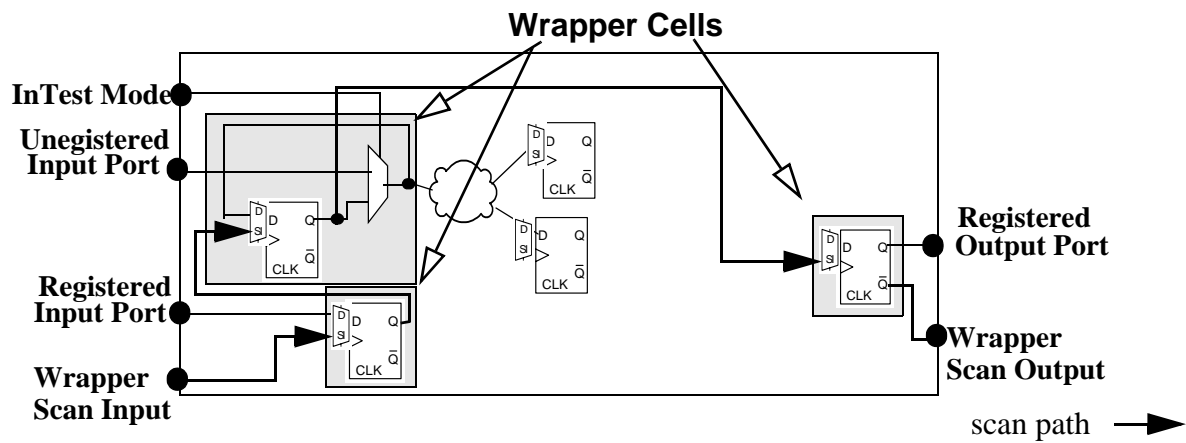


# IEEE Std. 1500 Compliant Wrapper Boundary Register Cell

**Author: Teresa L. McLaurin**

The IEEE 1500 standard was created to address test complexity of System on Chips (SoCs). It provides a standard interface and a set of rules for creation of an isolation boundary between a core and the logic external to the core. The purpose of the isolation boundary or wrapper is to allow for isolated test of the core with minimal signals that must be ported out to the SoC level. The wrapper comprises wrapper cells for each functional input and output port. The flip-flops with the shaded boundaries, shown in Figure 1, are utilized as wrapper cells and are connected into a wrapper scan chain or chains. The wrapper chains are used in place of the functional ports during test to control and observe the core logic. This is what allows the core to be isolated from any external logic during test and still get full coverage of the core. Note that the Unregistered Input Port, in the example shown in Figure 1, does not reuse a functional Mux-D flip-flop. A dedicated test wrapper cell is added to this port. The register is used for direct control during Intest Mode and at all other times, the functional unregistered port value can be captured into the wrapper cell, providing the clock is active (e.g. during test of the logic external to the core).

**Figure 1 Wrapper Cells Example**



The entire wrapper can also be reused to help achieve full test coverage of the logic external to the core.

The IEEE 1500 standard requires that cores include a IEEE 1500 wrapper to be compliant. This wrapper comprises a Wrapper Boundary Register (WBR), a Wrapper Instruction Register (WIR) and a Wrapper Bypass Register (WBY). There is also a set of signals called the Wrapper Serial Port (WSP) that are used for all serial instructions and any communication with the WIR or the WBY. The WSP comprises eight mandatory and one optional signal. The signals are as follows:

- WRSTN - Reset for IEEE 1500 logic - leaves the wrapper in functional mode
- WSI - Wrapper Serial Input - used as the scan input
- WSO - Wrapper Serial Output - used as the scan output
- WRCK - Wrapper Clock - always clocks the WIR, WBY and during serial instructions must clock the WBR
- ShiftWR - the shift enable signal
- UpdateWR - the update enable signal
- CaptureWR - the capture enable signal

- SelectWIR - selects between the instruction register (i.e. WIR) and a data register (e.g. WBR, WBY)
- TransferDR - optional - used only for the WBR if required by the WBR cells.

The IEEE 1500 standard mandates one serial instruction (WS\_EXTEST), so all IEEE 1500 wrapped cores must have the eight required ports, but not all need connect to all of the WBR cells. The WIR requires the shift and update events, but has an optional capture event. The WBR requires the shift and capture event, but may not require the update event or the transfer event as these are WBR cell dependent.

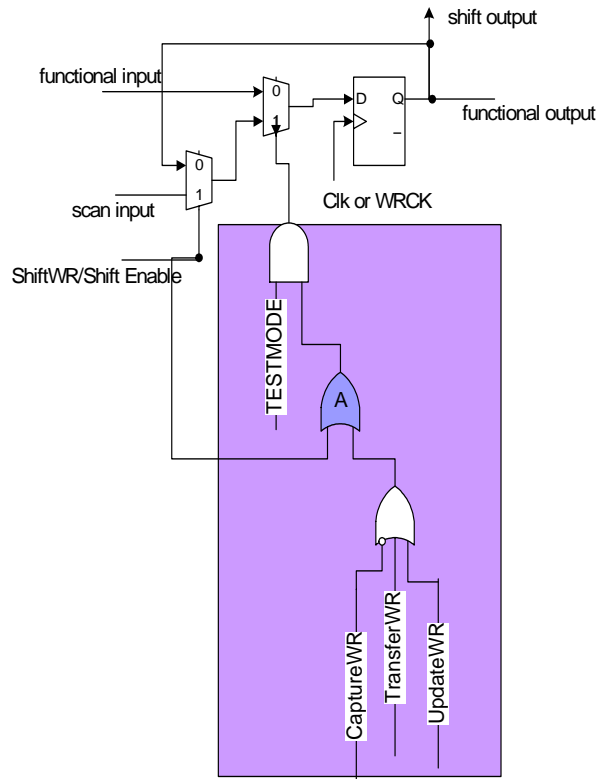
The WBR is composed of cells whose primary function is to control and observe the ports of the core. These cells are connected into a scan chain to supply the proper values for control of the inputs or to shift out the values that were observed on the outputs. These WBR cells must follow certain rules. The rules do not describe the circuitry of the WBR cell other than the input and output ports. Instead, the rules describe the behavior of the WBR cells. This allows for much freedom in the design of the WBR cell as long as the behaviors are met. These behaviors include the following:

1. Every WBR cell must have at least one storage element that is connected within a scan chain.
2. Every WBR cell must have a storage cell that can be utilized for the capture operation (for observability)
3. If an update storage element is provided for the capture operation, the transfer event must be supported.
4. Storage elements in the shift path cannot respond to the update operation
5. WBR cells must not change state in the absence of an active edge on WRCK or a pulse on WRSTN
6. The WBR cells must support and cannot interfere with normal functional mode.
7. During internal test mode, the WBR input cells must respond to the shift event, apply event and if provisioned, the transfer or update events. The WBR output cells must respond to the shift event, capture event and if provisioned the transfer event.
8. During external test mode, the WBR output cells must respond to the shift event, apply event and if provisioned, the transfer or update events. The WBR input cells must respond to the shift event, capture event and if provisioned the transfer event.
9. In the case that an event is not required (i.e. update or transfer), the WBR cell must have the capability to hold its value while other core wrappers are executing these events.

The WBR cells on ARM cores currently do not require the update or transfer event, but must be able to hold their values if other wrappers, being concurrently run, do require these events.

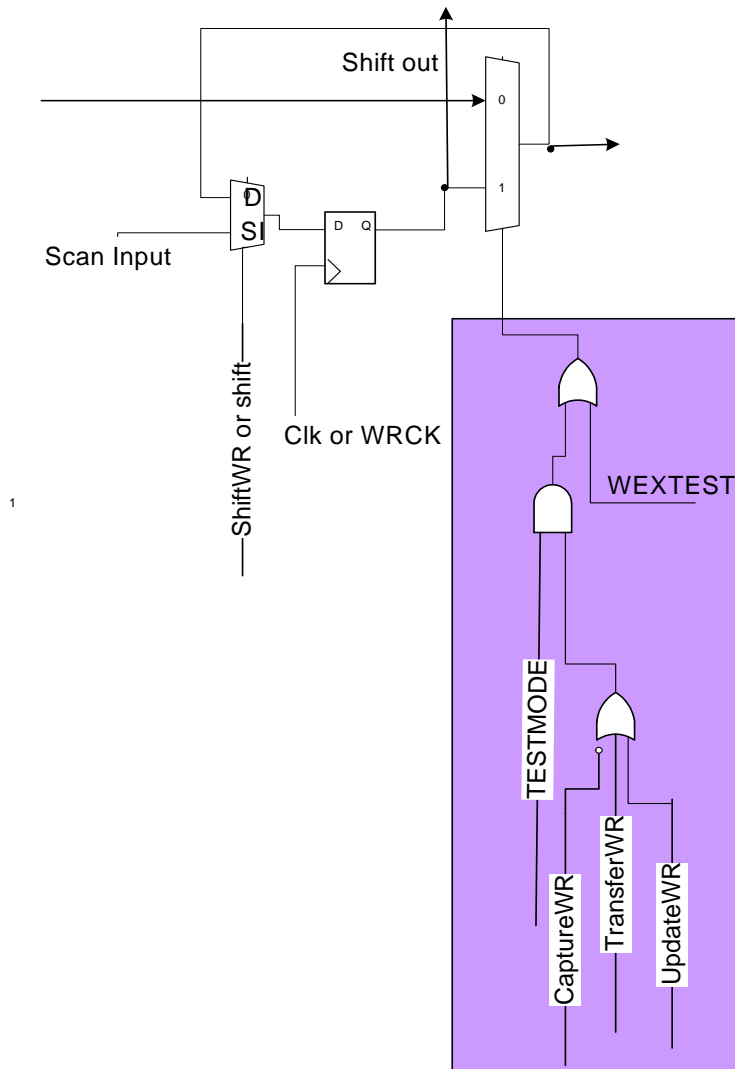
The update and transfer signals could be routed to each WBR cell to indicate when a hold is required as shown in Figure 2. This wrapper cell meets all of the requirements listed above.

Note that the logic in the shaded area can be moved out of the wrapper cell. It can be created once, and then just the single hold wire would be fanned out to each wrapper cell.

**Figure 2 IEEE 1500 Compliant Shared Wrapper Cell - Example 1**

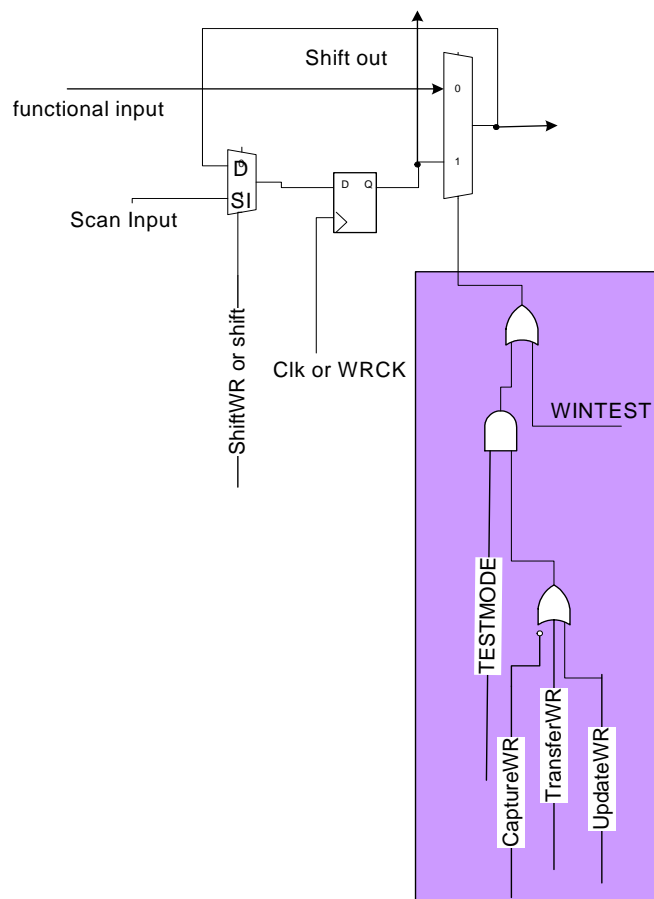
In Figure 2, if TESTMODE is low, the functional path will always be the chosen path. However, if TESTMODE is high (enabled), and if TransferDR or UpdateWR are enabled or CaptureWR is disabled, the cell will hold its value. Note that there is an override for the shift function. If shift and TESTMODE are enabled, nothing must override this. If, in a design, this is not possible, then OR gate A is not needed. The extra multiplexer needed for the hold function is kept out of the functional path.

The above cell is a shared wrapper cell. This same type of logic can be applied to dedicated wrapper boundary register cell as shown in Figure 3.

**Figure 3 IEEE 1500 Compliant Dedicated Output Wrapper Cell**

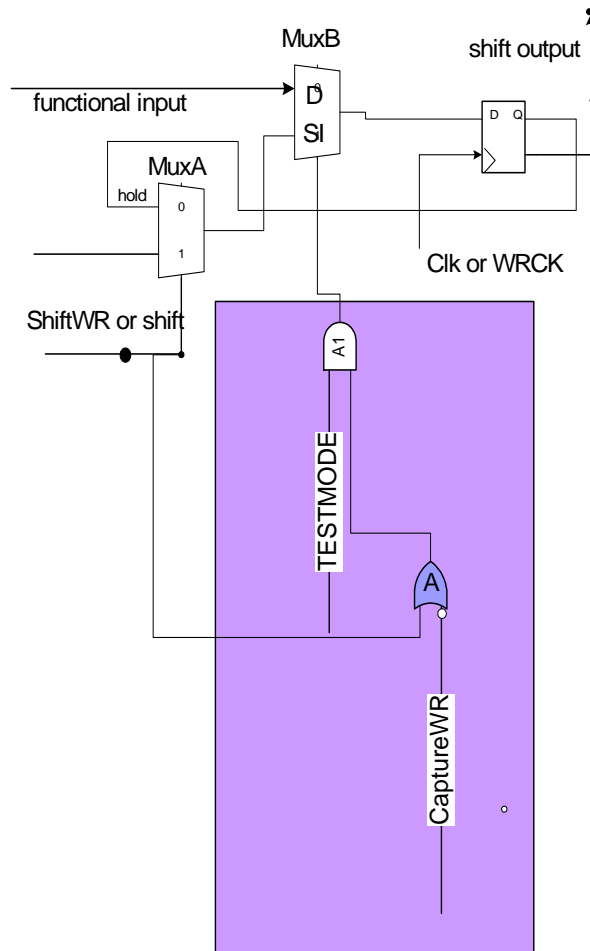
The WEXTEST signal shown in Figure 3 is asserted (high) during external test modes. An external test mode is when logic external to the core is being tested using the core wrapper. WEXTEST only factors into dedicated wrapper cells that are adjacent to output ports. While the WEXTEST signal is high, the dedicated output wrapper cell is either shifting or holding. An output wrapper cell cannot capture during an external test mode, just as an input wrapper cell cannot capture during an internal test mode. All wrapper cells adjacent to input ports utilize a signal called WINTEST (shown in Figure 4), which is enabled high during internal test mode (testing of the core).

### Figure 4 IEEE 1500 Dedicated Input Wrapper Cell



The method described thus far requires two signals, TransferDR and UpdateDR, to be routed to the core that are not required for the function of this particular WBR. Following is a proposal to remove those two superfluous signals and still meet all of the WBR cell behavior requirements of the IEEE 1500 standard. The hold function is determined by the state of the shift, capture and TESTMODE. If TESTMODE is enabled, and both shift and capture are disabled, this infers that an update or transfer is occurring.

In all current ARM designs, only a shift enable (SE) signal goes to our wrapper cells and the capture is  $\sim\text{shift}$  ( $\sim\text{SE}$ ), which is the same state as functional mode. In this case, the wrapper cell would not understand when to hold and when to capture - it cannot go into the hold state during functional mode ( $\sim\text{shift}$ ) or capture mode ( $\sim\text{shift}$ ). So, another signal must be added to differentiate hold from capture and from functional mode (see Figure 5). This methodology works for both shared and dedicated wrapper cells.

**Figure 5 IEEE 1500 Compliant Shared Wrapper Cell - Example 2**

The signal that ultimately connects to the SI port of MuxB is the shift or hold signal. If shift is enabled and capture is disabled during TESTMODE, the value going into SI of MuxB is the shift value. If shift and capture are both disabled, during TESTMODE, the hold value is input to the SI port of MuxB. The D port of MuxB always gets functional data (called capture data during scan testing). Capture data occurs if shift is disabled and capture is enabled or during functional mode ( $\sim$ TESTMODE). The extra multiplexer needed for the hold function is kept out of the functional path.

This logic could be internal to the wrapper cell as shown in Figure 5. In this case, the shift, capture and TESTMODE signal would have to be routed to each cell. However, the update and transfer would not have to be routed. The AND gate logic A1 can be removed from the cell, such that only one signal must be routed to each wrapper cell. During IEEE 1500 serial instructions, the ShiftWR and the CaptureWR signals would control the shift and capture signals on a wrapper cell.

Perhaps during internal test mode, the IEEE 1500 WSP (Wrapper Serial Port) signals are not used. Instead, WSE (wrapper scan enable) is utilized for the wrapper chain - as is in fact is the case with many ARM cores. There is no need for the hold capability during this test.

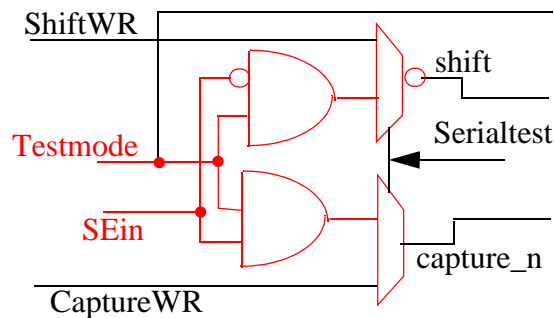
For this scenario, the equation for the shift and capture logic is as follows:

$$\text{shift} = \text{SE} \ \& \ \text{Testmode}$$

$\text{capture} = \sim\text{SE} \ \& \ \text{Testmode}$

The control signal for the muxes in Figure 6 can be any signal needed that will switch between the IEEE 1500 WSP ShiftWR/CaptureWR signals and the SEin signal. During this example, the WSP signals are only used during IEEE 1500 serial instructions. IEEE 1500 serial instructions require that only the WSP signals be utilized during test. Thus the name of the signal controlling the muxes, in the Figure 6 example, is Serialtest. If there are multiple wrapper scan enables, one structure per scan enable is needed. Also note that the AND gate A1 (in Figure 5) has been moved out of the wrapper cell and into this logic structure in Figure 6.

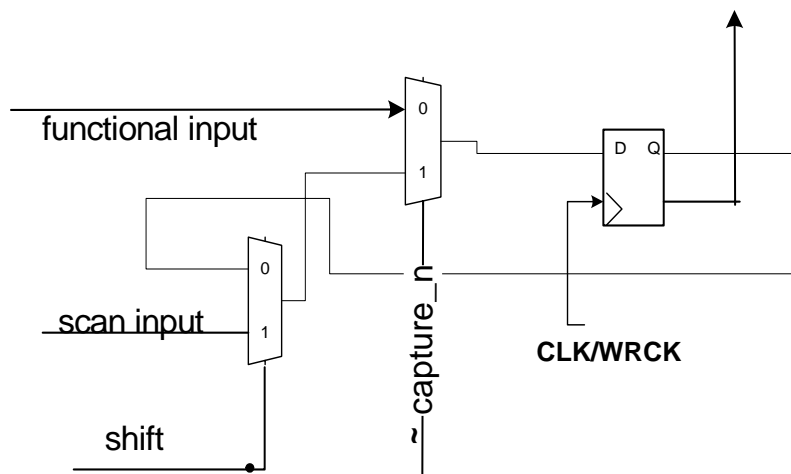
**Figure 6 shift/capture with SE control**



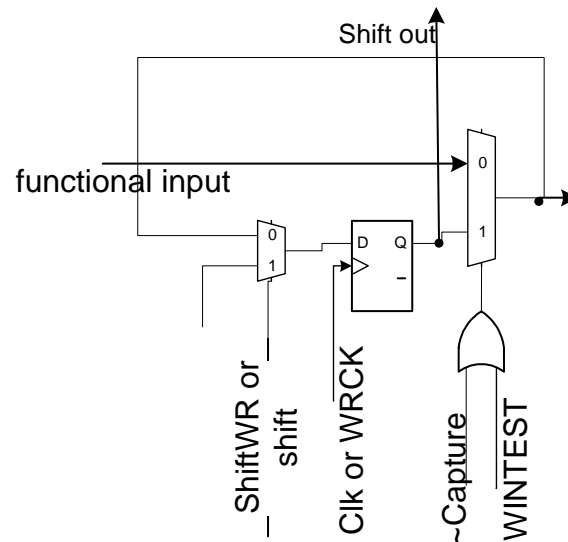
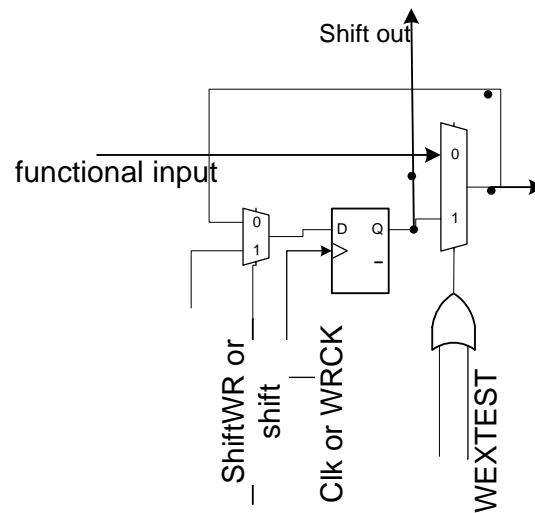
One other benefit of creating the hold signal in this manner is that since the TransferWR signal is optional, it does not have to be on the core and the SoC designer does not have to route that signal to the core.

The WBR cell that connects to the circuitry in Figure 6 is illustrated in Figure 7. Note that the shift signal could also be called  $\sim\text{hold}$  and the  $\sim\text{capture}$  signal could also be called shift.

**Figure 7 Shared WBR cell**



The WBR cell illustrated in Figure 7 can also be replaced with dedicated wrapper cells shown in Figure 8 and Figure 9.

**Figure 8 Dedicated Input Wrapper Cell****Figure 9 Dedicated Output WBR Cell**

## Conclusion

There are many variations of a WBR cell that can be created that exhibit the behavior required by the IEEE Std. 1500. This document only describes a couple of these variations. The WBR cell designs described in this document are a practical way to both meet the IEEE 1500 standard specifications and have minimal timing and area impact on the current ARM WBR by eliminating the need to route separate TransferDR and UpdateWR signals to each wrapper cell. In fact, TransferDR need not be routed to the core at all.