : SKP[1:0] with VSYNC, others none. Bit HDS is stored in a master-slave register to prevent undesired launching of header searches.

Description

The following write-only bits are used to control a variety of operations.

AVS[1:0] These bits select which data is available when the CMD register is read.

AVS[1]	AVS[0]	CMD data
0	0	CDcount[15:0]
0	1	CDcount[23:16]
1	0	SCDcount[15:0]
1	1	SCDcount[23:16]

SBM Select block move. When this bit is set block move operation is enabled and the BMS register is accessible at addresses 10 and 11. Otherwise (the default state) the MRF and MWF registers are accessible and read/write operations to the memory are enabled.

BBG Select BBG register. When this bit is set, the BBG register is accessible at addresses 24 and 25. Otherwise (the default state) the BBS register is accessible.

SKP[1:0] These bits are part of the instruction register, and are synchronized by the signal VSYNC. They define the skipping of one or two pictures, as defined in the following table :

SKP[1]	SKP[0]	
0	0	No skip (default)
0	1	Skip one picture, decode next
1	0	Skip two pictures, decode next
1	1	Illegal

7929237 0073014 120

If skipping is required, then these bits must be set up as part of the instruction for the picture which will be decoded. The skipping and the decoding of the following picture represent one task.

INS This bit selects which instruction register, INS1 or INS2, is accessible. When set, INS2 is selected, when reset (the default state), INS1 is selected.

QMN These two bits are used to control access

QMI to the inverse quantizer tables.

QMN	QMI	
0	0	Tables not accessible (lock)
0	1	Select the intra table
1	0	Select the non-intra table
1	1	Enable writing to the selected table

For example, to write a new intra table, the following steps are required :

Write CMD.QMI = 1, CMD.QMN = 0

Write CMD.QMI = CMD.QMN = 1

Write 64 weights to QMW

Write CMD.QMI = CMD.QMW = 0

HDS Writing a 1 to this bit starts a header search. Completion of the header search is indicated by the setting of bit STA.SCH.

CMD - Command (Read)

15	0
CD/SCDcount[15:8]	CD/SCDcount[23:16],[7:0]

Address : 02-03
Type : R
Reset State : 0

Description

The contents of the bit buffer input and output counters, "CDcount" and "SCDcount" can be read from addresses 02 and 03. Which data is accessible is defined by the state of bits CMD.AVS[1:0].

AVS[1]	AVS[0]	CMD data
0	0	CDcount[15:0]
0	1	CDcount[23:16]
1	0	SCDcount[15:0]
1	1	SCDcount[23:16]

CDcount[23:16] or SDCcount[23:16]is available in bits CMD[7:0]. CDcount[15:0]or SCDcount[15:0]is available in bits CMD[15:0].

CSO - SRC Chrominance Offset

7	0
CSO[7:0]	

Address : 36
Type : R/W
Reset State : 0
Synchronization : VSYNC

Description

This register is set up with a value calculated from the fractional part of the pan vector. If no pan vector is defined, this register can be left in its reset (default) state.

The method of calculation of the CSO value is given in the PSV register description.

CSR - SRC Chrominance Resolution

7	0
CSR[7:0]	

Address : 37
Type : R/W
Reset State : 0
Synchronization : VSYNC

Description

The register holds the upsampling factor of the chrominance SRC (sample rate converter). This value must always be equal to that loaded into the LSR register.

The method of calculation of this value is given in the LSR register description.

CTL - Control

15	14	13	12	11	10	9		7	6	5	4	3	2	1	0
35A	DEC	S8M	PBO	MP2	HRD	EPR		EC3	EC2	ECK	EDI	EVI	PRS	SRS	EDC

Address : 06-07
Type : R/W
Reset State : 0
Synchronization : None

This is implemented as a master-slave register in order to prevent problems of erroneous commands being issued during the write cycle.

SGS-THOMSON MICROELECTRONICS

7929237 0073015 067

Description

35A — Enable STi3500A features. When this bit is set, the additional functions of the STi3500A which are not compatible with the STi3500 are activated. When this bit is reset, the STi3500A behaves in a manner which is compatible with the STi3500.
The features which are enabled by this bit are :
- GCF.DFA[7:0] is defined.
- DFW, DFS are synchronized to DSYNC.
- YC output never has values 00h and FFh.
- PSV.V[7:1] and PSV.H[8] are defined.
- CSO, CSR, DCF.PXD, DCF.EOS, DCF.DSR, LSO, LSR, XDO, XDS, YDO, YDS are synchronized to VSYNC.
- XDO[9,8] and XDS[9,8] are defined.

DEC — Disable error concealment. When this bit is set, automatic error concealment is disabled. This should only be done for debugging purposes, since the occurrence of a syntax error with error concealment disabled will cause decoding to stop. Recovery is only possible with a pipeline or a soft reset.

S8M — Select 8Mbit memory mode. When this bit is set, 8Mbit memory mode is entered.

PBO — Prevent bit buffer overflow. When this bit is set, bit buffer overflow (and thus the loss of data) is prevented by disabling the transfer of data from the compressed data FIFO to the bit buffer whenever the bit buffer level reaches the threshold defined in the BBT register.

MP2 — MPEG-2 mode. When this bit is set, the STi3500A expects an MPEG-2 video bitstream. If it is reset, then an MPEG-1 bitstream is expected.

HRD — Half-resolution display. This bit must be set if the decoded picture vertical resolution is half that of the display, i.e. the whole decoded picture is displayed in both fields. (This bit determines whether buffer overwriting can commence after 8 (CTL.HRD = 0) or 16 (CTL.HRD = 1) lines).

EPR — Enable pipeline reset. When this bit is set, a pipeline reset is automatically generated if more than 64 samples are decoded in a block (i.e. in case of pipeline error).

EC3 — Enable "clock 3". When this bit is reset, the internal "clock 3" is disabled. This bit must be set for normal operation and for reduced power mode. It is reset in low power mode. (Power-down modes are defined in section VI.2, "Power-Down Modes").

EC2 — Enable "clock 2". When this bit is reset, the internal "clock 2" is disabled. This bit must be set for normal operation, and reset in reduced and low power modes. (Power-down modes are defined in section VI.2, "Power-Down Modes").

ECK — Enable clocks. When this bit is reset, all internal clocks are disabled. This bit must be set for normal operation and for reduced power mode. It is reset in low power mode. (Power-down modes are defined in section VI.2, "Power-Down Modes").

EDI — Enable DRAM interface. When this bit is reset the DRAM interface ($\overline{DD63}$-$\overline{DD0}$, $\overline{AA8}$-$\overline{AA0}$, $\overline{RAS1}$, $\overline{RAS0}$, $\overline{CAS}$, $\overline{OE}$ and $\overline{WE}$) and the signal CDREQ are put into their high impedance state. This bit must be set for normal operation and for reduced power mode. It is reset in low power mode. (Power-down modes are defined in section VI.2, "Power-Down Modes").

EVI — Enable video interface. When this bit is reset the video interface (YC7-YC0) is put into its high impedance state and the internal PIXCLK disabled. This bit must be set for normal operation and for reduced power mode (if the display interface is used). It is reset in low power mode. (Power-down modes are defined in section VI.2, "Power-Down Modes").

PRS — Pipeline reset. In order to generate a pipeline reset, this bit must be kept set for a duration of at least 3 primary clock cycles (55ns with a 55MHz primary clock). The effect of a pipeline reset is described in section VI.1, "Resets".

7929237 0073016 TT3

SRS Soft reset. In order to generate a soft reset, this bit must be kept set for a duration of at least 40 primary clock cycles (730ns with a 55MHz primary clock). The effect of a soft reset is described in section VI.1, "Resets".

EDC Enable decoding. This bit must be set to enable the decoding pipeline to run. It's action is asynchronous and independent of the task control state machine (see Figure 29). If EDC is reset while the pipeline is idle, then it is still possible for the next instruction to be executed on the next VSYNC, and for a DSYNC to be generated. However, in this case, the decoding task will not start until EDC is set.

DCF - Display Configuration

12	11 9	8	7	6	5	4	3	2 0
XYE	DAM[2:0]	FLD	USR	PXD	EVD	EOS	DSR	VFC[2:0]

Address : 38-39
Type : R/W
Reset State : 0
Synchronization : XYE, USR : None
Others : VSYNC

Description

XYE Select XDO/XDS extensions. When this bit is set, register bits XD0[9,8] and XDS[9,8] are accessible at addresses 2Dh and 2Fh. When is it reset, XDO[7:0] and XDS[7:0] are accessible.

DAM[2:0] Display access mode. These bits are only active when bit DCF.USR is set. They are used in conjunction with the bit DCF.FLD. Their use is described in section IX.7, "Displayed Field Sequence Control".

FLD Field bit. This bit is only active when bit DCF.USR is set. It defines which field is to be displayed : 1 for the bottom field, 0 for the top.
Bits DCF.DAM[2:0] and DCF.FLD must be programmed in conjunction with the bits DCF.VFC[2:0], as defined in the table appearing in section IX.7, "Displayed Field Sequence Control".

USR Enable user field sequence control. When this bit is set, bits DCF.DAM[2:0] and DCF.FLD can be programmed field-by-field to enable a user-defined display sequence. Otherwise, these bits have no effect, and the fields are displayed in a sequence determined by the B/$\overline{T}$ signal.

PXD Add one PIXCLK delay. When this bit is set the active video is delayed by one PIXCLK cycle with respect to $\overline{HSYNC}$. This is to allow its horizontal position to be defined more precisely than is possible with XDO and XDS, which have a resolution of 2 PIXCLK cycles. Changing the value of PXD also has the effect of inverting the phasing of the Y/C output samples with respect to $\overline{HSYNC}$.

EVD Enable video display. When this bit is reset, the video output has a constant value of Y = 16, $C_B = C_R = 128$. OSD is still displayed.

EOS Enable OSD. When this bit is set, the OSD (on-screen display) bitmap defined in the top and bottom field OSD buffers is displayed over the picture.

DSR Disable SRC. When this bit is set, both luminance and chrominance SRCs (sample rate converters) are disabled. In this case no horizontal filtering can occur, as would be required when the horizontal resolution of the decoded picture is equal to the horizontal resolution of the display.

VFC[2:0] Vertical filter configuration. These bits define the vertical filter mode.

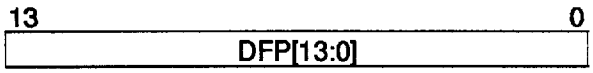
VFC[2]	VFC[1]	VFC[0]	Mode
0	0	0	Full resolution, chrominance line repeat with interpolation
0	0	1	Full resolution, chrominance line repeat
0	1	0	Full resolution, chrominance field repeat with interpolation
0	1	1	Full resolution, chrominance field repeat
1	0	0	Half resolution, chrominance interpolation
1	0	1	Half resolution, chrominance repeat
1	1	0	Half resolution, luminance interpolation
1	1	1	Illegal

The vertical filtering modes are described in detail in section IX.5, "Vertical Filter".

SGS-THOMSON MICROELECTRONICS

7929237 0073017 93T

DFP - Displayed Frame Pointer



Address : 1A-1B
Type : R/W
Reset State : 0
Synchronization : VSYNC

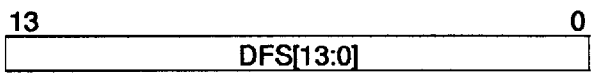
Description

This register holds the start address, defined in units of 32 words, of the frame which is currently being displayed. When a new value is written this is used at the start of the next field.

The bottom two bits, DFP[1] and DFP[0], must be zero. The displayed frame buffer is thus constrained to start at an address which is a multiple of 128 words.

When DFP is set to same value as RFP (i.e. the decoder is writing the reconstructed picture into the buffer which is being displayed), bit INS1.OVW must be set.

DFS - Decoded Frame Size

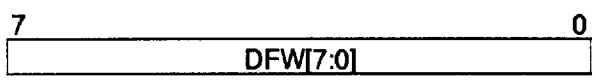


Address : 2A-2B
Type : R/W
Reset State : 0
Synchronization : DSYNC

Description

This register is set up with a value equal to the number of macroblocks in the decoded picture. This is derived from the horizontal_size and vertical_size values transmitted in the sequence header.

DFW - Decoded Frame Width



Address : 29
Type : R/W
Reset State : 0
Synchronization : DSYNC

Description

This register is set up with a value equal to the width in macroblocks of the decoded picture. This is derived from the horizontal_size value transmitted in the sequence header.

FFP - Forward Frame Pointer

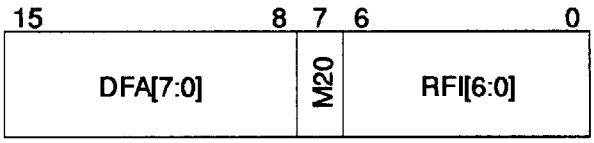
Address : 1E-1F
Type : R/W
Reset State : 0
Synchronization : DSYNC

Description

This register holds the start address of the forward prediction frame picture buffer, defined in units of 32 words.

If this buffer is also to be used for display, then the bottom two bits, FFP[1] and FFP[0], must be zero. In this case the forward frame buffer is constrained to start at an address which is a multiple of 128 words.

GCF - General Configuration



Address : 04-05
Type : R/W
Reset State : 0
Synchronization : DFA[7:0] synchronized to DSYNC
Others : None

Description

DFA[7:0] DFS adjustment. 4 x DFA[7:0] defines the offset, in units of macroblocks, between the end of the luminance storage area and the beginning of the chrominance storage area in the frame buffers. This offset is normally only used in 8-Mbit mode, in which it is set equal to the size in macroblocks of the picture overlap required for motion compensation minus 1, rounded up to the next highest multiple of 4. For example, if the maximum vertical vector range is -128 to 127 and the picture width is 720 pels, then the area needed for motion compensation is :
(720 x (128 + 16)/256) - 1
= 404 macroblocks
thus, DFA[7:0] = 101

7929237 0073018 876

M20 Select 20Mbit memory mode. When this bit is set, 20Mbit memory mode (one 16Mbit bank plus one 4Mbit bank) is entered.

RFI[6:0] DRAM refresh interval. This is defined in units of 24 primary clock periods. For example if each row must be refreshed every 8 ms, with a primary clock of 55 MHz, the following value must be stored in GCF.RFI[6:0]:
$(8 \times 10^{-3}/512) \times (55 \times 10^{6}/24) \rightarrow 35$

HDF - Header Data FIFO

15	0
HDF[15:0]	

Address : 00-01
Type : R
Reset State : Undefined

Description

When the start code detector has found a start code, the header data FIFO must be read in order to identify the start code and if required to obtain the header data. The start code identification procedure is described in section VII.2, "Start Code Detection".

Before reading the header FIFO, status bit STA.HFE should be checked to ensure that it is not empty. Bit STA.HFF set indicates that the header FIFO contains at least 66 bytes of data.

INS1 - Instruction 1

15	14	13 10	9 6	5 4	3	2	1	0
TFF	OVW	BFH[3:0]	FFH[3:0]	PCT[1] PCT[0]	SEQ	EXE	RPT	CMV

Address : 0E-0F
Type : R/W
Reset State : 0
Synchronization : "new instruction" (see Figure 29), except bits OVW, SEQ, EXE and RPT, which are updated by VSYNC

Description

This register contains 16 bits of the decoding instruction. The other bits are CMD.SKP[1:0] and INS2[15:0]. The mechanism of instruction execution is described in section VIII.4, "Decoding Task Control".

TFF This bit is set equal to the top_field_first bit of the MPEG-2 picture coding extension. It is only taken into account when decoding MPEG-2 dual-prime pictures.

OVW This bit must be set when the displayed picture and the reconstructed picture share the same buffer (i.e. DFP = RFP). It enables the overwrite mode which ensures that the reconstructed picture does not overwrite data which has not yet been displayed.

BFH[3:0] In MPEG-1 mode BFH[3] is set equal to full_pel_backward_vector of the picture header, and BFH[2:0] is set equal to backward_f_code of the picture header.
In MPEG-2 mode BFH[3:0] is set equal to backward_horizontal_f_code of the picture coding extension.

FFH[3:0] In MPEG-1 mode FFH[3] is set equal to full_pel_forward_vector of the picture header, and FFH[2:0] is set equal to forward_f_code of the picture header.
In MPEG-2 mode BFH[3:0] is set equal to forward_horizontal_f_code of the picture coding extension.

PCT[1:0] This is set equal to the two least significant bits of picture_coding_type in the picture header. The STi3500A does not support D-pictures, and therefore the code "00" must not be used.

SEQ Search for next sequence. This bit must be set while the decoder is searching for the start of a new sequence after a soft reset. It has the effect of putting the controller into a state in which VSYNCs are ignored.

EXE When this bit is not set, no decoding task is executed for one or two VSYNC periods, depending on the state of RPT.

RPT When this bit is set, the task duration is two VSYNC periods. In many applications, when the frame display rate is equal to the picture decoding rate, RPT will always be set.

CMV This bit is set equal to the concealment_motion_vectors bit of the MPEG-2 picture coding extension. It indicates that motion vectors are coded for intra macroblocks. It is only taken into account when decoding MPEG-2 pictures.

This register shares the same addresses as the INS2 register. Bit CMD.INS selects which of the two registers is accessible at these addresses. When this bit is reset, the INS1 register is selected.

SGS-THOMSON MICROELECTRONICS

7929237 0073019 702

INS2 - Instruction 2

15 14 13	10 9	6 5 4	3	2	1	0
PST[1] PST[0]	BFV[3:0]	FFV[3:0]	DCP[1] DCP[0]	FRM	QST	AZZ	IVF

Address : 0E-0F
Type : R/W
Reset State : 0
Synchronization : "new instruction" (see Figure 29)

Description

This register contains 16 bits of the decoding instruction. The other bits are CMD.SKP[1:0] and INS1[15:0]. The mechanism of instruction execution is described in section VIII.4, "Decoding Task Control".

In MPEG-1 mode (i.e. when CTL.MP2 is reset), INS2 is not used, and must be kept reset.

PST[1:0] This is set equal the picture_structure bits of the MPEG-2 picture coding extension. Note that code "00" also indicates frame structure, even though this value is illegal in the MPEG-2 variable.

PST[1]	PST[0]	
0	0	Frame picture
0	1	Top field
1	0	Bottom field
1	1	Frame picture

BFV[3:0] INS2.BFV[3:0] is set equal to backward_vertical_f_code of the picture coding extension.

FFV[3:0] INS2.FFV[3:0] is set equal to forward_vertical_f_code of the picture coding extension.

DCP[1:0] INS2.DCP[1:0] is set equal to intra_dc_precision of the picture coding extension. The value "11", defining a precision of 11 bits, is not allowed.

FRM This bit is set equal to the frame_pred_frame_dct bit of the picture coding extension.

QST This bit is set equal to the q_scale_type bit of the picture coding extension.

AZZ This bit is set equal to the alternate_scan bit of the picture coding extension.

IVF This bit is set equal to the intra_vlc_format bit of the picture coding extension.

ITM - Interrupt Mask

14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SER	BMI	HFF	RFF	WFE	PID	PER	PSD	VST	VSB	BBE	BBF	HFE	BFF	SCH

Address : 0A-0B
Type : R/W
Reset State : 0 (all interrupts disabled)
Synchronization : None

Description

Any bit set in this register will enable the corresponding interrupt. An interrupt is generated whenever at bit in the STA register changes from 0 to 1 and the corresponding mask bit is set.

ITS - Interrupt Status

14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SER	BMI	HFF	RFF	WFE	PID	PER	PSD	VST	VSB	BBE	BBF	HFE	BFF	SCH

Address : 0C-0D
Type : R
Reset State : 0

After clocks have been enabled, the state changes to be the same as that of STA.

Description

When a bit in the STA register changes from 0 to 1, the corresponding bit in the ITS register is set, independent of the state of ITM. If any set ITS bit is unmasked, the signal $\overline{IRQ}$ is asserted. Reading the most significant byte of ITS clears it, leaving $\overline{IRQ}$ in its de-asserted (high) state.

See section IV.3, "Interrupts" for more information on interrupt handling.

LSO - SRC Luminance Offset

7	0
LSO[7:0]	

Address : 34
Type : R/W
Reset State : 0
Synchronization : VSYNC

Description

This register is set up with a value calculated from the fractional part of the pan vector. If no pan vector is defined, this register can be left in its reset (default) state.

The method of calculation of the LSO value is given in the PSV register description.

7929237 0073020 424

LSR - SRC Luminance Resolution

7	0
LSR[7:0]	

Address : 35
Type : R/W
Reset State : 0
Synchronization : VSYNC

Description

This register holds the upsampling factor of the luminance SRC (sample rate converter).

The upsampling factor is equal to 256/LSR. Below are given some examples of upsampling factors, where in each case the displayed picture has a nominal width of 720 pels*. Also shown are the numbers of valid pels generated, "N", calculated as shown in section IX.4.1, "Sample Rate Converter".

* Displayed picture widths other than 720 can of course be supported.

Decoded Picture Width	LSR	N
640	228	715
640	227	718
544	193	717
544	192	721
480	170	717
480	169	722
352	125	713
352	124	719
704	250	717
704	249	720

MRF - Memory Read FIFO

7	0
MRF[7:0]	

Address : 10
Type : R
Reset State : Undefined

Description

The memory read FIFO data is read from this address. The mechanism of data transfer from the memory into the FIFO is described in section V.6, "Memory Read and Write through the Microcontroller Interface". Each 64-bit (or 32-bit, when in 8-Mbit mode) word is accessed most significant-byte-first.

This register shares the same address as the most significant byte of the BMS register. Bit CMD.SBM selects whether MRF or BMS is accessible at this address. When this bit is reset, the MRF register is selected.

MRP - Memory Read Pointer

18	0
MRP[18:0]	

Address : 13-14-15
Type : R/W
Reset State : Undefined
Synchronization : None

Description

This register holds the address of the word which will be transferred next from the memory into the memory read FIFO. It is incremented each time a word is transferred from the memory to the FIFO. It is thus not the address of the word currently available in the FIFO, but one or two addresses ahead.

MRP is only set up at the beginning of a memory read sequence, at which time the memory read FIFO is cleared. The mechanism of data transfer from the memory into the FIFO is described in section V.6, "Memory Read and Write through the Microcontroller Interface".

MWF - Memory Write FIFO

7	0
MWF[7:0]	

Address : 11
Type : W
Reset State : Undefined
Synchronization : None (not a register)

Description

The memory write FIFO data is written to this address. The mechanism of data transfer from the FIFO to the memory is described in section V.6, "Memory Read and Write through the Microcontroller Interface". Each 64-bit (or 32-bit, when in 8-Mbit mode) word must be written most significant-byte-first.

This register shares the same address as the least significant byte of the BMS register. Bit CMD.SBM selects whether MWF or BMS is accessible at this address. When this bit is reset, the MWF register is selected.

SGS-THOMSON MICROELECTRONICS

7929237 0073021 360

MWP - Memory Write Pointer

18	0
MWP[18:0]	

Address : 17-18-19
Type : R/W
Reset State : Undefined
Synchronization : None

Description

This register holds the address of the word which will be transferred next from the memory write FIFO into the memory. It is incremented each time a word is transferred from the FIFO to the memory. It is thus not the address of the word last written to the FIFO, but one or two addresses behind.

MWP is only set up at the beginning of a memory write sequence, at which time the memory write FIFO is cleared. The mechanism of data transfer from the FIFO to the memory is described in section V.6, "Memory Read and Write through the Microcontroller Interface".

OBP - OSD Bottom Field Pointer*

13	0
OBP[13:0]	

Address : 30-31
Type : R/W
Reset State : 0
Synchronization : VSYNC bottom

Description

The register holds the start address, in units of 32 words, of the current OSD specification buffer for the bottom field. This specification will be decoded during bottom fields when OSD is enabled (bit DCT.EOS is set).

* Formerly called OEP.

OTP - OSD Top Field Pointer*

13	0
OTP[13:0]	

Address : 32-33
Type : R/W
Reset State : 0
Synchronization : VSYNC top

Description

The register holds the start address, in units of 32 words, of the current OSD specification buffer for the top field. This specification will be decoded during top fields when OSD is enabled (bit DCF.EOS is set).

* Formerly called OOP.

PSV - Pan/Scan Vector

15	9	8	0
V[7:1]		H[8:0]	

Address : 3A-3B
Type : R/W
Reset State : 0
Synchronization : VSYNC

Description

V[7:1] Vertical Offset. The vertical component of the pan/scan vector is programmed into V[7:1], in units of 2 field lines, or 4 frame lines. For example, if V[7:1] is set to the value 16, then the 32 top lines of every field would not be output to the display.

H[8:0] Horizontal Offset. These bits are set up with the integer part of the horizontal component of the pan/scan vector. This number defines, in the decoded picture, the location of the first displayed luminance sample relative to the first luminance sample in the line. The LSO and CSO registers are set up with the fractional part of the pan vector, as follows :
PSV = ⌊pan vector⌋
where "⌊x⌋" indicates the integer part of x.
LSO = 256 x (pan vector - PSV)
CSO = LSO/2 (+ 1 if PSV is odd)
Refer to section IX.4.1, "Sample Rate Converter" for more details.

QMW - Quantization Matrix Data

7	0
Q[7:0]	

Address : 3C or 3D
Type : W
Reset State : Undefined
Synchronization : None (not a register)

Description

To either of these addresses are written the quantization coefficients in the order in which they appear in the bitstream, i.e. in zig-zag order. Which matrix (intra or non-intra) is written is defined by bits CMD.QMN and CMD.QMI.

There are no built-in default quantization matrices.

7929237 0073022 2T7

RFP - Reconstructed Frame Pointer

13	0
RFP[13:0]	

Address : 1C-1D
Type : R/W
Reset State : 0
Synchronization : DSYNC

Description

This register holds the start address of the reconstructed (decoded) frame picture buffer, defined in units of 32 words.

If this buffer is also to be used for display, then the bottom two bits, RFP[1] and RFP[0], must be zero. In this case the reconstructed frame buffer is constrained to start at an address which is a multiple of 128 words.

STA - Status

14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SER	BMI	HFF	RFF	WFE	PID	PER	PSD	VST	VSB	BBE	BBF	HFE	BFF	SCH

Address : 08-09
Type : R
Reset State : 0

After clocks have been enabled, the state changes to :

SER = 0
BMI = 1
HFF = 0
RFF = 1
WFE = 1
PID = 1
PER = 0
PSD = 0
VST = 0
VSB = 0
BBE = 1
BBF = 1
HFE = 1
BFF = 0
SCH = 0

Description

This register contains a set of bits which represent the status of the decoder at any instant. Any change from 0 to 1 of any of these bits sets the corresponding bit of the ITS register, and can thus potentially cause an interrupt.

The status vector is sampled internally at the start of the read cycle accessing the most significant byte of STA (address 08). VST, VSB and PSD are pulses and are unlikely ever to be read as a 1.

The status bits have the following significance :

SER — Severe Error. This bit is set when more than the programmed number of macroblocks (defined by DFS) have been decoded, either due to a data or a programming error. Decoding is halted automatically when this error condition is detected. This bit is reset by all 3 types of reset.

BMI — Block Move Idle. This bit is set when a block move operation has terminated. It is automatically reset at the start of a block move.

HFF — Header FIFO Full. This bit is set when the header FIFO contains at least 66 bytes.

RFF — Read FIFO Full. This bit is set when the memory read FIFO contains 16 bytes.

WFE — Write FIFO Empty. This bit is set when the memory write FIFO is empty.

PID — Pipeline Idle. This bit is set when the STi3500A is not in the course of decoding a picture, i.e. when the pipeline is inactive. It becomes low when the decoding of a picture starts and high when picture decoding is complete.

PER — Pipeline Error. This bit is set when due to a data error, more than 64 coefficients are reconstructed for a block. If bit CTL.EPR is set, a pipeline reset is generated. This bit is reset by all 3 types of reset.

PSD — Pipeline Starting to Decode. This bit is set for a short period at the instant the pipeline starts decoding a picture.

VST — VSYNC Top. This bit is set for a short time at the beginning of the top field, corresponding to the falling edge of the B/$\overline{T}$ signal.

VSB — VSYNC Bottom. This bit is set for a short time at the beginning of the bottom field, corresponding to the rising edge of the B/$\overline{T}$ signal.

BBE — Bit Buffer Empty. This bit is set when the bit buffer contains no data.

BBF — Bit Buffer Full. This bit is set when the bit buffer level (= BBL) is greater than or equal to the value loaded into the BBT register.

SGS-THOMSON MICROELECTRONICS

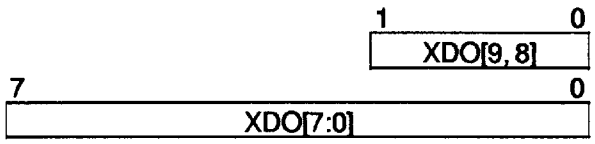
7929237 0073023 133

HFE Header FIFO Empty. This bit is set when the header FIFO is empty.

BFF Compressed Data (bitstream) FIFO Full. This bit is set when the CD FIFO is full. This bit is equivalent to the signal $\overline{\text{CDREQ}}$.

SCH Start Code Hit. This bit is set whenever the first 16-bit word available in the header FIFO contains one of the start codes recognised (see section VII.2, "Start Code Detection"). While data is being read from the header FIFO, this bit can be tested to determine whether the next word contains a start code.

XDO - Display X Offset



Address : 2D
Type : R/W
Reset State : 0
Synchronization : VSYNC

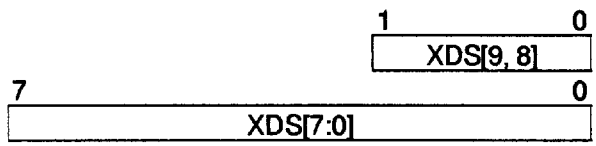
Description

This register is set up with a number defining the first active video sample of each line. Preceding pels are output as black (unless OSD is defined for this region).

When bit DCF.XYE is set, XDO[9,8] is accessible. When bit DCF.XYE is reset, XDO[7:0] is accessible.

XDO programming is described in section IX.3, "Setting up the Display".

XDS - Display X End



Address : 2F
Type : R/W
Reset State : 0
Synchronization : VSYNC

Description

This register is set up with a number defining the last active video sample of each line. Succeeding pels are output as black (unless OSD is defined for this region).

When bit DCF.XYE is set, XDS[9,8] is accessible. When bit DCF.XYE is reset, XDS[7:0] is accessible.

Note : When bit CTL.35A is set, care must be taken to set up XDS[9,8], since the default value is 0, while in the STi3500, in which these bits are not programmable, this value is 3.

XDS programming is described in section IX.3, "Setting up the Display".

YDO - Display Y Offset

Address : 2C
Type : R/W
Reset State : 0
Synchronization : VSYNC

Description

This register is set up with a number defining the first active video line of a field. Preceding lines are output as black (unless OSD is defined for this region).

YDO programming is described in section IX.3, "Setting up the Display".

YDS - Display Y End

Address : 2E
Type : R/W
Reset State : 0
Synchronization : VSYNC

Description

This register is set up with a number defining the last active video line of a field. Succeeding lines are output as black (unless OSD is defined for this region).

YDS programming is described in section IX.3, "Setting up the Display".

7929237 0073024 07T

XIII - ELECTRICAL CHARACTERISTICS

XIII.1 - Absolute Maximum Ratings

	Parameter	Min.	Max.	Unit
V_{DD}	Power Supply	-	6	V
V_I, V_O	Voltages on Input and Output Pins	-1	V_{DD} + 0.5	V
T_{stg}	Storage Temperature	-65	150	°C
T_{oper}	Ambient Operating Temperature	0	70	°C

XIII.2 - DC Electrical Characteristics

Operating conditions : V_{DD} = 5V ±5%, T_A = 0 to 70°C unless otherwise specified.

	Parameter	Test Conditions	Min.	Typ.	Max.	Unit
V_{DD}	Operating Voltage		4.75		5.25	V
I_{DD}	Average Power Supply Current	C_{LOAD} = 50pF on all outputs $f_{primary}$ = 56MHz, all inputs at V_{DD} or 0V $f_{primary}$ = 0Hz DRAM and video interfaces disabled $f_{primary}$ = 56MHz, all inputs at V_{DD} or 0V $f_{primary}$ = 0Hz			200	mA
V_{IL}	Input Logic Low Voltage (except CLK)				0.8	V
V_{IH}	Input Logic High Voltage (except CLK, PIXCLK)		2			V
V_{IL}(CLK)	Input Logic Low Voltage (CLK)				0.6	V
V_{IH}(CLK)	Input Logic High Voltage (CLK)		3			V
V_{IH} (PIXCLK)	Input Logic High Voltage (PIXCLK)		2.5			V
	Input Leakage Current Inputs I/Os	V_{DD} = 5.25V, 0 ≤ V_{IN} ≤ V_{DD}	 -5 -1		 +5 +1	 µA µA
V_{OL}	Output Logic Low Voltage	V_{DD} = 5.25V, I_{LOAD} = 500µA			0.4	V
V_{OH}	Output Logic High Voltage	V_{DD} = 4.75V, I_{LOAD} = -500µA	2.4			V
C_{IN}	Input Capacitance	V_{offset} = 2.5V, f = 1MHz			10	pF

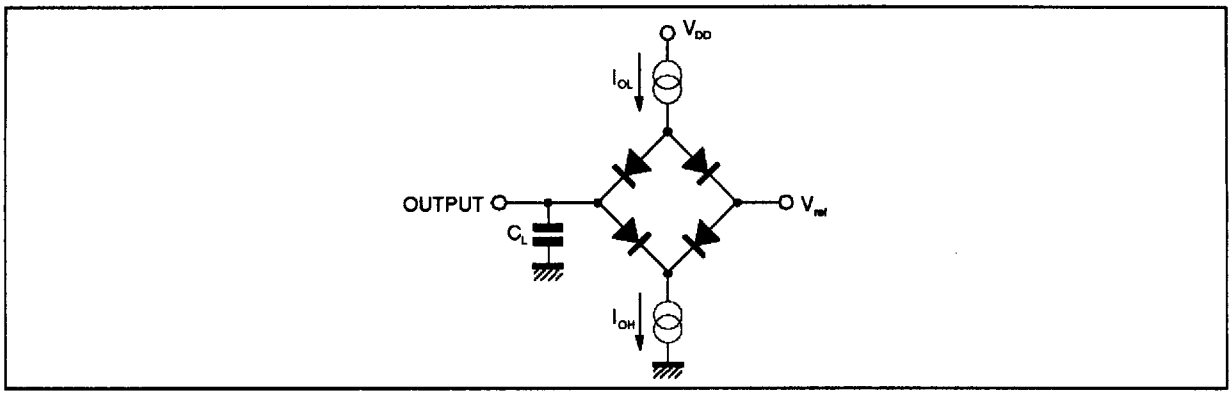
SGS-THOMSON MICROELECTRONICS

7929237 0073025 T06

XIII.3 - AC Electrical Characteristics

Test Conditions : V_{DD} = 5V ±5%, T_A = 0 to 70°C

Figure 63 : Test Load Circuit



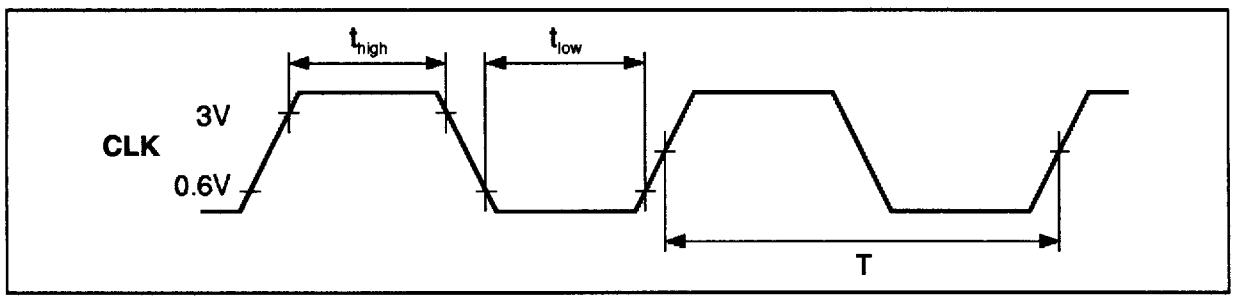
Test Loads

Output	I_{OL}	I_{OH}	C_L	V_{ref}
YC7-YC0, D7-D0, $\overline{CDREQ}$, $\overline{WAIT}$	500µA	500µA	50pF	1.5V
$\overline{IRQ}$	5mA	0	50pF	3.5V
$\overline{OE}$	200µA	200µA	56pF	1.5V
$\overline{WE}$, $\overline{CAS}$	200µA	200µA	112pF	1.5V
$\overline{AA8}$ - $\overline{AA0}$	200µA	200µA	40pF	1.5V
$\overline{RAS1}$, $\overline{RAS0}$, DD63-DD0	200µA	200µA	30pF	1.5V

XIV - TIMING DIAGRAMS

XIV.1 - Primary Clock

Figure 64 : Primary Clock



Symbol	Parameter	Min.	Max.	Units
T	Primary clock period (1,2)	17.8		ns
t_{high}	Clock high time	6		ns
t_{low}	Clock low time	6		ns

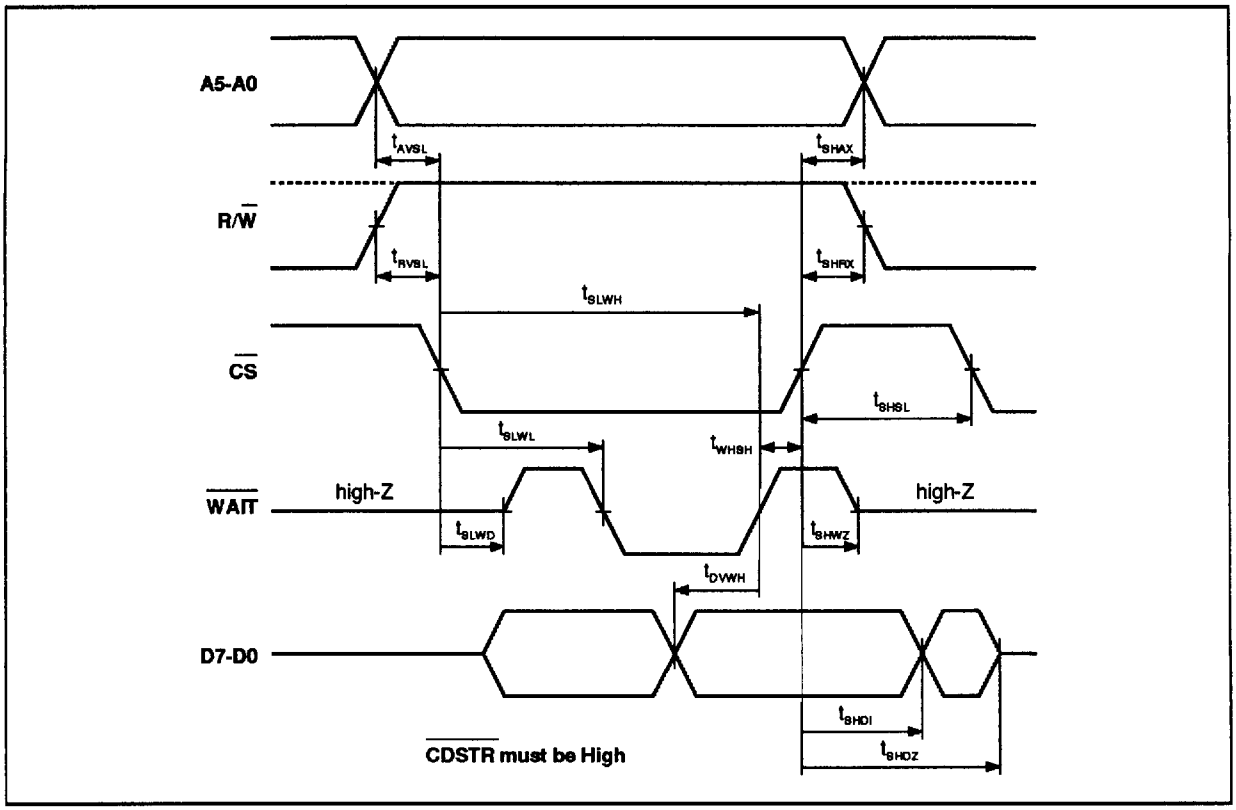
Notes : 1. This corresponds to a maximum primary clock frequency of 56MHz.

2. No maximum value is given since, although the STi3500A is fully static, large values of T would conflict with performance and memory refresh constraints.

7929237 0073026 942

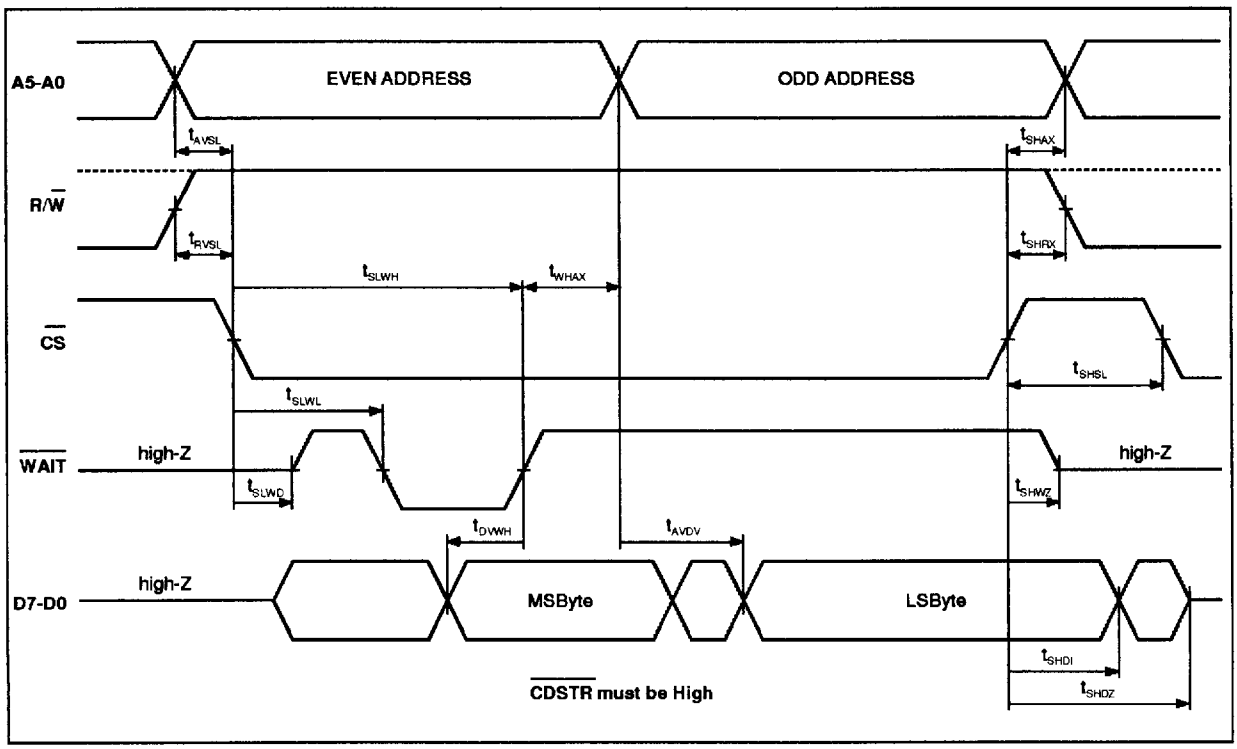
XIV.2 - Register Read and Write Cycles

Figure 65 : Single-Byte Register Read Cycle



3500A-67.EPS

Figure 66 : Two-Byte Register Read Cycle

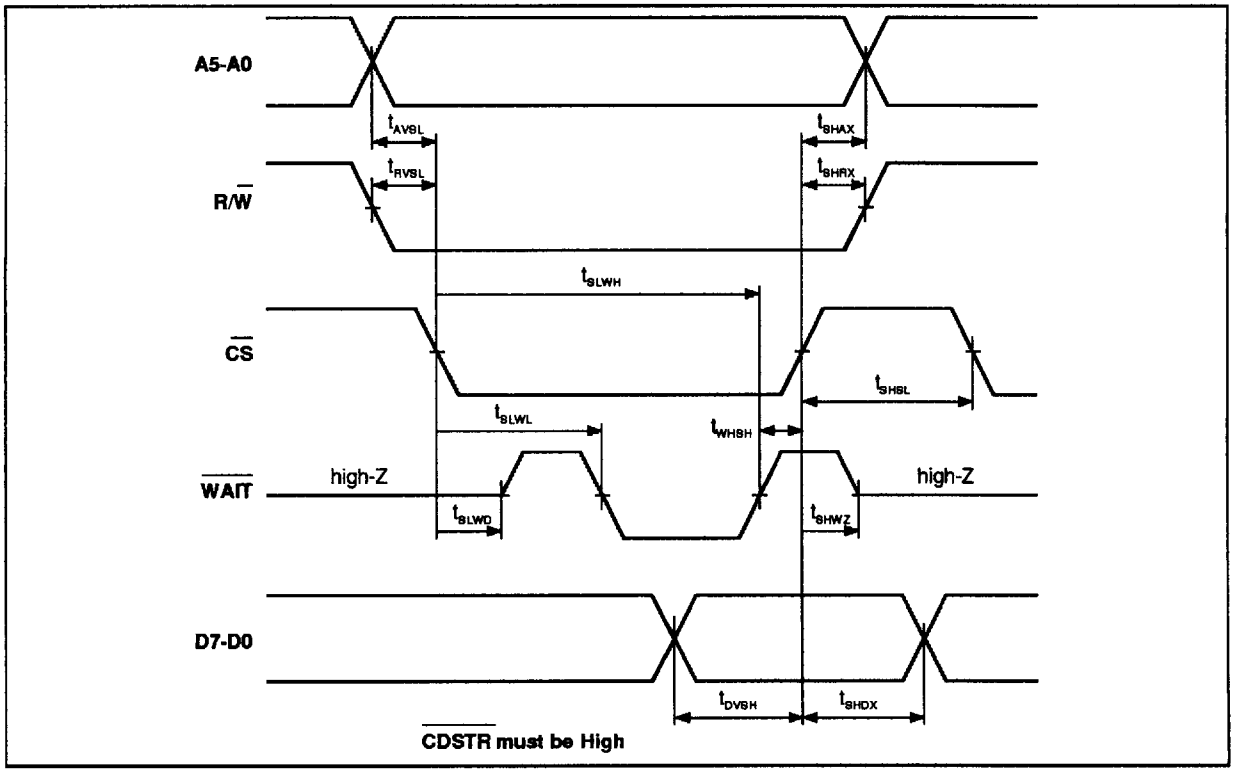


3500A-68.EPS

SGS-THOMSON MICROELECTRONICS

7929237 0073027 889

Figure 67 : Register Write Cycle



Register Read and Write Cycles (1)

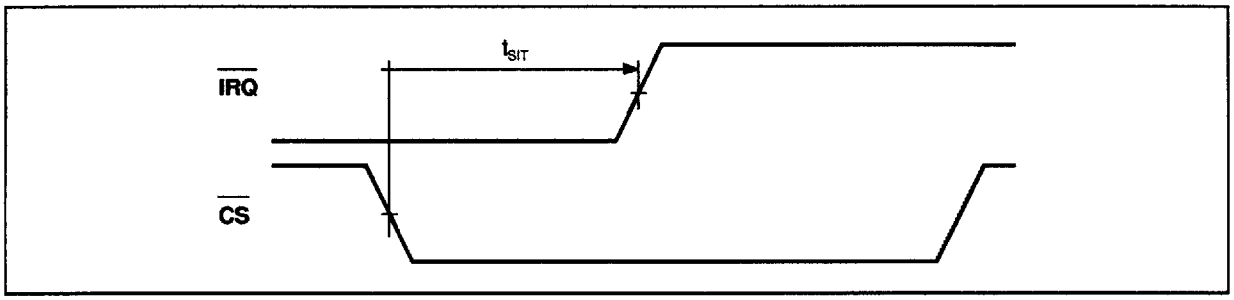
Symbol	Parameter	Min.	Max.	Units
t_{AVSL}	Address to $\overline{CS}$ set-up time	5		ns
t_{RVSL}	R/W to $\overline{CS}$ set-up time	5		ns
t_{SHAX}	Address from $\overline{CS}$ hold time	5		ns
t_{SHRX}	R/W from $\overline{CS}$ hold time	5		ns
t_{SLWD}	$\overline{CS}$ low to $\overline{WAIT}$ on		10	ns
t_{SLWL}	$\overline{CS}$ low to $\overline{WAIT}$ low		25	ns
t_{SLWH}	$\overline{CS}$ low to $\overline{WAIT}$ high		50	ns
t_{SHWZ}	$\overline{CS}$ high to $\overline{WAIT}$ off (hi-Z)		10	ns
t_{SHSL}	$\overline{CS}$ high to $\overline{CS}$ or $\overline{CDSTR}$ low again	20		ns
t_{WHSH}	$\overline{CS}$ high from $\overline{WAIT}$ high hold time	0		ns
t_{DVWH}	Data valid before $\overline{WAIT}$ high	5		ns
t_{SHDI}	$\overline{CS}$ high to data invalid	10		ns
t_{SHDZ}	$\overline{CS}$ high to data off (hi-Z)		20	ns
t_{WHAX}	Address from $\overline{WAIT}$ high hold time (first byte)	0		ns
t_{AVDV}	Address valid to data valid (second byte)		20	ns
t_{DVSH}	Data valid to $\overline{CS}$ high set-up time	15		ns
t_{SHDX}	Data from $\overline{CS}$ high hold time	5		ns

Note : 1. Timing measurements are made with respect to thresholds of 1.5V.

7929237 0073028 715

XIV.3 - Interrupt Acknowledge

Figure 68 : Interrupt Acknowledge

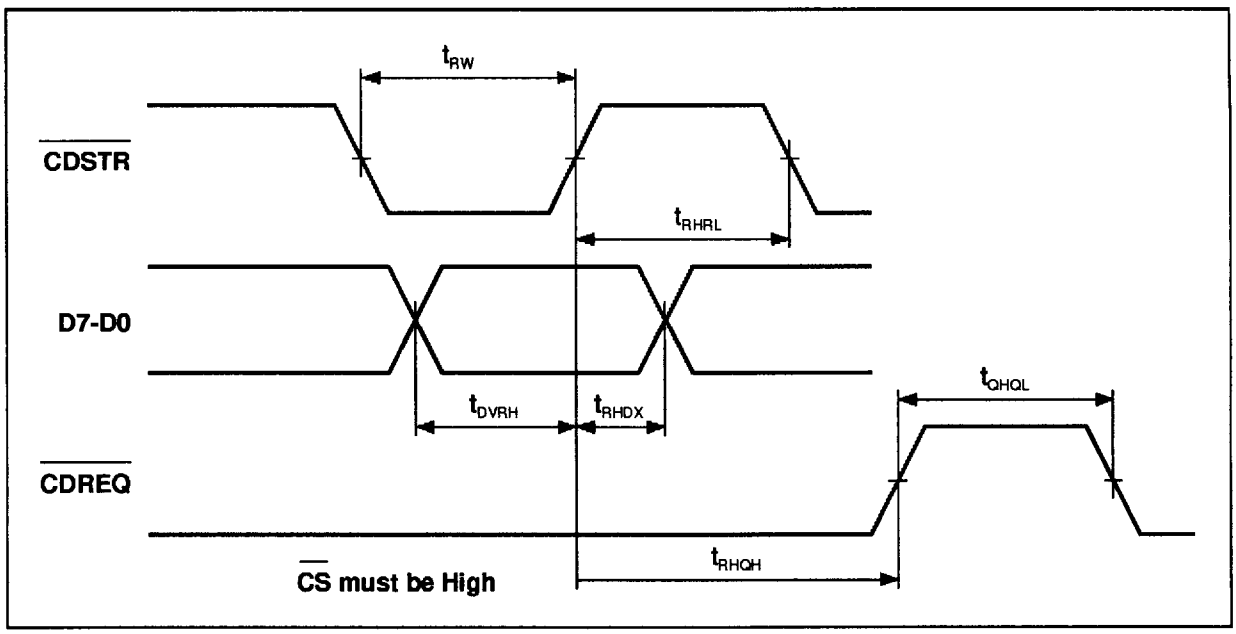


Symbol	Parameter	Min.	Max.	Units
t_{SIT}	$\overline{CS}$ low to $\overline{IRQ}$ high (1)		50	ns

Note : 1. Timing measurements are made with respect to thresholds of 1.5V.

XIV.4 - Compressed Data Write Cycle

Figure 69 : Compressed Data Write



SGS-THOMSON MICROELECTRONICS

7929237 0073029 651

Compressed Data Write (1)

Symbol	Parameter	Min.	Max.	Units
t_{RW}	$\overline{CDSTR}$ pulse width	15		ns
t_{RHRL}	$\overline{CDSTR}$ high to $\overline{CDSTR}$ or $\overline{CS}$ low again (2)	20		ns
t_{DVRH}	D7 - D0 set-up time to $\overline{CDSTR}$ rising edge	15		ns
t_{RHDX}	D7 - D0 hold time from $\overline{CDSTR}$ rising edge	5		ns
t_{RHQH}	$\overline{CDREQ}$ high from $\overline{CDSTR}$ high (3,4)		40	ns
t_{QHQL}	$\overline{CDREQ}$ high to low again : Normal mode 8- Mbit mode		 180 300	 primary clock cycles

Notes : 1. Measurements are made with respect to thresholds of 1.5V.
2. If $\overline{CDREQ}$ handshake not used.
3. $\overline{CDREQ}$ goes high if the write caused CD FIFO to become almost full.
4. $\overline{CDREQ}$ becoming high indicates that only 3 more bytes can be written. For the CD FIFO to become empty there is a maximum delay of 350 primary clock cycles (or 580 in 8-Mbit mode) after the rising edge of CDSTR.

XIV.5 - DRAM Interface

Test Conditions

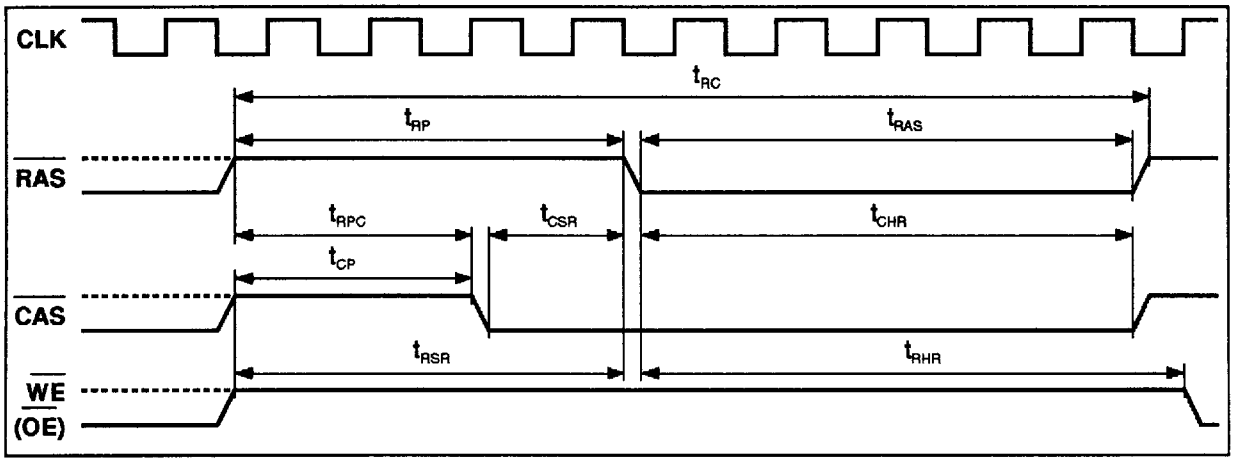
Unless otherwise stated, the worst case values correspond to maximum loading, i.e. 2 banks each with 4 memory packages. Also, unless otherwise stated, it is assumed that the memories have dual $\overline{WE}$ and $\overline{CAS}$ inputs each of which are linked.

Measurement Thresholds

All timing measurements are made with respect to thresholds V_{IH} = 4V, V_{IL} = 0.2V, V_{OH} = 2.4V, V_{OL} = 0.8V.

7929237 0073030 373

Figure 70 : $\overline{CAS}$ Before $\overline{RAS}$ Refresh Cycle



$\overline{CAS}$ before $\overline{RAS}$ Refresh Cycle (1)

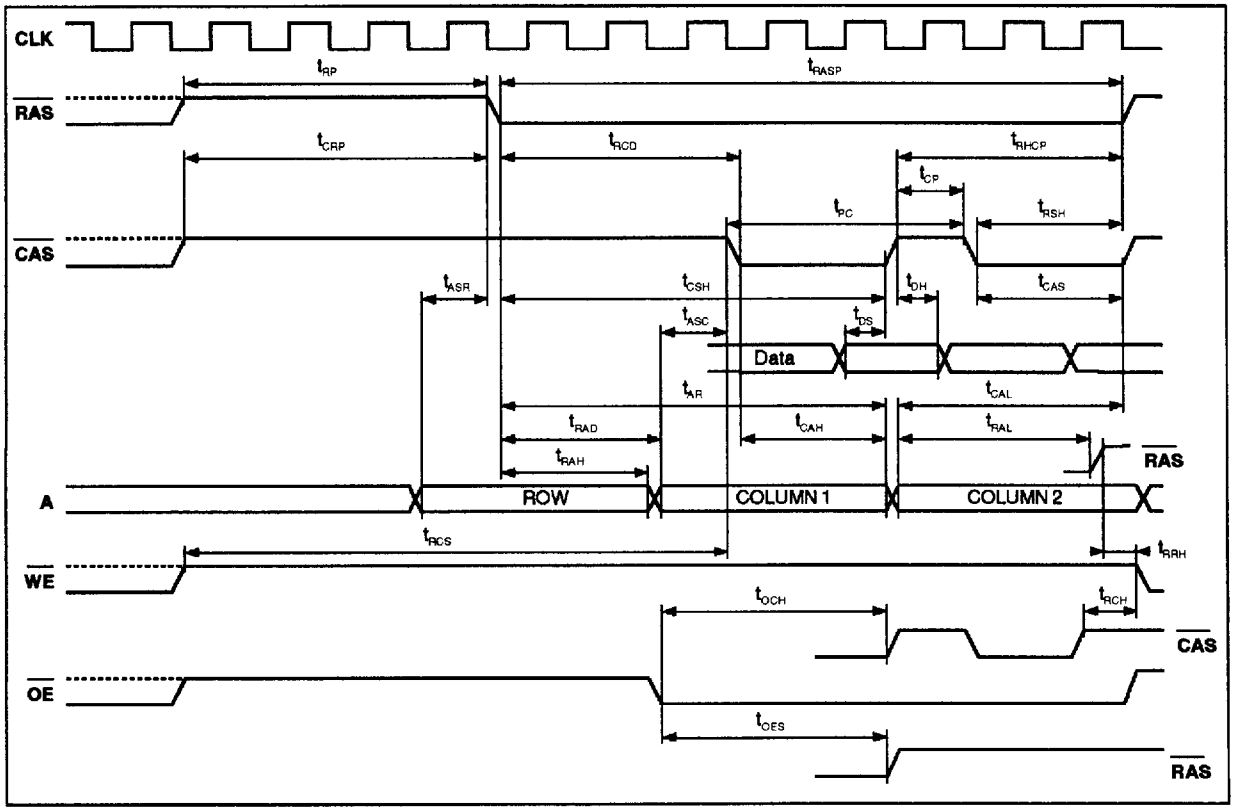
Symbol	Parameter	Min.	Max.	Units
t_{RC}	Cycle time	9T		ns
t_{RP}	$\overline{RAS}$ precharge time	4T-7		ns
t_{RAS}	$\overline{RAS}$ pulse width	5T-7		ns
t_{RPC}	$\overline{RAS}$ to $\overline{CAS}$ precharge time (2)	3T-8		ns
t_{CSR}	$\overline{CAS}$ to $\overline{RAS}$ set-up time	T-6		ns
t_{CHR}	$\overline{CAS}$ from $\overline{RAS}$ hold time (2)	5T-8		ns
t_{CP}	$\overline{CAS}$ precharge time	3T-6		ns
t_{RSR}	Read command to $\overline{RAS}$ set-up time (3)	4T-10		ns
t_{RHR}	Read command from $\overline{RAS}$ hold time (4,5)	11T-5		ns

Notes : 1. T is the primary clock period.
2. Worst case is with one bank of memory, and each package having a single $\overline{CAS}$ Pin.
3. Sometimes referred to as t_{WSR}.
4. Sometimes referred to as t_{WHR}.
5. Worst case is with one bank of memory, and each package having a single $\overline{WE}$ Pin.

SGS-THOMSON MICROELECTRONICS

7929237 0073031 20T

Figure 71 : Page Mode Read Cycle



7929237 0073032 146

Page Mode Read Cycle (1)

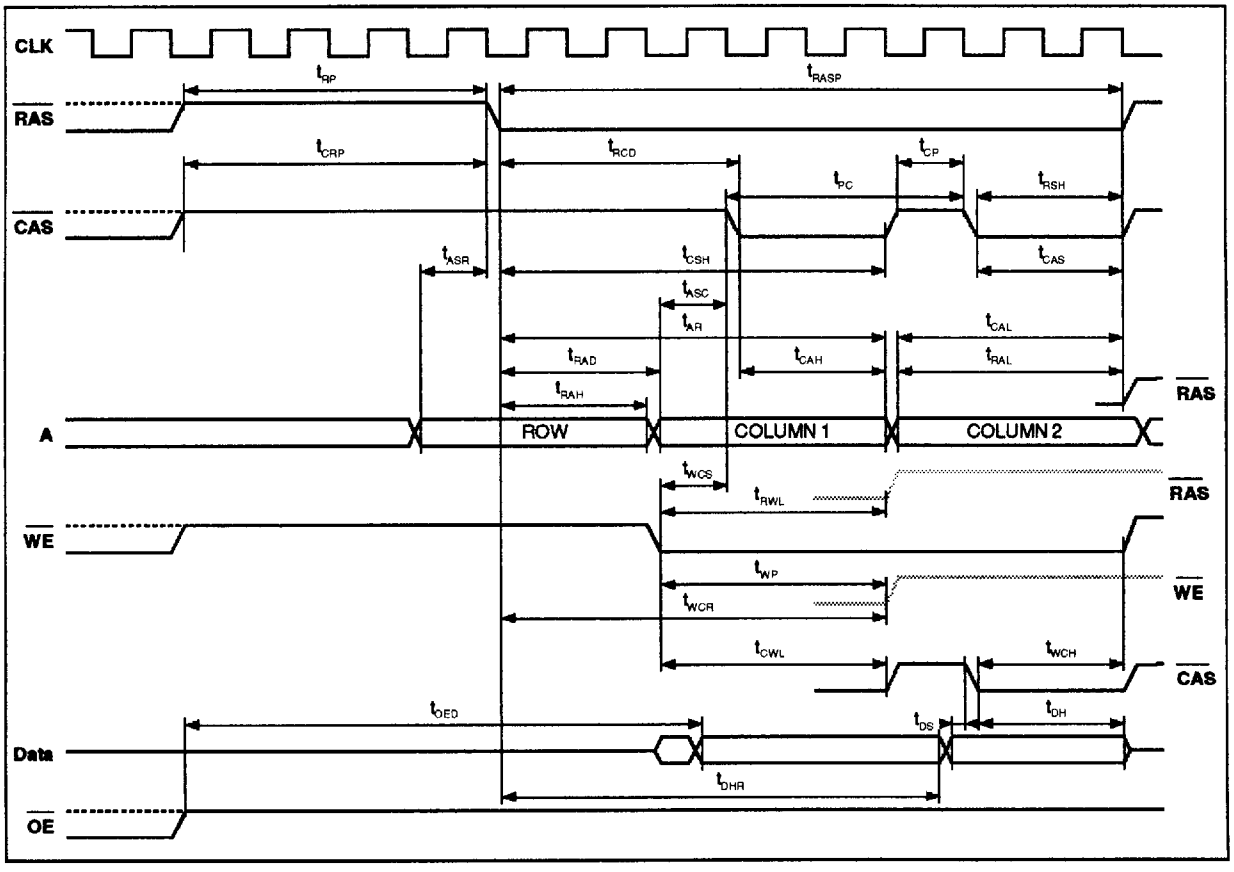
Symbol	Parameter	Min.	Max.	Units
t_{RC}	Read cycle time	9T		ns
t_{RP}	$\overline{RAS}$ precharge time	4T-7		ns
t_{RASP}	$\overline{RAS}$ pulse width	5T-7		ns
t_{CRP}	$\overline{CAS}$ to $\overline{RAS}$ precharge time	4T-7		ns
t_{RCD}	$\overline{RAS}$ low to $\overline{CAS}$ low delay time (2)	3T-4	3T+3	ns
t_{PC}	Fast page mode read cycle time	3T		ns
t_{CP}	$\overline{CAS}$ precharge time	T-6		ns
t_{RHCP}	$\overline{RAS}$ hold time after $\overline{CAS}$ precharge (3)	3T-6		ns
t_{RSH}	$\overline{RAS}$ hold time after $\overline{CAS}$	2T-6		ns
t_{CSH}	$\overline{CAS}$ hold time after $\overline{RAS}$ (2)	5T-8		ns
t_{CAS}	$\overline{CAS}$ pulse width	2T-6		ns
t_{ASR}	Row address set-up time to $\overline{RAS}$	T-13		ns
t_{RAH}	Row address hold time after $\overline{RAS}$ (4)	2T		ns
t_{ASC}	Column address set-up time to $\overline{CAS}$ (5)	T-14		ns
t_{RAD}	Column address delay time from $\overline{RAS}$ (4)	2T	2T+15	ns
t_{CAH}	Column address hold time from $\overline{CAS}$	2T		ns
t_{AR}	Column address hold time from $\overline{RAS}$ (4)	5T		ns
t_{CAL}	Column address to $\overline{CAS}$ lead time	3T-14		ns
t_{RAL}	Column address to $\overline{RAS}$ lead time	3T-13		ns
t_{DS}	Data in set-up time to $\overline{CAS}$ rising edge	-2		ns
t_{DH}	Data in hold time from $\overline{CAS}$ rising edge	7.5		ns
t_{RCS}	Read command to $\overline{CAS}$ set-up time	7T-15		ns
t_{RCH}	Read command from $\overline{CAS}$ hold time	6T		ns
t_{RRH}	Read command from $\overline{RAS}$ hold time (6)	6T		ns
t_{OCH}	$\overline{CAS}$ from $\overline{OE}$ hold time	3T-7		ns
t_{OES}	$\overline{RAS}$ from $\overline{OE}$ hold time	3T-7		ns

Notes : 1. T is the primary clock period.
2. Worst case for min. value is with one bank of memory, and each package having a single $\overline{CAS}$ Pin.
3. Sometimes referred to as t_{CPRH}.
4. Worst case for min. value is with one bank of memory.
5. Worst case is with two banks of memory, and each package having a single $\overline{CAS}$ Pin.
6. Worst case is with one bank of memory, and each package having a single $\overline{WE}$ Pin.

SGS-THOMSON MICROELECTRONICS

7929237 0073033 082

Figure 72 : Page Mode Early Write Cycle



7929237 0073034 T19

Page Mode Early Write Cycle (1)

Symbol	Parameter	Min.	Max.	Units
t_{RP}	$\overline{RAS}$ precharge time	4T-7		ns
t_{RASP}	$\overline{RAS}$ pulse width	5T-7		ns
t_{CRP}	$\overline{CAS}$ to $\overline{RAS}$ precharge time	4T-7		ns
t_{RCD}	$\overline{RAS}$ low to $\overline{CAS}$ low delay time (2)	3T-4	3T+3	ns
t_{PC}	Fast page mode read cycle time	3T		ns
t_{CP}	$\overline{CAS}$ precharge time	T-6		ns
t_{RSH}	$\overline{RAS}$ hold time after $\overline{CAS}$	2T-6		ns
t_{CSH}	$\overline{CAS}$ hold time after $\overline{RAS}$ (2)	5T-8		ns
t_{CAS}	$\overline{CAS}$ pulse width	2T-6		ns
t_{ASR}	Row address set-up time to $\overline{RAS}$	T-13		ns
t_{RAH}	Row address hold time after $\overline{RAS}$ (3)	2T		ns
t_{ASC}	Column address set-up time to $\overline{CAS}$ (4)	T-14		ns
t_{RAD}	Column address delay time from $\overline{RAS}$ (3)	2T	2T+15	ns
t_{CAH}	Column address hold time from $\overline{CAS}$	2T		ns
t_{AR}	Column address hold time from $\overline{RAS}$ (3)	5T		ns
t_{CAL}	Column address to $\overline{CAS}$ lead time	3T-14		ns
t_{RAL}	Column address to $\overline{RAS}$ lead time	3T-13		ns
t_{WCS}	Write command set-up time to $\overline{CAS}$	T-14		ns
t_{RWL}	Write command to $\overline{RAS}$ lead time	3T-15		ns
t_{CWL}	Write command to $\overline{CAS}$ lead time	3T-14		ns
t_{WCR}	Write command hold time after $\overline{RAS}$ (5)	5T-8		ns
t_{WCH}	Write command hold time after $\overline{CAS}$	2T-6		ns
t_{WP}	Write command pulse width	3T-13		ns
t_{DS}	Data in set-up time to $\overline{CAS}$ falling edge	T-18		ns
t_{DH}	Data in hold time from $\overline{CAS}$ falling edge	2T		ns
t_{OED}	$\overline{OE}$ high before data valid	6T-10		ns
t_{OEH}	$\overline{OE}$ high hold time after write command ($\overline{WE}$ falling edge) (6)	9T-14		ns
t_{DHR}	Data hold time after $\overline{RAS}$ (3)	5T		ns

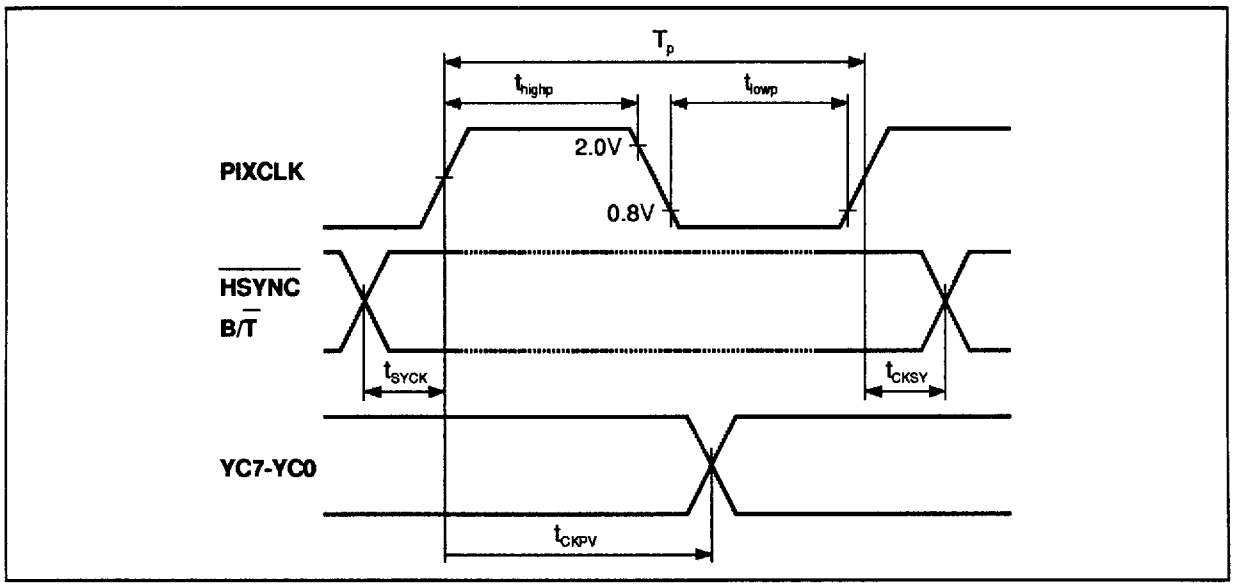
Notes : 1. T is the primary clock period.
2. Worst case for min. value is with one bank of memory, and each package having a single $\overline{CAS}$ Pin.
3. Worst case for min. value is with one bank of memory.
4. Worst case is with two banks of memory, and each package having a single $\overline{CAS}$ Pin.
5. Worst case is with one bank of memory, and each package having a single $\overline{WE}$ Pin.
6. Not shown on timing diagram.

SGS-THOMSON MICROELECTRONICS

7929237 0073035 955

XIV.6 - Video Interface Timing

Figure 73 : Video Interface



Video Interface (1)

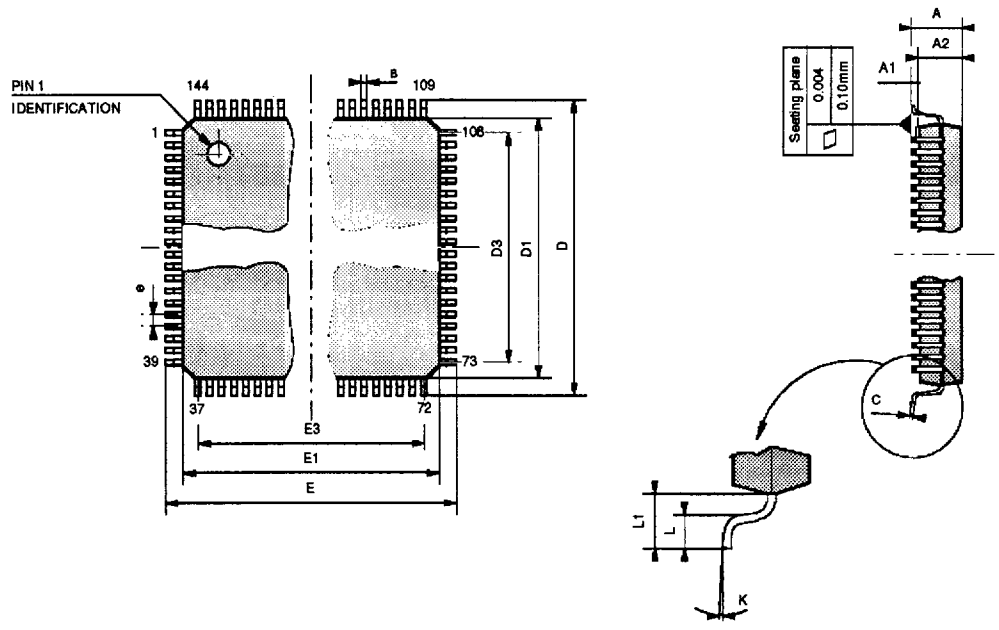
Symbol	Parameter	Min.	Max.	Units
T_p	Pel clock period (1)	33.3		ns
t_{highp}	Pel clock high time	10		ns
t_{lowp}	Pel clock low time	10		ns
t_{SYCK}	$\overline{HSYNC}$, B/$\overline{T}$ set-up time to PIXCLK rising edge	7		ns
t_{CKSY}	$\overline{HSYNC}$, B/$\overline{T}$ from PIXCLK rising edge hold time	4		ns
t_{CKPV}	PIXCLK to YC7 - YC0 valid	4	25	ns

Notes : 1. This corresponds to a PIXCLK frequency of 30MHz.

7929237 0073036 891

XV - PACKAGE MECHANICAL DATA

144 PINS - PLASTIC QUAD FLAT PACK



PMPQF144.EPS

Dim.	mm			inches		
	Min	Typ	Max	Min	Typ	Max
A			4.07			0.160
A1	0.25			0.010		
A2	3.17	3.42	3.67	0.125	0.133	0.144
B	0.22		0.38	0.009		0.015
C	0.13		0.23	0.005		0.009
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
D3		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
E3		22.75			0.896	
L	0.65	0.80	0.95	0.026	0.031	0.037
L1		1.60			0.063	
K	0^0 (min.), 7^0 (max.)					

PQFP144.TBL

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 licence 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 information previously supplied. SGS-THOMSON Microelectronics products are not authorized for use as critical components in life support devices or systems without express written approval of SGS-THOMSON Microelectronics.

© 1995 SGS-THOMSON Microelectronics - All Rights Reserved

Purchase of I^2C Components of SGS-THOMSON Microelectronics, conveys a license under the Philips I^2C Patent. Rights to use these components in a I^2C system, is granted provided that the system conforms to the I^2C Standard Specifications as defined by Philips.

SGS-THOMSON Microelectronics GROUP OF COMPANIES

Australia - Brazil - China - France - Germany - Hong Kong - Italy - Japan - Korea - Malaysia - Malta - Morocco
The Netherlands - Singapore - Spain - Sweden - Switzerland - Taiwan - Thailand - United Kingdom - U.S.A.

File Name : STI3500A

SGS-THOMSON MICROELECTRONICS

7929237 0073037 728