TELEFUNKEN
Semiconductors

TF112 Pass-Thru Bits User's Guide

APPLICATION NOTE 300
Bill Aronson

Introduction

The TF112 is used in systems that incorporate the IEEE-1149.n test standards. It is used to expand the number of scan chains available to any one (or two) 1149.n test masters. It is capable of being used as an addressable port in a multidrop system as well as a scan chain mux. This allows for a greater flexibility in targeting and combining multiple scan chains.

Pass-Thru bits are a feature on the TF112 that allow for additional signals from the test master to be presented to the devices on a particular LSP (Local Scan Port). These signal could include the R/W bit on a flash device or a control line that is need by a microprocessor for ICE (In Circuit Emulation).

Control Mechanisms

The Pass-Thru bits on the TF112 have two control mechanisms. These are designed to allow for greater flexibility of control compared to similar devices. The first method is the compatibility mode where the bits become enabled once the LSP has been UNPARKED. The logic value present on the outputs will be controlled by the input data (e.g. $Y0_{B1}$ = $A0_{B0}$ when B_0 is the master port and LSP0 is UNPARKED). The second method is new to the TF112 and consist of an internal control register. This register is accessed in the same manner as other internal data registers in an IEEE-1149.n compatible device. First the instruction register is loaded with the required instruction, in this case 0xBF. The internal muxes then select the Pass-Thru bit control register as the active data register This a an 8-bit register and has the following structure.

Bit	Default	Function
0	0	LSP0 Y_0 Data bit
1	0	Enable LSP0 Y_0 as GPIO
2	0	LSP0 Y_1 Data bit
3	0	Enable LSP0 Y_1 as GPIO
4	0	LSP1 Y_0 Data bit
5	0	Enable LSP1 Y_0 as GPIO
6	0	LSP1 Y_1 Data bit
7	0	Enable LSP1 Y_1 as GPIO

Table 1. Pass-Thru bit Data Register

There are two Pass-Thru bits available to LSP0 and LSP1 for a total of four bits. Each of the Pass-Thru bits has two bits in the data register associated with its functionality. One bit enables and disables the output, the second bit determines the logic value that will be driven. Once the instruction register has been loaded with the 0xBF instruction the user may write to the data register and configure the Pass-Thru bits.

There is an operational order to the two control mechanisms whereas the enable bits in the data register override the functionality of the compatibility mode.

Architectural Note

The pins that are utilized as LSP0 are selected based on which of the two master ports is selected. The master port is selected with the MP_{sel} pin where if it is asserted to a logic 1, the B1 port is the master and the B0 port becomes LSP0. The opposite is true when MP_{sel} is deasserted to a logic 0, the B0 port becomes the master port and the B1 port becomes the LSP0 port.

Compatibility Mode

Compatibility mode is designed to be identical in functionality to other ScanBridge devices. The output pins are controlled by the input pins (e.g. $Y0_{B1} = A0_{B0}$) once the associated LSP as been UNPARKED. If the LSP is in the PARKED state the outputs remain in the High-Z state. In Figure 1 the transitions on the inputs at t_1 and t_2 have no effect on the outputs. Once the LSP has been UNPARKED at t_4 the outputs begin driving data that is controlled by the inputs. When the inputs transitions at t_5 and t_7 results in the outputs following the inputs.

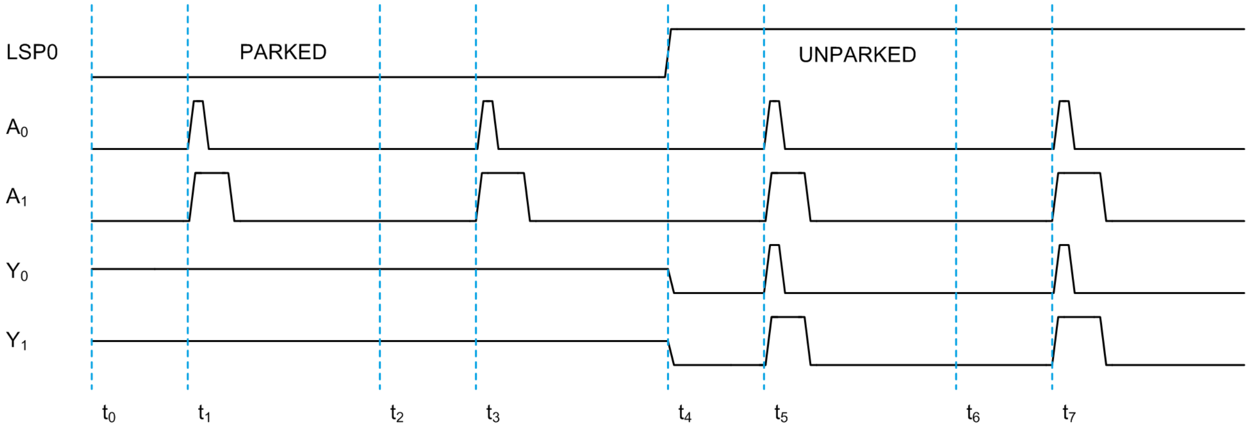

Figure 1. Logic Diagram of Compatibility Mode

Internal Control Register Mode

In Figure 2 the functionality from t_0 until t_2 is identical in functionality to figure 1 because the LSP has not been UNPARKED and the enable bits in the internal control register are deasserted resulting in the outputs being in the High-Z state. When at t_2 the enable bit in the internal control register is asserted to a logic 1 the outputs begin to drive data that corresponds to the data in the internal control register even while the LSP is PARKED. At t_4 the LSP is UNPARKED and the Pass-Thru bit outputs are still controlled by the internal control register and the input pins are ignored. While the LSP is still UNPARKED at t_6 the enable bit in the internal control register is deasserted and control of the Pass-Thru bit outputs is returned to the Pass-Thru bit inputs. The transitions on the inputs at t_7 results in the outputs following the inputs.

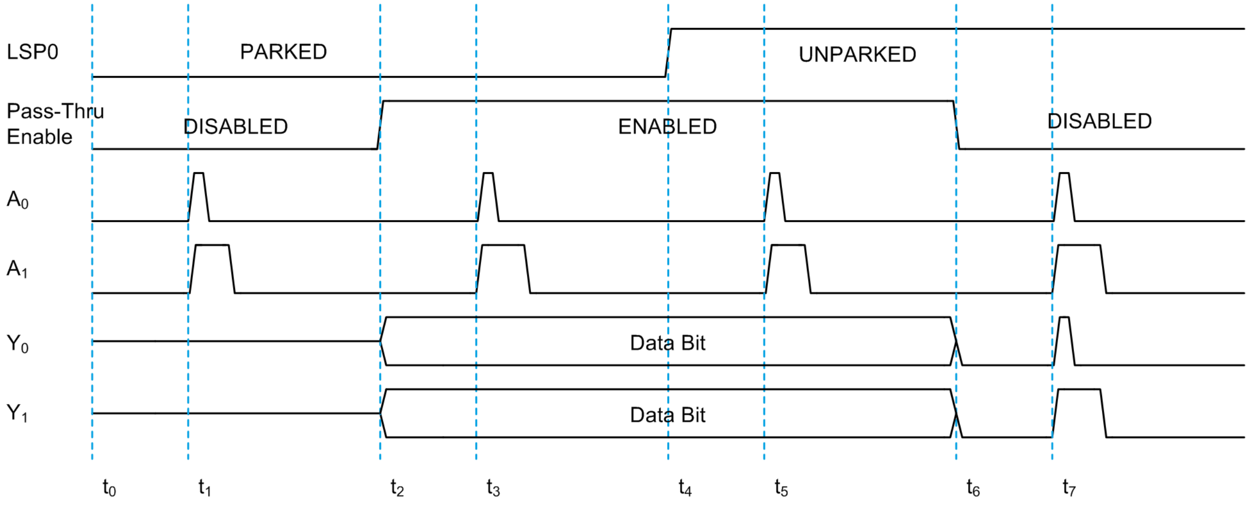

Figure 2. Logic Diagram of Internal Control Register Mode

SVF Example

Figure 3 is an example SVF file that controls the Pass-Thru output bits

```
trst on;
runtest 1000 TCK;
trst off;
state reset;
state idle;
sir 8 tdi(09) tdo(00) mask(00); ! set bridge address
sir 8 tdi(8e) tdo(25) mask(ff); ! modesel
sdr 8 tdi(01) tdo(01) mask(00); ! LSP0
sir 8  tdi(bf) tdo(00) mask(00); ! Access pass-thru bit register
sdr 8  tdi(0A) tdo(00) mask(00); ! Enable Y0 and Y1 to drive a '0'
sir 8  tdi(e7) tdo(25) mask(ff); ! unpark
sir 16 tdi(ff11) tdo(224A) mask(ffff); ! bypass=ff
sdr 41 tdi(1AAAAAAAAAA) tdo(00000000000) mask(00000000000);
sir 16 tdi(bf11) tdo(224A) mask(ffff); ! Access pass-thru bit register
sdr 48 tdi(0BAAAAAAAAAA) tdo(000000000000) mask(000000000000); ! Y0 = 1 Y1 = 0
runtest 100 TCK;
sdr 48 tdi(0EAAAAAAAAAA) tdo(000000000000) mask(000000000000); ! Y0 = 0 Y1 = 1

runtest 100 TCK;
sdr 48 tdi(0AAAAAAAAAAA) tdo(000000000000) mask(000000000000); ! Y0 = 0 Y1 = 0
runtest 100 TCK;
```

Figure 3. SVF example

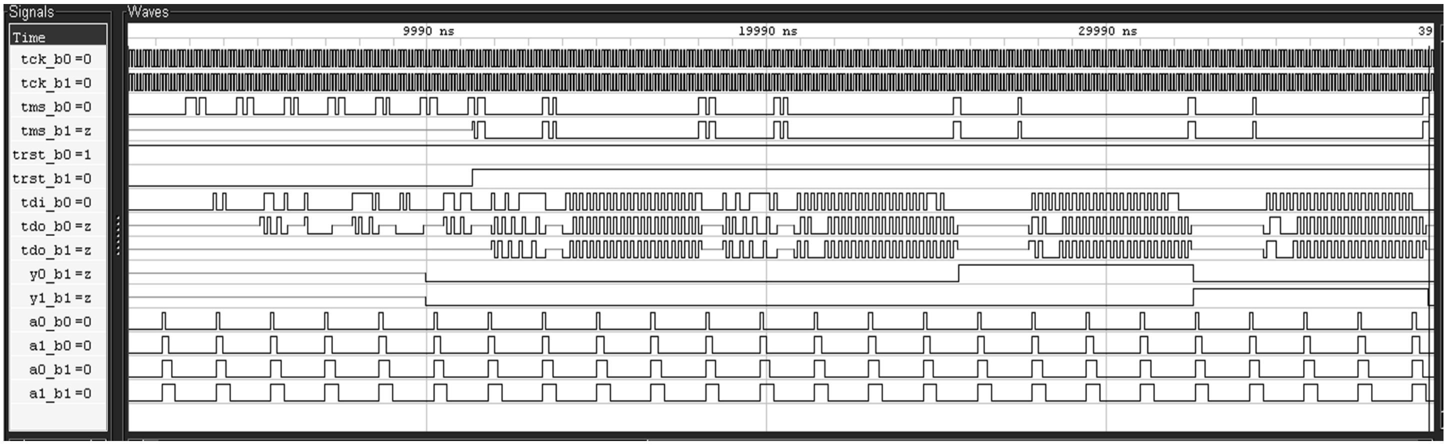

Figure 4. Logic Diagram SVF example