

Multiple Drivers

- Each concurrent signal assignment statement creates a driver for the signal being assigned. What happens when there is more than one assignment to the same signal?
- In this case, the signal has more than one driver and a mechanism is needed to compute the effective value of the signal.

Multiple Drivers cont..

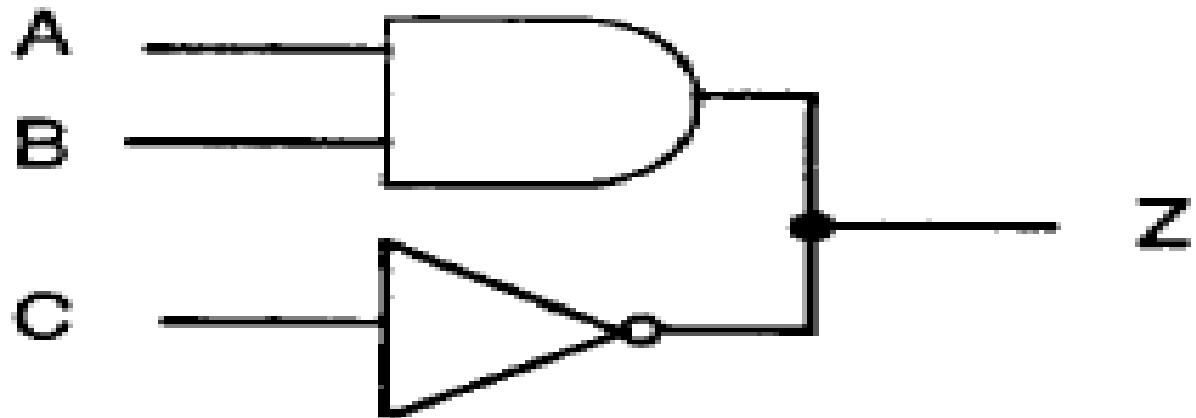


Figure 5.8 Two drivers driving signal Z.

Multiple Drivers cont..

```
entity TWO_DR_EXAMPLE is  
port (A, B, C: in BIT; Z: out BIT);  
end TWO_DR_EXAMPLE;  
architecture TWO_DR_EX_BEH of  
    TWO_DR_EXAMPLE is  
begin  
    Z <= A and B after 10 ns;  
    Z <= not C after 5 ns;  
end;
```

Multiple Drivers and Resolution Function

- In this example, there are two gates driving the output signal Z. How is the value of Z determined? It is determined by using a user-defined *resolution function that considers the values of both the drivers for Z and determines the effective value. Consider the following architecture body.*

architecture NO_ENTITY of DUMMY is

begin

Z <= '1' after 2 ns, '0' after 5 ns, '1' after 10 ns;

Z <= '0' after 4 ns, '1' after 5 ns, '0' after 20 ns;

Z <= '1' after 10 ns, '0' after 20 ns;

end NO_ENTITY;

Resolution Function cont..

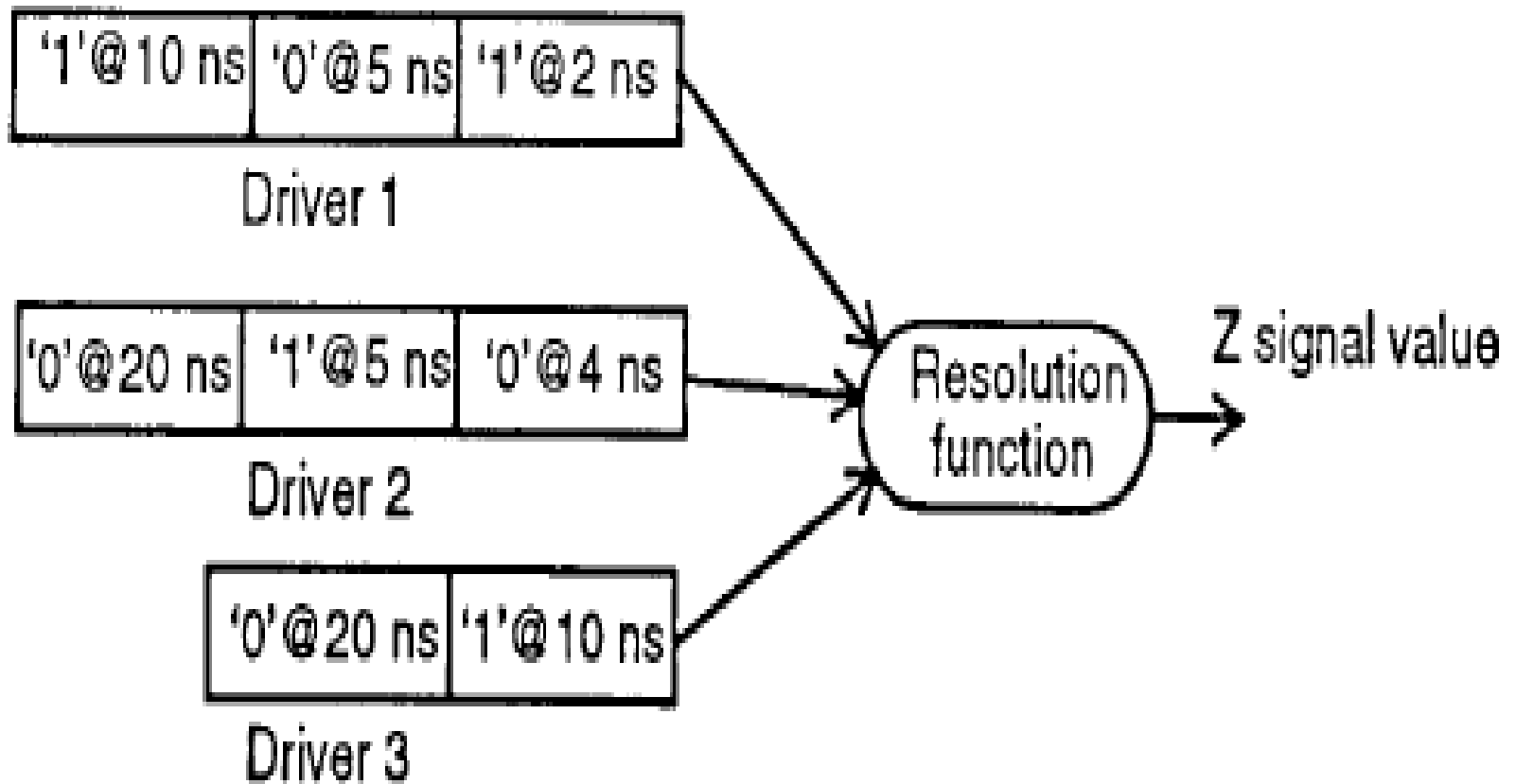


Figure 5.9 Resolving signal drivers.

Resolution Function cont..

- The value of each driver is an input to the resolution function and based on the computation performed within the resolution function, the value returned by this function becomes the resolved value for the signal. The resolution function is user-written.
- A signal with more than one driver must have a resolution function associated with it, otherwise, it is an error. Such a signal is called a *resolved signal*.

Conditional Signal Assignment Statement

- The conditional signal assignment statement selects different values for the target signal based on the specified, possibly different, conditions (it is like an if statement). A typical syntax for this statement is
 - *Target - signal* **\leftarrow** [*waveform-elements when condition else*]
 - [*waveform-elements when condition else*]
 - . . .
 - *waveform-elements*;

Example of Conditional Signal Assignment Statement

For example,

**Z <= IN0 after 10ns when S0 = '0' and
S1 = '0' else**

**IN1 after 10ns when S0 = '1' and S1 = '0'
else**

**IN2 after 10ns when S0 = '0' and S1 = '1'
else**

IN3 after 10 ns;

Selected Signal Assignment Statement

- The selected signal assignment statement selects different values for a target signal based on the, value of a select expression (it is like a case statement). A typical syntax for this statement is
- **with *expression select***
 - *target-signal* <= *waveform-elements when choices*,
 - *waveform-elements when choices*,
 - ...
- *waveform-elements when choices* ;

Here is an example of a selected signal assignment statement.

type OP is (ADD, SUB, MUL, DIV);

signal OP_CODE: OP;

...

with OP_CODE select

Z <= A+B after ADD_PROP_DLY when ADD,

A - B after SUB_PROP_DLY when SUB,

A * B after MUL_PROP_DLY when MUL,

A / B after DIV_PROP_DLY when DIV;