Using the TMX320VC5510 APLL

Clay Turner *C5000 Applications*

ABSTRACT

Certain revisions of the prototype TMX320VC5510 DSP (hereafter referred to as VC5510) support an analog phase-locked loop (APLL) clock generator. This document provides a description of the programming and use of this APLL, including the control register description, how to program the APLL for the desired clock frequency, and how to determine, through software, whether the device contains a digital PLL (DPLL) or an APLL.

Contents

1 Identification of the DPLL/APLL Presence Through Software 2
2 Programming the APLL Clock Mode Register 3
3 Reference 5
Appendix A Example Code to Identify the PLL Type 6

List of Figures

Figure 1 DIE_ID Register 2
Figure 2 APLL Clock Mode Register 3

List of Tables

Table 1 TMX320VC5510 Prototype Revisions 2
Table 2 APLL CLKMD Register Functions 4
Table 3 APLL Multiply/Divide Factors 4
Table 4 APLL CLKMD Programming Values 5

1 Identification of the DPLL/APLL Presence Through Software

Certain revisions of the prototype TMX320VC5510 DSP support an analog phase-locked loop (APLL) clock generator. Each of the prototype silicon revisions and their descriptions are shown in Table 1 below.

Table 1. TMX320VC5510 Prototype Revisions

Revision Number	PLL Type	REV_ID[15:0] (hex)	DIE_ID[48:46] (binary)
1.0/1.0A	DPLL	8050	011
1.2	APLL	8050	101

The DPLL and APLL have different clock-mode register structures and, consequently, are programmed differently. As a result, there may be a need to identify the device revision (and the PLL type) through software. This can be achieved by examining the contents of the DIE_ID and REV_ID registers.

The REV_ID register (port address 0x3804) contains 0x8050 for all VC5510 revision 1.x material. Four DIE_ID registers (port addresses 0x3800–0x3803) function as a virtual 64-bit register containing coded manufacturing information about the device, as shown below in Figure 1.

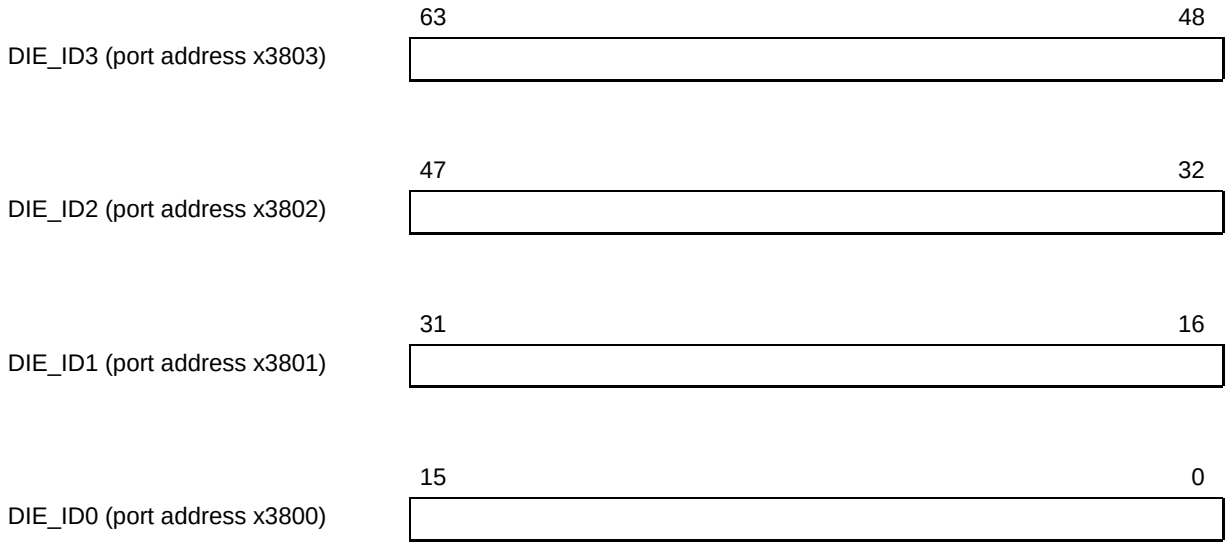

Figure 1. DIE_ID Register

Bits 48-46 of the DIE_ID register can be examined to determine whether the device contains an APLL or a DPLL.

For revision 1.0 / 1.0A material (DPLL):

DIE_ID [63:48] (port address 0x3803) = xxxx xxxx xxxx xxx0 (binary)

DIE_ID [47:32] (port address 0x3802) = 11xx xxxx xxxx xxxx (binary)

REV_ID (port address 0x3804) = 8050 (hex)

For revision 1.2 material (APLL):

DIE_ID [63:48] (port address 0x3803) = xxxx xxxx xxxx xxx1 (binary)

DIE_ID [47:32] (port address 0x3802) = 01xx xxxx xxxx xxxx (binary)

REV_ID (port address 0x3804) = 8050 (hex)

Example code to identify the PLL type is included in Appendix A.

NOTE: The DIE_ID register requires at least 3 cycles to occur on the TCK signal following reset. If the DIE_ID register is read before this conditions occurs, the value read may be incorrect. When the emulator pod is disconnected, the TCK signal is not normally driven. In this state, a clock can either be connected to TCK (such as CLKIN), or a GPIO pin can be used to toggle the TCK prior to reading the DIE_ID register.

2 Programming the APLL Clock Mode Register

For revisions supporting the DPLL, a description of the structure and use of the DPLL is available in the *TMS320C55x DSP Peripherals Reference Guide* (SPRU317).

Similar to the DPLL, the APLL is controlled by a single clock mode register (CLKMD) shown below in Figure 2. The APLL clock mode register is located at port address 0x1C00.

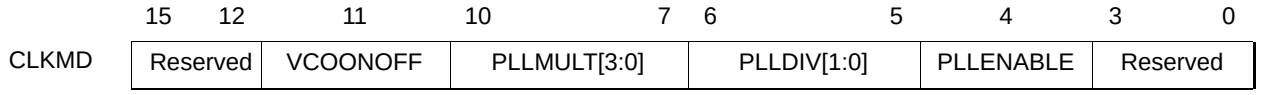

Figure 2. APLL Clock Mode Register

The operation of each of the fields in the APLL CLKMD register is given below in Table 2.

Table 2. APLL CLKMD Register Functions

Bit(s)	Field	Symval	Description	
15–12	Reserved		Reserved. These bits can always be written as 0.	
11	VCOONOFF		APLL VCO† Control	
		OFF	0	VCO is disabled.
		ON	1	VCO is enabled.
10–7	PLLMULT[3:0]	OF(value)	APLL Multiply Factor	
			0–15	PLLMULT[3:0] +1 = K (multiply factor)
6–5	PLLDIV[1:0]	OF(value)		APLL Divide Factor
			00, 01	Bypass Mode
			10	Divide-by-1
			11	Divide-by-2 or Divide-by-4 (see Table 3).
4	PLLENABLE		APLL Enable	
		OFF	0	PLLENABLE = 0 APLL is disabled (bypass mode).
		ON	1	PLLENABLE = 1 APLL is enabled.
3–0	Reserved		Reserved. These bits can always be written as 0.	

† Voltage-controlled oscillator

For the APLL to operate, both VCOONOFF and PLLENABLE must be enabled (logic 1). The APLL will generate a clock frequency according to Table 3. After the CLKMD register is written, the APLL will lock within 50 uS.

Table 3. APLL Multiply/Divide Factors

PLLDIV[1]	PLLDIV[0]	K (= PLLMULT[3:0] + 1)	APLL Frequency Generated
0	X	X	Input frequency – Bypass Mode
1	0	1 to 15	(Input frequency) x (K)
1	0	16	(Input frequency) x (1)
1	1	ODD	(Input frequency) x (K/2)
1	1	EVEN	(Input frequency) x ((K–1)/4)

The desired frequency modification is generated by programming PLLDIV[1:0] and PLLMULT[3:0] to the appropriate values. For easy reference, some common frequency multiplications are given in Table 4 below with the appropriate value for programming the CLKMD register.

Table 4. APLL CLKMD Programming Values

To generate this frequency…	Program the APLL CLKMD Register With This Value	To generate this frequency…	Program the APLL CLKMD Register With This Value
Input frequency x 1	0FD0	Input frequency x 0.5	0870
Input frequency x 2	08D0	Input frequency x 1.5	0970
Input frequency x 3	0950	Input frequency x 2.5	0A70
Input frequency x 4	09D0	Input frequency x 3.5	0B70
Input frequency x 5	0A50	Input frequency x 4.5	0C70
Input frequency x 6	0AD0	Input frequency x 5.5	0D70
Input frequency x 7	0B50	Input frequency x 6.5	0E70
Input frequency x 8	0BD0	Input frequency x 7.5	0F70
Input frequency x 9	0C50		
Input frequency x 10	0CD0	Input frequency x 0.25	08F0
Input frequency x 11	0D50	Input frequency x 0.75	09F0
Input frequency x 12	0DD0	Input frequency x 1.25	0AF0
Input frequency x 13	0E50	Input frequency x 1.75	0BF0
Input frequency x 14	0ED0	Input frequency x 2.25	0CF0
Input frequency x 15	0F50	Input frequency x 2.75	0DF0
Input frequency x 15	0F50	Input frequency x 3.25	0EF0

3 Reference

1. *TMS320C55x DSP Peripherals Reference Guide* (SPRU317).

Appendix A Example Code to Identify the PLL Type

```
DIE_ID3  .set  03803h
DIE_ID2  .set  03802h

  mov port(#DIE_ID3)<<#16,ac0       ;load ac0 with upper 32 bits of DIE_ID reg's
  or  port(#DIE_ID2),ac0
  sftl   ac0,#–14,ac0               ;shift the 3 bits to be tested down to the LSB
  and #00007h,ac0                   ;mask off all other bits
  add #0h, ac1                      ;bug work-around for CPU_38 on rev 1.x Si

  bcc DPLL_config, AC0==#3 ;branch to DPLL configuration if bits = 011
  bcc APLL_config, AC0==#5 ;branch to APLL configuration if bits = 101
```

The label *DPLL_config* is the location of the code to configure the clock generator for a DPLL.

The label *APLL_config* is the location of the code to configure the clock generator for an APLL.

NOTES: 1. The Compare-and-Branch instruction (BCC) used in this example is a relative branch to a location within +/– 127 byte addresses from the current PC location. If the distance to branch to the PLL configuration code is farther than +/– 127 bytes, another branch syntax must be used.

2. The DIE_ID register requires at least 3 cycles to occur on the TCK signal following reset. If the DIE_ID register is read before this conditions occurs the value read may be incorrect. When the emulator pod is disconnected, the TCK signal is not normally driven. In this state, a clock can either be connected to TCK (such as CLKIN), or a GPIO pin can be used to toggle the TCK prior to reading the DIE_ID register.

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third–party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Mailing Address:

Texas Instruments
Post Office Box 655303
Dallas, Texas 75265