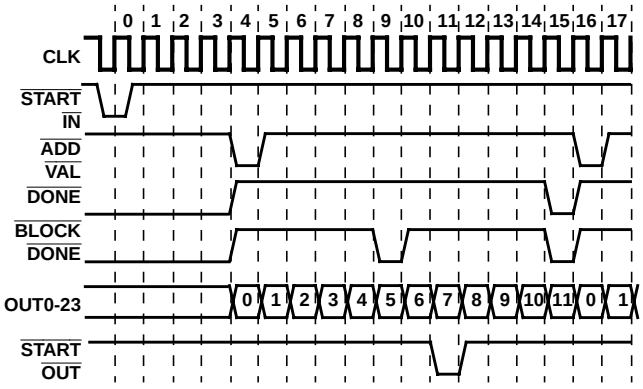
The Timing Diagram in Figure 1 shows the timing relationship between the various output signals of the HSP45240 when the sequence generator is programmed for **One-Shot Mode with Restart** (see Sequence Generator Section of Data Sheet). In this example, the HSP45240 is configured to generate a sequence consisting of two address blocks. Each block is 6 addresses long, and the end of a block is denoted by the assertion of $\overline{\text{BLOCKDONE}}$. As the final address in the second block is generated, both $\overline{\text{DONE}}$ and $\overline{\text{BLOCKDONE}}$ are asserted to signal the end of the address sequence. On the next clock, a new address sequence is started (see assertion of $\overline{\text{ADDVAL}}$) because the Sequencer was configured to restart. In this mode the $\overline{\text{STARTOUT}}$ signal is asserted prior to the end of the address sequence for the synchronization of multiple HSP45240's.



NOTE: Asserting $\overline{\text{STARTIN}}$ after an addressing sequence has been started will cause the sequencer to restart from the beginning of the sequence.

FIGURE 1. SIGNAL RELATIONSHIPS FOR ONE-SHOT MODE WITH RESTART

The Timing Diagram in Figure 2 shows the timing relationship between the various output signals of the HSP45240 when the sequence generator is programmed for **One-Shot Mode without Restart** (see Sequence Generator Section of Data Sheet). As in the above example, the HSP45240 is configured to generate a sequence consisting of two address blocks. Each block is 6 addresses long, and the end of a block is denoted by the assertion of $\overline{\text{BLOCKDONE}}$. As the final address in the second block is generated, both $\overline{\text{DONE}}$ and $\overline{\text{BLOCKDONE}}$ are asserted and addressing is halted.

The Timing Diagram in Figure 3 shows the timing relationship between the various output signals of the HSP45240 when the sequence generator is internally started by writing the Sequencer "START" address (see Table 1 of Data Sheet). The output signals are shown with respect to the rising edge of $\overline{\text{WR}}$ responsible for the internal START. The address generation parameters are as above.

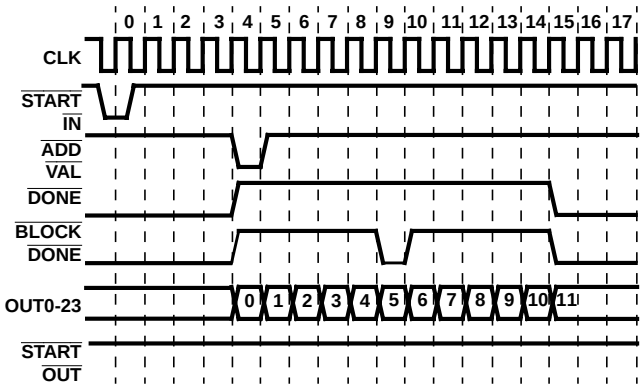


FIGURE 2. SIGNAL RELATIONSHIPS FOR ONE-SHOT MODE WITHOUT RESTART

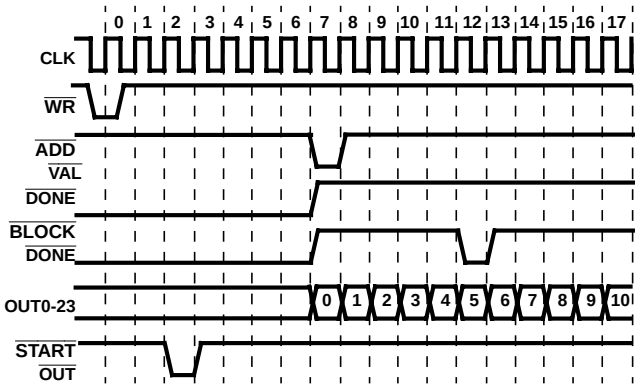


FIGURE 3. SIGNAL RELATIONSHIPS FOR INTERNALLY GENERATED START

DLYBLK Operation

Address generation can be halted by assertion of DLYBLK prior to the completion of an address block (Figures 4 and 5). Addressing will resume once DLYBLK is de-asserted. Since there is a pipeline delay between the assertion of DLYBLK at the pin and when it is internally active, DLYBLK must be asserted prior to the end of an address block. The pipeline delay associated with DLYBLK differs for halting address generation in mid-sequence and halting address generation after the final address block of a sequence.

For halting address generation in mid-sequence, DLYBLK must be asserted 3 clocks prior to the end of the addressing block as shown in Figure 4. In this example, DLYBLK is asserted for one clock cycle which delays the generation of the next address block by one clock. If addressing has been halted in mid-sequence, addressing will resume 4 clocks after de-asserting DLYBLK. Note: $\overline{\text{BLOCKDONE}}$ will be asserted and OUT0-23 will be held until addressing resumes.

1-888-INTERSIL or 321-724-7143 | Copyright © Intersil Corporation 1999

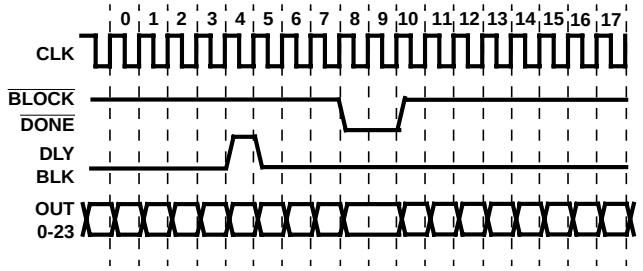


FIGURE 4. SIGNAL RELATIONSHIPS FOR A ONE CYCLE BLOCK DELAY IN MID SEQUENCE

For halting address generation after the final block of addresses in a sequence, DLYBLK must be asserted 4 clocks prior to the end of the addressing block as shown in Figure 5. In this example, DLYBLK is asserted for one clock cycle which delays the start of a new address sequence by one clock. The part is assumed to be configured for One-Shot Mode with Restart. Addressing will resume 5 clocks after de-asserting DLYBLK. Note: BLOCKDONE and DONE will be asserted and OUT0-23 will be held until addressing resumes. Also, STARTOUT will be asserted one clock after the assertion of DLYBLK.

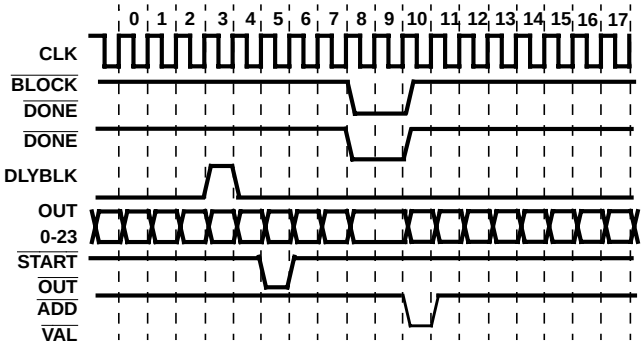


FIGURE 5. SIGNAL RELATIONSHIPS FOR A ONE CYCLE BLOCK DELAY AFTER FINAL BLOCK IN AN ADDRESSING SEQUENCE

STARTIN Operation

The STARTIN pin has two functions: first, it downloads the configuration data in the processor interface into the register bank that controls the operation of the part; second, it starts the address sequence using the updated configuration. When STARTIN is deasserted, the part continues on with the new sequence. Note that there are four stages of pipeline delay between the sequence generator and the output of the part; all of the output signals will continue on using the original sequence for those four clock cycles.

After the assertion of STARTIN, the first value in the sequence appears on the output after four pipeline delays. The part will remain in this state for the remainder of the time that STARTIN is low, and for four clocks after STARTIN returns high. This is shown in Figure 6, note that the old sequence ends at clock 3; the first address of the new sequence goes from clock 4 to clock 12; the second address of the new sequence appears on clock 13.

Asserting STARTIN in the middle of a sequence demonstrates the sequence restart function as described above. The internal count of the HSP45240 returns to the starting point (the value in the Start Address Register - not the Current Block Start Address Register) on the first rising edge of CLK that STARTIN is low. The first address of the sequence is output four clocks after the assertion of STARTIN. The Sequencer goes to the second address in the sequence when STARTIN goes away; this address appears on the output pins four clocks later. Sequencing continues based on the updated configuration. In Figure 7, the new sequence is started on clock 1; the old sequence will continue unaffected until clock 4, and the first address of the new sequence becomes valid on the outputs during clock 5.

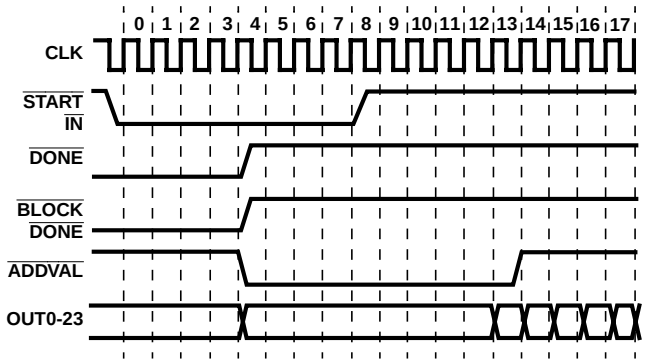


FIGURE 6. INPUT, OUTPUT SIGNALS WHEN STARTIN IS LONGER THAN ONE CLOCK CYCLE

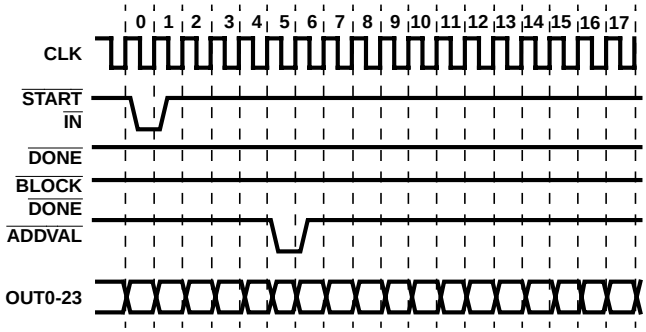


FIGURE 7. USING STARTIN TO RESTART SEQUENCE DURING OPERATION