Interfacing With the HSP50210 DCL Lock Detector

Application Note | May 1999 | AN9656

Introduction

This Application Note will provide additional information on interfacing the DCL Lock Detector circuitry. The primary purpose of this Application Note is to answer some commonly asked questions about the DCL Lock detector.

Question #1: What is the process for reading values from the Lock Detector in the HSP50210?

Question #2: What is the process for writing values to the Lock Detector in the HSP50210?

Question #3: What is the relationship between the processor clock and CLK of the DCL? Do they need to be synchronized?

Answers to Questions 1 and 2:

The first two questions require a more detailed description of the hardware DCL processor to Lock Detector interface. Figure 1 provides the necessary detail for understanding the location of the registers, the register addresses, the partitioning into read and write registers and the destination addresses for the write C0-7 contents. It may also be helpful to have a copy of the DQT/DCL detailed Block Diagram on hand to understand how this Lock Detection Circuitry fits into the larger picture of the HSP50210.

Answer to Question 3:

There is no requirement for the processor clock to be synchronous to the CLK driving the DCL. On our evaluation board, there are two crystal oscillators: a 40MHz one for CLK and an 8MHz one for the processor clock. All of the Timing Diagrams shown in the HSP50210 Data Sheet (Figures 19-22) are only representative since making the processor an asynchronous clock that is much slower than CLK complicates conveying the waveform idea.

Two things to remember:

1. Compliance to the timing detailed in Figure 25 of the Data Sheet is required at all times.
2. The clock of the processor is usually significantly slower than CLK, allowing for a second CLK edge to comply with the waveform requirements when the compliance is not met with the first CLK edge.

Lock Detection Control Modes

There are two control modes in which you could choose to operate the DCL acquisition and tracking circuitry:

1. The Internal State Machine Control mode.
2. The Microprocessor Control mode.

Selection of the Lock Detector mode is made in Destination Address = 23, bit 4.

0 = Processor Control mode; 1 = Internal State Machine Control mode

The requirements for interfacing with the Lock Detector are different for each mode. The Internal State Machine mode allows the internal circuitry of the DCL to control the declaration of lock function. The Microprocessor Control mode allows a microprocessor access to all of the necessary parameters to perform the “Declaration of Lock” function. Additionally, the LKINT (Lock Detector Interrupt - pin 52) signal can be used to streamline the “read the lock detector” sequence. On the evaluation board, a jumper via JP6 pin 1 and 2 connects the LKDETINT signal to the on-board microprocessor IRQB input for implementing this function.

The “Declaration of Lock” function involves integration of Carrier Phase Error, False Lock (or Frequency Error) and Gain Error. Because of this integration process, Lock Detection will require that these error accumulators be initialized (or reset), started, monitored, halted (or identified as halted) and read. The sequence can then be repeated. A dwell counter and an integration counter control when and how long the error accumulators run. The dwell counter holds off the lock integration counter for a programmed number of integration cycles. The integration counter controls the number of Phase Error samples accumulated in the lock accumulator. These functional items are found in the Lock Detector block of the HSP50210 Block Diagram.

The Internal State Machine Mode

Figure 16 of the HSP50210 Data Sheet describes in detail the Internal State Machine implementation in the DCL. The processor control command sequence required for this mode is:

1. Initialize the Lock Detector; Reset the Integration Counter.

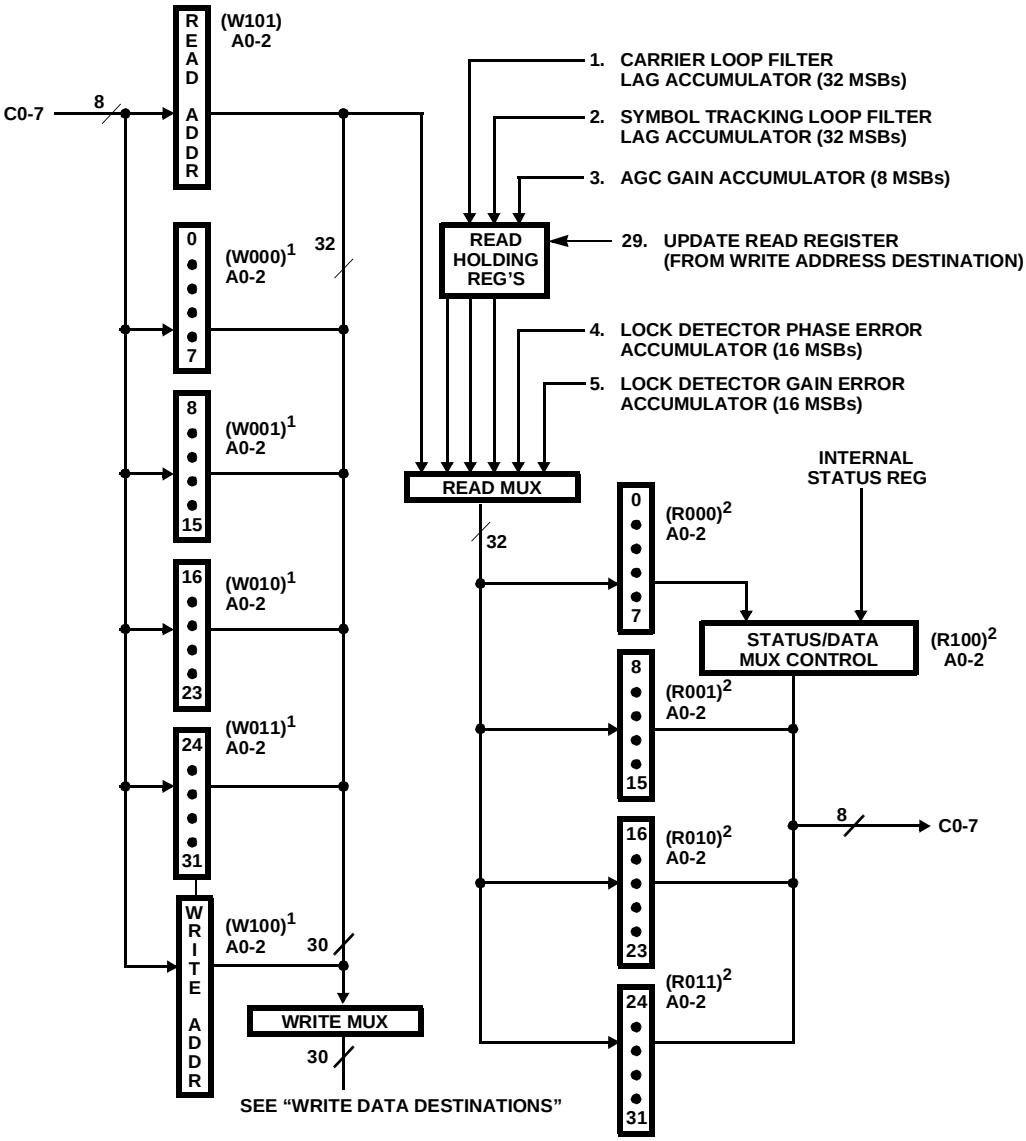

FIGURE 1. A CONCEPTUAL BLOCK DIAGRAM OF THE READ/WRITE MICROPROCESSOR INTERFACE

→	1.	DATA PATH CONFIGURATION
→	2.	POWER DETECT THRESHOLD REGISTER
→	3.	AGC LOOP PARAMETERS
→	4.	CARRIER PHASE ERROR DETECTOR
→	5.	FREQUENCY DETECTOR
→	6.	FREQUENCY ERROR DETECTOR
→	7.	CARRIER LOOP FILTER CONTROL #1
→	8.	CARRIER LOOP FILTER CONTROL #2
→	9.	CARRIER LOOP FILTER UPPER LIMIT
→	10.	CARRIER LOOP FILTER LOWER LIMIT
→	11.	CARRIER ACQUISITION LOOP FILTER GAINS
→	12.	CARRIER TRACKING LOOP FILTER GAINS
→	13.	FREQUENCY SEEP/AFC LOOP CONTROL
→	14.	CARRIER LAG ACCUMULATOR INITIALIZATION
→	15.	SYMBOL TRACKING LOOP CONFIGURATION
→	16.	SYMBOL TRACKING LOOP FILTER UPPER LIMIT
→	17.	SYMBOL TRACKING LOOP FILTER LOWER LIMIT
→	18.	SYMBOL TRACKING LOOP FILTER GAINS: ACQUISITION
→	19.	SYMBOL TRACKING LOOP FILTER GAINS: TRACKING
→	20.	SYMBOL TRACKING LOOP FILTER LAG ACCUMULATOR INITIALIZATION
→	21.	LOCK DETECTOR CONFIGURATION
→	22.	LOCAL ACCUMULATOR PRE-LOADS
→	23.	FALSE LOCK ACCUMULATOR PRE-LOADS
→	24.	ACQ/TRK CONTROL
→	25.	HALT LOCK DETECTOR FOR READING
→	26.	RESTART LOCK DETECTOR
→	27.	SOFT DECISION SLICER CONFIGURATION
→	28.	SERIAL OUTPUT CONFIGURATION OUTPUT SELECTOR CONFIGURATION
→	29.	UPDATE READ REGISTER → TO PAGE 1
→	30.	INITIALIZE LOCK DETECTOR (μP CONTROL MODE)

NOTES:

1. The prefix W indicates that a rising edge of $\overline{WR}$ is required to load the contents (C0-7) of the addressed requestor (A0-2).
2. The prefix R indicates that a low assertion of $\overline{RD}$ is required to load the contents (C0-7) of the addressed register (A0-2. The contents will be available on the first rising edge of CLK after $\overline{RD}$ is asserted low.

WRITE DATA DESTINATIONS

2. Start the Lock Detector.
3. Stop the Lock Detector.
4. Monitor the Lock Status Bit.

Note that on the write commands (steps 1-3), the C0-7 values will become valid processor clock edges and on the read commands (step 4), the C0-7 values will become valid on the CLK rising edges. The $\overline{RD}$, $\overline{WR}$ and A0-2 signals are always synchronous to the processor clock edges.

The Processor Control Mode

The Processor Control Mode allows a designer the ability to implement a design unique “Lock Declaration” function. The primary difference between this mode and the State Machine mode is in the way the error accumulators work. In the Processor Mode, the error integration process stops after every integration, rather than continuing into another integration cycle. This allows the processor to take advantage of the LKINT signal to streamline the process of knowing when the error accumulators are finished an integration cycle and can be read. Thus, there are two ways to operate in the Processor Mode:

1. Using the LKINT signal as an interrupt for recognizing the end of an integration cycle.
2. Reading the internal status bit SR-7 to determine the integration cycle state.

LKINT Interrupt Driven Operation

With the LKINT Interrupt Driven technique, the processor command sequence is:

1. Initialize the Lock Detector; Reset the Integration Counter.
2. Start the Lock Detector.
3. Set the Read Address for a Phase Error or Gain Error Read.
4. Read the Lock Detector Error MSByte.
5. Read the Lock Detector Error LSByte.
6. Processor uses Application Specific Lock Declaration Logic on Values (Note).
7. Restart the Lock Detector.

NOTE: This step does not involve writing to the DCL.

Figure 22 of the HSP50210 Data Sheet details the timing waveforms required to implement this processor command sequence. Note that on the write commands (steps 1-3, 7), the C0-7 values will become valid on the processor clock edges and on the read commands (steps 4-5), the C0-7 values will become valid on the CLK rising edges. The $\overline{RD}$, $\overline{WR}$ and A0-2 signals are always synchronous to the processor clock edges.

Processor Read Status Driven Operation

With the Processor Read Status Driven technique, the processor command sequence is:

1. Initialize the Lock Detector; Reset the Integration Counter.
2. Start the Lock Detector.
3. Halt the Lock Detector at end of cycle.
4. Set the Read Address for Phase Error or Gain Error Read.
5. Read (Monitor) the Internal Status (SR-7).
6. Read the Lock Detector Error MSByte.
7. Read the Lock Detector Error LSByte.
8. Processor uses Application Specific Lock Declaration Logic on Values (Note).
9. Monitor the Lock Status Bit.
10. Restart the Lock Detector.

NOTE: This step does not involve writing to the DCL.

Figure 21 of the HSP50210 Data Sheet details the timing waveforms required to implement this processor command sequence. Note that on the write commands (steps 1-4, 10), the C0-7 values will become valid on the processor clock edges and on the read commands (steps 5-7, 9), the C0-7 values will become valid on the CLK rising edges. The $\overline{RD}$, $\overline{WR}$ and A0-2 signals are always synchronous to the processor clock edges.