AL300 Data Sheets

Version 2.0

Amendments (Since April 2, 1999)

04.02.99 "Section 6.13 Zoom" is added to the document.

04.15.99 Pin 131 is redefined as a GND pin instead of a NC pin. Sections 4.0 and 5.0 are modified accordingly.

05.11.99 Power consumption data and Programming flowchart are provided.

05.12.99 Registers #06h, 07h, 1Bh, 36h, 4Eh, 50~57h, 66h, 67h, 71h, 85h, 86h, 89h and 8Ah are modified due to the ver. B change. V_{IH} modified.

05.26.99 Pin definition of R/YIN<7:0>, G/UVIN<7:0> and BIN<7:0> is modified to reflect ver. B change.

06.15.99 Pin definition of odd and even output replaced by B and A.

07.02.99 Register table and some description added and updated.

07.13.99 Added more PLL description in section 6.4 and the register definition 10h and 11h.

08.03.99 Definition of registers 80h, 84h and 88h modified.

08.25.99 Definition of registers 06h, 80h modified. Definition of registers 05h, 19h, 1Ah, 3Bh and 3Ch modified. Description of #17h and #18h modified.

9.15.99 Added section 6.14, Initialization, modified section 6.7.

9.29.99 Registers #80h and #82h modified to reflect version B-0 change.

10.5.99 Section 10.0 Power consumption updated

8.17.00 Diagram AL300-11 in section "6.11.2 ROM mode" and definition of register 13h & 35h are modified.

6.27.01 Version 2.0, removed CCIR-656 8-bit interface support from data sheets and add timing diagram

Contents

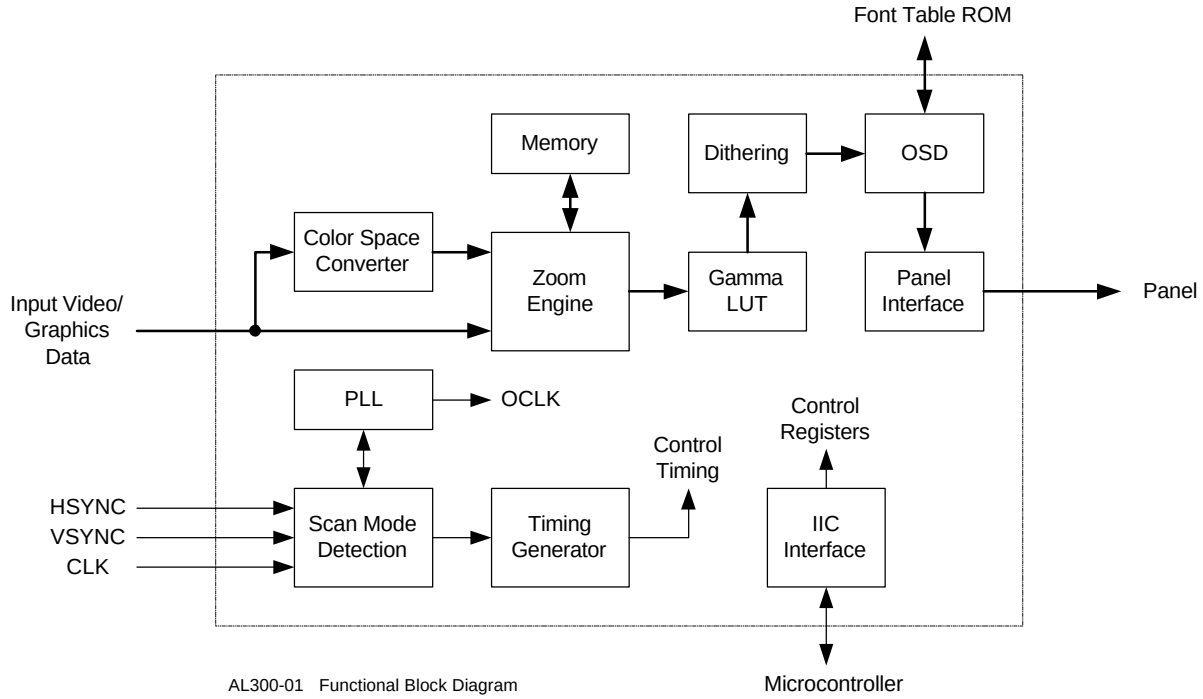
AL300
LCD Monitor Controller

1.0 Features

- Converts PC's or TV's signals for flat panel displays
- Supports active matrix up to 1280x1024 resolution
- De-interlacing support for video inputs
- Automatic screen positioning support
- Fully programmable zoom ratios
- Independent linear zoom in H and V directions
- Supports single and dual pixel per clock panels
- Dithering logic to enhance color resolution for 12-bit or 18-bit panels
- Built-in high speed PLL
- User-definable font table supporting different languages and font sizes
- Two built-in OSD windows
- I^2C programmable
- No external memory required
- Single 3.3 volt power with 5 volt tolerant I/O
- 160-pin 28x28 mm PQFP package

2.0 Applications

- TFT LCD Monitor
- LCD TV
- LCD Projector with PC and/or TV Input
- Other Flat Panel Displays
- TV to PC Monitor Scan Converter
- Progressive Scan TV



AL300-01 Functional Block Diagram

3.0 General Description

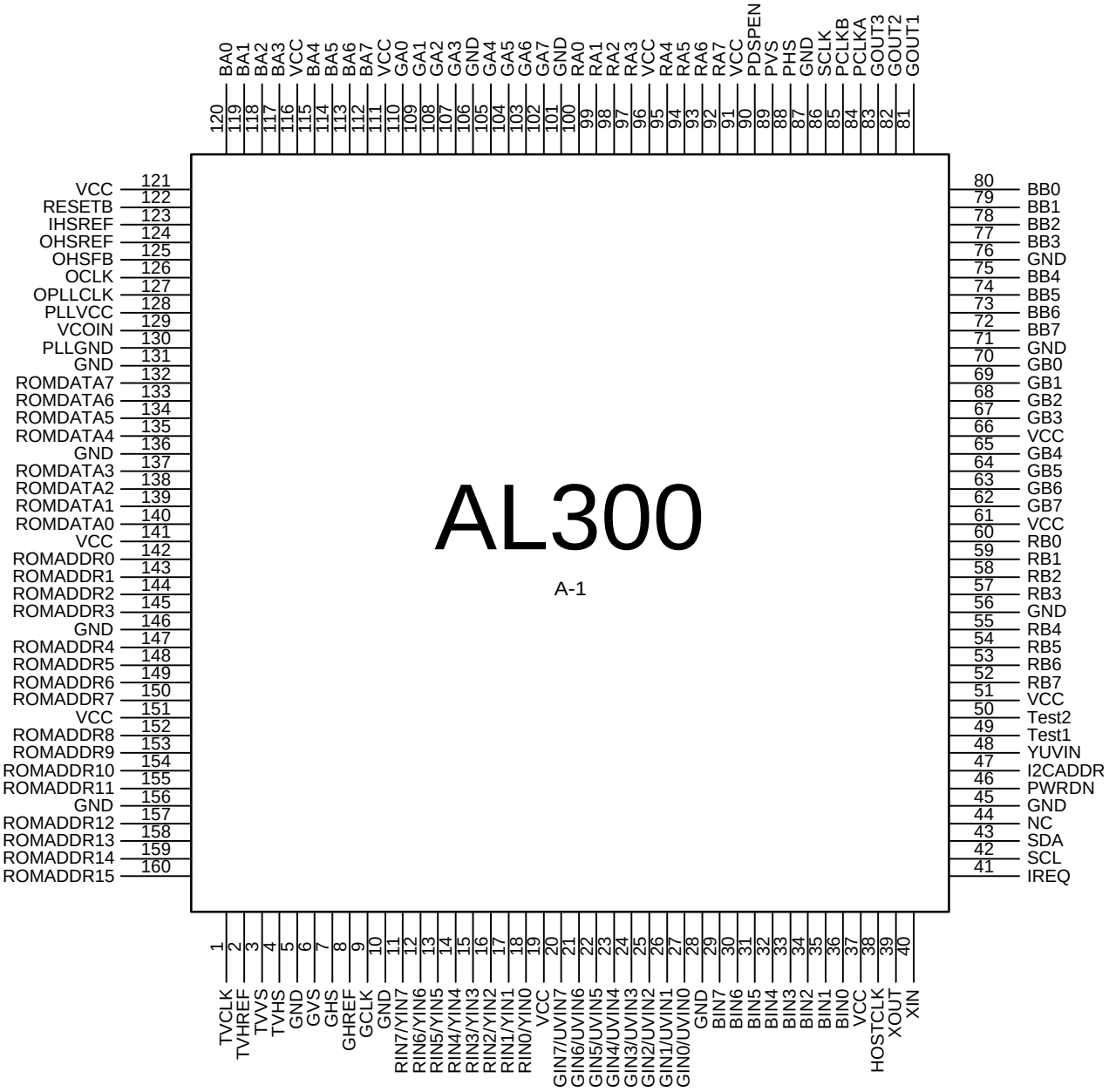
The AL300 is designed to enable simple connection from PC's or video devices to flat panel displays. It provides LCD/PDP monitor and projector manufacturers with a low-cost, easy solution to bring TV or PC video to LCD panels.

The AL300 is equipped with a high quality zoom engine that automatically maintains full screen output display, regardless of the resolution of the incoming signal. The input video can be linearly and independently zoomed in the x and y directions. The AL300 also provides de-interlacing, filtering, and scaling support for interlaced video to be displayed on a LCD panel.

Two integrated On Screen Display (OSD) windows provide overlay of a control menu, text, or caption on the output display. With the internal OSD RAM, OSD bitmaps of up to 8K pixels are supported. With optional external user-defined font table ROM, the AL300 OSD functionality is very flexible with font size and display location; virtually all languages and fonts are supported. Special OSD effects such as translucency and blinking offer the manufacturer a unique and vivid way of presenting monitor status, control menu, or other display information.

Used with an AL875 (high speed 3-channel ADC with PLL, 100-pin QFP), the AL300 (in 160-pin QFP) offers the best cost-performance and total solution for LCD monitors or projectors, or other flat panel devices.

4.0 Pinout Diagrams



AL300 pinout diagram

5.0 Pin Definition and Description

Following is the pin definition of the AL300:

Pin Name	Type	Pin #	Note
Video Interface			
TVCLK	IN (CMOSd)	1	Video Clock from Video Source
TVHREF	IN (CMOSd)	2	Video Horizontal Active Data Reference This signal is used to indicate valid data of the YUV input.
TVVS	IN (CMOSs)	3	Video Vertical Sync Signal
TVHS	IN (CMOSs)	4	Video Horizontal Sync Signal
Graphic Interface			
GVS	IN (CMOSs)	6	Graphic Vertical Sync Signal
GHS	IN (CMOSs)	7	Graphic Horizontal Sync Signal
GHREF	IN (CMOS)	8	Graphic Horizontal Active Data Reference
GCLK	IN (CMOSd)	9	Graphic Input Clock
R/YIN<7:0>	IN (CMOS)	11-18	Red Input When in RGB Mode Y Input When in CCIR601 422 and 444 Modes
G/UVIN<7:0>	IN (CMOS)	20-27	Green Input When in RGB Mode CbCr Input When in CCIR601 422 Mode Cb Input When in CCIR601 444 Mode Refer to register #1Bh for details.
BIN<7:0>	IN (CMOS)	29-36	Blue Input When in RGB Mode Cr Input When in CCIR601 444 Mode Refer to register #1Bh for details.
Host Interface			
HOSTCLK	OUT (CMOS)	38	Buffered Output of the Clock Input for Host Interface such as a Micro-controller
XOUT	OUT (CMOS)	39	Crystal Output
XIN	IN (CMOS)	40	Crystal Input; the frequency provided is for I^2C sampling and for output reference timing when input sync signals are missing or undetectable. Usually in the range of 10~50MHz. Refer to section 6.7 Internal Timing Generator for details.

IREQ	OUT (CMOS)	41	Interrupt Request, active high
SCL	IN (CMOSs)	42	I2C Serial Clock Input
SDA	INOUT (COMSsu)	43	I2C Serial Data Input/Output
GOUT1	OUT (CMOS)	81	General Purpose Output Connected to Register 0x1B bit 2
GOUT2	OUT (CMOS)	82	General Purpose Output Connected to Register 0x1B bit 3
GOUT3	OUT (CMOS)	83	General Purpose Output Connected to Register 0x1B bit 1
Configuration			
PWRDN	IN (CMOSd)	46	Power Down 0, Normal Operation 1, Power Down
I2CADDR	IN (CMOSd)	47	I2C Bus Slave Address Select 0, write address = 70, read address = 71 1, write address = 72, read address = 73
YUVIN	IN (CMOSd)	48	YUV Input 0, RGB Format Video Input 1, CCIR YUV Format Video Input Refer to RIN, GIN, BIN pins
Test1	IN (CMOSd)	49	Test Pin
Test2	IN (CMOSd)	50	Test Pin
Panel/Display Interface			
RB<7:0>	OUT (CMOS)	52-55, 57-60	Right Pixel of Interleaved Red Output in Dual Pixel Mode Valid when Register 0x43 bit4 = '1'. Data are output with PCLKB. For AL300 ver. A, the B data lag A data by 90° (half SCLK). For AL300 ver. B, A and B data are aligned.
GB<7:0>	OUT (CMOS)	62-65, 67-70	Right Pixel of Interleaved Green Output in Dual Pixel Mode Valid when Register 0x43 bit4 = '1'. Data are output with PCLKB.

			For AL300 ver. A, the B data lag A data by 90° (half SCLK). For AL300 ver. B, A and B data are aligned.
BB<7:0>	OUT (CMOS)	72-75, 77-80	Right Pixel of Interleaved Blue Output in Dual Pixel Mode Valid when Register 0x43 bit4 = '1'. Data are output with PCLKB. For AL300 ver. A, the B data lag A data by 90° (half SCLK). For AL300 ver. B, A and B data are aligned.
PCLKA	OUT (CMOS)	84	Leading Pixel Clock of Interleaved Video Output for Right data in Dual Pixel Mode. Polarity is programmable
PCLKB	OUT (CMOS)	85	Lagging Pixel Clock of Interleaved Video Output for Right data in Dual Pixel Mode. Polarity is programmable. Default PCLKB lags PCLKA by 180° (one SCLK).
SCLK	OUT (CMOS)	86	Display Pixel Clock (for single pixel per clock mode)
PHS	OUT (CMOS)	88	Panel/Display Hsync Can be programmed to either polarity.
PVS	OUT (CMOS)	89	Panel/Display Vsync Can be programmed to either polarity.
PDSPEN	OUT (CMOS)	90	Panel/Display Display Enable; used to indicate active output pixels (HDE). Can be programmed to either polarity.
RA<7:0>	OUT (CMOS)	92-95, 97-100	Red Data Output When Register 0x43 bit4 = '0', data are output every SCLK. When Register 0x43 bit4 = '1', the left pixel of interleaved red data are output with PCLKA.
GA<7:0>	OUT (CMOS)	102-105, 107-110	Green Data Output When Register 0x43 bit4 = '0', data are output every SCLK. When Register 0x43 bit4 = '1', the left pixel of interleaved red data are output with PCLKA.

BA<7:0>	OUT (CMOS)	112-115, 117-120	Blue Data Output When Register 0x43 bit4 = '0', data are output every SCLK. When Register 0x43 bit4 = '1', the left pixel of interleaved red data are output with PCLKA.
PLL (Phase Lock Loop) Interface			
IHSREF	OUT (CMOS)	123	Input Hsync Reference, buffered and polarity adjusted, usually for input PLL to regenerate input pixel clock. Always positive polarity. When no input HSYNC is present, virtual IHSREF can be generated by programming registers 41h & 42h
OHSREF	OUT (CMOS)	124	Output Hsync Reference, for output PLL to generate output pixel clock. Always positive polarity. OHSREF is either equivalent to IHSREF or the equally divided IHSREF. Refer to registers 03h, 10h~13h.
OHSFB	OUT (CMOS)	125	Output PLL Feedback; works with OHSREF to generate output pixel clock
OCLK	IN (CMOSd)	126	Output Clock, connected to OPLLCLK when internal PLL is used; connected to external PLL clock output when external PLL is used
OPLLCLK	OUT (CMOS)	127	Recovered Output Clock generated by the internal PLL
VCOIN	IN	129	PLL External VCO Filter Circuit Input
OSD ROM Interface			
ROMDATA <7:0>	IN (CMOSd)	132-135, 137-140	OSD ROM Data
ROMADDR <15:0>	OUT (CMOS)	160-157, 155-152, 150-147, 145-142	OSD ROM Address
Power, Ground, Reset			
RESETB	IN (CMOS)	122	Reset, active low
PLLVCC	POWER	128	VCC of Internal PLL, 3.3V
PLLGND	GROUND	130	GND of Internal PLL
VCC	POWER	19, 37, 51, 61,	Digital VCC, 3.3V

		66, 91, 96, 111, 116, 121, 141, 151	
GND	GROUND	5, 10, 28, 45, 56, 71, 76, 87, 101, 106, 131, 136, 146, 156	Digital Ground
NC	-	44	No connection

Remarks:
CMOSd: CMOS with internal pull-down
CMOSs: CMOS with Schmitt trigger
CMOSsu: CMOS with Schmitt trigger and internal pull-up

6.0 Functional Description

The AL300 provides a simple way of connecting video or graphic sources to a flat panel display such as a LCD. A LCD panel normally supports only one fixed resolution of digital non-interlaced data format; therefore, it can not be connected directly to a video source due to the difference in format. Additionally, VGA or TV video usually are in analog format and can be of many different resolutions and refresh rates. The AL300 is designed to change the various video/graphics formats into a fixed data format to be accepted by a specific LCD panel.

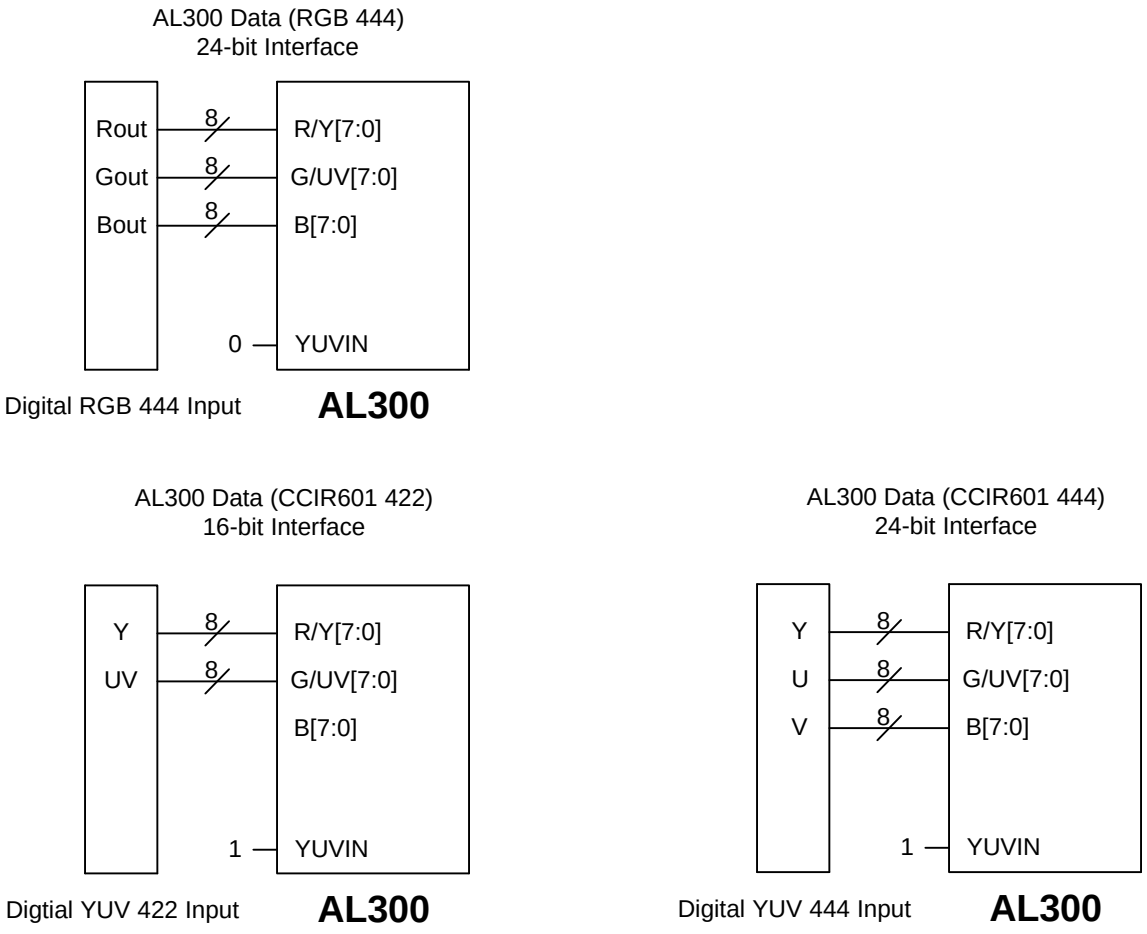
The input video format can be 24-bit digital RGB or 16-bit digital YUV. Different video sources can be connected to an AL300 via certain digitization devices. For example, analog NTSC/PAL/SECAM can be connected to the AL300 via a digital video decoder. Analog VGA RGB signal can be connected to the AL300 after a high-speed three-channel analog-to-digital converter (ADC), such as the AL875. Alternatively, digital PC graphics can also be applied to the AL300 through a Panel Link or LVDS (low voltage differential signal) receiver.

The scan rate and resolution of the incoming video are recognized by the scan mode detection circuits. The detected mode information stored in the AL300 registers is read through the I^2C bus by the firmware. The firmware then will decide how to program the chip such that the input video can be scaled up to fit into the full screen size of the display. To achieve this, the linear zoom engine needs to be programmed to properly scale the input video to the desired output resolution. Also, the frequency of the display clock generated from the internal PLL needs to be programmed to the correct level.

After the digital data is scaled, optional dithering is performed to retain color resolution for LCD panels that support only 12-bit or 18-bit color resolutions. The processed data is overlaid with internal on-screen-display (OSD) for display status or control menu.

6.1 Input Data Format

The AL300 accepts three input data formats: 24-bit RGB, 16-bit CCIR601 422 and 24-bit CCIR601 444 data. The clock and sync signal pins are separate for RGB or YUV while the YUV data share the same pins as the RGB data. The input video format is defined by the YUVIN pin. Diagram AL300-06 Input Data Format shows how to connect the AL300 for each format.



AL300-06 Input Data Format

YUV input data is selected when the YUVIN pin is high. This hardware pin can be disabled by "SoftConfig" (in register #03h, GENERAL). When the bit is set to one, the input video format is controlled by "VideoIn" (in register #02h, BOARDCONFIG). RGB input data is selected when the YUVIN pin is low.

6.2 Scan Mode Detection

The resolution of the analog input is determined by two hardware registers, LINERATE (Reg. #61h and #62h) and INVTOTAL (Reg. #63h and #64h), which store the detected input line rate and the

detected total number of vertical lines, respectively. The line rate and frame rate values can be obtained by counting the clock provided from XIN pin and/or calculations and some calculations.

When the line rate of the input video changes, a programmable interrupt signal is sent out to the micro-controller for mode setting or other controls. Details about this can be found in the Interrupt section.

6.3 Automatic Positioning Control

The active video (horizontal and vertical) starting and ending positions are detected to ensure that the whole picture fits into the displayable region of the screen. The threshold of the minimum active video value is user-definable. Position detection can be performed for any row or column across the whole screen and it is programmable. Related registers are:

HNUMBER (horizontal line number, Reg. #70h),
DATA_TH (data threshold, Reg. #71h),
HDE_ST (horizontal active start high and low, Reg. #72h and #73h),
HDE_END (horizontal active end high and low, Reg. #74h and #75h),
VCOLUMN (selected vertical column number, Reg. #79h),
VDE_ST (vertical active start high and low, Reg. #7Ah and #7Bh),
and VDE_END (vertical active end high and low, Reg. #7Ch and #7Dh).

Details about these registers can be found in the Register Definition section.

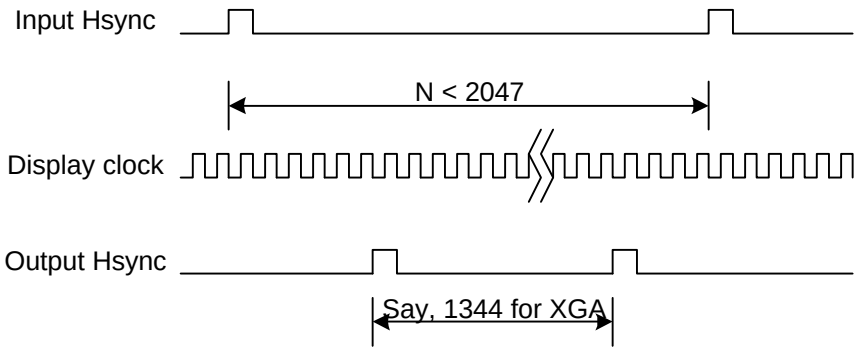
6.4 Output Clock PLL

The internal PLL can regenerate a display clock from the incoming horizontal sync signal. The maximum multiplication factor (N) is 2047 defined by register PLLDIV (Reg. #10h and #11h). The N value is derived from the following formula:

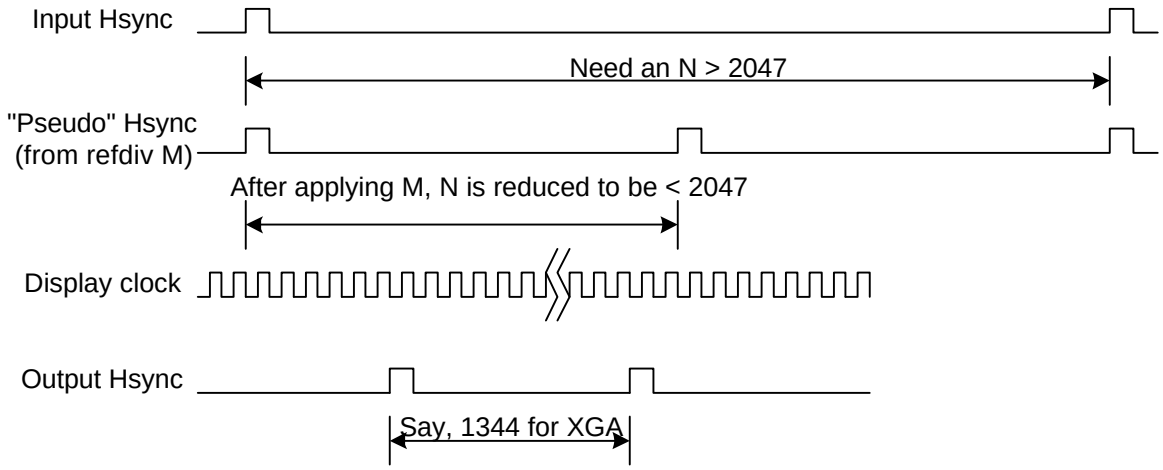
$$PLLDIV(N) = \frac{Vo}{Vi} \bullet Ho$$

Where, Ho is the total number of pixels per output line
Vo is the output active lines per frame
Vi is the input active lines per frame

The following drawing shows how the input Hsync is divided by the PLL to generate the display clock, which in turn generates the output Hsync.

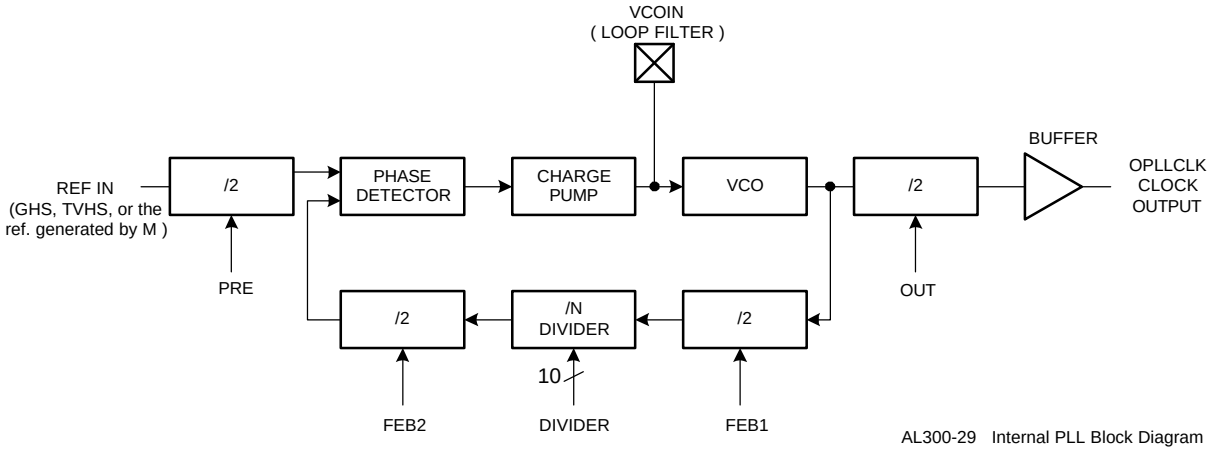


If the scan rate of input video HSYNC is much slower, as in the case of interlaced video input, PLLREFDIV can evenly divide the input HSYNC into small segments of PLL reference input to achieve a higher multiplication rate. This "pseudo" Hsync is generated by equally dividing the input Hsync into 2 or more segments. The length of the pseudo Hsync is defined by the scaling divider (M) in register PLLREFDIV (Reg. #12h and #13h). The following diagram shows the relationship among these signals:



AL300-30 HSYNC and reference clock

Following is the AL300 internal PLL block diagram:



AL300-29 Internal PLL Block Diagram

The AL300's internal PLL has better performance at high frequency than low frequency when the output rate is within 110MHz. To take advantage of this property, there are four "divide-by-2" dividers for optimal operation. The FEB1 and FEB2 work as frequency multipliers; the PRE and OUT work as frequency dividers. To turn on or off these dividers, control register #11h. The recommended values for the register setting at different input modes are as follows:

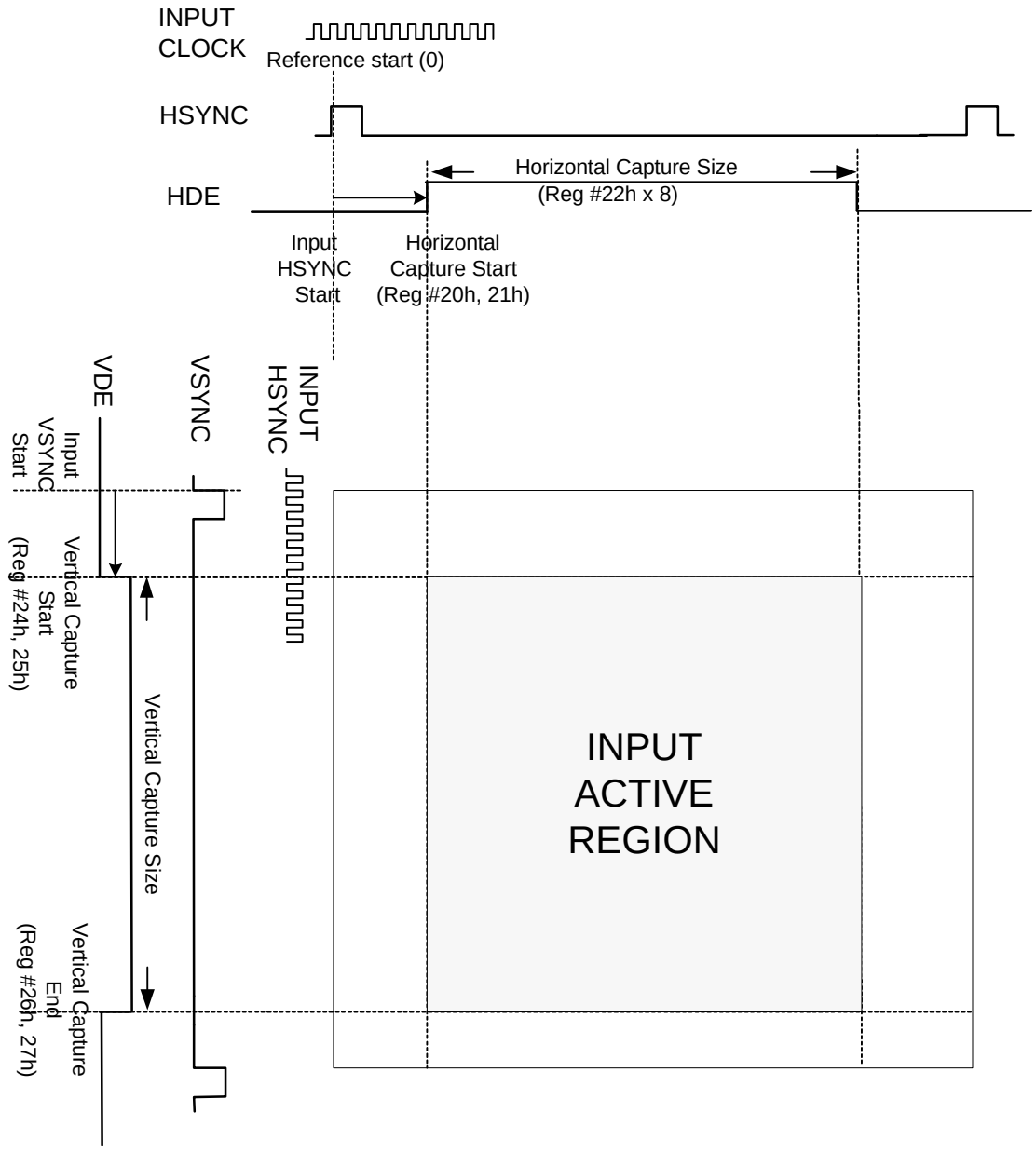
Ref freq. (KHz)	Fout (MHz)	Divider value	FEB2	FEB1	OUT	PRE	DIVIDER	Reg. 11h (Hex)	Reg. 10h (Hex)
31.5	25.2	800	1	0	1	0	011,00011110	A3	1E
37.5	31.5	840	1	0	1	0	011,01000110	A3	46
37.88	40	1056	1	1	1	0	010,00001110	E2	0E
46.75	49.368	1056	1	1	1	0	010,00001110	E2	0E
48.0	64.8	1350	0	1	0	0	010,10100001	42	A1
56.5	74.92	1326	0	1	0	0	010,10010101	42	95
60.0	78.72	1312	0	1	0	0	010,10001110	42	8E
64	85.5	1336	0	1	0	0	010,10011010	42	9A
64	107.52	1680	0	1	0	0	011,01000110	43	46

The output pixel clock can be either from the internal PLL locked to the ref_in clock, or from the external clock pin, OCLK. The ref_in reference clock is the GHS, TVHS, or the pseudo-reference clock generated by the PLLREFDIV.

The frequency of the reference clock ref_in is between 10 KHz and 1 MHz. The output frequency is adjustable and the maximum frequency is up to 110 MHz. Details about the PLL related registers can be found in the Register Definition section.

6.5 Input/Capture Timing

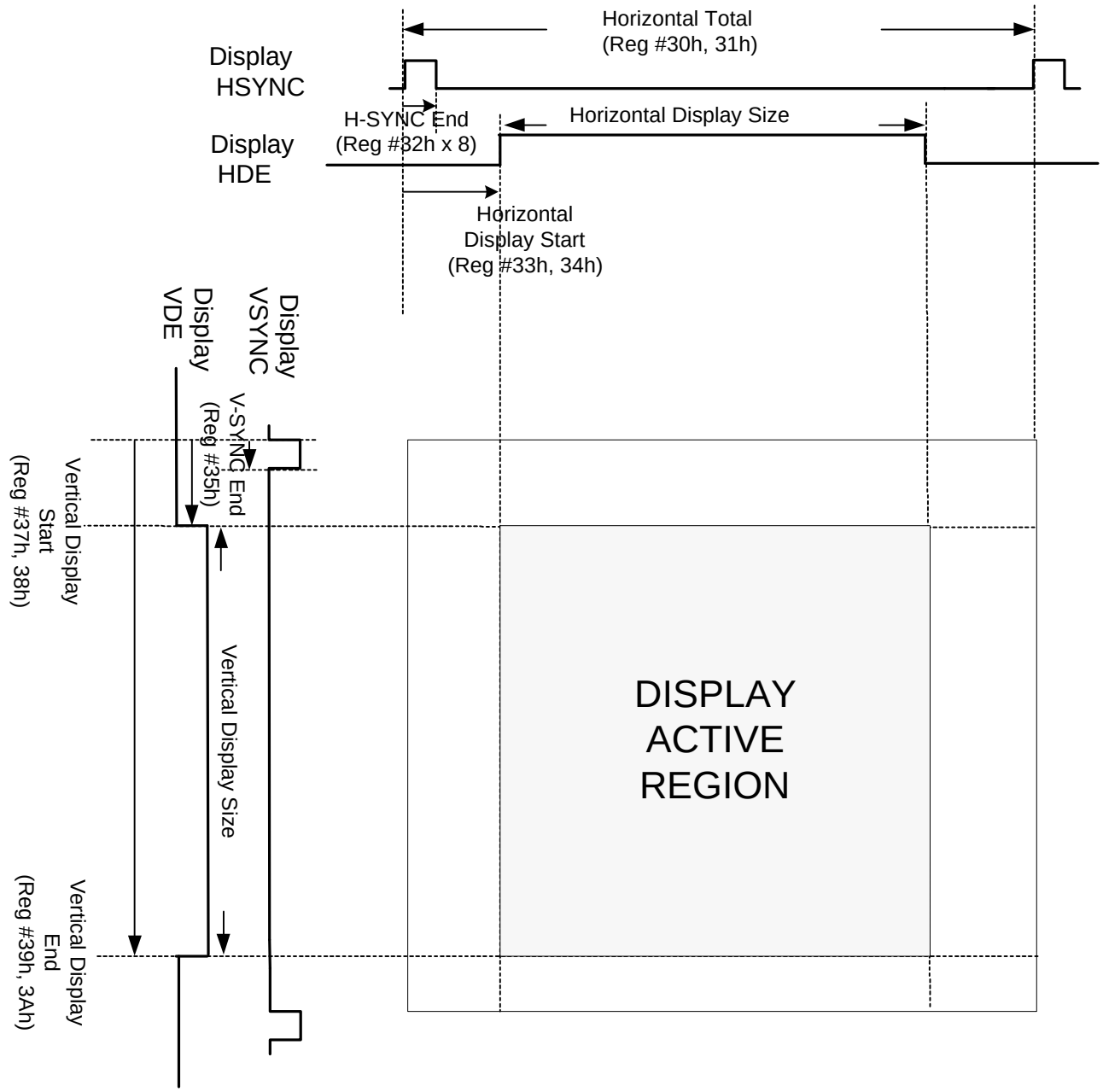
The input timing defines the active region of the input video/graphics to be processed and eventually displayed on the screen. The "captured" window for display is defined by registers CAPHSTART (Reg. #20h and #21h), CAPHSIZE (Reg. #22h), CAPVSTART (Reg. #24h and #25h), and CAPVEND (Reg. #26h and #27h). The output vertical sync position relative to the input vertical sync is defined by register FRAMEVSTART (Reg. #28h and #29h). All the parameters are relative to the leading edges of the input HSYNC and VSYNC. The following diagram shows the input active window timing and the related registers.



AL300-07 Input video timing

6.6 Output/Display Timing

The output display resolution, sync signal duty cycles, and active display position are fully programmable. They are defined by registers DSPHTOTAL (Reg. #30h and #31h), DSPHSEND (Reg. #32h), DSPHDESTART (Reg. #33h), DSPHDEEND (Reg. #34h), DSPVDESTART (Reg. #37h and #38h), and DSPVDEEND (Reg. #39h and #3Ah). The following diagram shows the output active window timing and the related registers.



AL300-08 Output video timing

The output HSYNC delay relative to the input HSYNC is defined by FRAMEDELAY (Reg. #3Bh), and FRAMEDELAYODD (Reg. #3Ch). For interlaced video input, FRAMEDELAYODD is half a line more than FRAMEDELAY. For non-interlaced video input, FRAMEDELAYODD has the same value as FRAMEDELAY.

6.7 Internal Timing Generator

The AL300 normally synchronizes to the external video source timing provided by a graphic chip (or recovered by an external PLL) or a video decoder. It can also generate internal reference timing from the clock provided by the XIN pin. When no external video source is connected or the input timing is not supported by the firmware, the internal HSYNC and VSYNC can be used to generate timing for the OSD display to show monitor status, warning signals and/or other messages. The desired internal HSYNC and VSYNC frequency can be generated by programming the REFHTOTAL (Reg. #41h) and REFVTOTAL (Reg. #42h) registers. If the XIN frequency is 14.31818 MHz and the expected vertical frame rate is 60 Hz, the following table shows the register values for some typical resolutions.

	REFHTOTAL (Reg. #41h)	REFVTOTAL (Reg. #42h)	Remarks
640x480	450	520	14.3MHz =: 450 x 520 x 60
800x600	376	632	14.3MHz =: 376 x 632 x 60
1024x768	296	800	14.3MHz =: 296 x 800 x 60
1280x1024	216	1072	14.3MHz =: 216 x 1072 x 60

To detect if there is input signal, turn on VSYNCIrq by writing “10” to register #40h<1:0>. “1” status of register #65h<1> indicates that input VSYNC is detected. Reset register #40h as “00h” before next input VSYNC detection is needed. When input VSYNC is missing, turn on the internal timing as mentioned above.

If input VSYNC is detected but the mode is not supported by the firmware (such as out-of-spec resolution), also turn on the internal timing to avoid sync conflict. In addition, set register #43h<6> as 1and write 00h to “color 3” registers (#B9h~BBh) to blank the input graphics/video. Turn on the OSD to show warning signals (such as “out of sync”).

To detect if “out of sync” status is changed, turn on LineRateIrq by writing “01” to register #40h<1:0>. “1” status of register #65h<0> indicates that input mode is changed. The firmware should then perform mode detection. If a valid input mode is detected, turn off the internal timing.

Other control algorithm is also available.

6.8 Zoom Engine

The AL300 has two independent horizontal and vertical scale engines performing proprietary scaling operations. The range of linear scaling can be from 1 up to 8 times.

The horizontal interpolation coefficient is generated by the 11-bit HRATIO (Reg. #14h and #15h), which defines the scaling ratio with the 3 MSB as the integer part and the 8 LSB as the mantissa part. The HRATIO is derived from the size of the displayed input video and the actual scaled display size. The related registers include CAPHSTART (Reg. #20h and #21h), CAPHEND (Reg. #22h), DSPHDESTART (Reg. #33h), and DSPHDEEND (Reg. #34h). The initial value for the interpolation coefficient generator is defined by HINITPHASE (Reg. #18h).

The vertical interpolation coefficient generator, the 15-bit VRATIO (Reg. #16h and #17h) defines the scaling ratio with the 3 MSB being the integer part and the 12 LSB being the mantissa part. The VRATIO is derived from the size of the displayed input video and the actual scaled display size. The registers involved include CAPVSTART (Reg. #24h and #25h), CAPVEND (Reg. #26h and #27h), DSPVDESTART (Reg. #37h and #38h), and DSPVDEEND (Reg. #39h and #3Ah). The initial value for the interpolation coefficient generator is defined by VINITPHASE (Reg. #1Ah). All of these registers can be used to achieve optimum sharpness and smoothness results for different scaling ratios.

6.9 Operating Modes

The AL300 operates in three different modes: non-interlaced zoom mode, de-interlaced zoom mode, and bypass mode

6.9.1 Non-interlaced Zoom Mode

Non-interlaced zoom mode is for synchronizing the output display with the non-interlaced input video with a zooming ratio of larger than or equal to one. The output video can have the same or higher resolution than the input video. First of all, the output timing parameters are decided depending on the size of the display device. Secondly, the firmware determines what area of picture needs to be displayed with the help of internal scan rate and position detection circuits. The input timing parameters are set accordingly. The next step is to set the parameters of the scaling engine by the ratio between the output video and input video. Details about programming the related registers can be found in the Register Definition section.

6.9.2 De-interlaced Zoom Mode

De-interlaced zoom mode is different from Non-interlaced zoom mode in that the input video is interlaced and needs to be de-interlaced. Some parameters need to be treated differently for different operations required in odd and even fields in interlaced modes compared to non-interlaced modes. Details about programming the related registers can be found in the Register Definition section.

6.9.3 Bypass Mode

This mode is only good for the application in which the input and output resolution are the same. Basically, the input clock and H-sync timing are used for the output timing and therefore the picture is not zoomed. However, the data can be processed by all of the internal data processing hardware and OSD circuit, except for the scaling function. Only some registers that are used for the previous two modes are affected. Details about programming the related registers can be found in the Register Definition section.

6.10 Interrupt

The interrupt signal can be triggered by one or a combination of any of the three following events.

1. When the line rate of the input video changes.
2. When input VSYNC arrives.
3. When the odd field arrives.

Register IREQSOURCE (Reg. #40h) decides which interrupt event(s) are used to trigger the interrupt. The first interrupt source is helpful in automatically detecting the mode changes. The second interrupt source can be used to do a periodic routine during the vertical blanking region. The last interrupt source can be used to do a periodic routine in the odd field of interlaced input video.

Upon receiving the interrupt request from the AL300, the micro-controller needs to first check the interrupt event by reading the register IREQSTATUS (Reg. #65h) to decide what operations to perform. After the interrupt routine is complete, the interrupt status register is reset by writing the register IREQSOURCE (Reg. #40h) and then it becomes ready for the next interrupt request.

6.11 On Screen Display (OSD)

Two independent On-Screen-Display (OSD) windows provide overlay for a control menu, text, or caption on the output display. The AL300's OSD is very flexible in the way that the font, size, and display location are all programmable. The internal 1K byte SRAM provides storage for the OSD information. The OSD can be operated with only this internal SRAM or with an external ROM to store font tables or even larger bitmaps.

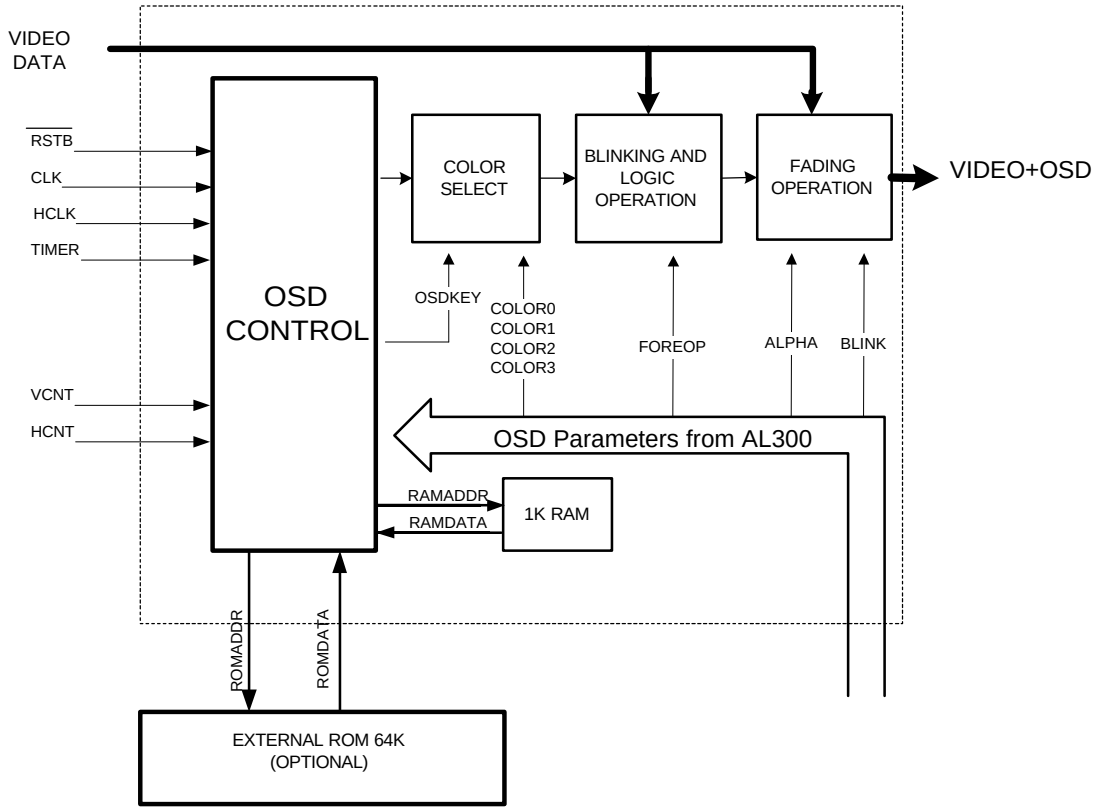
The OSD key from the OSD controller selects the colors to be overlaid on the screen based on the data stored in RAM or ROM as well as the control registers that decide when and where these colors are displayed on the screen. The OSD key is used to select four out of 16 million OSD colors.

The selected OSD color data is then passed through the internal blink control circuits. Blinking is supported in 2-byte font addressing ROM mode only and will be further described later. The blinking frequency is controlled by the value defined by BLINKCTRL (Reg. #36h) with VYSNC as the reference counting clock. The blinking information is stored as data embedded in character codes stored in RAM.

After the processed blinking data is passed to the logic operation section, the logic operation section performs special OSD effects such as translucent, opaque, negative, and posterization. The logic operation feature, which provides a vivid and unique way of implementing display menu and status, is controlled by the FOREOP (Reg. #81h) register.

The fading operation block creates the effect of smoothly fading in and fading out the OSD titles or menu. Fading is controlled by register FADEALPHA (Reg. #82h).

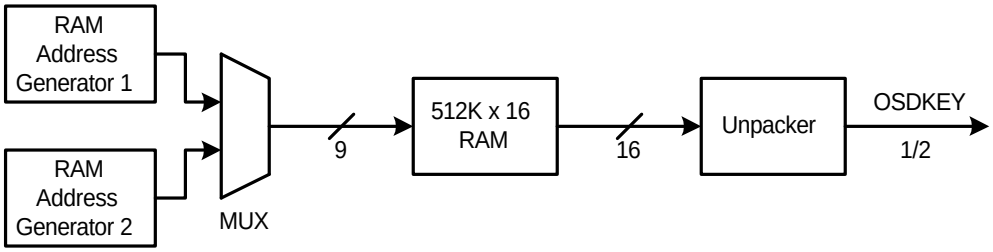
Following is the block diagram of the AL300 OSD.



AL300-09 OSD Block Diagram

6.11.1 RAM mode

If external ROM is not used, OSD data can be stored in the internal 1K Byte of RAM. In this mode, the OSD data is stored in bitmap form, which is written by the host through the I^2C bus. Two independent OSD RAM address generators generate addresses for indexing the OSD data stored in RAM. The data written from the RAM are unpacked to either 1-bit or 2-bit data depending if 2-color mode or 4-color mode is selected. MSB bits are read out as OSD data before the LSB bits. The following diagram shows how the OSD key generator works in RAM mode.

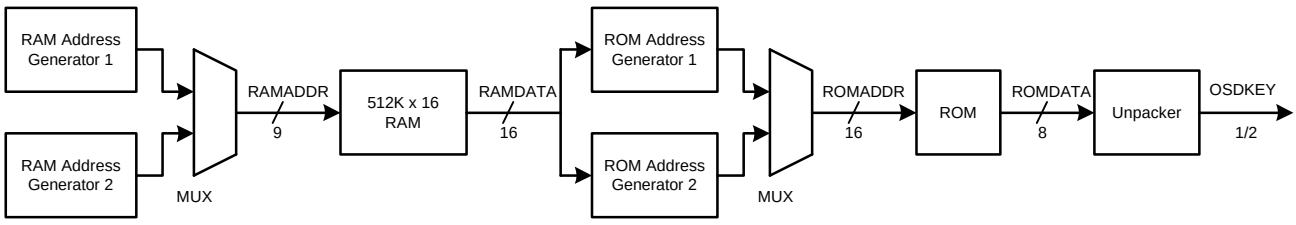


AL300-10 OSD Key Generator in RAM Mode

The RAMADDRST1 (Reg. #92h) and RAMADDRST2 (Reg. #A2h) are the starting address pointers indicating the starting addresses of the bitmaps of OSD1 and OSD2. RAMSTRIDE1 (Reg. #93h and #8Bh) and RAMSTRIDE2 (Reg. #A3h and #8Ch) are the line offsets in the RAM for OSD1 bitmap and OSD2 bitmap, respectively.

6.11.2 ROM mode

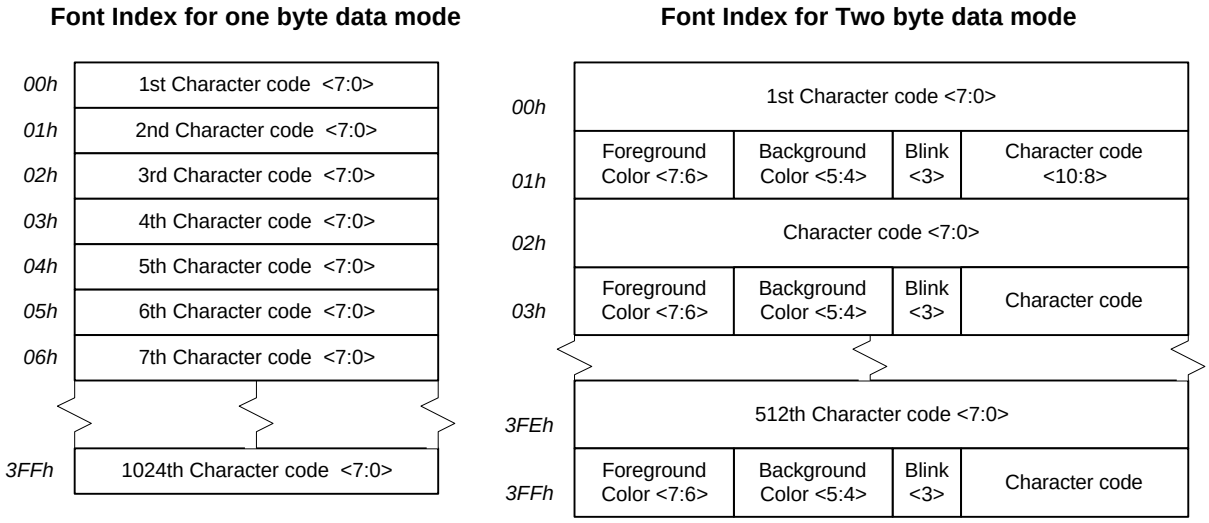
If external ROM is used to store font tables or bitmaps, the internal RAM is used to store the font code or index for assessing a specific font (a font can be viewed as a pre-defined bitmap). The addresses of the font indexes are generated first to retrieve the index address to a specific font or bitmap. The following ROM address generators then calculate the address of the font stored in ROM based on the font index read from RAM. The 8-bit data read from ROM is then unpacked into 1-bit or 2-bit data depending if 2-color mode or 4-color mode is selected. MSB bits are read out as OSD data before the LSB bits. This is illustrated as follows:



The font addressing code can be in 1-byte or 2-byte formats. Blinking function is only supported in 2-byte font index format.
RAMADDRST1 (Reg. #92h) and RAMADDRST2 (Reg. #A2h) are the starting addresses of the font index codes of OSD1 and OSD2 stored in RAM. The font index is used to address the font data store in ROM. The address of a data of a specific font is calculated as:

Address = ROMADDRST + (OSD font index * FONTADDRESSUNIT) + (OSD line count * FONTLINESIZE)

The data format stored in RAM for addressing of OSD bitmap data in ROM mode is illustrated in the following diagram.

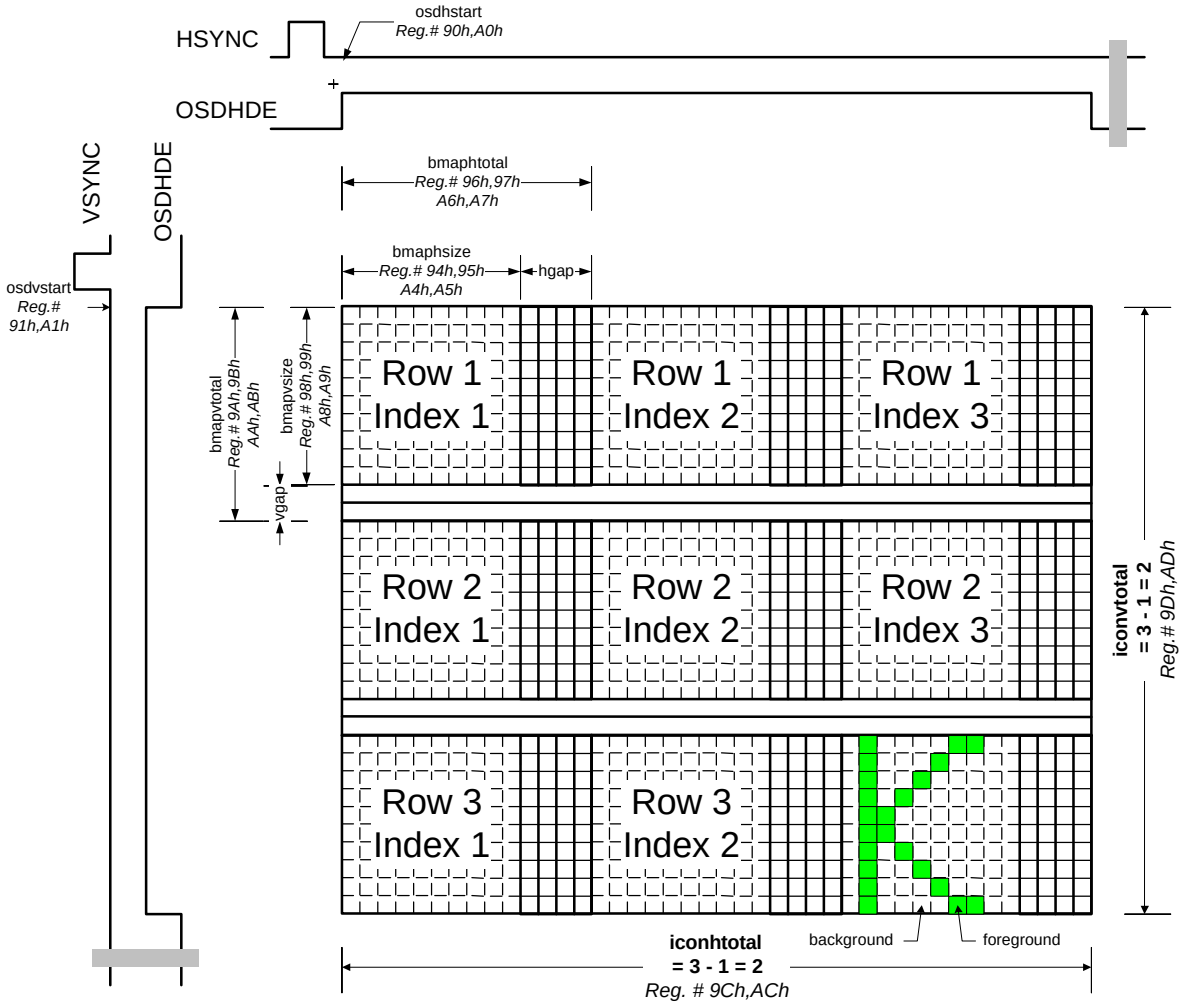


AL300-11 OSD Data Addressing in ROM Mode

6.11.3 OSD Timing

The register OSDHSTART (Reg. #90h and #A0h) refers to the horizontal starting position of the OSD window relative to the leading edge of HSYNC. OSDVSTART (Reg. #91h and #A1h) refers to the vertical starting position of the OSD window relative to the leading edge of VSYNC. The BMAPHSIZE (Reg. #94h, #95h, #A4h, and #A5h) defines the horizontal visible portion of the font (it can be smaller or equal to the actual horizontal font size). BMAPHTOTAL (Reg. #96h, #97h, #A6h and #A7h) is BMAPHSIZE plus the horizontal size of the border or gap in between fonts. The BMAPVSIZE (Reg. #98h, #99h, #A8h, and #A9h) defines the vertical visible portion of the font (it can be smaller or equal to the actual vertical font size). BMAPVTOTAL (Reg. #9Ah, #9Bh, #AAh,

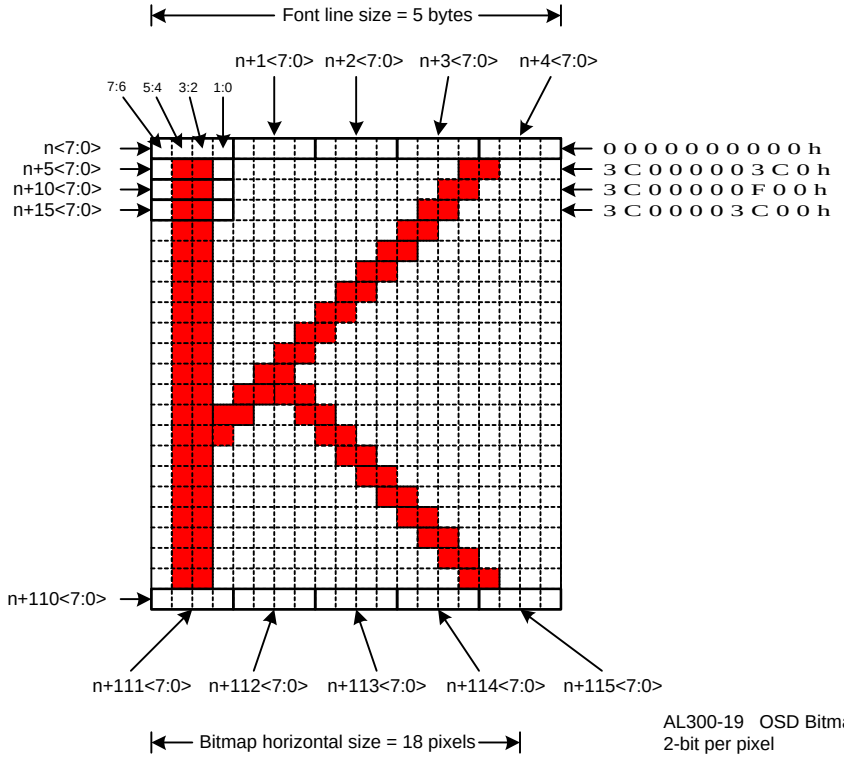
and #ABh) is BMAPVSIZE plus the vertical size of border or gap in between fonts. The total numbers of fonts in the horizontal and vertical directions are defined by ICONHTOTAL (Reg. #9Ch and #ACh) and ICONVTOTAL (Reg #9Dh and #ADh).
The screen timing diagram of the OSD window is as follows:



AL300-15 OSD Scren Timing Diagram

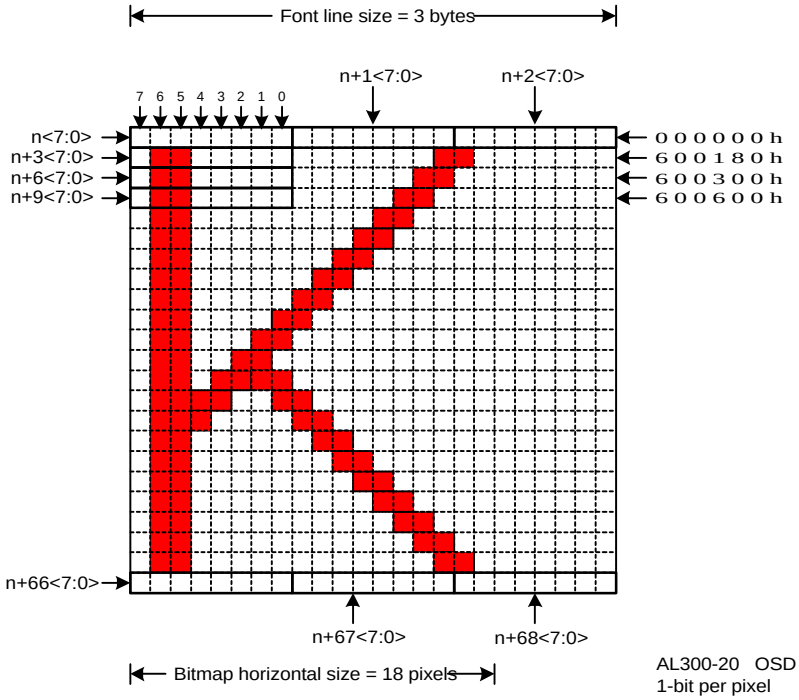
6.11.4 OSD Bitmap/Font Formatting

The following two examples show how the bitmaps or fonts should be formatted for OSD RAM or ROM. The first example uses 2-bit per pixel formatting:



AL300-19 OSD Bitmap_Font Formatting
2-bit per pixel

The next example uses 1-bit per pixel formatting:



AL300-20 OSD Bitmap_Font Formatting
1-bit per pixel

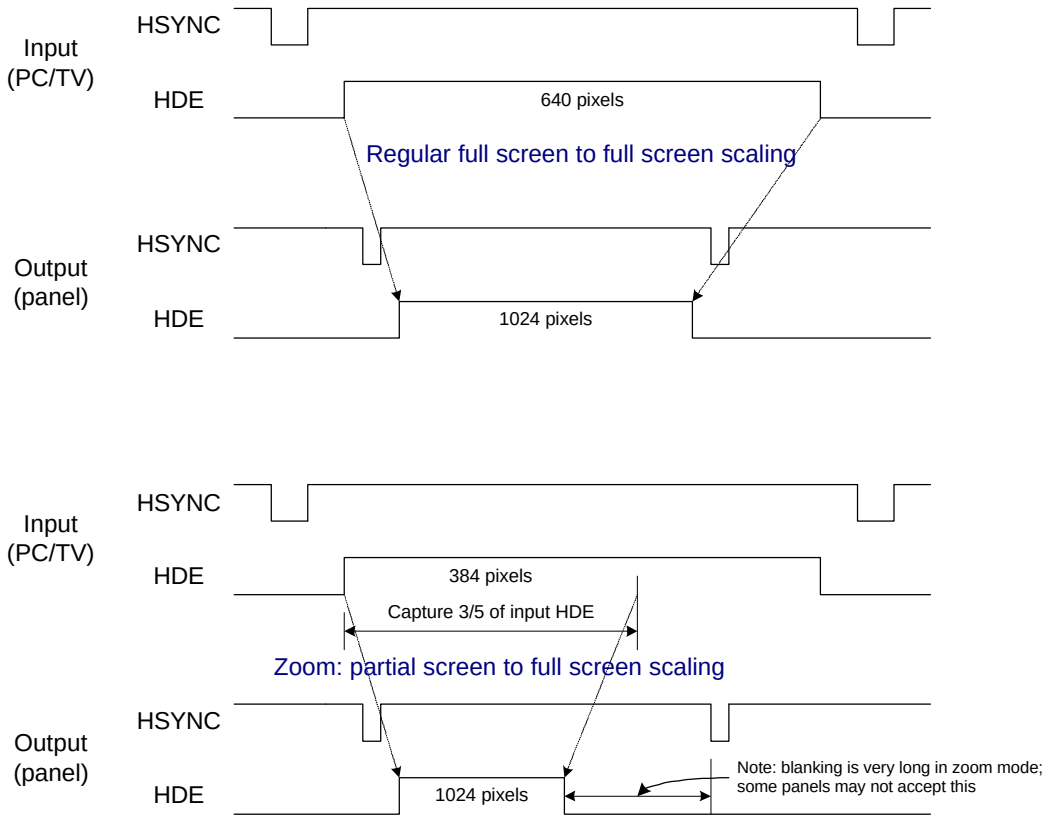
6.12 Output Data Format

Different output data formats for various types of panels are supported. Data can be programmed to be single pixel per clock mode or dual pixels per clock mode. This is controlled by "DualOut" (in register OUTPUTCONTROL, #43h). When DualOut = '0', output is single pixel per clock. When DualOut = '1', output is dual pixels per clock.

Dithering, controlled by "DithMode" (in register DITHER, Reg. #06h), is performed to retain color resolution for LCD panels that support only 12-bit or 18-bit color depths. Twenty-four-bit RGB data are output when DithMode = '00', 18-bit RGB data are output when DithMode = '01', and 12-bit RGB data are output when DithMode = '10'.

6.13 Zoom

The AL300 can scale the input resolution up to 8x8 times. However, since the AL300 does not perform frame rate conversion, nor does it store the whole input frame, full zoom functionality is not available. Nevertheless, the AL300 can zoom the scaled screen up to approximately1.8x1.8 times (depending on the panel timing tolerance) by capturing only part of the input pixels and using a higher scaling factor. The following drawing shows how this is performed:



AL300-22 Zoom mode (VGA scale up to XGA)

6.14 Initialization

The AL300 does not need initialization. However, for panels that need all data and control signals to be low during initialization, the AL300 register #43h<6> can be set to 1 to pull all output data pins low by writing 00h to "color 3" registers (#B9h~BBh). The AL300 PHS, PVS, PDSPEN, and clock outputs are low before programming.

6.15 I²C Programming

The AL300 I²C programming interface follows the Philips I²C standard. The I²C interface consists of the SCL (clock) and SDA (data) signals. Data can be written to or read from the AL300. For both read and write, each byte is transferred MSB first and LSB last, and the SDA data bit is valid when the SCL is pulled high.

The read/write command format is as follows:

Write: <S> <Write SA> <A> <Register Index> <A> <Data> <A> <P>

Read: <S> <Write SA> <A> <Register Index> <A> <S> <Read SA> <A> <Data> <NA> <P>

Following are the details:

<**S**>:
Start signal

SCL	SDA
High	High
High	Low

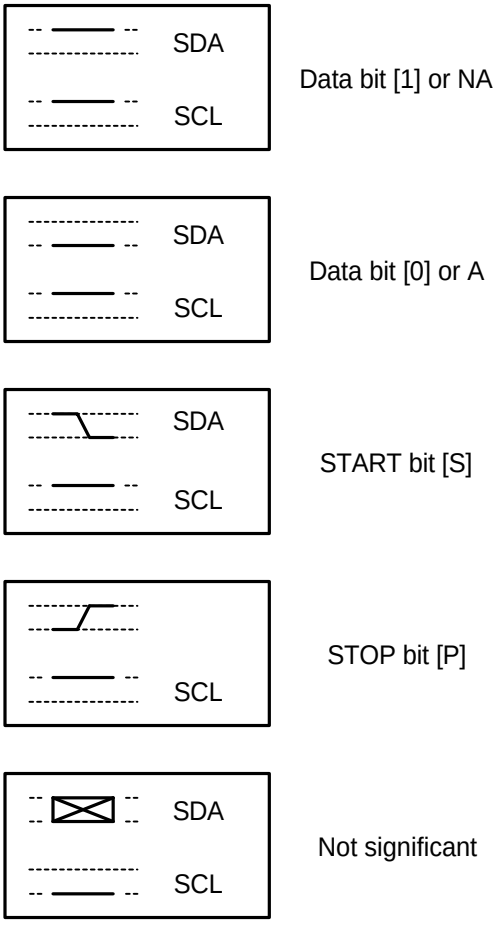
The Start signal is HIGH to LOW transition on the SDA line when SCL is HIGH.

<**WRITE SA**>:
Write Slave Address: 70h or 72h

<**READ SA**>:
Read Slave Address: 71h or 73h

<**REGISTER INDEX**>:
Value of the AL300 register index.

<**A**>:
Acknowledge stage
The acknowledge-related clock pulse is generated by the host (master). The host releases the SDA line (HIGH) for the AL300 (slave) to



AL250-15 I2C drawing

pull down the SDA line during the acknowledge clock pulse.

<**NA**>:
Not Acknowledged stage
The acknowledge-related clock pulse is generated by the host (master). The host releases the SDA line (HIGH) during the acknowledge clock pulse, but the AL300 does not pull it down during this stage.

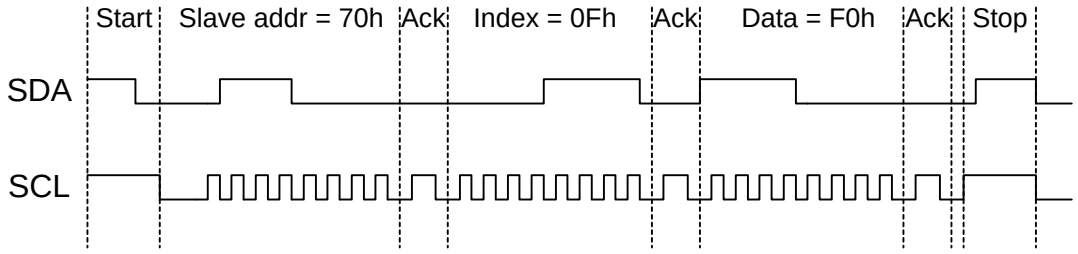
<**DATA**>:
Data byte write to or read from the register index.
In read operation, the host must release the SDA line (high) before the first clock pulse is transmitted to the AL300.

<**P**>:
Stop signal

SCL	SDA
High	Low
High	High

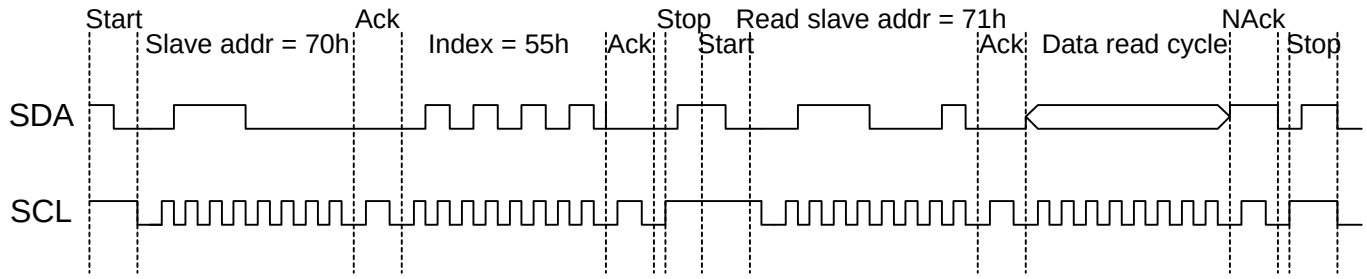
The Stop signal is LOW to HIGH transition on the SDA line when SCL is HIGH.

Suppose data F0h is to be written to register 0Fh using write slave address 70h, the timing is as follows:



AL300-13 I2C Write timing

Suppose data is to be read from register 55h using read slave address 71h, the timing is as follows:



AL300-14 I2C Read timing

More information on the AL300 functionality can be found in the Register Definition section.

7.0 Electrical Characteristics

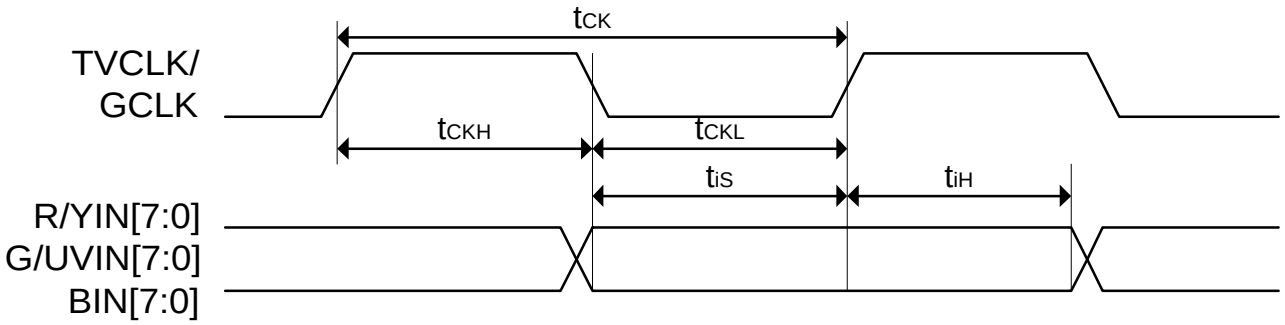
7.1 Recommended Operating Conditions

Parameter		Min	Max	Unit
VDD	Supply Voltage	+3.0	+4.0	V
TAMB	Ambient Operating Temperature	0	+70	°C

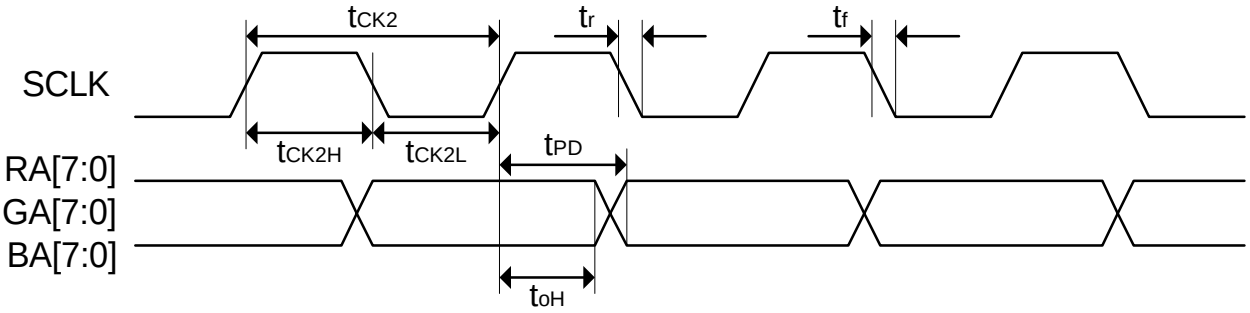
7.2 Characteristics

Parameter		Test Conditions	Min	Max	Unit
I_{DD}	Supply current		80	300	mA
P	Power consumption		240	1200	mW
V_{IH}	Hi-level input voltage		0.7VDD	5.5	V
V_{IL}	Lo-level input voltage		-0.5	+0.8	V
V_{OH}	Hi-level output voltage		2.4	VDD	V
V_{OL}	Lo-level output voltage		-	0.5	V
I_O	Output current, data	$-0.5V<V_O<V_{DD}+0.5$	-8	8	mA
	Output current, bus driver	$-0.5V<V_O<V_{DD}+0.5$	-12	12	mA
I_{LI}	Input leakage current		-	1	μA
C_i	Input pin capacitance		-	8	PF
δ_{CK2}	Duty factor (t_{CK2H}/t_{CK2})		40	60	%
t_{iS}	Input data set-up time		5	-	ns
t_{iH}	Input data hold time		3	-	ns
t_r	Input rise time	Vi = 0.6 to 2.6V	-	5	ns
t_f	Input fall time	Vi = 2.6 to 0.6V	-	5	ns
C_L	Digital output load cap.		15	50	PF
t_{oH}	Output hold time	C_L = 15pF	3	-	ns
t_{PD}	Propagation delay	C_L = 40pF	-	5	ns

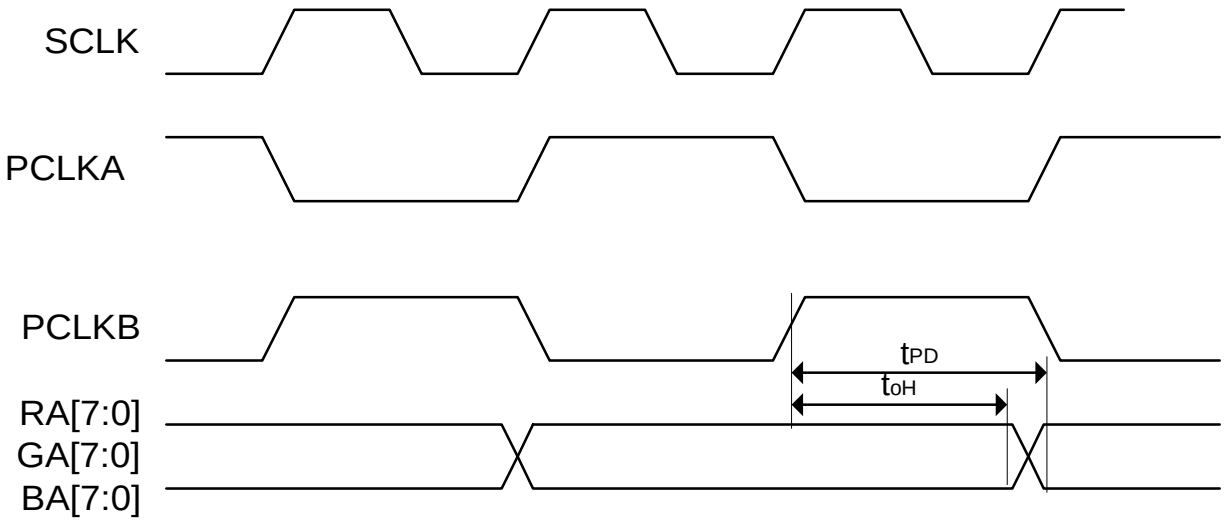
7.3 Timing Diagram



AL300 Input timing



AL300 Single pixel output timing



AL300 Dual pixel output timing

8.0 AL300 Register Definition

Following is the summary of the AL300 control registers:

Register	R/W	Address	Function
Configuration			
COMPANYID	R	00h	Company ID (46h)
REVISION	R	01h	Revision number
BOARDCONFIG	R/W	02h	Board configuration
GENERAL	R/W	03h	General control
FAMILY	R	04h	Chip family number
POLARITY	R/W	05h	Polarity control
DITHER	R/W	06h	Dither control
ADJUSTMENT	R/W	07h	Adjustment control
Output Clock PLL and Zoom Ratio			
PLLDIV	R/W	10h, 11h	PLL divider number
PLLREFDIV	R/W	12h, 13h	Display PLL reference input divider number
HRATIO	R/W	14h, 15h	Horizontal scale ratio
VRATIO	R/W	16h, 17h	Vertical scale ratio
HINITPHASE	R/W	18h	Horizontal zoom scaler initial phase
VINITPHASEODD	R/W	19h	Vertical zoom scalar initial phase in odd field of interlaced input
VINITPHASE	R/W	1Ah	Vertical zoom scalar initial phase
GOUT	R/W	1Bh	Panel Power Control
Input Timing			
CAPHSTART	R/W	20h, 21h	Horizontal capture start position
CAPHEND	R/W	22h	Horizontal capture size
RESETTEST	R/W	23h	Testing
CAPVSTART	R/W	24h, 25h	Vertical capture start position
CAPVEND	R/W	26h, 27h	Vertical capture end position
FRAMEVSTART	R/W	28h, 29h	Display vertical counter start position relative to input vertical counter

Output Timing			
DSPHTOTAL	R/W	30h, 31h	Display horizontal total
DSPHSEND	R/W	32h	Output display horizontal sync end position
DSPHDESTART	R/W	33h	Output horizontal display start
DSPHDEEND	R/W	34h	Output horizontal display end
DSPVSEND	R/W	35h	Output display vertical sync end position
BLINKCTRL	R/W	36h	OSD blinking control
DSPVDESTART	R/W	37h, 38h	Output vertical display start position
DSPVDEEND	R/W	39h, 3Ah	Output vertical display end position
FRAMEDELAY	R/W	3Bh	Output HSYNC delay adjustment relative to input HSYNC
FRAMEDELAYODD	R/W	3Ch	Output HSYNC delay adjustment relative to input HSYNC in odd field
Interrupt and internal timing			
IREQSOURCE	R/W	40h	Interrupt Source
REFHTOTAL	R/W	41h	Horizontal total of internal reference timing
REFVTOTAL	R/W	42h	Vertical total of internal reference timing
OUTPUTCONTROL	R/W	43h	Output Control
Look-up Table			
RLUTPORT	R/W	48h	Red LUT write data port
GLUTPORT	R/W	49h	Green LUT write data port
BLUTPORT	R/W	4Ah	Blue LUT write data port
LUTWADDR	R/W	4Bh	Address of LUT wrote port
OSD RAM			
OSDRAMWADDR	R/W	4Ch, 4Dh	OSD RAM write address
OSDRAMWPORT	R/W	4Eh	OSD RAM write port
Blanking/Border Control			
HBLANKSTART	R/W	50h, 51h	Horizontal blank start
HBLANKEND	R/W	52h, 53h	Horizontal blank end
VBLANKSTART	R/W	54h, 55h	Vertical blank start
VBLANKEND	R/W	56h, 57h	Vertical blank end
Input Timing Measurement			
INPUTSTATUS	R	60h	Input Status
LINERATE	R	61h, 62h	Input line rate, which is counted by clock from XIN pin

INVTOTAL	R	63h, 64h	Total vertical line count of input video
IREQSTATUS	R	65h	Interrupt Status
INHTOTAL	R	66h, 67h	Input horizontal pixel total
Automatic positioning			
HLINENUMBER	R/W	70h	Horizontal line number for horizontal active start and end detection
DATATHRESHOLD	R/W	71h	Data threshold value used to determine non-blanking pixel
HDESTART	R	72h, 73h	Detected horizontal active start pixel position
HDEEND	R	74h, 75h	Detected horizontal active end pixel position
VCOLUMN	R/W	79h	Vertical column for vertical active start and end detection
VDESTART	R	7Ah, 7Bh	Detected vertical active start line
VDEEND	R	7Ch, 7Dh	Detected vertical active end line
OSD Control			
OSDMODE	R/W	80h	OSD(On Screen Display) modes
FOREOP	R/W	81h	Logic operation
FADEALPHA	R/W	82h	Fading alpha value
OSD 1			
OSDCONTROL1	R/W	84h	OSD1 Control
ROMSTARTADDR1	R/W	85h	OSD1 ROM start address
FONTADDRUNIT1	R/W	86h	OSD1 font address unit
OSDHSTART1	R/W	90h	On Screen Display horizontal start position
OSDVSTART1	R/W	91h	On Screen Display vertical start position
RAMADDRST1	R/W	92h	OSD1 RAM start address
RAMSTRIDE1	R/W	93h, 8Bh	OSD1 RAM line stride
BMAPHSIZE1	R/W	94h, 95h	OSD1 horizontal bitmap size
BMAPHTOTAL1	R/W	96h, 97h	OSD1 bitmap horizontal total
BMAPVSIZE1	R/W	98h, 99h	OSD1 bitmap vertical size
BMAPVTOTAL1	R/W	9Ah, 9Bh	OSD1 bitmap vertical total
ICONHTOTAL1	R/W	9Ch	OSD1 horizontal icon total
ICONVTOTAL1	R/W	9Dh	OSD1 vertical icon total
FONTLINESIZE1	R/W	AEh	OSD1 font line size
OSD 2			
OSDCONTROL2	R/W	88h	OSD2 Control
ROMSTARTADDR2	R/W	89h	OSD2 ROM start address

FONTADDRUNIT2	R/W	8Ah	OSD2 font address unit
OSDHSTART2	R/W	A0h	On Screen Display horizontal start position
OSDVSTART2	R/W	A1h	On Screen Display vertical start position
RAMADDRST2	R/W	A2h	OSD2 RAM start address
RAMSTRIDE2	R/W	A3h, 8Ch	OSD2 RAM line stride
BMAPHSIZE2	R/W	A4h, A5h	OSD2 horizontal bitmap size
BMAPHTOTAL2	R/W	A6h, A7h	OSD2 bitmap horizontal total
BMAPVSIZE2	R/W	A8h, A9h	OSD2 bitmap vertical size
BMAPVTOTAL2	R/W	AAh,ABh	OSD2 bitmap vertical total
ICONHTOTAL2	R/W	ACh	OSD2 horizontal icon total
ICONVTOTAL2	R/W	ADh	OSD2 vertical icon total
FONTLINESIZE2	R/W	AFh	OSD2 font line size
OSD Color Registers			
COLOR0RED	R/W	B0h	Color 0 Red Component
COLOR0GREEN	R/W	B1h	Color 0 Green Component
COLOR0BLUE	R/W	B2h	Color 0 Blue Component
COLOR1RED	R/W	B3h	Color 1 Red Component
COLOR1GREEN	R/W	B4h	Color 1 Green Component
COLOR1BLUE	R/W	B5h	Color 1 Blue Component
COLOR2RED	R/W	B6h	Color 2 Red Component
COLOR2GREEN	R/W	B7h	Color 2 Green Component
COLOR2BLUE	R/W	B8h	Color 2 Blue Component
COLOR3RED	R/W	B9h	Color 3 Red Component
COLOR3GREEN	R/W	BAh	Color 3 Green Component
COLOR3BLUE	R/W	BBh	Color 3 Blue Component

8.1 Register Description

Configuration:

00h: Company ID (R) [COMPANYID]

CompanyId	<7:0>	Company ID (46h)

01h: Revision (R) [REVISION]

Revision	<7:0>	Revision number

02h: Board Configuration (R only; R/W for bit 0) [BOARDCONFIG]

If SoftCinfig (0x03<4>) = 0, this register setting represent board configuration.
If SoftCinfig (0x03<4>) = 1, bit 0 (only) is programmable by software. The other bits still represent board configuration.

VideoIn	<0>	0	Accept graphic input
		1	Accept video input
I^2C	<6>	0	Serial I^2C input
		1	Reserved
I^2C Addr	<7>	0	I^2C write address is 70h, read address is 71h.
		1	I^2C write address is 72h, read address is 73h.

03h: General (R/W) [GENERAL]

PwrDown	<0>	0	Normal operation
		1	Power down mode
SoftTiming	<1>	Should always be 1 to enable s/w timing configuration.	
DisablePll*	<2>	Should always be 0 to enable internal/external PLL.	
PllDivHs	<3>	0	Use input HSYNC as the display clock PLL input reference to generate output pixel clock. Use registers #10h and 11h to define the PLL divider.
		1	Use divided input HSYNC as display clock PLL input reference. It needs to be enabled when required output PLL divider is higher than 2047, but can be used any time when output pixel rate is higher than input pixel rate. The divider ratio is defined by registers #10h ~ #13h. Refer to registers #10h ~ #13h for additional reference.
SoftConfig	<4>	0	Input video type is defined by external pin VIDEOIN.
		1	Input video type is defined by bit 0 of register 02h. Refer to register 02h for additional reference.
Bypass	<5>	Video pass through without scaling.	
		0	Enable scaling defined in registers 14h, 15h, 16h and 17h
		1	Bypass
		When in Bypass mode, only output timing registers 30h, 31h, 37h, 38h, 39h and 3Ah affect the timing control. The PLL and input timing registers are ignored.	
Ddr12*	<6>	Reserved. Should be always 0.	
RefTiming	<7>	0	Normal operation
		1	Free running

In free running mode, external input HSYNC and VSYNC are disabled. Nominal input timing is defined by registers 41h and 42h instead. This is useful when external video source is disconnected or unstable. The reference clock is given by XIN pin. Refer to registers 36h, 41h and 42h for additional reference.

04h: Chip Family (R) [FAMILY]

Family	<7:0>	00110000, AL300 series

05h: Polarity (R/W) [POLARITY]

InvPllRef	<0>	0	Display clock PLL reference signal is of negative polarity
		1	Display clock PLL reference signal is of positive polarity
InvPllFb	<1>	0	Display clock PLL feedback signal is of negative polarity
		1	Display clock PLL feedback signal is of positive polarity
HsPol	<2>	0	Output HSYNC is of positive polarity
		1	Output HSYNC is of negative polarity
BlnkPol	<3>	0	Output display enable PDSPEN is of positive polarity
		1	Output display enable PDSPEN is of negative polarity
VsPol	<4>	0	Output VSYNC is positive polarity
		1	Output VSYNC is negative polarity
ABDelay	<5>	Delay A data so that A and B data outputs are of the same timing. Usually written as 1 for dual pixel panels. NC for single pixel panels. Please also program registers #07h and #43h accordingly for dual pixel panels.	
CsyncOut	<6>	0	Normal HSYNC output
		1	Send composite sync output from PHS pin
InvOddFiled	<7>	For CCIR601 or square pixel video input, this bit should be written as 1 for correct interlaced odd/even field detection. In case of error, rewrite this bit as 0.	

06h: Dither control (R/W) [DITHER]

Sharpness	<1:0>	Sharpness control	
		00	turn off sharpness control
		01	sharpness level 1
		10	sharpness level 2 (sharper than level 1)
		11	sharpness level 3 (sharpest)
Reserved	<2>		
Reserved	<3>	Should be always 1 for optimized output.	
DithMode	<5:4>	Dither mode select	
		00	RGB 888
		01	RGB 666
		10	RGB 444
		11	RGB 444 without dithering.
		It is recommended not to use dithering for graphics input for best sharpness.	
SoftTVRef	<6>	0	YUV video horizontal capture start is defined by the hardware TVREF pin

		1	YUV Video horizontal capture start is defined by register #20h and #21h. Note that when #20h values changes between odd and even, the output U/V may flip. #20h should stay either odd or even at all times to ensure correct U/V.
			When SoftTVRef is 0, register#20h<0> will be used as U/V Flip
			The counterpart for controlling graphic reference is defined by SoftGraRef (register #80h<6>)
Interpolation	<7>	0	turn on interpolation
		1	turn off interpolation (duplicate pixels only)

07h: Adjustment (R/W) [ADJUSTMENT]

Reserved	<7>	
HDEDelay	<6:4>	PDSPEN delay relative to output data phase
Reserved	<3:0>	

Optimized value is 00h for single pixel mode, 20h for dual pixel mode

Please also program registers #05h<5>, #43h<7>, and #43h<4> accordingly for dual pixel mode.

Output Clock PLL and Zoom Ratio:

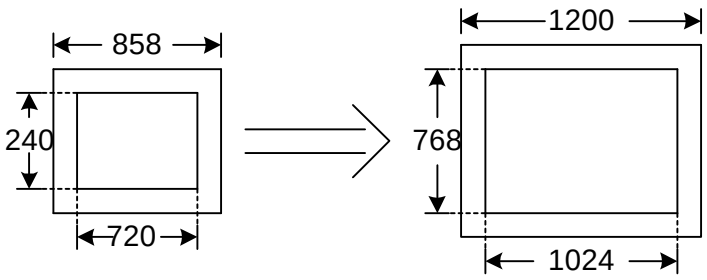
The output pixel clock is generated from the input HSYNC (or with input pixel clock) relative to the ratio between the output and input display resolution. The PLL divider (PLLDIV, defined by registers #10h and 11h) has a maximum value of 2047 programmed by an 11-bit word. PLLDIV is defined by the following equation:

$$PLLDIV(N) = \frac{Vo}{Vi} \bullet Ho$$

Where, Ho is the total number of pixels per output line
Vo is the output active lines per frame
Vi is the input active lines per frame

For the best scaling quality, PLLREFDIV (defined by registers #12h and #13h) is used as a PLL reference divider. In this case, PLLREFDIV (M) should be equal to the total number of pixels per input line (Hi). If N is greater than 2047, both N and M need to be adjusted by dividing the scaling ratio by a simple integer as in the following example:

Example: (N greater than 2047)



Input Display

Output Display

Vo = 768
Vi = 240
Ho = 1200
Therefore, using the equation for PLLDIV, N = 3840 which is much greater than 2047. The scaling ratio is:

$$\frac{N}{M} = \frac{3840}{858}$$

Since N is greater than 2047, the scaling ratio can be divided by 2 for the best scaling quality. Dividing this ratio by 2, the scaling ratio becomes:

$$\frac{N}{M} = \frac{1920}{429}$$

These scaled values are ideal for N and M. Note: Ideally, for the PLL, N should be an even integer.

10h: PLL Divider Low (R/W) [PLLDIV]

PllDivL	<7:0>	Bits <7:0> of PLL divider number

11h: PLL Divider High (R/W) [PLLDIV]

Feb2_d2	<7>	Feedback Divider 2 control 0: bypass Feedback Divider 2 1: enable Feedback Divider 2. Output clock frequency is doubled when the Feedback Divider is turned on.
Feb1_d2	<6>	Feedback Divider 1 control 0: bypass Feedback Divider 1 1: enable Feedback Divider 1. Output clock frequency is doubled when the Feedback Divider is turned on.
Out_d2	<5>	Output 1-bit Divider control 0: bypass Output Divider 2 1: enable Output Divider 2. Output clock frequency is divided by 2 when the Output Divider is turned on.
Pre_d2	<4>	Input 1-bit Divider control 0: bypass Input Divider 2

1: enable Input Divider 2. Output clock frequency is divided by 2 when the Input Divider is turned on.

PllDivH (10:8) <2:0> Bits 8~10 of the PLL divider

The value of PLLDIV is determined by subtracting from the actual target divider by 2 when internal PLL is used. For example, if the desired divider value is 840, then the value set to the related registers should be 838 = 346h. Therefore, PllDivH[2:0] should be set to 3h ("011") and PllDivL[7:0] should be set to 46h ("01000110").
However, when an external genlock PLL such as ICS9173 is used, programming registers 10h and 11h is still needed. In such applications, the PLLDIV should be the actual target divider instead of being subtracted by 2 as in the above case.
Please review the AL300Programming algorithm for more details.

12h: PLL Reference Divider Low (R/W) [PLLREFDIV]
PllRefDivL <7:0> Bits <7:0> of display PLL reference input divider number (Unit: 1 pixel)

13h: PLL Reference Divider High (R/W) [PLLREFDIV]
PllRefDivH <2:0> Bits <10:8> of display PLL reference input divider number

Registers 14h~17h are used to define the independent horizontal and vertical scaling ratio.

14h: Horizontal Zoom Ratio Low (R/W) [HRATIO]
HRatioL <7:0> Bits <7:0> of horizontal scale ratio
Value valid only when register 03h<5> = 0.

15h: Horizontal Zoom Ratio High (R/W) [HRATIO]
HRatioH <2:0> Bits <10:8> of horizontal scale ratio
Value valid only when register 03h<5> = 0.

Horizontal Zoom Ratio = (HDST x 256) / HSRC
HSRC = Horizontal source capture size (register 22h)
HDST = Horizontal destination active width (registers 33h and 34h)

16h: Vertical Zoom Ratio Low (R/W) [VRATIO]
VRatioL <7:0> Bits <7:0> of vertical scale ratio
Value valid only when register 03h<5> = 0.

17h: Vertical Zoom Ratio High (R/W) [VRATIO]
VRatioH <6:0> Bits <14:8> of vertical scale ratio
Value valid only when register 03h<5> = 0.

Vertical Zoom Ratio = (VDST x 4096) / VSRC

VSRC = (output vertical active line count x output horizontal line total) / N, or (Voa x Hot) / N. This value should be very close to Vertical source active lines (register 24h, 25h, 26h and 27h), but please keep the decimal.
VDST = Vertical destination active lines (registers 37h, 38h, 39h and 3Ah)
Note: when vertical zoom ratio is to be changed, input capture timing, PLL divider and M/N values should also be modified. Please refer to the AL300 Programming Algorithm for detailed description.

18h: Horizontal Zoom Initial Phase (R/W) [HINITPHASE]

HinitPh <7:0> Horizontal zoom scaler initial phase
The purpose of phase adjustment is to optimize the scaling quality when the scaling ratio is a simple fraction such as 2/1 or 3/2. The initial interpolation phase can be 0 up to close to 1 pixel,
HinitPh = (1 –Hia/2/Hoa) x 32. Please refer to the AL300 Programming Algorithm for detailed description.
The vertical counterpart is defined by register #1Ah

19h: Vertical Zoom Initial Phase at Odd Field (R/W) [VINITPHASEODD]

VinitPhOdd <7:0> Vertical zoom scalar initial phase for the odd field of interlaced input.
Enter the same value as register 1Ah for non-interlaced input.
Refer to the AL300 Programming Algorithm for more details.

1Ah: Vertical Zoom Initial Phase (R/W) [VINITPHASE]

VinitPh <7:0> Vertical zoom scalar initial phase.
This is the initial phase for graphics input or even field of interlaced input.
The recommended value is 20h.
The horizontal zoom initial phase is defined by register #18h

1Bh: Panel Power Control (R/W) [GOUT]

ControlEn	<0>		Enable panel interface control signals
Gout3	<1>		General purpose register for GOUT3 pin.
Gout1	<2>		General purpose register for GOUT1 pin.
Gout2	<3>		General purpose register for GOUT2 pin.
			The Gout bits are reserved in case additional control pins are required for specific panels.
YUV444In	<4>	0	YUV422 input
		1	YUV444 input (Set <5>=0)
CCIR601	<5>	0	CCIR601 input timing format
		1	Reserved
YUVOut	<6>	0	RGB output
		1	YUV output
YUV444Out	<7>	0	YUV422 output
		1	YUV444 output

Bits<7:4> is valid only if register #02h<0>, VideoIn is 1

Input timing:

Please refer to section 6.5 Input Video Timing for additional reference. The input timing diagram (AL300-07) shows it pictorially.

20h: Horizontal Capture Start Low (R/W) [CAPHSTART]
CapHStartL <7:0> Bits<7:0> of horizontal capture start position (Unit: 1 pixel)

21h: Horizontal Capture Start High (R/W) [CAPHSTART]
CapHStartH <2:0> Bits <10:8> of horizontal capture start position

22h: Horizontal Capture Size (R/W) [CAPHSIZE]
CapHSize <7:0> Horizontal capture size (Unit: 8 pixels)

23h: Testing (R/W) [RESETTEST]
Reserved <7:0> Reserved

24h: Vertical Capture Start Low (R/W) [CAPVSTART]
CapVStartH <7:0> Bits <7:0> of vertical capture start position (Unit: 1 line)

25h: Vertical Capture Start High (R/W) [CAPVSTART]
CapVStartL <2:0> Bits <10:8> of vertical capture start position

26h: Vertical Capture End Low (R/W) [CAPVEND]
CapVEndL <7:0> Bits <7:0> of vertical capture end position (Unit: 1 line)

27h: Vertical Capture End High (R/W) [CAPVEND]
CapVEndH <2:0> Bits <10:8> of vertical capture end position

Registers #28h and #29h are used to control the delay of vertical lines between the output and the input. The counterpart for horizontal delay is controlled by the registers #3Bh and #3Ch.

28h: Vertical display Frame Start Position Low (R/W) [FRAMEVSTART]
FrameVStartL <7:0> Bits <7:0> of display vertical counter start position relative to input vertical counter. (Unit: 1 line)

29h: Vertical display Frame Start Position High (R/W) [FRAMEVSTART]
FrameVStartH <2:0> Bits <10:8> of display vertical counter start position relative to input vertical counter.

Output timing:
Please refer to section 6.6 Output Video Timing for additional reference. The output timing diagram (AL300-08) shows it pictorially.

30h: Horizontal Total Low (R/W) [DSPHTOTAL]
DspHTotalL <7:0> Bits <7:0> of display horizontal total. (Unit: 1 pixel)

31h: Horizontal Total High (R/W) [DSPHTOTAL]

DspHTotalH <2:0> Bits <10:8> of display horizontal total.

32h: Horizontal Sync End (R/W) [DSPHSEND]

DspHsEnd <7:0> Output display horizontal sync end position (Unit: 8 pixels)

33h: Horizontal Display Start (R/W) [DSPHDESTART]

DspHdeStart <7:0> Output horizontal display start (Unit: 8 pixels)

34h: Horizontal Display End (R/W) [DSPHDEEND]

DspHdeEnd <7:0> Output horizontal display end (Unit: 8 pixels)

35h: Vertical Sync End (R/W) [DSPVSEND]

DspVsEnd <7:0> Output display vertical sync end position (Unit: 8 line)

36h: Blink Control (R/W) [BLINKCTRL]

BlinkTime <1:0> OSD blinking time constant
- 00 32 frames of internal reference timing per blink
- 01 64 frames of internal reference timing per blink
- 10 128 frames of internal reference timing per blink
- 11 256 frames of internal reference timing per blink

(Refer to registers 41h, 42h, 84h<1> and 88h<1> for additional reference)

BlinkType <2>
- 0 Blinking is defined in BlinkTime
- 1 No blinking, just reverse the index color

37h: Vertical Display Start Low (R/W) [DSPVDESTART]

DspVdeStartL <7:0> Bits <7:0> of vertical display start position. (Unit: 1 line)

38h: Vertical Display Start High (R/W) [DSPVDESTART]

DspVdeStartH <2:0> Bits <10:8> of vertical display start position.

39h: Vertical Display End Low (R/W) [DSPVDEEND]

DspVdeEndL <7:0> Bits <7:0> of vertical display end position. (Unit: 1 line)

3A: Vertical Display End High (R/W) [DSPVDEEND]

DspVdeEndH <2:0> Bits <10:8> of vertical display end position.

Registers #3Bh and #3Ch are used to control the delay of horizontal pixels between the output and the input. The counterpart for vertical delay is controlled by the registers #28h and #29h.

3B: Frame Head Delay Adjustment (R/W) [FRAMEDELAY]

FrameDly <7:0> Output HSYNC delay adjustment relative to input HSYNC (Unit: 16 pixels)

Refer to the AL300 Programming Algorithm for more details.

3C: Frame Head Delay Adjustment for Odd Field (R/W) [FRAMEDELAYODD]

FrameDlyOdd <7:0> Output HSYNC delay adjustment relative to input HSYNC in odd field. (Unit: 16 pixels)

For non-interlaced video, FrameDlyOdd has the same value as FrameDly.
For interlaced video, FrameDlyOdd is half line more than FrameDly.
Refer to the AL300 Programming Algorithm for more details.

Interrupt and internal timing:

40h: Interrupt Source (R/W) [IREQSOURCE]

LineRateIreq	<0>	If 1, enable interrupt if line rate changes
VSYNCIreq	<1>	If 1, enable interrupt if VSYNC comes
OddFieldIreq	<2>	If 1, enable interrupt when odd field of input video comes

To check what causes the interruption when more than one Interrupt Source is enabled, read register #65h for the status and then clear it for the next interruption.

41h: Reference H Total (R/W) [REFHTOTAL]

RefHTotal <7:0> Horizontal total of internal reference timing (Unit: 8 pixels).
Enabled when register 03h<7>=1

Registers #41h and #42h are used to define free running mode when input is disconnected or is undefined. This is especially useful for showing error messages with OSD. Refer to register #42h for examples.

42h: Reference V Total (R/W) [REFVTOTAL]

RefVTotal <7:0> Vertical total of internal reference timing (Unit: 8 lines)
Enabled when register 03h<7>=1

Register 41h and 42h are used to define free running mode when input is disconnected or is undefined. This is especially useful for showing error message with OSD.
For instance, a reference clock of 14.31818 MHz is used to generate HSYNC and VSYNC with vertical frame refresh rate of 60Hz, the resulted H-total and V-total would be as follows:

PANEL	H-total (Reg.#41h x 8)	V-total (Reg.#42h x 8)
640x480:	456	520
800x600:	376	632
1024x768:	296	800
1280x1024	216	1072

To correct the H-total if necessary, program the input PLL circuitry to generate correct input pixel clock rate.

43h: Output Control (R/W) [OUTPUTCONTROL]

Reserved	<2:0>	Should be written as 000.
NoBlank	<3>	No blanking, all data pass through; for testing only.
DualOut	<4>	output odd/even pixels for dual pixel panels
LutEn	<5>	Enable LUT

ColorOut	<6>	Replace the input video with color 3 (refer to registers #B9h, #BAh, and #BBh for additional reference)
InvOutClk	<7>	Invert the phase of SCLK, PCLKA, and PCLKB. This bit is usually 1 for dual pixel panels.

Please also program registers #05h and #07h accordingly for dual pixel panels.

65h: Interrupt Status (R only) [IREQSTATUS]

Please refer to the register 65h description in the Input Timing Measurement section, after register 64h.

Look-up Table:

48h: Red LUT write data port (R/W) [RLUTPORT]

RLutPort	<7:0>	Red LUT write data port

49h: Green LUT write data port (R/W) [GLUTPORT]

GLutPort	<7:0>	Green LUT write data port

4Ah: Blue LUT write data port (R/W) [BLUTPORT]

BLutPort	<7:0>	Blue LUT write data port

4Bh: LUT write port address (R/W) [LUTWADDR]

LutWrAddr	<7:0>	Address of LUT wrote port

OSD RAM:

4Ch: OSD RAM address Low byte (R/W) [OSDRAMWADDR]

OsdRamAddrL	<7:0>	Bits <7:0> of OSD RAM write address

4Dh: OSD RAM address high byte (R/W) [OSDRAMWADDR]

OsdRamAddrH	<1:0>	Bits <9:8> of OSD RAM write address

4Eh: OSD RAM write data port (R/W) [OSDRAMWPORT]

OsdRamPort	<7:0>	OSD RAM write port

Writing data to this register #4Eh automatically increases the OSD RAM address defined in registers #4Ch and #4Dh.

Blanking/border control:

The blanking control is used as output data cropping

50h: Horizontal Blank Start Low (R/W) [HBLANKSTART]

HBlankStartL	<7:0>	Bits <7:0> of Horizontal blank start position + 12. (Unit: 1 pixel)

51h: Horizontal Blank Start High (R/W) [HBLANKSTART]
HBlankStartH <2:0> Bits <10:8> of Horizontal blank start position + 12. (Unit: 1 pixel)

52h: Horizontal Blank End Low (R/W) [HBLANKEND]
HBlankEndL <7:0> Bits <7:0> of Horizontal blank end position + 12. (Unit: 1 pixel)

53h: Horizontal Blank End High (R/W) [HBLANKEND]
HBlankEndH <2:0> Bits <10:8> of Horizontal blank end position + 12. (Unit: 1 pixel)

54h: Vertical Blank Start Low (R/W) [VBLANKSTART]
VBlankStartL <7:0> Bits <7:0> of Vertical blank start position. (Unit: 1 line)

55h: Vertical Blank Start High (R/W) [VBLANKSTART]
VBlankStartH <2:0> Bits <10:8> of Vertical blank start position. (Unit: 1 line)

56h: Vertical Blank End Low (R/W) [VBLANKEND]
VBlankEndL <7:0> Bits <7:0> of Vertical blank end position. (Unit: 1 line)

57h: Vertical Blank End High (R/W) [VBLANKEND]
VBlankEndH <2:0> Bits <10:8> of Vertical blank end position. (Unit: 1 line)

Input Timing Measurement:

60h: Input Status (R only) [INPUTSTATUS]

InHsPol	<0>	Input HSYNC Polarity. 0 Active low/ negative polarity 1 Active high/ positive polarity
InVsPol	<1>	Input VSYNC polarity. 0 Active low/ negative polarity 1 Active high/ positive polarity
OddFIeld	<2>	1, if input is in odd field
Reserved	<3>	
InVde	<4>	1, if input video is in active region
OutVde	<5>	1, if output video is in active region
RefVs	<6>	1, if refence timing is in VSYNC cycle (in free running mode)
InVSYNC	<7>	1, if input timing is in VYSNC cycle

61h: Input Line Rate Low (R only) [LINERATE]
LineRateL <7:0> Bits <7:0> of input line rate, which is in terms of how many reference lock cycles per horizontal line.
Refresh Rate = (Reference clock frequency) / (LINERATE x INVTOTAL)
Reference clock is the clock from XIN pin.
LINERATE is the input line rate defined in registers #61h and #62h
INVTOTAL is the input vertical total line defined in registers #63h and #64h

62h: Input Line Rate High (R only) [LINERATE]

LineRateH	<3:0>	Bits <11:8> of input line rate, which is in terms of how many reference lock cycle per horizontal line.

63h: Input Vertical Line Total Low (R only) [INVTOTAL]

InVTotalL	<7:0>	Bits <7:0> of total vertical line count of input video

Refresh Rate = (Reference clock frequency) / (LINERATE x INVTOTAL)
Reference clock is the clock from XIN pin.
LINERATE is the input line rate defined in registers #61h and #62h
INVTOTAL is the input vertical total line defined in registers #63h and #64h

64h: Input Vertical Line Total High (R only) [INVTOTAL]

InVTotalH	<3:0>	Bits <11:8> of total vertical line count of input video

65h: Interrupt Status (R only) [IREQSTATUS]

NewHSYNCRate	<0>	1, if HSYNC line rate of input video changes. Reset by clearing the related bit of register #40h, then enable it if desired.
NewVysnc	<1>	1, if VSYNC of input video happens. Reset by clearing the related bit of register #40h, then enable it if desired.
NewField	<2>	1, if odd field of input video comes. Reset by clearing the related bit of register #40h, then enable it if desired.

66h: Input Horizontal Pixel Total Low (R only) [INHTOTAL]

InHTotalL	<7:0>	Bits<7:0> of total horizontal pixel count of input video

67h: Input Horizontal Pixel Total High (R only) [INHTOTAL]

InHTotalH	<2:0>	Bits<10:8> of total horizontal pixel count of input video

The two registers can be used to reconfirm if the input pixel-total number is reasonable or as expected. For instance, an odd number may indicate that the PLL of the preceding ADC may not have worked as how it is programmed.

Automatic positioning:

70h: Horizontal Line Number for HDEST & HDEEND detection (R/W) [HLINENUMBER]

HLineNumber	<7:0>	Horizontal line number for horizontal active start and end detection; refer to register #71h for additional reference. (Unit: 8 lines)

71h: Data Threshold (R/W) [DATATHRESHOLD]

DataThreshold	<6:0>	Luma (brightness) threshold value. DataThreshold is used to determine whether an input pixel exists for both horizontal and vertical directions. Any pixel luma value less than this value is regarded as blank.
LinePosDetect	<7>	0 whole frame active data detection. 1 Single line/column active data detection.

When LinePosDetect is 1, the horizontal line used to detect horizontal active start and end is defined in register 70h, and the vertical column used to detect vertical active start and end is defined in register 79h.

Values of register #72h~#75h reflect the detected input active pixels start/end positions or the GHREF signal star/end positions, depending on the status of register #82h<7:6>.

72h: Horizontal Active Start Low (R only) [HDESTART]

HDEStartL	<7:0>	Bits <7:0> of detected horizontal active start pixel position. (Unit: 1 pixel)

73h: Horizontal Active Start High (R only) [HDESTART]

HDEStartH	<2:0>	Bits <10:8> of detected horizontal active start pixel position.

74h: Horizontal Active End Low (R only) [HDEEND]

HDEEndL	<7:0>	Bits <7:0> of detected horizontal active end pixel position (Unit: 1 pixel)

75h: Horizontal Active End High (R only) [HDEEND]

HDEEndH	<2:0>	Bits <10:8> of detected horizontal active end pixel position.

79h: Vertical Column for VDESTART & VDEEND detection (R/W) [VCOLUMN]

VColumn	<7:0>	Vertical column number for vertical active start and end detection; refer to register #71h for additional reference. (Unit: 8 pixels)

7Ah: Vertical Active Start Low (R only) [VDESTART]

VDEStartL	<7:0>	Bits <7:0> of detected vertical active start line (Unit: 1 line)

7Bh: Vertical Active Start High (R only) [VDESTART]

VDEStartH	<2:0>	Bits <10:8> of detected vertical active start line

7Ch: Vertical Active End Low (R only) [VDEEND]

VDEEndL	<7:0>	Bits <7:0> of detected vertical active end line (Unit: 1 line)

7Dh: Vertical Active End High (R only) [VDEEND]

VDEEndH	<2:0>	Bits <10:8> of detected vertical active end line

OSD Control:

80h: OSD Modes (R/W) [OSDMODE]

RomMode	<0>	Enable ROM mode 0 Internal RAM mode 1 External ROM mode This bit should be 0 when OSD1 and OSD2 are both disabled.
BypassHPos	<5>	0 The output horizontal start/end position is defined by register #33h & #34h, which resolution is 8-pixel. 1 the output horizontal start/end position is defined by registers #20h~22h

SoftGraRef <6> 0 Graphic horizontal capture start is defined by the hardware GHREF pin
1 Graphic horizontal capture start is defined by register #20h and #21h.
The counterpart for controlling graphic reference is defined by SoftTVRef (register #06h<6>)

81h: Logic Operation (R/W) [FOREOP]

Color0Op <1:0> Logic operation between color 0 and video
00 NOP show only OSD
01 OR video or color 0
10 AND video and color 0
11 XOR video xor color 0

Color1Op <3:2> Logic operation between color 1 and video
00 NOP show only OSD
01 OR video or color 1
10 AND video and color 1
11 XOR video xor color 1

Color2Op <5:4> Logic operation between color 2 and video
00 NOP show only OSD
01 OR video or color 2
10 AND video and color 2
11 XOR video xor color 2

Color3Op <7:6> Logic operation between color 3 and video
00 NOP show only OSD
01 OR video or color 3
10 AND video and color 3
11 XOR video xor color 3

Color 0, 1, 2 and 3 are defined in registers B0h ~ BBh.

82h: Fading Alpha Value (R/W) [FADEALPHA]

FadeAlpha <5:0> The alpha factor for fading effect ranging from 00h to 20h, i.e., there is 33-level of fade-in/fade-out effect.
Output = input * alpha + OSD * (1 – (alpha/32))
000000 - minimum alpha value. Show only OSD.
100000 - maximum alpha value. Show only video

GHREFPol <7:6> 00 Auto positioning registers #72h~#75h controlled by hardware internal detection.
01 Reserved
10 Auto positioning registers #72h~#75h controlled by GHREF pin, if GHREF is active high.
11 Auto positioning registers #72h~#75h controlled by GHREF pin, if GHREF is active low.

On Screen Display 1 Registers:

84h: OSD1 Control (R/W) [OSDCONTROL1]

PixDepth1 <0> Number of bits per pixel of On Screen Display (OSD) 1

0 One bit per pixel
1 Two bits per pixel

BlinkEn1 <1> OSD1 blinking enable, effective only when RomMode = '1'.
0 Disable blinking
1 Enable blinking

HZoom1 <3:2> OSD1 horizontal zoom factor
00 OSD pixel H size equals to 1X of video pixel
01 OSD pixel H size equals to 2X of video pixel
10 OSD pixel H size equals to 4X of video pixel
11 OSD pixel H size equals to 8X of video pixel

VZoom1 <5:4> OSD1 vertical zoom factor
00 OSD line V size equals to 1X of video line
01 OSD line V size equals to 2X of video line
10 OSD line V size equals to 4X of video line
11 OSD line V size equals to 8X of video line

Font2Byte1 <6> Two-byte font charter code mode, effective only when RomMode = '1'
0 One-byte character code mode
1 Two-byte character code mode
Please refer to OSD data addressing in ROM mode diagram (AL300-11)

OsdEn1 <7> On Screen Display (OSD) 1 enable
0 Disable OSD1; 80h<0> should be 0 when OSD1 and OSD2 are both disabled.
1 Enable OSD1

85h: OSD1 ROM Start Address (R/W) [ROMSTARTADDR1]

RomStAddr1H <7:0> Bits<11:4> of OSD1 ROM start address (Unit: 16 bytes)
Bits<3:0> of OSD1 ROM start address is defined in register #86h<3:0>

OSD1 ROM start address = RomStAddr1H * 256 + RomStAddr1L * 16

86h: OSD1 Font Address Unit (R/W) [FONTADDRUNIT1]

FontAddrUnit1 <7:4> OSD1 font address unit
0000 font address is multiple of 2^5 bytes
0001 font address is multiple of 2^6 bytes
0010 font address is multiple of 2^7 bytes
0011 font address is multiple of 2^8 bytes
0100 font address is multiple of 2^9 bytes
0101 font address is multiple of 2^{10} bytes
0110 font address is multiple of 2^{11} bytes
0111 font address is multiple of 2^{12} bytes
1000 font address is multiple of 2^{13} bytes
1001 font address is multiple of 2^{14} bytes
1010 font address is multiple of 2^{15} bytes
1011 font address is multiple of 2^{16} bytes

RomStAddr1L <3:0> Bits<3:0> OSD1 ROM start address (Unit: 16 bytes)

Font address = (Character Code) * (Font Address Unit) + (OSD1 ROM Start Address)
'Character Code' is the data retrieved from internal OSD RAM.

'Font Address Unit' is defined in register #86h.
'OSD1 ROM Start Address' is defined in registers #85h and #86h.

90h: OSD1 Horizontal Start (R/W) [OSDHSTART1]

OsdHStart1 <7:0> On Screen Display horizontal start position (Unit: 8 video pixels)
Relative to display HSYNC start

Please also refer to OSD screen timing diagram (AL300-15) for additional reference.

91h: OSD1 Vertical Start (R/W) [OSDVSTART1]

OsdVStart1 <7:0> On Screen Display vertical start position (Unit: 8 video lines)
Relative to display VSYNC start

Please also refer to OSD screen timing diagram (AL300-15) for additional reference.

92h: OSD1 RAM Start Address (R/W) [RAMADDRST1]

RamAddrSt1 <7:0> OSD1 RAM start address (Unit: 4 or 8 bytes)

OSD1 RAM Start Address =

1. Register 92h<7:0> * 4, if register 84h<6> = 0
2. Register 92h<7:0> * 8, if register 84h<6> = 1

AL300 has built in 1K Byte RAM

In RAM mode (register 80h<0> = 0):

RAM start address defines the start address of stored internal OSD RAM bitmap.

In ROM mode (register 80h<0> = 1):

RAM start address defines the start address of OSD ROM font character codes retrieved from internal OSD RAM.

font address = (Character Code) * (Font Address Unit) + (OSD1 ROM Start Address)
'Character Code' is the data retrieved from internal OSD RAM.
'Font Address Unit' is defined in register #86h.
'OSD1 ROM Start Address' is defined in register #85h.

93h: OSD1 RAM Horizontal Stride Low (R/W) [RAMSTRIDE1]

RamStride1L <7:0> Bits<7:0> of OSD1 RAM line stride (Unit: 1 bytes)

In RAM mode (register 80h<0> = 0):

RAM horizontal stride defines the number of bytes occupied in internal OSD RAM for each OSD bitmap line.
Total bitmap bytes stored in OSD RAM:
'RAM Horizontal Stride' x 'Bitmap Vertical Size' (bytes)
where
'RAM Horizontal Stride' is defined by register #8Bh and #93h.
'Bitmap Vertical Size' is defined by register #98h and #99h.

In ROM mode (register 80h<0> = 1):

RAM horizontal stride defines the total bytes occupied in internal OSD RAM for each OSD text row.
Total character code bytes stored in OSD RAM:
'RAM Horizontal Stride' x 'Icon Vertical Total' (bytes)
Total character code bytes retrieved in OSD RAM:
'Icon Horizontal Total' x 'Icon Vertical Total' x 'Character code size' (bytes)

Where
'RAM Horizontal Stride' is defined in register #8Bh and #93h.
'Icon Horizontal Total' is defined in register #9Ch.
'Icon Vertical Total' is defined in register #9Dh.
'Character Code Size' is defined in register #84h<6>.

8Bh: OSD1 RAM Horizontal Stride High (R/W) [RAMSTRIDE1]
RamStride1H <1:0> Bits <9:8> of OSD1 RAM line stride (Unit: 1 bytes)
The lower byte is defined by register #93h.

94h: OSD1 Bitmap Horizontal Size Low (R/W) [BMAPHSIZE1]
BMAPHSIZE1 = Actual Bitmap Horizontal Size – 1
BmapHSize1L <7:0> Bits<7:0> of OSD1 horizontal bitmap size (Unit: 1 OSD pixel)
Value of ('Bitmap Horizontal Total' – 'Bitmap Horizontal Size') defines the extra gap between each character shown on OSD screen.
'Bitmap Horizontal Size' is defined by register #94h and #95h.
'Bitmap Horizontal Total' is defined by register #96h and #97h.
Please refer to OSD screen timing diagram (AL300-15) for additional reference.

95h: OSD1 Bitmap Horizontal Size High (R/W) [BMAPHSIZE1]
BMAPHSIZE1 = Actual Bitmap Horizontal Size – 1
BmapHSize1H <1:0> Bits<9:8> of OSD1 bitmap horizontal size
The lower byte is defined by register #94h.

96h: OSD1 Bitmap Horizontal Total Pixels Low (R/W) [BMAPHTOTAL1]
BMAPHTOTAL1 = Actual Bitmap Horizontal total – 1
BmapHTotal1L <7:0> Bits<7:0> of OSD1 bitmap horizontal total (Unit: 1 OSD pixel)
Value of ('Bitmap Horizontal Total' – 'Bitmap Horizontal Size') defines the extra gap between each character shown on OSD screen.
'Bitmap Horizontal Size' is defined by register 94h and 95h.
'Bitmap Horizontal Total' is defined by register 96h and 97h.
Please refer to OSD screen timing diagram (AL300-15) for additional reference.

97h: OSD1 Bitmap Horizontal Total Pixels High (R/W) [BMAPHTOTAL1]
BMAPHTOTAL1 = Actual Bitmap Horizontal Total – 1
BmapHTotal1H <1:0> Bits<9:8> of OSD1 bitmap horizontal total
The lower byte is defined by register #96h.

98h: OSD1 Bitmap Vertical Size Low (R/W) [BMAPVSIZE1]
BMAPVSIZE1 = Actual Bitmap Vertical Size – 1
BmapVSize1L <7:0> Bits<7:0> of OSD1 bitmap vertical size (Unit: 1 OSD line)
Value of ('Bitmap Vertical Total' – 'Bitmap Vertical Size') defines the extra line(s) between each text row shown on OSD screen.
'Bitmap Vertical Size' is defined by register #98h and #99h.
'Bitmap Vertical Total' is defined by register #9Ah and #9Bh.
Please refer to OSD screen timing diagram (AL300-15) for additional reference.

99h: OSD1 Bitmap Vertical Size High (R/W) [BMAPVSIZE1]

BMAPVSIZE1 = Actual Bitmap Vertical Size – 1
BmapVSize1H <1:0> Bits<9:8> of OSD1 bitmap vertical size
The lower byte is defined by register #98h.

9Ah: OSD1 Bitmap Vertical total Lines Low (R/W) [BMAPVTOTAL1]

BMAPVTOTAL1 = Actual Bitmap Vertical Total – 1
BmapVTotal1L <7:0> Bits<7:0> of OSD1 bitmap vertical total (Unit: 1 OSD line)
Value of ('Bitmap Vertical Total' – 'Bitmap Vertical Size') defines the extra line(s) between each text row shown on OSD screen.
'Bitmap Vertical Size' is defined by register #98h and #99h.
'Bitmap Vertical Total' is defined by register #9Ah and #9Bh.
Please refer to OSD screen timing diagram (AL300-15) for additional reference.

9Bh: OSD1 Bitmap Vertical Total Lines High (R/W) [BMAPVTOTAL1]

BMAPVTOTAL1 = Actual Bitmap Vertical Total – 1
BmapVTotal1H <1:0> Bits<9:8> of OSD1 bitmap vertical total
The lower byte is defined by register #9Ah.

9Ch: OSD1 Icon Horizontal Total (R/W) [ICONHTOTAL1]

ICONHTOTAL1 = actual icon horizontal total – 1
IconHtotal1 <7:0> OSD1 horizontal icon total (Unit: 1 icon)
Icon Horizontal total defines how many character codes should be retrieved from internal OSD RAM and shown on OSD screen per OSD line.
Total character code bytes stored in OSD RAM:
'RAM Horizontal Stride' x 'Icon Vertical Total' (bytes)
Total character code bytes retrieved in OSD RAM:
'Icon Horizontal Total' x 'Icon Vertical Total' x 'Character Code Size' (bytes)
Where
'RAM Horizontal Stride' is defined by register #8Bh and #93h.
'Icon Horizontal Total' is defined by register #9Ch.
'Icon Vertical Total' is defined by register #9Dh.
'Character Code Size' is defined by register #84h<6>.
Please refer to OSD screen timing diagram (AL300-15) for additional reference.

9Dh: OSD1 Icon Vertical Total (R/W) [ICONVTOTAL1]

ICONVTOTAL1 = actual vertical icon total – 1
IconVTotal1 <7:0> OSD1 vertical icon total (Unit: 1 icon)
Icon vertical total defines how many text rows shown on OSD screen.
Total character code bytes stored in OSD RAM:
'RAM Horizontal Stride' x 'Icon Vertical Total' (bytes)
Total character code bytes retrieved in OSD RAM:
'Icon Horizontal Total' x 'Icon Vertical Total' x 'Character code size' (bytes)
Where
'RAM Horizontal Stride' is defined by register #8Bh and #93h.
'Icon Horizontal Total' is defined by register #9Ch.

'Icon Vertical Total' is defined by register #9Dh.
'Character Code Size' is defined by register #84h<6>.
Please refer to OSD screen timing diagram (AL300-15) for additional reference.

AEh: OSD1 Font Line Size (R/W) [FONTLINESIZE1]

Fontlinesize1 <7:0> memory size of a line of font (Unit: 1 byte)
Registers #94h and #95h define the size by bits, but #AEh defines it by bytes.

On Screen Display 2 Registers:

88h: OSD2 Control (R/W) [OSDCONTROL2]

PixDepth2 <0> Number of bits per pixel of On Screen Display (OSD) 2
- 0 One bit per pixel
- 1 Two bits per pixel

BlinkEn2 <1> OSD2 blinking enable, effective only when RomMode = '1'.
- 0 Disable blinking
- 1 Enable blinking

HZoom2 <3:2> OSD2 horizontal zoom factor
- 00 OSD pixel H size equals to 1X of video pixel
- 01 OSD pixel H size equals to 2X of video pixel
- 10 OSD pixel H size equals to 4X of video pixel
- 11 OSD pixel H size equals to 8X of video pixel

VZoom2 <5:4> OSD2 vertical zoom factor
- 00 OSD line V size equals to 1X of video line
- 01 OSD line V size equals to 2X of video line
- 10 OSD line V size equals to 4X of video line
- 11 OSD line V size equals to 8X of video line

Font2Byte2 <6> Two-byte font charter code mode, effective only when RomMode = '1'
- 0 One-byte character code mode
- 1 Two-byte character code mode

Please refer OSD data addressing in ROM mode diagram (AL300-11)

OsdEn2 <7> On Screen Display (OSD) 2 enable
- 0 Disable OSD2; 80h<0> should be 0 when OSD1 and OSD2 are both disabled.
- 1 Enable OSD2

89h: OSD2 ROM Start Address (R/W) [ROMSTARTADDR2]

RomStAddr2H <7:0> Bits<11:4> of OSD2 ROM start address (Unit: 16 bytes)
Bits<3:0> of OSD2 ROM start address is defined in register #8Ah<3:0>

OSD1 ROM start address = RomStAddr1H * 256 + RomStAddr1L * 16

8Ah: OSD2 Font Address Unit (R/W) [FONTAADDRUNIT2]

FontAddrUnit2 <7:4> OSD2 font address unit
- 0000 font address is multiple of 2^5 bytes
- 0001 font address is multiple of 2^6 bytes

0010 font address is multiple of 2^7 bytes
0011 font address is multiple of 2^8 bytes
0100 font address is multiple of 2^9 bytes
0101 font address is multiple of 2^{10} bytes
0110 font address is multiple of 2^{11} bytes
0111 font address is multiple of 2^{12} bytes
1000 font address is multiple of 2^{13} bytes
1001 font address is multiple of 2^{14} bytes
1010 font address is multiple of 2^{15} bytes
1011 font address is multiple of 2^{16} bytes

RomStAddr2L <3:0> Bits<3:0> OSD2 ROM start address (Unit: 16 bytes)
Font address = (Character Code) * (Font Address Unit) + (OSD2 ROM Start Address)
'Character Code' is the data retrieved from internal OSD RAM.
'Font Address Unit' is defined by register #8Ah.
'OSD2 ROM Start Address' is defined by register #89h and #8Ah.

A0h: OSD2 Horizontal Start (R/W) [OSDHSTART2]

OsdHStart2 <7:0> On Screen Display horizontal start position (Unit: 8 video pixels)
Relative to display HSYNC start
Please refer to OSD screen timing diagram (AL300-15) for additional reference.

A1h: OSD2 Vertical Start (R/W) [OSDVSTART2]

OsdVStart2 <7:0> On Screen Display vertical start position (Unit: 8 video lines)
Relative to display VSYNC start
Please refer to OSD screen timing diagram (AL300-15) for additional reference.

A2h: OSD2 RAM Start Address (R/W) [RAMADDRST2]

RamAddrSt2 <7:0> OSD2 RAM start address (Unit: 4 or 8 bytes)
OSD2 RAM Start Address =
1. Register A2h<7:0> * 4, if register 88h<6> = 0
2. Register A2h<7:0> * 8, if register 88h<6> = 1

AL300 has built in 1K Byte RAM
In RAM mode (register 80h<0> = 0):
RAM start address defines the start address of stored internal OSD RAM bitmap.
In ROM mode (register 80h<0> = 1):
RAM start address defines the start address of OSD ROM font character codes retrieved from internal OSD RAM.
font address = (Character Code) * (Font Address Unit) + (OSD2 ROM Start Address)
Where
'Character Code' is the data retrieved from internal OSD RAM.
'Font Address Unit' is defined by register #8Ah.
'OSD2 ROM Start Address' is defined by register #89h.

A3h: OSD2 RAM Horizontal Stride Low (R/W) [RAMSTRIDE2]

RamStride2L <7:0> Bits<7:0> of OSD2 RAM line stride (Unit: 1 bytes)
In RAM mode (register 80h<0> = 0):

RAM horizontal stride defines the amount of bytes occupied in internal OSD RAM for each OSD bitmap line.
Total bitmap bytes stored in OSD RAM:
'RAM Horizontal Stride' x 'Bitmap Vertical Size' (bytes)
Where
'RAM Horizontal Stride' is defined by registers #8Ch and #A3h.
'Bitmap Vertical Size' is defined by registers #A8h and #A9h.

In ROM mode (register 80h<0> = 1):
RAM horizontal stride defines the total bytes occupied in internal OSD RAM for each OSD text row.
Total character code bytes stored in OSD RAM:
'RAM Horizontal Stride' x 'Icon Vertical Total' (bytes)
Total character code bytes retrieved in OSD RAM:
'Icon Horizontal Total' x 'Icon Vertical Total' x 'Character code size' (bytes)
Where
'RAM Horizontal Stride' is defined by register #8Bh and #93h.
'Icon Horizontal Total' is defined by register #9Ch.
'Icon Vertical Total' is defined by register #9Dh.
'Character Code Size' is defined by register #84h<6>.
Please refer to OSD screen timing diagram (AL300-15) for additional reference.

8Ch: OSD2 RAM Horizontal Stride High (R/W) [RAMSTRIDE2]
RamStride2H <1:0> Bits<9:8> of OSD2 RAM line stride (Unit: 1 bytes)
Refer to register #A3h for additional reference.

A4h: OSD2 Bitmap Horizontal Size Low (R/W) [BMAPHSIZE2]
BMAPHSIZE2 = Actual Bitmap Horizontal Size – 1
BmapHSize2L <7:0> Bits<7:0> of OSD2 horizontal bitmap size (Unit: 1 OSD pixel)
Value of ('Bitmap Horizontal Total' – 'Bitmap Horizontal Size') defines the extra gap between each character shown on OSD screen.
'Bitmap Horizontal Size' is defined by registers #A4h and #A5h.
'Bitmap Horizontal Total' is defined by registers #A6h and #A7h.
Please refer to OSD screen timing diagram (AL300-15) for additional reference.

A5h: OSD2 Bitmap Horizontal Size High (R/W) [BMAPHSIZE2]
BMAPHSIZE2 = Actual Bitmap Horizontal Size – 1
BmapHSize2H <1:0> Bits<9:8> of OSD2 bitmap horizontal size
The lower byte is defined by register #A4h.

A6h: OSD2 Bitmap Horizontal Total Pixels Low (R/W) [BMAPHTOTAL2]
BMAPHTOTAL2 = Actual Bitmap Horizontal total – 1
BmapHTotal2L <7:0> Bits<7:0> of OSD2 bitmap horizontal total (Unit: 1 OSD pixel)
Value of ('Bitmap Horizontal Total' – 'Bitmap Horizontal Size') defines the extra gap between each character shown on OSD screen.
'Bitmap Horizontal Size' is defined by registers #A4h and #A5h.
'Bitmap Horizontal Total' is defined by registers #A6h and #A7h.

Please refer to OSD screen timing diagram (AL300-15) for additional reference.

A7h: OSD2 Bitmap Horizontal Total Pixels High (R/W) [BMAPHTOTAL2]

BMAPHTOTAL2 = Actual Bitmap Horizontal Total – 1
BmapHTotal2H <1:0> Bits<9:8> of OSD2 bitmap horizontal total
The lower byte is defined by register #A6h.

A8h: OSD2 Bitmap Vertical Size Low (R/W) [BMAPVSIZE2]

BMAPVSIZE2 = Actual Bitmap Vertical Size – 1
BmapVSize2L <7:0> Bits<7:0> of OSD1 bitmap vertical size (Unit: 1 OSD line)
Value of ('Bitmap Vertical Total' – 'Bitmap Vertical Size') defines the extra line(s) between each text row shown on OSD screen.
'Bitmap Vertical Size' is defined by registers #A8h and #A9h.
'Bitmap Vertical Total' is defined by registers #AAh and #ABh.
Please refer to OSD screen timing diagram (AL300-15) for additional reference.

A9h: OSD2 Bitmap Vertical Size High (R/W) [BMAPVSIZE2]

BMAPVSIZE2 = Actual Bitmap Vertical Size – 1
BmapVSize2H <1:0> Bits<9:8> of OSD2 bitmap vertical size
The lower byte is defined by register #A8h.

AAh: OSD2 Bitmap Vertical total Lines Low (R/W) [BMAPVTOTAL2]

BMAPVTOTAL2 = Actual Bitmap Vertical Total – 1
BmapVTotal2L <7:0> Bits<7:0> of OSD2 bitmap vertical total (Unit: 1 OSD line)
Value of ('Bitmap Vertical Total' – 'Bitmap Vertical Size') defines the extra line(s) between each text row shown on OSD screen.
'Bitmap Vertical Size' is defined by registers #A8h and #A9h.
'Bitmap Vertical Total' is defined by registers #AAh and #ABh.
Please refer to OSD screen timing diagram (AL300-15) for additional reference.

ABh: OSD2 Bitmap Vertical Total Lines High (R/W) [BMAPVTOTAL2]

BMAPVTOTAL2 = Actual Bitmap Vertical Total – 1
BmapVTotal2H <1:0> Bits<9:8> of OSD2 bitmap vertical total
The lower byte is defined by register #AAh.

ACh: OSD2 Icon Horizontal Total (R/W) [ICONHTOTAL2]

ICONHTOTAL2 = actual icon horizontal total – 1
IconHTotal2 <7:0> OSD1 horizontal icon total (Unit: 1 icon)
Icon Horizontal total defines how many character codes should be retrieved from internal OSD RAM and shown on OSD screen per OSD line.
Total character code bytes stored in OSD RAM:
'RAM Horizontal Stride' x 'Icon Vertical Total' (bytes)
Total character code bytes retrieved in OSD RAM:
'Icon Horizontal Total' x 'Icon Vertical Total' x 'Character code size' (bytes)
Where
'RAM Horizontal Stride' is defined by registers #8Ch and #A3h.
'Icon Horizontal Total' is defined by register #ACh.

'Icon Vertical Total' is defined by register #ADh.
'Character Code Size' is defined by register #88h<6>.
Please refer to OSD screen timing diagram (AL300-15) for additional reference.

ADh: OSD2 Icon Vertical Total (R/W) [ICONVTOTAL2]
ICONVTOTAL2 = actual vertical icon total – 1
IconVTotal2 <7:0> OSD2 vertical icon total (Unit: 1 icon)
Icon vertical total defines how many text rows shown on OSD screen.
Total character code bytes stored in OSD RAM:
'RAM Horizontal Stride' x 'Icon Vertical Total' (bytes)
Total character code bytes retrieved in OSD RAM:
'Icon Horizontal Total' x 'Icon Vertical Total' x 'Character code size' (bytes)
Where
'RAM Horizontal Stride' is defined by registers #8Ch and #A3h.
'Icon Horizontal Total' is defined by register #ACh.
'Icon Vertical Total' is defined by register #ADh.
'Character Code Size' is defined by register #88h<6>.
Please refer to OSD screen timing diagram (AL300-15) for additional reference.

AFh: OSD2 Font Line Size (R/W) [FONTLINESIZE2]
Fontlinesize2 <7:0> memory size of a line of font (Unit: 1 byte)
Please refer to OSD screen timing diagram (AL300-15) for additional reference.

On Screen Display Color Registers:

Color 0 is the color of pixel with value:
'0' when in 1-bit/2-color mode or
"00" when in 2-bit/4-color mode.
Color 1 is the color of pixel with value:
'1' when in 1-bit/ 2-color mode or
"01" when in 2-bit/4-color mode.
Color 2 is the color of pixel with value "10" when in 2-bit, 4-color mode.
Color 3 is the color of pixel with value "11" when in 2-bit, 4-color mode.

B0h: Color 0 Red (R/W) [COLOR0RED]
Color0Red <7:0> Color 0 Red Component

B1h: Color 0 Green (R/W) [COLOR0GREEN]
Color0Green <7:0> Color 0 Green Component

B2h: Color 0 Blue (R/W) [COLOR0BLUE]
Color0Blue <7:0> Color 0 Blue Component

B3h: Color 1 Red (R/W) [COLOR1RED]
Color1Red <7:0> Color 1 Red Component

B4h: Color 1 Green (R/W) [COLOR1GREEN]

Color1Green <7:0> Color 1 Green Component

B5h: Color 1 Blue (R/W) [COLOR1BLUE]
Color1Blue <7:0> Color 1 Blue Component

B6h: Color 2 Red (R/W) [COLOR2RED]
Color2Red <7:0> Color 2 Red Component

B7h: Color 2 Green (R/W) [COLOR2GREEN]
Color2Green <7:0> Color 2 Green Component

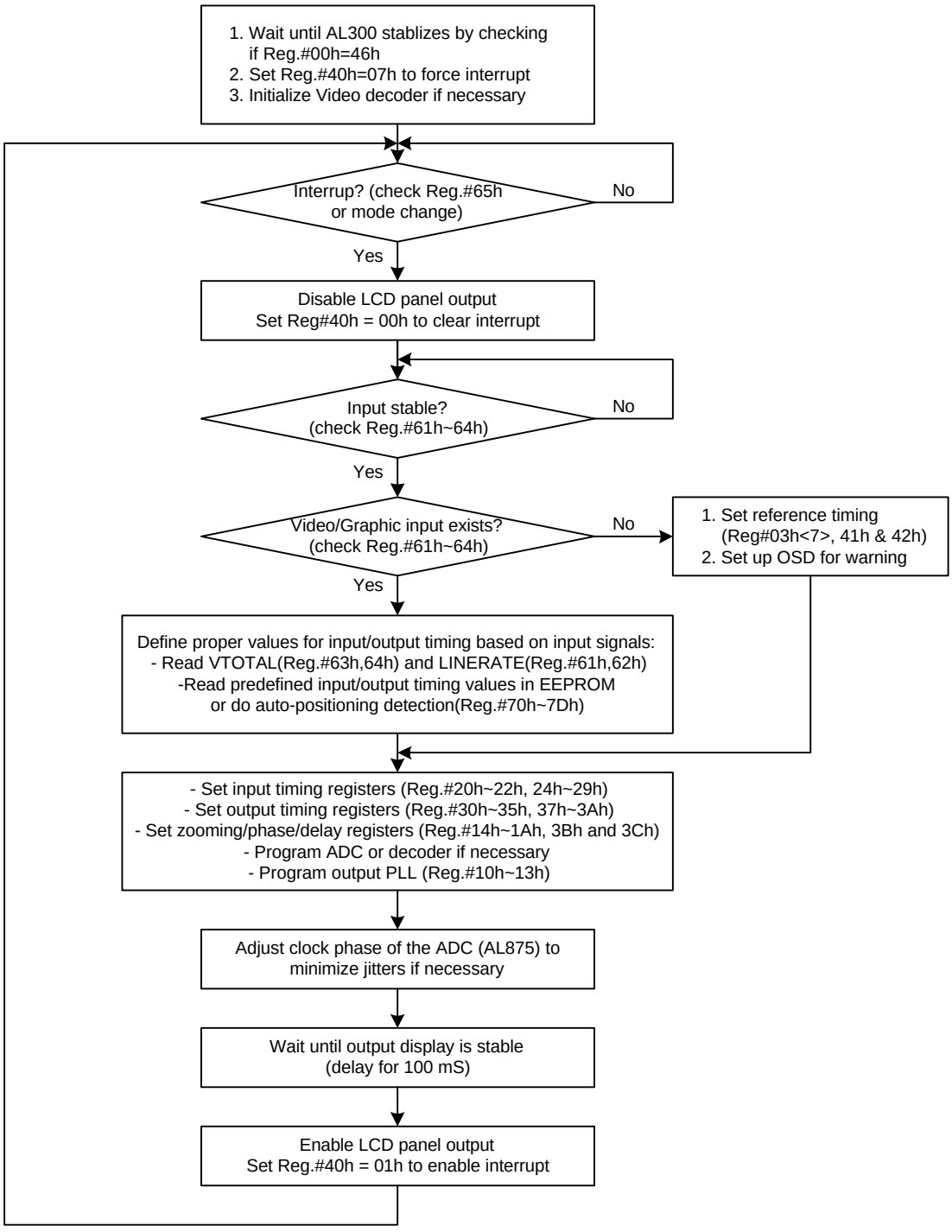
B8h: Color 2 Blue (R/W) [COLOR2BLUE]
Color2Blue <7:0> Color 2 Blue Component

B9h: Color 3 Red (R/W) [COLOR3RED]
Color3Red <7:0> Color 3 Red Component

BAh: Color 3 Green (R/W) [COLOR3GREEN]
Color3Green <7:0> Color 3 Green Component

BBh: Color 3 Blue (R/W) [COLOR3BLUE]
Color3Blue <7:0> Color 3 Blue Component

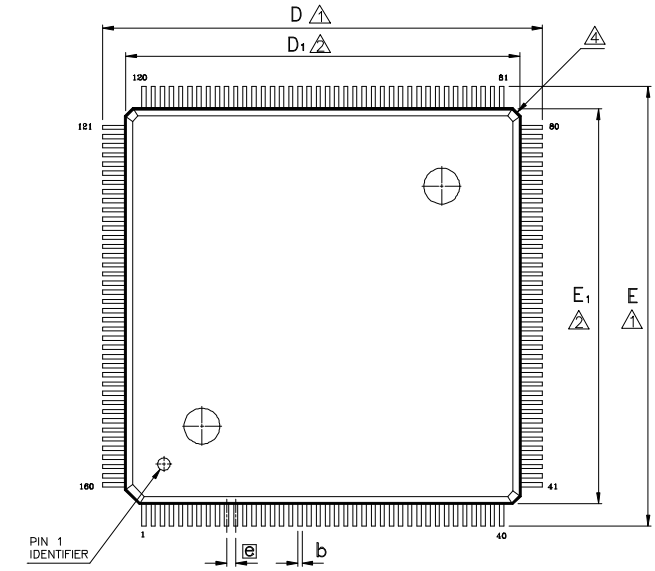
8.2 Programming Flowchart

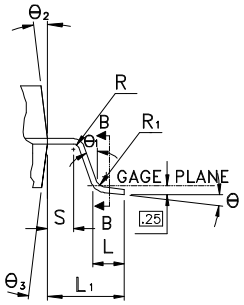


AL300-25 Programming Flowchart_ver B

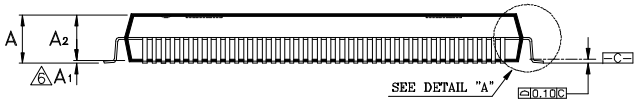
9.0 Mechanical Drawing

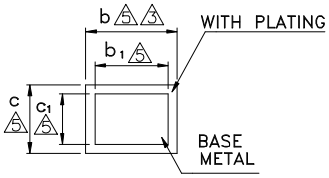
AL300: 28mm x 28mm 160-pin 0.65-pitch PQFP package





DETAIL "A"





SECTION B–B

Symbol	Dimension in mm			Dimension in inch		
	Min	Nom	Max	Min	Nom	Max
A	—	—	4.07	—	—	0.160
A1	0.25	—	—	0.010	—	—
A2	3.20	—	3.30	0.126	—	0.130
b	0.25	—	0.35	0.010	—	0.014
b1	0.30 BSC			0.012 BSC		
c	0.13	—	0.23	0.005	—	0.009
c1	0.15 BSC			0.006 BSC		
D	30.95	31.20	31.45	1.218	1.228	1.238
D1	27.90	28.00	28.10	1.098	1.102	1.106
E	30.95	31.20	31.45	1.218	1.228	1.238
E1	27.90	28.00	28.10	1.098	1.102	1.106
e	0.65 BSC			0.026 BSC		
L	0.73	—	1.03	0.029	—	0.040
L1	1.60 REF			0.063 REF		
R	0.15 REF			0.006 REF		
R1	0.15 REF			0.006 REF		
S	0.46 REF			0.018 REF		
θ	0°	—	7°	0°	—	7°
θ1	5° REF			5° REF		
θ2	15° REF			15° REF		
θ3	10° REF			10° REF		

NOTE :
1. TO BE DETERMINED AT SEATING PLANE C .
2. DIMENSIONS D1 AND E1 DO NOT INCLUDE MOLD PROTRUSION. D1 AND E1 ARE MAXIMUM PLASTIC BODY SIZE DIMENSIONS INCLUDING MOLD MISMATCH.
3. DIMENSION b DOES NOT INCLUDE DAMBAR PROTRUSION. DAMBAR CAN NOT BE LOCATED ON THE LOWER RADIUS OF THE FOOT.
4. EXACT SHAPE OF EACH CORNER IS OPTIONAL.
5. THESE DIMENSIONS APPLY TO THE FLAT SECTION OF THE LEAD BETWEEN 0.10 mm AND 0.25 mm FROM THE LEAD TIP.
6. A1 IS DEFINED AS THE DISTANCE FROM THE SEATING PLANE TO THE LOWEST POINT OF THE PACKAGE BODY.
7. CONTROLLING DIMENSION : MILLIMETER.
8. REFERENCE DOCUMENT : JEDEC MS–022 DD–1.

TITLE: 160LD QFP (28x28mm) PACKAGE OUTLINE –A42 L/F,FOOTPRINT 3.2mm			
L/F MATERIAL: ALLOY 42			
APPR.		DWG NO.	Q160–SW1
Q.M		REV NO.	B
PT		SCALE	
R&D		DATE	Feb. 16,'99
CHK.		DEN.	
AverLogic Technologies, Inc.			

10.0 Power Consumption

The AL300 works at single 3.3V power. The following table shows the current consumption of the AL300 (with output buffers to the panel) at different operating frequencies and supply voltages.

	Output frequency	Current consumption
AL300@3.3V	110MHz	135 mA (typ.)
AL300@3.3V	70MHz	95 mA (typ.)
AL300@3.3V	50MHz	75 mA (typ.)
AL300@3.3V	30MHz	55 mA (typ.)
Power down		5 mA (typ.)

The current consumption may be somewhat higher when the output of the AL300 drives a panel directly.

For more information about the AL300 or other AverLogic products, please contact your local authorized representatives, visit our website, or contact us directly.

CONTACT INFORMATION

AverLogic Technologies, Inc.
6840 Via Del Oro
Suite 160
San Jose, CA 95119
USA

Tel : 1 408 361-0400
Fax : 1 408 361-0404
E-mail : sales@averlogic.com
URL : www.averlogic.com