Digi®

NS9775 Print Engine Controller & JBIG Reference

Making DEVICE NETWORKING easy™

90000525_C

NS9775 Print Engine Controller & JBIG Reference

Part number/version: 90000525_C
Release date: March 2006
www.digi.com

Information in this document is subject to change without notice and does not represent a committment on the part of Digi International.

Digi provides this document "as is," without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of, fitness or merchantability for a particular purpose. Digi may make improvements and/or changes in this manual or in the product(s) and/or the program(s) described in this manual at any time.

This product could include technical inaccuracies or typographical errors. Changes are made periodically to the information herein; these changes may be incorporated in new editions of the publication.

Digi International
11001 Bren Road East
Minnetonka, MN 55343 U.S.A.
United States: +1 877 912-3444
Other locations: +1 952 912-3444

www.digi.com/support/
www.digi.com

Contents

Using This Guide

Review this section for basic information about the guide you are using, as well as general support and contact information.

About this guide

This guide provides information about the NS9775 print engine controller and JBIG decompression. The NS9775 is part of the Digi NET+ARM line of SoC (System-on-Chip) products, and supports high-bandwidth applications for intelligent networked devices.

The NET+ARM family is part of the NET+Works integrated product family, which includes the NET+OS network software suite.

Who should read this guide

This guide is for hardware developers, system software developers, and applications programmers who want to use the NS9775 for development.

To complete the tasks described in this guide, you must:

- Understand the basics of hardware and software design, operating systems, and microprocessor design.
- Understand the NS9775 architecture.

What’s in this guide

This table shows where you can find specific information in this guide:

To read about	See
NS9775 Print Engine controller interface	Chapter 1, "Printer Interface and JBIG"
JBIG Decoder module	Chapter 2, "JBIG Decoder"
Print Engine controller and JBIG registers	Chapter 3, "Registers"
JBIG DMA controller	Chapter 4, "JBIG DMA Controller"
Configuring the Print Engine controller	Chapter 5, "Print Engine Controller Configuration"

Conventions used in this guide

This table describes the typographic conventions used in this guide:

This convention	Is used for
italic type	Emphasis, new terms, variables, and document titles.
`monospaced type`	Filenames, pathnames, signal names, and code examples
_ (underscore)	Defines a signal as being active low
‘b	Indicates that the number following this indicator is in binary radix

Related documentation

The *NS9775 Hardware Reference* provides details about the NS9775, Digi's 32-bit microprocessor for color network printers and multifunction devices.

Review the documentation CD-ROM that came with your development kit for information on third-party products and other components.

Documentation updates

Digi occasionally provides documentation updates on the Web site.

Be aware that if you see differences between the documentation you received in your NET+Works package and the documentation on the Web site, the Web site content is the latest version.

Customer support

To get help with a question or technical problem with this product, or to make comments and recommendations about our products or documentation, use the contact information listed in this tables.

For	Contact information
Technical support	Telephone: 1 877 912-3444/ 1 952 912-3456 Fax: 1 952 912-4960
Documentation	techpubs@digi.com
Digi home page	www.digi.com
Online problem reporting	www.digi.com/support/eservice/eservicelogin.jsp

Printer Interface and JBIG

CHAPTER 1

The Print Engine Controller is an application-specific function designed for high-end monochrome laser printers and low-to-medium range color laser printers. The controller contains an embedded hardware JBIG decompressor, a precision clock generator, and an integrated multiprint-engine video signal interface.

Overview

The Print Engine Controller interfaces, gluelessly, with 4-pass color laser engines and tandem engines, and can interface with other major printer engines currently on the market.

Figure 1 shows the Print Engine Controller.

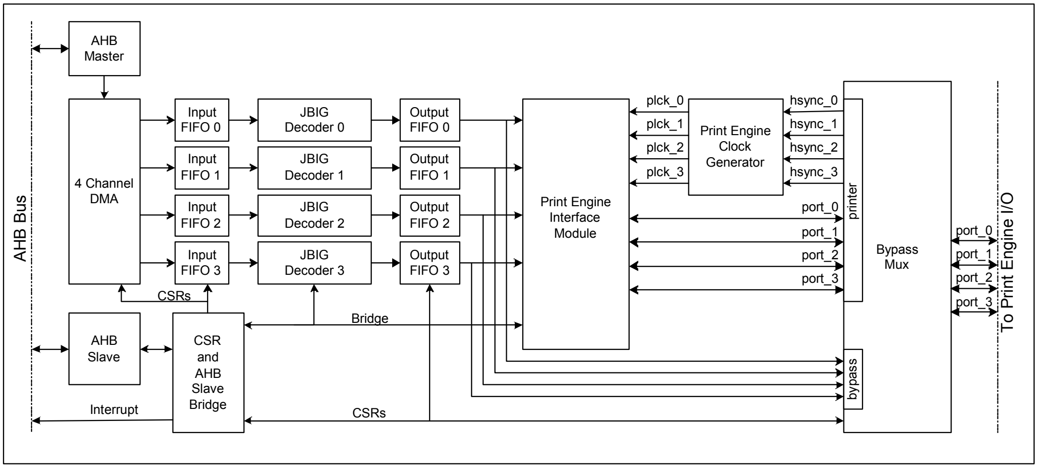


Figure 1: Print Engine Controller block diagram

- The JBIG decoder can be bypassed if the image to be printed is not compressed.
- There is a bypass mode to which decompressed data can be sent, bypassing the Print Engine Interface Module block.

JBIG requirements

To determine the worst case performance of the JBIG decompressor, use these formulas:

For a monochrome printer

```
[Resolution (1200x600) x Pagesize (8.5x11) x PPM] / 60 sec = JBIG performance
(pixels/sec)
```

For a 4-pass color printer

```
[Resolution (1200x600) x Pagesize (8.5x11) x PPM x 4] / 60 sec = JBIG
performance (pixels/sec)
```

For a tandem color printer

```
{[Resolution (1200x600) x Pagesize (8.5x11) x PPM x 4] / 60 sec} / # of JBIG
decoders = JBIG performance (pixels/sec)
```

Note: A 4-pass color printer uses only one of the JBIG decoders, as the four color bit-plane images must be sent sequentially.

Video clock generation

The video clock must take these critical issues into consideration:

- Print engine's horizontal and vertical correction factors.
- A very large jitter range exists on HSYNC signal.
- Pixel clock jitter cannot be more that 1/4 of a pixel clock; less than 1/8 of a pixel is preferred.

Horizontal and vertical correction factors

The video clock is used to clock pixel data to the print engine. During laser beam retraction and synchronization, however, the pixel data is stopped temporarily. To accomplish the performance goal, then, the actual video clock rate is much higher than the calculated average pixel rate based on the resolution and page-per-minute (PPM) requirement. Print engine manufacturers correct this problem by providing a *horizontal correction factor* (Hcorr) and *vertical correction factor* (Vcorr).

Examples

The appropriate video clock rate can be calculated for a monochrome/tandem printers and for 4-pass color printer using this equation:

```
VCLK rate (pixels/sec)=

{[Resolution (pixels/sq in) x passes x Page size (sq in/pg) x Page rate (pg/min)] x
(1/60) min/sec}
(horizontal correction x vertical correction)
```

With correction factors of:

Hcorr	0.6872
Vcorr	0.7353

video rates can be calculated as follows:

For a monochrome or tandem color engine at 30 ppm

```
VCLK rate = {[Resolution (1200x600) pixels/sq in x (1) x Pagesize (8.5x11)
sq in/pg x pagerate (30) pg/min] x (1/60) min/sec} / (0.6872 x 0.7353) =
66,614,135.31 Hz (66.6 MHz)
```

For a 4-pass color engine at 10 ppm

```
VCLK rate = {[Resolution (1200x600) pixels/sq in x (4) x Pagesize (8.5x11)
sq in/pg x PPM (10) pg/min] x (1/60) min/sec} / (0.6872 x 0.7353) =
88,818,847.074 Hz (88.8 MHz)
```

HSYNC jitter

The HSYNC signal from the print engine indicates the start of a horizontal line. Ideally, HSYNC can be used to synchronize with the video clock through a PLL. Due to the excessive jittering range of a typical HSYNC signal, however, the PLL tends to lose the lock and requires a resynchronization each time the HSYNC is presented. The VCLK generation circuit must generate a “super-stable” video clock independent of HSYNC, and use HSYNC as an enable.

Video clock jitter

Because of the asynchronous nature of the HSYNC signal and VCLK, the space between HSYNC and the first rising edge of the VCLK can be as long as a full VCLK. The pixels printed on two adjacent lines on the same paper can have up to one pixel length of jittering, which is an unacceptable value; the acceptable value is less than 1/4 pixel. To resolve this problem, the VCLK generation circuit must create a “super clock” that is 4X or 8X VCLK frequency. The super clock is enabled by HSYNC, and its first rising edge creates the first VCLK — with a VCLK jitter that is less than 1/4 (or 1/8) of a pixel.

Printer interface mode signals

Figure 2 details all signals on the printer interface. Table 1 explains the signals.

A *non-tandem engine* is a monochrome or 4-pass color engine. A *tandem engine* is a single-pass color engine.

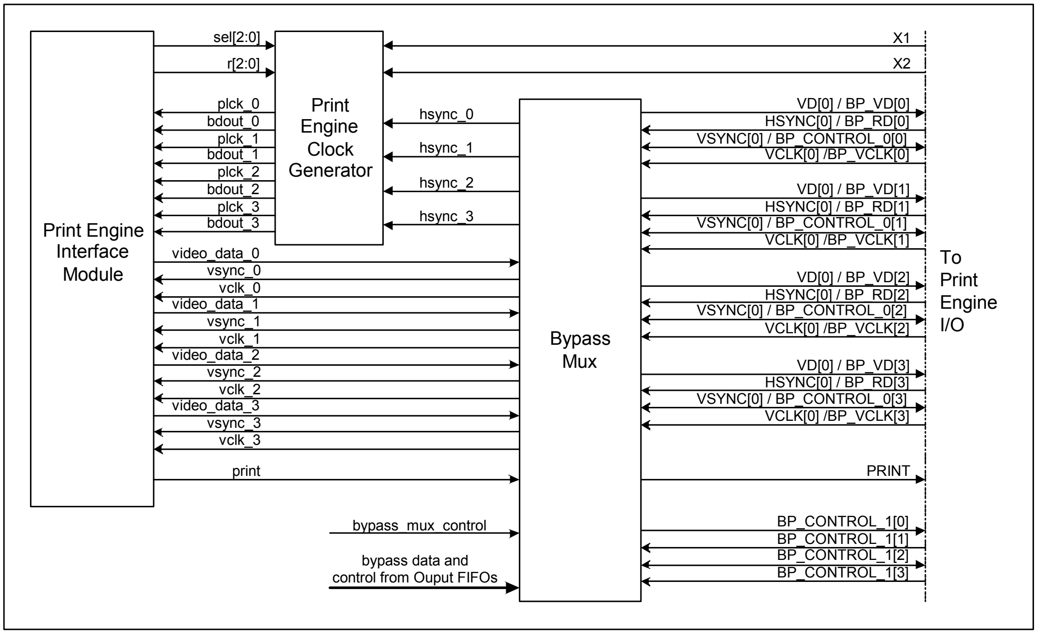


Figure 2: Detailed print engine inputs/outputs

Signal	Description
VD [3:0] Video data	Four output signals, with programmable polarity. These are the video data output signals that modulate the laser beam, or LED array, ON or OFF corresponding to the image to be printed. ■ Non-tandem engines. Only one video data out signal is used. ■ Tandem engines. All four video data out signals are used, and correspond to the Y, M, C, and K color planes.

Table 1: Print engine signals

Signal	Description
HSYNC [3:0] Horizontal synchronization	Four input signals, with programmable polarity. The active edges will always be the leading edges. An engine provides the horizontal scanning direction (main scanning direction) synchronization signals, which time and start each scan line illumination of the OPC drum for each color plane. ■ Non-tandem engines. Only one HSYNC signal is used. ■ Tandem engines. One of two situations: • All four HSYNC signals are used, and correspond to the Y. M. C, and K color planes. • The tandem engine provides only one common HSYNC signal, applicable to all four planes, and all four HSYNC signals must be connected to that engine's HSYNC signal.
VSYNC [3:0] Vertical synchronization	Four input signals, with programmable polarity. The active edges will always be the leading edges. An engine provides the vertical scanning direction (subscanning direction) synchronization signals, which time and start the scan-by-scan illumination of the OPC drum for each color plane. ■ Non-tandem engines. Only one VSYNC signal is used. ■ Tandem engines. All four VSYNC signals are used, and correspond to the Y, M, C, and K color planes. The VSYNC signals are used for flow control and are outputs in bypass mode.
VCLK [3:0] Video or pixel clock	Four input signals, with programmable active edges and synchronous to HSYNC signals. Synchronous engines provide these video clock signals for timing of video data. ■ Non-tandem engines. Only one VCLK signal is used. ■ Tandem engines. One of two situations: • All four signals are used, one for each color plane. • The tandem engine provides only one common video clock signal, and all VCLK signals must be connected to that engine's video clock signal.
PRINT Print signal	An output signal with programmable polarity. This signal directs the print engine to start or continue print operations. The signal typically is asserted after the frame FIFO(s) has been filled initially with the beginning data for the next page to be printed. After receiving this signal, the engine starts or continues the physical imaging process.
Crystal inputs	Used with asynchronous print engines. High precision external clock sources (that is, crystals) are required to generate accurate video clock.

Table 1: Print engine signals

Signal	Description
General communication signals	In addition to the video interface signals, print engines include communication lines for transmitting or receiving command and status signals. The status and command codes usually are exchanged over a serial port link (rather than I2C, for example)) between the engine and the controller. Other signals (such as printer ready, controller ready, or reset) can be exchanged over the general purpose I/O interface. Note, however, that these signals are provided using the GPIO and serial modules, and are not dedicated only to the video interface.

Table 1: Print engine signals

Bypass mode

The video interface includes a bypass mode, which transmits decompressed bit-plane data to an external programmable device for formatting and timing. Bypass mode is provided for operational requirements that are not available using the Print Engine Interface module.

Figure 3 shows a Bypass Mux module for one port.

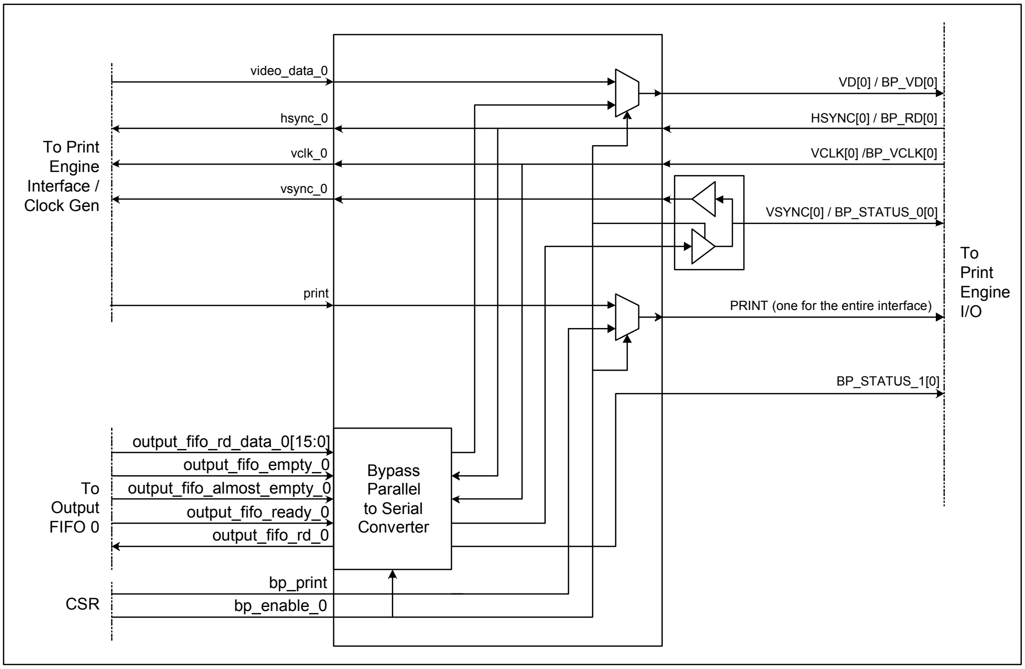


Figure 3: IBypass mode for one port

The Print Engine Interface is set to synchronous mode so the VCLK is passed directly to the output FIFO. The signals to the external Print Engine interface are described in Table 2. Figure 4 shows the timing of I/O signals in bypass mode.

Signal	I/O	Description
BP_VD[n]	Output	Serial image bit data.
BP_RD[n]	Input	Reads one bit of image data.
BP_VCLK[n]	Input	The data, control, and read signals are synchronous to this clock

Table 2: Bypass mode signal description

Signal	I/O	Description
BP_STATUS_1[n] BP_STATUS_0[n]	Output	Indicate the status of the serial image bit data as shown: BP_STATUS_1:0 00 FIFO empty 01 FIFO not empty; two or more bits ready 10 FIFO almost empty, only one bit remaining 11 Reserved
PRINT	Output	Control signal under software control.

Table 2: Bypass mode signal description

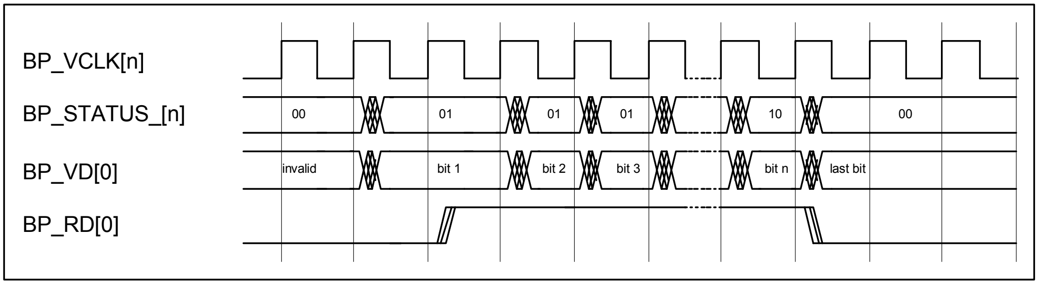


Figure 4: Bypass mode timing

The status signals allow the output circuit to read the bit stream in a burst mode. The status combination 10 indicates that the bit being read is the last bit, and the read signal should be deasserted.

Print Engine Interface module

The Print Engine Interface module allows the NS9775 ASIC to connect to different types of print engines (monochrome, 4-pass color, one pass tandem color) and to drive these engines with a minimum of external circuitry.

The Print Engine Interface accommodates both synchronous and asynchronous video operating modes.

- In synchronous mode, the print engine provides a clock and the interface provides the image bit stream using this clock. The maximum clock rate for synchronous mode is 200 MHz.
- In asynchronous mode, the NS9775 provides the clock used to transmit the image bit stream to the print engine. The NS9775 must synchronize the clock to the `HSYNC` input from the print engine. `HSYNC` to clock synchronization is explained in "Print engine clock generator," beginning on page 12. The maximum clock rate for asynchronous mode is 100 MHz.

Figure 5 shows the interface between the output FIFO and the Print Engine Interface module.

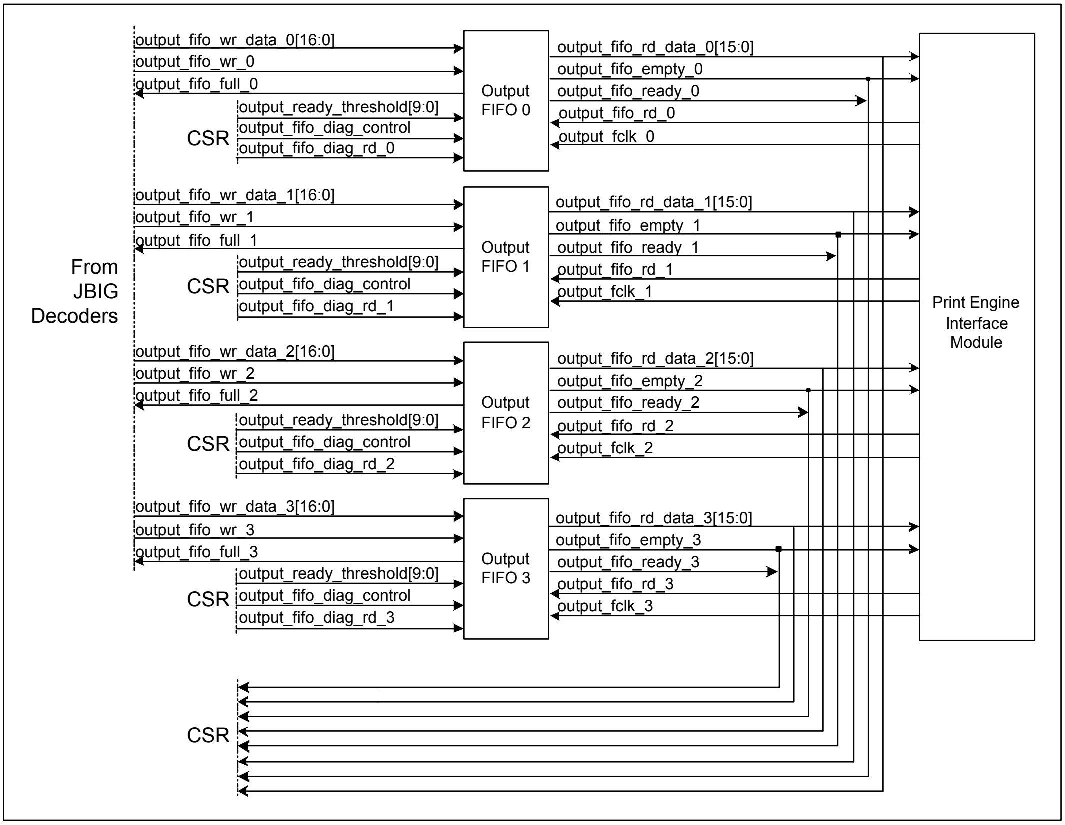


Figure 5: Output FIFO to Print Engine Interface

Output FIFO is sized at 32, 768 bits. Memory configuration is a 2048x16 bit single port RAM.

Figure 6 shows the timing between the output FIFOs and the Print Engine Interface.

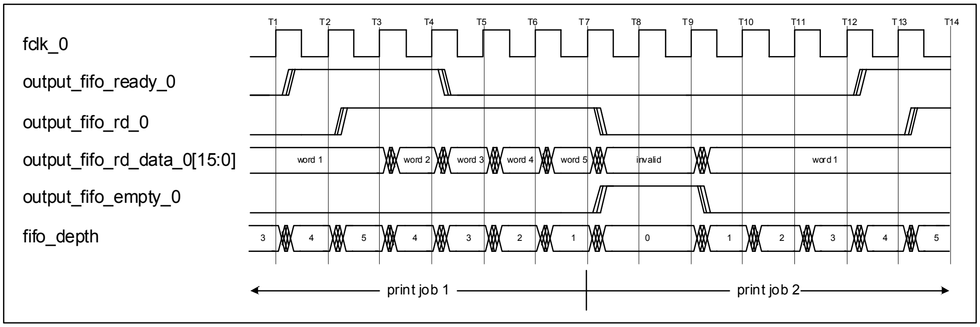


Figure 6: Output FIFO to Print Engine Interface timing

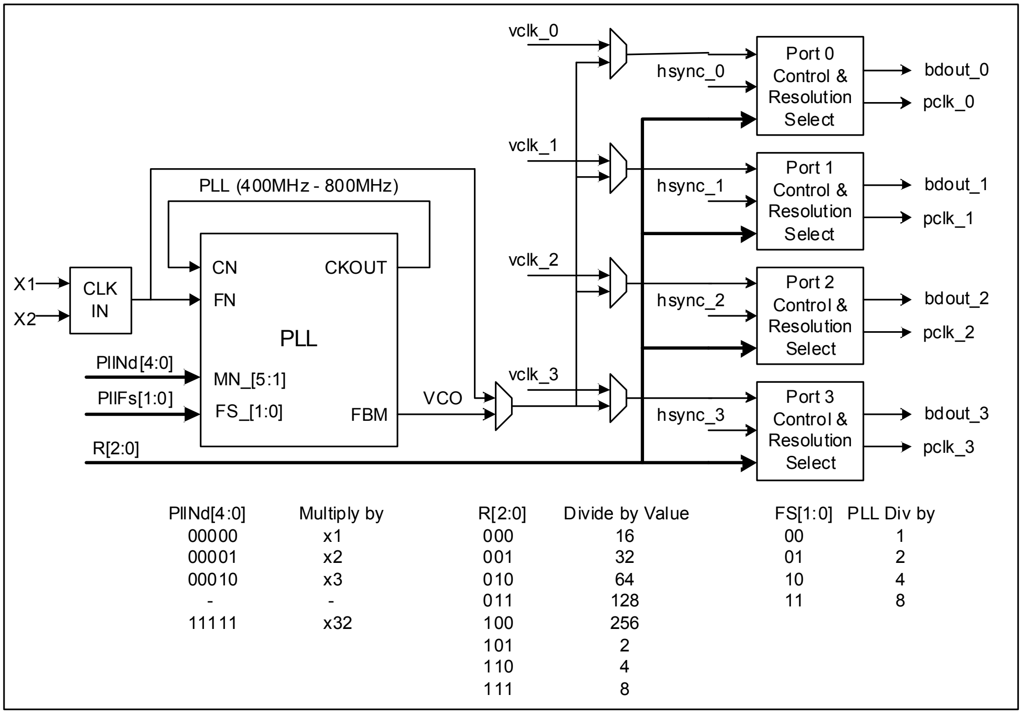
- The output FIFOs provide synchronization between the system clock used to perform writes and the video clock used to perform reads.
- The CSR inputs and outputs configure the output FIFOs, and allow the FIFOs to be read from when in diagnostic mode.
- The `output_ready_threshold` value control when the `output_fifo_ready` signals are asserted. At the beginning of a print job, the `ready` signal allows up to a line's worth of data to be decoded before the `ready` signal is asserted. The `ready` signal is asserted when the number of 16y-bit words in the FIFO reaches the threshold value.
- The `ready` signal generates an interrupt to the CPU, at which time the Print Engine Interface can be armed. The `ready` signal is deasserted when the number of 16-bit words in the FIFO drops below the threshold value.
- The threshold value is programmable.

Note: The FIFO must never become empty during a multi-page print job.

Print engine clock generator

The print engine clock generator circuit provides a pixel clock for four print engine interfaces.

Figure 7 shows the print engine clock generator. The control settings `PllND[4:0]`, `PllFs[1:0]`, and `R[2:0]` are set in the Video PLL Configuration register (see "Video PLL Configuration register," beginning on page 43).



PllNd[4:0]	Multiply by
00000	x1
00001	x2
00010	x3
-	-
11111	x32

R[2:0]	Divide by Value
000	16
001	32
010	64
011	128
100	256
101	2
110	4
111	8

FS[1:0]	PLL Div by
00	1
01	2
10	4
11	8

Figure 7: Print engine clock generator

The `hsync_0-3` signals are inputs from a print engine external to the NS9775. The active edge of this signal synchronizes `pclk_0-3` and `bdout_0-3`. The `bdout_0-3` signals are used as a start enable signal to determine when to start sending serial pixel data to each engine port. The rising edge of `pclk_0-3` actually transmits the serial pixel data.

Figure 8 shows how `pclk_0-3` and `bdout_0-3` are used.

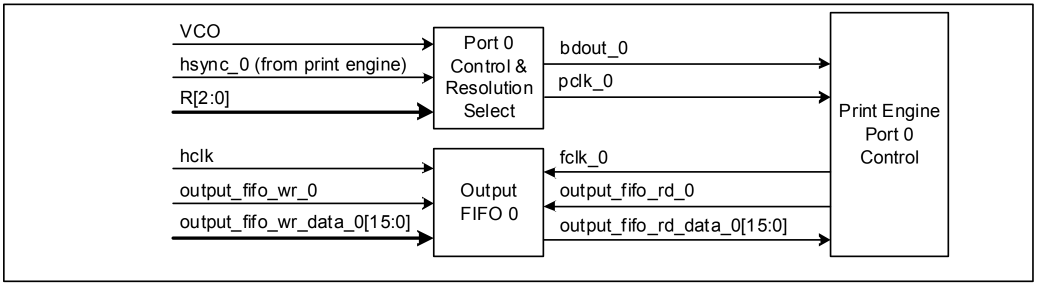

VCO
hsync_0 (from print engine)
R[2:0]
hclk
output_fifo_wr_0
output_fifo_wr_data_0[15:0]
Port 0 Control & Resolution Select
Output FIFO 0
bdout_0
pclk_0
fclk_0
output_fifo_rd_0
output_fifo_rd_data_0[15:0]
Print Engine Port 0 Control

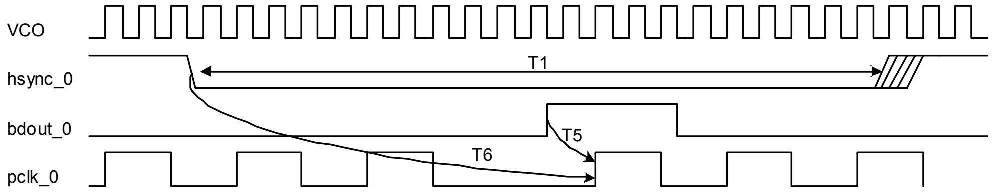
Figure 8: Print engine clock generator system example

Print engine timing

This section provides timing information for the print engine clock generator , the print engine controller, and the print engine clock.

Print engine clock generator

Figure 9 shows a basic timing diagram for the print engine clock generator; Table 3 provides the timing requirements.



	MIN	MAX
T1	22 Tvco	--
T6	19 Tvco + 0.97 Tpclk - 0.1ns	20 Tvco + 1.0 Tpclk + 0.1ns

T5 must meet technology dependent setup time to be sampled by pclk_0

Note: technology dependent minimum pulse widths must be met while synchronizing pclk_0 to hsync_0

Figure 9: Print engine clock generator timing diagram

Description	Minimum	Maximum
pclk_0-3 peak-to-peak jitter	0ps	200ps
Crystal input frequency	20 MHz	40 MHz
PLL frequency	400 MHz	800 MHz
VCO frequency	50 MHz	400 MHz
pclk_0-3 frequency	781 kHz (at 4x)	100 MHz

Table 3: Print engine clock generator requirements

Print engine controller

Figure 10 shows timing for the print engine controller; Table 4 provides the parameters.

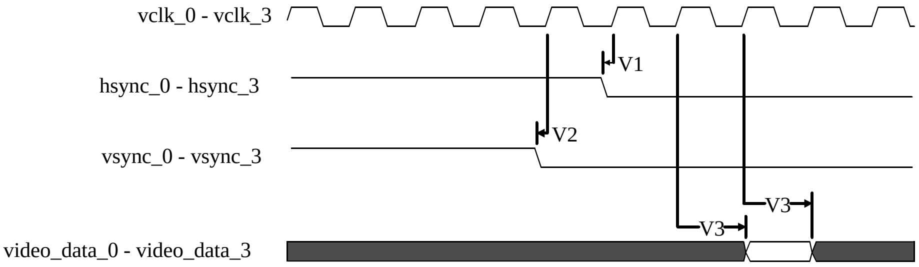


Figure 10: Print engine controller timing

Parm	Description	Min	Max	Unit	Note
V1	hsync_0 – hsync_3 (input) to vclk setup	1		ns	1
V2	vsync_0 – vsync_3 (input) to vclk setup	1		ns	1
V3	video_data_0 – video_data_3 (output) to vclk setup	2	6.5	ns	

Table 4: Print engine controller timing parameters

1 Hold time is 0.5ns.

Print engine clock

Figure 11 shows timing for the print engine clock; Table 5 provides the parameters.

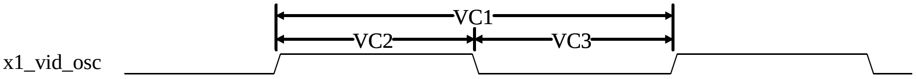


Figure 11: Print engine clock timing

Parm	Description	Min	Max	Unit	Notes
VC1	x1_vid_osc cycle time	2.5	10	ns	1
VC2	x1_vid_osc high time	(VC1/2) x 0.45	(VC1/2) x 0.55	ns	
VC3	x1_vid_osc low time	(VC1/2) x 0.45	(VC1/2) x 0.55	ns	

Table 5: Print engine clock timing parameters

1 The video PLL can be bypassed.

JBIG Decoder

CHAPTER 2

The JBIG Decoder module is a complete generic core for the loss-less decoding process of still images.

Overview

The JBIG Decoder module is instantiated four times in the NS9775, one instantiation for each bit plane in a four-color image. Compressed data is input to the JBIG Decoder through the input FIFO, and decompressed data is written to the output FIFO.

The JBIG Decoder requires two processing RAM memories per instantiation: a context DPRAM and a line-memory SRAM. The JBIG Decoder also has a host interface for accessing control and status registers.

The JBIG Decoder can be programmed to operate in one of two modes:

- **Manual mode.** The CPU must program all of the configuration registers in the JBIG Decoder using the host interface.
- **Automatic Header Processing mode.** The configuration registers are configured automatically from the contents of the 20-byte JBIG header.

See Chapter 3, "Registers," for a description of each JBIG Decoder register.

Figure 12 shows the JBIG Decoder interface to the NS9775, using JBIG Decoder 1 as an example. All instantiations of the JBIG Decoder (0 – 3) are identical.

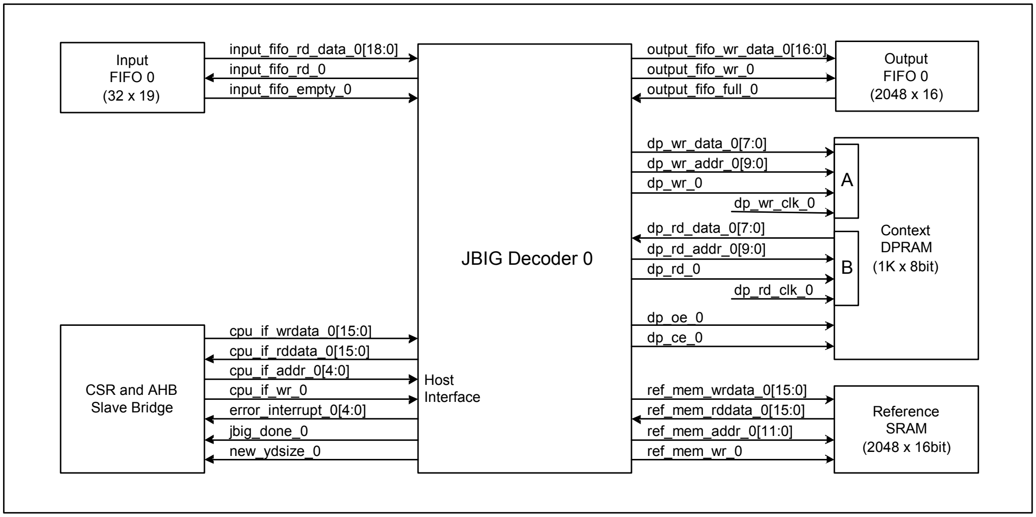


Figure 12: JBIG Decoder I/O signals

JBIG Decoder considerations

When using the JBIG Decoder, be aware of these facts:

- The JBIG compressed image must be a multiple of 16-bit words.
- The minimum bits per line is 64.
- For 2400 dpi images, 2-line template encoding must be used.

  For 1200 dpi images and below, 2-line or 3-line template encoding can be used. Note, however, that using 2-line template encoding rather than 3-line results in 5% reduction compression efficiency.
- The Mx field in the JBIG header to must be set to 0. Modify the JBIG driver on the host machine performing the JBIG compression to ensure that Mx is set to 0.

Input FIFO interface

The Input FIFO interface holds compressed or uncompressed image data that has been read from external memory through the AHB Master module and DMA. The FIFO is sized to 32 16-bit words. There are three tag bits in the Input FIFO interface — `input_fifo_rd_data[18:16]` — that identify the data in the FIFO, as shown:

Tag bits	Description
000	First 16-bit word of JBIG header (2 bytes)
001	Remaining 16-bit words of JBIG header (18 bytes)
010	First 16-bit word of compressed image data
011	Remaining 16-bit words of compressed image data
100	reserved
101	Reserved
110	First 16-bit word of uncompressed image data
111	Remaining 16-bit words of uncompressed image data

Table 6: Input FIFO tag bit description

When the `110` and `111` tag bits are found, the JBIG Decoder operates in uncompressed bypass mode and simply transfers the data from the input FIFO to the output FIFO.

Table 7 describes all signals in the input FIFO interface.

Signal name	Description
input_fifo_rd_data[18:16]	Tag bits (see Table 6: "Input FIFO tag bit description").
input_fifo_rd_data[15:00]	JBIG header, JBIG compressed image, or uncompressed image data, as indicated by the tag bits (`input_fifo_rd_data[18:16]`).
input_fifo_rd	The JBIG Decoder asserts a 1 to read one 16-bit word.
input_fifo_empty	0 The FIFO has one or more 16-bit words 1 The FIFO is empty

Table 7: Input FIFO signal bit definition

Output FIFO interface

The Output FIFO interface holds uncompressed image data that is ready to be sent to the Print Engine Interface module or the bypass mux. The FIFO is sized to hold one line of data with a resolution of 2400 dpi and a page width of 13 inches. This requires 21,200 bits of memory; a 2048 x 16-bit RAM is used for the FIFO memory.

There is one tag bit in the Output FIFO interface — `output_fifo_rd_data[16]`.

Table 8 describes all signals in the Output FIFO interface.

Signal name	Description
output_fifo_wr_data[16]	Tag bit 0 The first 16-bit word in the image 1 Remaining 16-bit words in the image
output_fifo_wr_data[15:0]	Uncompressed image data.
output_fifo_wr	The JBIG Decoder asserts a 1 to write one16-bit word.
output_fifo_full	0 The FIFO can accept at least one 16-bit word 1 The FIFO is full

Table 8: Output FIFO signal bit definition

Context DPRAM interface

The JBIG Decoder requires 1k bytes for context memory for decoding the JBIG image. The context memory is a DPRAM with one write port and one read port, both independent of each other.

Reference Memory interface

The JBIG Decoder requires a certain amount of line reference memory for decoding the JBIG image. The memory configuration is a single port SRAM. The size is determined by the maximum horizontal line size and the maximum resolution. The NS9775 supports 13-inch wide images at a resolution of 2400 dpi, for a total of 31,200 bits of memory. The NS9775 ASIC has a 2048 x 16-bit memory, for a total of 32,768 bits.

Figure 13 shows reference memory read and write timing.

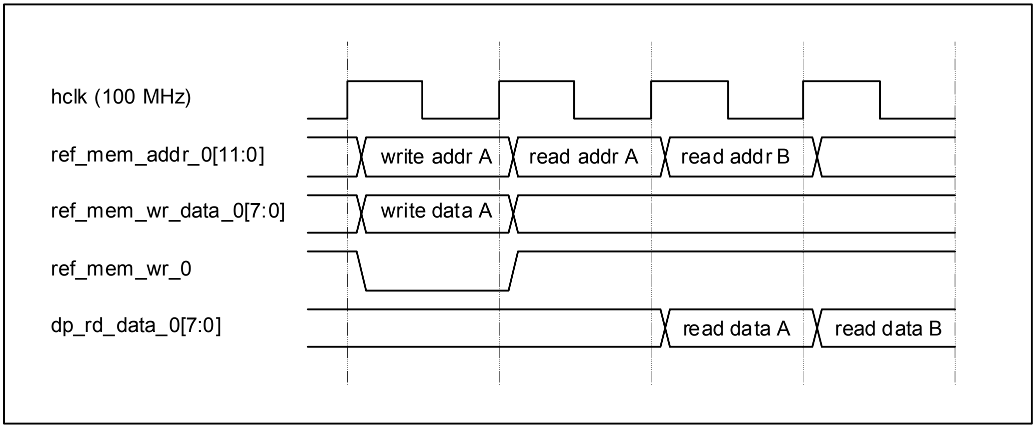


Figure 13: Reference SRAM timing

JBIG Decoder Host interface

The CPU has access to the control and status registers in the JBIG Decoder through the JBIG Decoder Host interface. In addition, the JBIG Decoder has several interrupt signals that are captured in the CSR and Slave AHB Bridge module. The bridge function converts the slave access into a JBIG Decoder Host interface access. The error and interrupt signals are captured and handled in the CSR and AHB Slave Bridge module.

See Chapter 3, "Registers," for register-specific information.

Registers

CHAPTER 3

The video control and status registers configure the “glue” logic for the DMA controller, input and output FIFOs, and bypass mux. The AHB slave function contains registers that allow access to the JBIG decoders and Print Engine Interface module. These registers are discussed in this chapter.

Address map

The 1 meg address space dedicated to the Print Engine Controller Interface module AHB slave is divided as shown:

0x000–0x0FF	Digi glue module CSRs
0x100–0x1FF	JBIG Decoder 3
0x200–0x2FF	JBIG Decoder 2
0x300–0x3FF	JBIG Decoder 1
0x400–0x4FF	JBIG Decoder 0
0x500–0x5FF	Print Engine Interface module

Table 9 lists the configuration and status registers for the print engine controller module, including JBIG decoder and Print Engine Interface registers.

Offset	[31:24]	[23:16]	[15:08]	[07:00]
0x000	GenConfig			
0x004	DMA Channel 3 Initial Buffer Descriptor Pointer			
0x008	DMA Channel 2 Initial Buffer Descriptor Pointer			
0x00C	DMA Channel 1 Initial Buffer Descriptor Pointer			
0x010	DMA Channel 0 Initial Buffer Descriptor Pointer			
0x014	DMA Channel 3 Current Buffer Descriptor Pointer			
0x018	DMA Channel 2 Current Buffer Descriptor Pointer			
0x01C	DMA Channel 1 Current Buffer Descriptor Pointer			
0x020	DMA Channel 0 Current Buffer Descriptor Pointer			
0x024	Interrupt Status register			
0x028	Interrupt Enable register			
0x02C	JBIG Decoder Error Interrupt Status register			
0x030	Output FIFO Ready Threshold			
0x034	Output FIFO 3–0 Status register			

Table 9: Video (JBIG) registers

Offset	[31:24]	[23:16]	[15:08]	[07:00]
0x038	Output FIFO 3 Diagnostic Read register			
0x03C	Output FIFO 2 Diagnostic Read register			
0x040	Output FIFO 1 Diagnostic Read register			
0x044	Output FIFO 0 Diagnostic Read register			
0x048	Video PLL Configuration register			
0x04C	Video Clock Fine Tune register			
0x050	Horizontal & Vertical Delay Channel 3			
0x054	Horizontal & Vertical Delay Channel 2			
0x058	Horizontal & Vertical Delay Channel 1			
0x05C	Horizontal & Vertical Delay Channel 0			
0x060	Output FIFO Ready Interrupt Control and Status			
0x064–0x0FF	Reserved			
0x100–0x1FF	JBIG Decoder Host Interface			
0x100–0x120	Reserved			
0x124	Lines per Stripe			
0x128–0x140	Reserved			
0x144	Pixels per Line in Image			
0x148	Total Lines in Image			
0x14C	JBIG Control register			
0x150	Reserved			
0x154	Auto Header Enable			
0x158–0x178	Reserved			
0x17C	Enable Status register			
0x200–0x2FF	JBIG Decoder 2 Host Interface			
0x200–0x220	Reserved			
0x224	Lines per Stripe			
0x228–0x240	Reserved			

Table 9: Video (JBIG) registers

Offset	[31:24]	[23:16]	[15:08]	[07:00]
0x244	Pixels per Line in Image			
0x248	Total Lines in Image			
0x24C	JBIG Control register			
0x250	Reserved			
0x254	Auto Header Enable			
0x258–0x278	Reserved			
0x27C	Enable Status register			
0x300–0x3FF	JBIG Decoder 1 Host Interface			
0x300–0x320	Reserved			
0x324	Lines per Stripe			
0x328–0x340	Reserved			
0x344	Pixels per Line in Image			
0x348	Total Lines in Image			
0x34C	JBIG Control register			
0x350	Reserved			
0x354	Auto Header Enable			
0x358–0x378	Reserved			
0x37C	Enable Status register			
0x400–0x4FF	JBIG Decoder 0 Host Interface			
0x400–0x420	Reserved			
0x424	Lines per Stripe			
0x428–0x440	Reserved			
0x444	Pixels per Line in Image			
0x448	Total Lines in Image			
0x44C	JBIG Control register			
0x450	Reserved			
0x454	Auto Header Enable			

Table 9: Video (JBIG) registers

Offset	[31:24]	[23:16]	[15:08]	[07:00]
0x458–0x478	Reserved			
0x47C	Enable Status register			
0x500–0x5FF	Print Engine Interface Module CPU Interface			
0x500	Video Control register			
0x510	Video Status register			
0x520	Video Vertical Margin and Data register			
0x530	Video Horizontal Margin and Data register			

Table 9: Video (JBIG) registers

CSR and AHB slave registers

Print Engine Controller General Configuration register

The Print Engine Controller General Configuration register contains miscellaneous control settings for the Print Engine Controller module.

Address: A050 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved		BPMS	BPMP	BPME	DMAEC	IFBS	OFDE	DCA3	DCA2	DCA1	DCA0	DCE3	DCE2	DCE1	DCE0

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ABS	Rsvd	DCR	PEIR	OFR3	OFR2	OFR1	OFR0	IFR3	IFR2	IFR1	IFR0	JDR3	JDR2	JDR1	JDR0

Register bit assignment

Bits	Access	Mnemonic	Reset	Description
D31:30	N/A	Reserved	N/A	N/A

Table 10: Print Engine Controller General Configuration register

Bits	Access	Mnemonic	Reset	Description
D29	R/W	BPMS	0x0	BypassShiftMode 0 Shift data out from the right (least significant byte (lsb) first) 1 Shift data out from the left (most significant byte (msb) first)
D28	R/W	BPMP	0x0	BypassModePrint Controls the print output signal when in bypass mode.
D27	R/W	BPME	0x0	BypassModeEnable 0 Normal Print Engine interface 1 Bypass mode enabled
D26	R/W	DMAEC	0x0	DmaErrClear 0 Normal operation 1 Clear DMA error condition
D25	R/W	IFBS	0x0	InputFifoByteSwap 0 Normal operation for JBIG images; swap the bytes in the 16-bit words 1 Do not swap bytes in the 16-bit words
D24	R/W	OFDE	0x0	OutputFifoDiagEn 0 Normal operation 1 The contents of the four Output FIFOs can be read using the OutputFifoRead registers (see page 41 through page 43).
D23	R/W	DCA3	0x0	DmaCh*N*Abort N = 3, 2, 1, or 0 0 Normal operation 1 Close current buffer descriptor, then stop operation. This bit must be cleared to start the next print job. The DmaCh*N*En bit must also be cleared during the same write, after the DmaCh*N*AIP interrupt indicating that the abort has finished.
D22	R/W	DCA2	0x0	
D21	R/W	DCA1	0x0	
D20	R/W	DCA0	0x0	

Table 10: Print Engine Controller General Configuration register

Bits	Access	Mnemonic	Reset	Description
D19	R/W	DCE3	0x0	DmaCh*N*En N = 3, 2, 1, or 0 0 Disable the DMA channel 1 Enable the DMA channel These bits must be set high by system software to start sending image data. After a single or multi-page print job has completed, the bit must be cleared before being set high for the next print job.
D18	R/W	DCE2	0x0	
D17	R/W	DCE1	0x0	
D16	R/W	DCE0	0x0	
D15	R/W	ABS	0x0	AhbBurstSize 0 Use burst INCR8 1 Use burst INCR16
D14	N/A	Reserved	N/A	N/A
D13	R/W	DCR	0x0	DmaControllerReset 0 Soft reset 1 Enable
D12	R/W	PEIR	0x0	PrintEngineReset 0 Soft reset 1 Enable
D11	R/W	OFR3	0x0	OutputFifo3Reset 0 Soft reset 1 Enable
D10	R/W	OFR2	0x0	OutputFifo2Reset 0 Soft reset 1 Enable
D09	R/W	OFR1	0x0	OutputFifo1Reset 0 Soft reset 1 Enable
D08	R/W	OFR0	0x0	OutputFifo0Reset 0 Soft reset 1 Enable
D07	R/W	IFR3	0x0	InputFifo3Reset 0 Soft reset 1 Enable

Table 10: Print Engine Controller General Configuration register

Bits	Access	Mnemonic	Reset	Description
D06	R/W	IFR2	0x0	InputFifo2Reset 0 Soft reset 1 Enable
D05	R/W	IFR1	0x0	InputFifo1Reset 0 Soft reset 1 Enable
D04	R/W	IFR0	0x0	InputFifo0Reset 0 Soft reset 1 Enable
D03	R/W	JDR3	0x0	JbigDecoder3Reset 0 Soft reset 1 Enable
D02	R/W	JDR2	0x0	JbigDecoder2Reset 0 Soft reset 1 Enable
D01	R/W	JDR1	0x0	JbigDecoder1Reset 0 Soft reset 1 Enable
D00	R/W	JDR0	0x0	JbigDecoder0Reset 0 Soft reset 1 Enable

Table 10: Print Engine Controller General Configuration register

DMA Channel 3–0 Initial Buffer Descriptor Pointer registers

Address: A050 0004 / 0008 / 000C / 0010

The DMA Channel *N* Initial Buffer Descriptor Pointer registers provide the address of the first buffer descriptor field for each DMA channel. The address is a 32-bit field.

The register shown is for DMA Channel 3 Initial Buffer Descriptor (DIBD3). The format for registers DIBD2, DIBD1, and DIBD0 is the same as for this register.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DIBD3															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DIBD3															

Register bit assignment

Bits	Access	Mnemonic	Reset	Description
D31:00	R/W	DIBD3	0x0	DmaCH*N*InitBdPtr The first buffer descriptor in the ring. Used when the `Wrap` bit is found, indicating the last buffer descriptor in the list.
D31:00	R/W	DIBD2	0x0	
D31:00	R/W	DIBD1	0x0	
D31:00	R/W	DIBD0	0x0	

Table 11: DMA Channel 3–0 Initial Buffer Descriptor Pointer register

DMA Channel 3–0 Current Buffer Descriptor Pointer register

Address: A050 0014 / 0018 / 001C / 0020

The DMA Channel *N* Current Buffer Descriptor Pointer registers provide the address of the first buffer descriptor in the next print job to be started.

The register shown is for DMA Channel 3 Current Buffer Descriptor (DCBD3). The format for registers DCBD2, DCBD1, and DCBD0 is the same as for this register.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DCBD3															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DCBD3															

Register bit assignment

Bits	Access	Mnemonic	Reset	Description
D31:00	R/W	DCBD3	0x0	DmaCh*N*CurrBdPtr The current buffer descriptor pointer. After system software has set up the buffer descriptors and buffers for a print job, it writes the address of the first buffer descriptor in the list here.
D31:00	R/W	DCBD2	0x0	
D31:00	R/W	DCBD1	0x0	
D31:00	R/W	DCBD0	0x0	

Table 12: DMA Channel 3–0 Current Buffer Descriptor Pointer register

Interrupt Status register

Address: A050 0024

The Interrupt Status register allows system software to determine the source of the interrupt from the Video Interface module.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OFIUP	OFRIP	AERR IP	PEIIP	NLIP3	NLIP2	NLIP1	NLIP0	ERR IP3	ERR IP2	ERR IP1	ERR IP0	NCIP3	NCIP2	NCIP1	NCIP0

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AIP3	AIP2	AIP1	AIP0	BN RIP3	BN RIP2	BN RIP1	BN RIP0	LIP3	LIP2	LIP1	LIP0	EIP3	EIP2	EIP1	EIP0

Register bit definition

Bits	Access	Mnemonic	Reset	Description
D31	R/W	OFUIP	0x0	OutputFifoUnderflowIP The output FIFO underflow IP is active. The Output FIFO modules must be reset if this interrupt occurs.
D30	R/W	OFRIP	0x0	OutputFIFOReadyIP The output FIFO ready interrupt is active. The Print Engine Interface module can be enabled after this interrupt is received.

Table 13: Interrupt Status register

Bits	Access	Mnemonic	Reset	Description
D29	R/W	AERRIP	0x0	AHB error An AHB error was found by one of the DMA controllers. The DMA channels stop until the DmaErrClr (see page 28) bit is toggled.
D28	R/W	PEIIP	0x0	Print Engine Interface IP The Print Engine Interface module has an interrupt pending. Read the Video Status register (see page 60) to determine the source of the interrupt.
D27	R/W	NLIP3	0x0	JbigDecoder*N*NLIP N = 3, 2, 1, or 0 JBIG decoder NEWLEN marker received. **Note:** Applies only to fax applications. This interrupt should be disabled for printer applications.
D26	R/W	NLIP2	0x0	
D25	R/W	NLIP1	0x0	
D24	R/W	NLIP0	0x0	
D23	R/W	ERRIP3	0x0	JbigDecoderNERRIP N = 3, 2, 1, or 0 JBIG decoder error interrupt. Indicates the JBIG decoder stopped due to an error condition. Read the JBIG Decoder Error Interrupt Status register (see page 37) to determine the source of the interrupt.
D22	R/W	ERRIP2	0x0	
D21	R/W	ERRIP1	0x0	
D20	R/W	ERRIP0	0x0	
D19	R/W	NCIP3	0x0	JbigDecoder*N*NCIP N = 3, 2, 1, or 0 JBIG decoder normal completion interrupt. The done signal is asserted, indicating a complete JBIG plane has been decoded.
D18	R/W	NCIP2	0x0	
D17	R/W	NCIP1	0x0	
D16	R/W	NCIP0	0x0	
D15	R/W	AIP3	0x0	DmaCh*N*AIP N = 3, 2, 1, or 0 The DMA controller stopped because the DMA channel abort bit is set for a particular channel.
D14	R/W	AIP2	0x0	
D13	R/W	AIP1	0x0	
D12	R/W	AIP0	0x0	

Table 13: Interrupt Status register

Bits	Access	Mnemonic	Reset	Description
D11	R/W	BNRIP3	0x0	DmaCH*N*BNRIP A buffer descriptor with the F bit not set was read. When this happens, the DMA channel stops until the DmaErrClr bit (see page 28) is toggled.
D10	R/W	BNRIP2	0x0	
D09	R/W	BNRIP1	0x0	
D08	R/W	BNRIP0	0x0	
D07	R/W	LIP3	0x0	DmaCh*N*Last The last buffer descriptor in an image has been read (L bit set).
D06	R/W	LIP2	0x0	
D05	R/W	LIP1	0x0	
D04	R/W	LIP0	0x0	
D03	R/W	EIP3	0x0	DmaCh*N*End The last buffer descriptor in a print job has been read (E bit set).
D02	R/W	EIP2	0x0	
D01	R/W	EIP1	0x0	
D00	R/W	EIP0	0x0	

Table 13: Interrupt Status register

Interrupt Enable register

Address: A050 0028

The Interrupt Enable register enables interrupt sources to generate an interrupt to the CPU.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OFUE	OFRIE	AERR IE	PEIIE	NLIE3	NLIE2	NLIE1	NLIE0	ERR IE3	ERR IE2	ERR IE1	ERR IE0	NCIE3	NCIE2	NCIE1	NCIE0

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AIE3	AIE2	AIE1	AIE0	BN RIE3	BN RIE2	BN RIE1	BN RIE0	LIE3	LIE2	LIE1	LIE0	EIE3	EIE2	EIE1	EIE0

Register bit assignment

Bits	Access	Mnemonic	Reset	Description
D31	R/W	OFUE	0x0	OutputFifoUnderflowEnable 0 Interrupt source is disabled 1 Interrupt source is enabled. The interrupt source is visible in the Interrupt Status register but does not cause the interrupt to be asserted to the CPU.
D30	R/W	OFRIE	0x0	OutputFifoReadyEnable 0 Interrupt source is disabled 1 Interrupt source is enabled. The interrupt source is visible in the Interrupt Status register but does not cause the interrupt to be asserted to the CPU.
D29	R/W	AERRIE	0x0	AHBErrorIntEnable 0 Interrupt source is disabled 1 Interrupt source is enabled. The interrupt source is visible in the Interrupt Status register but does not cause the interrupt to be asserted to the CPU.
D28	R/W	PEIIE	0x0	Print Engine Interface IP enable 0 Interrupt source is disabled 1 Interrupt source is enabled. The interrupt source is visible in the Interrupt Status register but does not cause the interrupt to be asserted to the CPU.
D27	R/W	NLIE3	0x0	JbigDecoder*N*NLIPEnable N = 3, 2, 1, or 0; JBIG decoder NEWLEN marker. **Note:** Applies only to fax applications. This interrupt should be disabled for printer applications.
D26	R/W	NLIE2	0x0	
D25	R/W	NLIE1	0x0	
D24	R/W	NLIE0	0x0	

Table 14: Interrupt Enable register

Bits	Access	Mnemonic	Reset	Description
D23	R/W	ERRIE3	0x0	JbigDecoderNERRIPEnable N = 3, 2, 1, or 0; JBIG decoder error interrupt. 0 Interrupt source is disabled 1 Interrupt source is enabled. The interrupt source is visible in the Interrupt Status register but does not cause the interrupt to be asserted to the CPU.
D22	R/W	ERRIE2	0x0	
D21	R/W	ERRIE1	0x0	
D20	R/W	ERRIE0	0x0	
D19	R/W	NCIE3	0x0	JbigDecoder*N*NCIPEnable N = 3, 2, 1, or 0; JBIG decoder normal completion interrupt. 0 Interrupt source is disabled 1 Interrupt source is enabled. The interrupt source is visible in the Interrupt Status register but does not cause the interrupt to be asserted to the CPU.
D18	R/W	NCIE2	0x0	
D17	R/W	NCIE1	0x0	
D16	R/W	NCIE0	0x0	
D15	R/W	AIE3	0x0	DmaCh*N*AIPEnable N = 3, 2, 1, or 0 0 Interrupt source is disabled 1 Interrupt source is enabled. The interrupt source is visible in the Interrupt Status register but does not cause the interrupt to be asserted to the CPU.
D14	R/W	AIE2	0x0	
D13	R/W	AIE1	0x0	
D12	R/W	AIE0	0x0	
D11	R/W	BNRIE3	0x0	DmaCh*N*BNRIPEnable N = 3, 2, 1, or 0 0 Interrupt source is disabled 1 Interrupt source is enabled. The interrupt source is visible in the Interrupt Status register but does not cause the interrupt to be asserted to the CPU.
D10	R/W	BNRIE2	0x0	
D09	R/W	BNRIE1	0x0	
D08	R/W	BNRIE0	0x0	
D07	R/W	LIE3	0x0	DmaCh*N*LastEnable N = 3, 2, 1, or 0; last buffer descriptor. 0 Interrupt source is disabled 1 Interrupt source is enabled. The interrupt source is visible in the Interrupt Status register but does not cause the interrupt to be asserted to the CPU.
D06	R/W	LIE2	0x0	
D05	R/W	LIE1	0x0	
D04	R/W	LIE0	0x0	

Table 14: Interrupt Enable register

Bits	Access	Mnemonic	Reset	Description
D03	R/W	EIE3	0x0	DmaCh*N*EndEnable N = 3, 2, 1, or 0; end buffer descriptor. 0 Interrupt source is disabled 1 Interrupt source is enabled. The interrupt source is visible in the Interrupt Status register but does not cause the interrupt to be asserted to the CPU.
D02	R/W	EIE2	0x0	
D01	R/W	EIE1	0x0	
D00	R/W	EIE0	0x0	

Table 14: Interrupt Enable register

JBIG Decoder Error Interrupt Status register

Address: A050 002C

The JBIG Decoder Error Interrupt Status register allows system software to determine the source of the error interrupts from the JBIG Decoder modules.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved												DERR3			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DERR 3	DERR2					DERR1					DERR0				

Register bit assignment

Bits	Access	Mnemonic	Reset	Description
D31:20	N/A	Reserved	N/A	N/A

Table 15: JBIG Decoder Error Interrupt Status register

Bits	Access	Mnemonic	Reset	Description
D19:15	RR	DERR3	0x0	JbigDecoder*N*Error N = 3, 2, 1, or 0 00000 No error xx001 Invalid escape code found xx010 Non-zero AT marker at start of image xx011 ATPIXEL in X direction out of range (>127) xx100 ATPIXEL movement in Y direction xx101 NEWLEN marker out of range (>64K) xx110 Comment LC out of range (>64K) xx111 Missing end of stripe x1xxx JBIG header detected when the decoder was expecting JBIG compressed data for the current image 1xxxx Reserved
D14:10	RR	DERR2	0x0	
D09:05	RR	DERR1	0x0	
D04:00	RR	DERR0	0x0	

Table 15: JBIG Decoder Error Interrupt Status register

Output FIFO Ready Threshold register

Address: A050 0030

The Output FIFO ready Threshold register determines the threshold for the `output_fifo_ready` signals to be asserted. This register is used for all output FIFOs.

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr>
<tr><td colspan="16">Reserved</td></tr>
</table>

<table>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="5">Reserved</td><td colspan="11">OFRT</td></tr>
</table>

Register bit assignment

Bits	Access	Mnemonic	Reset	Description
D31:11	R	Reserved	0x0	N/A
D10:00	R/W	OFRT	0x0	OutputFifoReadyThreshold The `output_fifo_ready` signals are asserted when the FIFO depth is greater than this programmable value.

Table 16: Output FIFO Ready Threshold register

Output FIFO [3–0] Status register

Address: A050 0034

The Output FIFO [3-0] Status register can be read to verify the status of the output FIFOs. This register should be read before reading the output FIFOs in diagnostic mode.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFF3	OFR3	Rsvd	OFE3	OFF2	OFR2	Rsvd	OFE2	OFF1	OFR1	Rsvd	OFE1	OFF0	OFR0	Rsvd	OFE0

Register bit assignment

Bits	Access	Mnemonic	Reset	Description
D31:16	NA	Reserved	N/A	N/A
D15	R	OFF3	0x0	OutputFifo3Full 1 FIFO full
D14	R	OFR3	0x0	OutputFifo3Ready 1 FIFO ready
D13	N/A	Reserved	N/A	N/A
D12	R	OFE3	0x1	OutputFifo3Empty 1 FIFO empty

Table 17: Output FIFO Status register

Bits	Access	Mnemonic	Reset	Description
D11	R	OFF2	0x0	OutputFifo2Full 1 FIFO full
D10	R	OFR2	0x0	OutputFifo2Ready 1 FIFO ready
D09	N/A	Reserved	N/A	N/A
D08	R	OFE2	0x1	OutputFifo2Empty 1 FIFO empty
D07	R	OFF1	0x0	OutputFifo1Full 1 FIFO full
D06	R	OFR1	0x0	OutputFifo1Ready 1 FIFO ready
D05	N/A	Reserved	N/A	N/A
D04	R	OFE1	0x1	OutputFifo1Empty 1 FIFO empty
D03	R	OFF0	0x0	OutputFifo0Full 1 FIFO full
D02	R	OFR0	0x0	OutputFifo0Ready 1 FIFO ready
D01	N/A	Reserved	N/A	N/A
D00	R	OFE0	0x1	OutputFifo0Empty 1 FIFO empty

Table 17: Output FIFO Status register

Output FIFO 3 Diagnostic Read register

Address: A050 0038

When the output FIFOs are set to diagnostic mode (`OutputFifoDiagEn = 1`), Output FIFO 3 can be read from this register.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFRD3															

Register bit assignment

Bits	Access	Mnemonic	Reset	Description
D31:16	R	Reserved	0x0	N/A
D15:00	R	OFRD3	0x0	OutputFifo3ReadData Read the Output FIFO Status register before reading this register, to find out if data is available.

Table 18: Output FIFO 3 Diagnostic Read register

Output FIFO 2 Diagnostic Read register

Address: A050 003C

When the output FIFOs are set to diagnostic mode (`OutputFifoDiagEn = 1`), Output FIFO 2 can be read from this register.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFRD2															

Register bit assignment

Bits	Access	Mnemonic	Reset	Description
D31:16	R	Reserved	0x0	N/A
D15:00	R	OFRD2	0x0	OutputFifo2ReadData Read the Output FIFO Status register before reading this register, to find out if data is available.

Table 19: Output FIFO 2 Diagnostic Read register

Output FIFO 1 Diagnostic Read register

Address: A050 0040

When the output FIFOs are set to diagnostic mode (`OutputFifoDiagEn = 1`), Output FIFO 1 can be read from this register.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFRD1															

Register bit assignment

Bits	Access	Mnemonic	Reset	Description
D31:16	R	Reserved	0x0	N/A
D15:00	R	OFRD1	0x0	OutputFifo1ReadData Read the Output FIFO Status register before reading this register, to find out if data is available.

Table 20: Output FIFO 1 Diagnostic Read register

Output FIFO 0 Diagnostic Read register

Address: A050 0044

When the output FIFOs are set to diagnostic mode (`OutputFifoDiagEn = 1`), Output FIFO 0 can be read from this register.

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr>
<tr><td colspan="16">Reserved</td></tr>
</table>

<table>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="16">OFRD0</td></tr>
</table>

Register bit assignment

Bits	Access	Mnemonic	Reset	Description
D31:16	R	Reserved	0x0	N/A
D15:00	R	OFRD0	0x0	OutputFifo0ReadData Read the Output FIFO Status register before reading this register, to find out if data is available.

Table 21: Output FIFO 0 Diagnostic Read register

Video PLL Configuration register

Address: A050 0048

The Video PLL Configuration register configures the video PLL and clock generation circuit.

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr>
<tr><td>VLPLL</td><td colspan="14">Reserved</td><td>VHM</td></tr>
</table>

<table>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>OXM</td><td>PLL BM</td><td>UPLLS</td><td colspan="3">DVR</td><td>PLL BP</td><td colspan="2">PLLFS</td><td colspan="2">Reserved</td><td colspan="5">PLLND</td></tr>
</table>

Register bit assignment

Bits	Access	Mnemonic	Reset	Description
D31	R	VPLLL	0x0	VideoPllLocked You must monitor this register after the video PLL settings have been changed. This bit must be read as a 1 before printing operation can begin.
D30:17	R	Reserved	0x0	N/A
D16	R/W	VHM	0x1	VsyncHsyncMode 0 Do not synchronize the VSYNC signal to the HSYNC signal 1 Synchronize the VSYNC signal to HSYNC. This is necessary for applications that use one VSYNC/HSYNC pair for all four planes, and VSYNC and HSYNC are asynchronous. **Note:** This bit must be programmed after the Horizontal Sync Polarity and Vertical Sync Polarity bits have been programmed in the Video Control register (see "Video Control register" on page 57). When recovering from an abort due to a paper jam or other printer problem, VsyncHsyncMode must be set before the module resets occur.
D15	R/W	OXM	0x0	One X mode 0 Use the PLL divider as programmed in the DividerValue field (D12:10) 1 Set the divide value to 1
D14	R/W	PLLBM	0x0	PllBypassMultisource 0 Use the source as selected by PllBypass (D09) 1 Bypass the PLL and use the four external clocks (vclk_0-vclk_3) to drive the clock generators
D13	R/W	UPLLS	0x0	Update PLL settings Write a 1 to this bit to cause the PLL settings to be written to the PLL. This bit must be cleared before the next PLL change.

Table 22: Video PLL Configuration register

Bits	Access	Mnemonic	Reset	Description
D12:10	R/W	DVR	0x6	DividerValueR Controls the R divide value in the video clock generation circuit, as shown: 000 Divide by 16 001 Divide by 32 010 Divide by 64 011 Divide by 128 100 Divide by 256 101 Divide by 2 110 Divide by 4 111 Divide by 8
D09	R/W	PLLBP	0x0	PLL bypass 0 Use the PLL to generate the VCO clock 1 Bypass the PLL and use the input reference clock as the VCO clock
D08:07	R/W	PLLFS	0x0	Frequency select — PLL output divider 00 Divide by 1 01 Divide by 2 10 Divide by 4 11 Divide by 8
D06:05	R	Reserved	0x0	N/A
D04:00	R/W	PLLND	0x1A	PLL multiplier setting The multiplier is PllNd + 1.

Table 22: Video PLL Configuration register

Video Clock Fine Tune register

Address: A050 004C

The Video Clock Fine Tune register finetunes the video clocks generated for each color plane. The register can be used to independently delay the starting edge of each video clock from the falling edge of `hsync_n`. The delay unit is one VCO clock cycle.

Note: If the VsyncHsyncMode bit is in the Video PLL Configuration register, the minimum value must be 8 times the DividerValueR decoded value. For example, if DividerValueR is `3'b111`, the divider value is 8. The minimum value for VideoClockDelay is 8x8 = 64.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
VCD3								VCD2							

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VCD1								VCD0							

Register bit assignment

Bits	Access	Mnemonic	Reset	Description
D31:24	R/W	VCD3	0xB	VideoClockDelay3 Video clock delay, channel 3
D23:16	R/W	VCD2	0xB	VideoClockDelay2 Video clock delay, channel 2
D15:08	R/W	VCD1	0xB	VideoClockDelay1 Video clock delay, channel 1
D07:00	R/W	VCD0	0xB	VideoClockDelay0 Video clock delay, channel 0

Table 23: Video Clock Fine Tune register

Video HSYNC VSYNC Delay Channel 3

Address: A050 0050

The Video HSYNC VSYNC Delay Channel 3 register delays the HSYNC and VSYNC signals for channel 3:

- The delay unit for HSYNC is one VCO clock cycle.
- The delay unit for VSYNC is the HSYNC signal.

Notes:

- A value of 0 disables this feature.
- If this feature is used on a tandem engine, the minimum value for HSYNC and VSYNC is 1 for all channels.

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr>
<tr><td colspan="16">VOC3</td></tr>
</table>

<table>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="16">HOC3</td></tr>
</table>

Register bit assignment

Bits	Access	Mnemonic	Reset	Description
D31:16	R/W	VOC3	0x0	VSYNC offset channel 3
D15:00	R/W	HOC3	0x0	HSYNC offset channel 3

Table 24: Video HSYNC VSYNC Delay Channel 3 register

Video HSYNC VSYNC Delay Channel 2

Address: A050 0054

The Video HSYNC VSYNC Delay Channel 2 register delays the HSYNC and VSYNC signals for channel 2:

- The delay unit for HSYNC is one VCO Clock cycle.
- The delay unit for VSYNC is the HSYNC signal.

Notes:

- A value of 0 disables this feature.
- If this feature is used on a tandem engine, the minimum value for HSYNC and VSYNC is 1 for all channels.

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr>
<tr><td colspan="16">VOC2</td></tr>
</table>

<table>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="16">HOC2</td></tr>
</table>

Register bit assignment

Bits	Access	Mnemonic	Reset	Description
D31:16	R/W	VOC2	0x0	VSYNC offset channel 2
D15:00	R/W	HOC2	0x0	HSYNC offset channel 2

Table 25: Video HSYNC VSYNC Delay Channel 2 register

Video HSYNC VSYNC Delay Channel 1

Address: A050 0058

The Video HSYNC VSYNC Delay Channel 1 register delays the HSYNC and VSYNC signals for channel 1:

- The delay unit for HSYNC is one VCO Clock cycle.
- The delay unit for VSYNC is the HSYNC signal.

Notes:

- A value of 0 disables this feature.
- If this feature is used on a tandem engine, the minimum value for HSYNC and VSYNC is 1 for all channels.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
VOC1															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HOC1															

Register bit assignment

Bits	Access	Mnemonic	Reset	Description
D31:16	R/W	VOC1	0x0	VSYNC offset channel 1
D15:00	R/W	HOC1	0x0	HSYNC offset channel 1

Table 26: Video HSYNC VSYNC Delay Channel 1 register

Video HSYNC VSYNC Delay Channel 0

Address: A050 005C

The Video HSYNC VSYNC Delay Channel 0 register delays the HSYNC and VSYNC signals for channel 0:

- The delay unit for HSYNC is one VCO Clock cycle.
- The delay unit for VSYNC is the HSYNC signal.

Notes:

- A value of 0 disables this feature.
- If this feature is used on a tandem engine, the minimum value for HSYNC and VSYNC is 1 for all channels.

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr>
<tr><td colspan="16">VOC0</td></tr>
</table>

<table>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="16">HOC0</td></tr>
</table>

Register bit assignment

Bits	Access	Mnemonic	Reset	Description
D31:16	R/W	VOC0	0x0	VSYNC offset channel 0
D15:00	R/W	HOC0	0x0	HSYNC offset channel 0

Table 27: Video HSYNC VSYNC Delay Channel 0 register bit definition

Output FIFO Ready/Underflow Interrupt Control and Status

Address: A050 0060

The Output FIFO Ready/Underflow Interrupt Control and Status register configures and reads the status of the output FIFO ready/underflow interrupt. The output FIFO ready and output FIFO underflow signals are combined to create two interrupts as configured by bits 0-3.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				OFUS3	OFUS2	OFUS1	OFUS0	OFRS3	OFRS2	OFRS1	OFRS0	OFR UE3	OFR UE2	OFR UE1	OFR UE0

Register bit assignment

Bits	Access	Mnemonic	Reset	Description
D31:12	R	Reserved	0x0	N/A
D11	R	OFUS3	0x0	Output FIFO underflow channel 3 status Raw status of output FIFO underflow status signals.
D10	R	OFUS2	0x0	Output FIFO underflow channel 2 status Raw status of output FIFO underflow status signals.
D09	R	OFUS1	0x0	Output FIFO underflow channel 1 status Raw status of output FIFO underflow status signals.
D08	R	OFUS0	0x0	Output FIFO underflow channel 0 status Raw status of output FIFO underflow status signals.
D07	R	OFRS3	0x0	Output FIFO ready channel 3 status Raw status of output FIFO ready status signals.
D06	R	OFRS2	0x0	Output FIFO ready channel 2 status Raw status of output FIFO ready status signals.
D05	R	OFRS1	0x0	Output FIFO ready channel 1 status Raw status of output FIFO ready status signals.
D04	R	OFSR0	0x0	Output FIFO ready channel 0 status Raw status of output FIFO ready status signals.
D03	R/W	OFRUE3	0x0	Output FIFO ready/underflow enable channel 3 0 Disable channel in interrupt generation 1 Enable channel in interrupt generation

Table 28: Output FIFO Ready/Underflow Interrupt Control and Status register

Bits	Access	Mnemonic	Reset	Description
D02	R/W	OFRUE2	0x0	Output FIFO ready/underflow enable channel 2 0 Disable channel in interrupt generation 1 Enable channel in interrupt generation
D01	R/W	OFRUE1	0x0	Output FIFO ready/underflow enable channel 1 0 Disable channel in interrupt generation 1 Enable channel in interrupt generation
D00	R/W	OFRUE0	0x0	Output FIFO ready/underflow enable channel 0 0 Disable channel in interrupt generation 1 Enable channel in interrupt generation

Table 28: Output FIFO Ready/Underflow Interrupt Control and Status register

JBIG Decoder Host Interface registers

The JBIG Decoder Host Interface registers are the same for each JBIG Decoder Host interface. Each register is explained only once in this section.

JBIG Decoder 3 Host Interface

Address range: A050 0100–01FF
Reserved addresses within the interface are: A050 0100-0120, A050 0128-0140, A050 0150, and A050 0158-0178.

JBIG Decoder 2 Host Interface

Address range: A050 0200–02FF
Reserved addresses within the interface are: A050 0200-0220, A050 0228-0240, A050 0250, and A050 0258-0278.

JBIG Decoder 1 Host Interface

Address range: A050 0300–03FF
Reserved addresses within the interface are: A050 0300-0320, A050 0328-0340, A050 0350, and A050 0358-0378.

JBIG Decoder 0 Host Interface

Address range: A050 0400–04FF
Reserved addresses within the interface are: A050 0400-0420, A050 0428-0440, A050 0450, and A050 0458--478.

Lines per Stripe in Image

Address: A050 0124 / 0224 / 0324 / 0424
The Line per Stripe in Image register sets the number of lines per stripe in the image being decoded.

Note: This field does not apply if using auto-header processing mode.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SL															

Register bit assignment

Bits	Access	Mnemonic	Reset	Description
D31:16	R	Reserved	0x0	N/A
D15:00	R/W	SL	0x0	Stripe_Lines Number of lines per stripe in image

Table 29: Lines per Stripe in Image

Pixels per Line in Image

Address: A050 0144 / 0244 / 0344 / 0444

The Pixels per Line in Image register sets the number of pixels per line in the image being decoded.

Note: This field does not apply if using auto-header processing mode.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PIXIN															

Register bit assignment

Bits	Access	Mnemonic	Reset	Description
D31:16	R	Reserved	0x0	N/A
D15:00	R/W	PIXIN	0x0	Pixel_In Number of pixels per line in image

Table 30: Pixels per Line in Image

Total Lines in Image

Address: A050 0148 / 0248 / 0348 / 0448

The Total Lines in Image register sets the number of lines in the image to be decoded.

Note: This field does not apply if using auto-header processing mode.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
INL															

Register bit assignment

Bits	Access	Mnemonic	Reset	Description
D31:16	R	Reserved	0x0	N/A
D15:00	R/W	INL	0x0	In_Lines Number of lines in image

Table 31: Total Lines in Image

JBIG Control register

Address: A050 14C / 024C / 034C / 044C

The JBIG Control register sets the JBIG header parameters for the image to be decoded.

Note: This field does not apply if using auto-header processing mode.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				Rsvd	Rsvd	Rsvd	Rsvd	CX3	ATP						

Register bit assignment

Bits	Access	Mnemonic	Reset	Description
D31:12	N/A	Reserved	0x0	N/A
D11	N/A	Reserved	0x1	N/A
D10	N/A	Reserved	0x0	N/A
D09	N/A	Reserved	0x1	N/A
D08	N/A	Reserved	0x0	N/A
D07	R/W	CX3	0x0	Cx_3line 0 Three line template encoded 1 Two line template encoded

Table 32: JBIG Control register

Bits	Access	Mnemonic	Reset	Description
D06:00	R/W	ATP	0x0	At_Pixel AT pixel location

Table 32: JBIG Control register

Auto-Header mode

Address: A050 0154 / 0254 / 0354 / 0454

The Auto-Header Mode register controls the JBIG Decoder operating mode.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved												ABP	BP2 MH	BP1 MH	AHME

Register bit assignment

Bits	Access	Mnemonic	Reset	Description
D31:04	R	Reserved	0x0	N/A
D03	R/W	ABP	0x0	Active bit plane Always set to 1 (Bit plane 1)
D02	R/W	BP2MH	0x0	Bit plane 2 manual header mode 0 Auto-header mode 1 Manual header mode
D01	R/W	BP1MH	0x0	Bit plane 1 manual header mode 0 Auto-header mode 1 Manual header mode

Table 33: Auto-Header Mode register

Bits	Access	Mnemonic	Reset	Description
D00	R/W	AHME	0x0	Auto-header mode enable 0 You parse the JBIG header and manually program the JBIG settings. 1 The JBIG decoder parses the JBIG header and extracts the JBIG control parameters automatically.

Table 33: Auto-Header Mode register

Enable Status register

Address: A050 017C / 027C / 037C / 047C

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th></tr>
<tr><td colspan="16">Reserved</td></tr>
</table>

<table>
<tr><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="15">Reserved</td><td>Enable</td></tr>
</table>

Register bit assignment

Bits	Access	Mnemonic	Reset	Description
D31:01	R	Reserved	0x0	N/A
D00	R	Enable	0x0	Enable 0 JBIG decoder idle 1 JBIG decoder active

Table 34: Enable Status register

Print Engine Interface Module CPU Interface registers

Address range: A050 0500–05FF

Video Control register

Address: A050 0500

The Video Control register sets the printer-dependent control settings in the Print Engine Interface module.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved							MEPI_				MSPI_				VARM

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved						HSP	VSP	VDSD	VDP	VCAE	SYASY	TAND	PPOL	PRT	VRST_

Register bit assignment

Bits	Access	Mnemonic	Reset	Description
D31:25	R	Reserved	0x0	N/A
D24:21	R/W	MEPI_	0x0	Mask end of planes (EOP) interrupt 0 Cleared bit. The corresponding EOP[3:0] signal is prevented from causing an interrupt to the CPU (signal interrupt). 1 Set bit. The corresponding EOP[3:0] signal causes an interrupt to the CPU (signal interrupt).
D20:17	R/W	MSPI_	0x0	Mask start of planes (SOP) interrupt 0 Cleared bit. The corresponding SOP[3:0] signal is prevented from causing an interrupt to the CPU (signal interrupt). 1 Set bit. The corresponding SOP[3:0] signal causes an interrupt to the CPU (signal interrupt).

Table 35: Video Control register

Bits	Access	Mnemonic	Reset	Description
D16	R/W	VARM	0x0	Video arm 0 Cleared bit. Disables the detection of vertical and horizontal sync signals from the engine. 1 Set bit. Allows detection of sync signals.
D15:10	R	Reserved	0x0	N/A
D09	R/W	HSP	0x0	Horizontal sync polarity 0 HSYNC_3:0 signals are active low 1 HSYNC_3:0 signals are active high Defines the polarity of the input horizontal sync HSYNC_3:0 signals provided by the engine.
D08	R/W	VSP	0x0	Vertical sync polarity 0 VSYNC_3:0 signals are active low 1 VSYNC_3:0 signals are active high Defines the polarity of the input vertical sync VSYNC_3:0 signals provided by the engine.
D07	R/W	VDSD	0x0	Video data scan direction 0 Scan direction is forward 1 Scan direction is reverse Defines the scan direction of video data. This bit may be needed to accommodate special video data scan requirements for engines with duplex capabilities.
D06	R/W	VDP	0x0	Video data polarity 0 VDO_3:0 signals are active low 1 VDO_3:0 signals are active high Defines the polarity of the output VEDO_3:0 signals going to the engine.
D05	R/W	VCAE	0x0	Video clock active edge 0 Video clock active edge is rising edge 1 Video clock active edge is falling edge This bit is applies only to synchronous engines, and its choice is determined by which video clock edge the engine uses to latch video data.

Table 35: Video Control register

Bits	Access	Mnemonic	Reset	Description
D04	R/W	SYASY	0x0	Synchronous/Asynchronous 0 The video clock source is asynchronous through the internal video clock synthesizer/ synchronizer. 1 The video clock source is synchronous through `VCLK_3:0` pins.
D03	R/W	TAND	0x0	Tandem 0 The engine used is a non-tandem engine; that is, either monochrome or sequential four-pass color) 1 The engine used is a tandem engine (that is, one-pass color).
D02	R/W	PPOL	0x0	Print polarity 0 PRINT signal is active low 1 PRINT signal is active high Defines the polarity of the output PRINT signal going to the engine.
D01	R/W	PRT	0x0	Print 0 PRINT signal will not be asserted 1 PRINT signal will be asserted When set to 1, drives the PRINT signal to the engine and tells the engine to start printing
D00	R/W	VRST_	0x0	Video logic reset 0 All video logic and byte FIFOs are held to proper initial conditions. 1 All video logic and byte FIFOs are enabled for normal operation.

Table 35: Video Control register

Video Status register

Address: A050 0510

The Video Status register reads the status of the Print Engine Interface module.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved													HS		

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VS				BFEF				EOP				SOP			

Register bit assignment

Bits	Access	Mnemonic	reset	Description
D31:20	R	Reserved	0x0	N/A
D19:16	R	HS	0x0	Horizontal sync 0 The corresponding HSYNC signal is not asserted. 1 The corresponding HSYNC signal is asserted. Reflect the current state of the video horizontal sync signals. These bits are intended for debugging purposes, and are not needed for normal operation.
D15:12	R	VS	0x0	Vertical sync 0 The corresponding VSYNC signal is not asserted. 1 The corresponding VSYNC signal is asserted. Reflect the current state of the video vertical sync signals. These bits are intended for debugging purposes, and are not needed for normal operation.

Table 36: Video Status register

Bits	Access	Mnemonic	reset	Description
D11:08	R	BFEF	0x0	Byte FIFO empty flags 0 The corresponding byte FIFO is not empty. 1 The corresponding byte FIFO is empty. Reflect the current state of the video byte FIFOs empty flags. These bits are read-only, but because signal VRST_ resets all byte FIFOs, these bits will be 1s when VRST_ is cleared to 0. These bits are intended for debugging purposes, and are not needed for normal operation.
D07:04	R	EOP	0x0	End of planes When set to 1, an end-of-plane condition (VRST_ is set to 1, VARM is set to 1, and vertical as well as horizontal null and image run counts are expired), has been found since the last time that bit was cleared. Each bit can be cleared by writing the appropriate data word to offset 0x18 (for example, write 0x00000010 to clear only EOP(0), 0x00000080 to clear only EOP(3), 0x000000F0 to clear all four bits, and so on). Each bit can cause an interrupt to the CPU (signal interrupt) if its corresponding mask bit — MEPI_(3:0) — is set to 1. After clearing any of the EOP(3:0) bits, this register must be read to verify that those bits are cleared.
D03:00	R	SOP	0x0	Start of planes When set to 1, a start-of-plane condition (VRST_ is set to 1, VARM is set to 1, and the corresponding VSYNC_3:0 signal is asserted) has been found since the last time that bit was cleared. Each bit can be cleared by writing the appropriate data word to offset 0x18 (for example, 0x00000001 to clear only SOP(0), 0x00000008 to clear only SOP(3), 0x0000000F to clear all four bits, and so on). Each bit can cause an interrupt to the CPU (signal interrupt) if its corresponding mask bit — MSPI_(3:0) — is set to 1. After clearing any of the SOP(3:0) bits, this register must be read to verify that those bits are cleared.

Table 36: Video Status register

Video Vertical Margin and Data register

Address: A050 0520

The Video Vertical Margin and Data register provides enough bits to span a maximum page length of 27 inches (685.5 mm) at up to 2400 dpi. A single register applies to all four planes for color printers. On hardware reset, all bits are cleared to 0. This register can be written to or read from offset `0x20`.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
VIRC															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VNRC															

Register bit assignment

Bits	Access	Mnemonic	Reset	Description
D31:16	R/W	VIRC	0x0	Vertical image run count Number of scan lines of the image to be sent to the engine after the VNRC (D15:00) expires.
D15:00	R/W	VNRC	0x0	Vertical null run count Number of scan lines to be left blank after `VSYNC_3:0` is asserted.

Table 37: Video Vertical Margin and Data register

Video Horizontal Margin and Data register

Address: A050 0530

The Video Horizontal Margin and Data register provides enough bits to span a maximum page width of 27 inches (685.5 mm) at up to 2400 dpi. A single register applies to all four planes for color printers. On hardware reset, all bits are cleared to 0.

This register can be written to or read from offset `0x30`. All *Reserved* bits must be set to zeros when writing this register.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved			HIRC												

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved			HNRC												

Register bit assignment

Bits	Access	Mnemonic	Reset	Description
D31:29	R	Reserved	0x0	N/A
D28:16	R/W	HIRC	0x0	Horizontal image run count Number of pixel bytes (8 pixels) of the image to be sent to the engine for each scan line after the HNRC (D12:00) expires.
D15:13	R	Reserved	0x0	N/A
D12:00	R/W	HNRC	0x0	Horizontal null run count Number of pixel bytes (8 pixels) to be left blank on each scan line after `HSYNC_3:0` is asserted.

Table 38: Video Horizontal Margin and Data register bit definition

JBIG DMA Controller

CHAPTER 4

The four-channel DMA controller services the four input FIFOs.

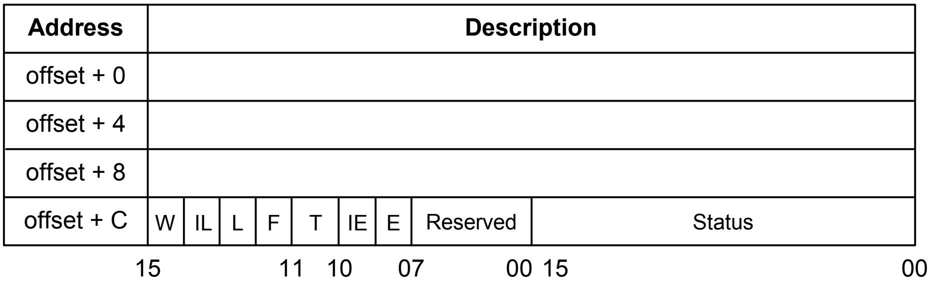
Overview

The DMA controller handles the four input FIFOs in round-robin manner. When one of the FIFOs is not full — that is, it can receive an 8- or 16-burst of 32-bit words as programmed in a CSR — the DMA controller requests the AHB bus through the AHB master. After the request has been granted, the image data is transferred from system memory.

Note: Buffer descriptors and buffers must start on a 16-bit boundary (per the JBIG decoder).

Buffer Descriptor

Image data is held in buffers in external memory linked together using buffer descriptors. Buffer descriptors are 16 bytes in length, and are located contiguously in external memory. Figure 14 shows the buffer descriptor format. Table 39 describes each buffer descriptor field.



Address	Description								
offset + 0									
offset + 4									
offset + 8									
offset + C	W	IL	L	F	T	IE	E	Reserved	Status
	15				11	10		07 00	15 00

Figure 14: Buffer descriptor

Field	Definition
Buffer pointer	32-bit pointer to the start of the buffer in system memory.l
Buffer length	Length of the buffer in bytes. The buffer allows an image size of 2^{16} -1 bytes to be in a single buffer. **Note:** Bits 31:16 are not used.
W (Control[15])	Wrap bit. When set, this bit informs the DMA controller that this is the last buffer descriptor within the contiguous list of descriptors. The next descriptor is found using the initial DMA channel buffer descriptor pointer. When the Wrap bit is not set, the next buffer descriptor is found using the 16-byte offset.
IL (Control[14])	Interrupt on Last bit. When set, this bit tells the DMA controller to issue an interrupt when the buffer is closed due to normal channel completion. This bit is set only if this is the last buffer in the image, as indicated when the Last bit is set.
L (Control[13])	Last bit. This bit is set when the current buffer is the last in the image. The DMA channel uses this bit to generate an interrupt to the CPU indicating that the last image buffer has been retrieved from system memory. The DmaChNLast bit is set in the Interrupt Status register (see "Interrupt Status register," beginning on page 32), where *N* is 0, 1, 2, or 3.
F (Control[12])	Full bit. This bit is set by firmware after the image data is written to a buffer. The DMA controller clears this bit as each buffer is read from external memory. If the DMA controller finds that this bit is not set when the buffer descriptor is read, the DmaChNBNRIP bit is set in the Interrupt Status register (see "Interrupt Status register," beginning on page 32) and the DMA controller halts immediately. (*N* is 0, 1, 2, or 3). The DMA controller must be soft-reset after the buffer descriptor problem has been resolved.
T (Control [11:10])	Type bits. This value identifies the type of data in the buffer: 00 JBIG header 01 JBIG compressed image data 10 Reserved 11 Uncompressed image data

Table 39: Buffer descriptor field definition

Field	Definition
IE (Control [09])	Interrupt on End bit. When set, this bit tells the DMA controller to issue an interrupt when the buffer is closed due to normal channel completion. This bit is set only if this is the last buffer in the print job, as indicated when the End bit is set.
E (Control[08])	End bit. Indicates that this is the last buffer in the last image of a print job. The DmaChNLast bit is set in the Interrupt Status register (see "Interrupt Status register," beginning on page 32) after this bit is found and the Output FIFO is emptied. *N* is 0, 1, 2, or 3.
Control[7:0]	Reserved
Status [15:0]	Reserved

Table 39: Buffer descriptor field definition

The image is ready to be transferred after the last buffer in the image has been placed in system memory and after the F and L bits have been set in the last buffer descriptor. The system software configures the DmaChNCurrBdPtr bit in the DMA Channel 3-0 Current Buffer Descriptor Pointer register and then sets the related DmaChNEn bit in the GenConfig register.

Buffer descriptor read process

When the appropriate bits are set in the DMA Channel 3-0 Current Buffer Descriptor Pointer and GenConfig registers, the DMA begins the buffer descriptor and buffer data read process. Data is read from external memory through the AHB master.

The read process is as follows:

1 The DmaChNCurrBdPtr pointer is written to the t_adr bus. The t_trans bus is set to indicate a 4-word transfer. The t_ads_n signal is pulsed low for one clock cycle to initiate the transfer on the AHB bus. The AHB master asserts the t_rdy signal when the data is read from external memory. The DMA controller now has the first buffer descriptor.

2 The DMA controller starts to read the image data from external memory, by writing the buffer pointer address to the t_adr bus. The t_trans bus is set to read either 32 or 64 bytes, as set in the GenConfig register. The AHB master asserts the t_rdy_n signal for each burst of data. The data is tagged according to the Type field in the buffer descriptor and written to the corresponding input FIFO. The buffer length is then decremented by either 32 or 64.

– If the remaining length is greater than the 32- or 64-byte configured transfer size, the burst is repeated.

– If the remaining length is less than the configured transfer size, a burst of unspecified length (INCR) is generated to read the exact number of remaining bytes in the buffer.

3 The F bit is cleared and the buffer descriptor is written back to external memory after the last byte has been transferred to the input FIFO.

– If the L bit is set, the DMA controller generates an interrupt to the CPU but continues with the next image as defined in the next buffer descriptor.

– If the E bit is set, the DMA controller goes to the idle state.

An interrupt is generated after the Output FIFO becomes empty. This indicates the last page in the print job.

Figure 15 shows a sample buffer-descriptor-to-buffer-link list.

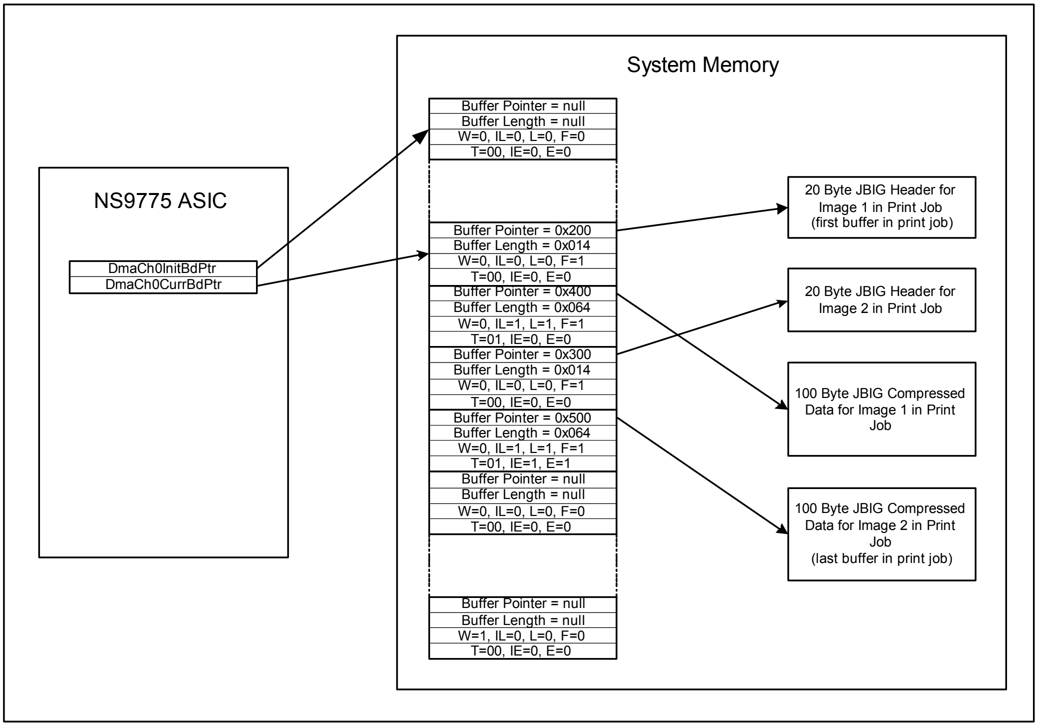


Figure 15: Buffer linking example

Print Engine Controller Configuration

CHAPTER 5

The Print Engine Controller module uses several registers to configure its operation. This chapter details the programming sequence required to send a page of data to a print engine. Three scenarios are described:

- Sending data to a synchronous print engine
- Sending data to an asynchronous print engine
- Sending data to a multi-clock source asynchronous print engine

Working with a synchronous configuration

Printing an image to a synchronous engine means that the print engine provides the pixel data clock to the NS9775.

Sample image characteristics

- Page size: 8.5 x 11 inches
- Resolution: 1200 dpi horizontal x 600 dpi vertical
- Top margin: 1 inch
- Left margin: 1.5 inch

The sample image is divided into four buffers for this example.

- First buffer: 20 bytes long and contains the JBIG header
- Remaining three buffers: Contain compressed image data

The sample image is monochrome; therefore, only one channel is detailed. With a tandem engine, all four channels are used.

Table 40 shows the buffer descriptors for the image. Table 41 shows the data buffers for the image.

Buffer descriptor	Location
JBIG header buffer descriptor	0x1000_0000
First compressed buffer descriptor	0x1000_0010
Middle compressed buffer descriptor	0x1000_0020
Last compressed buffer descriptor	0x1000_0030

Table 40: Buffer descriptor map

Data buffer	Location
JBIG header, 20 bytes	0x1001_0000
First compressed data buffer, 204 bytes	0x1001_0200
Middle compressed data buffer, 500 bytes	0x1001_0540
Last compressed data buffer, 56 bytes	0x1001_0800

Table 41: Data buffer map

What to do

▶ **To configure the Print Engine Controller module to print the image contained in the four buffers:**

1 Write `0x0000_3111` to the GenConfig register:

- a Remove soft reset from the DMA controller, Input FIFO 0, Output FIFO 0, Print Engine Interface module, and JBIG Decoder module.
- b Select Print Engine Interface mode rather than bypass mode.

2 Write `0x1000_0000` to `DmaCh0InitBdPtr`:

The buffer descriptor pointing to the JBIG header is the first buffer descriptor in the ring. When the `Wrap` bit is set, hardware reads from this buffer descriptor again.

3 Write `0x1000_0000` to `DmaCh0CurrBdPtr`:

This tells hardware which is the first buffer descriptor to read once the `DmaCh0En` bit is set.

4 Set up the first buffer descriptor for the JBIG header in system memory, as follows:

- a Write `0x1001_0000` to address `0x1000_0000`.
- b Write `0x0000_0014` to address `0x1000_0004`.
- c Write `0x0000_0000` to address `0x1000_0008`.
- d Write `0x1000_0000` to address `0x1000_000C`.

This initializes the first buffer descriptor for the JBIG header to these values:

W	= 0
IL	= 0
I	= 0
E	= 0
IE	= 0
Pointer	= 0x1001_0000
Status	= 0x0000
F	= 1
Length	= 0x14 (20 bytes)
T	= 0x0 (JBIG header)

5 Set up the first buffer descriptor for the first compressed image data buffer in system memory, as follows:

a Write `0x1001_0200` to address `0x1000_0010`

b Write `0x0000_00CC` to address `0x1000_0014`

c Write `0x0000_0000` to address `0x1000_0018`

d Write `0x1400_0000` to address `0x1000_001C`

This initializes the buffer descriptor to these values:

W	= 0
IL	= 0
I	= 0
E	= 0
IE	= 0
Pointer	= 0x1001_0200
Status	= 0x0000
F	= 1
Length	= 0xCC (204 bytes)
T	= 0x1 (JBIG compressed image data)

6 Set up the first buffer descriptor for the middle compressed image data buffer in system memory, as follows:

a Write `0x1001_0540` to address `0x1000_0020`

b Write `0x0000_01F4` to address `0x1000_0024`

c Write `0x0000_0000` to address `0x1000_0028`

d Write `0x1400_0000` to address `0x1000_002C`

This initializes the buffer descriptor to these values:

W	= 0
IL	= 0
I	= 0
E	= 0
IE	= 0
Pointer	= 0x1001_0540
Status	= 0x0000
F	= 1
Length	= 0x1F4 (500 bytes)
T	= 0x1 (JBIG compressed image data)

7 Set up the first buffer descriptor for the last compressed image data buffer in system memory, as follows:

a Write `0x1001_0800` to address `0x1000_0030`

b Write `0x0000_0038` to address `0x1000_0034`

c Write `0x0000_0000` to address `0x1000_0038`

d Write `0x7700_0000` to address `0x1000_003C`

This initializes the buffer descriptor to these values:

W	= 0
IL	= 1
I	= 1
E	= 1
IE	= 1
Pointer	= 0x1001_0800
Status	= 0x0000
F	= 1
Length	= 0x38 (56 bytes)
T	= 0x1 (JBIG compressed image data)

8 Write `0x7111_1111` to the Interrupt Enable register:

- Bit 30, `OutputFIFOReadyEnable`: Enable the Output FIFO Ready interrupt.
- Bit 28, `Print Engine Interface IP Enable`: Enable Print Engine Interface interrupt.
- Bit 24, `JbigDecoder0NLIPEnable`: Enable JBIG Decoder 0 NewLen marker error interrupt.
- Bit 20, `JbigDecoder0ERRIPEnable`: Enable JBIG Decoder 0 error interrupt.
- Bit 16, `JbigDecoderNCIPEnable`: Enable JBIG Decoder 0 normal completion interrupt.
- Bit 12, `DmaCh0AIPEnable`: Enable DMA controller 0 channel abort interrupt.
- Bit 8, `DmaCh0BNRIPEnable`: Enable DMA controller 0 buffer not ready interrupt.
- Bit 4, `DmaCh0LastEnable`: Enable DMA controller 0 last buffer interrupt.
- Bit 0, `DmaCh0EndEnable`: Enable DMA controller 0 end buffer interrupt.

9 Write `0x0000_020D` to `OutputFifoReadyThreshold`:

a Set the FIFO threshold to one line of uncompressed data, in units of 16-bit words. The sample image width is 7 inches, with a resolution of 1200 dpi. This calculates to a total of 8400 pixels or 525 16-bit words.

10 Write `0x0000_0009` to the Auto Header Enable register for JBIG Decoder 0:

– Bit 3: Set the active plane to 1. The design only supports decoding one plane at a time per decoder.

– Bit 1: Enable automatic JBIG header processing. No further programming of the JBIG decoder control registers is required in this mode.

11 Write `0x0000_0001` to the Output FIFO Ready Interrupt Control and Status register:

– Bit 0: Enable channel 0 only.

12 Write `0x0022_0013` to the Video Control register. The Print Engine Interface block can now be enabled to send the image to the printer.

– Bits 24:21: Enable a start of plane interrupt for plane 0. The interrupt alerts software that printing has started.

– Bits 20:17: Enable an end of plane interrupt for plane 0. The interrupt alerts software that the entire page has printed.

– Bit 16: Disable the Print Engine Interface block until a line is available.

– Bit 9: Set the horizontal sync polarity to active low.

– Bit 8: Set the vertical sync polarity to active low.

– Bit 7: Set the video data scan direction to forward.

– Bit 6: Set the video data polarity to active low.

– Bit 5: Set the video clock active edge to the rising edge.

– Bit 4: Set the interface mode to synchronous. The print engine provides the clock.

– Bit 3: Set the mode to monochrome; only one channel — 0 — is used.

– Bit 2: Set the “print” output signal polarity to active low.

– Bit 1: Assert the “print” output signal. This tells the engine that a page is ready to print.

– Bit 0: Take the Print Engine Interface block out of reset.

13 Write `0x0000_187A` to the Video PLL Configuration register:

– Bit 16: Clear `VsyncHsyncMode` to prevent `VSYNC/HSYNC` synchronization.

Leave all other bits in their default state.

14 Write `0x0258_1770` to the Video Vertical Margin and Data register. The image is to be printed on 8.5 x 11 inch paper, with a 600 dpi vertical resolution and a 1-inch margin:

- Bits 31:16: Set the top margin to 600 (`0x258`) scan lines, for a 1-inch margin.
- Bits 15:0: Set the data length to 6000 (`0x1770`) scan lines. This accounts for the remainder of the 11-inch page width.

15 Write `0x00E1_141A` to the Video Horizontal Margin and Data register. The image is to be printed on 8.5 x 11 inch paper, with a 1200 dpi horizontal resolution and a 1.5-inch margin. The units are in 8-pixel words.

- Bits 31:16: Set the left margin to 225 (`0x00E1`) 8-bit pixel words, for a 1.5-inch left margin.
- Bits 15:0: Set the data width to 1050 (`0x041A`) 8-bit pixel words. This accounts for the remainder of the 8.5-inch page width.

16 Write `0x0001_3111` to the GenConfig register:

- Bit 16: Enable the DMA engine for channel 0. Hardware now transfers the encoded JBIG image from system memory.

17 Wait for the Output FIFO ready interrupt, to indicate that a full line has been decoded and placed in the Output FIFO.

Read the Interrupt Status register to verify that the Output FIFO ready interrupt is asserted.

- Bit 30: Verify that the OutputFIFOReadyIP bit is set. This indicates that a full line has been decoded and written to the Output FIFO for channel 0.

18 Write `0x3111_1111` to the Interrupt Enable register:

- Bit 30, `OutputFIFOReadyEnable`: Disable the Output FIFO ready interrupt until the start of the next print job.

19 Write `0x0023_0013` to the Video Control register. The Print Engine Interface block can now be enabled to send the image to the printer:

- Bits 24:21: Enable a start of plane interrupt for plane 0. The interrupt alerts software that printing has started.
- Bits 20:17: Enable an end of plane interrupt for plane 0. The interrupt alerts software that the entire page has printed.
- Bit 16: Enable the Print Engine Interface module block to send print data to the print engine as controlled by horizontal sync and vertical sync.
- Bit 9: Set the horizontal sync polarity to active low.

- Bit 8: Set the vertical sync polarity to active low.
- Bit 7: Set the video data scan direction to forward.
- Bit 6: Set the video data polarity to active low.
- Bit 5: Set the video clock active edge to the rising edge.
- Bit 4: Set the interface mode to synchronous. The print engine provides the clock.
- Bit 3: Set the mode to monochrome; only one channel — 0 — is used.
- Bit 2: Set the “print” output signal polarity to active low.
- Bit 1: Assert the “print” output signal. This tells the engine that a page is ready to print.
- Bit 0: Continue to keep the Print Engine Interface block out of reset.

20 Wait for the sequence of interrupts to indicate the processing of the JBIG image and printing status of the page. The last interrupt from the DMA engine indicates that the last data buffer at location `0x1001_0800` has been transferred from system memory.

The last interrupt is from the Print Engine Interface block — the end of plane interrupt for channel 0 — to indicate that the last scan line has finished.

Working with an Asynchronous Configuration

Printing an image to an asynchronous print engine means that the NS9775 provides the clock for the pixel data, synchronized to the horizontal `SYNC` signal provided by the print engine.

Sample image characteristics

- Page size: 8.5 x 11 inches
- Resolution: 1200 dpi horizontal x 600 dpi vertical
- Top margin: 1 inch
- Left margin: 1.5 inch

Clock details

- Reference oscillator clock frequency: 26.4 MHz
- VCO frequency: 132 MHz
- PLL frequency: 528 MHz
- Pixel clock frequency: 33 MHz
- PLL multiplier setting: 20
- FS divider setting: 4
- Divider setting: 4

The formula for figuring the clock details for an asynchronous interface is determined as follows:

- Determine the required pixel clock frequency — `fpixel`.
- Determine the horizontal synchronization resolution factor — `hre`s. The minimum for most applications is 1/4 pixel. This value determines the VCO clock frequency — f_{VCO}.
- Determine the reference oscillator clock frequency — `fref` — as shown, with the allowable range of 20–40 MHz:

  ```
  fref = (fpixel x (1/hres)) / PLL multiplier
  ```

  Note that the PLL frequency must be kept in the 400–800 MHz range.

The sample image is divided into four buffers for this example.

- First buffer: 20 bytes long and contains the JBIG header
- Remaining three buffers: Contain compressed image data

The sample image is monochrome; therefore, only one channel is detailed. With a tandem engine, all four channels are used.

Table 42 shows the buffer descriptors for the image. Table 43 shows the data buffers for the image.

Buffer descriptor	Location
JBIG header buffer descriptor	0x1000_0000
First compressed buffer descriptor	0x1000_0010
Middle compressed buffer descriptor	0x1000_0020
Last compressed buffer descriptor	0x1000_0030

Table 42: Buffer descriptor map

Data buffer	Location
JBIG header, 20 bytes	0x1001_0000
First compressed data buffer, 204 bytes	0x1001_0200
Middle compressed data buffer, 500 bytes	0x1001_0540
Last compressed data buffer, 56 bytes	0x1001_0800

Table 43: Data buffer map

What to do

▶ **To configure the Print Engine Controller module to print the image contained in the four buffers:**

1 Write `0x0001_3849` to the Video PLL Configuration register:

- Bit 14: Leave `VsyncHsyncMode` set until the Vertical Sync Polarity and Horizontal Sync Polarity bits have been programmed in the Video Control register.
- Bit 13: Update the PLL settings, as set by the other fields in this register.
- Bits 12:10: Set the divider value to `0x6` to provide a divide-by-4 setting.
- Bit 9: Use the PLL to provide the VCO clock; do not bypass the PLL.
- Bits 8:7: Set the frequency select value to `10`, and the divider value to 4.
- Bits 4:0 Set the PLL multiplier setting to 9.

  Note that the value to write here is the actual multiplier value, 10 in this sample case, minus 1.

2 Read the Video PLL Configuration register to determine if the PLL has locked. This normally takes about 4 milliseconds. The lock indication is bit 31; a 1 indicates that the PLL has locked.

3 Write `0x0000_3111` to the GenConfig register:

   a Remove soft reset from the DMA controller, Input FIFO 0, Output FIFO 0, Print Engine Interface module, and JBIG Decoder module.

   b Select Print Engine Interface mode rather than bypass mode.

4 Write `0x1000_0000` to `DmaCh0InitBdPtr`:

   The buffer descriptor pointing to the JBIG header is the first buffer descriptor in the ring. When the `Wrap` bit is set, hardware reads from this buffer descriptor again.

5 Write `0x1000_0000` to `DmaCh0CurrBdPtr`:

   This tells hardware which is the first buffer descriptor to read once the `DmaCh0En` bit is set.

6 Set up the first buffer descriptor for the JBIG header in system memory, as follows:

   a Write `0x1001_0000` to address `0x1000_0000`

   b Write `0x0000_0014` to address `0x1000_0004`

   c Write `0x0000_0000` to address `0x1000_0008`

   d Write `0x1000_0000` to address `0x1000_000C`

   This initializes the first buffer descriptor for the JBIG header to these values:

W	= 0
IL	= 0
I	= 0
E	= 0
IE	= 0
Pointer	= 0x1001_0000
Status	= 0x0000
F	= 1
Length	= 0x14 (20 bytes)
T	= 0x0 (JBIG header)

7 Set up the first buffer descriptor for the first compressed image data buffer in system memory, as follows:

a Write `0x1001_0200` to address `0x1000_0010`

b Write `0x0000_00CC` to address `0x1000_0014`

c Write `0x0000_0000` to address `0x1000_0018`

d Write `0x1400_0000` to address `0x1000_001C`

This initializes the buffer descriptor to these values:

W	= 0
IL	= 0
I	= 0
E	= 0
IE	= 0
Pointer	= 0x1001_0200
Status	= 0x0000
F	= 1
Length	= 0xCC (204 bytes)
T	= 0x1 (JBIG compressed image data)

8 Set up the first buffer descriptor for the middle compressed image data buffer in system memory, as follows:

a Write `0x1001_0540` to address `0x1000_0020`

b Write `0x0000_01F4` to address `0x1000_0024`

c Write `0x0000_0000` to address `0x1000_0028`

d Write `0x1400_0000` to address `0x1000_002C`

This initializes the buffer descriptor to these values:

W	= 0
IL	= 0
I	= 0
E	= 0
IE	= 0
Pointer	= 0x1001_0540
Status	= 0x0000
F	= 1
Length	= 0x1F4 (500 bytes)
T	= 0x1 (JBIG compressed image data)

9 Set up the first buffer descriptor for the last compressed image data buffer in system memory, as follows:

a Write `0x1001_0800` to address `0x1000_0030`

b Write `0x0000_0038` to address `0x1000_0034`

c Write `0x0000_0000` to address `0x1000_0038`

d Write `0x7700_0000` to address `0x1000_003C`

This initializes the buffer descriptor to these values:

W	= 0
IL	= 1
I	= 1
E	= 1
IE	= 1
Pointer	= 0x1001_0800
Status	= 0x0000
F	= 1
Length	= 0x38 (56 bytes)
T	= 0x1 (JBIG compressed image data)

10 Write 0x7111_1111 to the Interrupt Enable register:

- Bit 30, `OutputFIFOReadyEnable`: Enable the Output FIFO ready interrupt.
- Bit 28, `Print Engine Interface IP Enable`: Enable the Print Engine Interface interrupt.
- Bit 24, `JbigDecoder0NLIPEnable`: Enable JBIG Decoder 0 NewLen marker error interrupt.
- Bit 20, `JbigDecoder0ERRIPEnable`: Enable JBIG Decoder 0 error interrupt.
- Bit 16, `JbigDecoderNCIPEnable`: Enable JBIG Decoder 0 normal completion interrupt.
- Bit 12, `DmaCh0AIPEnable`: Enable DMA controller 0 channel abort interrupt.
- Bit 8, `DmaCh0BNRIPEnable`: Enable DMA controller 0 buffer not ready interrupt.
- Bit 4, `DmaCh0LastEnable`: Enable DMA controller 0 last buffer interrupt.
- Bit 0, `DmaCh0EndEnable`: Enable DMA controller 0 end buffer interrupt.

11 Write `0x0000_020D` to `OutputFifoReadyThreshold`:

a Set the FIFO threshold to one line of uncompressed data, in units of 16-bit words. The sample image width is 7 inches, with a resolution of 1200 dpi. This calculates to a total of 8400 pixels, or 525 16-bit words.

12 Write `0x0000_0009` to the Auto Header Enable register for JBIG decoder 0:

- Bit 3: Set the active plane to 1. The design only supports decoding one plane at a time per decoder.
- Bit 1: Enable automatic JBIG header processing. The JBIG decoder parses JBIG settings from the JBIG header. No further programming of JBIG Decoder Control registers is required in this mode.

13 Write `0x0000_0001` to the Output FIFO Ready Interrupt Control and Status register:

- Bit 0: Enable channel 0 only.

14 Write `0x0022_0003` to the Video Control register:

- Bits 24:21: Enable a start of plane interrupt for plane 0. The interrupt alerts software that printing has started.
- Bits 20:17: Enable an end of plane interrupt for plane 0. The interrupt alerts software that the entire page has been printed.
- Bit 16: Disable the Print Engine Interface block until a line of data is ready.
- Bit 9: Set the horizontal sync polarity to active low.
- Bit 8: Set the vertical sync polarity to active low.
- Bit 7: Set the video data scan direction to go forward.

- Bit 6: Set the video data polarity to active low.
- Bit 5: Set the video clock active edge to the rising edge.
- Bit 4: Set the interface mode to asynchronous. The NS9775 provides the clock.
- Bit 3: Set the mode to monochrome. Only one channel — 0 — is used.
- Bit 2: Set the "print" output signal polarity to active low.
- Bit 1: Assert the "print" output signal. This tells the engine that a page is ready to print.
- Bit 0: Take the Print Engine Interface block out of reset.

15 Write `0x0000_1849` to the Video PLL Configuration register:

- Bit 14: Clear the `VsyncHsyncMode` bit. Do not synchronize `VSYNC` to `HSYNC`.

Leave remaining bits as programmed in Step 1.

16 Write `0x0258_1770` to the Video Vertical Margin and Data register. The image is to be printed on 8.5 x 11 inch paper, with a 600 dpi vertical resolution and a 1-inch margin.

- Bits 31:16: Set the top margin to 600 (`0x258`) scan lines, for a one-inch margin.
- Bits 15:0: Set the data length to 6000 (`0x1770`) scan lines. This accounts for the remainder of the 11-inch page length.

17 Write `0x00E1_041A` to the Video Horizontal Margin and Data register. The image is to be printed on 8.5 x 11 inch paper, with a 1200 dpi horizontal resolution and a 1.5-inch margin. The units are in 8-bit pixel words.

- Bits 31:16: Set the left margin to 225 (`0x00E1`) 8-bit pixel words, for a 1.5-inch left margin.
- Bits 15:0: Set the data width to 1050 (`0x041A`) 8-bit pixel words. This accounts for the remainder of the 8.5-inch page width.

18 Write `0x0001_3111` to the GenConfig register:

- Bit 16: Enable the DMA engine for channel 0. Hardware will now transfer the encoded JBIG image from system memory.

19 Wait for the Output FIFO ready interrupt, to indicate that a full line has been decoded and placed in the Output FIFO.

Read the Interrupt Status register to verify that the Output FIFO ready interrupt has become active.

- Bit 30: Verify that the `OutputFIFOReadyIP` bit is set. This indicates that a full line has been decoded and written to the Output FIFO for channel 0.

20 Write `0x3111_1111` to the Interrupt Enable register:

– Bit 30, `OutputFIFOReadyEnable`: Disable the Output FIFO ready interrupt until the start of the next print job.

21 Write `0x0023_0003` to the Video Control register:

– Bits 24:21: Enable a start-of-plane interrupt for plane 0. The interrupt alerts software that printing has started.

– Bits 20:17: Enable an end-of-plane interrupt for plane 0. The interrupt alerts software that the entire page has printed.

– Bit 16: Enable the Print Engine Interface block to send print data to the print engine as controlled by horizontal sync and vertical sync.

– Bit 9: Set the horizontal sync polarity to active low.

– Bit 8: Set the vertical sync polarity to active low.

– Bit 7: Set the video data scan direction to forward.

– Bit 6: Set the video data polarity to active low.

– Bit 5: Set the video clock active edge to the rising edge.

– Bit 4: Set the interface mode to asynchronous. The NS9775 provides the clock.

– Bit 3: Set the mode to monochrome. Only one channel — 0 — is used.

– Bit 2: Set the “print” output signal polarity to active low.

– Bit 1: Assert the “print” output signal. This tells the engine that a page is ready to print.

– Bit 0: Continue to keep the Print Engine Interface block out of reset.

22 Wait for the sequence of interrupts to indicate the JBIG image processing and printing status of the page. The last interrupt from the DMA engine indicates that the last data buffer at location `0x1001_0800` has been transferred from system memory.

The last interrupt is from the Print Engine Interface block — the end-of-plane interrupt for channel 0 — to indicate that the last scan line has finished.

Multi-Clock Source Asynchronous Configuration

Printing an image to a multi-clock source asynchronous print engine means that the NS9775 provides the clock for pixel data, synchronized to the horizontal `SYNC` signal provided by the print engine. Instead of using the PLL output as the source of VCO, however, the clock generator uses `vclk_0` as the source.

Sample image characteristics

- Page size: 8.5 x 11 inches
- Resolution: 1200 dpi horizontal x 600 dpi vertical
- Top margin: 1 inch
- Left margin: 1.5 inch

Clock details

- `vclk_0` input frequency: 145 MHz
- Pixel clock frequency: 72.5 MHz
- Divider setting: 2

The sample image is divided into four buffers for this example.

- First buffer: 20 bytes long and contains the JBIG header
- Remaining three buffers: Contain compressed image data

The sample image is monochrome; therefore, only one channel is detailed. With a tandem engine, all four channels are used.

Table 44 shows the buffer descriptors for the image. Table 45 shows the data buffers for the image.

Buffer descriptor	Location
JBIG header buffer descriptor	0x1000_0000
First compressed buffer descriptor	0x1000_0010
Middle compressed buffer descriptor	0x1000_0020
Last compressed buffer descriptor	0x1000_0030

Table 44: Buffer descriptor map

Data buffer	Location
JBIG header, 20 bytes	0x1001_0000
First compressed data buffer, 204 bytes	0x1001_0200
Middle compressed data buffer, 500 bytes	0x1001_0540
Last compressed data buffer, 56 bytes	0x1001_0800

Table 45: Data buffer map

What to do

▶ **To configure the Print Engine Controller module to print the image contained in the four buffers:**

1 Write `0x0001_543A` to the Video PLL Configuration register:

– Bit 14: Use `vclk_0` input as the source clock for the clock generator.

– Bits 12:10: Set the divider value to `0x5` to provide a divide-by-2 setting.

– Bits 9:0: Leave in default state (“don’t care” bits)

2 Write `0x0000_3111` to the GenConfig register:

– Remove soft reset from the DMA controller, Input FIFO 0, Output FIFO 0, Print Engine Interface module, and JBIG Decoder module.

– Select Print Engine Interface mode rather than bypass mode.

3 Write `0x1000_0000` to `DmaCh0InitBdPtr`:

The buffer descriptor pointing to the JBIG header is the first buffer descriptor in the ring. When the `Wrap` bit is set, Hardware reads from this buffer descriptor again.

4 Write `0x1000_0000` to `DmaCh0CurrBdPtr`:

This tells hardware which is the first buffer to read once the `DmaCh0En` bit is set.

5 Set up the first buffer descriptor for the JBIG header in system memory, as follows:

a Write `0x1001_0000` to address `0x1000_0000`.

b Write `0x0000_0014` to address `0x1000_0004`.

c Write `0x0000_0000` to address `0x1000_0008`.

d Write `0x1000_0000` to address `0x1000_000C`.

This initializes the first buffer descriptor for the JBIG header to these values:

W	= 0
IL	= 0
I	= 0
E	= 0
IE	= 0
Pointer	= 0x1001_0000
Status	= 0x0000
F	= 1
Length	= 0x14 (20 bytes)
T	= 0x0 (JBIG header)

6 Set up the first buffer descriptor for the first compressed image data buffer in system memory, as follows:

a Write `0x1001_0200` to address `0x1000_0010`.

b Write `0x0000_00CC` to address `0x1000_0014`.

c Write `0x0000_0000` to address `0x1000_0018`.

d Write `0x1400_0000` to address `0x1000_001C`.

This initializes the buffer descriptor to these values:

W	= 0
IL	= 0
I	= 0
E	= 0
IE	= 0
Pointer	= 0x1001_0200
Status	= 0x0000
F	= 1
Length	= 0xCC (204 bytes)
T	= 0x1 (JBIG compressed image data)

7 Set up the first buffer for the middle compressed image data buffer in system memory, as follows:

a Write `1001_0540` to address `0x1000_0020`.

b Write `0x0000_01F4` to address `0x1000_0024`.

c Write `0x0000_0000` to address `0x1000_0028`.

d Write `0x1400_0000` to address `0x1000_002C`.

This initializes the buffer descriptor to these values:

W	= 0
IL	= 0
I	= 0
E	= 0
IE	= 0
Pointer	= 0x1001_0540
Status	= 0x0000
F	= 1
Length	= 0x1F4 (500 bytes)
T	= 0x1 (JBIG compressed image data)

8 Set up the first buffer descriptor for the last compressed image data buffer in system memory, as follows:

- **a** Write `0x1001_0800` to address `0x1000_0030`.
- **b** Write `0x0000_0038` to address `0x1000_0034`.
- **c** Write `0x0000_0000` to address `0x1000_0038`.
- **d** Write `0x7700_0000` to address `0x1000_003C`.

This initializes the buffer descriptor to these values:

W	= 0
IL	= 1
I	= 1
E	= 1
IE	= 1
Pointer	= 0x1001_0800
Status	= 0x0000
F	= 1
Length	= 0x38 (56 bytes)
T	= 0x1 (JBIG compressed image data)

9 Write `0x7111_1111` to the Interrupt Enable register:

- Bit 30, `OutputFIFOReadyEnable`: Enable the Output FIFO ready interrupt.
- Bit 28, `Print Engine Interface IP Enable`: Enable the Print Engine Interface interrupt.
- Bit 24, `JbigDecoder0NLIPEnable`: Enable the JBIG Decoder 0 NewLen marker error interrupt.
- Bit 20, `JbigDecoder0ERRIPEnable`: Enable JBIG Decoder 0 error interrupt.
- Bit 16, `JbigDecoderNCIPEnable`: Enable the JBIG Decoder 0 normal completion interrupt.
- Bit 12, `DmaCh0AIPEnable`: Enable the DMA controller 0 channel abort interrupt.
- Bit 8, `DmaCh0BNRIPEnable`: Enable DMA controller 0 buffer not ready interrupt.
- Bit 4, `DmaCh0LastEnable`: Enable DMA controller 0 last buffer interrupt.
- Bit 0, `DmaCh0EndEnable`: Enable DMA controller 0 end buffer interrupt.

10 Write `0x0000_020D` to `OutputFIFOReadyThreshold`.

a Set the FIFO threshold to one line of uncompressed data, in units of 16-bit words. The sample image width is 7 inches, with a resolution of 1200 dpi. This calculates to a total of 8400 pixels or 525 16-bit words.

11 Write `0x0000_0009` to the Auto Header Enable register for JBIG decoder 0.

- Bit 3: Set the active plane to 1. The design only supports decoding one plane at a time per decoder.
- Bit 1: Enable automatic JBIG header processing. The JBIG decoder parses the settings from the JBIG header. No other programming of JBIG Decoder Control registers is required in this mode.

12 Write `0x0000_0001` to the Output FIFO Ready Interrupt Control and Status register:

- Bit 0: Enable channel 0 only.

13 Write `0x0022_0003` to the Video Control register:

- Bits 24:21: Enable a start-of-plane interrupt for plane 0. The interrupt alerts software that printing has started.
- Bits 20:17: Enable an end-of-plane interrupt for plane 0. The interrupt alerts software that the entire page has printed.
- Bit 16: Disable the Print Engine Interface block until a line of data is ready.
- Bit 9: Set the horizontal sync polarity to active low.
- Bit 8: Set the vertical sync polarity to active low.
- Bit 7: Set the video data scan direction to forward.
- Bit 6: Set the video data polarity to active low.
- Bit 5: Set the video clock active edge to the rising edge.
- Bit 4: Set the interface mode to asynchronous. The NS9775 provides the clock.
- Bit 3: Set the mode to monochrome. Only one channel — 0 — is used.
- Bit 2: Set the “print” output signal polarity to active low.
- Bit 1: Assert the “print” output signal. This tells the engine that a page is ready to print.
- Bit 0: Take the Print Engine Interface block out of reset.

14 Write `0x0258_1770` to the Video Vertical Margin and Data register. The image is to print on 8.5 x 11 inch paper, with a 600 dpi vertical resolution and a 1-inch margin.

- Bits 31:16: Set the top margin to 600 (`0x258`) scan lines, for a 1-inch margin.
- Bits 15:0: Set the data length to 6000 (`0x1770`) scan lines. This accounts for the remainder of the 11-inch page length.

15 Write `0x00E1_141A` to the Video Horizontal Margin and Data register. The image is to print on 8.5 x 11 inch paper, with a 1200 dpi horizontal resolution and a 1.5-inch margin. The units are in 8-bit pixel words.

– Bits 31:16: Set the left margin to 225 (`0x00E1`) 8-bit pixel words, for a 1.5-inch left margin.

– Bits 15:0: Set the data width to 1050 (`0x041A`) 8-bit pixel words. This accounts for the remainder of the 8.5-inch page width.

16 Write `0x0001_3111` to the GenConfig register:

– Bit 16: Enable the DMA engine for channel 0. Hardware transfers the encoded JBIG image from system memory.

17 Wait for the Output FIFO ready interrupt, which indicates that a full line has been decoded and placed in the Output FIFO.

Read the Interrupt Status register to verify that the Output FIFO ready interrupt has become active.

– Bit 30: Verify that this bit is set, to indicate that a full line has been decoded and written to the Output FIFO for channel 0.

18 Write `0x3111_1111` to the Interrupt Enable register:

– Bit 30, `OutputFIFOReadyEnable`: Disable the Output FIFO ready interrupt until the start of the next print job.

19 Write `0x0023_0003` to the Video Control register:

– Bits 24:21: Enable a start-of-plane interrupt for plane 0. The interrupt alerts software that printing has started.

– Bits 20:17: Enable an end-of-plane interrupt for plane 0. The interrupt alerts software that the entire page has printed.

– Bit 16:Enable the Print Engine Interface block to send print data to the print engine as controlled by horizontal sync and vertical sync.

– Bit 9: Set the horizontal sync polarity to active low.

– Bit 8: Set the vertical sync polarity to active low.

– Bit 7: Set the video data scan direction to forward.

– Bit 6: Set the video data polarity to active low.

– Bit 5: Set the video clock active edge to the rising edge.

– Bit 4: Set the interface mode to asynchronous. The NS9775 provides the clock.

– Bit 3: Set the mode to monochrome. Only one channel — 0 — is used.

– Bit 2: Set the “print” output signal polarity to active low.

- Bit 1: Assert the “print” output signal. This tells the engine that a page is ready to print.
- Bit 0: Continue to keep the Print Engine Interface block out of reset.

20 Wait for the sequence of interrupts to indicate JBIG image processing and printing status of the page. The last interrupt from the DMA engine indicates that the last data buffer at location `0x1001_0800` has been transferred from system memory.

The last interrupt is from the Print Engine Interface block — the end-of-plane interrupt for channel 0 — to indicate that the last scan line has finished.

Digi®