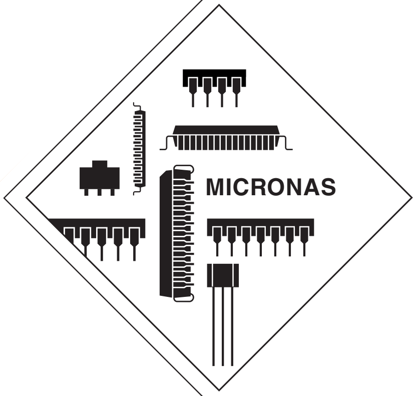
ADVANCE INFORMATION

VCT 38xxA Video/Controller/Teletext IC Family



Edition Oct. 17, 2000
6251-518-1AI

MICRONAS

Contents

Contents, continued

Contents, continued

Contents, continued

Contents, continued

Video/Controller/Teletext IC Family

Release Note: This data sheet describes functions and characteristics of the VCT 38xxA-B2.

1. Introduction

The VCT 38xxA is an IC family of high-quality single-chip TV processors. Modular design and a submicron technology allow the economic integration of features in all classes of TV sets. The VCT 38xxA family is based on functional blocks contained and approved in existing products like VDP 3120B, TPU 3050S, and CCZ 3005K.

Each member of the family contains the entire video, display and deflection processing for 4:3 and 16:9 50/60-Hz TV sets. The integrated microcontroller is supported by a powerful OSD generator with integrated teletext acquisition which can be upgraded with on-chip page memory. With volume control and audio input select the basic audio features for mono TV sets are integrated. An overview of the VCT 38xxA single-chip TV processor family is given in Fig. 1–1 on page 7.

The VCT 38xxA family offers a rich feature set, covering the whole range of state-of-the-art 50/60-Hz TV applications.

VCT 38xxA Family	8-Bit Microcontroller 96 kB ROM, 1 kB RAM Flash Option Color Decoder Tube Control OSD Generator Audio Control	Picture Improvements (Color Transient Improv., Soft Limiter, Black-Level Expander)	Adaptive Comb Filter	Panorama Scaler	1 Page Teletxt	10 Page Teletxt	ext. Page Memory	ext. Prog. Memory
VCT 3801A	✔						PMQFP128	PMQFP128
VCT 3802A	✔	✔						
VCT 3803A	✔	✔	✔					
VCT 3804A	✔	✔	✔	✔				
VCT 3811A	✔				✔			
VCT 3831A	✔					✔		
VCT 3832A	✔	✔				✔		
VCT 3833A	✔	✔	✔			✔		
VCT 3834A	✔	✔	✔	✔		✔		

Fig. 1–1: VCT 38xxA family overview

1.1. Features

1.1.1. Video Features

- four composite video inputs, two S-VHS inputs
- analog YC_rC_b input
- composite video monitor
- multistandard color decoder (1 crystal)
- multistandard sync decoder
- black-line detector
- adaptive 2H comb filter Y/C separator
- horizontal scaling (0.25 to 4)
- Panoramavision
- black-level expander
- dynamic peaking
- soft limiter (gamma correction)
- color transient improvement
- programmable RGB matrix
- analog RGB/Fastblank input
- half-contrast switch
- picture frame generator
- scan velocity modulation output
- high-performance H/V deflection
- angle and bow correction
- separate ADC for tube measurements
- EHT compensation

1.1.2. Microcontroller Features

- 8-bit, 10-MHz CPU (65C02)
- 96 kB program ROM on chip
- 1 kB program RAM on chip
- memory banking
- 16-input, 16-level interrupt controller
- patch modul for 10 ROM locations
- two 16-bit reloadable timers
- capture compare modul
- watchdog timer
- 14-bit PWM for voltage synthesis
- four 8-bit PWMs
- 10-bit ADC with 15:1 input MUX
- I^2C bus master interface
- 24 programmable I/O ports

1.1.3. OSD Features

- 3 kB OSD RAM on chip
- WST level 1.5 compliant
- WST level 2 parallel attributes
- 32 foreground/background colors
- programmable color look-up table
- 1024 mask programmable characters
- 24 national languages
  (Latin, Cyrillic, Greek, Arabic, Farsi, Hebrew)
- character matrix 8x8, 8x10, 8x13, 10x8, 10x10, 10x13
- vertical soft scroll
- 4-color mode for user font

1.1.4. Teletext Features

- four programmable video inputs
- acquisition is independent from display part
- adaptive data slicer
- signal quality detection
- WST, PDC, VPS, and WSS acquisition
- high-level command language
- EPG, FLOF, and TOP support
- 10 pages memory on chip
- up to 500 pages with external SRAM

1.1.5. Audio Features

- three mono inputs
- two mono outputs
- programmable channel select
- volume control for one mono channel

1.1.6. General Features

- submicron CMOS technology
- low-power standby mode
- single 20.25-MHz crystal
- 64-pin PSDIP package
- 128-pin PMQFP package
- emulator chip for software development

1.2. Chip Architecture

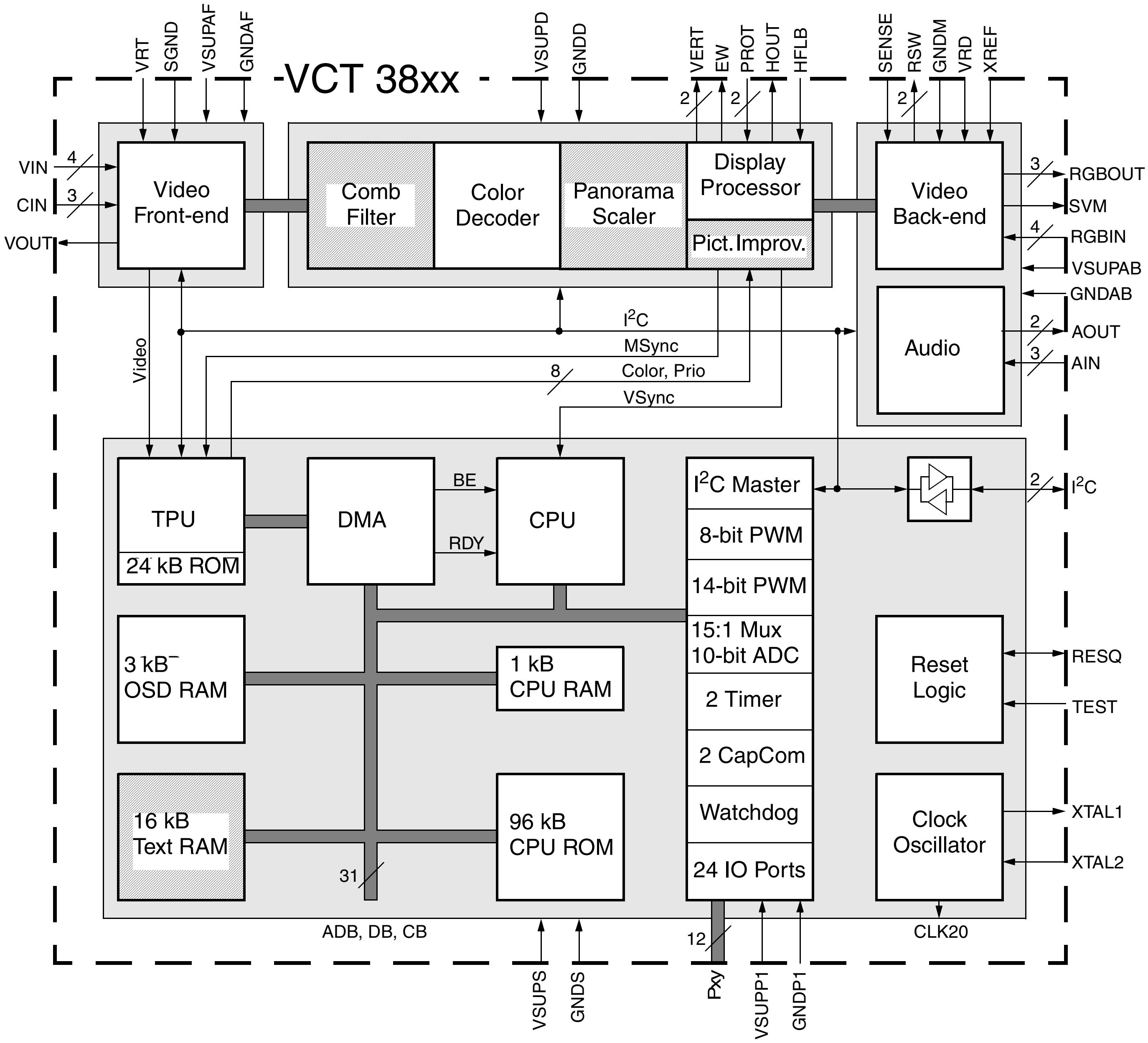


Fig. 1–2: Block diagram of the VCT 38xxA (shaded blocks are optional)

1.3. System Application

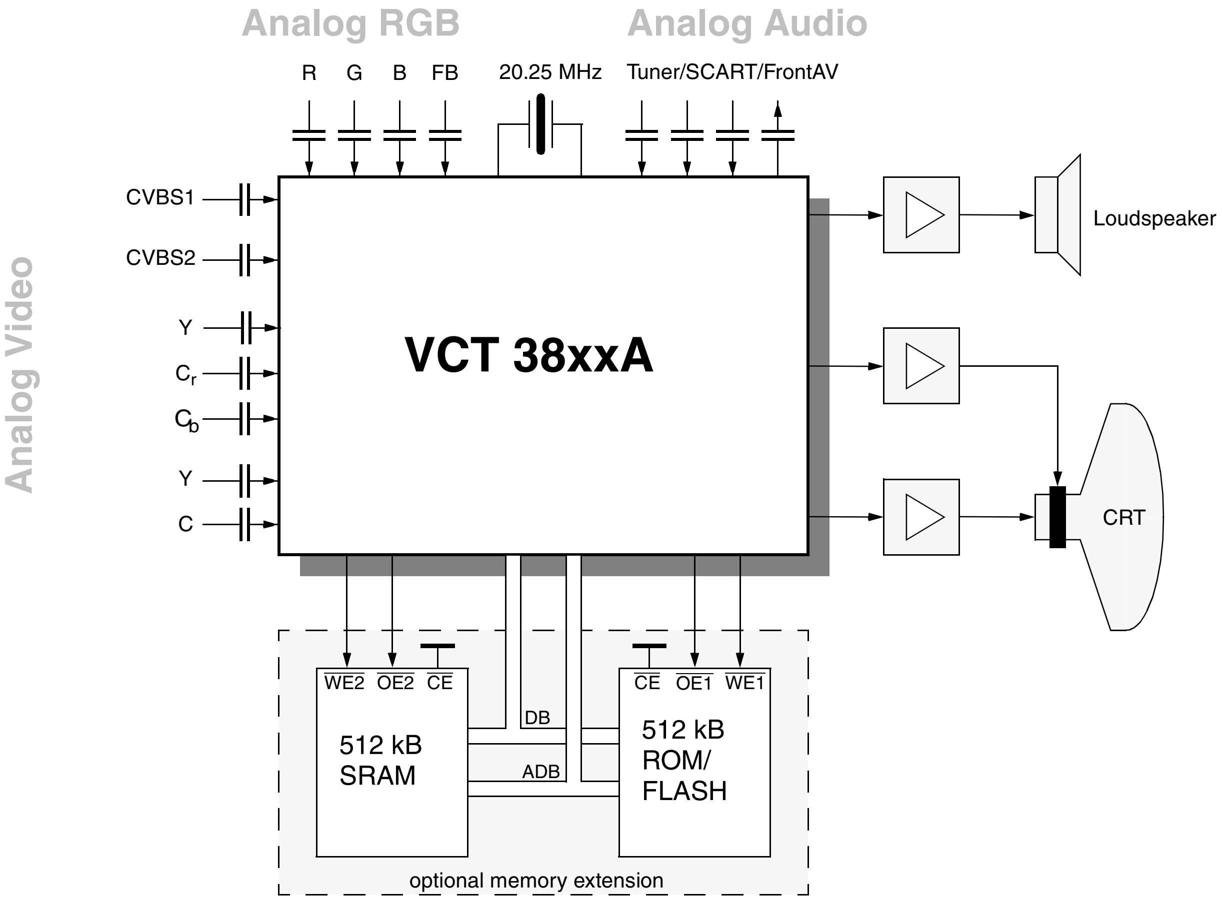


Fig. 1–3: Single-chip TV with VCT 38xxA

2. Video Processing

2.1. Introduction

The VCT 38xxA includes complete video, display, and deflection processing. In the following sections the video processing part of the VCT 38xxA will be named VDP for short.

All processing is done digitally, the video front-end and video back-end are interfacing to the analog world. Most functions of the VDP can be controlled by software via I^2C bus slave interface (see Section 2.15. on page 32).

2.2. Video Front-end

This block provides the analog interfaces to all video inputs and mainly carries out analog-to-digital conversion for the following digital video processing. A block diagram is given in Fig. 2–1.

Most of the functional blocks in the front-end are digitally controlled (clamping, AGC, and clock-DCO). The control loops are closed by the Fast Processor ('FP') embedded in the video decoder.

2.2.1. Input Selector

Up to seven analog inputs can be connected. Four inputs are for input of composite video or S-VHS luma signal. These inputs are clamped to the sync back porch and are amplified by a variable gain amplifier. Two chroma inputs can be used for connection of S-VHS carrier-chrominance signal. These inputs are internally biased and have a fixed gain amplifier. For analog YC_rC_b signals (e.g. from DVD players) one of the selected luminance inputs is used together with CBIN and CRIN inputs.

2.2.2. Clamping

The composite video input signals are AC-coupled to the IC. The clamping voltage is stored on the coupling capacitors and is generated by digitally controlled current sources. The clamping level is the back porch of the video signal. S-VHS chrominance is also AC-coupled. The input pin is internally biased to the center of the ADC input range. The chrominance inputs for YC_rC_b need to be AC-coupled by 220 nF clamping capacitors. It is strongly recommended to use 5-MHz anti-alias low-pass filters on each input. Each channel is sampled at 10.125 MHz with a resolution of 8 bit and a clamping level of 128.

2.2.3. Automatic Gain Control

A digitally working automatic gain control adjusts the magnitude of the selected baseband by +6/–4.5 dB in 64 logarithmic steps to the optimal range of the ADC. The gain of the video input stage including the ADC is 213 steps/V with the AGC set to 0 dB. The gain of the chrominance path in the YC_rC_b mode is fix and adapted to a nominal amplitude of 0.7 V_{pp}. However, if an overflow of the ADC occurs an extended signal range from 1 V_{pp} can be selected.

2.2.4. Analog-to-Digital Converters

Two ADCs are provided to digitize the input signals. Each converter runs with 20.25 MHz and has 8 bit resolution. An integrated bandgap circuit generates the required reference voltages for the converters. The two ADCs are of a 2-stage subranging type.

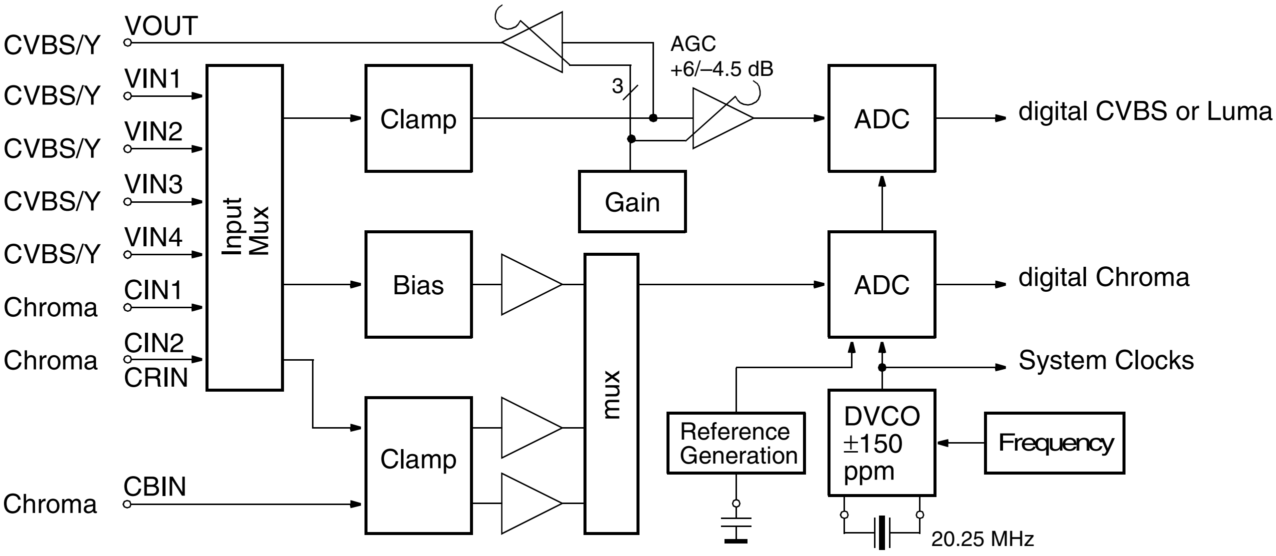


Fig. 2–1: Video front-end

2.2.5. Digitally Controlled Clock Oscillator

The clock generation is also a part of the analog front-end. The crystal oscillator is controlled digitally by the control processor. The clock frequency can be adjusted within ±150 ppm.

2.2.6. Analog Video Output

The input signal of the Luma ADC is available at the analog video output pin. The signal at this pin must be buffered by a source follower. The output voltage is 2 V, thus the signal can be used to drive a 75-Ω line. The magnitude is adjusted with an AGC in 8 steps together with the main AGC.

2.3. Adaptive Comb Filter

The adaptive comb filter is used for high-quality luminance/chrominance separation for PAL or NTSC signals. The comb filter improves the luminance resolution (bandwidth) and reduces interferences like cross-luminance and cross-color artifacts. The adaptive algorithm can eliminate most of the mentioned errors without introducing new artifacts or noise.

A block diagram of the comb filter is shown in Fig. 2–2. The filter uses two line delays to process the information of three adjacent video lines. To have a fixed phase relationship of the color subcarrier in the three channels, the system clock (20.25 MHz) is fractionally locked to the color subcarrier. This allows the processing of all color standards and substandards using a single crystal frequency.

The CVBS signal in the three channels is filtered at the subcarrier frequency by a set of bandpass/notch filters. The output of the three channels is used by the adaption logic to select the weighting that is used to reconstruct the luminance/chrominance signal from the 4 bandpass/notch filter signals. By using soft mixing of the 4 signals switching artifacts of the adaption algorithm are completely suppressed.

The comb filter uses the middle line as reference, therefore, the comb filter delay is one line. If the comb filter is switched off, the delay lines are used to pass the luma/ chroma signals from the A/D converters to the luma/ chroma outputs. Thus, the comb filter delay is always one line.

Various parameters of the comb filter are adjustable, hence giving to the user the ability to adjust his own desired picture quality.

Two parameters (KY, KC) set the global gain of luma and chroma comb separately; these values directly weigh the adaption algorithm output. In this way, it is possible to obtain a luma/chroma separation ranging from standard notch/bandpass to full comb decoding.

The parameter KB allows to choose between the two proposed comb booster modes. This so-called feature widely improves vertical high-to-low frequency transitions areas, the typical example being a multiburst to DC change. For KB=0, this improvement is kept moderate, whereas, in case of KB=1, it is maximum, but the risk to increase the “hanging dots” amount for some given color transitions is higher.

Using the default setting, the comb filter has separate luma and chroma decision algorithms; however, it is possible to switch the chroma comb factor to the current luma adaption output by setting CC to 1.

Another interesting feature is the programmable limitation of the luma comb amount; proper limitation, associated to adequate luma peaking, gives rise to an enhanced 2-D resolution homogeneity. This limitation is set by the parameter CLIM, ranging from 0 (no limitation) to 31 (max. limitation).

The DAA parameter (1:off, 0:on) is used to disable/ enable a very efficient built-in “rain effect” suppressor; many comb filters show this side effect which gives some vertical correlation to a 2-D uniform random area, due to the vertical filtering. This unnatural-looking phenomenon is mostly visible on tuner images, since they are always corrupted by some noise; and this looks like rain.

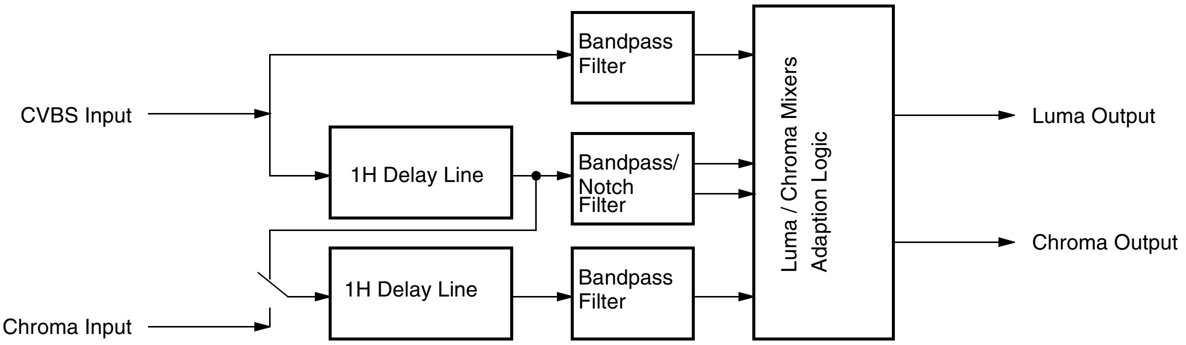


Fig. 2–2: Block diagram of the adaptive comb filter (PAL mode)

2.4. Color Decoder

In this block, the standard luma/chroma separation and multi-standard color demodulation is carried out. The color demodulation uses an asynchronous clock, thus allowing a unified architecture for all color standards.

A block diagram of the color decoder is shown in Fig. 2–3. The luma as well as the chroma processing, is shown here. The color decoder provides also some special modes, e.g. wide band chroma format which is intended for S-VHS wide bandwidth chroma.

If the adaptive comb filter is used for luma chroma separation, the color decoder uses the S-VHS mode processing. The output of the color decoder is YC_rC_b in a 4:2:2 format.

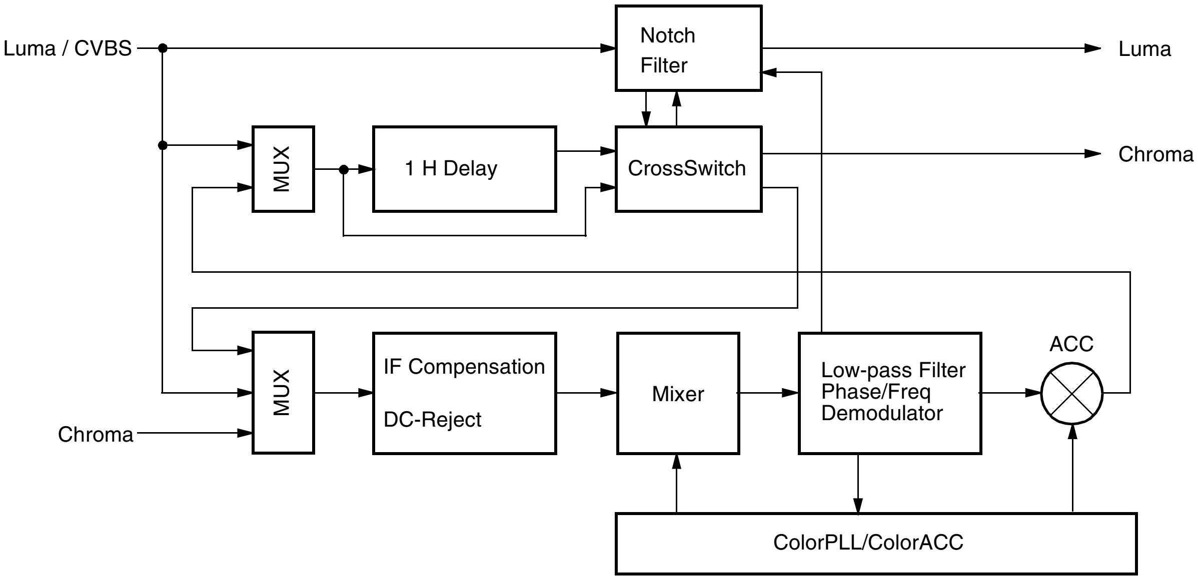


Fig. 2–3: Color decoder

2.4.1. IF-Compensation

With off-air or mistuned reception, any attenuation at higher frequencies or asymmetry around the color subcarrier is compensated. Four different settings of the IF-compensation are possible:

– flat (no compensation)

– 6 dB/octave

– 12 dB/octave

– 10 dB/MHz

The last setting gives a very large boost to high frequencies. It is provided for SECAM signals that are decoded using a SAW filter specified originally for the PAL standard.

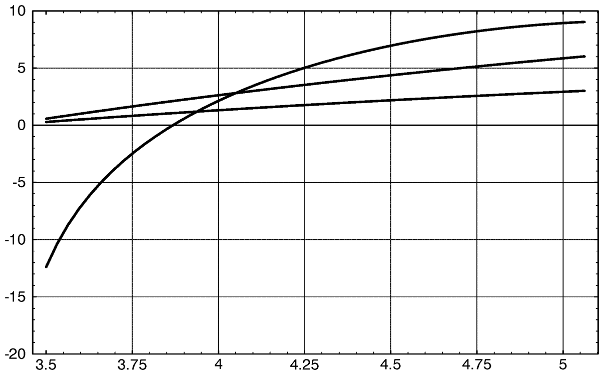


Fig. 2–4: Frequency response of chroma IF-compensation

2.4.2. Demodulator

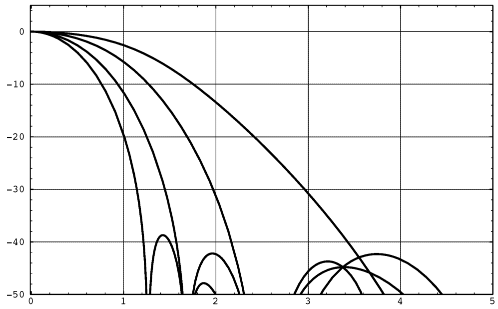
The entire signal (which might still contain luma) is now quadrature-mixed to the baseband. The mixing frequency is equal to the subcarrier for PAL and NTSC, thus achieving the chroma demodulation. For SECAM, the mixing frequency is 4.286 MHz giving the quadrature baseband components of the FM modulated chroma. After the mixer, a low-pass filter selects the chroma components; a downsampling stage converts the color difference signals to a multiplexed half-rate data stream.

The subcarrier frequency in the demodulator is generated by direct digital synthesis; therefore, substandards such as PAL 3.58 or NTSC 4.43 can also be demodulated.

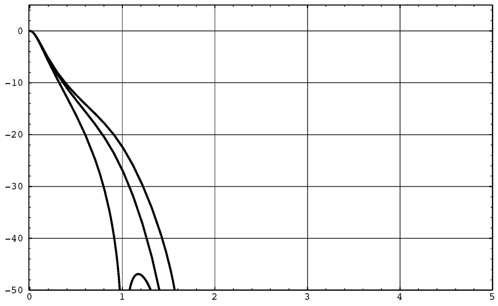
2.4.3. Chrominance Filter

The demodulation is followed by a low-pass filter for the color difference signals for PAL/NTSC. SECAM requires a modified low-pass function with bell-filter characteristic. At the output of the low-pass filter, all luma information is eliminated.

The low-pass filters are calculated in time multiplex for the two color signals. Three bandwidth settings (narrow, normal, broad) are available for each standard. For PAL/NTSC, a wide band chroma filter can be selected. This filter is intended for high bandwidth chroma signals, e.g. a non-standard wide bandwidth S-VHS signal.



PAL/NTSC



SECAM

Fig. 2–5: Frequency response of chroma filters

2.4.4. Frequency Demodulator

The frequency demodulator for demodulating the SECAM signal is implemented as a CORDIC structure. It calculates the phase and magnitude of the quadrature components by coordinate rotation.

The phase output of the CORDIC processor is differentiated to obtain the demodulated frequency. After the deemphasis filter, the Dr and Db signals are scaled to standard C_rC_b amplitudes and fed to the crossover switch.

2.4.5. Burst Detection / Saturation Control

In the PAL/NTSC-system the burst is the reference for the color signal. The phase and magnitude outputs of the CORDIC are gated with the color key and used for controlling the phase-locked-loop (APC) of the demodulator and the automatic color control (ACC) in PAL/NTSC.

The ACC has a control range of +30...−6 dB.

Color saturation can be selected once for all color standards. In PAL/NTSC it is used as reference for the ACC. In SECAM the necessary gains are calculated automatically.

For SECAM decoding, the frequency of the burst is measured. Thus, the current chroma carrier frequency can be identified and is used to control the SECAM processing. The burst measurements also control the color killer operation; they are used for automatic standard detection as well.

2.4.6. Color Killer Operation

The color killer uses the burst-phase/burst-frequency measurement to identify a PAL/NTSC or SECAM color signal. For PAL/NTSC, the color is switched off (killed) as long as the color subcarrier PLL is not locked. For SECAM, the killer is controlled by the toggle of the burst frequency. The burst amplitude measurement is used to switch-off the color if the burst amplitude is below a programmable threshold. Thus, color will be killed for very noisy signals. The color amplitude killer has a programmable hysteresis.

2.4.7. Automatic Standard Recognition

The burst-frequency measurement is also used for automatic standard recognition (together with the status of horizontal and vertical locking) thus allowing a completely independent search of the line and color standard of the input signal. The following standards can be distinguished:

PAL B,G,H,I; NTSC M; SECAM; NTSC 44; PAL M;
PAL N; PAL 60

For a preselection of allowed standards, the recognition can be enabled/disabled via I^{2}C bus for each standard separately.

If at least one standard is enabled, the VCT 38xxA regularly checks the horizontal and vertical locking of the input signal and the state of the color killer. If an error exists for several adjacent fields a new standard search is started. Depending on the measured line number and burst frequency, the current standard is selected.

For error handling the recognition algorithm delivers the following status information:

- search active (busy)
- search terminated, but failed
- found standard is disabled
- vertical standard invalid
- no color found

2.4.8. PAL Compensation/1-H Comb Filter

The color decoder uses one fully integrated delay line. Only active video is stored.

The delay line application depends on the color standard:

- NTSC: 1-H comb filter **or** color compensation
- PAL: color compensation
- SECAM: crossover switch

In the NTSC compensated mode, (Fig. 2–6c), the color signal is averaged for two adjacent lines. Thus, cross-color distortion and chroma noise is reduced. In the NTSC comb filter mode, (Fig. 2–6d), the delay line is in the composite signal path, thus allowing reduction of cross-color components, as well as cross-luminance. The loss of vertical resolution in the luminance channel is compensated by adding the vertical detail signal with removed color information.

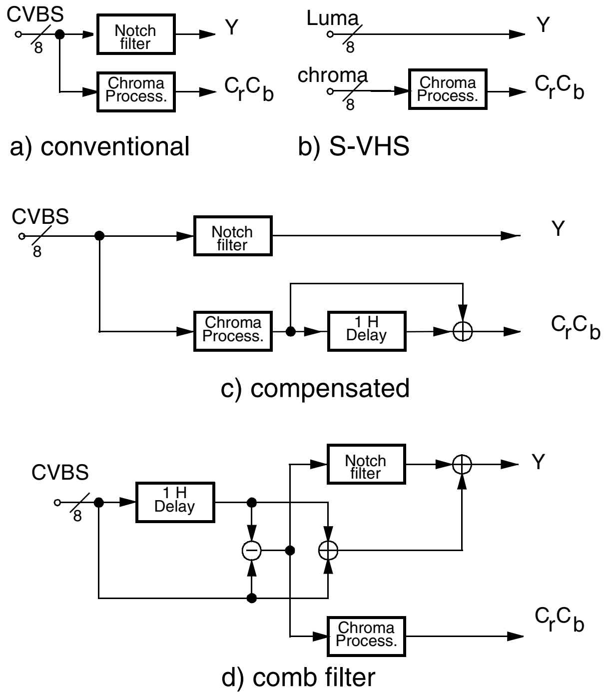


Fig. 2–6: NTSC color decoding options

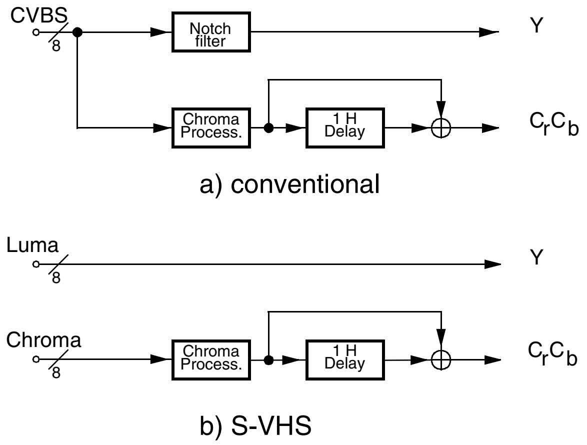


Fig. 2–7: PAL color decoding options

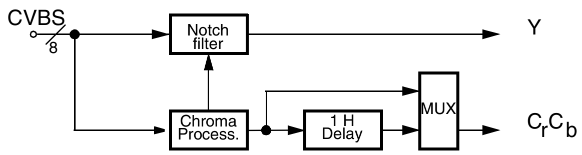
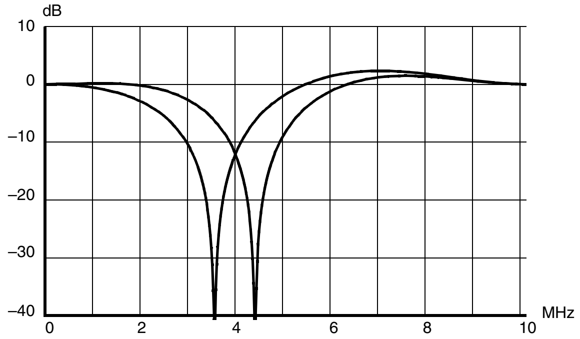


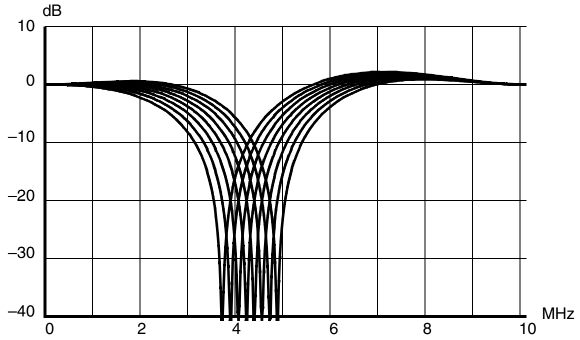
Fig. 2–8: SECAM color decoding

2.4.9. Luminance Notch Filter

If a composite video signal is applied, the color information is suppressed by a programmable notch filter. The position of the filter center frequency depends on the subcarrier frequency for PAL/NTSC. For SECAM, the notch is directly controlled by the chroma carrier frequency. This considerably reduces the cross-luminance. The frequency responses for all three systems are shown in Fig. 2–9.



PAL/NTSC notch filter



SECAM notch filter

Fig. 2–9: Frequency responses of the luma notch filter for PAL, NTSC, SECAM

2.4.10. Skew Filtering

The system clock is free-running and not locked to the TV line frequency. Therefore, the ADC sampling pattern is not orthogonal. The decoded YC_rC_b signals are converted to an orthogonal sampling raster by the skew filters, which are part of the scaler block.

The skew filters allow the application of a group delay to the input signals without introducing waveform or frequency response distortion.

The amount of phase shift of this filter is controlled by the horizontal PLL1. The accuracy of the filters is 1/32 clocks for luminance and 1/4 clocks for chroma. Thus the 4:2:2 YC_rC_b data is in an orthogonal pixel format even in the case of nonstandard input signals such as VCR.

2.5. Horizontal Scaler

The 4:2:2 YC_rC_b signal from the color decoder is processed by the horizontal scaler. The scaler block allows a linear or nonlinear horizontal scaling of the input video signal in the range of 0.25 to 4. Nonlinear scaling, also called “Panoramavision”, provides a geometrical distortion of the input picture. It is used to fit a picture with 4:3 format on a 16:9 screen by stretching the picture geometry at the borders. Also, the inverse effect can be produced by the scaler. A summary of scaler modes is given in Table 2–1.

The scaler contains a programmable decimation filter, a 1-line FIFO memory, and a programmable interpolation filter. The scaler input filter is also used for pixel skew correction (see Section 2.4.10. on page 16). The decimator/interpolator structure allows optimal use of the FIFO memory. The controlling of the scaler is done by the internal Fast Processor.

Table 2–1: Scaler modes

Mode	Scale Factor	Description
Compression 4:3 → 16:9	0.75 linear	4:3 source displayed on a 16:9 tube, with side panels
Panorama 4:3 →16:9	non-linear compr	4:3 source displayed on a 16:9 tube, Borders distorted
Zoom 4:3 → 4:3	1.33 linear	Letterbox source (PAL+) displayed on a 4:3 tube, vertical overscan with cropping of side panels
Panorama 4:3 → 4:3	non-linear zoom	Letterbox source (PAL+) displayed on a 4:3 tube, vertical overscan, borders distorted, no cropping

2.6. Black-line Detector

In case of a letterbox format input video, e.g. Cinemascope, PAL+ etc., black areas at the upper and lower part of the picture are visible. It is suitable to remove or reduce these areas by a vertical zoom and/or shift operation.

The VCT 38xxA supports this feature by a letterbox detector. The circuitry detects black video lines by measuring the signal amplitude during active video. For every field the number of black lines at the upper and lower part of the picture are measured, compared to the previous measurement and the minima are stored in the I^2C-register BLKLIN. To adjust the picture amplitude, the external controller reads this register, calculates the vertical scaling coefficient and transfers the new settings, e.g. vertical sawtooth parameters, horizontal scaling coefficient etc., to the VCT 38xxA.

Letterbox signals containing logos on the left or right side of the black areas are processed as black lines, while subtitles, inserted in the black areas, are processed as non-black lines. Therefore, the subtitles are visible on the screen. To suppress the subtitles, the vertical zoom coefficient is calculated by selecting the larger number of black lines only. Dark video scenes with a low contrast level compared to the letterbox area are indicated by the BLKPIC bit.

2.7. Test Pattern Generator

The YC_rC_b outputs can be switched to a test mode where YC_rC_b data are generated digitally in the VCT 38xxA. Test patterns include luma/chroma ramps and flat fields.

2.8. Video Sync Processing

Fig. 2–10 shows a block diagram of the front-end sync processing. To extract the sync information from the video signal, a linear phase low-pass filter eliminates all noise and video contents above 1 MHz. The sync is separated by a slicer; the sync phase is measured. A variable window can be selected to improve the noise immunity of the slicer. The phase comparator measures the falling edge of sync, as well as the integrated sync pulse.

The sync phase error is filtered by a phase-locked loop that is computed by the FP. All timing in the front-end is derived from a counter that is part of this PLL, and it thus counts synchronously to the video signal.

A separate hardware block measures the signal back porch and also allows gathering the maximum/minimum of the video signal. This information is processed by the FP and used for gain control and clamping.

For vertical sync separation, the sliced video signal is integrated. The FP uses the integrator value to derive vertical sync and field information.

The information extracted by the video sync processing is multiplexed onto the hardware front sync signal (FSY) and is distributed to the rest of the video processing system.

The data for the vertical deflection, the sawtooth, and the East-West correction signal is calculated by the VCT 38xxA. The data is buffered in a FIFO and transferred to the back-end by a single wire interface.

Frequency and phase characteristics of the analog video signal are derived from PLL1. The results are fed to the scaler unit for data interpolation and orthogonalization and to the clock synthesizer for line-locked clock generation. Horizontal and vertical syncs are latched with the line-locked clock.

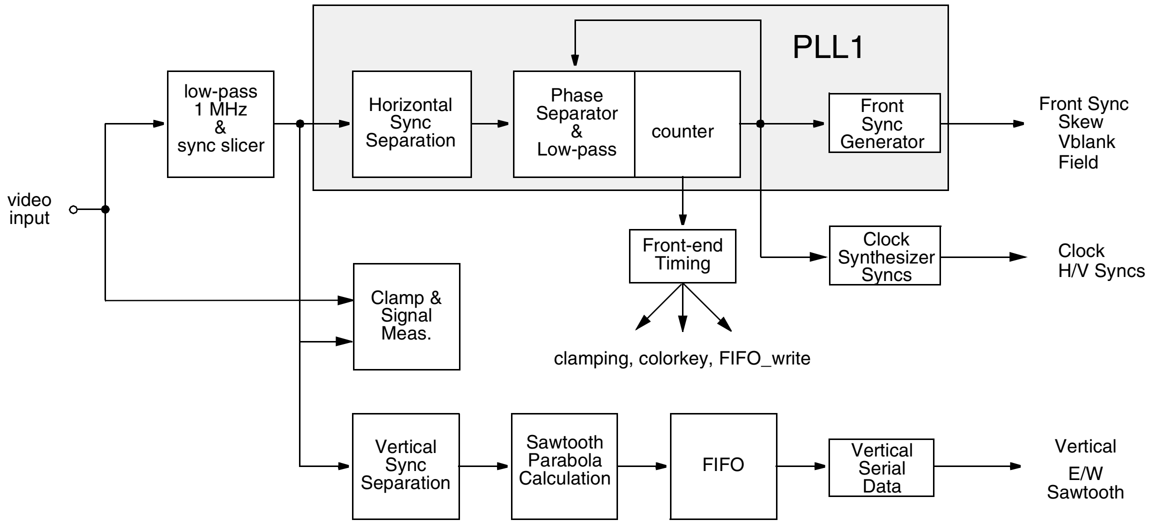


Fig. 2–10: Sync separation block diagram

2.9. Macrovision Detection

Video signals from Macrovision encoded VCR tapes are decoded without loss of picture quality. However, it might be necessary in some applications to detect the presence of Macrovision encoded video signals. This is possible by reading the Macrovision status register (FP-RAM 0x170).

Macrovision encoded video signals typically have AGC pulses and pseudo sync pulses added during VBI. The amplitude of the AGC pulses is modulated in time. The Macrovision detection logic measures the VBI lines and compares the signal against thresholds.

The window wherein the video lines are checked for Macrovision pulses can be defined in terms of start and stop line (e.g. 6-15 for NTSC).

2.10.Display Processing

In the display processing the conversion from digital YC_rC_b to analog RGB is carried out. A block diagram is shown in Fig. 2–18 on page 24. In the luminance processing path, contrast and brightness adjustments and a variety of features, such as black-level expansion, dynamic peaking and soft limiting, are provided. In the chrominance path, the C_rC_b signals are converted to 4:4:4 format and filtered by a color transient improvement circuit. The YC_rC_b signals are converted by a programmable matrix to RGB color space.

The display processor provides separate control settings for two pictures, i.e. different coefficients for a ‘main’ and a ‘side’ picture.

The digital OSD insertion circuit allows the insertion of a 5-bit OSD signal. The color space for this signal is controlled by a partially programmable color look-up table (CLUT) and contrast adjustment.

The OSD signals and the display clock are synchronized to the horizontal flyback. For the display clock, a gate delay phase shifter is used. In the analog backend, three 10-bit digital-to-analog converters provide the analog output signals.

2.10.1.Luma Contrast Adjustment

The contrast of the luminance signal can be adjusted by multiplication with a 6-bit contrast value. The contrast value corresponds to a gain factor from 0 to 2, where the value 32 is equivalent to a gain of 1. The contrast can be adjusted separately for main picture and side picture.

2.10.2.Black-Level Expander

The black-level expander enhances the contrast of the picture. Therefore the luminance signal is modified with an adjustable, non-linear function. Dark areas of the picture are changed to black, while bright areas remain unchanged. The advantage of this black-level expander is that the black expansion is performed only if it will be most noticeable to the viewer.

The black-level expander works adaptively. Depending on the measured amplitudes ‘L_{min}’ and ‘L_{max}’ of the low-pass-filtered luminance and an adjustable coefficient BTLT, a tilt point ‘L_t’ is established by

$$L_t = L_{min} + BTLT\ (L_{max} - L_{min}).$$

Above this value there is no expansion, while all luminance values below this point are expanded according to:

$$L_{out} = L_{in} + BAM\ (L_{in} - L_t)$$

A second threshold, L_{tr}, can be programmed, above which there is no expansion. The characteristics of the black-level expander are shown in Fig. 2–11and Fig. 2–12.

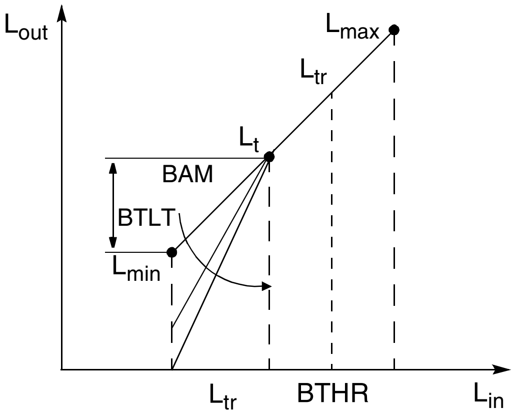


Fig. 2–11: Characteristics of the black-level expander

The tilt point L_t is a function of the dynamic range of the video signal. Thus, the black-level expansion is only performed when the video signal has a large dynamic range. Otherwise, the expansion to black is zero. This allows the correction of the characteristics of the picture tube.

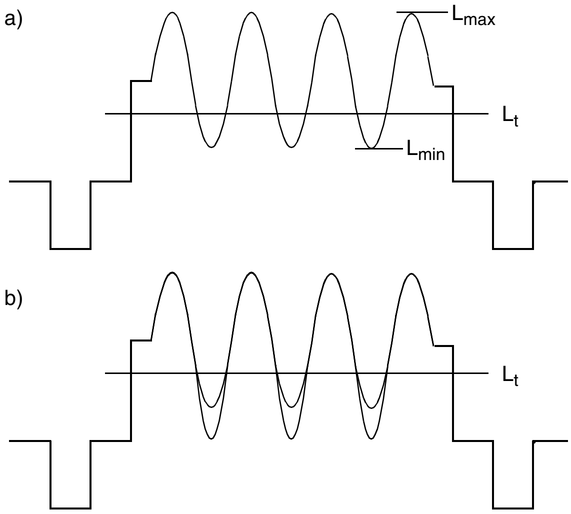


Fig. 2–12: Black-level expansion
a) luminance input
b) luminance input and output

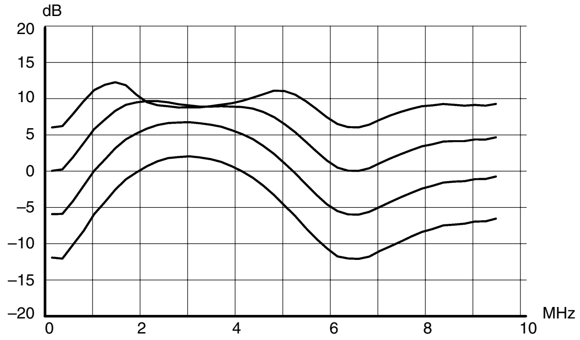


Fig. 2–13: Dynamic peaking frequency response

2.10.3.Dynamic Peaking

Especially with decoded composite signals and notch filter luminance separation, as input signals, it is necessary to improve the luminance frequency characteristics. With transparent, high-bandwidth signals, it is sometimes desirable to soften the image.

In the VCT 38xxA, the luma response is improved by ‘dynamic’ peaking. The algorithm has been optimized regarding step and frequency response. It adapts to the amplitude of the high-frequency part. Small AC amplitudes are processed, while large AC amplitudes stay nearly unmodified.

The dynamic range can be adjusted from –14 to +14 dB for small high-frequency signals. There is separate adjustment for signal overshoot and for signal undershoot. For large signals, the dynamic range is limited by a non-linear function that does not create any visible alias components. The peaking can be switched over to “softening” by inverting the peaking term by software.

The center frequency of the peaking filter is switchable from 2.5 MHz to 3.2 MHz. For S-VHS and for notch filter color decoding, the total system frequency responses for both PAL and NTSC are shown in Fig. 2–14.

Transients, produced by the dynamic peaking when switching video source signals, can be suppressed via the priority bus.

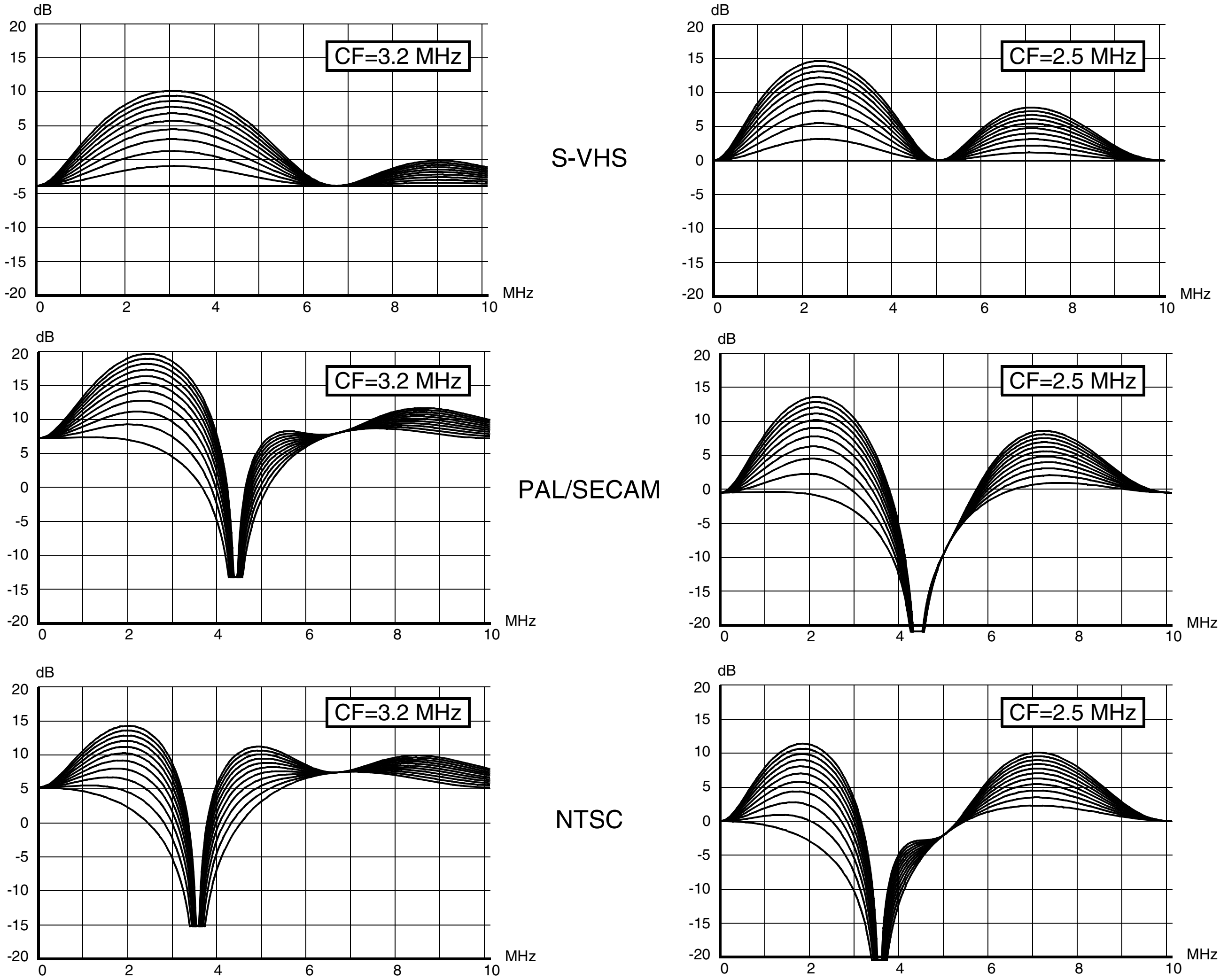


Fig. 2–14: Total frequency response for peaking filter and S-VHS, PAL, NTSC

2.10.4. Digital Brightness Adjustment

The DC-level of the luminance signal can be adjusted by adding an 8-bit number in the luminance signal path in front of the softlimiter.

With a contrast adjustment of 32 (gain+1) the signal can be shifted by 100 %. After the brightness addition, the negative going signals are limited to zero. It is desirable to keep a small positive offset with the signal to prevent undershoots produced by the peaking from being cut. The digital brightness adjustment works separately for main and side picture.

2.10.5. Soft Limiter

The dynamic range of the processed luma signal must be limited to prevent the CRT from overload. An appropriate headroom for contrast, peaking and brightness can be adjusted by the TV manufacturer according to the CRT characteristics. All signals above this limit will be ‘soft’-clipped. A characteristic diagram of the soft limiter is shown in Fig. 2–15. The total limiter consists of three parts:

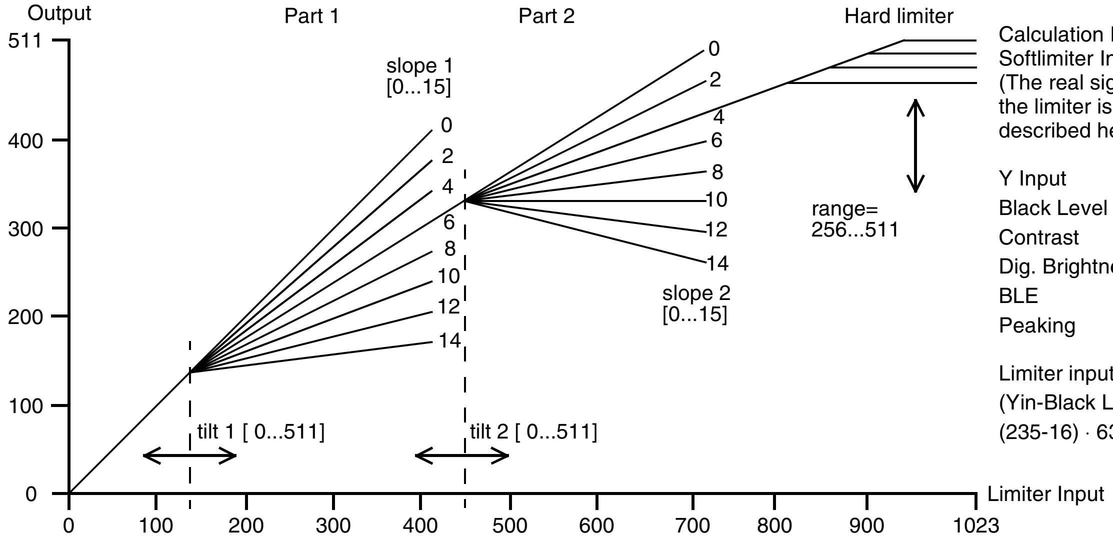
Part 1 includes adjustable tilt point and gain. The gain before the tilt value is 1. Above the tilt value, a part (0...15/16) of the input signal is subtracted from the input signal itself. Therefore, the gain is adjustable from 16/16 to 1/16, when the slope value varies from 0 to 15. The tilt value can be adjusted from 0 to 511.

Part 2 has the same characteristics as part 1. The subtracting part is also relative to the input signal, so the total differential gain will become negative if the sum of slope 1 and slope 2 is greater than 16 and the input signal is above the both tilt values (see characteristics).

Finally, the output signal of the soft limiter will be clipped by a hard limiter adjustable from 256 to 511.

2.10.6. Chroma Interpolation

A linear phase interpolator is used to convert the chroma sampling rate from 10.125 MHz (4:2:2) to 20.25 MHz (4:4:4). All further processing is carried out at the full sampling rate.



Calculation Example for the Softlimiter Input Amplitude. (The real signal processing in the limiter is 2 bit more than described here)

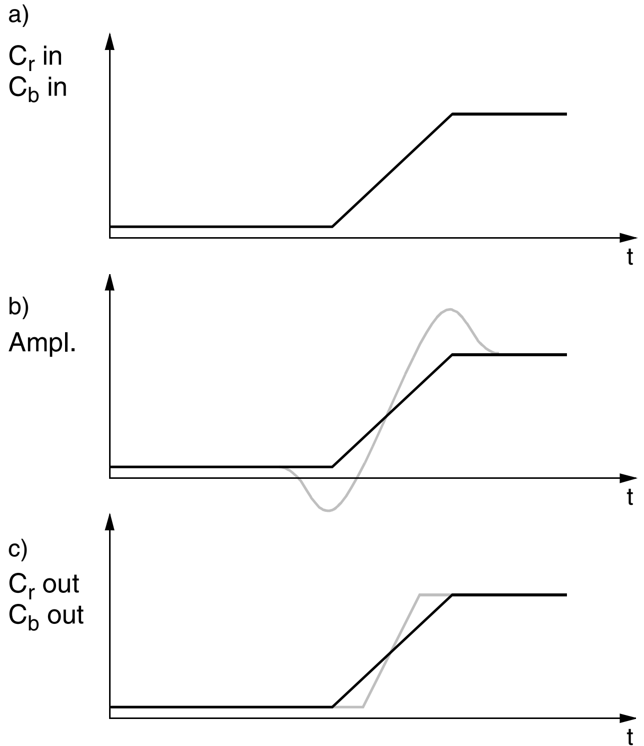
Y Input	16...235 (ITUR)
Black Level	16 (constant)
Contrast	63
Dig. Brightness	20
BLE	off
Peaking	off

Limiter input signal:
(Yin-Black Level)·Contr./32 + Brightn.
(235-16) · 63/32 + 20 = 451

Fig. 2–15: Characteristic of soft limiter A and B and hard limiter

2.10.7. Chroma Transient Improvement

The intention of this block is to enhance the chroma resolution. A correction signal is calculated by differentiation of the color difference signals. The differentiation can be selected according to the signal bandwidth, e.g. for PAL/NTSC/SECAM or digital component signals, respectively. The amplitude of the correction signal is adjustable. Small noise amplitudes in the correction signal are suppressed by an adjustable coring circuit. To eliminate ‘wrong colors’, which are caused by over and undershoots at the chroma transition, the sharpened chroma signals are limited to a proper value automatically.



a) C_rC_b input of DTI
b) C_rC_b input + correction signal
c) sharpened and limited C_rC_b

Fig. 2–16: Digital color transient improvement

2.10.8.Inverse Matrix

A 6-multiplier matrix transcodes the C_r and C_b signals to R-Y, B-Y, and G-Y. The multipliers are also used to adjust color saturation in the range of 0 to 2. The coefficients are signed and have a resolution of 9 bits. There are separate matrix coefficients for main and side pictures. The matrix computes:

$$
\begin{aligned}
R-Y &= MR1*C_b+MR2*C_r \\
G-Y &= MG1*C_b+MR2*C_r \\
B-Y &= MB1*C_b+MR2*C_r
\end{aligned}
$$

The initialization values for the matrix are computed from the standard ITUR (CCIR) matrix:

$$
\begin{array}{c} R \\ G \\ B \end{array} = \begin{pmatrix} 1 & 0 & 1.402 \\ 1 & -0.345 & -0.713 \\ 1 & 1.773 & 0 \end{pmatrix} \begin{array}{c} Y \\ C_b \\ C_r \end{array}
$$

For a contrast setting of CTM+32, the matrix values are scaled by a factor of 64 (see Table 2–4 on page 34).

2.10.9.RGB Processing

After adding the post-processed luma, the digital RGB signals are limited to 10 bits. Three multipliers are used to digitally adjust the white drive. Using the same multipliers an average beam current limiter is implemented (see Section 2.11.1. on page 25).

2.10.10.OSD Color Look-up Table

The VCT 38xxA has five input lines for an OSD signal. This signal forms a 5-bit address for a color look-up table (CLUT). The CLUT is a memory with 32 words where each word holds a RGB value.

Bits 0 to 3 (bit 4=0) form the addresses for the ROM part of the OSD, which generates full RGB signals (bit 0 to 2) and half-contrast RGB signals (bit 3).

Bit 4 addresses the RAM part of the OSD with 16 freely programmable colors, addressable with bit 0 to 3. The programming is done via the I^2C bus.

The amplitude of the CLUT output signals can be adjusted separately for R, G, and B via the I^2C bus. The switchover between video RGB and OSD RGB is done via the priority decoder.

2.10.11.Picture Frame Generator

When the picture does not fill the total screen (height or width too small) it is surrounded with black areas. These areas (and more) can be colored with the picture frame generator. This is done by switching over the RGB signal from the matrix to the signal from the OSD color look-up table.

The width of each area (left, right, upper, lower) can be adjusted separately. The generator starts on the right, respectively lower side of the screen and stops on the left, respectively upper side of the screen. This means, it runs during horizontal, respectively vertical flyback. The color of the complete border can be stored in the programmable OSD color look-up table in a separate address. The format is 3 x 4-bit RGB. The contrast can be adjusted separately.

The picture frame generator includes a priority master circuit. Its priority is programmable and the border is generated only if the priority is higher than the priority of the other sources (video/OSD). Therefore, the border can be underlay or overlay depending on the picture source.

2.10.12.Priority Decoder

The priority decoder selects the picture source depending on the programmed priorities. Up to eight levels can be selected for OSD and the picture frame – where 0 is the highest. The video source always has the lowest priority. A 5-bit information is attached to each priority (see Table 2–4 on page 34). These bits are programmable via the I^2C bus and have the following meanings:

- one of two contrast, brightness and matrix values for main and side picture
- RGB from video signal or color look-up table
- disable/enable black-level expander
- disable/enable peaking transient suppression when signal is switched
- disable/enable analog Fast-Blank input

2.10.13.Scan Velocity Modulation

The RGB input signal of the SVM is converted to Y in a simple matrix. Then the Y signal is differentiated by a filter of the transfer function $1\text{-}Z^{-N}$, where N is programmable from 1 to 6. With a coring, some noise can be suppressed. This is followed by a gain adjustment and an adjustable limiter. The analog output signal is generated by an 8-bit D/A converter.

The signal delay can be adjusted by ±3.5 clocks in half-clock steps. For the gain and filter adjustment there are two parameter sets. The switching between these two sets is done with the same RGB switch signal that is used for switching between video-RGB and OSD-RGB for the RGB outputs (see Fig. 2–17).

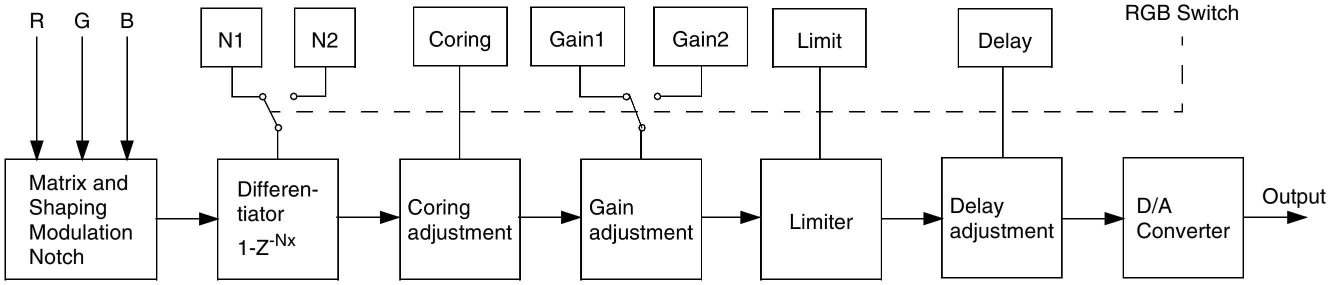


Fig. 2–17: SVM Block diagram

2.10.14.Display Phase Shifter

A phase shifter is used to partially compensate the phase differences between the video source and the flyback signal. By using the described clock system, this phase shifter works with an accuracy of approximately 1 ns. It has a range of 1 clock period which is equivalent to ±24.7 ns at 20.25 MHz. The large amount of phase shift (full clock periods) is realized in the front-end circuit.

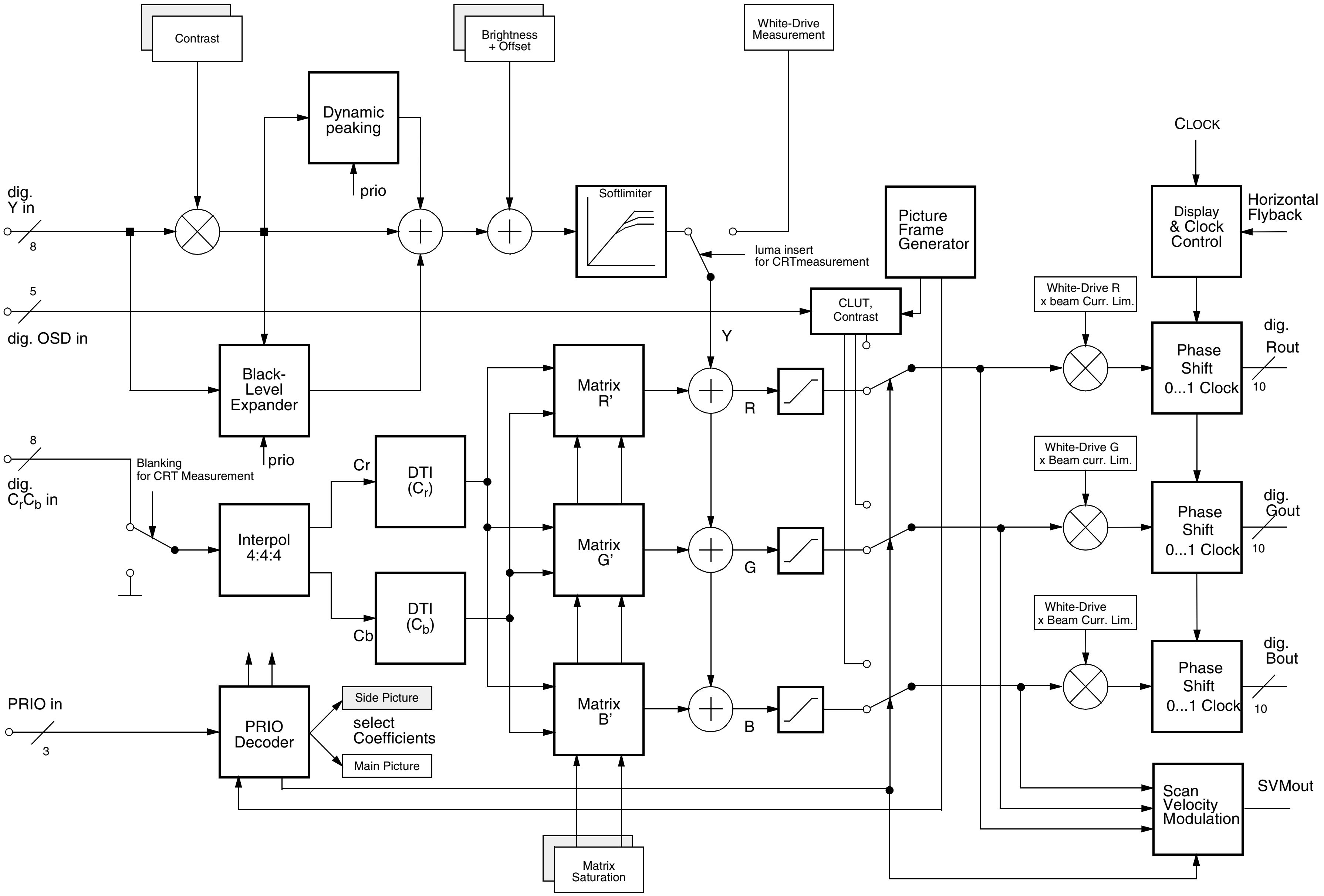


Fig. 2–18: Digital back-end

2.11.Video Back-end

The digital RGB signals are converted to analog RGBs using three video digital-to-analog converters (DAC) with 10-bit resolution. An analog brightness value is provided by three additional DACs. The adjustment range is 40 % of the full RGB range.

Controlling the white-drive/analog brightness and also the external contrast and brightness adjustments is done via the Fast Processor, located in the front-end. Control of the cutoff DACs is done via I^2C bus registers.

Finally cutoff and blanking values are added to the RGB signals. Cutoff (dark current) is provided by three 9-bit DACs. The adjustment range is 60 % of full scale RGB range.

The analog RGB-outputs are current outputs with current-sink characteristics. The maximum current drawn by the output stage is obtained with peak white RGB. An external half contrast signal can be used to reduce the output current of the RGB outputs to 50 %.

2.11.1.CRT Measurement and Control

The display processor is equipped with an 8-bit PDM-ADC for all measuring purposes. The ADC is connected to the SENSE input pin, the input range is 0 to 1.5V. The bandwidth of the PDM filter can be selected; it is 40/80 kHz for small/large bandwidth setting. The input impedance is more than 1 MΩ.

Cutoff and white-drive current measurement are carried out during the vertical blanking interval. They always use the small bandwidth setting. The current range for the cutoff measurement is set by connecting a sense resistor to the MADC input. For the white-drive measurement, the range is set by using another sense resistor and the range select switch 2 output pin (RSW2). During the active picture, the minimum and maximum beam current is measured. The measurement range can be set by using the range select switch 1 pin (RSW1) as shown in Fig. 2–19 and Fig. 2–20. The timing window of this measurement is programmable. The intention is, to automatically detect letterbox transmission or to measure the actual beam current. All control loops are closed via the external control microprocessor.

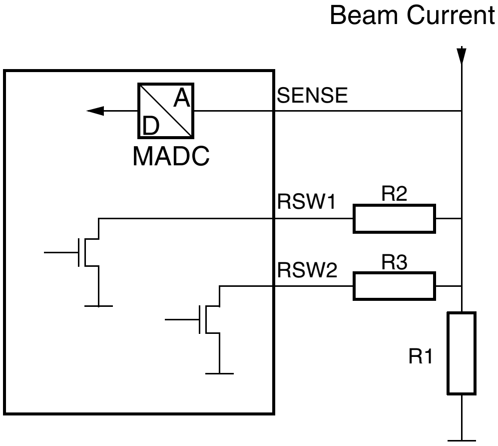


Fig. 2–19: MADC range switches

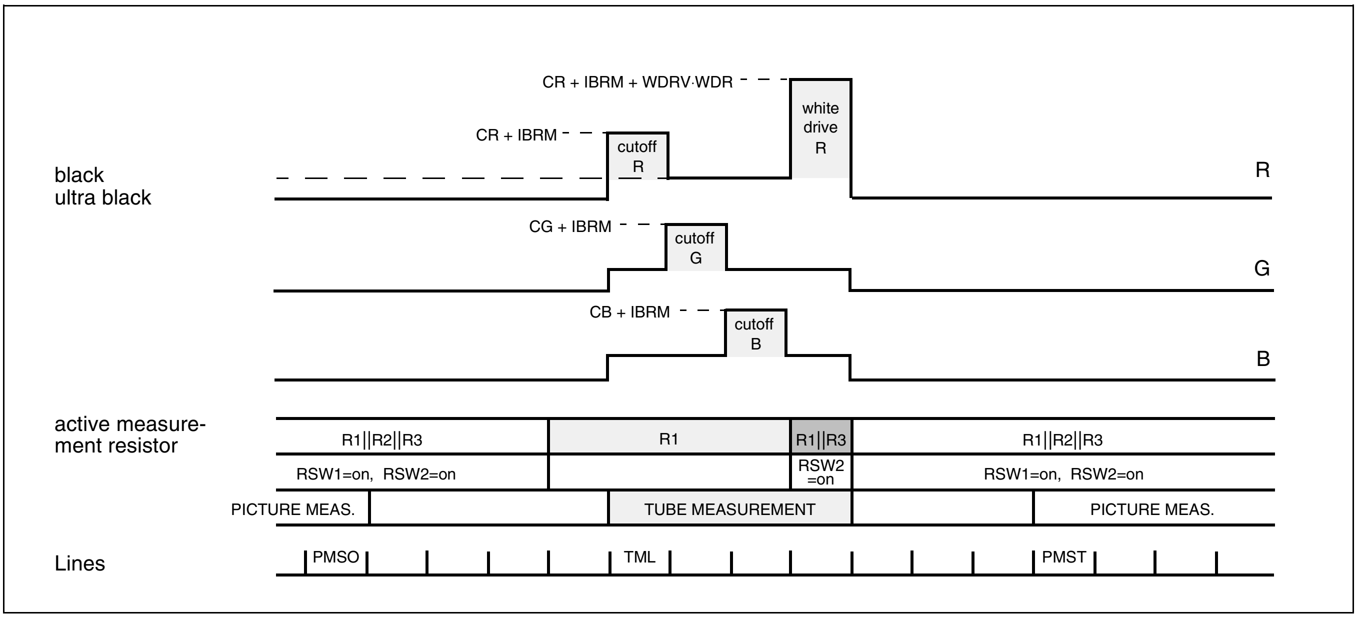


Fig. 2–20: MADC measurement timing

In each field two sets of measurements can be taken:

a) The picture tube measurement returns results for

- cutoff R
- cutoff G
- cutoff B
- white-drive R or G or B (sequentially)

b) The picture measurement returns data on

- active picture maximum current
- active picture minimum current

The tube measurement is automatically started when the cutoff blue result register is read. Cutoff control for RGB requires one field only, whereas a complete white-drive control requires three fields. If the measurement mode is set to ‘offset check’, a measurement cycle is run with the cutoff/white-drive signals set to zero. This allows to compensate the MADC offset as well as input the leakage currents. During cutoff and white-drive measurements, the average beam current limiter function (see Section 2.11.3. on page 27) is switched off and a programmable value is used for the brightness setting. The start line of the tube measurement can be programmed via I^2C bus, the first line used for the measurement, i.e. measurement of cutoff red, is 2 lines after the programmed start line.

The picture measurement must be enabled by the control microprocessor after reading the min./max. result registers. If a ‘1’ is written into bit 2 in subaddress 25, the measurement runs for one field. For the next measurement a ‘1’ has to be written again. The measurement is always started at the beginning of active video.

The vertical timing for the picture measurement is programmable, and may even be a single line. Also the signal bandwidth is switchable for the picture measurement.

Two horizontal windows are available for the picture measurement. The large window is active for the entire active line. Tube measurement is always carried out with the small window. Measurement windows for picture and tube measurement are shown in Fig. 2–21.

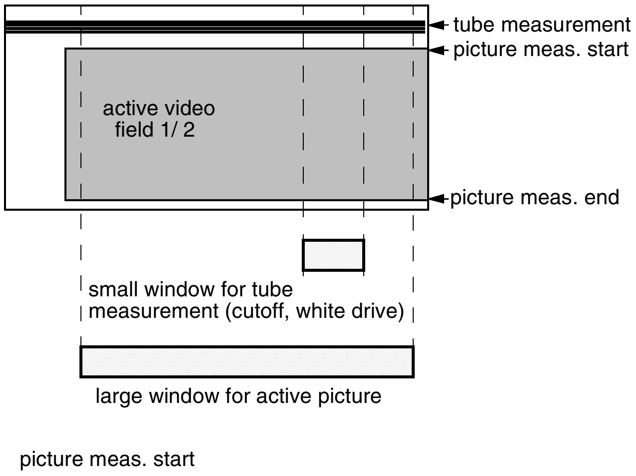


Fig. 2–21: Windows for tube and picture measurements

2.11.2.SCART Output Signal

The RGB output of the VCT 38xxA can also be used to drive a SCART output. In the case of the SCART signal, the parameter CLMPR (clamping reference) has to be set to 1. Then, during blanking, the RGB outputs are automatically set to 50 % of the maximum brightness. The DC offset values can be adjusted with the cutoff parameters CR, CG, and CB. The amplitudes can be adjusted with the drive parameters WDR, WDG, and WDB.

2.11.3.Average Beam Current Limiter

The average beam current limiter (BCL) uses the SENSE input for the beam current measurement. The BCL uses a different filter to average the beam current during the active picture. The filter bandwidth is approx. 2 kHz. The beam current limiter has an automatic offset adjustment that is active two lines before the first cutoff measurement line.

The beam current limiter function is located in the front-end. The data exchange between the front-end and the back-end is done via a single-wire serial interface.

The beam current limiter allows the setting of a threshold current. If the beam current is above the threshold, the excess current is low-pass filtered and used to attenuate the RGB outputs by adjusting the white-drive multipliers for the internal (digital) RGB signals, and the analog contrast multipliers for the analog RGB inputs, respectively. The lower limit of the attenuator is programmable, thus a minimum contrast can always be set. During the tube measurement, the ABL attenuation is switched off. After the white-drive measurement line it takes 3 lines to switch back to BCL limited drives and brightness.

Typical characteristics of the ABL for different loop gains are shown in Fig. 2–22; for this example the tube has been assumed to have square law characteristics.

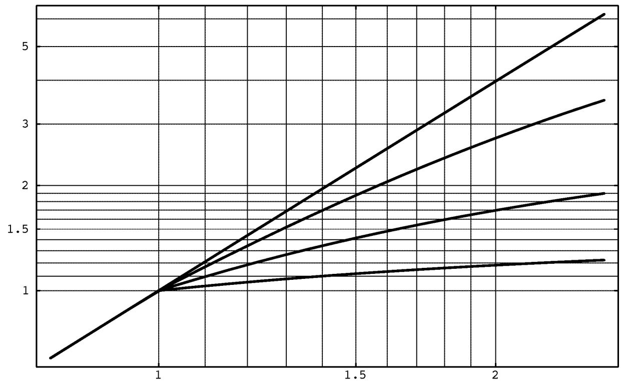


Fig. 2–22: Beam current limiter characteristics:
beam current output vs. drive
BCL threshold: 1

2.11.4.Analog RGB Insertion

The VCT 38xxA allows insertion of external analog RGB signals. The RGB signal is key-clamped and inserted into the main RGB by the Fast-Blank switch. The external RGB input can be overlaid or underlaid to the digital picture. The external RGB signals can be adjusted independently as regards DC level (brightness) and magnitude (contrast).

All signals for analog RGB insertion (RIN, GIN, BIN, FBLIN) must be synchronized to the horizontal flyback, otherwise a horizontal jitter will be visible. The VCT 38xxA has no means for timing correction of the analog RGB input signals.

2.11.5.Fast-Blank Monitor

The presence of external analog RGB sources can be detected by means of a Fast-Blank monitor. The status of the Fast-Blank input can be monitored via an I^2C bus register. There is a 2 bit information, giving static and dynamic indication of a Fast-Blank signal. The static bit is directly reading the Fast-Blank input line, whereas the dynamic bit is reading the status of a flip-flop triggered by the negative edge of the Fast-Blank signal.

With this monitor logic it is possible to detect if there is an external RGB source active and if it is a full screen insertion or only a box. The monitor logic is connected directly to the FBLIN pin.

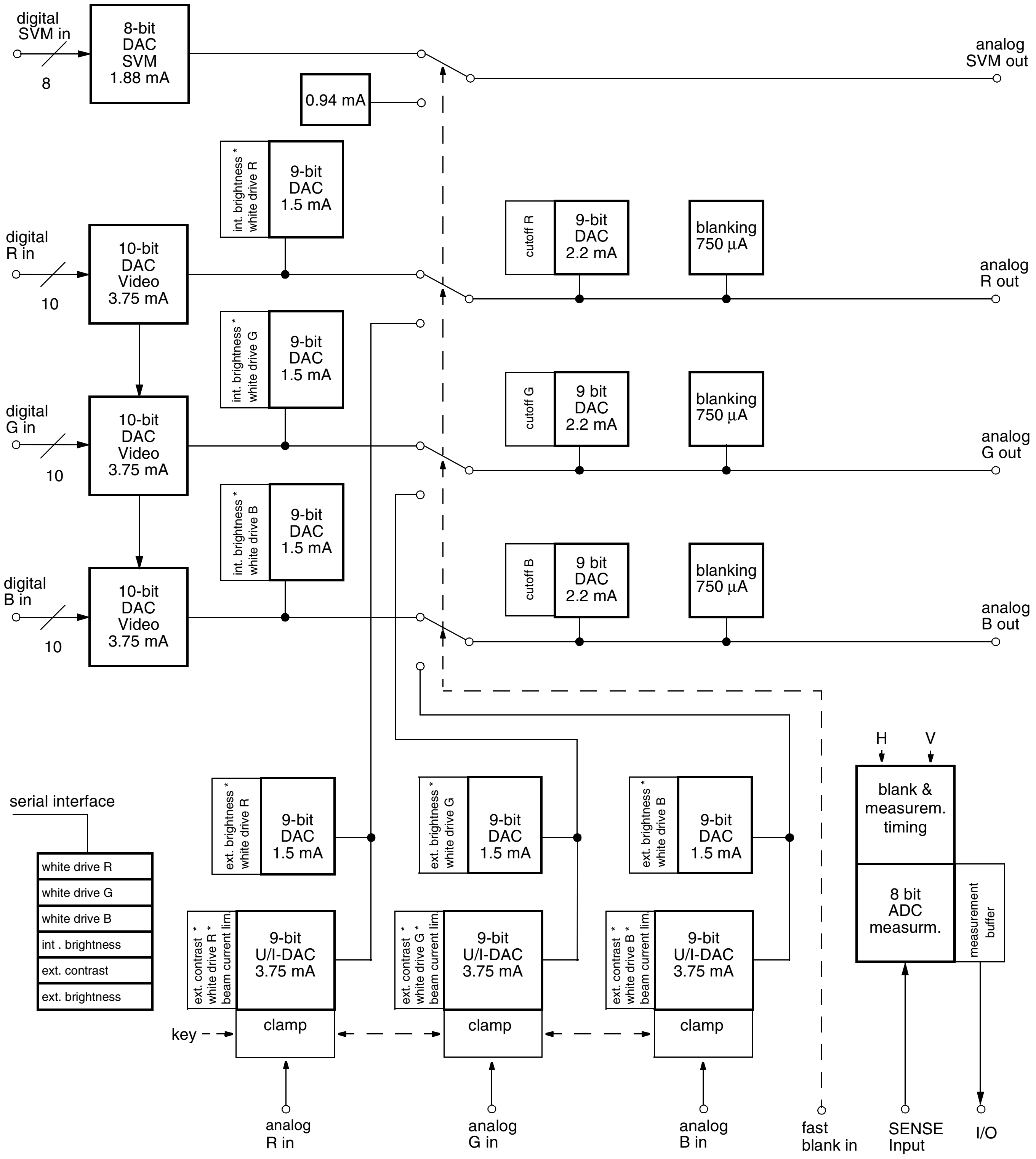


Fig. 2–23: Video back-end

2.12.Synchronization and Deflection

The synchronization and deflection processing is distributed over front-end and back-end. The video clamping, horizontal and vertical sync separation and all video related timing information are processed in the front-end. Most of the processing that runs at the horizontal frequency is programmed on the internal Fast Processor (FP). Also the values for vertical and East/West deflection are calculated by the FP software.

The generation of horizontal and vertical drive signals can be synchronized to the video timing extracted in the front-end or to a free running line counter in the back-end.

2.12.1.Deflection Processing

The deflection processing generates the signals for the horizontal and vertical drive (see Fig. 2–24). This block contains two phase-locked loops:

- PLL2 generates the horizontal and vertical timing, e.g. blanking, clamping and composite sync. Phase and frequency are synchronized by the front sync signal.
- PLL3 adjusts the phase of the horizontal drive pulse and compensates for the delay of the horizontal output stage. Phase and frequency are synchronized by the oscillator signal of PLL2.

The horizontal drive circuitry uses a digital sine wave generator to produce the exact (subclock) timing for the drive pulse HOUT. The generator runs at 1 MHz. Under control of the EHPLL bit and the internal voltage supervision it is either synchronized by the deflection PLL or it is free running. In the output stage the frequency is divided down to give drive-pulse period and width. The drive pulse width is programmable. The horizontal drive uses an open drain output transistor.

After power on or during reset the HOUT generation is switched to a free running mode with a fix duty cycle of 50 %. For normal operation the EHPLL bit has to be set first. During the switch the actual period of HOUT can vary by up to 1 µs.

2.12.2.Angle and Bow Correction

The Angle and Bow correction is part of the horizontal drive PLL. This feature allows a shift of the horizontal drive pulse phase depending on the vertical position on the screen. The phase correction has a linear (angle) and a quadratic term (bow).

2.12.3.Horizontal Phase Adjustment

This section describes a simple way to align PLL phases and the horizontal frame position.

1. With HDRV the duration of the horizontal drive pulse has to be adjusted
2. With POFS2 the delay between input video and display timing (e.g. clamping pulse for analog RGB) has to be adjusted
3. With CSYDEL the delay between video and analog RGB (OSD) has to be adjusted.
4. With CSYDEL and HPOS the horizontal position of both, the digital and analog RGB signal (from SCART) relative to the clamping pulse has to be adjusted to the correct position, e.g. the pedestal of the generator signal.
5. With POFS3 the position of horizontal drive/flyback relative to RGB has to be adjusted
6. With NEWLIN the position of a scaled video picture can be adjusted (left, middle, center, etc; versions with panorama scaler only).
7. With HBST and HBSO, the start and stop values for the horizontal blanking have to be adjusted.

Note: The processing delay of the internal digital video path differs depending on the comb filter option of the VCT 38xxA. The versions with comb filter have an additional delay of 34 clock cycles.

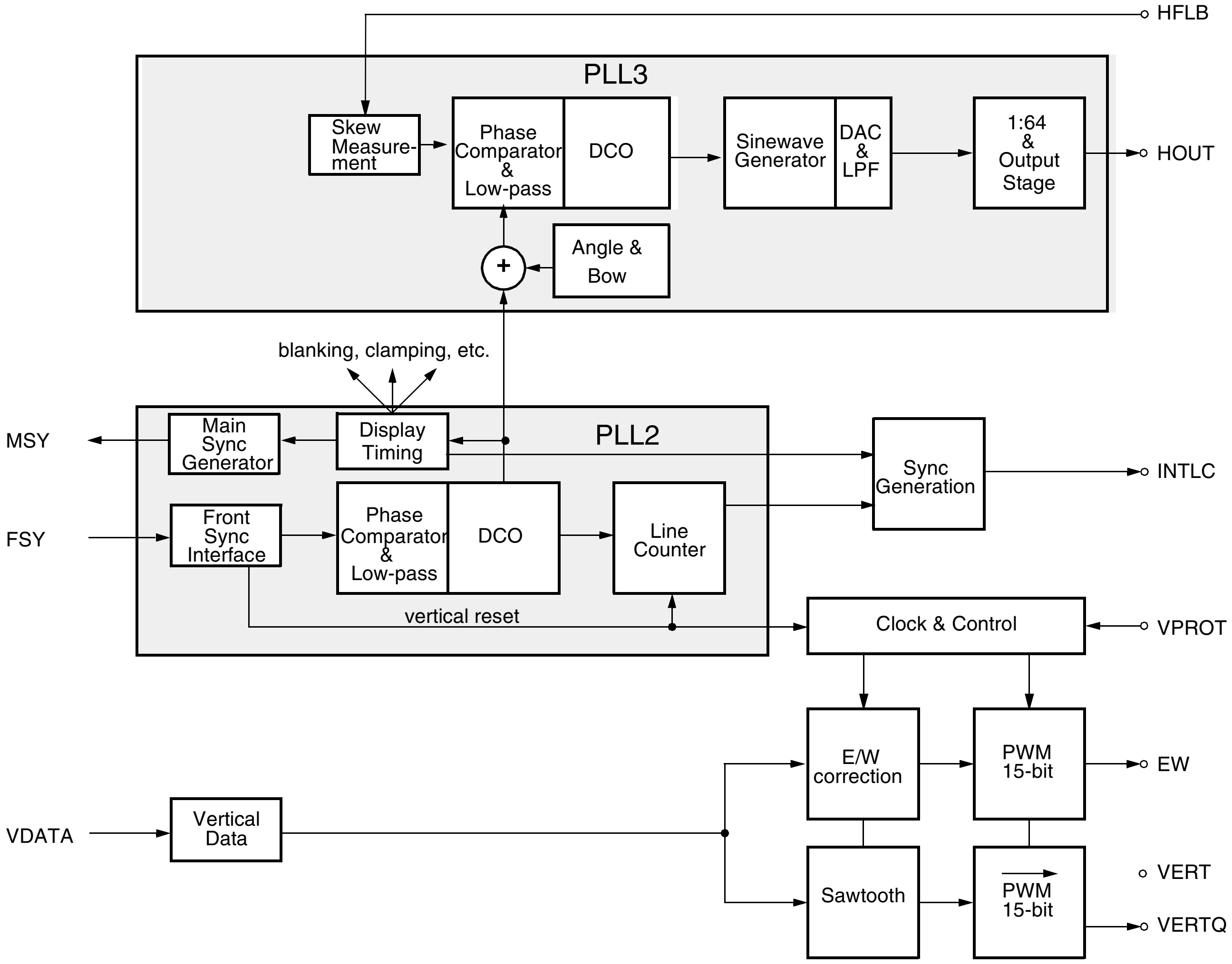


Fig. 2–24: Deflection processing block diagram

2.12.4. Vertical and East/West Deflection

The calculations of the vertical and East/West deflection waveforms is done by the internal Fast Processor (FP). The algorithm uses a chain of accumulators to generate the required polynomial waveforms. To produce the deflection waveforms, the accumulators are initialized at the beginning of each field. The initialization values must be computed by the TV control processor and are written to the front-end once. The waveforms are described as polynomials in x, where x varies from 0 to 1 for one field.

$P: a + b(x\text{-}0.5) + c(x\text{-}0.5)^2 + d(x\text{-}0.5)^3 + e(x\text{-}0.5)^4$

The initialization values for the accumulators a0..a3 for vertical deflection and a0..a4 for East/West deflection are 12-bit values.

Fig. 2–25 shows several vertical and East/West deflection waveforms. The polynomial coefficients are also stated.

In order to get a faster vertical retrace timing, the output impedance of the vertical D/A-converter can be reduced by 50 % during the retrace.

2.12.5. EHT Compensation

The vertical waveform can be scaled according to the average beam current. This is used to compensate the effects of electric high-tension changes due to beam current variations. EHT compensation for East/West deflection is done with an offset corresponding to the average beam current.

2.12.6.Protection Circuitry

Picture tube and drive stage protection is provided through the following measures:

- Vertical flyback protection input:
  This pin searches for a negative edge in every field, otherwise the RGB drive signals are blanked.
- Drive shutoff during flyback:
  This feature can be selected by software.
- Safety input pin:
  This input has two thresholds. Between zero and the lower threshold, normal functioning takes place. Between the lower and the higher threshold, the RGB signals are blanked. Above the higher threshold, the RGB signals are blanked and the horizontal drive is shut off. Both thresholds have a small hysteresis.

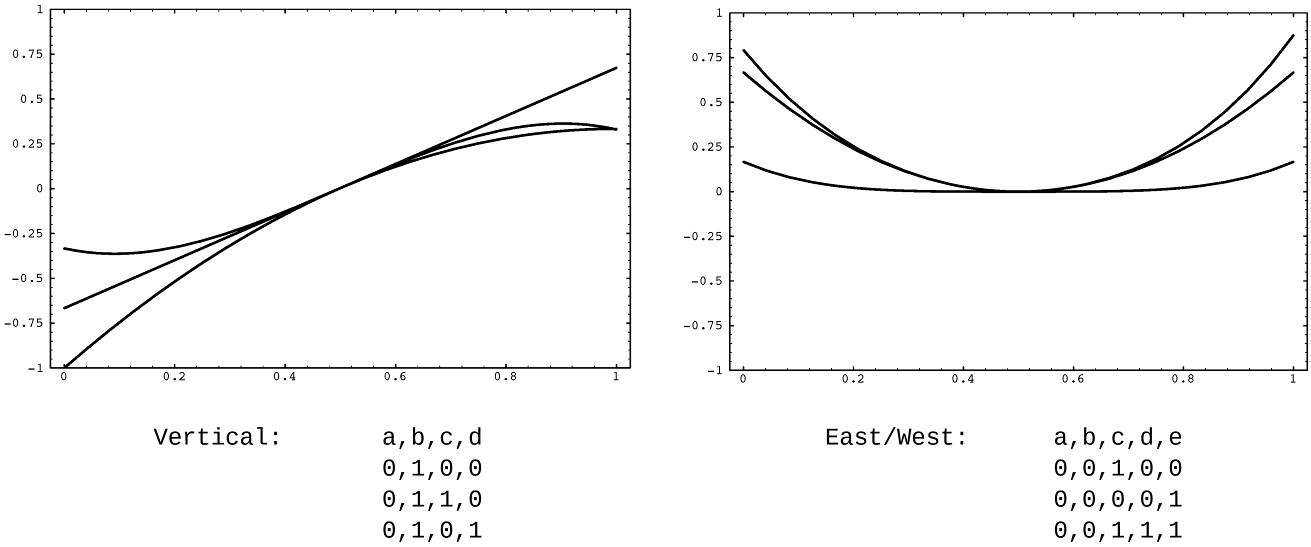


Fig. 2–25: Vertical and East/West deflection waveforms

2.13.Reset Function

Reset of all VDP functions is performed by the RESQ pin. When this pin becomes active, all internal registers and counters are lost. The TV controller can activate the RESQ pin by software (see Section 5.7.2. on page 90).

When the RESQ pin is released, the internal reset is still active for 4 µs. After that time, the initialization of all required registers is performed by the internal Fast Processor. This takes approximately 60 µs. During this initialization procedure it is not possible to access the VDP via the I^2C interface.

The VDP has clock and voltage supervision circuits to generate a stable HOUT signal. The voltage supervision activates an internal reset signal when the supply for the digital circuits ($VSUP_D$) goes below ~2.5 V for more than 50 ns. This reset signal is extended by 50 µs after $VSUP_D$ is back again.

2.14.Standby and Power-On

The VDP does not have a standby mode. To disable all the analog and digital video functions, it is necessary to switch off the supplies for analog front-end ($VSUP_{AF}$), analog back-end ($VSUP_{AB}$) and digital circuitry ($VSUP_D$).

2.15. I^{2}C Bus Slave Interface

Communication between the VDP and the TV controller is done via I^{2}C bus. For detailed information on the I^{2}C bus please refer to the Philips manual ‘I^{2}C bus Specification’.

The VDP has two I^{2}C bus slave interfaces (for compatibility with VPC/DDP applications) – one in the front-end and one in the back-end. Both I^{2}C bus interfaces use I^{2}C clock synchronization to slow down the interface if required. Both I^{2}C bus interfaces use one level of subaddress: the I^{2}C bus chip address is used to address the VDP and a subaddress selects one of the internal registers. The I^{2}C bus chip addresses are given below:

Table 2–2: I^{2}C chip addresses

Chip Address	A6	A5	A4	A3	A2	A1	A0	R/W
front-end	1	0	0	0	1	1	1	1/0
back-end	1	0	0	0	1	0	1	1/0

The registers of the VDP have 8 or 16-bit data size; 16-bit registers are accessed by reading/writing two 8-bit data words.

Fig. 2–26 shows I^{2}C bus protocols for read and write operations of the interface; the read operation requires an extra start condition and repetition of the chip address with read command set.

2.15.1. Control and Status Registers

Table 2–3 gives definitions of the VDP control and status registers. The number of bits indicated for each register in the table is the number of bits implemented in hardware, i.e. a 9-bit register must always be accessed using two data bytes but the 7 MSB will be ‘don’t care’ on write operations and ‘0’ on read operations. Write registers that can be read back are indicated in Table 2–3.

Functions implemented by software in the on-chip control microprocessor (FP) are explained in Table 2–5.

A hardware reset initializes all control registers to 0. The automatic chip initialization loads a selected set of registers with the default values given in Table 2–3.

The register modes given in Table 2–3 are

- w: write only register
- w/r: write/read data register
- r: read data from VDP
- v: register is latched with vertical sync
- h: register is latched with horizontal

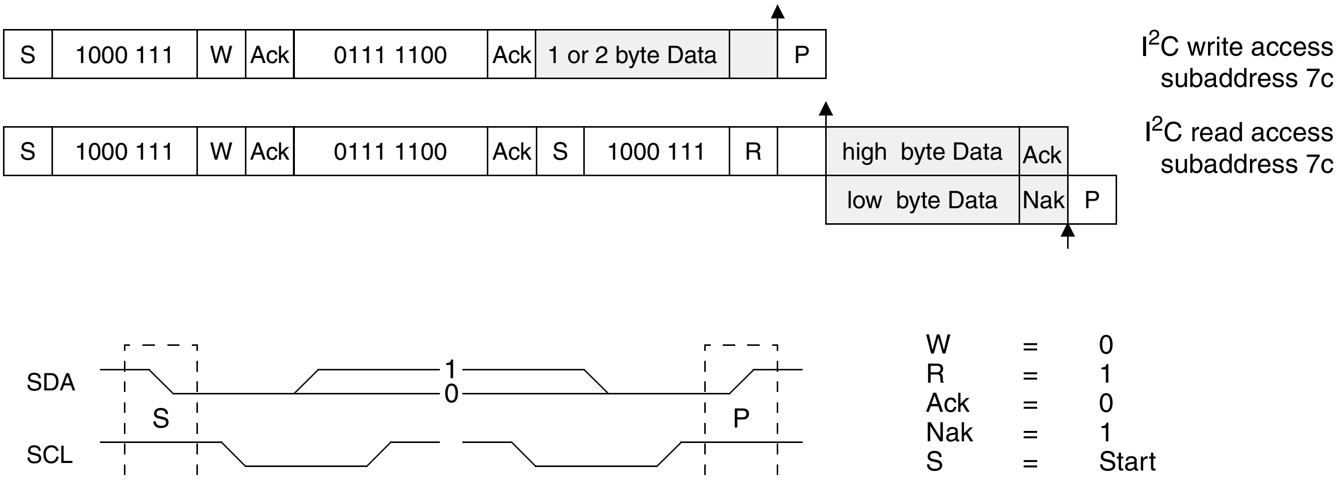


Fig. 2–26: I^{2}C bus protocols

Table 2–3: I^{2}C control and status registers of the video front-end

I^{2}C Sub address	Number of bits	Mode	Function	Default	Name
FP Interface					
h’35	8	r	FP status bit [0] write request bit [1] read request bit [2] busy		FPSTA
h’36	16	w	bit[8:0] 9-bit FP read address bit[11:9] reserved, set to zero		FPRD
h’37	16	w	bit[8:0] 9-bit FP write address bit[11:9] reserved, set to zero		FPWR
h’38	16	w/r	bit[11:0] FP data register, reading/writing to this register will autoincrement the FP read/write address. Only 16 bit of data are transferred per I^{2}C telegram.		FPDAT
Black Line Detector					
h’12	16	r	read only register, do not write to this register! after reading, LOWLIN and UPLIN are reset to 127 to start a new measurement bit[6:0] number of lower black lines bit[7] always 0 bit[14:8] number of upper black lines bit[15] normal/black picture		BLKLIN LOWLIN UPLIN BLKPIC
Miscellaneous					
h’29	16	w/r	Test pattern generator: bit[10:0] reserved (set to 0) bit[11] 0/1 disable/enable test pattern generator bit[13:12] output mode: 00 Y/C = ramp (240 ... 17) 01 Y/C = 16 10 Y/C = 90 11 Y/C = 240 bit[15:14] 0/1 reserved (set to 0)	 0 0 0 0	TPG TPGEN TPGMODE
h’22	16	w/r	NEWLINE (available for versions with panorama scaler only): bit[10:0] NEWLINE register This register defines the readout start of the next line in respect to the value of the sync counter. bit [15:11] reserved (set to 0)	 0	NEWLIN

Table 2–4: I^2C control and status registers of the video back-end

I^2C Sub address	Number of bits	Mode	Function	Default	Name
1) priority mask register If bit[x] is set to 1 then the function is active for the respective signal priority					
Luminance Channel					
h'61	9	w v	bit [5:0] 0..63/32 main picture contrast	32	CTM
h'65	9	w v	bit [5:0] 0..63/32 side picture contrast	32	CTS
h'51	9	w v	bit [8:0] –256..255 main picture brightness	0	BRM
h'55	9	w v	bit [8:0] –256..255 side picture brightness	0	BRS
h'75	9	w v	luma channel, priority mask register bit [7:0] 0/1 select contrast, brightness, matrix for main/side picture	0	PBCT 1)
h'71	9	w v	luma channel, priority mask register bit [7:0] 0/1 select main (video) / external (via CLUT) RGB	0	PBERGB 1)
Black-Level Expander					
h'59	9	w v	black-level expander bit [3:0] 0..15 tilt coefficient bit [8:4] 0...31 amount	 8 12	BLE1 BTLT BAM
h'5d	9	w v	black-level expander, threshold: bit [8:0] 0..511 disable expansion, threshold value	 200	BLE2 BTHR
h'73	9	w v	black-level expander, measurement bit[0] 0/1 50/60 Hz measurement windowlength bit [8:1] 0..255 vstart/2 start line = vstart stop line = 336/283 – vstart or vertical blanking	 0 15	BLE3 BWL BVST
h'7d	9	w v	black-level expander, priority mask register bit [7:0] 0/1 enable/disable black-level expander	0	PBBLE 1)
Dynamic Peaking					
h'69	9	w v	luma peaking filter, the gain at high frequencies and small signal amplitudes is: 1 + (k1+k2)/8 bit [3:0] 0..15 k1: peaking level undershoot bit [7:4] 0..15 k2: peaking level overshoot bit [8] 0/1 peaking value normal/inverted (peaking/softening)	 4 4 0	PK1 PKUN PKOV PKINV
h'6d	9	w v	luma peaking filter, coring bit [4:0] 0..31 coring level bit [7:5] reserved bit [8] 0/1 peaking filter center frequency high/low	 3 0	PK2 COR PFS
h'79	9	w v	luma peaking filter, priority mask register bit [7:0] 0/1 disable/enable peaking transient suppression when signal is switched	0	PBPK 1)

I^2C Sub address	Number of bits	Mode	Function	Default	Name
Soft Limiter					
h’41	9	w v	luma soft limiter, slope A and B bit [3:0] slope segment A bit [7:4] slope segment B	 0 0	LSL1 LSLSA LSLSB
h’45	9	w v	luma soft limiter, absolute limit bit [7:0] luma soft limiter absolute limit (unsigned) bit [8] 0/1 modulation off/on	 255 1	LSL2 LSLAL LSLM
h’49	9	w v	bit [8:0] luma soft limiter segment B tilt point (unsigned)	300	LSLTB
h’4d	9	w v	bit [8:0] luma soft limiter segment A tilt point (unsigned)	250	LSLTA
Chrominance Channel					
h’14	8	w/r	luma/chroma matching bit [2:0] –3...3 variable chroma delay bit [7:3] reserved, set to 0	 0	LCM CDEL
h’5e	9	w v	digital transient improvement bit [3:0] 0..15 coring value bit [7:4] 0..15 DTI gain bit [8] 0/1 narrow/wide bandwidth mode	 1 5 1	DTI DTICO DTIGA DTIMO
Inverse Matrix					
 h’7c h’74	 9 9	 w v w v	main picture matrix coefficient R–Y = MR1M*C_B + MR2M*C_R bit [8:0] –256/128 ... 255/128 bit [8:0] –256/128 ... 255/128	 0 86	 MR1M, MR2M
 h’6c h’64	 9 9	 w v w v	main picture matrix coefficient G–Y = MG1M*C_B + MG2M*C_R bit [8:0] –256/128 ... 255/128 bit [8:0] –256/128 ... 255/128	 –22 –44	 MG1M, MG2M
 h’5c h’54	 9 9	 w v w v	main picture matrix coefficient B–Y = MB1M*C_B + MB2M*C_R bit [8:0] –256/128 ... 255/128 bit [8:0] –256/128 ... 255/128	 113 0	 MB1M, MB2M
 h’78 h’70	 9 9	 w v w v	side picture matrix coefficient R–Y = MR1S*C_B + MR2S*C_R bit [8:0] –256/128 ... 255/128 bit [8:0] –256/128 ... 255/128	 0 73	 MR1S, MR2S
 h’68 h’60	 9 9	 w v w v	side picture matrix coefficient G–Y = MG1S*C_B + MG2S*C_R bit [8:0] –256/128 ... 255/128 bit [8:0] –256/128 ... 255/128	 –19 –37	 MG1S, MG2S
 h’58 h’50	 9 9	 w v w v	side picture matrix coefficient B–Y = MB1S*C_B + MB2S*C_R bit [8:0] –256/128 ... 255/128 bit [8:0] –256/128 ... 255/128	 97 0	 MB1S, MB2S

I^{2}C Sub address	Number of bits	Mode	Function	Default	Name
Color Lookup Table					
h’00– h’0f	16	w h	color look-up table : 16 entries, 12 bit wide, The CLUT registers are initialized at power-up bit [3:0] 0..15 blue amplitude bit [7:4] 0..15 green amplitude bit [11:8] 0..15 red amplitude	000h f00h 0f0h ff0h 00fh f0fh 0ffh fffh 7ffh 700h 070h 770h 007h 707h 077h 777h	CLUT0 CLUT15
h’4c	9	w v	digital OSD insertion contrast for R (amplitude range: 0 to 255) bit [3:0] 0..13 R amplitude = CLUTn · (DRCT + 4) 14,15 invalid picture frame insertion contrast for R (ampl. range: 0 to 255) bit [7:4] 0..13 R amplitude = PFCR · (PFRCT + 4) 14,15 invalid	 8 8	RCT DRCT PFRCT
h’48	9	w v	digital OSD insertion contrast for G (amplitude range: 0 to 255) bit [3:0] 0..13 G amplitude = CLUTn · (DGCT + 4) 14,15 invalid picture frame insertion contrast for G (ampl. range: 0 to 255) bit [7:4] 0..13 G amplitude = PFCG · (PFGCT + 4) 14,15 invalid	 8 8	GCT DGCT PFGCT
h’44	9	w v	digital OSD insertion contrast for B (amplitude range: 0 to 255) bit [3:0] 0..13 B amplitude = CLUTn · (DBCT + 4) 14,15 invalid picture frame insertion contrast for B (ampl. range: 0 to 255) bit [7:4] 0..13 B amplitude = PFCB · (PFBCT + 4) 14,15 invalid	 8 8	BCT DBCT PFBCT
Picture Frame Generator					
h’11	16	w h	picture frame color bit [3:0] 0..15 blue amplitude bit [7:4] 0..15 green amplitude bit [11:8] 0..15 red amplitude	 0 0 0	PFC PFCB PFCG PFCR
h’47	9	w v	bit [2:0] picture frame generator priority id bit [8] enable prio id for picture frame generator	0	PFGID PFGEN
h’4F	9	w v	bit [8:0] horizontal picture frame begin code 0 = picture frame generator horizontally disabled code 1FF = full frame	0	PFGHB
h’53	9	w v	bit [8:0] horizontal picture frame end	0	PFGHE
h’63	9	w v	bit [8:0] vertical picture frame begin code 0 = picture frame generator vertically disabled	270	PFGVB
h’6f	9	w v	bit [8:0] vertical picture frame end	56	PFGVE

I^{2}C Sub address	**Number of bits**	**Mode**	**Function**	**Default**	**Name**
			Scan Velocity Modulation		
h’5a	9	w v	video mode coefficients bit [5:0] gain1 bit [8:6] differentiator delay 1 (0= filter off, 1...6= delay)	 60 4	SVM1 SVG1 SVD1
h’56	9	w v	text mode coefficients bit [5:0] gain 2 bit [8:6] differentiator delay 2 (0= filter off, 1...6= delay)	 60 4	SVM2 SVG2 SVD2
h’52	9	w v	limiter bit [6:0] limit value bit [8:5] not used, set to “0”	 100 0	SVM3 SVLIM
h’4e	9	w v	delay and coring bit [3:0] adjustable delay, in 1/2 display clock steps, (value 5 : delay of SVMOUT is the same as for RGBOUT bit [7:4] coring value bit [8] not used, set to “0”	 7 0	SVM4 SVDEL SVCOR
			Display Controls		
h’4a	9	w v	cutoff Red	0	CR
h’46	9	w v	cutoff Green	0	CG
h’42	9	w v	cutoff Blue	0	CB
			Tube- and Picture-Measurements		
h’7b	9	w v	picture measurement start line bit [8:0] (TML+9)..511 first line of picture measurement	 23	PMST
h’6b	9	w v	picture measurement stop line bit [8:0] (PMST+1)..511 last line of picture measurement	 308	PMSO
h’7f	9	w v	tube measurement line bit [8:0] 0..511 start line for tube measurement	 15	TML
h’25	8	w/r	tube and picture measurement control bit [0] 0/1 disable/enable tube measurement bit [1] 0/1 80/40 kHz bandwidth for picture measurement bit [2] 0/1 disable/enable picture measurement (writing a ’1’ starts one measurement cycle) bit [3] 0/1 large/small picture measurement window, will be disabled from bit[3] in address h’32 bit [4] 0/1 measure / offset check for adc bit [7:5] reserved	0	TPM TMEN PMBW PMEN PMWIN OFSEN
h’13	16	w/r	white drive measurement control bit [9:0] 0..1023 RGB values for white drive beam current measurement bit [10] reserved bit [11] 0/1 RGB values for white drive beam current measurement disabled/enabled	 512 0	WDM WDRV EWDM

I^2C Sub address	Number of bits	Mode	Function	Default	Name
 h'18 h'19 h'1a h'1d h'1c h'1b	8	r	measurement result registers minimum in active picture maximum in active picture white drive cutoff/leakage red cutoff/leakage green cutoff/leakage blue, read pulse starts tube measurement	–	 MRMIN MRMAX MRWDR MRCR MRCG MRCB
h'1e	8	r	measurement adc status and Fast-Blank input status measurement status register bit [0] 0/1 tube measurement active / complete bit [2:1] white drive measurement cycle 00 red 01 green 10 blue 11 reserved bit [3] 0/1 picture measurement active / complete bit [4] 0/1 Fast-Blank input Low / High (static) bit [5] 1 Fast-Blank input negative transition since last read (bit reset at read) bit [7:6] reserved	–	PMS
			Vertical Timing		
h'67	9	w v	vertical blanking start bit [8:0] 0..511 first line of vertical blanking	305	VBST
h'77	9	w v	vertical blanking stop bit [8:0] 0..511 last line of vertical blanking	25	VBSO
h'5f	9	w v	vertical free run period bit [8:0] free running field period = (value+4) lines	309	VPER
			Horizontal Deflection and Timing		
h'7a	9	w v	quadratic term of angle & bow correction bit [8:0] –256..+255 (± 500 ns)	0	BOW
h'76	9	w v	linear term of angle & bow correction bit [8:0] –256..+255 (± 500 ns)	0	ANGLE
h'6e	9	w v	adjustable delay of PLL2, clamping, and blanking (relative to front sync) adjust clamping pulse for analog RGB input bit [8:0] –256..+255 (± 8 µs)	–141	POFS2
h'72	9	w v	adjustable delay of flyback, main sync, csync and analog RGB (relative to PLL2) adjust horizontal drive or csync bit [8:0] –256..+255 (± 8 µs)	0	POFS3
h'7e	9	w v	adjustable delay of main sync (relative to flyback) adjust horizontal position for digital picture bit [8:0] 20 steps=1 µs	120	HPOS
h'5b	9	w v	start of horizontal blanking bit [8:0] 0..511	1	HBST

I^2C Sub address	Number of bits	Mode	Function	Default	Name
h’57	9	w v	end of horizontal blanking bit [8:0] 0..511	48	HBSO
			PLL2/3 filter coefficients, 1of5 bit code (n+ set bit number)		
h’62	9	w v	bit [5:0] proportional coefficient PLL3, 2^{-n-1}	2	PKP3
h’66	9	w v	bit [5:0] proportional coefficient PLL2, 2^{-n-1}	1	PKP2
h’6a	9	w v	bit [5:0] integral coefficient PLL2, 2^{-n-5}	2	PKI2
h’15	16	w/r	horizontal drive and vertical signal control register		HVC
			bit [5:0] 0..63 horizontal drive pulse duration in µs (internally limited to 4..61)	32	HDRV
			bit [6] 0/1 disable/enable horizontal PLL2 and PLL3	0	EHPLL
			bit [7] 0/1 1: disable horizontal drive pulse during flyback	0	EFLB
			bit [8] reserved, set to ’0’	0	
			bit [9] 0/1 enable/disable ultra black blanking	0	DUBL
			bit [10] 0/1 0: all outputs blanked 1: normal mode	1	EBL
			bit [11] 0/1 enable/disable clamping for analog RGB input	0	DCRGB
			bit [12] 0/1 disable/enable vertical free running mode (FIELD is set to field2, no interlace)	0	SELFT
			bit [13] 0/1 enable/disable vertical protection	0	DVPR
			bit [14] reserved, set to ’0’	0	
			bit [15] 0/1 disable/enable phase shift of display clock	1	DISKA
h’9d	8	w/r	sync output control	0	SYCTRL
			bit [0] invert INTLC		INTLCINV
			bit [4:1] reserved, set to ’0’		
			bit [5] force INTLC to polarity defined in ‘INTLCINV’		INTLCFO
Miscellaneous					
h’32	8	w/r	Fast-Blank interface mode	0	FBMOD
			bit [0] 0 internal Fast-Blank from FBLIN pin 1 force internal Fast-Blank signal to High		FBFOH
			bit [1] 0/1 internal Fast-Blank active High/Low		FBPOL
			bit [2] 0/1 disable/enable clamping reference for RGB outputs		CLMPR
			bit [3] 1 full line MADC measurement window, disables bit [3] in address h’25		FLMW
			bit [4] 0/1 horizontal flyback input active High/Low		FLPOL
			bit [6:5] reserved (set to 0)		
			bit [7] vertical output select 0 VERTQ output 1 INTLC output		VOS
h’4b	9	w v	Fast-Blank input, priority mask register bit [7:0] 0/1 disable/enable analog Fast-Blank input	0	PBFB 1)

Table 2–5: Control registers of the Fast Processor for control of the video front-end functions

– default values are initializied at reset

FP Sub-address	Function	Default	Name
	Standard Selection		
h’20	Standard select:	0	SDT
	bit[2:0] standard		
	0 PAL B,G,H,I (50 Hz) 4.433618		PAL
	1 NTSC M (60 Hz) 3.579545		NTSC
	2 SECAM (50 Hz) 4.286		SECAM
	3 NTSC44 (60 Hz) 4.433618		NTSC44
	4 PAL M (60 Hz) 3.575611		PALM
	5 PAL N (50 Hz) 3.582056		PALN
	6 PAL 60 (60 Hz) 4.433618		PAL60
	7 NTSC COMB (60 Hz) 3.579545		NTSCC
	bit[3] 0/1 standard modifier		SDTMOD
	PAL modified to simple PAL		
	NTSC modified to compensated NTSC		
	SECAM modified to monochrome 625		
	NTSCC modified to monochrome 525		
	bit[4] reserved (set to 0)		
	bit[5] 0/1 2-H comb filter off/on		COMB
	bit[6] 0/1 S-VHS mode off/on (2-H comb is switched off)		SVHS
	Option bits allow to suppress parts of the initialization, this can be used for color standard search:		SDTOPT
	bit[7] no hpll setup		
	bit[8] no vertical setup		
	bit[9] no acc setup		
	bit[10] 2-H comb filter set-up only		
	bit[11] status bit, normally write 0. After the FP has switched to a new standard, this bit is set to 1 to indicate operation complete. Standard is automatically initialized when the insel register is written.		
h’148	Enable automatic standard recognition (ASR)	0	ASR_ENA
	bit[0] 0/1 PAL B,G,H,I (50 Hz) 4.433618		
	bit[1] 0/1 NTSC M (60 Hz) 3.579545		
	bit[2] 0/1 SECAM (50 Hz) 4.286		
	bit[3] 0/1 NTSC44 (60 Hz) 4.433618		
	bit[4] 0/1 PAL M (60 Hz) 3.575611		
	bit[5] 0/1 PAL N (50 Hz) 3.582056		
	bit[6] 0/1 PAL 60 (60 Hz) 4.433618		
	bit[10:7] reserved set to 0		
	bit[11] 1 reset status information ‘switch’ in asr_status (cleared automatically)		
	0: disable recognition; 1: enable recognition Note: For correct operation don’t change FP reg. 20h and 21h, while ASR is enabled!		

FP Sub-address	Function			Default	Name
h’14e	Status of automatic standard recognition			0	ASR_STATUS
	bit[0]	1	error of the vertical standard (neither 50 nor 60 Hz)		VWINERR
	bit[1]	1	detected standard is disabled		DISABLED
	bit[2]	1	search active		BUSY
	bit[3]	1	search terminated, but failed		FAILED
	bit[4]	1	no color found		NOCOLOR
	bit[5]	1	standard has been switched (since last reset of this flag with bit[11] of asr_enable)		SWITCH
	bit[4:0]	00000	all ok		
		00001	search not started, because vwin error detected (no input or SECAM L)		
		00010	search not started, because detected vert. standard not enabled		
		0x1x0	search started and still active		
		01x00	search failed (found standard not correct)		
		01x10	search failed, (detected color standard not enabled)		
		10000	no color found (monochrome input or switch betw. CVBS/SVHS necessary)		
h’21	Input select:		writing to this register will also initialize the standard		INSEL
	bit[1:0]		luma selector		
		00	VIN1	00	VIS
		01	VIN2		
		10	VIN3		
		11	VIN4		
	bit[2]		chroma selector		
		0	CIN1	0	CIS
		1	CIN2		
	bit[4:3]		IF compensation		
		00	off	00	IFC
		01	6 dB/Okt		
		10	12 dB/Okt		
		11	10 dB/MHz only for SECAM		
	bit[6:5]		chroma bandwidth selector		
		00	narrow	01	CBW
		01	normal		
		10	broad		
		11	wide		
	bit[7]	0/1	adaptive/fixed SECAM notch filter		
	bit[8]	0/1	enable luma lowpass filter		FNTCH
	bit[10:9]		hpll speed		LOWP
		00	no change		HPLLMD
		01	terrestrial		
		10	vcr		
		11	mixed		
	bit[11]		status bit, write 0, this bit is set to 1 to indicate operation complete.		
h’22	Available for versions with panorama scaler only! picture start position, this register sets the start point of active video, this can be used e.g. for panning. The setting is updated when ’sdt’ register is updated.			0	SFIF
h’23	luma/chroma delay adjust. The setting is updated when ’sdt’ register is updated.			0	LDLY
	bit[5:0]		reserved, set to zero		
	bit[11:6]		luma delay in clocks, allowed range is +1 ... –7		

FP Sub-address	Function	Default	Name
h’2f	YC_rC_b mode control register bit[6:0] reserved (set to 0) bit[7] 1 ADC over-/underflow (has to be reset after read if used) bit[8] 0 disable/enable YC_rC_b bit[9] ADC range 0 nominal input amplitude (±350 mV) 1 extented input amplitude (±500 mV) bit[11:10] reserved (set to 0) Note: Activate the YC_rC_b mode by – enabling YC_rC_b – selecting simple PAL or NTSC M, svhs=1, comb=0 in the std register – setting cbw=2 in the insel register	0	YC_rC_b
	Comb Filter		
h’27	comb filter control register bit[0] 0 comb coefficients are calculated for luma/chroma 1 comb coefficients for luma are used for luma and chroma bit[1] 0 luma comb strength depends on signal amplitude 1 luma comb strength is independent of amplitude bit[2] 0 reduced comb booster 1 max comb booster bit[4:3] 0..3 comb strength for chroma signal bit[6:5] 0..3 comb strength for luma signal bit[11:7] 0..31 overall limitation of the calculated comb coefficients 0 no limitation 31 max limitation (1/2)	 0 0 1 3 2 0	CMB_UC CC DAA KB KC KY CLIM
	Color Processing		
h’30	Saturation control bit[11:0] 0...4094 (2070 corresponds to 100% saturation) 4095 disabled (test mode only)	2070	ACC_SAT
h’39	amplitude killer level (0:killer disabled)	25	KILVL
h’3a	amplitude killer hysteresis	5	KILHY
h’dc	NTSC tint angle, $\pm 512 = \pm\pi/4$	0	TINT
	DVCO		
h’f8	crystal oscillator center frequency adjust, –2048 ... 2047	–720	DVCO
h’f9	crystal oscillator center frequency adjustment value for line-lock mode True adjust value is DVCO – ADJUST. For factory crystal alignment, using standard video signal: set DVCO = 0, set lock mode, read crystal offset from ADJUST register and use negative value for initial center frequency adjustment via DVCO.	read only	ADJUST
h’f7	crystal oscillator line-locked mode, lock command/status write: 100 enable lock 0 disable lock read: 0 unlocked >2047 locked	0	XLCK
h’b5	crystal oscillator line-locked mode, autolock feature. If autolock is enabled, crystal oscillator locking is started automatically. bit[11:0] threshold; 0: autolock off	400	AUTOLOCK

FP Sub-address	Function	Default	Name
	FP Status		
h’12	general purpose control bits bit[2:0] reserved, do not change bit[3] vertical standard force bit[8:4] reserved, do not change bit[9] disable flywheel interlace bit[11:10] reserved, do not change to enable vertical free run mode set vfrc to 1 and dflw to 0	 0 1	GPC VFRC DFLW
h’13	standard recognition status bit[0] 1 vertical lock bit[1] 1 horizontally locked bit[2] no signal detected bit[3] 1 color amplitude killer active bit[4] 1 disable amplitude killer bit[5] 1 color ident killer active bit[6] 1 disable ident killer bit[7] 1 interlace detected bit[8] 1 no vertical sync detection bit[9] 1 spurious vertical sync detection bit[11:10] reserved	–	ASR
h’14	input noise level	read only	NOISE
h’cb	number of lines per field, P/S: 312, N: 262	read only	NLPF
h’15	vertical field counter, incremented per field		VCNT
h’74	measured sync amplitude value, nominal: 768 (PAL), 732 (NTSC)	read only	SAMPL
h’36	measured burst amplitude	read only	BAMPL
h’f0	firmware version number bit[7:0] internal revision number bit[11:8] firmware release	read only	SW_VERSION
h’170	status of macrovision detection bit[0] AGC pulse detected bit[1] pseudo sync detected	read only	MCV_STATUS
h’171	bit[11:0] first line of macrovision detection window	6	MCV_START
h’172	bit[11:0] last line of macrovision detection window	15	MCV_STOP
	Horizontal Scaler		
	[1] these registers are updated when the scaler mode register is written		
h’40	scaler mode register bit[1:0] scaler mode 0 linear scaling mode 1 nonlinear scaling mode, ’panorama’ 2 nonlinear scaling mode, ’waterglass’ 3 reserved bit[10:2] reserved, set to 0 bit[11] scaler update 0 start scaler update command, when the registers are updated the bit is set to 1	0	SCMODE MODE SCUP

FP Sub-address	Function	Default	Name
h'41	luma offset register 1) bit[6:0] luma offset 0..127 ITU-R output format: 57 CVBS output format: 4	57	YOFFS
h'42	active video length for 1-h FIFO 1) bit[11:0] length in pixels	1080	FFLIM
h'43	scaler1 compression coefficient 1) For compression by a factor c the value c*1024 is required. bit[11:0] allowed values from 1024..4095	1024	SCINC1
h'44	scaler2 expansion coefficient 1) For expansion by a factor c the value 1/c*1024 is required. bit[11:0] allowed values from 256..1024	1024	SCINC2
h'45	scaler1/2 nonlinear scaling coefficient 1)	0	SCINC
h'47 – h'4b	scaler1 window controls 1) 5 12-bit registers for control of the nonlinear scaling	0	SCW1_0 – 4
h'4c – h'50	scaler2 window controls 1) 5 12-bit registers for control of the nonlinear scaling	0	SCW2_0 – 4

2.15.1.1. Scaler Adjustment

In case of linear scaling, most of the scaler registers need not be set. Only the scaler mode, active video length, and the fixed scaler increments (scinc1/scinc2) must be written.

The adjustment of the scaler for nonlinear scaling modes should use the parameters given in Table 2–6.

Table 2–6: Set-up values for nonlinear scaler modes

Register	Scaler Modes			
	'waterglass' border 35%		'panorama' border 30%	
	center compression			
	3/4	5/6	4/3	6/5
scinc1	1643	1427	1024	1024
scinc2	1024	1024	376	611
scinc	90	56	85	56
fflim	945	985	921	983
scw1 – 0	110	115	83	94
scw1 – 1	156	166	147	153
scw1 – 2	317	327	314	339
scw1 – 3	363	378	378	398
scw1 – 4	473	493	461	492
scw2 – 0	110	115	122	118
scw2 – 1	156	166	186	177
scw2 – 2	384	374	354	363
scw2 – 3	430	425	418	422
scw2 – 4	540	540	540	540

Table 2–7: Control Registers of the Fast Processor for control of the video backend functions

– default values are initializied at reset

FP Sub-address	Function	Default	Name
FP Display Control Register			
h’130	White Drive Red (0...1023)	700	WDR [1]
h’131	White Drive Green (0...1023)	700	WDG [1]
h’132	White Drive Blue (0...1023)	700	WDB [1]
h’139	Internal Brightness, Picture (0...511), the center value is 256, the range allows for both increase and reduction of brightness.	256	IBR
h’13c	Internal Brightness, Measurement (0...511), the center value is 256, the brightness for measurement can be set to measure at higher cutoff current. The measurement brightness is independent of the drive values.	256	IBRM
h’13a	Analog Brightness for external RGB (0...511), the center value is 256, the range allows for both increase and reduction of brightness.	256	ABR
h’13b	Analog Contrast for external RGB (0...511)	350	ACT
1) The white drive values will become active only after writing the blue value WDB, latching of new values is indicated by setting the MSB of WDB.			
FP Display Control Register, BCL			
h’144	BCL threshold current, 0...2047 (max ADC output ~1152)	1000	BCLTHR
h’142	BCL time constant 0...15 ➠13 ... 1700 msec	15	BCLTM
h’143	BCL loop gain. 0..15	0	BCLG
h’145	BCL minimum contrast 0...1023	307	BCLMIN
h’105	Test register for BCL/EHT comp. function, register value: 0 normal operation 1 stop ADC offset compensation x>1 use x in place of input from Measurement ADC	0	BCLTST
FP Display Control Register, Deflection			
h’103	interlace offset, –2048..2047 This value is added to the SAWTOOTH output during one field.	0	INTLC
h’102	discharge sample count for deflection retrace, SAWTOOTH DAC output impedance is reduced for DSCC lines after vertical retrace.	7	DSCC
h’11f	vertical discharge value, SAWTOOTH output value during discharge operation, typically same as A0 init value for sawtooth.	–1365	DSCV
h’10b	EHT compensation vertical gain coefficient, 0...511	0	EHTV
h’10a	EHT compensation time constant, 0...15 --> 3.2..410 msec	15	EHTTM
h’10f	EHT compensation east/west gain coefficient, –1024...1023	15	EHTEW

FP Display Control Register, Vertical Sawtooth			
h’110	DC offset of SAWTOOTH output This offset is independent of EHT compensation.	0	OFS
h’11b	accu0 init value	–1365	A0
h’11c	accu1 init value	900	A1
h’11d	accu2 init value	0	A2
h’11e	accu3 init value	0	A3
FP Display Control Register, East-West Parabola			
h’12b	accu0 init value	–1121	A0
h’12c	accu1 init value	219	A1
h’12d	accu2 init value	479	A2
h’12e	accu3 init value	–1416	A3
h’12f	accu4 init value	1052	A4

2.15.1.2. Calculation of Vertical and East-West Deflection Coefficients

In Table 2–8 the formula for the calculation of the deflection initialization parameters from the polynominal coefficients a,b,c,d,e is given for the vertical and East-West deflection. Let the polynomial be:

$$P = a + b(x - 0.5) + c(x - 0.5)^2 + d(x - 0.5)^3 + e(x - 0.5)^4$$

The initialization values for the accumulators a0..a3 for vertical deflection and a0..a4 for East-West deflection are 12-bit values. The coefficients that should be used to calculate the initialization values for different field frequencies are given below, the values must be scaled by 128, i.e. the value for a0 of the 50 Hz vertical deflection is:

$$a0 = (a \cdot 128 - b \cdot 1365.3 + c \cdot 682.7 - d \cdot 682.7) / 128$$

Table 2–8: Calculation of Initialization values for Vertical Sawtooth and East-West Parabola

Vertical Deflection 50 Hz				
	a	b	c	d
a0	128	–1365.3	+682.7	–682.7
a1		899.6	–904.3	+1363.4
a2			296.4	–898.4
a3				585.9

Vertical Deflection 60 Hz				
	a	b	c	d
a0	128	–1365.3	+682.7	–682.7
a1		1083.5	–1090.2	+1645.5
a2			429.9	–1305.8
a3				1023.5

East-West Deflection 50 Hz					
	a	b	c	d	e
a0	128	–341.3	1365.3	–85.3	341.3
a1		111.9	–899.6	84.8	–454.5
a2			586.8	–111.1	898.3
a3				72.1	–1171.7
a4					756.5

East-West Deflection 60 Hz					
	a	b	c	d	e
a0	128	–341.3	1365.3	–85.3	341.3
a1		134.6	–1083.5	102.2	–548.4
a2			849.3	–161.2	1305.5
a3				125.6	–2046.6
a4					1584.8

3. Text and OSD Processing

3.1. Introduction

The VCT 38xxA includes a World System Teletext (WST) decoder, whose display capabilities are also used for OSD generation. In the following sections the text and OSD processing part of the VCT 38xxA will be named TPU for short.

With integrated CPU, RAM and ROM, an adaptive data slicer, a display controller, and a number of interfaces, the TPU offers acquisition and display of various teletext and data services such as WST, PDC, VPS, and WSS. Fig. 3–1 shows the functional block diagram of the TPU.

The TPU operates independently from the TV controller and can be controlled by software via I^{2}C bus interface (see Section 3.14. on page 82). The TV controller is not burdened with the task of teletext decoding and communicates with the TPU via a high-level command language.

The TPU performs the following tasks:

– teletext data acquisition	(hardware)
– teletext data decoding	(software)
– page generation	(software)
– page memory management	(software)
– page display	(hardware)
– user interface	(software)

3.2. SRAM Interface

The SRAM interface connects a standard SRAM to the internal bus structure. The address bus is 19 bit wide, addressing SRAMs up to 4 Mbit. Smaller SRAMs can also be connected.

The SRAM interface has to handle 3 asynchronous data streams. The CPU needs access to every memory location of the SRAM. During VBI the slicer writes up to 22 teletext lines of 43 Bytes into the acquisition scratch memory. During text display the display controller copies teletext rows from display memory into its internal row buffer.

On VCT 38xxA the SRAM interface of the TPU is connected to the memory bus of the TV controller. This is done to save pins and to give the TV controller faster access to the display memory. Refer to DMA Interface (chapter 5.9. on page 96) for more details.

After reset the TPU will not use the SRAM interface until receiving the I^{2}C command “DRAM_MODE” (see Section 3.12. on page 68).

3.3. Text Controller

The TPU operates with its own 65C02 core running at 10.125 MHz. The core can address up to 64 kBytes of memory.

The CPU memory contains 640 Bytes RAM, 12 kBytes program ROM and 12 kBytes character ROM. The character ROM holds the font data and is separated from the program ROM to save CPU time. The CPU can still access the character ROM via a DMA interface including wait cycles. The display controller can also access the CPU memory via the same DMA interface. By this means it is possible to locate part of the character font in program ROM or part of the program code in character ROM.

Table 3–1: Memory map of text controller

Interrupt Vector	**Absolute Address (High Byte, Low Byte)**
IRQ	FFFF, FFFE
Reset	FFFD, FFFC
NMI	FFFB, FFFA
Control Word	FFF9
Memory Segment	**Absolute Address**
Zero Page	0000 – 00FF
Stack Page	0100 – 01FF
OSD Buffer	0100 – 019F
I/O Page	0200 – 02FF
Extra Page	0300 – 037F
Character ROM	5000 – 7FFF
Program ROM	D000 – FFFF

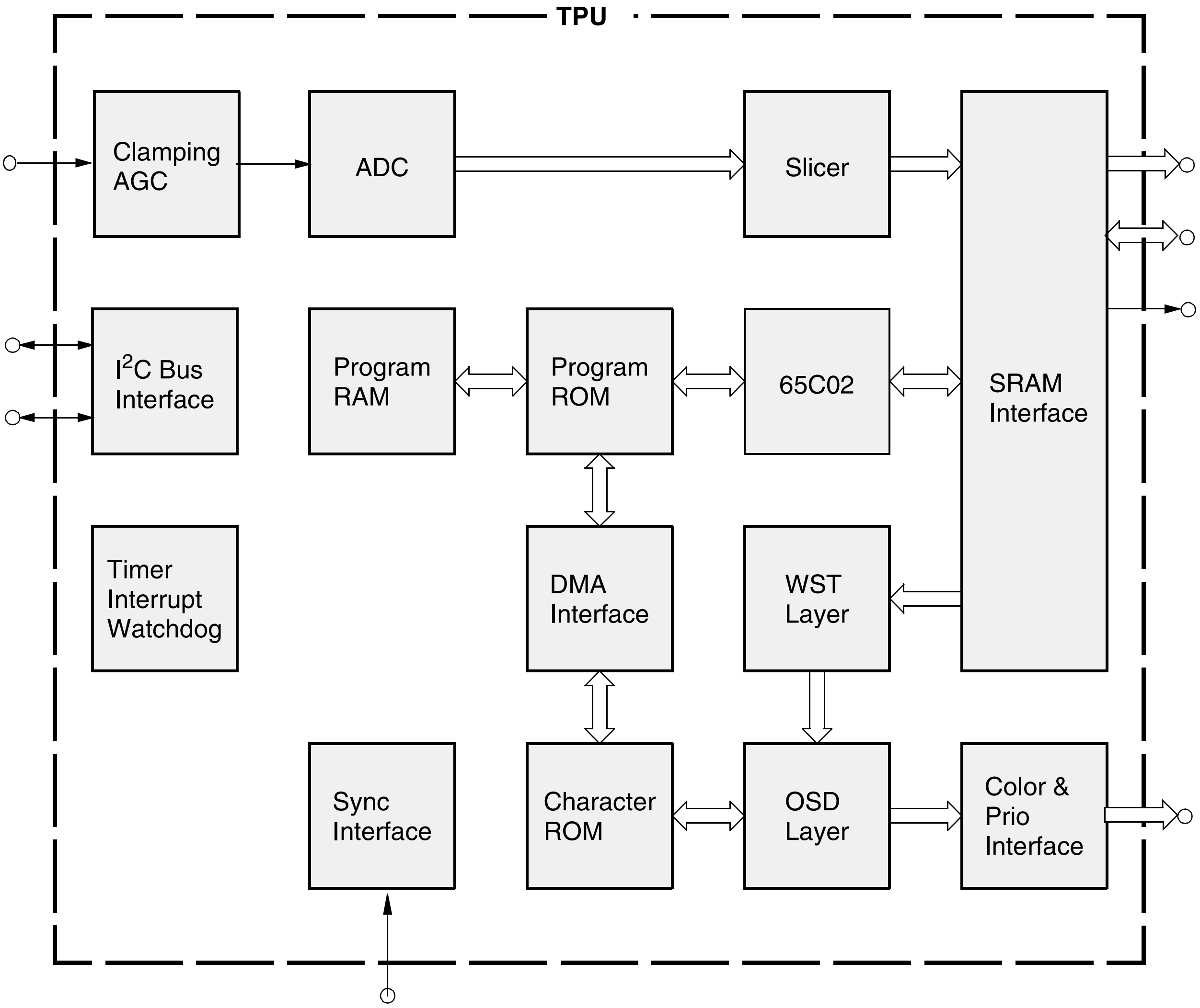


Fig. 3–1: Block diagram of the TPU

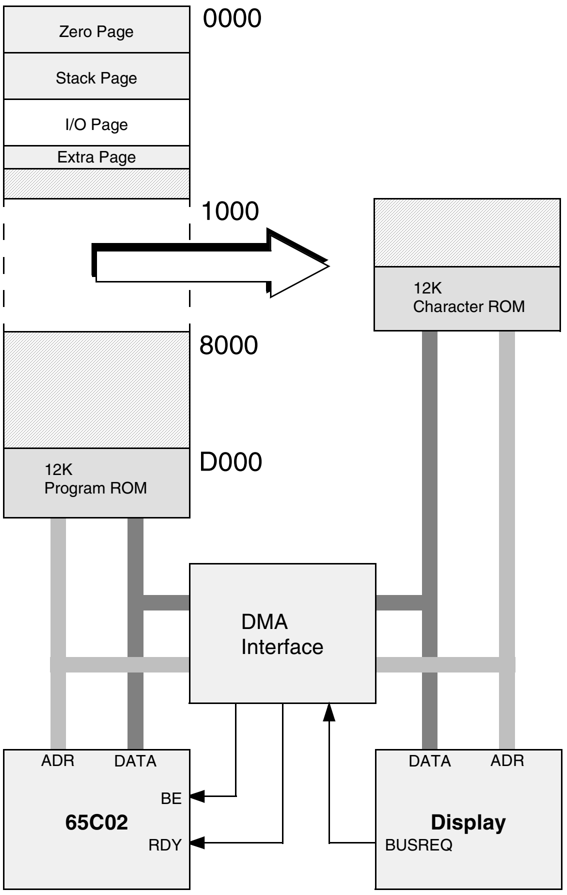


Fig. 3–2: Memory environment of text controller

3.4. Teletext Acquisition

The only task of the slicer circuit is to extract teletext lines from the incoming composite video signal and to store them into the acquisition scratch buffer of the internal/external SRAM. No page selection is done at this hardware level.

Four analog sources can be connected, thus it is possible to receive text from one channel while watching another on the screen. After clamping and AGC amplifier the analog video signal is converted into binary data. Sync separation is done by a sync slicer and a horizontal PLL, which generate the horizontal and vertical timing. By these means, no external sync signals are needed and any available signal source can be used for teletext reception.

The teletext information itself is acquired using adaptive slicers on bit and byte level with soft error detection to decrease the bit error rate under bad reception conditions. The slicer can be programmed to different bit rates for reception of PAL, NTSC or MAC world system teletext as well as VPS, WSS, or CAPTION signals.

3.5. Teletext Page Management

As a state-of-the-art teletext decoder, the TPU is able to store and manage a sufficient number of teletext pages to absorb the annoying transmission cycle times. The number of available pages is only limited by the memory size. With an intelligent software and a 4-Mbit SRAM it is possible to store and to control more than 500 teletext pages.

The management of such a data base is a typical software task and is therefore performed by the 65C02. Using a fixed length page table with one entry for every possible page, the software distributes the content of the acquisition scratch buffer among the page memory. The page size is fixed to 1 kByte, only ghost rows are chained in 128-Byte segments to avoid unused memory space.

A stored teletext page cannot be displayed directly, because of the row-adaptive transmission and the level 2 enhancements (row 26–29). Therefore, the CPU has to transfer the selected teletext page into a display page buffer, adding extra data such as character set extension and non-spacing attributes.

3.5.1. Memory Manager

The Memory manager is the core of the internal TPU firmware. Most of the acquisition and display related functions are controlled by this management.

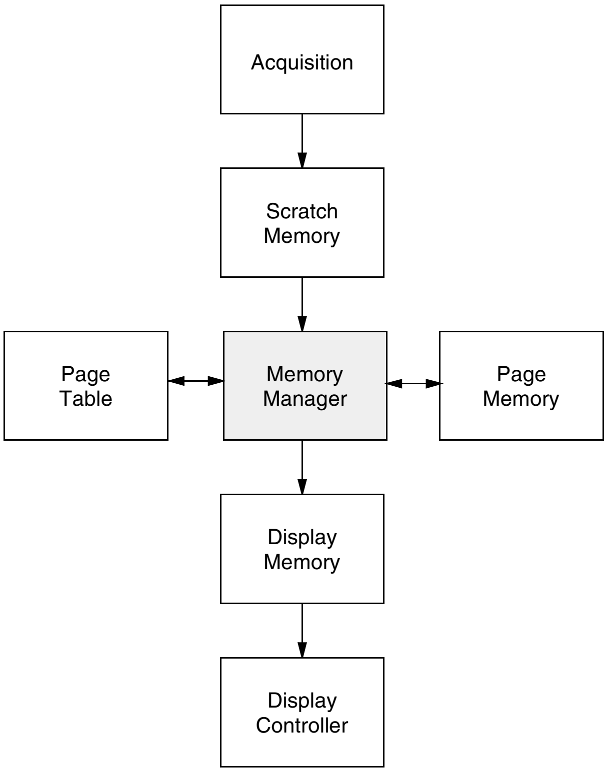


Fig. 3–3: Memory Manager

3.5.2. Memory Organization

The upper end of the memory is defined by the SRAM size, the lower end can be defined with the *PAGE_MEMORY* command. Default memory organization is shown in Fig. 3–4.

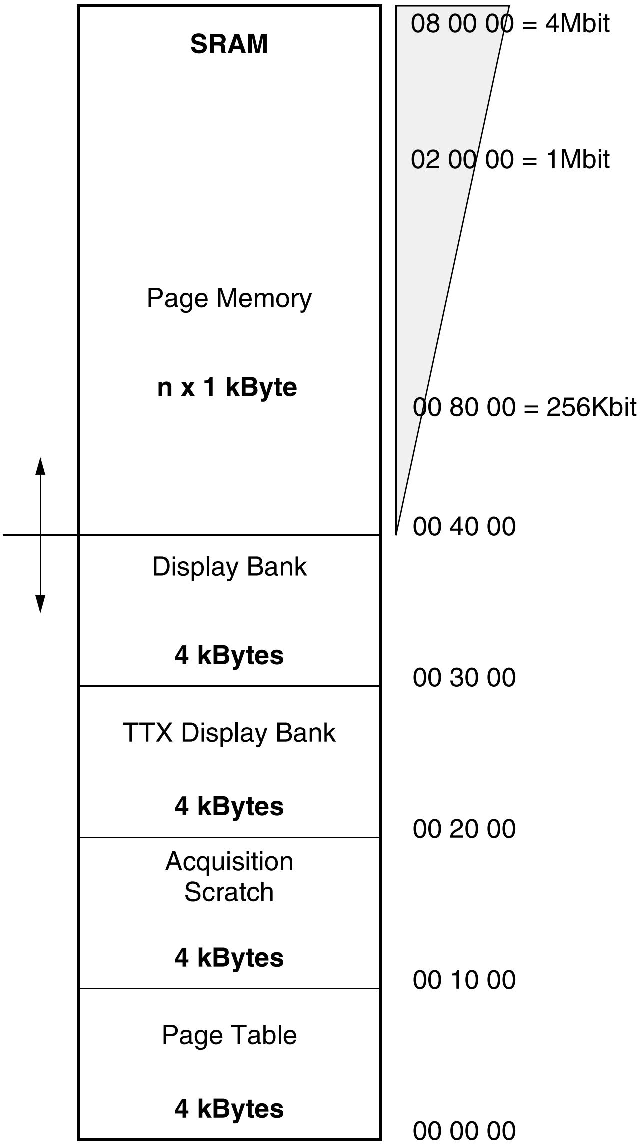


Fig. 3–4: Memory organization

Table 3–1: Memory Organisation

Memory Segment Address	**SRAM Size**			
	≥128k	**19k**	**16k**	**3k**
Display Bank	h’3000	h’4000	h’3000	h’0000
TTX Bank	h’2000	h’4000	h’3000	h’0000
Page Table	h’0000	h’0000	h’0000	no
Acquisition Scratch	h’1000	h’1000	h’1000	no
Page Memory	h’4000	h’1800	h’1800	no

The memory organization depends on available SRAM size. If external SRAM is not available, there is only one display bank for OSD and teletext and the page memory starts at a different location (see Table 3–1).

3.5.3. Page Table

The memory management is based on a fixed size page table, which has entries for every hexadecimal page number from 100 to 8FF. The page table starts with page 800 and contains a 2-Byte page pointer for every page.

The page table can be read with the command *READ_PAGE_INFO* sending the page number and reading the 2-Byte page pointer containing:

- SRAM pointer
- cycle flag
- memory flag
- subpage flag
- update flag
- protection flag

The SRAM pointer gives the location where the page is stored in memory. The page size is fixed to 1 kByte, only ghost rows are allocated dynamically.

The cycle flag will be set as soon as this page is detected in the transmission cycle even if it cannot be stored in memory. Only if the page is really stored in memory, the memory flag will be set. The subpage flag will be set for every page in cycle if the page subcode is different from 0000H or 3F7FH. The update flag is set every time a page is stored and will be reset only for the display page after updating the display memory. A page with protection flag set will never be removed from memory.

The memory manager uses page priorities to decide which pages should be stored or removed from memory. If no more memory is available, pages with lowest priority are removed automatically and the higher priority pages are stored at their place. By setting the page priority the programmer has control over the memory management.

The page table is fully controlled by the memory manager and should never be written by external software. To change the page table flags the command *CHANGE_PAGE_INFO* can be used.

Table 3–2: Page Table Format

<table>
<tr><th>Index</th><th colspan="6">2-Byte Page Pointer</th></tr>
<tr><td>000</td><td colspan="6">start magazine 8</td></tr>
<tr><td>001</td><td colspan="6"></td></tr>
<tr><td>...</td><td colspan="6"></td></tr>
<tr><td>100</td><td>Cycle Flag</td><td>Memory Flag</td><td>Subpage Flag</td><td>11-bit SRAM Pointer</td><td>Update Flag</td><td>Protect Flag</td></tr>
<tr><td>...</td><td colspan="6"></td></tr>
<tr><td>1F0</td><td colspan="6">hexadecimal pages (e.g. TOP)</td></tr>
<tr><td>...</td><td colspan="6"></td></tr>
<tr><td>7FE</td><td colspan="6"></td></tr>
<tr><td>7FF</td><td colspan="6">end magazine 7</td></tr>
</table>

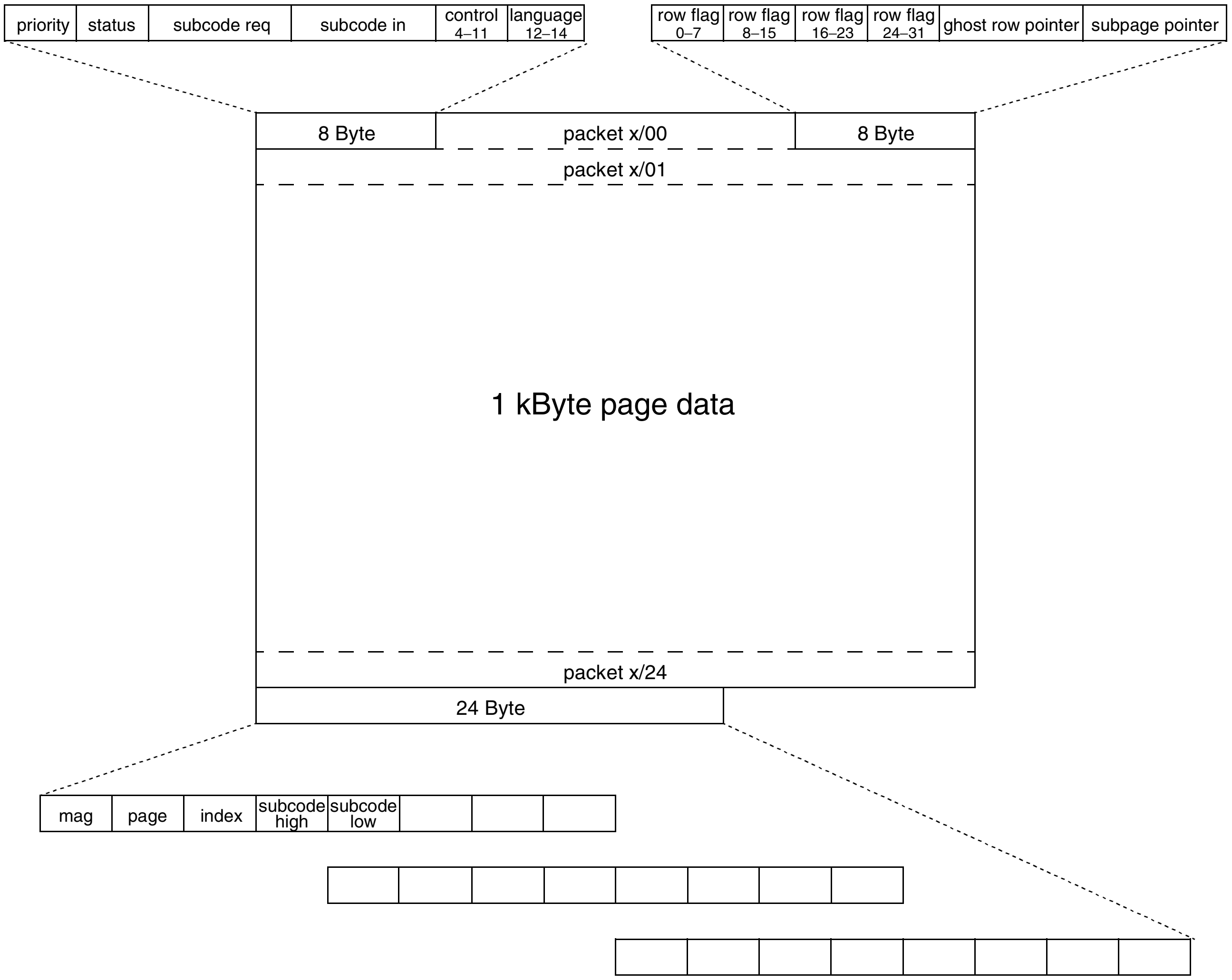


Fig. 3–5: Page format

3.5.4. Ghost Row Organization

Page-related ghost rows are stored in blocks of 128 Bytes. These ghost blocks are linked together using 2-Byte ghost row pointers. The first pointer can be found in the basic page, all following pointers are part of the block header. A zero pointer indicates the end of the chain.

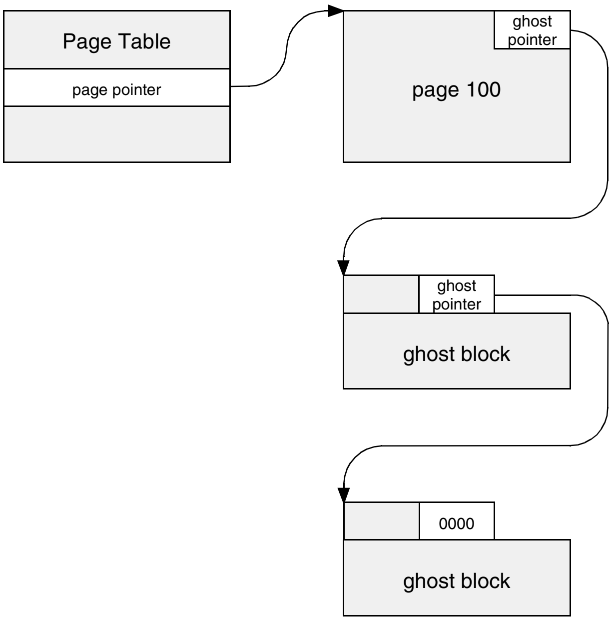


Fig. 3–6: Ghost row organization

Every ghost block contains 3 ghost rows which can be identified by 3 row identification bytes in the block header. The row identification contains designation code and row number. The row number is reduced to a 3-bit tag. All ghost rows in one block belong to the same page. If the memory manager removes a page from memory, the linked ghost blocks will also be removed.

Table 3–3: Ghost Row Identification

Row Number Tag	Row
000	empty
001	row 25
010	row 26
011	row 27
100	row 28
101	row 29
110	row 30
111	row 31

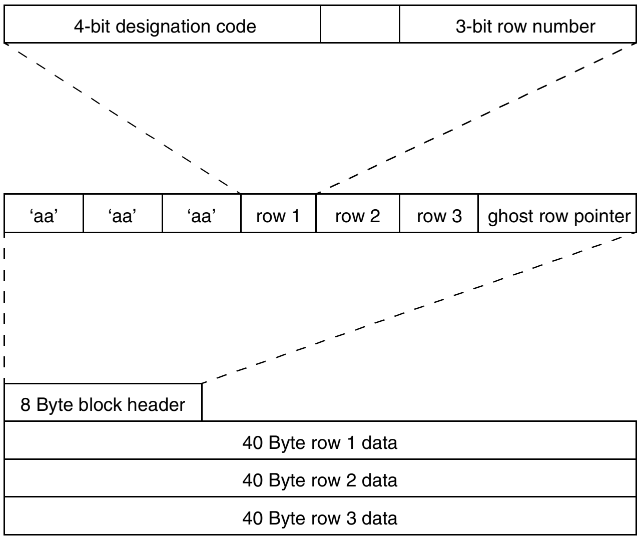


Fig. 3–7: Ghost block structure

3.5.5. Subpage Manager

Any page in cycle can have a number of subpages, identified by subcode. In normal mode the subpage manager will acquire only one subpage of every requested page. This subpage can be any if subcode FFFF is requested or it will be selected according to the requested subcode.

After a *PAGE_REQUEST* command with subcode F0xx, the subpage manager will acquire all subpages of the requested page. The subpages will be chained in the same order as they are transmitted, i.e. every new subcode will be added at the end of chain. The page table entry points to the subpage which was transmitted first after the page request. The *READ_PAGE_INFO* command will reply the page table pointer and the actual number of subpages in chain.

After a *PAGE_REQUEST* command with subcode F1xx, the subpage manager will acquire all subpages of the requested page but will allocate only a limited amount of memory to store these subpages. The parameter “page subcode low” will define the length (in number of subpages) of a ring buffer in page memory which will hold the recently received subpages. In this case, the *READ_PAGE_INFO* command will return an index pointing to the most recently updated subpage in chain, together with the subcode of this page.

The *DISPLAY_PAGE_REQUEST* command searches and displays a page according to the requested display subcode. The search starts from page table and continues through the subpage chain if there is any. A rolling header will be displayed if the requested subpage cannot be found in memory.

A requested display subcode FFFF (don't care subcode) will only search and display the first subpage in chain, thus there is no rolling subpage anymore. A *DISPLAY_PAGE_REQUEST* command with subcode F0xx (follow subcode) will search and display the last received subpage in chain, thus it is possible to request all subpages in background while still showing rolling subpages in display.

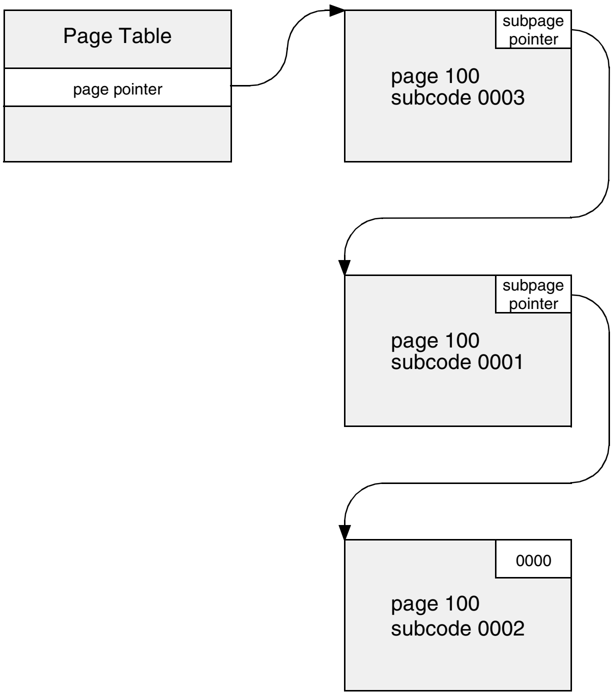


Fig. 3–8: Subpage organization

3.6. WST Display Controller

The display controller reads data from a display page buffer in the internal/external SRAM. The display page buffer is organized in rows which are separated into level 1 data such as character codes and spacing attributes and into level 2 data, such as character set extension and non-spacing attributes. To limit the memory amount for level 2 data, a slightly modified stack model is used, in which one pointer bit for every character location indicates the presence of additional parallel attributes. Fig. 3–9 shows the organization of the stack row buffer. In this stack model the number of non-spacing attributes per row is limited to 40, which agrees with the WST and CEPT specification.

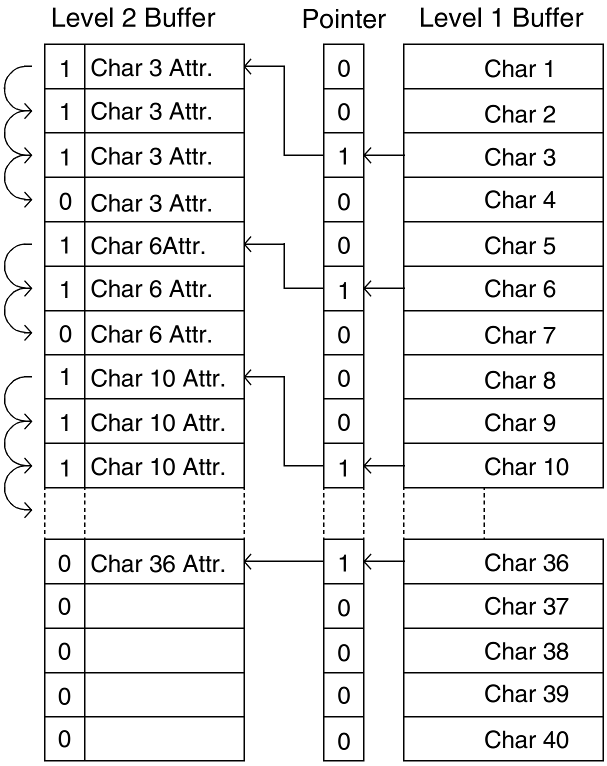


Fig. 3–9: Stack Row Buffer

The display controller includes two row buffers. The first row buffer holds a copy of a teletext row from the display page buffer. This decreases the data rate through the SRAM interface by a factor of 10 or 8, because new teletext row data is needed only after 10 lines in PAL or 8 lines in NTSC mode. The second row buffer stores all display attributes in parallel, to allow level 2 display without additional decoding.

To present a WST level 2 display, the teletext display controller has to evaluate the following attributes in parallel, that is for every character location:

- 10-bit character code
- 5-bit foreground color
- 5-bit background color
- 2-bit size
- 5-bit flash
- 1-bit invert
- 1-bit separated
- 1-bit conceal
- 1-bit underline
- 1-bit boxing/window

Additional attributes are defined to improve the display of CAPTION and OSD text:

- 1-bit italics
- 1-bit shadow
- 1-bit color mode

The display controller delivers 5-bit digital color information, a shadow signal for contrast reduction, and a fast blank signal. The color bus is used to address the color-lookup-table (CLUT) in the video processor. By this means, the full level 2 color spectrum can be displayed.

3.7. Display Memory

The TPU supports a variable number of display memories, each 4 kBytes large. One bank is used to store the display information of the selected teletext page. The bank location can be defined with the command DISPLAY_TTX_POINTER. Other banks can be used to store any kind of display data in level 1 or level 2 format. Switching between these banks is fast and can be programmed with the command DISPLAY_POINTER. Bank switching allows generation of OSD menus without affecting the teletext display.

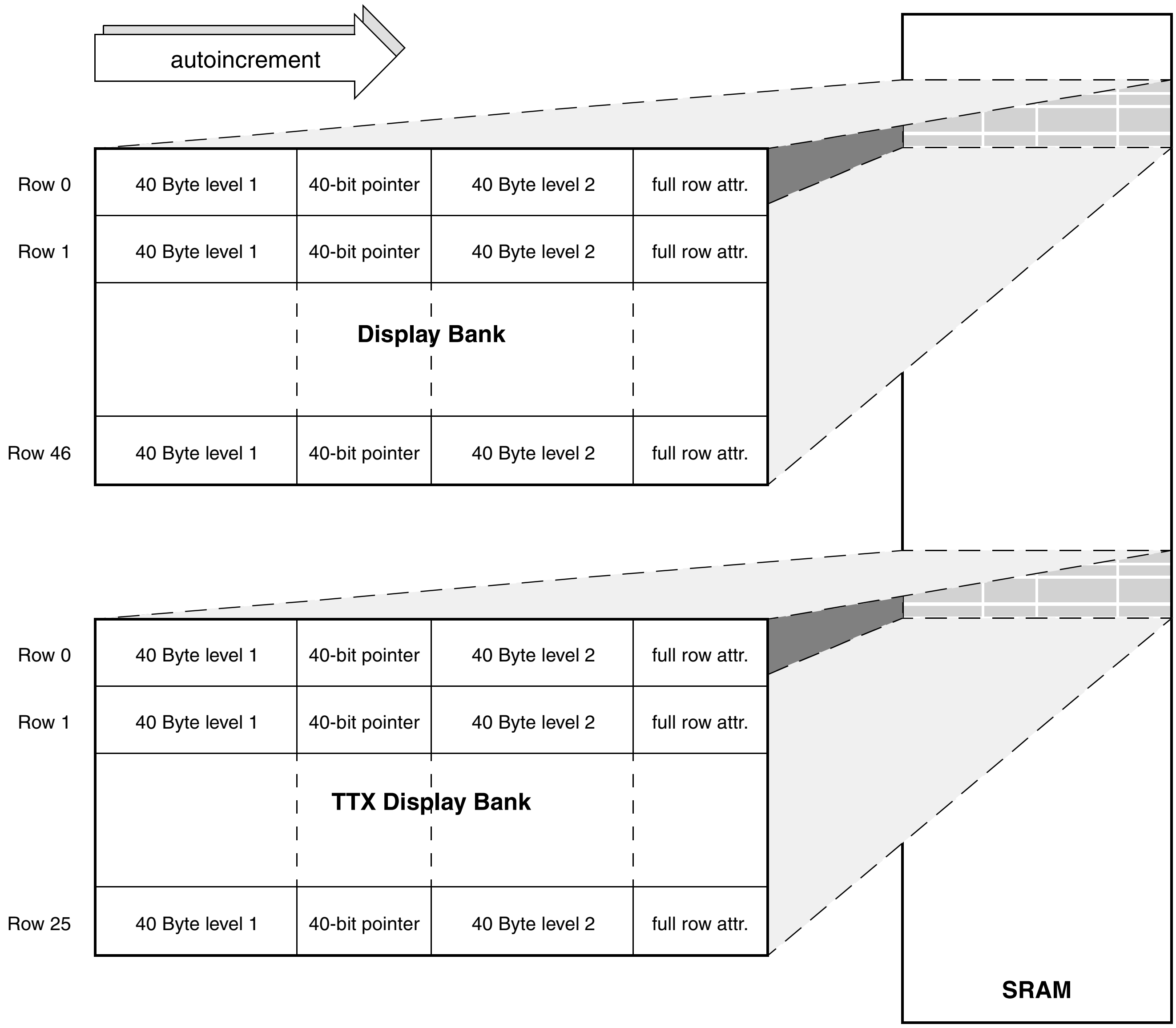


Fig. 3–10: Display memory organization (level 2)

Table 3–5: Full row attribute

+ 55H	R/W	Full Row Attribute
Bit	Reset	Function
7	-	1 = row is displayed blank 0 = row is displayed using row data
6	-	1 = row is displayed in double height 0 = row is displayed in normal height
5	-	1 = row is displayed in level 2 mode 0 = row is displayed in level 1 mode
4 to 0	-	5-bit value defining full row background color

Table 3–6: Level 1 spacing attributes

Code	Function	Action	Notes
00	Alpha Black		set alpha mode and foreground color of following alpha characters select character set 0
01	Alpha Red		
02	Alpha Green		
03	Alpha Yellow		
04	Alpha Blue		
05	Alpha Magenta		
06	Alpha Cyan		
07	Alpha White		
08	Flash Normal		
09	Flash Off	set at	
0A	Boxing Off	set at double	
0B	Boxing On	set at double	
0C	Size Normal	set at	
0D	Size Double Height		
0E	Size Double Width		
0F	Size Double		
10	Mosaic Black		set mosaic mode and foreground color of following mosaic characters select character set 1
11	Mosaic Red		
12	Mosaic Green		
13	Mosaic Yellow		
14	Mosaic Blue		
15	Mosaic Magenta		
16	Mosaic Cyan		
17	Mosaic White		
18	Conceal	set at	
19	Contiguous Mosaic	set at	
1A	Separated Mosaic	set at	
1B	ESC		
1C	Black Background	set at	
1D	New Background	set at	
1E	Hold Mosaic	set at	
1F	Release Mosaic		
Shaded attributes are default at start of each display row.			

Table 3–7: Level 2 parallel attributes

7	6	5	4	3	2	1	0	Function
P	0	0	Color					Foreground Color
P	0	1	Color					Background Color
P	1	0	Flash					Flash Mode
P	1	1	0	0	L	Set		Character Set
P	1	1	0	1	0	DH	DW	Size
P	1	1	0	1	1	0	U	Underline/Separated
P	1	1	0	1	1	1	I	Inverted
P	1	1	1	0	0	0	C	Conceal
P	1	1	1	0	0	1	W	Window/Boxing
P	1	1	1	0	1	0	S	Shadow
P	1	1	1	0	1	1	IT	Italic
P	1	1	1	1	0	0	CM	Color Mode

Table 3–8: Color look-up table

4	3	2	1	0	Display Color
0	0	0	0	0	Black
0	0	0	0	1	Red
0	0	0	1	0	Green
0	0	0	1	1	Yellow
0	0	1	0	0	Blue
0	0	1	0	1	Magenta
0	0	1	1	0	Cyan
0	0	1	1	1	White
0	1	0	0	0	Transparent
0	1	0	0	1	Reduced Red
0	1	0	1	0	Reduced Green
0	1	0	1	1	Reduced Yellow
0	1	1	0	0	Reduced Blue
0	1	1	0	1	Reduced Magenta
0	1	1	1	0	Reduced Cyan
0	1	1	1	1	Reduced White
1	x	x	x	x	Programmable

Table 3–4: Flash modes

4	3	2	1	0	Function
0	0	0	0	0	Off
0	0	0	0	1	Normal
0	0	1	0	1	Normal Fast Phase 1
0	1	0	0	1	Normal Fast Phase 2
0	1	1	0	1	Normal Fast Phase 3
0	0	0	1	0	Inverted
0	0	1	1	0	Inverted Fast Phase 1
0	1	0	1	0	Inverted Fast Phase 2
0	1	1	1	0	Inverted Fast Phase 3
0	0	0	1	1	Color Table
0	0	1	1	1	Color Table Phase 1
0	1	0	1	1	Color Table Phase 2
0	1	1	1	1	Color Table Phase 3
1	0	0	x	x	Incremental
1	0	1	x	x	Decremental

3.8. Character Generator

Characters are addressed using a 10-bit character code. The 2 MSBs of the character code define 1 of 4 character sets. Character set selection is done using level 2 parallel attributes (see Table 3–7 on page 56). Each character set contains 224 characters. The first 32 characters in each character set are reserved for control codes (see Table 3–6 on page 56). On a single screen, 896 different characters can be displayed.

Characters can be displayed in several pixel resolutions provided that the according font is available. The character generator supports horizontal resolution of 8 or 10 pixel/char and vertical resolution of 8, 10, or 13 lines/char. Characters can be combined without separating borders to create more complex character definitions (e.g. kanji or icons).

Table 3–9: Character resolutions

matrix (h x v)	**char/screen (PAL)**	**char/screen (NTSC)**	**osd width**	**# char in 12k font**	**# char in 20k font**
single character					
8 x 8	40 x 32	40 x 28	32µs	1600	2560
10 x 8	40 x 32	40 x 28	40µs	1280	2048
8 x 10	40 x 26	40 x 22	32µs	1280	2048
10 x 10	40 x 26	40 x 22	40µs	1024	1638
8 x 13	40 x 20	40 x 17	32µs	800	1280
10 x 13	40 x 20	40 x 17	40µs	640	1024
combined character (2 x 2)					
16 x 16	20 x 16	20 x 14	32µs	400	640
20 x 16	20 x 16	20 x 14	40µs	320	512
16 x 20	20 x 13	20 x 11	32µs	320	512
20 x 20	20 x 13	20 x 11	40µs	256	409
16 x 26	20 x 10	20 x 8.5	32µs	200	320
20 x 26	20 x 10	20 x 8.5	40µs	160	256
combined character (2 x 1)					
16 x 10	20 x 26	20 x 22	32µs	640	1024
16 x 13	20 x 20	20 x 17	32µs	400	640
20 x 13	20 x 20	20 x 17	40µs	320	512
combined character (1x 2)					
10 x 16	40 x 16	40 x 14	40µs	640	1024

The pixel clock can be either 10.125 MHz or 20.25 MHz. To get 10-bit pixel information from the character font, two memory cycles are needed. The character font is part of the mask-programmable ROM, but supplied with its own bus structure (see Fig. 3–2 on page 49). By this means the data transfer between character ROM and teletext display controller does not stop the CPU.

Both bus structures are connected via a memory interface which allows cross-connections using DMA or wait cycles. If the character font size exceeds 12 kBytes, part of the character font can be shifted into the program ROM which causes DMA cycles. Therefore only less frequently used characters should be placed into the program ROM. Vice versa seldom used CPU code can be put into the character ROM.

The WST specification defines a number of 7-bit code tables, which are filled with 96 characters only (the MSB is used for parity check). In the G0 code table some characters have several language dependent variations. Additionally characters from the G0 code table can be combined with diacritical marks from the G2 code table (row 26). Furthermore different code tables are defined for languages like cyrillic, greek or arabic. Thus it is not possible to simply transform the code tables into a continuous character font ROM without getting unused ROM space and multiple defined character fonts.

This problem is solved by implementing a character code mapping (see Fig. 3–11 on page 58). The 5 MSBs of each character code are mapped into another 5-bit code which is then used to address the character font ROM. By this means the whole character font is subdivided into 32 blocks of 32 characters which can freely be distributed over the 4 character sets.

The character code mapping is implemented as RAM and can be programmed by software. After reset the TPU initializes the mapping RAM for standard WST latin code tables.

The TV controller can select predefined mappings for latin, cyrillic and arabic teletext via the command DISPLAY_MODE (see Table 3–16 on page 70). The same command allows selection of a user defined mapping which has to programmed in advance using command USER_MAPPING.

3.8.1. Character Code Mapping

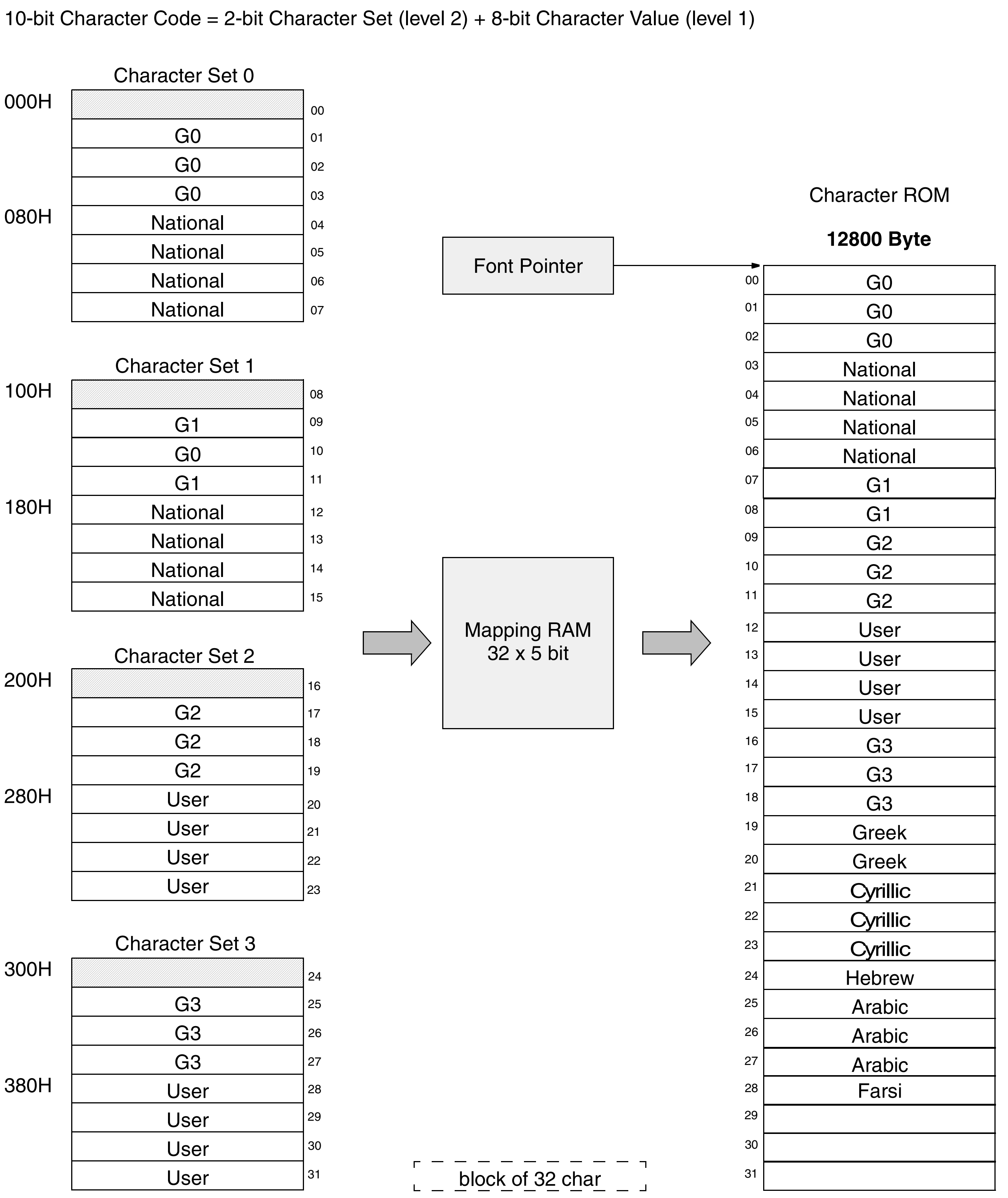


Fig. 3–11: Character code mapping

3.8.2. Character Font ROM

The character font ROM is mask-programmable. Design of customer specific characters (user font) is supported by a Windows™ based PC tool named MOFA (Micronas OSD and Font Assembler). In combination with the VCT 38xxA emulator board it is possible to download character fonts and verify them on the TV screen.

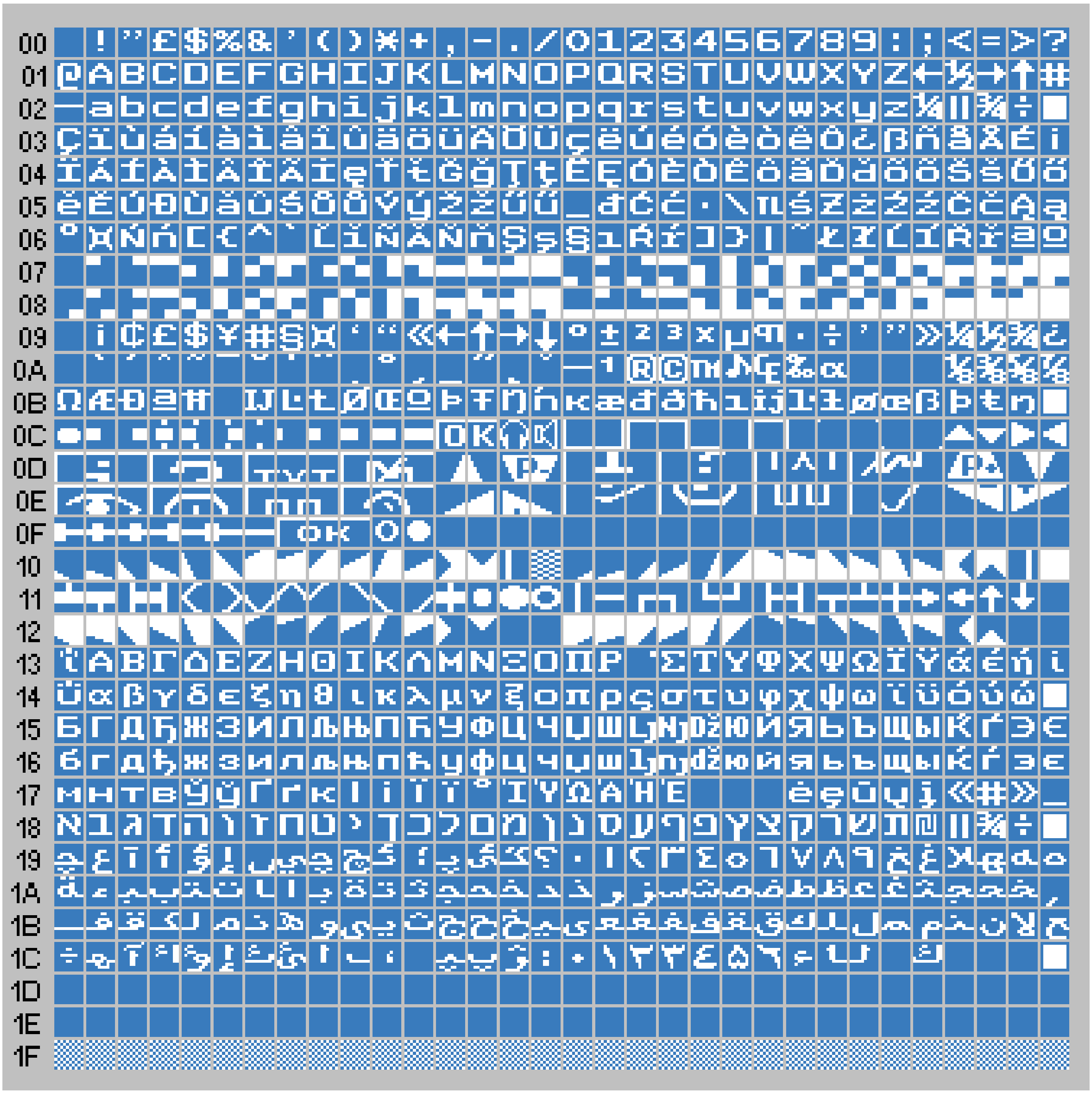


Fig. 3–12: Character font ROM

3.8.3. Latin Font Mapping

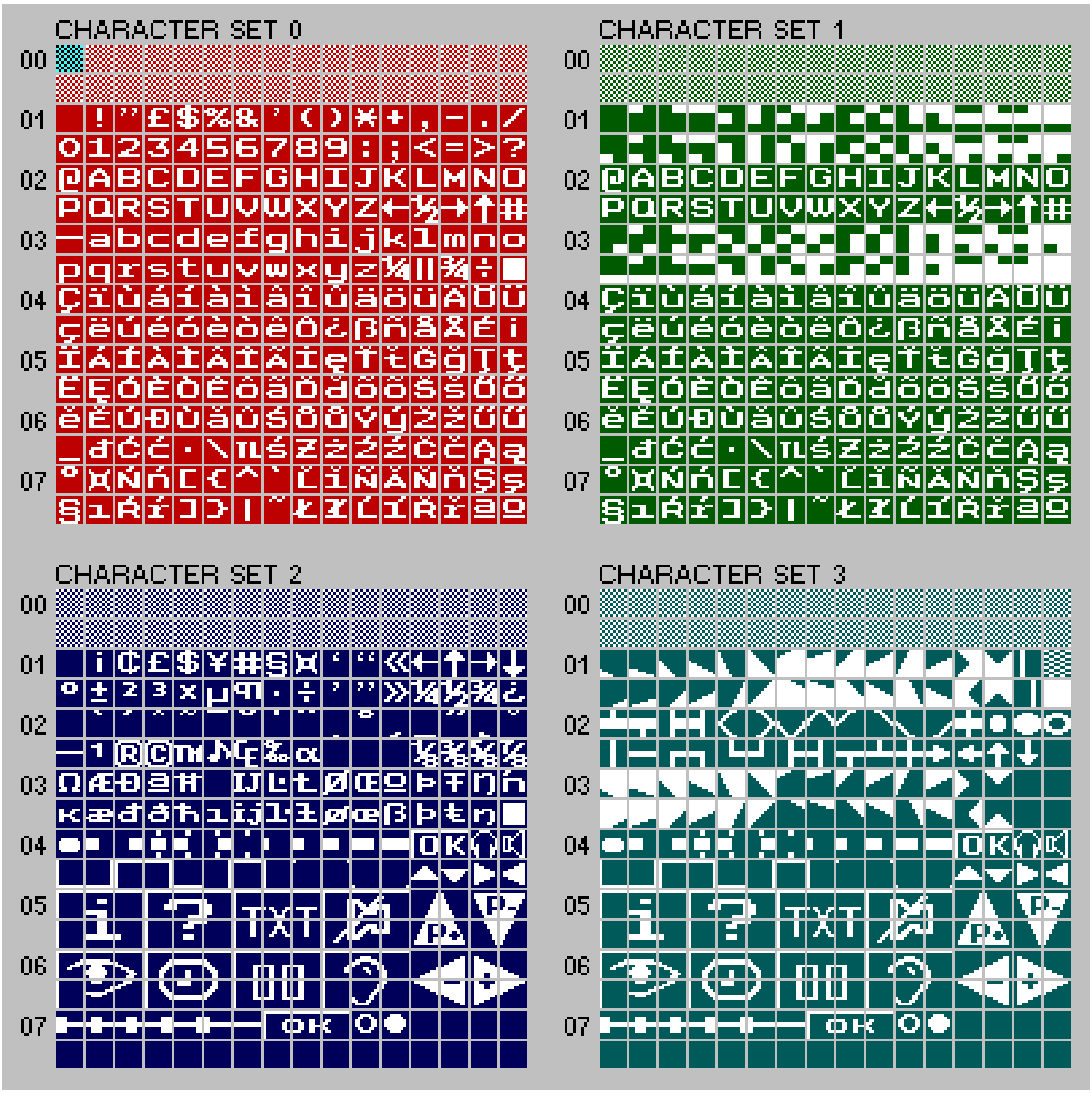


Fig. 3–13: Latin font mapping

3.8.4. Cyrillic Font Mapping

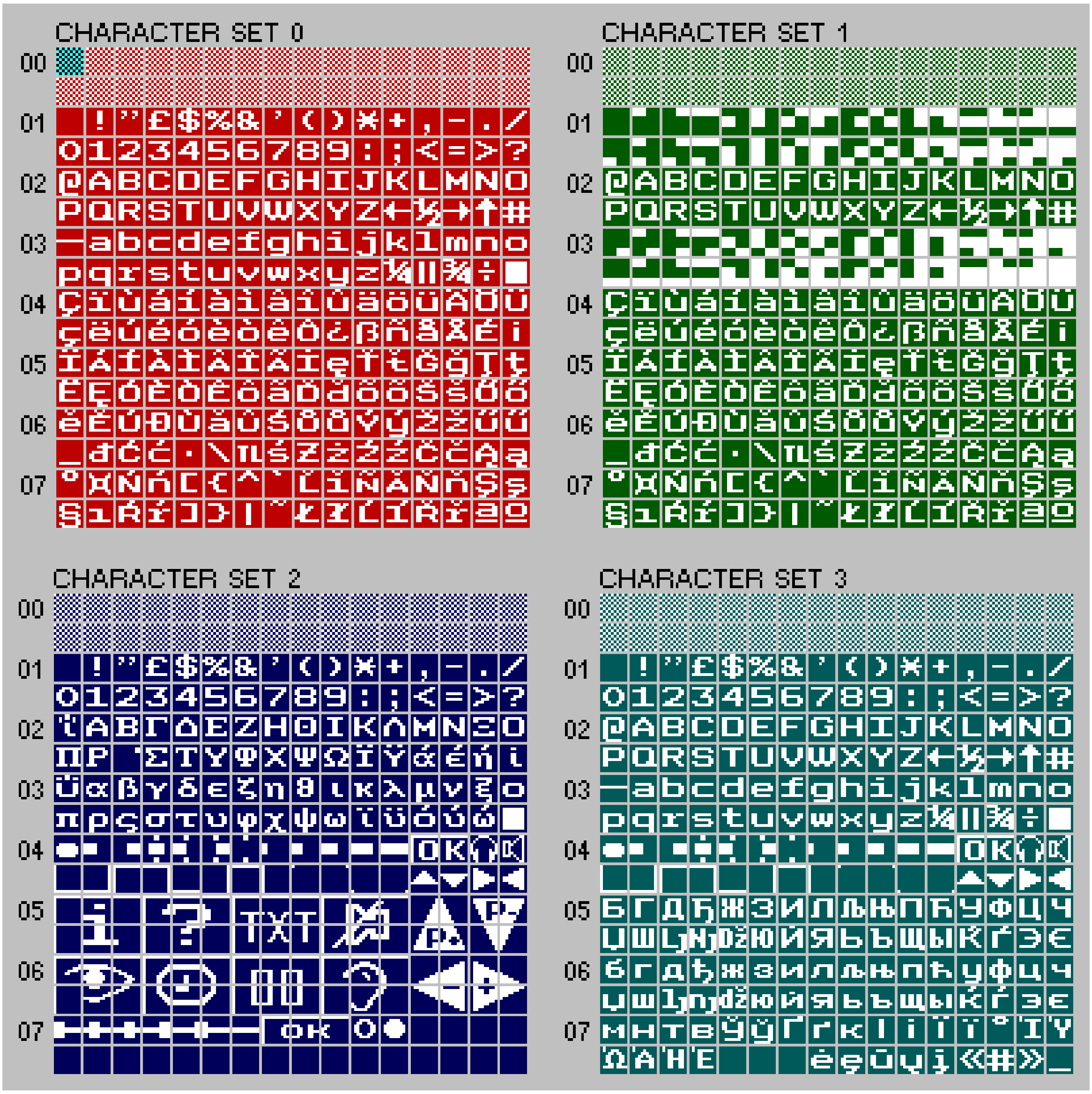


Fig. 3–14: Cyrillic font mapping

3.8.5. Arabic Font Mapping

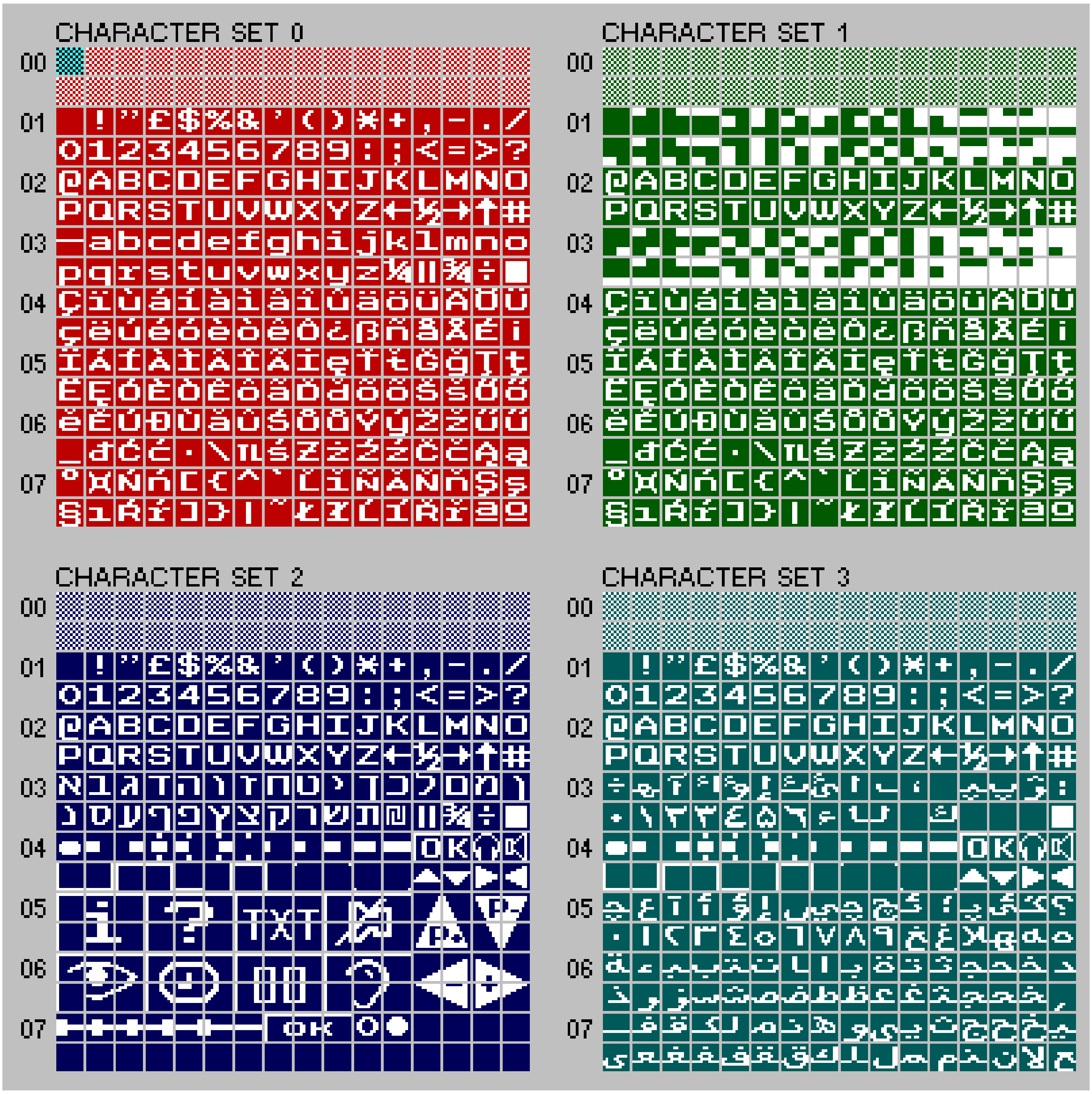


Fig. 3–15: Arabic font mapping

3.8.6. Character Font Structure

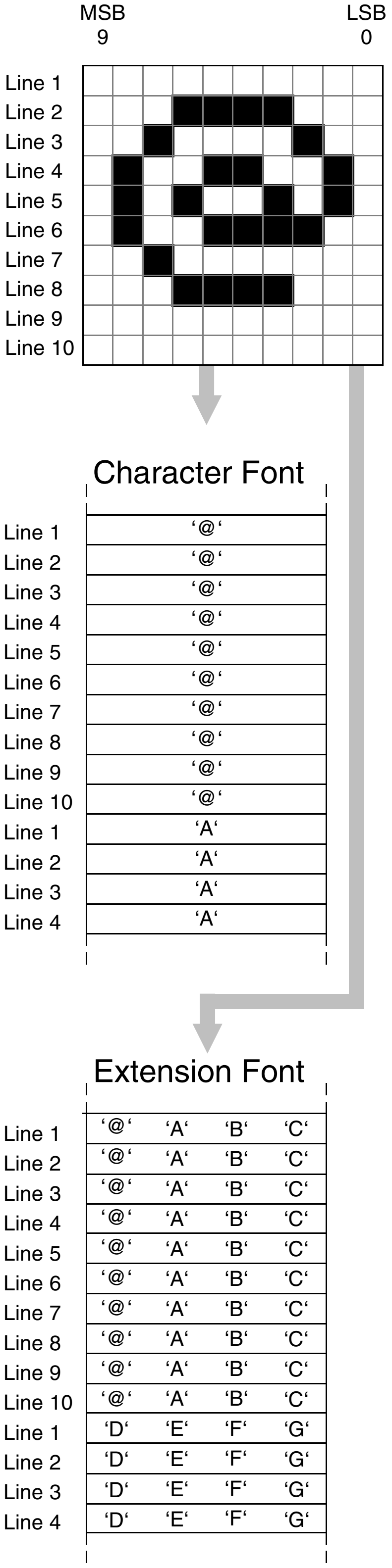


Fig. 3–16: Font Structure 10 x 10

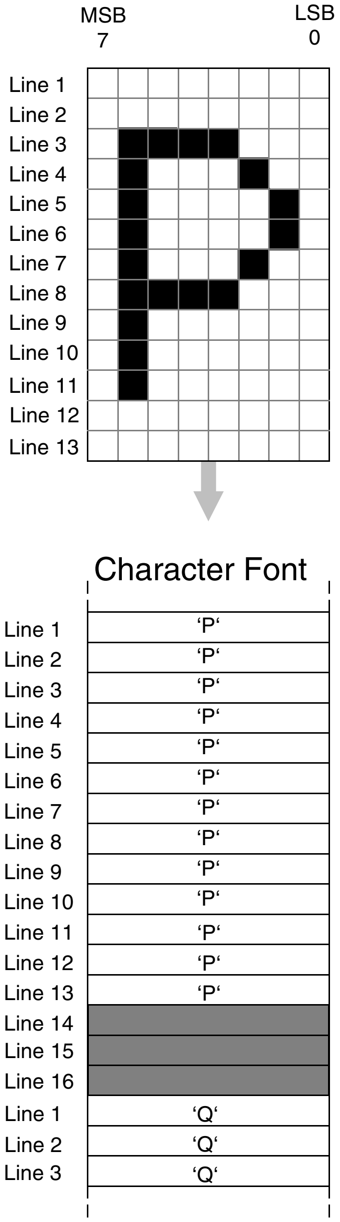


Fig. 3–17: Font structure 8 x 13

3.9. National Character Mapping

Table 3–10: Character set options

Option Bits	Character Set					
C14,C13,C12	6	38	40	55	70	128
000	English	Polish	English (US)	English	English (US)	programmable
001	French	French	French	French	Slovakian	programmable
010	Swedish	Swedish	Swedish	Swedish	Hungarian	programmable
011	Czech	Czech	Czech	Turkish	Serbian	programmable
100	German	German	German	German	Albanian	programmable
101	Spanish	Serbian	Spanish	Spanish	Polish	programmable
110	Italian	Italian	Italian	Italian	Turkish	programmable
111	Estonian	Estonian	Estonian	Estonian	Rumanian	programmable

Table 3–11: Language codes

Code	Language
0	English
1	French
2	Swedish, Finnish
3	Czech
4	German
5	Spanish
6	Italian
7	Estonian, Finnish
8	English (US)
9	Slovakian
10	Hungarian
11	Serbian, Croatian, Slovene
12	Albanian
13	Polish
14	Turkish
15	Rumanian
16	Cyrillic (Russian, Bulgarian)
17	Greek
18	Cyrillic (Serbian, Montenegro)
19	YU Latin
20	Arabic
21	Hebrew
22	Farsi
23	Lettish, Lithuanian
24	Cyrillic (Ukrainian)
25–255	not defined

Table 3–12: National option mapping

Language	**G0/G1 Table Position**												
	2/3	**2/4**	**4/0**	**5/11**	**5/12**	**5/13**	**5/14**	**5/15**	**6/0**	**7/11**	**7/12**	**7/13**	**7/14**
Albanian	5/15	2/4	13/12	13/2	12/12	12/3	11/12	9/1	13/13	13/3	12/13	13/1	11/13
Czech	5/15	12/9	13/13	10/11	12/13	12/11	8/4	15/13	9/3	8/3	12/0	9/2	11/13
English	2/3	2/4	4/0	5/11	5/12	5/13	5/14	5/15	6/0	7/11	7/12	7/13	7/14
English (US)	5/15	2/4	4/0	14/4	13/5	15/4	14/6	13/0	14/7	14/5	15/6	15/5	15/7
Estonian, Finnish	5/15	11/11	11/12	8/13	8/14	12/12	8/15	11/10	11/13	8/10	8/11	12/13	8/12
French	9/3	8/1	8/5	9/1	9/7	8/2	8/8	5/15	9/5	8/7	9/8	8/9	9/0
German	5/15	2/4	15/0	8/13	8/14	8/15	14/6	13/0	14/0	8/10	8/11	8/12	9/10
Hungarian	5/15	9/2	9/14	8/4	8/14	10/1	12/15	11/15	9/3	9/4	8/11	8/3	8/12
Italian	2/3	2/4	9/3	14/0	9/0	5/13	5/14	5/15	8/2	8/5	9/6	9/5	8/6
Polish	5/15	14/3	13/15	13/8	12/7	15/8	13/3	9/4	10/9	13/9	13/7	15/9	13/11
Rumanian	5/15	14/1	10/14	10/5	14/14	14/11	10/6	15/1	10/15	8/7	14/15	12/5	8/8
Serbian, Croatian	5/15	2/4	13/12	13/2	12/12	12/3	11/12	13/0	13/13	13/3	12/13	13/1	11/13
Slovakian	5/15	12/9	13/13	10/11	12/13	12/11	8/4	15/13	9/3	8/3	12/0	9/2	11/13
Spanish	9/0	2/4	9/15	8/3	9/3	8/4	9/4	9/2	9/9	8/12	9/11	9/5	8/5
Swedish, Finnish	5/15	14/1	9/14	8/13	8/14	9/13	8/15	13/0	9/3	8/10	8/11	9/12	8/12
Turkish	13/6	10/13	10/8	14/14	8/14	8/0	8/15	10/12	15/1	14/15	8/11	9/0	8/12
YU Latin	5/15	2/4	13/12	13/2	12/12	12/3	11/12	13/0	13/13	13/3	12/13	13/1	11/13

3.10.Four-Color Mode

In “Four-Color Mode” the color depth of single or multiple characters can be increased to 4 colors (e.g. to display icons or 3-D effects). A special font organization is required because two consecutive characters will be combined. The number of 4-colored characters is only limited by font size.

The “Four-Color Mode” is controlled via the level 2 parallel attribute “Color Mode”. Setting the bit CM to 1 activates the “Four-Color Mode” until the end of row or until the bit CM is set to 0 again. At the start of each display row the “Four-Color Mode” is disabled.

A character with active “Four-Color Mode” attribute will be combined with its font neighbor to define a 2 bit/pixel character matrix. The 2 additional colors are derived from the active foreground and background colors by inverting bit 3 of the color code. Using the programmable part of the CLUT it is possible to display characters in 4 out of 4096 colors.

If the “Four-Color Mode” attribute is set for a character with even character code n, this character is combined with its font neighbor addressed by code n + 1. If the “Four-Color Mode” attribute is set for a character with odd character code, this character is combined with itself.

The neighbor character does not change the definition of foreground and background pixel which is used to control flash and mix mode.

Table 3–13: Color Allocation

Pixel Definition		Color Allocation
Character n	Character n+1	
0	0	background
1	0	foreground
0	1	background .xor. 8
1	1	foreground .xor. 8

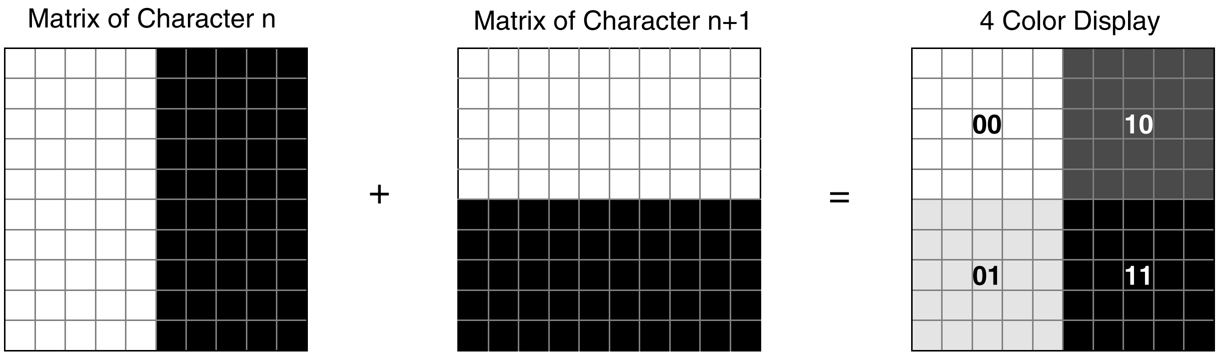


Fig. 3–18: Four-color mode

3.11. OSD Layer

Apart from the WST layer, there is an additional OSD layer on chip. The OSD layer accesses the CPU memory via DMA to read text, display attributes, and character font information. The color outputs of the OSD layer can have higher priority than the WST layer outputs. Thus, it is possible to overlay the teletext display with an additional layer for user guidance (see Fig. 3–19).

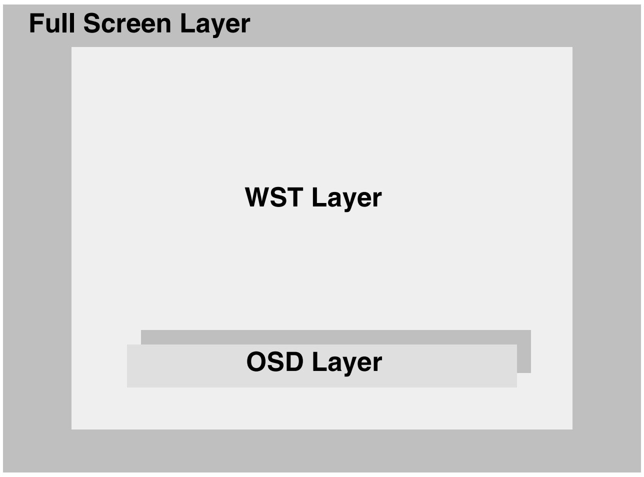


Fig. 3–19: Display layer

The OSD layer reads text strings addressed by a programmable text pointer. Codes smaller than 80h will address the character font, codes greater or equal 80h are interpreted as control codes to change color or character set (see Table 3–14). After reading a control code the OSD layer will do an additional read to get the next character code.

Code	Function	Notes
01	Underline On	only for 13 scanlines/character
02	Underline Off	
03	Flash On	
04	Flash Off	
05	Italics On	
06	Italics Off	
07	Transparent	layer becomes transparent
08	Shadow	layer becomes transparent and contrast is reduced to 66%
0C	END	end of layer
0D	CR	end of text line
0E – 7F	ASCII Character	using font 1 or font 2
80 – FF	Control Code	Only one control code per character is allowed. Depending on OSD Mode, the control code defines either color or character set.
	Color	bit 0 = foreground color blue bit 1 = foreground color green bit 2 = foreground color red bit 3 = background color blue bit 4 = background color green bit 5 = background color red bit 6 = replace white by transparent bit 7 = 1
	Character Set	bit 0 = bit 7 of character code bit 1 = bit 8 of character code bit 2 = bit 9 of character code bit 3 = bit 4 = bit 5 = bit 6 = latching shift to character set bit 7 = 1
Shaded attributes are default at start of each text line.		

3.12.Command Language

The TPU supports a command language, allowing the TV controller to start complex processing inside the TPU with simple commands. The TV controller is not burdened with time consuming tasks like page searching or data shuffling.

Table 3–15 lists all available commands. For a more detailed description of the command language see Table 3–16.

The application software has to send commands to the TPU via I^2C bus using the command subaddresse SUB4 (see Section 3.14.1.3. on page 83).

Table 3–15: Command language cross reference

Code Dec.	Code Hex.	Command Name	No. Write Parameter	No. Read Parameter	Status Register
0	00	Dummy	0	0	x000 0000
1	01	Reset	0	0	x000 0000
2	02	Escape	0	0	x000 0000
3	03	Version	0	2	x000 0000
4	04	Test	0	0	x000 0000
5	05	Test	0	0	x000 0000
6	06	DRAM Mode	3	0	x000 0000
7	07	Acquisition Mode	5	2	x000 0000
8	08	Display Mode	4	0	x000 0000
9	09	Display TTX Pointer	2	0	x000 0000
10	0a	Display Pointer	3	0	x000 0000
11	0b	Display Clear	2	0	x000 0000
12	0c	Page Request	8	3	x0x0 0000
13	0d	Display Time Pointer	2	0	x000 0000
14	0e	Read DRAM Size	0	3	x000 0000
15	0f	Read VPS	0	15	x0x0 0000
16	10	Read Quality	0	4	x000 0000
17	11	Read Display Mode	0	4	x000 0000
18	12	Read Reset Source	0	1	x000 0000
19	13	Read Rolling Header	0	24	x000 0000
20	14	Read Page Info	2	7	x000 0000
21	15	Read Page Row	5	40	x0x0 0000
22	16	Change Page Info	3	0	x000 0000
23	17	Search MPET	0	1 + (n*4)	x0x0 0000
24	18	Read Display Page	0	4	x000 0000
25	19	Page Memory	2	0	x000 0000
26	1a	Display Page Request	5	0	x000 0000
27	1b	Page Table Reset	0	0	x000 0000
28	1c	Search Next Page	3	6	x0x0 0000
29	1d	Read Page Cycle	0	9	x000 0000
30	1e	Read TOP Code	2	2	x000 0000
31	1f	Read Rolling Time	0	8	x000 0000
32	20	Copy Page Row	8	0	x0x0 0000
33	21	Copy Data	7	0	x000 0000
34	22	Search Next TOP Code	3	4	x0x0 0000
35	23	Read Ghost Row	6	40	x0x0 0000

Table 3–15: Command language cross reference

Code Dec.	Code Hex.	Command Name	No. Write Parameter	No. Read Parameter	Status Register
36	24	Read 8/30 Row	1	40	x0x0 0000
37	25	Read Priority	0	5	x000 0000
38	26	Page Priority	2	0	x000 0000
39	27	Search AIT	0	1 + (n*4)	x0x0 0000
40	28	Read TOP Status	0	2	x000 0000
41	29	Search AIT Title	2	17	x0x0 0000
42	2a	Reset Ghost Row Status	0	0	x000 0000
43	2b	Search MPT	0	1 + (n*4)	x0x0 0000
44	2c	Copy AIT Title	5	17	x0x0 0000
45	2d	Search Direct Choice	1	1 + (n*2)	x0x0 0000
46	2e	Read Hamming	1	1	x000 0000
47	2f	Read Hamming 2	3	3	x000 0000
48	30	Display Column	3+length	0	x000 0000
49	31	Display Fill	4	0	x000 0000
50	32	Read BTTL	0	9	x0x0 0000
51	33	Read Next Page	2	2	x000 0000
52	34	Change BTT magazine	1	0	x000 0000
53	35	Read WSS	0	15	x0x0 0000
54	36	Read CAPTION 1	0	7	x0x0 0000
55	37	Read CAPTION 2	0	7	x0x0 0000
56	38	OSD Font Pointer	5	0	x000 0000
57	39	Display Read Column	3	length	x000 0000
58	3a	User Character Set	8	0	x000 0000
59	3b	User ESC Character Set	8	0	x000 0000
60	3c	Full Row Attribute	3	0	x000 0000
61	3d	User Mapping	32	0	x000 0000

Note:
If not otherwise designated, all parameters in the following table are specified as single bytes. As write parameter magazine numbers 8 and 0 have the same meaning, as read parameter the magazine number is a true 4-bit number (e.g. magazine 8=00001000). For write parameters the values in parentheses indicate default values after reset (in hex notation). For compatibility reasons every undefined bit in a write parameter should be set to ‘0’. Undefined bits in a read parameter should be treated as “don’t care”.

Table 3–16: Command language

Code	Function	Write Parameter	Read Parameter	Notes
Operational & Test Commands				
00	Dummy			no action
01	Reset			software reset of 65C02
02	Escape			escape to other codes
03	Version		CPU pointer high CPU pointer low	show version in OSD layer CPU pointer to text in ROM
04	Test			reserved for testing
05	Test			reserved for testing
06	DRAM Mode	dram mode (06) flash inc (05) control enable (FF)		dram mode = I/O page register 028EH flash freq = flash inc / (256 * 0.00324) control enable: bit0 = C4 erase page bit1 = C5 news flash bit2 = C6 subtitle bit3 = C7 suppress header bit4 = C8 update indicator bit5 = C9 interrupted sequence bit6 = $\overline{C10}$ inhibit display bit7 = $\overline{C11}$ magazine parallel
07	Acquisition Mode	acquisition mode (00) init subcode high (FF) init subcode low (FF) gain max (1F) filter max (1F)	gain filter	acquisition mode: bit0 = no slicer adaption bit1 = no bit error in framing code bit2 = limit slicer adaption init subcode: automatic subcode request after page table reset gain max: only used if bit2 = 1 filter max: only used if bit2 = 1
Memory Management Commands				

14	Read DRAM Size		dram size high
			dram size low
			dram mode
				dram size:
				`000CH =   3kByte SRAM`
				`004CH =  19kByte SRAM`
				`0200H = 128kByte SRAM`
				`0240H = 144kByte SRAM`
				`0400H = 256kByte SRAM`
				`0800H = 512kByte SRAM`
				dram mode:
				see I/O page register 028EH

25	Page Memory	dram bank (00) dram high (40)		start of page memory execute page table reset
27	Page Table Reset			reset page table reset ghost row status reset data service status reset cycle count reset memory count reset ghost count reset priorities clear rolling header clear VPS data clear WSS data

Table 3–16: Command language, continued

Code	Function	Write Parameter	Read Parameter	Notes
42	Reset Ghost Row Status			ghost row status: bit0 = row 24 in cycle bit1 = row 25 in cycle bit2 = row 26 in cycle bit3 = row 27 in cycle bit4 = row 28 in cycle bit5 = row 29 in cycle bit6 = row 30 in cycle bit7 = row 31 in cycle
29	Read Page Cycle		ghost row status 2 Byte cycle count 2 Byte memory count 2 Byte ghost count data service status memory status	= number of pages in cycle = number of pages in memory = number of ghost blocks in memory data service status: bit0 = 8/30 format 1 updated bit1 = 8/30 format 2 updated bit2 = VPS updated bit3 = WSS updated bit4 = CAPTION 1st field updated bit5 = CAPTION 2nd field updated memory status: bit0 = memory full
38	Page Priority	enable (00) border (FF)		enable: bit0 = enable priority manager border: min/max border for page priorities
37	Read Priority		highest priority lowest priority border priority magazine number page number	= max priority in page memory = min priority in page memory = min/max border for page priorities = page with lowest priority
Page Related Commands				
12	Page Request	magazine number page number page subcode high page subcode low priority quantity start magazine number start page number	number of open requests removed magazine number removed page number	remove pages from memory beginning at start page if page priority is disabled, ignores start page if page priority is enabled magazine number: bit0–3 = magazine number bit4 = not used bit5 = hex request bit6 = backward request bit7 = forced request = ignore cycle flag
20	Read Page Info	magazine number page number	page pointer high page pointer low subpage count ghost row count ring buffer index page subcode high page subcode low	= pointer from page table = number of subpages in chain = number of ghost rows in chain if page request with subcode F1xx
22	Change Page Info	magazine number page number page table flags		page table flags: bit0 = protection bit1 = update bit2 = not used bit3 = not used bit4 = not used bit5 = subpage bit6 = memory bit7 = cycle

Table 3–16: Command language, continued

Code	Function	Write Parameter	Read Parameter	Notes
28	Search Next Page	magazine number page number search code	magazine number page number page pointer high page pointer low subpage count ghost row count	search in page table for cycle flag magazine number: bit0–3 = magazine number bit4 = take search code bit5 = hex search bit6 = backward search bit7 = include start page search code: bit0 = search protection flag bit1 = search update flag bit2–4 = not used bit5 = search subpage flag bit6 = search memory flag bit7 = search cycle flag
51	Read Next Page	magazine number page number	magazine number page number	calculate next page number magazine number: bit0–3 = magazine number bit4 = not used bit5 = hex calculation bit6 = backward calculation bit7 = not used
21	Read Page Row	magazine number page number subpage number high subpage number low row number	40 Byte row data	row 0 – 24
32	Copy Page Row	magazine number page number subpage number high subpage number low row number destination dram bank destination dram high destination dram low		copy 40Byte text row from page memory into DRAM
35	Read Ghost Row	magazine number page number subpage number high subpage number low row number designation code	40 Byte row data	row 25 – 28
TOP Commands				
40	Read TOP Status		TOP status 1 TOP status 2	TOP status 1: bit0 = not used bit1 = MPT link in PLT bit2 = MPET link in PLT bit3 = AIT link in PLT bit4 = BTT in memory bit5 = MPT in memory bit6 = MPET in memory bit7 = AIT in memory TOP status 2: bit0–5 = not used bit6 = all MPET in memory bit7 = all AIT in memory
30	Read TOP Code	magazine number page number	BTT code MPT code	code: bit0–3 = data bit6 = hamming error
50	Read BTTL		BTTL error 8 Byte BTTL data	BTTL error: bit6 = hamming error in BTTL BTTL data: bit0–3 = data bit6 = hamming error
52	Change BTT magazine	magazine number (01)		all TOP commands then refer to this magazine
43	Search MPT		number of MPTs magazine number page number subpage number high subpage number low ...	search in PLT

Table 3–16: Command language, continued

Code	Function	Write Parameter	Read Parameter	Notes
23	Search MPET		number of MPETs magazine number page number subpage number high subpage number low ...	search in PLT
39	Search AIT		number of AITs magazine number page number subpage number high subpage number low ...	search in PLT
41	Search AIT Title	magazine number page number	5 Byte data 12 Byte title	search in AIT magazine number: bit0–3 = magazine number (0#8) bit4–6 = not used bit7 = ignore title language data: bit0–3 = data bit6 = hamming error
44	Copy AIT Title	magazine number page number destination dram bank destination dram high destination dram low	5 Byte data 12 Byte title	search in AIT and copy title into dram magazine number: bit0–3 = magazine number (0#8) bit4–6 = not used bit7 = ignore title language data: bit0–3 = data bit6 = hamming error
34	Search Next TOP Code	magazine number page number code condition	magazine number page number code code flag	search in BTT magazine number: bit0–3 = magazine number bit4–5 = not used bit6 = backward search bit7 = include start page code condition: low nibble = BTT code high nibble = search condition 0 = BTT code in low nibble 1 = BTT code # 0 2 = block page 3 = group page 4 = normal page 5 = subtitle page 6 = TV page 7 = block/TV page 8 = group/block/TV page 9 = subpage a = block/TV subpage b = group/block/TV subpage c = title page d = future page e = future page f = future page code: bit0–3 = BTT code bit6 = hamming error code flag: bit0 = subtitle page found bit1 = TV page found bit2 = block page found bit3 = group page found bit4 = normal page found bit5 = future page found bit6 = title page found bit7 = subpage found
45	Search Direct Choice	direct choice code	number of AIT entries magazine number page number ...	search in AIT

Table 3–16: Command language, continued

Code	Function	Write Parameter	Read Parameter	Notes
Miscellaneous Data Commands				
36	Read 8/30 Row	designation code	40 Byte row data	only format 1 and 2 are supported 1st byte of row data is already hamming decoded
15	Read VPS		framing code counter 13 Byte VPS data	= 51H = incremented every VPS reception = biphase decoded VPS bytes 3–15
53	Read WSS		framing code counter 13 Byte WSS data	= 78H = incremented every WSS reception = 102 WSS elements from group 1 on
54	Read CAPTION 1		counter 6 Byte CAPTION data	= incremented every reception in field 1 = 3x oversampling
55	Read CAPTION 2		counter 6 Byte CAPTION data	= incremented every reception in field 2 = 3x oversampling
19	Read Rolling Header		24 Byte rolling header	every row 0 in cycle
31	Read Rolling Time		8 Byte rolling time	using time pointer
16	Read Quality		text lines hamming errors parity errors soft errors	updated every VBI
18	Read Reset Source		reset source	reset source: bit0 = clock supervision bit1 = voltage supervision bit2 = watchdog all bits in reset source are reset after read
46	Read Hamming	hamming (8,4) Byte	data	hamming Byte: bit0–3 = data bit6 = hamming error
47	Read Hamming 2	hamming (24,18) 1st Byte hamming (24,18) 2nd Byte hamming (24,18) 3rd Byte	address mode data	address: bit0–5 = address bit7 = hamming error mode: bit0–4 = mode data: bit0–6 = data
33	Copy Data	source dram bank source dram high source dram low length destination dram bank destination dram high destination dram low		copy data from DRAM to DRAM
Display Commands				
17	Read Display Mode		display mode character set font mapping	display mode: bit0 = forced boxing bit1 = reveal bit2 = box bit3 = time hold bit4 = page hold bit5 = row 24 hold bit6 = row 25 hold bit7 = row 26 hold
08	Display Mode	display mode (18) character set (06) font mapping (00)		display mode: see above character set: 6,38,40,55,70,128 font mapping: 0=latin 1=cyrillic/greek 2=arabic/farsi/hebrew 128=user defined
09	Display TTX Pointer	dram high (20) dram low (00)		page memory is copied to TTX pointer
10	Display Pointer	dram high (20) dram low (00) scroll counter (00)		display starts at pointer using scroll counter as line offset
11	Display Clear	dram high dram low		clear display bank beginning at pointer (26 rows * 86 Bytes)

Table 3–16: Command language, continued

Code	Function	Write Parameter	Read Parameter	Notes
13	Display Time Pointer	dram high (20) dram low (20)		8 Byte time string from packet x/00 is copied to time pointer
26	Display Page Request	magazine number page number subpage number high subpage number low display delay (1E)		magazine number: bit0–3 = magazine number bit4 = change display delay bit5 = display clear (on update) bit6–7 = not used subpage number: F0xx for rolling subpages display delay: delay after row 0 reception in steps of 3.24ms (255 = no update) only used if bit4 = 1
24	Read Display Page		magazine number page number subpage number high subpage number low	current page in display
48	Display Column	dram high dram low length Byte list ...		write to dram with increment of 86 Bytes = number of bytes in list
49	Display Fill	dram high dram low length character		repeated write of 1 character to dram = number of repeated writes
56	OSD Font Pointer	font mode (00) font pointer high font pointer low extension font pointer high extension font pointer low		font mode: bit0 = 0 = reset OSD font 2 pointer bit0 = 1 = load OSD font 2 pointer with following parameters
57	Display Read Column	dram high dram low length	Byte list ...	read from dram with increment of 86 Bytes = number of Bytes to read
58	User Character Set	language 000 (00) language 001 (01) language 010 (02) language 011 (03) language 100 (04) language 101 (05) language 110 (06) language 111 (07)		If character set 128 is selected via command 08 “Display Mode”, these 8 languages will be selected by option bits C14,C13,C12 when ESC code is inactive.
59	User ESC Character Set	esc language 000 (00) esc language 001 (00) esc language 010 (00) esc language 011 (00) esc language 100 (00) esc language 101 (00) esc language 110 (00) esc language 111 (00)		If character set 128 is selected via command 08 “Display Mode”, these 8 languages will be selected by option bits C14,C13,C12 when ESC code is active.
60	Full Row Attribute	full row attribute number of rows start row		set full row attribute of specified rows without changing level 2 bit
61	User Mapping	32 Byte mapping data		32 Bytes are copied into mapping ram via I/O page register 0276H

3.13.I/O Register

Most hardware-related functions of the TPU are controlled by memory mapped I/O of the 65C02. The application software has access to the I/O registers via I^{2}C bus using the CPU subaddresses SUB1 and SUB2 (see Section 3.14.1.1. on page 82).

Most of the I/O registers can only be written and will not return useful data when read by application software. Reset values are written by TPU during initialization.

Note:
For compatibility reasons, every undefined bit of a write register should be set to ‘0’. Undefined bits of a read register should be treated as “don’t care”.

0200 H	**R/W**	**CONTROL REGISTER**	
Bit	Reset	Write Function	Read Function
all	00 H	During reset the control register is loaded with the contents of the address FFF9H, but it can be read and written via software.	
7	0	1 = CPU disable 0 = CPU enable	
6	0	1 = program RAM disable 0 = program RAM enable	
5	0	1 = program ROM disable 0 = program ROM enable	
4	0	1 = character ROM disable 0 = character ROM enable	
3	0	1 = DMA interface disable 0 = DMA interface enable	
2	0	1 = I/O page disable 0 = I/O page enable	
1	0	1 = test mode on 0 = test mode off	
0	0	1 = burnin test mode (only if test pin high) 0 = normal test mode	1 = burnin test mode 0 = normal test mode

0202 H	**Write**	**STANDBY**
Bit	Reset	Function
2	0	1 = digital circuitry power off(CPU still active with slow clock) 0 = digital circuitry power on
1	0	1 = analog front-end power off 0 = analog front-end power on

0213 H	**Write**	**INTERFACE MODE**
Bit	Reset	Function
1	0	1 = standby enable (if bit 2 of register 0202H = 1) 0 = standby disable

0251 H	**Write**	**BLANKING STOP**
Bit	Reset	Function
all	07 H	horizontal stop of blanking pulse in character increments correct blanking pulse cannot be guaranteed if blanking start = blanking stop

0252 H	**Write**	**BLANKING START**
Bit	Reset	Function
all	00 H	horizontal start of blanking pulse or self-timed HSYNC in character increments correct blanking pulse cannot be guaranteed if blanking start = blanking stop

0254 H	**Write**	**DISPLAY MODE 1**
Bit	Reset	Function
7	0	1 = OSD layer always uses FONT 1 0 = OSD layer changes from FONT 1 to FONT 2 if ASCII≥20H
6	1	1 = enable OSD layer 0 = disable OSD layer
5	1	1 = active flash phase of OSD layer 0 = inactive flash phase of OSD layer
4	0	1 = 13 scanlines/character 0 = 8 scanlines/character
3 to 0	0	With this scan line the OSD layer starts display of the first text line. By slow incrementing of this value soft scroll begins.

0255 H	**Write**	**DISPLAY MODE 2**
Bit	Reset	Function
7	0	1 = OSD layer control code defines character set 0 = OSD layer control code defines color
3	1	1 = 10.125MHz display clock 0 = 20.25MHz display clock
2	1	1 = font pointer offset 10 scanlines/character 0 = font pointer offset 8 or 16 scanlines/character (depending on bit 1)
1	0	1 = font pointer offset 16 scanlines/character 0 = font pointer offset 8 scanlines/character
0	1	1 = 10 scanlines/character 0 = 8 or 13 scanlines/character (depending on bit 4 in register 0254 H)

025A H	**Write**	**PRIO MODE**
Bit	Reset	Function
5 to 3	110	prio code for shadow pixel
2 to 0	101	prio code for normal pixel

025B H	**R/W**	**FB Mode**	
Bit	Reset	Write Function	Read Function
all	00 H		every read resets status
7	0	color bit 4(color output of OSD layer)	
6	0	color bit 3(color output of OSD layer)	
4	0	1 = inverted color output 0 = normal color output	

0260 H	**Write**	**OSD LAYER VERTICAL START**
Bit	Reset	Function
all	00 H 60 H	9-bit value defining vertical position (in scanline) 1st write: bit 0 = MSB 2nd write: bit 7 to 0 = 8 LSBs

0261 H	**Write**	**OSD LAYER VERTICAL STOP**
Bit	Reset	Function
all	01 H 28 H	9-bit value defining vertical position (in scanline) 1st write: bit 0 = MSB 2nd write: bit 7 to 0 = 8 LSBs

0262 H	**Write**	**OSD LAYER HORIZONTAL START**
Bit	Reset	Function
all	16 H	8-bit value defining horizontal start position (in character)

0264 H	**Write**	**OSD LAYER TEXTPOINTER**
Bit	Reset	Function
all	–	16-bit value defining memory address of text 1st write: bit 7 to 0 = 8 MSBs 2nd write: bit 7 to 0 = 8 LSBs

0265 H	**Write**	**OSD LAYER 2nd COLOR START**
Bit	Reset	Function
all	01 H 38 H	9-bit value defining vertical start for 2nd color (in scanline) 1st write: bit 0 = MSB 2nd write: bit 7 to 0 = 8 LSBs

0266 H	**Write**	**OSD LAYER 2nd COLOR**
Bit	Reset	Function
6 to 0	0C H	7-bit value defining 2nd color 2nd color is used during 1 text row (8, 10 or 13 scanlines) after 2nd color start

0267 H	**Write**	**WST LAYER VERTICAL START**
Bit	Reset	Function
all	00 H 24 H	9-bit value defining vertical position (in scanline) 1st write: bit 0 = MSB 2nd write: bit 7 to 0 = 8 LSBs

0268 H	**Write**	**WST LAYER HORIZONTAL START**
Bit	Reset	Function
all	0F H	8-bit value defining horizontal start position (in character)

026A H	**Write**	**WST LAYER VERTICAL STOP**
Bit	Reset	Function
all	01 H 28 H	9-bit value defining vertical position (in scanline) 1st write: bit 0 = MSB 2nd write: bit 7 to 0 = 8 LSBs

026B H	**Write**	**WST LAYER LAST ROW**
Bit	Reset	Function
all	01 H 1E H	9-bit value defining last scanline of the last row to display level 1 double height after this scanline the level 1 double height attribute will not be decoded anymore 1st write: bit 0 = MSB 2nd write: bit 7 to 0 = 8 LSBs

026C H	**Write**	**RGB MODE**
Bit	Reset	Function
5	0	1 = WST layer mixed mode 0 = WST layer normal mode
4 to 3	0	11 = WST layer top 10 = WST layer opaque bottom 01 = WST layer transparent bottom 00 = WST layer disable
2	0	1 = OSD layer mixed mode 0 = OSD layer normal mode
1 to 0	0	11 = OSD layer top 10 = OSD layer opaque bottom 01 = OSD layer transparent bottom 00 = OSD layer disable

026D H	**Write**	**SYNC MODE**
Bit	Reset	Function
5	0	1 = double scan enable 0 = double scan disable
4	0	1 = blanking disable 0 = blanking enable

026F H	**Write**	**DISPLAY MODE 3**
Bit	Reset	Function
7	1	1 = 10 pixel/character 0 = 8 pixel/character
6	0	1 = double dot size in vertical direction(OSD layer only) 0 = normal dot size in vertical direction
5	0	1 = double dot size in horizontal direction(OSD layer only) 0 = normal dot size in horizontal direction
4	0	1 = black colors replaced by transparent & shadow(OSD layer only) 0 = black colors displayed black
3 to 0	F H	4-bit value defining delay of horizontal start for both layers (in pixel) delay = mod_{16} (character_width – 2 – value)(leftmost position should not be used!)

0270 H	**Write**	**DISPLAY MODE 4**
Bit	Reset	Function
4	0	1 = new mosaic mode (single switch to character set 1) 0 = old mosaic mode (static switch to character set 1)
3	0	1 = level 1 display mode (read 40 Byte from display bank) 0 = level 2 display mode (read 86 Byte from display bank)
2	0	1 = boxing enable 0 = boxing disable
1	0	1 = reveal enable 0 = reveal disable
0	0	This bit is taken as flash clock for the WST layer, the frequency should be around 6 Hz.

0273 H	**Write**	**DISPLAY MODE 5**
Bit	Reset	Function
4	0	WST layer scan line counter preset (LSB for zoom mode)
3 to 0	0	WST layer scan line counter preset

028E H	**Write**	**DRAM MODE**
Bit	Reset	Function
4	0	1 = next CPU write without WEQ but with address increment 0 = normal CPU write mode
3	0	1 = reset address pointer and switch off refresh during standbyt 0 = keep address pointer and refresh during standby
2	1	1 = display channel enable 0 = display channel disable
1	1	1 = slicer channel enable 0 = slicer channel disable

029C H	**Read**	**ACQ SOFT ERROR COUNTER**
Bit	Reset	Function
5 to 0	–	6-bit soft error counter counts number of soft error corrected bytes counter stops at 63 reset after read

029E H	**Read**	**ACQ SYNC STATUS**	
Bit	Reset	Function	
7	–	1 = field 1 0 = field 2	set at line 624 (PAL) or line 524 (NTSC) reset at line 313 (PAL) or line 263 (NTSC)
6	–	1 = vertical retrace 0 = vertical window	set at line 628 (PAL) or line 528 (NTSC) reset at line 624 (PAL) or line 524 (NTSC)

029F H	Write	ACQ STANDARD
Bit	Reset	Function
7	0	1 = CAPTION enable in field 2 0 = CAPTION disable in field 2
6	0	1 = CAPTION enable in field 1 0 = CAPTION disable in field 1
5	0	1 = VPS enable 0 = VPS disable
7 to 5	0	VPS and CAPTION cannot be used at the same time, therefore these combinations are used to enable WSS reception on a PAL+ signal 0 = 1 = VPS 2 = CAPTION field 1 3 = WSS & VPS 4 = CAPTION field 2 5 = WSS & VPS 6 = CAPTION field 1&2 7 = WSS
4	1	1 = acquisition enable 0 = acquisition disable
1 to 0	0	00 = PAL mode 10 = NTSC mode 11 = Caption full field mode

02A3 H	Write	ACQ VIDEO INPUT
Bit	Reset	Function
1 to 0	0	00 = VIN1 01 = VIN2 10 = VIN3 11 = VIN4

02A4 H	Read	ACQ HSYNC COUNTER
Bit	Reset	Function
7 to 0	0	number of detected horizontal sync pulses per frame divided by 4 sync pulse is detected if within horizontal window of HPLL counter is latched with vertical sync, the register can be read at any time

3.14. I^{2}C-Bus Slave Interface

Communication between the TPU and the TV controller is done via I^{2}C bus. For detailed information on the I^{2}C bus please refer to the Philips manual ‘I^{2}C bus Specification’.

The TPU acts as a slave transmitter/receiver and uses clock synchronization to slow down the data transfer if necessary. General call address will not be acknowledged.

Different memories and functions of TPU can be accessed by subaddressing. The byte following the slave address byte is defined as the subaddress byte.

Maximum length of an I^{2}C telegram is 256 Bytes following slave address and subaddress byte. The interface supports data transfer with autoincrement.

The I^{2}C bus interface is interrupt-driven and uses an internal 48-Byte buffer to collect I^{2}C data in real-time without disturbing internal processes. This is done to avoid clock synchronization as far as possible. When the TPU has to process the I^{2}C buffer and the I^{2}C telegram has not yet been stopped, the I^{2}C clock line will be held down.

The time required to process the I^{2}C buffer depends on other processes running inside the TPU firmware. Thus the following I^{2}C telegram addressing the TPU can be held after the slave address byte until the old telegram is completely processed.

3.14.1. Subaddressing

Access to all memory locations and to the command interface is achieved by subaddressing. Both the external DRAM and the internal CPU memory can be addressed completely. The TPU acknowledges 6 different subaddresses following the slave address (see Table 3–17 on page 82).

The following symbols are used to describe the I^{2}C example telegrams:

`<`	`start condition`
`>`	`stop condition`
`ab`	`address bank byte`
`ah`	`address high byte`
`al`	`address low byte`
`cc`	`command byte`
`dd`	`data byte`
`ss`	`status byte`
`..`	`0 – n continuation bytes`

Table 3–17: I^{2}C bus subaddresses

Name	Binary Value	Hex Value	Mode	Function
TPU	0010 001x	22, 23	W, R	TPU slave address
Sub 1	0111 1000	78	W	subaddressing CPU (static)
Sub 2	0111 1001	79	W	subaddressing CPU (autoincrement)
Sub 3	0111 1010	7A	W	subaddressing DRAM (autoincrement)
Sub 4	0111 1011	7B	W	subaddressing command language
Data	0111 1100	7C	R/W	subaddressing data register
Status	0111 1101	7D	R	status register bit 7 = command wait bit 6 = command invalid bit 5 = command found no data bit 4 = not used bit 3 = not used bit 2 = not used bit 1 = 0 bit 0 = 0

3.14.1.1. CPU Subaddressing

There are 2 CPU subaddresses to access CPU memory: either with static memory address or with autoincrementing memory address. The main purpose of CPU subaddressing is to write text into the OSD buffer and to access the I/O page (see Section 3.13. on page 76). The static CPU subaddress can be used to write more than 1 Byte into the same I/O page register.

The CPU subaddress has to be followed by 2 address bytes defining the CPU memory address. The following data byte is written into this address. In the case of autoincrement the continuation bytes are written into incrementing memory addresses.

The CPU telegram can be stopped after the 2 memory address bytes. The following I^{2}C telegram subaddressing the data register will continue data transfer to or from the CPU memory. The data transfer will always start at the CPU memory address (autoincrement is not saved).

```
< 22 78 ah al dd .. >
< 22 79 ah al dd .. >
< 22 79 ah al  > <  22 7C dd .. >
```

Data is directly written into CPU memory without using the I^{2}C buffer of TPU and without waiting for a stop condition.

3.14.1.2. DRAM Subaddressing

DRAM access is necessary to generate level 2 displays. The external DRAM can be addressed on byte level. The maximum DRAM size of 16 Mbit requires a 21-bit memory address pointer. The format of the DRAM address pointer is shown in Fig. 3–20.

▨	5-bit Bank	8-bit High	8-bit Low

Fig. 3–20: DRAM address pointer

The DRAM subaddress has to be followed by 3 address bytes defining the DRAM address pointer. The following data byte is written into this address.

DRAM subaddressing always uses autoincrement. Separate read and write DRAM address pointers are saved for autoincrement.

The DRAM telegram can be stopped after the 3 address pointer bytes. The following I^{2}C telegram subaddressing the data register will continue data transfer to or from the DRAM.

When reading the DRAM, the first data byte the TPU returns is a dummy byte, which has to be ignored.

```
< 22 7A ab ah al dd .. >
< 22 7A ab ah al  > < 22 7C dd .. >
< 22 7A ab ah al  > < 22 7C <  23 dd ..>
```

Data written to the DRAM subaddress is collected first in the I^{2}C buffer of TPU and is copied to DRAM when the buffer is full (48 Bytes) or after stop condition. During the time the buffer is copied to DRAM the TPU will hold the I^{2}C clock line down.

Reading data from the DRAM subaddress is also buffered internally. Reading the first byte will only empty the I^{2}C buffer. Every time the buffer is empty, the TPU will copy 48 Bytes from DRAM into the I^{2}C buffer. During this time the TPU will hold the I^{2}C clock line down.

3.14.1.3. Command Subaddressing

TPU supports a command language, allowing the host controller to start complex processing inside the TPU with simple commands (see Section 3.12. on page 68). Commands have to be sent to the command subaddress.

The command subaddress has to be followed by the command code. The following data bytes are taken as command parameters.

The execution time for commands depends on other processes running inside the TPU firmware, therefore the host controller has to read the status register to get information about the running command before reading command parameter or starting other commands.

The status register returns information about the command interface. The ‘command wait’ bit is set during execution of a command and is reset when a command is executed completely and read parameters are available. If a non-existing command is sent to the TPU, the ‘command invalid’ bit is set. If a command could not be executed successfully, the ‘command found no data’ bit is set. In this case the read parameters of this command are not valid.

Reading status from TPU is done by subaddressing the status register followed by repeated start condition and slave read address (see Fig. 3–21).

```
< 22 7B cc dd ..  >
< 22 7D <  23 ss .. >
< 22 7C <  23 dd .. >
```

Telegrams subaddressing the command interface are buffered and processed after receiving the stop condition. Therefore the command code and all necessary command parameters have to be included in a single telegram.

3.14.1.4. Data Subaddressing

Writing data to TPU memory is possible by subaddressing the data register directly. The data is then written into memory addressed by the foregoing telegram.

```
< 22 7C dd .. >
```

Reading data from TPU is done by subaddressing the data register followed by a repeated start condition and slave read address (see Fig. 3–21). The returned data depend on the subaddress selected in the preceding TPU telegram.

```
< 22 7C <  23 dd .. >
```

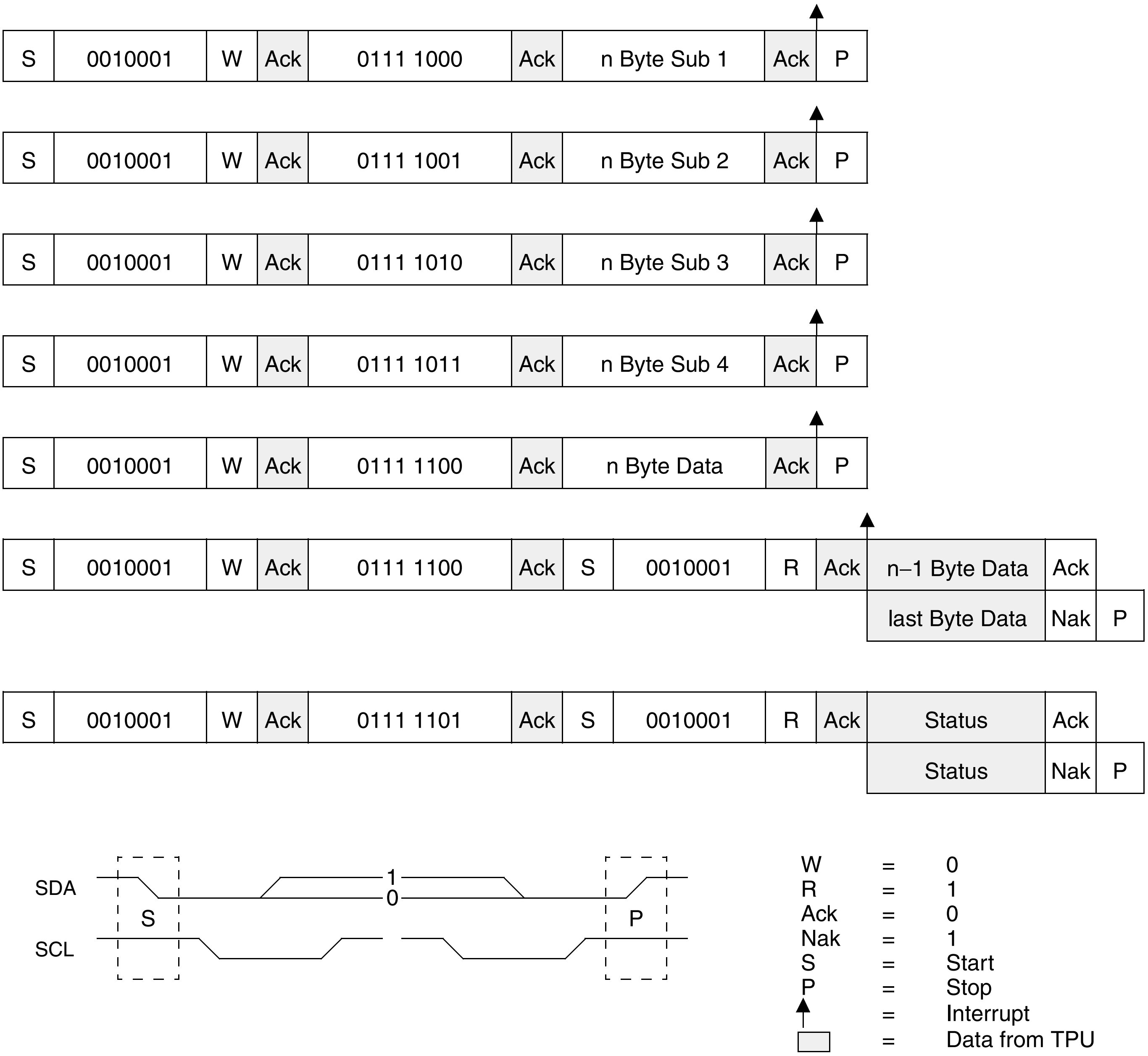


Fig. 3–21: I^{2}C bus protocol

3.14.1.5. Hardware Identification

A separate I^{2}C bus slave register is reserved to read out the hardware version of VCT 38xxA. This register is active in standby mode.

I^{2}C Sub address	Number of bits	Mode	Function	Default	Name
h’9F	16	r	Hardware version number bit[7:0] hardware id (A3=h’13, B1=h’21 a.s.o.) bit[15:8] product code VCT38xy (VCT3832=h’32)	read only	HWID TC PROD

4. Audio Processing

4.1. Introduction

The audio processing allows input selection and volume control for mono audio sources either from tuner or from SCART input.

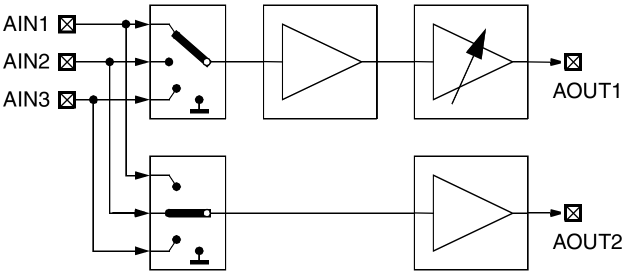


Fig. 4–1: Audio processing

4.2. Input Select

Both audio output channels can be switched to any of the three audio input channels. Only the audio output channel AOUT1 can be volume controlled.

4.3. Volume Control

The analog volume control covers a range from +18 dB and –75 dB. The lowest step is the mute position. Step size is split into a 3-dB and a 1.5-dB range.

–75 dB...–54 dB :	3 dB step size
–54 dB...+18 dB :	1.5 dB step size

4.4. I^2C-Bus Slave Interface

The input selection and analog volume is controlled via the audio control register ACON. This I^2C register is activated by the chip address of the video back-end processing (see Table 2–2 on page 32).

Table 4–1: Audio control register

I^2C Sub address	Number of bits	Mode	Function	Default	Name
h'34	16	w	Audio Control bit [5:0] volume control 000000 mute 000001 –75db ... 000111 –57.0dB 001000 –54.0dB ... 101011 –1.5dB 101100 0.0dB 101101 +1.5dB ... 110110 +15.0dB 110111 +16.5dB 111000 +18.0dB	0	ACON AVOL
			bit [7:6] reserved		
			bit [9:8] audio input select 1 00 mute 01 AIN1 10 AIN2 11 AIN3		ASEL1
			bit [11:10] audio input select 2 00 mute 01 AIN1 10 AIN2 11 AIN3		ASEL2
			bit[12] low power mode 0 disable low power mode 1 enable low power mode		ALPM
			bit[15:13] reserved		

5. TV Controller

5.1. Introduction

The TV controller basically consists of the CPU, RAM, ROM, and a number of peripheral modules.

For instance:

- a memory banking module is included to allow access to more than 64 kB memory.
- a bootloader software is included to allow in-system-downloading of external code to Flash memory via the I^2C interface.

The TV controller runs the complete software necessary to control a TV set. The software includes control of the audio, video, OSD, and text processors on chip, as well, as control of external devices like tuner or stereo decoder.

Communication between the TV controller and external devices is done either via I^2C bus interface or via programmable port pins.

The TV Controller is clocked with $f_{OSC} = f_{XTAL}/2$.

5.2. CPU

The CPU is fully compatible to WDC's W65C02 microprocessor. The processor has 8-bit registers/accumulator, an 8-bit data bus, and a 16-bit address bus. For further information about the CPU core, please refer to the WDC W65C02 data sheet.

5.2.1. CPU Slow Mode

To reduce power consumption considerably, the user can reduce the internal CPU clock frequency to 1/256 of the normal f_{CPU} value. In this CPU Slow mode, program execution is reduced to 1/256 of the normal speed, but clocking of most other modules remains unaffected. The modules that are affected by CPU Slow mode are:

1. CPU and Interrupt Controller with all internal and external interrupts
2. RAM, ROM and DMA
3. Watchdog

Some modules must not be operated during CPU Slow mode. Refer to module sections for details.

After reset the CPU is in Fast mode ($f_{CPU} = f_{OSC}$).

CPU Slow mode is enabled by clearing flag CPUFST in standby register SR1. The CPU clock frequency reduction to $f_{OSC}/256$ will take effect after a maximum delay of 256 f_{OSC} periods.

Returning CPU to Fast mode is done by setting flag CPUFST to High. The CPU clock frequency will immediately change to its normal f_{OSC} value.

Fig. 5–1 shows the memory access signals during CPU fast and slow mode.

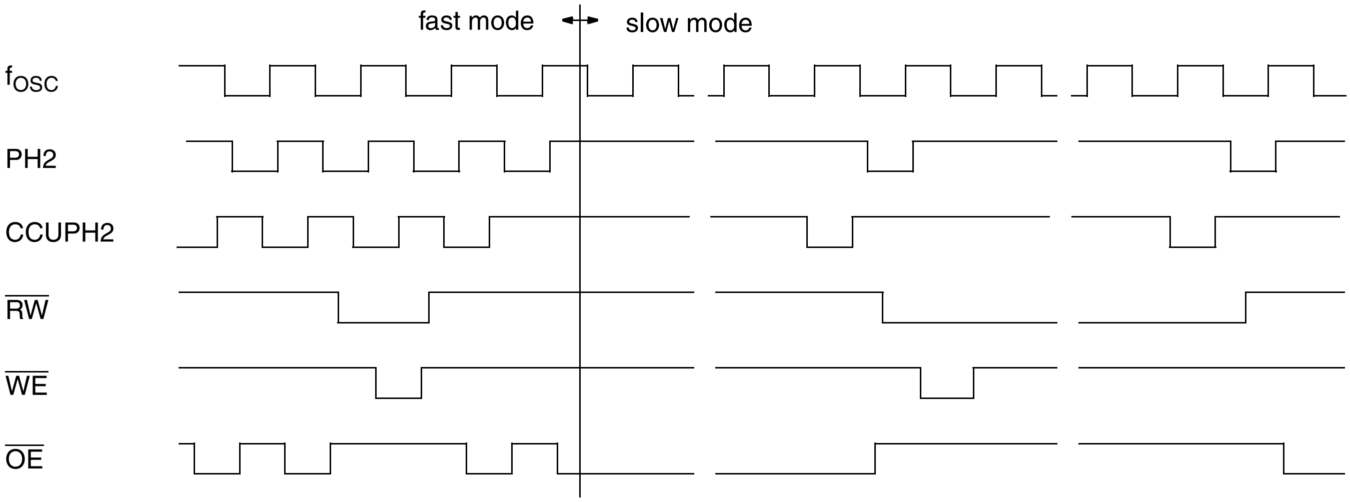


Fig. 5–1: Memory access signals

5.3. RAM and ROM

On-chip RAM is composed of static RAM cells. The RAM will hold all information during reset, as long as the specified operating voltages are available.

The 64PSDIP Multi Chip Module contains a 128-KByte Flash EEPROM of the ST M29W010B type. These devices exhibit electrical Byte program and block erase functions. Refer to the ST M29W010B data sheet for details.

5.3.1. Address Map

The following ROM addresses are reserved and cannot be used to store program code.

Table 5–1: Reserved (physical) addresses

Addresses	Usage
00FFC6 – 00FFD5	Manufacturer ROM ID
00FFD6 – 00FFD7	reserved for bootloader
00FFD8 – 00FFF7	Interrupt Vectors
00FFF8	reserved
00FFF9	Control Word (during reset)
00FFFA – 00FFFB	NMI Vector (expanded by Interrupt Controller)
00FFFC – 00FFFD	Reset Vector
0xFFFE – 0xFFFF	IRQ/BRK Vector

A 16-Byte address space is reserved as “Manufacturer ROM ID”. This area contains a unique ROM ID number which has to be agreed between Micronas and the customer. Especially the first 6 digits identify customer and version. As an example a Micronas demo software is identified like “MI1108 240700 TV”.

Table 5–2 shows the internal memory segmentation. Internal program RAM and ROM can be disabled via the Control Register (chapter 5.4. on page 87). The internal text RAM can be disabled via Standby Register 0 (see page 89).

All memory locations not available internally will be addressed as external memory. It is possible to operate with internal and external memory in parallel, but overlapping memory segments will always be addressed internally.

During internal memory access, the pins DB0-DB7, WExQ and OExQ are tristate. For emulation and test purposes it is possible to change this behavior via the

Table 5–2: Internal Memory Locations

Addresses	Internal Memory
000000 – 000FFF	4k Program RAM
001E00 – 001FFF	I/O Register
002000 – 0023FF	1k Bootloader ROM
002400 – 019FFF	95k Program ROM
0A0000 – 0A3FFF	16k Text RAM

5.3.2. Bootloader

A segment of the internal ROM is reserved for bootloader code. Via this bootloader code it is possible to download additional code into the internal RAM and execute this code. The downloaded code can be used to program the external Flash EEPROM.

After reset the bootloader checks the I^2C bus pins SDA and SCL for a special identification sequence. If no identification sequence is detected, the bootloader starts the application program code.

The bootloader checks the address FFD6/FFD7 of the external memory if there is a predefined pattern (A55Ah). If so, it starts the external application software else it starts the internal application software.

5.4. Control Register

The Control Register CR serves to configure the ways, by which certain system resources are accessed during operation. The main purpose is to obtain a variable system configuration during IC test.

Upon each High transition on the RESQ pin internal hardware reads data from address location 00FFF9h and stores it to the CR. The state of the TEST pin at this timepoint specifies which program storage source is accessed for this read:

– With the TEST pin Low, the control byte is read from internal program storage (mask ROM). With location 00FFF9h set to FFh, this is the setting for stand-alone operation.

– With the TEST pin High, the control byte is read from external memory via the test bus (for test purposes only).The system will thus start up according to the configuration defined in address location 00FFF9h and automatically copied to register CR.

1F01		CR		Control Register				
bit	7	6	5	4	3	2	1	0
r/w	RESLNG	TSTTOG	DISEXT	MFM	TSTROM	IROM	IRAM	ICPU
reset	Value of 00FFF9h							

RESLNG **Reset Pulse Length**
r/w1: Pulse length is 4095/f_{OSC}.
r/w0: Pulse length is 16/f_{OSC}.

This bit specifies the length of the reset pulse which is output at pin RESQ following an internal reset. If pin TEST is 1 the first reset after power on is short. The following resets are as programmed by RESLNG. If pin TEST is 0 all resets are long.

TSTTOG **TEST Pin Toggle**
r/w1: Pin TEST can toggle the Multi Function pins.
r/w0: Pin TEST can't toggle the Multi Function pins.

This bit is used for test purposes only. If TSTTOG is true in IC active mode, pin TEST can toggle the Multi Function pins between Bus mode and normal mode.

DISEXT **Disable External Memory Access**
r/w1: DB0–DB7, WExQ and OExQ output pins are tristate during internal memory access (see Fig. 5–2 on page 89).
r/w0: DB0–DB7, WExQ and OExQ output pins are active during internal memory access.

MFM **Multi Function pin Mode**
r/w1: Enable normal mode.
r/w0: Enable Test Bus mode.

TSTROM **Test ROM (mask ROM parts only)**
r/w1: Disable internal Test ROM.
r/w0: Enable internal Test ROM (@ IROM=1).

IROM **Internal ROM**
r/w1: Enable internal CPU ROM.
r/w0: Disable internal CPU ROM.

IRAM **Internal RAM**
r/w1: Enable internal CPU RAM.
r/w0: Disable internal CPU RAM.

ICPU **Internal CPU**
r/w1: Enable internal CPU.
r/w0: Disable internal CPU.

Table 5–5: Some commonly used settings for address location 00FFF9h.

Code	TEST Pin	Operation Mode
FFh	0	Stand-alone with internal ROM or Flash
DFh	0	Emulator mode (CPGA257 package)
ABh	1	External program storage connected to Multi Function pins in Bus mode

Table 5–3: TSTROM and IROM usage in mask ROM parts

TSTROM	IROM	selected program storage
1	1	internal CPU ROM
0	1	internal Test ROM
x	0	external on Multi Function pins in Bus mode

Table 5–4: TSTTOG and MFM usage

TSTTOG	MFM	TEST pin	Multi Function Pins
x	1	x	normal mode
1	0	1	normal mode
1	0	0	Bus mode
0	0	x	Bus mode

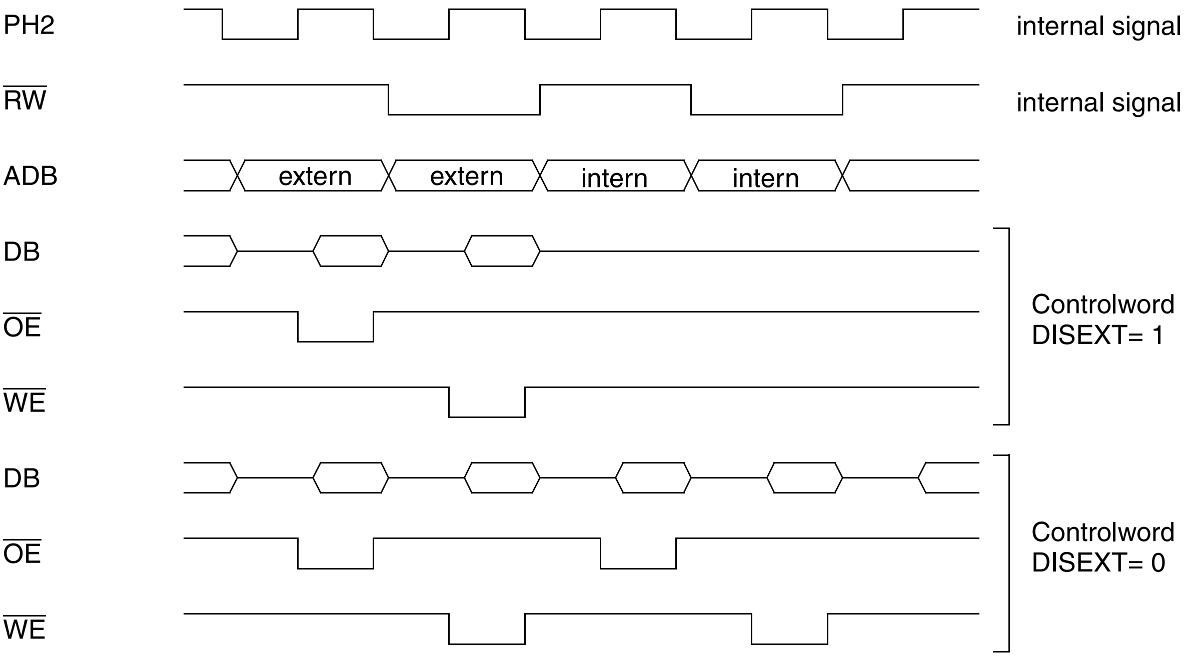


Fig. 5–2: Internal/external memory access

5.5. Standby Registers

The Standby registers allow the user to switch on/off power or clock supply of single modules. With these flags it is possible to greatly influence power consumption and its related electromagnetic interference.

For details about enabling and disabling procedures and the standby state refer to the specific module descriptions.

The minimum IC current consumption is obtained with all standby registers set to 00h.

1F08	SR0			Standby Register 0				
bit	7	6	5	4	3	2	1	0
r/w		PWM1	PWM0			TRAM	CCC	TVPWM
reset	0	0	0	0	0	0	0	0

PWM1	**Pulse Width Modulator 1**
r/w1:	Module active.
r/w0:	Module off.
PWM0	**Pulse Width Modulator 0**
r/w1:	Module active.
r/w0:	Module off.
TRAM	**Text RAM**
r/w1:	Module active
r/w0:	Module off
CCC	**Capture Compare Counter**
r/w1:	Module active.
r/w0:	Module off.
TVPWM	**Tuning Voltage Pulse Width Modulator**
r/w1:	Module active.
r/w0:	Module off.

1F09	SR1			Standby Register 1				
bit	7	6	5	4	3	2	1	0
r/w		CPUFST			ADC		TIM1	TIM0
reset	0	1	0	0	0	0	0	0

CPUFST	**CPU Fast Mode**
r/w1:	Fast mode: $f_{CPU} = f_{XTAL} / 2$
r/w0:	Slow mode: $f_{CPU} = f_{XTAL} / 512$
ADC	**ADC Module**
r/w1:	Module active.
r/w0:	Module off.
TIM1	**Timer 1**
r/w1:	Module active.
r/w0:	Module off.
TIM0	**Timer 0**
r/w1:	Module active.
r/w0:	Module off.

1F0A		SR2		Standby Register 2				
bit	7	6	5	4	3	2	1	0
r/w		PWM3	PWM2			I2C		MB
reset	0	0	0	0	0	0	0	0

PWM3	**Pulse Width Modulator 3**
r/w1:	Module active.
r/w0:	Module off.
PWM2	**Pulse Width Modulator 2**
r/w1:	Module active.
r/w0:	Module off.
I2C	**I^2C-Bus Master Interface**
r/w1:	Module active.
r/w0:	Module off.
MB	**Memory Banking**
r/w1:	Module active.
r/w0:	Module off.

5.6. Test Registers

Test registers are for manufacturing test only. They must not be written by the user with values other than their reset values (00h). They are valid independent of the TEST input state.

In all applications where a hardware reset may not occur over long times, it is good practice to force a software reset on these registers within appropriate intervals.

1FFE		TST1		Test Register 1				
bit	7	6	5	4	3	2	1	0
w	For testing purposes only							
reset	0	0	0	0	0	0	0	0

1FFF		TST2		Test Register 2				
bit	7	6	5	4	3	2	1	0
w	For testing purposes only							
reset	0	0	0	0	0	0	0	0

1FFD		TST3		Test Register 3				
bit	7	6	5	4	3	2	1	0
w	For testing purposes only							
reset	0	0	0	0	0	0	0	0

1FFC		TST4		Test Register 4				
bit	7	6	5	4	3	2	1	0
w	For testing purposes only							
reset	0	0	0	0	0	0	0	0

1FFB		TST5		Test Register 5				
bit	7	6	5	4	3	2	1	0
r	For testing purposes only							
reset	0	0	0	0	0	0	0	0

5.7. Reset Logic

5.7.1. Alarm Function

An alarm comparator on the pin RESQ allows the detection of a threshold higher than the reset threshold. An alarm interrupt can be triggered with the output of this comparator.

The interrupt source output of this module is routed to the Interrupt Controller logic. But this does not necessarily select it as input to the Interrupt Controller. Check section “Interrupt Controller” for the actually selectable sources and how to select them.

The intended use of this function is made, when a system uses a 3.3V regulator with an unregulated input. In this case, the unregulated input, scaled down by a resistive divider, is fed to the RESQ pin. With falling regulator input voltage this alarm interrupt is triggered first. Then the reset threshold is reached and VCT 38xxA is reset before the regulator drops out.

The time interval between the occurrence of the alarm interrupt and the reset may be used to save process data to nonvolatile memory. In addition, power saving steps like turning off other devices may be taken to increase the time interval until reset. The alarm interrupt is a level triggered interrupt. The interrupt is active as long as the voltage on pin RESQ remains between the two thresholds of alarm and reset (see Fig. 5–3 on page 92).

5.7.2. Software Reset

The TV controller software can generate a reset via the Reset Control Register (see page 94). To prevent the TV controller from carrying out a reset in this case, the internal CPU reset can be disconnected from the RESQ pin.

5.7.2.1. From Standby into Normal Mode

To switch the whole TV application from standby operation into normal mode the controller has to perform the following sequence:

- RC.RESDIS = 1, RC.DCOCLP = 1
- RC.RESOUT =1
- switch on power supply
- wait for stable power supply
- RC.SELCLK = 1, RC.I2CEN = 1
- RC.DCOCLP = 0, RC.RESOUT = 0
- wait for RC.ALI = 0 (ext. capacitor!)
- RC.RESDIS = 0
- init DMA interface
- init TPU, VDP and Audio
- init external devices

5.7.2.2. From Normal into Standby Mode

To switch the whole TV application from normal mode into standby operation the controller has to perform the following sequence:

- RC.DCOCLP = 1, RC.I2CEN = 0
- wait 1ms for stable 20.25MHz DCO
- RC.SELCLK = 0
- turn off power supply
- set TPU into standby mode
- SR0 = 2, SR1 = 8, SR2 = 0

5.7.3. Internal Reset Sources

The VCT 38xxA contains three internal circuits that are able to generate a system reset: watchdog, supply supervision, and clock supervision.

All internal resets are directed to the open drain output of pin RESQ. Thus a “wired or” combination with external reset sources is possible. The RESQ pin is current limited and therefore large external capacitances may be connected.

All internal reset sources initially set a reset request flag. This flag activates the pull-down transistor on the RESQ pin. An internal reset prolongation counter starts, as soon as no internal reset source is active any more. It counts 4096 f_{CPU} periods (for alternative settings refer to register CR) and then resets the reset request flag, thus releasing the RESQ pin.

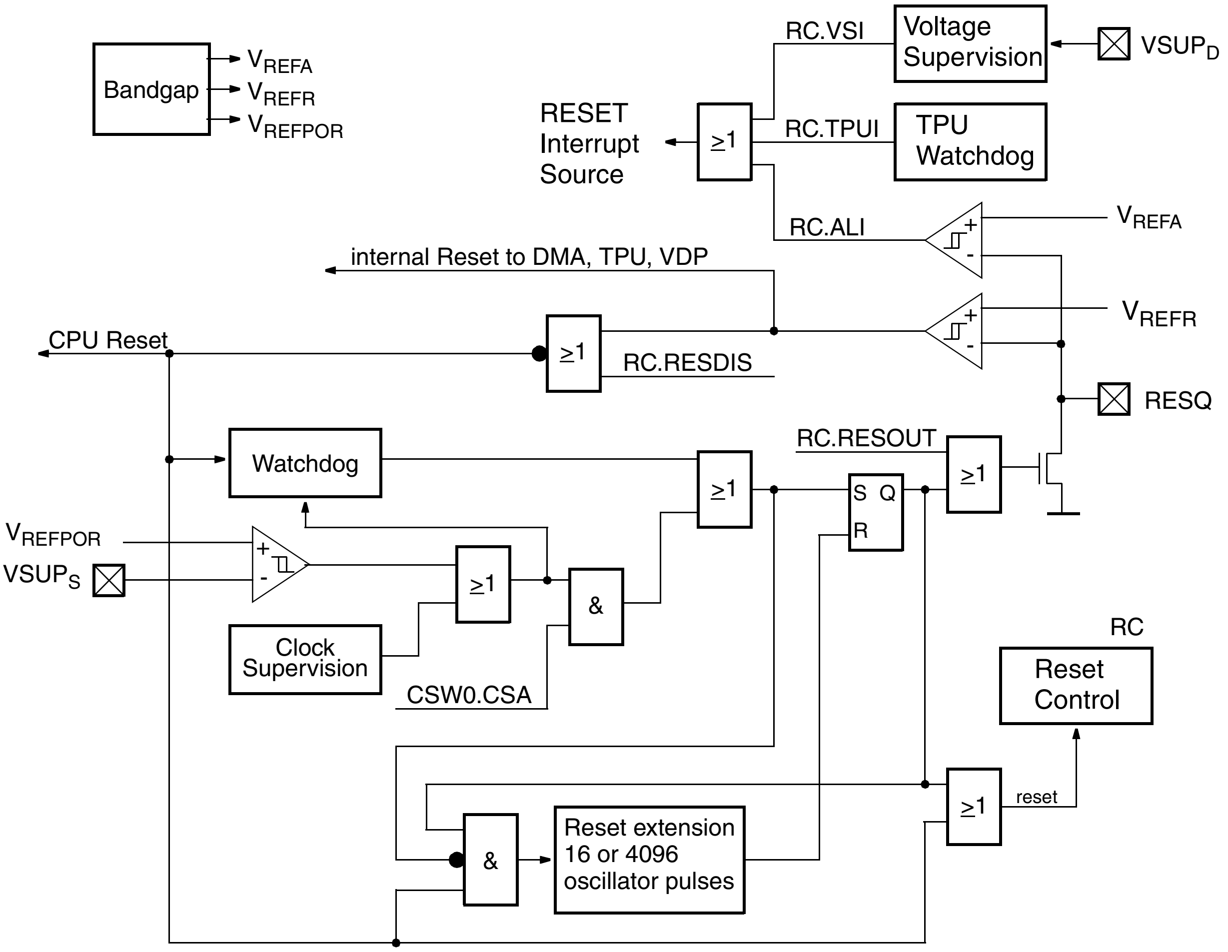


Fig. 5–3: Block diagram of reset logic

5.7.3.1. Supply Supervision

An internal bandgap reference voltage is compared to $VSUP_S$. A $VSUP_S$ level below the Supply Supervision threshold VREFPOR will permanently pull the pin RESQ low and thus hold the VCT 38xxA in reset state (see Fig. 5–3 on page 92). This reset source is active after reset and can be enabled/disabled by flag CSA in register CSW0.

5.7.3.2. Clock Supervision

The Clock Supervision monitors the CPU clock frequency f_{CPU}. A frequency level below the clock supervision threshold of approx. 200 kHz will permanently pull the pin RESQ low and thus hold the IC in reset (see Fig. 5–3 on page 92). This reset source is active after reset and can be enabled/disabled by flag CSA in register CSW0.

A frequency exceeding the specified clock frequency is not detected.

5.7.3.3. Watchdog

The Watchdog module serves to monitor undisturbed program execution. A failure of the program to retrigger the Watchdog within a preselectable time will pull the RESQ pin low and thus reset the VCT 38xxA (see Fig. 5–3 and Fig. 5–4). The Watchdog reset source is only enabled after the first write access to register CSW1 (see Section 5.7.3.2. on page 92).

Once the Watchdog is enabled, it cannot be disabled anymore, neither by software nor by pulling down the external RESQ pin. Only after power up the watchdog is disabled.

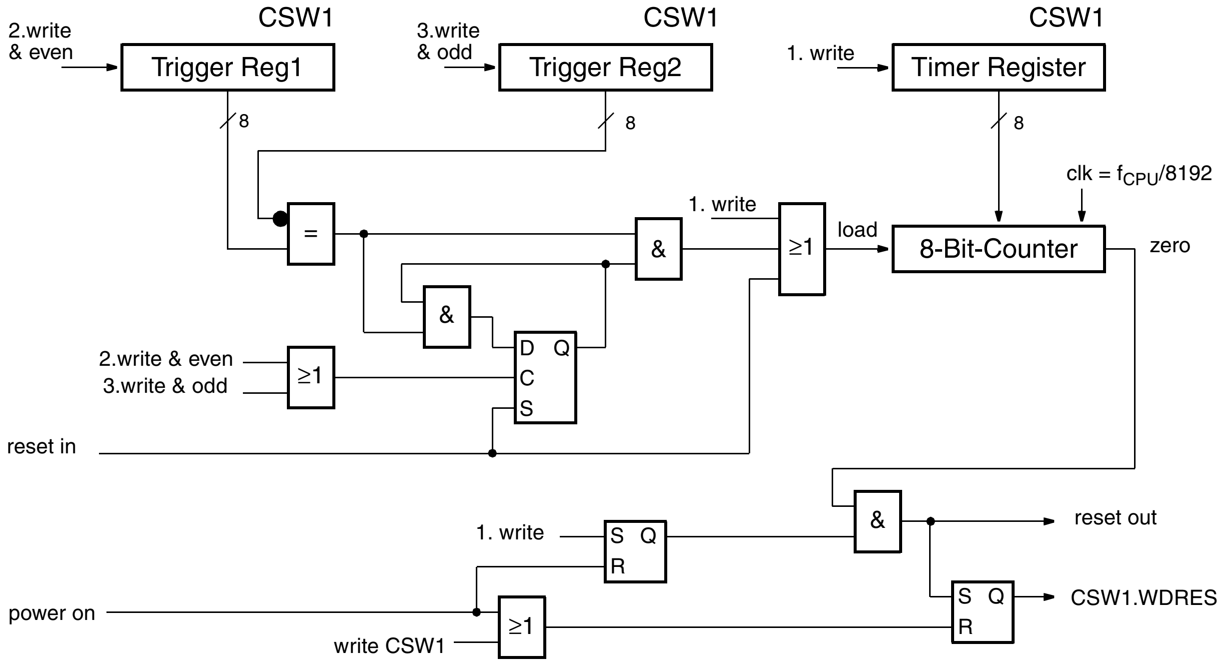


Fig. 5–4: Block diagram of watchdog

The Watchdog contains a down-counter that generates a reset when it wraps from zero to FFh. It is reloaded with the content of the watchdog timer register, when, on a write access to register CSW1, watchdog trigger registers 1 and 2 contain bit complemented values. Resetting the VCT 38xxA initializes the watchdog timer register to FFh, thus forcing the Watchdog to create a maximum reset interval.

The Watchdog is controlled by register CSW1. The first write access to it loads the timer register value setting the Watchdog's unretriggered reset interval. The desired interval can be programmed by setting the CSW1 value to:

$$\text{Value} = \frac{\text{Interval} \times f_{CPU}}{8192} - 1$$

The resolution of the Watchdog is $8192/f_{CPU}$. In CPU Slow mode (see Section 5.2.1. on page 86), the watchdog is clocked with the reduced CPU clock.

The second and all following even numbered write accesses load watchdog trigger register 1, the third and all following odd numbered write accesses load watchdog trigger register 2.

In all future, the CPU has to write alternatingly to register CSW1 value and bit complement value, thus retriggering the up-counter. Failure to retrigger will result in an overflow of the up-counter generating a Watchdog reset.

It is not allowed to change a chosen value. Writing a wrong value to CSW1 immediately sets the flag CSW1.WDRES and prohibits further retriggering of the watchdog counter.

CSW1.WDRES is true after a Watchdog reset. Only a Supply Supervision reset or a write access to register CSW1 clears it.

5.7.4. External Reset Sources

As long as the reset input comparator on the pin RESQ detects the Low level, the VCT 38xxA is in reset state. On this pin, external reset sources may be wire-ored with the internal reset sources, leading to a system-wide reset signal combining all system reset sources.

5.7.5. Summary of Module Reset States

After reset, the controller modules are set to the following reset states:

Table 5–6: Status after reset

Module	Status
CPU	CPU Fast mode.
Interrupt Controller	Interrupts are disabled. Priority registers, request flip-flops and stack are cleared.
Ports	Normal mode. Output is tristate.
Watchdog	Switched off. SW activation is possible.
Clock monitor	EMU IC: Active. SW may toggle. normal IC: Permanently active.

5.7.6. Reset Registers

1F07		RC		Reset Control Register				
bit	7	6	5	4	3	2	1	0
w	ALI	VSI	TPUI	I2CEN	DCOCLP	SELCLK	RESDIS	RESOUT
r	ALI	VSI	TPUI	I2CEN	DCOCLP	SELCLK	RESDIS	0
reset	0	0	0	0	1	0	0	0

This register controls the reset logic and clock generation.

ALI **Alarm Interrupt**
r1: Alarm was interrupt source
r0: no pending alarm interrupt
w1: reset alarm interrupt

VSI **$VSUP_D$ Voltage Supervision Interrupt**
r1: $VSUP_D$ supervision was interrupt source
r0: no pending $VSUP_D$ supervision interrupt
w1: reset $VSUP_D$ supervision interrupt

TPUI **TPU Watchdog Interrupt**
r1: TPU watchdog was interrupt source
r0: no pending TPU watchdog interrupt
w1: reset TPU interrupt flag

If the source of one of these interrupts is still active, resetting the interrupt flag will not work and no further interrupt will be generated.

I2CEN **I2C Enable**
r/w1: Enable I2C output from FE/BE.
r/w0: Disable I2C output.

DCOCLP **DCO clamping**
r/w1: DCO input clamped to 0.
r/w0: DCO input controlled by front-end.

SELCLK **Select clock source**
r/w1: From PLL.
r/w0: From DCO.

RESDIS **Reset Disable**
r/w1: Disable internal CPU reset.
r/w0: Enable internal CPU reset.

RESOUT **RESQ Output**
w1: RESQ output active.
w0: RESQ output inactive.

1F00		CSW0		Clock, Supply & Watchdog Register 0				
bit	7	6	5	4	3	2	1	0
w	x	x	x	x	x	x	x	CSA
reset	x	x	x	x	x	x	x	1

This register controls the Supply and Clock Supervision modules.

CSA **Clock and Supply Supervision Active**
w1: Both Enabled.
w0: Both Disabled.

1F60		CSW1		Clock, Supply & Watchdog Register 1				
bit	7	6	5	4	3	2	1	0
r	x	x	x	x	x	x	x	WDRES
w	Watchdog Time and Trigger Value							
reset	1	1	1	1	1	1	1	1

This register controls the Watchdog module. Only values between 1 and 255 are allowed.

WDRES **Watchdog Reset Source**
r1: Watchdog was reset source.
w: Any write access to CSW1 resets this flag.

First write the desired watchdog time value to this register. On further writes, to retrigger the Watchdog, alternatingly write a value (not necessarily the former time value) and its bit complemented value. Never change the latter value.

5.8. Memory Banking

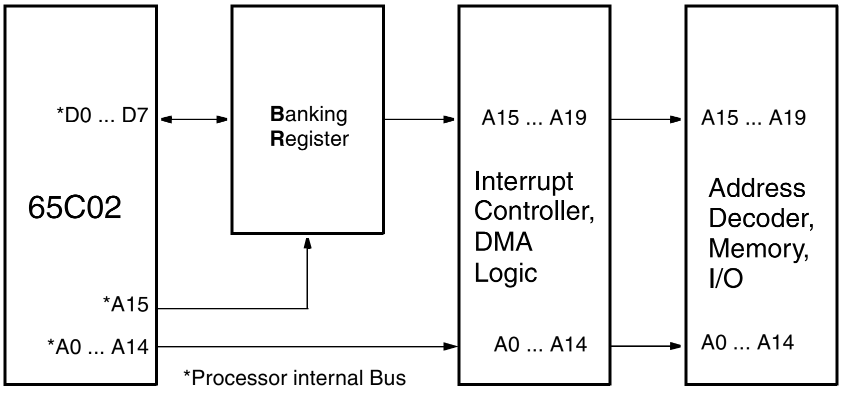


Fig. 5–5: Block diagram of Memory Banking

The 8-bit processor W65C02 only allows access to 64 kByte of memory space. To allow access to the expanded memory range above 64 kByte, a specific banking hardware is implemented. The physical address range above 32 kBytes (A15 = 1) is separated into several banks of which only one at a time is enabled and selected by the Banking register (BR), which is programmable as any other standard peripheral register by writing the desired value into its specific address. The content of the BR is also readable, so the software may check the current bank at any time. The applied software is responsible to program the BR with the correct bank number at the right time. Since the upper 32 kBytes range is switched immediately after programming the BR, correct function is not guaranteed if it is changed by a program sequence running in a switched bank. BR settings need to be done in the lower 32 kBytes (A15 = 0), which is the non-switchable master bank (bank 0).

Setting BN = 0 should be avoided because it will mirror the non-switchable master bank (bank 0) into the upper 32-kByte area (A15 = 1). RAM, I/O pages and reserved addresses may be manipulated unintentionally.

RESET initializes BN = 1 to read control byte and reset vector from bank 1. Also, interrupt vectors have to reside in bank 1, because the Interrupt Controller generates the appropriate address of bank 1, but it does not change the contents of the BR. Interrupt functions have to reside in the non-switchable master bank (bank 0). Otherwise,they need to be in each used bank, because after getting the vector the unchanged contents of the BR determine the current bank which is valid if A15 is “1”.

5.8.1. Banking Register

1F0F		BR		Banking Register				
bit	7	6	5	4	3	2	1	0
r/w				BN				
reset	0	0	0	0	0	0	0	1

BN	**Bank Number**
r/w:	number of 32 kByte memory bank

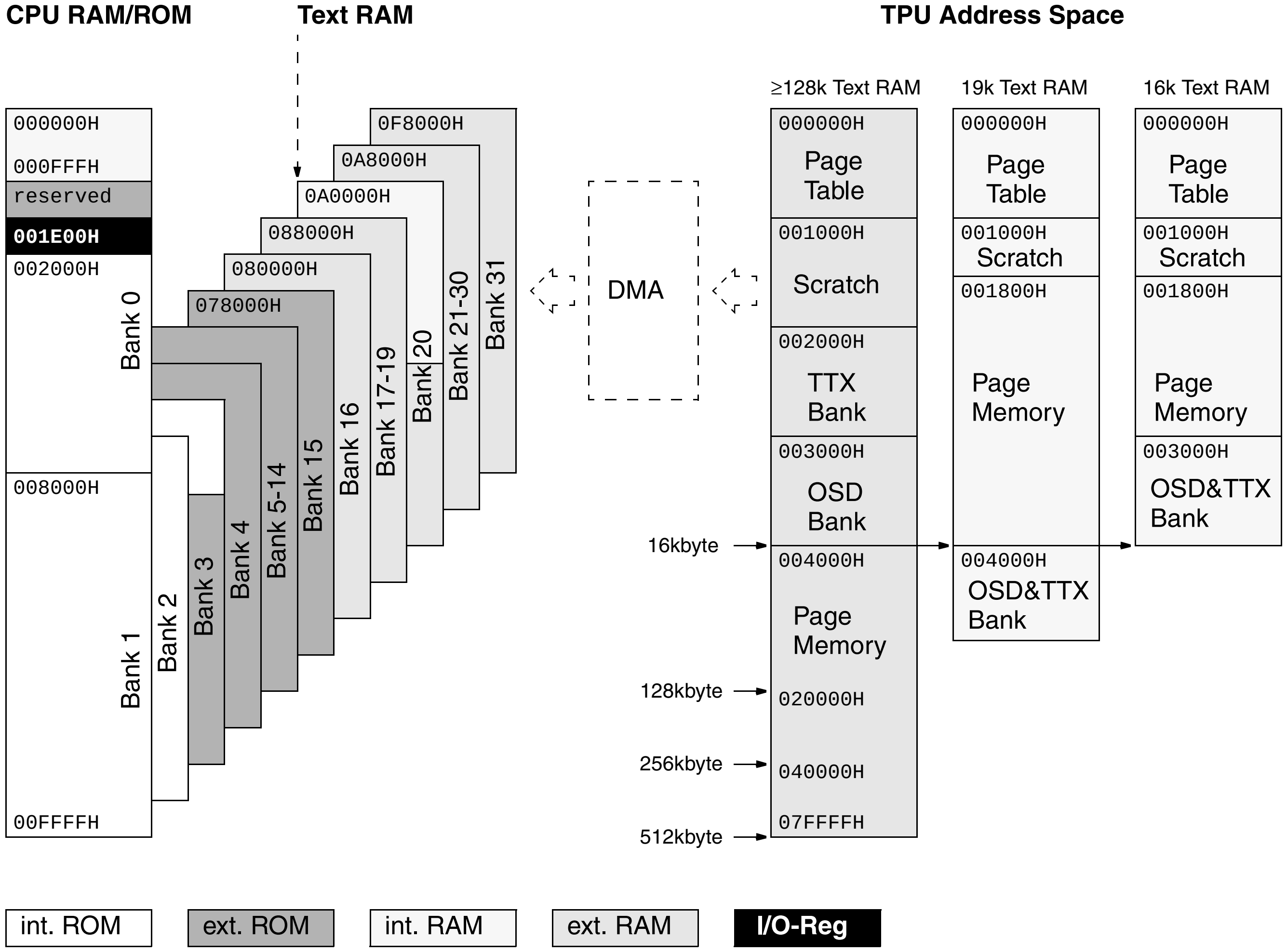


Fig. 5–6: Memory Banking shown with the maximal size of addressable memory

5.9. DMA Interface

The DMA interface connects the TPU SRAM interface to the CPU memory bus (see Fig. 5–7). This is done to avoid extra pins for external TPU page memory.

The DMA interface must not be operated during CPU Slow mode. The DMA interface can be disabled via DMAIM.DMAEN.

As long as the DMA interface is disabled, the TPU cannot access the CPU address bus and therefore should not transfer data to/from the internal/external SRAM. To ensure this, the controller should reset the TPU before disabling the DMA interface. After reset the TPU will not access the memory until receiving the I^2C command “DRAM_MODE” (see Section 3.12. on page 68).

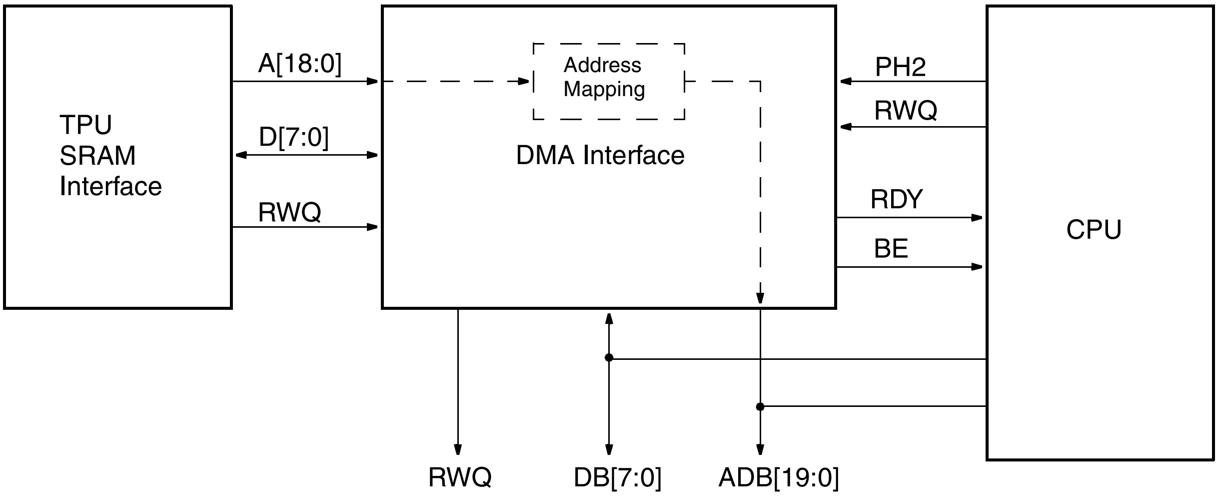


Fig. 5–7: Block diagram of DMA interface

In general, all TPU addresses are mapped into bank 16 to 31 of the CPU address space by forcing the MSB of the address bus to “1” (see Fig. 5–8). Additionally 4 memory segments can be mapped into any address area by programming a set of DMA registers (see Fig. 5–9).

Special care should be taken when mapping TPU addresses into the RAM area of bank 0. Any overlap between TPU memory (e.g. OSD Bank) and controller memory (e.g. non zero page variables) must be avoided.

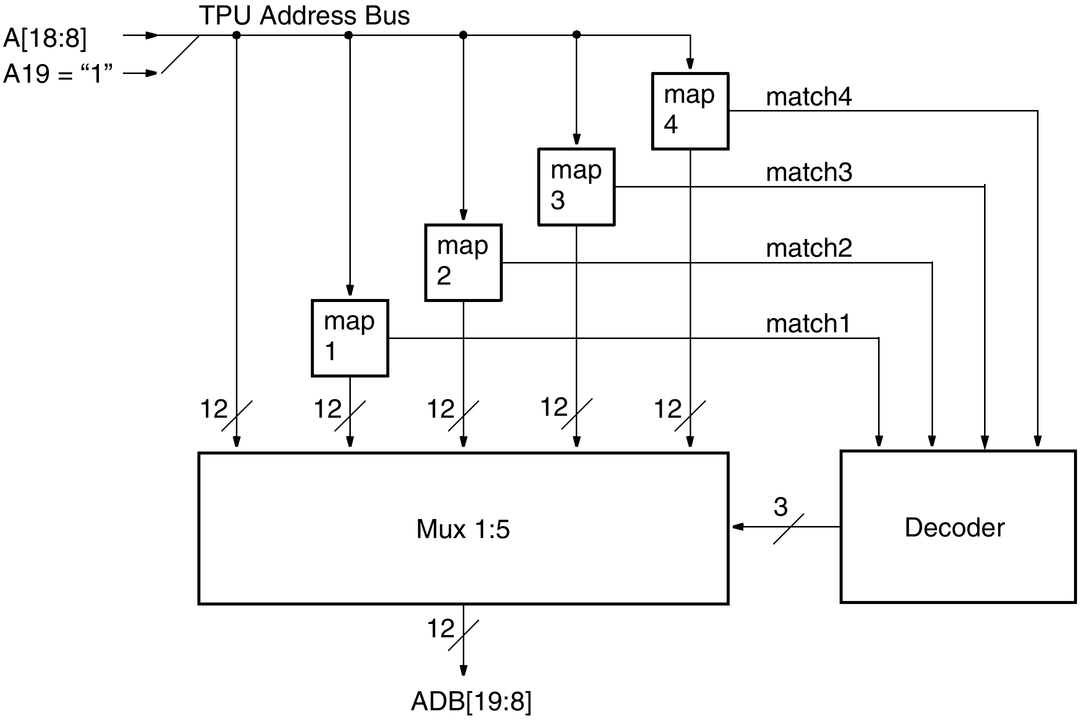
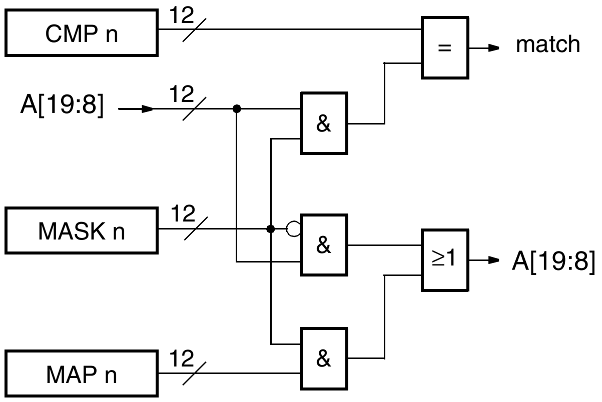


Fig. 5–8: DMA address mapping

If the mapping logic does not find any address match, the TPU address is directly put on the CPU address bus with A19 set to “1”. In case of multiple matches, the priority is map1 > map2 > map3 > map4.



n: mapping logic 1 to 4

Fig. 5–9: DMA mapping logic

5.9.1. DMA Registers

1E00	MASK1L	Mask 1 Low Byte						
1E01	MASK2L	Mask 2 Low Byte						
1E02	MASK3L	Mask 3 Low Byte						
1E03	MASK4L	Mask 4 Low Byte						
bit	7	6	5	4	3	2	1	0
w	MA15	MA14	MA13	MA12	MA11	MA10	MA9	MA8
reset	1	1	1	1	1	1	1	1

1E04	MASK1H	Mask 1 High Byte						
1E05	MASK2H	Mask 2 High Byte						
1E06	MASK3H	Mask 3 High Byte						
1E07	MASK4H	Mask 4 High Byte						
bit	7	6	5	4	3	2	1	0
w					MA19	MA18	MA17	MA16
reset					1	1	1	1

MA19 to 8 Mask Address
TPU address is masked with this value.

1E08	CMP1L	Compare 1 Low Byte						
1E09	CMP2L	Compare 2 Low Byte						
1E0A	CMP3L	Compare 3 Low Byte						
1E0B	CMP4L	Compare 4 Low Byte						
bit	7	6	5	4	3	2	1	0
w	CA15	CA14	CA13	CA12	CA11	CA10	CA9	CA8
reset	1	1	1	1	1	1	1	1

1E0C	CMP1H	Compare 1 High Byte						
1E0D	CMP2H	Compare 2 High Byte						
1E0E	CMP3H	Compare 3 High Byte						
1E0F	CMP4H	Compare 4 High Byte						
bit	7	6	5	4	3	2	1	0
w					CA19	CA18	CA17	CA16
reset					1	1	1	1

CA19 to 8 Compare Address
Masked TPU address is compared with this value.

1E10	MAP1L	Map 1 Low Byte						
1E11	MAP2L	Map 2 Low Byte						
1E12	MAP3L	Map 3 Low Byte						
1E13	MAP4L	Map 4 Low Byte						
bit	7	6	5	4	3	2	1	0
w	MPA15	MPA14	MPA13	MPA12	MPA11	MPA10	MPA9	MPA8
reset	1	1	1	1	1	1	1	1

1E14	MAP1H	Map 1 High Byte						
1E15	MAP2H	Map 2 High Byte						
1E16	MAP3H	Map 3 High Byte						
1E17	MAP4H	Map 4 High Byte						
bit	7	6	5	4	3	2	1	0
w					MPA19	MPA18	MPA17	MPA16
reset					1	1	1	1

MPA19 to 8 Map Address
Matching TPU address is replaced with this value.

1E18	DMAIM	DMA Interface Mode						
bit	7	6	5	4	3	2	1	0
w	DMAEN				MAP4E	MAP3E	MAP2E	MAP1E
reset	0				0	0	0	0

DMAEN	**DMA Enable**
w1:	Enable DMA Interface
w0:	Disable DMA Interface

MAPxE	**Mapping Logic x Enable**
w1:	Enable mapping logic x
w0:	Disable mapping logic x

5.10.Interrupt Controller

The Interrupt Controller has 16 input channels (see Fig. 5–10 on page 100). Each input has its own interrupt vector pointing to an interrupt service routine (ISR). One of 15 priority levels can be assigned to each input or the input can be disabled. The Interrupt Controller is connected to the NMI input of the CPU. But despite of the non-maskable interrupt input, it is possible to disable all interrupt sources in total in the Interrupt Controller.

5.10.1.Features

– 16 interrupt inputs.

– 16 interrupt vectors.

– 15 individual priority levels.

– Global/individual disable of interrupts.

– Single interrupt service mode.

5.10.2.General

Interrupt requests are served in the order of their programmed priority level. Interrupt requests of the same priority level are served in descending order of interrupt input number.

Each of the 16 interrupt inputs clears a flag in the interrupt pending register (IRRET and IRP), which can be read by the user. A pending interrupt enables the output of the corresponding priority register (IRPRI10 to IRPRIFE) which is connected to a parallel priority decoder together with the other priority registers. The decoder outputs the highest priority and its input number to a latch. The latched priority is compared with the top entry of the priority stack. The top entry of the priority stack contains the priority of the actual served interrupt. Lower entries contain interrupts with lower priority whose interrupt service routines were started but interrupted by the higher priority interrupts above. If the latched priority is lower or equal than the top of stack priority, nothing happens. If the latched priority is higher than the top of stack priority, a NMI is sent to the CPU and the latched priority is pushed on the stack.

The Interrupt Controller signals an interrupt by NMI input to the CPU. After the current instruction is finished the CPU starts an interrupt sequence. First it puts the program bank register, the program counter High byte, the program counter Low byte and the program status register to the stack. Then the CPU writes the vector address Low byte (00FFFAh) to the bus. The Interrupt Controller recognizes this address and stops the CPU by the RDY signal. Now the Interrupt Controller writes the vector address Low and High byte of the corresponding interrupt number to the bus and releases the CPU by releasing RDY. The CPU now operates with the new vector of the interrupt service routine.

When the Interrupt Controller writes the new vector to the address bus, the interrupt pending flag of this vector is set, indicating that no interrupt is pending.

The software must pull the top entry from the priority stack at the end of an interrupt service routine. This happens with the write access to the interrupt return register IRRET. Then the next entry (with lower priority) is visible at top of stack and is compared with the priority latch.

The Interrupt Controller and related circuitry is clocked by the CPU clock and participates in CPU Fast and Slow mode.

5.10.3.Initialization

After reset, all internal registers are cleared but the Interrupt Controller is active. When an interrupt request arrives, it will be stored in the respective pending register IRP/IRRET. But it will not trigger an interrupt as long as its interrupt priority register IRPRIxy is set to zero.

Proper SW configuration of the interrupt sources in peripheral modules has to be made prior to operation.

Before enabling individual inputs, make sure that no previously received signal on that input has cleared its pending flag which may trigger the Interrupt Controller. Clear all pending interrupts with the flag IRC.CLEAR to avoid such an effect.

5.10.4.Operation

Activation of an interrupt input is done by writing a priority value ranging from 1h to Fh to the respective IRPRIxy register. Upon an interrupt request, pending or fresh, the Interrupt Controller will immediately generate an interrupt.

During operation, changes in the priority register setting may be made to obtain varying interrupt servicing strategies. Flags IRC.DAINT, IRC.DINT and IRC.A1INT allow some variation in the Interrupt Controller response behavior.

5.10.5.Inactivation

There are two possibilities to disable an interrupt within the Interrupt Controller. Changing the priority of an interrupt input to zero disables this interrupt locally. Interrupts are globally disabled by writing a zero to flag IRC.DINT of register IRC.

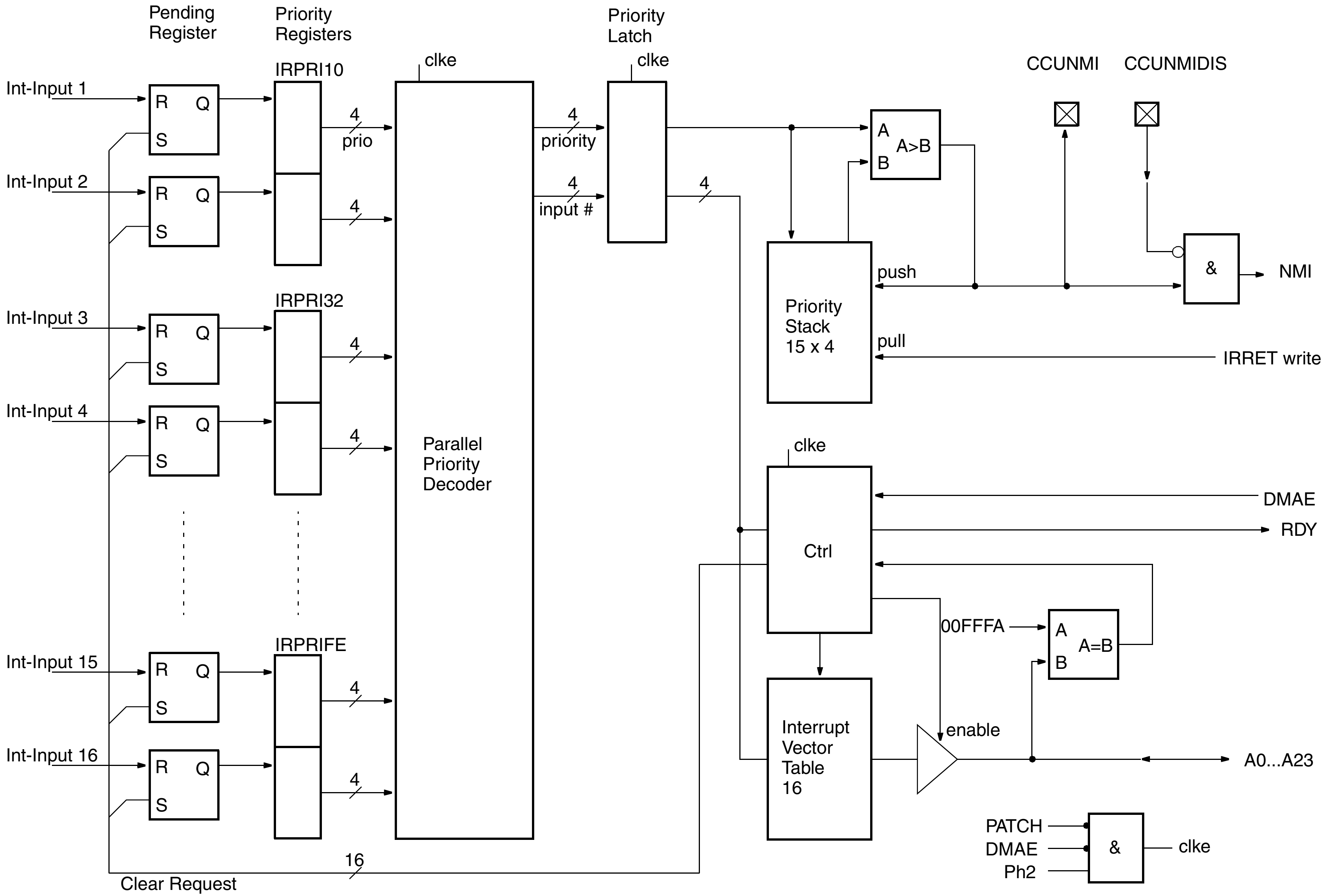


Fig. 5–10: Block diagram of interrupt logic

Within the evaluation period (see Section 5.10.10. on page 106) it's not possible to suppress an interrupt by changing priority.

A zero in the flag IRC.DINT of register IRC prevents the Interrupt Controller from pulling the signal $\overline{\text{NMI}}$ Low. However, if this flag is set after the falling edge of $\overline{\text{NMI}}$, the corresponding interrupt cannot be cancelled.

5.10.6.Precautions

The write access to the IRRET must be performed just before the RTI command at the end of the interrupt service routine. After a write access to this location it is guaranteed that the next command (should be RTI) will be processed completely before a new interrupt request is signaled to the CPU. If the RTI command does not immediately follow the write to IRRET, an interrupt with the same priority may be detected before the corresponding RTI is processed. A stack underflow may occur because this may happen several times.

If an opcode fetch of a disable interrupt instruction (DI) happens one clock cycle after the falling edge of $\overline{\text{NMI}}$ (see Section 5.10.10. on page 106), it is possible, that an interrupt service routine (ISR) is active, though the corresponding interrupt is disabled. That is why after disabling an interrupt, and before accessing critical data, at least one uncritical instruction is necessary. This guarantees that the ISR is finished before critical data access and no further ISR can interrupt it.

Because it is now possible that an ISR can lengthen the time between DI and enable interrupt (EI) indefinitely, it is necessary that an ISR first saves registers and enables interrupt flags, and then enables interrupts. After interrupt execution, enable flags and registers must be restored. This guarantees, that other interrupts are not locked out during interrupt execution.

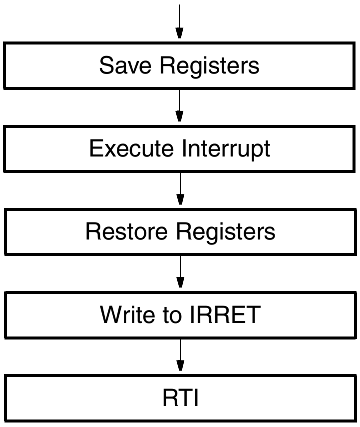


Fig. 5–11: Interrupt service routine

5.10.7.Interrupt Registers

1F20	IRC	Interrupt Control Register						
bit	7	6	5	4	3	2	1	0
r	x	x	x	x	DAINT	DINT	x	x
w	x	x	x	RESET	DAINT	DINT	A1INT	CLEAR
reset				x	1	1	x	x

RESET	**Reset**
w1:	No action.
w0:	Momentary reset of the Interrupt Controller, all internal registers are cleared.

The reset of the Interrupt Controller happens with writing zero to this flag. It is not necessary to write a one to finish the reset.

The standard interrupt controller function is performed by setting all flags to one. A hardware reset of the Interrupt Controller is performed by setting the RESET flag to Low and the other flags to High.

DAINT	**Disable after interrupt**
r1:	Don't disable after interrupt.
r0:	Disable Interrupt Controller after interrupt.
w1:	Cancel this feature.
w0:	Disable Interrupt Controller after interrupt.

This is the enable flag for the flag A1INT function.

DINT	**Disable interrupt**
r1:	Interrupts are enabled.
r0:	All interrupts are disabled.
w1:	Enable interrupts according to priority setting.
w0:	Disable all interrupts.

A1INT	**Allow one interrupt**
w1:	No action.
w0:	Serve one interrupt.

This is a momentary signal. With DAINT = 0, only one interrupt (with the highest priority) will be served.

The Flags DAINT and A1INT must be considered in common. They provide the possibility to serve interrupts one by one, only when the main program has enough time.

CLEAR	**Clear all requests**
w1:	No action.
w0:	Momentarily clears all interrupt requests.

Table 5–7: Single interrupt service

DAINT	A1INT	Resulting Function
0	1	Disable after current interrupt.
0	0	Serve one interrupt request.
1	x	Normal interrupt mode.

1F21		IRRET		Interrupt Return Register				
bit	7	6	5	4	3	2	1	0
r	IPF7	IPF6	IPF5	IPF4	IPF3	IPF2	IPF1	IPF0
w	A write access signals the Interrupt Controller that the current request has been served.							
reset	0	0	0	0	0	0	0	0

IPF0 to 7	**Interrupt Pending Flag of Input 0 to 7**
r1:	No interrupt is pending.
r0:	Interrupt is pending.
w:	Current request is finished.

For interrupt pending flags 8 to 15 refer to description of register IRP.

A write access to this memory location signals to the Interrupt Controller that the current request has been served.

1F22		IRPRI10		Interrupt Priority Register, Input 0 and 1				
bit	7	6	5	4	3	2	1	0
r/w	PRIO1				PRIO0			
reset	0	0	0	0	0	0	0	0

1F23		IRPRI32		Interrupt Priority Register, Input 2 and 3				
bit	7	6	5	4	3	2	1	0
r/w	PRIO3				PRIO2			
reset	0	0	0	0	0	0	0	0

1F24		IRPRI54		Interrupt Priority Register, Input 4 and 5				
bit	7	6	5	4	3	2	1	0
r/w	PRIO5				PRIO4			
reset	0	0	0	0	0	0	0	0

1F25		IRPRI76		Interrupt Priority Register, Input 6 and 7				
bit	7	6	5	4	3	2	1	0
r/w	PRIO7				PRIO6			
reset	0	0	0	0	0	0	0	0

1F26		IRPRI98		Interrupt Priority Register, Input 8 and 9				
bit	7	6	5	4	3	2	1	0
r/w	PRIO9				PRIO8			
reset	0	0	0	0	0	0	0	0

1F27		IRPRIBA		Interrupt Priority Register, Input 10 and 11				
bit	7	6	5	4	3	2	1	0
r/w	PRIO11				PRIO10			
reset	0	0	0	0	0	0	0	0

1F28		IRPRIDC		Interrupt Priority Register, Input 12 and 13				
bit	7	6	5	4	3	2	1	0
r/w	PRIO13				PRIO12			
reset	0	0	0	0	0	0	0	0

1F29		IRPRIFE		Interrupt Priority Register, Input 14 and 15				
bit	7	6	5	4	3	2	1	0
r/w	PRIO15				PRIO14			
reset	0	0	0	0	0	0	0	0

PRIOn	**Priority of interrupt input n**
r:	Priority of the corresponding interrupt input.
w:	Priority of the corresponding interrupt input.

Priority zero prevents the Interrupt Controller from being triggered but the pending register is not affected. All incoming requests are stored in the pending registers. With two inputs having the same PRIO setting, the higher numbered input has priority.

Table 5–8: PRIOn usage

PRIOn	Resulting Function
0h	Interrupt input is disabled
1h	Interrupt input is enabled with lowest priority
:	:
Fh	Interrupt input is enabled with highest priority

1F2A		IRP		Interrupt Pending Register				
bit	7	6	5	4	3	2	1	0
r	IPF15	IPF14	IPF13	IPF12	IPF11	IPF10	IPF9	IPF8
reset	0	0	0	0	0	0	0	0

IPF8 to 15	**Interrupt Pending Flag of Input 8 to 15**
r1:	No interrupt is pending.
r0:	Interrupt is pending.

For interrupt pending flags 0 to 7 refer to description of register IRRET.

5.10.8.Interrupt Assignment

While most interrupt assignments are hard-wired, some can be configured by software (see Fig. 5–12 on page 104).

Table 5–9: Interrupt assignment

Interrupt Input	Interrupt Vector Address	Interrupt Source
0	00FFF6–F7	I2C
1	00FFF4–F5	T0
2	00FFF2–F3	T1
3	00FFF0–F1	CCCOFL
4	00FFEE–EF	CC0OR
5	00FFEC–ED	CC0COMP
6	00FFEA–EB	CC1OR
7	00FFE8–E9	CC1COMP
8	00FFE6–E7	TVPWM
9	00FFE4–E5	VSYNC
10	00FFE2–E3	RESET
11	00FFE0–E1	CMPO
12	00FFDE–DF	PINT0
13	00FFDC–DD	PINT1
14	00FFDA–DB	PINT2
15	00FFD8–D9	PINT3

5.10.8.1. Interrupt Multiplexer

Interrupt inputs 0–11 are directly connected to the respective module’s interrupt output. Four interrupt inputs 12 to 15 allow source selection via multiplexers. The source can be any of the 15 special input ports (see Section 5.18.1. on page 126). The multiplexers are configured by registers IRPMUX0 and IRPMUX1.

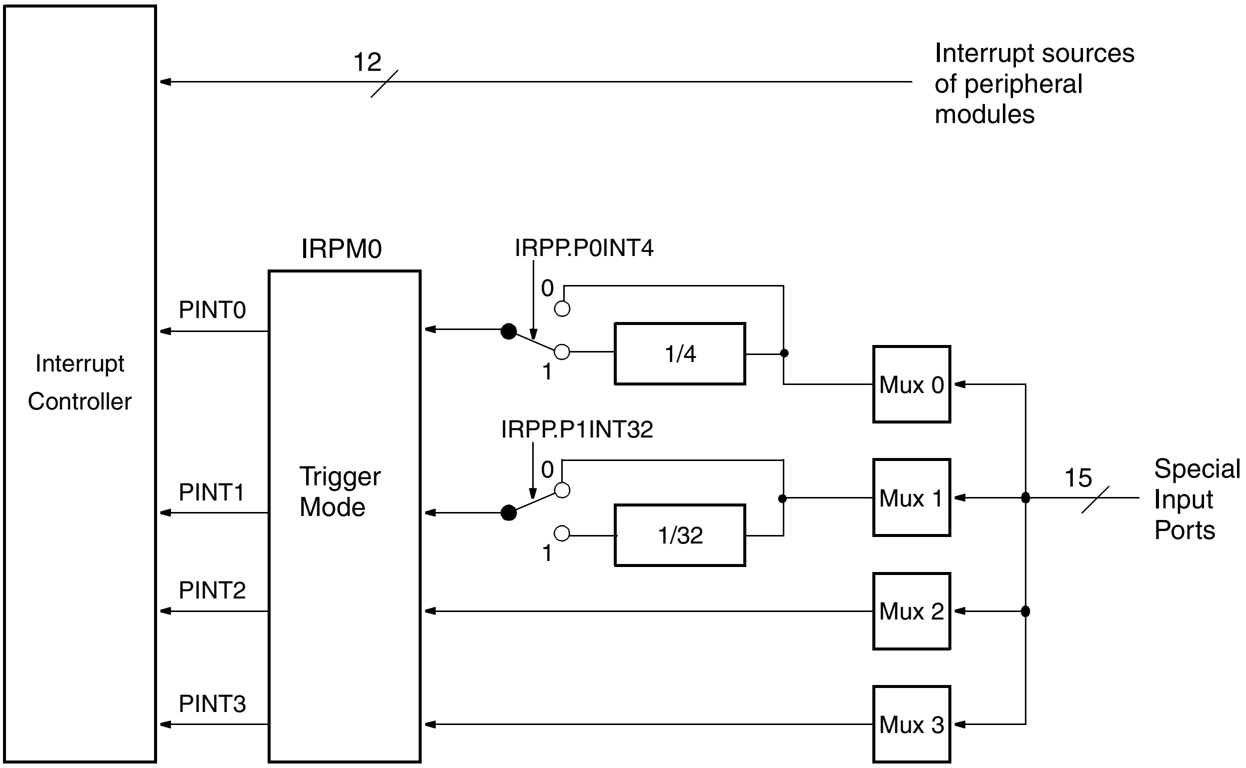


Fig. 5–12: Interrupt assignment and multiplexer

5.10.9.Port Interrupt Module

Port interrupts are the interface of the Interrupt Controller to the external world. Four port pins are connected to the module via their special input lines. Port interrupt 0 and 1 can scale down the interrupt load by prescalers. Port interrupt 2 and 3 are directly connected to the special input multiplexer.

The user can define the trigger mode for each port interrupt by the interrupt port mode register. The Port interrupt prescaler can be switched by the interrupt port prescaler register. The pulse duty factor of the prescaler output is 50 %.

The Trigger mode defines on which edge of the interrupt source signal the Interrupt Controller is triggered. The triggering of the Interrupt Controller is shown in Fig. 5–13 and Fig. 5–14 for port prescaler active (P1INT32 or P0INT4 = 1).

1F2B	IRPM0		Interrupt Port Mode					
bit	7	6	5	4	3	2	1	0
w	PIT3		PIT2		PIT1		PIT0	
reset	0	0	0	0	0	0	0	0

PITn **Port interrupt trigger n**

This field defines the trigger behavior of the associated port interrupt.

Table 5–10: PITn usage

PITn	Trigger Mode
0h	Interrupt source is disabled
1h	Rising edge
2h	Falling edge
3h	Rising and falling edges

1F2C	IRPP		Interrupt Port Prescaler					
bit	7	6	5	4	3	2	1	0
w	x	x	x	x	x	x	P1INT32	P0INT4
reset							0	0

P1INT32	**Port 1 interrupt prescaler**
w1:	Indirect mode, 1:32 prescaler
w0:	Direct mode, bypass prescaler

P0INT4	**Port 0 interrupt prescaler**
w1:	Indirect mode, 1:4 prescaler
w0:	Direct mode, bypass prescaler

1E71	IRPMUX0		Interrupt Port Multiplex 0					
bit	7	6	5	4	3	2	1	0
w	PISIP1				PISIP0			
reset	0	0	0	0	0	0	0	0

1E72	IRPMUX1		Interrupt Port Multiplex 1					
bit	7	6	5	4	3	2	1	0
w	PISIP3				PISIP2			
reset	0	0	0	0	0	0	0	0

PISIPn **Port interrupt special input port n**

This field defines the special input port connected to the associated port interrupt (see Table on page 126).

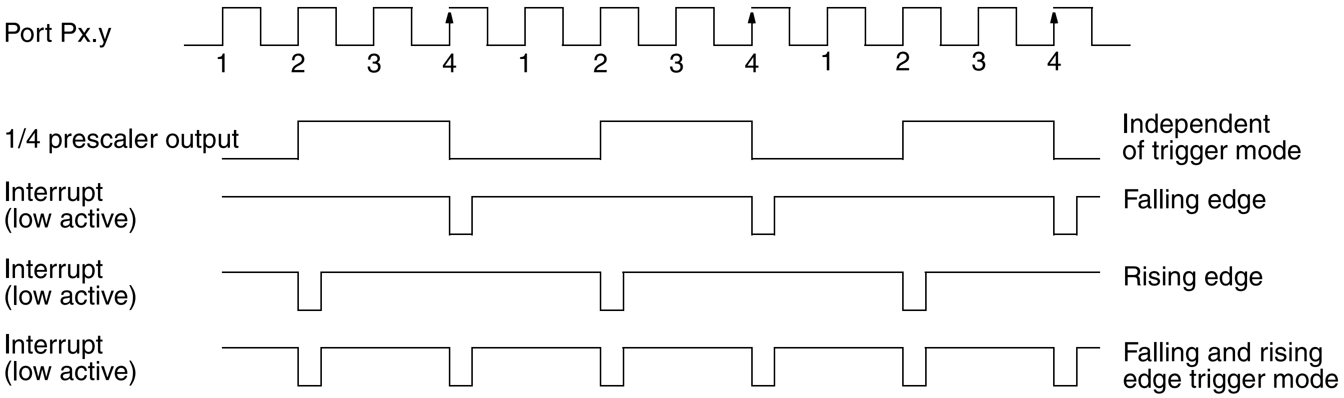


Fig. 5–13: Interrupt timing (1/4 Prescaler On)

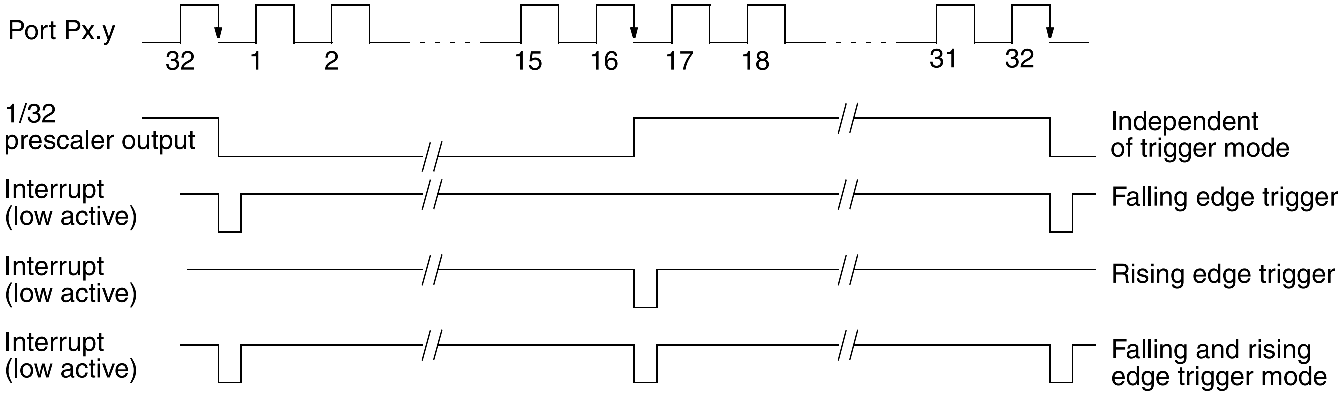


Fig. 5–14: Interrupt timing (1/32 Prescaler On)

5.10.10.Interrupt Timing

The interrupt response time is calculated from the interrupt event up to the first interrupt vector on the address bus (see Fig. 5–15 on page 106).

After an interrupt event, the Interrupt Controller starts evaluation with the first falling edge of PH2.

Evaluation needs one clock cycle until the Interrupt Controller pulls the signal $\overline{\text{NMI}}$ Low.

After the falling edge of $\overline{\text{NMI}}$ the CPU finishes the actual command. If the falling edge of $\overline{\text{NMI}}$ happens one clock cycle before an opcode fetch, the following command will be finished too. Then PC and status will be saved on stack before the Low byte of the interrupt vector is written to the address bus.

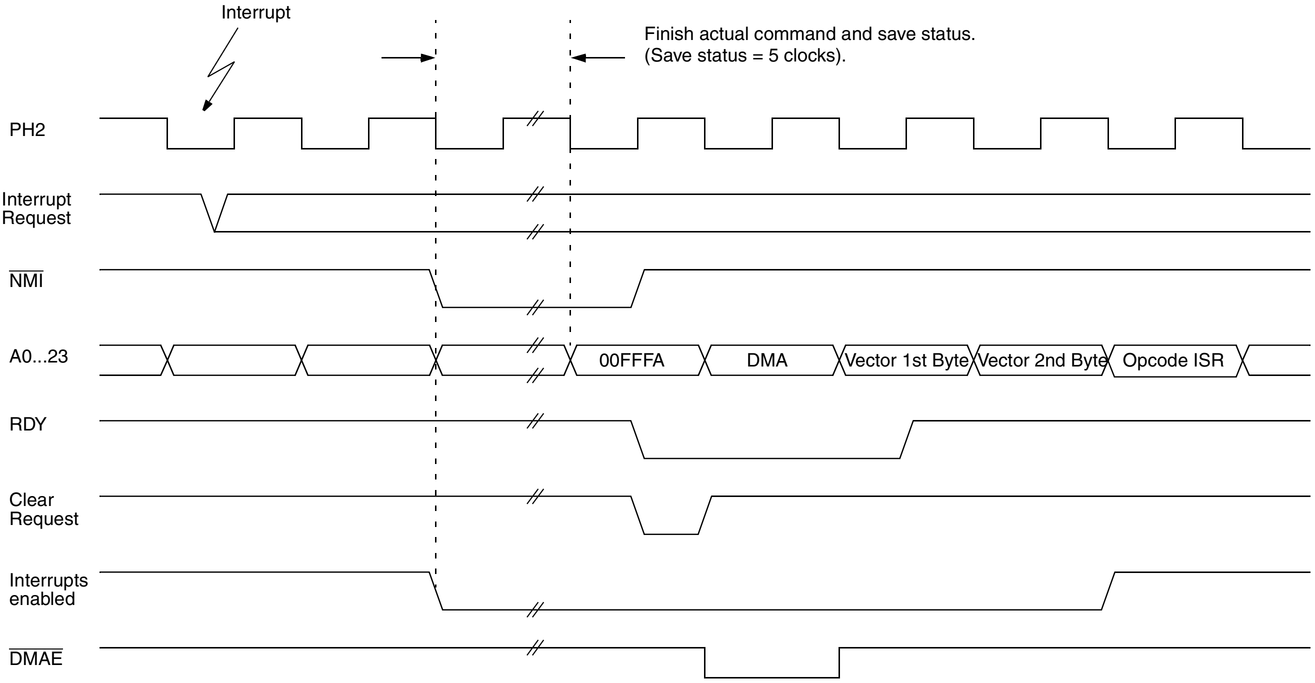


Fig. 5–15: Interrupt timing diagram

5.11.Memory Patch Module

The Memory Patch Module allows the user to modify up to ten hard-wired ROM locations by external means. This function is useful if faulty parts of software or data are detected after the ROM code has been cast into mask ROM.

Software loads addresses and the corrected code e.g. from external non-volatile memory into respective registers of the module. The module then will replace faulty code upon address match.

Single ROM locations are directly replaced. Longer faulty sequences may be repaired by introducing a jump to a new subroutine in RAM (e.g. opcode JSR requires 3 consecutive bytes to be patched). The RAM subroutine then may consist of any number of instructions, ending with a return to the next correct instruction in ROM. In such a way it is possible to include also complex software modules.

5.11.1.Features

– patching of read data from up to 10 different ROM locations (24 bit physical address)

– automatic insertion of 1 CPU wait state for each patched access

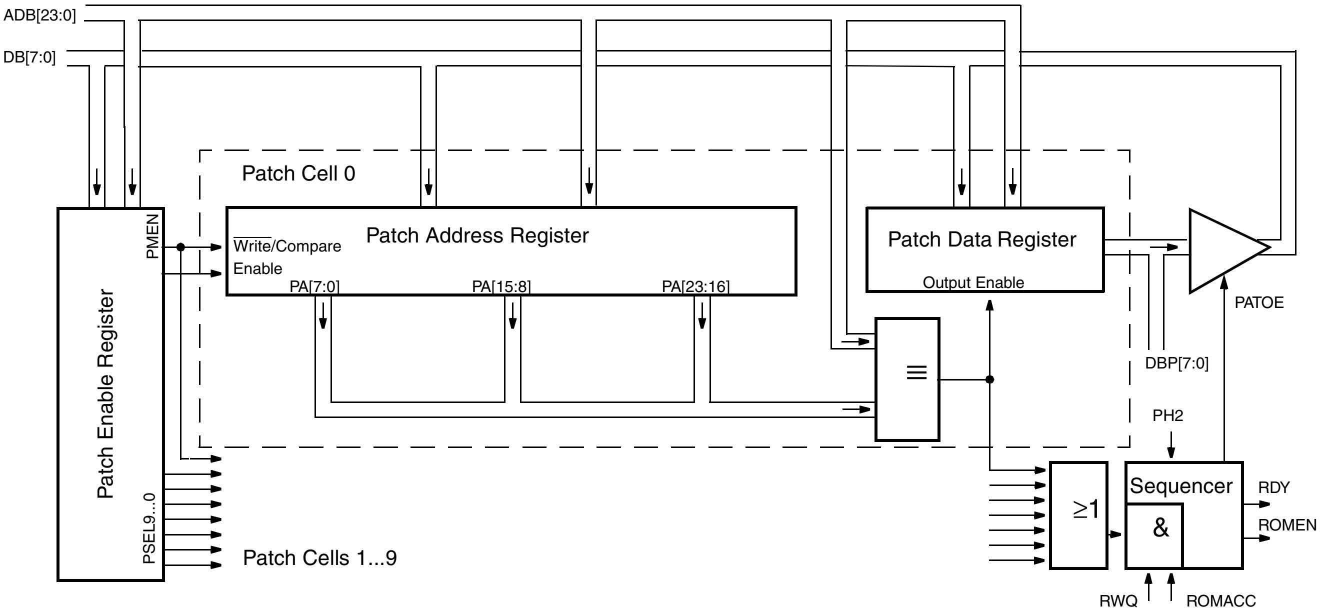


Fig. 5–16: Block diagram of patch module

5.11.2.General

The logic contains ten patch cells (see Fig. 5–16 on page 107), each consisting of a 24-bit compare register (Patch Address register, PARn), a 24-bit address comparator, a Patch Enable register (PERn) bit and an 8-bit Patch Data register (PDR).

The current address information for a ROM access is fed to a bank of ten patch cells. In case of a match in one patch cell, and provided that the corresponding Patch Enable register bit is set, a wait cycle for CPU is included by pulling down the RDY input of CPU for one cycle (see Fig. on page 108). In the meantime the module's logic disables the ROM data bus drivers and instead places the data information from the corresponding Patch Data register on the data bus.

5.11.3.Initialization

After reset, as bit PER0.PMEN is reset to 0, all patch cell registers are in Write mode and patch operation is disabled.

To initialize a patch cell, first set the corresponding PSEL bit in register PER0 or PER1 as a pointer. Then enter the 24bit address to registers PAR2 (High byte), PAR1 (middle byte) and PAR0 (Low byte) and the desired patch code to register PDR.

If desired, repeat the above sequence for other patch cells. Only set one PSEL pointer bit in registers PER0 and PER1 at a time.

5.11.4. Patch Operation

To activate a number of properly initialized patch cells for ROM code patching, set all the corresponding PSEL bits in registers PER1, then PER0, setting bit PER0.PMEN to 1.

The Memory Patch Module will immediately start comparing the current address to the setting of the enabled patch cells. In case of a match, the ROM data will be replaced by the corresponding patch cell data register setting.

To reconfigure the Memory Patch Module, first set PER0.PMEN to 0. The module will immediately terminate patch operation.

.

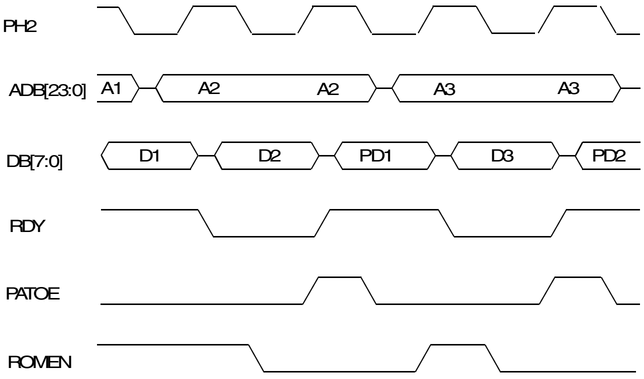


Fig. 5–17: Patch timing

5.11.5. Patch Registers

1E64	PAR0	Patch Address Register 0						
bit	7	6	5	4	3	2	1	0
w	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
reset	1	1	1	1	1	1	1	1

1E65	PAR1	Patch Address Register 1						
bit	7	6	5	4	3	2	1	0
w	PA15	PA14	PA13	PA12	PA11	PA10	PA9	PA8
reset	1	1	1	1	1	1	1	1

1E66	PAR2	Patch Address Register 2						
bit	7	6	5	4	3	2	1	0
w	PA23	PA22	PA21	PA20	PA19	PA18	PA17	PA16
reset	1	1	1	1	1	1	1	1

1E67	PDR	Patch Data Register						
bit	7	6	5	4	3	2	1	0
w	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
reset	0	0	0	0	0	0	0	0

1E68	PER0	Patch Enable Register 0						
bit	7	6	5	4	3	2	1	0
w	PSEL6	PSEL5	PSEL4	PSEL3	PSEL2	PSEL1	PSEL0	PMEN
reset	0	0	0	0	0	0	0	0

1E69	PER1	Patch Enable Register 1						
bit	7	6	5	4	3	2	1	0
w	x	x	x	x	x	PSEL9	PSEL8	PSEL7
reset	x	x	x	x	x	0	0	0

PA23 to 0 Patch Address
Upon occurrence of this address the patch cell replaces ROM data with data from PDR.

PD7 to 0 Patch Data
Data to replace false ROM data at certain address.

PSEL0 to 9 Select Patch Cell
w1: select cell for write or enable for patch
w0: disable patch cell

Before writing compare address or replace data of a patch cell, only one cell must be selected. In compare mode one or more patch cells can be selected.

PMEN Patch Mode Enable
w1: enable patch mode of all cells
w0: enable write mode of all cells

5.12. I^{2}C-Bus Master Interface

The I^{2}C bus interface is a pure Master system, Multimaster busses are not realizable. The clock and data terminal pins have open-drain outputs.

The I^{2}C bus master interface can operate on two terminals. Terminal 1 is connected to the pins SDA/SCL, terminal 2 can be connected either to the pins P36/P37 or to the pins P22/P23. Please refer to chapter 5.18. on page 126 how to set up the corresponding port pins.

The I^{2}C bus master interface is not affected by CPU Slow mode.

The bit rate is programmable using a clock prescaler.

A complete telegram is assembled by the software out of individual sections. Each section contains an 8-bit data. This data is written into one of the six possible Write registers. Depending on the chosen address, a certain part of an I^{2}C bus cycle is generated. By means of corresponding calling sequences it is therefore possible to join even very long telegrams (e.g. long data files for auto increment addressing of I^{2}C slaves).

The software interface contains a 5 word deep Write-FIFO for the control data registers, as well, as a 3 word deep Read FIFO for the received data. Thus most of the I^{2}C telegrams can be transmitted to the hardware without the software having to wait for empty space in the FIFO.

An interrupt is generated on two conditions:

- The Write-FIFO was filled and reaches the ‘half full’ state.
- The Write-FIFO is empty and stop condition is completed.

All address and data fields appearing on the bus are constantly monitored and written into the Read-FIFO. The software can then check these data in comparison with the scheduled data. If a read instruction is handled, the interface must set the data word FFH, so that the responding slave can insert its data. In this case the Read-FIFO contains the read-in data.

If telegrams longer than 3 bytes (1 address, 2 data bytes) are received, the software must check the filling condition of the Write-FIFO and, if necessary, fill it up (or read out the Read-FIFO). A variety of status flags is available for this purpose:

- The ‘half full’ flag I2CRS.WFH is set if the Write-FIFO is filled with three bytes.
- The ‘empty’ flag I2CRS.RFE is set if there is no more data available in the Read-FIFO.
- The ‘busy’ flag I2CRS.BUSY is activated by writing any byte to any one of the Write registers. It stays active until the I^{2}C bus activities are stopped after the stop condition generation.

Moreover, the ACK-bit is recorded separately on the bus lines for the address and the data fields. However, the interface itself can set the address ACK=0. In any case the two ACK flags show the actual bus condition. These flags remain until the next I^{2}C start condition is generated.

For example, the software has to work off the following sequence (ACK=1) to read a 16-bit word from an I^{2}C device address 10H (on condition that the bus is not active):

– write 021H to	I2CWS0
– write 0FFH to	I2CWD0
– write 0FFH to	I2CWP0
– read RFE bit from	I2CRS
– read dev. address from	I2CRD
– read RFE bit from	I2CRS
– read 1st data byte from	I2CRD
– read RFE bit from	I2CRS
– read 2nd data byte from	I2CRD

The value 21H in the first step results from the device address in the 7 MSBs and the R/W-bit (read=1) in the LSB. If the telegrams are longer, the software has to ensure that neither the Write-FIFO nor the Read-FIFO can overflow.

- To write data to this device:

– write 20H to	I2CWS0
– write 1st databyte to	I2CWD0
– write 2nd databyte to	I2CWP0

The bus activity starts immediately after the first write to the Write-FIFO. The transmission can be synchronized by an artificial extension of the Low phase of the clock line. Transmission is not continued until the state of the clock line is High once again. Thus, an I^{2}C slave device can adjust the transmission rate to its own abilities.

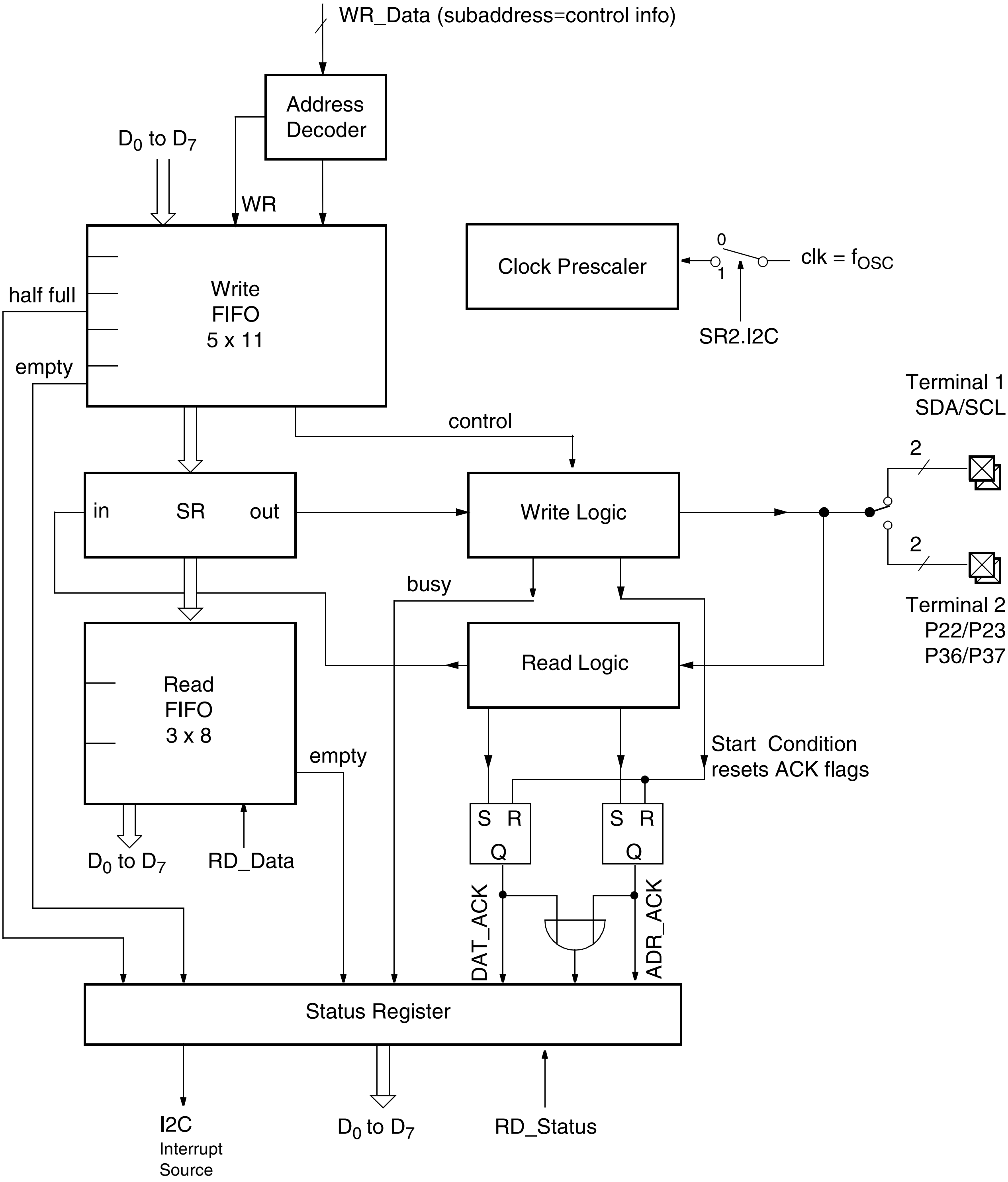


Fig. 5–18: Block diagram of I^{2}C bus master interface

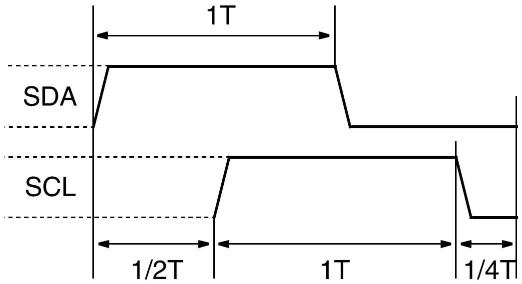


Fig. 5–19: Start condition I^{2}C bus

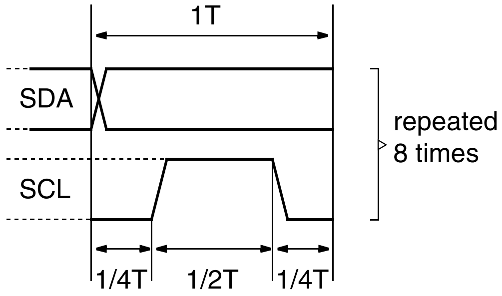


Fig. 5–20: Single bit on I^{2}C bus

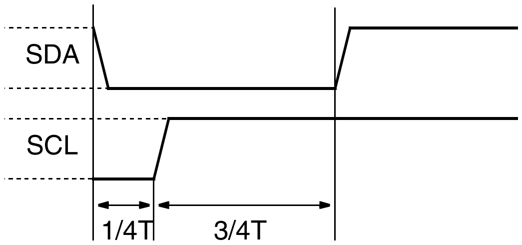


Fig. 5–21: Stop condition I^{2}C bus

5.12.1. I^{2}C Bus Master Interface Registers

1FD0	I2CWS0	I2C Write Start Register 0						
bit	7	6	5	4	3	2	1	0
w	I2C Address							
reset	0	0	0	0	0	0	0	0

Writing this register moves I2C start condition, I2C Address and ACK=1 into the Write FIFO.

1FD1	I2CWS1	I2C Write Start Register 1						
bit	7	6	5	4	3	2	1	0
w	I2C Address							
reset	0	0	0	0	0	0	0	0

Writing this register moves I2C start condition, I2C Address and ACK=0 into the Write FIFO.

1FD2	I2CWD0	I2C Write Data Register 0						
bit	7	6	5	4	3	2	1	0
w	I2C Data							
reset	0	0	0	0	0	0	0	0

Writing this register moves I2C Data and ACK=1 into the Write FIFO.

1FD3	I2CWD1	I2C Write Data Register 1						
bit	7	6	5	4	3	2	1	0
w	I2C Data							
reset	0	0	0	0	0	0	0	0

Writing this register moves I2C Data and ACK=0 into the Write FIFO.

1FD4	I2CWP0	I2C Write Stop Register 0						
bit	7	6	5	4	3	2	1	0
w	I2C Data							
reset	0	0	0	0	0	0	0	0

Writing this register moves I2C Data, ACK=1 and I2C stop condition into the Write FIFO.

1FD5	I2CWP1	I2C Write Stop Register 1						
bit	7	6	5	4	3	2	1	0
w	I2C Data							
reset	0	0	0	0	0	0	0	0

Writing this register moves I2C Data, ACK=0 and I2C stop condition into the Write FIFO.

1FD6		I2CRD		I2C Read Data Register				
bit	7	6	5	4	3	2	1	0
r	I2C Data							
reset	0	0	0	0	0	0	0	0

Reading this register returns the content of the Read FIFO.

1FD7		I2CRS		I2C Read Status Register				
bit	7	6	5	4	3	2	1	0
r	x	OACK	AACK	DACK	BUSY	WFH	RFE	x
reset	0	0	0	0	0	0	0	0

OACK	**“OR”ed Acknowledge**
r:	AACK || DACK

AACK	**Address Acknowledge**
r:	Acknowledge state of address field

DACK	**Data Acknowledge**
r:	Acknowledge state of data field

BUSY	**Busy**
r1:	I^{2}C Master Interface is busy
r0:	I^{2}C Master Interface is not busy

WFH	**Write-FIFO Half Full**
r1:	Write-FIFO is filled with 3 Bytes
r0:	Write-FIFO is not half full

RFE	**Read-FIFO Empty**
r1:	Read-FIFO is empty
r0:	Read-FIFO is not empty

1FDB		I2CM		I2C Mode Register				
bit	7	6	5	4	3	2	1	0
w	TERM	SPEED						
reset	1	0	0	0	0	0	1	0

TERM	**Terminal Select**
w1:	Terminal 1
w0:	Terminal 2

SPEED	**Speed Select**
w:	I^{2}C Bit Rate = f_{OSC} / (4 * SPEED)

1E73		I2CPS		I2C Port Select Register				
bit	7	6	5	4	3	2	1	0
w								SIPS
reset	0	0	0	0	0	0	0	0

Table 5–11: I^{2}C Bit Rates

SPEED	Bit Rate
0	19.776 Kbit/s
1	2.531 Mbit/s
2	1.266 Mbit/s
3	844 Kbit/s
4	633 Kbit/s
...	...
127	19.931 Kbit/s

SIPS	**Special Input Port Select**
w1:	use port pair P36, P37 for terminal 2
w0:	use port pair P22, P23 for terminal 2

5.13.Timer T0 and T1

Timer T0 and T1 are 16-bit auto reload down counters. They serve to deliver a timing reference signal, to output a frequency signal or to produce time stamps.

5.13.1.Features

- 16-bit auto reload counter
- Time value readable
- Interrupt source output
- Frequency output

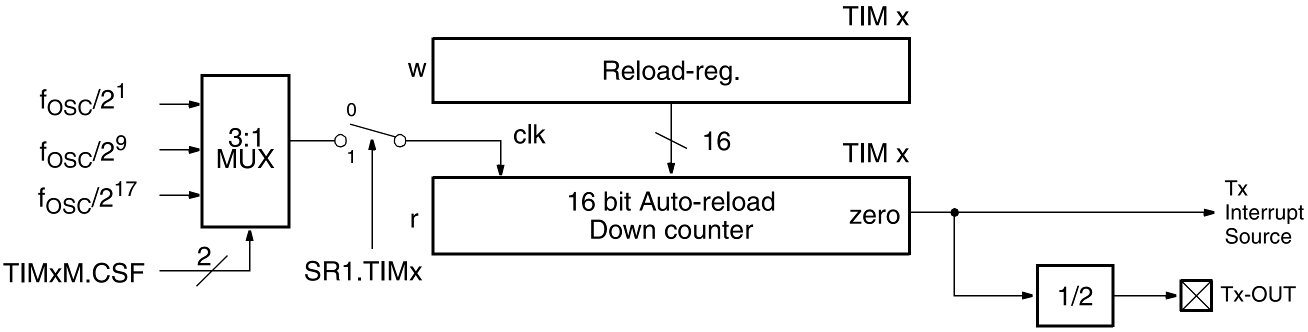


Fig. 5–22: Block diagram of timer T0 and T1

5.13.2.Operation

The timer's 16-bit down-counter is clocked by the input clock and counts down to zero. Reaching zero, it generates an output pulse, reloads with the content of the TIMx reload register and restarts its travel.

T0 and T1 are not affected by CPU Slow mode.

The clock input frequency can be selected from three possible values by programming the timer mode register TIMxM.CSF. After reset, both timers are in standby mode (inactive).

Prior to entering active mode, proper SW initialization of the Ports assigned to function as Tx-OUT outputs has to be made. The ports have to be configured Special Out (see Section 5.18. on page 126).

To initialize a timer, Reload register TIMx has to set to the desired time value, still in standby mode. For entering active mode, set the corresponding enable bit in the Standby register. The timer will immediately start counting down from the time value present in register TIMx.

During active mode, a new time value is loaded by writing to the 16-bit register TIMx, High byte first. Upon writing the Low byte, the reload register is set to the new 16-bit value, the counter is reset, and immediately starts down-counting with the new value.

On reaching zero, the counter generates a reload signal, which can be used to trigger an interrupt. The same signal is connected to a divide by two scaler to generate the output signal Tx-OUT with a pulse duty factor of 50 %.

The interrupt source output of this module is routed to the Interrupt Controller logic (see Section 5.10. on page 99).

The state of the down-counter is readable by reading the 16-bit register TIMx, Low byte first. Upon reading the Low byte, the High byte is saved to a temporary latch, which is then accessed during the subsequent High byte read.

Thus, for time stamp applications, read consistency between Low and High byte is guaranteed.

Returning a timer to standby mode by resetting the corresponding Enable bit will halt its counter and will set its output to Low. The register TIMx remains unchanged.

5.13.3.Timer Registers

1F4E		**TIM0L**		**Timer 0 Low Byte**				
1F4C		**TIM1L**		**Timer 1 Low Byte**				
bit	7	6	5	4	3	2	1	0
r	**Read Low Byte of down-counter and latch High Byte.**							
w	**Write Low Byte of reload value and reload down-counter.**							
reset	1	1	1	1	1	1	1	1

1F4F		**TIM0H**		**Timer 0 High Byte**				
1F4D		**TIM1H**		**Timer 1 High Byte**				
bit	7	6	5	4	3	2	1	0
r	**Read latched High Byte of down-counter.**							
w	**Write High Byte of reload value.**							
reset	1	1	1	1	1	1	1	1

TIMx have to be read Low byte first and written High byte first.

1F11		**TIM0M**		**Timer 0 Mode**				
1F13		**TIM1M**		**Timer 1 Mode**				
bit	7	6	5	4	3	2	1	0
r/w							**CSF**	
reset	0	0	0	0	0	0	0	0

CSF	**Clock Selection Field**
r/w:	Source of timer clock (see Table 5–12)

Table 5–12: CSF usage

CSF	Clock Divider	Timer Clock	Timer Increment	Timer Period
00	$f_{OSC}/2^1$	5.0625 MHz	197.53 ns	12.945 ms
01	$f_{OSC}/2^9$	19.775 KHz	50.568 µs	3.3140 s
1x	$f_{OSC}/2^{17}$	77.248 Hz	12.945 ms	848.39 s

5.14.Capture Compare Module (CAPCOM)

The Capture Compare Module (CAPCOM) is a complex relative timer. It comprises a free running 16-bit Capture Compare Counter (CCC) and 2 Capture Compare Subunits (SU). The CCC provides an interrupt on overflow and the timervalue can be read by software.

A SU is able to capture the relative time of an external event input and to generate an output signal when the CCC passes a predefined timer value. Three types of interrupts enable interaction with SW. Special functionality provides an interface to the asynchronous external world.

5.14.1.Features

- 16-bit free running counter with read out.
- 16-bit capture register.
- 16-bit compare register.
- Input trigger on rising, falling or both edges.
- Output action: toggle, Low or High level.
- Three different interrupt sources: overflow, input, compare
- Designed for interfacing to asynchronous external events

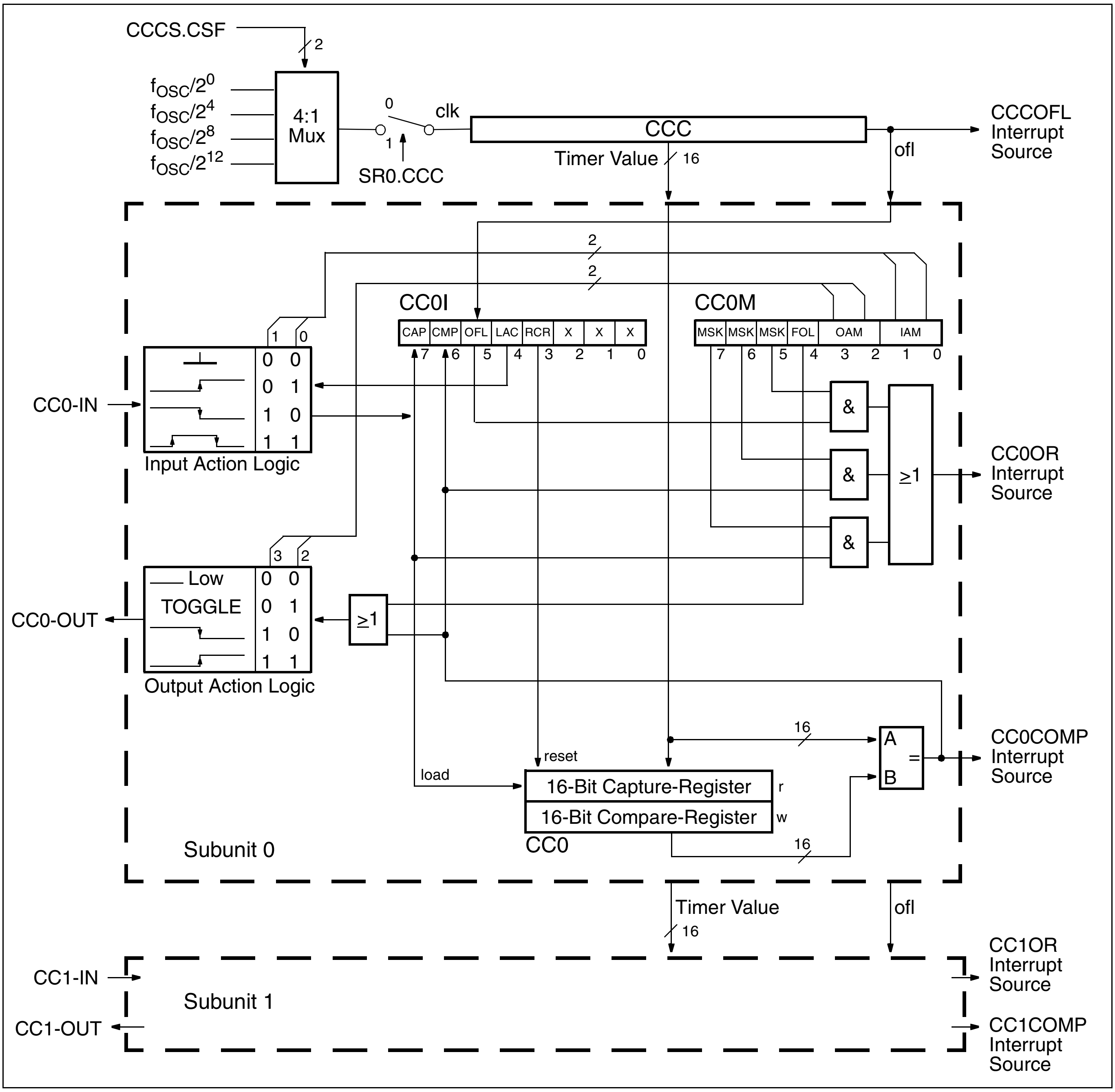


Fig. 5–23: Block diagram of CAPCOM module

5.14.2.Initialization

After system reset the CCC and all SUs are in standby mode (inactive).

In standby mode, the CCC is reset to value 0000h. Capture and compare registers CCx are reset. No information processing will take place, e.g. update of interrupt flags. However, the values of registers CCxI and CCxM are only reset by system reset, not by standby mode. Thus, it is possible to program all mode bits in standby mode and a predetermined start-up out of standby mode is guaranteed.

Prior to entering active mode, proper SW configuration of the Ports assigned to function as Input Capture inputs and Output Action outputs has to be made. The Output Action ports have to be configured as Special Out and the Input Capture ports as special in (see Section 5.18. on page 126).

Please note, that the compare register CCx is reset in standby mode. It can only be programmed in active mode.

5.14.3.Operation of CCC

For entering active mode of the entire CAPCOM module set, the enable bit in the standby register.

The CCC will immediately start up-counting with the selected clock frequency and will deliver this 16-bit value to the SUs.

The state of the counter is readable by reading the 16-bit register CCC, Low byte first. Upon reading the Low byte, the High byte is saved to a temporary latch, which is then accessed during the subsequent High byte read. Thus, for time stamp applications, read consistency between Low and High byte is guaranteed.

The CCC is free running and will overflow from time to time. This will cause generation of an overflow interrupt event. The interrupt (CCCOFL) is directly fed to the Interrupt Controller and also to all SUs where further processing takes place.

5.14.3.1. Operation of Subunit

For a proper setup the SW has to program the following SU control bits in registers CCxI and CCxM: Interrupt Mask (MSK), Force Output Logic (FOL, 0 recommended), Output Action mode (OAM), Input Action mode (IAM), Reset Capture register (RCR, 0 recommended), and Lock After Capture (LAC). Refer to section 5.14.5. for details.

Each SU is able to capture the CCC value at a point of time given by an external input event processed by an Input Action Logic.

A SU can also change an output line level via an Output Action Logic at a point of time given by the CCC value.

Thus, a SU contains a 16-bit capture register CCx to store the input event CCC value, a 16-bit compare register CCx to program the Output Action CCC value, an 8-bit interrupt register CCxI and an 8-bit mode register CCxM. Two types of interrupts per SU enable interaction with SW.

For limitations on operating the CAPCOM module in CPU Slow mode, see section 5.14.3.1.15. on page 117.

5.14.3.1.13. Compare and Output Action

To activate a SUs compare logic the respective 16-bit compare register CCx has to be programmed, Low byte first. The compare action will be locked until the High byte write is completed. As soon as CCx setting and CCC value match, the following actions are triggered:

- The flag CMP in the CCxI register is set.
- The CCxCOMP interrupt source is triggered.
- The CCxOR interrupt source is triggered when activated.
- The Output Action logic is triggered.
  Four different reactions are selectable for the Output Action signal: according to field CCxM.OAM (Table 5–17) the equal state will lead to a High or Low level, or toggling or inactivity on this output.
  Another means to control the Output Action is bit CCxM.FOL. E.g. rise-mode and force will set the output pin to High level, fall-mode and force to Low level. This forcing is static, i.e. it will be permanently active and may override compare events. Thus, it is recommended to set and reset shortly after that, i.e. to pulse the bit with SW. Toggle mode of the Output Action logic and forcing leads to a burst with clock-frequency and is not recommended.

5.14.3.1.14. Capture and Input Action

The Input Action logic operates independently of the Output Action logic and is triggered by an external input in a way defined by field CCxM.IAM. Following Table 5–18 it can completely ignore events, trigger on rising or falling edge or on both edges. When triggered, the following actions take place:

- Flag CCxI.CAP is set.
- The CCxOR interrupt source is triggered when activated.
- The 16-bit capture register CCx stores the current CCC value, i.e. the “time” of the external event. Read CCx Low byte first. Further compare action will be locked until the subsequent High byte read is completed. Thus a coherent result is ensured, no matter how much time has elapsed between the two reads.

Some applications suffer from fast input bursts and a lot of capture events and interrupts in consequence. If the SW cannot handle such a rate of interrupts, this could evoke stack overflow and system crash. To prevent such fatal situations the Lock After Capture (LAC) mode is implemented. If bit CCxI.LAC is set, only one capture event will pass. After this event has triggered a capture, the Input Action logic will lock until it is unlocked again by writing an arbitrary value to register CCxM. Make sure that this write only restores the desired setting of this register.

Programming the Input Action logic while an input transition occurs may result in an unexpected triggering. This may overwrite the capture register, lock the Input Action logic if in LAC mode and generate an interrupt. Make sure that SW is prepared to handle such a situation.

For testing purposes, a permanent reset (FFFFh) may be forced on capture register CCx by setting bit CCxI.RCR. Make sure that the reset is only temporary.

5.14.3.1.15. Interrupts

Each SU supplies two internal interrupt events:

1. Input Capture event and
2. Comparator equal state.

As previously explained, interrupt events will set the corresponding flags in register CCxI. In addition to the above mentioned two, the CCC Overflow interrupt event sets flag CCxI.OFL in each SU. Thus, three interrupt events are available in each SU. The corresponding flags are masked with their mask bits in register CCxM and passed to a logical or. The result (CCxOR) is fed to the Interrupt Controller as a first interrupt source. In addition, the Comparator equal (CCxCOMP) interrupt is directly passed to the Interrupt Controller as second interrupt source. Thus a SU offers four types of interrupts: CCC overflow (maskable ored), input capture event (maskable ored) and comparator equal state (maskable ored and non-maskable direct).

All interrupt sources act independently, parallel interrupts are possible. The interrupt flags enable SW to determine the interrupt source and to take the appropriate action. Before returning from the interrupt routine the corresponding interrupt flag should thus be cleared by writing a 1 to the corresponding bit location in register CCxI.

The interrupts generated by internal logic (CCC Overflow and Comparator equal) will trigger in a predetermined and known way. But as explained in 5.14.3.1.14. erroneous input signals may cause some difficulties concerning the Input Capture input, as well, as interrupt handling. To overcome possible problems the Input Capture Interrupt flag CCxI.CAP is double buffered. If a second or even more input capture interrupt events occur before the interrupt flag is cleared (i.e. SW was not able to keep track), the flag goes to a third state. Two consecutive writes to this bit in register CCxI are then necessary to clear the flag. This enables SW to detect such a multiple interrupt situation and eventually to discard the capture register value which always relates to the latest input capture event and interrupt.

The internal CAPCOM module control logic always runs on the oscillator frequency, regardless of CPU Slow mode. Avoid write accesses to the CCxI register in CPU Slow mode, since the logic would interpret one CPU access as many consecutive accesses. This may yield unexpected results concerning the functionality of the interrupt flags. The following procedure should be followed to handle the capture interrupt flag CAP:

1. SW responds to a CAPCOM interrupt, switching to CPU Fast mode if necessary and determining that the source is a capture interrupt (CAP flag =1).
2. The interrupt service routine is processed.
3. Just before returning to main program, the service routine acknowledges the interrupt by writing a 1 to flag CAP.
4. The service routine reads CAP again. If it is reset, the routine can return to main program as usual. If it is still set an external capture event overrun has happened. Appropriate actions may be taken (i.e. discarding the capture register value etc.).
5. go to 3.

5.14.4.Inactivation

The CAPCOM module is inactivated and returned to standby mode (power down mode) by setting the Enable bit to 0. Section 5.14.2. applies. CCxI and CCxM are only reset by system reset, not by standby mode.

5.14.5. CAPCOM Registers

1F7C	CCCL		CAPCOM Counter Low Byte					
bit	7	6	5	4	3	2	1	0
r	Read Low Byte and lock CCC.							
reset	0	0	0	0	0	0	0	0

1F7D	CCCH		CAPCOM Counter High Byte					
bit	7	6	5	4	3	2	1	0
r	Read High Byte and unlock CCC.							
reset	0	0	0	0	0	0	0	0

The CAPCOM module counter has to be read Low byte first to avoid inconsistencies.

1F14	CCCS		CAPCOM Clock Select					
bit	7	6	5	4	3	2	1	0
w							CSF	
reset	0	0	0	0	0	0	0	0

CSF **Clock Selection Field**
w: Source of CCC clock (see Table 5–16)

Table 5–16: CSF usage

CSF	Clock Divider	Timer Clock	Timer Increment	Timer Period
00	$f_{OSC}/2^0$	10.125 MHz	98.765 ns	6.4727 ms
01	$f_{OSC}/2^4$	632.81 KHz	1.5802 µs	103.56 ms
10	$f_{OSC}/2^8$	39.551 KHz	25.284 µs	1.6570 s
11	$f_{OSC}/2^{12}$	2.4719 KHz	404.54 µs	26.512 s

1F6C	CC0M		CAPCOM 0 Mode Register					
1F70	CC1M		CAPCOM 1 Mode Register					
bit	7	6	5	4	3	2	1	0
r	MSK	MSK	MSK	FOL	OAM		IAM	
reset	0	0	0	0	0	0	0	0

MSK **Mask Flag**
w1: Enable.
w0: Disable.

These mask flags refer to the corresponding event flags in CAPCOM interrupt register.

FOL **Force Output Action Logic**
r/w1: Force Output Action logic.
r/w0: Release Output Action logic.

This flag is static. As long as FOL is true neither comparator can trigger nor SW can force, by writing another “one”, the Output Action logic. After forcing it is recommended to clear FOL unless Output Action logic should not be locked.

OAM **Output Action Mode**
r/w: Defines behavior of Output Action logic.

IAM **Input Action Mode**
r/w: Defines behavior of Input Action logic.

Table 5–17: OAM usage

OAM	Output Action Logic Modes
0 0	Disabled, ignore trigger, output Low level.
0 1	Toggle output.
1 0	Output Low level.
1 1	Output High level.

Table 5–18: IAM usage

IAM	Input Action Logic Modes
0 0	Disabled, don’t trigger.
0 1	Trigger on rising edge.
1 0	Trigger on falling edge.
1 1	Trigger on rising and falling edge.

1F6D	CC0I	CAPCOM 0 Interrupt Register						
1F71	CC1I	CAPCOM 1 Interrupt Register						
bit	7	6	5	4	3	2	1	0
r/w	CAP	CMP	OFL	LAC	RCR	x	x	x
reset	0	0	0	0	0	0	0	0

CAP	**Capture Event**
r1:	Event.
r0:	No Event.
w1:	Clear flag.

CMP	**Compare Event**
r1:	Event.
r0:	No Event.
w1:	Clear flag.

OVL	**Overflow Event**
r1:	Event.
r0:	No Event.
w1:	Clear flag.

LAC	**Lock After Capture**
r/w1:	Enable.
r/w0:	Disable.

RCR	**Reset Capture Register**
r/w1:	Reset capture register to FFFFh.
r/w0:	Release capture register.

1F6E	CC0L	CAPCOM 0 Capture/Compare Low Byte						
1F72	CC1L	CAPCOM 1 Capture/Compare Low Byte						
bit	7	6	5	4	3	2	1	0
r	Read Low Byte of capture register and lock it.							
w	Write Low Byte of compare register and lock it.							
reset	1	1	1	1	1	1	1	1

1F6F	CC0H	CAPCOM 0 Capture/Compare High Byte						
1F73	CC1H	CAPCOM 1 Capture/Compare High Byte						
bit	7	6	5	4	3	2	1	0
r	Read High Byte of capture register and unlock it.							
w	Write High Byte of compare register and unlock it.							
reset	1	1	1	1	1	1	1	1

1E70	CCIMUX	CAPCOM Input Multiplex Register						
bit	7	6	5	4	3	2	1	0
w	CCSIP1				CCSIP0			
reset	0	0	0	0	0	0	0	0

CCSIPn **CAPCOM Special Input Port n**

This field defines the special input port connected to the associated SU (see Table on page 126).

5.15. Pulse Width Modulator

Each of the 4 available PWMs is an 8-bit reload down-counter with fixed reload interval. It serves to generate a frequency signal with variable pulse width or, with an external low-pass filter, as a digital to analog converter.

5.15.1. Features

- 8-bit resolution
- standby mode

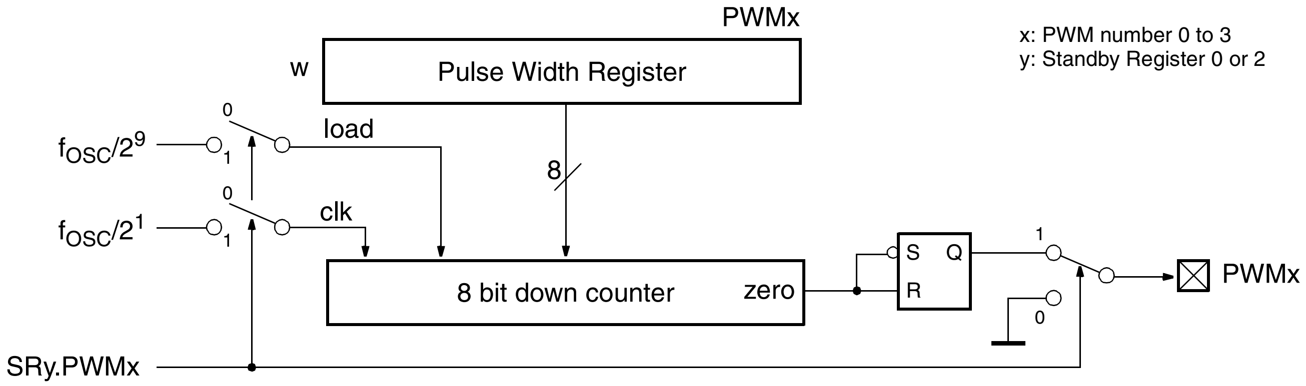


Fig. 5–1: Block diagram of 8-bit PWM

5.15.2. General

A PWM's 8-bit down-counter is clocked by its input clock and counts down to zero. Reaching zero, it stops and sets the output to Low. A load pulse reloads the counter with the content of the PWM register, restarts it and sets the output to High. The repetition rate is 19.775 KHz, the reload period is 50.57 µs.

The PWMs are not affected by CPU Slow mode. It is recommended that the CPU should not write the PWM registers during Slow mode.

5.15.3. Initialization

Prior to entering active mode, proper SW initialization of the Ports assigned to function as PWMx outputs has to be made. The ports have to be configured Special Out (see Section 5.18. on page 126).

5.15.4. Operation

After reset, all PWMs are in standby mode (inactive) and the output signal PWMx is Low.

For entering active mode, the enable bit in the corresponding standby register has to be set (see Section 5.5. on page 89). The desired pulse width value is then written into register PWMx. Each PWM will start producing its output signal immediately after the next subsequent load pulse.

During active mode, a new pulse width value is set by simply writing to the register PWMx. Upon the next subsequent load pulse the PWM will start producing an output signal with the new pulse width value, starting with a High level.

Returning a PWM to standby mode by resetting its respective enable flag will immediately set its output Low.

The state of the down-counters is not readable.

5.15.5. PWM Registers

1F50	**PWM0**	**PWM 0 Register**						
1F51	**PWM1**	**PWM 1 Register**						
1F52	**PWM2**	**PWM 2 Register**						
1F53	**PWM3**	**PWM 3 Register**						
bit	7	6	5	4	3	2	1	0
w	**Pulse width value**							
reset	0	0	0	0	0	0	0	0

Table 5–19: Pulse Width Programming

Pulse width value	Pulse duty factor
00h	0% (Output is static Low)
01h	1/256
02h	2/256
:	:
FEh	254/256
FFh	100% (Output is static High) 1)

1) Pulse duty factor 255/256 is not selectable.

5.16.Tuning Voltage Pulse Width Modulator

The Tuning Voltage Pulse Width Modulator (TVPWM), in combination with an external low pass filter, serves as a digital to analog converter to control voltage synthesis tuning. It can also be operated as a normal 8-bit PWM.

5.16.1.Features

- 14bit resolution
- standby mode

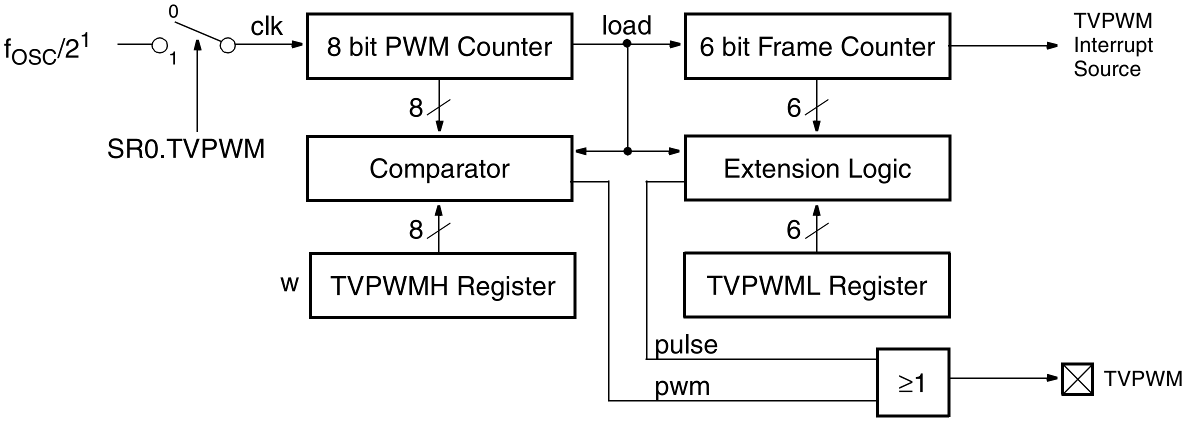


Fig. 5–2: Block Diagram of 14bit Tuning Voltage PWM

5.16.2.General

The TVPWM is based on an 8-bit PWM built by a counter and a programmable comparator (see Fig. 5–2). The overflow of the counter reloads the comparator with the content of the TVPWMH register and sets the TVPWM output to High. Matching the counter value, the comparator sets the TVPWM output to Low. The counter is continually running, producing PWM cycles with a length of 256 T.

Depending on the content of the TVPWML register, the 6-bit pulse extension logic will add additional single clock pulses distributed over a frame of 64 reload cycles (see Fig. 5–3). This gives 14-bit resolution when integrating over a complete frame. The frame rate is 309 Hz, the frame period is 3.24 ms.

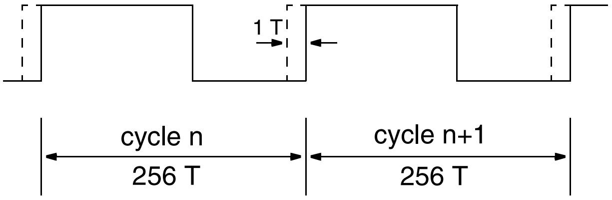


Fig. 5–3: TVPWM Timing

An interrupt is generated after completion of a frame of 64 reload cycles. The interrupt source output of this module is routed to the Interrupt Controller logic (see Section 5.10. on page 99).

The TVPWM is not affected by CPU Slow mode. It is recommended that the CPU should not write the TVPWM registers during Slow mode.

5.16.3. Initialization

Prior to entering active mode, proper SW initialization of the Ports assigned to function as TVPWM output has to be made. The ports have to be configured Special Out (see Section 5.18. on page 126).

5.16.4. Operation

After reset, the TVPWM is in standby mode (inactive) and the output signal TVPWM is Low.

For entering active mode, the enable bit in the corresponding standby register has to be set (see Section 5.5. on page 89). The desired pulse width value is then written into the registers TVPWML and TVPWMH. The TVPWM will start producing its output signal immediately after the next subsequent load pulse.

During active mode, a new pulse width value is set by simply writing to the register TVPWML and TVPWMH. Writing TVPWMH will update the comparator and the extension logic with the new register values. Upon the next subsequent load pulse the TVPWM will start producing an output signal with the new pulse width value, starting with a High level.

Returning the TVPWM to standby mode by resetting its respective enable flag will not reset its output signal.

The state of the counters and the extension logic is not readable.

5.16.5. TVPWM Registers

1F4A	TVPWML		TV PWM Low Byte					
bit	7	6	5	4	3	2	1	0
w			**Pulse width value Low**					
reset			0	0	0	0	0	0

1F4B	TVPWMH		TV PWM High Byte					
bit	7	6	5	4	3	2	1	0
w	**Pulse width value High**							
reset	0	0	0	0	0	0	0	0

TVPWM has to be written Low byte first.

5.17. A/D Converter (ADC)

This 10-bit analog to digital converter allows the conversion of an analog voltage in the range of 0 to U_{Ref} into a digital value. A multiplexer connects the ADC to one of 15 analog input ports. A sample-and-hold circuit holds the analog voltage during conversion. The duration of the sampling time is programmable. The A/D conversion is done by a charge balance A/D converter using successive approximation.

5.17.1. Features

- A/D converter with 10-bit resolution.
- Successive approximation, charge balance type.
- Input multiplexer with 15 analog channels.
- Sample and hold circuit.
- 4/8/16/32 µs conversion selectable for optimum throughput/accuracy balance.
- Zero standby current, 300 µA active current.

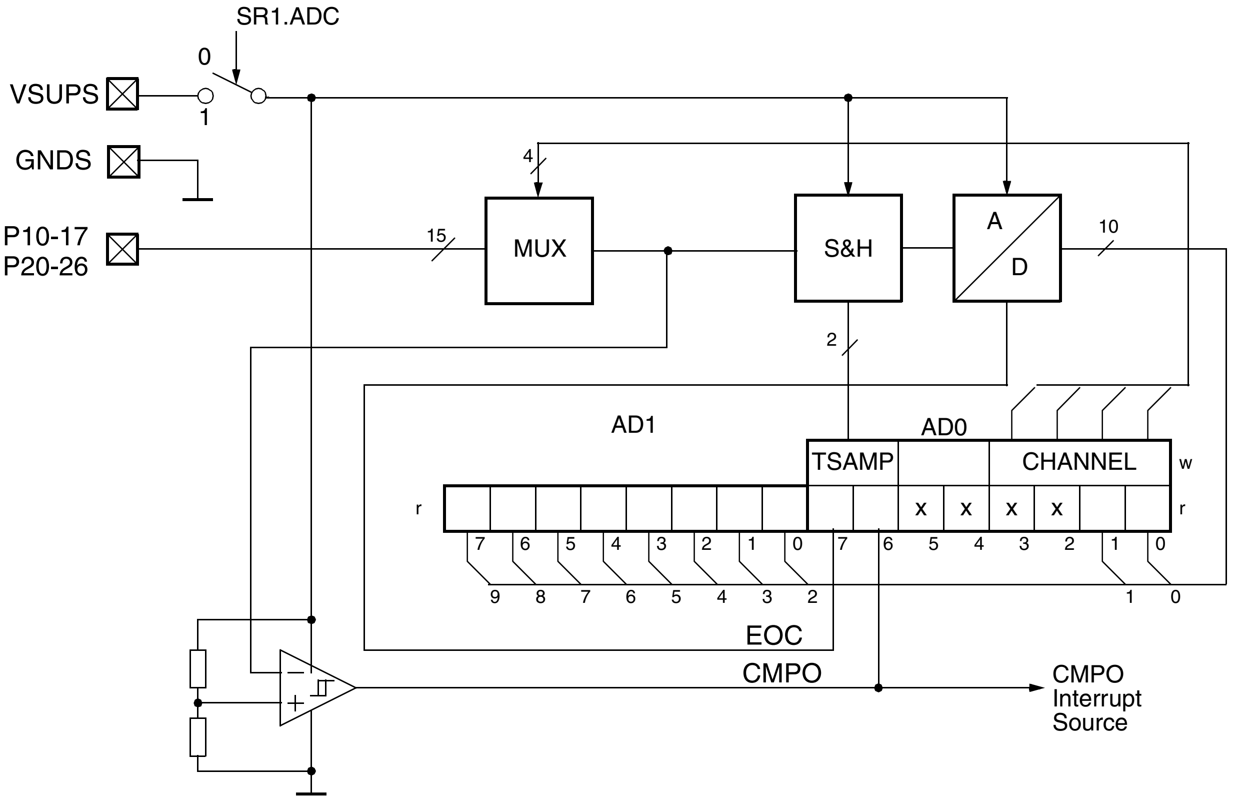


Fig. 5–4: Block Diagram of the ADC

5.17.2. Operation

After reset, the module is off (zero standby current). The module is enabled by the flag SR1.ADC. The user must ensure that the flag End of Conversion (EOC) in register AD0 is true, before he starts to operate the module.

A write access to register AD0 indicating sample time and channel number starts the conversion. The flag EOC signalizes the end of conversion. The 10-bit result is stored in the registers AD1 (8 MSB) and AD0. The conversion rate depends on the software, the oscillator frequency and the programmed sample time.

The ADC module is not affected by CPU Slow mode.

5.17.3.Measurement Errors

The result of the conversion mirrors the voltage potential of the sampling capacitance (typically 15 pF) at the end of the sampling time. This capacitance has to be charged by the source through the source impedance within the sampling time period. To avoid measurement errors, system design has to make sure that at the end of the sampling period, the potential error on the sampling capacitance is less than ±0.1 LSB.

Measurement errors may occur, when the voltage of high-impedance sources has to be measured:

- To reduce these errors, the sampling time may be increased by programming the field TSAMP in register AD1.
- In cases where high-impedance sources are only rarely sampled, a 100-nF capacitor from the input to GNDS is a sufficient measure to ensure that the potential on the sampling capacitance reaches the full source potential, even with the shortest sampling time.
- In some high-impedance applications a charge pumping effect may influence the measurement result when two sources are measured alternatingly.

5.17.4.Comparator

In addition to the A/D converter the module contains a comparator. The level at the A/D converter input is compared to VSUPS/2. The state of the comparator output can be read at flag CMPO in register AD0.

The interrupt source output of this module is routed to the Interrupt Controller logic. The CMPO interrupt source is gated with an internal clock. This is the reason why interrupts are generated as long as the level at the comparator is lower than the internal reference.

5.17.5.ADC Registers

A write access to register AD0 starts the A/D conversion of the written channel number and sampling duration. The flag EOC signals the end of conversion. The result is stored in register AD1 (bit 9 to 2) and in register AD0 (bit 1 and 0).

1FA8		AD0		ADC Register 0				
bit	7	6	5	4	3	2	1	0
r	EOC	CMPO	x	x	x	x	AN1	AN0
w	TSAMP				CHANNEL			
reset	0	0	0	0	0	0	0	0

1FA9		AD1		ADC Register 1				
bit	7	6	5	4	3	2	1	0
r	AN9	AN8	AN7	AN6	AN5	AN4	AN3	AN2
reset								

EOC **End of Conversion**
r1: End of conversion
r0: Busy

EOC is reset by a write access to the register AD0. EOC must be true before starting the first conversion after enabling the module by setting SR1.ADC.

CMPO **Comparator Output**
r1: Input is lower than reference voltage.
r0: Input is higher than reference voltage.

TSAMP **Sampling Time**

TSAMP adjusts the sample time and the conversion time. The total conversion time is 20 clock cycles longer than the sample time. Sampling starts one clock cycle after completion of the write access to AD0.

Table 5–20: Sampling time adjustment

TSAMP	t_{Sample}	$t_{Conversion}$
0H	20 T_{OSC}	40 T_{OSC}
1H	60 T_{OSC}	80 T_{OSC}
2H	140 T_{OSC}	160 T_{OSC}
3H	300 T_{OSC}	320 T_{OSC}

CHANNEL **Channel of Input Multiplexer**

CHANNEL selects from which port pin the conversion is done. The MSB of CHANNEL is bit 3. No port pin is connected to the ADC if the channel 0 is selected. In this case the input of the A/D converter is connected to ground. After reset, CHANNEL is set to zero.

Table 5–21: ADC input multiplexer

CHANNEL	Port Pin
0	none
1	P10
2	P11
3	P12
4	P13
5	P14
6	P15
7	P16
8	P17
9	P20
10	P21
11	P22
12	P23
13	P24
14	P25
15	P26

AN 9 to 0 **Analog Value Bit 9 to 0**

The 10 bit analog value is in the range of 0 to 1023. The 8 MSB can be read from register AD1. The two LSB can be read from register AD0. The result is available until a new conversion is started.

5.18.Ports

There exist different kinds of ports. The universal ports, P1 to P3, serve as digital I/O and have additional special input and output functions. A subset of the universal ports (P10-P17, P20-P26) serves as input for the analog-to-digital converter. The I^2C ports SDA, SCL can alternatively be used as digital I/O ports. The analog audio ports AIN1–3, AOUT1–2 can alternatively be used as digital input ports. The 20.25 MHz system clock output CLK20 can alternatively be used as digital output port.

5.18.1.Port Assignment

Table 5–23 shows the assignment of port pins to Special Input and Output functions.

Every Special Output function is connected to 2 port pins in parallel and can be activated via the MOD flag in the corresponding port register.

The ADC input multiplexer can be connected to 1 of 15 port pins. The output driver of the selected port pin is then forced to open-drain mode. Additionally it can be disabled using the EN flag in the corresponding port register.

Every special input function can be connected to 1 of 15 input ports (see Table 5–22). If port number 0 is selected the special input function is connected to ground. Changing the input port may produce temporary glitch signals. Therefore, the corresponding special input function should be disabled before the input port is changed.

Table 5–22: Special input configuration

Special Input Number	Special Input Function	Special Input Port
1	CC0–IN	0–15
2	CC1–IN	0–15
3	PINT0	0–15
4	PINT1	0–15
5	PINT2	0–15
6	PINT3	0–15

Table 5–23: Port pin configuration

Port Name	ADC Input	Special Output	Special Input Port
P10	1	Timer 0	1
P11	2	Timer 1	2
P12	3	CC0–OUT	3
P13	4	CC1–OUT	4
P14	5	TVPWM	5
P15	6	PWM 0	6
P16	7	PWM 1	7
P17	8	PWM 2	8
P20	9	PWM 3	9
P21	10	CLK20	10
P22	11	SDA 2	11
P23	12	SCL 2	12
P24	13	Timer 0	
P25	14	Timer 1	
P26	15	CC0–OUT	
P27		CC1–OUT	
P30		TVPWM	
P31		PWM 0	
P32		PWM 1	
P33		PWM 2	
P34		PWM 3	
P35		CLK20	
P36		SDA 2	
P37		SCL 2	
P40		SDA	
P41		SCL	
P42		AOUT1	
P43		AOUT2	13
P44		AIN1	
P45		AIN2	14
P46		AIN3	15

5.18.2.Universal Ports P1 to P3

There are 24 universal port pins. The universal ports P1 to P3 are each 8 bits wide. In the 64-pin PSDIP package only 12 universal port pins are available (P10–P17, P20–P23).

5.18.2.1. Features

- digital I/O port
- special input and output function
- analog input function
- Schmitt trigger input buffer
- tristate output
- push-pull or open-drain output
- 10-mA output current
- output supply either 3.3 V or 5.0 V

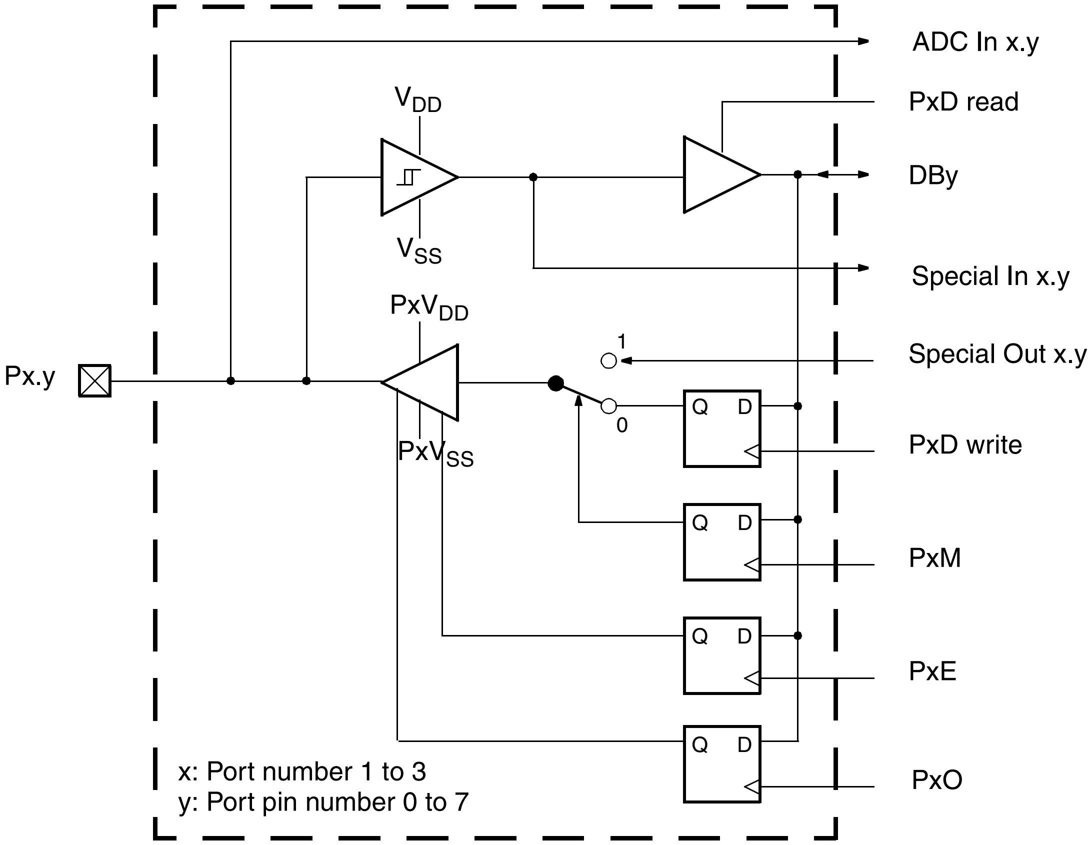


Fig. 5–5: Universal port circuit

Universal ports can be operated in different modes:

Table 5–24: Universal ports operating modes

Modes		Function
Port Mode	Normal Input	The SW uses the ports as digital input.
	Special Input	The port input is additionally connected to specific hardware modules.
	Normal Output	The SW uses the ports as latched digital tristate output.
	Special Output	The port output is directly driven by specific hardware modules.

After reset, all Universal Ports are in normal mode, tristate condition.

5.18.2.2. Universal Port Mode

Each port bit can be individually configured to several port modes. The output driver of each pin has to be enabled by setting the EN flag. Using the OUT flag the output stage can be configured to either open drain or push pull output. The MOD flag selects the source of the output value.

Table 5–25: Port mode register settings

Mode	MOD	EN	D	Function
Normal Input	x	0	x	READ of register PxD returns port pin input levels to data bus.
Normal Output	0	1	Data	WRITE to register PxD changes level of port pin output drivers. READ of register PxD returns the PxD register setting to the data bus.
Special Input	x	x	x	Port pin input level is presented to special hardware.
Special Output	1	1	x	Special hardware drives port pin. READ of register PxD returns port pin input levels to data bus.

The Special Input mode is always active. This allows manipulating the input signal to the special hardware through Normal Output operations by software.

As the Special Output mode allows reading the pin levels, the output state of the special hardware may be read by the CPU.

5.18.3.Universal Port Registers

Universal Port Data registers PxD contain input/output data of the corresponding port. The “x” in PxD means the number of the port. Thus PxD stands for P1D to P3D.

	1F90	P1D	Port 1 Data Register					
	1F94	P2D	Port 2 Data Register					
	1F98	P3D	Port 3 Data Register					
bit	7	6	5	4	3	2	1	0
r/w	D7	D6	D5	D4	D3	D2	D1	D0
reset	0	0	0	0	0	0	0	0

D0–7 **Universal Port Data Input/Output**
r: Read pin level resp. data latch.
w: Write data to data latch.

To use a port pin as software output, the appropriate driver must be activated by setting the EN flag and the MOD flag must be programmed to Normal mode.

	1F91	P1O	Port 1 Output Register					
	1F95	P2O	Port 2 Output Register					
	1F99	P3O	Port 3 Output Register					
bit	7	6	5	4	3	2	1	0
w	OUT7	OUT6	OUT5	OUT4	OUT3	OUT2	OUT1	OUT0
reset	0	0	0	0	0	0	0	0

OUT0–7 **Output Flag**
w1: Output driver is open drain
w0: Output driver is push pull

	1F92	P1M	Port 1 Mode Register					
	1F96	P2M	Port 2 Mode Register					
	1F9A	P3M	Port 3 Mode Register					
bit	7	6	5	4	3	2	1	0
w	MOD7	MOD6	MOD5	MOD4	MOD3	MOD2	MOD1	MOD0
reset	0	0	0	0	0	0	0	0

MOD0–7 **Normal/Special Mode Flag**
w1: Special Output Mode
w0: Normal Output Mode

The MOD flag defines from which source the pin is driven if the EN flag is true.

	1F93	P1E	Port 1 Enable Register					
	1F97	P2E	Port 2 Enable Register					
	1F9B	P3E	Port 3 Enable Register					
bit	7	6	5	4	3	2	1	0
w	EN7	EN6	EN5	EN4	EN3	EN2	EN1	EN0
reset	0	0	0	0	0	0	0	0

EN0–7 **Enable Flag**
w1: Output driver is enabled
w0: Output driver is disabled

5.18.4. I^{2}C Ports P40 and P41

The I^{2}C ports SDA and SCL can alternatively be used as digital I/O ports. To activate the I^{2}C function of the port pin the corresponding MOD flag has to be set to special mode. In normal mode the port pin serves as digital I/O. The output stage is open-drain only. After reset, the I^{2}C ports are in special mode.

5.18.4.1. Features

- digital I/O port
- I^{2}C input and output function
- Schmitt trigger input buffer
- open-drain output
- connected to standby supply

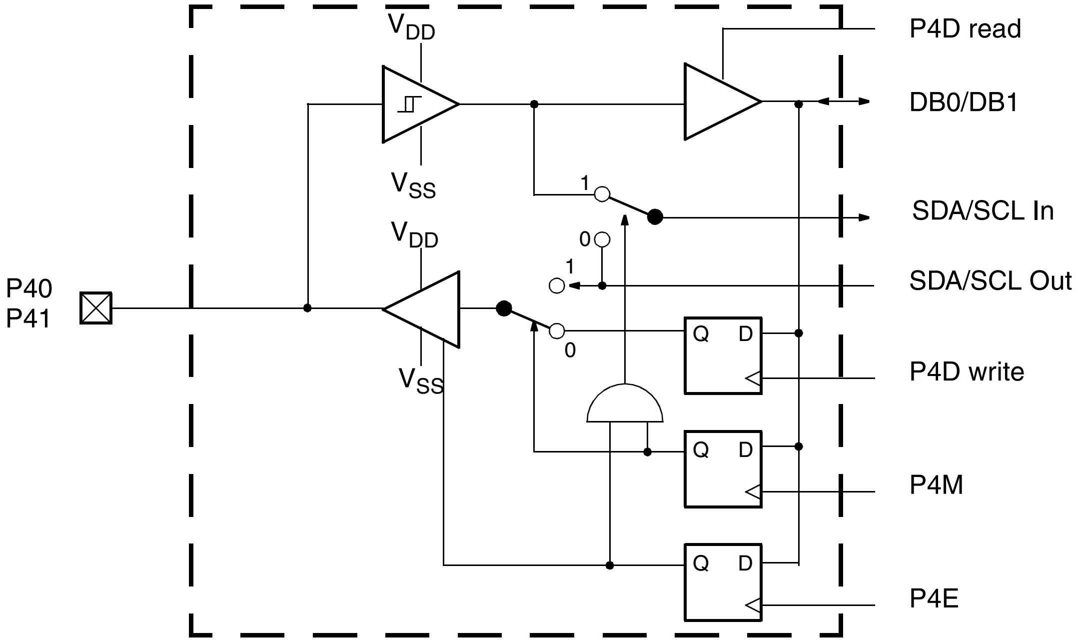


Fig. 5–6: I^{2}C port circuit

.

1F9C		P4D			Port 4 Data Register			
bit	7	6	5	4	3	2	1	0
r		AIN3D	AIN2D	AIN1D	AOUT2D	AOUT1D	SCLD	SDAD
w							SCLD	SDAD
reset		0	0	0	0	0	0	0

SCLD **SCL Data Input/Output**
r: Read pin level resp. data latch.
w: Write data to data latch.

SDAD **SDA Data Input/Output**
r: Read pin level resp. data latch.
w: Write data to data latch.

To use the I^{2}C ports as software output, the appropriate drivers must be activated by setting the SCLEN and SDAEN flag and resetting the SCLM and SDAM flags.

1F9E		P4M			Port 4 Mode Register			
bit	7	6	5	4	3	2	1	0
w		AIN3M	AIN2M	AIN1M	AOUT2M	AOUT1M	SCLM	SDAM
reset		1	1	1	0	0	1	1

SCLM **SCL Normal/Special Mode Flag**
w1: Special I2C Output Mode
w0: Normal Output Mode

SDAM **SDA Normal/Special Mode Flag**
w1: Special I2C Output Mode
w0: Normal Output Mode

1F9F		P4E			Port 4 Enable Register			
bit	7	6	5	4	3	2	1	0
w							SCLEN	SDAEN
reset							1	1

SCLEN **SCL Enable Flag**
w1: Output driver is enabled
w0: Output driver is disabled

SDAEN **SDA Enable Flag**
w1: Output driver is enabled
w0: Output driver is disabled

5.18.5. Audio Ports P42 to P46

The analog audio ports AIN1–3, AOUT1–2 can alternatively be used as digital input ports. To activate the audio function of the port pin, the corresponding MOD flag has to be set to Special mode. In Normal mode the port pin serves as digital input. After reset the audio ports are in Normal mode.

5.18.5.1. Features

- analog audio input or output
- digital input port
- Schmitt trigger input buffer
- special input function

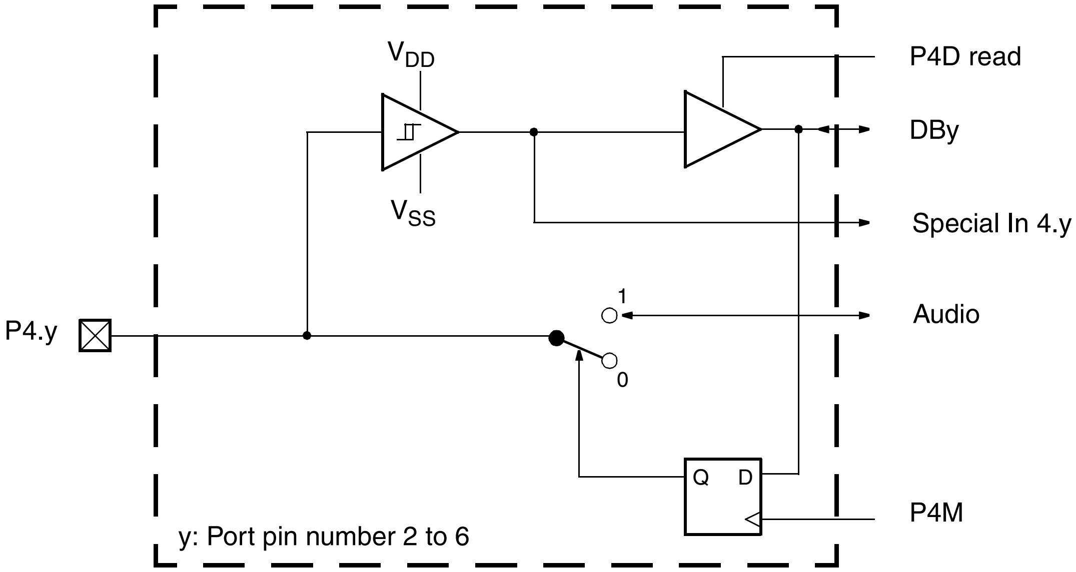


Fig. 5–7: Audio port circuit

AINxD	**AINx Data Input**
r:	Read pin level resp. data latch.

AOUTxD	**AOUTx Data Input**
r:	Read pin level resp. data latch.

To use the Audio ports as software input, the corresponding flags must be programmed to Normal Input mode.

AINxM	**AINx Normal/Special Mode Flag**
w1:	Special Audio Input Mode
w0:	Normal Input Mode

AOUTxM	**AOUTx Normal/Special Mode Flag**
w1:	Special Audio Output Mode
w0:	Normal Input Mode

5.18.6.CLK20 Output Port

The CLK20 pin delivers the internal 20.25-MHz clock. The output stage is push-pull with programmable driver strength (C20M.DSTR). The CLK20 pin can alternatively be used as digital output port. It is possible to force the CLK20 output either to High or Low (C20M.FSO) or to switch it into tristate mode (C20M.DOD). After reset, the CLK20 port is enabled.

5.18.6.1. Features

- programmable driver strength
- tristate mode
- digital output port

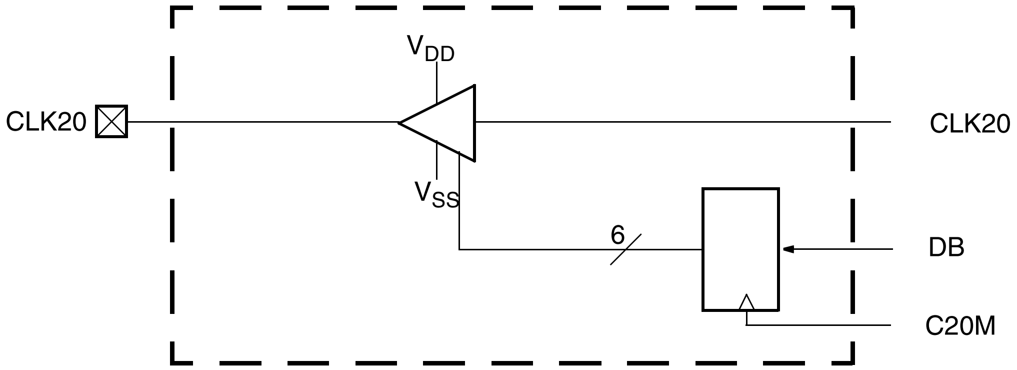


Fig. 5–8: CLK20 Port Circuit

1F9D		C20M			CLK20 Mode Register			
bit	7	6	5	4	3	2	1	0
w			FSO		DOD	DSTR		
reset		0	0	0	0	0	0	0

DSTR	**Driver Strength**
w000:	Output driver strong
w111:	Output driver weak

DOD	**Disable Output Driver**
w1:	Output driver is high-impedance
w0:	Output driver is enabled

FSO	**Force Static Output**
w10:	Output driver is forced to 1
w11:	Output driver is forced to 0

5.19.I/O Register Cross Reference

Table 5–26: I/O Register Map

Addr.	Mnemonic	Name	Mode	Reset	Section
1E00	MASK1L	Mask 1 Low Byte	w	FF	DMA Interface (chapter 5.9. on page 96)
1E01	MASK2L	Mask 2 Low Byte	w	FF	
1E02	MASK3L	Mask 3 Low Byte	w	FF	
1E03	MASK4L	Mask 4 Low Byte	w	FF	
1E04	MASK1H	Mask 1 High Byte	w	FF	
1E05	MASK2H	Mask 2 High Byte	w	FF	
1E06	MASK3H	Mask 3 High Byte	w	FF	
1E07	MASK4H	Mask 4 High Byte	w	FF	
1E08	CMP1L	Compare 1 Low Byte	w	FF	
1E09	CMP2L	Compare 2 Low Byte	w	FF	
1E0A	CMP3L	Compare 3 Low Byte	w	FF	
1E0B	CMP4L	Compare 4 Low Byte	w	FF	
1E0C	CMP1H	Compare 1 High Byte	w	FF	
1E0D	CMP2H	Compare 2 High Byte	w	FF	
1E0E	CMP3H	Compare 3 High Byte	w	FF	
1E0F	CMP4H	Compare 4 High Byte	w	FF	
1E10	MAP1L	Map 1 Low Byte	w	FF	
1E11	MAP2L	Map 2 Low Byte	w	FF	
1E12	MAP3L	Map 3 Low Byte	w	FF	
1E13	MAP4L	Map 4 Low Byte	w	FF	
1E14	MAP1H	Map 1 High Byte	w	FF	
1E15	MAP2H	Map 2 High Byte	w	FF	
1E16	MAP3H	Map 3 High Byte	w	FF	
1E17	MAP4H	Map 4 High Byte	w	FF	
1E18	DMAIM	DMA Interface Mode	w	00	
1E64	PAR0	Patch Address Register 0	w	FF	Memory Patch Module (chapter 5.11. on page 107)
1E65	PAR1	Patch Address Register 1	w	FF	
1E66	PAR2	Patch Address Register 2	w	FF	
1E67	PDR	Patch Data Register	w	00	
1E68	PER0	Patch Enable Register 0	w	00	
1E69	PER1	Patch Enable Register 1	w	00	

Table 5–26: I/O Register Map

Addr.	Mnemonic	Name	Mode	Reset	Section
1F01	CR	Control Register	r/w	–	Control Register (chapter 5.4. on page 87)
1F0F	BR	Banking Register	r/w	01	Memory Banking (chapter 5.8. on page 95)
1F00	CSW0	Clock, Supply & Watchdog Register 0	w	01	Reset Logic (chapter 5.7. on page 90)
1F60	CSW1	Clock, Supply & Watchdog Register 1	r/w	FF	
1F07	RC	Reset Control Register	r/w	00	
1F08	SR0	Standby Register 0	r/w	00	Standby Registers (chapter 5.5. on page 89)
1F09	SR1	Standby Register 1	r/w	40	
1F0A	SR2	Standby Register 2	r/w	00	
1FD0	I2CWS0	I2C Write Start Register 0	w	00	I2C-Bus Master Interface (chapter 5.12. on page 109)
1FD1	I2CWS1	I2C Write Start Register 1	w	00	
1FD2	I2CWD0	I2C Write Data Register 0	w	00	
1FD3	I2CWD1	I2C Write Data Register 1	w	00	
1FD4	I2CWP0	I2C Write Stop Register 0	w	00	
1FD5	I2CWP1	I2C Write Stop Register 1	w	00	
1FD6	I2CRD	I2C Read Data Register	r	00	
1FD7	I2CRS	I2C Read Status Register	r	00	
1FDB	I2CM	I2C Mode Register	w	02	
1E73	I2CPS	I2C Port Select Register	w	00	

Table 5–26: I/O Register Map

Addr.	Mnemonic	Name	Mode	Reset	Section
1F20	IRC	Interrupt Control Register	r/w	0C	Interrupt Controller (chapter 5.10. on page 99)
1F21	IRRET	Interrupt Return Register	r/w	00	
1F22	IRPRI10	Interrupt Priority Register, Input 0 and 1	r/w	00	
1F23	IRPRI32	Interrupt Priority Register, Input 2 and 3	r/w	00	
1F24	IRPRI54	Interrupt Priority Register, Input 4 and 5	r/w	00	
1F25	IRPRI76	Interrupt Priority Register, Input 6 and 7	r/w	00	
1F26	IRPRI98	Interrupt Priority Register, Input 8 and 9	r/w	00	
1F27	IRPRIBA	Interrupt Priority Register, Input 10 and 11	r/w	00	
1F28	IRPRIDC	Interrupt Priority Register, Input 12 and 13	r/w	00	
1F29	IRPRIFE	Interrupt Priority Register, Input 14 and 15	r/w	00	
1F2A	IRP	Interrupt Pending Register	r	00	
1F2B	IRPM0	Interrupt Port Mode	w	00	
1F2C	IRPP	Interrupt Port Prescaler	w	00	
1E71	IRPMUX0	Interrupt Port Multiplex 0	w	00	
1E72	IRPMUX1	Interrupt Port Multiplex 1	w	00	
1FA8	AD0	ADC Register 0	r/w	00	A/D Converter (ADC) (chapter 5.17. on page 123)
1FA9	AD1	ADC Register 1	r	00	
1F4E	TIM0L	Timer 0 Low Byte	r/w	FF	Timer T0 and T1 (chapter 5.13. on page 113)
1F4F	TIM0H	Timer 0 High Byte	r/w	FF	
1F11	TIM0M	Timer 0 Mode	w	00	
1F4C	TIM1L	Timer 1 Low Byte	r/w	FF	
1F4D	TIM1H	Timer 1 High Byte	r/w	FF	
1F13	TIM1M	Timer 1 Mode	w	00	
1F4A	TVPWML	TV PWM Low Byte	w	00	Pulse Width Modulator (chapter 5.15. on page 120)
1F4B	TVPWMH	TV PWM High Byte	w	00	
1F50	PWM0	PWM 0 Register	w	00	
1F51	PWM1	PWM 1 Register	w	00	
1F52	PWM2	PWM 2 Register	w	00	
1F53	PWM3	PWM 3 Register	w	00	

Table 5–26: I/O Register Map

Addr.	Mnemonic	Name	Mode	Reset	Section
1F6C	CC0M	CAPCOM 0 Mode Register	r/w	00	Capture Compare Module (CAPCOM) (chapter 5.14. on page 115)
1F6D	CC0I	CAPCOM 0 Interrupt Register	r/w	00	
1F6E	CC0L	CAPCOM 0 Capture/Compare Low Byte	r/w	FF	
1F6F	CC0H	CAPCOM 0 Capture/Compare High Byte	r/w	FF	
1F70	CC1M	CAPCOM 1 Mode Register	r/w	00	
1F71	CC1I	CAPCOM 1 Interrupt Register	r/w	00	
1F72	CC1L	CAPCOM 1 Capture/Compare Low Byte	r/w	FF	
1F73	CC1H	CAPCOM 1 Capture/Compare High Byte	r/w	FF	
1F7C	CCCL	CAPCOM Counter Low Byte	r	00	
1F7D	CCCH	CAPCOM Counter High Byte	r	00	
1F14	CCCS	CAPCOM Clock Select	w	00	
1E70	CCIMUX	CAPCOM Input Multiplex Register	w	00	
1F90	P1D	Port 1 Data Register	r/w	00	Ports (chapter 5.18. on page 126)
1F91	P1O	Port 1 Output Register	w	00	
1F92	P1M	Port 1 Mode Register	w	00	
1F93	P1E	Port 1 Enable Register	w	00	
1F94	P2D	Port 2 Data Register	r/w	00	
1F95	P2O	Port 2 Output Register	w	00	
1F96	P2M	Port 2 Mode Register	w	00	
1F97	P2E	Port 2 Enable Register	w	00	
1F98	P3D	Port 3 Data Register	r/w	00	
1F99	P3O	Port 3 Output Register	w	00	
1F9A	P3M	Port 3 Mode Register	w	00	
1F9B	P3E	Port 3 Enable Register	w	00	
1F9C	P4D	Port 4 Data Register	r/w	00	
1F9E	P4M	Port 4 Mode Register	w	73	
1F9F	P4E	Port 4 Enable Register	w	03	
1F9D	C20M	CLK20 Mode Register	w	00	
1FFB	TST5	Test Register 5	r	00	Test Registers (chapter 5.6. on page 90)
1FFC	TST4	Test Register 4	w	00	
1FFD	TST3	Test Register 3	w	00	
1FFE	TST1	Test Register 1	w	00	
1FFF	TST2	Test Register 2	w	00	

6. Specifications

6.1. Outline Dimensions

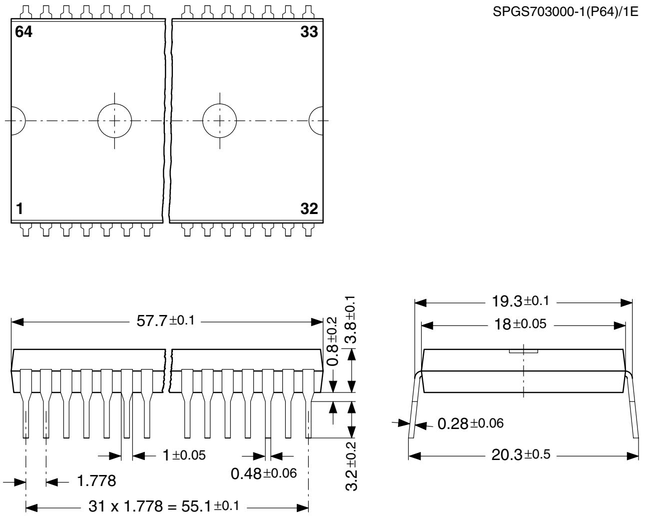
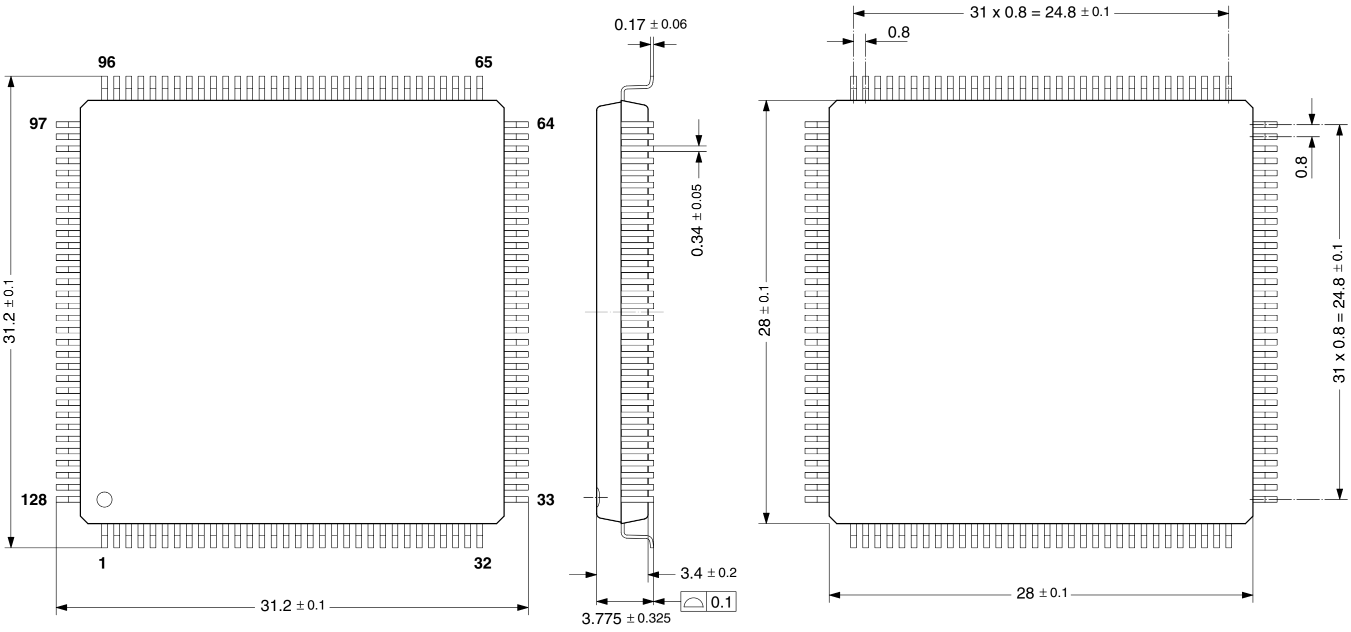


Fig. 6–1: 64-Pin Plastic Shrink Dual-Inline Package **(PSDIP64)**
Weight approximately 9.0 g
Dimensions in mm



SPGS706000-5(P128)/1E

Fig. 6–2: 128-Pin Plastic Metric Quad Flat Package **(PMQFP128)**
Weight approximately 5.4 g
Dimensions in mm

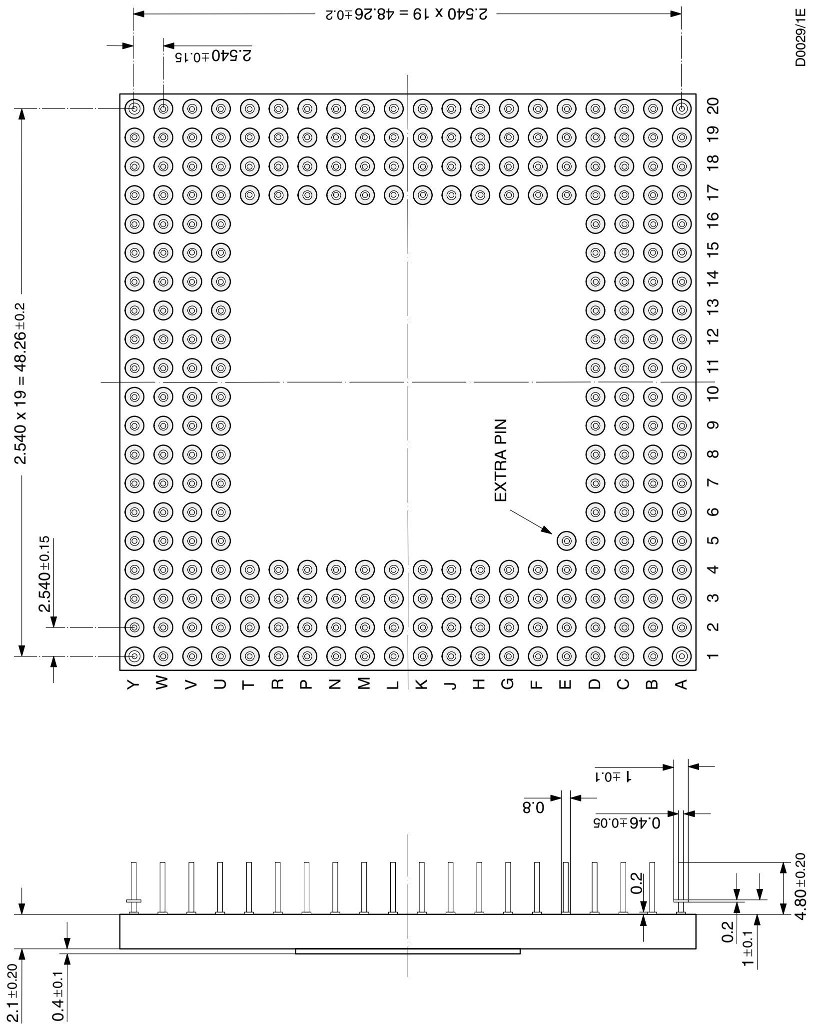


Fig. 6–3: 257-Pin Ceramic Pin Grid Array **(CPGA257)**
Weight approximately 32 g
Dimensions in mm

6.2. Pin Connections and Short Descriptions

NC = not connected
LV = if not used, leave vacant
X = obligatory; connect as described in circuit diagram

IN = Input
OUT = Output
SUPPLY = Supply Pin

Pin No.			Pin Name	Type	Connection (If not used)	Short Description
PSDIP 64-pin	PMQFP 128-pin	CPGA 257-pin				
1	33	A-1	P17	IN/OUT	LV	Port 1, Bit 7
2	34	C-2	P16	IN/OUT	LV	Port 1, Bit 6
3	37	C-1	VSUP$_{P1}$	SUPPLY	X	Supply Voltage, Port 1
4	38	E-2	GND$_{P1}$	SUPPLY	X	Ground, Port 1
5	35	B-1	P15	IN/OUT	LV	Port 1, Bit 5
6	36	D-2	P14	IN/OUT	LV	Port 1, Bit 4
7	39	D-1	P13	IN/OUT	LV	Port 1, Bit 3
8	40	F-2	P12	IN/OUT	LV	Port 1, Bit 2
9	41	E-1	P11	IN/OUT	LV	Port 1, Bit 1
10	42	G-2	P10	IN/OUT	LV	Port 1, Bit 0
11	43	F-1	VOUT	OUT	LV	Analog Video Output
12	44	H-2	VRT	IN	X	Reference Voltage Top, Video ADC
13	45	G-1	SGND	IN	GND$_{AF}$	Signal Ground for Analog Input
14	46	H-1	GND$_{AF}$	SUPPLY	X	Ground, Analog Front-end
15	47	J-1	VSUP$_{AF}$	SUPPLY	X	Supply Voltage, Analog Front-end
16	48	K-1	CBIN	IN	VRT	Analog Component Cb Input
17	49	L-1	CIN1	IN	VRT	Analog Chroma 1 Input
18	50	M-1	CIN2/ CRIN	IN	VRT	Analog Chroma 2 Input / Analog Component Cr Input
19	51	N-1	VIN1	IN	VRT	Analog Video 1 Input
20	52	N-2	VIN2	IN	VRT	Analog Video 2 Input
21	53	P-1	VIN3	IN	VRT	Analog Video 3 Input
22	54	P-2	VIN4	IN	VRT	Analog Video 4 Input
23	89	Y-16	TEST	IN	GND$_{S}$	Test Pin, reserved for Test
24	76	W-8	HOUT	OUT	X	Horizontal Drive Output
25	77	Y-7	VSUP$_{D}$	SUPPLY	X	Supply Voltage, Digital Circuitry
26	78	Y-8	GND$_{D}$	SUPPLY	X	Ground, Digital Circuitry
27	90	W-16	FBLIN	IN	GND$_{AB}$	Fast Blank Input
28	91	Y-17	RIN	IN	GND$_{AB}$	Analog Red Input
29	92	W-17	GIN	IN	GND$_{AB}$	Analog Green Input
30	93	Y-18	BIN	IN	GND$_{AB}$	Analog Blue Input

Pin No.			Pin Name	Type	Connection (If not used)	Short Description
PSDIP 64-pin	PMQFP 128-pin	CPGA 257-pin				
31	94	W-18	VPROT	IN	GND$_{D}$	Vertical Protection Input
32	95	Y-19	SAFETY	IN	GND$_{D}$	Safety Input
33	96	W-19	HFLB	IN	HOUT	Horizontal Flyback Input
34	97	Y-20	VERTQ / INTLC	OUT	LV	Differential Vertical Sawtooth Output Interlace Control Output
35	98	V-19	VERT	OUT	LV	Differential Vertical Sawtooth Output
36	99	W-20	EW	OUT	LV	Vertical Parabola Output
37	100	U-19	SENSE	IN	GND$_{AB}$	Sense ADC Input
38	101	V-20	GNDM	SUPPLY	X	Ground, MADC Input
39	102	T-19	RSW1	OUT	LV	Range Switch1 for Measurement ADC
40	103	U-20	RSW2	OUT	LV	Range Switch2 for Measurement ADC
41	104	R-19	SVMOUT	OUT	VSUP$_{AB}$	Scan Velocity Modulation Output
42	105	T-20	ROUT	OUT	VSUP$_{AB}$	Analog Red Output
43	106	P-19	GOUT	OUT	VSUP$_{AB}$	Analog Green Output
44	107	R-20	BOUT	OUT	VSUP$_{AB}$	Analog Blue Output
45	108	N-19	VSUP$_{AB}$	SUPPLY	X	Supply Voltage, Analog Back-end
46	109	P-20	GND$_{AB}$	SUPPLY	X	Ground, Analog Back-end
47	110	N-20	VRD	IN	X	DAC Reference
48	111	M-20	XREF	IN	X	Reference Input for RGB DACs
49	112	L-20	AIN3	IN	GND$_{S}$	Analog Audio 3Input
50	113	K-20	AIN2	IN	GND$_{S}$	Analog Audio 2Input
51	114	J-20	AIN1	IN	GND$_{S}$	Analog Audio 1 Input
52	115	H-20	AOUT2	OUT	LV	Analog Audio 2 Output
53	116	H-19	AOUT1	OUT	LV	Analog Audio 1 Output
54	122	E-19	VSUP$_{S}$	SUPPLY	X	Supply Voltage, Standby
55	123	D-20	GND$_{S}$	SUPPLY	X	Ground, Standby
56	120	F-19	XTAL1	IN	X	Analog Crystal Input
57	121	E-20	XTAL2	OUT	X	Analog Crystal Output
58	117	G-20	RESQ	IN/OUT	X	Reset Input/Output, Active Low
59	118	G-19	SCL	IN/OUT	X	I^{2}C Bus Clock
60	119	F-20	SDA	IN/OUT	X	I^{2}C Bus Data
61	62	V-2	P23	IN/OUT	LV	Port 2, Bit 3
62	63	W-1	P22	IN/OUT	LV	Port 2, Bit 2
63	64	W-2	P21	IN/OUT	LV	Port 2, Bit 1

Pin No.			**Pin Name**	**Type**	**Connection (If not used)**	**Short Description**
PSDIP 64-pin	**PMQFP 128-pin**	**CPGA 257-pin**				
64	65	Y-1	P20	IN/OUT	LV	Port 2, Bit 0
	1	A-20	ADB17	OUT	LV	Address Bus 17
	2	B-18	VSUP_{ADB}	SUPPLY	X	Supply Voltage, Address Bus
	3	A-19	GND_{ADB}	SUPPLY	X	Ground, Address Bus
	4	B-17	ADB16	OUT	LV	Address Bus 16
	5	A-18	ADB15	OUT	LV	Address Bus 15
	6	B-16	ADB14	OUT	LV	Address Bus 14
	7	A-17	ADB13	OUT	LV	Address Bus 13
	8	B-15	ADB12	OUT	LV	Address Bus 12
	9	A-16	ADB11	OUT	LV	Address Bus 11
	10	B-14	ADB10	OUT	LV	Address Bus 10
	11	A-15	ADB9	OUT	LV	Address Bus 9
	12	B-13	ADB8	OUT	LV	Address Bus 8
	13	A-14	ADB7	OUT	LV	Address Bus 7
	14	A-13	ADB6	OUT	LV	Address Bus 6
	15	A-12	ADB5	OUT	LV	Address Bus 5
	16	A-11	VSUP_{ADB}	SUPPLY	X	Supply Voltage, Address Bus
	17	A-10	GND_{ADB}	SUPPLY	X	Ground, Address Bus
	18	A-9	ADB4	OUT	LV	Address Bus 4
	19	A-8	ADB3	OUT	LV	Address Bus 3
	20	B-8	ADB2	OUT	LV	Address Bus 2
	21	A-7	ADB1	OUT	LV	Address Bus 1
	22	B-7	ADB0	OUT	LV	Address Bus 0
	23	A-6	DB0	IN/OUT	LV	Data Bus 0
	24	B-6	DB1	IN/OUT	LV	Data Bus 1
	25	A-5	DB2	IN/OUT	LV	Data Bus 2
	26	B-5	DB3	IN/OUT	LV	Data Bus 3
	27	A-4	VSUP_{DB}	SUPPLY	X	Supply Voltage, Data Bus
	28	B-4	GND_{DB}	SUPPLY	X	Ground, Data Bus
	29	A-3	DB4	IN/OUT	LV	Data Bus 4
	30	B-3	DB5	IN/OUT	LV	Data Bus 5
	31	A-2	DB6	IN/OUT	LV	Data Bus 6
	32	B-2	DB7	IN/OUT	LV	Data Bus 7
	55	R-1	DISINTROM	IN	X	Disable Internal ROM

Pin No.			**Pin Name**	**Type**	**Connection (If not used)**	**Short Description**
PSDIP 64-pin	**PMQFP 128-pin**	**CPGA 257-pin**				
	56	R-2	P27	IN/OUT	LV	Port 2, Bit 7
	57	T-1	P26	IN/OUT	LV	Port 2, Bit 6
	58	T-2	P25	IN/OUT	LV	Port 2, Bit 5
	59	U-1	P24	IN/OUT	LV	Port 2, Bit 4
	60	U-2	VSUP$_{P2}$	SUPPLY	X	Supply Voltage, Port 2
	61	V-1	GND$_{P2}$	SUPPLY	X	Ground, Port 2
	66	W-3	VBCLK	IN	GND$_D$	Video Bus Clock
	67	Y-2	VB7	IN	GND$_D$	Video Bus 7
	68	W-4	VB6	IN	GND$_D$	Video Bus 6
	69	Y-3	VB5	IN	GND$_D$	Video Bus 5
	70	W-5	VB4	IN	GND$_D$	Video Bus 4 / Bond 0=16k Text RAM
	71	Y-4	VB3	IN	GND$_D$	Video Bus 3 / Bond 1=CTI
	72	W-6	VB2	IN	GND$_D$	Video Bus 2 / Bond 2=Scaler
	73	Y-5	VB1	IN	GND$_D$	Video Bus 1 / Bond 3=Comb Filter
	74	W-7	VB0	IN	GND$_D$	Video Bus 0 / Bond 4=VDP Full/Lite
	75	Y-6	CLK20	OUT	LV	20 MHz System Clock Output
	79	Y-9	P37	IN/OUT	LV	Port 3, Bit 7
	80	Y-10	P36	IN/OUT	LV	Port 3, Bit 6
	81	Y-11	P35	IN/OUT	LV	Port 3, Bit 5
	82	Y-12	P34	IN/OUT	LV	Port 3, Bit 4
	83	Y-13	VSUP$_{P3}$	SUPPLY	X	Supply Voltage, Port 3
	84	W-13	GND$_{P3}$	SUPPLY	X	Ground, Port 3
	85	Y-14	P33	IN/OUT	LV	Port 3, Bit 3
	86	W-14	P32	IN/OUT	LV	Port 3, Bit 2
	87	Y-15	P31	IN/OUT	LV	Port 3, Bit 1
	88	W-15	P30	IN/OUT	LV	Port 3, Bit 0
	124	D-19	WE1Q	OUT	LV	$\overline{\text{Write Enable}}$ Output 1
	125	C-20	WE2Q	OUT	LV	$\overline{\text{Write Enable}}$ Output 2
	126	C-19	OE1Q	OUT	LV	$\overline{\text{Output Enable}}$ Output 1
	127	B-20	OE2Q	OUT	LV	$\overline{\text{Output Enable}}$ Output 2
	128	B-19	ADB18	OUT	LV	Address Bus 18

6.3. Pin Descriptions for PSDIP64 package

Pin 1,2,5-10, **P10**–**P17** – I/O Port (Fig. 6–28)
These pins provide CPU controlled I/O ports.

Pin 3, **VSUPP1*** – Supply Voltage, Port 1 Driver
This pin is used as supply for the I/O port 1 driver.

Pin 4, **GNDP1*** – Ground, Port 1 Driver
This is the ground reference for the I/O port 1 driver.

Pin 11, **VOUT**– Analog Video Output (Fig. 6–13)
The analog video signal that is selected for the main (luma, CVBS) adc is output at this pin. An emitter follower is required at this pin.

Pin 12, **VRT** – Reference Voltage Top (Fig. 6–14)
Via this pin, the reference voltage for the A/D converters is decoupled. The pin is connected with 10 µF/47 nF to the Signal Ground Pin.

Pin 13, **SGND** – Signal GND for Analog Input
This is the high quality ground reference for the video input signals.

Pin 14, **GNDAF*** – Ground, Analog Front-end
This pin has to be connected to the analog ground. No supply current for the digital stages should flow through this line.

Pin 15, **VSUPAF*** – Supply Voltage, Analog Front-end
This pin has to be connected to the analog supply voltage. No supply current for the digital stages should flow through this line.

Pin 16,18, **CBIN,CRIN** – Analog Chroma Component Input (Fig. 6–11)
These pins are used as the chroma component (CB, CR) inputs required for the analog YUV Interface. The input signal must be AC-coupled. The CRIN pin can alternatively be used as the second SVHS chroma input (CIN2).

Pin 17,18, **CIN1,CIN2** – Analog Chroma Input (Fig. 6–11)
These are the analog chroma inputs. A S-VHS chroma signal is converted using the chroma (Video 2) AD converter. A resistive divider is used to bias the input signal to the middle of the converter input range. The input signal must be AC-coupled. The CIN2 pin can alternatively be used as the chroma component (CR) input required for the analog YUV Interface.

Pins 19–22, **VIN1–4** – Analog Video Input (Fig. 6–10)
These are the analog video inputs. A CVBS or S-VHS luma signal is converted using the luma (Video 1) AD converter. The input signal must be AC-coupled.

Pin 23, **TEST** – Test Input (Fig. 6–6)
This pin enables factory test modes. For normal operation, it must be connected to ground.

Pin 24, **HOUT** – Horizontal Drive Output (Fig. 6–17)
This open drain output supplies the drive pulse for the horizontal output stage. The polarity and gating with the flyback pulse are selectable by software.

Pin 25, **VSUPD*** – Supply Voltage, Digital Circuitry

Pin 26, **GNDD*** – Ground, Digital Circuitry
This is the ground reference for the digital circuitry.

Pin 27, **FBLIN** – Fast Blank Input (Fig. 6–19)
These pins are used to switch the RGB outputs to the external analog RGB inputs. The active level (low or high) can be selected by software.

Pin 28,29,30, **RIN, GIN, BIN** – Analog RGB Input (Fig. 6–15)
These pins are used to insert an external analog RGB signal, e.g. from a SCART connector which can by switched to the analog RGB outputs with the fast blank signal. The analog back-end provides separate brightness and contrast settings for the external analog RGB signals.

Pin 31, **VPROT** – Vertical Protection Input (Fig. 6–18)
The vertical protection circuitry prevents the picture tube from burn-in in the event of a malfunction of the vertical deflection stage. During vertical blanking, a signal level of 2.5V is sensed. If a negative edge cannot be detected, the RGB output signals are blanked.

Pin 32, **SAFETY** – Safety Input (Fig. 6–18)
This is a three-level input. Low level means normal function. At the medium level RGB output signals are blanked. At high level RGB output signals are blanked and horizontal drive is shut off.

Pin 33, **HFLB** – Horizontal Flyback Input (Fig. 6–18)
Via this pin the horizontal flyback pulse is supplied to the VCT 38xxA.

Pin 34, **VERTQ, INTLC** – Inverted Vertical Sawtooth Output (Fig. 6–21) / Interlace Output (Fig. 6–20)
This pin supplies the inverted signal of VERT. Together with the VERT pin it can be used to drive symmetrical deflection amplifiers. The drive signal is generated with 15-bit precision. The analog voltage is generated by a 4 bit current-DAC with external resistor and uses digital noise shaping. Alternatively this pin supplies the interlace information, the polarity is programmable.

Pin 35, **VERT** – Vertical Sawtooth Output (Fig. 6–21)
This pin supplies the drive signal for the vertical output stage. The drive signal is generated with 15-bit precision. The analog voltage is generated by a 4 bit current-DAC with external resistor and uses digital noise shaping.

Pin 36, **EW** – East-West Parabola Output (Fig. 6–22)
This pin supplies the parabola signal for the East-West correction. The drive signal is generated with 15 bit precision. The analog voltage is generated by a 4 bit current-DAC with external resistor and uses digital noise shaping.

Pin 37, **SENSE** – Measurement ADC Input (Fig. 6–24)
This is the input of the analog to digital converter for the picture and tube measurement. Three measurement ranges are selectable with RSW1 and RSW2.

Pin 38, **GNDM** – Measurement ADC Reference Input
This is the ground reference for the measurement A/D converter. Connect this pin to GND_{AB}

Pin 39, 40, **RSW1**, **RSW2** – Range Switch for Measuring ADC (Fig. 6–23)
These pins are open drain pulldown outputs. RSW1 is switched off during cutoff and whitedrive measurement. RSW2 is switched off during cutoff measurement only.

Pin 41, **SVMOUT** – Scan Velocity Modulation Output (Fig. 6–16)
This output delivers the analog SVM signal. The D/A converter is a current sink like the RGB D/A converters. At zero signal the output current is 50% of the maximum output current.

Pin 42, 43, 44, **ROUT, GOUT, BOUT** – Analog RGB Output (Fig. 6–16)
These pins are the analog Red/Green/Blue outputs of the back-end. The outputs are current sinks.

Pin 45, **VSUPAB*** – Supply Voltage, Analog Back-end
This pin has to be connected to the analog supply voltage. No supply current for the digital stages should flow through this line.

Pin 46, **GNDAB*** – Ground, Analog Back-end
This pin has to be connected to the analog ground. No supply current for the digital stages should flow through this line.

Pin 47, **VRD** – DAC Reference Decoupling (Fig. 6–25)
Via this pin the DAC reference voltage is decoupled by an external capacitor. The DAC output currents depend on this voltage, therefore a pulldown transistor can be used to shut off all beam currents. A decoupling capacitor of 4.7 µF in parallel to 100 nF (low inductance) is required.

Pin 48, **XREF** – DAC Current Reference (Fig. 6–25)
External reference resistor for DAC output currents, typical 10 kΩ to adjust the output current of the D/A converters. (see recommended operating conditions). This resistor has to be connected to analog ground as closely as possible to the pin.

Pin 49, 50, 51, **AIN1**–**3** – Analog Audio Input (Fig. 6–26)
The analog input signal from TUNER or SCART is fed to this pin. The input signal must be AC-coupled. Alternatively these pins can be used as digital I/O ports (Fig. 6–29).

Pin 52,53, **AOUT1, AOUT2** – Analog Audio Output (Fig. 6–27)
These pins are the analog audio outputs. Connections to these pins must use a 680 ohm series resistor as closely as possible to these pins. The output signals are intended to be AC coupled. Alternatively these pins can be used as digital I/O ports (Fig. 6–29).

Pin 54, **VSUPS*** – Supply Voltage, Standby

Pin 55, **GNDS*** – Ground, Standby
This is the ground reference for the standby circuitry.

Pins 56 and 57, **XTAL1** Crystal Input and **XTAL2** Crystal Output (Fig. 6–8)
These pins are connected to an 20.25 MHz crystal oscillator which is digitally tuned by integrated shunt capacitances. The CLK20 clock signal is derived from this oscillator.

Pin 58, **RESQ** – Reset Input/Output (Fig. 6–7)
A low level on this pin resets the VCT 38xxA. The internal CPU can pull down this pin to reset external devices connected to this pin.

Pin 59, **SCL** – I^2C Bus Clock (Fig. 6–7)
This pin connects to the I^2C bus clock line. The signal can be pulled down by external slave ICs to slow down data transfer.

Pin 60, **SDA** – I^2C Bus Data (Fig. 6–7)
This pin connects to the I^2C bus data line.

Pin 61–64, **P20**–**P23** – I/O Port (Fig. 6–28)
These pins provide CPU controlled I/O ports.

6.4. Pin Descriptions for PMQFP128 package

Pins 1, 4–15, 18–22, 128, **ADB0**–**ADB18** – Address Bus Output (Fig. 6–30)
These 19 lines provide the CCU address bus output to access external memory.

Pin 2, 16, **VSUPADB*** – Supply Voltage, Address Bus Driver
This pin is used as supply for the address bus driver.

Pin 3, 17, **GNDADB*** – Ground, Address Bus Driver
This is the ground reference for the address bus driver.

Pins 23–26, 29–32, **DB0**–**DB7** – Data Bus Input/Output (Fig. 6–31)
These 8 lines provide the bidirectional CCU data bus to access external memory.

Pin 27, **VSUPDB*** – Supply Voltage, Data Bus Driver
This pin is used as supply for the CCU data bus driver.

Pin 28, **GNDDB*** – Ground, Data Bus Driver
This is the ground reference for the CCU data bus driver.

Pin 55, **DISINTROM** – Disable Internal ROM Input (Fig. 6–6)
A high level at this pin disables the internal CCU program memory during reset. In this case the CCU loads the control word from external address location h’FFF9.

Pin 56–59, **P27**–**P24** – I/O Port (Fig. 6–28)
These pins provide CCU controlled I/O ports.

Pin 60, **VSUPP2*** – Supply Voltage, Port 2 Driver
This pin is used as supply for the I/O port 2 driver.

Pin 61, **GNDP2*** – Ground, Port 2 Driver
This is the ground reference for the I/O port 2 driver.

Pins 66–74, **VBCLK, VB0**–**VB7** – Digital Video Bus Input (Fig. 6–32)
In future versions of VCT 38xxA these pins will provide the ITU–R 656 video interface. As long as the ITU-R 656 video interface is not available, these pins have to be connected to GND$_D$.

Pin 75, **CLK20** – Main Clock Output (Fig. 6–9)
This is the 20.25 MHz main clock output.

Pin 79–82, 85–88, **P37**–**P30** – I/O Port (Fig. 6–28)
These pins provide CCU controlled I/O ports.

Pin 83, **VSUPP3*** – Supply Voltage, Port 3Driver
This pin is used as supply for the I/O port 3 driver.

Pin 84, **GNDP3*** – Ground, Port 3Driver
This is the ground reference for the I/O port 3 driver.

Pin 124, **WE1Q** – $\overline{\text{Write Enable}}$ Output 1 (Fig. 6–30)
This pin controls the direction of data exchange between the CCU and the external program memory device.

Pin 125, **WE2Q** – $\overline{\text{Write Enable}}$ Output 2 (Fig. 6–30)
This pin controls the direction of data exchange between the CCU and external the teletext page memory device.

Pin 126, **OE1Q** – $\overline{\text{Output Enable}}$ Output 1 (Fig. 6–30)
This pin is used to enable the output driver of the external program memory device for read access.

Pin 127, **OE2Q** – $\overline{\text{Output Enable}}$ Output 1 (Fig. 6–30)
This pin is used to enable the output driver of the external teletext page memory device for read access.

*** Application Note:**
All ground pins should be connected to one low-resistive ground plane. All supply pins should be connected separately with short and low-resistive lines to the power supply. Decoupling capacitors from VSUP$_{xx}$ to GND$_{xx}$ are recommended as closely as possible to these pins. Decoupling of VSUP$_D$ and GND$_D$ is most important. We recommend using more than one capacitor. By choosing different values, the frequency range of active decoupling can be extended.

6.5. Pin Configuration

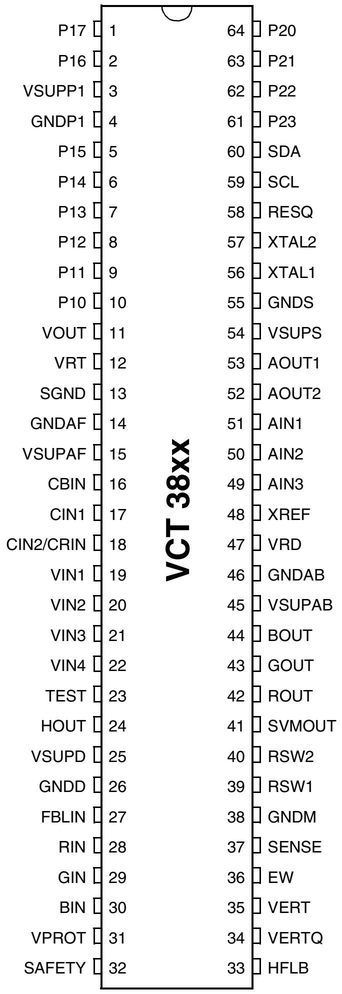


Fig. 6–4: 64-pin PSDIP package

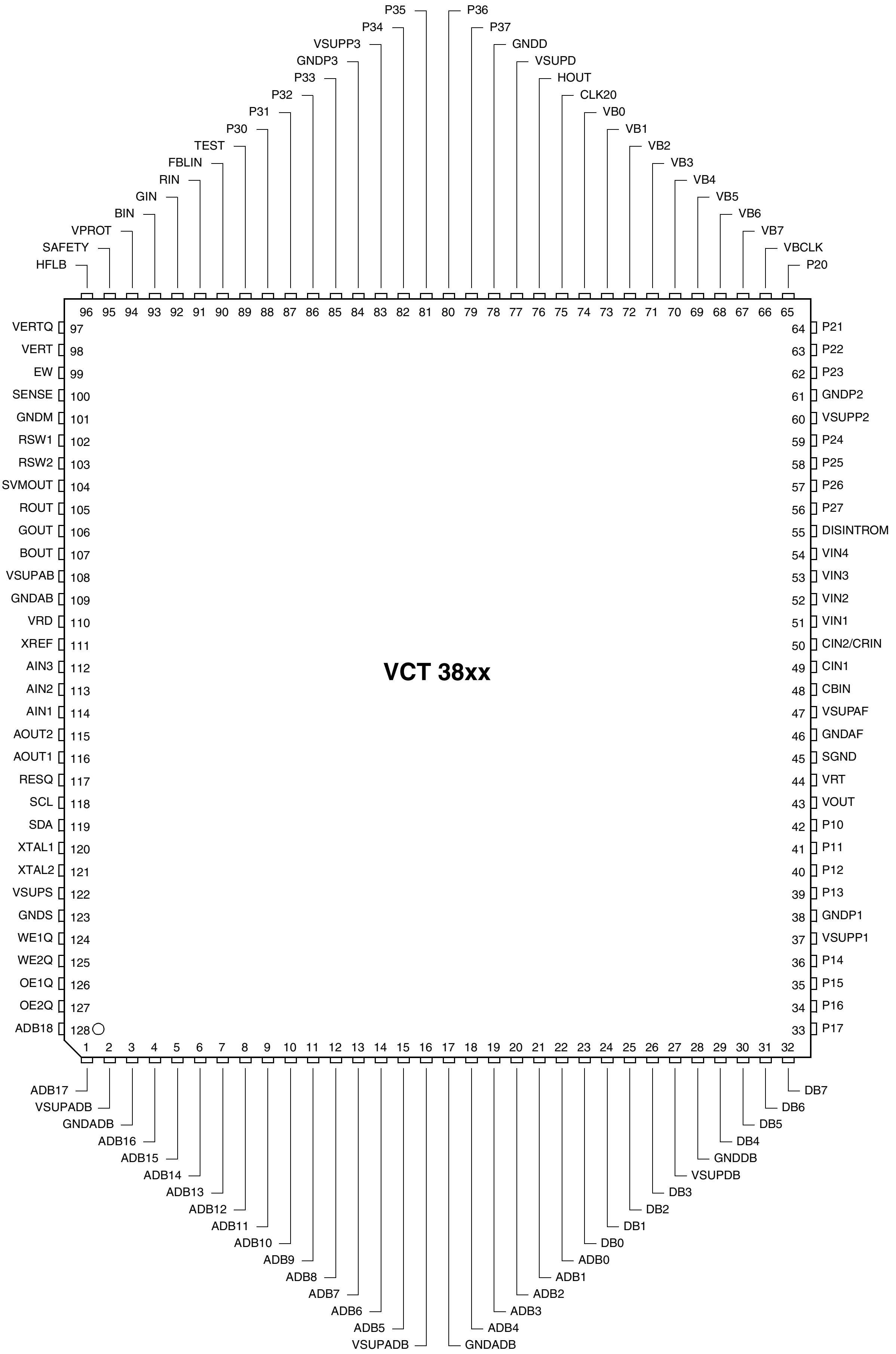

VCT 38xx
96 HFLB
95 SAFETY
94 VPROT
93 BIN
92 GIN
91 RIN
90 FBLIN
89 TEST
88 P30
87 P31
86 P32
85 P33
84 GNDP3
83 VSUPP3
82 P34
81 P35
80 P36
79 P37
78 GNDD
77 VSUPD
76 HOUT
75 CLK20
74 VB0
73 VB1
72 VB2
71 VB3
70 VB4
69 VB5
68 VB6
67 VB7
66 VBCLK
65 P20
97 VERTQ
98 VERT
99 EW
100 SENSE
101 GNDM
102 RSW1
103 RSW2
104 SVMOUT
105 ROUT
106 GOUT
107 BOUT
108 VSUPAB
109 GNDAB
110 VRD
111 XREF
112 AIN3
113 AIN2
114 AIN1
115 AOUT2
116 AOUT1
117 RESQ
118 SCL
119 SDA
120 XTAL1
121 XTAL2
122 VSUPS
123 GNDS
124 WE1Q
125 WE2Q
126 OE1Q
127 OE2Q
128 ADB18
64 P21
63 P22
62 P23
61 GNDP2
60 VSUPP2
59 P24
58 P25
57 P26
56 P27
55 DISINTROM
54 VIN4
53 VIN3
52 VIN2
51 VIN1
50 CIN2/CRIN
49 CIN1
48 CBIN
47 VSUPAF
46 GNDAF
45 SGND
44 VRT
43 VOUT
42 P10
41 P11
40 P12
39 P13
38 GNDP1
37 VSUPP1
36 P14
35 P15
34 P16
33 P17
1 ADB17
2 VSUPADB
3 GNDADB
4 ADB16
5 ADB15
6 ADB14
7 ADB13
8 ADB12
9 ADB11
10 ADB10
11 ADB9
12 ADB8
13 ADB7
14 ADB6
15 ADB5
16 VSUPADB
17 GNDADB
18 ADB4
19 ADB3
20 ADB2
21 ADB1
22 ADB0
23 DB0
24 DB1
25 DB2
26 DB3
27 VSUPDB
28 GNDDB
29 DB4
30 DB5
31 DB6
32 DB7

Fig. 6–5: 128-pin PMQFP package

6.6. Pin Circuits

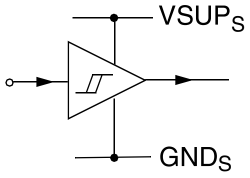


Fig. 6–6: Input pins TEST, DISINTROM

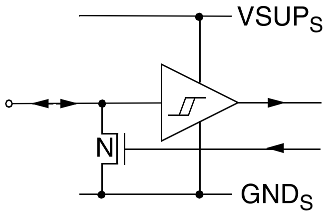


Fig. 6–7: Input/Output pins RESQ, SDA, SCL

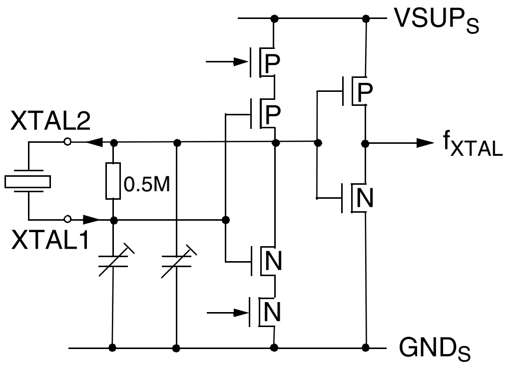


Fig. 6–8: Input/Output pins XTAL1, XTAL2

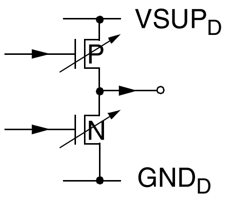


Fig. 6–9: Output pin CLK20

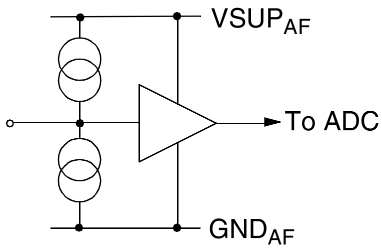


Fig. 6–10: Input pins VIN1–VIN4

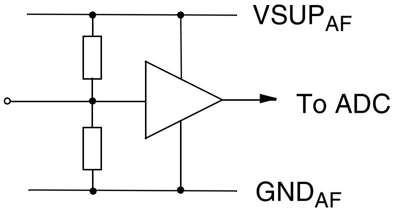


Fig. 6–11: Input pins CIN1, CIN2

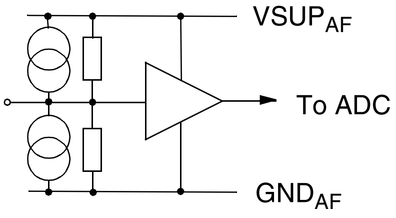


Fig. 6–12: Input pins CRIN, CBIN

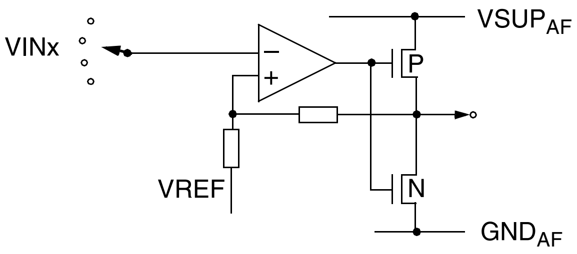


Fig. 6–13: Output pin VOUT

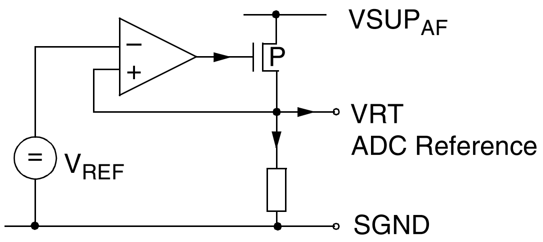


Fig. 6–14: Supply pins VRT, SGND

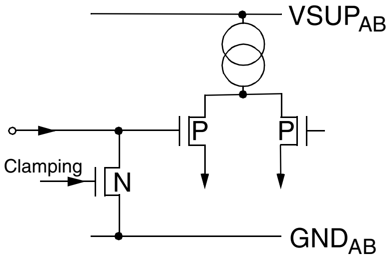


Fig. 6–15: Input pins RIN, GIN, BIN

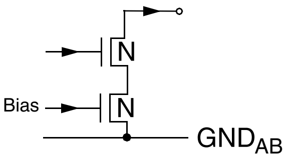


Fig. 6–16: Output pins ROUT, GOUT, BOUT, SVMOUT

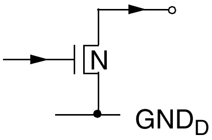


Fig. 6–17: Output pin HOUT

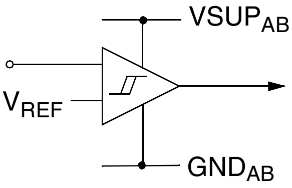


Fig. 6–18: Input pins SAFETY, VPROT, HFLB

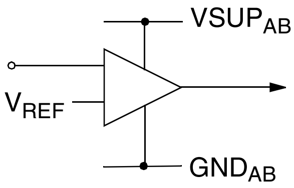


Fig. 6–19: Input pin FBLIN

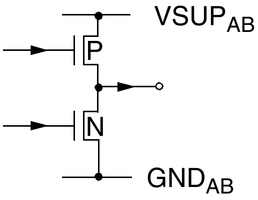


Fig. 6–20: Output pin INTLC

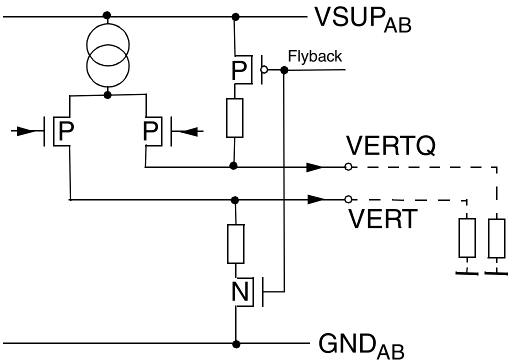


Fig. 6–21: Output pins VERT, VERTQ

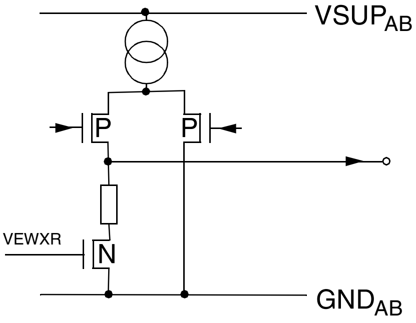


Fig. 6–22: Output pin EW

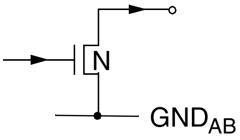


Fig. 6–23: Output pins RSW1, RSW2

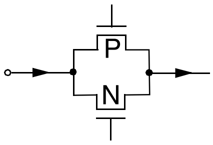


Fig. 6–24: Input pin SENSE

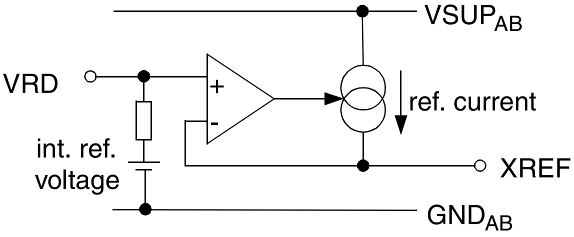


Fig. 6–25: Supply pins XREF, VRD

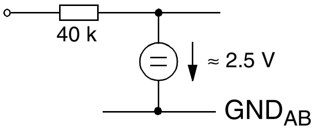


Fig. 6–26: Input pins AIN1-3

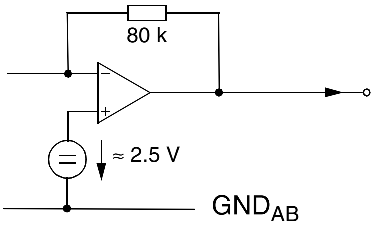


Fig. 6–27: Output pins AOUT1, AOUT2

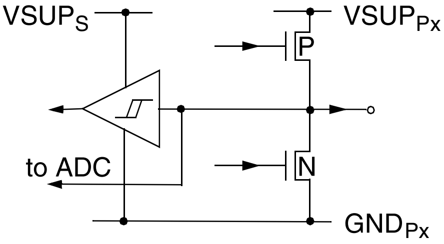


Fig. 6–28: Input/Output pins P10-P17, P20-P27, P30-P37

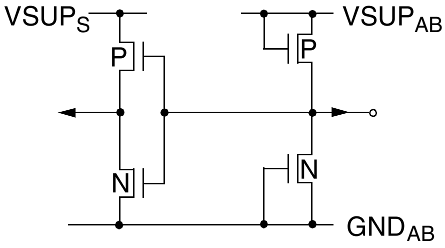


Fig. 6–29: Input pins P42-P46

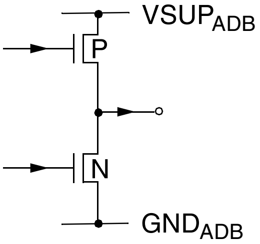


Fig. 6–30: Output pins ADB0-ADB18, $\overline{\text{OE1}}$, $\overline{\text{OE2}}$, $\overline{\text{WE1}}$, $\overline{\text{WE2}}$

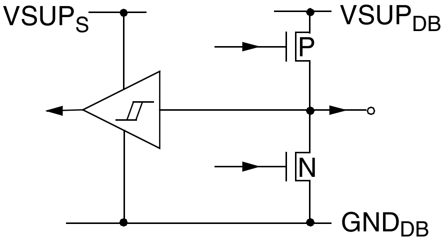


Fig. 6–31: Input/Output pins DB0-DB7

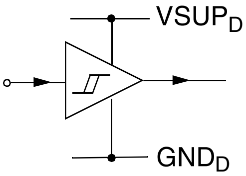


Fig. 6–32: Input pins VB0-VB7, VBCLK

6.7. Electrical Characteristics

6.7.1. Absolute Maximum Ratings

Symbol	Parameter	Pin Name	Min.	Max.	Unit
T_A	Ambient Operating Temperature		–	65	°C
T_C	Case Temperature (PSDIP64)		–	115	°C
T_C	Case Temperature (PMQFP128)		–	115	°C
T_S	Storage Temperature		–40	125	°C
P_{TOT}	Total Power Dissipation		–	1400	mW
$VSUP_x$	Supply Voltage	$VSUP_x$	–0.3	6	V
V_I	Input Voltage, all Inputs		–0.3	$VSUP_x$+0.3[1]	V
V_O	Output Voltage, all Outputs		–0.3	$VSUP_x$+0.3[1]	V
V_{IO}	Input/Output Voltage, all Open Drain Outputs		–0.3	6	V

[1] Refer to Pin Circuits (chapter 6.6. on page 147)

Stresses beyond those listed in the “Absolute Maximum Ratings” may cause permanent damage to the device. This is a stress rating only. Functional operation of the device at these or any other conditions beyond those indicated in the “Recommended Operating Conditions/Characteristics” of this specification is not implied. Exposure to absolute maximum ratings conditions for extended periods may affect device reliability.

6.7.2. Recommended Operating Conditions

6.7.2.1. General Recommendations

Symbol	Parameter	Pin Name	Min.	Typ.	Max.	Unit
T_A	Ambient Operating Temperature		0	–	65	°C
f_{XTAL}	Clock Frequency	XTAL1/2	–	20.25	–	MHz
$VSUP_A$	Analog Supply Voltage	$VSUP_{AF}$ $VSUP_{AB}$	4.75	5.0	5.25	V
$VSUP_D$	Digital Supply Voltage	$VSUP_S$ $VSUP_D$ $VSUP_{VDP}$ $VSUP_{TPU}$ $VSUP_{CCU}$	3.15	3.3	3.45	V
$VSUP_P$	Port Supply Voltage	$VSUP_{Px}$ $VSUP_{DB}$ $VSUP_{ADB}$	3.15	3.3/5.0	5.25	V
$VSUP_{OFF}$	Standby Supply Voltage	$VSUP_{AF}$ $VSUP_{AB}$ $VSUP_D$	0	–	0.5	V
$VSUP_{\Delta}$	Difference between Standby and Emulator Supply Voltage	$VSUP_S$ $VSUP_{VDP}$ $VSUP_{TPU}$ $VSUP_{CCU}$	0	–	0.3	V

6.7.2.2. Analog Input and Output Recommendations

Symbol	Parameter	Pin Name	Min.	Typ.	Max.	Unit
Audio						
C_{AIN}	Input Coupling Capacitor Audio Inputs	AIN1–3	–	330	–	nF
V_{AIN}	Audio Input Level		–	–	1.0	V_{RMS}
R_{LAOUT}	Audio Output Load Resistance	AOUT1–2	10	–	–	kΩ
C_{LAOUT}	Audio Output Load Capacitance		–	–	1.0	nF
R_{SAOUT}	Audio Output Serial Resistance		–	680	–	Ω
Video						
V_{VIN}	Video Input Level	VIN1–4, CIN1–2	0.5	1.0	3.5	V
V_{CIN}	Chroma Input Level	CRIN, CBIN	–	700	–	mV
C_{VIN}	Input Coupling Capacitor Video Inputs	VIN1–4	–	680	–	nF
C_{CIN}	Input Coupling Capacitor Chroma Inputs	CIN1–2	–	1	–	nF
C_{CCIN}	Input Coupling Capacitor Component Inputs	CRIN, CBIN	–	220	–	nF
RGB						
R_{xref}	RGB–DAC Current defining Resistor	XREF	9.5	10	10.5	kΩ
C_{RGBIN}	RGB Input Coupling Capacitor	RIN GIN BIN	–	15	–	nF
Deflection						
R_{load}	Deflection Load Resistance	EW, VERT, VERTQ	–	6.8	–	kΩ
C_{load}	Deflection Load Capacitance		–	68	–	nF

6.7.2.3. Recommended Crystal Characteristics

Symbol	Parameter	Min.	Typ.	Max.	Unit
T_A	Operating Ambient Temperature	0	–	65	°C
f_P	Parallel Resonance Frequency with Load Capacitance **C_L = 13 pF**	–	20.250000	–	MHz
$\Delta f_P/f_P$	Accuracy of Adjustment	–	–	±20	ppm
$\Delta f_P/f_P$	Frequency Temperature Drift	–	–	±30	ppm
R_R	Series Resistance	–	–	25	Ω
C_0	Shunt Capacitance	3	–	7	pF
C_1	Motional Capacitance	20	–	30	fF
Load Capacitance Recommendation					
C_{Lext}	External Load Capacitance [1] from pins to Ground (pin names: Xtal1 Xtal2)	–	3.3	–	pF
DCO Characteristics [2,3]					
$C_{ICLoadmin}$	Effective Load Capacitance @ min. DCO–Position, Code 0, package: 64PSDIP	3	4.3	5.5	pF
$C_{ICLoadrng}$	Effective Load Capacitance Range, DCO Codes from 0..255	11	12.7	15	pF

[1] **Remarks on defining the External Load Capacitance:**

External capacitors at each crystal pin to ground are required. They are necessary to tune the effective load capacitance of the PCBs to the required load capacitance C_L of the crystal. The higher the capacitors, the lower the clock frequency results. The nominal free running frequency should match f_p MHz. Due to different layouts of customer PCBs the matching capacitor size should be determined in the application. The suggested value is a figure based on experience with various PCB layouts.
Tuning condition: Code DVCO Register=–720

[2] **Remarks on Pulling Range of DCO:**

The pulling range of the DCO is a function of the used crystal and effective load capacitance of the IC ($C_{ICLoad} + C_{LoadBoard}$). The resulting frequency f_L with an effective load capacitance of $C_{Leff} = C_{ICLoad} + C_{LoadBoard}$ is:

$$f_L = f_P * \frac{1 + 0.5 * [\, C_1 / (C_0 + C_{Leff}) \,]}{1 + 0.5 * [\, C_1 / (C_0 + C_L) \,]}$$

[3] **Remarks on DCO codes**

The DCO hardware register has 8 bits, the fp control register uses a range of –2048...2047

6.7.3. Characteristics

If not otherwise designated under test conditions, all characteristics are specified for recommended operating conditions (see Section 6.7.2. on page 150).

6.7.3.1. General Characteristics

Symbol	Parameter	Pin Name	Min.	Typ.	Max.	Unit	Test Conditions
P_{TOT}	Total Power Dissipation		–	850	1350	mW	
P_{STDBY}	Standby Power Dissipation	$VSUP_S$	–	tbd	tbd	mW	$VSUP_D = VSUP_P =$ $VSUP_{AF} = VSUP_{AB} =$ $VSUP_{DB} = VSUP_{ADB} = 0V$ SR0 = SR1 = SR2 = 0
I_{VSUPS}	Current Consumption Standby Mode		–	tbd	tbd	mA	
I_{VSUPS}	Current Consumption Standby Supply		–	15	23	mA	normal operation
I_{VSUPD}	Current Consumption Digital Circuitry	$VSUP_D$	–	55	83	mA	
I_{VSUPP}	Current Consumption Port Circuitry	$VSUP_P$	–	–	–	mA	depends on port load
I_{VSUPAF}	Current Consumption Analog Front-end	$VSUP_{AF}$	–	48	72	mA	
I_{VSUPAB}	Current Consumption Analog Back-end	$VSUP_{AB}$	–	50	100	mA	depends on contrast and brightness settings
I_L	Input and Output Leakage Current	All I/O Pins	–	–	1	μA	

6.7.3.2. Test Input

Symbol	Parameter	Pin Name	Min.	Typ.	Max.	Unit	Test Conditions
V_{IL}	Input Low Voltage	TEST	–	–	0.8	V	
V_{IH}	Input High Voltage		2.0	–	–	V	

6.7.3.3. Reset Input

Symbol	Parameter	Pin Name	Min.	Typ.	Max.	Unit	Test Conditions
VBG	Internal Reference Voltage	RESQ	1.125	1.25	1.375	V	
tBG	Internal Voltage Reference Setup Time after Power-Up		–	30	60	us	
VREFR	RESET Comparator Reference Voltage		–	1*VBG	–	V	
RVlh–RVhl	RESET Comparator Hysteresis, symmetrical to VREFR		0.25	0.313	0.375	V	
VREFA	ALARM Comparator Reference Voltage		–	2*VBG	–	V	
AVlh–AVhl	ALARM Comparator Hysteresis, symmetrical to VREFA		0.1	0.135	0.15	V	
tCDEL	RESET, ALARM Comparator Delay Time		–	–	100	ns	Overdrive=50mV
VREFPOR	Power On Reset Comparator Reference Voltage	$VSUP_S$	–	2*VBG	–	V	

6.7.3.4. I^2C Bus Interface

Symbol	Parameter	Pin Name	Min.	Typ.	Max.	Unit	Test Conditions
V_{IL}	Input Low Voltage	SDA, SCL	–	–	0.3* $VSUP_S$	V	
V_{IH}	Input High Voltage		0.6* $VSUP_S$	–	–	V	
V_{OL}	Output Low Voltage		–	–	0.4 0.6	V V	I_l = 3 mA I_l = 6 mA
V_{IH}	Input Capacitance		–	–	5	pF	
t_F	Signal Fall Time		–	–	300	ns	C_L = 400 pF
t_R	Signal Rise Time		–	–	300	ns	C_L = 400 pF
f_{SCL}	Clock Frequency	SCL	0	–	400	kHz	
t_{LOW}	Low Period of SCL		1.3	–	–	µs	
t_{HIGH}	High Period of SCL		0.6	–	–	µs	
$t_{SU\ Data}$	Data Set Up Time to SCL high	SDA	100	–	–	ns	
$t_{HD\ Data}$	DATA Hold Time to SCL low		0	–	0.9	µs	

6.7.3.5. 20-MHz Clock Output

Symbol	Parameter	Pin Name	Min.	Typ.	Max.	Unit	Test Conditions
V_{OL}	Output Low Voltage	CLK20	–	–	0.4	V	I_{OL} = tbd, strength tbd
V_{OH}	Output High Voltage		$VSUP_D$ – 0.4	–	$VSUP_D$	V	$-I_{OL}$ = tbd, strength tbd

6.7.3.6. Analog Video Output

Symbol	Parameter	Pin Name	Min.	Typ.	Max.	Unit	Test Conditions
V_{OUT}	Output Voltage	VOUT	1.7	2.0	2.3	V_{PP}	V_{VIN} = 1 V_{PP}, AGC = 0 dB
AGC_{VOUT}	AGC Step Width, VOUT			1.333		dB	3 Bit Resolution = 7 Steps 3 MSB's of main AGC
DNL_{AGC}	AGC Differential Non-Linearity				±0.5	LSB	
V_{OUTDC}	DC-Level			1		V	clamped to back porch
BW	V_{OUT} Bandwidth		6	10	–	MHz	Input: –2 dBr of main ADC range, $C_L \leq 10$ pF
THD	V_{OUT} Total Harmonic Distortion			–45	–40	dB	Input: –2 dBr of main ADC range, $C_L \leq 10$ pF 1 MHz, 5 Harmonics
C_{LVOUT}	Load Capacitance		–	–	10	pF	
I_{LVOUT}	Output Current		–	–	±0.1	mA	

6.7.3.7. A/D Converter Reference

Symbol	Parameter	Pin Name	Min.	Typ.	Max.	Unit	Test Conditions
V_{VRT}	Reference Voltage Top	VRT	2.5	2.6	2.8	V	10 µF/10 nF, 1 GΩ Probe
V_{VRTN}	Reference Voltage Top Noise		–	–	100	mV_{PP}	

6.7.3.8. Analog Video Front-End and A/D Converters

Symbol	Parameter	Pin Name	Min.	Typ.	Max.	Unit	Test Conditions
Luma – Path (Composite)							
R_{VIN}	Input Resistance	VIN1–4	1	–	–	MΩ	Code Clamp–DAC=0
C_{VIN}	Input Capacitance		–	5	–	pF	
V_{VIN}	Full Scale Input Voltage		1.8	2.0	2.2	V_{PP}	min. AGC Gain
V_{VIN}	Full Scale Input Voltage		0.5	0.6	0.7	V_{PP}	max. AGC Gain
AGC	AGC step width		–	0.166	–	dB	6-Bit Resolution= 64 Steps f_{sig}=1MHz, – 2 dBr of max. AGC–Gain
DNL_{AGC}	AGC Differential Non-Linearity		–	–	±0.5	LSB	
V_{VINCL}	Input Clamping Level, CVBS		–	1.0	–	V	Binary Level = 64 LSB min. AGC Gain
Q_{CL}	Clamping DAC Resolution		–16	–	15	steps	5 Bit – I–DAC, bipolar V_{VIN}=1.5 V
I_{CL-LSB}	Input Clamping Current per step		0.7	1.0	1.3	μA	
DNL_{ICL}	Clamping DAC Differential Non-Linearity		–	–	±0.5	LSB	
Chroma – Path (Composite)							
R_{CIN}	Input Resistance SVHS Chroma	CIN1 CIN2	1.4	2.0	2.6	kΩ	
V_{CIN}	Full Scale Input Voltage, Chroma		1.08	1.2	1.32	V_{PP}	
V_{CINDC}	Input Bias Level, SVHS Chroma		–	1.5	–	V	
	Binary Code for Open Chroma Input		–	128	–	–	
Chroma – Path (Component)							
R_{CIN}	Input Resistance	CRIN CBIN	1	–	–	MΩ	Code Clamp–DAC=0
C_{CIN}	Input Capacitance		–	–	4.5	pF	
V_{CIN}	Full Scale Input Voltage		0.76	0,84	0.92	V_{PP}	minimal Range
V_{CIN}	Full Scale Input Voltage		1.08	1.2	1.32	V_{PP}	extended Range
V_{CINCL}	Input Clamping Level C_r, C_b		–	1.5	–	V	Binary Level = 128 LSB
Q_{CL}	Clamping DAC Resolution		–32	–	31	steps	6 Bit – I–DAC, bipolar V_{VIN}=1.5 V
I_{CL-LSB}	Input Clamping Current per step		0.59	0.85	1.11	μA	
DNL_{ICL}	Clamping DAC Differential Non-Linearity		–	–	±0.5	LSB	

Symbol	Parameter	Pin Name	Min.	Typ.	Max.	Unit	Test Conditions
Dynamic Characteristics for all Video-Paths (Luma + Chroma)							
BW	Bandwidth	VIN1–4 CIN1–2 CBIN	8	10	–	MHz	–2 dBr input signal level
XTALK	Crosstalk, any Two Video Inputs		–	–56	–	dB	1 MHz, –2 dBr signal level
THD	Total Harmonic Distortion		–	50	–	dB	1 MHz, 5 harmonics, –2 dBr signal level
SINAD	Signal-to-Noise and Distortion Ratio		–	45	–	dB	1 MHz, all outputs, –2 dBr signal level
INL	Integral Non-Linearity		–	–	±1	LSB	Code Density, DC-ramp
DNL	Differential Non-Linearity		–	–	±0.8	LSB	
DG	Differential Gain		–	–	±3	%	–12 dBr, 4.4 MHz signal on DC-ramp
DP	Differential Phase		–	–	1.5	deg	

6.7.3.9. Analog RGB and FB Inputs

Symbol	Parameter	Pin Name	Min.	Typ.	Max.	Unit	Test Conditions
RGB Input Characteristics							
V_{RGBIN}	External RGB Inputs Voltage Range	RIN GIN BIN	–0.3	–	1.1	V	
V_{RGBIN}	nominal RGB Input Voltage peak-to-peak		0.5	0.7	1.0	V_{PP}	SCART Spec: 0.7V ±3dB
V_{RGBIN}	RGB Inputs Voltage for Maximum Output Current		–	0.44	–	V	Contrast setting: 511
			–	0.7	–	V	Contrast setting: 323
			–	1.1	–	V	Contrast setting: 204
t_{CLP}	Clamp Pulse Width		1.6	–	–	µs	
C_{IN}	Input Capacitance		–	–	13	pF	
I_{IL}	Input Leakage Current		–0.5	–	0.5	µA	Clamping OFF, V_{IN} = –0.3...3 V
V_{CLIP}	RGB Input Voltage for Clipping Current		–	2	–	V	
V_{CLAMP}	Clamp Level at Input		40	60	80	mV	Clamping ON
V_{INOFF}	Offset Level at Input		–10	–	10	mV	Extrapolated from V_{IN} = 100 and 200 mV
V_{INOFF}	Offset Level Match at Input		–10	–	10	mV	Extrapolated from V_{IN} = 100 and 200 mV
R_{CLAMP}	Clamping-ON-Resistance		–	140	–	Ω	
Fast Blank Input Characteristics							
V_{FBLOFF}	FBLIN Low Level	FBLIN	–	–	0.5	V	
V_{FBLON}	FBLIN High Level		0.9	–	–	V	
$V_{FBLTRIG}$	Fast Blanking Trigger Level typical		–	0.7	–	V	
t_{PID}	Delay Fast Blanking to RGB_{OUT} from midst of FBLIN–transition to 90% of RGB_{OUT} transition		–	8	15	ns	Internal RGB = 3.75 mA Full Scale Int. Brightness = 0 External Brightness = 1.5 mA (Full Scale) RGBin = 0 V_{FBLOFF} = 0.4 V V_{FBLON} = 1.0 V Rise and fall time = 2 ns
	Difference of Internal Delay to External RGBin Delay		–5	–	+5	ns	
	Switch-Over-Glitch		–	0.5	–	pAs	Switch from 3.75 mA (int) to 1.5 mA (ext)

6.7.3.10. Horizontal Flyback Input

Symbol	Parameter	Pin Name	Min.	Typ.	Max.	Unit	Test Conditions
V_{IL}	Input Low Voltage	HFLB	–	–	1.8	V	
V_{IH}	Input High Voltage		2.6	–	–	V	
V_{IHST}	Input Hysteresis		0.1	–	–	V	
$PSRR_{HF}$	Power Supply Rejection Ratio of Trigger Level		0	–	–	dB	f = 20 MHz
$PSRR_{MF}$	Power Supply Rejection Ratio of Trigger Level		–20	–	–	dB	f < 15 kHz
$PSRR_{LF}$	Power Supply Rejection Ratio of Trigger Level		–40	–	–	dB	f < 100 Hz

6.7.3.11. Horizontal Drive Output

Symbol	Parameter	Pin Name	Min.	Typ.	Max.	Unit	Test Conditions
V_{OL}	Output Low Voltage	HOUT	–	–	0.4	V	I_{OL} = 10 mA
V_{OH}	Output High Voltage (Open Drain Stage)		–	–	5	V	external pull-up resistor
t_{OF}	Output Fall Time		–	8	20	ns	C_{LOAD} = 30pF
I_{OL}	Output Low Current		–	–	10	mA	

6.7.3.12. Vertical Safety Input

Symbol	Parameter	Pin Name	Min.	Typ.	Max.	Unit	Test Conditions
V_{ILA}	Input Low Voltage A	SAFETY	–	–	1.8	V	
V_{IHA}	Input High Voltage A		2.6	–	–	V	
V_{ILB}	Input Low Voltage B		–	–	3.0	V	
V_{IHB}	Input High Voltage B		3.8	–	–	V	
V_{IHST}	Input Hysteresis A and B		0.1	–	–	V	
t_{PID}	Internal Delay		–	–	100	ns	

6.7.3.13. Vertical Protection Input

Symbol	Parameter	Pin Name	Min.	Typ.	Max.	Unit	Test Conditions
V_{IL}	Input Low Voltage	VPROT	–	–	1.8	V	
V_{IH}	Input High Voltage		2.6	–	–	V	
V_{IHST}	Input Hysteresis		0.1	–	–	V	

6.7.3.14. Vertical and East/West D/A Converter Output

Symbol	Parameter	Pin Name	Min.	Typ.	Max.	Unit	Test Conditions
V_{OMIN}	Minimum Output Voltage	EW VERT VERTQ	–	0	–	V	R_{load} = 6.8 kΩ R_{xref} = 10 kΩ
V_{OMAX}	Maximum Output Voltage		2.82	3	3.2	V	R_{load} = 6.8 kΩ R_{xref} = 10 kΩ
I_{DACN}	Full scale DAC Output Current		415	440	465	µA	R_{xref} = 10 kΩ
PSRR	Power Supply Rejection Ratio		–	20	–	dB	

6.7.3.15. Interlace Output

Symbol	Parameter	Pin Name	Min.	Typ.	Max.	Unit	Test Conditions
V_{OL}	Output Low Voltage	INTLC	–	0.2	0.4	V	I_{OL} = 1.6 mA
V_{OH}	Output High Voltage		VSUP_{AB} – 0.4	VSUP_{AB} – 0.2	–	V	$-I_{OL}$ = 1.6mA

6.7.3.16. Sense A/D Converter Input

Symbol	Parameter	Pin Name	Min.	Typ.	Max.	Unit	Test Conditions
V_I	Input Voltage Range	SENSE	0	–	VSUP_{AB}	V	
V_{I255}	Input Voltage for code 255		1.4	1.54	1.7	V	Read cutoff blue register
C_0	Digital Output for zero Input		–	–	16	LSB	Offset check, read cutoff blue register
R_I	Input Impedance		1	–	–	MΩ	

6.7.3.17. Range Switch Output

Symbol	Parameter	Pin Name	Min.	Typ.	Max.	Unit	Test Conditions
R_{ON}	Output On Resistance	RSW1 RSW2	–	–	50	Ω	I_{OL} = 10 mA
I_{Max}	Maximum Current		–	–	15	mA	
I_{LEAK}	Leakage Current		–	–	600	nA	RSW High Impedance
C_{IN}	Input Capacitance		–	–	5	pF	

6.7.3.18. D/A Converter Reference

Symbol	Parameter	Pin Name	Min.	Typ.	Max.	Unit	Test Conditions
V_{DACREF}	DAC-Ref. Voltage	VRD	2.38	2.50	2.67	V	
V_{DACR}	DAC-Ref. Output resistance		18	25	32	kΩ	
V_{XREF}	DAC-Ref. Voltage Bias Current Generation	XREF	2.38	2.5	2.67	V	R_{xref} = 10 kΩ

6.7.3.19. Analog RGB Outputs, D/A Converters

Symbol	Parameter	Pin Name	Min.	Typ.	Max.	Unit	Test Conditions
Internal RGB Signal D/A Converter Characteristics							
	Resolution	ROUT GOUT BOUT	–	10	–	bit	
I_{OUT}	Full Scale Output Current		3.6	3.75	3.9	mA	R_{ref} = 10 kΩ
I_{OUT}	Differential Nonlinearity		–	–	0.5	LSB	
I_{OUT}	Integral Nonlinearity		–	–	1	LSB	
I_{OUT}	Glitch Pulse Charge		–	0.5	–	pAs	Ramp signal, 25 Ω output termination
I_{OUT}	Rise and Fall Time		–	3	–	ns	10% to 90%, 90% to 10%
I_{OUT}	Intermodulation		–	–	–50	dB	2/2.5MHz full scale
I_{OUT}	Signal to Noise		+50	–	–	dB	Signal: 1MHz full scale Bandwidth: 10MHz
I_{OUT}	Matching R-G, R-B, G-B		–2	–	2	%	
	R/B/G Crosstalk one channel talks two channels talk		–	–	–46	dB	Passive channel: I_{OUT} =1.88 mA Crosstalk-Signal: 1.25 MHz, 3.75 mA_{PP}
	RGB Input Crosstalk from external RGB one channel talks two channels talk three channels talk		 – – –	 – – –	 –50 –50 –50	 dB dB dB	
Internal RGB Brightness D/A Converter Characteristics							
	Resolution	ROUT GOUT BOUT	–	9	–	bit	
I_{BR}	Full Scale Output Current relative		39.2	40	40.8	%	Ref to max. digital RGB
I_{BR}	Full Scale Output Current absolute		–	1.5	–	mA	
I_{BR}	Differential Nonlinearity		–	–	1	LSB	
I_{BR}	Integral Nonlinearity		–	–	2	LSB	
I_{BR}	Match R-G, R-B, G-B		–2	–	2	%	
I_{BR}	Match to digital RGB R-R, G-G, B-B		–2	–	2	%	

Symbol	Parameter	Pin Name	Min.	Typ.	Max.	Unit	Test Conditions
External RGB Voltage/Current Converter Characteristics							
	Resolution	ROUT GOUT BOUT	–	9	–	bit	
I_{EXOUT}	Full Scale Output Current relative		96	100	104	%	Ref. to max. Digital RGB V_{IN} = 0.7 $V_{PP,}$ contrast = 323
	Full Scale Output Current absolute		–	3.75	–	mA	Same as Digital RGB
CR	Contrast Adjust Range		–	16:511	–		
	Gain Match R-G, R-B, G-B		–3	–	3	%	Measured at RGB Outputs V_{IN} = 0.7 V, contrast = 323
	Gain Match to RGB-DACs R-R, G-G, B-B		–3	–	3	%	Measured at RGB Outputs V_{IN} = 0.7 V, contrast = 323
	R/B/G Input Crosstalk one channel talks two channels talk		–	–	–46	dB	Passive channel: V_{IN} = 0.7V, contrast = 323 Crosstalk signal: 1.25 MHz, 3.75 mA_{PP}
	RGB Input Crosstalk from Internal RGB one channel talks two channels talk tree channels talk		–	–	–50	dB	
	RGB Input Noise and Distortion		–	–	–50	dB	V_{IN}=0.7 V_{PP} at 1 MHz contrast = 323 Bandwidth: 10 MHz
	RGB Input Bandwidth -3dB			15	–	MHz	V_{IN} = 0.7 $V_{PP,}$ contrast =323
	RGB Input THD		– –	–50 –40	– –	dB dB	Input signal 1 MHz Input signal 6 MHz V_{IN} = 0.7 V_{PP} contrast =323
	Differential Nonlinearity of Contrast Adjust		–	–	1	LSB	V_{IN} = 0.44V
	Integral Nonlinearity of Contrast Adjust		–	–	2	LSB	
V_{RGBO}	R,G,B Output Voltage		–1.0	–	0.3	V	Referred to V_{SUPO}
	R,G,B Output Load Resistance		–	–	100	Ω	Ref. to V_{SUPO}
V_{OUTC}	RGB Output Compliance		–1.5	–1.3	–1.2	V	Ref. to V_{SUPO} Sum of max. Current of RGB-DACs and max. Current of Int. Brightness DACs is 2% degraded

Symbol	Parameter	Pin Name	Min.	Typ.	Max.	Unit	Test Conditions
External RGB Brightness D/A Converter Characteristics							
	Resolution	ROUT GOUT BOUT	–	9	–	bit	
I_{EXBR}	Full Scale Output Current relative		39.2	40	40.8	%	Ref to max. digital RGB
	Full Scale Output Current absolute		–	1.5	–	mA	
	Differential Nonlinearity		–	–	1	LSB	
	Integral Nonlinearity		–	–	2	LSB	
	Matching R-G, R-B, G-B		–2	–	2	%	
	Matching to digital RGB R-R, G-G, B-B		–2	–	2	%	
RGB Output Cutoff D/A Converter Characteristics							
	Resolution	ROUT GOUT BOUT	–	9	–	bit	
I_{CUT}	Full Scale Output Current relative		58.8	60	61.2	%	Ref to max. digital RGB
	Full Scale Output Current absolute		–	2.25	–	mA	
	Differential nonlinearity		–	–	1	LSB	
	Integral nonlinearity		–	–	2	LSB	
	Matching to digital RGB R-R, G-G, B-B		–2	–	2	%	
RGB Output Ultrablack D/A Converter Characteristics							
	Resolution	ROUT GOUT BOUT	–	1	–	bit	
I_{UB}	Full Scale Output Current relative		19.6	20	20.4	%	Ref to max. digital RGB
	Full Scale Output Current absolute		–	0.75	–	mA	
	Match to digital RGB R–R, G–G, B–B		–2	–	2	%	

6.7.3.20. Scan Velocity Modulation Output

Symbol	Parameter	Pin Name	Min.	Typ.	Max.	Unit	Test Conditions
	Resolution	SVMOUT	–	8	–	bit	
I_{OUT}	Full Scale Output Current		1.55	1.875	2.25	mA	
I_{OUT}	Differential Nonlinearity		–	–	0.5	LSB	
I_{OUT}	Integral Nonlinearity		–	–	1	LSB	
I_{OUT}	Glitch Pulse Charge		–	0.5	–	pAs	Ramp, output line is terminated on both ends with 50 Ω
I_{OUT}	Rise and Fall Time		–	3	–	ns	10% to 90%, 90% to 10%

6.7.3.21. Analog Audio Inputs and Outputs

Symbol	Parameter	Pin Name	Min.	Typ.	Max.	Unit	Test Conditions
R_{AIN}	Audio Input Resistance	AIN1-3 AOUT1-2	25	40	58	kΩ	f_{signal} = 1 kHz, I = 0.05 mA
dV_{AOUT}	Deviation of DC-Level at Audio Output from GND_{AB} Voltage		–20	–	+20	mV	
A_{Audio}	Gain from Audio Input to Audio Output		–1.0	0.0	+0.5	dB	f_{signal} = 1 kHz AVOL = 0dB
f_{rAudio}	Frequency Response from Audio Input to Audio Output bandwidth: 50 Hz to 15000 Hz		–0.5	0.0	+0.5	dB	with resp. to 1 kHz
PSRR	Power Supply Rejection Ratio for Audio Output		tbd	50	–	dB	1 kHz sine at 100 mV_{RMS}
			tbd	20	–	dB	≤ 100 kHz sine at 100 mV_{RMS}
V_{NOISE}	Noise Output Voltage		–	tbd	20	µV	R_{GEN} = 1kΩ, equally weighted 50 Hz...15 kHz
V_{Mute}	Mute Output Voltage		–	tbd	2	µV	AVOL = mute, equally weighted 50 Hz...15 kHz
THD	Total Harmonic Distortion from Audio Input to Audio Output		–	–	0.1	%	Input Level = 0.7V_{RMS}, f_{sig} = 1 kHz, equally weighted 50 Hz...15 kHz
XTALK	Crosstalk attenuation between Audio Input and Audio Output		70	–	–	dB	Input Level = 0.7V_{RMS}, f_{sig} = 1 kHz, equally weighted 50 Hz...15 kHz, unused analog inputs connected to ground by Z < 1 kΩ

6.7.3.22. ADC Input Port

Symbol	Parameter	Pin Name	Min.	Typ.	Max.	Unit	Test Conditions
VREF	ADC Comparator Reference Voltage	Px.y	–	0.5* VSUP$_S$	–	V	
Vlh– Vhl	ADC Comparator Hysteresis, symmetrical to VREF		0.1	0.17	0.24	V	
tCDEL	ADC Comparator Delay Time		–	–	100	ns	Overdrive=50mV
LSB	LSB Value		–	VSUP$_S$ /1024	–	V	
R	Conversion Range		GND$_S$	–	VSUP$_S$	V	
A	Conversion Result		–	INT (Vin/ LSB)	–	hex	GND$_S$<Vin<VSUP$_S$
			000	–	–	hex	Vin<=GND$_S$
			–	–	3FF	hex	Vin>=VSUP$_S$
tc	Conversion Time		–	4	–	µs	
ts	Sample Time		–	2	–	υs	
TUE	Total Unadjusted Error		–6	–	6	LSB	
DNL	Differential Non-Linearity		–3	–	3	LSB	
Ci	Input Capacitance during Sampling Period		–	15	–	pF	
Ri	Serial Input Resistance during Sampling Period		–	5	–	kΩ	

6.7.3.23. Universal Port

Symbol	Parameter	Pin Name	Min.	Typ.	Max.	Unit	Test Conditions
Vol	Output Low Voltage	P1x P2x P3x	–	–	0.4 1.0	V	Io=3mA Io=10mA
Voh	Output High Voltage		VSUP$_P$–0.4 VSUP$_P$–1.0	–	–	V	Io=–3mA Io=–10mA
V_{IL}	Input Low Voltage	P1x P2x P3x P42–P46	–	–	0.8	V	
V_{IH}	Input High Voltage		2.0	–	–	V	
Ii	Input Leakage Current		–0.1	–	0.1	µA	0<Vi<VSUP$_P$

6.7.3.24. Memory Port

Symbol	Parameter	Pin Name	Min.	Typ.	Max.	Unit	Test Conditions
t_{CYC}	PH2 Cycle Time		-	98.77	-	ns	
t_{ADS}	Address Setup Time	ADB[18-0]	-	15 + 0.5 20 30	19 + 0.7 26 40	ns ns/pf ns ns	C_{ADB} = 0 pF C_{ADB} = 10 pF C_{ADB} = 30 pF
t_{ADH}	Address Hold Time		-	8	10	ns	C_{ADB} = 10 pF
t_{DWS}	Data Write Setup Time	DB[7:0]	-	9 + 0.5 14 24	14 + 0.7 21 35	ns ns/ pF ns ns	C_{DB} = 0 pF C_{DB} = 10 pF C_{DB} = 30 pF
t_{DWH}	Data Write Hold Time		-	6	8	ns	C_{DB} = 0 pF
t_{DRS}	Data Read Setup Time		12	-	-	ns	
t_{DRH}	Data Read Hold Time		6	-	-	ns	
t_{ENS}	Enable Setup Time	OE1Q OE2Q WE1Q WE2Q	-	6	10	ns	$C_{OEQ,WEQ}$ = 0 pF
t_{ENH}	Enable Hold Time		-	6	9	ns	$C_{OEQ,WEQ}$ = 0 pF

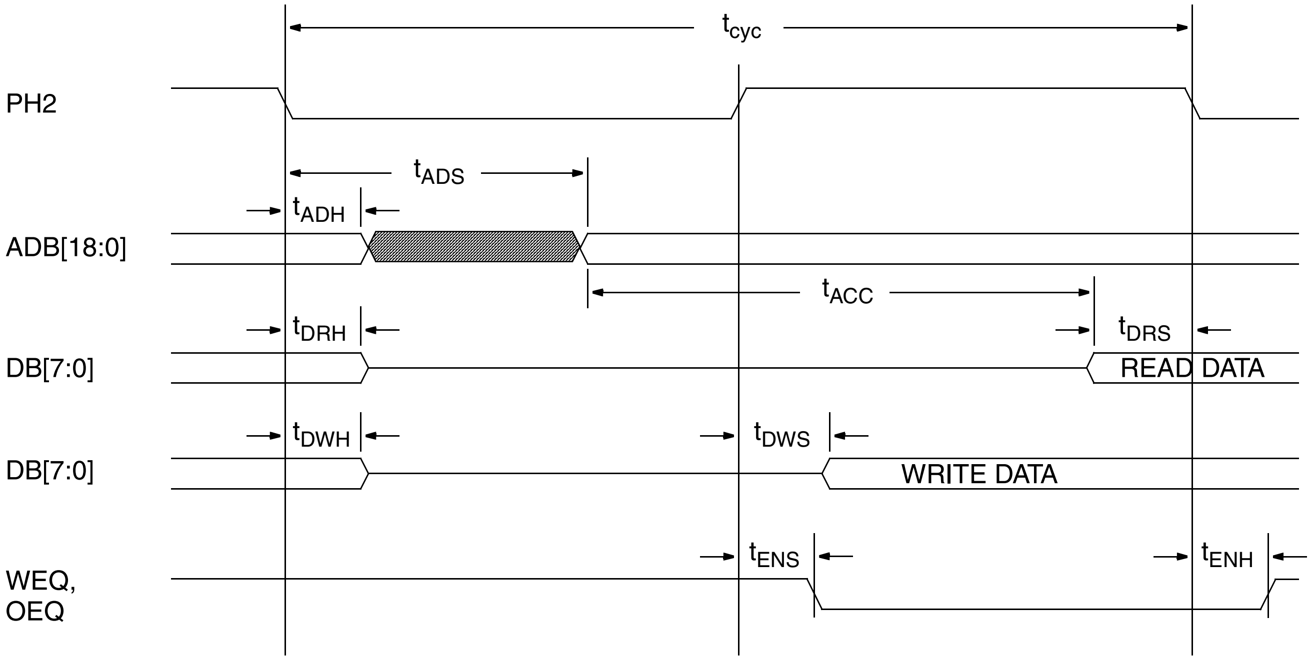


Fig. 6–33: Memory port timing

7. Application

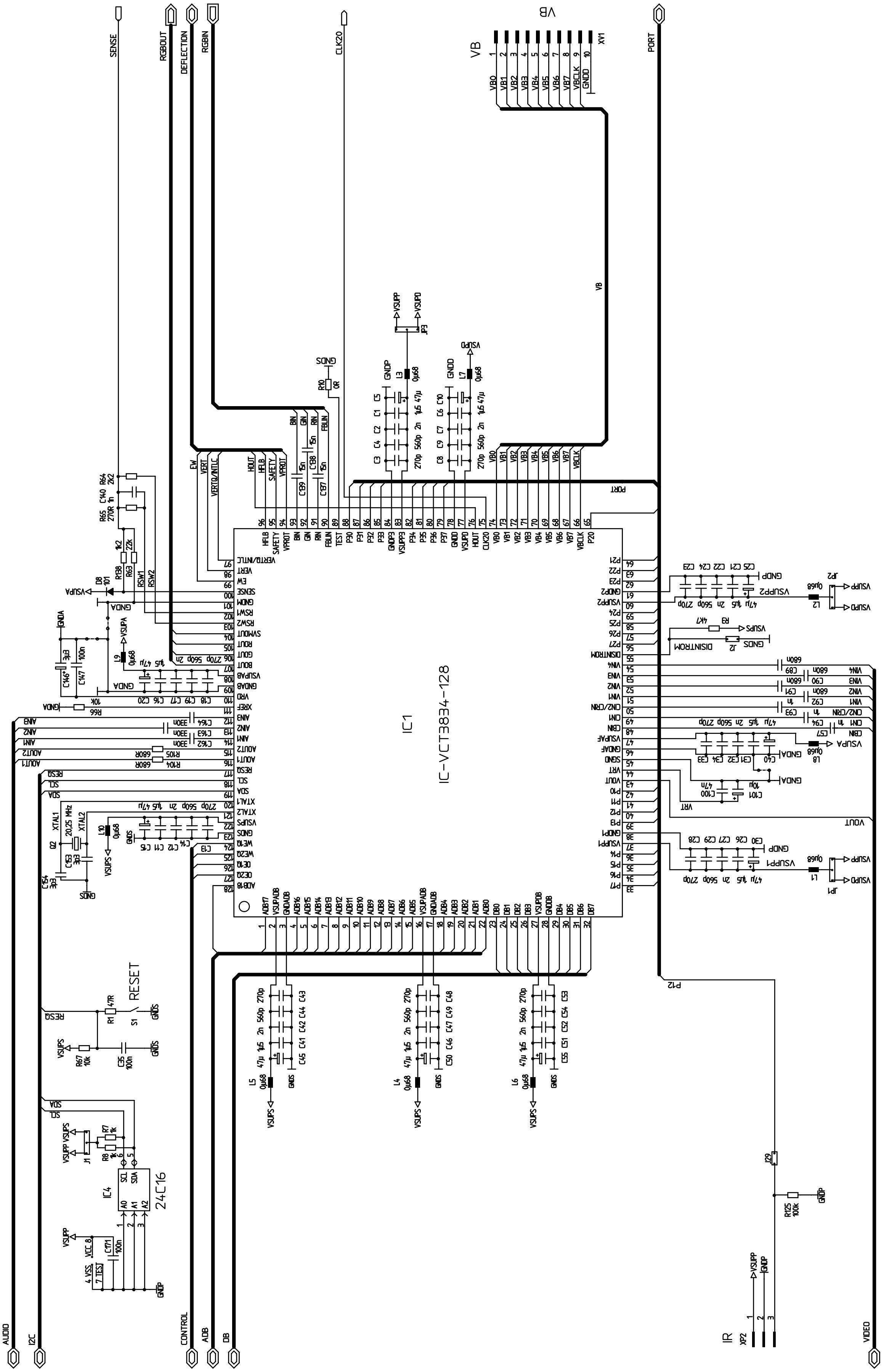


Fig. 7–1: VCT 38xxA application circuit, part 1/3

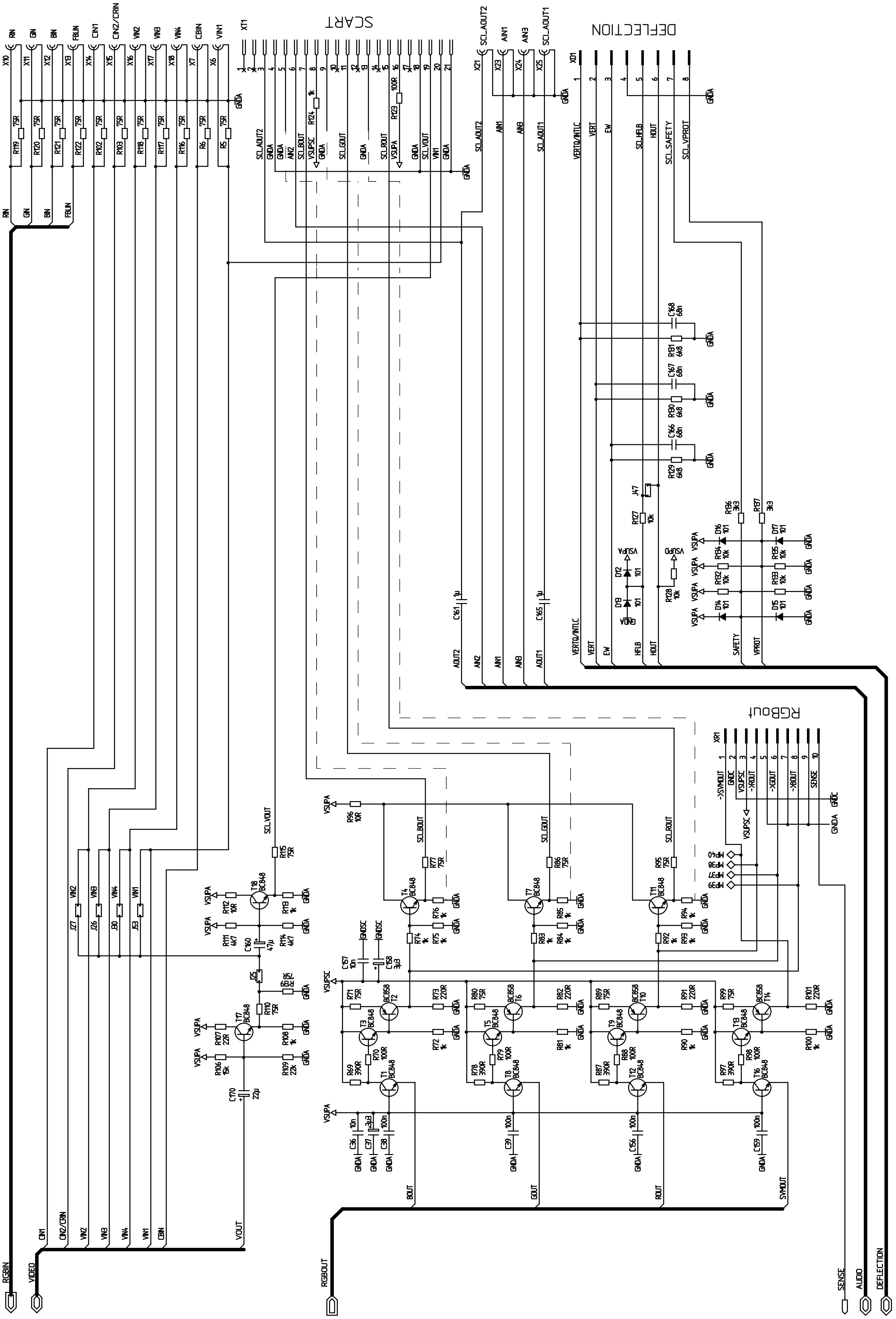


Fig. 7–2: VCT 38xxA application circuit, part 2/3

MEMORY

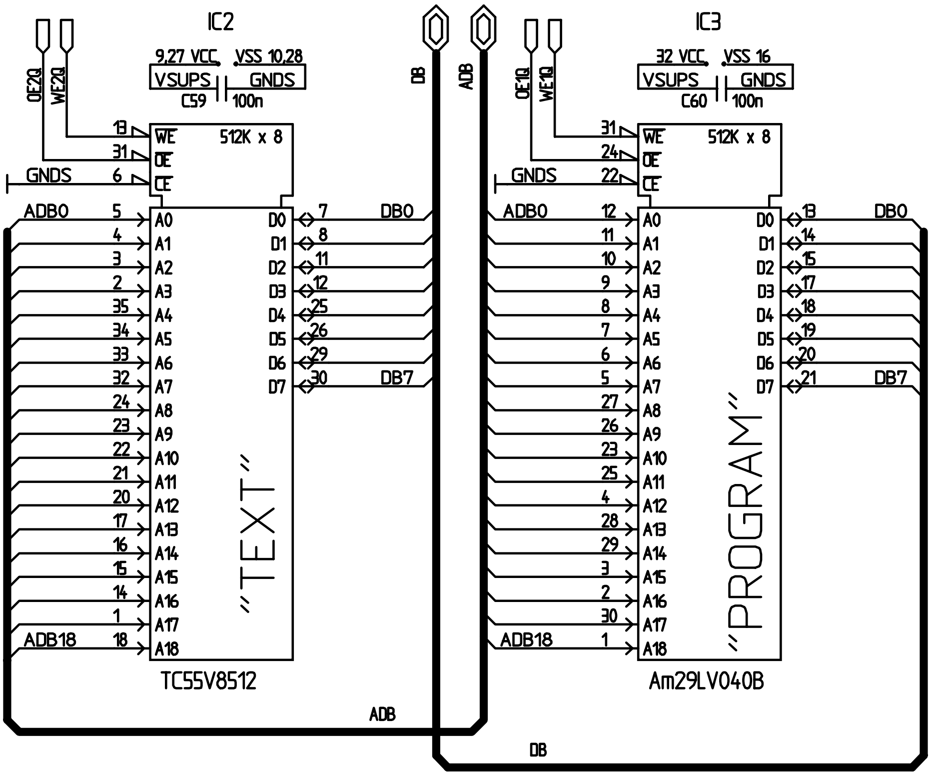


Fig. 7–3: VCT 38xxA application circuit, part 3/3

8. Glossary of Abbreviations

AIT	Additional Information Table
BTT	Basic TOP Table
BTTL	Basic TOP Table List
CCU	Central Control Unit
CLUT	Color Look Up Table
CPU	Central Processing Unit
CRI	Clock Running-in
DMA	Direct Memory Access
DRAM	Dynamic Random Access Memory
FLOF	Full Level One Features
FRC	Framing Code
MPT	Multipage Table
MPET	Multipage Extension Table
NMI	Non Maskable Interrupt
OSD	On Screen Display
PDC	Programme Delivery Control
PLT	Page Linking Table
RAM	Random Access Memory
ROM	Read Only Memory
SRAM	Static Random Access Memory
TOP	Table Of Pages
TPU	Teletext Processing Unit
TTX	Teletext
VBI	Vertical Blanking Interval
VPS	Video-Programm-System
WSS	Wide Screen Signalling
WST	World System Teletext

9. References

1. Preliminary Data Sheet: “VDP 31xxB”,
Sept. 25, 1998
6251-437-2PD

2. Preliminary Data Sheet: “TPU 3035, TPU 3040, TPU 3050”,
Feb. 23, 1999
6251-349-6PD

3. Preliminary Data Sheet: “W65C02”,
Oct. 2, 1991
6251-364-1PD

4. “Enhanced Teletext Specification”. European Telecommunication Standard ETS 300 706. ETSI, May1997.

5. “Television systems; 625-line television Wide Screen Signalling (WSS)”. European Telecommunication Standard ETS 300 294. ETSI, May1996.

6. “Television systems; Specification of the domestic video Programme Delivery Control system (PDC)”. European Telecommunication Standard ETS 300 231. ETSI, August1996.

7. “Electronic Programme Guide (EPG)”. European Telecommunication Standard ETS 300 707. ETSI, May1997.

10. Data Sheet History

1. Advance Information: “VCT 38xxA Video/Controller/Teletext IC Family”, Edition Oct. 17, 2000, 6251-518-1AI. First release of the advance information.

Micronas GmbH
Hans-Bunte-Strasse 19
D-79108 Freiburg (Germany)
P.O. Box 840
D-79008 Freiburg (Germany)
Tel. +49-761-517-0
Fax +49-761-517-2174
E-mail: docservice@micronas.com
Internet: www.micronas.com

Printed in Germany
Order No. 6251-518-1AI

All information and data contained in this data sheet are without any commitment, are not to be considered as an offer for conclusion of a contract, nor shall they be construed as to create any liability. Any new issue of this data sheet invalidates previous issues. Product availability and delivery are exclusively subject to our respective order confirmation form; the same applies to orders based on development samples delivered. By this publication, Micronas GmbH does not assume responsibility for patent infringements or other rights of third parties which may result from its use.
Further, Micronas GmbH reserves the right to revise this publication and to make changes to its content, at any time, without obligation to notify any person or entity of such revisions or changes.
No part of this publication may be reproduced, photocopied, stored on a retrieval system, or transmitted without the express written consent of Micronas GmbH.