1.8V MULTI-QUEUE FLOW-CONTROL DEVICES (128 QUEUES) 40 BIT WIDE CONFIGURATION

5,242,880 bits
10,485,760 bits

IDT72P51767
IDT72P51777

FEATURES

- Choose from among the following memory density options:
  IDT72P51767 — Total Available Memory = 5,242,880 bits
  IDT72P51777 — Total Available Memory = 10,485,760 bits
- Configurable from 1 to 128 Queues
- Multiple default configurations of symmetrical queues
- Default multi-queue device configurations
  – IDT72P51767: 512 x 40 x 128Q
  – IDT72P51777: 1,024 x 40 x 128Q
- Number of queues and queue sizes may be configured; at master reset, though serial programming, (via the queue address bus)
- 166 MHz High speed operation (6ns cycle time)
- 0.48ns access time
- Independent Read and Write access per queue
- Echo Read Clock available
- Internal PLL
- On-chip Output Impedance matching
- User Selectable Bus Matching Options:
  – x40 in to x40 out *– x20 in to x20 out*
  – x40 in to x20 out *– x20in to x40out*
- User selectable I/O: 1.5V HSTL or 1.8V eHSTL
- 100% Bus Utilization, Read and Write on every clock cycle
- Selectable Back off one (BOI) or IDT standard mode of operation
- Ability to operate on packet or word boundaries
- Mark and Re-Write operation
- Mark and Re-Read operation
- Individual, Active queue flags ($\overline{\text{EF}}$, $\overline{\text{FF}}$, $\overline{\text{PAE}}$, $\overline{\text{PAF}}$)
- 8 bit parallel flag status on both read and write ports
- Direct or polled operation of flag status bus
- Expansion of up to 256 queues
- JTAG Functionality (Boundary Scan)
- Available in a 376-pin BGA, 1mm pitch, 23mm x 23mm
- HIGH Performance submicron CMOS technology
- Industrial temperature range (-40°C to +85°C) is available
- Green parts available, seeing Ordering Information

FUNCTIONAL BLOCK DIAGRAM

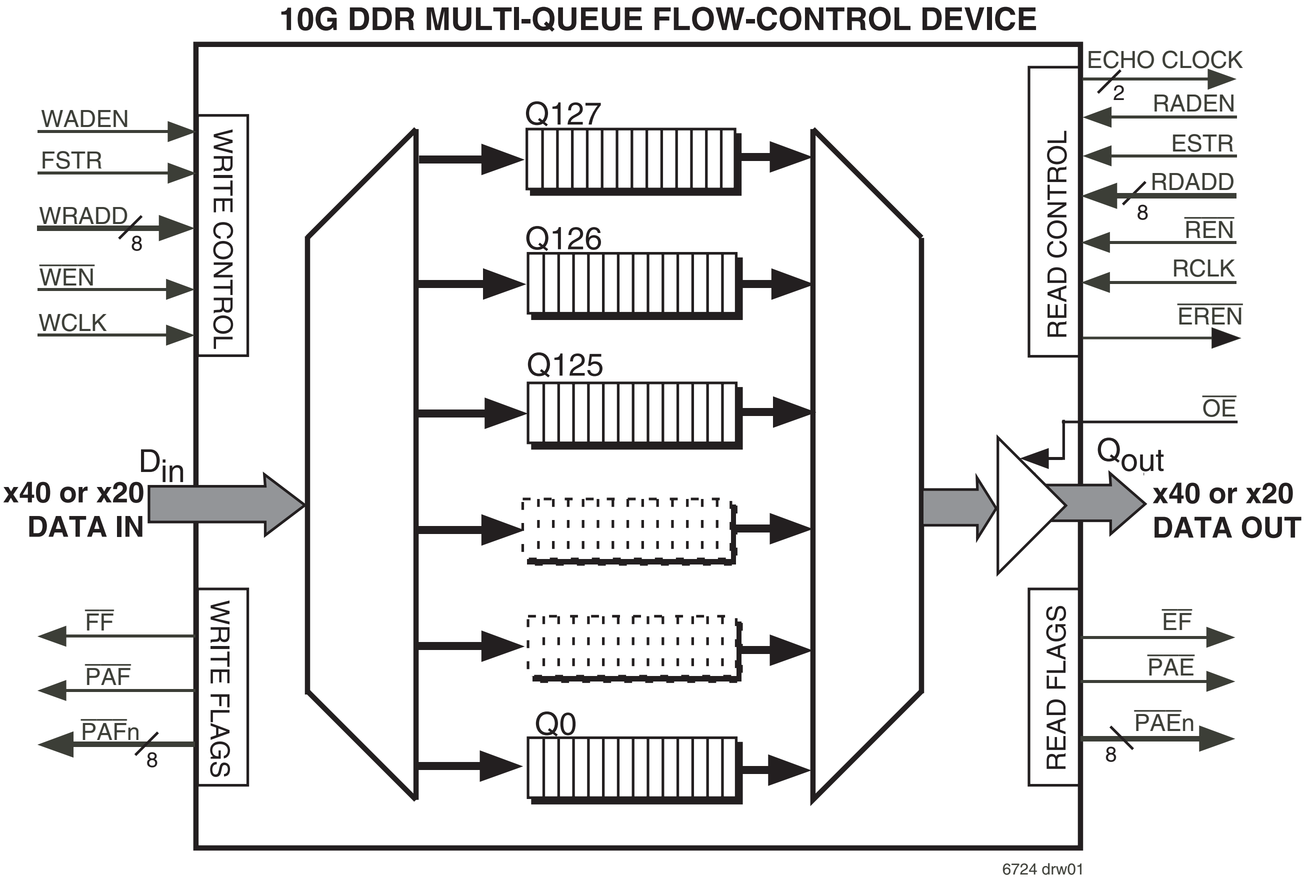


Table of Contents

List of Tables

List of Figures

List of Figures (Continued)

DESCRIPTION

The IDT72P51767/ IDT72P51777 multi-queue flow-control devices are single chip solutions containing up to 128 configurable queues. All queues within the device have a common data input bus, Din [39:0] (write port) and a common data output bus Qout [39:0], (read port). Data written into the write port is directed to a respective queue via an integrated de-multiplex function. Data read from the read port is accessed from a given queue transparently via an internal multiplex operation. Data writes and reads can be performed at high speeds up to 166MHz DDR allowing data rates up to 10Gigabits/s (OC-192). By utilizing high speed interfaces such as 1.5V HSTL, coupled with a x40 bit data bus and 10Mb of data storage, the 10G Multi-Queue can interface with the industry standard 10 Gigabits/sec Media Independent Interface (XGMII) to allow high speed data transmission over 10G Ethernet and SONET line cards. Data write and read operations are totally independent of each other. The Write Clock and Read Clock can operate at independent frequencies. A different queue may be selected on the write port and read port or both ports may select the same queue simultaneously. Multiple clocking schemes are offered for this device as well. The user can utilize either single ended or differential clocking for DDR read operations. DDR write operation utilize a single ended clock. SDR write and read operations utilize a single ended clock.

The devices provide Full flag and Empty flag status for the queue selected for write and read operations respectively. Also a Programmable Almost Full ($\overline{\text{PAF}}$) and Programmable Almost Empty ($\overline{\text{PAE}}$) flag for each queue is provided. Two 8 bit programmable flag busses ($\overline{\text{PAF}}$n, $\overline{\text{PAE}}$n) are available, providing status of queues that are not the present queue selected for write or read operations. When 8 or fewer queues are configured in the device, these flag busses provide an individual flag per queue, when more than 8 queues are used; the queue status is multiplexed through the 8 stus lines. The multiplexing can be configured either a Polled or Direct mode of bus.

Bus Matching is available on this device; either port can be x20 bits or x40 bits wide. When Bus Matching is used the device ensures the logical transfer of data throughput.. With a 40 data bits configuration parity checking and packet tagging is achievable if desired. Parity checking is available through the use of 4 user selectable bits as part of the 40 bit word. The user will be able to pass along parity bits through the Multi-Queue to use for error detection in a up/down stream device. The Multi-Queue device does not provide parity checking circuits.

In Back off One mode, the user can switch queues without having to read the last pipelined data word that is stored in the output register which in IDT standard mode is required to be read out during a queue switch . The last pipelined data word in BOI mode is retained in the output data register until it is actively read.

A Mark and Re-write and a Mark and Re-read function are available on the write and read ports respectively. These functions allows for a mark location to be independently issued on the read and/or write ports, in their respective queues. The option to reset a given queue to the mark location effectively dropping data written into the queue or allow data to be read again from the device.

The devices offer a default configuration upon reset, offering 128 symmetrical queues configured at start-up, which means the user can program the number of queues to divide the 10Mb/5Mb of memory depending on the device. The Multi-Queues can even be programmed to support one single queue to be used as a FIFO for high performance applications of sequential queuing. The programmable flag positions are also user programmable. If the user does not wish to program the multi-queue device, a default option is available that configures the device in a predetermined manner. A Master Reset latches in all configuration setup pins and must be performed before programming of the device can take place.

The multi-queue flow-control devices have the capability of operating its I/O in either 1.5V HSTL , or 1.8V eHSTL mode. The type of I/O is selected via the IOSEL input. The core supply voltage (VCC) to the multi-queue is always 1.8V, however the output levels can be set independently via a separate supply, VDDQ. The package used will be a 23mm x 23mm, BB-376 BGA package for better noise immunity and ground bounce prevention.

A JTAG test port is provided, here the multi-queue flow-control device has a fully functional Boundary Scan feature, compliant with IEEE 1149.1 Standard Test Access Port and Boundary Scan Architecture.

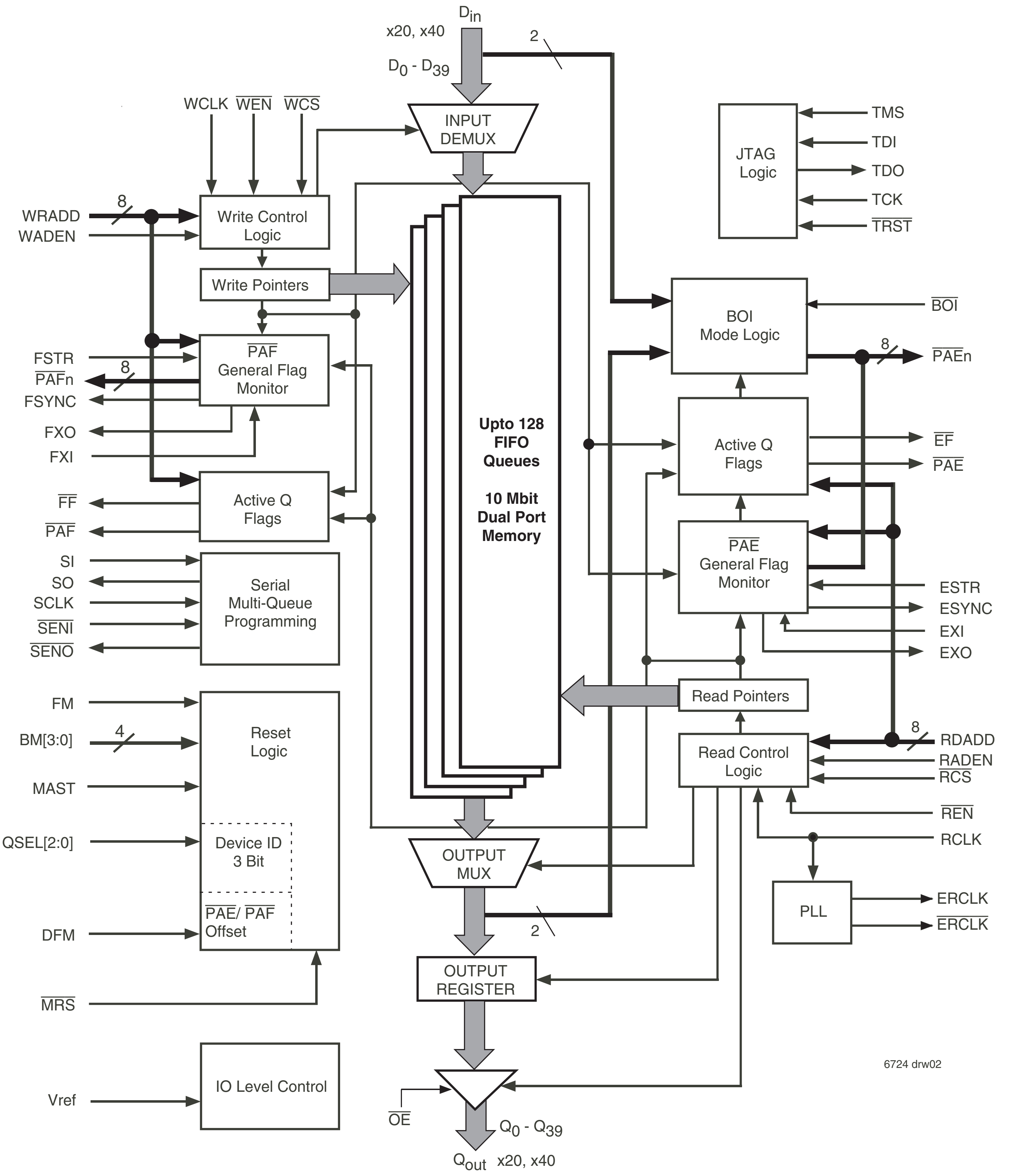


Figure 1. Multi-Queue Flow-Control Device Block Diagram

PIN CONFIGURATION

A1 BALL PAD CORNER

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
A	GND	GND	D11	D9	D7	D5	D3	D1	$\overline{\text{TRST}}$	TMS	TDI	ID2	ID0	Q1	Q3	Q5	Q7	Q9	Q11	Q13	GND	GND
B	GND	GND	D12	D10	D8	D6	D4	D2	D0	TCK	TDO	ID1	Q0	Q2	Q4	Q6	Q8	Q10	Q12	Q14	GND	GND
C	GND	GND	D13	D14	VDDQ	VDDQ	VDDQ	VDDQ	VDDQ	VDDQ	GND	GND	VDDQ	VDDQ	VDDQ	VDDQ	VDDQ	VDDQ	Q16	Q15	GND	GND
D	D17	D16	D15	VDD	VDD	VDD	VDD	VDD	VDD	VDD	GND	GND	VDD	VDD	VDD	VDD	VDD	VDD	VDD	Q17	Q18	Q19
E	$\overline{\text{FF}}$	D19	D18	VDD	VDD	VDD	VDD	VDD	VDD	GND	GND	GND	GND	VDD	VDD	VDD	VDD	VDDQ	VDD	DNC	$\overline{\text{EREN}}$	ERCLK
F	$\overline{\text{PAF6}}$	$\overline{\text{PAF7}}$	$\overline{\text{PAF}}$	VDDQ	VDDQ													VDDQ	VDDQ	$\overline{\text{ERCLK}}$	$\overline{\text{PAE7}}$	$\overline{\text{PAE6}}$
G	$\overline{\text{PAF3}}$	$\overline{\text{PAF4}}$	$\overline{\text{PAF5}}$	VDDQ	VDDQ													VDD	VDDQ	$\overline{\text{PAE5}}$	$\overline{\text{PAE4}}$	$\overline{\text{PAE3}}$
H	$\overline{\text{PAF0}}$	$\overline{\text{PAF1}}$	$\overline{\text{PAF2}}$	VDD	VDD													VDD	VDD	$\overline{\text{PAE2}}$	$\overline{\text{PAE1}}$	$\overline{\text{PAE0}}$
J	VREF	FSYNC	FXO	VDD	GND				GND	GND	GND	GND	GND	GND				GND	VDD	$\overline{\text{REN}}$	$\overline{\text{RCS}}$	RCLK
K	$\overline{\text{SENI}}$	$\overline{\text{SENO}}$	SCLK	GND	GND				GND	GND	GND	GND	GND	GND				GND	GND	GND	AVDD	AVSS
L	$\overline{\text{WCS}}$	SI	SO	GND	GND				GND	GND	GND	GND	GND	GND				GND	GND	GND	AVSS	AVSS
M	$\overline{\text{WEN}}$	WCLK	WADEN	GND	GND				GND	GND	GND	GND	GND	GND				GND	GND	GND	VREF	AVDD
N	WRADD0	WRADD1	WRADD2	GND	GND				GND	GND	GND	GND	GND	GND				GND	GND	$\overline{\text{PAE}}$	$\overline{\text{EF}}$	ZQ
P	WRADD3	WRADD4	WRADD5	VDD	GND				GND	GND	GND	GND	GND	GND				GND	VDD	$\overline{\text{BOI}}$	ESYNC	EXO
R	WRADD6	WRADD7	VDD	VDD	VDD													VDD	VDD	ESTR	EXI	$\overline{\text{OE}}$
T	FXI	FSTR	$\overline{\text{MRS}}$	VDDQ	VDDQ													VDDQ	VDDQ	RDADD1	RDADD0	RADEN
U	MAST	FM	TP	VDDQ	VDDQ													VDDQ	VDDQ	RDADD4	RDADD3	RDADD2
V	PLLON	D20	D21	VDD	VDD	VDD	VDD	VDD	GND	GND	GND	GND	GND	GND	VDD	VDD	VDD	VDD	VDD	RDADD7	RDADD6	RDADD5
W	D22	D23	D24	VDD	VDD	VDD	VDD	VDD	VDD	VDD	GND	GND	VDD	VDD	VDD	VDD	VDD	VDD	VDD	Q22	Q21	Q20
Y	GND	GND	D26	D25	VDDQ	VDDQ	VDDQ	VDDQ	VDDQ	VDDQ	GND	GND	GND	VDDQ	VDDQ	VDDQ	VDDQ	VDDQ	Q23	Q24	GND	GND
AA	GND	GND	D27	D29	D31	D33	D35	D37	D39	QSEL1	DFM	BM2	Q39	Q37	Q35	Q33	Q31	Q29	Q27	Q25	GND	GND
BB	GND	GND	D28	D30	D32	D34	D36	D38	QSEL0	QSEL2	BM0	BM1	BM3	Q38	Q36	Q34	Q32	Q30	Q28	Q26	GND	GND

6724 drw03

NOTE:
1. DNC - Do Not Connect.

PBGA (BB376-1, order code: BB)
TOP VIEW

DETAILED DESCRIPTION

MULTI-QUEUE STRUCTURE

The IDT multi-queue flow-control device has a single data input port and single data output port with up to 128 FIFO queues in parallel buffering between the two ports. The user can setup between 1 and 128 Queues within the device. These queues can be configured to utilize the total available memory, providing the user with full flexibility and ability to configure the queues to be various depths, independent of one another.

MEMORY ORGANIZATION/ ALLOCATION

The memory is organized into what is known as "blocks", each block being 256 x40 bits. When the user is configuring the number of queues and individual queue sizes the user must allocate the memory to respective queues, in units of blocks, that is, a single queue can be made up from 0 to m blocks, where m is the total number of blocks available within a device. Also the total size of any given queue must be in increments of 256 x40. For the IDT72P51767 and IDT72P51777 the Total Available Memory is 1024 and 512 blocks respectively (a block being 256 x40). Queues can be built from these blocks to make any size queue desired and any number of queues desired.

BUS WIDTHS

The input port is common to all queues within the device, as is the output port. The device provides the user with Bus Matching options such that the input port and output port can be either x20, x40 bits wide, the read and write port widths can be set independently of one another. Because a ports are common to all queues the width of the queues is not individually set. The input width of all queues are the same and the output width of all queues are the same.

WRITING TO & READING FROM THE MULTI-QUEUE

Data being written into the device via the input port is directed to a discrete queue via the write queue address input. Conversely, data being read from the device read port is read from a queue selected via the read queue address input. Data can be simultaneously written into and read from the same queue or different queues. Once a queue is selected for data writes or reads, the writing and reading operation is performed in the same manner as a conventional IDT synchronous FIFO, utilizing clocks and enables, there is a single clock and enable per port. When a specific queue is addressed on the write port, data placed on the data inputs is written to that queue sequentially based on the rising edge of a write clock provided setup and hold times are met. Conversely, data is read on to the output port after an access time from a rising edge on a read clock.

The operation of the write port is comparable to the function of a conventional FIFO operating in standard IDT mode. Write operations can be performed on the write port provided that the queue currently selected is not full, a full flag output provides status of the selected queue. When a queue is selected on the output port, the next word in that queue will be available for reading on the output register. All subsequent words from that queue require an enabled read cycle. Data cannot be read from a selected queue if that queue is empty, the read port provides an Empty flag indicating when data read out is valid. If the user switches to a queue that is empty, the last word from the previous queue will remain on the output bus. The device can operate in IDT Standard mode or $\overline{\text{BOI}}$ mode. In IDT Standard mode the read port provides a word to the output bus (Qout) for each clock cycle that $\overline{\text{REN}}$ is asserted. Refer to Figure 46, *SDR Read Queue Select, Read Operation (IDT Mode)*.

As mentioned, the write port has a full flag, providing full status of the selected queue. Along with the full flag a dedicated almost full flag is provided, this almost full flag is similar to the almost full flag of a conventional IDT FIFO. The device provides a user programmable almost full flag for all 128 queues and when a respective queue is selected on the write port, the almost full flag provides status for that queue. Conversely, the read port has an Empty flag, providing status of the data being read from the queue selected on the read port. As well as the Empty flag the device provides a dedicated almost empty flag. This almost empty flag is similar to the almost empty flag of a conventional IDT FIFO. The device provides a user programmable almost empty flag for each 128 queues and when a respective queue is selected on the read port, the almost empty flag provides status for that queue.

PROGRAMMABLE FLAG BUSSES

In addition to these dedicated flags, full & almost full on the write port and Output Ready & almost empty on the read port, there are two flag status busses. An almost full flag status bus is provided, this bus is 8 bits wide. Also, an almost empty flag status bus is provided, again this bus is 8 bits wide. The purpose of these flag busses is to provide the user with a means by which to monitor the data levels within queues that may not be selected on the write or read port. As mentioned, the device provides almost full and almost empty registers (programmable by the user) for each of the 128 queues in the device.

In the IDT72P51767/72P51777 multi-queue flow-control devices the user has the option of utilizing 1 to 128 queues, therefore the 8 bit flag status busses are multiplexed between the 128 queues, a flag bus can only provide status for 8 of the 128 queues at any moment, this is referred to as a "Status Word", such that when the bus is providing status of queues 1 through 8, this is status word 1, when it is queues 9 through 16, this is status word 2 and so on up to status word 16. If less than 128 queues are setup in the device, there are still 4 status words, such that in "Polled" mode of operation the flag bus will still cycle through 4 status words. If for example only 22 queues are setup, status words 1 and 2 will reflect status of queues 1 through 8 and 9 through 16 respectively. Status word 3 will reflect the status of queues 17 through 22 on the least significant 6 bits, the most significant 2 bits of the flag bus are don't care. The remaining status words are not used as there are no queues to report.

The flag busses are available in two user selectable modes of operation, "Polled" or "Direct". When operating in polled mode a flag bus provides status of each status word sequentially, that is, on each rising edge of a clock the flag bus is updated to show the status of each status word in order. The rising edge of the write clock will update the almost full bus and a rising edge on the read clock will update the almost empty bus. The mode of operation is always the same for both the almost full and almost empty flag busses. When operating in direct mode, the status word on the flag bus is selected by the user. So the user can actually address the status word to be placed on the flag status busses, these flag busses operate independently of one another. Addressing of the almost full flag bus is done via the write port and addressing of the almost empty flag bus is done via the read port.

EXPANSION

Expansion of multi-queue devices is possible. Expansion achieves either depth or queue expansion. Depth Expansion means expanding the depths of individual queues. Queue expansion means increasing the total number of queues available. Depth expansion is possible by virtue of the fact that more memory blocks within a multi-queue device can be allocated to a fewer number of queues to increase the depth of each queue. For example, depth expansion of 2 devices provides the possibility of 2 queues, each queue being setup within a single device utilizing all memory blocks available to produce a single queue. This is the deepest queue that can setup within a device.

For queue expansion a maximum number of 256 queues (2 x 128 queues) may be setup. If fewer queues are desired, then more memory blocks will be available to increase queue depths if desired. Refer to Figure 61, *Connecting two 10G Multi-Queue 128Q devices in Expansion Mode*, and Figure 62, *Connecting three or more 10G Multi-Queue 128Q in Expansion mode using WRADD bit 7 / RDADD bit 7* for device connection details.

10Gbps MULTI-QUEUE DIFFERENCES FROM THE 4M MULTI-QUEUE

The 10G Multi-Queue was developed to support very high performance applications that needed 10Gb/s of bandwidth, and the flexibility of buffering packets of information in large bursts such as Jumbo Ethernet packets that can be as large as 9KBs. Listed below are the differences between the 10G Multi-Queue and the previous 4M Multi-Queue with descriptions of the enhancements made to support performance functions in queuing.

PERFORMANCE ENHANCEMENTS

- 333.34 Mbps (per pin) High speed data rate in DDR mode
- x40 Din and x40 Qout (8 more pins for user selectable operation such as parity check or packet tagging)
- Electrical compatibility to 802.3ae XGMII specification for passive interconnection to Ethernet devices.
  - Single clocking in DDR and SDR, PLL on/off Mode. (PAD_PLLON pin) allowing data latency to be the same for SDR and DDR.
- Burst of 2 timing and interface logic
  - Output impedance matching for signal quality on the output pins.
  - More Data latency (same cycle on write, 1 cycle on read)
  - Three "echo" output pins: ERCLK, ERCLK, and EREN used for Source Synchronous data on the output. Data can be center aligned on the Echo Clock or issued on the rising edge of the Echo Clock.
  - Access Time (Ta) reduced to 0.48ns with Echo Clock used for faster Synchronized data delivery down stream

USER FLEXIBILITY IMPROVEMENTS

- 10Mbits of storage and queuing density for support large packet frames such as Jumbo Ethernet
- During a Queue switch, BOI mode preserves the data word in the output register until it's read.
- "Real Time" Flags, for both DDR and SDR.
  - PAF/PAE have 1 more cycle (WCLK/RCLK) latency (3 vs. 2)
  - Tskew of EF/PAE with respect to WCLK has 1 WCLK cycle delay.
  - Tskew of FF/PAF with respect to RCLK has 1 RCLK cycle delay.
- Programmable Default configuration of 128, 64, 32, 16, 8 or 4 symmetrical queues are available using DFM, QSEL[2:0] pins
- User selectable I/O: 1.5V HSTL, or 1.8V eHSTL for faster switching I/O
- Expansion of up to 256 queues and/or 80Mbit logical configuration using up to 8 multi-queue devices
- Default flag offset value is defined according to bus matching configuration
  - The PAE flag can be used as a packet indicator

TABLE 1 — SUMMARY OF THE DIFFERENCES BETWEEN THE 4M MQ AND 10G MQ

FEATURE	4M MQ (IDT72P51769)	10M MQ (IDT72P51777)
Data Transfer Modes	SDR	SDR, DDR
Bus Width	x36, x18, x9	x40, x20
XGMII Compatibility	no	yes
Access time (ta)	3.6 ns max	0.48ns max
Data Storage Capacity	4Mb	10Mb
Data Throughput	7.2Gbps	10Gbps
Operating Frequency	200mhz	166mhz
Configurable Queues	Up to 128	Up to 128
Package	256 pin PBGA	376 pin BGA
Output Impedance Technology	no	yes
I/O Voltages	1.5V,1.8V, 2.5V	1.5V,1.8V
Echo read Clock	no	yes
Modes of Operation	FWFT, IDT, Packet	IDT, BOI
Output data Clocking	Edge aligned	Centered aligned

PIN DESCRIPTIONS

Symbol & (Pin No.)	Name	I/O TYPE	Description
BM [3:0] (BM3-BB13 BM2-AA12 BM1-BB12 BM0-BB11)	Bus Matching	1.8V LVTTL INPUT	These pins define the bus width and data transfer rate (DDR/SDR) of the input write port and the output read port of the device. The bus widths/data rates are set during a Master Rest cycle. The BM[3:0] signals must meet the setup and hold time requirements of Master Reset and must not toggle/change state after a Master Reset cycle.
$\overline{\text{BOI}}$ (P20)	Back Off One Mode	HSTL INPUT	When in BOI, data is back-off one position in which Packet 1 and Packet 2 are out again during second Queue Switch. See section on 10Gbps Multi-queue Differences from the 4M multi-queue, previous page.
D[39:0] (See Pin No. table for details)	Data Input Bus	HSTL INPUT	These are the 32 data input pins. Data is written into the device via these input pins on the rising edge of WCLK provided that $\overline{\text{WEN}}$ is LOW. Any unused data input pins should be tied HIGH. D[39:36] user definable input bits D[33] user definableD[32] user definableD[31:0] data input bits
DFM (AA11)	Default Mode	1.8V LVTTL INPUT	The 10G multi-queue device requires programming after master reset. The user can do this serially via the serial port, or the user can use the default method. If DFM is LOW at Master Reset then serial mode will be selected, if DFM is HIGH then default mode is selected.
$\overline{\text{EF}}$ (N21)	Empty Flag	HSTL OUTPUT	The Empty Flag ($\overline{\text{EF}}$) provides valid status for the selected queue. The Empty Flag indicates the selected queue is empty, all words have been read. This flag is delayed to match the data output path delay.
ERCLK (E22)	Echo Read Clock	HSTL OUTPUT	The rising edge of this clock is centered aligned with Qout data.
$\overline{\text{ERCLK}}$ (F20)	Echo Read Clock	HSTL OUTPUT	Read Clock Echo is the inverse of ERCLK.
$\overline{\text{EREN}}$ (E21)	Echo Read Enable	HSTL OUTPUT	Echo Read Enable output, used in conjunction with ERCLK and $\overline{\text{ERCLK}}$.
ESTR (R20)	$\overline{\text{PAE}}$n Flag Bus	HSTL INPUT	If direct operation of the $\overline{\text{PAE}}$n bus has been selected, the ESTR input is used in conjunction with RCLK and the RDADD bus to select a quadrant of queues to be placed on to $\overline{\text{PAE}}$n output. A quadrant addressed via the RDADD bus is selected on the rising edge of RCLK provided that ESTR is HIGH. If Polled operation has been selected, ESTR should be tied inactive, LOW. Note, that a $\overline{\text{PAE}}$n flag bus selection cannot be made, (ESTR must NOT go active) until programming of the part has been completed and SENO has gone LOW.
ESYNC (P21)	$\overline{\text{PAE}}$n Bus Sync	HSTL OUTPUT	ESYNC is an output from the multi-queue device that provides a synchronizing pulse for the $\overline{\text{PAE}}$n bus during Polled operation of the $\overline{\text{PAE}}$n bus. During Polled operation each quadrant of queue status flags is loaded on to the $\overline{\text{PAE}}$n bus outputs sequentially based on RCLK. The first RCLK rising edge loads quadrant 1 on to $\overline{\text{PAE}}$n, the second RCLK rising edge loads quadrant 2 and so on. The fifth RCLK rising edge will again load quadrant 1. During the RCLK cycle that quadrant 1 of a selected device is placed on to the $\overline{\text{PAE}}$n bus, the ESYNC output will be HIGH. For all other quadrants of that device, the ESYNC output will be LOW.
EXI (R21)	$\overline{\text{PAE}}$n Bus Expansion In	HSTL INPUT	The EXI input is used when multi-queue devices are connected in expansion mode and Polled $\overline{\text{PAE}}$n bus operation has been selected. EXI of device 'N' connects directly to EXO of device 'N-1'. The EXI receives a token from the previous device in a chain. In single device mode the EXI input must be tied LOW if the $\overline{\text{PAE}}$n bus is operated in direct mode. If the $\overline{\text{PAE}}$n bus is operated in polled mode the EXI input must be connected to the EXO output of the same device. In expansion mode the EXI of the first device should be tied LOW, when direct mode is selected.
EXO (P22)	$\overline{\text{PAE}}$n Bus Expansion Out	HSTL OUTPUT	EXO is an output that is used when multi-queue devices are connected in expansion mode and Polled $\overline{\text{PAE}}$n bus operation has been selected. EXO of device 'N' connects directly to EXI of device 'N+1'. This pin pulses when device N has placed its final (4th) quadrant on to the $\overline{\text{PAE}}$n bus with respect to RCLK. This pulse (token) is then passed on to the next device in the chain 'N+1' and on the next RCLK rising edge the first quadrant of device N+1 will be loaded on to the $\overline{\text{PAE}}$n bus. This continues through the chain and EXO of the last device is then looped back to EXI of the first device. The ESYNC output of each device in the chain provides synchronization to the user of this looping event.
$\overline{\text{FF}}$ (E1)	Full Flag	HSTL OUTPUT	This pin provides the full flag output for the active Queue, that is, the queue selected on the input port for write operations, (selected via WCLK, WRADD bus and WADEN). On the WCLK cycle after a queue selection, this flag will show the status of the newly selected queue. Data can be written to this queue provided $\overline{\text{FF}}$ is HIGH. This flag has High-Impedance capability, this is important during expansion of

PIN DESCRIPTIONS (CONTINUED)

Symbol & Pin No.	Name	I/O TYPE	Description
$\overline{\text{FF}}$ (Continued) (E1)	Full Flag	HSTL OUTPUT	devices, when the $\overline{\text{FF}}$ flag output of up to 8 devices may be connected together on a common line. The device with a queue selected takes control of the $\overline{\text{FF}}$ bus, all other devices place their $\overline{\text{FF}}$ output into High-Impedance. When a queue selection is made on the write port this output will switch from High-Impedance control on the next WCLK cycle. This flag is asserted synchronous to WCLK.
FM (U2)	Flag Mode	1.8V LVTTL INPUT	This pin is setup before a Master Reset and must not toggle during any device operation. The state of the FM pin during Master Reset will determine whether the $\overline{\text{PAF}}$n and $\overline{\text{PAE}}$n flag busses operate in either Polled or Direct mode. If FM is HIGH, Polled mode is selected, if FM LOW, Direct mode is selected.
FSTR (T2)	$\overline{\text{PAF}}$n Flag Bus Strobe	HSTL INPUT	If direct mode for the $\overline{\text{PAF}}$n bus has been selected, the FSTR input is used in conjunction with WCLK and the WRADD bus to select a quadrant of queues to be placed on to the $\overline{\text{PAF}}$n bus outputs. A quadrant addressed via the WRADD bus is selected on the rising edge of WCLK provided that FSTR is HIGH. If polled operations has been selected, FSTR should be tied inactive, LOW. Note, that a $\overline{\text{PAF}}$n flag bus selection cannot be made, (FSTR must NOT go active) until programming of the part has been completed and SENO has gone LOW.
FSYNC (J2)	$\overline{\text{PAF}}$n Bus Sync	HSTL OUTPUT	FSYNC is an output from the multi-queue device that provides a synchronizing pulse for the $\overline{\text{PAF}}$n bus during Polled operation of the $\overline{\text{PAF}}$n bus. During Polled operation each quadrant of queue status flags is loaded on to the $\overline{\text{PAF}}$n bus outputs sequentially based on WCLK. The first WCLK rising edge loads quadrant 1 on to $\overline{\text{PAF}}$n, the second WCLK rising edge loads quadrant 2 and so on. The fifth WCLK rising edge will again load quadrant 1 queue status flags. During the WCLK cycle that quadrant 1 of a selected device is placed on to the $\overline{\text{PAF}}$n bus, the FSYNC output will be HIGH. For all other quadrants of that device, the FSYNC output will be LOW.
FXI (T1)	$\overline{\text{PAF}}$n Bus Expansion In	HSTL INPUT	The FXI input is used when multi-queue devices are connected in expansion mode and Polled $\overline{\text{PAF}}$n bus operation has been selected . FXI of device 'N' connects directly to FXO of device 'N-1'. The FXI receives a token from the previous device in a chain. In single device mode the FXI input must be tied LOW if the $\overline{\text{PAF}}$n bus is operated in direct mode. If the $\overline{\text{PAF}}$n bus is operated in polled mode the FXI input must be connected to the FXO output of the same device. In expansion mode the FXI of the first device should be tied LOW, when direct mode is selected.
FXO (J3)	$\overline{\text{PAF}}$n Bus Expansion Out	HSTL OUTPUT	FXO is an output that is used when multi-queue devices are connected in expansion mode and Polled $\overline{\text{PAF}}$n bus operation has been selected . FXO of device 'N' connects directly to FXI of device 'N+1'. This pin pulses when device N has placed its final (4th) quadrant on to the $\overline{\text{PAF}}$n bus with respect to WCLK. This pulse (token) is then passed on to the next device in the chain 'N+1' and on the next WCLK rising edge the first quadrant of device N+1 will be loaded on to the $\overline{\text{PAF}}$n bus. This continues through the chain and FXO of the last device is then looped back to FXI of the first device. The FSYNC output of each device in the chain provides synchronization to the user of this looping event.
ID[2:0] (ID2-A12 ID1-B12 ID0-A13)	Device ID Pins	1.8V LVTTL INPUT	The ID[2:0] pins are used to uniquely address individual devices when multiple Multi-Queue devices are connected in expansion mode. Addressing devices in expansion mode requires matching WRADD/ RDADD address bits with the address that is assigned to each device by the ID[2:0] pins. During write/ read operations the WRADD/RDADD address are compared to the device ID [2:0] value. Note: expansion mode supports a maximum 256 queues, regardless of the number of devices used in expansion mode. The first device in a chain of multi-queue's (connected in expansion mode), may be setup as '000', the second as '001". In single device mode the ID[2:0] pins should be setup as '0xx' and the MSb (bit 7) of the WRADD and RDADD address busses should be zero. The ID[2:0] inputs setup a respective device ID during Master Reset. These ID pins must not toggle during any device operation. Note, the device selected as the 'Master' does not have to have the ID of '000'.
MAST (U1)	Master Device	1.8V LVTTL INPUT	The state of this input at Master Reset determines whether a given device (within a chain of devices), is the Master device or a Slave. If this pin is HIGH, the device is the master if it is LOW then it is a Slave. The master device is the first to take control of all outputs after a Master Reset, all slave devices go to High-Impedance, preventing bus contention. If a multi-queue device is being used in single device mode, this pin must be set HIGH.
$\overline{\text{MRS}}$ (T3)	Master Reset	HSTL INPUT	The Master Reset is used to configure the device. To configure the device configuration signals must be asserted that meet the setup time and hold time requirements of a Master Reset cycle. Transitioning $\overline{\text{MRS}}$ from HIGH to LOW then LOW to HIGH performs a complete Master Reset cycle. Note, additional device programming is required after master reset.

PIN DESCRIPTIONS (CONTINUED)

Symbol & Pin No.	Name	I/O TYPE	Description
$\overline{OE}$ (R22)	Output Enable	HSTL INPUT	The Output Enable signal is the three-state control of the multi-queue data output bus Q[39:0], Qout. If a device has been configured as a "Master" device, the Qout data outputs will be in a low impedance condition if the $\overline{OE}$ input is LOW. If $\overline{OE}$ is HIGH then the Qout data outputs will be in high impedance. If a device is configured a "Slave" device, then the Qout data outputs will always be in high impedance until that device has been selected on the Read Port, at which point $\overline{OE}$ provides three-state of that respective device.
$\overline{PAE}$ (N20)	Programmable Almost-Empty Flag	HSTL OUTPUT	This pin provides the Almost-Empty flag status for the Queue that has been selected on the output port for read operations, (selected via RCLK, RDADD and RADEN). This pin is LOW when the selected Queue is almost-empty. This flag output may be duplicated on one of the $\overline{PAE}$n bus lines. This flag is synchronized to RCLK.
$\overline{PAE}$n[7:0] ($\overline{PAE}$7-F21 $\overline{PAE}$6-F22 $\overline{PAE}$5-G20 $\overline{PAE}$4-G21 $\overline{PAE}$3-G22 $\overline{PAE}$2-H20 $\overline{PAE}$1-H21 $\overline{PAE}$0-H22)	Programmable Flag Bus	HSTL OUTPUT	The $\overline{PAE}$n bus is 8 bits wide. During a Master Reset this bus is setup for Almost Empty configuration. This output bus provides $\overline{PAE}$ status of 8 queues (1 quadrant), within a selected device. During Queue read/ write operations these outputs provide programmable empty flag status or packet data available status, in either polled or direct mode. The mode of flag operation is determined during master reset via the state of the FM input. This flag bus is capable of High-Impedance state, this is important during expansion of multi-queue devices. During direct operation the $\overline{PAE}$n bus is updated to show the $\overline{PAE}$ status of a quadrant of queues within a selected device. Selection is made using RCLK, ESTR and RDADD. During Polled operation the $\overline{PAE}$n bus is loaded with the $\overline{PAE}$ status of multi-queue flow-control quadrants sequentially based on the rising edge of RCLK.
$\overline{PAF}$ (F3)	Programmable Almost–Full Flag	HSTL OUTPUT	This pin provides the Almost-Full flag status for the Queue that has been selected on the input port for write operations, (selected via WCLK, WRADD and WADEN). This pin is LOW when the selected Queue is almost-full. This flag output may be duplicated on one of the $\overline{PAF}$n bus lines. The $\overline{PAE}$ flag is asserted synchronous to WCLK.
$\overline{PAF}$n[7:0] ($\overline{PAF}$7-F2 $\overline{PAF}$6-F1 $\overline{PAF}$5-G3 $\overline{PAF}$4-G2 $\overline{PAF}$3-G1 $\overline{PAF}$2-H3 $\overline{PAF}$1-H2 $\overline{PAF}$0-H1)	Programmable Almost-Full Flag Bus	HSTL OUTPUT	The $\overline{PAF}$n bus is 8 bits wide. At any one time this output bus provides $\overline{PAF}$ status of 8 queues (1 quadrant), within a selected device. During Queue read/write operations these outputs provide programmable full flag status, in either direct or polled mode. The mode of flag operation is determined during master reset via the state of the FM input. This flag bus is capable of High-Impedance state, this is important during expansion of multi-queue devices. During direct operation the $\overline{PAF}$n bus is updated to show the $\overline{PAF}$ status of a quadrant of queues within a selected device. Selection is made using WCLK, FSTR, WRADD and WADEN. During Polled operation the $\overline{PAF}$n bus is loaded with the $\overline{PAF}$ status of multi-queue flow-control quadrants sequentially based on the rising edge of WCLK.
PLL ON (V1)	PLL ON	HSTL INPUT	This pin is used to enable the PLL. When PLL is activated, data will be clocked out by PLL generated clock.
Q[39:0](Qout) (See Pin No. table for details)	Data Output Bus	HSTL OUTPUT	These are the 40 data output pins. Data is read out of the device via these output pins on the rising edge of RCLK provided that $\overline{REN}$ is LOW, $\overline{OE}$ is LOW and the Queue is selected. Due to bus-matching not all outputs may be used, any unused outputs should not be connected.
QSEL[2:0] (QSEL2-BB10 QSEL1-AA10 QSEL0-BB9)	Queue Select	1.8V LVTTL INPUT	The QSEL pins provides various queue programming options. Refer to Table 10, Write Queue Switch Operation for details.
RADEN (T22)	Read Address Enable	HSTL INPUT	The RADEN input is used in conjunction with RCLK and the RDADD address bus to select a queue to be read from. A queue addressed via the RDADD bus is selected on the rising edge of RCLK provided that RADEN is HIGH. RADEN should be asserted (HIGH) only during a queue change cycle(s). RADEN should not be permanently tied HIGH. RADEN cannot be HIGH for the same RCLK cycle as ESTR. Note, that a read queue selection cannot be made, (RADEN must NOT go active) until programming of the part has been completed and SENO has gone LOW.
RCLK (J22)	Read Clock	HSTL INPUT	When enabled by $\overline{REN}$, the rising edge of RCLK reads data from the selected queue via the output bus Qout. The queue to be read is selected via the RDADD address bus and a rising edge of RCLK while RADEN is HIGH. A rising edge of RCLK in conjunction with ESTR and RDADD will also select the $\overline{PAE}$n flag quadrant to be placed on the $\overline{PAE}$n bus during direct flag operation. During polled flag operation the $\overline{PAE}$n bus is cycled with respect to RCLK and the ESYNC signal is synchronized to RCLK. The $\overline{PAE}$, and

PIN DESCRIPTIONS (CONTINUED)

Symbol & Pin No.	Name	I/O TYPE	Description
RCLK (Cont'd) (J22)	Read Clock	HSTL INPUT	$\overline{EF}$ outputs are all synchronized to RCLK. During device expansion the EXO and EXI signals are based on RCLK. RCLK must be continuous and free-running.
$\overline{RCS}$ (J21)	Read Chip Select	HSTL INPUT	The $\overline{RCS}$ signal in concert with $\overline{REN}$ signal provides control to enable data on to the output read data bus. During a Master Reset cycle the $\overline{RCS}$ it is don't care signal.
RDADD[7:0] (RDADD7-V20 RDADD6-V21 RDADD5-V22 RDADD4-U20 RDADD3-U21 RDADD2-U22 RDADD1-T20 RDADD0-T21)	Read Address Bus	HSTL INPUT	For the 128Q device the RDADD bus is 8 bits. The RDADD bus is a dual purpose address bus. The first function of RDADD is to select a Queue to be read from. The least significant 5 bits of the bus, RDADD[4:0]are used to address 1 of 128 possible queues within a multi-queue device. The most significant 3 bits, RDADD[7:5] are used to select 1 of 8 possible multi-queue devices that may be connected in expansion mode. These 3 MSB's will address a device with the matching ID code. (See ID[2:0] description for more detail on matching ID code. The second function of the RDADD bus is to select the quadrant of queues to be loaded on to the $\overline{PAE}$n bus during strobed flag mode. The least significant 4 bits, RDADD[3:0] are used to select the quadrant of a device to be placed on the $\overline{PAE}$n bus. The most significant 3 bits, RDADD[7:5] are again used to select 1 of 8 possible multi-queue devices that may be connected in expansion mode. Address bit RDADD[4] is don't care during quadrant selection.
$\overline{REN}$ (J20)	Read Enable	HSTL INPUT	The $\overline{REN}$ input enables read operations from a selected Queue based on a rising edge of RCLK. A queue to be read from can be selected via RCLK, RADEN and the RDADD address bus regardless of the state of $\overline{REN}$. Data from a newly selected queue will be available on the Qout output bus on the second RCLK cycle after queue selection regardless of $\overline{REN}$. A read enable is not required to cycle the $\overline{PAE}$n bus (in polled mode) or to select the $\overline{PAE}$n quadrant , (in direct mode).
SCLK (K3)	Serial Clock	HSTL INPUT	If serial programming of the multi-queue device has been selected during master reset, the SCLK input clocks the serial data through the multi-queue device. Data setup on the SI input is loaded into the device on the rising edge of SCLK provided that SENI is enabled, LOW. When expansion of devices is performed the SCLK of all devices should be connected to the same source.
SENI (K1)	Serial Input Enable	HSTL INPUT	During serial programming of a multi-queue device, data loaded onto the SI input will be clocked into the part (via a rising edge of SCLK), provided the SENI input of that device is LOW. If multiple devices are cascaded, the SENI input should be connected to the SENO output of the previous device. So when serial loading of a given device is complete, its SENO output goes LOW, allowing the next device in the chain to be programmed (SENO will follow SENI of a given device once that device is programmed). The SENI input of the master device (or single device), should be controlled by the user.
SENO (K2)	Serial Output Enable	HSTL OUTPUT	This output is used to indicate that serial programming or default programming of the multi-queue device has been completed. SENO follows SENI once programming of a device is complete. Therefore, SENO will go LOW after programming provided SENI is LOW, once SENI is taken HIGH again, SENO will also go HIGH. When the SENO output goes LOW, the device is ready to begin normal read/write operations. If multiple devices are cascaded and serial programming of the devices will be used, the SENO output should be connected to the SENI input of the next device in the chain. When serial programming of the first device is complete, SENO will go LOW, thereby taking the SENI input of the next device LOW and so on throughout the chain. When a given device in the chain is fully programmed the SENO output essentially follows the SENI input. The user should monitor the SENO output of the final device in the chain. When this output goes LOW, serial loading of all devices has been completed.
SI (L2)	Serial In	HSTL INPUT	During serial programming this pin is loaded with the serial data that will configure the multi-queue devices. Data present on SI will be loaded on a rising edge of SCLK provided that SENI is LOW. In expansion mode the serial data input is loaded into the first device in a chain. When that device is loaded and its SENO has gone LOW, the data present on SI will be directly output to the SO output. The SO pin of the first device connects to the SI pin of the second and so on. The multi-queue device setup registers are shift registers.
SO (L3)	Serial Out	HSTL OUTPUT	This output is used in expansion mode and allows serial data to be passed through devices in the chain to complete programming of all devices. The SI of a device connects to SO of the previous device in the chain. The SO of the final device in a chain should not be connected.
TCK (B10)	JTAG Clock	HSTL INPUT	Clock input for JTAG function. One of four terminals required by IEEE Standard 1149.1-1990. Test operations of the device are synchronous to TCK. Data from TMS and TDI are sampled on the rising edge of TCK and outputs change on the falling edge of TCK. If the JTAG function is not used this signal needs to be tied to GND.

PIN DESCRIPTIONS (CONTINUED)

Symbol & (Pin No.)	Name	I/O TYPE	Description
TDI (A11)	JTAG Test Data	HSTL INPUT	One of four terminals required by IEEE Standard 1149.1-1990. During the JTAG boundary scan operation, Input test data serially loaded via the TDI on the rising edge of TCK to either the Instruction Register, ID Register and Bypass Register. An internal pull-up resistor forces TDI HIGH if left unconnected
TDO (B11)	JTAG Test Data Output	HSTL	One of four terminals required by IEEE Standard 1149.1-1990. During the JTAG boundary scan operation, test data serially loaded output via the TDO on the falling edge of TCK from either the Instruction Register, ID Register and Bypass Register. This output is high impedance except when shifting, while in SHIFT-DR and SHIFT-IR controller states.
TP (U3)	IDT Internal Test Pin	LVTTL	For IDT internal test purpose only, must be tied to GND for normal/correct operation.
TMS (A10)	JTAG Mode Select	HSTL INPUT	TMS is a serial input pin. One of four terminals required by IEEE Standard 1149.1-1990. TMS directs the device through its TAP controller states. An internal pull-up resistor forces TMS HIGH if left unconnected.
$\overline{TRST}$ (A9)	JTAG Reset	HSTL INPUT	$\overline{TRST}$ is an asynchronous reset pin for the JTAG controller. The JTAG TAP controller does not automatically reset upon power-up, thus it must be reset by either this signal or by setting TMS= HIGH for five TCK cycles. If the TAP controller is not properly reset then the outputs will always be in high-impedance. If the JTAG function is used but the user does not want to use $\overline{TRST}$, then $\overline{TRST}$ can be tied with $\overline{MRS}$ to ensure proper queue operation. If the JTAG function is not used then this signal needs to be tied to GND. An internal pull-up resistor forces $\overline{TRST}$ HIGH if left unconnected.
WADEN (M3)	Write Address Enable	HSTL INPUT	The WADEN input is used in conjunction with WCLK and the WRADD address bus to select a queue to be written in to. A queue addressed via the WRADD bus is selected on the rising edge of WCLK provided that WADEN is HIGH. WADEN should be asserted (HIGH) only during a queue change cycle(s). WADEN should not be permanently tied HIGH. WADEN cannot be HIGH for the same WCLK cycle as FSTR. Note, that a write queue selection cannot be made, (WADEN must NOT go active) until programming of the part has been completed and SENO has gone LOW.
WCLK (M2)	Write Clock	HSTL INPUT	When enabled by $\overline{WEN}$, the rising edge of WCLK writes data into the selected Queue via the input bus, Din. The Queue to be written to is selected via the WRADD address bus and a rising edge of WCLK while WADEN is HIGH. A rising edge of WCLK in conjunction with FSTR and WRADD will also select the flag quadrant to be placed on the $\overline{PAF}$n bus during direct flag operation. During polled flag operation the $\overline{PAF}$n bus is cycled with respect to WCLK and the FSYNC signal is synchronized to WCLK. The $\overline{PAF}$n, $\overline{PAF}$ and $\overline{FF}$ outputs are all synchronized to WCLK. During device expansion the FXO and FXI signals are based on WCLK. The WCLK must be continuous and free-running
$\overline{WCS}$ (L1)	Write Chip Select	HSTL INPUT	The $\overline{WCS}$ signal in concert with $\overline{WEN}$ signal provides control to enable data from the input write data bus to be written into the device. During a Master Reset cycle the $\overline{WCS}$ it is don't care signal.
$\overline{WEN}$ (M1)	Write Enable	HSTL INPUT	The $\overline{WEN}$ input enables write operations to a selected Queue based on a rising edge of WCLK. A queue to be written to can be selected via WCLK, WADEN and the WRADD address bus regardless of the state of $\overline{WEN}$. Data present on Din can be written to a newly selected queue on the second WCLK cycle after queue selection provided that $\overline{WEN}$ is LOW. A write enable is not required to cycle the $\overline{PAF}$n bus (in polled mode) or to select the $\overline{PAF}$n quadrant , (in direct mode).
WRADD[7:0] (WRADD7-R2 WRADD6-R1 WRADD5-P3 WRADD4-P2 WRADD3-P1 WRADD2-N3 WRADD1-N2 WRADD0-N1)	Write Address Bus	HSTL INPUT	The WRADD bus is 8 bits. The WRADD bus is a dual purpose address bus. The first function of WRADD is to select a Queue to be written to. The least significant 5 bits of the bus, WRADD[4:0] are used to address 1 of 128 possible queues within a multi-queue device. The most significant 3 bits, WRADD[7:5] are used to select 1 of 8 possible multi-queue devices that may be connected in expansion mode. These 3 MSB's will address a device with the matching ID code. (See ID[2:0] description for more detail on matching ID code. The second function of the WRADD bus is to select the quadrant of queues to be loaded on to the $\overline{PAF}$n bus during strobed flag mode. The least significant 4 bits, WRADD[3:0] are used to select the quadrant of a device to be placed on the $\overline{PAF}$n bus. The most significant 3 bits, WRADD[7:5] are again used to select 1 of 8 possible multi-queue devices that may be connected in expansion mode. Address bit WRADD[4] is don't care during quadrant selection.
ZQ (N22)	ZQ	HSTL INPUT	Output Impedance Matching Input. This input is used to tune the device outputs to the system data bus impedance. Q[39:0] output impedance is set to 0.2 x RQ, where RQ is a resistor connected between ZQ and ground. This pin cannot be connected directly to GND or left unconnected.

PIN DESCRIPTIONS (CONTINUED)

Symbol & Pin No.	Name	I/O TYPE	Description
VDD (See below)	+1.8V Supply	Power	These are VCC power supply pins and must all be connected to a +1.8V supply
VDDQ (See below)	Output Voltage	Power	These pins must be tied to the desired output supply voltage (=1.5V for HSTL and =1.8V for eHSTL).
Vref (J1, M21)	Reference	INPUT	This is a Voltage Reference input and must be connected to a voltage level determined from the table Voltage "Recommended DC Operating Conditions". The input provides the reference level for HSTL/ eHSTL inputs.
GND (See below)	Ground	Ground	These are Ground pins and must all be connected to the Ground of the power supply.
AVDD (K21, M22)	PLL Power	Power	1.8V PLL Power Supply.
AVSS (L(21,22), K22)	PLL Ground	Ground	Ground for the PLL device. Should be connected to ground of the system.

NOTE:
1. Inputs should not change after Master Reset.

PIN NUMBER TABLE

Symbol	Name	I/O TYPE	Pin Number
D[39:0]	Data Input Bus	HSTL-LVTTL INPUT	D39-AA9, D38-BB8, D37-AA8, D36-BB7, D35-AA7, D34-BB6, D33-AA6, D32-BB5, D31-AA5, D30-BB4, D29-AA4, D28-BB3, D27-AA3, D(26,25)-Y(3,4), D(24-22)-W(3-1), D(21,20)-V(3,2), D(19,18)-E(2,3), D(17-15)-D(1-3), D(14,13)-C(4,3), D12-B3, D11-A3, D10-B4, D9-A4, D8-B5, D7-A5, D6-B6, D5-A6, D4-B7, D3-A7, D2-B8, D1-A8, D0-B9
Q[39:0]	Data Output Bus	HSTL-LVTTL	Q39-AA13, Q38-BB14, Q37-AA14, Q36-BB15, Q35-AA15, Q34-AA15, Q33-AA16, Q32-BB17, Q31-AA17, Q30-BB18, Q29-AA18, Q28-BB19, Q27-AA19, Q26-BB20, Q25-AA20, Q(24,23)-Y(20,19), Q(22-20)-W(20-22), Q(19-17)-D(22-20), Q(16,15)-C(19,20), Q14-B20, Q13-A20, Q12-B19, Q11-A19, Q10-B18, Q9-A18, Q8-B17, Q7-A17, Q6-B16, Q5-A16, Q4-B15, Q3-A15, Q2-B14, Q1-A14, Q0-B13
VDD	+1.8V Supply	Power	D(4-10,13-19), E(4-9,14-17,19), G18, H(4,5,18,19), J(4,19), P(4,19), R(3-5,18,19), V(4-8,15-19), W(4-10,13-19)
VDDQ	O/P Rail Voltage	Power	C(5-10,13-18) E18, F(4,5,18,19), G(4,5,19), T(4,5,18,19), U(4,5,18,19), Y(5-10,14-18)
GND	Ground Pin	Ground	A(1,2,21,22), B(1,2,21,22), C(1,2,11,12,21,22), D(11,12), E(10-13), J(5,9-14,18), K(4,5,9-14,18-20), L(4,5,9-14,18-20), M(4,5,9-14,18-20), N(4,5,9-14,18,19), P(5,9-14,18), V(9-14), W(11,12), Y(1,2,11-13,21,22), AA(1,2,21,22), BB(1,2,21,22)

ABSOLUTE MAXIMUM RATINGS

Symbol	Rating	Commercial	Unit
VTERM	Terminal Voltage with respect to GND	–0.5 to +2.9(2)	V
TSTG	Storage Temperature	–55 to +125	°C
IOUT	DC Output Current	–50 to +50	mA

NOTES:

1. Stresses greater than those listed under ABSOLUTE MAXIMUM RATINGS may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect reliability.
2. Compliant with JEDEC JESD8-5. VDD terminal only.

CAPACITANCE (TA = +25°C, f = 1.0MHz)

Symbol	Parameter(1)	Conditions	Max.	Unit
CIN(2,3)	Input Capacitance	VIN = 0V	10(3)	pF
COUT(1,2)	Output Capacitance	VOUT = 0V	15	pF

NOTES:

1. With output deselected, ($\overline{OE}$ ≥ VIH).
2. Characterized values, not currently tested.
3. CIN for Vref is 20pF.

RECOMMENDED DC OPERATING CONDITIONS

Symbol	Parameter		Min.	Typ.	Max.	Unit
VDD	Supply Voltage		1.7	1.8	1.9	V
VDDQ	Output Rail Voltage for I/Os	— eHSTL — HSTL	1.7 1.4	1.8 1.5	1.9 1.6	V V
GND	Supply Voltage		0	0	0	V
VIH(2)	Input High Voltage	— eHSTL — HSTL	VREF+0.2 VREF+0.2	— —	— —	V V
VIL	Input Low Voltage	— eHSTL — HSTL	— —	— —	VREF-0.2 VREF-0.2	V V
VREF(1) (HSTL only)	Voltage Reference Input	— eHSTL — HSTL	0.8 0.68	0.9 0.75	1.0 0.9	V V
TA	Operating Temperature Commercial		0	—	70	°C
TA	Operating Temperature Industrial		-40	—	85	°C

NOTES:

1. VREF is only required for HSTL or eHSTL inputs.
2. VIH AC Component = VREF + 0.4V

DC ELECTRICAL CHARACTERISTICS

(Commercial: VDD = 1.8V ± 0.10V, TA = 0°C to +70°C;Industrial: VDD = 1.8V ± 0.10V, TA = -40°C to +85°C)

Symbol	Parameter	Min.	Max.	Unit
ILI	Input Leakage Current	–10	10	µA
ILO	Output Leakage Current	–10	10	µA
VOH1(7)	Output High Voltage (test conditions: RQ = 205Ω IOH = -8mA)	VDDQ/2 -0.12	VDDQ/2 +0.12	V
VOL1(8)	Output Low Voltage (test conditions: RQ = 205Ω IOL = 8mA)	VDDQ/2 -0.12	VDDQ/2 +0.12	V
VOH2(9)	Output High Voltage (test conditions: IOH = -0.1mA)	VDDQ -0.12	VDDQ	V
VOL2(10)	Output Low Voltage (test conditions: IOL = 0.1mA)	VSS	0.2	V
IDD1(1,2)	Active VDD Current (VDD = 1.8V) I/O = HSTL I/O = eHSTL	— —	200 200	mA mA
IDD2(1, 5)	Standby VDD Current (VDD = 1.8V) I/O = HSTL I/O = eHSTL	— —	120 120	mA mA
IDDQ(1,2)	Active VDDQ Current (VDDQ = 1.5V HSTL) I/O = HSTL (VDDQ = 1.8V eHSTL) I/O = eHSTL	— —	20 20	mA mA

NOTES:

1. Both WCLK and RCLK toggling at 20MHz.
2. Data inputs toggling at 10MHz.
3. Total Power consumed: PT = [(VDD x IDD) + (VDDQ x IDDQ)].
4. Outputs are not 2.5V or 3.3V tolerant.
5. The following inputs should be pulled to GND: WRADD, RDADD, WADEN, FSTR, ESTR, SCLK, SI, EXI, FXI and all Data Inputs.
   The following inputs should be pulled to VDD: $\overline{\text{WEN}}$, $\overline{\text{REN}}$, $\overline{\text{SENI}}$, $\overline{\text{MRS}}$, TDI, TMS and $\overline{\text{TRST}}$.
   All other inputs are don't care and should be at a known state.
6. The ZQ pin is used to control the device outputs (Q[39:0], $\overline{\text{EREN}}$, ERCLK, and $\overline{\text{ERCLK}}$).
7. Outputs are impedance-controlled. IOH= -(VDDQ/2)/(RQ/5) and is guaranteed by device characterization for 175Ω< RQ < 350Ω. This parameter is tested at RQ = 250Ω which gives a nominal 50Ω output impedance.
8. Outputs are impedance-controlled. IOL = (VDDQ/2)/(RQ/5) and is guaranteed by device characterization for 175Ω< RQ < 350Ω This parameter is tested at RQ = 250Ω which gives a nominal 50Ω output impedance.
9. This measurement is taken to ensure that the output has the capability of pulling to the VDDQ rail, and is not intended to be used as an impedance measurement point.
10. This measurement is taken to ensure that the output has the capability of pulling to VSS, and is not intended to be used as an impedance measurement point.

HSTL
1.5V AC TEST CONDITIONS

Input Pulse Levels	0.25 to 1.25V
Input Rise/Fall Times	0.4ns
Input Timing Reference Levels	0.75
Output Reference Levels	VDDQ/2

NOTE:
1. VDDQ = 1.5V ± 0.1V.

EXTENDED HSTL
1.8V AC TEST CONDITIONS

Input Pulse Levels	0.4 to 1.4V
Input Rise/Fall Times	0.4ns
Input Timing Reference Levels	0.9
Output Reference Levels	VDDQ/2

NOTE:
1. VDDQ = 1.8V ± 0.1V.

AC TEST LOADS

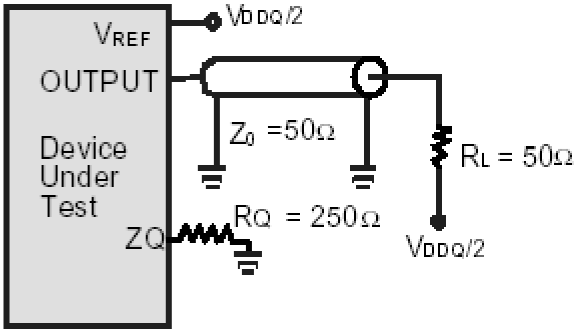


Figure 2a. AC Test Load

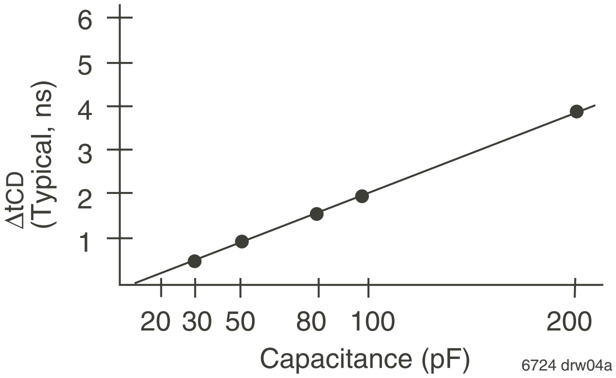
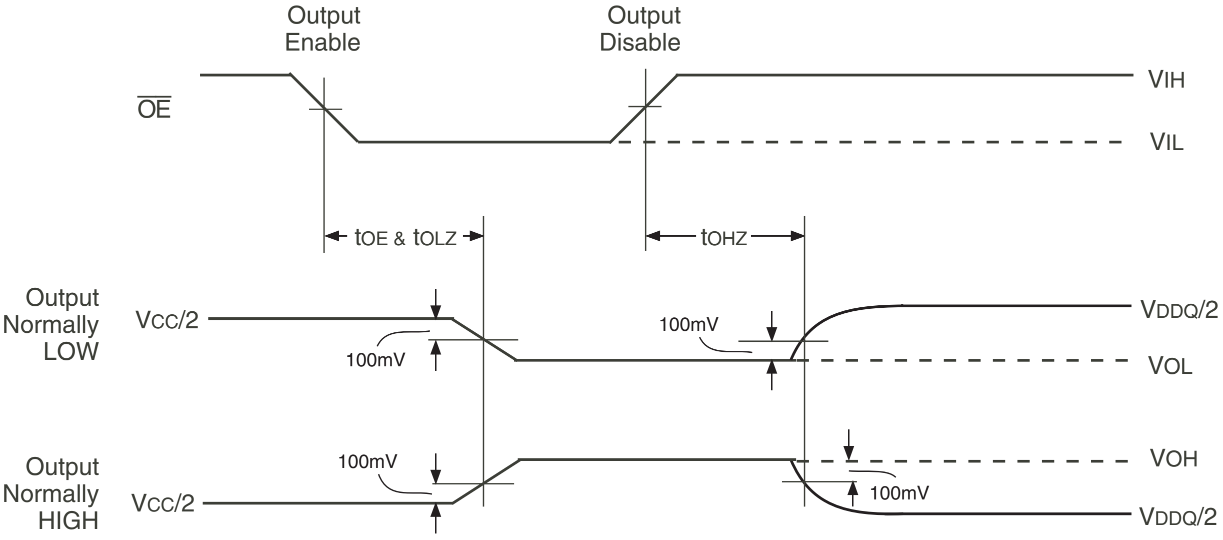


Figure 2b. Lumped Capacitive Load, Typical Derating

OUTPUT ENABLE & DISABLE TIMING



NOTE:
1. REN is HIGH.

AC ELECTRICAL CHARACTERISTICS

(Commercial: V_{DD} = 1.8V ± 0.10V, T_A = 0°C to +70°C;Industrial: V_{DD} = 1.8V ± 0.10V, T_A = -40°C to +85°C; JEDEC JESD8-A compliant)

		Commercial		Com'l and Ind'l		
		IDT72P51767L6 IDT72P51777L6		IDT72P51767L7-5 IDT72P51777L7-5		
Symbol	Parameter	Min.	Max.	Min.	Max.	Unit
f_C	Clock Cycle Frequency	—	166	—	133	MHz
t_A (PLL ON)	Data Access Time	-1.0	1.0	-1.2	1.2	ns
t_A (PLL OFF)	Data Access Time	0.6	3.6	0.8	3.8	ns
t_{CLK}	Clock Cycle Time	6.0	—	7.5	—	ns
t_{CLKH}	Clock High Time	2.8	—	3.0	—	ns
t_{CLKL}	Clock Low Time	2.8	—	3.0	—	ns
t_{DS}	Data Setup Time	0.48	—	0.7	—	ns
t_{DH}	Data Hold Time	0.48	—	0.7	—	ns
t_{ENS}	Enable Setup Time	2.0	—	2.2	—	ns
t_{ENH}	Enable Hold Time	0.5	—	0.7	—	ns
t_{RS}	Reset Pulse Width[3]	30	—	30	—	ns
t_{RSS}	Reset Setup Time	15	—	15	—	ns
t_{RSR}	Reset Recovery Time	10	—	10	—	ns
t_{OHZ}	Output Enable to Output in High Z	0.6	3.6	0.8	3.8	ns
t_{OE}	Output Enable to $\overline{OE}$	0.6	3.6	0.8	3.8	ns
f_S	Clock Cycle (SCLK)	—	10	—	10	MHz
t_{SCLK}	Serial Clock Cycle	100	—	100	—	ns
t_{SCKH}	Serial Clock High	45	—	45	—	ns
t_{SCKL}	Serial Clock Low	45	—	45	—	ns
t_{SDS}	Serial Data in Setup	20	—	20	—	ns
t_{SDH}	Serial Data in Hold	0.8	—	0.8	—	ns
t_{SENS}	Serial Enable Setup	20	—	20	—	ns
t_{SENH}	Serial Enable Hold	0.8	—	0.8	—	ns
t_{SDO}	SCLK to Serial Data Out	—	20	—	20	ns
t_{SENO}	SCLK to Enable Out	—	20	—	20	ns
t_{SDOP}	Serial Data Out Delay	0.8	3.6	0.8	3.6	ns
t_{SENOP}	Serial Enable Delay	0.8	3.6	0.8	3.6	ns
t_{PCWQ}	Programming to Write Queue Selection	—	7	—	7	cycles
t_{PCRQ}	Programming to Read Queue Selection	—	7	—	7	cycles
t_{AS}	Address Setup	2.0	—	2.2	—	ns
t_{AH}	Address Hold	0.5	—	0.7	—	ns
t_{WFF}	Write Clock to Full Flag ($\overline{FF}$)	-—	3.6	—	3.8	ns
t_{REF}	Read Clock to Empty Flag ($\overline{EF}$)	-—	3.6	—	3.8	ns
t_{STS}	Strobe Setup	2.0	—	2.2	—	ns
t_{STH}	Strobe Hold	0.5	—	0.7	—	ns
t_{QS}	Queue Setup	2.0	—	2.2	—	ns
t_{QH}	Queue Hold	0.5	—	0.7	—	ns
t_{WAF}	WCLK to $\overline{PAF}$ flag	0.6	3.6	0.8	3.8	ns
t_{RAE}	RCLK to $\overline{PAE}$ flag	0.6	3.6	0.8	3.8	ns
t_{PAF}	WCLK to Sync $\overline{PAF}$ bus	0.6	3.6	0.8	3.8	ns
t_{PAE}	RCLK to Sync $\overline{PAE}$ bus	0.6	3.6	0.8	3.8	ns
t_{PAELZ}	RCLK to Low-Z	0.6	3.6	0.8	3.8	ns
t_{PAEHZ}	RCLK to High-Z	0.6	3.6	0.8	3.8	ns
t_{PAFLZ}	WCLK to $\overline{PAF}$ Bus Low-Z	0.6	3.6	0.8	3.8	ns

NOTES:
1. Industrial temperature range product for the 7-5ns is available as a standard device. All other speed grades available by special order.
2. Values guaranteed by design, not currently tested.

AC ELECTRICAL CHARACTERISTICS (CONTINUED)

(Commercial: V_{DD} = 1.8V ± 0.10V, T_A = 0°C to +70°C;Industrial: V_{DD} = 1.8V ± 0.10V, T_A = -40°C to +85°C; JEDEC JESD8-A compliant)

		Commercial		Com'l and Ind'l		
		IDT72P51767L6 IDT72P51777L6		IDT72P51767L7-5 IDT72P51777L7-5		
Symbol	Parameter	Min.	Max.	Min.	Max.	Unit
t_{PAFHZ}	WCLK to $\overline{\text{PAF}}$ Bus High-Z	0.6	3.6	0.8	3.8	ns
t_{FFHZ}	WCLK to $\overline{\text{FF}}$ High-Z	0.6	3.6	0.8	3.8	ns
t_{FFLZ}	WCLK to $\overline{\text{FF}}$ Low-Z	0.6	3.6	0.8	3.8	ns
t_{EFHZ}	RCLK to $\overline{\text{EF}}$ High-Z	0.6	3.6	0.8	3.8	ns
t_{EFLZ}	RCLK to $\overline{\text{EF}}$ Low-Z	0.6	3.6	0.8	3.8	ns
t_{FSYNC}	WCLK to $\overline{\text{PAF}}$ Bus Sync	0.6	3.6	0.8	3.8	ns
t_{FXO}	WCLK to $\overline{\text{PAF}}$ Bus Exp	0.6	3.6	0.8	3.8	ns
t_{ESYNC}	RCLK to $\overline{\text{PAF}}$ Bus Sync	0.6	3.6	0.8	3.8	ns
t_{EXO}	RCLK to $\overline{\text{PAF}}$ Bus Exp	0.6	3.6	0.8	3.8	ns
t_{ERCLK} (DDR)	RCLK to ERCLK (DDR)	—	2.5	—	3.2	ns
t_{ERCLK} (SDR)	RCLK to ERCLK (SDR)	—	4	—	5	ns
t_{SKEW1}	Skew time for $\overline{\text{EF}}$ and $\overline{\text{FF}}$	6.0	—	7.0	—	ns
t_{SKEW2}	Skew time for $\overline{\text{PAF}}$ and $\overline{\text{PAE}}$	6.0	—	7.0	—	ns
t_{SKEW3}	Skew time for $\overline{\text{PAF}}$/$\overline{\text{PAE}}$[0:7]	6.0	—	7.0	—	ns
t_{XIS}	Expansion Input Setup	2.0	—	2.2	—	ns
t_{XIH}	Expansion Input Hold	0.5	—	0.7	—	ns
t_{LOCK}	PLL Lock Time	—	15	—	15	µs

NOTES:

1. Industrial temperature range product for the 7-5ns is available as a standard device. All other speed grades available by special order.
2. Values guaranteed by design, not currently tested.

XGMII REFERENCE SPECIFICATION

The XGMII uses 1.5V High Speed Transceiver Logic (HSTL) signal levels.

The electrical characteristics of the XGMII are specified such that the XGMII can be applied within a variety of 10 Gb/s equipment types. The electrical specifications are selected for an integrated circuit to integrated circuit application. The electrical characteristics specified in this clause apply to all XGMII signals.

When implemented as a chip-to-chip interface, the XGMII uses High Speed Transceiver Logic (HSTL), specified for a 1.5 volt output buffer supply voltage. XGMII chip-to-chip signals shall comply with EIA/JEDEC Standard EIA/JESD8-6 using Class I, output buffers. Output impedance shall be greater than 38Ω to assure acceptable overshoot and undershoot performance in an un-terminated interconnection.

TABLE 2 — DC AND AC SPECIFICATIONS (INFORMATIVE)

Symbol	Parameter	Minimum	Nominal	Maximum	Units
VDDQ	Output Voltage Supply	1.4	1.5	1.6	V
VREF	Input Reference Voltage	0.68	0.75	0.90	V
VIH_DC	DC Input Logic High	VREF+0.10	-	VDDQ+0.3	V
VIL_DC	DC Input Logic Low	-0.30	-	VREF-0.1	V
VIH_AC	AC Input Logic High	VREF+0.20	-	-	V
VIL_AC	AC Input Logic Low	-	-	VREF-0.20	V

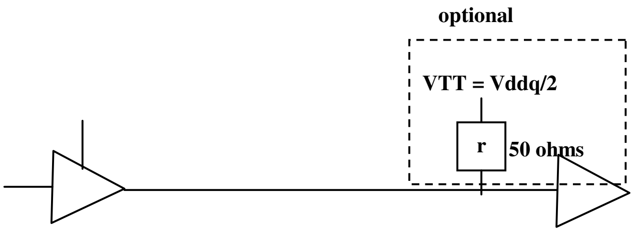


Figure 3. HSTL Termination for XGMII

TABLE 3 — IDT TO XGMII INTERFACE MAPPING SCHEMA

Signal Type	IDT Interface Signal Nomenclature	XGMII Signal Nomenclature
Input Port		
- Input Port Data	D[31:0] D[32:39]	TXD [31:0] User definable
- Input Port Enable	$\overline{\text{WEN}}$	N/A
- Input Port Control	N/A	TXC[3:0]
- Input Port Status	$\overline{\text{FF}}$, $\overline{\text{PAF}}$	N/A
- Input Port Clock	WCLK	TX_CLK
Output Port		
- Output Port Data	Q[31:0] Q[32:39]	RXD [31:0] User definable
- Output Port Enable	$\overline{\text{REN}}$	N/A
- Output Port Control	N/A	RXC [3:0}
- Output Port Status	$\overline{\text{EF}}$, $\overline{\text{PAE}}$	N/A
- Output Port Clock	RCLK	RX_CLK

FUNCTIONAL DESCRIPTION

MASTER RESET

A Master Reset is performed by toggling the $\overline{\text{MRS}}$ input from HIGH to LOW to HIGH. During a master reset all internal multi-queue device setup and control registers are initialized and require programming either serially by the user via the serial port, or via parallel programming or by using the default settings. Refer to Figure 6, *Device Programming Hierarchy* for the programming hierarchy structure. During a master reset the state of the following inputs determine the functionality of the part, these pins should be held HIGH or LOW.

FM – Flag bus Mode
BM [3:0] – Bus Matching options
MAST – Master Device
ID0, 1, 2 – Device ID
QSEL[2:0] Queue Select Mode
DFM – Programming mode, serial or default

Once a master reset has taken place, the device must be programmed either serially or via the default method before any read/write operations can begin.

See Figure 37, *Master Reset* for relevant timing.

PROGRAMMING MODE CAPTURED

On the rising of $\overline{\text{MRS}}$ the programming mode signals (QSEL [2:0], DFM) are captured. Once the programming mode signals are captured (latched), refer to Table 5, Setting the Queue Programming Mode during Master Reset for details. It will then require a number of clock cycles for the device to complete the configuration. Configuration completion is indicated when the $\overline{\text{SENO}}$ signal transitions from high to low. The configuration completion indication is consistent with the previous MQ device.

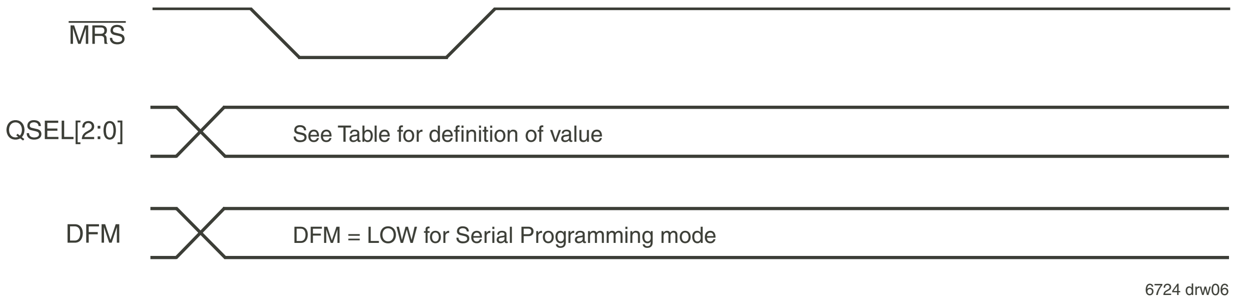

$\overline{\text{MRS}}$
QSEL[2:0]
See Table for definition of value
DFM
DFM = LOW for Serial Programming mode
6724 drw06

Figure 4. Reference Signals

TABLE 4 — DEVICE PROGRAMMING MODE COMPARISON

Programming Options	Serial Programming Mode	Default/Parallel Programming Mode
Queue Selection	Any number from 1 to 128	Any number from 1 to 128
Queue Depth	Each queue depth can be individualized	Default Value (total available memory divided by number of queues)
$\overline{\text{PAE}}$/$\overline{\text{PAF}}$	Programmable to any value	Default value
Bus-Matching	Any available option can be selected using BM[3:0] pins	Any available option can be selected using BM[3:0] pins
I/O voltage	Any available option can be selected	Any available option can be selected

TABLE 5 — SETTING THE QUEUE PROGRAMMING MODE DURING MASTER RESET

/MRS	Default Mode (DFM)	QSEL 2	QSEL 1	QSEL 0	Queue Programming Method
↑	0	0	0	0	Serial programming mode
↑	0	0	0	1	RESERVED
↑	0	0	1	0	RESERVED
↑	0	0	1	1	RESERVED
↑	0	1	0	0	RESERVED
↑	0	1	0	1	RESERVED
↑	0	1	1	0	RESERVED
↑	0	1	1	1	RESERVED
↑	1	0	0	0	Selects 128 Queues
↑	1	0	0	1	Selects 64 Queues
↑	1	0	1	0	Selects 32 Queues
↑	1	0	1	1	Selects 16 Queues
↑	1	1	0	0	Selects 8 Queues
↑	1	1	0	1	Selects 4 Queues
↑	1	1	1	0	Parallel programming enables the user to program the number of queues using the Write Address bus
↑	1	1	1	1	Parallel programming enables the user to program the number of queues using the Read Address bus

6724 drw07

SERIAL PROGRAMMING

The multi-queue flow-control device is a fully programmable device, providing the user with flexibility in how queues are configured in terms of the number of queues, depth of each queue and position of the $\overline{PAF}$/$\overline{PAE}$ flags within respective queues. All user programming is done via the serial port after a master reset has taken place. Internally the multi-queue device has setup registers which must be serially loaded, these registers contain values for every queue within the device, such as the depth and $\overline{PAE}$/$\overline{PAF}$ offset values. The IDT72P51767/72P51777 devices are capable of up to 128 queues and therefore contain 128 sets of registers for the setup of each queue.

During a Master Reset if the DFM (Default Mode) input is LOW, then the device will require serial programming by the user. It is recommended that the user utilize a 'C' program provided by IDT, this program will prompt the user for all information regarding the multi-queue setup. The program will then generate a serial bit stream which should be serially loaded into the device via the serial port. For the IDT72P51767/72P51777 devices the serial programming requires a total number of serially loaded bits per device, (SCLK cycles with $\overline{SENI}$ enabled), calculated by: 27+(Qx104) where Q is the number of queues the user wishes to setup within the device.

Once the master reset is complete and $\overline{MRS}$ is HIGH, the device can be serially loaded. Data present on the SI (serial in), input is loaded into the serial port on a rising edge of SCLK (serial clock), provided that $\overline{SENI}$ (serial in enable), is LOW. Once serial programming of the device has been successfully completed the device will indicate this via the $\overline{SENO}$ (serial output enable) going active, LOW. Upon detection of completion of programming, the user should cease all programming and take $\overline{SENI}$ inactive, HIGH. Note, $\overline{SENO}$ follows $\overline{SENI}$ once programming of a device is complete. Therefore, $\overline{SENO}$ will go LOW after programming provided $\overline{SENI}$ is LOW, once $\overline{SENI}$ is taken HIGH again, $\overline{SENO}$ will also go HIGH. The operation of the SO output is similar, when programming of a given device is complete, the SO output will follow the SI input.

If devices are being used in expansion configuration the serial ports of devices should be cascaded. The user can load all devices via the serial input port control pins, SI & $\overline{SENI}$, of the first device in the chain. Again, the user may utilize the 'C' program to generate the serial bit stream, the program prompting the user for the number of devices to be programmed. The $\overline{SENO}$ and SO (serial out) of the first device should be connected to the $\overline{SENI}$ and SI inputs of the second device respectively and so on, with the $\overline{SENO}$ & SO outputs connecting to the $\overline{SENI}$ & SI inputs of all devices through the chain. All devices in the chain should be connected to a common SCLK. The serial output port of the final device should be monitored by the user. When $\overline{SENO}$ of the final device goes LOW, this indicates that serial programming of all devices has been successfully completed. Upon detection of completion of programming, the user should cease all programming and take $\overline{SENI}$ of the first device in the chain inactive, HIGH.

As mentioned, the first device in the chain has its serial input port controlled by the user, this is the first device to have its internal registers serially loaded by the serial bit stream. When programming of this device is complete it will take its $\overline{SENO}$ output LOW and bypass the serial data loaded on the SI input to its SO output. The serial input of the second device in the chain is now loaded with the data from the SO of the first device, while the second device has its $\overline{SENI}$ input LOW. This process continues through the chain until all devices are programmed and the $\overline{SENO}$ of the final device (or master device, ID = '000') goes LOW.

Once all serial programming has been successfully completed, normal operations, (queue selections on the read and write ports) may begin. When connected in expansion configuration, the IDT72P51767/72P51777 devices require a total number of serially loaded bits per device to complete serial programming, (SCLK cycles with $\overline{SENI}$ enabled), calculated by: n[27+(Qx104)] where Q is the number of queues the user wishes to setup within the device, where n is the number of devices in the chain.

See Figure 40, *Serial Port Connection* and Figure 41, *Serial Programming* for connection and timing information.

The IDT72P51777/72P51767 device can be programmed using the serial input signals (SENI, SI, SCLK). Serial programming is accomplished by shifting in 26 bit words. It requires 1 Header Word to start the programming sequence and an additional 4 Programming Words for each queue that is configured within the device.

EACH OF THE 26 BIT WORDS ARE DESCRIBED BELOW:

Header Word: This is 1st 26-bit word and has the following bit assignments.

- Bits [25:7] is the Start of Header identifier.
- Bits [6:0] are the number of queues to be programmed.

The Start of Header identifier MUST be all ones (1's). The all 1's pattern in the Header word signifies the start of the programming cycle. The Header Word is only needed once for each device. For example, for 128 queues bits [6:0] = "1111111" for 32 queues bits [6:0] = "0011111".

Bits	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	Q64	Q32	Q16	Q8	Q4	Q2	Q1

$\overline{FF}$: This is the 2nd 26-bit word and represents the Full Flag programmed value. The Full Flag value is equal to the Queue depth-2. Each queue requires an individual $\overline{FF}$ value.

Bits	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Binary Value	0	0	0	0	0	0	524288	262144	131072	65536	32768	16384	8192	4096	2048	1024	512	256	128	64	32	16	8	4	2	1

$\overline{\text{PAE}}$: This is the 3rd 26-bit word and represents the Programmable Almost Empty ($\overline{\text{PAE}}$) value. Each queue requires an individual $\overline{\text{PAE}}$ value. The $\overline{\text{PAE}}$ value that is programmed into the device is the number of words. For example, for a $\overline{\text{PAE}}$ value = 52 words, bits [19:0] = "00000000000000110100".

Bits	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Binary Value	0	0	0	0	0	0	524288	262144	131072	65536	32768	16384	8192	4096	2048	1024	512	256	128	64	32	16	8	4	2	1

$\overline{\text{PAF}}$: This is the 4th 26-bit word and represents the Programmable Almost Full ($\overline{\text{PAF}}$) value. Each queue requires an individual $\overline{\text{PAF}}$ value. The $\overline{\text{PAF}}$ value that is programmed into the device is the "Queue Depth Value" – "The $\overline{\text{PAF}}$ Offset value". For example, with a Queue Depth of 16K (16384) and a desired $\overline{\text{PAF}}$ value = 39 words, bits [19:0] = "00000011111111011001".

Bits	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Binary Value	0	0	0	0	0	0	524288	262144	131072	65536	32768	16384	8192	4096	2048	1024	512	256	128	64	32	16	8	4	2	1

Queue End/Start Address

This is the 5th 26-bit word and represents both the start and end address of each queue.

End Address: The queue end address is bits [25:13] of the 26-bit. An end addresses is specified as; Queue Depth – 1k . Ending address are specified in increments of 1K words. For example, for a Queue Depth of 16K, the first queue would have a starting address of 0, bits [12:0] = "0000000000000" and an end address of 15K, bits [25:13] = "0000000001111",

Start Address: The queue starting address is bits [12:0] of the 26-bit word. Start addresses are specified in increments of words. The first queue should always start at address 0. The starting address of the next queue should be programmed at an address that is words greater the ending address of the previous queue.

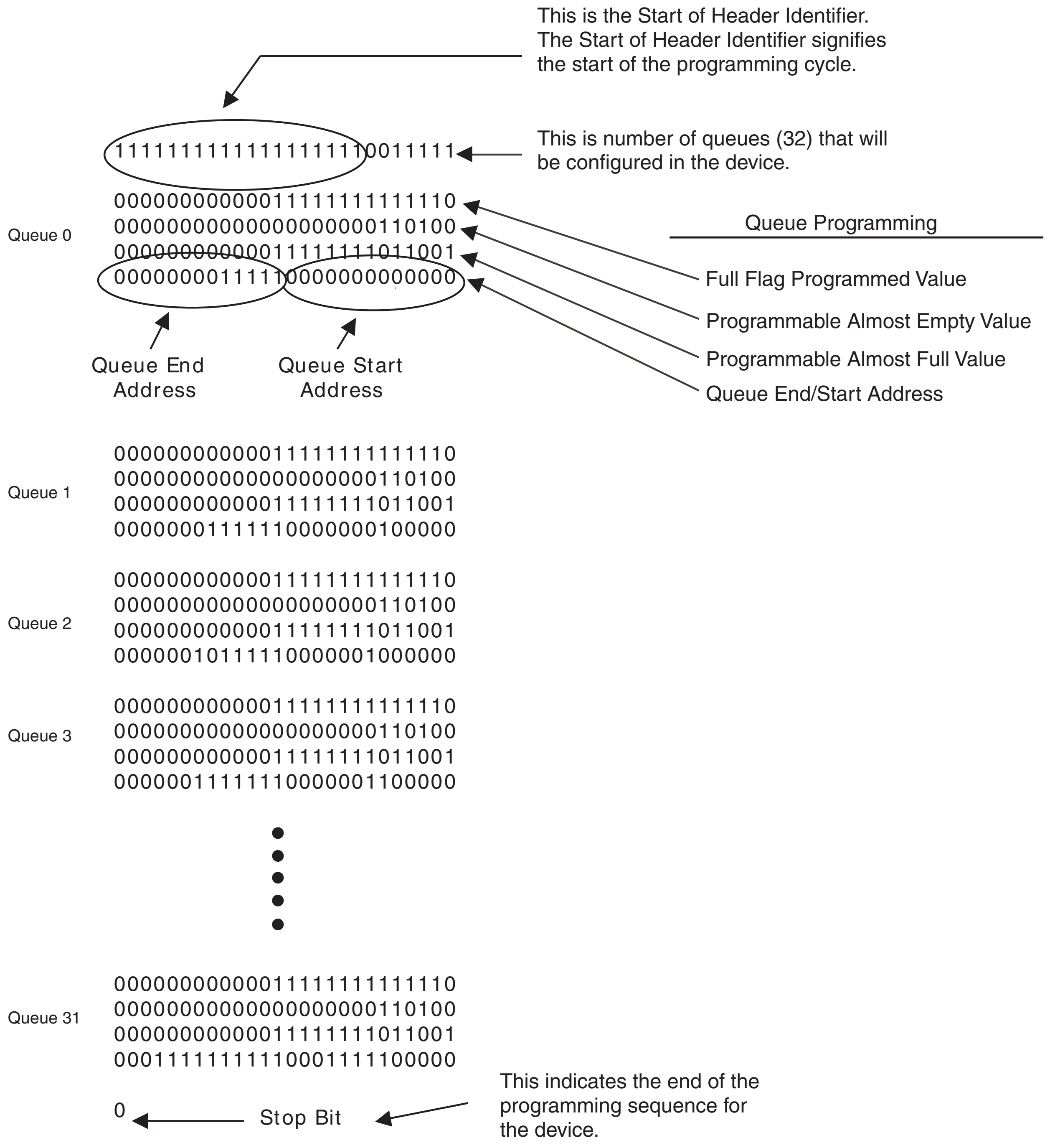
Bits	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Value	4096	2048	1024	512	256	128	64	32	16	8	4	2	1	4096	2048	1024	512	256	128	64	32	16	8	4	2	1

End Address ← → Start Address

The following is an explanation of the binary file created by the C program for the 10G MQ devices (IDT72P51767/72P51777).

The Desired Device Configuration is;

- Number of Queues = 32
- Each Queue Depth = 16k
- Each Queue $\overline{\text{PAE}}$ Value = 52
- Each Queue $\overline{\text{PAF}}$ Value = 39



This is the Start of Header Identifier. The Start of Header Identifier signifies the start of the programming cycle.

This is number of queues (32) that will be configured in the device.

Queue Programming

Full Flag Programmed Value

Programmable Almost Empty Value

Programmable Almost Full Value

Queue End/Start Address

Queue End Address

Queue Start Address

```
          11111111111111111110011111

          00000000000011111111111110
Queue 0   00000000000000000000110100
          00000000000011111111011001
          00000000111110000000000000

          00000000000011111111111110
Queue 1   00000000000000000000110100
          00000000000011111111011001
          00000001111110000000100000

          00000000000011111111111110
Queue 2   00000000000000000000110100
          00000000000011111111011001
          00000010111110000001000000

          00000000000011111111111110
Queue 3   00000000000000000000110100
          00000000000011111111011001
          00000011111110000001100000

                     ●
                     ●
                     ●
                     ●
                     ●

          00000000000011111111111110
Queue 31  00000000000000000000110100
          00000000000011111111011001
          00011111111110001111100000

          0
```

Stop Bit

This indicates the end of the programming sequence for the device.

DEFAULT PROGRAMMING

During a Master Reset if the DFM (Default Mode) input is HIGH the multi-queue device will be configured for default programming, (serial programming is not permitted). Default programming provides the user with a simpler, however limited means to setup the multi-queue flow-control device, rather than using the serial programming method. The default mode will configure a multi-queue device with the maximum number of queues setup, and the available memory allocated equally between the queues. The values of the PAE/PAF offsets is determined by the state of the (default) pin during a master reset.

For the IDT72P51767/72P51777 devices the default mode will setup 128 queues, each queue being 256 x40, 512 x40, and 1024 x40 deep respectively.

When configuring the IDT72P51767/72P51777 devices in default mode the user simply has to apply WCLK cycles after a master reset, until SENO goes LOW, this signals that default programming is complete. These clock cycles are required for the device to load its internal setup registers. When a single multi-queue device is used, the completion of device programming is signaled by the SENO output of a device going from HIGH to LOW. Note, that SENI must be held LOW when a device is setup for default programming mode.

When multi-queue devices are connected in expansion configuration, the SENI of the first device in a chain can be held LOW. The SENO of a device should connect to the SENI of the next device in the chain. The SENO of the final device is used to indicate that default programming of all devices is complete. When the master (ID='000') SENO goes LOW normal operations may begin. Again, all devices will be programmed with their maximum number of queues and the memory divided equally between them. Please refer to Figure 38, *Default Programming*.

PARALLEL PROGRAMMING

During a Master Reset cycle (i.e. the MRS signal transitions from HIGH to LOW then LOW to HIGH) if the DFM (Default Mode) input signal is HIGH and the QSEL[2:0] input signal is "110" for Write address and "111" for Read address, the Multi-Queue Flow Control device is configured for Parallel Programming. Parallel Programming enables the number of queues within the device to be set through either the Write Address (WRADD) bus or Read Address (RDADD) bus after the Master Reset cycle. Within Parallel Programming mode the Multi-Queue (MQ) device programmable parameters are; number of queues, queue depth, PAE/PAF flag offset value, bus matching and the I/O voltage level. As previously indicated, the number of queues are configured using the write or read address bus, however bus matching is set during the Master Reset cycle. The value that is set during the Master Reset cycle is determined by the Bus Matching (BM) bits. For the IDT72P51767/72P51777 devices in Parallel Programming Mode the value of the PAE/PAF offsets at master reset is determined by the state of the input.

When configuring the IDT72P51767/72P51777 devices in Parallel Programming Mode the user simply has to apply WCLK cycles after a master reset, until SENO goes LOW, this signals that Parallel Programming is complete. These clock cycles are required for the device to load its internal setup registers. When a single multi-queue device is used, the completion of device programming is signaled by the SENO output of a device going from HIGH to LOW. Note, that SENI must be held LOW when a device is setup for Parallel Programming mode.

ID[2:0] PINS AND WRADD/RDADD CONFIGURATIONS

The 10G DDR Multi-Queue will have the ability to expand up to a maximum of 256 Queues by 2 to 8 devices depending on the configuration setup. For programming the ID Codes for each device the WRADD/RDADD address buses impose the limitation of 256 Queues because, with an 8-bit address bus, the 3 MSB in the WRADD/RDADD are used to decode the ID[2:0] pins on each device. The slave devices must be configured before the master device as well. For 8-device expansion the least [4:0] bits are used for expansion addressing meaning a configuration of 32 queues can be accomplished. The MSBs [7:5] will select which of the eight devices to access depending on their MSB ID[2:0] settings. If ID[2:0] is 000, then it serves as the master device. Otherwise, the devices will serve as the slave devices. Ex: For using two 128 queue MQ devices, ID[2:0] = 0xx is for selecting the master device. For using 8 devices, ID[2:0] = 111 is for selecting the 8th slave device and WRADD/RDADD bits [4:0] for configuring 32 queues/device. And so on.

TABLE 6 — ID[2:0] AND WRADD[7:5]/RDADD[7:5] CONFIGURATION

Queues Addressed	x	128	64	32	16	8	4	2
ID[2:0] Master Device Configuration WRADD[7:5] or RDADD[7:5]		0xx	00x	000	000	000	000	000
ID[2:0] 2nd Device Configuration WRADD[7:5] or RDADD[7:5]		1xx	10x	100	100	100	100	100
ID[2:0] 3rd Device Configuration WRADD[7:5] or RDADD[7:5]		NA	01x	010	010	010	010	010
ID[2:0] 4th Device Configuration WRADD[7:5] or RDADD[7:5]		NA	11x	110	110	110	110	110
ID[2:0] 5th Device Configuration WRADD[7:5] or RDADD[7:5]		NA	NA	001	001	001	001	001
ID[2:0] 6th DeviceConfiguration WRADD[7:5] or RDADD[7:5]		NA	NA	101	101	101	101	101
ID[2:0] 7th Device Configuration WRADD[7:5] or RDADD[7:5]		NA	NA	011	011	011	011	011
ID[2:0] 8th Device Configuration WRADD[7:5] or RDADD[7:5]		NA	NA	111	111	111	111	111

TABLE 7 — PARALLEL PROGRAMMING MODE QUEUE CONFIGURATION EXAMPLE[1]

WRADD/RDADD[7:0]	7	6	5	4	3	2	1	0
128 Queues/Device	x	**1**	1	1	1	1	1	1
64 Queues/Device	x	0	**1**	1	1	1	1	1
32 Queues/Device	x	0	0	**1**	1	1	1	1
16 Queues/Device	x	0	0	0	**1**	1	1	1
8 Queues/Device	x	0	0	0	0	**1**	1	1
4 Queues/Device	x	0	0	0	0	0	**1**	1
1 Queue/Device	x	0	0	0	0	0	0	**0**

NOTE:
1. Users can also program 6 different settings in Default Programming mode using QSEL[2:0] and DFM pins instead of using Parallel Programming Mode. This table above shows how a user might program the number of queues in the device and can program any 1-128 queues in the device by using the WRADD/RDADD[7:0] pins.

The 10G Multi-Queue also has the capability of expanded theoretically to unlimited number of devices. As Figure 5 shows, the WRADD[7] and RDADD[7] will act as the Write Chip Select Enable and Read Chip Select Enable respectively, and the user can then decode each Multi-Queue device separately using these pins and still maintain 128 Queues per device. The WRADD[6:0] and RDADD[6:0] can still be routed as shared buses to each device, as can write and read data buses and external control pins and flags. To make this possible the ID pin, ID2 of all slave devices must be connected to "1" and the Master device ID Pins, ID[2:0], should equal 000. It is not recommend expanding greater than 10 devices because sufficient capacitive loading on each bus makes it diffcult to drive a greater multiple of devices per bus. The real $\overline{\text{WCS}}$ and $\overline{\text{RCS}}$ pins can be tied to a ground plane on the board to save FPGA pins.

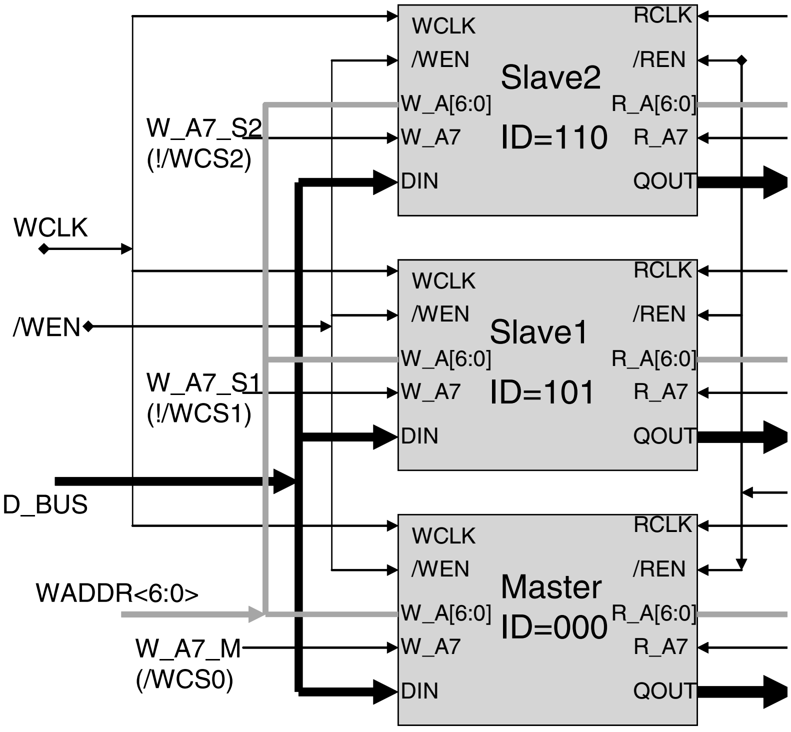


Figure 5. Expansion for Unlimited Number of Multi-Queue Devices Example

When Multi-Queue devices are connected in an Expansion Configuration, the $\overline{\text{SENI}}$ signal of the first device in a chain must be held LOW. The $\overline{\text{SENO}}$ signal of a device should connect to the $\overline{\text{SENI}}$ of the next device in the chain. The $\overline{\text{SENO}}$ of the final device is used to indicate that the programming of all devices is complete. When the master device (ID='000') $\overline{\text{SENO}}$ signal goes LOW the internal programming is complete and queue write/read operation may begin. Please refer to Figure 41, *Parallel Programming* for signal timing details.

PROGRAMMING HIERARCHY

Configuring the device is a 2 stage sequence. The first stage is to set the expansion device type, the desired programming mode and the device operating mode during the master reset cycle (i.e. on the rising edge of Master Reset ($\overline{\text{MRS}}$)). The second stage is to set values such as $\overline{\text{PAE}}$/$\overline{\text{PAF}}$, number of queues, queue depth, etc. using the programming mode (serial, parallel, default) selected in stage 1. Refer to Figure 6, *Device Programming Hierarchy*.

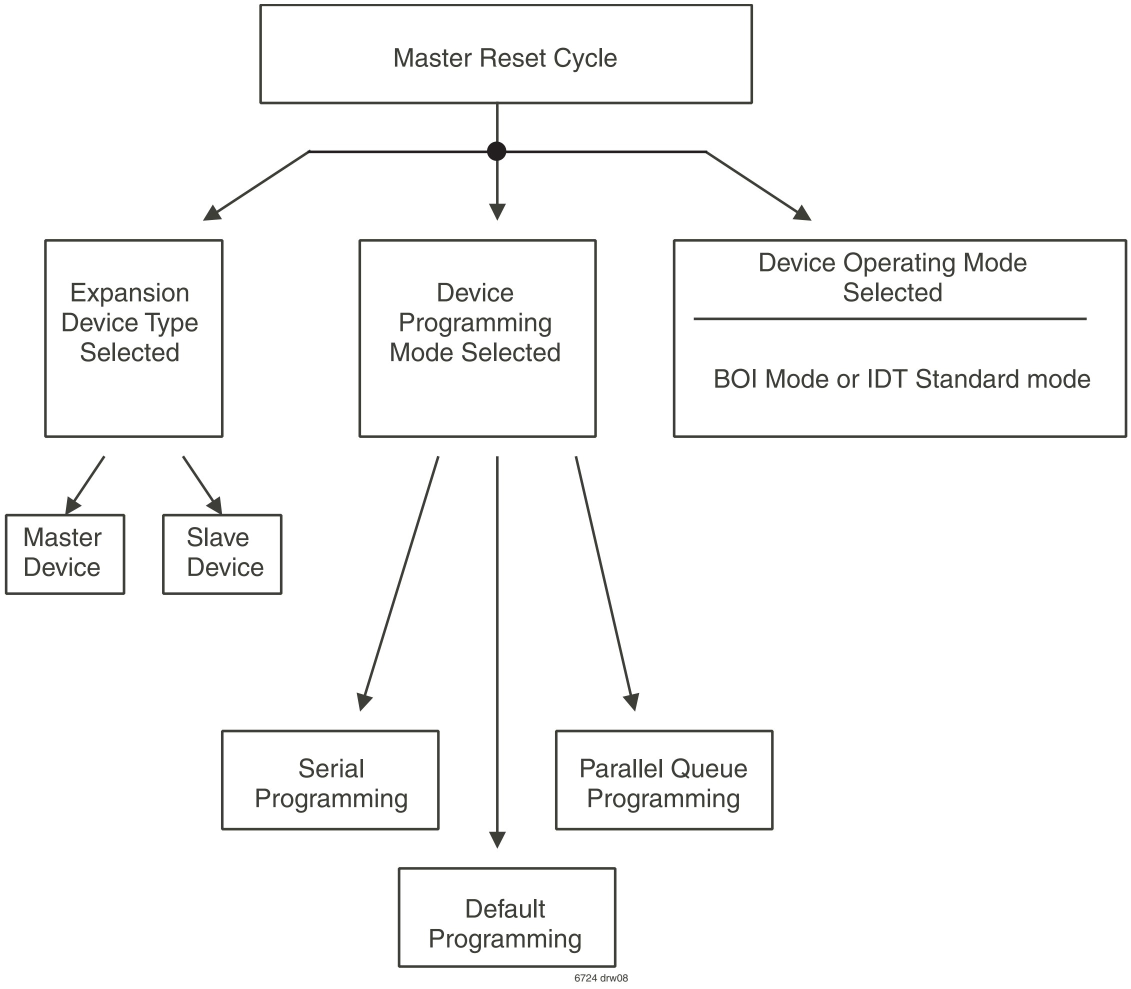

Master Reset Cycle
Expansion Device Type Selected
Device Programming Mode Selected
Device Operating Mode Selected
BOI Mode or IDT Standard mode
Master Device
Slave Device
Serial Programming
Parallel Queue Programming
Default Programming
6724 drw08

Figure 6. Device Programming Hierarchy

MODES OF OPERATION

STANDARD MODE OPERATION

The 10G Multi-queue supports two modes of operation, IDT Standard Mode and BOI Mode.

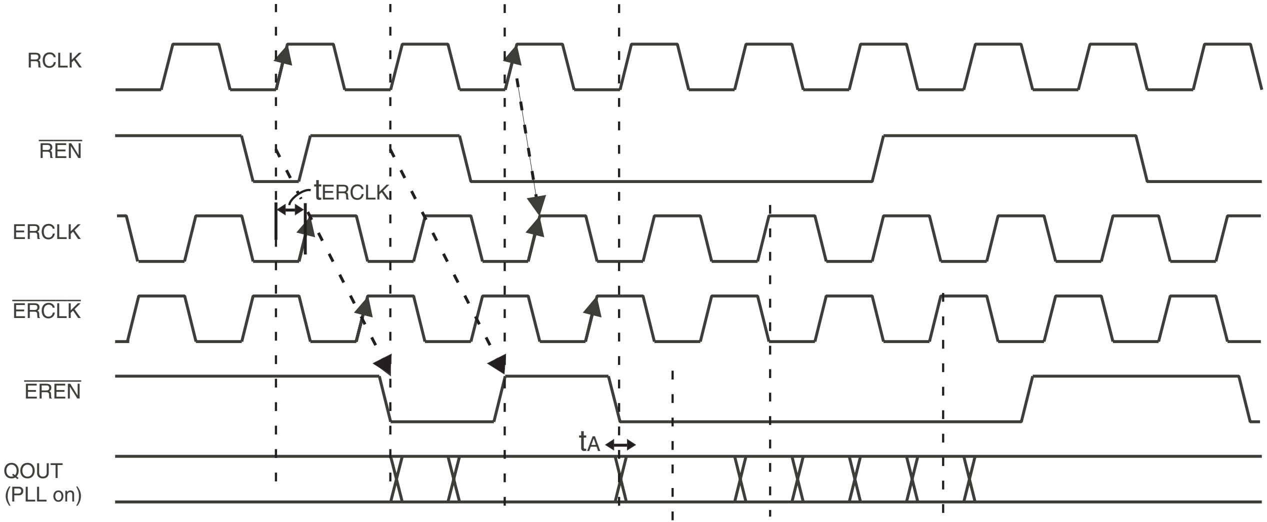
IDT STANDARD MODE VS. BOI MODE

The device mode is configured during the master reset cycle. Only the read side is affected by these two modes, the write side is identical in both modes. In IDT standard mode, the device operates as the IDT72P51769 (4M multi-queue) device as shown in Figure 49 and 50.

The other option, or BOI mode, has the device reissue the last data from a queue upon reentry to that queue. This allows for a customer to monitor a data bit and stop reading based on the output (e.g., End of Packet or EOP). Note: the $\overline{\text{EF}}$ flag is always updated "real time" using the read count in both modes.

PLL ON VS PLL OFF MODES

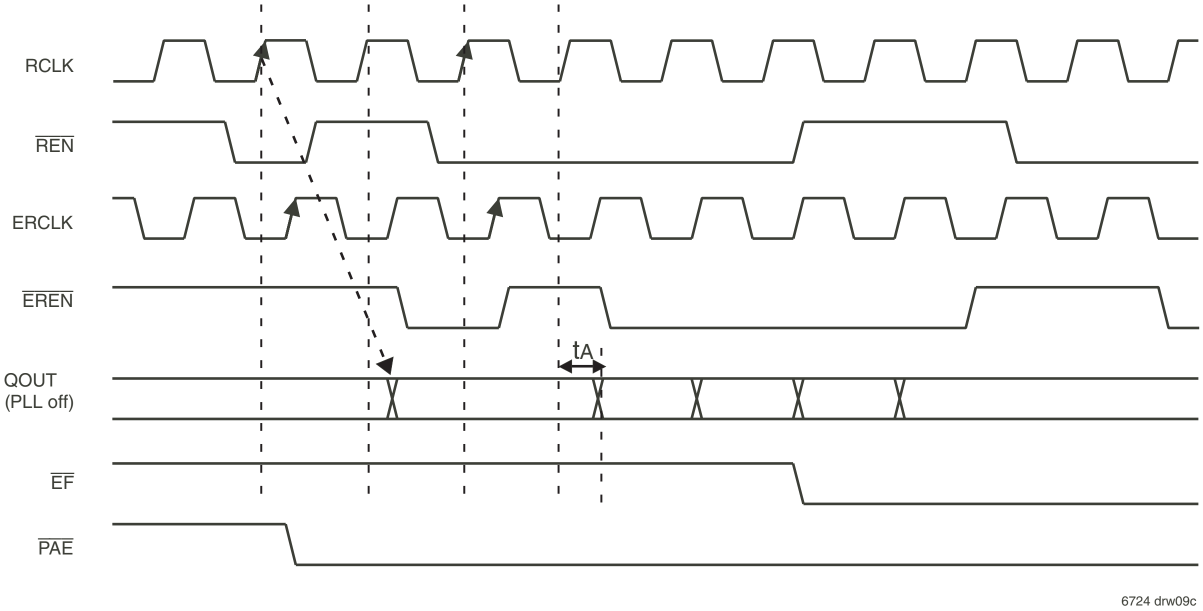
The PLL has a frequency response rate of 85-166MHz. The PLL reduces the access time (ta) for the DDR. Below 85MHz or in SDR mode, the PLL must be turned off. The following diagrams show the difference between the two modes:



NOTES:

1. Echo clocks are center aligned.
2. $\overline{ERCLK}$ is aligned with internal PLL generated clock.
3. QOUT is clocked out 1 cycle after valid $\overline{REN}$.
4. Data Access time (Ta) is +/-1.0ns when PLL is on, with respect to rising edges of RCLK (1 Clock Latency).

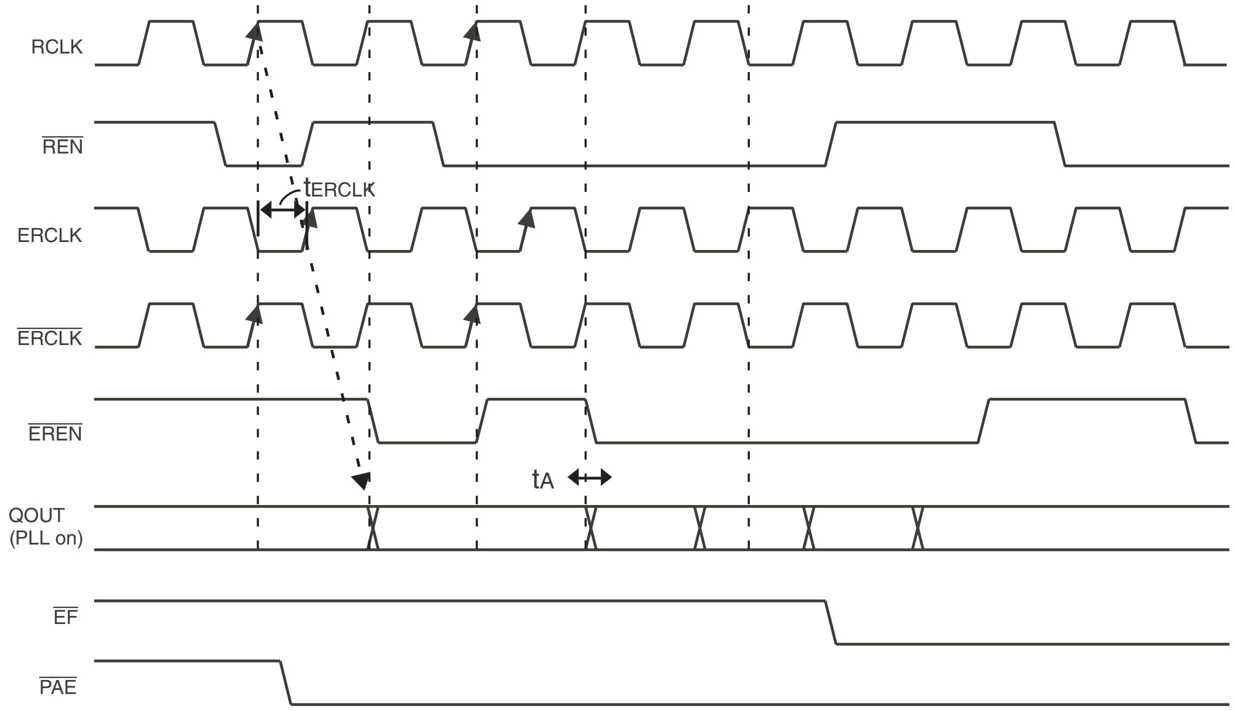
Figure 7. DDR Read Operation with PLL ON



NOTES:

1. $\overline{ERCLK}$ has uncontrollable delay when PLL is off.
2. DOUT is clocked out 1 cycle after valid $\overline{REN}$.
3. Data Access time (Ta) is <3.6ns when PLL is off, with respect to rising/falling edges of RCLK.

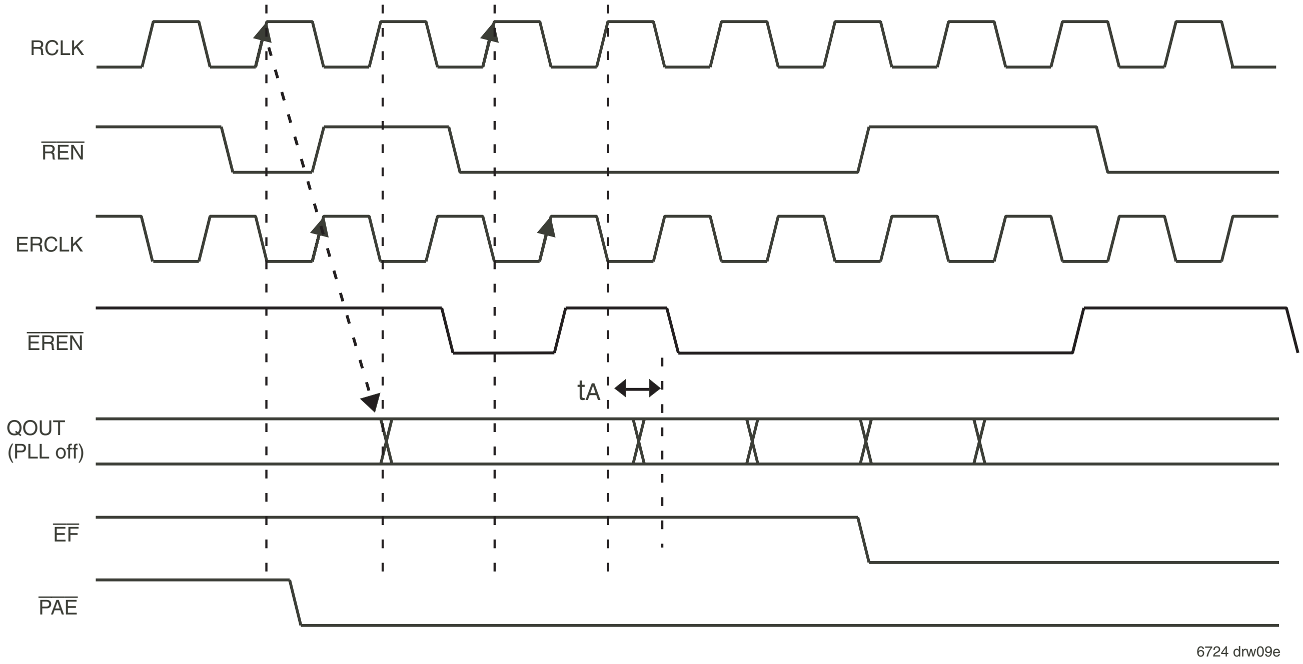
Figure 8. DDR Read Operation with PLL OFF



NOTES:

1. In single-ended clocking scheme, ERCLK is 50% duty cycle.
2. QOUT is clocked out 1 cycle after valid REN.
3. Data Access time (Ta) is +/-1.0ns when PLL is on, with respect to rising edges of RCLK.

Figure 9. SDR Read Operation with PLL ON



NOTES:

1. ERCLK has uncontrollable delay when PLL is off, Ta <3.6ns when PLL is off.
2. DOUT is clocked out 1 cycle after valid REN.
3. Data Access time (Ta) is <3.6ns when PLL is off, with respect to rising edges of RCLK.

Figure 10. SDR Read Operation with PLL OFF

STANDARD MODE OPERATION

WRITE QUEUE SELECTION AND WRITE OPERATION (STANDARD MODE)

The IDT72P51767/72P51777 multi-queue flow-control devices can be configured up to a maximum of 128 queues which data can be written via a common write port using the data inputs (Din), write clock (WCLK) and write enable ($\overline{\text{WEN}}$). The queue to be written is selected by the address present on the write address bus (WRADD) during a rising edge on WCLK while write address enable (WADEN) is HIGH. The state of $\overline{\text{WEN}}$ does not impact the queue selection. The queue selection requires 4 WCLK cycle. All subsequent data writes will be to this queue until another queue is selected.

Standard mode operation is defined as individual words will be written to the device. The write port is designed such that 100% bus utilization can be obtained. This means that data can be written into the device on every WCLK rising edge including the cycle that a new queue is being addressed.

Changing queues requires 4 WCLK cycles on the write port (see Figure 42, *SDR Write Queue Select, Write Operation and Full flag Operation*). WADEN goes high signaling a change of queue (clock cycle "A"). The address on WRADD at that time determines the next queue. Data presented during that cycle, will be written to the active (each) queue, provided $\overline{\text{WEN}}$ is LOW. If $\overline{\text{WEN}}$ is HIGH (inactive), data will not be written in a queue. The write port discrete full flag will update to show the full status of the newly selected queue. Data present on the data input bus (Din), can be written into the newly selected queue on the rising edge of WCLK a change of queue, provided $\overline{\text{WEN}}$ is LOW and the queue is not full. If the selected queue is full at the point of its selection, any writes to that queue will be prevented. Data cannot be written into a full queue.

Refer to Figure 42, *SDR Write Queue Select, Write Operation and Full flag Operation*, and Figure 45, *Full Flag Timing in Expansion Configuration* for timing diagrams.

TABLE 8 — WRITE ADDRESS BUS, WRADD[7:0]

Operation	WCLK	WADEN	FSTR	WRADD[7:0]
Write Queue Select	↑	1	0	7 6 ¦ 5 ¦ 4 3 2 1 0 Device Select (Compared to ID2,1,0) Write Queue Address (6 bits = 64 Queues 7 bits = 128 Queues)
$\overline{\text{PAF}}$n Quadrant Select	↑	0	1	7 6 5 | 4 | 3 2 1 0 Device Select (Compared to ID2,1,0) X Status Word Address

Status Word Address	Queue Status on $\overline{\text{PAF}}$n Bus
0000	Q0 : Q7 → $\overline{\text{PAF}}$0 : $\overline{\text{PAF}}$7
0001	Q8 : Q15 → $\overline{\text{PAF}}$0 : $\overline{\text{PAF}}$7
0010	Q16 : Q23 → $\overline{\text{PAF}}$0 : $\overline{\text{PAF}}$7
0011	Q24 : Q31 → $\overline{\text{PAF}}$0 : $\overline{\text{PAF}}$7
0100	Q32 : Q39 → $\overline{\text{PAF}}$0 : $\overline{\text{PAF}}$7
0101	Q40 : Q47 → $\overline{\text{PAF}}$0 : $\overline{\text{PAF}}$7
0110	Q48 : Q55 → $\overline{\text{PAF}}$0 : $\overline{\text{PAF}}$7
0111	Q56 : Q63 → $\overline{\text{PAF}}$0 : $\overline{\text{PAF}}$7
1000	Q64 : Q71 → $\overline{\text{PAF}}$0 : $\overline{\text{PAF}}$7
1001	Q72 : Q79 → $\overline{\text{PAF}}$0 : $\overline{\text{PAF}}$7
1010	Q80 : Q87 → $\overline{\text{PAF}}$0 : $\overline{\text{PAF}}$7
1011	Q88 : Q95 → $\overline{\text{PAF}}$0 : $\overline{\text{PAF}}$7
1100	Q96 : Q103 → $\overline{\text{PAF}}$0 : $\overline{\text{PAF}}$7
1101	Q104 : Q111 → $\overline{\text{PAF}}$0 : $\overline{\text{PAF}}$7
1110	Q112 : Q119 → $\overline{\text{PAF}}$0 : $\overline{\text{PAF}}$7
1111	Q120 : Q127 → $\overline{\text{PAF}}$0 : $\overline{\text{PAF}}$7

6724 drw11

READ QUEUE SELECTION AND READ OPERATION (STANDARD MODE)

The IDT72P51767/72P51777 multi-queue flow-control devices can be configured up to a maximum of 128 queues which data can be read via a common read port using the data outputs (Qout), read clock (RCLK) and read enable ($\overline{\text{REN}}$). An output enable, $\overline{\text{OE}}$ control pin is also provided to allow High-Impedance selection of the Qout data outputs. The multi-queue device read port operates in standard IDT mode or BOI mode. The queue to be read is selected by the address presented on the read address bus (RDADD) during a rising edge on RCLK while read address enable (RADEN) is HIGH. The state of $\overline{\text{REN}}$ does not impact the queue selection. The queue selection requires 4 RCLK cycles. All subsequent data reads will be from this queue until another queue is selected.

Standard mode operation is defined as individual words will be read from the device. The read port is designed such that 100% bus utilization can be obtained. This means that data can be read out of the device on every RCLK rising edge including the cycle that a new queue is being addressed.

Changing queues requires a minimum of four RCLK cycles on the read port (see Figure 46, *SDR Read Queue Select, Read Operation*). RADEN goes high signaling a change of queue (clock cycle “D”). The address on RDADD at that time determines the next queue. Data presented during that cycle will be read. Reading data can continue from the active queue, provided $\overline{\text{REN}}$ is LOW. If $\overline{\text{REN}}$ is HIGH (inactive), data will not be read from the queue. If a new selected queue is empty, reads from that queue will be prevented. Data cannot be read from an empty queue. Remember that $\overline{\text{OE}}$ allows the user to place the data output bus (Qout) into High-Impedance and the data can be read in to the output register regardless of $\overline{\text{OE}}$.

Refer to Table 9, for Read Address Bus arrangement.

TABLE 9 — READ ADDRESS BUS, RDADD[7:0]

Operation	RCLK	RADEN	ESTR	RDADD[7:0]		
Read Queue Select	↑	1	0	7 6 5 Device Select (Compared to ID2,1,0)		4 3 2 1 0 Read Queue Address (6 bits = 64 Queues 7 bits = 128 Queues)
$\overline{\text{PAE}}$n Quadrant Select	↑	0	1	7 6 5 Device Select (Compared to ID2,1,0)	4 X	3 2 1 0 Status Word Address

Status Word Address	Queue Status on $\overline{\text{PAE}}$n Bus
0000	Q0 : Q7 → $\overline{\text{PAF}}$0 : $\overline{\text{PAF}}$7
0001	Q8 : Q15 → $\overline{\text{PAF}}$0 : $\overline{\text{PAF}}$7
0010	Q16 : Q23 → $\overline{\text{PAF}}$0 : $\overline{\text{PAF}}$7
0011	Q24 : Q31 → $\overline{\text{PAF}}$0 : $\overline{\text{PAF}}$7
0100	Q32 : Q39 → $\overline{\text{PAF}}$0 : $\overline{\text{PAF}}$7
0101	Q40 : Q47 → $\overline{\text{PAF}}$0 : $\overline{\text{PAF}}$7
0110	Q48 : Q55 → $\overline{\text{PAF}}$0 : $\overline{\text{PAF}}$7
0111	Q56 : Q63 → $\overline{\text{PAF}}$0 : $\overline{\text{PAF}}$7
1000	Q64 : Q71 → $\overline{\text{PAF}}$0 : $\overline{\text{PAF}}$7
1001	Q72 : Q79 → $\overline{\text{PAF}}$0 : $\overline{\text{PAF}}$7
1010	Q80 : Q87 → $\overline{\text{PAF}}$0 : $\overline{\text{PAF}}$7
1011	Q88 : Q95 → $\overline{\text{PAF}}$0 : $\overline{\text{PAF}}$7
1100	Q96 : Q103 → $\overline{\text{PAF}}$0 : $\overline{\text{PAF}}$7
1101	Q104 : Q111 → $\overline{\text{PAF}}$0 : $\overline{\text{PAF}}$7
1110	Q112 : Q119 → $\overline{\text{PAF}}$0 : $\overline{\text{PAF}}$7
1111	Q120 : Q127 → $\overline{\text{PAF}}$0 : $\overline{\text{PAF}}$7

6724 drw12

SWITCHING QUEUES ON THE WRITE PORT

The IDT72P51767/72P51777 multi-queue flow-control devices can be configured up to a maximum of 128 queues. Data is written into a queue using the Data Input (Din) bus, Write Clock (WCLK) and Write Enable ($\overline{\text{WEN}}$) signals. Selecting a queue occurs by placing the queue address on the Write Address bus (WRADD) during a rising edge of WCLK while Write Address Enable (WADEN) is HIGH. For reference, the state of Write Enable ($\overline{\text{WEN}}$) is a "Don't Care" during a queue selection. $\overline{\text{WEN}}$ has significance during the queue mark operation. Selecting a queue requires 4 WCLK cycles. Refer to Figure 11, *Write Port Switching Queues Signal Sequence*.

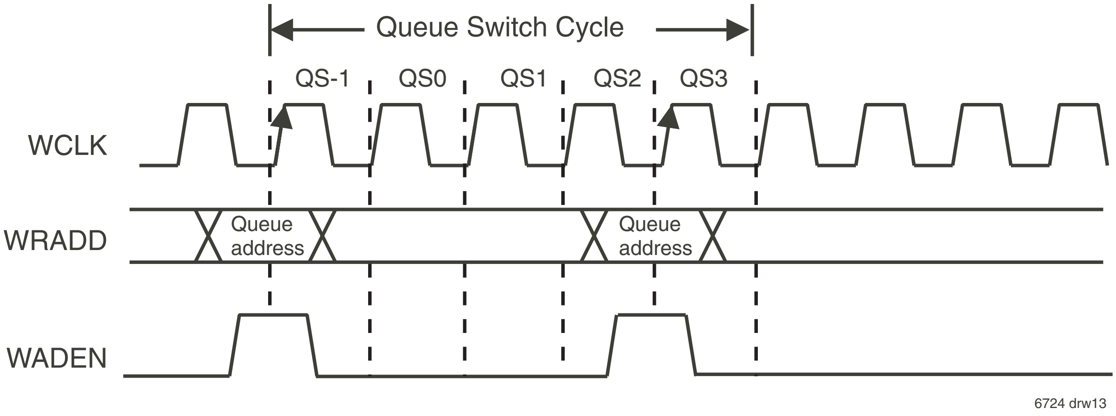
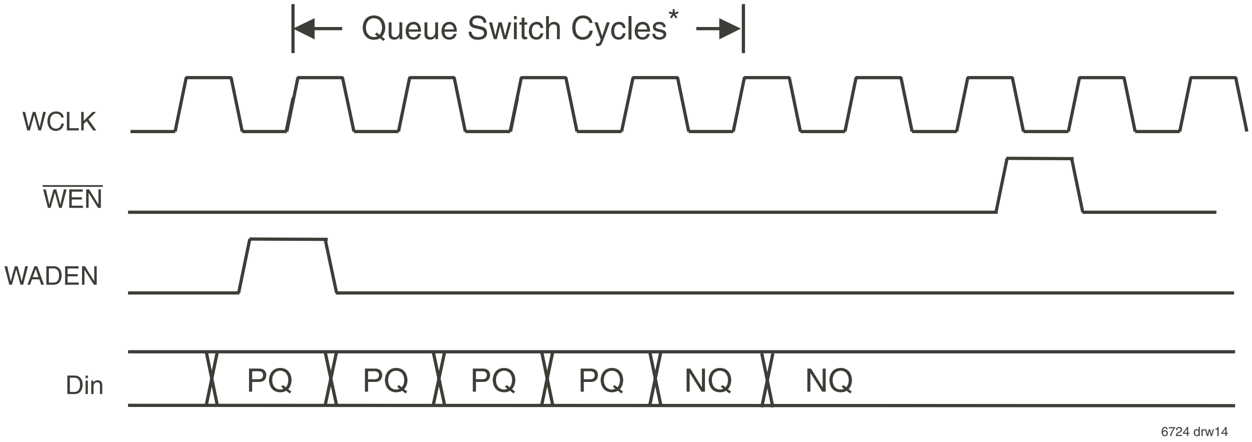


Figure 11. Write Port Switching Queues Signal Sequence

The IDT72P51767/72P51777 multi-queue flow-control device supports changing (switching) queues every four (4) clock cycles. To switch from the Present Queue (PQ) to another queue requires a queue address to be placed on the Write Address Bus (WRADD) bus and a rising edge of Write Clock (WCLK) and Write Address Enable (WADEN) is HIGH. There are no restrictions as to the order to which queues are selected or switched into or out of.

For maximum efficiency, during the 4 clock cycles required to switch queues the IDT72P51767/72P51777 multi-queue flow-control device can continue to write into the Present Queue (PQ). The Present Queue is defined as the current selected queue. Refer to Figure 12, *Switching Queues Bus Efficiency*.



NOTES:

1. PQ = Present Queue
   NQ = Next Queue

* Requires 4 clock cycles to switch queues.

Figure 12. Switching Queues Bus Efficiency

The IDT72P51767/72P51777 multi-queue flow-control device supports writing and reading from either the same queue of from different queues. The device also supports simultaneous queue switching on the write and read ports. The simultaneous queue switching may occur with either the Write Clock and Read Clock synchronous or asynchronous to each other. For reference refer to Figure 13, *Simultaneous Queue Switching*.

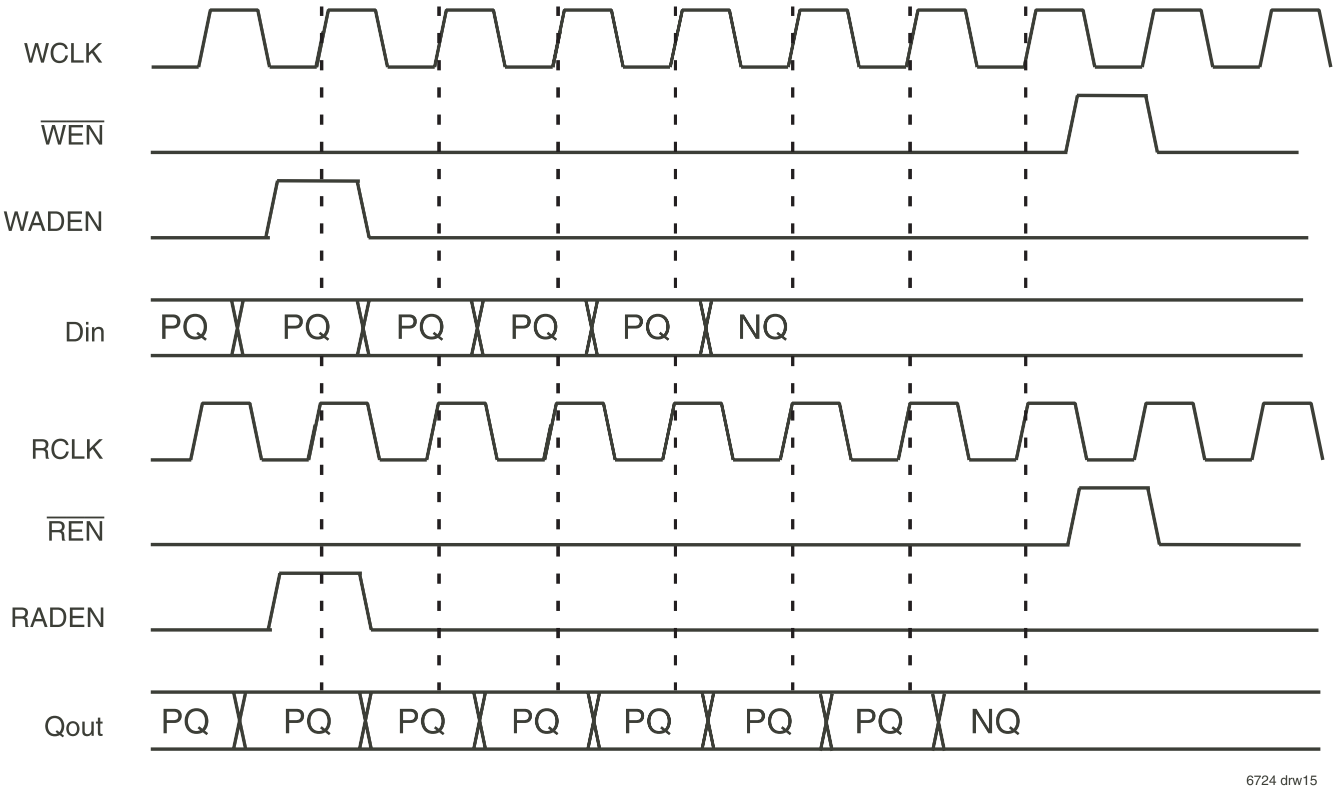


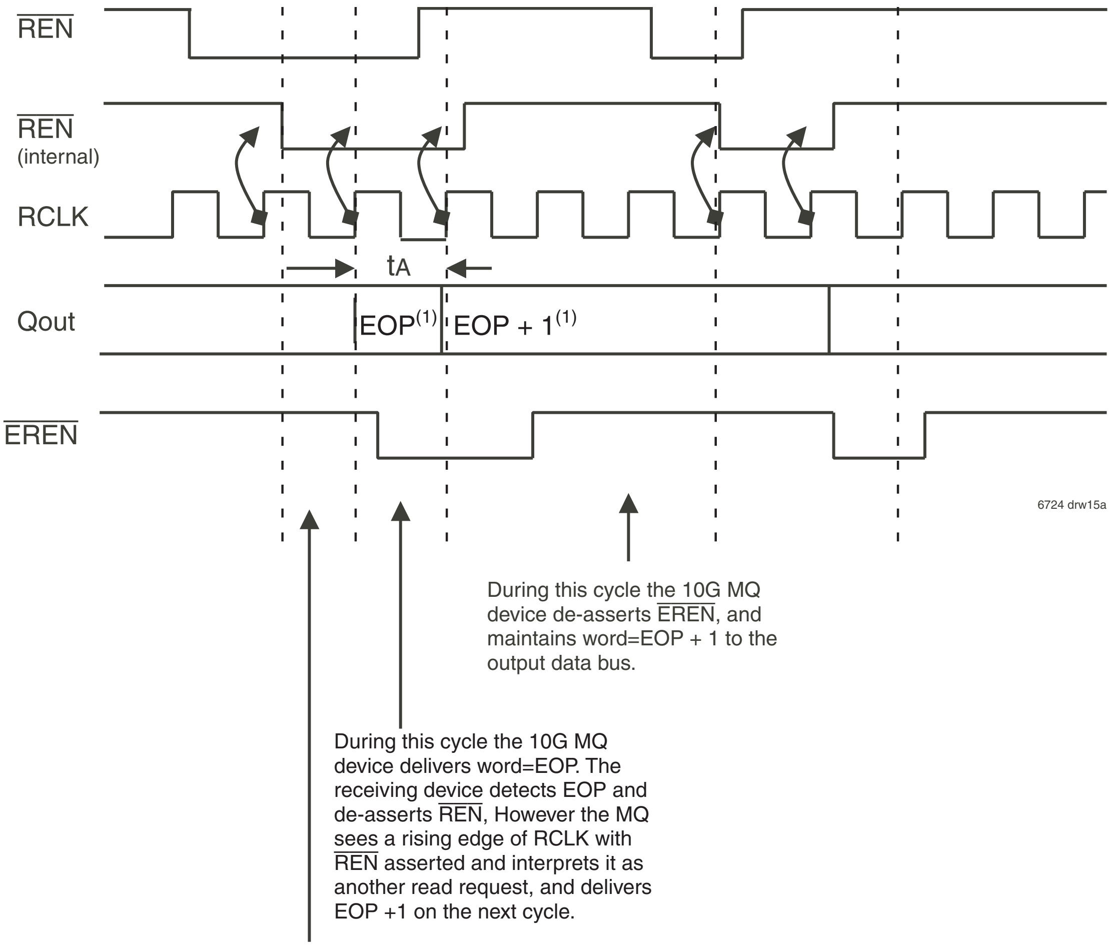
Figure 13. Simultaneous Queue Switching

The multi-queue flow-control device requires 4 clock cycles to switch queues on the write port. Refer to Table 10, Write Queue Switch Operation for a detailed description of each queue switch clock cycle.

TABLE 10 — WRITE QUEUE SWITCH OPERATION

Queue Switch Cycle	IDT Mode
QS-1	Queue Switch Initiated, Rewrite/No Rewrite selection
QS0	Queue MARK / Un-MARK
QS1	—
QS2	• $\overline{\text{PAF}}$ signal updated for Next Queue (NQ) • Full Flag ($\overline{\text{FF}}$) updated for NQ
QS3	Start of Write Data Operation

OUTPUT DATA DURING A QUEUE SWITCH



During this cycle the 10G MQ device de-asserts $\overline{\text{EREN}}$, and maintains word=EOP + 1 to the output data bus.

During this cycle the 10G MQ device delivers word=EOP. The receiving device detects EOP and de-asserts $\overline{\text{REN}}$, However the MQ sees a rising edge of RCLK with $\overline{\text{REN}}$ asserted and interprets it as another read request, and delivers EOP +1 on the next cycle.

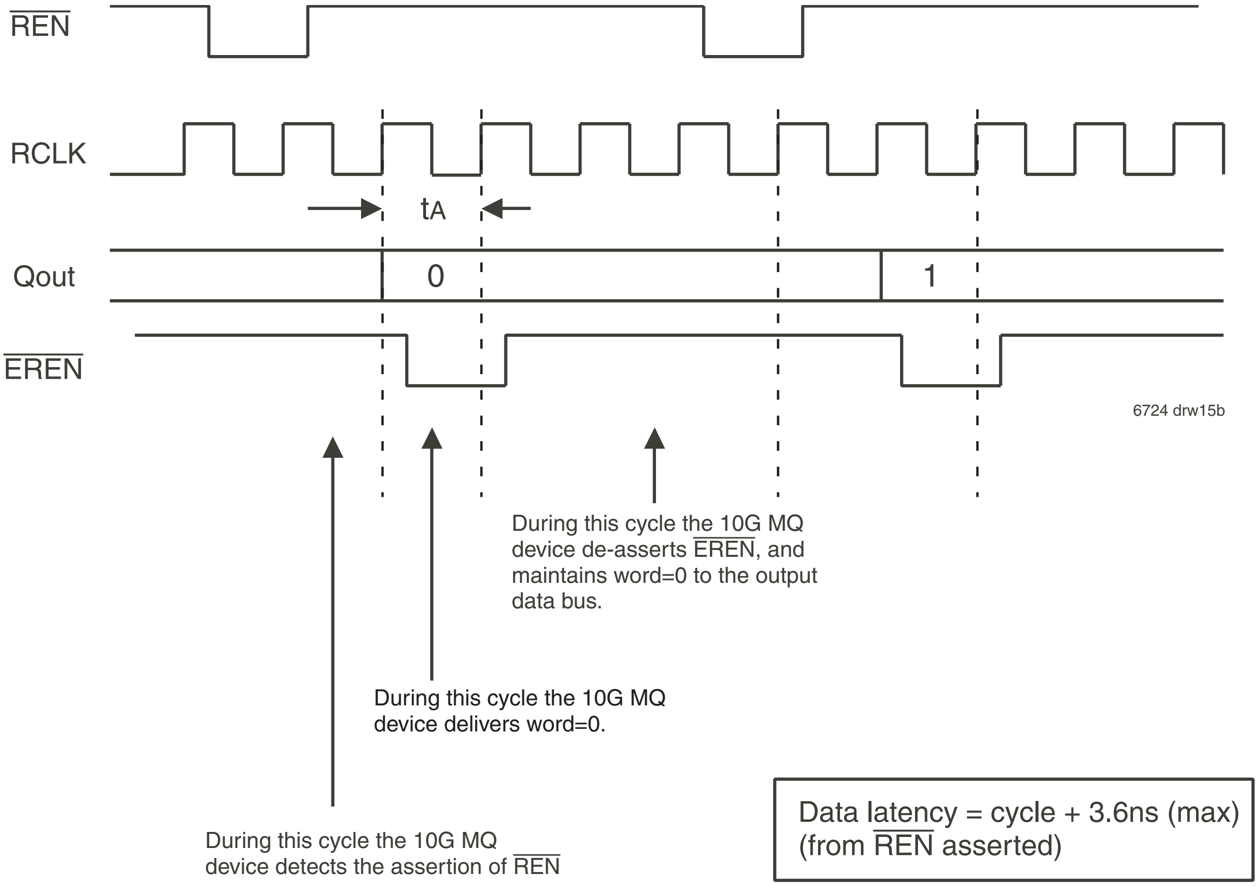
During this cycle the 10G MQ device detects the assertion of $\overline{\text{REN}}$

Data latency = cycle + 3.6ns (max)

NOTE:

1. Application specific.

Figure 14. Application: Reading words from the MQ using the EOP bit to end the read operation



During this cycle the 10G MQ device de-asserts $\overline{\text{EREN}}$, and maintains word=0 to the output data bus.

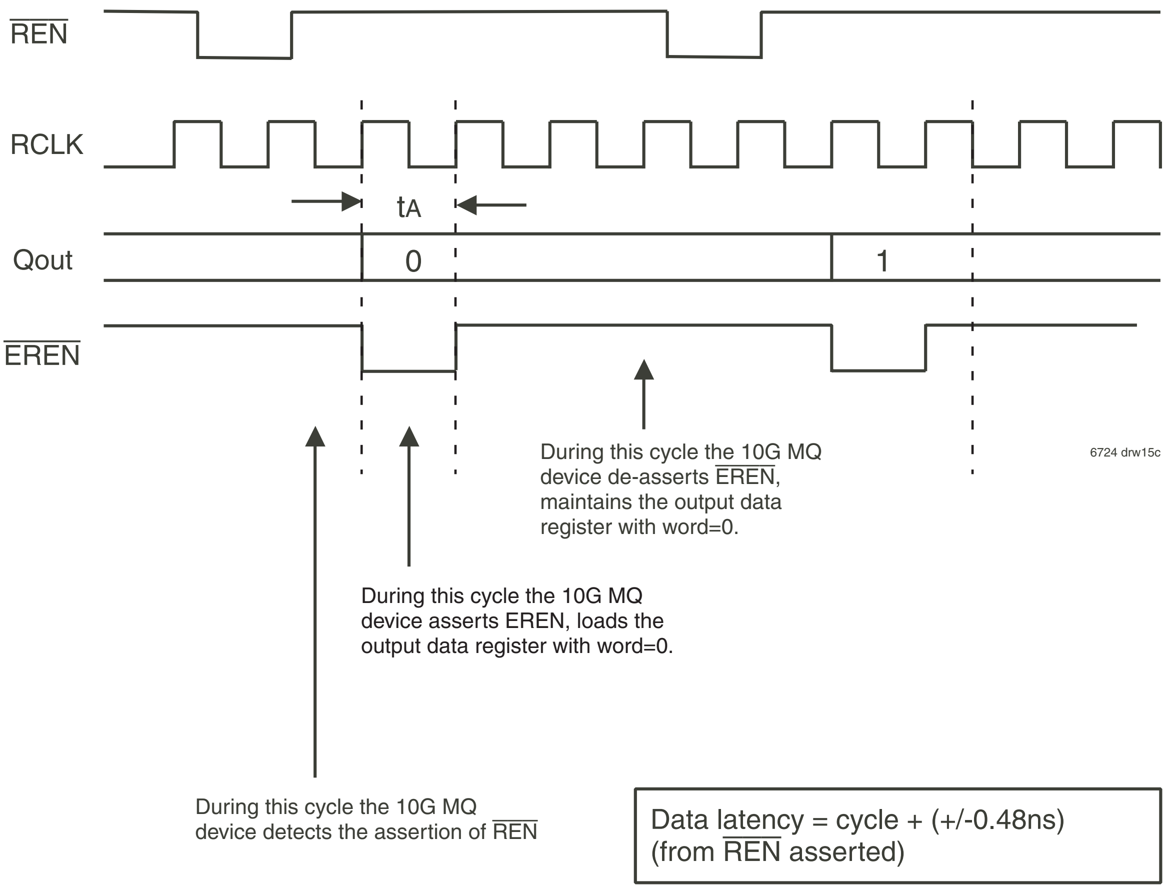
During this cycle the 10G MQ device delivers word=0.

During this cycle the 10G MQ device detects the assertion of $\overline{\text{REN}}$

Data latency = cycle + 3.6ns (max)
(from $\overline{\text{REN}}$ asserted)

Figure 15. Output Data during a Queue Switch (SDR w/o PLL)

Parameters: SDR Mode, PLL=ON, 1 read operation, centered aligned data/clock pause, 1 word read operation



During this cycle the 10G MQ device de-asserts EREN, maintains the output data register with word=0.

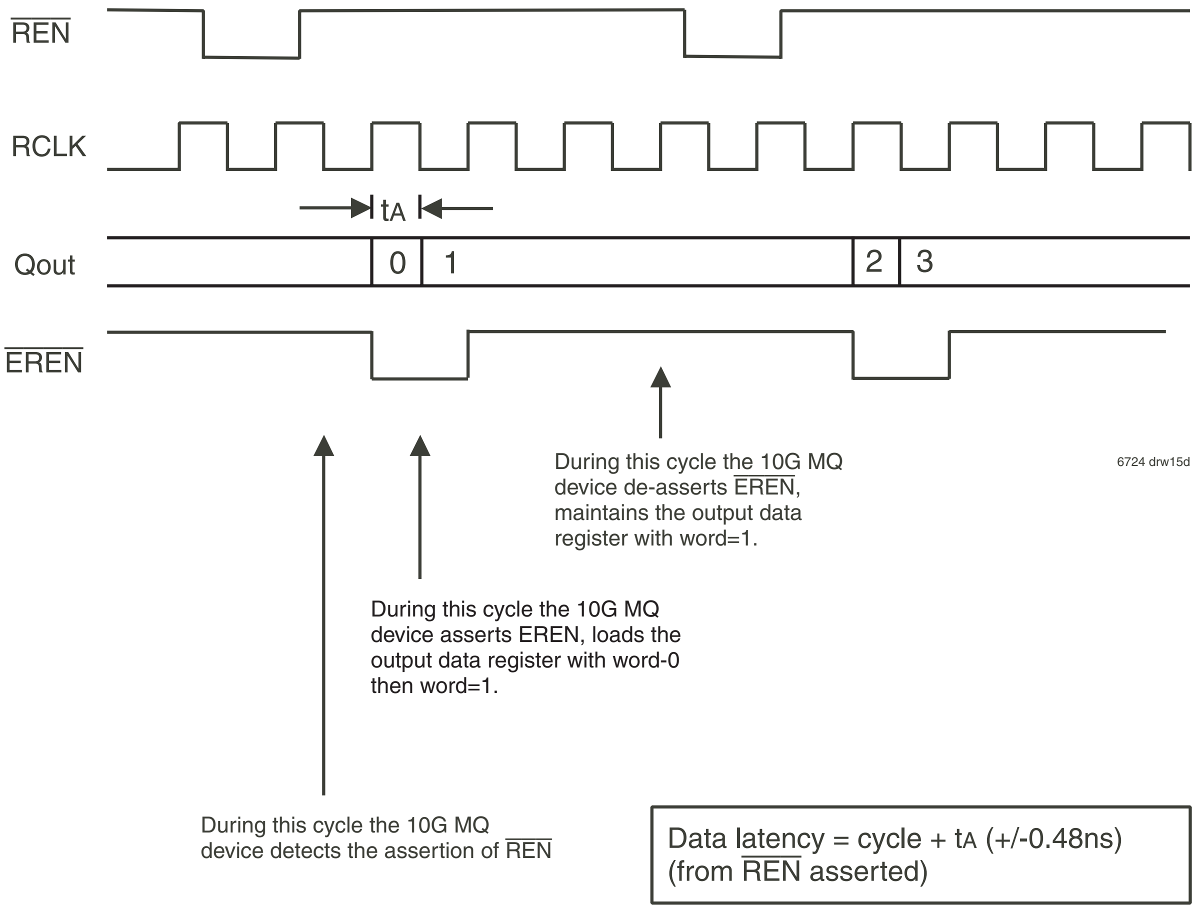
During this cycle the 10G MQ device asserts EREN, loads the output data register with word=0.

During this cycle the 10G MQ device detects the assertion of REN

Data latency = cycle + (+/-0.48ns) (from REN asserted)

Figure 16. Output Data during a Queue Switch (SDR w/ PLL)

Parameters: DDR Mode, PLL=ON, 1 word read operation, centered aligned data/clock pause with No queue switch, 1 word read operation



During this cycle the 10G MQ device de-asserts $\overline{\text{EREN}}$, maintains the output data register with word=1.

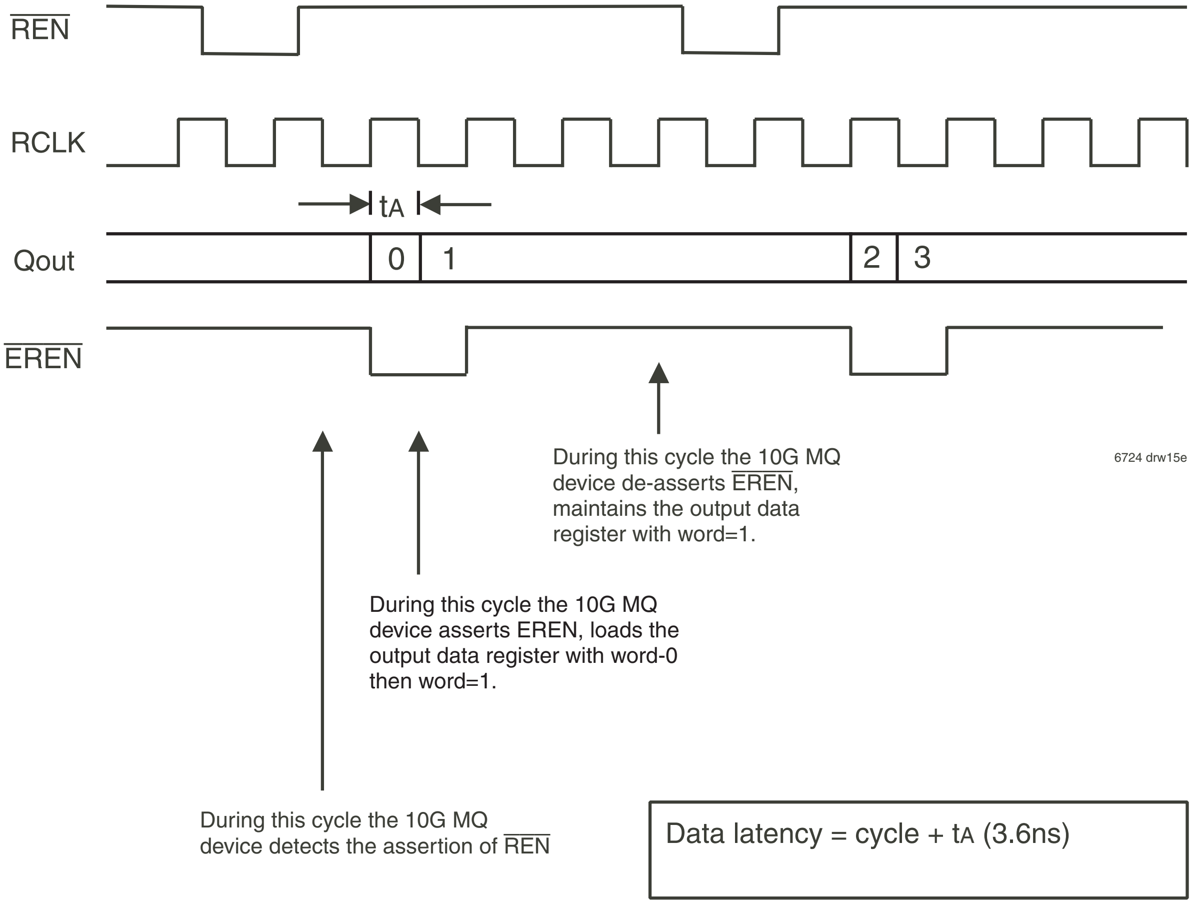
During this cycle the 10G MQ device asserts EREN, loads the output data register with word-0 then word=1.

During this cycle the 10G MQ device detects the assertion of $\overline{\text{REN}}$

Data latency = cycle + tA (+/-0.48ns)
(from $\overline{\text{REN}}$ asserted)

Figure 17. Output Data during a Queue Switch (DDR w/ PLL)

Parameters: DDR Mode, PLL=OFF, 1 word read operation, edge aligned data/clock
pause with No queue switch, 1 word read operation



During this cycle the 10G MQ device de-asserts $\overline{\text{EREN}}$, maintains the output data register with word=1.

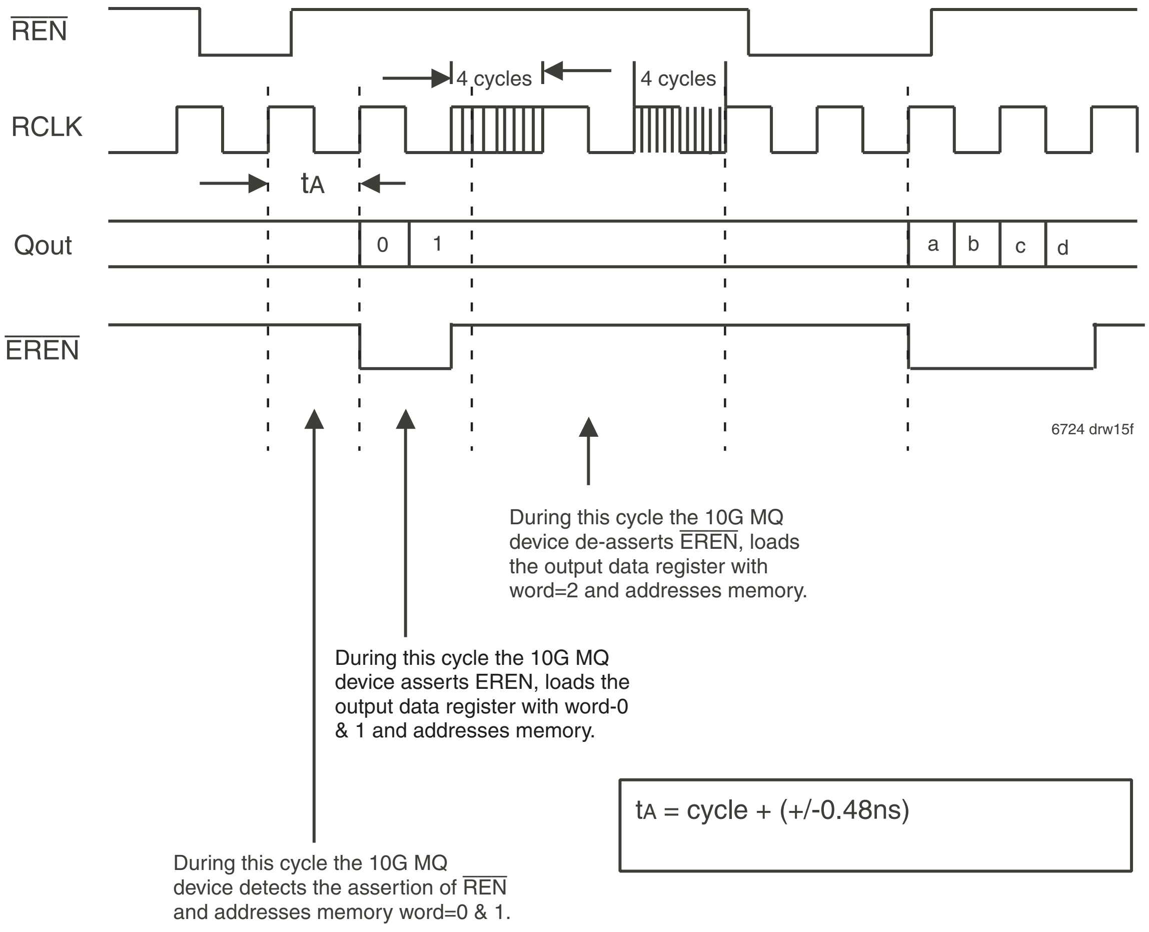
During this cycle the 10G MQ device asserts EREN, loads the output data register with word-0 then word=1.

During this cycle the 10G MQ device detects the assertion of $\overline{\text{REN}}$

Data latency = cycle + tA (3.6ns)

Figure 18. Output Data during a Queue Switch (DDR w/o PLL)

Parameters: DDR Mode, PLL=ON, 2 word read operation, WITH QUEUE SWITCH to NQ and QUEUE SWITCH BACK TO ORIGINAL QUEUE, 2 word read operation



During this cycle the 10G MQ device de-asserts $\overline{\text{EREN}}$, loads the output data register with word=2 and addresses memory.

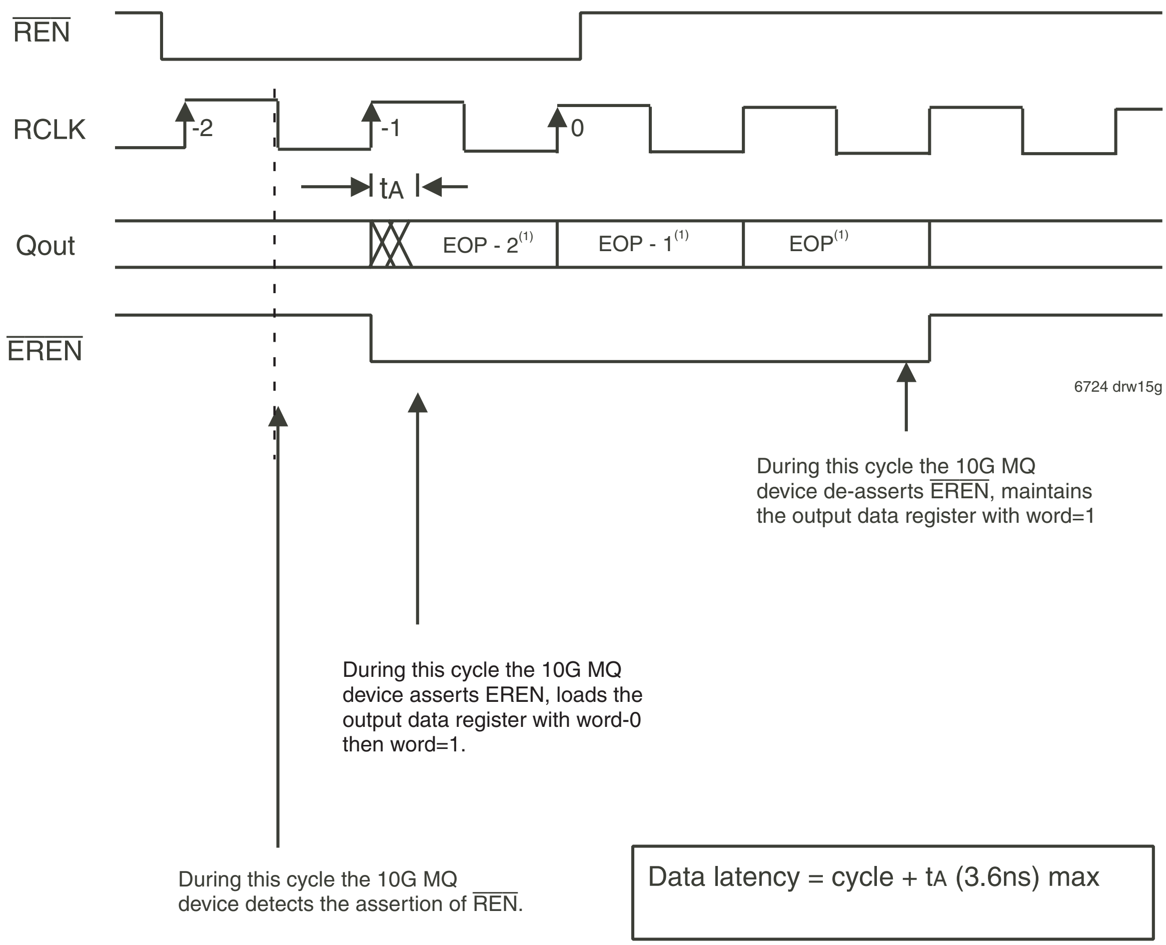
During this cycle the 10G MQ device asserts EREN, loads the output data register with word-0 & 1 and addresses memory.

tA = cycle + (+/-0.48ns)

During this cycle the 10G MQ device detects the assertion of $\overline{\text{REN}}$ and addresses memory word=0 & 1.

Figure 19. Output Data during two Queue Switches (DDR w/ PLL)

Parameters: SDR Mode, PLL=OFF, 3 word read operation, edge aligned data/clock, pause with No queue switch.



During this cycle the 10G MQ device de-asserts $\overline{\text{EREN}}$, maintains the output data register with word=1

During this cycle the 10G MQ device asserts EREN, loads the output data register with word-0 then word=1.

During this cycle the 10G MQ device detects the assertion of $\overline{\text{REN}}$.

Data latency = cycle + tA (3.6ns) max

NOTE:

1. Application specific.

Figure 20. Output Data during two Queue Switches (DDR w/o PLL)

TABLE 11 — BACKUP USAGE WHEN RE-ENTERING A QUEUE

Operating Frequency Range	Mode of Operation	Clock used by receiving device to latch data	Method to Stop Read Port operation	-1 word MQ backup required
1Mhz to 166Mhz	SDR PLL=OFF	Read Clock	Word Count	N
1Mhz to 166Mhz	SDR PLL=OFF	Read Clock	EOP Processing	Y (BOI)
83.33Mhz to 166Mhz	SDR PLL=ON	/ERCLK	Word Count	N
83.33Mhz to 166Mhz	DDR PLL=ON	/ERCLK	EOP Processing	Y (BOI)
1Mhz to 100Mhz	DDR PLL=OFF	Read Clock	Word Count	N
1Mhz to 100Mhz	SDR PLL=OFF	Read Clock	EOP Process	Y (BOI)
90Mhz to 100Mhz	SDR PLL=OFF	ERCLK	Word Count	N
90Mhz to 100Mhz	SDR PLL=OFF	ERCLK	EOP Process	Y (BOI)

SWITCHING QUEUES ON THE READ PORT

The IDT72P51767/72P51777 multi-queue flow-control devices can be configured up to a maximum of 128 queues. Data is read from a queue using the Data Output (Qout) bus, Read Clock (RCLK) and Read Enable (REN) signals. Selecting a queue on the read port occurs by placing the queue address on the Read Address bus (RDADD) during a rising edge of RCLK while Read Address Enable (RADEN) is HIGH. For reference, the state of Read Enable (REN) is a "Don't Care" during a read port queue selection. REN has significance during the queue mark operation. Selecting a queue requires 4 RCLK cycles. Refer to Figure 21, *Read Port Switching Queues Signal Sequence*.

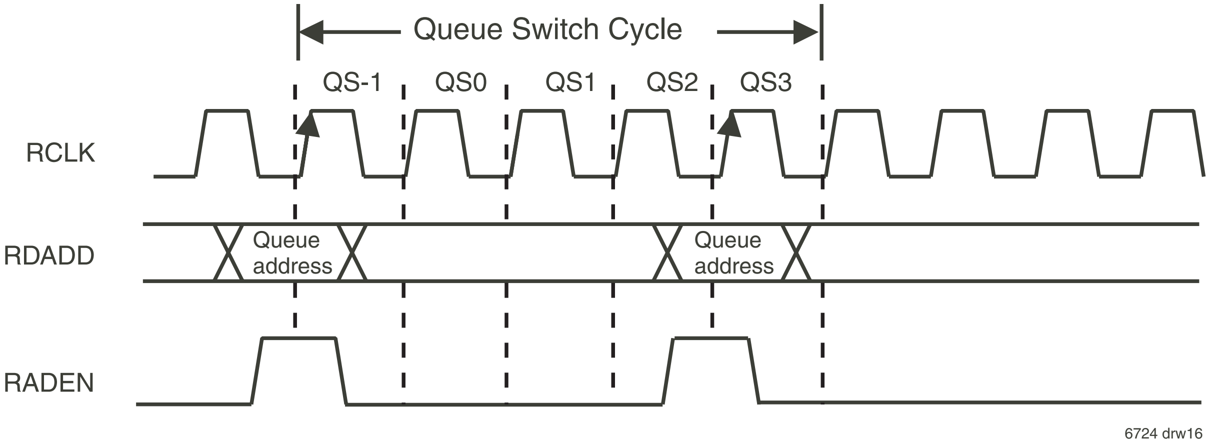
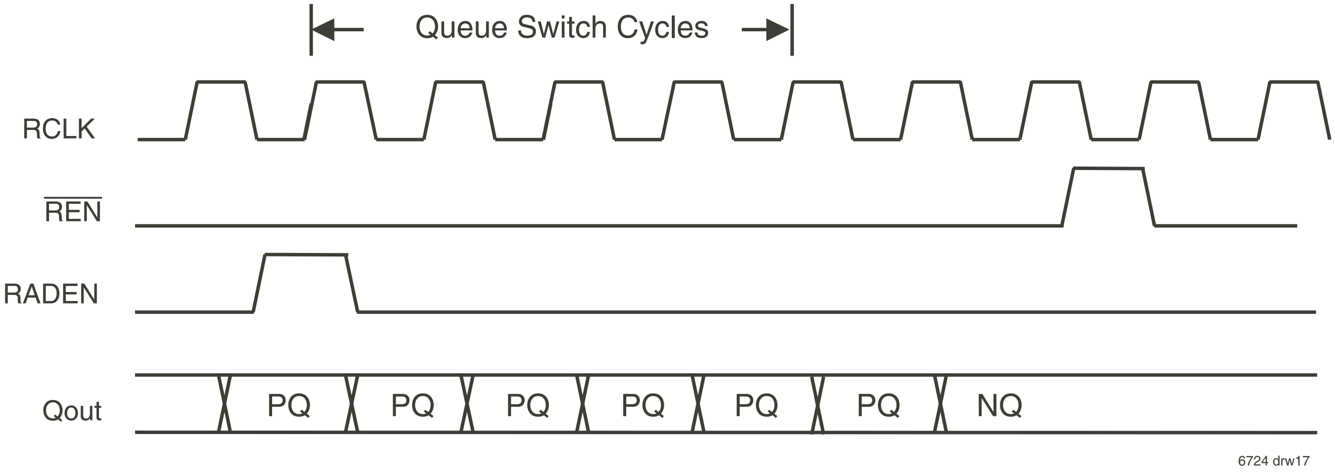


Figure 21. Read Port Switching Queues Signal Sequence

The IDT72P51767/72P51777 multi-queue flow-control device supports changing (switching) queues every four (4) clock cycles. To switch from the Present Queue (PQ) to another queue requires a queue address to be placed on the Read Address Bus (RDADD) bus and a rising edge of Read Clock (RCLK) and Read Address Enable (RADEN) is HIGH. There are no restrictions as to the order to which queues are selected or switched into or out of.

For maximum efficiency, during the 4 clock cycles required to switch queues the IDT72P51767/72P51777 multi-queue flow-control device can continue to read from the Present Queue (PQ). The Present Queue is defined as the current selected queue. Refer to Figure 22. Switching Queues Bus Efficiency.



NOTE:
PQ = Present Queue
NQ = Next Queue

Figure 22. Switching Queues Bus Efficiency

SIMULTANEOUS QUEUE SWITCHING

The IDT72P51767/72P51777 multi-queue flow-control device supports reading and writing from either the same queue or from different queues. The device also supports simultaneous queue switching on the read and write ports. The simultaneous queue switching may occur with either the Read Clock and Write Clock synchronous or asynchronous to each other. For reference refer to Figure 23, *Simultaneous Queue Switching*.

The multi-queue flow-control device requires 4 clock cycles to switch queues on the read port, refer to Table 12, Read Queue Switch Operation for a detailed description of each queue switch clock cycles.

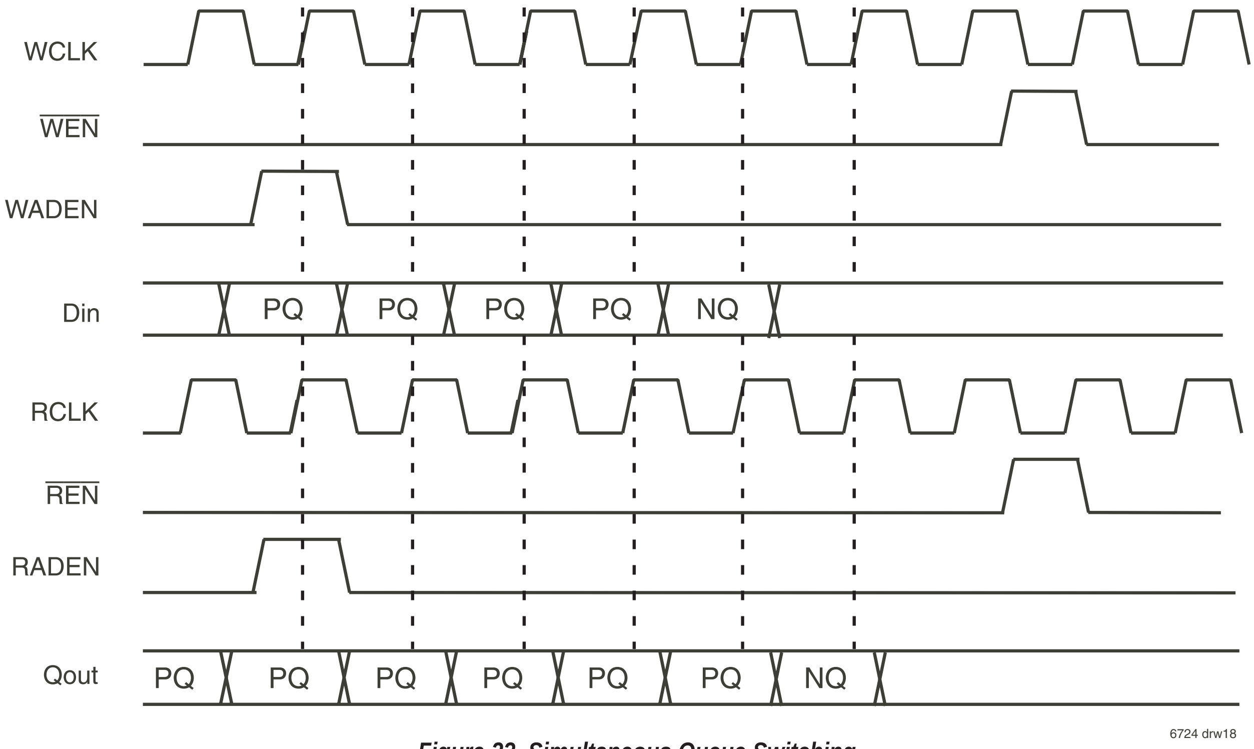


Figure 23. Simultaneous Queue Switching

TABLE 12 — READ QUEUE SWITCH OPERATION

Queue Switch Cycle	IDT Mode
QS-1	Queue Switch Initiated, Re-read/No Re-read selection
QS0	Queue MARK / Un-MARK
QS1	—
QS2	• $\overline{\text{PAE}}$ signal updated for Next Queue (NQ) • Empty Flag ($\overline{\text{EF}}$) updated for NQ
QS3	Start of Read Data Operation

TABLE 13 — SAME QUEUE SWITCH

PQ	NQ	Supported	Comment
Not Marked	Not Marked	Yes	Queue Switch is ignored
Not Marked	Marked	Yes	Add Mark to current queue
Marked	Not Marked, No Reread	Not Allowed	
Marked	Not Marked, Reread	Yes	Remove Mark
Marked	Marked, No Reread	Not Allowed	
Marked	Marked, Reread	Yes	Keep Mark

Legend:
PQ = Present Queue
NQ = Next Queue

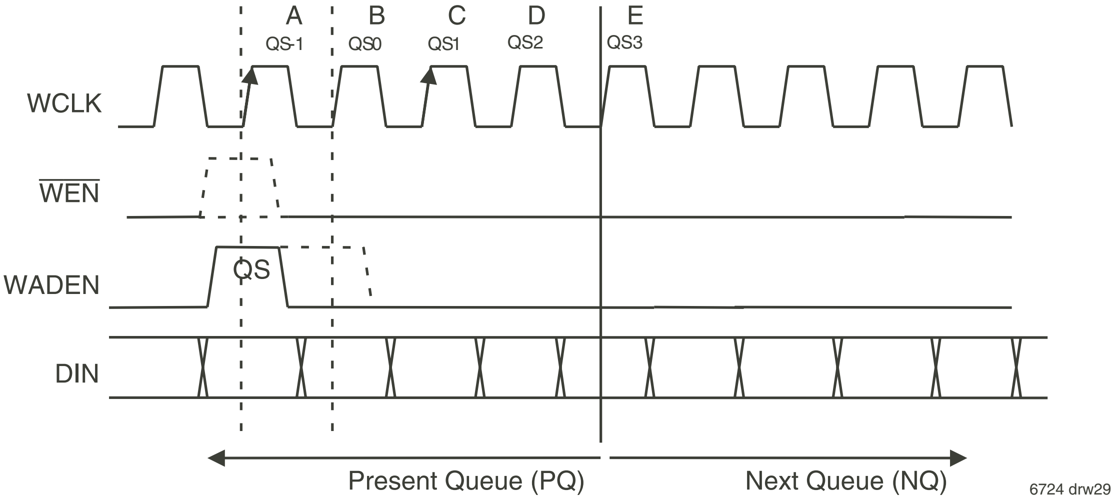
QUEUE MARKing

The overall intent of the MARK function is to provide the ability to either re-write and/or re-read information that is stored into a queue.

A queue can be MARKed by either the write port or the read port. The MARK operation is port independent. The same queue can be marked by the write port and the read port simultaneously. Only the active queue can be MARKed, multiple queues can NOT be MARKed by a port. A port (write or read) may only designate one queue MARKed at a time. Upon a queue switch a decision must be made as to whether to return to the Marked location or the last access address.

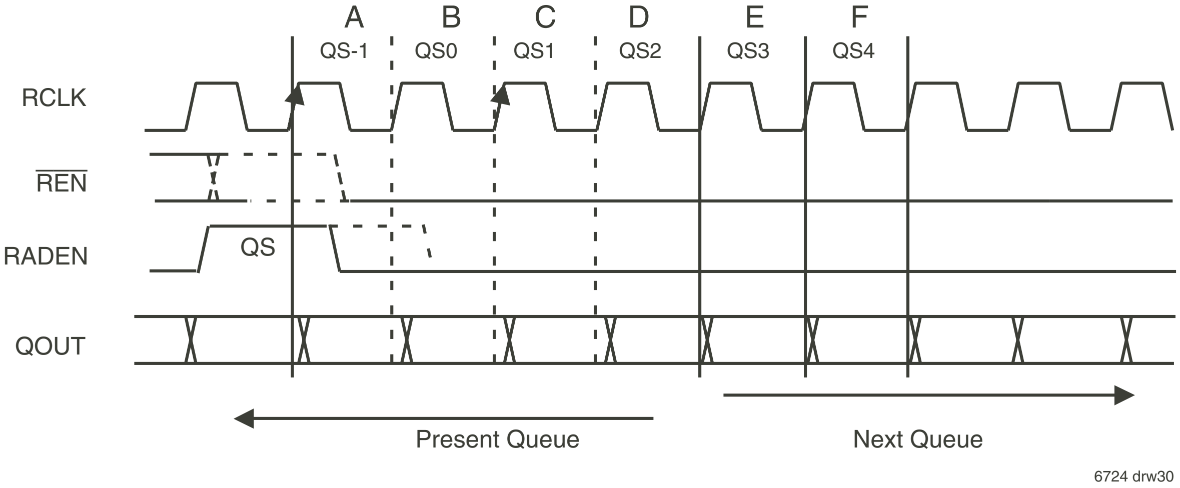
MARK AND REWRITE/ MARK AND REREAD

The MARK functionality operates in any mode combination (BOI mode, IDT Standard Mode). Queues on the Write Port are MARKed using the WCLK & WADEN signals. Queues on the Read Port are MARKed using the RCLK and RADEN signals. Refer to the following timing diagrams for additional queue MARK details. Refer to Figure 24 and 25 for further information.



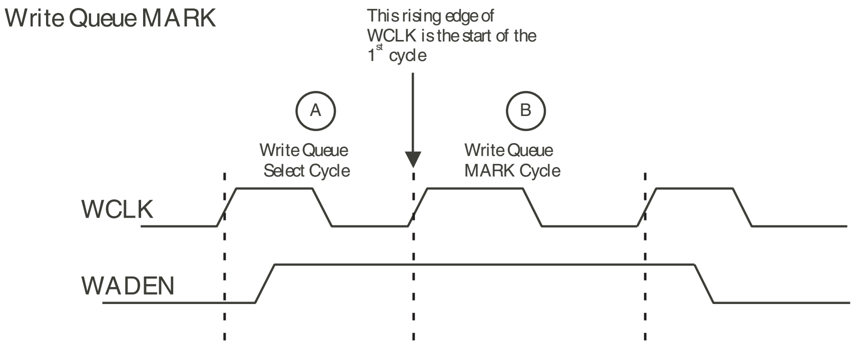
- @QS-1, if WEN=0 and WADEN=1, PQ will be updated in QS0,1, and 2, and NQ data will be written in QS3.
- @QS-1, if WEN_N=1 and WADEN=1, there is no update for PQ during QS0-QS2. Next time PQ is switched back, data will be written into last update location (rewrite).
- @QS0, WADEN status is used to determine if a "mark" is requested for NQ. If WADEN=1 in QS0, NQ will be marked. In IDT mode, the first NQ position after QS is marked (latch WFCR values before QS3), data can't be read out beyond this location.
- @QS0, if WADEN=0, NQ is not marked.

Figure 24. MARK and Re-Write Sequence



- @QS-1, if REN=0 and RADEN=1, Present Queue will be updated in QS0, QS1, and QS2, and the data from the Next Queue (NQ) will be available in QS4.
- @QS-1, if REN=1 and RADEN=1, Present Queue will not be updated QS0-QS2. The Next time PQ is selected, the data will be from the last updated location.
- @QS0, RADEN status is used to determine if a "mark" is requested for NQ. If RADEN=1 in QS0, NQ will be marked. In IDT mode, first NQ position after QS is marked (latch RFCR values before QS3), data can't overwrite this location.

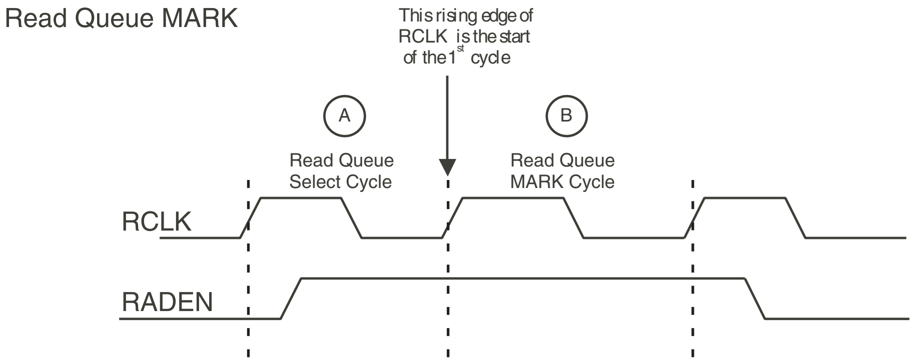
Figure 25. MARK and Re-Read Sequence



WADEN		ACTION
A	B	
1	1	Selects the Queue and MARK the Queue
1	0	Selects a Queue

6724 drw36

Figure 26. MARKing a Queue - Write Queue MARK



RADEN		ACTION
A	B	
1	1	Selects the Queue and MARK the Queue
1	0	Selects a Queue

6724 drw37

Figure 27. MARKing a Queue - Read Queue MARK

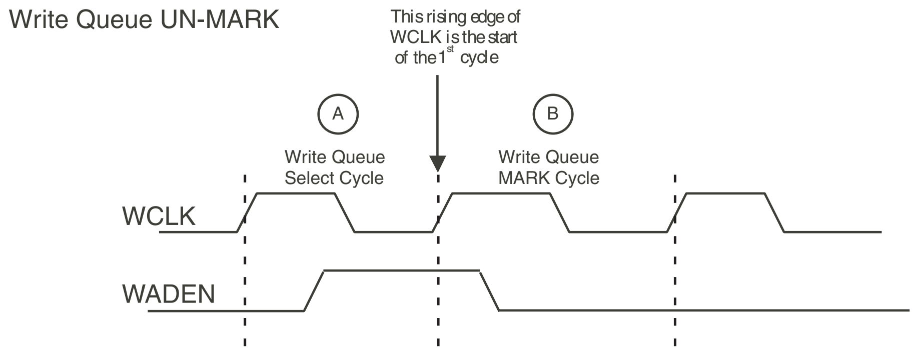
MARK Operational Notes:

Write Port
- MARKing can only occur during a Queue switch cycle
- The entire Queue is MARKed at a time.
- MARK is used to mark the first location of the Queue.
- MARK can NOT be moved within the queue.

Read Port
- MARKing can only occur during a Queue switch cycle
- Only the first location of the Queue can be MARKed.
- The MARK can NOT be moved location to location within the queue.

Un-MARKing a Queue

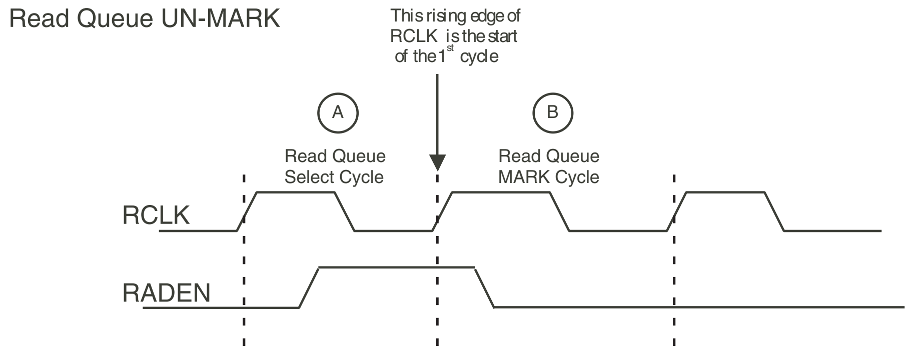
UN-MARKing a Queue



WADEN		ACTION
A	B	
1	0	Selects a Queue and UN-MARK the Queue

6724 drw38

Figure 28. UN-MARKing a Queue - Write Queue UN-MARK



RADEN		ACTION
A	B	
1	0	Selects a Queue and UN-MARK the Queue

6724 drw39

Figure 29. UN-MARKing a Queue - Read Queue UN-MARK

UN-MARK Operational Notes:

Write Port

- Un-MARKing can only occur during a Queue switch cycle.
- UN-MARKing a Queue can be accomplished by either switching to the same queue or switching to another queue.
- Note only 1 queue can be marked at any given time.
- The MARK can NOT be moved location to location within the queue.

Read Port

- Un-MARKing can only occur during a Queue switch cycle.
- UN-MARKing a Queue can be accomplished by either switching to the same queue or switching to another queue.
- Note only 1 queue can be marked at any given time.
- The MARK can NOT be moved location to location within the queue.

Leaving a MARK Active

During a Queue switch the value of $\overline{\text{WEN}}$ for the write port and $\overline{\text{REN}}$ for the read port determines whether the MARK remains active or is de-activated.

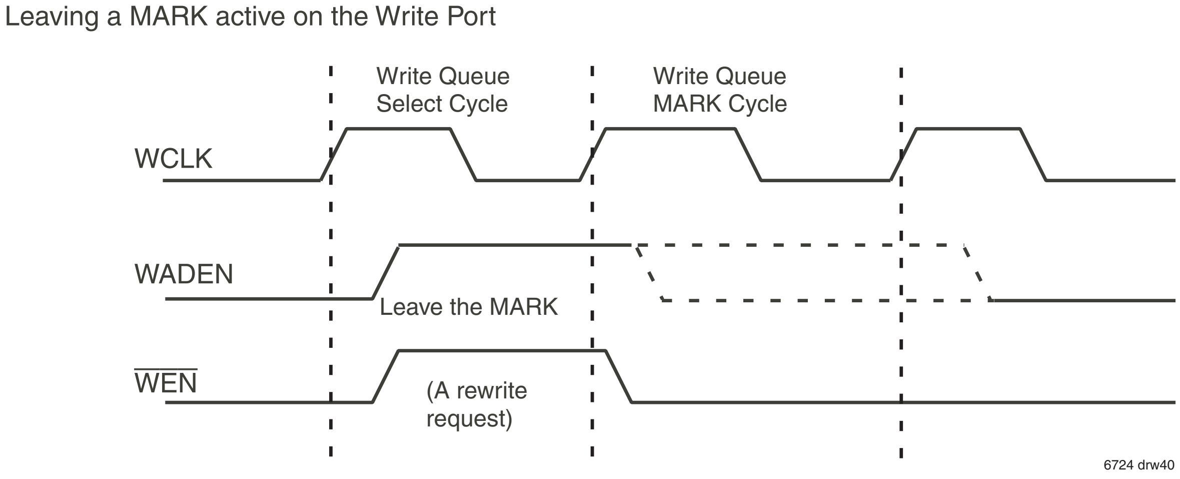


Figure 30. Leaving a MARK active on the Write Port

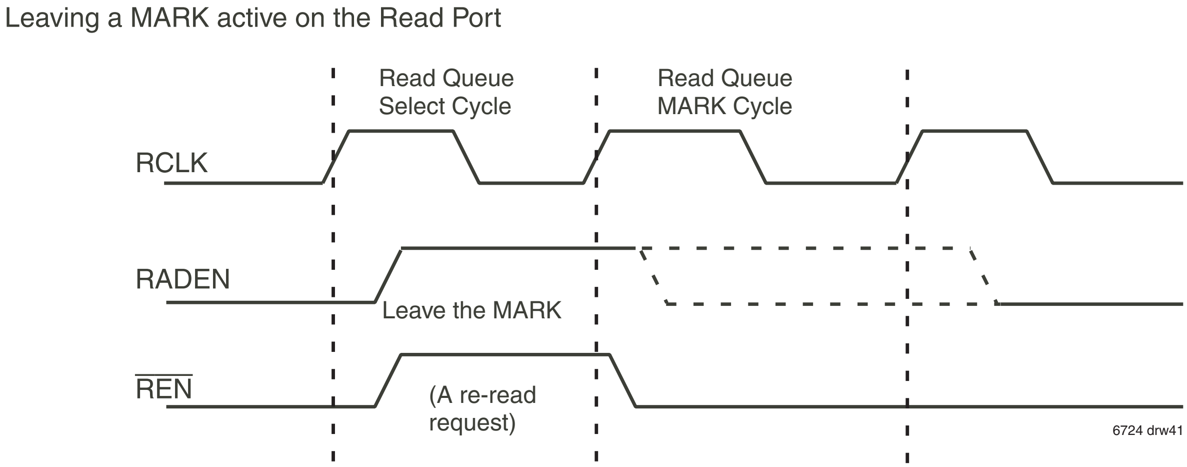


Figure 31. Leaving a MARK active on the Read Port

Inactivating a MARK

During a Queue switch the value of $\overline{\text{WEN}}$ for the write port and $\overline{\text{REN}}$ for the read port determines whether the MARK remains active or is de-activated.

Inactivating a MARK on the Write Port

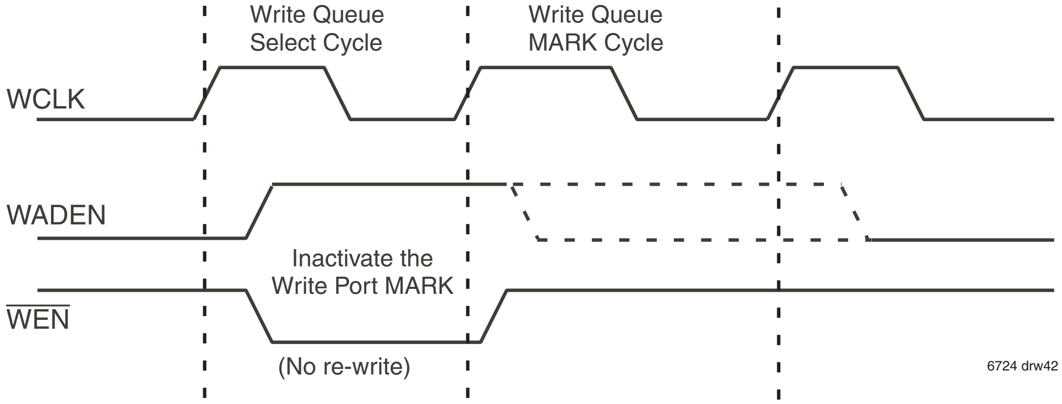


Figure 32. Inactivating a MARK on the Write Port Active

Inactivating a MARK on the Read Port

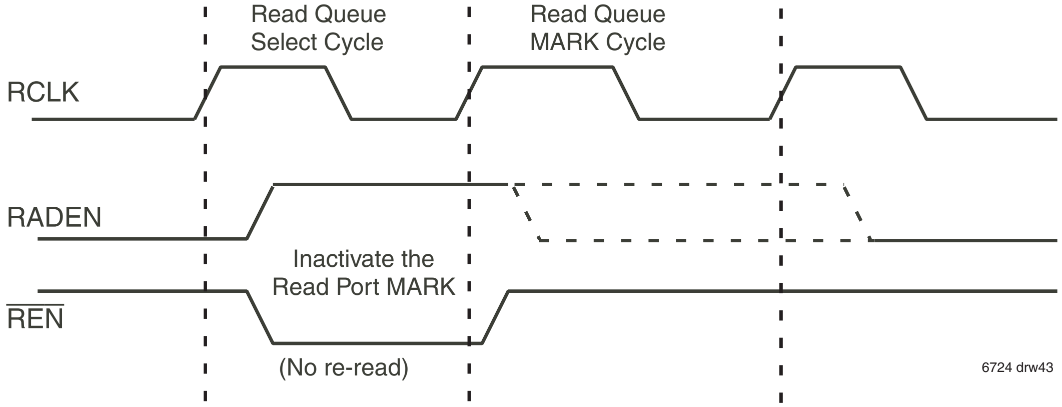
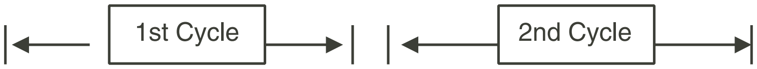


Figure 33. Inactivating a MARK on the Read Port Active

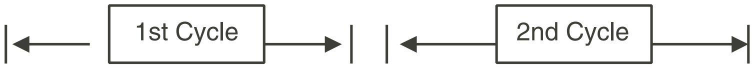
Write Cycle



Action	$\overline{\text{WEN}}$ (active LOW)	WADEN (active HIGH)		$\overline{\text{WEN}}$ (active LOW)	WADEN (active HIGH)
NO Operation	0	0		0	0
Selects a Queue	0	1		0	1
NO Operation	1	0		1	0
NO Operation	1	1		1	1

6724 drw44

Read Cycle



Action	$\overline{\text{REN}}$ (active LOW)	RADEN (active HIGH)		$\overline{\text{REN}}$ (active LOW)	RADEN (active HIGH)
NO Operation	0	0		0	0
Selects a Queue	0	1		0	1
NO Operation	1	0		1	0
NO Operation	1	1		1	1

6724 drw45

FLAG DESCRIPTION

$\overline{PAF}$n FLAG BUS OPERATION

The IDT72P51767/72P51777 multi-queue flow-control device can be configured for up to 128 queues, each queue having its own almost full status. An active queue has its flag status output to the discrete flags, $\overline{FF}$ and $\overline{PAF}$, on the write port. Queues that are not selected for a write operation can have their $\overline{PAF}$ status monitored via the $\overline{PAF}$n bus. The $\overline{PAF}$n flag bus is 8 bits wide, so that 8 queues at a time can have their status output to the bus. If 9 or more queues are setup within a device then there are 2 methods by which the device can share the bus between queues, “Direct” mode and “Polled” mode depending on the state of the FM (Flag Mode) input during a Master Reset. If 8 or less queues are setup within a device then each will have its own dedicated output from the bus. If 8 or less queues are setup in single device mode, it is recommended to configure the $\overline{PAF}$n bus to polled mode as it does not require using the write address (WRADD).

FULL FLAG OPERATION

The multi-queue flow-control device provides a single Full Flag output, $\overline{FF}$. The $\overline{FF}$ flag output provides a full status of the queue currently selected on the write port for write operations. Internally the multi-queue flow-control device monitors and maintains a status of the full condition of all queues within it, however only the queue that is selected for write operations has its full status output to the $\overline{FF}$ flag. This dedicated flag is often referred to as the “active queue full flag”.

When queue switches are being made on the write port, the $\overline{FF}$ flag output will switch to the new queue and provide the user with the new queue status, on the 3rd cycle after a new queue selection is made. The user then has a full status for the new queue one cycle ahead of the WCLK rising edge that data can be written into the new queue. That is, a new queue can be selected on the write port via the WRADD bus, WADEN enable and a rising edge of WCLK. On the 4th rising edge of WCLK, the $\overline{FF}$ flag output will show the full status of the newly selected queue. On the forth rising edge of WCLK following the queue selection, data can be written into the newly selected queue provided that data and enable setup & hold times are met.

Note, the $\overline{FF}$ flag will provide status of a newly selected queue three WCLK cycle after queue selection, which is one cycle before data can be written to that queue. This prevents the user from writing data to a queue that is full, (assuming that a queue switch has been made to a queue that is actually full).

The $\overline{FF}$ flag is synchronous to the WCLK and all transitions of the $\overline{FF}$ flag occur based on a rising edge of WCLK. Internally the multi-queue device monitors and keeps a record of the full status for all queues. It is possible that the status of a $\overline{FF}$ flag maybe changing internally even though that flag is not the active queue flag (selected on the write port). A queue selected on the read port may experience a change of its internal full flag status based on read operations.

See Figure 42, *SDR Write Queue Select, Write Operation and Full Flag Operation* and Figure 45, *Full Flag Timing in Expansion Configuration* for timing information.

EXPANSION CONFIGURATION - FULL FLAG OPERATION

When multi-queue devices are connected in Expansion configuration the $\overline{FF}$ flags of all devices should be connected together, such that a system controller monitoring and managing the multi-queue devices write port only looks at a single $\overline{FF}$ flag (as opposed to a discrete $\overline{FF}$ flag for each device). This $\overline{FF}$ flag is only pertinent to the queue being selected for write operations at that time. Remember, that when in expansion configuration only one multi-queue device can be written to at any moment in time, thus the $\overline{FF}$ flag provides status of the active queue on the write port.

This connection of flag outputs to create a single flag requires that the $\overline{FF}$ flag output have a High-Impedance capability, such that when a queue selection is made only a single device drives the $\overline{FF}$ flag bus and all other $\overline{FF}$ flag outputs connected to the $\overline{FF}$ flag bus are placed into High-Impedance. The user does not have to select this High-Impedance state, a given multi-queue flow-control device will automatically place its $\overline{FF}$ flag output into High-Impedance when none of its queues are selected for write operations.

When queues within a single device are selected for write operations, the $\overline{FF}$ flag output of that device will maintain control of the $\overline{FF}$ flag bus. Its $\overline{FF}$ flag will simply update between queue switches to show the respective queue full status.

The multi-queue device places its $\overline{FF}$ flag output into High-Impedance based on the 1-3 bit ID code (1 if two multi-queue are configured with a maximum total of 256 queues, 2 if four devices are used totalling a maximum of 256 queues, and 3 if there are up to eight devices with a maximum total of 256 queues) found in the 1-3 most significant bits of the write queue address bus, WRADD. If the 1-3 most significant bits of WRADD match the 1-3 bit ID code setup on the static inputs, ID0, ID1 and ID2 then the $\overline{FF}$ flag output of the respective device will be in a Low-Impedance state. If they do not match, then the $\overline{FF}$ flag output of the respective device will be in a High-Impedance state. See Figure 45, *Full Flag Timing in Expansion Configuration* for details of flag operation, including when more than one device is connected in expansion.

EMPTY FLAG OPERATION ($\overline{EF}$)

The multi-queue flow-control device provides a single Empty flag output, $\overline{EF}$. The rising edge of an RCLK cycle that places new data onto the output register of the read port. Internally the multi-queue flow-control device monitors and maintains a status of the empty condition of all queues within it.

See Figure 46, *SDR Read Queue Select, Read Operation* for details of the timing.

EXPANSION – EMPTY FLAG OPERATION

When multi-queue devices are connected in Expansion configuration, the $\overline{EF}$ flags of all devices should be connected together, such that a system controller monitoring and managing the multi-queue devices read port only looks at a single $\overline{EF}$ flag (as opposed to a discrete $\overline{EF}$ flag for each device). This $\overline{EF}$ flag is only pertinent to the queue being selected for read operations at that time. Remember, that when in expansion configuration only one multi-queue device can be read from at any moment in time, thus the $\overline{EF}$ flag provides status of the active queue on the read port.

This connection of flag outputs to create a single flag requires that the $\overline{EF}$ flag output have a High-Impedance capability, such that when a queue selection is made only a single device drives the $\overline{EF}$ flag bus and all other $\overline{EF}$ flag outputs connected to the $\overline{EF}$ flag bus are placed into High-Impedance. The user does not have to select this High-Impedance state, a given multi-queue flow-control device will automatically place its $\overline{EF}$ flag output into High-Impedance when none of its queues are selected for read operations.

When queues within a single device are selected for read operations, the $\overline{EF}$ flag output of that device will maintain control of the $\overline{EF}$ flag bus. Its $\overline{EF}$ flag will simply update between queue switches to show the respective queue status.

The multi-queue device places its $\overline{EF}$ flag output into High-Impedance based on the 1-3 bit ID code (1 if two multi-queue are configured with a maximum total of 256 queues, 2 if four devices are used totalling a maximum of 256 queues, and 3 if there are up to eight devices with a maximum total of 256 queues) found in the 3 most significant bits of the read queue address bus, RDADD. If the 3 most significant bits of RDADD match the 1-3 bit ID code setup on the static inputs, ID0, ID1 and ID2 then the $\overline{EF}$ flag output of the respective device will be in a Low-Impedance state. If they do not match, then the $\overline{EF}$ flag output of the respective device will be in a High-Impedance state.

ALMOST FULL FLAG

As previously mentioned the multi-queue flow-control device provides a single Programmable Almost Full flag output, PAF. The PAF flag output provides a status of the almost full condition for the active queue currently selected on the write port for write operations. Internally the multi-queue flow-control device monitors and maintains a status of the almost full condition of all queues within it, however only the queue that is selected for write operations has its full status output to the PAF flag. This dedicated flag is often referred to as the "active queue almost full flag". The position of the PAF flag boundary within a queue can be at any point within that queues depth. This location can be user programmed via the serial port or one of the default values (8 or 128) can be selected if the user has performed default programming.

As mentioned, every queue within a multi-queue device has its own almost full status, when a queue is selected on the write port, this status is output via the PAF flag. The PAF flag value for each queue is programmed during multi-queue device programming (along with the number of queues, queue depths and almost empty values). The PAF offset value, m, for a respective queue can be programmed to be anywhere between '0' and 'D', where 'D' is the total memory depth for that queue. The PAF value of different queues within the same device can be different values.

When queue switches are being made on the write port, the PAF flag output will switch to the new queue and provide the user with the new queue status, on the third cycle after a new queue selection is made, on the same WCLK cycle that data can actually be written to the new queue. That is, a new queue can be selected on the write port via the WRADD bus, WADEN enable and a rising edge of WCLK. On the third rising edge of WCLK following a queue selection, the PAF flag output will show the full status of the newly selected queue. The PAF is flag output is double register buffered, so when a write operation occurs at the almost full boundary causing the selected queue status to go almost full the PAF will go LOW 3 WCLK cycles after the write. The same is true when a read occurs, there will be a 3 WCLK cycle delay after the read operation.

So the PAF flag delay from a write operation to PAF flag LOW is 3 WCLK + tWAF. The delay from a read operation to PAF flag HIGH is tSKEW2 + WCLK + tWAF.

Note, if tSKEW is violated there will be one added WCLK cycle delay.

The PAF flag is synchronous to the WCLK and all transitions of the PAF flag occur based on a rising edge of WCLK. Internally the multi-queue device monitors and keeps a record of the almost full status for all queues. It is possible that the status of a PAF flag maybe changing internally even though that flag is not the active queue flag (selected on the write port). A queue selected on the read port may experience a change of its internal almost full flag status based on read operations. The multi-queue flow-control device also provides a duplicate of the PAF flag on the PAF[7:0] flag bus, this will be discussed in detail in a later section of the data sheet.

See Figures 23 and 24 for Almost Full flag timing and queue switching.

ALMOST EMPTY FLAG

As previously mentioned the multi-queue flow-control device provides a single Programmable Almost Empty flag output, PAE. The PAE flag output provides a status of the almost empty condition for the active queue currently selected on the read port for read operations. Internally the multi-queue flow-control device monitors and maintains a status of the almost empty condition of all queues within it, however only the queue that is selected for read operations has its empty status output to the PAE flag. This dedicated flag is often referred to as the "active queue almost empty flag". The position of the PAE flag boundary within a queue can be at any point within that queues depth. This location can be user programmed via the serial port or one of the default values (8 or 128) can be selected if the user has performed default programming.

As mentioned, every queue within a multi-queue device has its own almost empty status, when a queue is selected on the read port, this status is output via the PAE flag. The PAE flag value for each queue is programmed during multi-queue device programming (along with the number of queues, queue depths and almost full values). The PAE offset value, n, for a respective queue can be programmed to be anywhere between '0' and 'D', where 'D' is the total memory depth for that queue. The PAE value of different queues within the same device can be different values.

When queue switches are being made on the read port, the PAE flag output will switch to the new queue and provide the user with the new queue status, on the third cycle after a new queue selection is made, on the same RCLK cycle that data actually falls through to the output register from the new queue. That is, a new queue can be selected on the read port via the RDADD bus, RADEN enable and a rising edge of RCLK. On the third rising edge of RCLK following a queue selection, the data word from the new queue will be available at the output register and the PAE flag output will show the empty status of the newly selected queue. The PAE is flag output is double register buffered, so when a read operation occurs at the almost empty boundary causing the selected queue status to go almost empty the PAE will go LOW 3 RCLK cycles after the read. The same is true when a write occurs, there will be a 3 RCLK cycle delay after the write operation.

So the PAE flag delay from a read operation to PAE flag LOW is 3 RCLK + tRAE. The delay from a write operation to PAE flag HIGH is tSKEW2 + RCLK + tRAE.

Note, if tSKEW is violated there will be one added RCLK cycle delay.

The PAE flag is synchronous to the RCLK and all transitions of the PAE flag occur based on a rising edge of RCLK. Internally the multi-queue device monitors and keeps a record of the almost empty status for all queues. It is possible that the status of a PAE flag maybe changing internally even though that flag is not the active queue flag (selected on the read port). A queue selected on the write port may experience a change of its internal almost empty flag status based on write operations. The multi-queue flow-control device also provides a duplicate of the PAE flag on the PAE[7:0] flag bus, this will be discussed in detail in a later section of the data sheet.

See Figures 25 and 26 for Almost Empty flag timing and queue switching.

PAFn - DIRECT BUS

If FM is LOW at master reset then the PAFn bus operates in Direct (addressed) mode. In direct mode the user can address the status word of queues they require and it will be placed on to the PAFn bus. For example, consider the operation of the PAFn bus when 26 queues have been setup. To output status of the first status word, Queue[0:7] the WRADD bus is used in conjunction with the FSTR (PAF flag strobe) input and WCLK. The address present on the 4 least significant bits of the WRADD bus with FSTR HIGH will be selected as the status word address on a rising edge of WCLK. To address status word 0, Queue[0:7] the WRADD bus should be loaded with "0010000", the PAFn bus will change status to show the new status word selected 1 WCLK cycle after status word selection. PAFn[0:7] gets status of queues, Queue[0:7] respectively.

To address status word 1, Queue[8:15], the WRADD address is "00100001". PAFn[0:7] gets status of queues, Queue[8:15] respectively. To address the 2nd status word, Queue[16:23], the WRADD address is "00100010". PAF[0:7] gets status of queues, Queue[16:23] respectively. To address the 3rd status word, Queue[24:31], the WRADD address is "00100011". PAF[0:1] gets status of queues, Queue[24:25] respectively. Remember, only 26 queues were setup, so when status word 4 is selected the unused outputs PAF[2:7] will be don't care states.

Note, that if a read or write operation is occurring to a specific queue, say queue 'x' on the same cycle as a status word switch which will include the queue 'x', then there may be an extra WCLK cycle delay before that queues status is correctly shown on the respective output of the PAFn bus. However, the active PAF flag will show correct status at all times.

Status words can be selected on consecutive clock cycles, that is the status word on the PAFn bus can change every WCLK cycle. Also, data present on the input bus, Din, can be written into a Queue on the same WCLK rising edge that a status word is being selected, the only restriction being that a write queue selection and PAFn status word selection cannot be made on the same cycle.

If 8 or less queues are setup then queues, Queue[0:7] have their PAF status output on PAF[0:7] constantly.

When the multi-queue devices are connected in expansion of more than one device the PAFn busses of all devices are connected together, when switching between status words of different devices the user must utilize the 1-3 most significant bits of the WRADD address bus (as well as the 2 LSB's). These 1-3 MSb's correspond to the device ID inputs, which are the static inputs, ID0, ID1 & ID2.

Please refer to Figure 57 **PAF***n - Direct Mode Status Word Selection* for timing information. Also refer to Table 8, *Write Address Bus, WRADD*.

PAFn – POLLED BUS

If FM is HIGH at master reset then the PAFn bus operates in Polled (looped) mode. In polled mode the PAFn bus only cycles through the number of status words required to display the status of the number of queues that have been setup in the part. Every rising edge of the WCLK causes the next status word to be loaded on the PAFn bus. The device configured as the master (MAST input tied HIGH), will take control of the PAFn after MRS goes LOW. For the whole WCLK cycle that the first status word is on PAFn the FSYNC (PAFn bus sync) output will be HIGH, for all other status words, this FSYNC output will be LOW. This FSYNC output provides the user with a mark with which they can synchronize to the PAFn bus, FSYNC is always HIGH for the WCLK cycle that the first status word of a device is present on the PAFn bus.

When devices are connected in expansion configuration, only one device will be set as the Master (ID = '000'), MAST input tied HIGH, all other devices will have MAST tied LOW. The master device is the first device to take control of the PAFn bus and will place its first status word on the bus on the rising edge of WCLK. For the next n WCLK cycles (n= number of queues divided by 8 with n being increased by one for any remainder) the master device will maintain control of the PAFn bus and cycle its status words through it, all other devices hold their PAFn outputs in High-Impedance. When the master device has cycled all of its status words it passes a token to the next device in the chain and that device assumes control of the PAFn bus and then cycles its status words and so on, the PAFn bus control token being passed on from device to device. This token passing is done via the FXO outputs and FXI inputs of the devices ("PAF Expansion Out" and "PAF Expansion In"). The FXO output of the master device connects to the FXI of the second device in the chain and the FXO of the second connects to the FXI of the third and so on. The final device in a chain has its FXO connected to the FXI of the first device, so that once the PAFn bus has cycled through all status words of all devices, control of the PAFn will pass to the master device again and so on. The FSYNC of each respective device will operate independently and simply indicate when that respective device has taken control of the bus and is placing its first status word on to the PAFn bus.

When operating in single device mode the FXI input must be connected to the FXO output of the same device. In single device mode a token is still required to be passed into the device for accessing the PAFn bus.

Please refer to Figure 60, **PAF***n Bus – Polled Mode* for timing information.

PAEn FLAG BUS OPERATION

The IDT72P51767/72P51777 multi-queue flow-control device can be configured for up to 128 queues, each queue having its own almost empty/packet ready status. An active queue has its flag status output to the discrete flag, PAE, on the read port. Queues that are not selected for a read operation can have their PAE status monitored via the PAEn bus. The PAEn flag bus is 8 bits wide, so that 8 queues at a time can have their status output to the bus. If 9 or more queues are setup within a device then there are 2 methods by which the device can share the bus between queues, "Direct" mode and "Polled" mode depending on the state of the FM (Flag Mode) input during a Master Reset. If 8 or less queues are setup within a device then each will have its own dedicated output from the bus. If 8 or less queues are setup in single device mode, it is recommended to configure the PAFn bus to polled mode as it does not require using the write address (WRADD).

PAEn - DIRECT BUS

If FM is LOW at master reset then the PAEn bus operates in Direct (addressed) mode. In direct mode the user can address the status word of queues they require to be placed on to the PAEn bus. For example, consider the operation of the PAEn bus when 26 queues have been setup. To output status of the first status word, Queue[0:7] the RDADD bus is used in conjunction with the ESTR (PAE flag strobe) input and RCLK. The address present on the 2 least significant bits of the RDADD bus with ESTR HIGH will be selected as the status word address on a rising edge of RCLK. So to address status word 1, Queue[0:7] the RDADD bus should be loaded with "xxxx0000", the PAEn bus will change status to show the new status word selected 1 RCLK cycle after status word selection. PAEn[0:7] gets status of queues, Queue[0:7] respectively.

To address the second status word, Queue[8:15], the RDADD address is "xxxx0001". PAEn[0:7] gets status of queues, Queue[8:15] respectively. To address the third status word, Queue[16:23], the RDADD address is "xxxx0010". PAE[0:7] gets status of queues, Queue[16:23] respectively. To address the fourth status word, Queue[24:31], the RDADD address is "xxxx0011". PAE[0:1] gets status of queues, Queue[24:25] respectively. Remember, only 26 queues were setup, so when status word 4 is selected the unused outputs PAE[2:7] will be don't care states.

Note, that if a read or write operation is occurring to a specific queue, say queue 'x' on the same cycle as a status word switch which will include the queue 'x', then there may be an extra RCLK cycle delay before that queues status is correctly shown on the respective output of the PAEn bus.

Status words can be selected on consecutive clock cycles, that is the status word on the PAEn bus can change every RCLK cycle. Also, data can be read out of a Queue on the same RCLK rising edge that a status word is being selected, the only restriction being that a read queue selection and PAEn status word selection cannot be made on the same RCLK cycle.

If 8 or less queues are setup then queues, Queue[0:7] have their PAE status output on PAE[0:7] constantly.

When the multi-queue devices are connected in expansion of more than one device the PAEn busses of all devices are connected together, when switching between status words of different devices the user must utilize the 3 most significant bits of the RDADD address bus (as well as the 2 LSB's). These 3 MSb's correspond to the device ID inputs, which are the static inputs, ID0, ID1 & ID2.

Please refer to Figure 56, **PAE***n - Direct Mode Status Word Selection* for timing information. Also refer to Table 9, *Read Address Bus, RDADD*.

TABLE 14 — FLAG OPERATION BOUNDARIES & TIMING

Empty Flag, $\overline{EF}$ Flag Boundary	
I/O Set-Up	**$\overline{EF}$ Boundary Condition**
In40 to out40 (Almost Empty Mode) (Both ports selected for same queue when the 1st Word is written in)	$\overline{EF}$ Goes HIGH after 1st Write (see note 1 below for timing)
In40 to out20 (Both ports selected for same queue when the 1st Word is written in)	$\overline{EF}$ Goes HIGH after 1st Write (see note 1 below for timing)
In20 to out40 (Both ports selected for same queue when the 1st Word is written in)	$\overline{EF}$ Goes HIGH after 1st Write (see note 1 below for timing)

NOTE:

1. **$\overline{EF}$ Timing**

Assertion:

Write to $\overline{EF}$ HIGH: tSKEW1 + RCLK + tROV

If tSKEW1 is violated there may be 1 added clock: tSKEW1 + 2 RCLK + tR$\overline{EF}$

De-assertion:

Read Operation to $\overline{EF}$ HIGH: tREF

2. In40 = SDR40 or DDR20

In20 = SDR20

Full Flag, $\overline{FF}$ Boundary	
I/O Set-Up	**$\overline{FF}$ Boundary Condition**
In40 to out40 (Both ports selected for same queue when the 1st Word is written in)	$\overline{FF}$ Goes LOW after D Writes (see note below for timing)
In40 to out40 (Write port only selected for queue when the 1st Word is written in)	$\overline{FF}$ Goes LOW after D Writes (see note below for timing)
In40 to out20 (Both ports selected for same queue when the 1st Word is written in)	$\overline{FF}$ Goes LOW after D Writes (see note below for timing)
In40 to out20 (Write port only selected for queue when the 1st Word is written in)	$\overline{FF}$ Goes LOW after D Writes (see note below for timing)
In20 to out40 (Both ports selected for same queue when the 1st Word is written in)	$\overline{FF}$ Goes LOW after ([D] x 2) Writes (see note below for timing)
In20 to out40 (Write port only selected for queue when the 1st Word is written in)	$\overline{FF}$ Goes LOW after (D x 2) Writes (see note below for timing)

NOTE:

D = Queue Depth

$\overline{FF}$ Timing

Assertion:

Write Operation to $\overline{FF}$ LOW: tWFF

De-assertion:

Read to $\overline{FF}$ HIGH: tSKEW1 + tWFF

If tSKEW1 is violated there may be 1 added clock: tSKEW1+WCLK +tWFF

1. In40 = SDR40 or DDR20

In20 = SDR20

Programmable Almost Full Flag, $\overline{PAF}$ & $\overline{PAF}$n Bus Boundary	
I/O Set-Up	**$\overline{PAF}$ & $\overline{PAF}$n Boundary**
in40 to out40 (Both ports selected for same queue when the 1st Word is written in until the boundary is reached)	$\overline{PAF}$/$\overline{PAF}$n Goes LOW after D-m Writes (see note below for timing)
in40 to out40 (Write port only selected for same queue when the 1st Word is written in until the boundary is reached)	$\overline{PAF}$/$\overline{PAF}$n Goes LOW after D-m Writes (see note below for timing)
in40 to out20	$\overline{PAF}$/$\overline{PAF}$n Goes LOW after D-m Writes (see below for timing)

NOTE:

D = Queue Depth

m = Almost Full Offset value.

1. In40 = SDR40 or DDR20

In20 = SDR20

$\overline{PAF}$ Timing

Assertion: Write Operation to $\overline{PAF}$ LOW: 3 WCLK + tWAF

De-assertion: Read to $\overline{PAF}$ HIGH: tSKEW2 + WCLK + tWAF

If tSKEW2 is violated there may be 1 added clock: tSKEW2 + 3 WCLK + tWAF

$\overline{PAF}$n Timing

Assertion: Write Operation to $\overline{PAF}$n LOW: 2 WCLK* + tPAF

De-assertion: Read to $\overline{PAF}$n HIGH: tSKEW3 + WCLK* + tPAF

If tSKEW3 is violated there may be 1 added clock: tSKEW3 + 3 WCLK* + tPAF

* If a queue switch is occurring on the write port at the point of flag assertion or de-assertion there may be one additional WCLK clock cycle delay.

TABLE 14 — FLAG OPERATION BOUNDARIES & TIMING (CONTINUED)

Programmable Almost Empty Flag, PAE Boundary	
I/O Set-Up	**PAE Assertion**
in40 to out40 (Both ports selected for same queue when the 1st Word is written in until the boundary is reached)	PAE Goes HIGH after n+1 Writes (see note below for timing)
in40 to out20 (Both ports selected for same queue when the 1st Word is written in until the boundary is reached)	PAE Goes HIGH after n+1 Writes (see note below for timing)
in20 to out40 (Both ports selected for same queue when the 1st Word is written in until the boundary is reached)	PAE Goes HIGH after ([n+1] x 2) Writes (see note below for timing)

NOTE:

n = Almost Empty Offset value.

1. In40 = SDR40 or DDR20
   In20 = SDR20

PAE Timing

Assertion:	Read Operation to PAE LOW: 3 RCLK + tRAE
De-assertion:	Write to PAE HIGH: tSKEW2 + RCLK + tRAE If tSKEW2 is violated there may be 1 added clock: tSKEW2 + 3 RCLK + tRAE

Programmable Almost Empty Flag Bus, PAEn Boundary	
I/O Set-Up	**PAEn Boundary Condition**
in40 to out40 (Both ports selected for same queue when the 1st Word is written in until the boundary is reached)	PAEn Goes HIGH after n+1 Writes (see note below for timing)
in40 to out40 (Write port only selected for same queue when the 1st Word is written in until the boundary is reached)	PAEn Goes HIGH after n+1 Writes (see note below for timing)
in40 to out20	PAEn Goes HIGH after n+1 Writes (see below for timing)
in20 to out40 (Both ports selected for same queue when the 1st Word is written in until the boundary is reached)	PAEn Goes HIGH after ([n+1] x 2) Writes (see note below for timing)
in20 to out40 (Write port only selected for same queue when the 1st Word is written in until the boundary is reached)	PAEn Goes HIGH after ([n+1] x 2) Writes (see note below for timing)

NOTE:

n = Almost Empty Offset value.

1. In40 = SDR40 or DDR20
   In20 = SDR20

PAEn Timing

Assertion:	Read Operation to PAEn LOW: 3 RCLK* + tPAE
De-assertion:	Write to PAEn HIGH: tSKEW3 + RCLK* + tPAE If tSKEW3 is violated there may be 1 added clock: tSKEW3 + 3 RCLK* + tPAE

* If a queue switch is occurring on the read port at the point of flag assertion or de-assertion there may be one additional RCLK clock cycle delay.

PAEn – POLLED BUS

If FM is HIGH at master reset then the PAEn bus operates in Polled (looped) mode. In polled mode the PAEn bus automatically cycles through the 4 status words within the device regardless of how many queues have been setup in the part. Every rising edge of the RCLK causes the next status word to be loaded on the PAEn bus. The device configured as the master (MAST input tied HIGH), will take control of the PAEn after MRS goes LOW. For the whole RCLK cycle that the first status word is on PAEn the ESYNC (PAEn bus sync) output will be HIGH, for all other status words, this ESYNC output will be LOW. This ESYNC output provides the user with a mark with which they can synchronize to the PAEn bus, ESYNC is always HIGH for the RCLK cycle that the first status word of a device is present on the PAEn bus.

When devices are connected in expansion configuration, only one device will be set as the Master (ID='000'), MAST input tied HIGH, all other devices will have MAST tied LOW. The master device is the first device to take control of the PAEn bus and will place its first status word on the bus on the rising edge of RCLK after the MRS input goes LOW. For the next n RCLK cycles (n=number of queues divided by 8 with n incrementing by one should there be a remainder) the master device will maintain control of the PAEn bus and cycle its status words through it, all other devices hold their PAEn outputs in High-Impedance. When the master device has cycled all of its status words it passes a token to the next device in the chain and that device assumes control of the PAEn bus and then cycles its status words and so on, the PAEn bus control token being passed on from device to device. This token passing is done via the EXO outputs and EXI inputs of the devices ("PAE Expansion Out" and "PAE Expansion In"). The EXO output of the master device connects to the EXI of the second device in the chain and the EXO of the second connects to the EXI of the third and so on. The final device in a chain has its EXO connected to the EXI of the first device, so that once the PAEn bus has cycled through all status words of all devices, control of the PAEn will pass to the master device again and so on. The ESYNC of each respective device will operate independently and simply indicate when that respective device has taken control of the bus and is placing its first status word on to the PAEn bus.

When operating in single device mode the EXI input must be connected to the EXO output of the same device. In single device mode a token is still required to be passed into the device for accessing the PAEn bus.

TABLE 15 — INTERFACE DATA RATES

Input Interface	Output Interface	Supported Data Transfer Rate Combination
SDR	SDR	Yes
SDR	DDR	Yes
DDR	SDR	Yes
DDR	DDR	Yes

INPUT INTERFACE

The input port will support either Edge Aligned data clocking or Center Aligned data clocking. No device configuration is required. For reference XGMII uses center aligned clocking.

The input interface will support either Single Data Rate (SDR) data transfers or Double Data Rate (DDR) data transfers.

OUTPUT INTERFACE

The output interface will provide a Centered Aligned data clock for Double Data Rate (DDR) operation and Edge Aligned data clock for Single Data Rate (SDR) operation. The output interface will support either Single Data Rate (SDR) data transfers or Double Data Rate (DDR) data transfers.

Note, the sum of (ta max + tsu min) must be less than the cycle time.

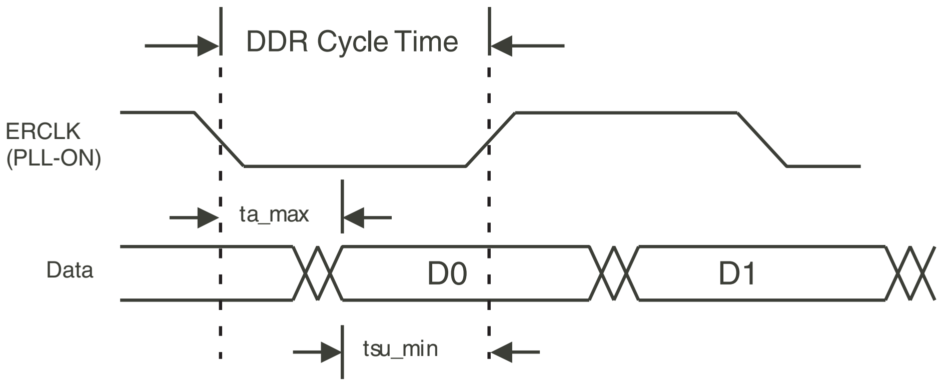


Figure 34. DDR Source Synchronous Center Aligned Clocking

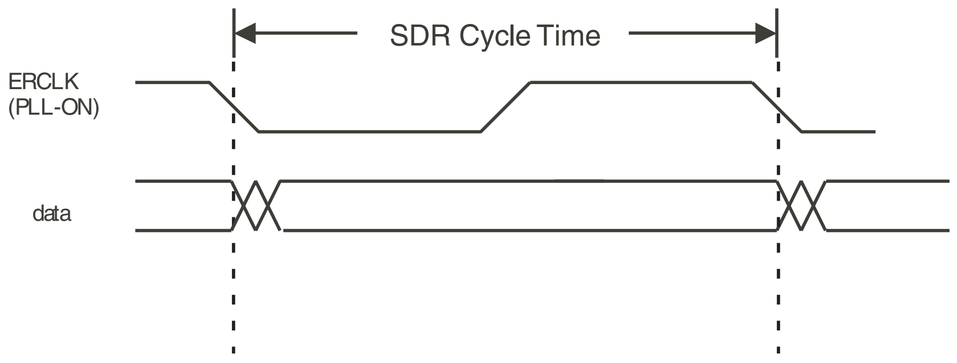


Figure 35. SDR Edge Aligned Clocking

BUS MATCHING OPERATION

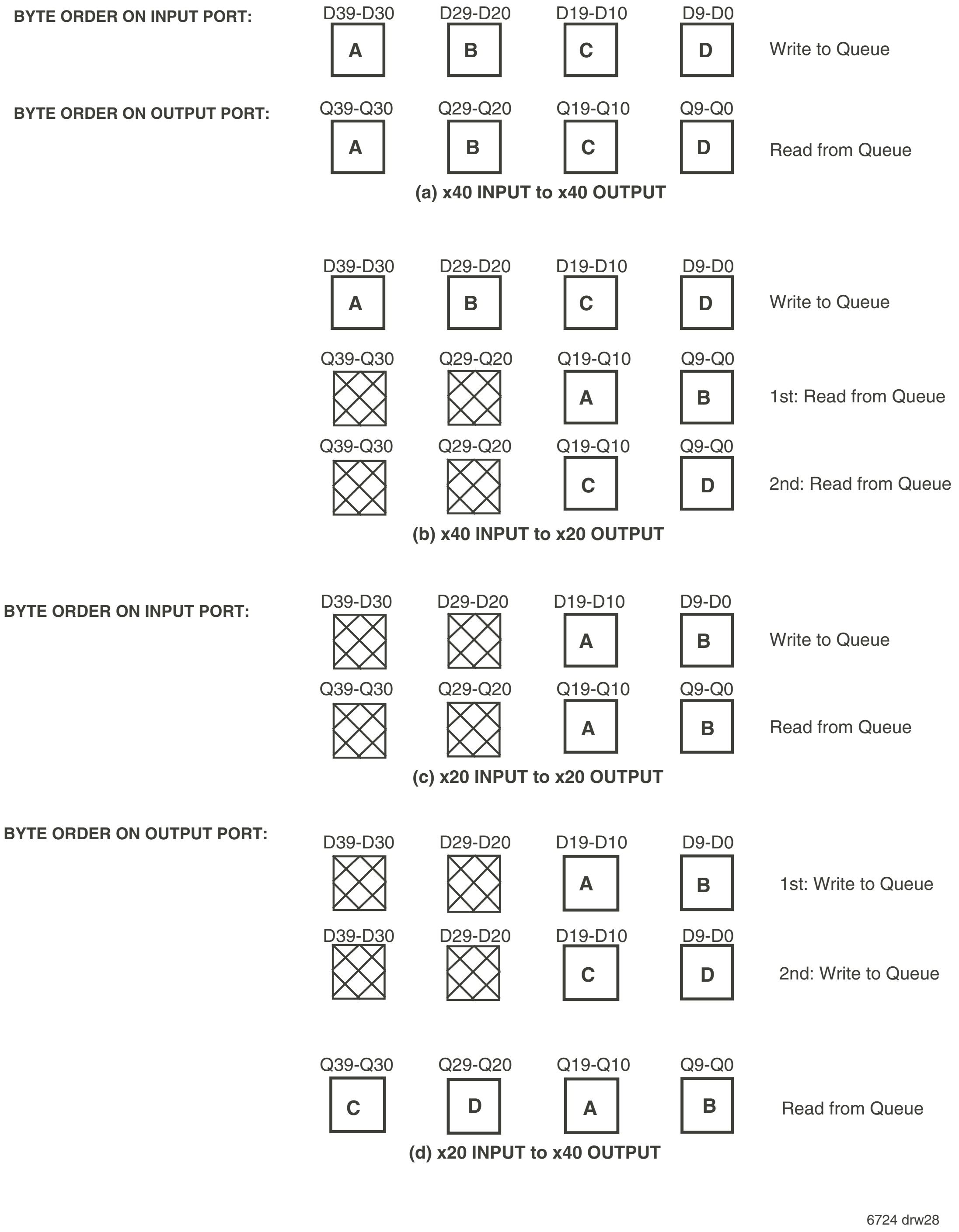
Bus Matching operation between the input port and output port is available. During a master reset of the multi-queue the state of the three setup pins, BM [3:0] (Bus Matching), determine the input and output port bus widths as shown in Table 16, "Bus Matching Configurations". 20 bit words and 40 bit words can be written into and read from the Queues. When writing to or reading from the multi-queue in a bus matching mode, the device orders data in a "Little Endian" format. See Figure 36, *Bus Matching Byte Arrangement* for details.

The Full flag and Almost Full flag operation is always based on writes and reads of data widths determined by the write port width. For example, if the input port is x40 and the output port is x20, then two data reads from a full queue will be required to cause the full flag to go HIGH (queue not full). Conversely, the Empty flag and Almost Empty flag operations are always based on writes and reads of data widths determined by the read port. For example, if the input port is x20 and the output port is x40, two write operations will be required to cause the Empty flag ($\overline{EF}$) of an empty queue to go HIGH (queue is not empty).

Note, that the input port serves all queues within a device, as does the output port, therefore the input bus width to all queues is equal (determined by the input port size) and the output bus width from all queues is equal (determined by the output port size).

TABLE 16 — BUS-MATCHING CONFIGURATIONS

BM3	BM2			Write	Read		
(IDR)	(ODR)	BM1	BM0	Port	Port	PAE Default	PAF Default
0	0	0	0	DDR x40	DDR x40	16	D-16
0	0	0	1	DDR x40	DDR x20	16	D-16
0	0	1	0	DDR x40	SDR x40	16	D-16
0	0	1	1	DDR x40	SDR x20	16	D-16
0	1	0	0	DDR x20	DDR x40	16	D-16
0	1	0	1	DDR x20	DDR x20	32	D-32
0	1	1	0	DDR x20	SDR x40	32	D-32
0	1	1	1	DDR x20	SDR x20	32	D-32
1	0	0	0	SDR x40	DDR x40	16	D-16
1	0	0	1	SDR x40	DDR x20	32	D-32
1	0	1	0	SDR x40	SDR x40	32	D-32
1	0	1	1	SDR x40	SDR x20	32	D-32
1	1	0	0	SDR x20	DDR x40	16	D-16
1	1	0	1	SDR x20	DDR x20	32	D-32
1	1	1	0	SDR x20	SDR x40	32	D-32
1	1	1	1	SDR x20	SDR x20	64	D-64



(a) x40 INPUT to x40 OUTPUT

(b) x40 INPUT to x20 OUTPUT

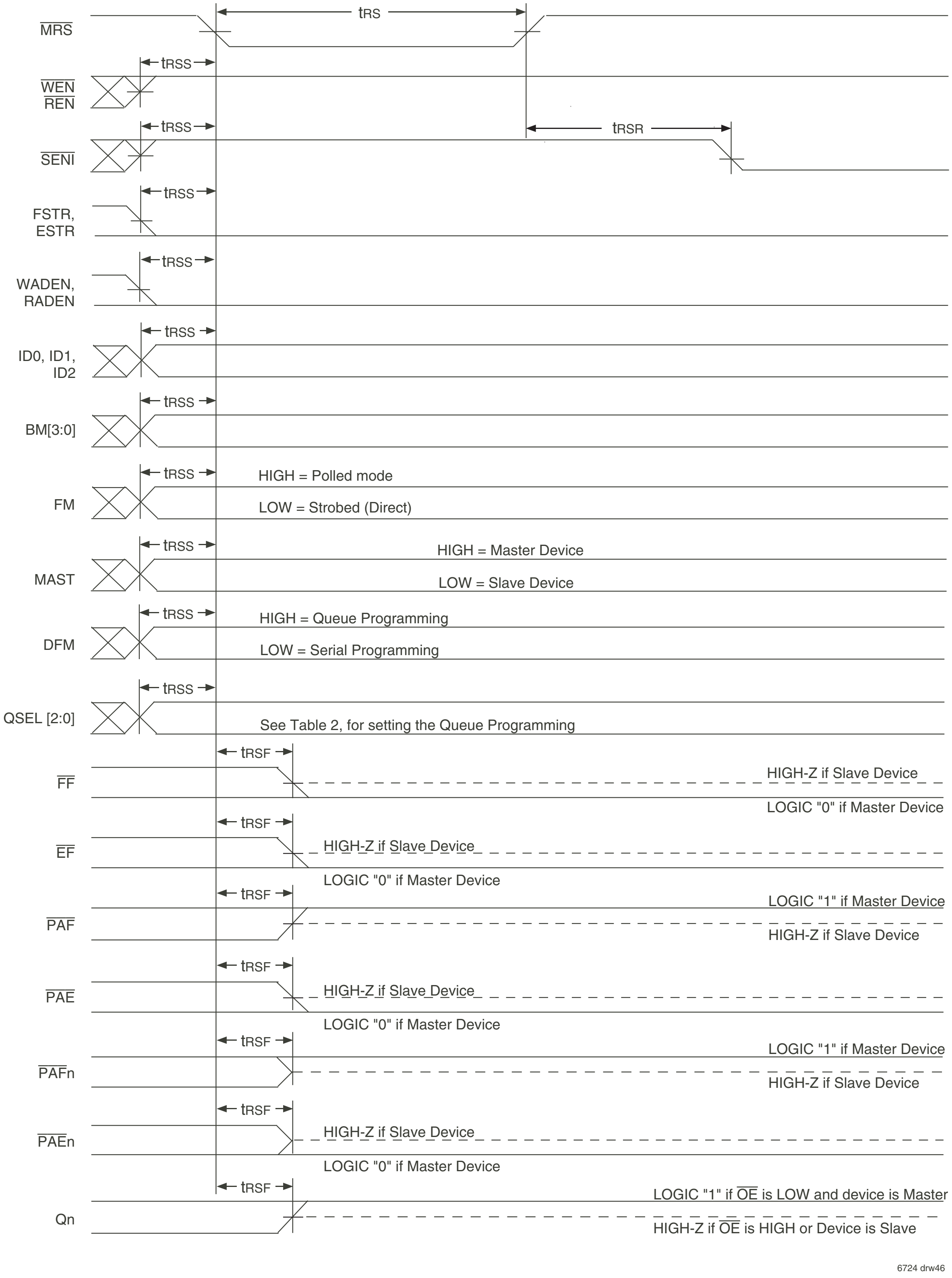
(c) x20 INPUT to x20 OUTPUT

(d) x20 INPUT to x40 OUTPUT

NOTES:

1. Please refer to Table 16, Bus-Matching set-up for details.
2. ☒ Data bits not used in this configuration.

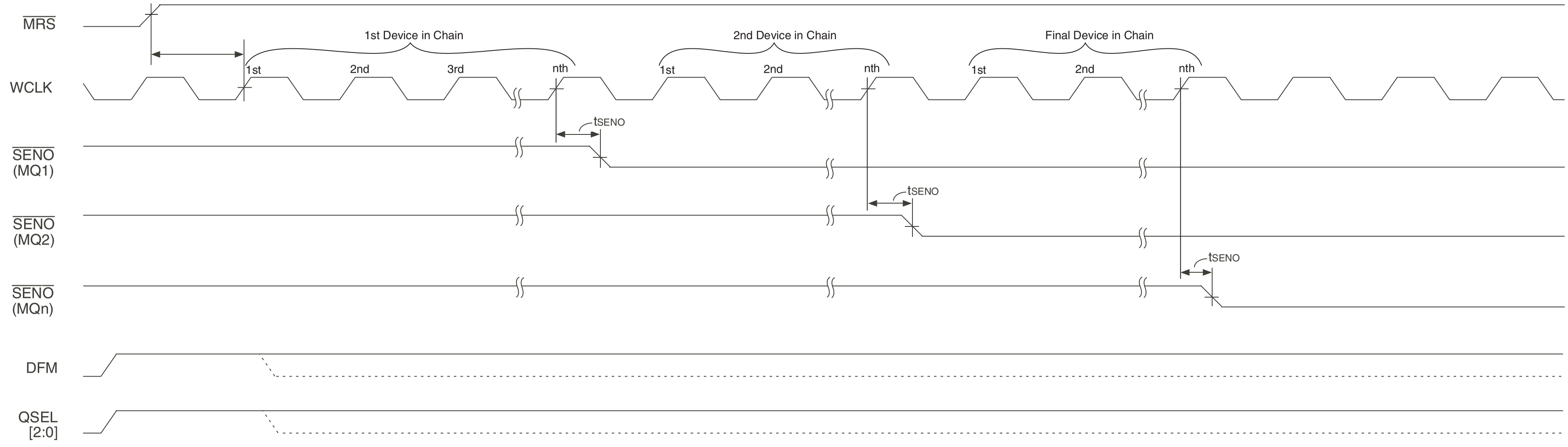
Figure 36. Bus-Matching Byte Arrangement



NOTE:

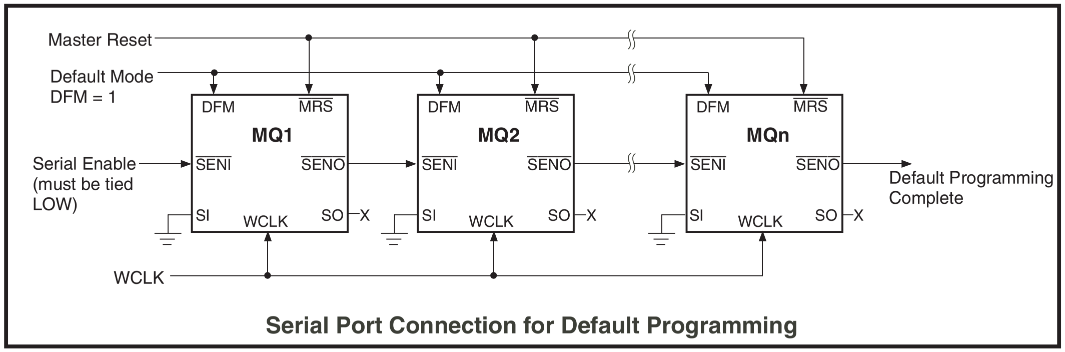
1. $\overline{OE}$ can toggle during this period.

Figure 37. Master Reset



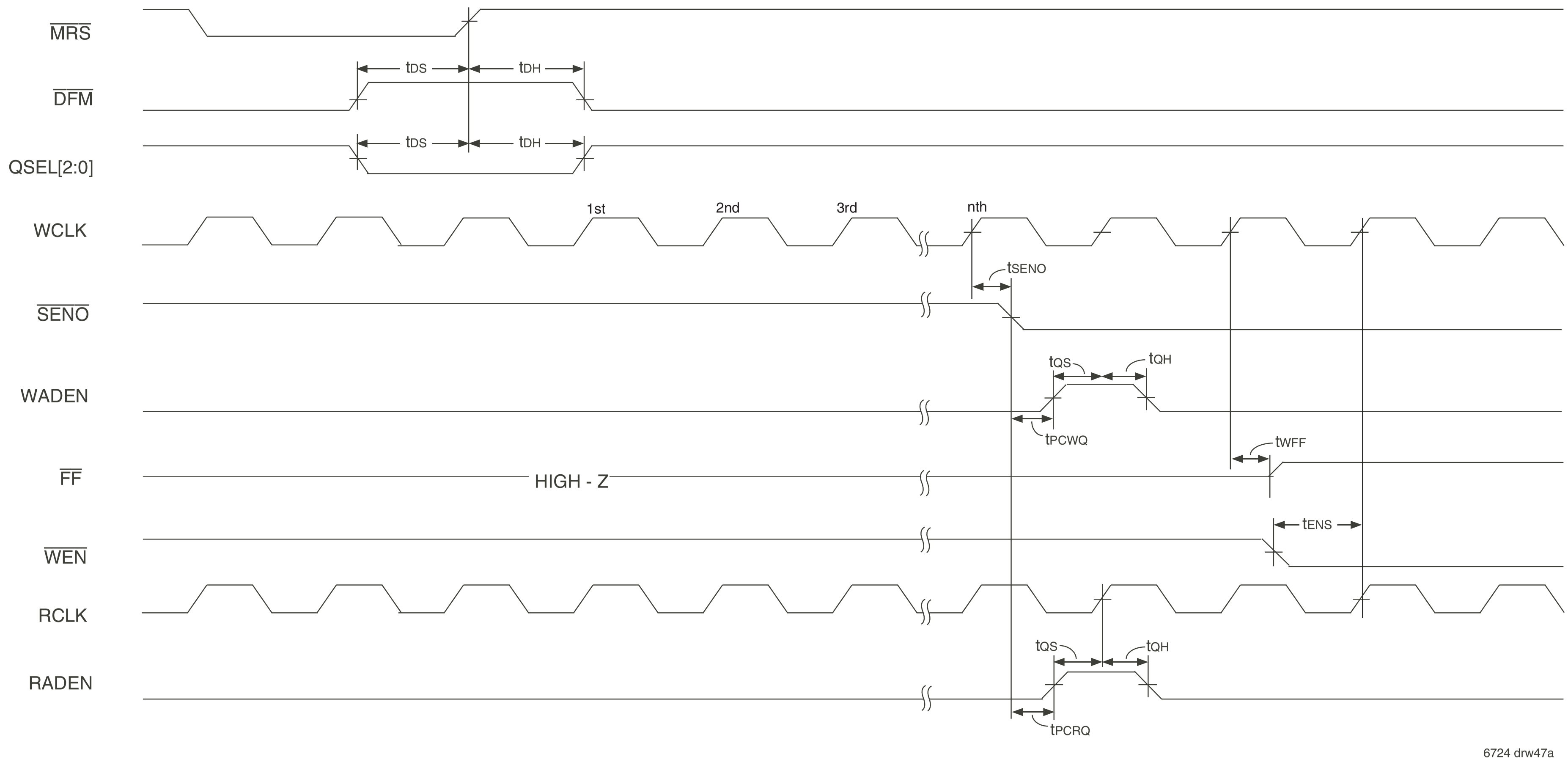
NOTES:

1. This diagram illustrates multiple devices connected in expansion. The $\overline{SENO}$ of the final device in a chain is the "programming complete" signal.
2. $\overline{SENI}$ of the first device in the chain must be held LOW
3. The $\overline{SENO}$ of a device should connect to the $\overline{SENI}$ of the next device in the chain. The final device $\overline{SENO}$ is used to indicate programming complete.
4. When Default Programming is complete the $\overline{SENO}$ of the final device will go LOW.
5. SCLK is not used and can be tied LOW.
6. Programming of all devices must be complete ($\overline{SENO}$ of the final device is LOW), before any write or read port operations can take place, this includes queue selections.



6724 drw47

Figure 38. Default Programming



NOTES:
1. The $\overline{\text{SENO}}$ is the "programming complete" signal.
2. $\overline{\text{SENI}}$ can be held LOW.
3. When Parallel Programming is complete the $\overline{\text{SENO}}$ of the device will go LOW.
4. SCLK is not used and can be tied LOW.
5. Programming of the device must be complete ($\overline{\text{SENO}}$ of the device is LOW), before any write or read port operations can take place, this includes queue selections.
6. See Table 5, Setting the Queue Programming Mode for setting default number of Queues rising (QSEL[2:0].
7. Depending on QSEL0 input, the Queue will be selected for programming the number of queues in parallel programming mode.

Figure 39. Write Address/Read Address Programming

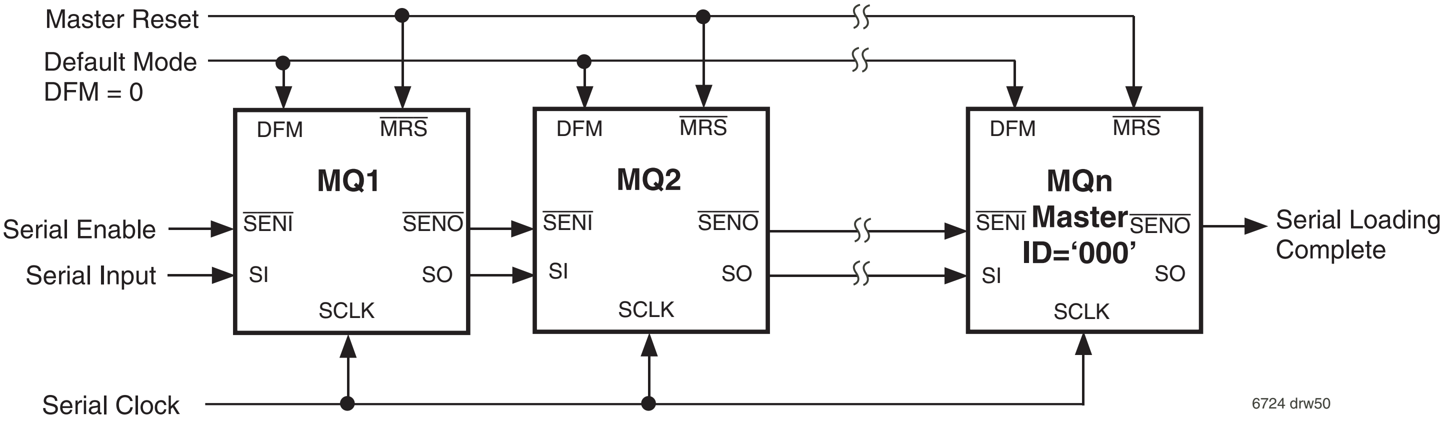
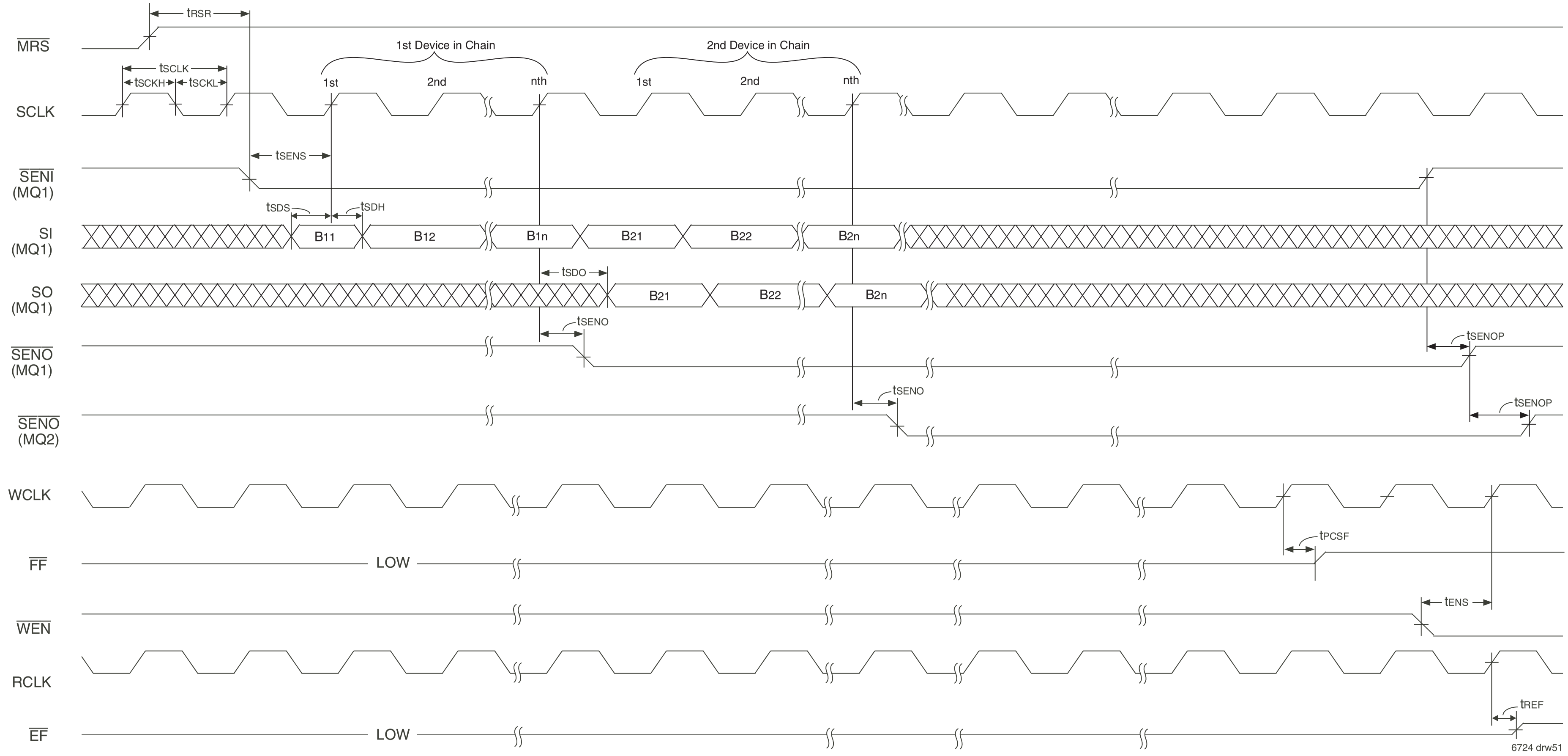
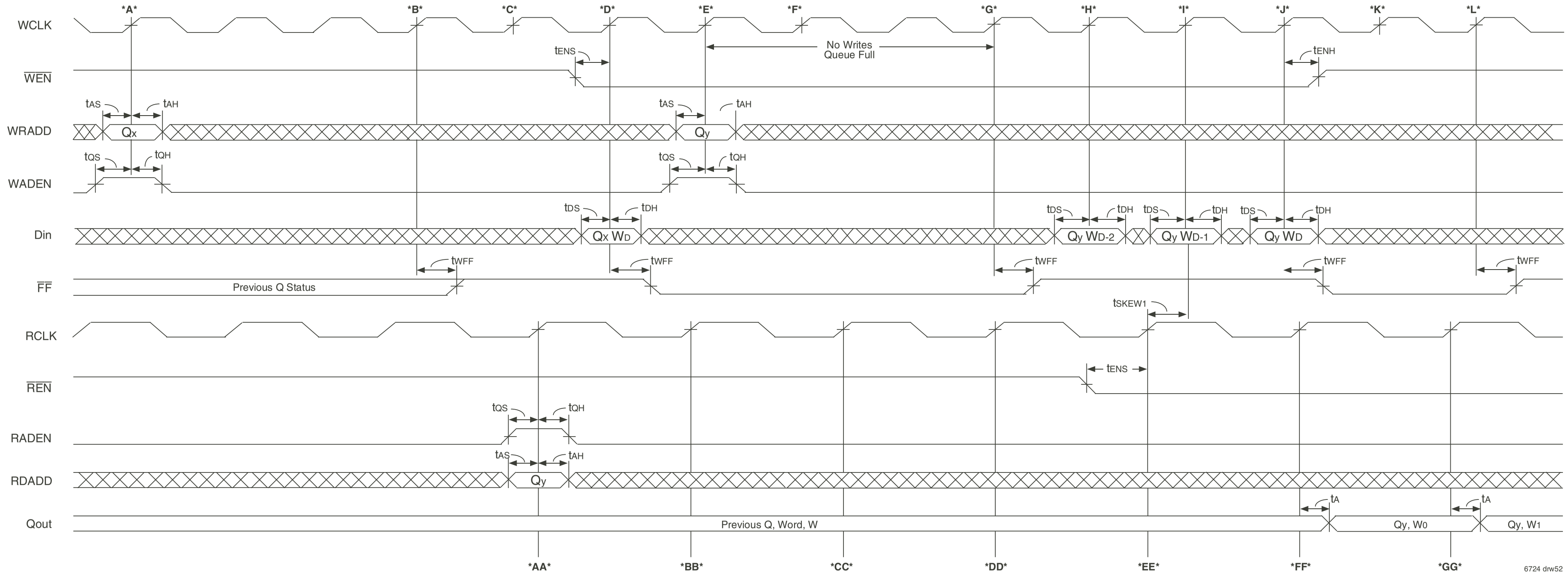


Figure 40. Serial Port Connection for Serial Programming



NOTES:
1. $\overline{\text{SENI}}$ can be toggled during serial loading. Once serial programming of a device is complete, the $\overline{\text{SENI}}$ and SI inputs become transparent. $\overline{\text{SENI}} \rightarrow \overline{\text{SENO}}$ and SI → SO.
2. DFM is LOW and QSEL[2:0] = LOW during Master Reset to provide Serial programming mode.
3. When $\overline{\text{SENO}}$ of the final device is LOW no further serial loads will be accepted.
4. n = 27+(Qx104); where Q is the number of queues required for the IDT72P51767/72P51777.
5. Programming of all devices must be complete ($\overline{\text{SENO}}$ of the 2nd device is LOW), before any write or read port operations can take place, this includes queue selections.

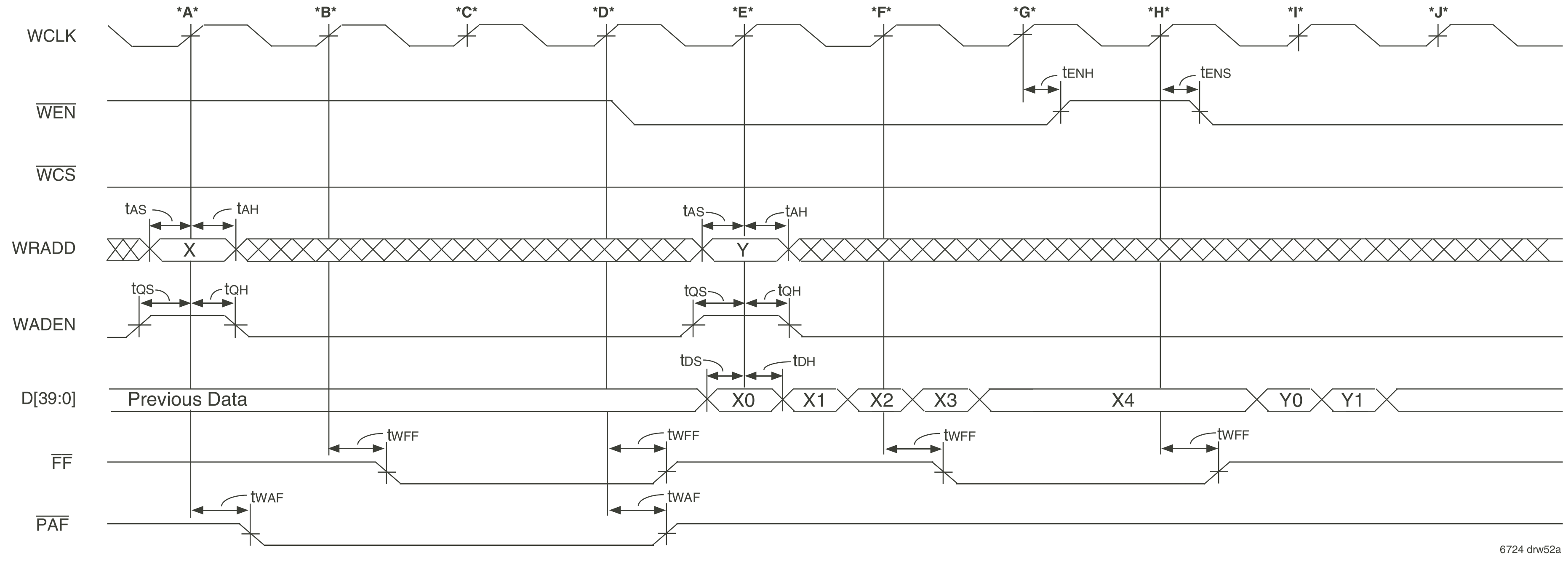
Figure 41. Serial Programming (2 Device Expansion)



NOTE:
$\overline{OE}$ is active LOW.
Cycle:

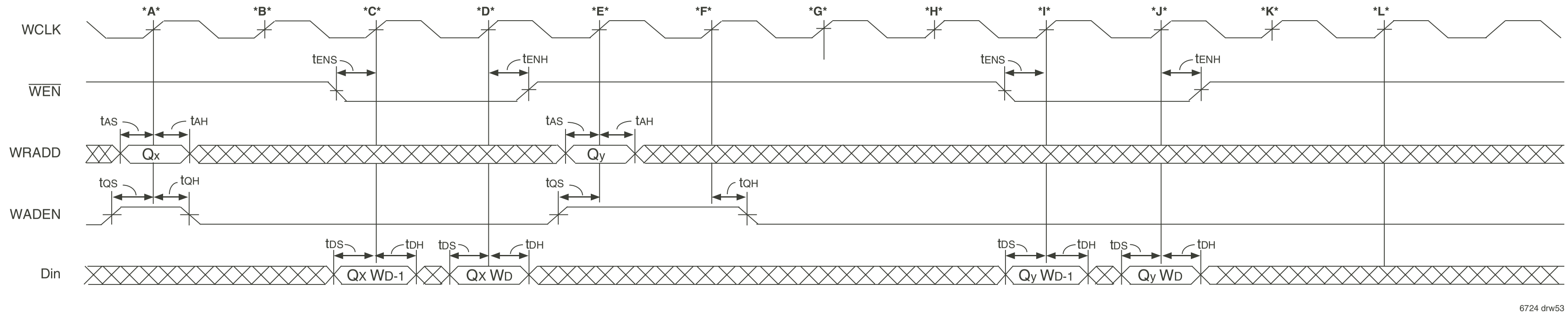
A Queue, Qx is selected on the write port.
The $\overline{FF}$ flag is providing status of a previously selected queue, within the same device.
AA Queue, Qy is selected for read operations.
B The $\overline{FF}$ flag output updates to show the status of Qx, it is not full.
BB Current word is kept on the output bus since $\overline{REN}$ is HIGH.
C The $\overline{FF}$ flag provides status of previous queue.
CC Word W+1 is read from the previous queue regardless of $\overline{REN}$ due to FWFT.
D Word, Wd is written into Qx. This causes Qx to go full.
DD The next available Word W0 of Qy is read out regardless of $\overline{REN}$, 3 RCLK cycles after queue selection. This is FWFT operation.
E Queue, Qy is selected within the same device as Qx. A write to Qx cannot occur on this cycle because it is full, $\overline{FF}$ is LOW.
EE No reads occur, $\overline{REN}$ is HIGH. Word, W0 is read from Qy, this causes Qy to go "not full", $\overline{FF}$ flag goes HIGH after time, t_{SKEW1} + t_{WFF}. Note, if t_{SKEW1} is violated the time $\overline{FF}$ HIGH will be: t_{SKEW1} + WCLK + t_{WFF}.
F Again, a write to Qx cannot occur on this cycle because it is full, $\overline{FF}$ is LOW.
FF Word, W0 is read from Qy.
G The $\overline{FF}$ flag updates after time t_{WFF} to show that queue, Qy is not full.
H Word, Wd-2 is written into Qy.
I Word, Wd-1 is written into Qy.
J Word, Wd is written into Qy, this causes Qy to go full, $\overline{FF}$ goes LOW.
K A write to Qy cannot occur on this cycle because it is full, $\overline{FF}$ is LOW.
L Qy goes "not full" based on reading word W0 from Qy on cycle *FF*.

Figure 42. SDR Write Queue Select, Write Operation and Full Flag Operation



NOTES:
1. First Queue is selected on the write port at Cycle 2.
2. User must wait 3 clock cycles after Queue switch is issued for data to be written in.
3. $\overline{\text{WEN}}$ is clocked in only on the rising edge of WCLK, set up time is 1.5ns, hold time is 0.5ns
4. The $\overline{\text{FF}}$ flag has "real time" response to active $\overline{\text{WEN}}$ and changes only on rising edge of WCLK. Flag access time < 3.6ns
5. Data X3 stays on the Write bus until Queue Y data is ready after 4 clock cycles
6. $\overline{\text{FF}}$ goes high reflects Queue Y's status
7. $\overline{\text{PAF}}$ flag updates 3 clock cycles after offset in register has been eclipsed

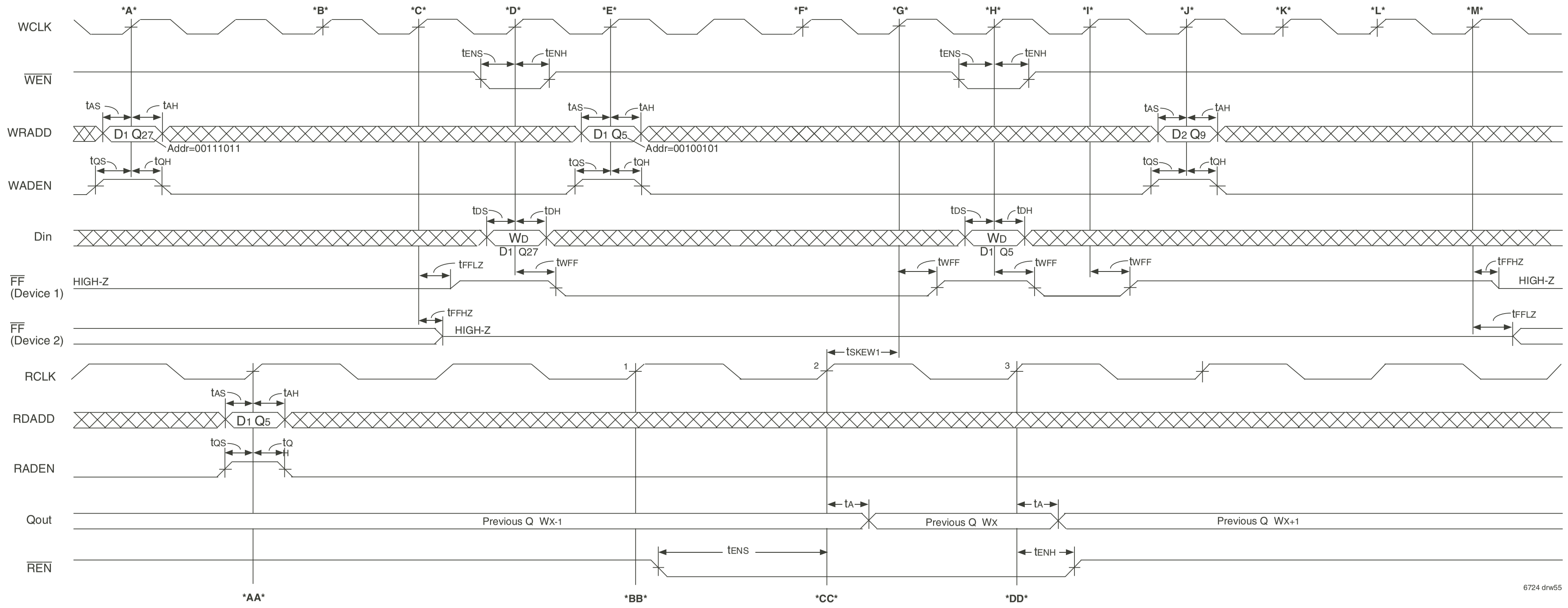
Figure 43. DDR Write Operation, Write Queue Select, Full Flag Operation



NOTES:
1. Only 1 queue can be marked at any given time.
2. Marking a queue can only occur during a queue switch.

Cycle:
A Queue "X" is selected but not marked.
E Queue "Y" is selected.
F Queue "Y" is marked.

Figure 44. Write Queue Select, Mark and Rewrite



Cycle:

A Queue, Q27 of device 1 is selected on the write port.
The $\overline{\text{FF}}$ flag of device 1 is in High-Impedance, the write port of device 2 was previously selected.
$\overline{\text{WEN}}$ is HIGH so no write occurs. The $\overline{\text{FF}}$ flag stays in High-Impedance for 3 WCLK cycles.

AA Queue, Q5 of device 1 is selected on the read port.

BB Word, Wx-1 is held on the outputs for 2 RCLK cycles after a read Queue switch.

C The $\overline{\text{FF}}$ flag of device 2 goes to High-Impedance and the $\overline{\text{FF}}$ flag of device 1 goes to Low-Impedance, logic HIGH indicating that D1 Q27 is not full.
$\overline{\text{WEN}}$ is HIGH so no write occurs.

CC Word, Wx is read from the previously selected queue.

D Word, Wd is written into Q27 of D1. This write operation causes Q27 to go full, $\overline{\text{FF}}$ goes LOW.

DD The first word from Q5 of D1 selected on cycle *AA* is read out. This read caused Q5 to go not full, therefore the $\overline{\text{FF}}$ flag will go HIGH after: t_{SKEW1} + t_{WFF}.
Note if t_{SKEW1} is violated the time to $\overline{\text{FF}}$ flag HIGH is t_{SKEW1} + WLCK + t_{WFF}.

E Queue, Q5 of device 1 is selected on the write port. No write occurs on this cycle.

G The $\overline{\text{FF}}$ flag updates to show the status of D1 Q5, it is not full, $\overline{\text{FF}}$ goes HIGH.

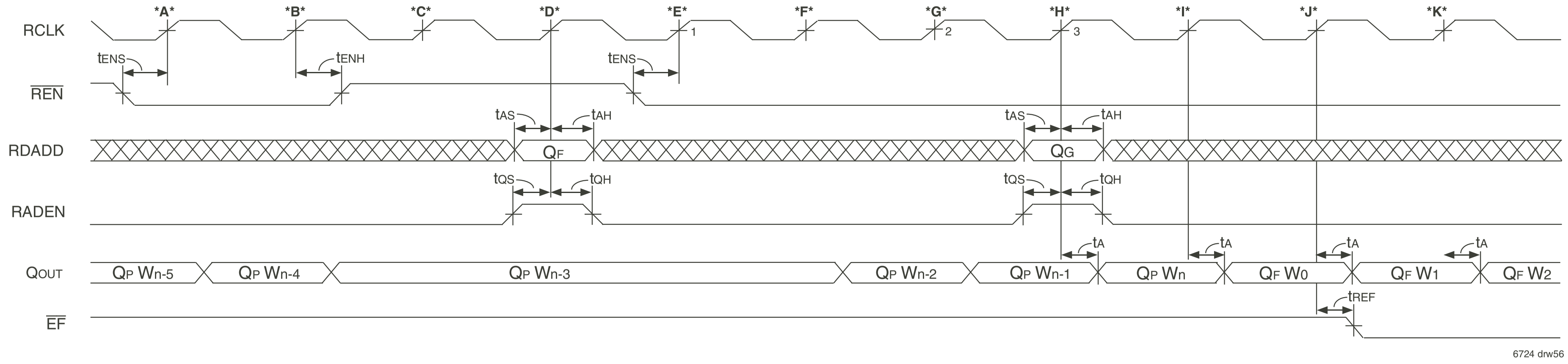
H Word, Wd is written into Q5 of D1. This causes the queue to go full, $\overline{\text{FF}}$ goes LOW.

I No write occurs regardless of $\overline{\text{WEN}}$, the $\overline{\text{FF}}$ flag is LOW preventing writes. The $\overline{\text{FF}}$ flag goes HIGH due to the read from Q5 of D1 on cycle *CC*. (This read is not an enabled read).

J Queue, Q9 of device 2 is selected on the write port.

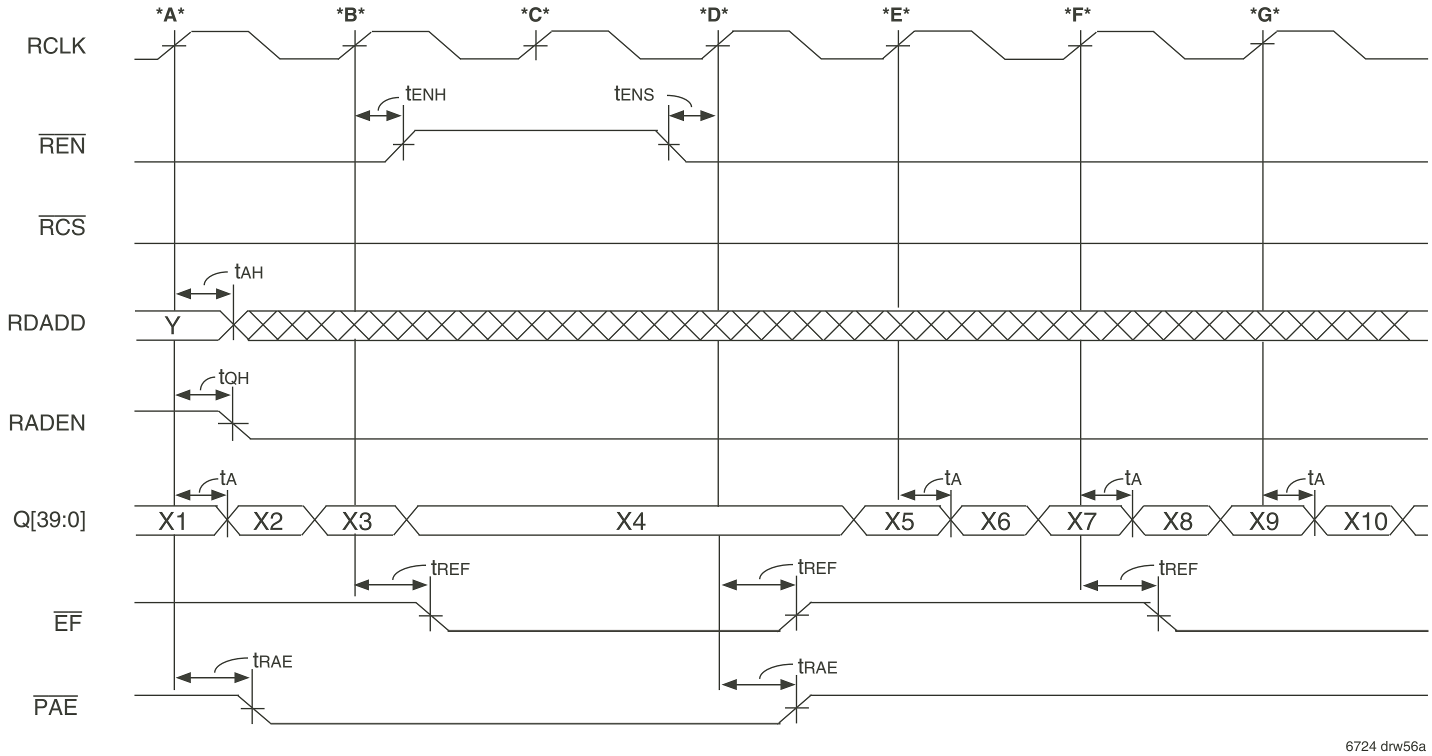
M The $\overline{\text{FF}}$ flag of device 1 goes to High-Impedance, this device was deselected on the write port on cycle *I*. The $\overline{\text{FF}}$ flag of device 2 goes to Low-Impedance and provides status of Q9 of D2.

Figure 45. Full Flag Timing in Expansion Configuration



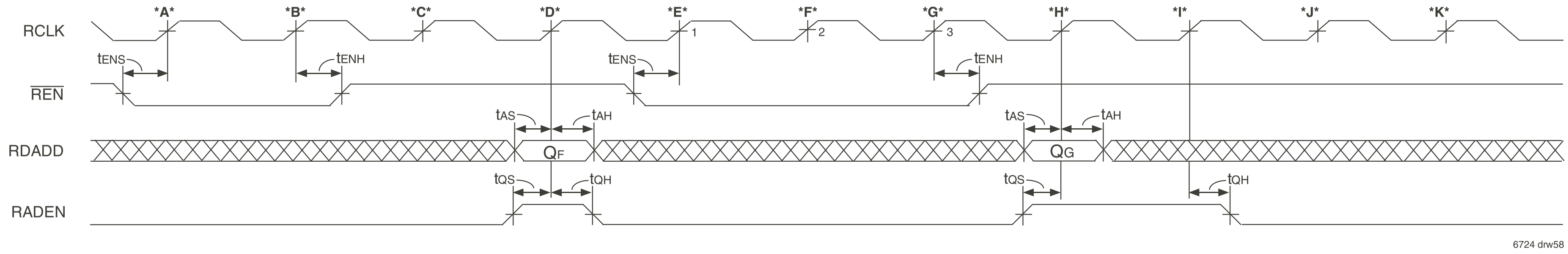
Cycle:
A Word W_{n-4} is on the Qout bus from the present selected queue.
B W_{n-3} is placed on Qout bus.
C Reads are disabled, W_{n-3} remains on Qout bus.
D Reads are disabled W_{n-3} remains on Qout bus.
E Read operation enabled.
F Word Q_P W_{n-3} is read from the Qout bus.

Figure 46. SDR Read Queue Select, Read Operation (IDT mode)



NOTES:
1. On Cycle 2, Queue Y is addressed to read data from.
2. On Cycle 6, Queue Y data is available on the Read bus
3. On Cycle 5, EF and PAE flags have updated to give the status for Queue Y
4. Previous Data from Queue X will be read during queue switch until 3 cycle latency from queue switch, plus 1 cycle Read delay of the first word in Queue Y is completed.
5. REN is high for falling edge of cycle 3 and all of cycle 4 which means Words X6-X8 will not be available for read operation

Figure 47. DDR Read Operation, Read Queue Select, EF & PAE Flag Operation



NOTES:
1. Only 1 queue can be marked at any given time.
2. Marking a queue can only occur during a queue switch.

Cycle:
D Queue "F" is selected but not marked.
H Queue "G" is selected.
I Queue "G" is marked.

Figure 48. Read Queue Select, Mark and Reread (IDT mode)

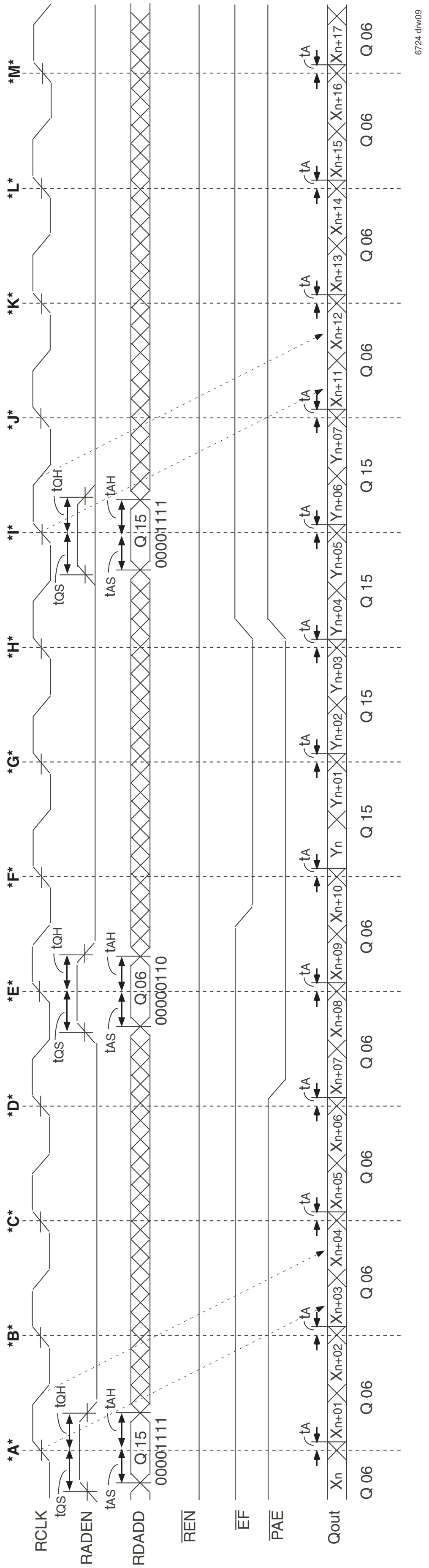


Figure 49. Standard Mode Pointers on Queue Re-entry for DDR Read Operation

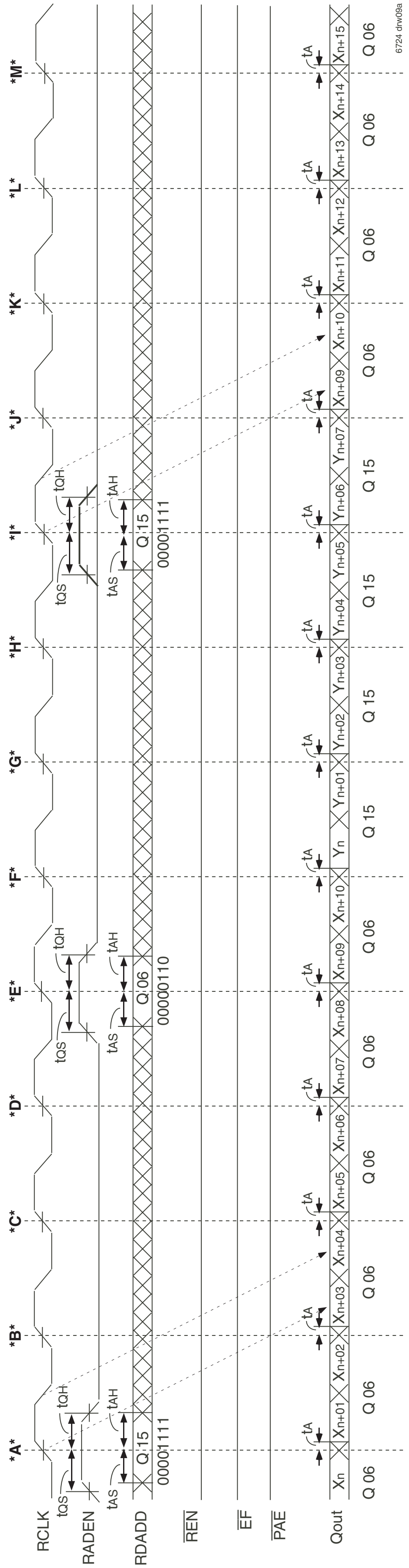
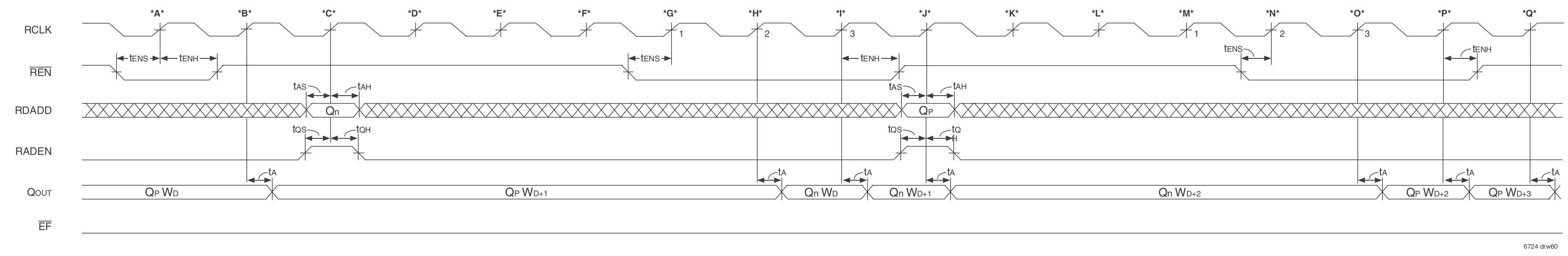
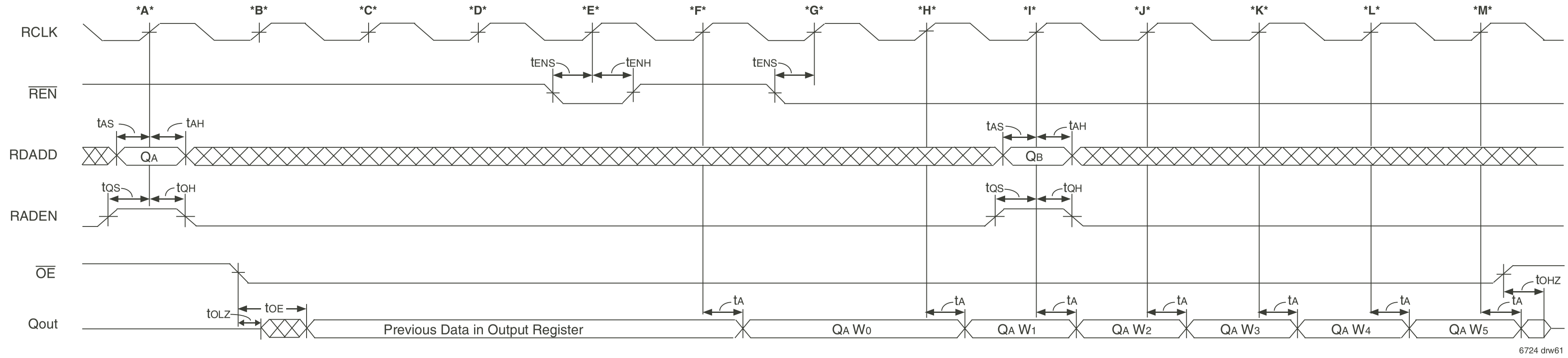


Figure 50. BOI Mode Pointers on Queue Re-entry for DDR Read Operation



Cycle:
A Word Qp WD is read from the present selected queue, Qp.
B Reads are disabled, word Qp WD+1 remains on the output bus.
C A new queue, Qn is selected for read port operations. Qp WD+1 remains on Qout bus.
D REN is not asserted therefore no read operation occurs, Qp WD+1 remains on Qout bus.
E REN is not asserted therefore no read operation occurs, Qp WD+1 remains on Qout bus.
F REN is not asserted therefore no read operation occurs, Qp WD+1 remains on Qout bus.
G Read operation is initiated.
H Word WD+1 of Qp is read.
I Word WD of Qn is read.
J The queue, Qp is again selected.
K Current Word is kept on the output bus since REN is HIGH.
L Word Qn WD+2 remains on the Qout bus.
M Word Qn WD+2 remains on the Qout bus.
N Read operation is initiated.
O Word WD+2 for Qn is read.
P Word WD+2 for Qp is read.

Figure 51. Read Queue Selection with Read Operations (IDT mode)
(SDR mode, PLL = OFF)



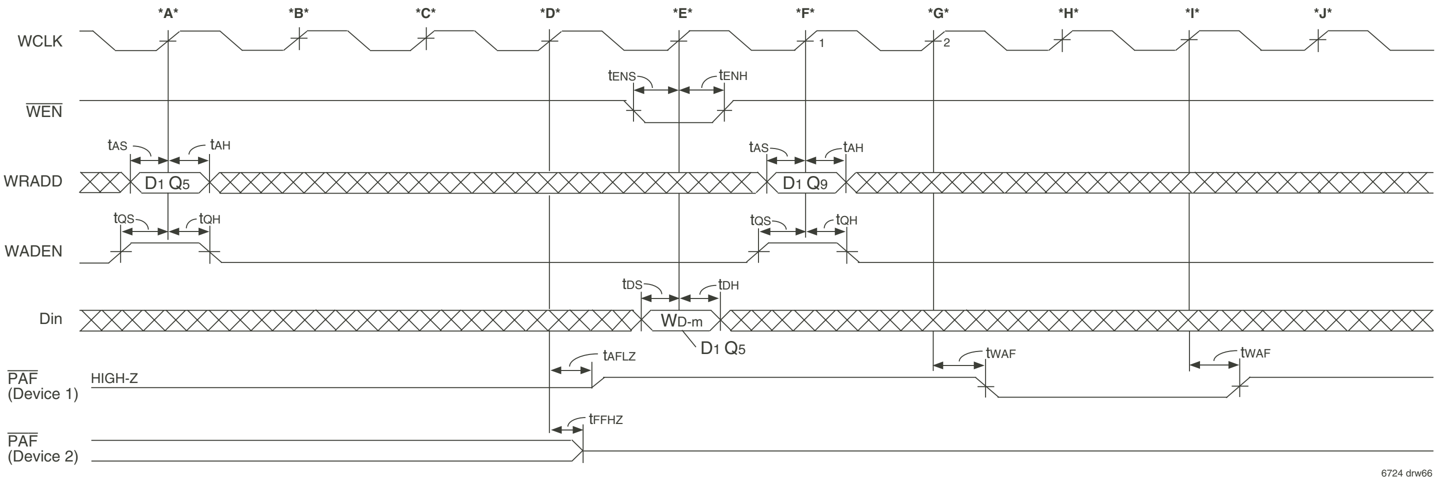
NOTES:

1. In expansion configuration the OE inputs of all devices should be connected together. This allows the output busses of all devices to be High-Impedance controlled.

Cycle:

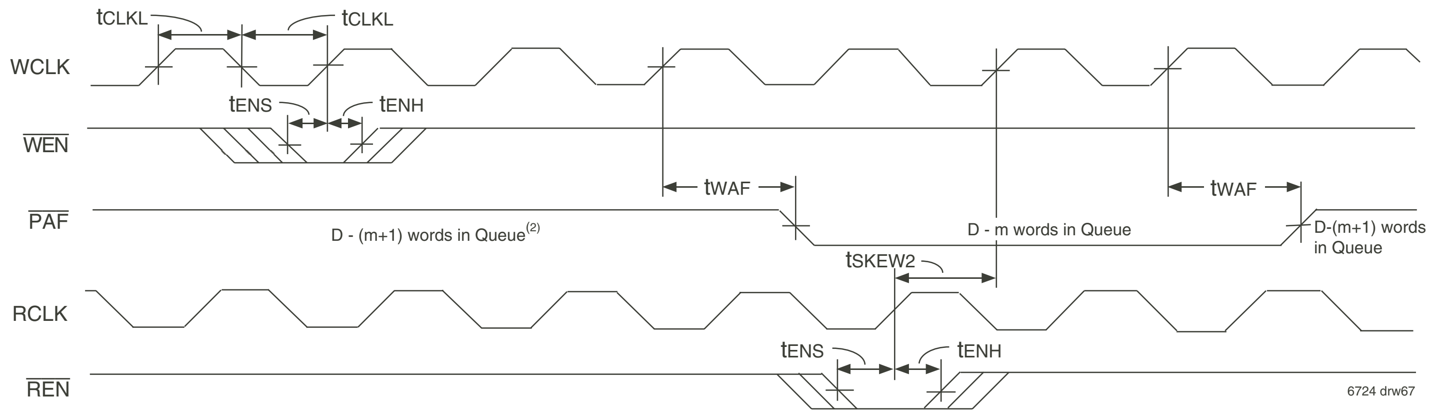
A Queue A is selected for reads. No data will fall through on this cycle, the previous queue was read to empty.
B No read operation REN = HIGH.
C Previous data kept on output bus since there is no read operation.
D Previous data kept on output bus since there is no read operation.
F Reads are disabled therefore word, W0 of QA remains on the output bus.
E Read from Queue A initiated.
G Reads are again enabled.
H Word W0 is read from QA.
I Queue, QB is selected on the read port. This queue is actually empty. Word, W1 is read from QA.
J Word, W2 is read from QA.
K Word W3 is read from QA.

Figure 52. Read Queue Select, Read Operation and OE Timing



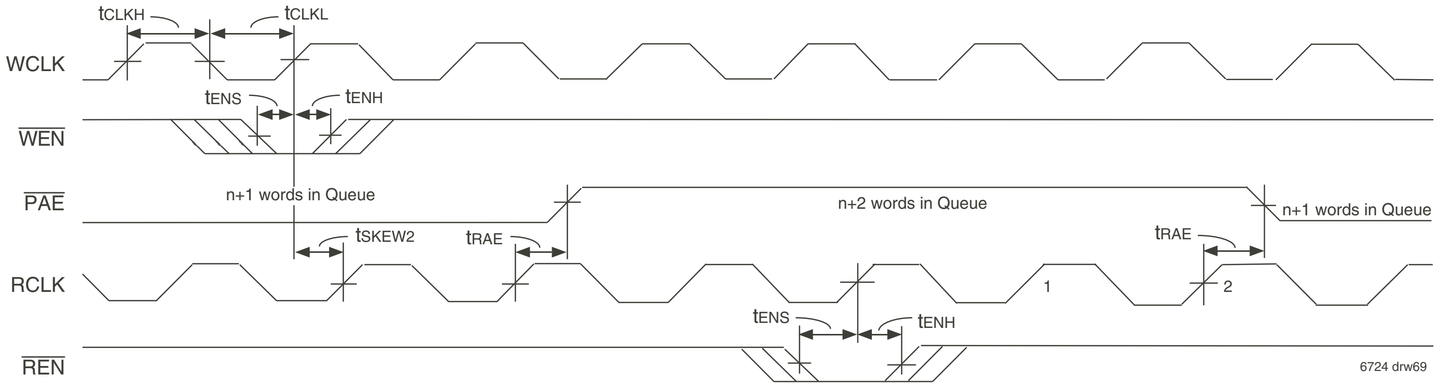
Cycle:
A Queue 5 of Device 1 is selected on the write port. A queue within Device 2 had previously been selected. The PAF output of device 1 is High-Impedance.
B No write occurs, WEN is HIGH.
C No write occurs, WEN is HIGH.
D No write occurs, WEN is HIGH.
E Word, Wd-m is written into Q5 causing the PAF flag to go from HIGH to LOW. The flag latency is 3 WCLK cycles + tWAF.
F Queue 9 in device 1 is now selected for write operations. This queue is not almost full, therefore the PAF flag will update after a 3 WCLK + tWAF latency.
G The PAF flag goes LOW based on the write 2 cycles earlier.
H No write occurs, WEN is HIGH.
I The PAF flag goes HIGH due to the queue switch to Q9.

Figure 53. Almost Full Flag Timing and Queue Switch



NOTE:
1. The waveform shows the PAF flag operation when no queue switch occurs and a queue is selected on both the write and read ports is being written to then read from at the almost full boundary.
2. Flag Latencies:
   Assertion: 2*WCLK + tWAF
   De-assertion: tSKEW2 + WCLK + tWAF
3. If tSKEW2 is violated there will be one extra WCLK cycle.

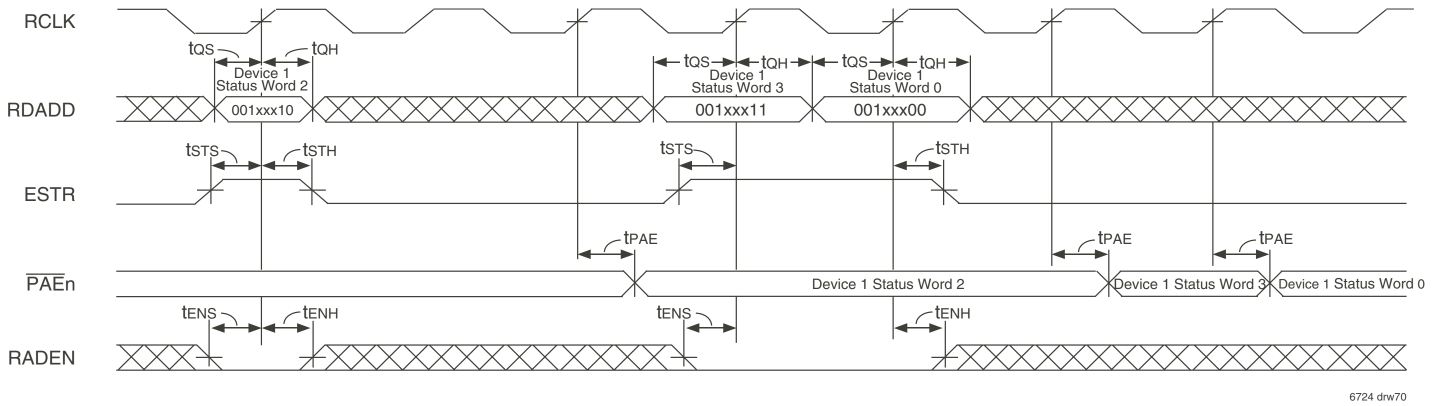
Figure 54. Almost Full Flag Timing



NOTE:

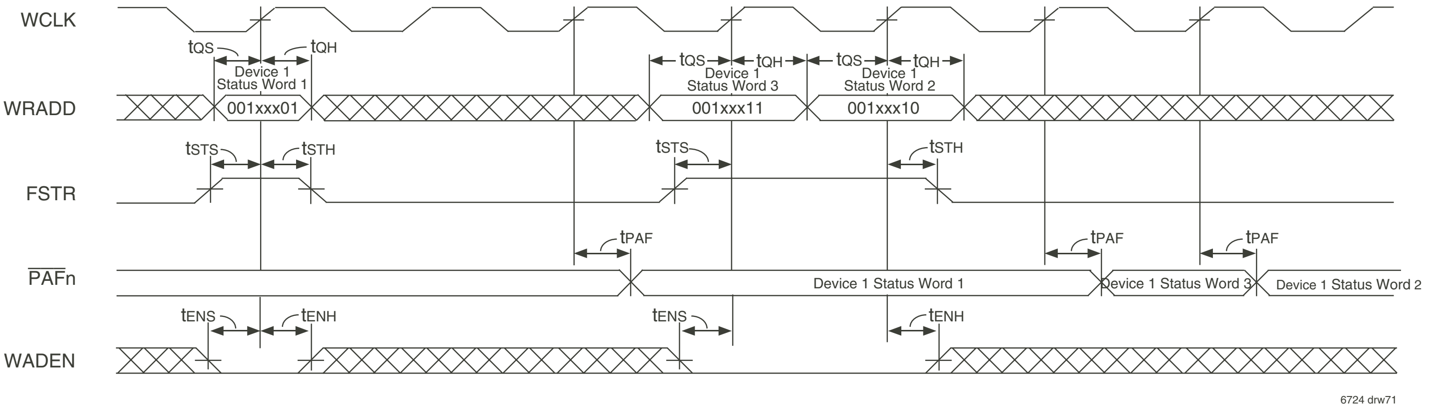
1. The waveform here shows the $\overline{\text{PAE}}$ flag operation when no queue switches are occurring and a queue selected on both the write and read ports is being written to then read from at the almost empty boundary.
   Flag Latencies:
2. Assertion: 2*RCLK + tRAE
   De-assertion: tSKEW2 + RCLK + tRAE
3. If tSKEW2 is violated there will be one extra RCLK cycle.

Figure 55. Almost Empty Flag Timing



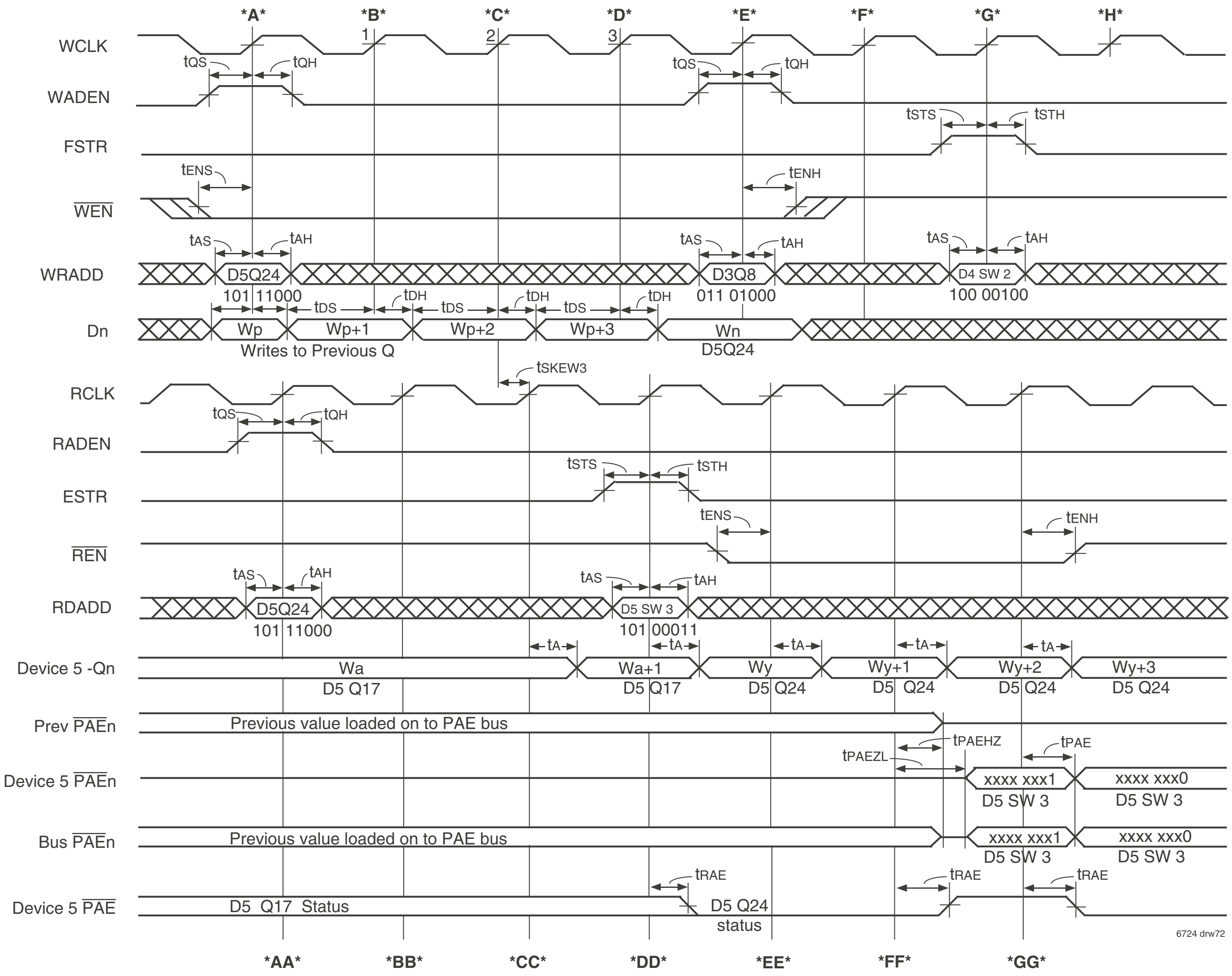
NOTES:
1. Status words can be selected on consecutive cycles.
2. On an RCLK cycle that the ESTR is HIGH, the RADEN input must be LOW.
3. There is a latency of 2 RCLK for the PAEn bus to switch.

Figure 56. PAEn - Direct Mode - Status Word Selection



NOTES:
1. Status words can be selected on consecutive cycles.
2. On a WCLK cycle that the FSTR is HIGH, the WADEN input must be LOW.
3. There is a latency of 2 WCLK for the PAFn bus to switch.

Figure 57. PAFn - Direct Mode - Status Word Selection



Cycle:

A Queue 24 of Device 5 is selected for write operations.
Word, Wp is written into the previously selected queue.

AA Queue 24 of Device 5 is selected for read operations.
A status word from another device has control of the PAEn bus.
The discrete PAE output of device 5 is currently in High-Impedance and the PAE active flag is controlled by the previously selected device.

B Word Wp+1 is written into the previously selected queue.

BB Current Word is kept on the output bus since REN is HIGH.

C Word Wp+2 is written into the previously selected queue.

D Word, Wn is written into the newly selected queue, Q24 of D5. This write will cause the PAE flag on the read port to go from LOW to HIGH (not almost empty) after time, tSKEW3 + RCLK + tRAE (if tSKEW3 is violated one extra RCLK cycle will be added).

DD Status word 4 of Device 5 is selected on the PAEn bus. Q24 of device 5 will therefore have is PAE status output on PAE[0]. There is a single RCLK cycle latency before the PAEn bus changes to the new selection.

E Queue 8 of Device 3 is selected for write operations.
Word Wn+1 is written into Q24 of D5.

EE Word, Wy+1 is read from Q24 of D5.

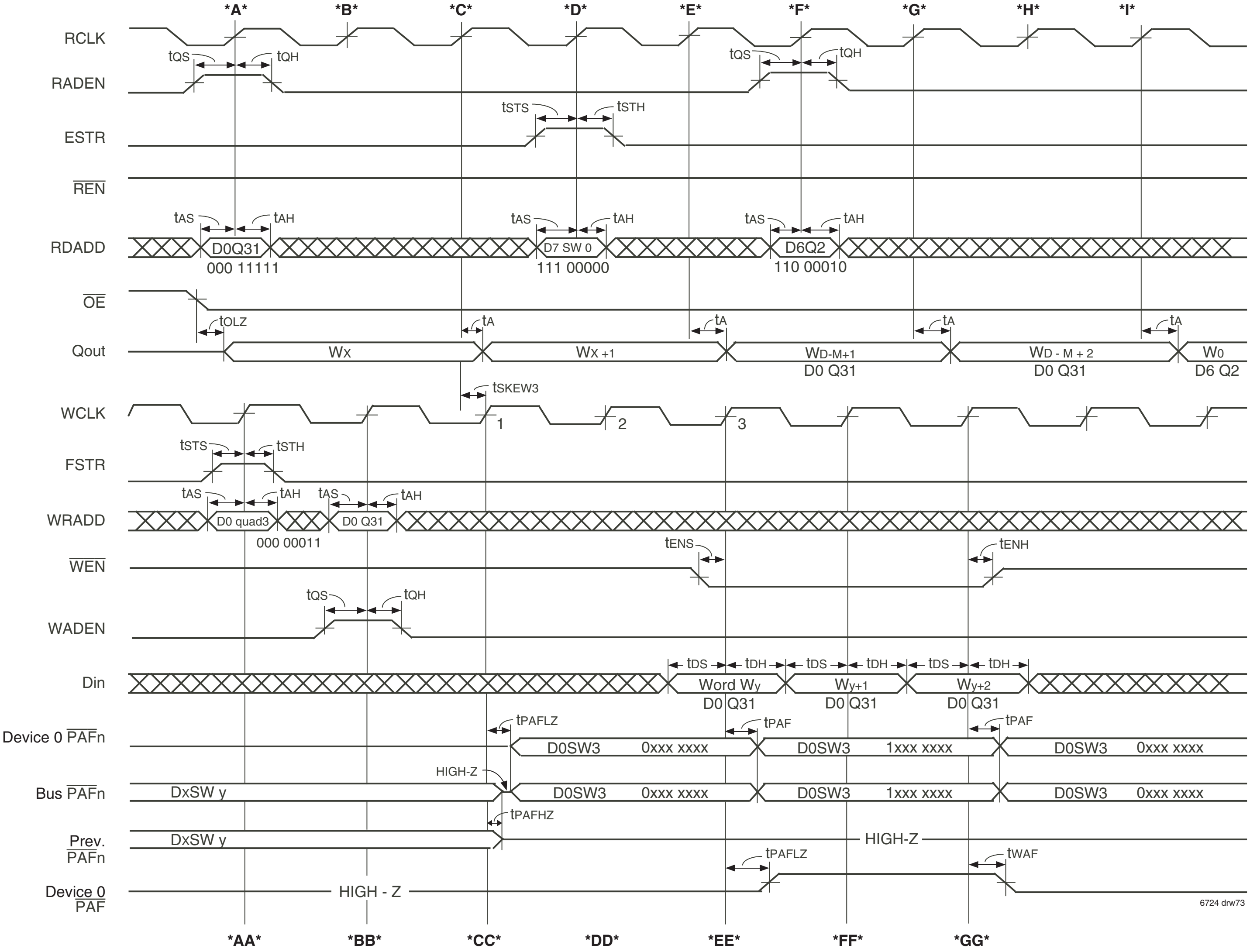
F No writes occur.

FF Word, Wy+2 is read from Q24 of D5.
The PAEn bus changes control to D5, the PAEn outputs of D5 go to Low-Impedance and status word 4 is placed onto the outputs. The device of the previously selected status word now places its PAEn outputs into High-Impedance to prevent bus contention.
The discrete PAE flag will go HIGH to show that Q24 of D5 is not almost empty. Q24 of device 5 will have its PAE status output on PAE[0].

G Status word 3 of device 4 is selected on the write port for the PAFn bus.

GG The PAEn bus updates to show that Q24 of D5 is almost empty based on the reading out of word, Wy+1.
The discrete PAE flag goes LOW to show that Q24 of D5 is almost empty based on the reading of Wy+1.

Figure 58. PAEn - Direct Mode, Flag Operation



Cycle:

A Queue 31 of device 0 is selected for read operations.
The last word in the output register is available on Qout. OE was previously taken LOW so the output bus is in Low-Impedance.

AA Status word 4 of device 0 is selected for the PAFn bus. The bus is currently providing status of a previously selected status word, Quad Y of device X.

B No read operation.

BB Queue 31 of device 0 is selected on the write port.

CC PAFn continues to show status of Quad4 D0.
The PAFn bus is updated with the status word selected on the previous cycle, D0 Quad 4. PAF[7] is LOW showing the status of queue 31.
The PAFn outputs of the device previously selected on the PAFn bus go to High-Impedance.

DD No write operation.

E No read operations occur, REN is HIGH.

EE PAF[7] goes HIGH to show that D0 Q31 is not almost empty due to the read on cycle *C*.
The active queue PAF flag of device 0 goes from High-Impedance to Low-Impedance.
Word, Wy is written into D0 Q31.

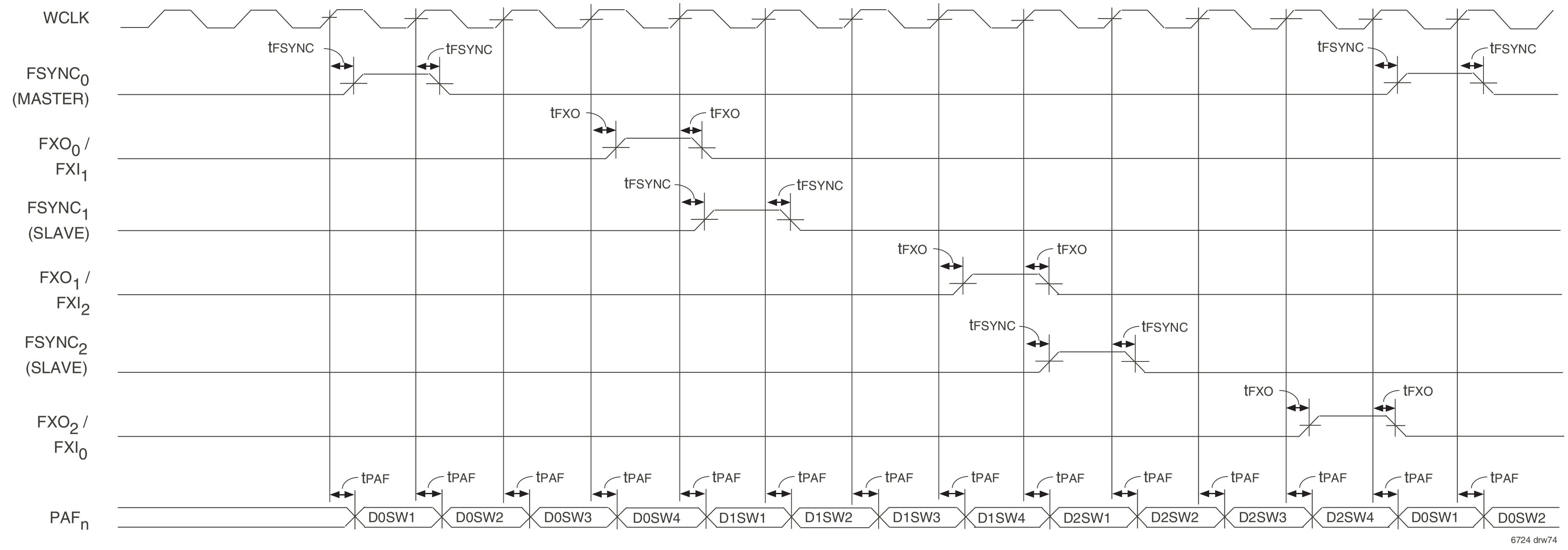
F Queue 2 of Device 6 is selected for read operations.

FF Word, Wy+1 is written into D0 Q31.

GG PAF[7] and the discrete PAF flag go LOW to show the write on cycle *DD* causes Q31 of D0 to again go almost full.
Word, Wy+2 is written into D0 Q31.

H No read operation.

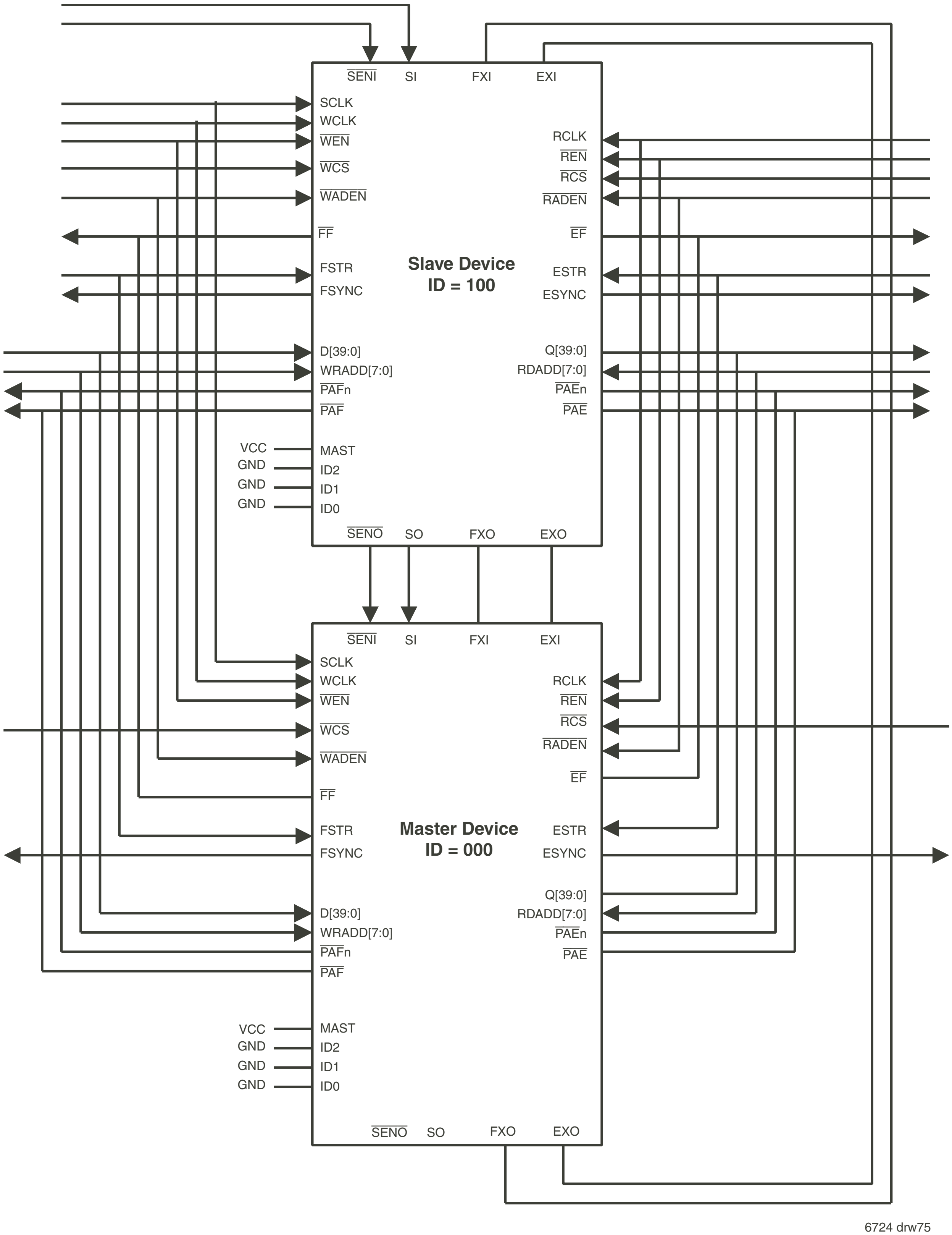
Figure 59. PAFn - Direct Mode, Flag Operation



NOTE:

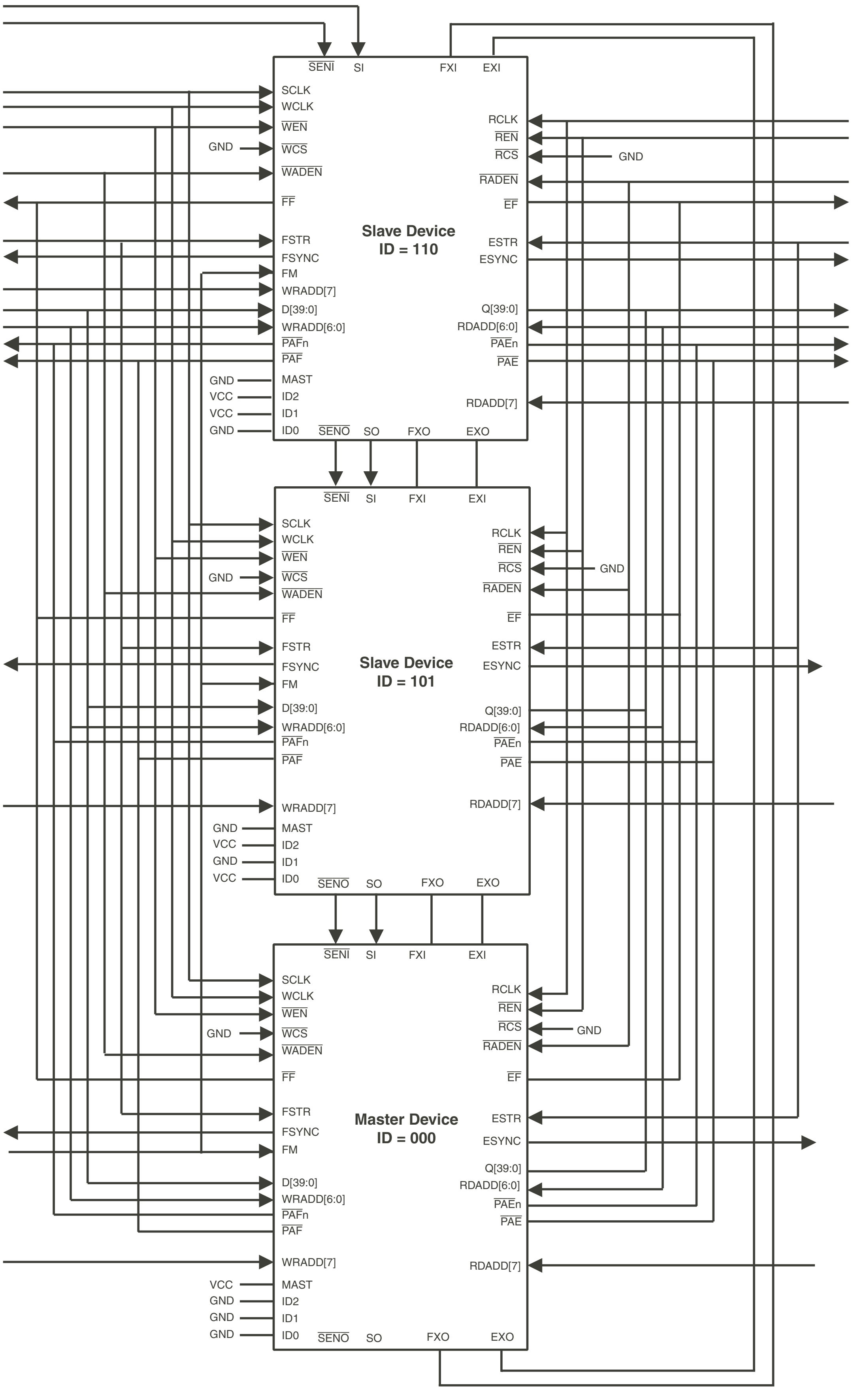
1. This diagram is based on 3 devices connected in expansion configuration.

Figure 60. $\overline{\textbf{PAF}}$n Bus - Polled Mode



NOTE:
1. ID2 MUST be unique between the devices.

Figure 61. Connecting two 10G MQ 128Q devices in Expansion Mode



6724 drw75a

Figure 62. Connecting THREE or more 10G MQ 128Q in Expansion Mode Using WADDR bit 7/RDADD bit 7

JTAG INTERFACE

Five additional pins (TDI, TDO, TMS, TCK and $\overline{\text{TRST}}$) are provided to support the JTAG boundary scan interface. The IDT72P51767/72P51777 incorporates the necessary tap controller and modified pad cells to implement the JTAG facility.

Note that IDT provides appropriate Boundary Scan Description Language program files for these devices.

The Standard JTAG interface consists of four basic elements:

- *Test Access Port (TAP)*
- *TAP controller*
- *Instruction Register (IR)*
- *Data Register Port (DR)*

The following sections provide a brief description of each element. For a complete description refer to the IEEE Standard Test Access Port Specification (IEEE Std. 1149.1-1990).

The Figure below shows the standard Boundary-Scan Architecture

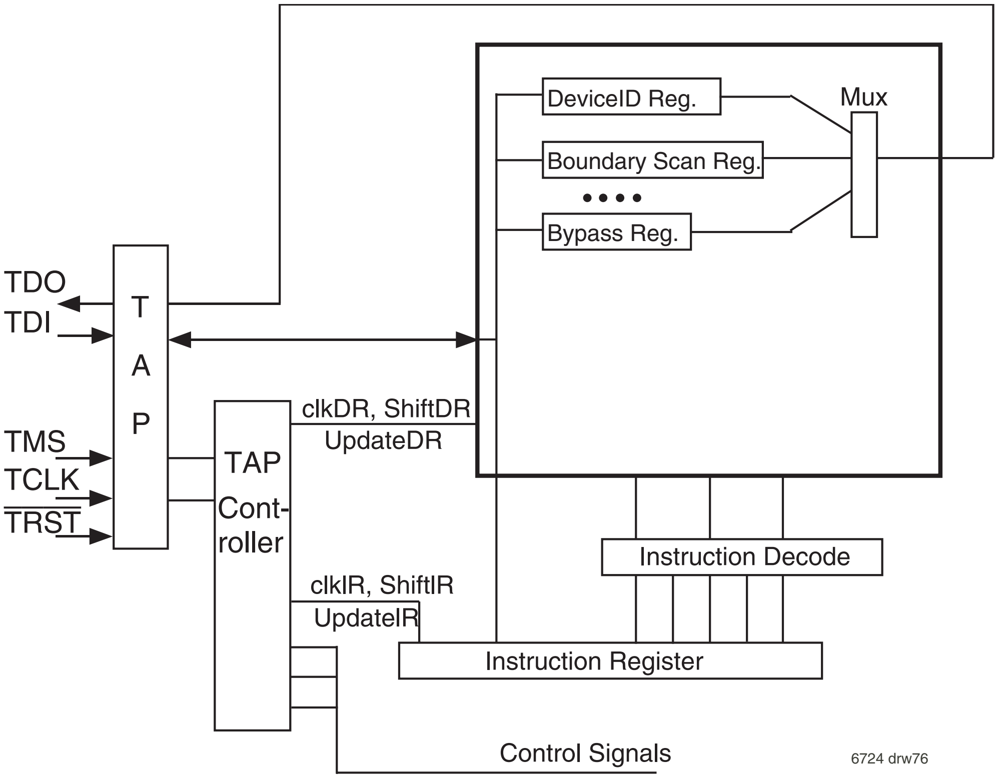


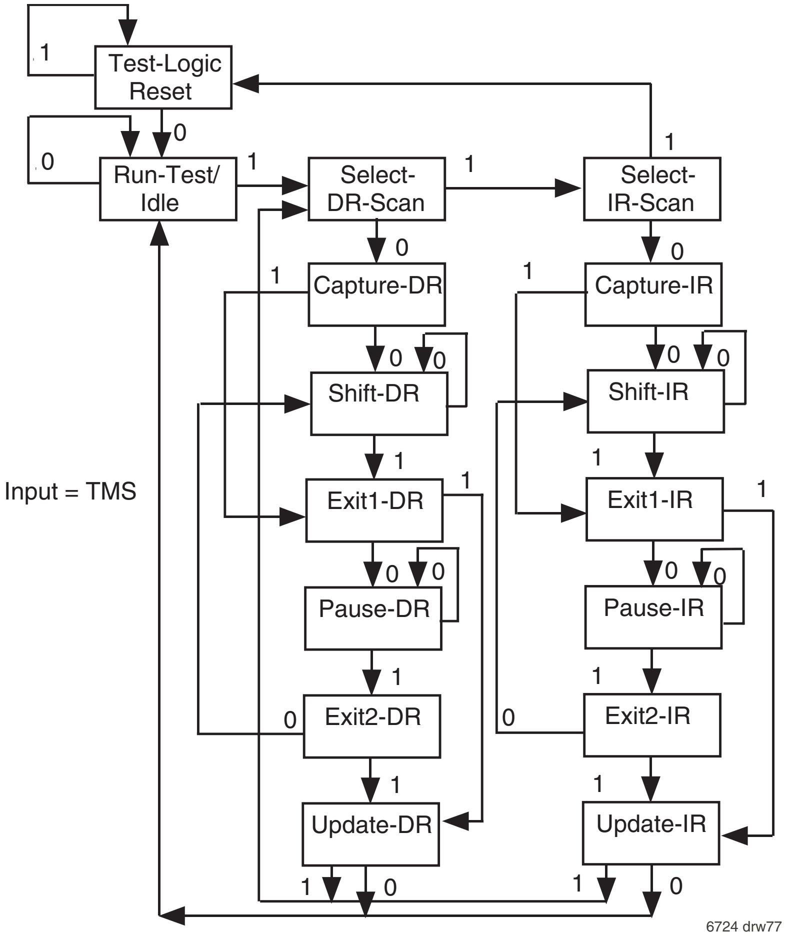
Figure 63. Boundary Scan Architecture

TEST ACCESS PORT (TAP)

The Tap interface is a general-purpose port that provides access to the internal of the processor. It consists of four input ports (TCLK, TMS, TDI, $\overline{\text{TRST}}$) and one output port (TDO).

THE TAP CONTROLLER

The Tap controller is a synchronous finite state machine that responds to TMS and TCLK signals to generate clock and control signals to the Instruction and Data Registers for capture and update of data.



NOTES:
1. Five consecutive TCK cycles with TMS = 1 will reset the TAP.
2. TAP controller does not automatically reset upon power-up. The user must provide a reset to the TAP controller (either by $\overline{\text{TRST}}$ or TMS).
3. TAP controller must be reset before normal Queue operations can begin.

Figure 64. TAP Controller State Diagram

Refer to the IEEE Standard Test Access Port Specification (IEEE Std. 1149.1) for the full state diagram.

All state transitions within the TAP controller occur at the rising edge of the TCLK pulse. The TMS signal level (0 or 1) determines the state progression that occurs on each TCLK rising edge. The TAP controller takes precedence over the Queue and must be reset after power up of the device. See $\overline{\text{TRST}}$ description for more details on TAP controller reset.

Test-Logic-Reset All test logic is disabled in this controller state enabling the normal operation of the IC. The TAP controller state machine is designed in such a way that, no matter what the initial state of the controller is, the Test-Logic-Reset state can be entered by holding TMS at high and pulsing TCK five times. This is the reason why the Test Reset ($\overline{\text{TRST}}$) pin is optional.

Run-Test-Idle In this controller state, the test logic in the IC is active only if certain instructions are present. For example, if an instruction activates the self test, then it will be executed when the controller enters this state. The test logic in the IC is idles otherwise.

Select-DR-Scan This is a controller state where the decision to enter the Data Path or the Select-IR-Scan state is made.

Select-IR-Scan This is a controller state where the decision to enter the Instruction Path is made. The Controller can return to the Test-Logic-Reset state other wise.

Capture-IR In this controller state, the shift register bank in the Instruction Register parallel loads a pattern of fixed values on the rising edge of TCK. The last two significant bits are always required to be "01".

Shift-IR In this controller state, the instruction register gets connected between TDI and TDO, and the captured pattern gets shifted on each rising edge of TCK. The instruction available on the TDI pin is also shifted in to the instruction register.

Exit1-IR This is a controller state where a decision to enter either the Pause-IR state or Update-IR state is made.

Pause-IR This state is provided in order to allow the shifting of instruction register to be temporarily halted.

Exit2-DR This is a controller state where a decision to enter either the Shift-IR state or Update-IR state is made.

Update-IR In this controller state, the instruction in the instruction register is latched in to the latch bank of the Instruction Register on every falling edge of TCK. This instruction also becomes the current instruction once it is latched.

Capture-DR In this controller state, the data is parallel loaded in to the data registers selected by the current instruction on the rising edge of TCK.

Shift-DR, Exit1-DR, Pause-DR, Exit2-DR and Update-DR These controller states are similar to the Shift-IR, Exit1-IR, Pause-IR, Exit2-IR and Update-IR states in the Instruction path.

THE INSTRUCTION REGISTER

The Instruction register allows an instruction to be shifted in serially into the processor at the rising edge of TCLK.

The Instruction is used to select the test to be performed, or the test data register to be accessed, or both. The instruction shifted into the register is latched at the completion of the shifting process when the TAP controller is at Update-IR state.

The instruction register must contain 4 bit instruction register-based cells which can hold instruction data. These mandatory cells are located nearest the serial outputs they are the least significant bits.

TEST DATA REGISTER

The Test Data register contains three test data registers: the Bypass, the Boundary Scan register and Device ID register.

These registers are connected in parallel between a common serial input and a common serial data output.

The following sections provide a brief description of each element. For a complete description, refer to the IEEE Standard Test Access Port Specification (IEEE Std. 1149.1-1990).

TEST BYPASS REGISTER

The register is used to allow test data to flow through the device from TDI to TDO. It contains a single stage shift register for a minimum length in serial path. When the bypass register is selected by an instruction, the shift register stage is set to a logic zero on the rising edge of TCLK when the TAP controller is in the Capture-DR state.

The operation of the bypass register should not have any effect on the operation of the device in response to the BYPASS instruction.

THE BOUNDARY-SCAN REGISTER

The Boundary Scan Register allows serial data TDI be loaded in to or read out of the processor input/output ports. The Boundary Scan Register is a part of the IEEE 1149.1-1990 Standard JTAG Implementation.

THE DEVICE IDENTIFICATION REGISTER

The Device Identification Register is a Read Only 32-bit register used to specify the manufacturer, part number and version of the processor to be determined through the TAP in response to the IDCODE instruction.

IDT JEDEC ID number is 0xB3. This translates to 0x33 when the parity is dropped in the 11-bit Manufacturer ID field.

For the IDT72P51767/72P51777, the Part Number field contains the following values:

Device	Part# Field (HEX)
IDT72P51767	048f
IDT72P51777	048e

31(MSb) 28	27 12	11 1	0(LSB)
Version (4 bits) 0X0	Part Number (16-bit)	Manufacturer ID (11-bit) 0X33	1

JTAG DEVICE IDENTIFICATION REGISTER

JTAG INSTRUCTION REGISTER

The Instruction register allows instruction to be serially input into the device when the TAP controller is in the Shift-IR state. The instruction is decoded to perform the following:

- Select test data registers that may operate while the instruction is current. The other test data registers should not interfere with chip operation and the selected data register.
- Define the serial test data register path that is used to shift data between TDI and TDO during data register scanning.

The Instruction Register is a 4 bit field (i.e. IR3, IR2, IR1, IR0) to decode 16 different possible instructions. Instructions are decoded as follows.

Hex Value	Instruction	Function
00	EXTEST	Select Boundary Scan Register
01	SAMPLE/PRELOAD	Select Boundary Scan Register
02	IDCODE	Select Chip Identification data register
03	HIGH-IMPEDANCE	JTAG
0F	BYPASS	Select Bypass Register

JTAG INSTRUCTION REGISTER DECODING

The following sections provide a brief description of each instruction. For a complete description refer to the IEEE Standard Test Access Port Specification (IEEE Std. 1149.1-1990).

EXTEST

The required EXTEST instruction places the IC into an external boundary-test mode and selects the boundary-scan register to be connected between TDI and TDO. During this instruction, the boundary-scan register is accessed to drive test data off-chip via the boundary outputs and receive test data off-chip via the boundary inputs. As such, the EXTEST instruction is the workhorse of IEEE. Std 1149.1, providing for probe-less testing of solder-joint opens/shorts and of logic cluster function.

IDCODE

The optional IDCODE instruction allows the IC to remain in its functional mode and selects the optional device identification register to be connected between TDI and TDO. The device identification register is a 32-bit shift register containing information regarding the IC manufacturer, device type, and version code. Accessing the device identification register does not interfere with the operation of the IC. Also, access to the device identification register should be immediately available, via a TAP data-scan operation, after power-up of the IC or after the TAP has been reset using the optional $\overline{\text{TRST}}$ pin or by otherwise moving to the Test-Logic-Reset state.

SAMPLE/PRELOAD

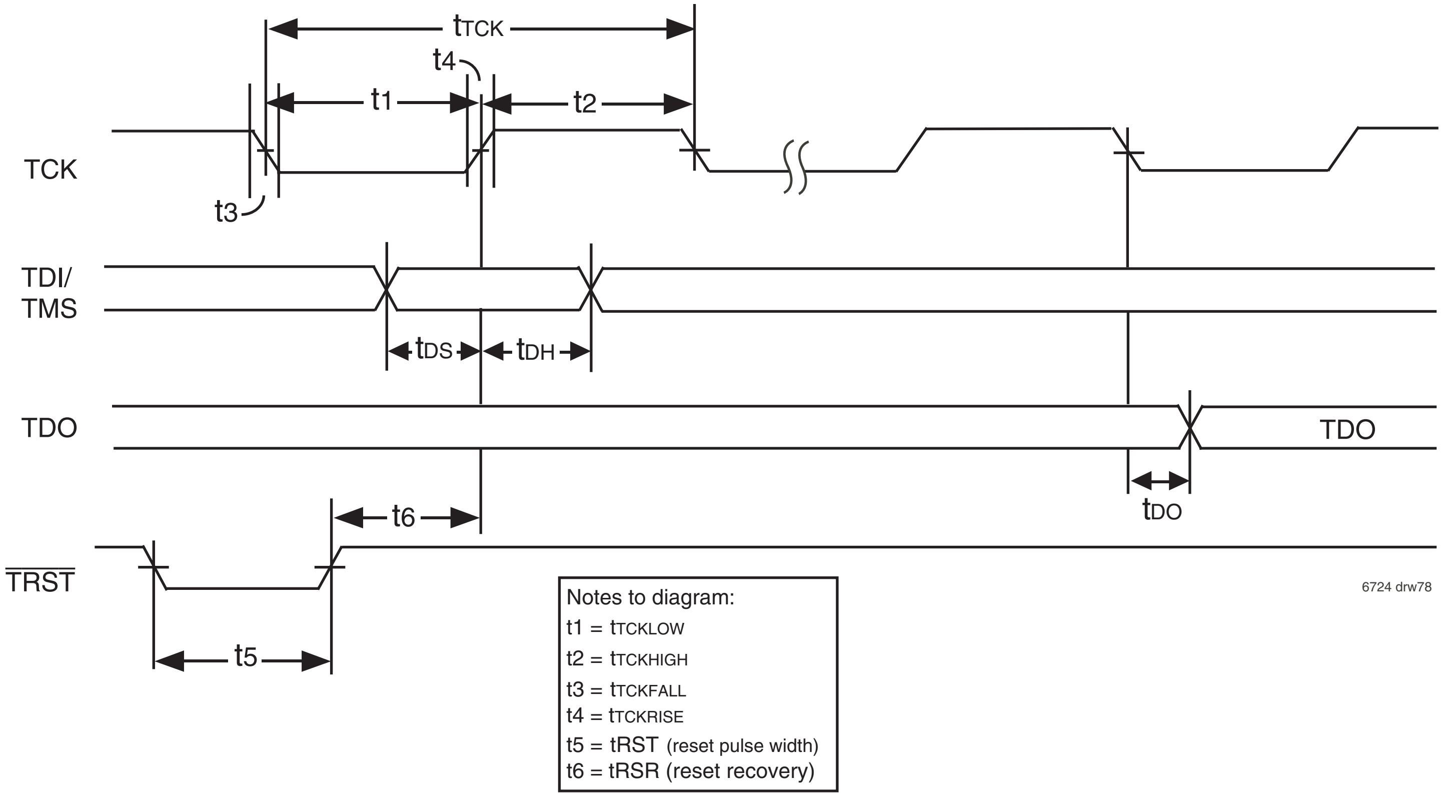
The required SAMPLE/PRELOAD instruction allows the IC to remain in a normal functional mode and selects the boundary-scan register to be connected between TDI and TDO. During this instruction, the boundary-scan register can be accessed via a date scan operation, to take a sample of the functional data entering and leaving the IC. This instruction is also used to preload test data into the boundary-scan register before loading an EXTEST instruction.

HIGH-IMPEDANCE

The optional High-Impedance instruction sets all outputs (including two-state as well as three-state types) of an IC to a disabled (high-impedance) state and selects the one-bit bypass register to be connected between TDI and TDO. During this instruction, data can be shifted through the bypass register from TDI to TDO without affecting the condition of the IC outputs.

BYPASS

The required BYPASS instruction allows the IC to remain in a normal functional mode and selects the one-bit bypass register to be connected between TDI and TDO. The BYPASS instruction allows serial data to be transferred through the IC from TDI to TDO without affecting the operation of the IC.



Notes to diagram:
t1 = tTCKLOW
t2 = tTCKHIGH
t3 = tTCKFALL
t4 = tTCKRISE
t5 = tRST (reset pulse width)
t6 = tRSR (reset recovery)

Figure 65. Standard JTAG Timing

SYSTEM INTERFACE PARAMETERS

Parameter	Symbol	Test Conditions	IDT72P51767 IDT72P51777		
			Min.	Max.	Units
Data Output	tDO[1]		-	20	ns
Data Output Hold	tDOH[1]		0	-	ns
Data Input	tDS	trise=3ns	10	-	ns
	tDH	tfall=3ns	10	-	

NOTE:
1. 50pf loading on external output signals.

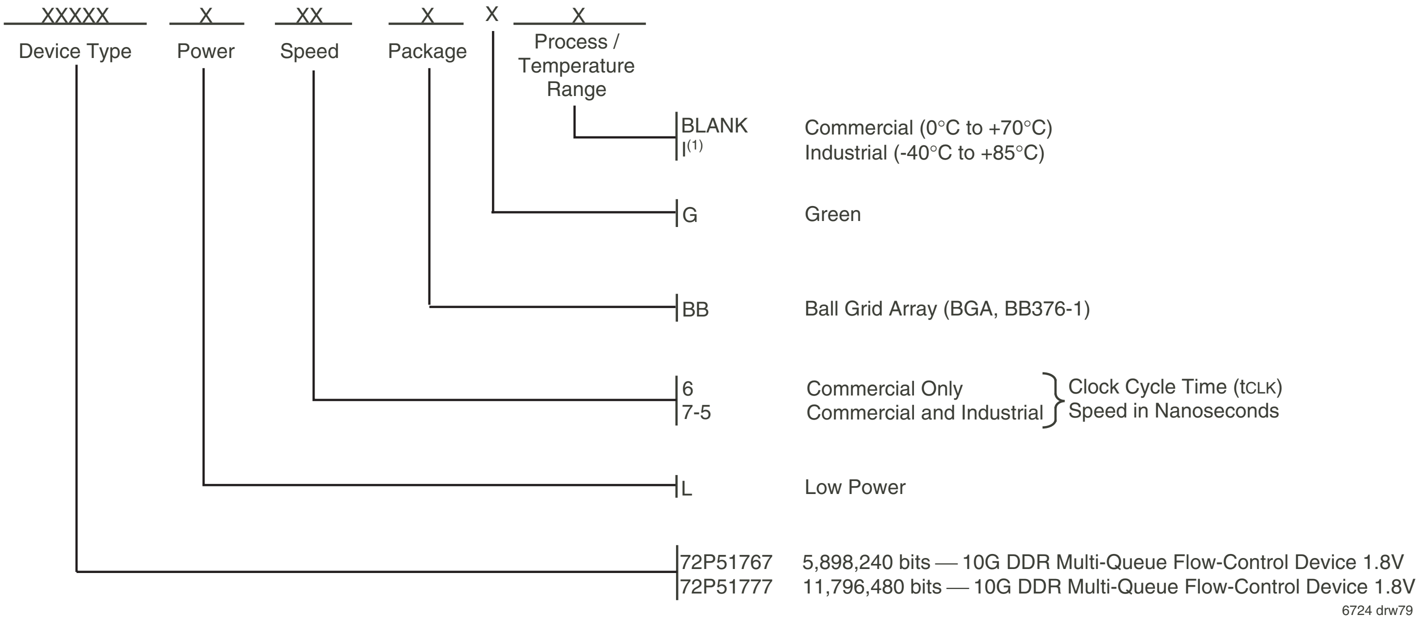
JTAG AC ELECTRICAL CHARACTERISTICS

(VDD = 2.5V ± 5%; Tcase = 0°C to +85°C)

Parameter	Symbol	Test Conditions	IDT72P51767 IDT72P51777		
			Min.	Max.	Units
JTAG Clock Input Period	tTCK	-	100	-	ns
JTAG Clock HIGH	tTCKHIGH	-	40	-	ns
JTAG Clock Low	tTCKLOW	-	40	-	ns
JTAG Clock Rise Time	tTCKRISE	-	-	5[1]	ns
JTAG Clock Fall Time	tTCKFALL	-	-	5[1]	ns
JTAG Reset	tRST	-	50	-	ns
JTAG Reset Recovery	tRSR	-	50	-	ns

NOTE:
1. Guaranteed by design.

ORDERING INFORMATION



XXXXX Device Type — X Power — XX Speed — X Package — X — X Process / Temperature Range

Process / Temperature Range	BLANK	Commercial (0°C to +70°C)
	I[1]	Industrial (-40°C to +85°C)
	G	Green
Package	BB	Ball Grid Array (BGA, BB376-1)
Speed	6	Commercial Only } Clock Cycle Time (tCLK) Speed in Nanoseconds
	7-5	Commercial and Industrial } Clock Cycle Time (tCLK) Speed in Nanoseconds
Power	L	Low Power
Device Type	72P51767	5,898,240 bits — 10G DDR Multi-Queue Flow-Control Device 1.8V
	72P51777	11,796,480 bits — 10G DDR Multi-Queue Flow-Control Device 1.8V

6724 drw79

NOTES:

1. Industrial temperature range product for the 7-5ns is available as a standard device. All other speed grades available by special order.
2. Green parts are available. For specific speeds contact your local sales office.

DATASHEET DOCUMENT HISTORY

01/18/2006	pgs.	1, 17, 19, 20, 30, and 31.
02/11/2009	pg.	88.

CORPORATE HEADQUARTERS
6024 Silver Creek Valley Road
San Jose, CA 95138

for SALES:
800-345-7015 or 408-284-8200
fax: 408-284-2775
www.idt.com

for Tech Support:
408-360-1533
email: Flow-Controlhelp@idt.com