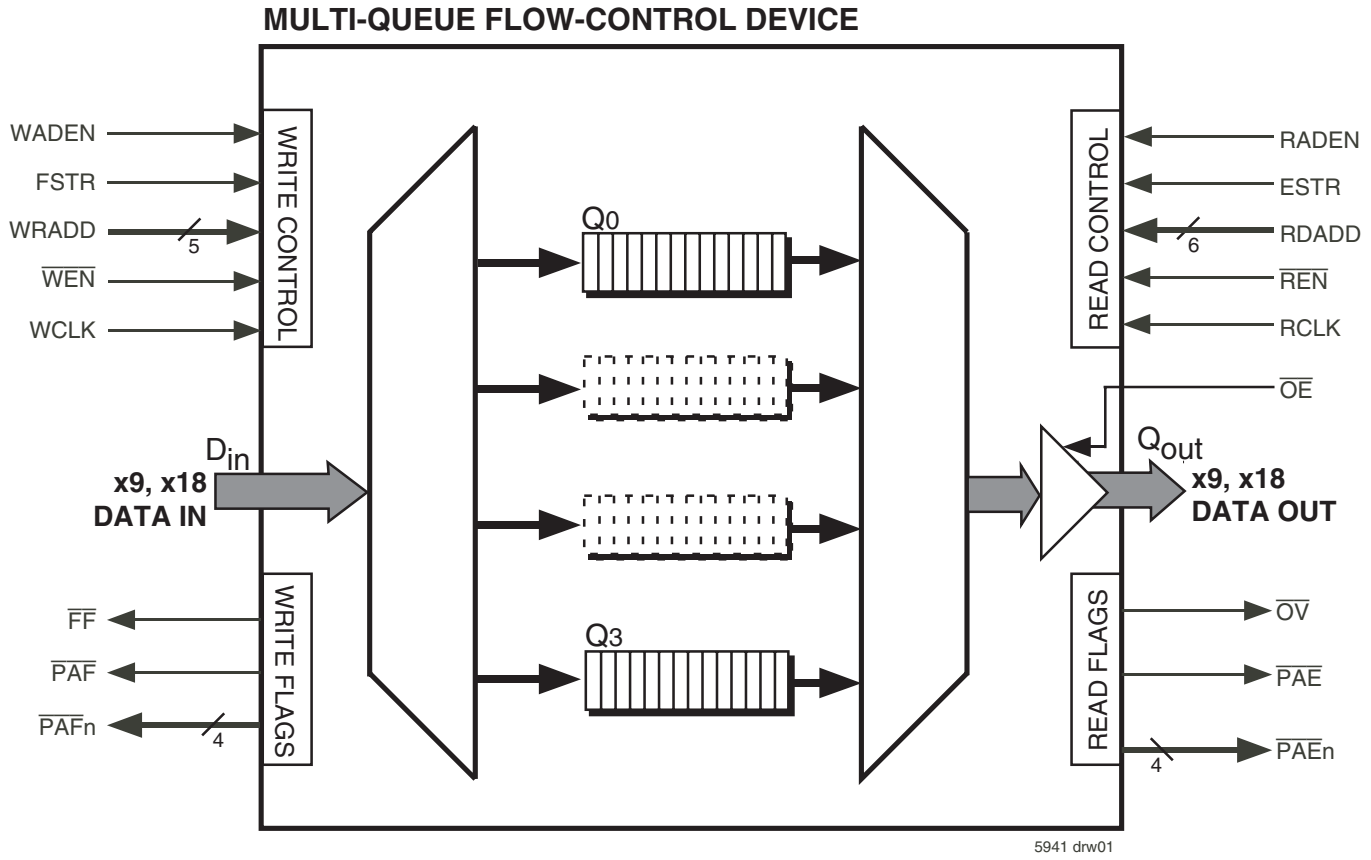
3.3V MULTI-QUEUE FLOW-CONTROL DEVICES
(4 QUEUES) 18 BIT WIDE CONFIGURATION
589,824 bits
1,179,648 bits
2,359,296 bits

IDT72V51233
IDT72V51243
IDT72V51253

FEATURES:

- Choose from among the following memory density options:
  IDT72V51233 — Total Available Memory = 589,824 bits
  IDT72V51243 — Total Available Memory = 1,179,648 bits
  IDT72V51253 — Total Available Memory = 2,359,296 bits
- Configurable from 1 to 4 Queues
- Queues may be configured at master reset from the pool of Total Available Memory in blocks of 512 x 18 or 1,024 x 9
- Independent Read and Write access per queue
- User programmable via serial port
- Default multi-queue device configurations
  -IDT72V51233: 8,192 x 18 x 4Q or 16,384 x 9 x 4Q
  -IDT72V51243: 16,384 x 18 x 4Q or 32,768 x 9 x 4Q
  -IDT72V51253: 32,768 x 18 x 4Q or 65,536 x 9 x 4Q
- 100% Bus Utilization, Read and Write on every clock cycle
- 166 MHz High speed operation (6ns cycle time)
- 3.7ns access time
- Individual, Active queue flags ($\overline{OV}$, $\overline{FF}$, $\overline{PAE}$, $\overline{PAF}$)
- 4 bit parallel flag status on both read and write ports
- Provides continuous $\overline{PAE}$ and $\overline{PAF}$ status of up to 4 Queues
- Global Bus Matching - (All Queues have same Input Bus Width and Output Bus Width)
- User Selectable Bus Matching Options:
  - x18in to x18out
  - x9in to x18out
  - x18in to x9out
  - x9in to x9out
- FWFT mode of operation on read port
- Partial Reset, clears data in single Queue
- Expansion of up to 8 multi-queue devices in parallel is available
- JTAG Functionality (Boundary Scan)
- Available in a 256-pin PBGA, 1mm pitch, 17mm x 17mm
- HIGH Performance submicron CMOS technology
- Industrial temperature range (-40°C to +85°C) is available

FUNCTIONAL BLOCK DIAGRAM



IDT and the IDT logo are registered trademarks of Integrated Device Technology, Inc.

© 2019 Renesas Electronics Corporation

DESCRIPTION:

The IDT72V51233/72V51243/72V51253 multi-queue flow-control devices are single chip within which anywhere between 1 and 4 discrete FIFO queues can be setup. All queues within the device have a common data input bus, (write port) and a common data output bus, (read port). Data written into the write port is directed to a respective queue via an internal de-multiplex operation, addressed by the user. Data read from the read port is accessed from a respective queue via an internal multiplex operation, addressed by the user. Data writes and reads can be performed at high speeds up to 166MHz, with access times of 3.7ns. Data write and read operations are totally independent of each other, a queue maybe selected on the write port and a different queue on the read port or both ports may select the same queue simultaneously.

The device provides Full flag and Output Valid flag status for the queue selected for write and read operations respectively. Also a Programmable Almost Full and Programmable Almost Empty flag for each queue is provided. Two 4 bit programmable flag busses are available, providing status of all queues, including queues not selected for write or read operations, these flag busses provide an individual flag per queue.

Bus Matching is available on this device, either port can be 9 bits or 18 bits wide. When Bus Matching is used the device ensures the logical transfer of data throughput in a Little Endian manner.

The user has full flexibility configuring queues within the device, being able to program the total number of queues between 1 and 4, the individual queue depths being independent of each other. The programmable flag positions are also user programmable. All programming is done via a dedicated serial port. 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.

Both Master Reset and Partial Reset pins are provided on this device. A Master Reset latches in all configuration setup pins and must be performed before programming of the device can take place. A Partial Reset will reset the read and write pointers of an individual queue, provided that the queue is selected on both the write port and read port at the time of partial reset.

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.

See Figure 1, *Multi-Queue Flow-Control Device Block Diagram* for an outline of the functional blocks within the device.

© 2019 Renesas Electronics Corporation

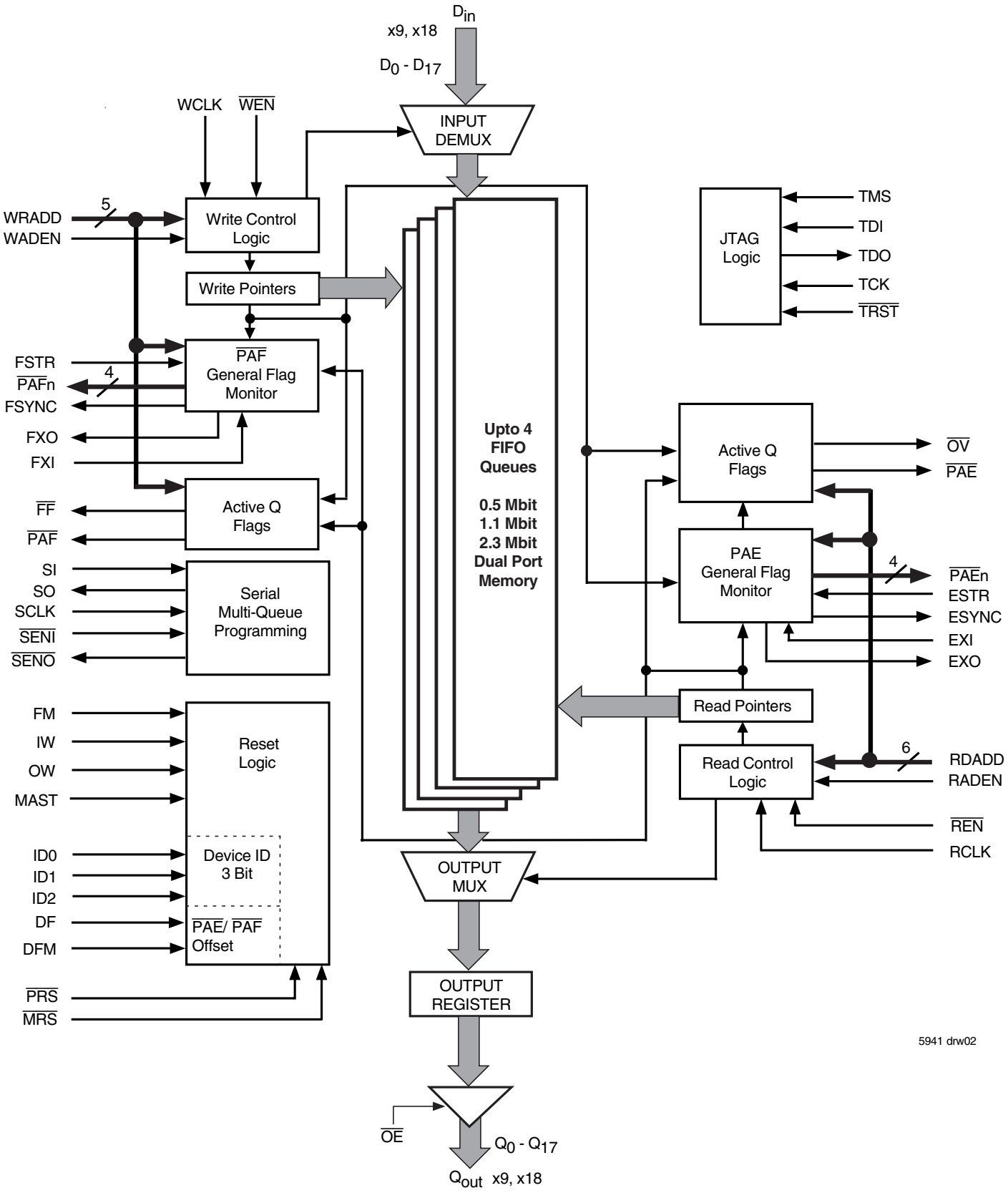
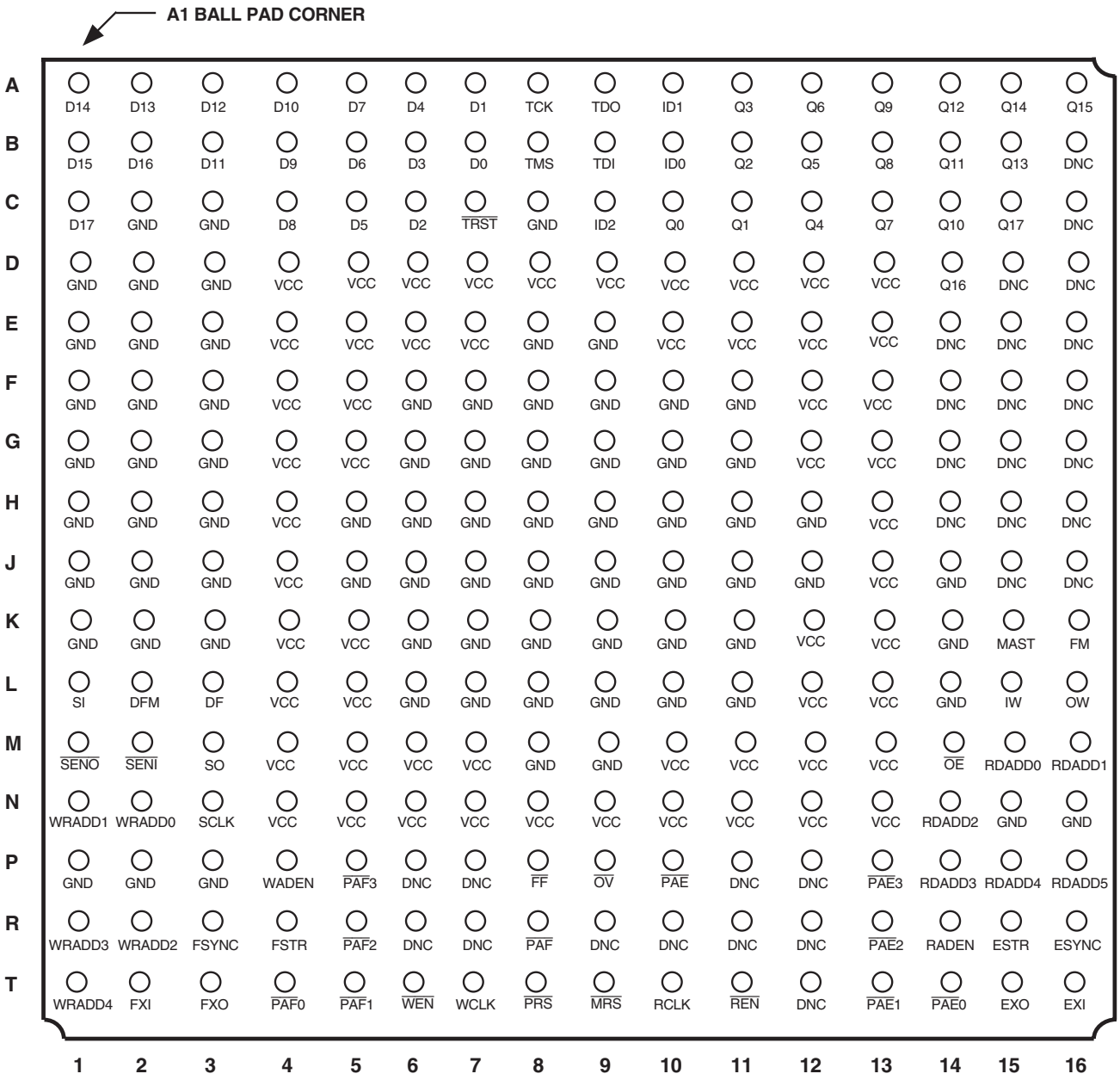


Figure 1. Multi-Queue Flow-Control Device Block Diagram

© 2019 Renesas Electronics Corporation

PIN CONFIGURATION



NOTE:

1. DNC - Do Not Connect.

5941 drw03

PBGA (BB256-1, order code: BB)
TOP VIEW

© 2019 Renesas Electronics Corporation

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 4 FIFO queues in parallel buffering between the two ports. The user can setup between 1 and 4 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 512 x 18 or 1,024 x 9 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 512 x 18 or 1,024 x 9. For the IDT72V51233, IDT72V51243 and IDT72V51253 the Total Available Memory is 64, 128 and 256 blocks respectively (a block being 512 x 18 or 1,024 x 9). If any port is configured for x18 bus width, a block size is 512 x 18. If both the write and read ports are configured for x9 bus width, a block size is 1,024 x 9. 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 x9 or x18 bits wide, the read and write port widths being set independently of one another. Because the ports are common to all queues the width of the queues is not individually set, so that the input width of all queues are equal and the output width of all queues are equal.

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 select address inputs. Conversely, data being read from the device read port is read from a queue selected via the read queue select address inputs. 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 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. The operation of the read port is comparable to the function of a conventional FIFO operating in FWFT mode. When a queue is selected on the output port, the next word in that queue will automatically fall through to 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 Output Valid 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 register.

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 4 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 output valid flag, providing status of the data being read from the queue selected on the read port. As well as the output valid 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 all 4 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 valid & almost empty on the read port, there are two flag status busses. An almost full flag status bus is provided, this bus is 4 bits wide. Also, an almost empty flag status bus is provided, again this bus is 4 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 4 queues in the device.

The 4 bit $\overline{PAE}$n and 4 bit $\overline{PAF}$n busses provide a discrete status of the Almost Empty and Almost Full conditions of all 4 queue's. If the device is programmed for less than 4 queue's, then there will be a corresponding number of active outputs on the $\overline{PAE}$n and $\overline{PAF}$n busses.

The flag busses can provide a continuous status of all queues. If devices are connected in expansion mode the individual flag busses can be left in a discrete form, providing constant status of all queues, or the busses of individual devices can be connected together to produce a single bus of 4 bits. The device can then operate in a "Polled" or "Direct" mode.

When operating in polled mode the flag bus provides status of each device sequentially, that is, on each rising edge of a clock the flag bus is updated to show the status of each device 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.

When operating in direct mode the device driving the flag bus is selected by the user. The user addresses the device that will take control of a respective flag bus, these $\overline{PAF}$n and $\overline{PAE}$n flag busses operating 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 also possible, up to 8 devices can be connected in a parallel fashion providing the possibility of both depth expansion 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 increase the depth of a queue. For example, depth expansion of 8 devices provides the possibility of 8 queues of 32K x 18 deep within the IDT72V51233, 64K x 18 deep within the IDT72V51243 and 128K x 18 deep within the IDT72V51253, 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 of the 4 queue device, a maximum number of 32 (8 x 4) queues may be setup, each queue being 4K x18 or 2K x 9 deep, if less queues are setup, then more memory blocks will be available to increase queue depths if desired. When connecting multi-queue devices in expansion mode all respective input pins (data & control) and output pins (data & flags), should be "connected" together between individual devices.

© 2019 Renesas Electronics Corporation

PIN DESCRIPTIONS

Symbol & Pin No.	Name	I/O TYPE	Description
D[17:0] Din (See Pin table for details)	Data Input Bus	LVTTL INPUT	These are the 18 data input pins. Data is written into the device via these input pins on the rising edge of WCLK provided that $\overline{WEN}$ is LOW. Due to bus matching not all inputs may be used, any unused inputs should be tied LOW.
DF[1] (L3)	Default Flag	LVTTL INPUT	If the user requires default programming of the multi-queue device, this pin must be setup before Master Reset and must not toggle during any device operation. The state of this input at master reset determines the value of the $\overline{PAE}$/$\overline{PAF}$ flag offsets. If DF is LOW the value is 8, if DF is HIGH the value is 128.
DFM[1] (L2)	Default Mode	LVTTL INPUT	The 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 HIGH then default mode is selected.
ESTR (R15)	$\overline{PAE}$n Flag Bus Strobe	LVTTL INPUT	If direct operation of the $\overline{PAE}$n bus has been selected, the ESTR input is used in conjunction with RCLK and the RDADD bus to select a device for its queues to be placed on to the $\overline{PAE}$n bus outputs. A device addressed via the RDADD bus is selected on the rising edge of RCLK provided that ESTR is HIGH. If Polled operations has been selected, ESTR should be tied inactive, LOW. Note, that a $\overline{PAE}$n flag bus selection cannot be made, (ESTR must NOT go active) until programming of the part has been completed and $\overline{SENO}$ has gone LOW.
ESYNC (R16)	$\overline{PAE}$n Bus Sync	LVTTL OUTPUT	ESYNC is an output from the multi-queue device that provides a synchronizing pulse for the $\overline{PAE}$n bus during Polled operation of the $\overline{PAE}$n bus. During Polled operation each devices queue status flags are loaded on to the $\overline{PAE}$n bus outputs sequentially based on RCLK. The first RCLK rising edge loads device 1 on to $\overline{PAE}$n, the second RCLK rising edge loads device 2 and so on. During the RCLK cycle that a selected device is placed on to the $\overline{PAE}$n bus, the ESYNC output will be HIGH.
EXI (T16)	$\overline{PAE}$n Bus Expansion In	LVTTL INPUT	The EXI input is used when multi-queue devices are connected in expansion mode and Polled $\overline{PAE}$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 $\overline{PAE}$n bus is operated in direct mode. If the $\overline{PAE}$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 (T15)	$\overline{PAE}$n Bus Expansion Out	LVTTL OUTPUT	EXO is an output that is used when multi-queue devices are connected in expansion mode and Polled $\overline{PAE}$n bus operation has been selected . EXO of device 'N' connects directly to EXI of device 'N+1'. This pin pulses HIGH when device N places its $\overline{PAE}$ status on to the $\overline{PAE}$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 $\overline{PAE}$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.
$\overline{FF}$ (P8)	Full Flag	LVTTL 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 on the next cycle provided $\overline{FF}$ is HIGH. This flag has High-Impedance capability, this is important during expansion of devices, when the $\overline{FF}$ 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 $\overline{FF}$ bus, all other devices place their $\overline{FF}$ 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 synchronized to WCLK.
FM[1] (K16)	Flag Mode	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 $\overline{PAF}$n and $\overline{PAE}$n flag busses operate in either Polled or Direct mode. If this pin is HIGH the mode is Polled, if LOW then it will be Direct.
FSTR (R4)	$\overline{PAF}$n Flag Bus Strobe	LVTTL INPUT	If direct operation of the $\overline{PAF}$n bus has been selected, the FSTR input is used in conjunction with WCLK and the WRADD bus to select a device for its queues to be placed on to the $\overline{PAF}$n bus outputs. A device 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 $\overline{PAF}$n flag bus selection cannot be made, (FSTR must NOT go active) until programming of the part has been completed and $\overline{SENO}$ has gone LOW.

© 2019 Renesas Electronics Corporation

PIN DESCRIPTIONS (CONTINUED)

Symbol & Pin No.	Name	I/O TYPE	Description
FSYNC (R3)	$\overline{PAF}$n Bus Sync	LVTTL OUTPUT	FSYNC is an output from the multi-queue device that provides a synchronizing pulse for the $\overline{PAF}$n bus during Polled operation of the $\overline{PAF}$n bus. During Polled operation each quadrant of queue status flags is loaded on to the $\overline{PAF}$n bus outputs sequentially based on WCLK. The first WCLK rising edge loads device1 on to the $\overline{PAF}$n bus outputs, the second WCLK rising edge loads device 2 and so on. During the WCLK cycle that a selected device is placed on to the $\overline{PAF}$n bus, the FSYNC output will be HIGH.
FXI (T2)	$\overline{PAF}$n Bus Expansion In	LVTTL INPUT	The FXI input is used when multi-queue devices are connected in expansion mode and Polled $\overline{PAF}$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 $\overline{PAF}$n bus is operated in direct mode. If the $\overline{PAF}$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 (T3)	$\overline{PAF}$n Bus Expansion Out	LVTTL OUTPUT	FXO is an output that is used when multi-queue devices are connected in expansion mode and Polled $\overline{PAF}$n bus operation has been selected . FXO of device 'N' connects directly to FXI of device 'N+1'. This pin pulses HIGH when device N places its $\overline{PAF}$ status on to the $\overline{PAF}$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 $\overline{PAF}$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][1] (ID2-C9 ID1-A10 ID0-B10)	Device ID Pins	LVTTL INPUT	For the 4Q multi-queue device the WRADD address bus is 5 bits and RDADD address bus is 6 bits wide. When a queue selection takes place the 3 MSb's of this address bus are used to address the specific device (the LSb's are used to address the queue within that device). During write/read operations the 3 MSb's of the address are compared to the device ID pins. The first device in a chain of multi-queue's (connected in expansion mode), may be setup as '000', the second as '001' and so on through to device 8 which is '111', however the ID does not have to match the device order. In single device mode these pins should be setup as '000' and the 3 MSb's of the WRADD and RDADD address busses should be tied LOW. The ID[2:0] inputs setup a respective devices 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'.
IW[1] (L15)	Input Width	LVTTL INPUT	IW selects the bus width for the data input bus. If IW is LOW during a Master Reset then the bus width is x18, if HIGH then it is x9.
MAST[1] (K15)	Master Device	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.
$\overline{MRS}$ (T9)	Master Reset	LVTTL INPUT	A master reset is performed by taking $\overline{MRS}$ from HIGH to LOW, to HIGH. Device programming is required after master reset.
$\overline{OE}$ (M14)	Output Enable	LVTTL INPUT	The Output enable signal is an Asynchronous signal used to provide three-state control of the multi-queue data output bus, 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.
$\overline{OV}$ (P9)	Output Valid Flag	LVTTL OUTPUT	This output flag provides output valid status for the data word present on the multi-queue flow-control device data output port, Qout. This flag is therefore, 2-stage delayed to match the data output path delay. That is, there is a 2 RCLK cycle delay from the time a given queue is selected for reads, to the time the $\overline{OV}$ flag represents the data in that respective queue. When a selected queue on the read port is read to empty, the $\overline{OV}$ flag will go HIGH, indicating that data on the output bus is not valid. The $\overline{OV}$ flag also has High-Impedance capability, required when multiple devices are used and the $\overline{OV}$ flags are tied together.
OW[1] (L16)	Output Width	LVTTL INPUT	OW selects the bus width for the data output bus. If OW is LOW during a Master Reset then the bus width is x18, if HIGH then it is x9.

© 2019 Renesas Electronics Corporation

PIN DESCRIPTIONS (CONTINUED)

Symbol & Pin No.	Name	I/O TYPE	Description
$\overline{PAE}$ (P10)	Programmable Almost-Empty Flag	LVTTL 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 ($\overline{PAE}$3-P13 $\overline{PAE}$2-R13 $\overline{PAE}$1-T13 $\overline{PAE}$0-T14)	Programmable Almost-Empty Flag Bus	LVTTL OUTPUT	On the 4Q device the $\overline{PAE}$n bus is 4 bits wide. This output bus provides $\overline{PAE}$ status of all 4 queues, within a selected device. During queue read/write operations these outputs provide programmable empty flag status in either director 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 $\overline{PAE}$n bus is updated to show the $\overline{PAE}$ status of queues within a selected device. Selection is made using RCLK, ESTR and Flag Bus RDADD. During Polled operation the $\overline{PAE}$n bus is loaded with the $\overline{PAE}$ status of multi-queue flow-control devices sequentially based on the rising edge of RCLK.
$\overline{PAF}$ (R8)	Programmable Almost-Full Flag	LVTTL 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. This flag is synchronized to WCLK.
$\overline{PAF}$n ($\overline{PAE}$3-P5 $\overline{PAE}$2-R5 $\overline{PAE}$1-T5 $\overline{PAE}$0-T4)	Programmable Almost-Full Flag Bus	LVTTL OUTPUT	On the 4Q device the $\overline{PAF}$n bus is 4 bits wide. This output bus provides $\overline{PAF}$ status of all 4 queues, 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 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 devices sequentially based on the rising edge of WCLK.
$\overline{PRS}$ (T8)	Partial Reset	LVTTL INPUT	A Partial Reset can be performed on a single queue selected within the multi-queue device. Before a Partial Reset can be performed on a queue, that queue must be selected on both the write port and read port 2 clock cycles before the reset is performed. A Partial Reset is then performed by taking $\overline{PRS}$ LOW for one WCLK cycle and one RCLK cycle. The Partial Reset will only reset the read and write pointers to the first memory location, none of the devices configuration will be changed.
Q[17:0] Qout (See Pin table for details)	Data Output Bus	LVTTL OUTPUT	These are the 18 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.
RADEN (R14)	Read Address Enable	LVTTL 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 $\overline{SENO}$ has gone LOW.
RCLK (T10)	Read Clock	LVTTL 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 device 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 $\overline{OV}$ 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.
RDADD [5:0] (See next page for details)	Read Address Bus	LVTTL INPUT	For the 4Q device the RDADD bus is 6 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 2 bits of the bus, RDADD[1:0] are used to address 1 of 4 possible queues within a multi-queue device. Address pin, RDADD[2] provides the user with a Null-Q address. If the user does not wish to address one of the 4 queues, a Null-Q can be addressed using this pin. The Null-Q operation is discussed in more detail later. The most significant 3 bits, RDADD[5:3] 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. The address present

© 2019 Renesas Electronics Corporation

PIN DESCRIPTIONS (CONTINUED)

Symbol & Pin No.	Name	I/O TYPE	Description
RDADD [5:0] (Continued) (RDADD5-P16 RDADD4-P15 RDADD3-P14 RDADD2-N14 RDADD1-M16 RDADD0-M15)	Read Address Bus	LVTTL INPUT	on the RDADD bus will be selected on a rising edge of RCLK provided that RADEN is HIGH, (note, that data can be placed on to the Qout bus, read from the previously selected queue on this RCLK edge). On the next rising RCLK edge after a read queue select, a data word from the previous queue will be placed onto the outputs, Qout, regardless of the $\overline{REN}$ input. Two RCLK rising edges after read queue select, data will be placed on to the Qout outputs from the newly selected queue, regardless of $\overline{REN}$ due to the first word fall through effect. The second function of the RDADD bus is to select the device of queues to be loaded on to the $\overline{PAE}$n bus during strobed flag mode. The most significant 3 bits, RDADD[5:3] are again used to select 1 of 8 possible multi-queue devices that may be connected in expansion mode. Address bits RDADD[2:0] are don't care during device selection. The device address present on the RDADD bus will be selected on the rising edge of RCLK provided that ESTR is HIGH, (note, that data can be placed on to the Qout bus, read from the previously selected queue on this RCLK edge). Please refer to Table 2 for details on RDADD bus.
$\overline{REN}$ (T11)	Read Enable	LVTTL 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 $\overline{REN}$ due to the FWFT operation. A read enable is not required to cycle the $\overline{PAE}$n bus (in polled mode) or to select the device , (in direct mode).
SCLK (N3)	Serial Clock	LVTTL 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 $\overline{SENI}$ is enabled, LOW. When expansion of devices is performed the SCLK of all devices should be connected to the same source.
$\overline{SENI}$ (M2)	Serial Input Enable	LVTTL 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 $\overline{SENI}$ input of that device is LOW. If multiple devices are cascaded, the $\overline{SENI}$ input should be connected to the $\overline{SENO}$ output of the previous device. So when serial loading of a given device is complete, its $\overline{SENO}$ output goes LOW, allowing the next device in the chain to be programmed ($\overline{SENO}$ will follow $\overline{SENI}$ of a given device once that device is programmed). The $\overline{SENI}$ input of the master device (or single device), should be controlled by the user.
$\overline{SENO}$ (M1)	Serial Output Enable	LVTTL OUTPUT	This output is used to indicate that serial programming or default programming of the multi-queue device has been completed. $\overline{SENO}$ 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. When the $\overline{SENO}$ 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 $\overline{SENO}$ output should be connected to the $\overline{SENI}$ input of the next device in the chain. When serial programming of the first device is complete, $\overline{SENO}$ will go LOW, thereby taking the $\overline{SENI}$ input of the next device LOW and so on throughout the chain. When a given device in the chain is fully programmed the $\overline{SENO}$ output essentially follows the $\overline{SENI}$ input. The user should monitor the $\overline{SENO}$ output of the final device in the chain. When this output goes LOW, serial loading of all devices has been completed.
SI (L1)	Serial In	LVTTL 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 $\overline{SENI}$ 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 $\overline{SENO}$ 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 (M3)	Serial Out	LVTTL 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[2] (A8)	JTAG Clock	LVTTL 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.
TDI[2] (B9)	JTAG Test Data Input	LVTTL INPUT	One of four terminals required by IEEE Standard 1149.1-1990. During the JTAG boundary scan operation, 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.

© 2019 Renesas Electronics Corporation

PIN DESCRIPTIONS (CONTINUED)

Symbol & Pin No.	Name	I/O TYPE	Description
TDO[2] (A9)	JTAG Test Data Output	LVTTL OUTPUT	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.
TMS[2] (B8)	JTAG Mode Select	LVTTL 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.
$\overline{TRST}$[2] (C7)	JTAG Reset	LVTTL 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 (P4)	Write Address Enable	LVTTL 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 $\overline{SENO}$ has gone LOW.
WCLK (T7)	Write Clock	LVTTL 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 device 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.
$\overline{WEN}$ (T6)	Write Enable	LVTTL 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 device , (in direct mode).
WRADD [4:0] (WRADD4-T1 WRADD3-R1 WRADD2-R2 WRADD1-N1 WRADD0-N2)	Write Address Bus	LVTTL INPUT	For the 4Q device the WRADD bus is 5 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 2 bits of the bus, WRADD[1:0] are used to address 1 of 4 possible queues within a multi-queue device. The most significant 3 bits, WRADD[4:2] 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. The address present on the WRADD bus will be selected on a rising edge of WCLK provided that WADEN is HIGH, (note, that data present on the Din bus can be written into the previously selected queue on this WCLK edge and on the next rising WCLK also, providing that $\overline{WEN}$ is LOW). Two WCLK rising edges after write queue select, data can be written into the newly selected queue. The second function of the WRADD bus is to select the device of queues to be loaded on to the $\overline{PAF}$n bus during strobed flag mode. The most significant 3 bits, WRADD[4:2] are again used to select 1 of 8 possible multi-queue devices that may be connected in expansion mode. Address bits WRADD[1:0] are don't care during device selection. The device address present on the WRADD bus will be selected on the rising edge of WCLK provided that FSTR is HIGH, (note, that data can be written into the previously selected queue on this WCLK edge). Please refer to Table 1 for details on the WRADD bus.
VCC (See Pin table for details)	+3.3V Supply	Power	These are VCC power supply pins and must all be connected to a +3.3V supply rail.
GND (See Pin table for details)	Ground Pin	Ground	These are Ground pins and must all be connected to the GND supply rail.

NOTES:
1. Inputs should not change after Master Reset.
2. These pins are for the JTAG port. Please refer to pages 45-49 and Figures 27-29.

***Please continue to next page for Pin Number Table.*

© 2019 Renesas Electronics Corporation

PIN NUMBER TABLE

Symbol	Name	I/O TYPE	Pin Number
D[17:0] Din	Data Input Bus	LVTTL INPUT	D17-C1, D(16,15)-B(2,1), D(14-12)-A(1-3), D11-B3, D10-A4, D9-B4, D8-C4, D7-A5, D6-B5, D5-C5, D4-A6, D3-B6, D2-C6, D1-A7, D0-B7
Q[17:0] Qout	Data Output Bus	LVTTL OUTPUT	Q17-C15, Q16-D14, Q(15,14)-A(16,15), Q13-B15, Q12-A14, Q11-B14, Q10-C14, Q9-A13, Q8-B13, Q7-C13, Q6-A12, Q5-B12, Q4-C12, Q3-A11, Q2-B11, Q(1,0)-C(11,10)
V_{CC}	+3.3V Supply	Power	D(4-13), E(4-7,10-13), F(4,5,12,13), G(4,5,12,13), H(4,13), J(4,13), K(4,5,12,13), L(4,5,12,13), M(4-7,10-13), N(4-13)
GND	Ground Pin	Ground	C(2,3,8), D(1-3), E(1-3,8,9), F(1-3,6-11), G(1-3,6-11), H(1-3,5-12), J(1-3,5-12,14), K(1-3,6-11,14), L(6-11,14), M(8,9), N(15,16), P(1-3)
DNC	Do Not Connect		B16, C16, D(15,16), E(14-16), F(14-16), G(14-16), H(14-16), J(15,16), P(6,7,11,12), R(6,7,9-12), T(12)

© 2019 Renesas Electronics Corporation

ABSOLUTE MAXIMUM RATINGS

Symbol	Rating	Com'l & Ind'l	Unit
V_{TERM}	Terminal Voltage with respect to GND	–0.5 to +4.5	V
T_{STG}	Storage Temperature	–55 to +125	°C
I_{OUT}	DC Output Current	–50 to +50	mA

NOTE:

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.

RECOMMENDED DC OPERATING CONDITIONS

Symbol	Parameter	Min.	Typ.	Max.	Unit
V_{CC}(1)	Supply Voltage (Com'l/Ind'l)	3.15	3.3	3.45	V
GND	Supply Voltage (Com'l/Ind'l)	0	0	0	V
V_{IH}	Input High Voltage (Com'l/Ind'l)	2.0	—	V_{CC}+0.3	V
V_{IL}	Input Low Voltage (Com'l/Ind'l)	—	—	0.8	V
T_A	Operating Temperature Commercial	0	—	+70	°C
T_A	Operating Temperature Industrial	-40	—	+85	°C

NOTE:

1. V_{CC} = 3.3V ± 0.15V, JEDEC JESD8-A compliant.

DC ELECTRICAL CHARACTERISTICS

(Commercial: V_{CC} = 3.3V ± 0.15V, T_A = 0°C to +70°C;Industrial: V_{CC} = 3.3V ± 0.15V, T_A = 40°C to +85°C; JEDEC JESD8-A compliant)

Symbol	Parameter	Min.	Max.	Unit
I_{LI}(1)	Input Leakage Current	–10	10	µA
I_{LO}(2)	Output Leakage Current	–10	10	µA
V_{OH}	Output Logic "1" Voltage, I_{OH} = –8 mA	2.4	—	V
V_{OL}	Output Logic "0" Voltage, I_{OL} = 8 mA	—	0.4	V
I_{CC1}(3,4,5)	Active Power Supply Current	—	100	mA
I_{CC2}(3,6)	Standby Current	—	25	mA

NOTES:

1. Measurements with $0.4 \leq V_{IN} \leq V_{CC}$.
2. $\overline{OE} \geq V_{IH}$, $0.4 \leq V_{OUT} \leq V_{CC}$.
3. Tested with outputs open (I_{OUT} = 0).
4. RCLK and WCLK toggle at 20 MHz and data inputs switch at 10 MHz.
5. Typical I_{CC1} = 16 + 3.14*fs + 0.02*C_L*fs (in mA) with V_{CC} = 3.3V, t_A = 25°C, fs = WCLK frequency = RCLK frequency (in MHz, using TTL levels), data switching at fs/2, C_L = capacitive load (in pF).
6. RCLK and WCLK, toggle at 20 MHz.
   The following inputs should be pulled to GND: WRADD, RDADD, WADEN, RADEN, FSTR, ESTR, SCLK, SI, EXI, FXI and all Data Inputs.
   The following inputs should be pulled to V_{CC}: $\overline{WEN}$, $\overline{REN}$, $\overline{SENI}$, $\overline{PRS}$, $\overline{MRS}$, TDI, TMS and $\overline{TRST}$.
   All other inputs are don't care, and should be pulled HIGH or LOW.

CAPACITANCE (T_A = +25°C, f = 1.0MHz)

Symbol	Parameter(1)	Conditions	Max.	Unit
C_{IN}(2)	Input Capacitance	V_{IN} = 0V	10	pF
C_{OUT}(1,2)	Output Capacitance	V_{OUT} = 0V	10	pF

NOTES:

1. With output deselected, ($\overline{OE} \geq V_{IH}$).
2. Characterized values, not currently tested.

© 2019 Renesas Electronics Corporation

AC TEST LOADS

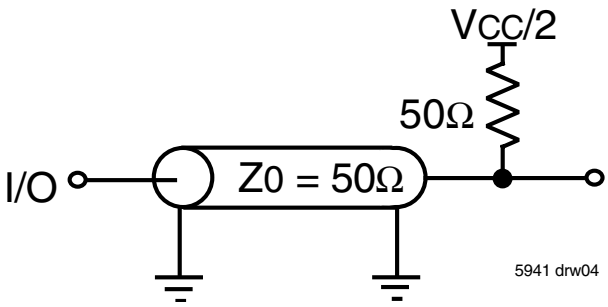


Figure 2a. AC Test Load

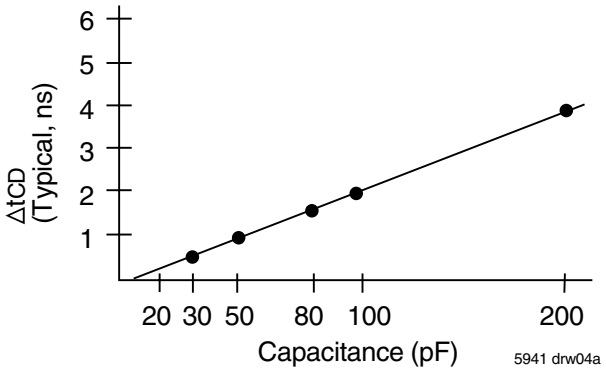
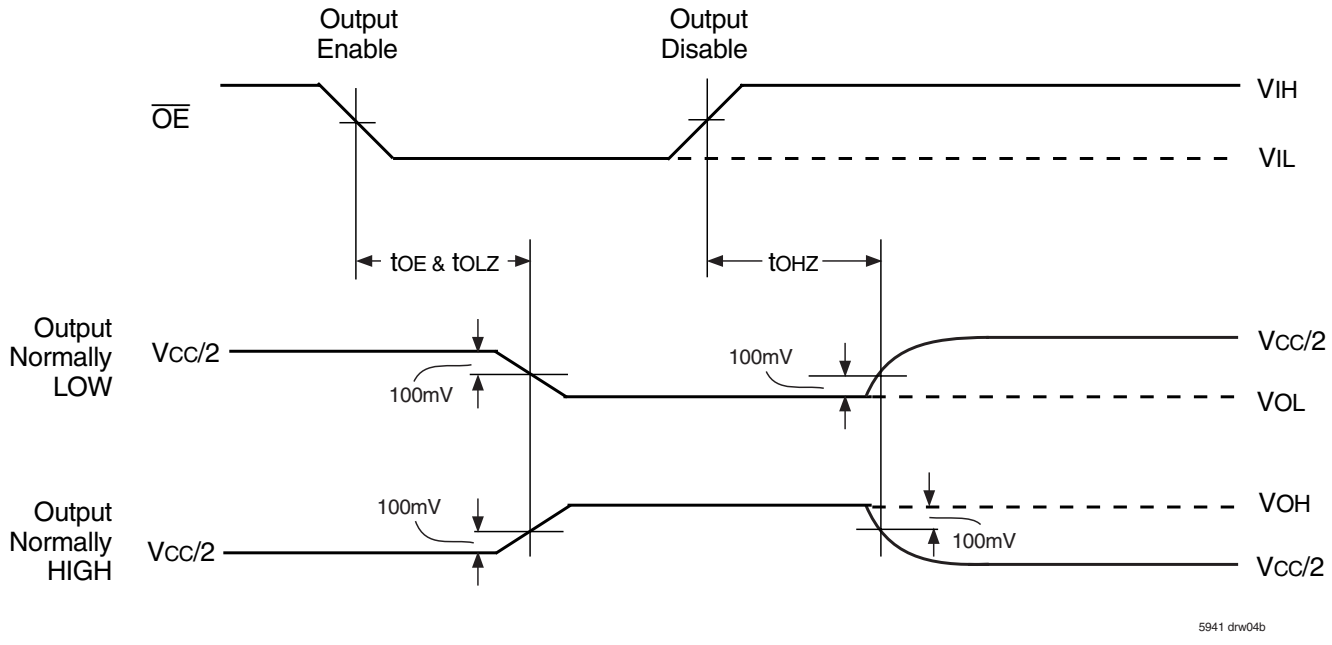


Figure 2b. Lumped Capacitive Load, Typical Derating

AC TEST CONDITIONS

Input Pulse Levels	GND to 3.0V
Input Rise/Fall Times	1.5ns
Input Timing Reference Levels	1.5V
Output Reference Levels	1.5V
Output Load	See Figure 2a & 2b

OUTPUT ENABLE & DISABLE TIMING



© 2019 Renesas Electronics Corporation

AC ELECTRICAL CHARACTERISTICS

(Commercial: V_{CC} = 3.3V ± 0.15V, T_A = 0°C to +70°C;Industrial: V_{CC} = 3.3V ± 0.15V, T_A = 40°C to +85°C; JEDEC JESD8-A compliant)

Symbol	Parameter	Commercial IDT72V51233L6 IDT72V51243L6 IDT72V51553L6		Com'l & Ind'l[1] IDT72V51233L7-5 IDT72V51243L7-5 IDT72V51553L7-5		Unit
		Min.	Max.	Min.	Max.	
f_S	Clock Cycle Frequency (WCLK & RCLK)	—	166	—	133	MHz
t_A	Data Access Time	0.6	3.7	0.6	4	ns
t_{CLK}	Clock Cycle Time	6	—	7.5	—	ns
t_{CLKH}	Clock High Time	2.7	—	3.5	—	ns
t_{CLKL}	Clock Low Time	2.7	—	3.5	—	ns
t_{DS}	Data Setup Time	2	—	2.0	—	ns
t_{DH}	Data Hold Time	0.5	—	0.5	—	ns
t_{ENS}	Enable Setup Time	2	—	2.0	—	ns
t_{ENH}	Enable Hold Time	0.5	—	0.5	—	ns
t_{RS}	Reset Pulse Width	10	—	10	—	ns
t_{RSS}	Reset Setup Time	15	—	15	—	ns
t_{RSR}	Reset Recovery Time	10	—	10	—	ns
t_{PRSS}	Partial Reset Setup	2.0	—	2.5	—	ns
t_{PRSH}	Partial Reset Hold	0.5	—	0.5	—	ns
t_{OLZ} ($\overline{OE}$-Q_n)[2]	Output Enable to Output in Low-Impedance	0.6	3.7	0.6	4	ns
t_{OHZ}[2]	Output Enable to Output in High-Impedance	0.6	3.7	0.6	4	ns
t_{OE}	Output Enable to Data Output Valid	0.6	3.7	0.6	4	ns
f_C	Clock Cycle Frequency (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	1.2	—	1.2	—	ns
t_{SENS}	Serial Enable Setup	20	—	20	—	ns
t_{SENH}	Serial Enable Hold	1.2	—	1.2	—	ns
t_{SDO}	SCLK to Serial Data Out	—	20	—	20	ns
t_{SENO}	SCLK to Serial Enable Out	—	20	—	20	ns
t_{SDOP}	Serial Data Out Propagation Delay	1.5	3.7	1.5	4	ns
t_{SENOP}	Serial Enable Propagation Delay	1.5	3.7	1.5	4	ns
t_{PCWQ}	Programming Complete to Write Queue Selection	20	—	20	—	ns
t_{PCRQ}	Programming Complete to Read Queue Selection	20	—	20	—	ns
t_{AS}	Address Setup	2.5	—	3.0	—	ns
t_{AH}	Address Hold	1	—	1	—	ns
t_{WFF}	Write Clock to Full Flag	—	3.7	—	5	ns
t_{ROV}	Read Clock to Output Valid	—	3.7	—	5	ns
t_{STS}	Strobe Setup	2	—	2	—	ns
t_{STH}	Strobe Hold	0.5	—	0.5	—	ns
t_{QS}	Queue Setup	2	—	2.5	—	ns
t_{QH}	Queue Hold	0.5	—	0.5	—	ns
t_{WAF}	WCLK to $\overline{PAF}$ flag	0.6	3.7	0.6	4	ns
t_{RAE}	RCLK to $\overline{PAE}$ flag	0.6	3.7	0.6	4	ns
t_{PAF}	Write Clock to Synchronous Almost-Full Flag Bus	0.6	3.7	0.6	4	ns
t_{PAE}	Read Clock to Synchronous Almost-Empty Flag Bus	0.6	3.7	0.6	4	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.

© 2019 Renesas Electronics Corporation

AC ELECTRICAL CHARACTERISTICS (CONTINUED)

(Commercial: V_{CC} = 3.3V ± 0.15V, T_A = 0°C to +70°C;Industrial: V_{CC} = 3.3V ± 0.15V, T_A = 40°C to +85°C; JEDEC JESD8-A compliant)

		Commercial		Com'l & Ind'l[1]		
		IDT72V51233L6 IDT72V51243L6 IDT72V51553L6		IDT72V51233L7-5 IDT72V51243L7-5 IDT72V51553L7-5		
Symbol	Parameter	Min.	Max.	Min.	Max.	Unit
t_{PAELZ}[2]	RCLK to $\overline{PAE}$ Flag Bus to Low-Impedance	0.6	3.7	0.6	4	ns
t_{PAEHZ}[2]	RCLK to $\overline{PAE}$ Flag Bus to High-Impedance	0.6	3.7	0.6	4	ns
t_{PAFLZ}[2]	WCLK to $\overline{PAF}$ Flag Bus to Low-Impedance	0.6	3.7	0.6	4	ns
t_{PAFHZ}[2]	WCLK to $\overline{PAF}$ Flag Bus to High-Impedance	0.6	3.7	0.6	4	ns
t_{FFHZ}[2]	WCLK to Full Flag to High-Impedance	0.6	3.7	0.6	4	ns
t_{FFLZ}[2]	WCLK to Full Flag to Low-Impedance	0.6	3.7	0.6	4	ns
t_{OVLZ}[2]	RCLK to Output Valid Flag to Low-Impedance	0.6	3.7	0.6	4	ns
t_{OVHZ}[2]	RCLK to Output Valid Flag to High-Impedance	0.6	3.7	0.6	4	ns
t_{FSYNC}	WCLK to $\overline{PAF}$ Bus Sync to Output	0.6	3.7	0.6	4	ns
t_{FXO}	WCLK to $\overline{PAF}$ Bus Expansion to Output	0.6	3.7	0.6	4	ns
t_{ESYNC}	RCLK to $\overline{PAE}$ Bus Sync to Output	0.6	3.7	0.6	4	ns
t_{EXO}	RCLK to $\overline{PAE}$ Bus Expansion to Output	0.6	3.7	0.6	4	ns
t_{SKEW1}	SKEW time between RCLK and WCLK for $\overline{FF}$ and $\overline{OV}$	4.5	—	5.75	—	ns
t_{SKEW2}	SKEW time between RCLK and WCLK for $\overline{PAF}$ and $\overline{PAE}$	6	—	7.5	—	ns
t_{SKEW3}	SKEW time between RCLK and WCLK for $\overline{PAF}$[0:7] and $\overline{PAE}$[0:7]	6	—	7.5	—	ns
t_{SKEW4}	SKEW time between RCLK and WCLK for $\overline{OV}$	6	—	7.5	—	ns
t_{XIS}	Expansion Input Setup	1.0	—	1.3	—	ns
t_{XIH}	Expansion Input Hold	0.5	—	0.5	—	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.

© 2019 Renesas Electronics Corporation

FUNCTIONAL DESCRIPTION

MASTER RESET

A Master Reset is performed by toggling the $\overline{MRS}$ 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 using the default settings. 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
IW, OW – Bus Matching options
MAST – Master Device
ID0, 1, 2 – Device ID
DFM – Programming mode, serial or default
DF – Offset value for $\overline{PAE}$ and $\overline{PAF}$

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 4, *Master Reset* for relevant timing.

PARTIAL RESET

A Partial Reset is a means by which the user can reset both the read and write pointers of a single queue that has been setup within a multi-queue device. Before a partial reset can take place on a queue, the respective queue must be selected on both the read port and write port a minimum of 2 RCLK and 2 WCLK cycles before the $\overline{PRS}$ goes LOW. The partial reset is then performed by toggling the $\overline{PRS}$ input from HIGH to LOW to HIGH, maintaining the LOW state for at least one WCLK and one RCLK cycle. Once a partial reset has taken place a minimum of 3 WCLK and 3 RCLK cycles must occur before enabled writes or reads can occur.

A Partial Reset only resets the read and write pointers of a given queue, a partial reset will not effect the overall configuration and setup of the multi-queue device and its queues.

See Figure 5, *Partial Reset* for relevant timing.

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 IDT72V51233/72V51243/72V51253 devices are capable of up to 4 queues and therefore contain 4 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 IDT72V51233/72V51243/72V51253 devices the serial programming requires a total number of serially loaded bits per device, (SCLK cycles with $\overline{SENI}$ enabled), calculated by: 19+(Qx72) where Q is the number of queues the user wishes to setup within the device. Please refer to the separate Application Note, AN-303 for recommended control of the serial programming port.

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 mode 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 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 mode, the IDT72V51233/72V51243/72V51253 devices require a total number of serially loaded bits per device to complete serial programming, (SCLK cycles with $\overline{SENI}$ enabled), calculated by: n[19+(Qx72)] 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 6, *Serial Port Connection* and Figure 7, *Serial Programming* for connection and timing information.

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 by which to setup the multi-queue flow-control device, rather than using the serial programming method. The default mode will configure a multi-queue device such that the maximum number of queues possible are setup, with all of the parts available memory blocks being allocated equally between the queues. The values of the $\overline{PAE}$/$\overline{PAF}$ offsets is determined by the state of the DF (default) pin during a master reset.

For the IDT72V51233/72V51243/72V51253 devices the default mode will setup 4 queues, each queue configured as follows: For the IDT72V51233 with x9 input and x9 output ports, 16,384 x 9. If one or both ports is x18, 8,192 x 18. For the IDT72V51243 with x9 input and x9 output ports, 32,768 x 9. If one or both ports is x18, 16,384 x 18. For the IDT72V51253 with x9 input and x9 output ports, 65,536 x 9. If one or both ports is x18, 32,768 x 18. For both devices the value of the $\overline{PAE}$/$\overline{PAF}$ offsets is determined at master reset by the state of the DF input. If DF is LOW then both the $\overline{PAE}$ & $\overline{PAF}$ offset will be 8, if HIGH then the value is 128.

© 2019 Renesas Electronics Corporation

When configuring the IDT72V51233/72V51243/72V51253 devices in default mode the user simply has to apply WCLK cycles after a master reset, until $\overline{SENO}$ 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 $\overline{SENO}$ output of a device going from HIGH to LOW. Note, that $\overline{SENI}$ must be held LOW when a device is setup for default programming mode.

When multi-queue devices are connected in expansion mode, the $\overline{SENI}$ of the first device in a chain can be held LOW. The $\overline{SENO}$ of a device should connect to the $\overline{SENI}$ of the next device in the chain. The $\overline{SENO}$ of the final device is used to indicate that default programming of all devices is complete. When the final $\overline{SENO}$ 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 8, *Default Programming*.

WRITE QUEUE SELECTION & WRITE OPERATION

The IDT72V51233/72V51243/72V51253 multi-queue flow-control devices have up to 4 queues that data can be written into via a common write port using the data inputs, Din, write clock, WCLK and write enable, $\overline{WEN}$. The queue address present on the write address bus, WRADD during a rising edge on WCLK while write address enable, WADEN is HIGH, is the queue selected for write operations. The state of $\overline{WEN}$ is don't care during the write queue selection cycle. The queue selection only has to be made on a single WCLK cycle, this will remain the selected queue until another queue is selected, the selected queue is always the last queue selected.

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. When a new queue is selected for write operations the address for that queue must be present on the WRADD bus during a rising edge of WCLK provided that WADEN is HIGH. A queue to be written to need only be selected on a single rising edge of WCLK. All subsequent writes will be written to that queue until a new queue is selected. A minimum of 2 WCLK cycles must occur between queue selections on the write port. On the next WCLK rising edge the write port discrete full flag will update to show the full status of the newly selected queue. On the second rising edge of WCLK, data present on the data input bus, Din can be written into the newly selected queue provided that $\overline{WEN}$ is LOW and the new queue is not full. The cycle of the queue selection and the next cycle will continue to write data present on the data input bus, Din into the previous queue provided that $\overline{WEN}$ is active LOW.

If $\overline{WEN}$ is HIGH, inactive for these 2 clock cycles, then data will not be written in to the previous queue.

If the newly selected queue is full at the point of its selection, then writes to that queue will be prevented, a full queue cannot be written into.

In the 4 queue multi-queue device the WRADD address bus is 5 bits wide. The least significant 2 bits are used to address one of the 4 available queues within a single multi-queue device. The most significant 3 bits are used when a device is connected in expansion mode, up to 8 devices can be connected in expansion, each device having its own 3 bit address. The selected device is the one for which the address matches a 3 bit ID code, which is statically setup on the ID pins, ID0, ID1, and ID2 of each individual device.

Note, the WRADD bus is also used in conjunction with FSTR (almost full flag bus strobe), to address the almost full flag bus of a respective device during direct mode of operation.

Refer to Table 1, for Write Address bus arrangement. Also, refer to Figure 9, *Write Queue Select, Write Operation and Full flag Operation* and Figure 11, *Full Flag Timing Expansion Mode* for timing diagrams.

TABLE 1 — WRITE ADDRESS BUS, WRADD[4:0]

Operation	WCLK	WADEN	FSTR	WRADD[4:0]	
Write Queue Select	↑	1	0	4 3 2 Device Select (Compared to ID0,1,2)	1 0 Write Queue Address (2 bits = 4 Queues)
$\overline{PAF}$n Flag Bus Device Select	↑	0	1	4 3 2 Device Select (Compared to ID0,1,2)	1 0 X X

5941 drw05

© 2019 Renesas Electronics Corporation

READ QUEUE SELECTION & READ OPERATION

The multi-queue flow-control device has up to 4 queues that data is read from via a common read port using the data outputs, Qout, read clock, RCLK and read enable, $\overline{REN}$. An output enable, $\overline{OE}$ control pin is also provided to allow High-Impedance selection of the Qout data outputs. The multi-queue device read port operates in a mode similar to "First Word Fall Through" on a traditional IDT FIFO, but with the added feature of data output pipelining. This data pipelining on the output port allows the user to achieve 100% bus utilization, which is the ability to read out a data word on every rising edge of RCLK regardless of whether a new queue is being selected for read operations.

The queue address present on the read address bus, RDADD during a rising edge on RCLK while read address enable, RADEN is HIGH, is the queue selected for read operations. A queue to be read from need only be selected on a single rising edge of RCLK. All subsequent reads will be read from that queue until a new queue is selected. A minimum of 2 RCLK cycles must occur between queue selections on the read port. Data from the newly selected queue will be present on the Qout outputs after 2 RCLK cycles plus an access time, provided that $\overline{OE}$ is active, LOW. On the same RCLK rising edge that the new queue is selected, data can still be read from the previously selected queue, provided that $\overline{REN}$ is LOW, active and the previous queue is not empty on the following rising edge of RCLK a word will be read from the previously selected queue regardless of $\overline{REN}$ due to the fall through operation, (provided the queue is not empty). Remember that $\overline{OE}$ allows the user to place the Qout, data output bus into High-Impedance and the data can be read onto the output register regardless of $\overline{OE}$.

When a queue is selected on the read port, the next word available in that queue (provided that the queue is not empty), will fall through to the output register after 2 RCLK cycles. As mentioned, in the previous 2 RCLK cycles to the new data being available, data can still be read from the previous queue, provided that the queue is not empty. At the point of queue selection, the 2-stage internal data pipeline is loaded with the last word from the previous queue and the next word from the new queue, both these words will fall through to the output register consecutively upon selection of the new queue. This pipelining effect provides the user with 100% bus utilization, but brings about the possibility that a "NULL" queue may be required within a multi-queue device. Null queue operation is discussed in the next section on.

If an empty queue is selected for read operations on the rising edge of RCLK, on the same RCLK edge and the following RCLK edge, 2 final reads will be made from the previous queue, provided that $\overline{REN}$ is active, LOW. On the next RCLK rising edge a read from the new queue will not occur, because the queue is empty. The last word in the data output register (from the previous queue), will remain there, but the output valid flag, $\overline{OV}$ will go HIGH, to indicate that the data present is no longer valid.

The RDADD bus is also used in conjunction with ESTR (almost empty flag bus strobe), to address the almost empty flag bus of a respective device during direct mode of operation. In the 4 queue multi-queue device the RDADD address bus is 6 bits wide. The least significant 2 bits are used to address one of the 4 available queues within a single multi-queue device. The 3rd least significant bit is used to select a "Null" Queue. During a Null-Q selection the 2 LSB's are don't care. The Null-Q is seen as an empty queue on the read port. Null-Q operation is discussed in more detail in a separate section. The most significant 3 bits are used when a device is connected in expansion mode, up to 8 devices can be connected in expansion, each device having its own 3 bit address. The selected device is the one for which the address matches a 3 bit ID code, which is statically setup on the ID pins, ID0, ID1, and ID2 of each individual device.

Refer to Table 2, for Read Address bus arrangement. Also, refer to Figures 12,14 & 15 for read queue selection and read port operation timing diagrams.

TABLE 2 — READ ADDRESS BUS, RDADD[5:0]

Operation	RCLK	RADEN	ESTR	RDADD[5:0]		
Read Queue Select	↑	1	0	5 4 3 Device Select (Compared to ID0,1,2)	2 Null-Q Select Pin	1 0 Read Queue Address (2 bits = 4 Queues)
Flag Bus Device Selection	↑	0	1	5 4 3 Device Select (Compared to ID0,1,2)	2 1 0 X X X	

5941 drw06

© 2019 Renesas Electronics Corporation

NULL QUEUE OPERATION (OF THE READ PORT)

Pipelining of data to the output port enables the device to provide 100% bus utilization in standard mode. Data can be read out of the multi-queue flow-control device on every RCLK cycle regardless of queue switches or other operations. The device architecture is such that the pipeline is constantly filled with the next words in a selected queue to be read out, again providing 100% bus utilization. This type of architecture does assume that the user is constantly switching queues such that during a queue switch, the last data word required from the previous queue will fall through the pipeline to the output.

Note, that if reads cease at the empty boundary of a queue, then the last word will automatically flow through the pipeline to the output.

The Null-Q is selected via read port address space RDADD[2]. The RDADD[5:0] bus should be addressed with xxx1xx, this address is the Null-Q. A null queue can be selected when no further reads are required from a previously selected queue. Changing to a null queue will continue to propagate data in the pipeline to the previous queue's output. The Null-Q can remain selected until a data becomes available in another queue for reading. The Null-Q can be utilized in either standard or packet mode.

Note: If the user switches the read port to the null queue, this queue is seen as and treated as an empty queue, therefore after switching to the null queue the last word from the previous queue will remain in the output register and the $\overline{OV}$ flag will go HIGH, indicating data is not valid.

The Null queue operation only has significance to the read port of the multi-queue, it is a means to force data through the pipeline to the output. Null-Q selection and operation has no meaning on the write port of the device. Also, refer to Figure 16, *Read Operation and Null Queue Select* for diagram.

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 two setup pins, IW (Input Width) and OW (Output Width) determine the input and output port bus widths as per the selections shown in Table 3, "Bus Matching Set-up". 9 bit bytes or 18 bit words can be written into and read form 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 3, *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 x18 and the output port is x9, then two data reads from a full queue will be required to cause the full flag to go HIGH (queue not full). Conversely, the Output Valid 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 x9 and the output port is x18, two write operations will be required to cause the output valid flag of an empty queue to go LOW, output valid (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 3 — BUS-MATCHING SET-UP

IW	OW	Write Port	Read Port
0	0	x18	x18
0	1	x18	x9
1	0	x9	x18
1	1	x9	x9

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 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 next rising edge of WCLK, the $\overline{FF}$ flag output will show the full status of the newly selected queue. On the second 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 one 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 9, *Write Queue Select, Write Operation and Full Flag Operation* and Figure 11, *Full Flag Timing in Expansion Mode* for timing information.

EXPANSION MODE - FULL FLAG OPERATION

When multi-queue devices are connected in Expansion mode 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 mode 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 3 bit ID code found in the 3 most significant bits of the write queue address bus, WRADD. If the 3 most significant bits of WRADD match the 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 11, *Full Flag Timing in Expansion Mode* for details of flag operation, including when more than one device is connected in expansion.

© 2019 Renesas Electronics Corporation

OUTPUT VALID FLAG OPERATION

The multi-queue flow-control device provides a single Output Valid flag output, $\overline{OV}$. The $\overline{OV}$ provides an empty status or data output valid status for the data word currently available on the output register of the read port. The rising edge of an RCLK cycle that places new data onto the output register of the read port, also updates the $\overline{OV}$ flag to show whether or not that new data word is actually valid. Internally the multi-queue flow-control device monitors and maintains a status of the empty condition of all queues within it, however only the queue that is selected for read operations has its output valid (empty) status output to the $\overline{OV}$ flag, giving a valid status for the word being read at that time.

The nature of the first word fall through operation means that when the last data word is read from a selected queue, the $\overline{OV}$ flag will go HIGH on the next enabled read, that is, on the next rising edge of RCLK while $\overline{REN}$ is LOW.

When queue switches are being made on the read port, the $\overline{OV}$ flag will switch to show status of the new queue in line with the data output from the new queue. When a queue selection is made the first data from that queue will appear on the Qout data outputs 2 RCLK cycles later, the $\overline{OV}$ will change state to indicate validity of the data from the newly selected queue on this 2nd RCLK cycle also. The previous cycles will continue to output data from the previous queue and the $\overline{OV}$ flag will indicate the status of those outputs. Again, the $\overline{OV}$ flag always indicates status for the data currently present on the output register.

The $\overline{OV}$ flag is synchronous to the RCLK and all transitions of the $\overline{OV}$ flag occur based on a rising edge of RCLK. Internally the multi-queue device monitors and keeps a record of the output valid (empty) status for all queues. It is possible that the status of an $\overline{OV}$ flag may be changing internally even though that respective 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 $\overline{OV}$ flag status based on write operations, that is, data may be written into that queue causing it to become "not empty".

See Figure 12, *Read Queue Select, Read Operation* and Figure 13, *Output Valid Flag Timing* for details of the timing.

EXPANSION MODE – OUTPUT VALID FLAG OPERATION

When multi-queue devices are connected in Expansion mode, the $\overline{OV}$ 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 $\overline{OV}$ flag (as opposed to a discrete $\overline{OV}$ flag for each device). This $\overline{OV}$ flag is only pertinent to the queue being selected for read operations at that time. Remember, that when in expansion mode only one multi-queue device can be read from at any moment in time, thus the $\overline{OV}$ flag provides status of the active queue on the read port.

This connection of flag outputs to create a single flag requires that the $\overline{OV}$ flag output have a High-Impedance capability, such that when a queue selection is made only a single device drives the $\overline{OV}$ flag bus and all other $\overline{OV}$ flag outputs connected to the $\overline{OV}$ 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 $\overline{OV}$ 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 $\overline{OV}$ flag output of that device will maintain control of the $\overline{OV}$ flag bus. Its $\overline{OV}$ flag will simply update between queue switches to show the respective queue output valid status.

The multi-queue device places its $\overline{OV}$ flag output into High-Impedance based on the 3 bit ID code found in the 3 most significant bits of the read queue address bus, RDADD. If the 3 most significant bits of RDADD match the 3 bit ID code setup on the static inputs, ID0, ID1 and ID2 then the $\overline{OV}$ flag output of the respective device will be in a Low-Impedance state. If they do not match, then the $\overline{OV}$ flag output of the respective device will be in a High-Impedance state. See Figure 13, *Output Valid Flag Timing* for details of flag operation, including when more than one device is connected in expansion.

ALMOST FULL FLAG

As previously mentioned the multi-queue flow-control device provides a single Programmable Almost Full flag output, $\overline{PAF}$. The $\overline{PAF}$ 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 $\overline{PAF}$ flag. This dedicated flag is often referred to as the "active queue almost full flag". The position of the $\overline{PAF}$ 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 $\overline{PAF}$ flag. The $\overline{PAF}$ 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 $\overline{PAF}$ 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 $\overline{PAF}$ value of different queues within the same device can be different values.

When queue switches are being made on the write port, the $\overline{PAF}$ flag output will switch to the new queue and provide the user with the new queue status, on the second 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 second rising edge of WCLK following a queue selection, the $\overline{PAF}$ flag output will show the full status of the newly selected queue. The $\overline{PAF}$ 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 $\overline{PAF}$ will go LOW 2 WCLK cycles after the write. The same is true when a read occurs, there will be a 2 WCLK cycle delay after the read operation.

So the $\overline{PAF}$ flag delays are:

from a write operation to $\overline{PAF}$ flag LOW is 2 WCLK + t_{WAF}

The delay from a read operation to $\overline{PAF}$ flag HIGH is t_{SKEW2} + WCLK + t_{WAF}

Note, if t_{SKEW} is violated there will be one added WCLK cycle delay.

The $\overline{PAF}$ flag is synchronous to the WCLK and all transitions of the $\overline{PAF}$ 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 $\overline{PAF}$ 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 $\overline{PAF}$ flag on the $\overline{PAF}$[3:0] flag bus, this will be discussed in detail in a later section of the data sheet.

See Figures 18 and 19 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, $\overline{PAE}$. The $\overline{PAE}$ 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 $\overline{PAE}$ flag. This dedicated flag is often referred to as the "active queue almost empty flag". The position of the $\overline{PAE}$ flag boundary

© 2019 Renesas Electronics Corporation

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 $\overline{PAE}$ flag. The $\overline{PAE}$ 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 $\overline{PAE}$ 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 $\overline{PAE}$ value of different queues within the same device can be different values.

When queue switches are being made on the read port, the $\overline{PAE}$ flag output will switch to the new queue and provide the user with the new queue status, on the second 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 second rising edge of RCLK following a queue selection, the data word from the new queue will be available at the output register and the $\overline{PAE}$ flag output will show the empty status of the newly selected queue. The $\overline{PAE}$ 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 $\overline{PAE}$ will go LOW 2 RCLK cycles after the read. The same is true when a write occurs, there will be a 2 RCLK cycle delay after the write operation.

So the $\overline{PAE}$ flag delays are:

from a read operation to $\overline{PAE}$ flag LOW is 2 RCLK + t_{RAE}

The delay from a write operation to $\overline{PAE}$ flag HIGH is t_{SKEW2} + RCLK + t_{RAE}

Note, if t_{SKEW} is violated there will be one added RCLK cycle delay.

The $\overline{PAE}$ flag is synchronous to the RCLK and all transitions of the $\overline{PAE}$ 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 $\overline{PAE}$ 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 $\overline{PAE}$ flag on the $\overline{PAE}$[3:0] flag bus, this will be discussed in detail in a later section of the data sheet.

See Figures 20 and 21 for Almost Empty flag timing and queue switching.

© 2019 Renesas Electronics Corporation

TABLE 4 — FLAG OPERATION BOUNDARIES & TIMING

Output Valid, $\overline{OV}$ Flag Boundary	
I/O Set-Up	**$\overline{OV}$ Boundary Condition**
In18 to out18 or In9 to out9 (Both ports selected for same queue when the 1st Word is written in)	$\overline{OV}$ Goes LOW after 1st Write (see note below for timing)
In18 to out9) (Both ports selected for same queue when the 1st Word is written in)	$\overline{OV}$ Goes LOW after 1st Write (see note below for timing)
In9 to out18 (Both ports selected for same queue when the 1st Word is written in)	$\overline{OV}$ Goes LOW after 2nd Write (see note below for timing)

NOTE:

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

Assertion:

Write to $\overline{OV}$ LOW: t_{SKEW1} + RCLK + t_{ROV}

If t_{SKEW1} is violated there may be 1 added clock: t_{SKEW1} + 2 RCLK + t_{ROV}

De-assertion:

Read Operation to $\overline{OV}$ HIGH: t_{ROV}

Full Flag, $\overline{FF}$ Boundary	
I/O Set-Up	**$\overline{FF}$ Boundary Condition**
In18 to out18 or In9 to out9 (Both ports selected for same queue when the 1st Word is written in)	$\overline{FF}$ Goes LOW after D+1 Writes (see note below for timing)
In18 to out18 or In9 to out9 (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)
In18 to out9 (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)
In18 to out9 (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)
In9 to out18 (Both ports selected for same queue when the 1st Word is written in)	$\overline{FF}$ Goes LOW after ([D+1] x 2) Writes (see note below for timing)
In9 to out18 (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: t_{WFF}

De-assertion:

Read to $\overline{FF}$ HIGH: t_{SKEW1} + t_{WFF}

If t_{SKEW1} is violated there may be 1 added clock: t_{SKEW1}+WCLK +t_{WFF}

© 2019 Renesas Electronics Corporation

TABLE 4 — FLAG OPERATION BOUNDARIES & TIMING (CONTINUED)

Programmable Almost Empty Flag, $\overline{PAE}$ Boundary	
I/O Set-Up	**$\overline{PAE}$ Assertion**
In18 to out18 or In9 to out9 (Both ports selected for same queue when the 1st Word is written in until the boundary is reached)	$\overline{PAE}$ Goes HIGH after n+2 Writes (see note below for timing)
In18 to out9 (Both ports selected for same queue when the 1st Word is written in until the boundary is reached)	$\overline{PAE}$ Goes HIGH after n+1 Writes (see note below for timing)
In9 to out18 (Both ports selected for same queue when the 1st Word is written in until the boundary is reached)	$\overline{PAE}$ Goes HIGH after ([n+2] x 2) Writes (see note below for timing)

NOTE:
n = Almost Empty Offset value.
Default values: if DF is LOW at Master Reset then n = 8
if DF is HIGH at Master Reset then n = 128

$\overline{PAE}$ Timing
Assertion: Read Operation to $\overline{PAE}$ LOW: 2 RCLK + t_{RAE}
De-assertion: Write to $\overline{PAE}$ HIGH: t_{SKEW2} + RCLK + t_{RAE}
If t_{SKEW2} is violated there may be 1 added clock: t_{SKEW2} + 2 RCLK + t_{RAE}

Programmable Almost Empty Flag Bus, $\overline{PAE}$n Boundary	
I/O Set-Up	**$\overline{PAE}$n Boundary Condition**
In18 to out18 or In9 to out9 (Both ports selected for same queue when the 1st Word is written in until the boundary is reached)	$\overline{PAE}$n Goes HIGH after n+2 Writes (see note below for timing)
In18 to out18 or In9 to out9 (Write port only selected for same queue when the 1st Word is written in until the boundary is reached)	$\overline{PAE}$n Goes HIGH after n+1 Writes (see note below for timing)
In18 to out9	$\overline{PAE}$n Goes HIGH after n+1 Writes (see below for timing)
In9 to out18 (Both ports selected for same queue when the 1st Word is written in until the boundary is reached)	$\overline{PAE}$n Goes HIGH after ([n+2] x 2) Writes (see note below for timing)
In9 to out18 (Write port only selected for same queue when the 1st Word is written in until the boundary is reached)	$\overline{PAE}$n Goes HIGH after ([n+1] x 2) Writes (see note below for timing)

NOTE:
n = Almost Empty Offset value.
Default values: if DF is LOW at Master Reset then n = 8
if DF is HIGH at Master Reset then n = 128

$\overline{PAE}$n Timing
Assertion: Read Operation to $\overline{PAE}$n LOW: 2 RCLK* + t_{PAE}
De-assertion: Write to $\overline{PAE}$n HIGH: t_{SKEW3} + RCLK* + t_{PAE}
If t_{SKEW3} is violated there may be 1 added clock: t_{SKEW3} + 2 RCLK* + t_{PAE}

* 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.

Programmable Almost Full Flag, $\overline{PAF}$ & $\overline{PAF}$n Bus Boundary	
I/O Set-Up	**$\overline{PAF}$ & $\overline{PAF}$n Boundary**
In18 to out18 or In9 to out9 (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+1-m Writes (see note below for timing)
In18 to out18 or In9 to out9 (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)
In18 to out9	$\overline{PAF}$/$\overline{PAF}$n Goes LOW after D-m Writes (see below for timing)
In9 to out18	$\overline{PAF}$/$\overline{PAF}$n Goes LOW after ([D+1-m] x 2) Writes (see note below for timing)

NOTE:
D = Queue Depth
m = Almost Full Offset value.
Default values: if DF is LOW at Master Reset then m = 8
if DF is HIGH at Master Reset then m= 128

$\overline{PAF}$ Timing
Assertion: Write Operation to $\overline{PAF}$ LOW: 2 WCLK + t_{WAF}
De-assertion: Read to $\overline{PAF}$ HIGH: t_{SKEW2} + WCLK + t_{WAF}
If t_{SKEW2} is violated there may be 1 added clock: t_{SKEW2} + 2 WCLK + t_{WAF}

$\overline{PAF}$n Timing
Assertion: Write Operation to $\overline{PAF}$n LOW: 2 WCLK* + t_{PAF}
De-assertion: Read to $\overline{PAF}$n HIGH: t_{SKEW3} + WCLK* + t_{PAF}
If t_{SKEW3} is violated there may be 1 added clock: t_{SKEW3} + 2 WCLK* + t_{PAF}

* 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.

© 2019 Renesas Electronics Corporation

PAFn FLAG BUS OPERATION

The IDT72V51233/72V51243/72V51253 multi-queue flow-control devices can be configured for up to 4 queues, each queue having its own almost full status. An active queue has its flag status output to the discrete flags, FF and PAF, on the write port. Queues that are not selected for a write operation can have their PAF status monitored via the PAFn bus. The PAFn flag bus is 4 bits wide, so that all 4 queues can have their status output to the bus. When a single multi-queue device is used anywhere from 1 to 4 queues may be set-up within the part, each queue having its own dedicated PAF flag output on the PAFn bus. Queues 1 through 4 have their PAF status to PAF[0] through PAF[3] respectively. If less than 4 queues are used then only the associated PAFn outputs will be required, unused PAFn outputs will be don't care outputs. When devices are connected in expansion mode the PAFn flag bus can also be expanded beyond 4 bits to produce a wider PAFn bus that encompasses all queues.

Alternatively, the 4 bit PAFn flag bus of each device can be connected together to form a single 4 bit bus, i.e. PAF[0] of device 1 will connect to PAF[0] of device 2 etc. When connecting devices in this manner the PAFn can only be driven by a single device at any time, (the PAFn outputs of all other devices must be in high impedance state). There are two methods by which the user can select which device has control of the bus, these are "Direct" (Addressed) mode or "Polled" (Looped) mode, determined by the state of the FM (flag Mode) input during a Master Reset.

PAFn BUS EXPANSION - DIRECT MODE

If FM is LOW at Master Reset then the PAFn bus operates in Direct (addressed) mode. In direct mode the user can address the device they require to control the PAFn bus. The address present on the 3 most significant bits of the WRADD[4:0] address bus with FSTR (PAF flag strobe), HIGH will be selected as the device on a rising edge of WCLK. So to address the first device in a bank of devices the WRADD[4:0] address should be "000xx" the second device "001xx" and so on. The 3 most significant bits of the WRADD[4:0] address bus correspond to the device ID inputs ID[2:0]. The PAFn bus will change status to show the new device selected 1 WCLK cycle after device selection. Note, that if a read or write operation is occurring to a specific queue, say queue 'x' on the same cycle as a PAFn bus switch to the device containing 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.

Devices can be selected on consecutive WCLK cycles, that is the device controlling 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 WLCK rising edge that a device is being selected on the PAFn bus, the only restriction being that a write queue selection and PAFn bus selection cannot be made on the same cycle.

PAFn BUS EXPANSION– POLLED MODE

If FM is HIGH at Master Reset then the PAFn bus operates in Polled (Looped) mode. In polled mode the PAFn bus automatically cycles through the devices connected in expansion. In expansion mode one device will be set as the Master, 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 place the PAF status of its queues onto the bus on the first rising edge of WCLK after the MRS input goes HIGH once a Master Reset is complete. The FSYNC (PAF sync pulse) output of the first device (master device), will be HIGH for one cycle of WCLK indicating that it is has control of the PAFn bus for that cycle.

The device also passes a "token" onto the next device in the chain, the next device assuming control of the PAFn bus on the next WCLK cycle. This token passing is done via the FXO outputs and FXI inputs of the devices ("PAFn Expansion Out" and "PAFn Expansion In"). The FXO output of the first device connecting to the FXI input of the second device in the chain, the FXO of the second device connects to the FXI of the third device and so on. The FXO of the final device in a chain connects to the FXI of the first device, so that once the PAFn bus has cycled through all devices control is again passed to the first device. The FXO output of a device will be HIGH for the WCLK cycle it has control of the bus.

Please refer to Figure 24, *PAFn Bus – Polled Mode* for timing information.

PAEn FLAG BUS OPERATION

The IDT72V51233/72V51243/72V51253 multi-queue flow-control devices can be configured for up to 4 queues, each queue having its own almost empty status. An active queue has its flag status output to the discrete flags, OV and 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 4 bits wide, so that all 4 queues can have their status output to the bus.

When a single multi-queue device is used anywhere from 1 to 4 queues may be set-up within the part, each queue having its own dedicated PAEn flag output on the PAEn bus. Queues 1 through 4 have their PAE status to PAE[0] through PAE[3] respectively. If less than 4 queues are used then only the associated PAEn outputs will be required, unused PAEn outputs will be don't care outputs. When devices are connected in expansion mode the PAEn flag bus can also be expanded beyond 4 bits to produce a wider PAEn bus that encompasses all queues.

Alternatively, the 4 bit PAEn flag bus of each device can be connected together to form a single 4 bit bus, i.e. PAE[0] of device 1 will connect to PAE[0] of device 2 etc. When connecting devices in this manner the PAEn bus can only be driven by a single device at any time, (the PAEn outputs of all other devices must be in high impedance state). There are two methods by which the user can select which device has control of the bus, these are "Direct" (Addressed) mode or "Polled" (Looped) mode, determined by the state of the FM (flag Mode) input during a Master Reset.

PAEn BUS EXPANSION- DIRECT MODE

If FM is LOW at Master Reset then the PAEn bus operates in Direct (addressed) mode. In direct mode the user can address the device they require to control the PAEn bus. The address present on the 3 most significant bits of the RDADD[5:0] address bus with ESTR (PAE flag strobe), HIGH will be selected as the device on a rising edge of RCLK. So to address the first device in a bank of devices the RDADD[5:0] address should be "000xx" the second device "001xx" and so on. The 3 most significant bits of the RDADD[5:0] address bus correspond to the device ID inputs ID[2:0]. The PAEn bus will change status to show the new device selected 1 RCLK cycle after device selection. Note, that if a read or write operation is occurring to a specific queue, say queue 'x' on the same cycle as a PAEn bus switch to the device containing 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. However, the "active" PAE flag will show correct status at all times.

Devices can be selected on consecutive RCLK cycles, that is the device controlling 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 device is being selected on the PAEn bus, the only restriction being that a read queue selection and PAEn bus selection cannot be made on the same cycle.

PAEn BUS EXPANSION- POLLED MODE

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

© 2019 Renesas Electronics Corporation

the devices connected in expansion. In expansion mode one device will be set as the Master, MAST input tied HIGH, all other devices will have MAST tied LOW. The master device is the first device to take control of the $\overline{PAE}$n bus and place the $\overline{PAE}$ status of its queues onto the bus on the first rising edge of RCLK after the $\overline{MRS}$ input goes HIGH once a Master Reset is complete. The ESYNC ($\overline{PAE}$ sync pulse) output of the first device (master device), will be HIGH for one cycle of RCLK indicating that it is has control of the $\overline{PAE}$n bus for that cycle.

The device also passes a "token" onto the next device in the chain, the next device assuming control of the $\overline{PAE}$n bus on the next RCLK cycle. This token passing is done via the EXO outputs and EXI inputs of the devices ("$\overline{PAE}$n Expansion Out" and "$\overline{PAE}$n Expansion In"). The EXO output of the first device connecting to the EXI input of the second device in the chain, the EXO of the second device connects to the EXI of the third device and so on. The EXO of the final device in a chain connects to the EXI of the first device, so that once the $\overline{PAE}$n bus has cycled through all devices control is again passed to the first device. The EXO output of a device will be HIGH for the RCLK cycle it has control of the bus.

Please refer to Figure 25, **$\overline{PAE}$***n Bus – Polled Mode* for timing information.

© 2019 Renesas Electronics Corporation

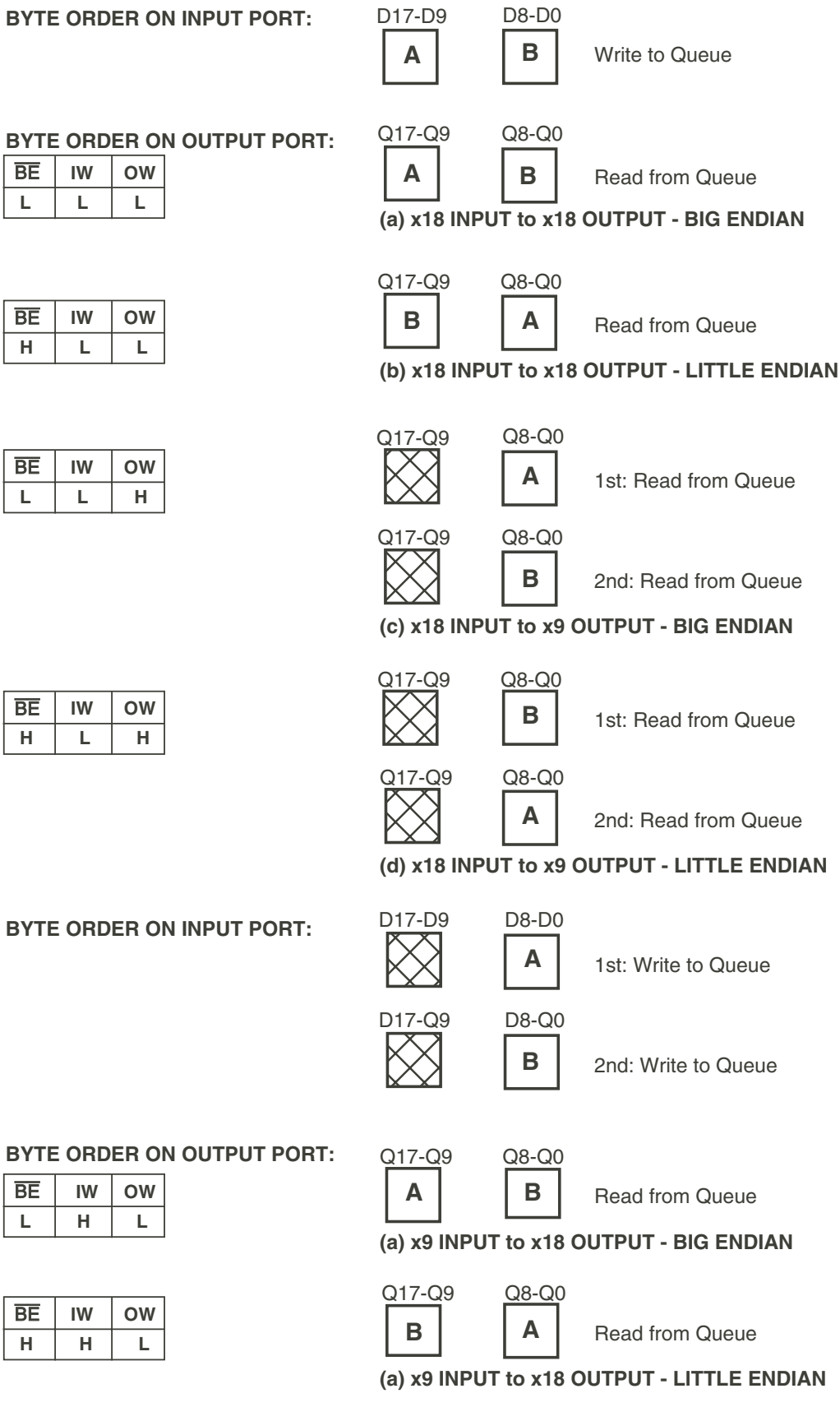
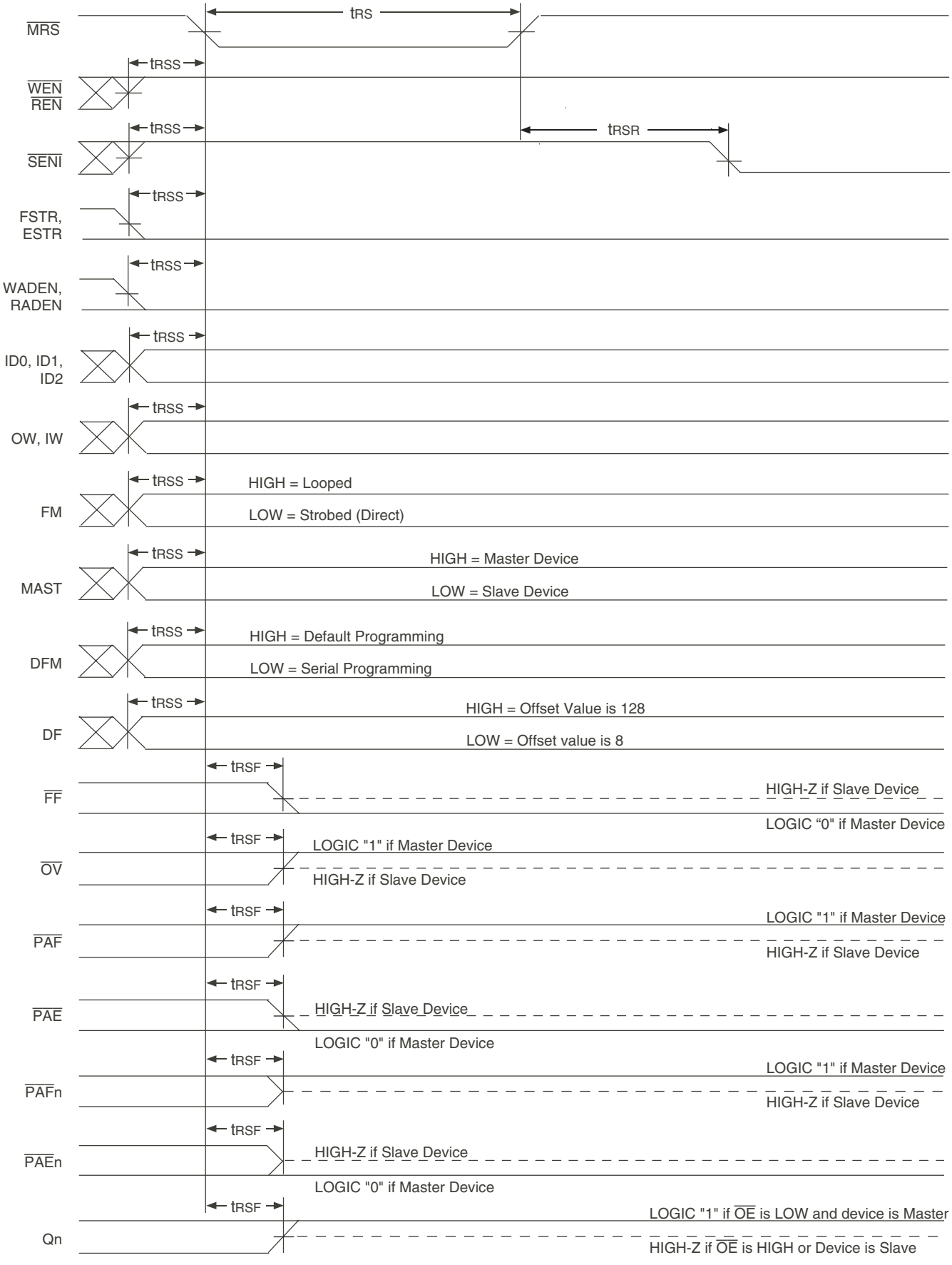


Figure 3. Bus-Matching Byte Arrangement

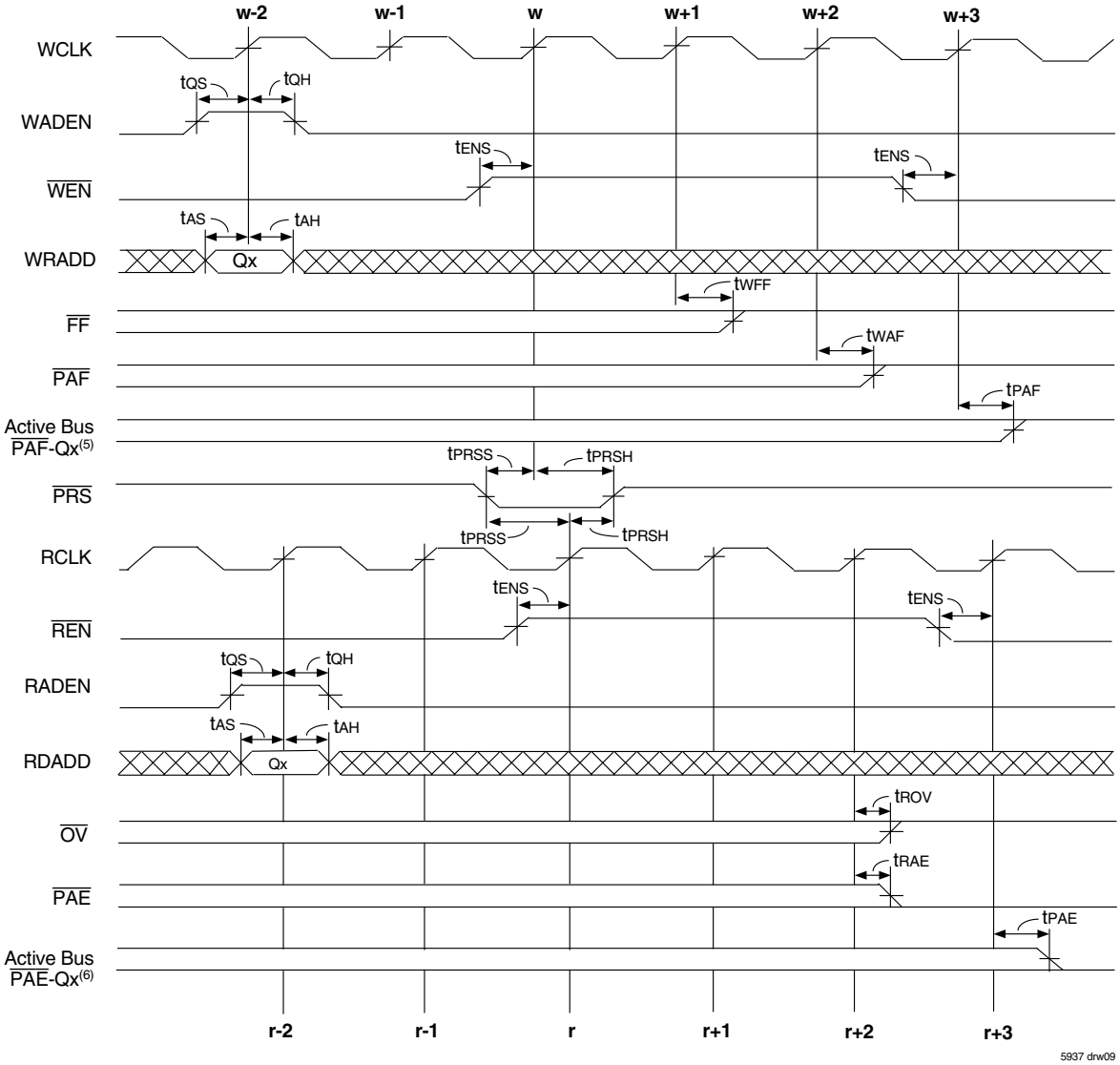
© 2019 Renesas Electronics Corporation



NOTES:
1. $\overline{OE}$ can toggle during this period.
2. $\overline{PRS}$ should be HIGH during a $\overline{MRS}$.

Figure 4. Master Reset

© 2019 Renesas Electronics Corporation



NOTES:

1. For a Partial Reset to be performed on a Queue, that Queue must be selected on both the write and read ports.
2. The queue must be selected a minimum of 2 clock cycles before the Partial Reset takes place, on both the write and read ports.
3. The Partial Reset must be LOW for a minimum of 1 WCLK and 1 RCLK cycle.
4. Writing or Reading to the queue (or a queue change) cannot occur until a minimum of 3 clock cycles after the Partial Reset has gone HIGH, on both the write and read ports.
5. The $\overline{PAF}$ flag output for Qx on the $\overline{PAF}$n flag bus may update one cycle later than the active $\overline{PAF}$ flag.
6. The $\overline{PAE}$ flag output for Qx on the $\overline{PAE}$n flag bus may update one cycle later than the active $\overline{PAE}$ flag.

Figure 5. Partial Reset

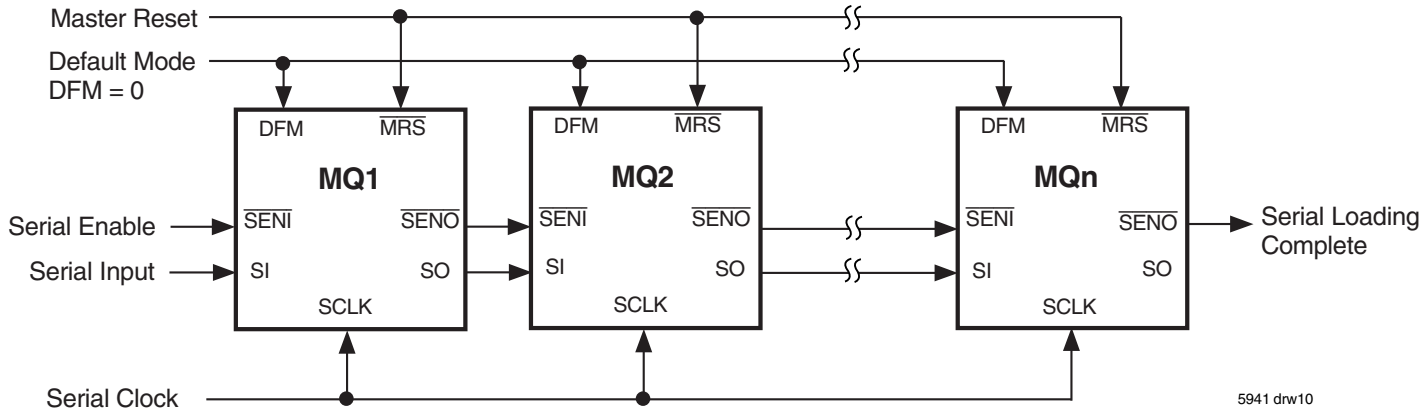
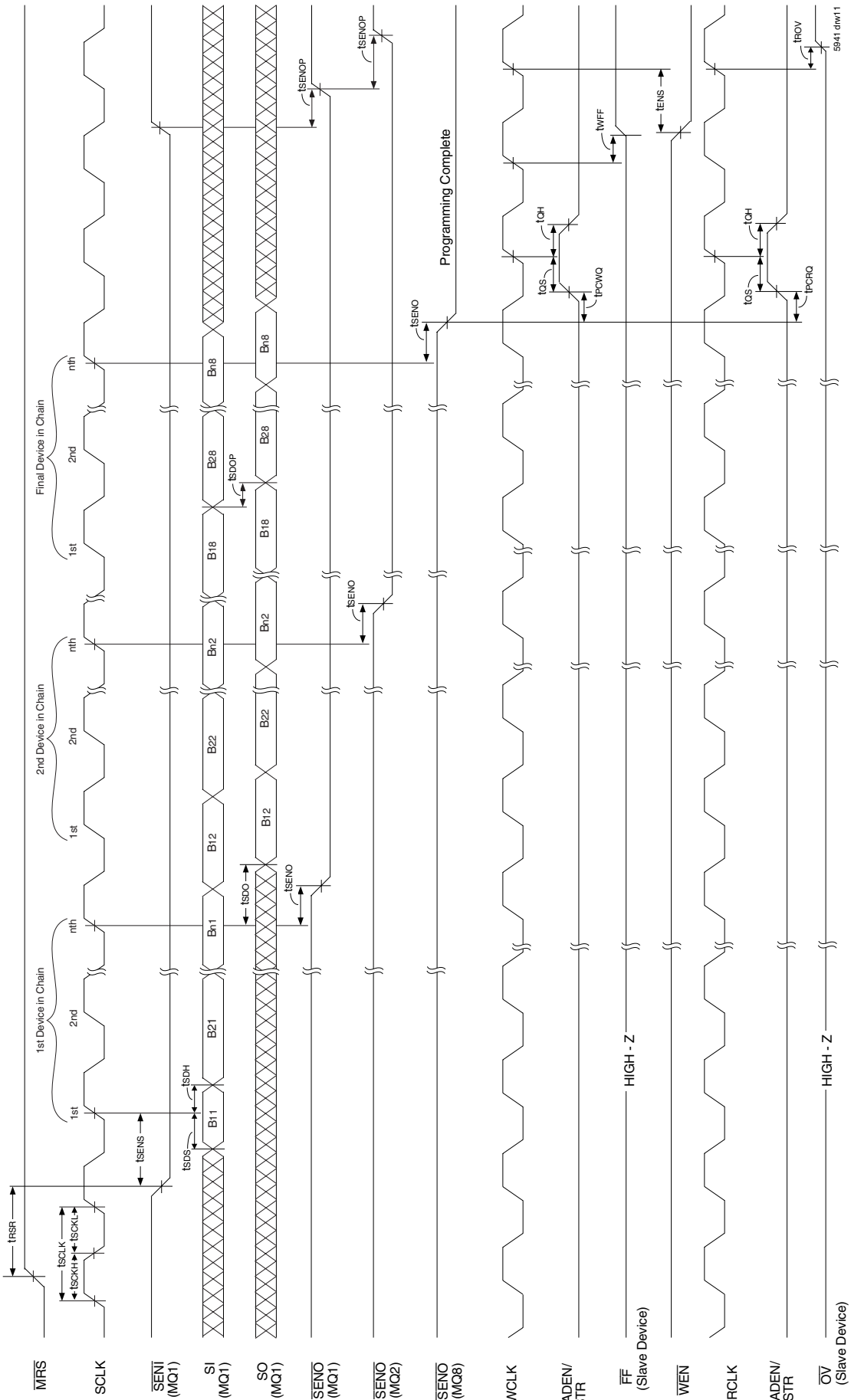


Figure 6. Serial Port Connection for Serial Programming

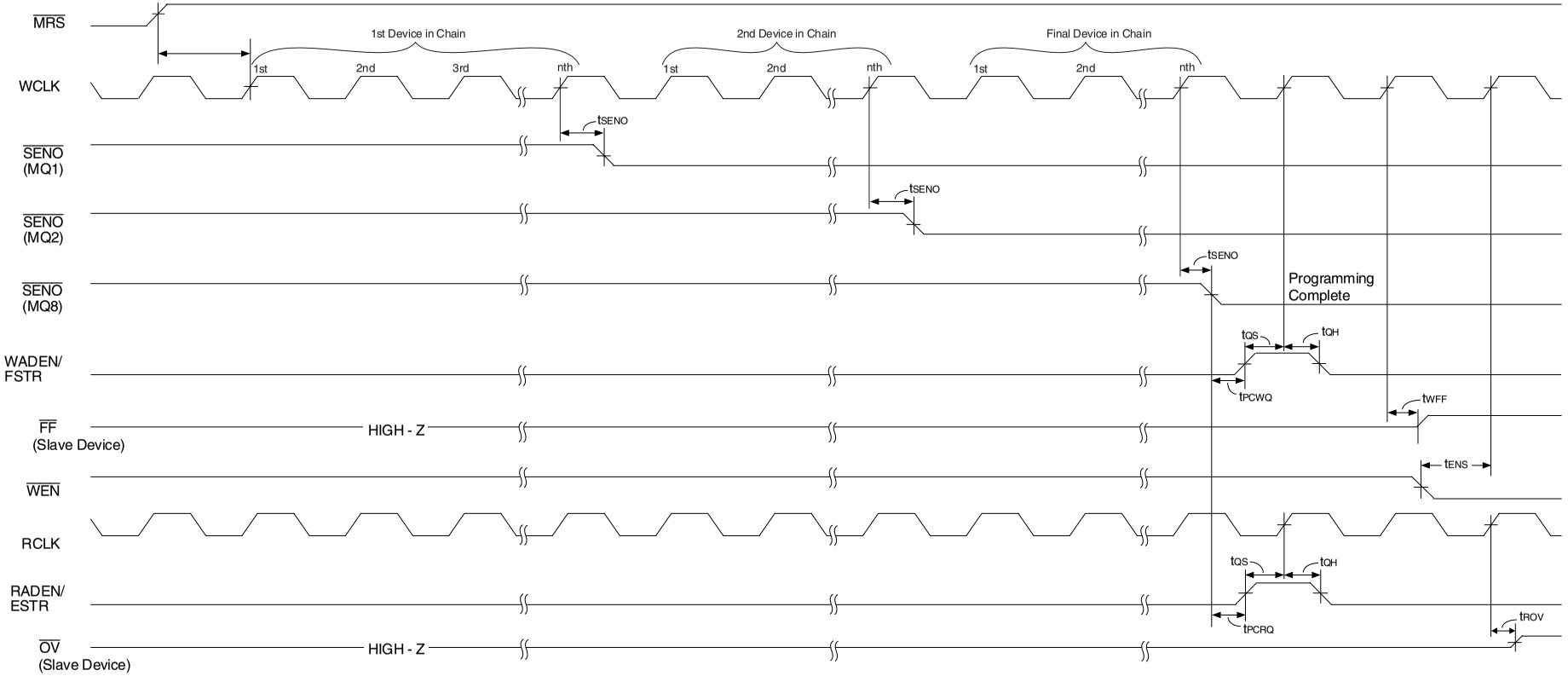
© 2019 Renesas Electronics Corporation

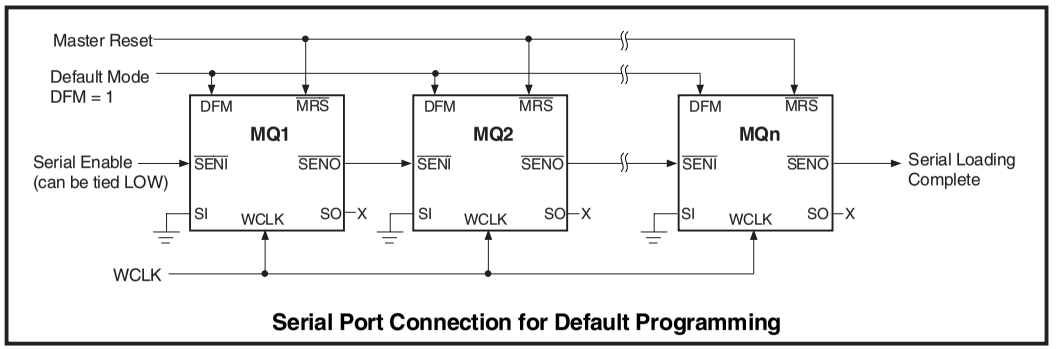


NOTES:
1. $\overline{SENI}$ can be toggled during serial loading. Once serial programming of a device is complete, the $\overline{SENI}$ and SI inputs become transparent. $\overline{SENI} \rightarrow \overline{SENO}$ and SI → SO.
2. DFM is LOW during Master Reset to provide Serial programming mode, DF is don't care.
3. When $\overline{SENO}$ of the final device is LOW no further serial loads will be accepted.
4. n = 19+(Qx72); where Q is the number of queues required for the IDT72V51233/72V51243/72V51253.
5. This diagram illustrates 8 devices in expansion.
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.

Figure 7. Serial Programming

© 2019 Renesas Electronics Corporation





Serial Port Connection for Default Programming

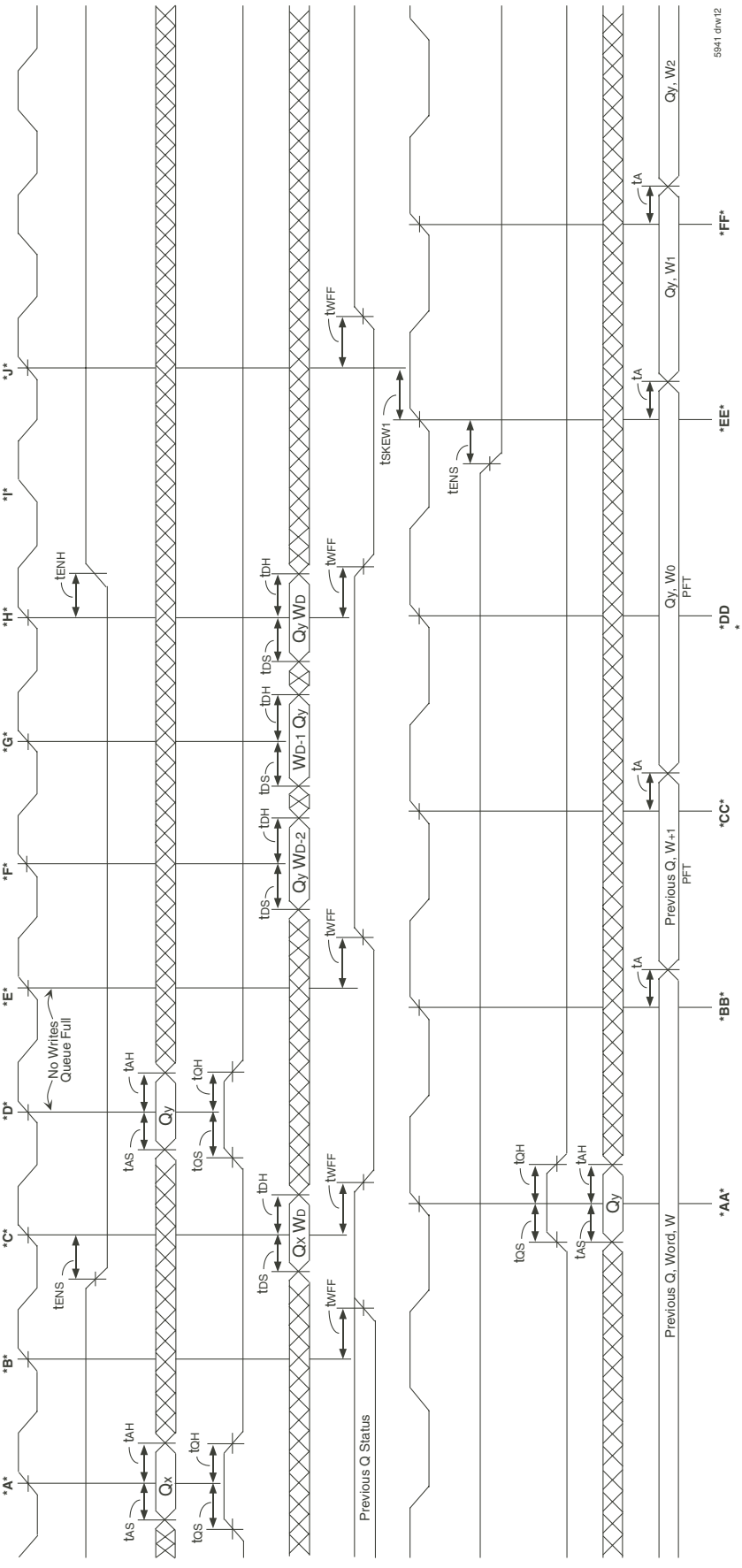
5941 drw11a

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 can 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.

Figure 8. Default Programming

© 2019 Renesas Electronics Corporation

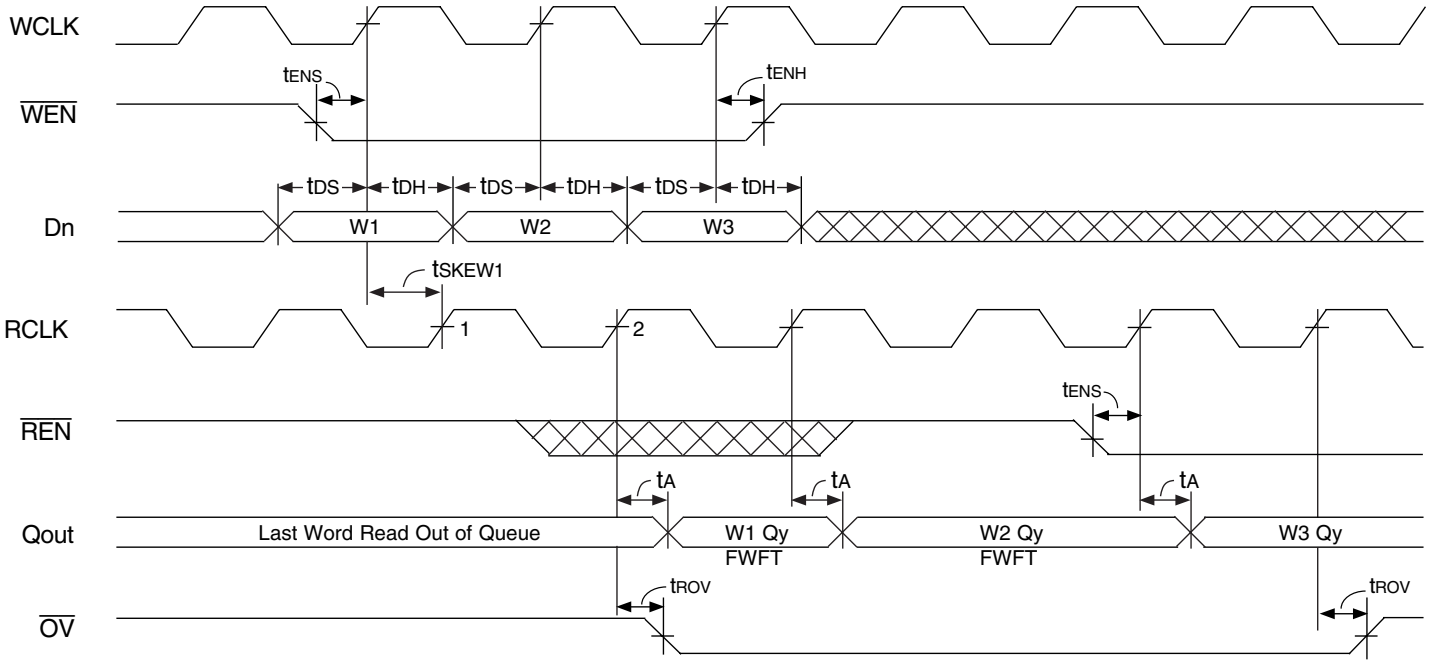


NOTE:
$\overline{OE}$ is active LOW.
Cycle:

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 Word W+1 is read from the previous queue regardless of $\overline{REN}$ due to FWFT.
C Word, Wd is written into Qx. This causes Qx to go full.
CC Word, W0 is read from Qy regardless of $\overline{REN}$, this is due to the FWFT effect.
D 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.
DD No reads occur, $\overline{REN}$ is HIGH.
E Again, a write to Qx cannot occur on this cycle because it is full, $\overline{FF}$ is LOW. The $\overline{FF}$ flag updates after time t_{WFF} to show that queue, Qy is not full.
EE Word, W1 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 Word, Wd-2 is written into Qy.
FF Word, W2 is read from Qy.
G Word, Wd-1 is written into Qy.
H Word, Wd is written into Qy, this causes Qy to go full, $\overline{FF}$ goes LOW.
I No writes occur to Qy.
J Qy goes "not full" based on reading word W1 from Qy on cycle *EE*.

Figure 9. Write Queue Select, Write Operation and Full Flag Operation

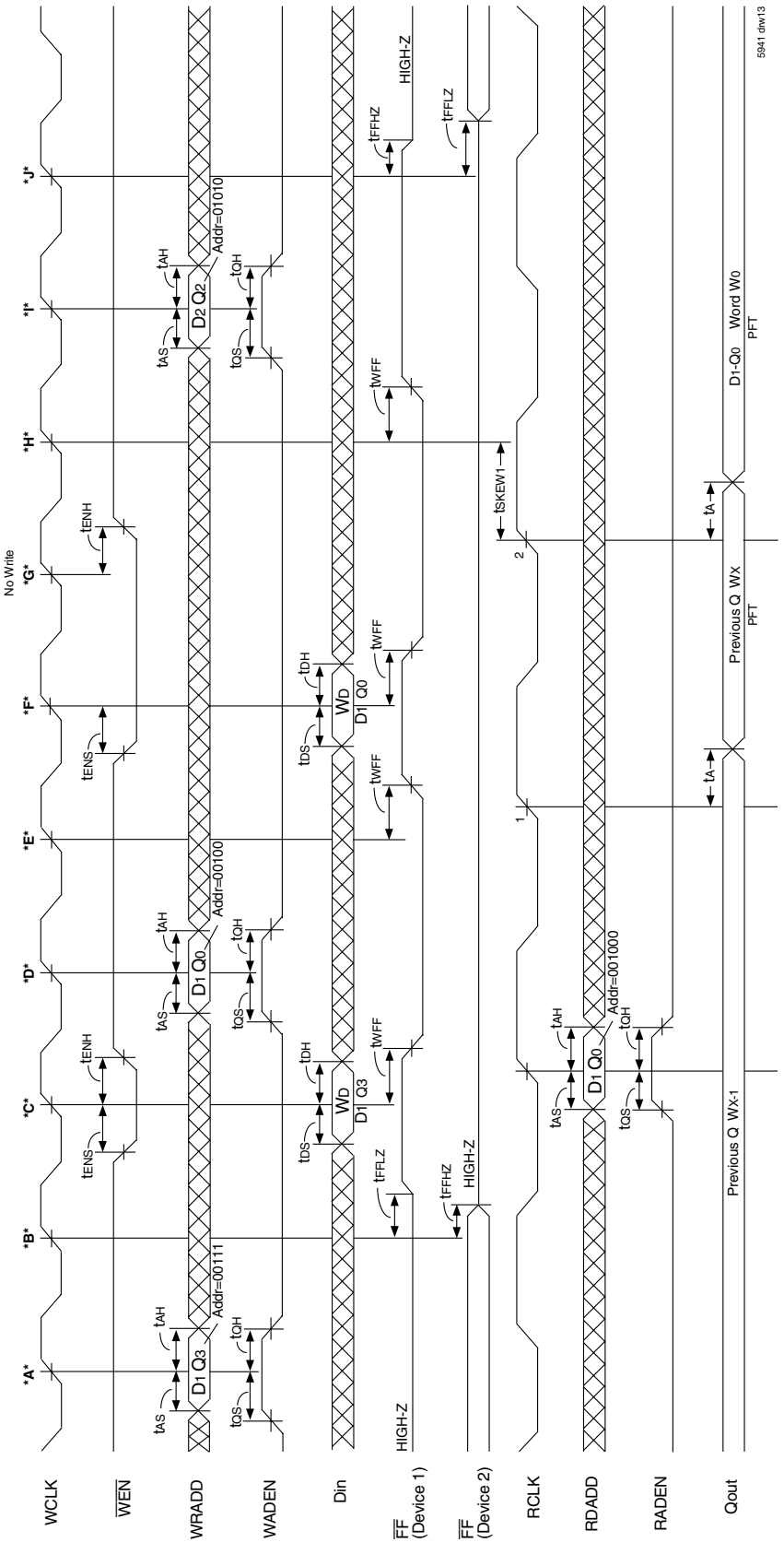
© 2019 Renesas Electronics Corporation



NOTES:
1. Qy has previously been selected on both the write and read ports.
2. $\overline{OE}$ is LOW.
3. The First Word Latency = tSKEW1 + RCLK + tA. If tSKEW1 is violated an additional RCLK cycle must be added.

Figure 10. Write Operations & First Word Fall Through

© 2019 Renesas Electronics Corporation



NOTE:
1. $\overline{REN}$ = HIGH.

Cycle:

A Queue, Q3 of device 1 is selected on the write port.
The $\overline{FF}$ flag of device 1 is in High-Impedance, the write port of device 2 was previously selected.
$\overline{WEN}$ is HIGH so no write occurs.

AA Queue, Q0 of device 1 is selected on the read port.

B The $\overline{FF}$ flag of device 2 goes to High-Impedance and the $\overline{FF}$ flag of device 1 goes to Low-Impedance, logic HIGH indicating that D1 Q3 is not full.
$\overline{WEN}$ is HIGH so no write occurs.

BB Word, Wx is read from the previously selected queue, (due to FWFT).

C Word, Wd is written into Q3 of D1. This write operation causes Q3 to go full, $\overline{FF}$ goes LOW.

CC The first word from Q0 of D1 selected on cycle *AA* is read out, this occurred regardless of $\overline{REN}$ due to FWFT. This read caused Q0 to go not full, therefore the $\overline{FF}$ flag will go HIGH after: t_{SKEW1} + t_{WFF}.
Note if t_{SKEW1} is violated the time to $\overline{FF}$ flag HIGH is t_{SKEW1} + WLCK + t_{WFF}.

D Queue, Q0 of device 1 is selected on the write port. No write occurs on this cycle.

E The $\overline{FF}$ flag updates to show the status of D1 Q0, it is not full, $\overline{FF}$ goes HIGH.

F Word, Wd is written into Q0 of D1. This causes the queue to go full, $\overline{FF}$ goes LOW.

G No write occurs regardless of $\overline{WEN}$, the $\overline{FF}$ flag is LOW preventing writes.

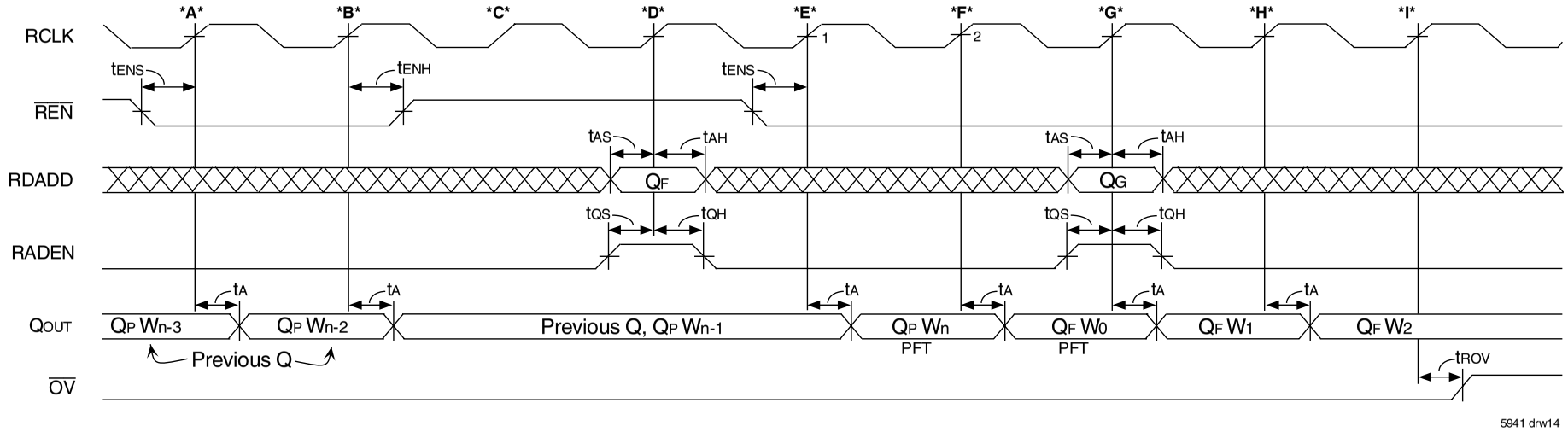
H The $\overline{FF}$ flag goes HIGH due to the read from Q0 of D1 on cycle *CC*. (This read is not an enabled read, it is due to the FWFT operation).

I Queue, Q2 of device 2 is selected on the write port.

J The $\overline{FF}$ flag of device 1 goes to High-Impedance, this device was deselected on the write port on cycle *I*. The $\overline{FF}$ flag of device 2 goes to Low-Impedance and provides status of Q2 of D2.

Figure 11. Full Flag Timing in Expansion Mode

© 2019 Renesas Electronics Corporation



Cycle:

A Word W_{n-3} is read from a previously selected queue Qp on the read port.

B W_{n-2} is read.

C Reads are disabled, W_{n-1} remains on the output bus.

D A new queue, Q_F is selected for read operations.

E Due to the First Word Fall Through (FWFT) effect, a read from the previous queue Qp will take place, Wn from Qp is placed onto the output bus regardless of $\overline{REN}$.

F The next word available in the new queue, Q_F-W_0 falls through to the output bus, again this is regardless of $\overline{REN}$.

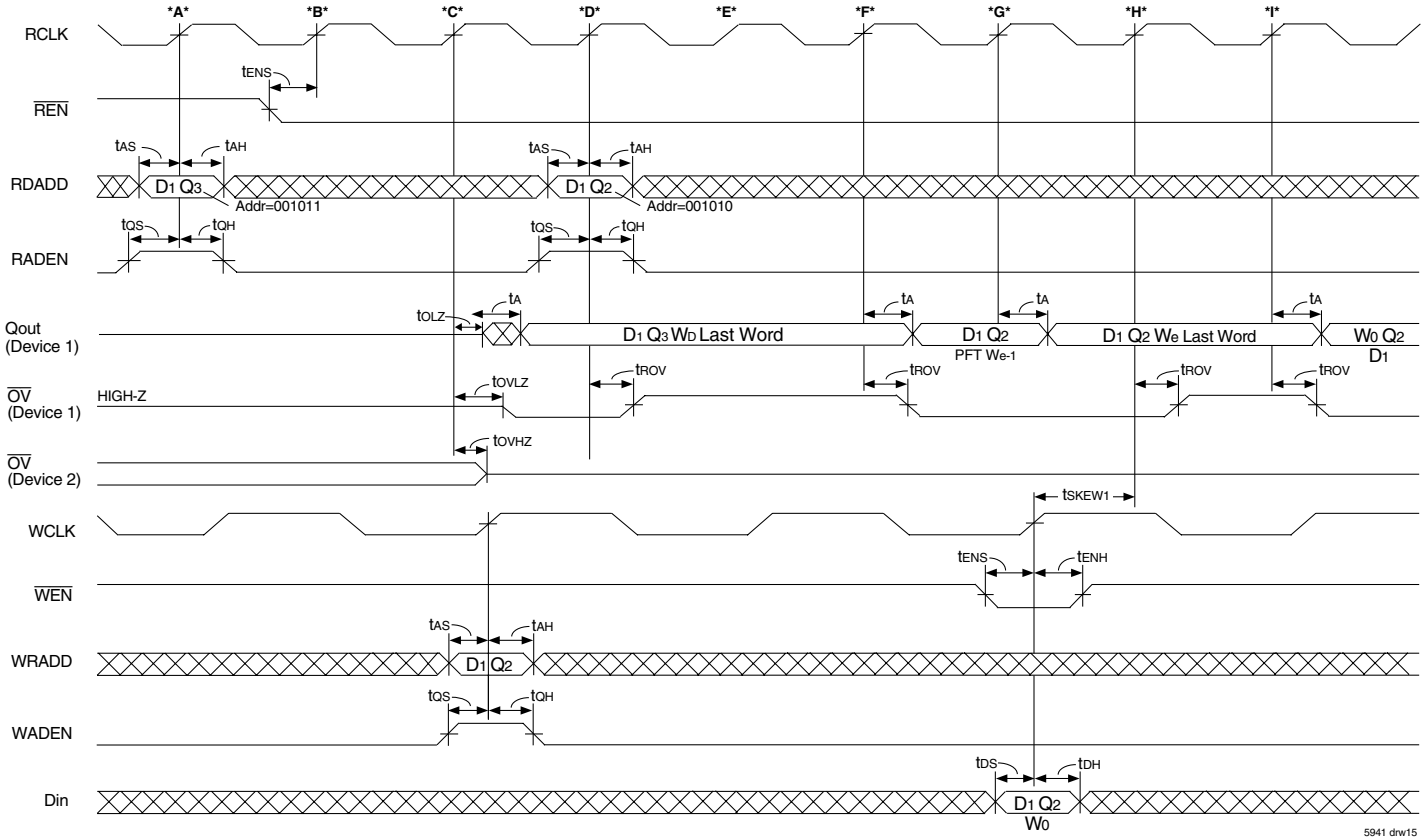
G A new queue, Q_G is selected for read operations. (This queue is an empty queue). Word, W_1 is also read from Q_F.

H Word, W_1 is read from Q_F. This occurs regardless of $\overline{REN}$ due to FWFT.

I Word W_2 from Q_F remains on the output bus because Q_G is empty. The Output Valid Flag, $\overline{OV}$ goes HIGH to indicate that the current word is not valid, i.e. Q_G is empty. W_2 is the last word in Q_G.

Figure 12. Read Queue Select, Read Operation

© 2019 Renesas Electronics Corporation



Cycle:

A Queue 3 of Device 1 is selected for read operations. The $\overline{OV}$ is currently being driven by Device 2, a queue within device 2 is selected for reads. Device 2 also has control of Qout bus, its Qout outputs are in Low-Impedance. This diagram only shows the Qout outputs of device 1. (Reads are disabled).

B Reads are now enabled. A word from the previously selected queue of Device 2 will be read out.

C The Qout of Device 1 goes to Low-Impedance and word Wd is read from Q3 of D1. This happens to be the last word of Q3. Device 2 places its Qout outputs into High-Impedance, device 1 has control of the Qout bus. The $\overline{OV}$ flag of Device 2 goes to High-Impedance and Device 1 takes control of $\overline{OV}$. The $\overline{OV}$ flag of Device 1 goes LOW to show that Wd of Q3 is valid.

D Queue 2 of device 1 is selected for read operations. The last word of Q3 was read on the previous cycle, therefore $\overline{OV}$ goes HIGH to indicate that the data on the Qout is not valid (Q3 was read to empty). Word, Wd remains on the output bus.

E The last word of Q3 remains on the Qout bus, $\overline{OV}$ is HIGH, indicating that this word has been previously read.

F The next word (We-1), available from the newly selected queue, Q2 of device 1 is now read out. This will occur regardless of $\overline{REN}$, 2 RCLK cycles after queue selection due to the FWFT operation. The $\overline{OV}$ flag now goes LOW to indicate that this word is valid.

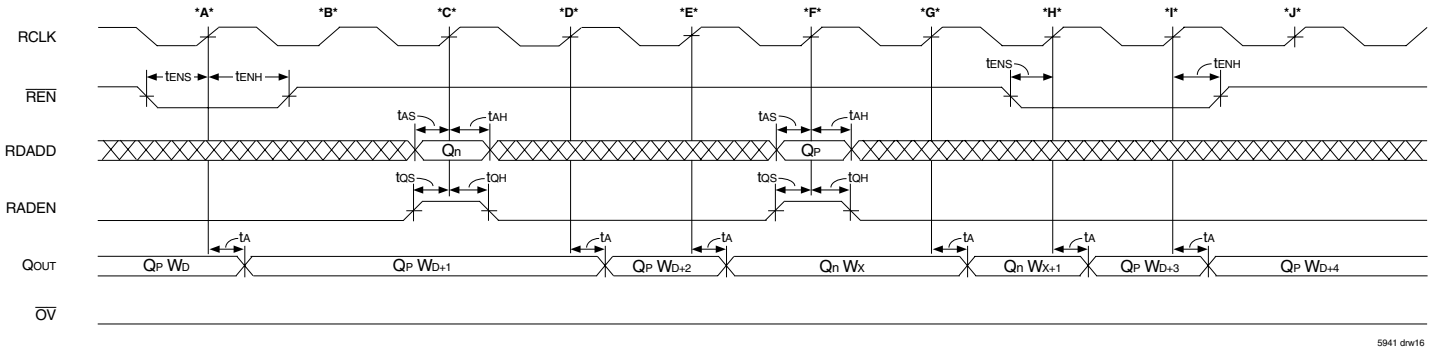
G The last word, We is read from Q2, this queue is now empty.

H The $\overline{OV}$ flag goes HIGH to indicate that Q2 was read to empty on the previous cycle.

I Due to a write operation the $\overline{OV}$ flag goes LOW and data word W0 is read from Q2. The latency is: t_{SKEW1} + 1*RCLK + t_{ROV}.

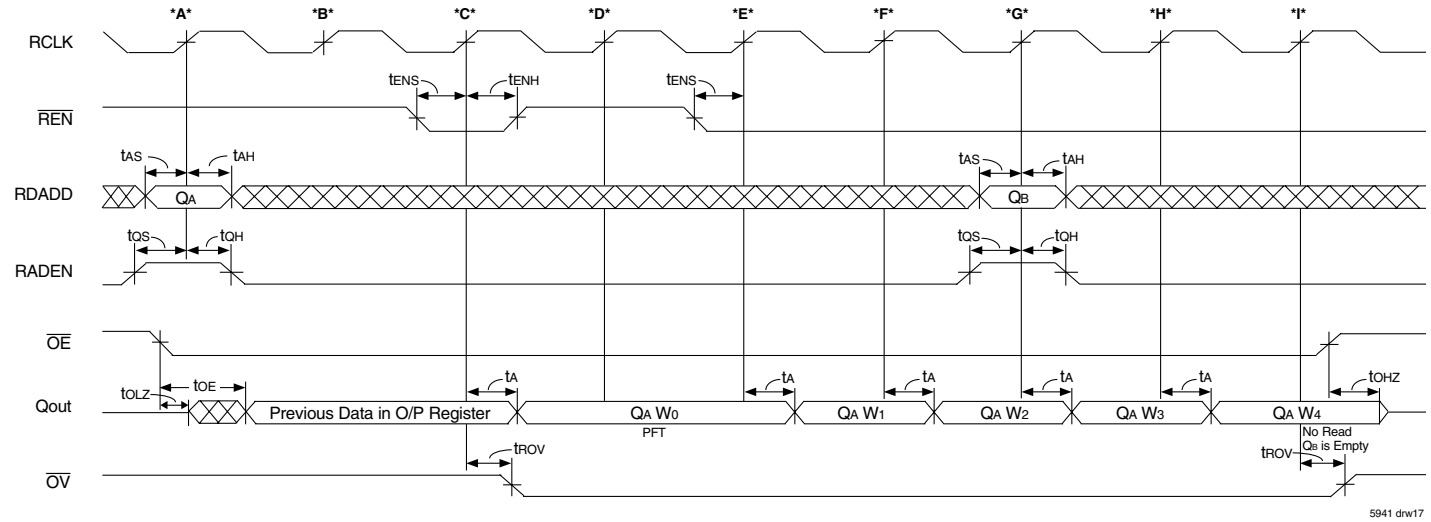
Figure 13. Output Valid Flag Timing (In Expansion Mode)

© 2019 Renesas Electronics Corporation



Cycle:
A Word Wd+1 is read from the previously selected queue, Qp.
B Reads are disabled, word Wd+1 remains on the output bus.
C A new queue, Qn is selected for read port operations.
D Due to FWFT operation Word, Wd+2 of Qp is read out regardless of $\overline{REN}$.
E The next available word Wx of Qn is read out regardless of $\overline{REN}$, 2 RCLK cycles after queue selection. This is FWFT operation.
F The queue, Qp is again selected.
G Word Wx+1 is read from Qn regardless of $\overline{REN}$, this is due to FWFT.
H Word Wd+3 is read from Qp, this read occurs regardless of $\overline{REN}$ due to FWFT operation.
I Word Wd+4 is read from Qp.
J Reads are disabled on this cycle, therefore no further reads occur.

Figure 14. Read Queue Selection with Reads Disabled

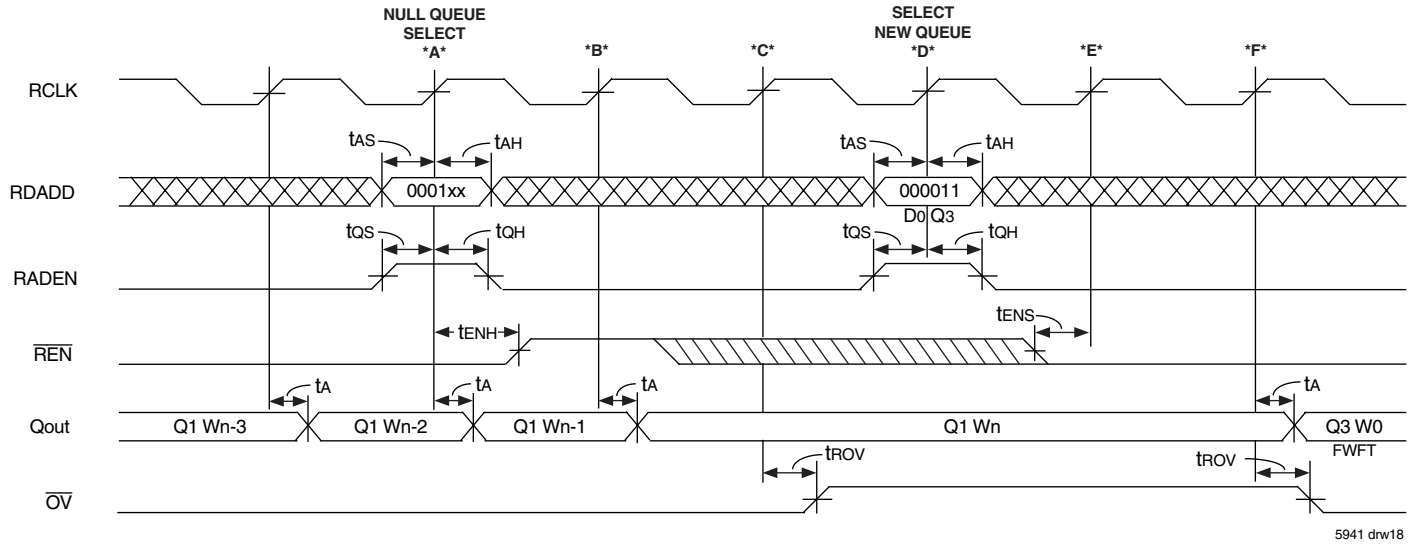


NOTES:
1. The Output Valid flag, $\overline{OV}$ is HIGH therefore the previously selected queue has been read to empty. The Output Enable input is Asynchronous, therefore the Qout output bus will go to Low-Impedance after time tOLZ.
   The data currently on the output register will be available on the output after time tOE. This data is the previous data on the output register, this is the last word read out of the previous queue.
2. In expansion mode the $\overline{OE}$ 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 data will fall through on this cycle, the previous queue was read to empty.
C Word, W0 from Qa is read out regardless of $\overline{REN}$ due to FWFT operation. The $\overline{OV}$ flag goes LOW indicating that Word W0 is valid.
D Reads are disabled therefore word, W0 of Qa remains on the output bus.
E Reads are again enabled so word W1 is read from Qa.
F Word W2 is read from Qa.
G Queue, Qb is selected on the read port. This queue is actually empty. Word, W3 is read from Qa.
H Word, W4 falls through from Qa.
I Output Valid flag, $\overline{OV}$ goes HIGH to indicate that Qb is empty. Data on the output port is no longer valid.
Output Enable is taken HIGH, this is Asynchronous so the output bus goes to High-Impedance after time, tOHZ.

Figure 15. Read Queue Select, Read Operation and $\overline{OE}$ Timing

© 2019 Renesas Electronics Corporation



NOTES:

1. The purpose of the Null queue operation is so that the user can stop reading a block (packet) of data from a queue without filling the 2 stage output pipeline with the next words from that queue.
2. Please see Figure 17, *Null Queue Flow Diagram*.

Cycle:

A Null-Q of device 0 is selected, when word Wn-1 from previously selected Q1 is read.

B $\overline{REN}$ is HIGH and Wn (Last Word of the Packet) of Q1 is pipelined onto the O/P register.

Note: *B* and *C* are a minimum 2 RCLK cycles between Q selects.

C The Null-Q is seen as an empty queue on the read side, therefore Wn of Q1 remains in the O/P register and $\overline{OV}$ goes HIGH.

D A new Q, Q3 is selected and the 1st word of Q3 will fall through present on the O/P register on cycle *F*.

Figure 16. Read Operation and Null Queue Select

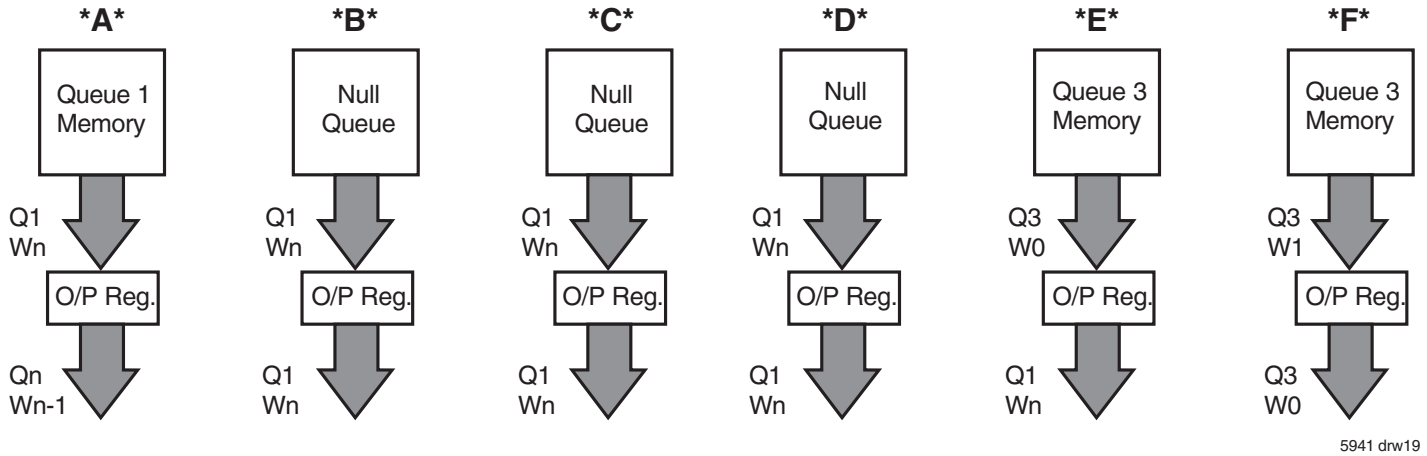
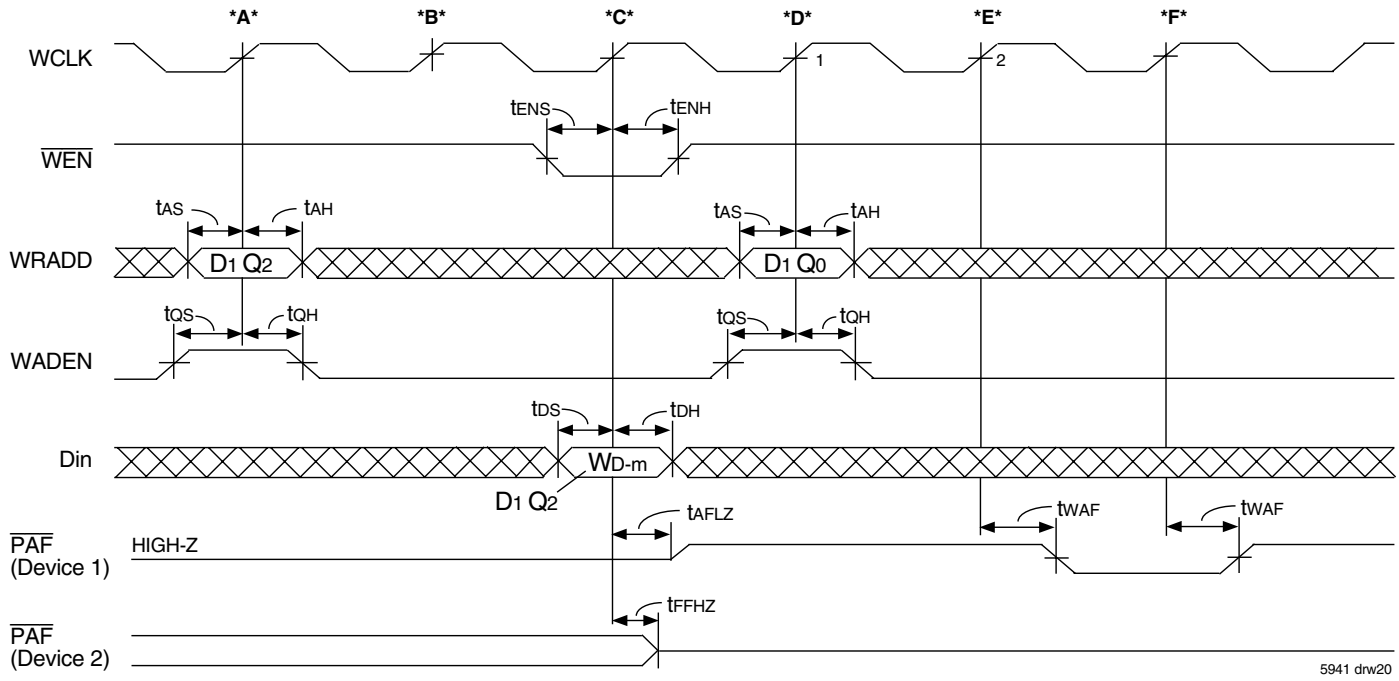


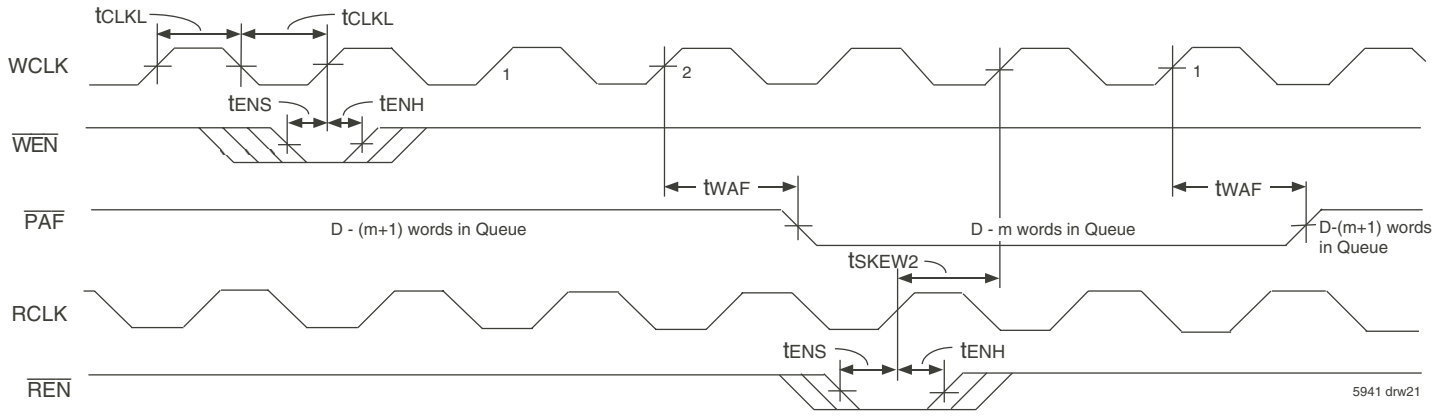
Figure 17. Null Queue Flow Diagram

© 2019 Renesas Electronics Corporation



Cycle:
A Queue 2 of Device 1 is selected on the write port. A queue within Device 2 had previously been selected. The $\overline{PAF}$ output of device 1 is High-Impedance.
B No write occurs.
C Word, Wd-m is written into Q2 causing the $\overline{PAF}$ flag to go from HIGH to LOW. The flag latency is 2 WCLK cycles + t_{WAF}.
D Queue 0 if device 1 is now selected for write operations. This queue is not almost full, therefore the $\overline{PAF}$ flag will update after a 2 WCLK + t_{WAF} latency.
E The $\overline{PAF}$ flag goes LOW based on the write 2 cycles earlier.
F The $\overline{PAF}$ flag goes HIGH due to the queue switch to Q0.

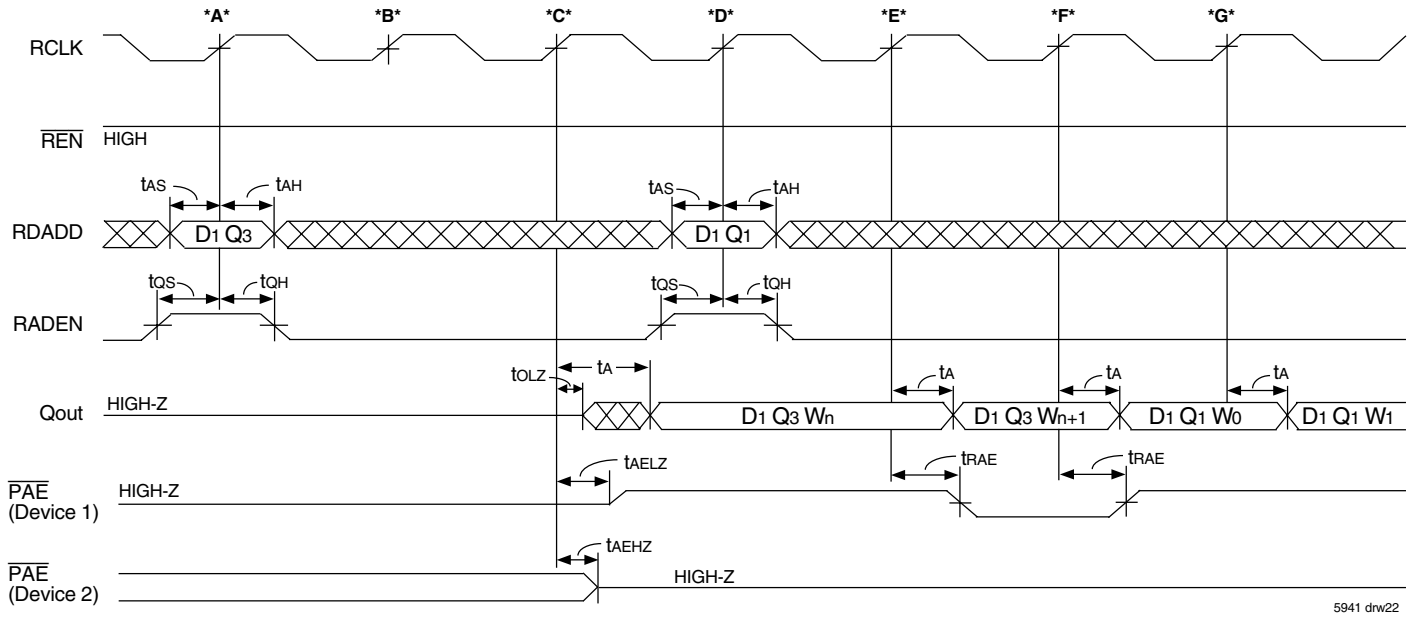
Figure 18. Almost Full Flag Timing and Queue Switch



NOTE:
1. The waveform here shows the $\overline{PAF}$ 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 full boundary.
   Flag Latencies:
   Assertion: 2*WCLK + t_{WAF}
   De-assertion: t_{SKEW2} + WCLK + t_{WAF}
   If t_{SKEW2} is violated there will be one extra WCLK cycle.

Figure 19. Almost Full Flag Timing

© 2019 Renesas Electronics Corporation



Cycle:

A Queue 3 of Device 1 is selected on the read port. A queue within Device 2 had previously been selected. The $\overline{PAE}$ flag output and the data outputs of device 1 are High-Impedance.

B No read occurs.

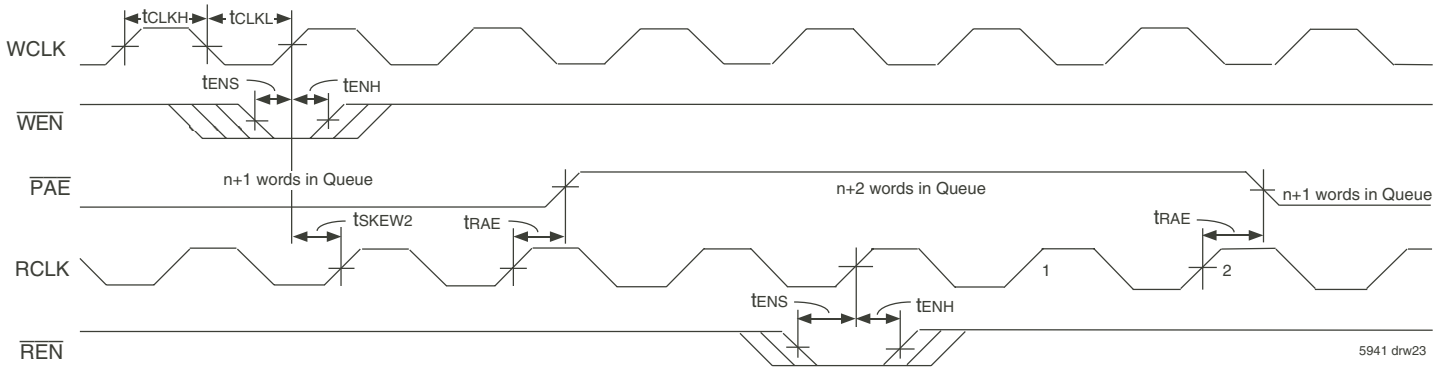
C The $\overline{PAE}$ flag output now switches to device 1. Word, Wn is read from Q3 due to the FWFT operation. This read operation from Q3 is at the almost empty boundary, therefore $\overline{PAE}$ will go LOW 2 RCLK cycles later.

D Q1 of device 1 is selected.

E The $\overline{PAE}$ flag goes LOW due to the read from Q3 2 RCLK cycles earlier. Word Wn+1 is read out due to the FWFT operation.

F Word, W0 is read from Q1 due to the FWFT operation. The $\overline{PAE}$ flag goes HIGH to show that Q1 is not almost empty.

Figure 20. Almost Empty Flag Timing and Queue Switch

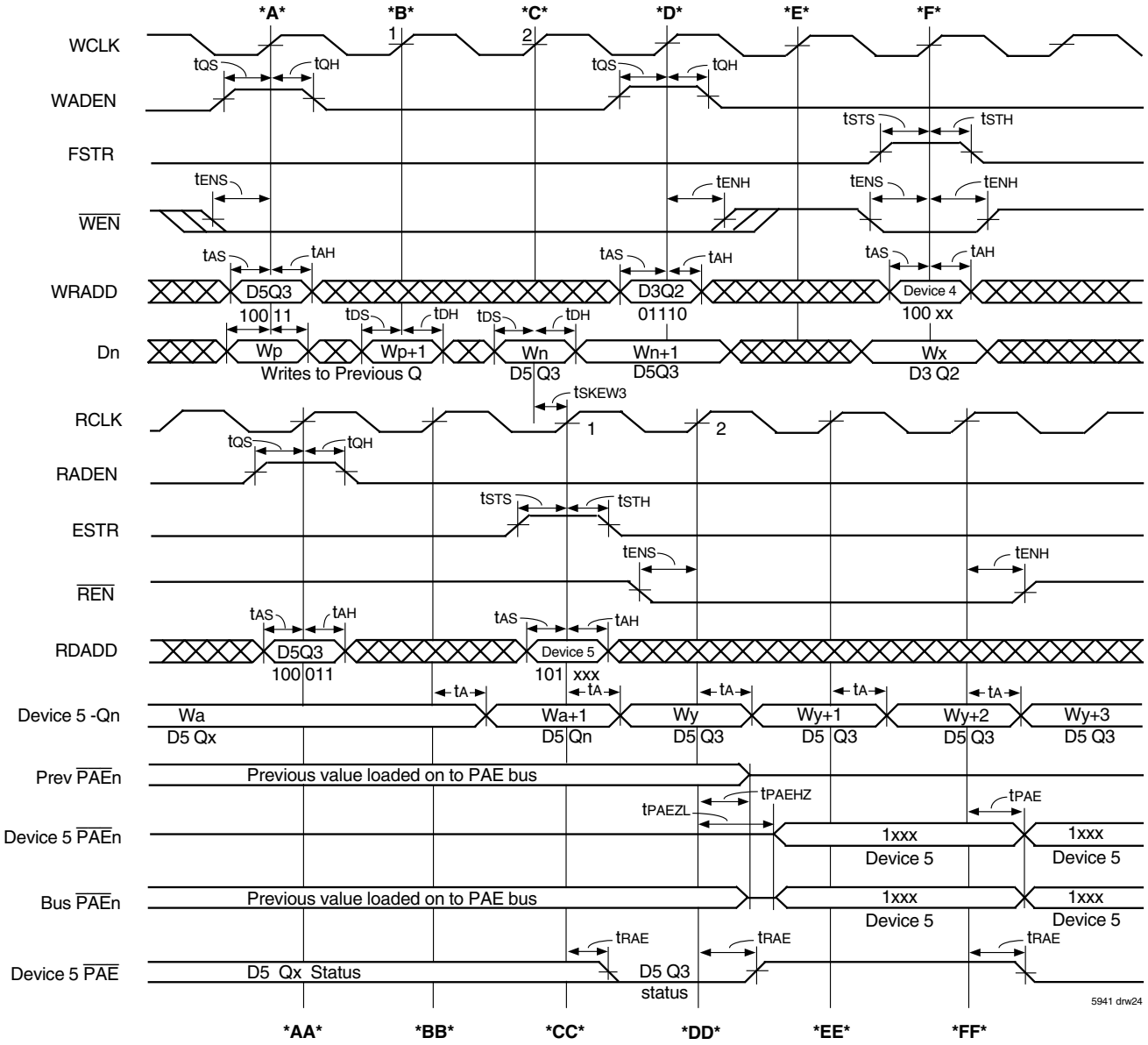


NOTE:

1. The waveform here shows the $\overline{PAE}$ 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:
   Assertion: 2*RCLK + t_{RAE}
   De-assertion: t_{SKEW2} + RCLK + t_{RAE}
   If t_{SKEW2} is violated there will be one extra RCLK cycle.

Figure 21. Almost Empty Flag Timing

© 2019 Renesas Electronics Corporation



Cycle:

A Queue 3 of Device 5 is selected for write operations.
Word, Wp is written into the previously selected queue.

AA Queue 3 of Device 5 is selected for read operations.
Another device has control of the $\overline{PAE}$n bus.
The discrete $\overline{PAE}$ output of device 5 is currently in High-Impedance and the $\overline{PAE}$ active flag is controlled by the previously selected device.

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

BB Word, Wa+1 is read from Qx of D5, due to FWFT operation.

C Word, Wn is written into the newly selected queue, Q3 of D5. This write will cause the $\overline{PAE}$ flag on the read port to go from LOW to HIGH (not almost empty) after time, t_{SKEW3} + RCLK + t_{RAE} (if t_{SKEW3} is violated one extra RCLK cycle will be added.

CC Word, Wy from the newly selected queue, Q3 will be read out due to FWFT operation.
Device 5 is selected on the $\overline{PAE}$n bus. Q3 of device 5 will therefore have is $\overline{PAE}$ status output on $\overline{PAE}$[3]. There is a single RCLK cycle latency before the $\overline{PAE}$n bus changes to the new selection.

D Queue 2 of Device 3 is selected for write operations.
Word Wn+1 is written into Q3 of D5.

DD The $\overline{PAE}$n bus changes control to D5, the $\overline{PAE}$n outputs of D5 go to Low-Impedance and are placed onto the outputs. The previously selected device now places its $\overline{PAE}$n outputs into High-Impedance to prevent bus contention. Word, Wy+1 is read from Q3 of D5.
The discrete $\overline{PAE}$ flag will go HIGH to show that Q3 of D5 is not almost empty. Q3 of device 5 will have its $\overline{PAE}$ status output on $\overline{PAE}$[3].

E No writes occur.

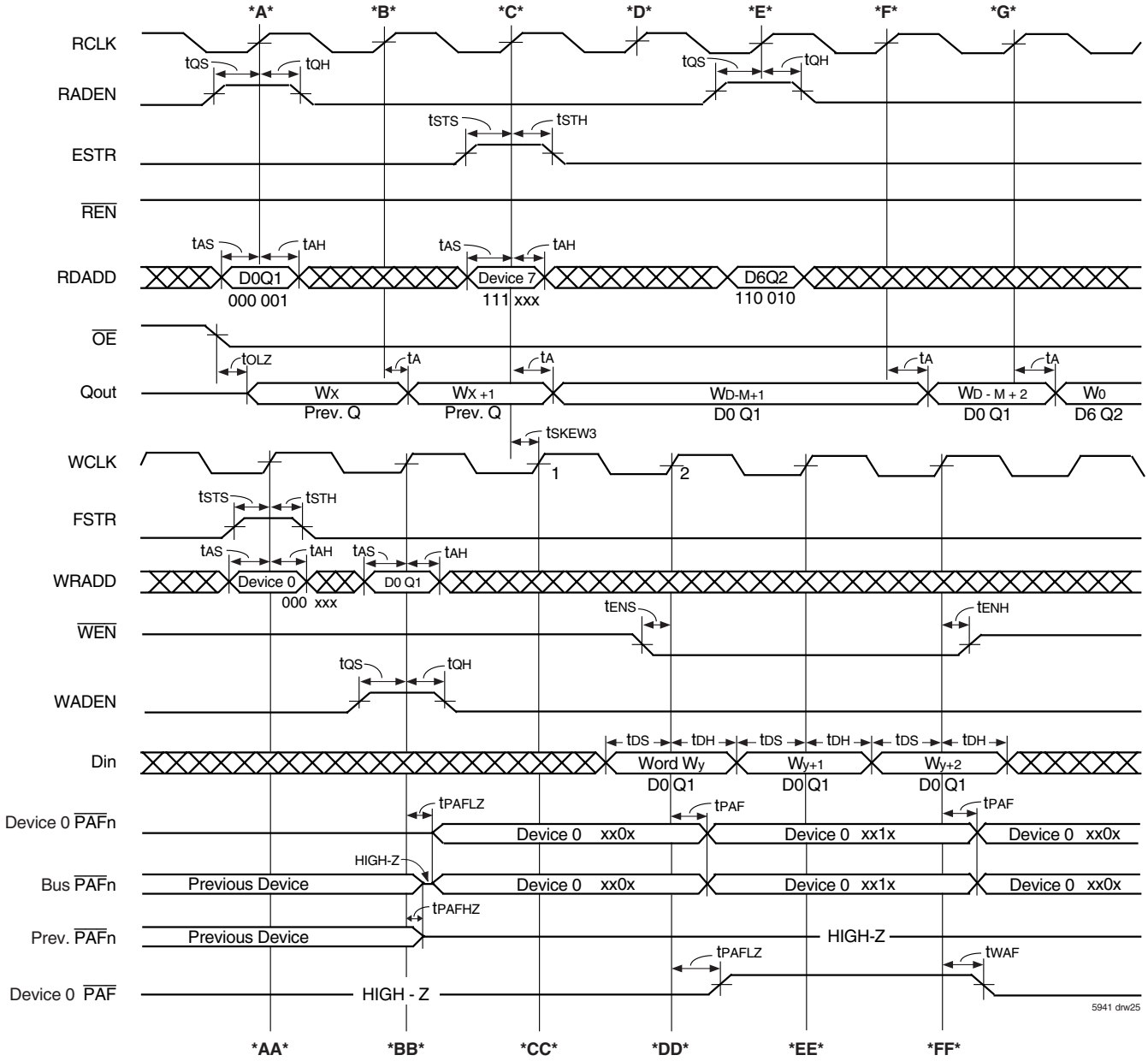
EE Word, Wy+2 is read from Q3 of D5.

F Device 4 is selected on the write port for the $\overline{PAF}$n bus.
Word, Wx is written into Q2 of D3.

FF The $\overline{PAE}$n bus updates to show that Q3 of D5 is almost empty based on the reading out of word, Wy+1.
The discrete $\overline{PAE}$ flag goes LOW to show that Q3 of D5 is almost empty based on the reading of Wy+1.

Figure 22. $\overline{PAE}$n - Direct Mode, Flag Operation – Devices in Expansion

© 2019 Renesas Electronics Corporation



Cycle:

A Queue 1 of device 0 is selected for read operations.
The last word in the output register is available on Qout. $\overline{OE}$ was previously taken LOW so the output bus is in Low-Impedance.

AA Device 0 is selected for the $\overline{PAF}$n bus. The bus is currently providing status of a previously selected device X.

B Word, Wx+1 is read out from the previous queue due to the FWFT effect.

BB Queue 1 of device 0 is selected on the write port.
The $\overline{PAF}$n bus is updated with the device selected on the previous cycle, device 0 $\overline{PAF}$[1] is LOW showing the status of queue 1.
The $\overline{PAF}$n outputs of the device previously selected on the $\overline{PAF}$n bus go to High-Impedance.

C Device 7 is selected for the $\overline{PAF}$n bus.
Word, Wd-m+1 is read from Q1 D0 due to the FWFT operation. This read is at the $\overline{PAF}$n boundary of queue D0 Q1. This read will cause the $\overline{PAF}$[1] output to go from LOW to HIGH (almost full to not almost full), after a delay t_{SKEW3} + WCLK + tPAF. If t_{SKEW3} is violated add an extra WCLK cycle.

CC $\overline{PAF}$n continues to show status of D0.

D No read operations occur, $\overline{REN}$ is HIGH.

DD $\overline{PAF}$[1] goes HIGH to show that D0 Q1 is not almost empty due to the read on cycle *C*.
The active queue $\overline{PAF}$ flag of device 0 goes from High-Impedance to Low-Impedance.
Word, Wy is written into D0 Q1.

E Queue 2 of Device 6 is selected for write operations.

EE Word, Wy+1 is written into D0 Q1.

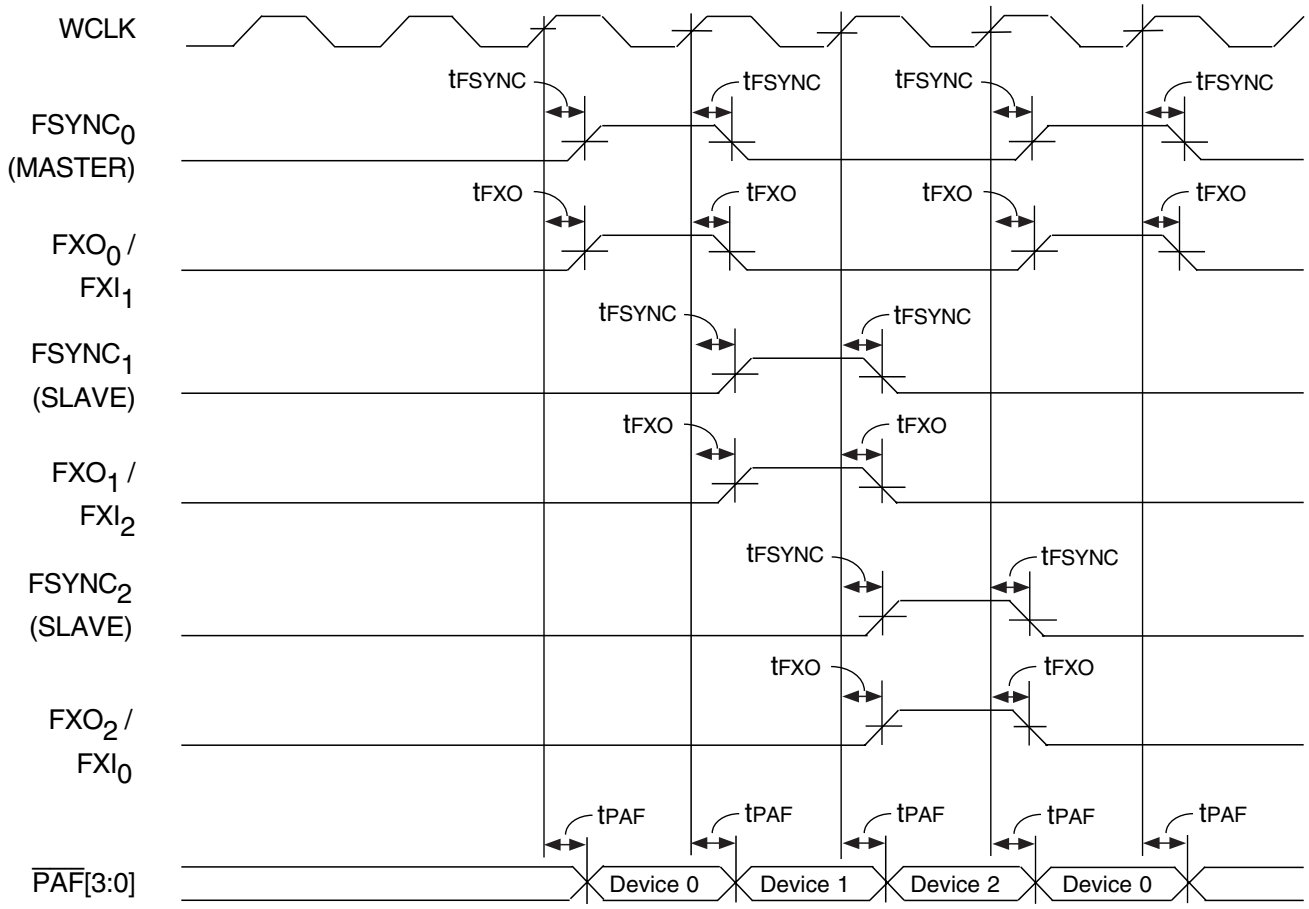
F Word, Wd-m+2 is read out due to FWFT operation.

FF $\overline{PAF}$[1] and the discrete $\overline{PAF}$ flag go LOW to show the write on cycle *DD* causes Q1 of D0 to again go almost full.
Word, Wy+2 is written into D0 Q1.

G Word, W0 is read from Q0 of D6, selected on cycle *E*, due to FWFT.

Figure 23. $\overline{PAF}$n - Direct Mode, Flag Operation – Devices in Expansion

© 2019 Renesas Electronics Corporation



NOTE:

1. This diagram is based on 3 devices connected to expansion mode.

Figure 24. PAFn Bus - Polled Mode

© 2019 Renesas Electronics Corporation

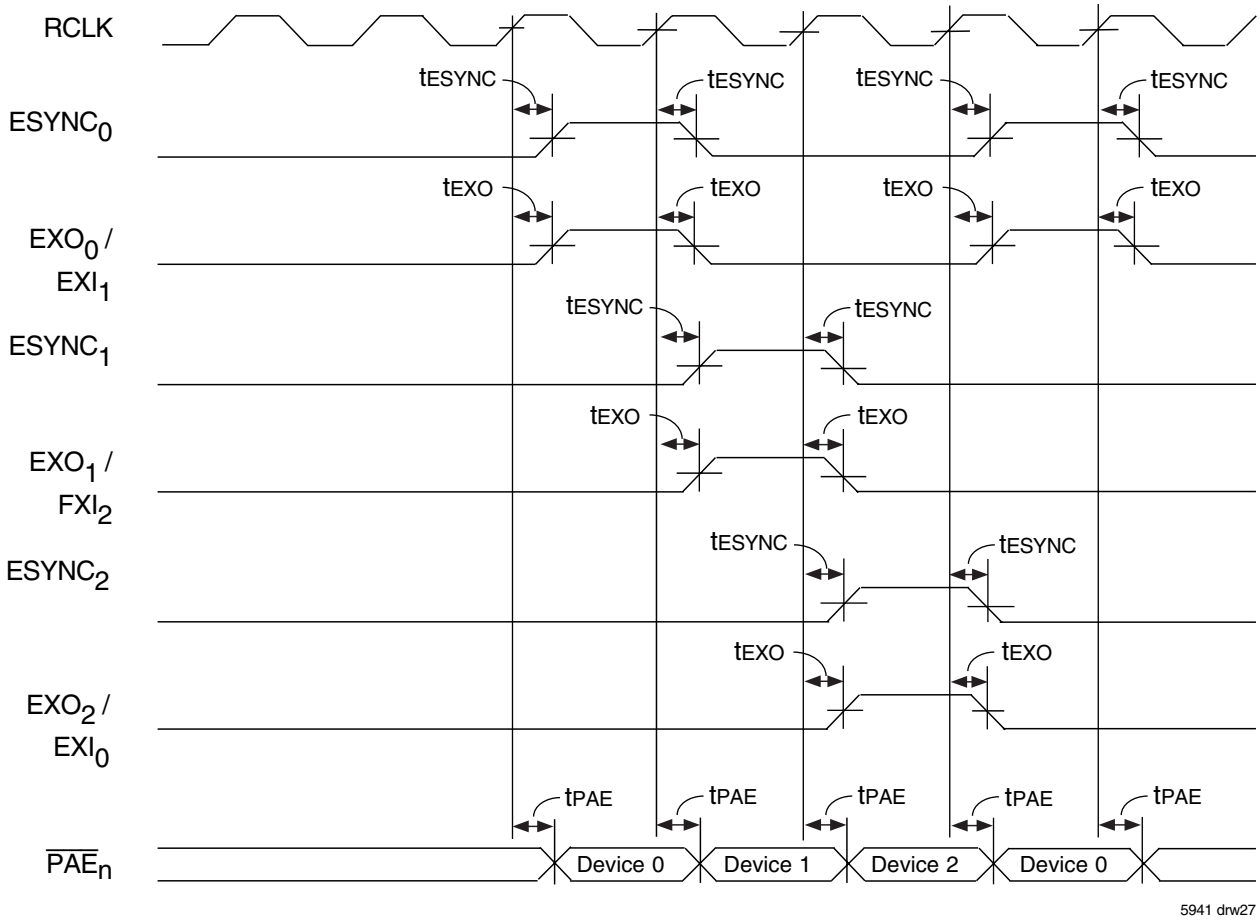
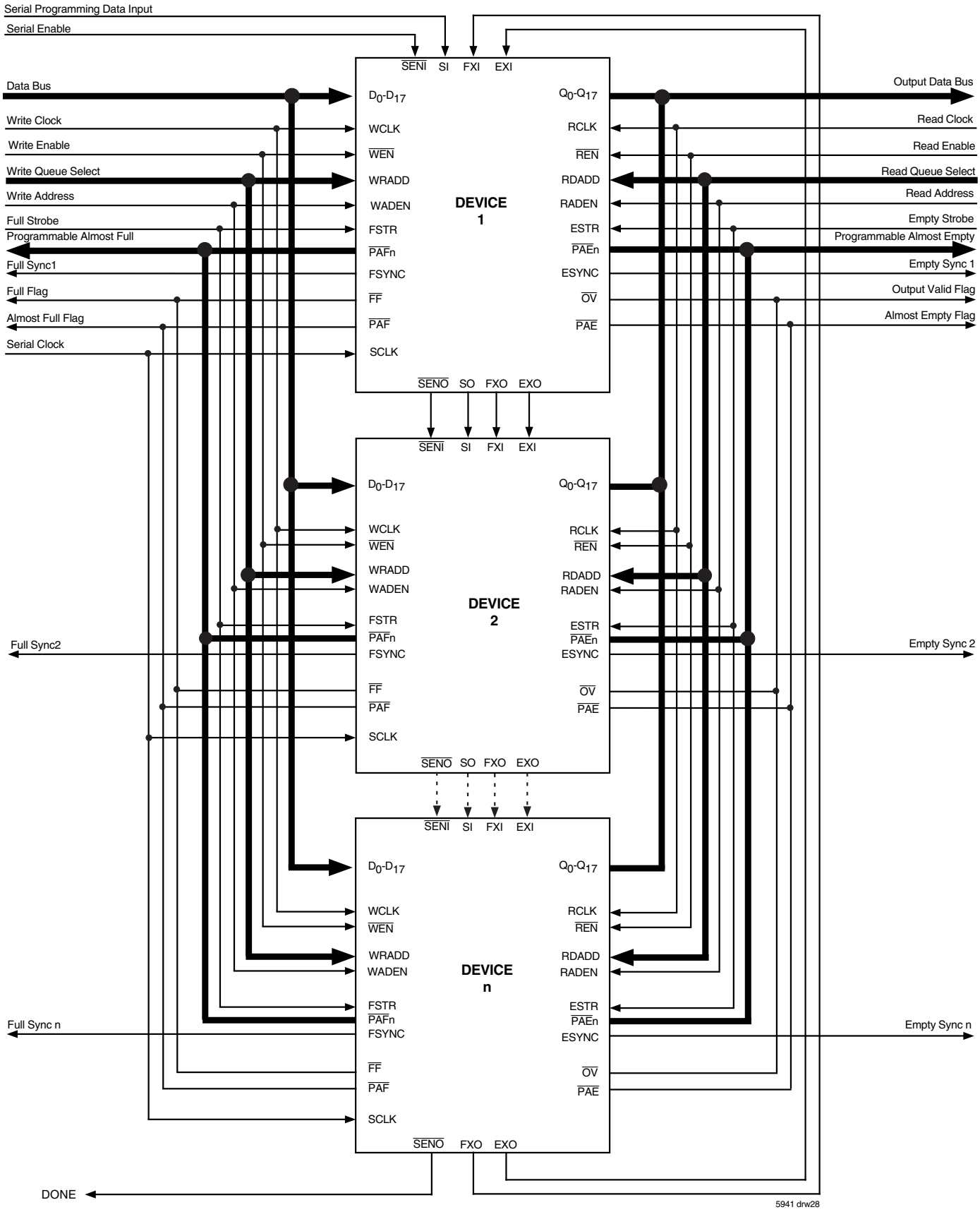


Figure 25. $\overline{PAE}_n$ Bus - Polled Mode

© 2019 Renesas Electronics Corporation



NOTES:

1. If devices are configured for Direct operation of the $\overline{PAF}$n/$\overline{PAE}$n flag busses the FXI/EXI of the MASTER device should be tied LOW. All other devices tied HIGH. The FXO/EXO outputs are DNC (Do Not Connect).
2. Q outputs must not be mixed between devices, i.e. Q0 of device 1 must connect to Q0 of device 2, etc.

Figure 26. Multi-Queue Expansion Diagram

© 2019 Renesas Electronics Corporation

JTAG INTERFACE

Five additional pins (TDI, TDO, TMS, TCK and $\overline{TRST}$) are provided to support the JTAG boundary scan interface. The IDT72V51233/72V51243/72V51253 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

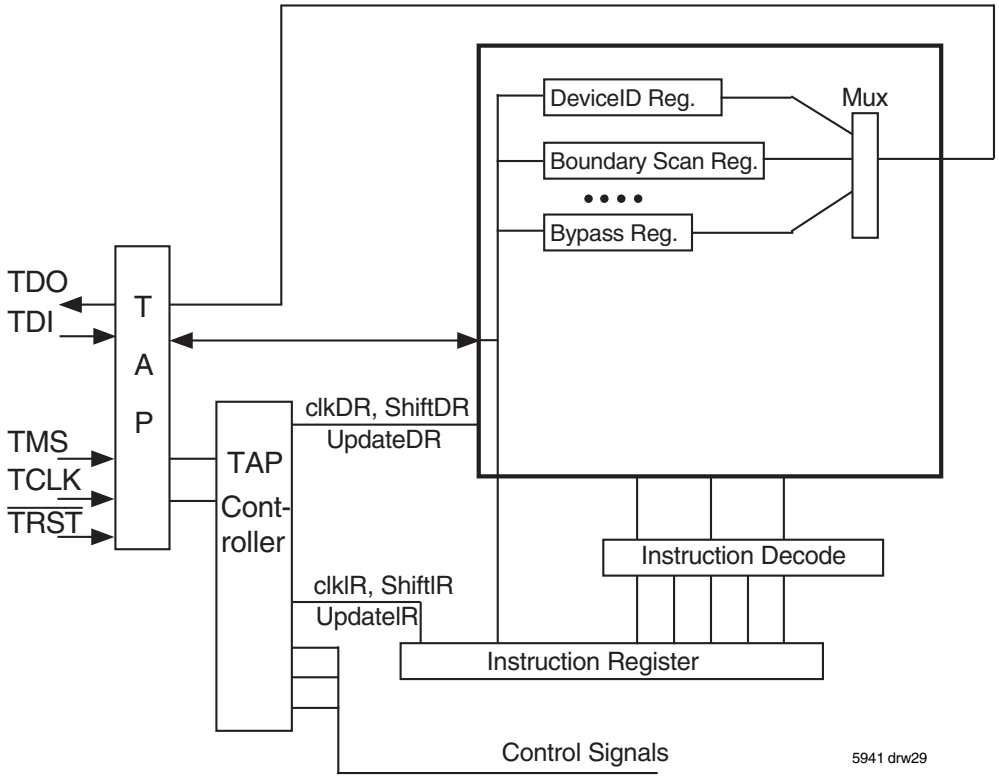


Figure 27. Boundary Scan Architecture

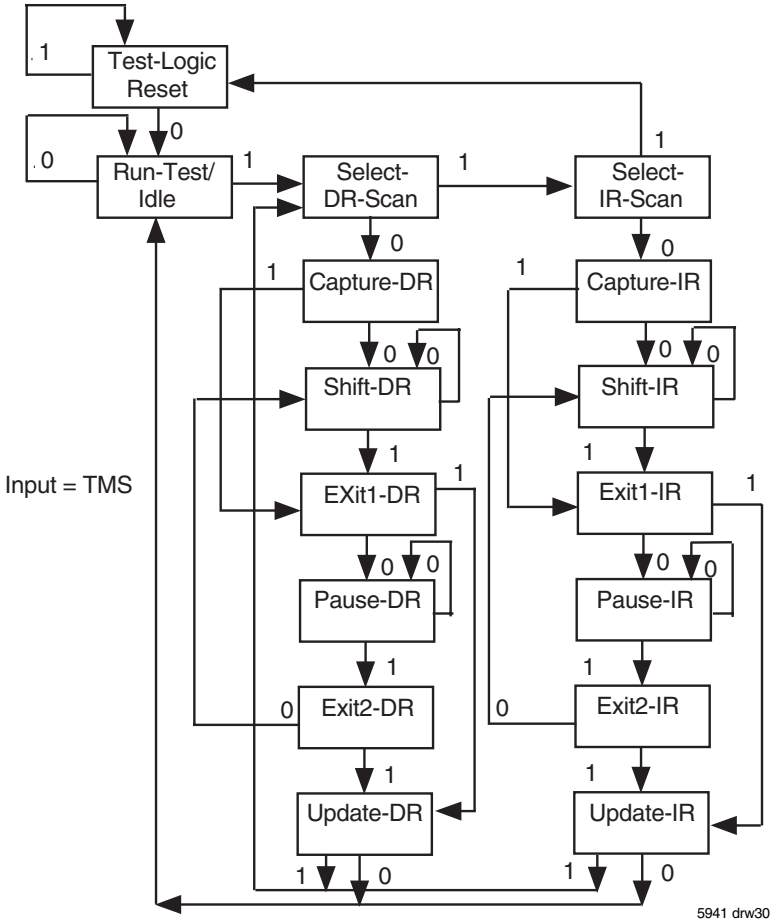
TEST ACCESS PORT (TAP)

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, $\overline{TRST}$) 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.

© 2019 Renesas Electronics Corporation



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 $\overline{TRST}$ or TMS).
3. TAP controller must be reset before normal Queue operations can begin.

Figure 28. 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 $\overline{TRST}$ 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 ($\overline{TRST}$) 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.

© 2019 Renesas Electronics Corporation

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 IDT72V51233/72V51243/72V51253, the Part Number field contains the following values:

Device	Part# Field (HEX)
IDT72V51233	0x411
IDT72V51243	0x412
IDT72V51253	0x413

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
04	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.

© 2019 Renesas Electronics Corporation

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.

© 2019 Renesas Electronics Corporation

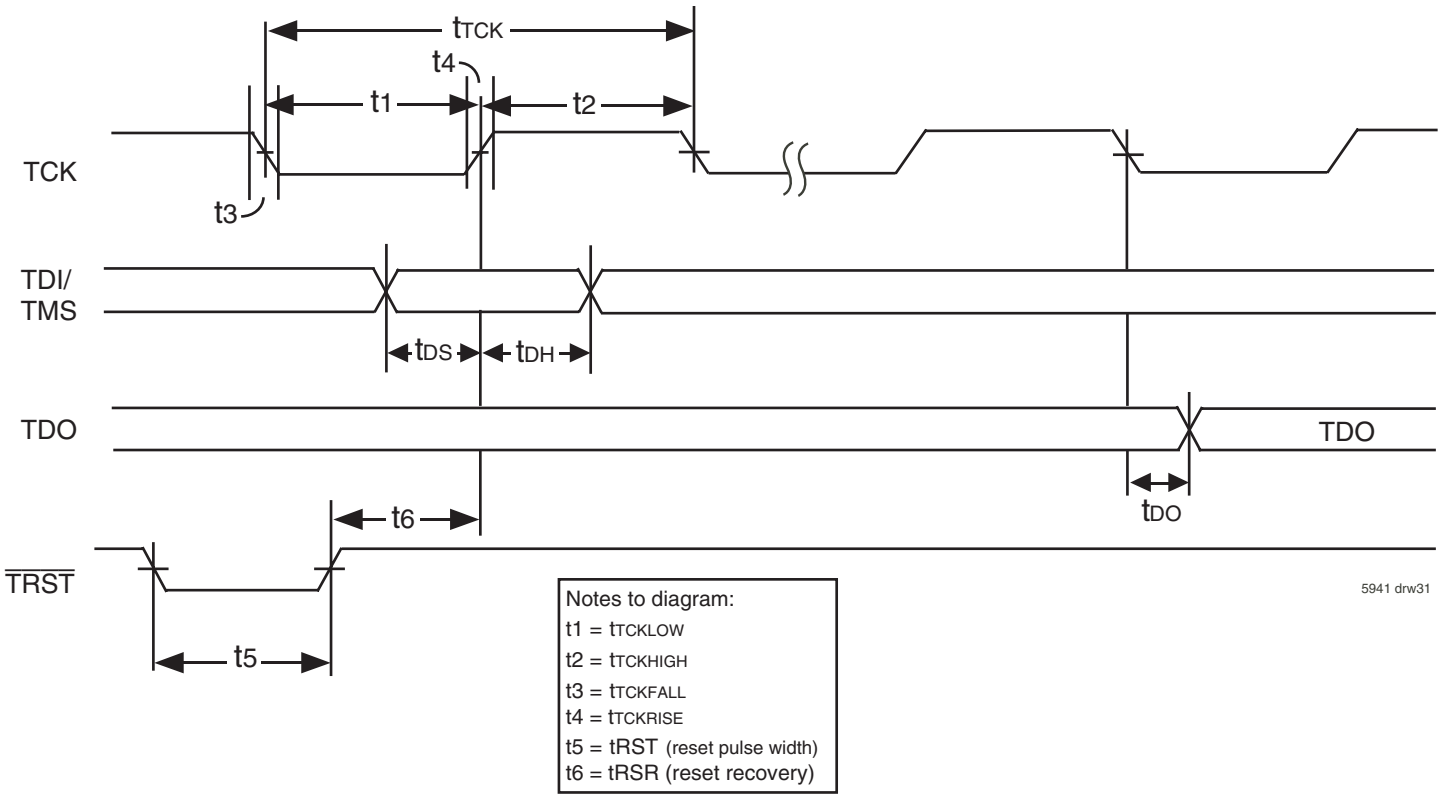


Figure 29. Standard JTAG Timing

SYSTEM INTERFACE PARAMETERS

Parameter	Symbol	Test Conditions	IDT72V51233 IDT72V51243 IDT72V51253		
			Min.	Max.	Units
Data Output	t_{DO}[1]		-	20	ns
Data Output Hold	t_{DOH}[1]		0	-	ns
Data Input	t_{DS}	trise=3ns	10	-	ns
	t_{DH}	tfall=3ns	10	-	

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

JTAG AC ELECTRICAL CHARACTERISTICS

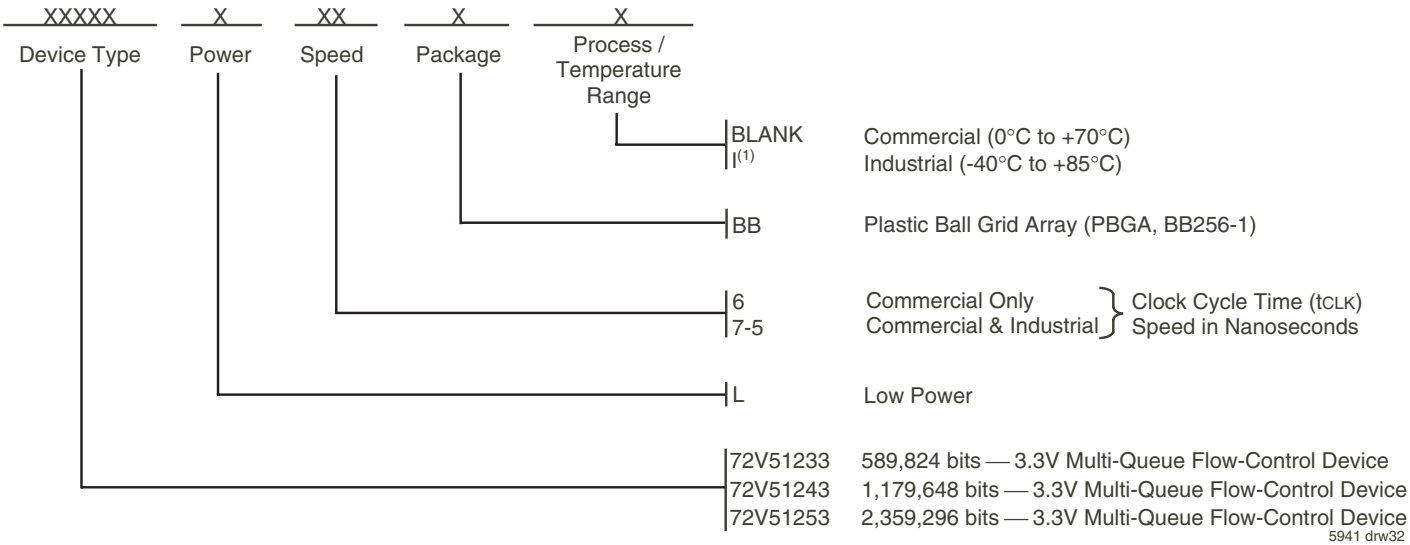
(Vcc = 3.3V ± 5%; Tcase = 0°C to +85°C)

Parameter	Symbol	Test Conditions			
			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.

© 2019 Renesas Electronics Corporation

ORDERING INFORMATION



5941 drw32

NOTE:

1. Industrial temperature range product for the 7-5ns is available as a standard device. All other speed grades are available by special order.

IMPORTANT NOTICE AND DISCLAIMER

RENESAS ELECTRONICS CORPORATION AND ITS SUBSIDIARIES (“RENESAS”) PROVIDES TECHNICAL SPECIFICATIONS AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for developers skilled in the art designing with Renesas products. You are solely responsible for (1) selecting the appropriate products for your application, (2) designing, validating, and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, or other requirements. These resources are subject to change without notice. Renesas grants you permission to use these resources only for development of an application that uses Renesas products. Other reproduction or use of these resources is strictly prohibited. No license is granted to any other Renesas intellectual property or to any third party intellectual property. Renesas disclaims responsibility for, and you will fully indemnify Renesas and its representatives against, any claims, damages, costs, losses, or liabilities arising out of your use of these resources. Renesas' products are provided only subject to Renesas' Terms and Conditions of Sale or other applicable terms agreed to in writing. No use of any Renesas resources expands or otherwise alters any applicable warranties or warranty disclaimers for these products.

(Rev.1.0 Mar 2020)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan
www.renesas.com

Contact Information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit:
www.renesas.com/contact/

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.

© 2020 Renesas Electronics Corporation. All rights reserved.