Output Compare TPU Function (OC)

By Sharon Darley

1 Functional Overview

The output compare (OC) function can generate a single output transition, a single pulse, or a continuous 50% duty cycle pulse train upon receiving a link from another channel. The first two actions require the CPU to initiate each output edge or pulse. The third action generates a continuous square wave without CPU intervention. OC can also be used to read the most recent TCR1 and TCR2 values.

2 Detailed Description

Depending on the state of the host service request bits, the OC function performs in one of two modes:

- If the host service request bits are %01 (host-initiated pulse mode), OC updates the parameter RAM word locations $YFFFEC and $YFFFEE with the most recent TCR1 and TCR2 values, respectively. If the host sequence bits are %1x, it requests an interrupt and completes execution. However, if the host sequence bits are %0x, it immediately forces the initial pin level to the state specified in the CHANNEL_CONTROL pin state control (PSC) field (if a state is specified). It then generates an output transition at a programmable delay from a user-specified time and requests an interrupt.
- If the host service request bits are %11 (continuous pulse mode), then when OC receives a link, it calculates the parameter OFFSET by multiplying the parameter RATIO by the contents of the parameter pointed to by REF_ADDR2. It then generates a 50% duty-cycle continuous square wave without CPU intervention. Each high/low time is equal to the calculated OFFSET. Each new link received causes a new OFFSET to be calculated. The new OFFSET is used at the subsequent transition in the generation of the square wave.

The two modes are described in greater detail in the following paragraphs.

2.1 Host-Initiated Pulse Mode

In host-initiated pulse mode, the CPU initiates a single transition: rising, falling, or a toggle of the previous state. The CPU can force an immediate output, generating a pulse whose duration is a programmable delay time. The CPU can also choose not to force an immediate output, causing the output transition to occur a programmable delay time from a reference time. Both of these options are shown in **Figure 1**. OFFSET contains this programmable delay. It is added to the reference time pointed to by REF_ADDR1 to form the output transition time. REF_ADDR1 can reference the most recent TCR1 or TCR2 time value, the time value of the last transition in REF_TIME, or a similar reference associated with another channel. This last capability allows the OC function to synchronize to an event on another channel. In addition, in this mode, the most recent TCR1 and TCR2 values are copied into the parameter RAM locations $EC and $EE.

The state of the pin state control (PSC) field in the channel control register determines whether or not the output is immediate. The PSC field can be set to force an output state immediately when the channel is enabled or it can be set to leave the pin at its previous state until a match occurs.

Freescale Semiconductor, Inc.

© Freescale Semiconductor, Inc., 2004. All rights reserved.

freescale™ semiconductor

© MOTOROLA INC, 1997

For More Information On This Product,
Go to: www.freescale.com

2.1.1 Immediate Output Selected

Immediate output is selected by setting the PSC field to force an output state immediately when the channel is enabled and by pointing REF_ADDR1 to an address that contains the most recent TCR1 or TCR2 time value. Immediately upon channel initialization, the TPU forces the channel pin to the state specified by the PSC field. The TPU waits the number of clock counts specified in OFFSET and then forces the channel pin to the state specified by the pin action control (PAC) field. For an example of a pulse generated by selecting immediate output, see **Figure 1a**. In this example, the previous pin state was low. OFFSET is a value less than $8000, and REF_ADDR1 points to the address containing the current TCR value. The PSC field is set to force the pin high, and the PAC field is set to force the pin low. As soon as the channel is enabled, the pin is forced high. Then, after the number of TCR clock counts specified in OFFSET has passed, the pin is forced low. Thus, a single output pulse is generated.

2.1.2 Immediate Output Not Selected

If the PSC field is set to “do not force any state,” the OC function generates a single transition instead of a pulse. For an example of a single transition generated, see **Figure 1b**. In this example, the previous pin state was low. OFFSET is a value less than $8000, and REF_ADDR1 points to either the most recent TCR time value or a reference value such as REF_TIME or a similar value on another channel. The PSC field is set to “do not force any state,” and the PAC field is set to force the pin high. Thus, in this example the previous pin state was low, and the pin does not go high until (REF_ADDR1) + OFFSET clock counts after TCR time (REF_ADDR1).

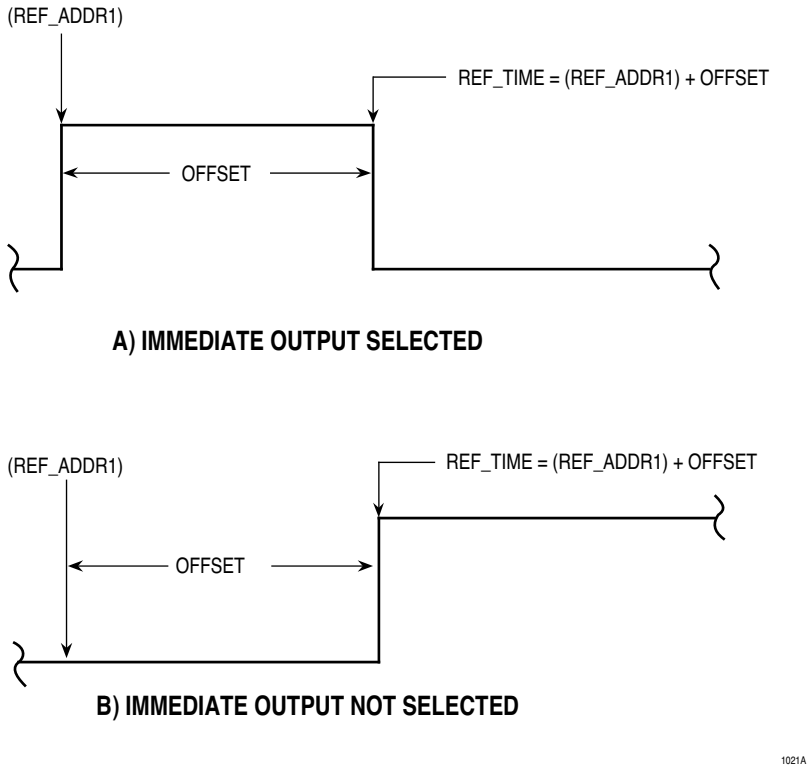


Figure 1 Host-Initiated Pulse Mode Diagram

Freescale Semiconductor, Inc.

For More Information On This Product,
Go to: www.freescale.com

2.2 Continuous Pulse Mode

In continuous mode, the OC function generates a continuous square wave with a 50% duty cycle after receiving a link service request. It calculates the value of the pulse width at each link. At initialization, the TBS field determines the time base of the channel, and the PSC field determines the initial level. The PSC should usually be set to 11 (do not force any state), unless two or more channels executing continuous pulse mode are required to be synchronized.

In this case, both PSC fields should be set to the same value (01 to force pin high, or 10 to force pin low) to produce two signals in phase, or set to opposite values to produce two signals 180 degrees out of phase. The PAC fields must also be set to appropriate values to keep the two channels synchronized. In a channel that has a PSC value set to 01 (force pin high), the PAC field should be set to 001 (force high on match). For a second channel to produce a complementary output, the PSC should be set to 10 (force pin low) and the PAC field to 010 (force low on match).

Continuous pulse generation does not begin until the channel receives the **first** link after initialization. When the first link is received, this function places the value pointed to by REF_ADDR into both the event register and the REF_TIME parameter. Each link, including the first, computes a new value for OFFSET as

$$OFFSET = (REF\_ADDR2) * RATIO$$

where () indicates the value pointed to by the address specified. OFFSET must be less than $8000 for proper operation.

When a match event occurs, OFFSET is added to the value in REF_TIME to form the next pulse edge, and this value replaces the value in REF_TIME. On all links subsequent to the first link, the value in REF_TIME is replaced with the value pointed to by REF_ADDR1. REF_ADDR and REF_ADDR1 can point to either a value updated by the linking channel or to the most recent match event value. This capability to reference parameters in another channel allows the OC function to be synchronized to a channel performing either a PPWA or ITC function. This allows the user to perform synchronized frequency multiplication or division of an input signal. **Figure 2** clarifies this capability.

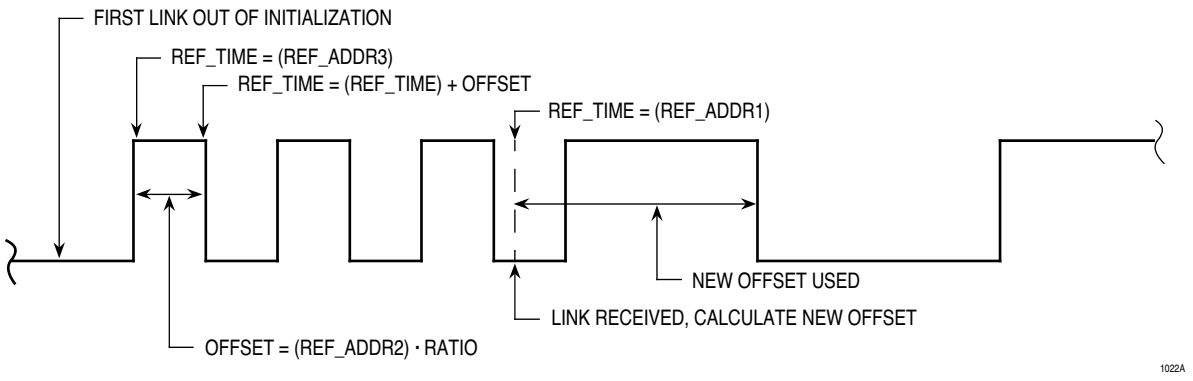


Figure 2 Continuous Pulse Mode Diagram

Each time a match event occurs, the pin state changes, a new REF_TIME is calculated by adding the parameters OFFSET and REF_TIME, and an interrupt request is issued to the host. As shown in **Figure 2**, the first rising edge occurs after the first link following initialization at the reference time pointed to by REF_ADDR. The pulse width is equal to the value in OFFSET, which the TPU calculates as (REF_ADDR2) * RATIO. OC then continuously generates this pulse in a 50% duty cycle waveform until the next link is received. When the new link is received, the TPU re-calculates OFFSET, using the current values in RATIO and the parameter pointed to by REF_ADDR2. The new pulse period then takes effect at the next rising edge.

Freescale Semiconductor, Inc.

If the OFFSET is less than or equal to zero, initialization for this mode is performed without intervention by the CPU, and the channel is configured as specified in CHANNEL_CONTROL. The pin remains in the state specified by CHANNEL_CONTROL until the first link service request is received. At this point operation continues as described above. Clearing OFFSET to zero can be done by action of a PPWA function on another channel that is providing the input values to which this OC function is synchronized, if the input signal slows to the extent that the input is presumed to be stopped.

The maximum usable value for any comparison with a TCR in the TPU is limited to $8000 more than the TCR value at the time that the match register is loaded. The TPU sees any value greater than TCR + $8000 as a time in the past and immediately generates a match. **Figure 3** illustrates this principle, assuming the TCR is at $2800 when the offset calculation is made, and the comparison value is written into the match register. Any value greater than $A800 causes a greater-than comparison, resulting in an immediate match event.

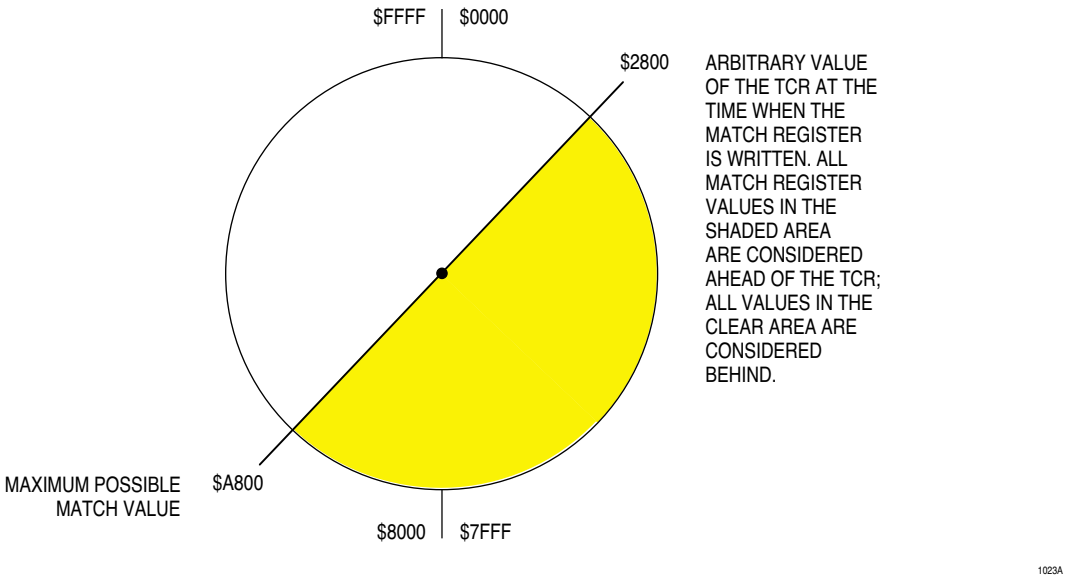


Figure 3 Offset Range

3 Function Code Size

Total TPU function code size determines what combination of functions can fit into a given ROM or emulation memory microcode space. OC function code size is:

31 μ instructions + 8 entries = **39 long words**

4 Function Parameters

This section provides detailed descriptions of output compare function parameters stored in channel parameter RAM. **Figure 4** shows TPU parameter RAM address mapping. **Figure 5** shows the parameter RAM assignment used by the OC function. In the diagrams, Y = M111, where M is the value of the module mapping bit (MM) in the system integration module configuration register (Y = $7 or $F).

Freescale Semiconductor, Inc.

Channel Number	Base Address	Parameter Address							
		0	1	2	3	4	5	6	7
0	$YFFF##	00	02	04	06	08	0A	—	—
1	$YFFF##	10	12	14	16	18	1A	—	—
2	$YFFF##	20	22	24	26	28	2A	—	—
3	$YFFF##	30	32	34	36	38	3A	—	—
4	$YFFF##	40	42	44	46	48	4A	—	—
5	$YFFF##	50	52	54	56	58	5A	—	—
6	$YFFF##	60	62	64	66	68	6A	—	—
7	$YFFF##	70	72	74	76	78	7A	—	—
8	$YFFF##	80	82	84	86	88	8A	—	—
9	$YFFF##	90	92	94	96	98	9A	—	—
10	$YFFF##	A0	A2	A4	A6	A8	AA	—	—
11	$YFFF##	B0	B2	B4	B6	B8	BA	—	—
12	$YFFF##	C0	C2	C4	C6	C8	CA	—	—
13	$YFFF##	D0	D2	D4	D6	D8	DA	—	—
14	$YFFF##	E0	E2	E4	E6	E8	EA	EC	EE
15	$YFFF##	F0	F2	F4	F6	F8	FA	FC	FE

— = Not Implemented (reads as $00)

Figure 4 TPU Channel Parameter RAM CPU Address Map

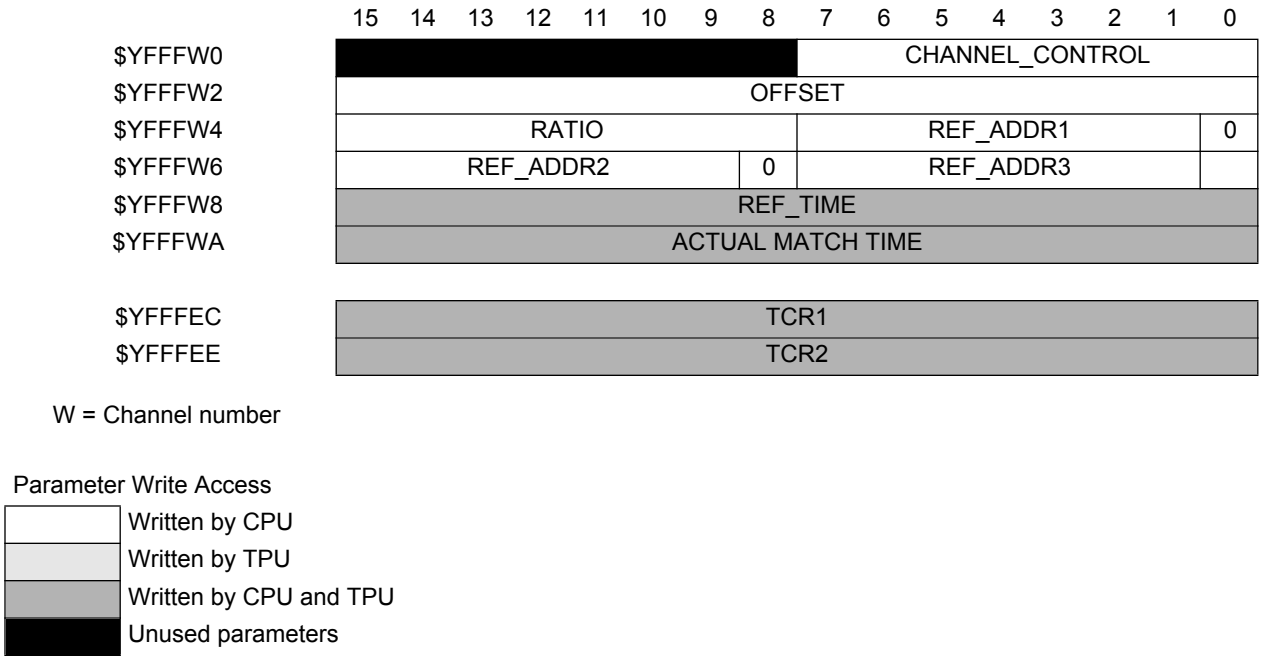


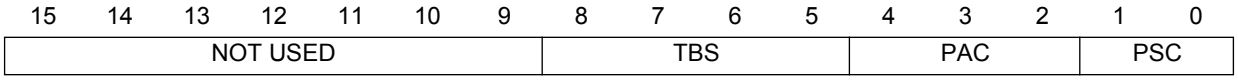
Figure 5 Parameter RAM Assignment

Freescale Semiconductor, Inc.

4.1 CHANNEL_CONTROL

CHANNEL_CONTROL contains the channel latch controls and configures the PSC, PAC, and TBS fields. The PSC field determines the initial pin level. For an output channel, the PAC field specifies the pin logic response when the match event occurs. The TBS field configures a channel pin as input or output and configures the time base for match or capture events.

The OC function is normally configured only as an output channel. It is not intended for channels configured for input. The function operates as described concerning generation of interrupts and use of parameters, etc., when programmed as an input channel; however, the CPU cannot discern whether a received interrupt is due to an input capture event or a match event.



The following table defines the allowable data for this parameter.

Table 1 OC CHANNEL_CONTROL Options

TBS	PAC	PSC	Action	
8 7 6 5	4 3 2	1 0	Input	Output
		0 0	—	Force Pin as Specified by PAC Latches
		0 1	—	Force Pin High
		1 0	—	Force Pin Low
		1 1	—	Do Not Force Any State
	0 0 0		Do Not Detect Transition	Do Not Change Pin State on Match
	0 0 1		Detect Rising Edge	High on Match
	0 1 0		Detect Falling Edge	Low on Match
	0 1 1		Detect Either Edge	Toggle on Match
	1 x x		Do Not Change PAC	Do Not Change PAC
0 1 x x				**Output Channel**
0 1 0 0			—	Capture TCR1, Match TCR1
0 1 0 1			—	Capture TCR1, Match TCR2
0 1 1 0			—	Capture TCR2, Match TCR1
0 1 1 1			—	Capture TCR2, Match TCR2
1 x x x			Do Not Change TBS	Do Not Change TBS

4.2 OFFSET

OFFSET contains the count value from a reference time value that indicates when the next match event will occur. In host-initiated pulse mode, the CPU writes this parameter. In continuous pulse mode, OFFSET determines the pulse width and is calculated by the TPU when a link is received as

$$OFFSET = (REF\_ADDR2) * RATIO$$

where () indicates the data pointed to by the specified address. This parameter must always be less than $8000.

4.3 RATIO

RATIO is an 8-bit fractional parameter (0.xxxxxxxx) that scales the contents indicated by REF_ADDR2 in the calculation of the OFFSET parameter when operating in the continuous pulse mode. The actual fraction used by the TPU is RATIO/$FF. Thus, a value of $80 in RATIO represents the fraction $80/$FF, which is approximately equal to 1/2. Similarly, a value of $FF in RATIO represents a ratio of 1/1.

Freescale Semiconductor, Inc.

4.4 REF_ADDR1

REF_ADDR1 is a pointer into the parameter RAM map. When REF_ADDR1 is used as a pointer, the upper 16 bits of the 24-bit address are assumed to be $YFFF, where Y can be %0111 or %1111 (the default is %1111). For example, if the value in REF_ADDR1 is $3A, then REF_ADDR1 points to the fifth parameter location for channel 3. In host-initiated pulse mode, REF_ADDR1 points to a reference value that is added to OFFSET, forming a new match. In continuous pulse mode, REF_ADDR1 points to a synchronization reference value used whenever a link is received subsequent to the first link after initialization. REF_ADDR1 can point to either a value updated by the linking channel or to the most recent match event value. It should not point to a static constant.

4.5 REF_ADDR2

REF_ADDR2 is also a pointer into the parameter RAM map. It is used only in continuous mode. It points to a reference value that is scaled by the RATIO parameter to form a new OFFSET value whenever a link is received.

4.6 REF_ADDR3

REF_ADDR3 is a pointer into the parameter RAM map that points to a synchronization reference value used only once in continuous pulse mode when the first link service request is serviced after initialization.

4.7 REF_TIME

REF_TIME contains the TCR value when the next match event will occur, or the TCR value of the most recent match event.

4.8 ACTUAL_MATCH_TIME

ACTUAL_MATCH_TIME contains the TCR value captured at the time of the last match event. This may be used by the host CPU to estimate time available before the next match and to confirm that a change in offset was effected as desired. The TBS field determines which TCR is captured and placed in this parameter. This parameter is used only in the host-initiated pulse mode.

4.8.1 TCR1 and TCR2

TCR1 and TCR2 are word locations $EC and $EE in the parameter RAM map. When the HOST_MATCH state (S1) is executed, the most recent TCR1 value updates word location $EC, and the most recent TCR2 value updates word location $EE.

5 Host Interface to Function

This section provides information concerning the TPU host interface to the OC function. **Figure 6** is a TPU address map. Detailed TPU register diagrams follow the figure. In the diagrams, Y = M111, where M is the value of the module mapping bit (MM) in the system integration module configuration register (Y = $7 or $F).

Freescale Semiconductor, Inc.

Address	15 8	7 0
$YFFE00	TPU MODULE CONFIGURATION REGISTER (TPUMCR)	
$YFFE02	TEST CONFIGURATION REGISTER (TCR)	
$YFFE04	DEVELOPMENT SUPPORT CONTROL REGISTER (DSCR)	
$YFFE06	DEVELOPMENT SUPPORT STATUS REGISTER (DSSR)	
$YFFE08	TPU INTERRUPT CONFIGURATION REGISTER (TICR)	
$YFFE0A	CHANNEL INTERRUPT ENABLE REGISTER (CIER)	
$YFFE0C	CHANNEL FUNCTION SELECTION REGISTER 0 (CFSR0)	
$YFFE0E	CHANNEL FUNCTION SELECTION REGISTER 1 (CFSR1)	
$YFFE10	CHANNEL FUNCTION SELECTION REGISTER 2 (CFSR2)	
$YFFE12	CHANNEL FUNCTION SELECTION REGISTER 3 (CFSR3)	
$YFFE14	HOST SEQUENCE REGISTER 0 (HSQR0)	
$YFFE16	HOST SEQUENCE REGISTER 1 (HSQR1)	
$YFFE18	HOST SERVICE REQUEST REGISTER 0 (HSRR0)	
$YFFE1A	HOST SERVICE REQUEST REGISTER 1 (HSRR1)	
$YFFE1C	CHANNEL PRIORITY REGISTER 0 (CPR0)	
$YFFE1E	CHANNEL PRIORITY REGISTER 1 (CPR1)	
$YFFE20	CHANNEL INTERRUPT STATUS REGISTER (CISR)	
$YFFE22	LINK REGISTER (LR)	
$YFFE24	SERVICE GRANT LATCH REGISTER (SGLR)	
$YFFE26	DECODED CHANNEL NUMBER REGISTER (DCNR)	

Figure 6 TPU Address Map

CIER — Channel Interrupt Enable Register **$YFFE0A**

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

CH	Interrupt Enable
0	Channel interrupts disabled
1	Channel interrupts enabled

CFSR[0:3] — Channel Function Select Registers **$YFFE0C – $YFFE12**

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CFS (CH 15, 11, 7, 3)				CFS (CH 14, 10, 6, 2)				CFS (CH 13, 9, 5, 1)				CFS (CH 12, 8, 4, 0)			

CFS[4:0] — OC Function Number (Assigned during microcode assembly)

Freescale Semiconductor, Inc.

HSQR[0:1] — Host Sequence Registers **$YFFE14 – $YFFE16**

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

CH	Action Taken
0x	All code associated with host pulse mode is executed.
1x	During execution of the host pulse mode, only the code that updates word locations $EC and $EE with TCR1 and TCR2, respectively, is executed

HSRR[1:0] — Host Service Request Registers **$YFFE18 – $YFFE1A**

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

CH	Initialization
00	No Host Service (Reset Condition)
01	Host-initiated pulse mode
10	Reserved
11	Initialize, continuous pulse mode

CPR[1:0] — Channel Priority Registers **$YFFE1C – $YFFE1E**

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

CH	Channel Priority
00	Disabled
01	Low
10	Middle
11	High

CISR — Channel Interrupt Status Register **$YFFE20**

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

CH	Interrupt Status
0	Channel interrupt not asserted
1	Channel interrupt asserted

Freescale Semiconductor, Inc.

6 Function Configuration

The CPU configures the OC function as follows:

1. Writes parameters CHANNEL_CONTROL, REF_ADDR1, and other parameters, depending on the mode of operation, to RAM.
2. Writes host sequence bit 1 according to mode of operation.
3. Issues an HSR %01 for initiation of host-initiated pulse mode, or %11 for initialization of continuous pulse mode. (The host-initiated pulse mode has no separate initialization state. Initialization occurs each time the host-initiated pulse mode is executed.)
4. Enables channel servicing by assigning a high, middle, or low priority.

The CPU should monitor the HSR register (or the channel interrupt) until the TPU clears the service request to 00 before changing parameters or issuing a new service request to this channel.

In the host-initiated pulse mode, the CPU configures CHANNEL_CONTROL, OFFSET, and REF_ADDR1 and initiates the match event to be scheduled by issuing an HSR%01. When the HSR is serviced, locations $EC and $EE are updated with the current values of TCR1 and TCR2, respectively, and the scheduled match event is loaded into the event register. When the scheduled match event occurs, the actual match time is loaded into ACTUAL_MATCH_TIME and an interrupt is requested. If host sequence bit 1 is set, only the update of locations $EC and $EE with TCR1 and TCR2, respectively, occurs and the function terminates after requesting an interrupt.

In continuous pulse mode, the CPU configures this mode by providing parameters CHANNEL_CONTROL, RATIO, REF_ADDR1, REF_ADDR2, and REF_ADDR3 before issuing an HSR %11. After each scheduled match event, an interrupt is issued to the host.

7 Performance and Use of Function

7.1 Performance

Like all TPU functions, OC function performance in an application is to some extent dependent upon the service time (latency) of other active TPU channels. This is due to the operational nature of the scheduler. The more TPU channels are active, the more performance decreases. However, worst-case latency in any TPU application can be closely estimated. To analyze the performance of an application that appears to approach the limits of the TPU, use the guidelines given in the TPU reference manual and the information in the OC state timing table below.

Table 2 OC State Timing

State Number and Name	Max. CPU Clock Cycles	RAM Accesses by TPU
S1 Init	6	2
S2 Offset_Cal (first link after Init)	40	7
S3 Offset_Compare 0 < OFFSET ≤ $7FFF $7FFF < OFFSET ≤ 0	 10 14	 4 4
S4 Host Compare HSB = 0X HSB = 1X	18 10	5 2
S5 Ref_Time_Compare	4	2

7.2 Changing Mode

The host sequence bits are used to select OC function operating mode. Change host sequence bit values only when the function is stopped or disabled (channel priority bits = %00). Disabling the channel before changing mode avoids conditions that cause indeterminate operation.

Freescale Semiconductor, Inc.

8 Examples

The following examples give an indication of the capabilities of the OC function. Each example includes a description of the example, a diagram of the initial parameter RAM content, initial control bit settings, and a diagram of the output waveform.

8.1 Example A

8.1.1 Description

This example uses the OC function in the continuous mode to output a continuous square wave after a certain number of input pulses have been counted. A pulse-width modulated square wave is initialized on channel 0. Channel 0 is physically connected to channel 1, which is initialized with the input transition counter (ITC) function. The ITC function is initialized for single-shot operation with links. This means that it counts a certain number of pulses from channel 0, then generates a link and ceases channel activity. Channel 2 is set up to run the OC function. Channel 2 receives the link from channel 1 and then generates a continuous 50% duty cycle square wave with an output frequency that is twice the input frequency.

8.1.2 Initialization

The PWM channel is set up to generate a 50% duty cycle square wave. It has a pulse hightime of $2000 TCR clock counts and a pulse period of $4000 TCR clock counts.

Load parameter RAM as shown.

Table 3 PWM Channel Parameter RAM

$YFFF00	0	0	0	0	0	0	0	0	1	0	0	1	0	0	1	0
$YFFF02	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
$YFFF04	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
$YFFF06	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
$YFFF08	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
$YFFF0A	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

The ITC function is initialized in the single shot mode with links by setting the host sequence field bits to %10. CHANNEL_CONTROL is set to detect falling edges. START_LINK_CHANNEL is set to two, and LINK_CHANNEL_COUNT is set to one. The BANK_ADDRESS increment is not needed, so it points to an unimplemented location in the parameter RAM. MAX_COUNT is set to $10. Thus, after $10 falling edges are counted, ITC generates a link to the OC function on channel 2.

Load parameter RAM as shown.

Table 4 ITC Channel Parameter RAM

$YFFF10	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1
$YFFF12	0	0	1	0	0	0	0	1	0	0	0	0	1	1	1	0
$YFFF14	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
$YFFF16	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
$YFFF18	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
$YFFF1A	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Freescale Semiconductor, Inc.

The OC function is initialized in the continuous pulse mode by setting the host service request bits to %11. After it receives a link from channel 1, the OC function generates a continuous 50% duty cycle square wave with a frequency equal to twice that of the PWM square wave from channel 0. The parameters are initialized as follows:

CHANNEL_CONTROL is configured as an output channel. It is set to capture and match TCR1, to force the initial pin level low, and to force the pin level low on a match.

RATIO is set to $80. This corresponds to the fraction $80/$FF, or approximately 1/2. The TPU multiplies 1/2 by the value pointed to by REF_ADDR2. In this case, REF_ADDR2 points to an address which contains the value $2000. Thus, the output pulse will have a pulse width of (1/2) ∗ $2000 = $1000.

REF_ADDR1 points to a reference value used whenever a link is received that is not the first link after initialization. In this example, it is a don't care value since only one link is received after initialization.

REF_ADDR2 points to a value that is scaled by RATIO to form the value OFFSET whenever a link is received. In this example, it points to an unused parameter RAM location that contains the value $2000.

REF_ADDR3 points to a reference value used when the first link is received after initialization. In this example, it points to the ITC parameter FINAL_TRANS_TIME.

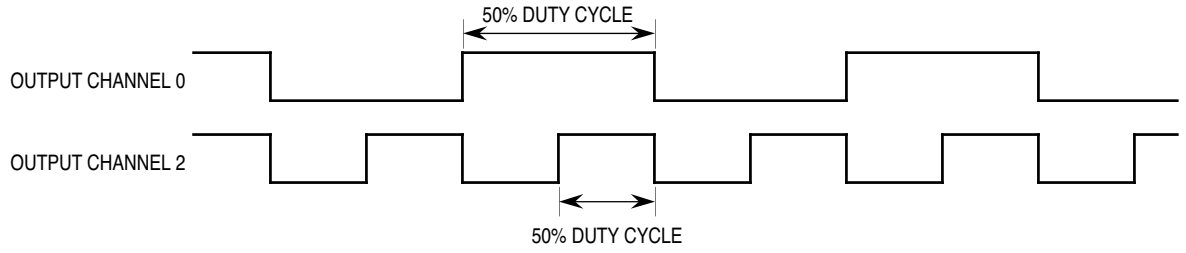
Load the parameter RAM location $YFFFFE with the value $2000. This location is chosen because channel 15 is not used in this example.

Load parameter RAM as shown.

Table 5 OC Channel Parameter RAM

$YFFF20	0	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0
$YFFF22	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
$YFFF24	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
$YFFF26	1	1	1	1	1	1	1	0	0	0	0	1	1	0	0	0
$YFFF28	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
$YFFF2A	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

8.1.3 Output Waveforms



TPU OC 50% DUTY TIM

Freescale Semiconductor, Inc.

8.2 Example B

8.2.1 Description

This example uses OC with the PPWA function to divide the input frequency to the PPWA channel by four. The PPWA function repeatedly accumulates sixteen input periods and then generates a link to the OC function. The OC function scales the accumulated period and then generates the scaled output waveform. The PPWA function is on channel 0, and the OC function is on channel 3. Both waveforms produce a 50% duty cycle.

8.2.2 Initialization

The PPWA function is assigned to channel 0 so that it has higher priority than the OC function. It is set up for mode 1, to accumulate 16-bit periods and generate links. The PPWA parameters are initialized as follows:

START_LINK_CHANNEL is set to 3, the number of the channel running the OC function.

LINK_CHANNEL_COUNT is set to 1, since there is one channel in the link block.

CHANNEL_CONTROL is set to $310B. This sets the TBS field to capture and match TCR1, the PAC field to detect falling edge for period accumulate, and the PSC field to “do not force any state”.

MAX_CNT is the number of periods per accumulation. In this case it is sixteen.

ACCUM_RATE determines how often ACCUM is updated during accumulations. Since a running accumulation is not desired, this parameter is set to the slowest rate possible, $FF.

The host sequence field bits are set to %01, to accumulate 16-bit periods and generate links. The host service request bits are set to %10, for initialization.

Load parameter RAM as shown.

Table 6 PPWA Channel Parameter RAM

$YFFF00	0	0	1	1	0	0	0	1	0	0	0	0	1	0	1	1
$YFFF02	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
$YFFF04	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
$YFFF06	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
$YFFF08	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0
$YFFF0A	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

The OC function is set up on channel 3 in continuous pulse mode. Its parameters are initialized as follows:

CHANNEL_CONTROL is set to $008A. This captures and matches TCR1, forces the pin low on a match, and forces the initial pin level low.

RATIO is an 8-bit fractional number between $00 and $FF used to scale the value indicated by REF_ADDR2 to form the output pulse hightime. Here, REF_ADDR2 points to PPWA_LW. Thus, for the input PPWA channel, PPWA_LW represents a period accumulation, while for the output OC channel, it represents the unscaled output pulse hightime. The following equation relates the input period accumulation in PPWA_LW, the desired output period, and RATIO:

Freescale Semiconductor, Inc.

$$T_o = 2 * A_i * (RATIO/255) \text{ seconds}$$

In this equation, A_i is the accumulation value in PPWA_LW multiplied by the resolution of the timer for the input channel, and T_o is the period of the output pulse multiplied by the resolution of the timer for the output channel. The factor of 2 is included since the OC channel scales the value in PPWA_LW to be the output pulse hightime instead of the output pulse period. Thus, the scaled output pulse hightime must be multiplied by two in order to calculate the output pulse period.

Solving this equation for RATIO yields the following:

$$RATIO = (T_o / A_i) * (255/2)$$

In this example, the desired ratio of the output period to the input period is 4/1. Since the accumulated value in PPWA_LB represents sixteen input periods, $T_o/A_i = 4/1 * 1/16 = 1/4$. Thus, $T_o/A_i * 255/2 = 1/4 * 255/2 = \sim 32 = \sim \$20$.

REF_ADDR1 points to a synchronization reference value used whenever a link is received that is not the first link after initialization. Here, it points to LAST_ACCUM.

REF_ADDR2 points to a reference value used in calculating the output pulse hightime. Here, it points to PPWA_LW.

REF_ADDR3 points to a synchronization reference value used when the first link service request is serviced after initialization. Here, it points to LAST_ACCUM.

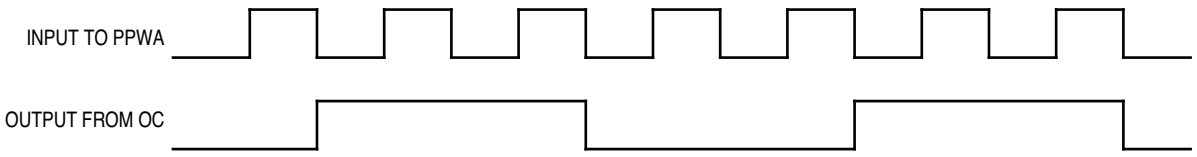
The host sequence field bits are initialized to%00, matches and pulses scheduled. The host service request bits are set to %11, initialization for the continuous mode.

Load parameter RAM as shown.

Table 7 OC Channel Parameter RAM

$YFFF30	0	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0
$YFFF32	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
$YFFF34	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0
$YFFF36	0	0	0	0	1	0	1	0	0	0	0	0	0	1	0	0
$YFFF38	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
$YFFF3A	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

8.2.3 Output Waveforms



TPU OC/PPWA TIM

Freescale Semiconductor, Inc.

8.3 Example C

8.3.1 Description

This example uses OC in host-initiated pulse mode to generate an output pulse similar to that shown in **Figure 1a**. The duration of the hightime is approximately 16.4 ms. Channel 0 is assigned the OC function.

CHANNEL_CONTROL is initialized to force the initial pin state high and to force the pin state low when a match occurs. The previous pin level is assumed to be low.

OFFSET is initialized to $2000 TCR clock counts. This is the duration of the pulse hightime.

REF_ADDR1 is initialized to $EC. This points it to the address $YFFFEC, which contains the most recent value of TCR1.

The host service request field should be set to %01, host-initiated pulse.

8.3.2 Initialization

Load parameter RAM as shown.

Table 8 OC Channel Parameter RAM

$YFFF00	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	1
$YFFF02	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
$YFFF04	0	0	0	0	0	0	0	0	1	1	1	0	1	1	0	0
$YFFF06	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
$YFFF08	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
$YFFF0A	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

8.4 Example D

8.4.1 Description

This example uses OC in host-initiated pulse mode to generate a rising transition similar to that shown in **Figure 1b**. The delay between the host service request and the rising edge is approximately 16.4 ms. Channel 0 is assigned the OC function.

CHANNEL_CONTROL is initialized as follows:

PSC field value is set to %11, so that no initial pin state is forced.

PAC field value is set to %001, so that the pin is driven high when a match occurs.

The previous pin level is assumed to be low.

OFFSET is initialized to $2000 TCR clock counts. This will be the duration of the delay to the transition.

REF_ADDR1 is initialized to $EC. This points it to the address $YFFFEC, which contains the most recent value of TCR1.

The host service request field should be set to %01, for host-initiated pulse mode.

8.4.2 Initialization

Load parameter RAM as shown.

Freescale Semiconductor, Inc.

Table 9 OC Channel Parameter RAM

$YFFF00	0	0	0	0	0	0	0	0	1	0	0	0	0	1	1	1
$YFFF02	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
$YFFF04	0	0	0	0	0	0	0	0	1	1	1	0	1	1	0	0
$YFFF06	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
$YFFF08	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
$YFFF0A	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

9 Function Algorithm

The following description is provided as a guide only, to aid understanding of the function. The exact sequence of operations in microcode may be different from that shown, in order to optimize speed and code size. TPU microcode source listings for all functions in the TPU function library can be downloaded from the Freescale Freeware bulletin board. Refer to *Using the TPU Function Library and TPU Emulation Mode* (TPUPN00/D) for detailed instructions on downloading and compiling microcode.

The output compare function consists of five states, described below. The host-initiated pulse mode of operation uses states 1 and 2, and the continuous pulse mode of operation uses states 3 through 5.

For clarity, reference is made to internal channel flags in the following descriptions. These internal TPU control bits are not available to the user. The channel flags, flag1 and flag0, indicate the mode of operation or direct the function to execute a routine. Flag0 is set during initialization of the continuous pulse mode and defines the current mode of operation. When set, the continuous pulse mode is operating. When clear, the host-initiated pulse mode is operating. Flag1 is set during initialization of the continuous pulse mode to implement a one-time execution after initialization of a microcode sequence in *Offset_Cal*.

9.1 State 1: *HOST_MATCH* (Host-Initiated Pulse Mode)

This state is entered to initialize the host-initiated pulse mode as a result of an HSR %01. Parameter RAM word locations $EC and $EE are updated with the most recent TCR1 and TCR2 values, respectively.

If host sequence bit 1 is one, an interrupt is asserted to the CPU and execution ends. If host sequence bit 1 is zero, match and input transition service requests are enabled, and the channel is configured to generate a match event at the new time. A new match time value is calculated, and the event register and REF_TIME are updated with the new desired match value (OFFSET plus (REF_ADDR1)). The channel is programmed to specify the pin direction, the time base for comparison, the initial pin state, and the desired pin action on a match event as specified by CHANNEL_CONTROL. Flag0 and flag1 are cleared.

Condition: HSR1, HSR0, M/TSR, LSR, Pin, Flag0 = 11xxxx
Match Enable: Don't Care

Configure channel latches via parameter CHANNEL_CONTROL
Enable M/TSR
Assert flag0, assert flag1
ACTUAL_MATCH_TIME = ERT
Negate MRL, TDL, LSR
Assert interrupt request

Freescale Semiconductor, Inc.

9.2 State 2: Ref_Time_Match

In the host-initiated pulse mode, this state is entered as a result of a match event with flag0 equal to zero. An interrupt is generated, signaling the CPU that the previously configured match occurred and that a new match may be configured. The time of the match event is stored in ACTUAL_MATCH_TIME.

Condition: HSR1, HSR0, M/TSR, LSR, Pin, Flag0 = 00x1x1
Match Enable: Don't Care

```
Configure channel latches via parameter CHANNEL_CONTROL
REF_TIME replaced by (REF_ADDR1)
If (channel flag1 = 1) then {
                                /* if first link after initialization: get a reference time and put in ERT */
    REF_TIME replaced by (REF_ADDR3)
    Clear flag1
    Generate a match at (REF_ADDR3)
}                                                    /* set up for first match after initialization */
OFFSET replaced by (REF_ADDR2) * RATIO
Negate LSR
```

9.3 State 3: *Init* (Continuous Pulse Mode)

This state is entered as a result of an HSR %11. The channel is configured as specified by CHANNEL_CONTROL. When asserted, flag0 indicates the continuous pulse mode.

Condition: HSR1, HSR0, M/TSR, LSR, Pin, Flag0 = 0010x1
Match Enable: Don't Care

```
Set PAC = toggle
If (OFFSET ≤ 0) then {                                   /* Do state 1 Init */
    Configure channel latches via parameter CHANNEL_CONTROL
    Enable M/TSR
    Assert flag1, assert flag0
    ACTUAL_MATCH_TIME = ERT
    Negate MRL, TDL, LSR
    Assert interrupt request
}
else {
    REF_TIME = REF_TIME + OFFSET
    Generate a match at REF_TIME + OFFSET
    Clear flag1, negate MRL
    Assert interrupt request
}
```

9.4 State 4: Offset_Cal

This state is entered as a result of an LSR received in the continuous pulse mode. The channel is configured as specified in CHANNEL_CONTROL. A new value of OFFSET is calculated, and REF_TIME is updated with a value pointed to by REF_ADDR1, unless this is the first execution of this state after execution of *Init*. In the first execution after *Init*, a match event is set up to be generated at the contents pointed to by REF_ADDR3, and REF_TIME is updated with the new match value. Flag1 is negated.

Condition: HSR1, HSR0, M/TSR, LSR, Pin, Flag0 = 01xxxx
Match Enable: Disable

```
TCR1 copied to absolute parameter word address $EC
TCR2 copied to absolute parameter word address $EE
```

Freescale Semiconductor, Inc.

```
If (host sequence bit 1 = 1) {
    Assert interrupt request
}
Else {
    Configure channel latches via parameter CHANNEL_CONTROL
    Enable M/TSR
    Clear flag0, clear flag1
    temp = (REF_ADDR1)
    Generate match at temp + OFFSET
    REF_TIME = temp + OFFSET
    Negate MRL
}
```

9.5 State 5: Offset_Match

This state is entered as a result of a match event in the continuous pulse mode. If the OFFSET parameter is less than or equal to zero, *Init* is executed, no new match is configured, and the pin state is configured as specified by CHANNEL_CONTROL. If OFFSET is greater than zero, the channel is configured to generate a match at time value (REF_TIME plus OFFSET). REF_TIME is updated with the time value of the new (next) match event. The pin toggles when the match event occurs.

Condition: HSR1, HSR0, M/TSR, LSR, Pin, Flag0 = 001000
Match Enable: Disable
    Negate MRL, TDL, LSR
    ACTUAL_MATCH_TIME = ERT
    Assert flag0
    Assert interrupt request

The following table shows the OC transitions listing the service request sources and channel conditions from current state to next state. **Figure 7** illustrates the flow of the OC states.

Table 10 OC State Transition Table

Current State	HSR	M/TSR	LSR	Pin	Flag0	Next State
Any State	01	—	—	—	—	S1 *HOST_MATCH* Host-Initiated Pulse Mode
	11	—	—	—	—	S3 *Init* Continuous Pulse Mode
S1 *HOST_MATCH* Host-Initiated Pulse Mode	00	1	0	—	0	S2 *Ref_Time_Match*
S2 *Ref_Time_Match*	01	1	—	—	0	S1 *HOST_MATCH* Host-Initiated Pulse Mode
S3 *Init* Continuous Pulse Mode	00	—	1	—	1	S4 *Offset_Cal*
S4 *Offset_Cal*	00	—	1	—	1	S4 *Offset_Cal*
	00	—	0	—	1	S5 *Offset_Match*
S5 *Offset_Match*	00	—	0	—	1	S5 *Offset_Match*
	00	—	1	—	1	S4 *Offset_Cal*
Unimplemented Conditions	10	—	—	—	—	—
	00	—	1	—	0	—

NOTES:
1. Conditions not specified are “don't care.”
2. HSR = Host service request
   LSR = Link service request
   M/TSR = Either a match or transition (input capture) service request occurred (M/TSR = 1) or neither occurred (M/TSR = 0).

Freescale Semiconductor, Inc.

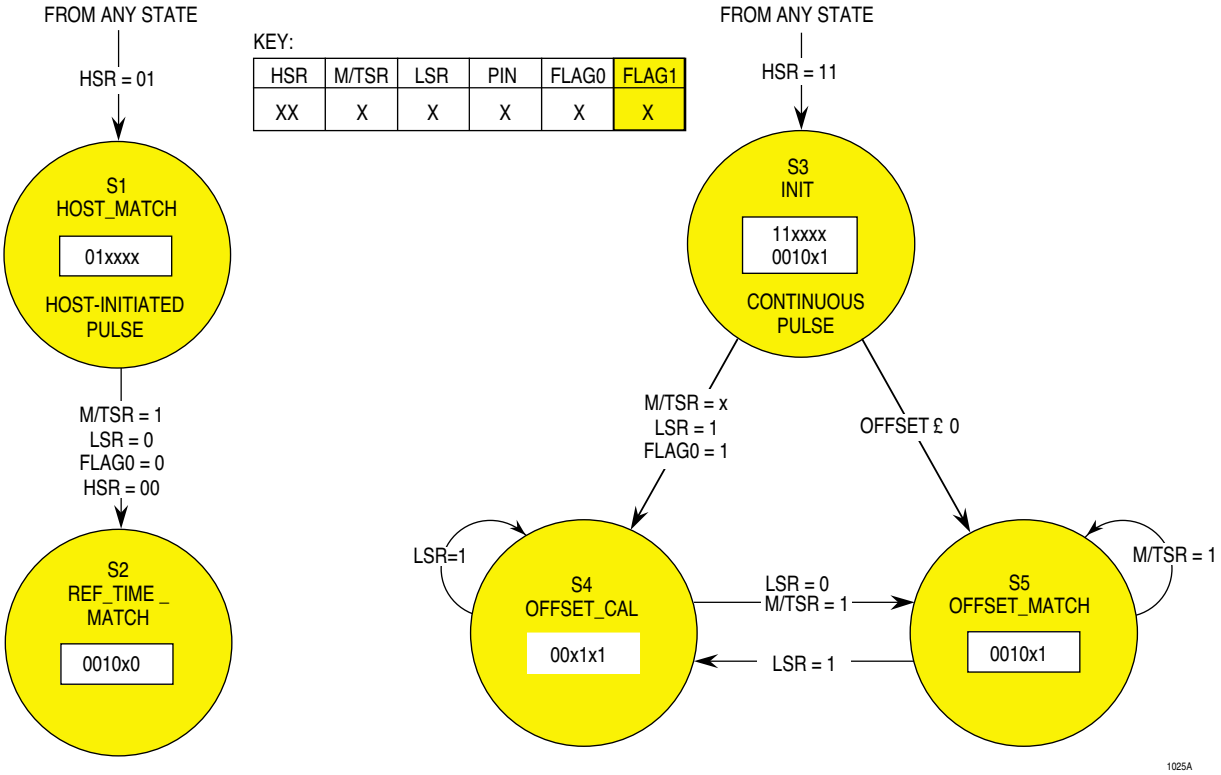


Figure 7 Output Compare State Flowchart

Freescale Semiconductor, Inc.

Freescale Semiconductor, Inc.

How to Reach Us:

Home Page:
www.freescale.com

E-mail:
support@freescale.com

USA/Europe or Locations Not Listed:
Freescale Semiconductor
Technical Information Center, CH370
1300 N. Alma School Road
Chandler, Arizona 85224
+1-800-521-6274 or +1-480-768-2130
support@freescale.com

Europe, Middle East, and Africa:
Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
support@freescale.com

Japan:
Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku,
Tokyo 153-0064
Japan
0120 191014 or +81 3 5437 9125
support.japan@freescale.com

Asia/Pacific:
Freescale Semiconductor Hong Kong Ltd.
Technical Information Center
2 Dai King Street
Tai Po Industrial Estate
Tai Po, N.T., Hong Kong
+800 2666 8080
support.asia@freescale.com

For Literature Requests Only:
Freescale Semiconductor Literature Distribution Center
P.O. Box 5405
Denver, Colorado 80217
1-800-441-2447 or 303-675-2140
Fax: 303-675-2150
LDCForFreescaleSemiconductor@hibbertgroup.com

Information in this document is provided solely to enable system and software implementers to use Freescale Semiconductor products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits or integrated circuits based on the information in this document.
Freescale Semiconductor reserves the right to make changes without further notice to any products herein. Freescale Semiconductor makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale Semiconductor assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. “Typical” parameters which may be provided in Freescale Semiconductor data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including “Typicals” must be validated for each customer application by customer’s technical experts. Freescale Semiconductor does not convey any license under its patent rights nor the rights of others. Freescale Semiconductor products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Freescale Semiconductor product could create a situation where personal injury or death may occur. Should Buyer purchase or use Freescale Semiconductor products for any such unintended or unauthorized application, Buyer shall indemnify and hold Freescale Semiconductor and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Freescale Semiconductor was negligent regarding the design or manufacture of the part.

freescale™
semiconductor