

```
END PROCESS;  
sr_tap_one <= sr(15);  
sr_tap_two <= sr(31);  
sr_tap_three <= sr(47);  
sr_out <= sr(63);  
END arch;
```

Register and Latch Coding Guidelines

This section provides device-specific coding recommendations for Altera registers and latches.

Understanding the architecture of the target Altera device helps ensure that your code produces the expected results and achieves the optimal quality of results.

Register Power-Up Values in Altera Devices

Registers in the device core always power up to a low (0) logic level on all Altera devices.

If your design specifies a power-up level other than 0, synthesis tools can implement logic that causes registers to behave as if they were powering up to a high (1) logic level.

If your design uses a preset signal on a device that does not support presets in the register architecture, your synthesis tool may convert the preset signal to a clear signal, which requires synthesis to perform an optimization referred to as NOT gate push-back. NOT gate push-back adds an inverter to the input and the output of the register so that the reset and power-up conditions will appear to be high, and the device operates as expected. In this case, your synthesis tool may issue a message informing you about the power-up condition. The register itself powers up low, but the register output is inverted, so the signal that arrives at all destinations is high.

Due to these effects, if you specify a non-zero reset value, you may cause your synthesis tool to use the asynchronous clear (`aclr`) signals available on the registers to implement the high bits with NOT gate push-back. In that case, the registers look as though they power up to the specified reset value.

When an asynchronous load (`aload`) signal is available in the device registers, your synthesis tools can implement a reset of 1 or 0 value by using an asynchronous load of 1 or 0. When the synthesis tool uses a load signal, it is not performing NOT gate push-back, so the registers power up to a 0 logic level.

For additional details, refer to the appropriate device family handbook or the appropriate handbook on the Altera website.

Designers typically use an explicit reset signal for the design, which forces all registers into their appropriate values after reset. Altera recommends this practice to reset the device after power-up to restore the proper state.

You can make your design more stable and avoid potential glitches by synchronizing external or combinational logic of the device architecture before you drive the asynchronous control ports of registers.

Related Information

[Design Recommendations for Altera Devices and the Quartus Prime Design Assistant](#) on page 11-1

Specifying a Power-Up Value

If you want to force a particular power-up condition for your design, you can use the synthesis options available in your synthesis tool.