module mem_interface (data [3:0], input data);

tri [7:0] data;

assign data = RW ? data_int : 8'b1;
endmodule

---

test fixture:
tri data;

mem_interface ul (... data )
res[7:0] mt_data
initial
begin
  mt_data = ...;
  control = 1'b1;
end
assign data = control ? mt_data : 8'h8;
Register or Serial Operation

Replace Register Cells with Scan Cell:

Synchronous Logic Under Normal Mode of Operation:

Scan-Based Testing
A test vectors
Compared on
expected against stored
Check if sec if clip is as
Check out result vector similarly
Mode = scan
Clock logic one cycle
Mode = normal
Clock test vector in serially
\[ \text{Mode} = \text{scan} \]
<table>
<thead>
<tr>
<th>a</th>
<th>b</th>
<th>c</th>
<th>Detect</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>NANO</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1KU</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>button</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>-1</td>
</tr>
</tbody>
</table>

Note: By trying to force a '1' at the output

2: Without fault, will see 0 at O1P of NANO. With fault will see 1
In theory, vectors to detect SA-O/1 might not detect other faults e.g. wire opens and shorts. In practice, 99%+ coverage of SA-O/1 faults — these vectors will exercise most other faults. However, <95% coverage will be too low.

\[
TE = 1 - 0.5 \times (1 - 0.95)
\]
\[
= 1 - 0.97 = 3 \% 
\]