



Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!



## Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832

Email & Skype: info@chipsmall.com Web: www.chipsmall.com

Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China





## 3.3V EEPROM PROGRAMMABLE CLOCK GENERATOR

**IDT5V9885T**

### FEATURES:

- Three internal PLLs
- Internal non-volatile EEPROM
- JTAG and FAST mode I<sup>2</sup>C serial interfaces
- Input Frequency Ranges: 1MHz to 400MHz
- Output Frequency Ranges: 4.9kHz to 500MHz
- Reference Crystal Input with programmable oscillator gain and programmable linear load capacitance
  - Crystal Frequency Range: 8MHz to 50MHz
- Each PLL has an 8-bit pre-scaler and a 12-bit feedback-divider
- 10-bit post-divider blocks
- Fractional Dividers
- Two of the PLLs support Spread Spectrum Generation capability
- I/O Standards:
  - Outputs - 3.3V LVTTTL/LVCMOS, LVPECL, and LVDS
  - Inputs - 3.3V LVTTTL/LVCMOS
- Programmable Slew Rate Control
- Programmable Loop Bandwidth Settings
- Programmable output inversion to reduce bimodal jitter
- Redundant clock inputs with glitchless auto and manual switchover options
- JTAG Boundary Scan
- Individual output enable/disable
- Power-down mode
- 3.3V V<sub>DD</sub>
- Available in TQFP and VFQFPN packages

### DESCRIPTION:

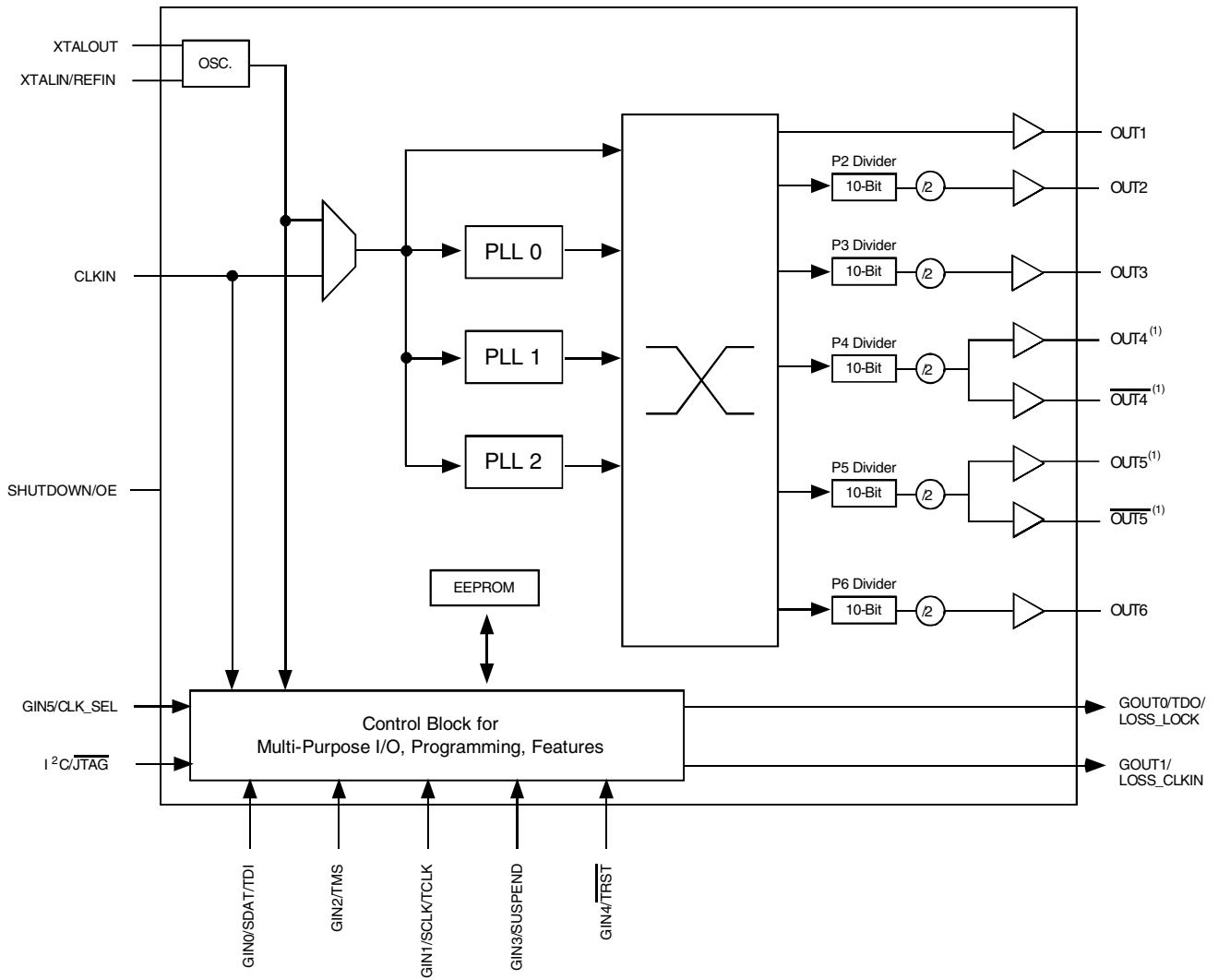
The IDT5V9885T is a programmable clock generator intended for high performance data-communications, telecommunications, consumer, and networking applications. There are three internal PLLs, each individually programmable, allowing for three unique non-integer-related frequencies. The frequencies are generated from a single reference clock. The reference clock can come from one of the two redundant clock inputs. A glitchless automatic or manual switchover function allows any one of the redundant clocks to be selected during normal operation.

The IDT5V9885T can be programmed through the use of the I<sup>2</sup>C or JTAG interfaces. The programming interface enables the device to be programmed when it is in normal operation or what is commonly known as in-system programmable. An internal EEPROM allows the user to save and restore the configuration of the device without having to reprogram it on power-up. JTAG boundary scan is also implemented.

Each of the three PLLs has an 8-bit pre-scaler and a 12-bit feedback divider. This allows the user to generate three unique non-integer-related frequencies. The PLL loop bandwidth is programmable to allow the user to tailor the PLL response to the application. For instance, the user can tune the PLL parameters to minimize jitter generation or to maximize jitter attenuation. Spread spectrum generation and fractional divides are allowed on two of the PLLs.

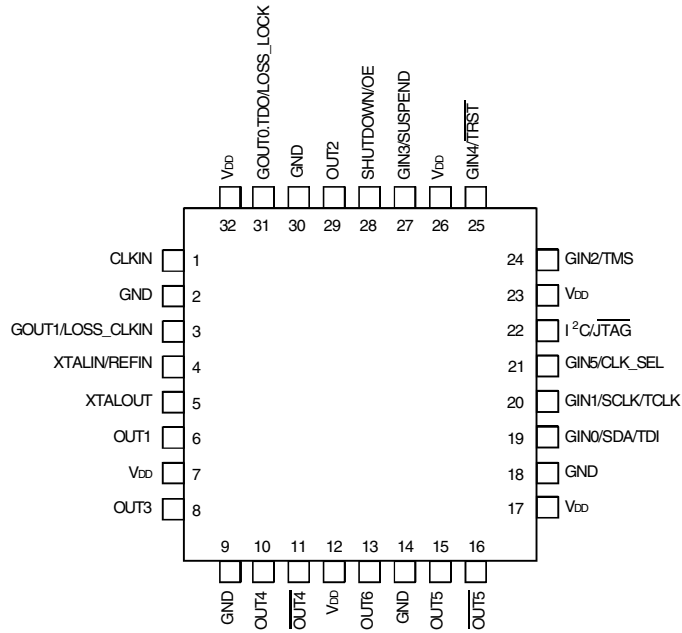
There are 10-bit post dividers on five of the six output banks. Two of the six output banks are configurable to be LVTTTL, LVPECL, or LVDS. The other four output banks are LVTTTL. The outputs are connected to the PLLs via the switch matrix. The switch matrix allows the user to route the PLL outputs to any output bank. This feature can be used to simplify and optimize the board layout. In addition, each output's slew rate and enable/disable function can be programmed.

## FUNCTIONAL BLOCK DIAGRAM

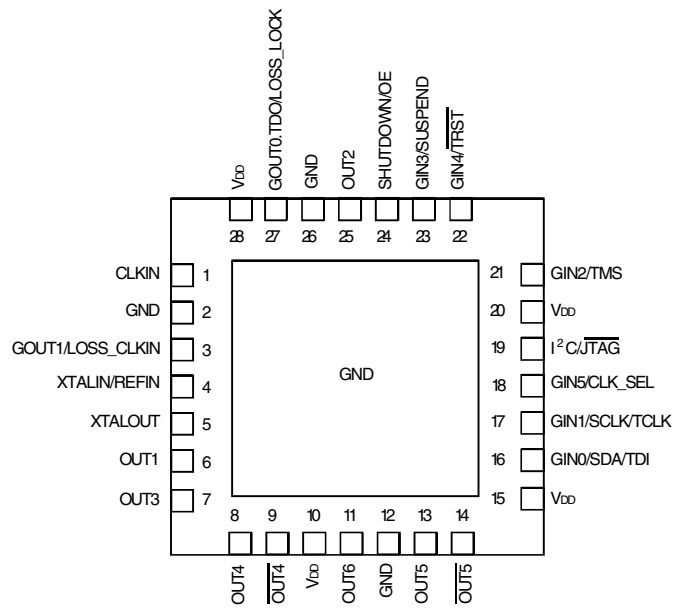


**NOTE:**  
1. OUT4 and OUT5 pairs can be configured to be LVDS, LVPECL, or two single-ended LVTTTL outputs. As LVTTTL, OUT4 and OUT5 can be configured to be non-inverting.

## PIN CONFIGURATION



**TQFP  
TOP VIEW**



**VFQFPN  
TOP VIEW**

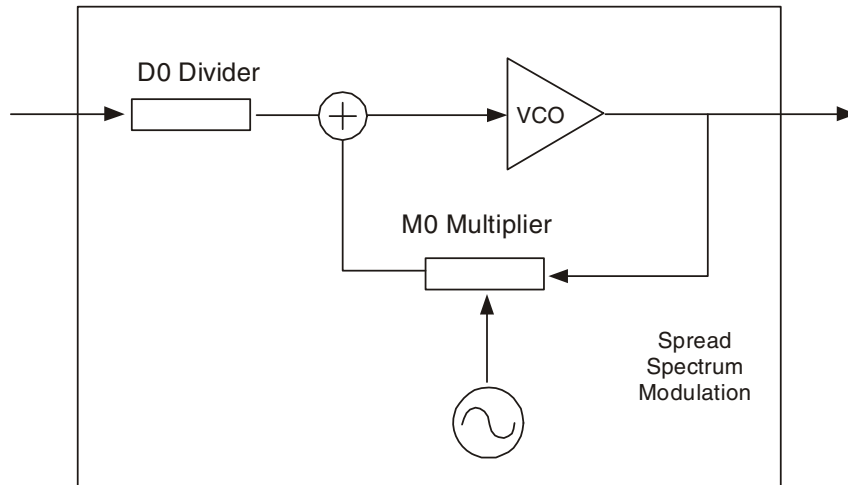
## PIN DESCRIPTION

Pin Name	PF32 Pin#	NL28 Pin#	I/O	Type	Description
CLKIN	1	1	I	LVTTTL	Input Clock
XTALIN/REFIN	4	4	I	LVTTTL	CRYSTAL_IN - Reference crystal input or external reference clock input
XTALOUT	5	5	O	LVTTTL	CRYSTAL_OUT - Reference crystal feedback
GIN0/SDAT/TDI	19	16	I	LVTTTL <sup>(1,2)</sup>	Multi-purpose inputs. Can be used for Frequency Control, SDAT(I <sup>2</sup> C), or TDI(JTAG).
GIN1/SCLK/TCK	20	17	I	LVTTTL <sup>(1,2)</sup>	Multi-Purpose inputs. Can be used for Frequency Control, SCLK(I <sup>2</sup> C), or TCK(JTAG).
GIN2/TMS	24	21	I	LVTTTL <sup>(1,2)</sup>	Multi-Purpose inputs. Can be used for Frequency Control or TMS (JTAG)
GIN3/SUSPEND	27	23	I	LVTTTL <sup>(1,2)</sup>	Multi-Purpose inputs. Can be used for Frequency Control or as a suspend mode control input (active HIGH).
GIN4/TRST	25	22	I	LVTTTL <sup>(1,2)</sup>	Multi-Purpose inputs. Can be used for Frequency Control or TRST (JTAG)
GIN5/CLK_SEL	21	18	I	LVTTTL <sup>(1,2)</sup>	Multi-Purpose inputs. Can be used for Frequency Control or input clock selector.
SHUTDOWN/OE	28	24	I	LVTTTL <sup>(1,2)</sup>	Enables/disables the outputs or powers down the chip. The SP bit (0x1C) controls the polarity of the signal to be either active HIGH or LOW. (Default is active HIGH.)
I <sup>2</sup> C/JTAG	22	19	I	3-level <sup>(3)</sup>	I <sup>2</sup> C (HIGH) or MFC Mode (MID) or JTAG Programming (LOW)
OUT1	6	6	O	LVTTTL	Configurable clock output 1. Can also be used to buffer the reference clock.
OUT2	29	25	O	LVTTTL	Configurable clock output 2
OUT3	8	7	O	LVTTTL	Configurable clock output 3
OUT4	10	8	O	Adjustable <sup>(4)</sup>	Configurable clock output 4, Single-Ended or Differential when combined with OUT4
OUT4	11	9	O	Adjustable <sup>(4)</sup>	Configurable complementary clock output 4, Single-Ended or Differential when combined with OUT4
OUT5	15	13	O	Adjustable <sup>(4)</sup>	Configurable clock output 5, Single-Ended or Differential when combined with OUT5
OUT5	16	14	O	Adjustable <sup>(4)</sup>	Configurable complementary clock output 5, Single-Ended or Differential when combined with OUT5
OUT6	13	11	O	LVTTTL	Configurable clock output 6
GOUT0/TDO/LOSS_LOCK	31	27	O	LVTTTL <sup>(1)</sup>	Multi-Purpose Output. Can be programmed to use as PLL LOCK signal, LOSS_LOCK or TDO in JTAG mode
GOUT1/LOSS_CLKIN	3	3	O	LVTTTL	Multi-Purpose Output. Can be programmed to use as LOSS_CLKIN
V <sub>DD</sub>	7,12,17, 23,26,32	10,15,20 28			3.3V Power Supply
GND	2,9,14, 18,30	2,12,26			Ground

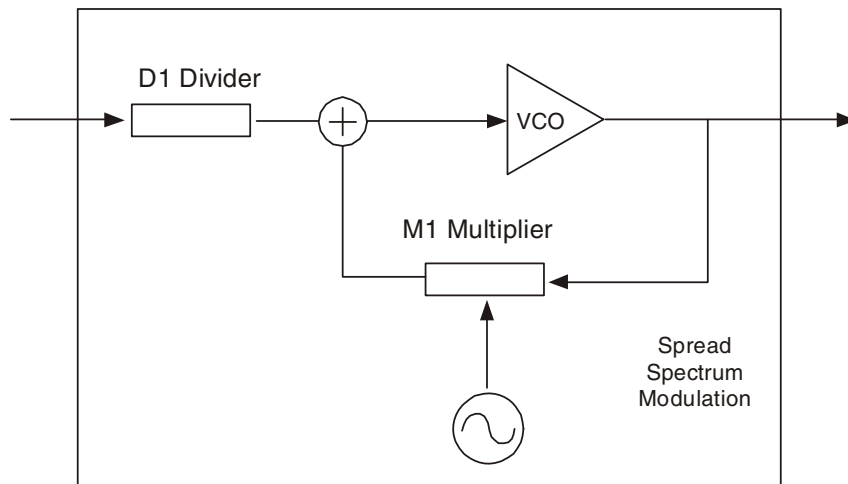
### NOTES:

1. The JTAG (TDO, TMS, TCLK, TRST, and TDI) and I<sup>2</sup>C (SCLK and SDAT) signals share the same pins with GIN signals.
2. Weak internal 100KΩ pull-down resistor.
3. 3-level inputs are static inputs and must be tied to V<sub>DD</sub> or GND or left floating. These inputs are internally biased to V<sub>DD</sub>/2. They are not hot-insertable or over voltage tolerant.
4. Outputs are user programmable to drive single-ended 3.3V LVTTTL, differential LVDS, or differential LVPECL interface levels.

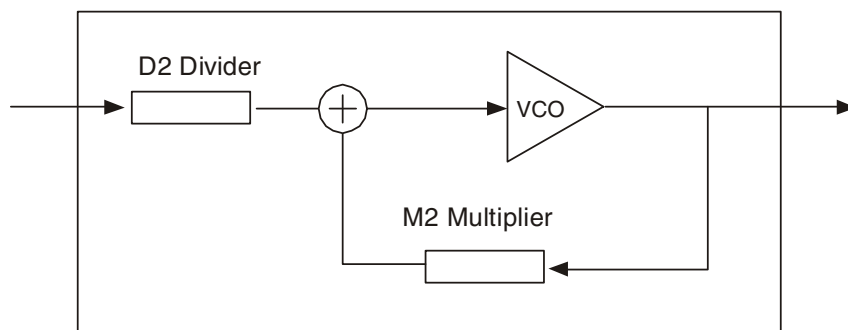
## PLL FEATURES AND DESCRIPTIONS



*PLL0 Block Diagram*



*PLL1 Block Diagram*



*PLL2 Block Diagram*

	Pre-Divider (D) Values	Multiplier (M) Values	Programmable Loop Bandwidth	Spread Spectrum Generation Capability
PLL0	1 - 255	2 - 8190	yes	yes
PLL1	1 - 255	2 - 8190	yes	yes
PLL2	1 - 255	1 - 4095	yes	no

## REFERENCE CLOCK INPUT PINS AND SELECTION

The 5V9885T supports up to two clock inputs. One of the clock inputs (XTALIN/ REFIN) can be driven by either an external crystal or a reference clock. The second clock input (CLKIN) can only be driven from an external reference clock. Either clock input can be set as the primary clock. The primary clock designation is to establish which is the main reference clock to the PLLs. The non-primary clock is designated as the secondary clock in case the primary clock goes absent and a backup is needed. The PRIMCLK bit (0x34) determines which clock input will be the primary clock. When PRIMCLK bit is "0", it will select XTALIN/REFIN as the primary, and when "1", it will select CLKIN as the primary. The two external reference clocks can be manually selected using the GIN5/CLK\_SEL pin, except in Manual Frequency Control (MFC) mode 2, or via programming by hard wiring the CLK\_SEL pin and toggling the PRIMCLK bit. For more details on the MFC modes, refer to the CONFIGURING MULTI-PURPOSE I/Os section. When CLK\_SEL is LOW, the primary clock is selected and when HIGH, the secondary clock is selected. The SM bits (0x34) must be set to "0x" for manual switchover which is detailed in SWITCHOVER MODES section.

GIN5/CLK_SEL	Selected Clock Input
L	Primary
H	Secondary

### Crystal Input (XTALIN/REFIN)

The crystal oscillators should be fundamental mode quartz crystals: overtone crystals are not suitable. Crystal frequency should be specified for parallel resonance with 50Ω maximum equivalent series resonance.

When the XTALIN/REFIN pin is driven by a crystal, it is important to set the internal oscillator inverter drive strength and internal tuning/load capacitor values correctly to achieve the best clock performance. These values are programmable through either I<sup>2</sup>C or JTAG interface to allow for maximum compatibility with crystals from various manufacturers, processes, performances, and qualities. The internal load capacitors are true parallel-plate capacitors for ultra-linear performance. Parallel-plate capacitors were chosen to reduce the frequency shift that occurs when non-linear load capacitance interacts with load, bias, supply, and temperature changes. External non-linear crystal load capacitors should not be used for applications that are sensitive to absolute frequency requirements. The value of the internal load capacitors are determined by XTALCAP[7:0] bits, (0x07). The load capacitance can be set with a resolution of 0.125 pF for a total crystal load range of 3.5pF to 35.5pF. This value should be set to two times the crystal load capacitance value stated by the vendor, subtracting out board capacitance value. Check with the vendor's crystal load capacitance specification for the exact setting to tune the internal load capacitor. The following equation governs how the total internal load capacitance is set.

Ex.: For crystal capacitance = 12pF

For board capacitance = 3pF each leg

XTALCAP = 2x [12-3] = 18pF

$$XTAL \text{ load cap} = 3.5pF + XTALCAP[7:0] * 0.125pF \text{ (Eq. 1)}$$

Parameter	Bits	Step	Min	Max	Units
XTALCAP	8	0.125	0	32	pF

When using an external reference clock instead of a crystal on the XTAL/ REFIN pin, the input load capacitors may be completely bypassed. This allows for the input frequency to be up to 200MHz. When using an external reference clock, the XTALOUT pin must be left floating, XTALCAP must be programmed to the default value of "0", and crystal drive strength bit, XDRV (0x06), must be set to the default value of "11".

### CLKIN Pin

CLKIN pin is a regular clock input pin, and can be driven up to 400MHz.

## PRE-SCALER, FEEDBACK-DIVIDER, AND POST-DIVIDER

Each PLL incorporates an 8-bit pre-scaler and a 12-bit feedback divider which allows the user to generate three unique non-integer-related frequencies. For output banks OUT2-OUT6, each bank has a 10-bit post-divider. The following equation governs how the frequency on output banks OUT2-6 is calculated.

$$F_{OUT} = \frac{F_{IN} * \left(\frac{M}{D}\right)}{P * 2} \text{ (Eq. 2)}$$

Where F<sub>IN</sub> is the reference frequency, M is the total feedback-divider value, D is the pre-scaler value, P is the total post-divider value, and F<sub>OUT</sub> is the resulting output bank frequency. The value 2 in the denominator is due to the divide-by-2 on each of the output banks OUT2-6. Note that OUT1 does not have any type of post-divider. Also, programming any of the dividers may cause glitches on the outputs.

### Pre-Scaler

D[7:0] are the bits used to program the pre-scaler for each PLL, D0 for PLL0, D1 for PLL1, and D2 for PLL2. The pre-scalers divide down the reference clock with integer values ranging from 1 to 255. To maintain low jitter, the divided down clock must be higher than 400KHz; it is best to use the smallest D divider value possible. If D is set to '0x00', then this will power down the PLL and all the outputs associated with that PLL.

**Feedback-Divider**

N[11:0] and A[3:0] are the bits used to program the feedback-divider for PLL0 (N0 and A0) and PLL1 (N1 and A1). If spread spectrum generation is enabled for either PLL0 or PLL1, then the SS\_OFFSET[5:0] bits (0x61, 0x69) would be factored into the overall feedback divider value. See the SPREAD SPECTRUM GENERATION section for more details on how to configure PLL0 and PLL1 when spread spectrum is enabled. The two PLLs can also be configured for fractional divide ratios. See FRACTIONAL DIVIDER for more details. For PLL2, only the N[11:0] bits (N2) are used to program its feedback divider and there is no spread spectrum generation and fractional divides capability. The 12-bit feedback-divider integer values range from 1 to 4095.

The following equations govern how the feedback divider value is set. Note that the equations are different for PLL0/PLL1 and PLL2

PLL0 and PLL1:

$$M = 2 * N[11:0] + A[3:0] + 1 + SS\_OFFSET[5:0] * 1/64 \quad (\text{Eq. 3})$$

$$M = 2 * N[11:0] + A[3:0] + 1 \quad (\text{spread spectrum disabled}) \quad (\text{Eq. 4})$$

- A[3:0] = 0000 = -1
- = 0001 = 1
- = 0010 = 2
- = 0011 = 3
- .
- .
- .
- = 1111 = 15

Note: A[3:0] < (N[11:0] - 5), must be met when using A. N cannot be programmed with a value of 4, 8, or 16 when using A.

PLL2:

$$M = N[11:0] \quad (\text{Eq. 5})$$

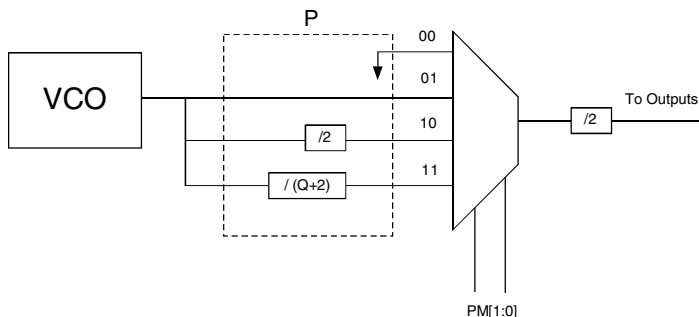
The user can achieve an even or odd integer divide ratio for both PLL0 and PLL1 by setting the A[3:0] bits accordingly and disabling the spread spectrum. A fractional divide can also be set for PLL0 and PLL1 by using the A[3:0] bits in conjunction with the SS\_OFFSET[5:0] bits, which is detailed in the FRACTIONAL DIVIDER section. Note that the VCO has a frequency range of 10MHz to 1200MHz. To maintain low jitter, it is best to maximize the VCO frequency. For example, if the reference clock is 100MHz and a 200MHz clock is required, to achieve the best jitter performance, multiply the 100MHz by 12 to get the VCO running at the highest possible frequency of 1200MHz and then divide it down to get 200MHz. Or if the reference clock is 25MHz and 20MHz is the required clock, multiply the 25MHz by 40 to get the VCO running at 1000MHz and then divide it down to get 20MHz. If N is set to '0x00', the VCO will slew to the minimum frequency.

**Post-Divider**

Q[9:0] are the bits used to program the 10-bit post-dividers on output banks OUT2-6. OUT1 bank does not have a 10-bit post-divider or any other post-divide along its path. The 10-bit post-dividers will divide down the output banks' frequency with integer values ranging from 1 to 1023.

There is the option to choose between disabling the post-divider, utilizing a div/1, a div/2, or the 10-bit post-divider by using the PM[1:0] bits. Each bank, except for OUT1, has a set of PM bits. When disabling the post-divider, no clock will appear at the outputs, but will remain powered on. The values are listed in the table below.

PM[1:0]	P Post-Divider
00	disabled
01	div/1
10	div/2
11	Q[9:0] + 2 (Eq. 6)



*Post-Divider Diagram*

Note that the actual 10-bit post-divider value has a 2 added to the integer value Q and the outputs are routed through another div/2 block. The post-divider should never be disabled unless the output bank will never be used during normal operation. The output frequency range for LVTTTL outputs are from 4.9KHz to 200MHz. The output frequency range for LVPECL/LVDS outputs are from 4.9KHz to 500MHz.

## SPREAD SPECTRUM GENERATION

PLL0 and PLL1 support spread spectrum generation capability, which users have the option of turning on and off. Spread spectrum profile, frequency, and spread are fully programmable (within limits). The programmable spread spectrum generation parameters are TSSC[3:0], NSSC[3:0], SS\_OFFSET[5:0], SD[3:0], DITH, and X2 bits. These bits are in the memory address range of 0x60 to 0x67 for PLL0 and 0x68 to 0x6F for PLL1. The spread spectrum generation on PLL0 & PLL1 can be enabled/disabled using the TSSC[3:0] bits. To enable spread spectrum, set TSSC > '0' and set NSSC, SD[3:0], SD[5:0], and the A[3:0] in the total M value accordingly. And to disable, set TSSC = '0'.

### TSSC[3:0]

These bits are used to determine the number of phase/frequency detector cycles per spread spectrum cycle (ssc) steps. The modulation frequency can be calculated with the TSSC bits in conjunction with the NSSC bits. Valid TSSC integer values for the modulation frequency range from 5 to 14.

### NSSC[3:0]

These bits are used to determine the number of delta-encoded samples used for a single quadrant of the spread spectrum waveform. All four quadrants of the spread spectrum waveform are mirror images of each other. The modulation frequency is also calculated based off the NSSC bits in conjunction with the TSSC bits. Valid NSSC integer values range from 1 to 6.

### SS\_OFFSET[5:0]

These bits are used to program the fractional offset with respect to the nominal M integer value. For center spread, the SS\_OFFSET should be set to '0' so the spread spectrum waveform is about the nominal M (Mnom) value. For down spread, the SS\_OFFSET > '0' so the spread spectrum waveform is about the (Mideal - 1 = Mnom) value. The downspread percentage can be thought of in terms of center spread. For example, a downspread of -1% can also be considered as a center spread of ±0.5% but with Mnom shifted down by one and offset. The SS\_OFFSET has integer values ranging from 0 to 63.

### SD[3:0]

These bits are used to shape the profile of the spread spectrum waveform. These are delta-encoded samples of the waveform. There are twelve sets of SD samples for each PLL. The NSSC bits determine how many of these samples are used for the waveform. The sum of these delta-encoded samples (sigma-delta-encoded samples) determine the amount of spread and should not exceed (63 - SS\_OFFSET). The maximum spread is inversely proportional to the nominal M integer value.

### DITH

This bit is for dithering the sigma-delta-encoded samples. This will randomize the least-significant bit of the input to the spread spectrum modulator. Set the bit to '1' to enable dithering.

### X2

This bit will double the total value of the sigma-delta-encoded-samples which will increase the amplitude of the spread spectrum waveform by a factor of two. When X2 is '0', the amplitude remains nominal but if set to '1', the amplitude is increased by x2.

The following equations govern how the spread spectrum is set:

$$T_{SSC} = TSSC[3:0] + 2 \quad (\text{Eq. 7})$$

$$N_{SSC} = NSSC[3:0] * 2 \quad (\text{Eq. 8})$$

$$SD[3:0]_k = S_{j+1}(\text{unencoded}) - S_j(\text{unencoded}) \quad (\text{Eq. 9})$$

where  $S_j$  is the unencoded sample out of a possible 12 and  $SD_k$  is the delta-encoded sample out of a possible 12.

$$\text{Amplitude} = \frac{(2 * N[11:0] + A[3:0] + 1) * \text{Spread}\%}{2} / 100 \quad (\text{Eq. 10})$$

if  $1 < \text{Amp} < 2$ , then set X2 bit to '1'.

**Modulation frequency:**

$$F_{\text{PFD}} = F_{\text{IN}} / D \quad (\text{Eq. 11})$$

$$F_{\text{VCO}} = F_{\text{PFD}} * M_{\text{NOM}} \quad (\text{Eq. 12})$$

$$F_{\text{SSC}} = F_{\text{PFD}} / (4 * N_{\text{SSC}} * T_{\text{SSC}}) \quad (\text{Eq. 13})$$

**Spread:**

$$\Sigma\Delta = SD_0 + SD_1 + SD_2 + \dots + SD_{11}$$

the number of samples used depends on the N<sub>SSC</sub> value

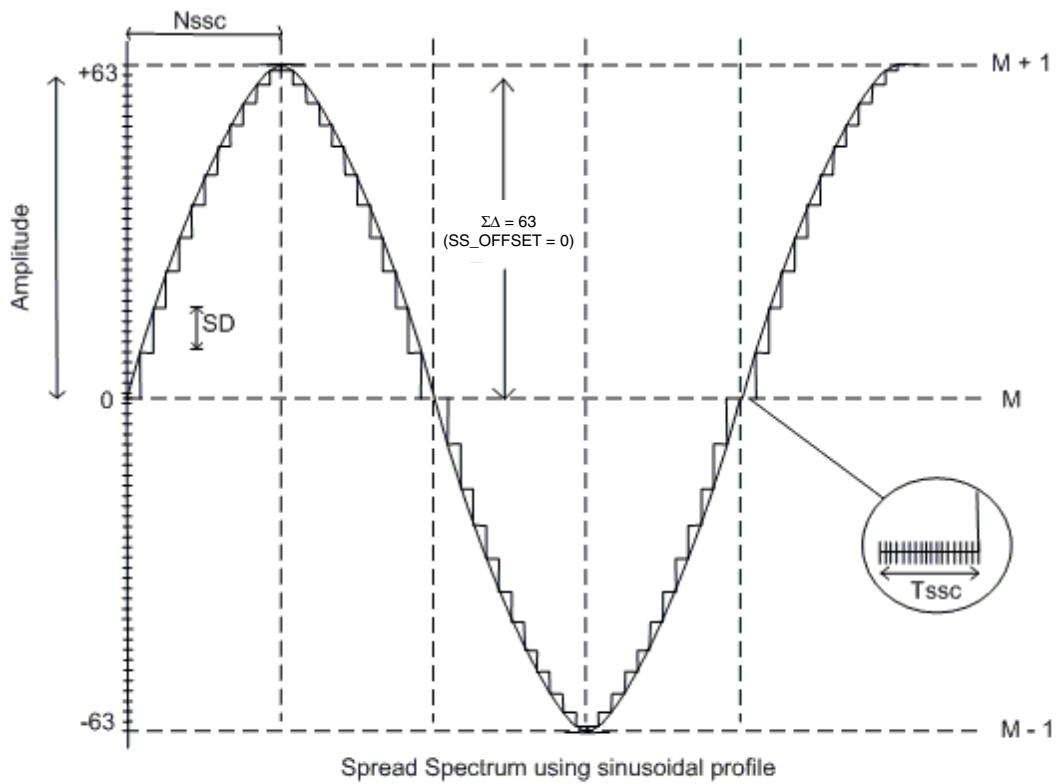
$$\Sigma\Delta \leq 63 - \text{SS\_OFFSET}$$

$$\pm \text{Spread\%} = \frac{\Sigma\Delta * 100}{64 * (2 * N[11:0] + A\{3:0\} + 1)} \quad (\text{Eq. 14})$$

$$\pm \text{Max Spread\%} / 100 = 1 / M_{\text{NOM}} \text{ or } 2 / M_{\text{NOM}} (X2=1)$$

**Profile:**

Waveform starts with SS\_OFFSET, SS\_OFFSET + SD<sub>J</sub>, SS\_OFFSET + SD<sub>J+1</sub>, etc.



*Spread Spectrum Using Sinusoidal Profile*

### Example

$F_{IN} = 25\text{MHz}$ ,  $F_{OUT} = 100\text{MHz}$ ,  $F_{SSC} = 33\text{KHz}$  with center spread of  $\pm 2\%$ . Find the necessary spread spectrum register settings.

Since the spread is center, the SS\_OFFSET can be set to '0'. Solve for the nominal M value; keep in mind that the nominal M should be chosen to maximize the VCO. Start with  $D = 1$ , using Eq.10 and Eq.11.

$$M_{NOM} = 1200\text{MHz} / 25\text{MHz} = 48$$

Using Eq.4, we arbitrarily choose  $N = 22$ ,  $A = 3$ . Now that we have the nominal M value, we can determine TSSC and NSSC by using Eq.12.

$$N_{SSC} * T_{SSC} = 25\text{MHz} / (33\text{KHz} * 4) = 190$$

However, using Eq. 7 and Eq.8, we find that the closest value is when  $T_{SSC} = 14$  and  $N_{SSC} = 6$ . Keep in mind to maximize the number of samples used to enhance the profile of the spread spectrum waveform.

$$T_{SSC} = 14 + 2 = 16$$

$$N_{SSC} = 6 * 2 = 12$$

$$N_{SSC} * T_{SSC} = 192$$

Use Eq. 14 to determine the value of the sigma-delta-encoded samples.

$$\pm 2\% = \frac{\Sigma\Delta * 100}{64 * 48}$$

$$\Sigma\Delta = 61.44$$

Either round up or down to the nearest integer value. Therefore, we end up with 61 or 62 for sigma-delta-encoded samples. Since the sigma-delta-encoded samples must not exceed 63 with SS\_OFFSET set to '0', 61 or 62 is well within the limits. It is the discretion of the user to define the shape of the profile that is better suited for the intended application.

Using Eq. 14 again, the actual spread for the sigma-delta-encoded samples of 61 and 62 are  $\pm 1.99\%$  and  $\pm 2.02\%$ , respectively.

Use Eq. 10 to determine if the X2 bit needs to be set;

$$\text{Amplitude} = \frac{48 * (1.99 \text{ or } 2.02) / 100}{2} = 0.48 < 1$$

Therefore, the X2 = '0'. The dither bit is left to the discretion of the user.

The example above was of a center spread using spread spectrum. For down spread, the nominal M value can be set one integer value lower to 43.

Note that the 5V9885T should not be programmed with  $T_{SSC} > '0'$ ,  $SS\_OFFSET = '0'$ , and  $SD = '0'$  in order to prevent an unstable state in the modulator. The PLL loop bandwidth must be at least 10x the modulation frequency along with higher damping (larger  $\omega_{\Sigma\Delta}$ ) to prevent the spread spectrum from being filtered and reduce extraneous noise. Refer to the LOOP FILTER section for more detail on  $\omega_{\Sigma\Delta}$ . The A[3:0] must be used for spread spectrum, even if the total multiplier value is an even integer.

## FRACTIONAL DIVIDER

There is the option for the feedback-divider to be programmed as a fractional divider for only PLL0 and PLL. By setting  $T_{SSC} > '0'$  and SD bits to '0', the SS\_OFFSET bits would determine the fractional divide value. See the SPREAD SPECTRUM GENERATION section for more details on the TSSC, SD, and SS\_OFFSET bits. The following equation governs how the fractional divide value is set.

$$M = 2 * N[11:0] + A[3:0] + 1 + SS\_OFFSET[5:0] * 1/64$$

The spread spectrum parameters such as the modulation frequency and profile will not be enabled nor will it have any impact on the PLL output when the PLL is programmed for fractional divide.

The following is an example of how to set the fractional divider.

**Example**

$F_{IN} = 20\text{MHz}$ ,  $F_{OUT1} = 168.75\text{MHz}$ ,  $F_{OUT2} = 350\text{MHz}$

Solving for 350MHz using Eq.2 and Eq.3 with PLL0 and spread spectrum off,

$$350\text{MHz} = 20\text{MHz} * (M / D) / (P * 2)$$

For better jitter performance, keep D as small as possible

$$(350\text{MHz} * 2 / 20\text{MHz}) = (M/P) = 35$$

Therefore, we have  $D = 1$ ,  $M = 35$  ( $N = 16$ ,  $A = 2$ ) for PLL0 with  $P = 1$  on output bank4 resulting in 350MHz.

Solving for 168.75MHz with PLL1 and fractional divide enabled:

$$168.75\text{MHz} = 20\text{MHz} * (M / D) / (P * 2)$$

$$168.75\text{MHz} * 2 / 20\text{MHz} = M/P = 16.875/1 \text{ or } 33.75/2$$

The 33.75 value is chosen to achieve the highest VCO frequency possible. Next step is to figure out the setting for the fractional divide using Eq.3.

$$33.75 = 2 * N + A + 1 + SS\_OFFSET * 1/64$$

Integer value 33 can be determined by N and A, thus leaving 0.75 left to be solved.

$$2 * N + A + 1 = 33$$

$$SS\_OFFSET = 64 * 0.75 = 48$$

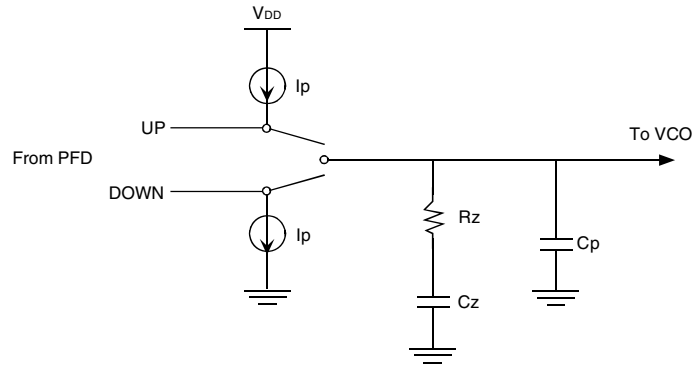
Therefore, we have  $D=1$ ,  $M=33.75$  ( $N=15$ ,  $A=2$ ,  $SS\_OFFSET=48$ ) for PLL1 with  $P=2$  on an output bank resulting in 168.75MHz.

The fractional divider can be determined if it is needed by following the steps in the previous example. Note that the 5V9885T should not be programmed with  $TSSC > '0'$ ,  $SS\_OFFSET = '0'$ , and  $SD = '0'$  in order to prevent an unstable state in the modulator. The  $A[3:0]$  must be used and set to be greater than '2' for a more accurate fractional divide.

## LOOP FILTER

The loop filter for each PLL can be programmed to optimize the jitter performance. The low-pass frequency response of the PLL is the mechanism that dictates the jitter transfer characteristics. The loop bandwidth can be extracted from the jitter transfer. A narrow loop bandwidth is good for jitter attenuation while a wide loop bandwidth is best for low jitter generation. The specific loop filter components that can be programmed are the resistor via the RZ[3:0] bits, pole capacitor via the CZ[3:0] bits, zero capacitor via the CP[3:0] bits, and the charge pump current via the IP[2:0] bits.

The following equations govern how the loop filter is set.



**Charge Pump and Loop Filter Configuration**

$$\text{Resistor (Rz)} = 0.3\text{K}\Omega + \text{RZ}[3:0] * 1\text{K}\Omega \quad (\text{Eq. 15})$$

$$\text{Zero capacitor (Cz)} = 6\text{pF} + \text{CZ}[3:0] * 27.2\text{pF} \quad (\text{Eq. 16})$$

$$\text{Pole capacitor (Cp)} = 1.3\text{pF} + \text{CP}[3:0] * 0.75\text{pF} \quad (\text{Eq. 17})$$

$$\text{Charge pump current (Ip)} = 5 * 2^{\text{IP}[2:0]} \mu\text{A} \quad (\text{Eq. 18})$$

Parameter	Bits	Step	Min	Max	Units
RZ	4	1	0.3	15.3	K $\Omega$
CZ	4	27.2	6	414	pF
CP	4	0.75	1.3	12.55	pF
IP	3	2 <sup>n</sup>	5	640	$\mu\text{A}$

PLL loop filter design is beyond the scope of this datasheet. Refer to design procedures for 3-order charge-pump based PLLs. For the sake of simplicity, the fastest and easiest way to calculate the PLL loop bandwidth (Fc) given the programmable loop filter parameters is as follows.

### PLL Loop Bandwidth:

$$\text{Charge pump gain (K}\phi\text{)} = \text{Ip} / 2\pi \quad (\text{Eq. 19})$$

$$\text{VCO gain (Kvco)} = 950\text{MHz/V} * 2\pi \quad (\text{Eq. 20})$$

M = Total multiplier value (See the PRE-SCALERS, FEEDBACK-DIVIDERS, POST-DIVIDERS section for more detail)

$$\omega_c = \frac{\text{Rz} * \text{K}\phi * \text{Kvco} * \text{Cz}}{\text{M} * (\text{Cz} + \text{Cp})} \quad (\text{Eq. 21})$$

$$\text{Fc} = \omega_c / 2\pi \quad (\text{Eq. 22})$$

Note, the phase/frequency detector frequency (F<sub>PFD</sub>) is typically seven times the PLL closed-loop bandwidth (Fc) but too high of a ratio will reduce your phase margin thus compromising loop stability.

To determine if the loop is stable, the phase margin ( $\phi_m$ ) would need to be calculated as follows.

**Phase Margin:**

$$\omega_z = 1 / (R_z * C_z) \quad (\text{Eq. 23})$$

$$\omega_p = \frac{C_z + C_p}{R_z * C_z * C_p} \quad (\text{Eq. 24})$$

$$\phi_m = (360 / 2\pi) * [\tan^{-1}(\omega_c / \omega_z) - \tan^{-1}(\omega_c / \omega_p)] \quad (\text{Eq. 25})$$

To ensure stability in the loop, the phase margin is recommended to be  $> 60^\circ$  but too high will result in the lock time being excessively long. Certain loop filter parameters would need to be compromised to not only meet a required loop bandwidth but to also maintain loop stability.

Example

$F_c = 150\text{KHz}$  is the desired loop bandwidth. The total M value is 850. The ratio of  $\omega_p / \omega_c$  should be at least 4. A rule of thumb that will help to aid the way, the  $\omega_p / \omega_c$  ratio should be at least 4. Given  $F_c$  and M, an optimal loop filter setting needs to be solved for that will meet both the PLL loop bandwidth and maintain loop stability.

The charge pump gain should be relatively small as possible to achieve a low loop bandwidth.

$$I_p = 40\mu\text{A}$$

$$K_\phi * K_{vco} = 950\text{MHz/V} * 40\mu\text{A} = 38000\text{A/Vs}$$

Loop Bandwidths

$$\omega_c = 2\pi * F_c = 9.42 \times 10^5 \text{ s}^{-1}$$

$$\omega_z = \omega_p / \omega_c = 4 \quad (\text{Eq. 26})$$

$$\omega_c^2 = \omega_p * \omega_z \quad (\text{Eq. 27})$$

$$\omega_p = \frac{C_z + C_p}{R_z * C_z * C_p} = \omega_c (1 + C_z / C_p)$$

Solving for  $C_z$ ,  $C_p$ , and  $R_z$

Knowing  $\omega_c = \frac{R_z * K_\phi * K_{vco} * C_z}{M * (C_z + C_p)}$  and substituting in the equations from above,

$C_z \gg C_p$ , therefore, we can easily derive  $C_p$  to be

$$C_p = \frac{K_\phi * K_{vco}}{M * \omega_c^2 * \omega_z} = 12.60\text{pF}$$

Similarly for  $C_z$  and  $R_z$

$$C_z = \frac{K_\phi * K_{vco} * (\omega_z^2 - 1)}{M * \omega_c^2 * \omega_z} = 189\text{pF}$$

$$R_z = \frac{M * \omega_c * \omega_z^2}{K_\phi * K_{vco} * (\omega_z^2 - 1)} = 22.48\text{K}\Omega$$

Based on the loop filter parameter equations from above, since there are no possible values of 12.60pF for  $C_p$ , 189pF for  $C_z$ , and 22.48K $\Omega$  for  $R_z$ , the next possible values within the loop filter settings are 12.55pF ( $CP[3:0]=1111$ ), 196.4pF ( $CZ[3:0]=0111$ ), and 15.3K $\Omega$  ( $RZ[3:0]=1111$ ), respectively. This loop filter setting will yield a loop bandwidth of about 102KHz. The phase margin must be checked for loop stability.

$$\phi_m = (360 / 2\pi) * [\tan^{-1}(6.41 \times 10^5 \text{ s}^{-1} / 3.33 \times 10^5 \text{ s}^{-1}) - \tan^{-1}(6.41 \times 10^5 \text{ s}^{-1} / 5.54 \times 10^6 \text{ s}^{-1})] = 56^\circ$$

Although slightly below  $60^\circ$ , the phase margin would be acceptable with a fairly stable loop.

## CONFIGURING THE MULTI-PURPOSE I/Os

The 5V9885T can operate in four distinct modes. These modes are controlled by the MFC bit (0x04) and the I<sup>2</sup>C/JTAG pin. The general purpose I/O pins (GIN0, GIN1, GIN2, GIN3, GIN4, GIN5) have different uses depending on the mode of operation. The four available modes of operation are:

- 1) Manual Frequency Control (MFC) Mode for PLL0 Only
- 2) Manual Frequency Control (MFC) Mode for all three PLLs
- 3) I<sup>2</sup>C Programming Mode
- 4) JTAG Programming Mode

Along with the GINx pins are also GOUTx output pins that can take up a different function depending on the mode of operation. See table below for description.

Multi-Purpose Pins	Other Signal Functions	Signal Description
GIN0	SDAT / TDI	I <sup>2</sup> C serial data input / JTAG serial data input
GIN1	SCLK / TCK	I <sup>2</sup> C clock input / JTAG clock input
GIN2	TMS	JTAG control signal to the TAP controller state machine
GIN3	SUSPEND	Suspends all outputs of PLL (Active High)
GIN4	$\overline{\text{TRST}}$	JTAG active LOW input to asynchronously reset the BST
GIN5	CLK_SEL	Reference clock select between XTALIN/REFIN and CLKIN
GOUT0	TDO / LOSS_LOCK	JTAG serial data output / Detects loss of PLL lock <sup>(1)</sup>
GOUT1	LOSS_CLKIN	Detects loss of the primary clock source <sup>(1)</sup>

**NOTE:**

1. Please see detail description in Loss of Lock and Input Clock section.

Each PLL's programming registers can store up to four different Dx and Mx configurations in combination with two different P configurations in MFC modes. The post-divider should never be disabled in any of the two P configurations unless the output bank will never be used during normal operation. The PLL's loop filter settings also has four different configurations to store and select from. This will be explained in the MODE1 and MODE2 sections. The use of the GINx pins in MFC mode control the selection of these configurations.

### MODE1 - Manual Frequency Control (MFC=1) Mode for PLL0 Only

In this mode, only 8 configurations of PLL0 can be changed during operation. The GIN0, GIN1 and GIN2 pins control the selection of eight different configurations (D, M, Rz, Cz, Cp and Ip) of PLL0. GIN3 becomes PLL SUSPEND pin, GIN4 is not available to users, and GIN5 becomes CLK\_SEL pin. The output GOUT0 will become an indicator for loss of PLL lock (LOSS\_LOCK). GOUT1 pin will become an indicator for loss of the primary clock (LOSS\_CLKIN).

The PLL0 has 4 sets of dedicated registers for D, M, Rz, Cz, Cp, Ip and ODIV. For additional 4 sets of registers, the PLL0 uses registers from CONFIG2 and CONFIG3 of PLL1 and PLL2. The PLL1 and PLL2 will still be fully operational, but have only one fixed configuration in this mode, and the default configuration will be set to CONFIG0 of PLL1 and CONFIG0 of PLL2. (Please see page 18 for register location.)

The output banks will each have two P configurations that can be associated with each of the PLL configurations. Each of the two P configurations has its own set of PM bits (See the PRE-SCALERS, FEEDBACK-DIVIDERS, POST-DIVIDERS section for more detail on the PM bits). Use the ODIV bit to choose which post-divider configuration to associate with a specific PLL configuration.

To enter this mode, users must set MFC bit to "1", and I<sup>2</sup>C/JTAG pin must be left floating.

GIN2 Pin	GIN1 Pin	GIN0 Pin	PLL0 Configuration Selection (Mode 1)
0	0	0	Configuration0
0	0	1	Configuration1
0	1	0	Configuration2
0	1	1	Configuration3
1	0	0	Configuration4
1	0	1	Configuration5
1	1	0	Configuration6
1	1	1	Configuration7

### MODE2 - Manual Frequency Control (MFC=0) Mode for all PLLs

In this mode, the configuration of PLL0, PLL1, and PLL2 can be changed during operation. The GINx pins are used to control the selection of up to four different Dx, Mx, RZx, CZx, CPx, and IPx configurations for each PLL. GIN0 and GIN1 become configuration selection pins for D0 and M0 of PLL0, GIN2 and GIN3 become configuration selection pins for PLL1, and GIN4 and GIN5 become configuration selection pins for D2 and M2 of PLL2. The output GOUT0 will become an indicator for loss of PLL lock (LOSS\_LOCK). GOUT1 pin will become an indicator for loss of the primary clock (LOSS\_CLKIN).

The output banks will have two different P configurations to choose from for each of the four PLL configurations. Each of the two P configurations has its own set of PM bits (See the PRE-SCALERS, FEEDBACK-DIVIDERS, POST-DIVIDERS section for more detail on the PM bits). Use the ODIV bit to choose which post-divider configuration to associate with a specific PLL configuration. For example, if ODIV2\_CONFIG2=1, then when Config2 is selected Qx[9:0]\_CONFIG1 is selected as the post-divider value to be used. Note that there is an ODIVx bit for each of the PLL configurations. In this way, the post-divider values can change with the configuration.

To enter this mode, users must set MFC bit to "0", and I<sup>2</sup>C/JTAG pin must be left floating.

GIN1 Pin	GIN0 Pin	PLL0 Configuration Selection (Mode 2)
0	0	Configuration 0
0	1	Configuration 1
1	0	Configuration 2
1	1	Configuration 3

GIN5 Pin	GIN4 Pin	PLL2 Configuration Selection (Mode 2)
0	0	Configuration 0
0	1	Configuration 1
1	0	Configuration 2
1	1	Configuration 3

GIN3 Pin	GIN2 Pin	PLL1 Configuration Selection (Mode 2)
0	0	Configuration 0
0	1	Configuration 1
1	0	Configuration 2
1	1	Configuration 3

### MODE3 - I<sup>2</sup>C Programming Mode

In this mode, GIN0, GIN1, GIN3 and GIN5 become SDAT (I<sup>2</sup>C data), SCLK (I<sup>2</sup>C clock), SUSPEND and CLK\_SEL signal pins, respectively. The output GOUT0 will become an indicator for loss of PLL lock (LOSS\_LOCK). GOUT1 pin will become an indicator for loss of the primary clock (LOSS\_CLKIN). GIN2 and GIN4 are not available to users.

To enter this mode, I<sup>2</sup>C/JTAG pin must be set HIGH.

### MODE4 - JTAG Programming Mode

In this mode, GIN0, GIN1, GIN2, GIN3, GIN4 and GIN5 will become TDI (JTAG data in), TCK (JTAG clock), TMS (JTAG control signal), SUSPEND,  $\overline{\text{TRST}}$  (JTAG reset) and CLK\_SEL signal pins, respectively. The output GOUT0 will become JTAG TDO signal, and GOUT1 will be an indicator for loss of the selected clock (LOSS\_CLKIN).

To enter this mode, I<sup>2</sup>C/JTAG pin must be set LOW.

Multi-Purpose pins	Manual Frequency Control modes			
	Mode1	Mode2	JTAG	I <sup>2</sup> C
GIN0	GIN0	GIN0	TDI	SDAT
GIN1	GIN1	GIN1	TCK	SCLK
GIN2	GIN2	GIN2	TMS	n/a
GIN3	SUSPEND	GIN3	SUSPEND	SUSPEND
GIN4	n/a	GIN4	$\overline{\text{TRST}}$	n/a
GIN5	CLK_SEL	GIN5 <sup>(1)</sup>	CLK_SEL	CLK_SEL
GOUT0	LOSS_LOCK	LOSS_LOCK	TDO	LOSS_LOCK
GOUT1	LOSS_CLKIN	LOSS_CLKIN	LOSS_CLKIN	LOSS_CLKIN

**NOTE:**

- The PLL(s) will lock onto the primary clock and the manual switchover can be controlled by the PRIMCLK bit.

### Understanding the GIN Signals

During power up, the part will virtually be in MFC mode2, therefore, the values of GIN4, GIN3, GIN2, GIN1 and GIN0 will be latched and used for PLL configuration selection, regardless of the state of the I<sup>2</sup>C/JTAG pin. GIN5 is not latched, and will assume the LOW state internally when in programming mode. This means that when in programming mode, the PLL configuration can only be changed by writing directly to the registers of the currently selected configuration. When in MFC mode 2, configuration 0 or 1 (GIN5=0) should be selected if you do not want to change configurations when entering or leaving programming mode. The GIN pins should be held LOW during power up to select configuration0 as default.

When not in programming mode, the GIN inputs directly control the selected configuration. The internal GINx signals can be individually disabled via programming the GINEN bits (0x06). When disabled by setting GINENx to "0", the GINx inputs may be left floating, but during power up, the GIN pins will still latch. Disabled inputs are interpreted as LOW by the internal state machines. Even if disabled, GIN2, GIN1, GIN0 and GIN4 pins will be enabled if required for I<sup>2</sup>C or JTAG programming functions when in programming mode. The SUSPEND and CLK\_SEL functions on the GIN3 and GIN5 pins, respectively, will be rendered completely non-functional when disabled.

## SHUTDOWN/SUSPEND/ENABLE OF OUTPUTS

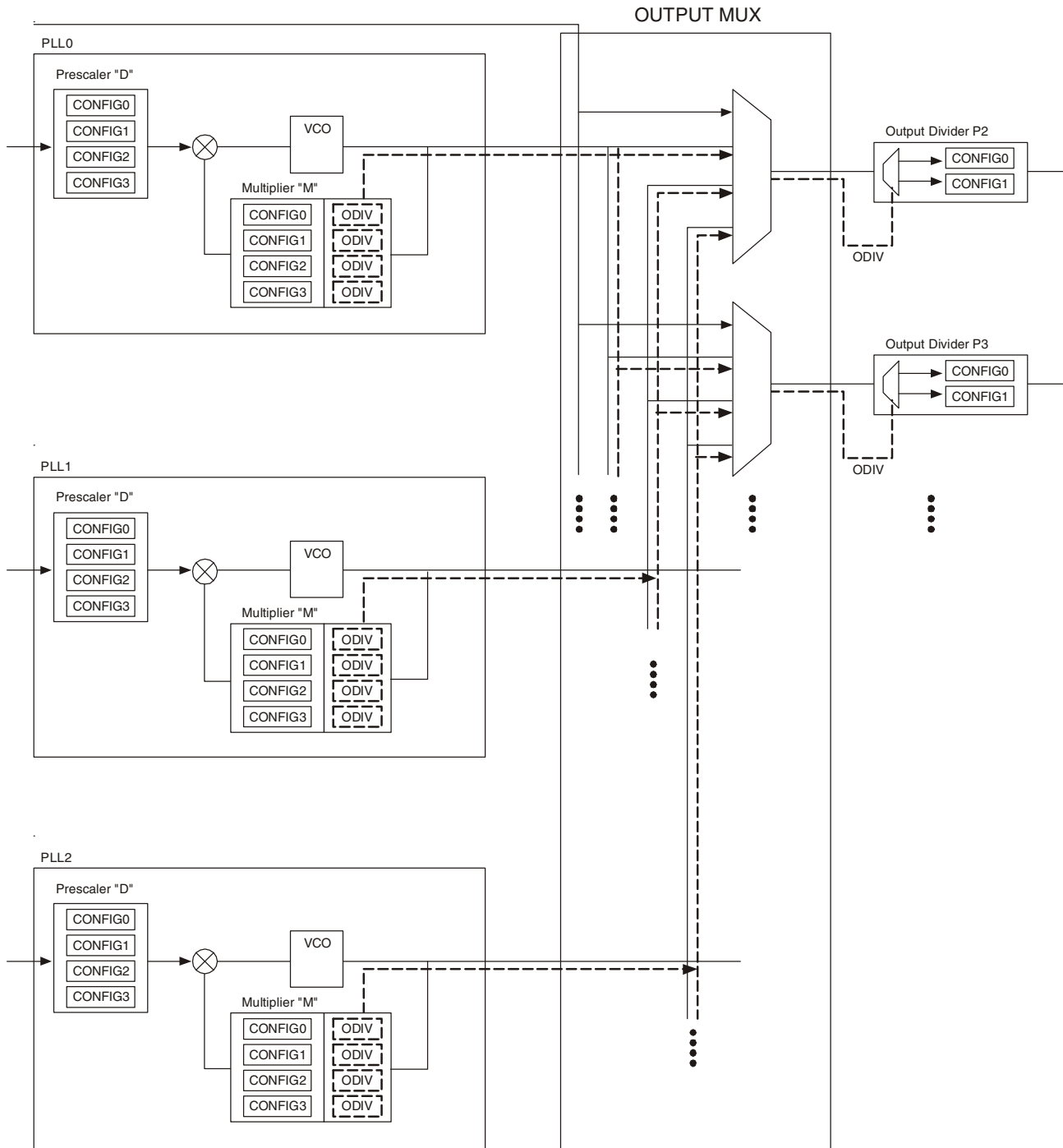
There are two external pins along with internal bits that control the enabling/disabling of the output banks. The two pins are the SHUTDOWN/OE pin and the GIN3/SUSPEND pin. The SHUTDOWN/OE pin can be programmed to function as an output enable or global shutdown. The polarity of the SHUTDOWN/OE signal pin can be programmed to be either active HIGH or LOW with the SP bit (0x1C). When SP is "0", the pin becomes active HIGH and when SP is "1", the pin becomes active LOW. The SH bit(0x1C) determines the function of the SHUTDOWN/OE signal pin. If SH is "1", the signal pin is SHUTDOWN and functions as a global shutdown. This will override the OEx (0x1C), OSx (0x1D), and PLLSx (0x1E) bits. If SH is "0", the signal pin is OE and functions as an enable/disable of the output banks. If used as an output enable/disable, each output bank can be individually programmed to be enabled or disabled by the OE pin by setting OEx bits to "1". If the OE signal pin is asserted, the output banks that has their corresponding OEx bit set to "1" will be disabled. The OEMx bits determine the outputs' disable state. When set to "0x" the outputs will be tristated. When set to "10", the outputs will be pulled low. When set to "11", the outputs will be pulled high. Inverted outputs will be parked in the opposite state. If the OEx bits are set to "0", the states of the corresponding output banks will not be impacted by the state of the OE pin. To individually enable/disable via programming instead of the OE pin, hard wire the OE pin to Vdd or GND (depending if it is active HIGH or LOW) as if to disable the outputs. Then toggle the OEx bits to either "0" to enable or "1" to disable.

When the chip is in shutdown, the outputs, the reference oscillator, and the I<sup>2</sup>C/JTAG pin are powered down. The outputs will be tristated and the I<sup>2</sup>C/JTAG pin will be set to MFC mode (MID level). Programming will not be allowed. The GINx pins and clock inputs remain operational. The PLL is not disabled. The SHUTDOWN pin must be reasserted in order to program the part or to resume operation.

The GIN3/SUSPEND pin, when used as a SUSPEND function, can be used to power down the PLL and/or output banks.. Each output bank can be individually programmed to be enabled or disabled by the SUSPEND signal pin by setting the OSx bits to "1". If the SUSPEND signal pin is asserted, the output banks that has their corresponding OSx bit set to "1" will be powered down and outputs tristated. If the OSx bits are set to "0", the states of the corresponding output banks will not be impacted by the state of the SUSPEND pin. There is also an option to suspend individual PLLs by setting the PLLSx bits (0x1E) to "1". This will associate the PLL to the SUSPEND pin. When the pin is asserted, the corresponding PLLs will be powered down. It will not only power down the PLL but also any output bank associated with it. The PLLSx bits will override the OSx bits.

In the event of a PLL suspend, the PLL must achieve lock again after it has been re-enabled, In the event of a global shutdown, the PLL does not have to re-acquire lock since it is not disabled.

## MANUAL FREQUENCY CONTROL (MFC) BLOCK DIAGRAM



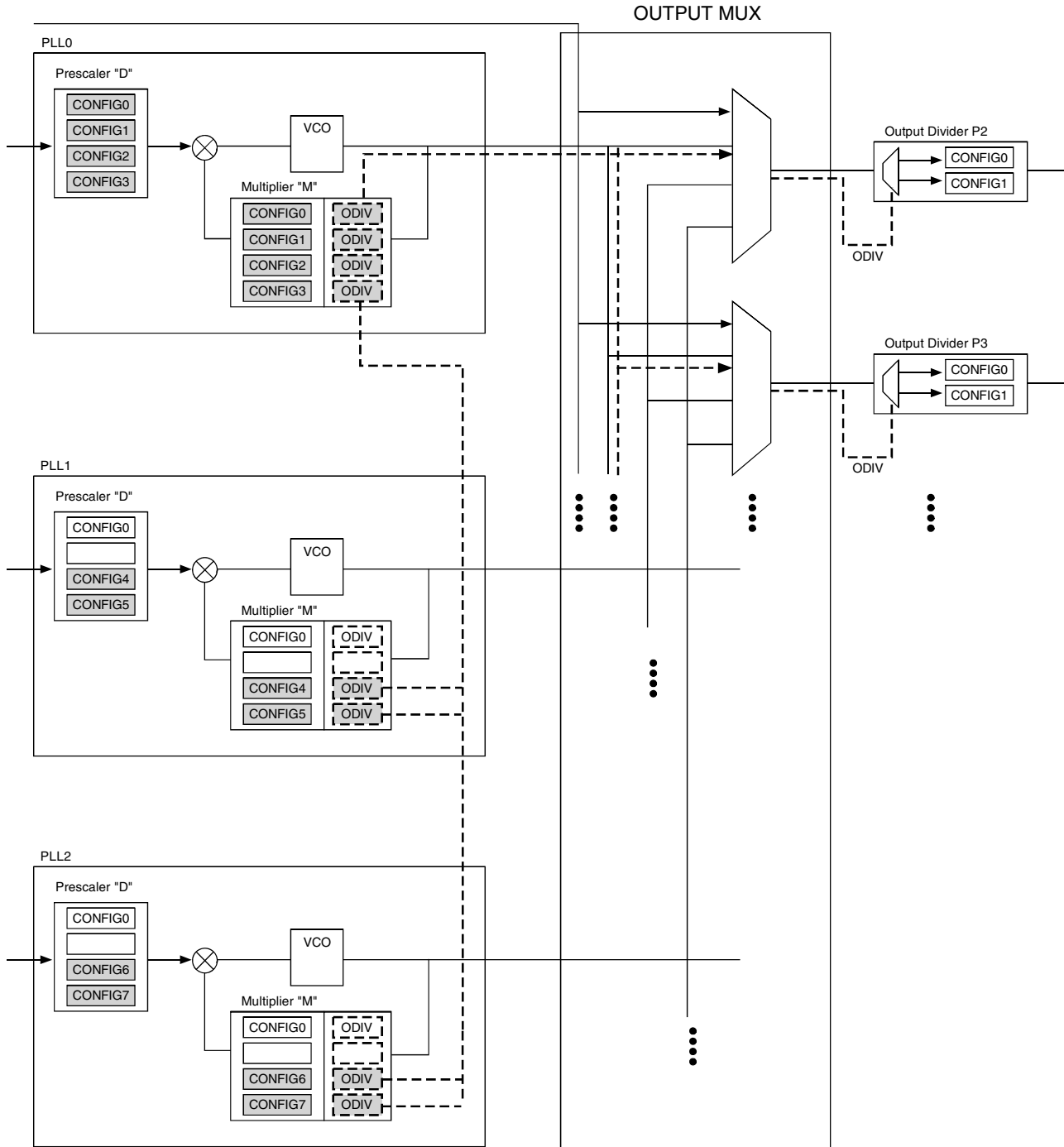
MFC = 0

### NOTES:

This illustration shows how the configurations are arranged for each PLL. There is an ODIV bit associated with each of the four configurations.

- GIN0 and GIN1 control four configurations from PLL0.
- GIN2 and GIN3 control four configurations from PLL1.
- GIN4 and GIN4 control four configurations from PLL2.
- ODIV from each configuration determines the selection of two Output Divider Px Configurations.

## MANUAL FREQUENCY CONTROL (MFC) BLOCK DIAGRAM



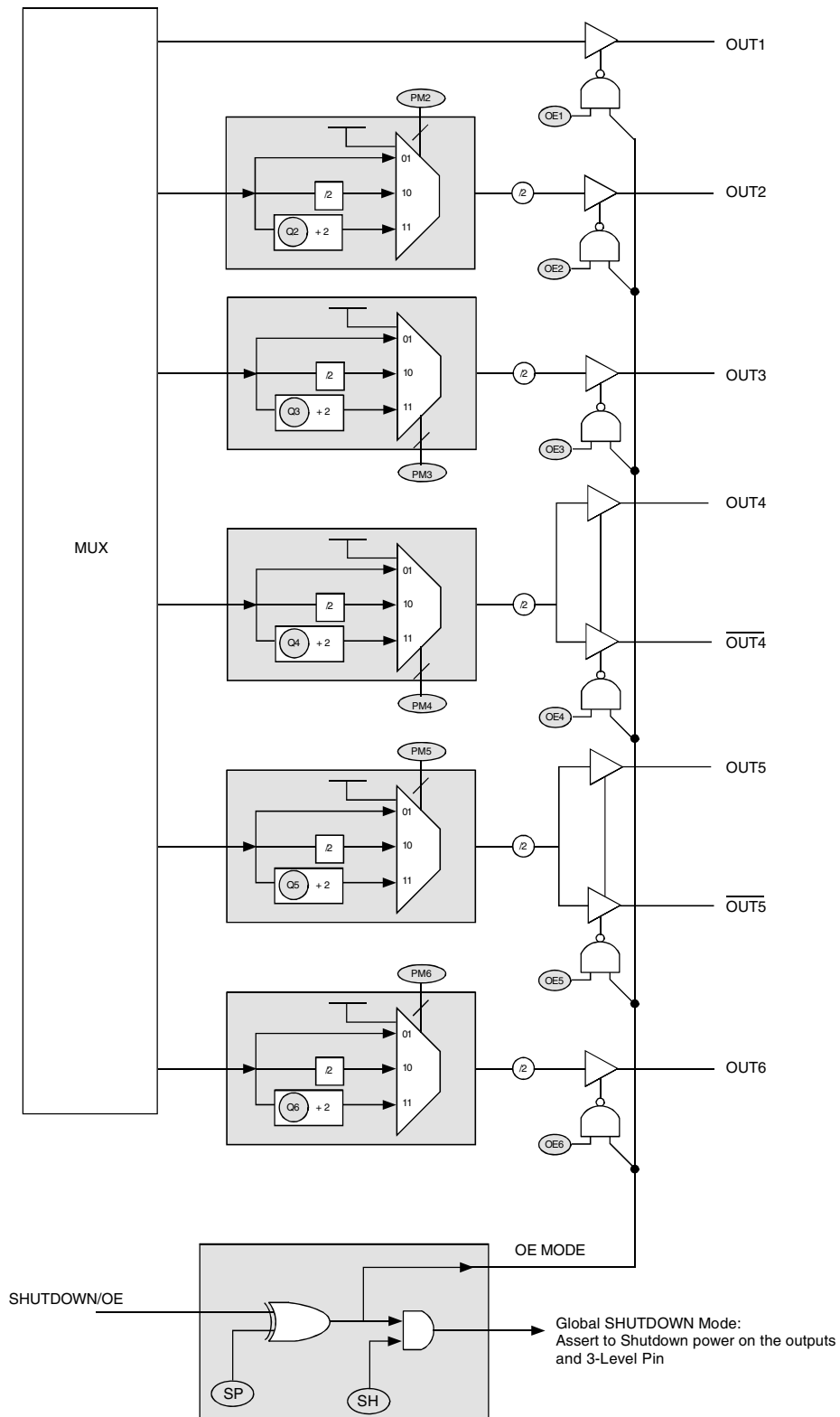
MFC = 1

### NOTES:

This illustration shows how the configurations are arranged for PLL0. Register location for Config\_4 and Config\_5 are taken from PLL1, and Config\_6 and Config\_7 are taken from PLL2. There is an ODIV bit associated with each of the configurations.

- GIN0, GIN1, and GIN2 control eight shaded configurations for PLL0.
- ODIV from each configuration determines the selection of two Output Divider Px Configurations.

### BLOCK DIAGRAM FOR SHUTDOWN/OE CONTROL SIGNAL



**NOTE:**  
This illustration shows the internal logic behind the SHUTDOWN/OE pin and the bits associated with it.

## POWER UP AND POWER SAVING FEATURES

If a global shutdown is enabled, SHUTDOWN pin asserted, most of the chip except for the PLLs will be powered down. In order to have a complete power down of the chip, the PLLs must be powered down via the SUSPEND function or by setting the pre-scaler bits to '0x00' and disable the internal GINx signals via the enable bits at memory address 0x05. Note that the register bits will not lose their state in the event of a chip power-down. The only possibility that the register bits will lose their state is if the part was power-cycled. After coming out of shutdown mode, the PLLs will require time to relock.

During power up, the values of GIN4, GIN3, GIN2, GIN1 and GIN0 will be latched and used for PLL configuration selection, regardless of the state of the I<sup>2</sup>C/JTAG pin and GINx being disabled via the GINENx bits. GIN5 will have an internal state of LOW. The GIN pins should be held LOW during power up to select configuration0 as default. The output levels will be at an undefined state during power up.

The post-divider should never be disabled via PM bits after power up, or else it will render the output bank completely non-functional during normal operation, (unless the output bank itself will not be used at all).

During power up, the V<sub>DD</sub> ramp must be monotonic.

## LOSS OF LOCK AND INPUT CLOCK

The device employs a loss of lock and loss of input clock detection circuitry. The GOUT0/LOSS\_LOCK and GOUT1/LOSS\_CLKIN are the outputs that indicate such failures. LOSS\_LOCK signal will be asserted if any of the three powered up PLLs loses frequency lock for any event other than PLL shutdown. Lock is determined by checking that the reference and feedback clocks are within 1/2 period of each other. Loss\_LOCK signal may be falsely asserted when

- Spread Spectrum is turned on for any of the PLLs
- Fractional divider is used for any of the PLLs
- the reference and feedback clocks are not within 1/2 period of each other.

LOSS\_CLKIN is asserted when the currently selected clock is lost or is asserted when both clocks are lost. In the event of the selected clock being absent up on power up, the loss of the selected clock detection circuitry will reference an internal oscillator. LOSS\_LOCK and LOSS\_CLKIN cannot be used as reliable inputs to other devices.

## SWITCHOVER MODES

The IDT5V9885T features redundant clock inputs which supports both Automatic and Manual switchover mode. These two modes are determined by the configuration bits, SM (0x34). The primary clock source can be programmed, via the PRIMCLK bit, to be either XTALIN/REFIN or CLKIN, which is determined by the PRIMCLK bit. The other clock source input will be considered as the secondary source. This is more detailed in the 'REFERENCE CLOCK INPUT PINS AND SELECTION'. Note that the switchover modes are asynchronous. If the reference clocks are directly routed to OUTx with no phase relationship, short pulses can be generated during switchover. The automatic switchover mode will work only when the primary clock source is XTALIN/REFIN.

### MANUAL SWITCHOVER MODE

When SM[1:0] is "0x", the redundant inputs are in manual switchover mode. In this mode, CLK\_SEL pin is used to switch between the primary and secondary clock sources. As previously mentioned, the primary and secondary clock source setting is determined by the PRIMCLK bit. During the switchover, no glitches will occur at the output of the device, although there may be frequency and phase drift, depending on the exact phase and frequency relationship between the primary and secondary clocks. If GOUT1 is used as LOSS\_CLKIN, it indicates loss of primary clock.

### AUTOMATIC SWITCHOVER MODE

When SM[1:0] is "11", the redundant inputs are in automatic revertive switchover mode.

*Revertive*

The input clock selection will switch to the secondary clock source when there are no transitions on the primary clock source. LOSS\_CLKIN signals will be asserted. After a stable and valid primary clock source is present, the input clock selection will automatically switch back to the primary clock source and LOSS\_CLKIN signal will be deasserted. The CLK\_SEL pin can be left floating in this auto-revertive mode. Note that both clock inputs must be at the same frequency (within 1000 ppm) in order for the auto-revertive switchover to function properly. If both reference clocks are at different frequencies, the device will always remain on the primary clock unless it is absent for two secondary clock cycles.

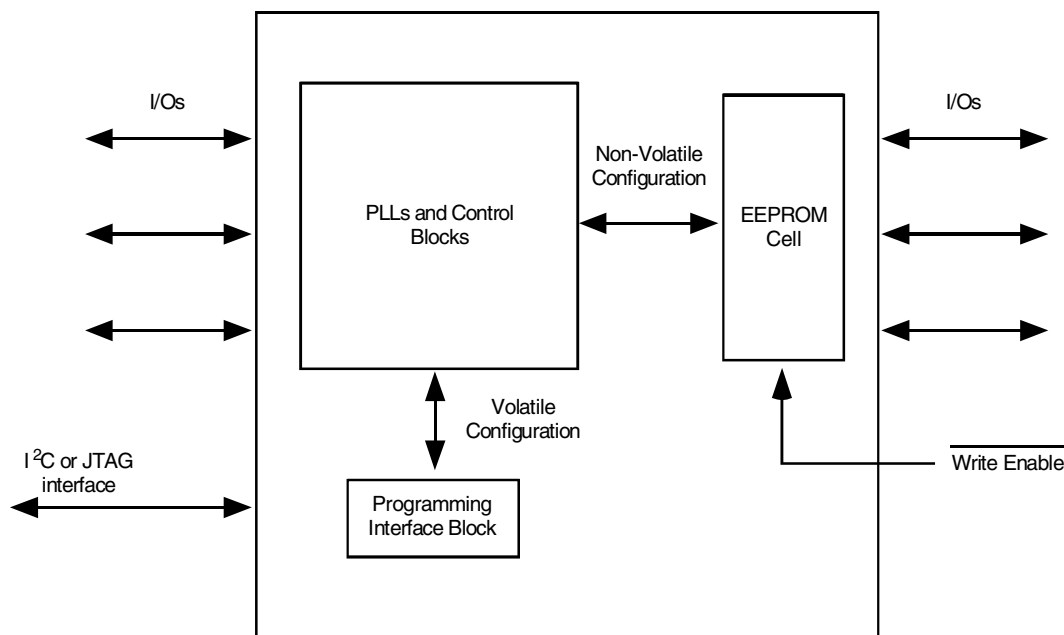
## CLOCK SWITCH MATRIX AND OUTPUTS

All three PLL outputs and the currently selected input clock source are routed into and through a clock matrix. The user is able to select which PLL output and clock source is routed to which output bank via the SRCx bits (0x34, 0x35). Each output bank has its own set of SRC bits. Refer to the RAM table for more information. Note that OUT1 will be based off the reference clock and the only output bank toggling under the default RAM bit settings.

Outputs 1, 2 and 3 are 3.3V LVTTTL. Outputs banks 4 and 5 can be 3.3V LVTTTL, LVPECL or LVDS. The LVDS and LVPECL selection is determined by the LVLx bits (0x54, 0x58). Each output bank has individual slew-rate control (SLEWx bits). Each output can be individually inverted (INVx bits); when using LVPECL or LVDS modes, one of the outputs in each LVPECL/LVDS pair should be inverted. All output banks except OUT1 have a programmable 10-bit post-divider (Qx bits) with two selectable divide configurations via the ODIVx bits.

There are four settings for the programmable slew rate, 0.7V/ns, 1.25V/ns, 2V/ns, and 2.75V/ns; this only applies to the 3.3V LVTTTL outputs. The differential outputs are not slew rate programmable in LVPECL or LVDS modes. SLEW4 and/or SLEW5 must be set to 2.75V/ns for stable output operation. For LVTTTL output frequency rates higher than 100MHz, a slew rate of 2V/ns or greater should be selected. The post-dividers can be disabled using the PMx bit, which is described in the PRE-SCALER, FEEDBACK-DIVIDER, AND POST-DIVIDER section. Each output can also be enabled/disabled, which is described in the 'SHUTDOWN/SUSPEND/ENABLE of OUTPUTS' section. Refer to the RAM table for all binary settings.

## HIGH LEVEL BLOCK DIAGRAM FOR CONFIGURATION SCHEME



NOTE: Diagram does not represent actual number of die on chip.

## PROGRAMMING THE DEVICE

I<sup>2</sup>C and JTAG may be used to program the 5V9885T. The I<sup>2</sup>C/JTAG pin selects the I<sup>2</sup>C when HIGH and JTAG when LOW. Note that the TRST pin needs to be LOW for I<sup>2</sup>C mode.

### Hardwired Parameters for the IDT5V9885T

JTAG identification number = 32'b0000\_0000001110101100\_00000110011\_1

Device (slave) address = 7'b1101010

ID Byte for the 5V9885T = 8'b00010000

### I<sup>2</sup>C PROGRAMMING

The 5V9885T is programmed through an I<sup>2</sup>C-Bus serial interface, and is an I<sup>2</sup>C slave device. The read and write transfer formats are supported. The first byte of data after a write frame to the correct slave address is interpreted as the register address; this address auto-increments after each byte written or read. The frame formats are shown below.

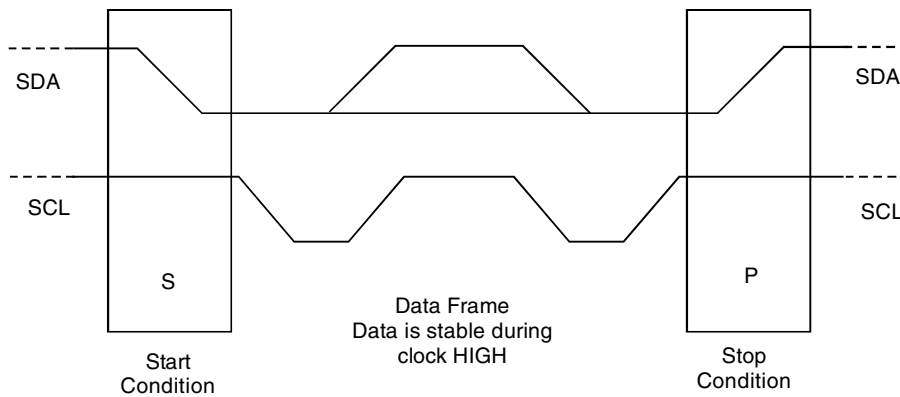
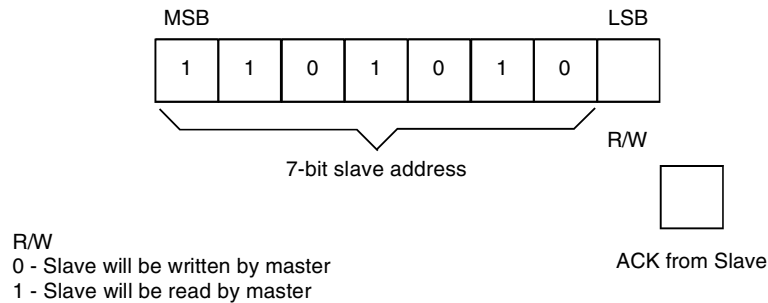


Figure 1: Framing

Each frame starts with a "Start Condition" and ends with an "End Condition". These are both generated by the Master device.



The first byte transmitted by the Master is the Slave Address followed by the R/W bit. The Slave acknowledges by sending a "1" bit.

Figure 2: First Byte Transmitted on I<sup>2</sup>C Bus

**EXTERNAL I<sup>2</sup>C INTERFACE CONDITION**

**KEY:**

- From Master to Slave
- From Master to Slave, but can be omitted if followed by the correct sequence  
Normally data transfer is terminated by a STOP condition generated by the Master. However, if the Master still wishes to communicate on the bus, it can generate a repeated START condition, and address another Slave address without first generating a STOP condition.
- From Slave to Master

**SYMBOLS:**

- ACK - Acknowledge (SDA LOW)
- NACK - Not Acknowledge (SDA HIGH)
- Sr - Repeated Start Condition
- S - START Condition
- P - STOP Condition

**PROGWRITE**

S	Address	R/W	ACK	Command Code	ACK	Register	ACK	Data	ACK	P
	7-bits	0	1-bit	8-bits: xxxxxx00	1-bit	8-bits	1-bit	8-bits	1-bit	

*Figure 3: Progwrite Command Frame*

Writes can continue as long as a Stop condition is not sent and each byte will increment the register address.

**PROGREAD**

Note: If the expected read command is not from the next higher register to the previous read or write command, then set a known "read" register address prior to a read operation by issuing the following command:

S	Address	R/W	ACK	Command Code	ACK	Register	ACK	P
	7-bits	0	1-bit	8-bits: xxxxxx00	1-bit	8-bits	1-bit	<input checked="" type="checkbox"/>

*Figure 4a: Prior to Progread Command Set Register Address*

The user can ignore the STOP condition above and use a repeated START condition instead, straight after the slave acknowledgement bit (i.e., followed by the Progread command):

Sr	Address	R/W	ACK	ID Byte	ACK	Data_1	ACK	Data_2	ACK	Data_last	NACK	P
	7-bits	1	1-bit	8 bits	1-bit	8-bits	1-bit	8-bits	1-bit	8-bits	1-bit	

*Figure 4b: Progread Command Frame*

Note: Figure 4b above by itself is the Progread command format. The ID byte for the 5V9885T is 10hex. Each byte recieved increments the register address.

**PROGSAVE**

S	Address	R/W	ACK	Command Code	ACK	P
	7-bits	0	1-bit	8-bits:xxxxxx01	1-bit	

**NOTE:**  
 PROGWRITE is for writing to the 5V9885T registers.  
 PROGREAD is for reading the 5V9885T registers.  
 PROGSAVE is for saving all the contents of the 5V9885T registers to the EEPROM.  
 PROGRESTORE is for loading the entire EEPROM contents to the 5V9885T registers.

**PROGRESTORE**

S	Address	R/W	ACK	Command Code	ACK	P
	7-bits	0	1-bit	8-bits:xxxxxx10	1-bit	

**JTAG INTERFACE**

In addition to the IEEE 1149.1 instructions EXTEST, SAMPLE/PRELOAD, CLAMP, HIGH-Z and BYPASS, the 5V9885T allows access to internal programming registers using the REGADDR (set register address), REGDATAR (read register) and REGDATW (write register instructions). Data is always accessed by byte, and the register address increments after each read or write. The full instruction set follows. The IDT5V9885T will be updating the registers during programming.

The JTAG TAP controller can be reset in one of four ways:

- 1) Power up in JTAG mode
- 2) Power up in I<sup>2</sup>C mode and then go into JTAG mode, or go out of and back into JTAG mode with the I<sup>2</sup>C/JTAG pin
- 3) Apply TRST while in JTAG mode
- 4) Apply five rising edges of TCK with TMS high while in JTAG mode

**JTAG INSTRUCTION REGISTER DESCRIPTION**

IR (3)	IR (2)	IR (1)	IR (0)	Instructions
0	0	0	0	EXTEST <sup>(1)</sup>
0	0	0	1	SAMPLE/PRELOAD <sup>(1)</sup>
0	0	1	0	IDCODE <sup>(1)</sup>
0	0	1	1	REGADDR <sup>(2)</sup>
0	1	0	0	REGDATAW / PROGWRITE <sup>(3)</sup>
0	1	0	1	REGDATAR / PROGREAD <sup>(4)</sup>
0	1	1	0	PROGSAVE <sup>(5)</sup>
0	1	1	1	PROGRESTORE <sup>(6)</sup>
1	0	0	0	CLAMP <sup>(1)</sup>
1	0	0	1	HIGHZ <sup>(1,7)</sup>
1	1	1	1	BYPASS <sup>(1)</sup>

- NOTES:**
1. IEEE 1149.1 definition
  2. REGADDR is for setting a specific 5V9885T register address.
  3. REGDATAW/PROGWRITE is for writing to the 5V9885T registers.
  4. REGDATAR/PROGREAD is for reading the 5V9885T registers.
  5. PROGSAVE is for saving all the contents of the 5V9885T registers to the EEPROM.
  6. PROGRESTORE is for loading the entire EEPROM contents to the 5V9885T registers.
  7. The OEMs bits for OUT1-6 must be set for tri-state when using the HIGHZ instruction

**EEPROM INTERFACE**

The IDT5V9885T can also store its configuration in an internal EEPROM. The contents of the device's internal programming registers can be saved to the EEPROM by issuing a save instruction (ProgSave) and can be loaded back to the internal programming registers by issuing a restore instruction (ProgRestore).

To initiate a save or restore using I<sup>2</sup>C, only two bytes are transferred. The Device Address is issued with the read/write bit set to "0", followed by the appropriate command code. The save or restore instruction executes after the STOP condition is issued by the Master, during which time the IDT5V9885T will not generate Acknowledge bits. The IDT5V9885T will acknowledge the instructions after it has completed execution of them. During that time, the I<sup>2</sup>C bus should be interpreted as busy by all other users of the bus.

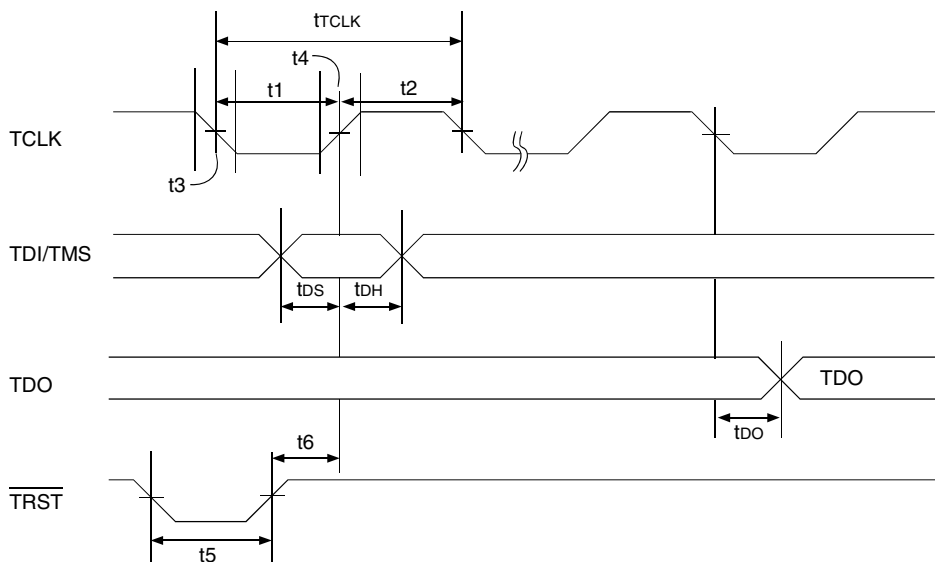
Using JTAG, the ProgSave and ProgRestore instructions selects the BYPASS register path for shifting the data from TDI to TDO during the data register scanning. During the execution of a ProgSave or ProgRestore instruction, the IDT5V9885T will not accept a new programming instruction (read, write, save, or restore). All non-programming JTAG instructions will function properly, but the user should wait until the save or restore is complete before issuing a new programming instruction. If a new programming instruction is issued before the save or restore completes, the new instruction is ignored, and the BYPASS register path remains in effect for shifting data from TDI to TDO during data register scanning.

The time it takes for the save (T<sub>SAVE</sub>) and restore (T<sub>RESTORE</sub>) instructions to complete is:

T<sub>SAVE</sub> = 100ms max, T<sub>RESTORE</sub> = 10 ms max

In order for the save and restore instructions to function properly, the IDT5V9885T must not be in shutdown mode (SHUTDOWN pin asserted). In the event of an interrupt of some sort such as a power down of the part in the middle of a save or restore operation, the contents to or from the EEPROM will be partially loaded, and a CRC error will be generated. The CERR bit (0x81) will be asserted to indicate that an error has occurred. The LOSS\_LOCK signal will also be asserted.

On power-up of the IDT5V9885T, an automatic restore is performed to load the EEPROM contents into the internal programming registers. The auto-restore will not function properly if the device is in shutdown mode (SHUTDOWN pin asserted). The IDT5V9885T will be ready to accept a programming instruction once it acknowledges its 7-bit I<sup>2</sup>C address.



Standard JTAG Timing

**NOTE:**

- t1 = tTCLKLOW
- t2 = tTCLKHIGH
- t3 = tTCLKFALL
- t4 = tTCLKRISE
- t5 = tRST (reset pulse width)
- t6 = tRSR (reset recovery)

**JTAG  
AC ELECTRICAL CHARACTERISTICS**

Symbol	Parameter	Min.	Max.	Units
tCLK	JTAG Clock Input Period	100	—	ns
tCLKHIGH	JTAG Clock HIGH	40	—	ns
tCLKLOW	JTAG Clock Low	40	—	ns
tCLKRISE	JTAG Clock Rise Time	—	5 <sup>(1)</sup>	ns
tCLKFALL	JTAG Clock Fall Time	—	5 <sup>(1)</sup>	ns
tRST	JTAG Reset	50	—	ns
tRSR	JTAG Reset Recovery	50	—	ns

**NOTE:**

- 1. Guaranteed by design.

**SYSTEM INTERFACE PARAMETERS**

Symbol	Parameter	Min.	Max.	Units
tDO	Data Output <sup>(1)</sup>	—	20	ns
tDOH	Data Output Hold <sup>(1)</sup>	0	—	ns
tDS	Data Input, tRISE = 3ns	10	—	ns
tDH	Data Input, tFALL = 3ns	10	—	ns

**NOTE:**

- 1. 50pF loading on external output signals.