

最適合車用電子晶片記憶體測試與開發的診斷工具

現今記憶體測試常使用自動測試裝置 (Automatic test equipment, ATE)，來進行測試，首先 START™ v3 會提供給 ATE 所需要的 memory test pattern 來進行測試，ATE 會產出測試結果 ATE log 檔，而 ATE log 的內容為 0、1 的訊號，要去診斷判讀會較繁瑣與不易，因此為了方便快速分析記憶體在 ATE 測試後的結果，START™ v3 開發診斷 ATE 工具，需搭配芯測科技自家工具產出的檔案做配合就可方便分析錯誤記憶體資訊。芯測科技還有開發一款以 PC base 的 JTAG 轉 USB 的診斷工具，可以針對非量產 chip 以及開發中的 chip 進行診斷，可以省下 ATE 測試的成本還可以即時的知曉診斷結果。

一、JTAG 轉 USB 的診斷工具

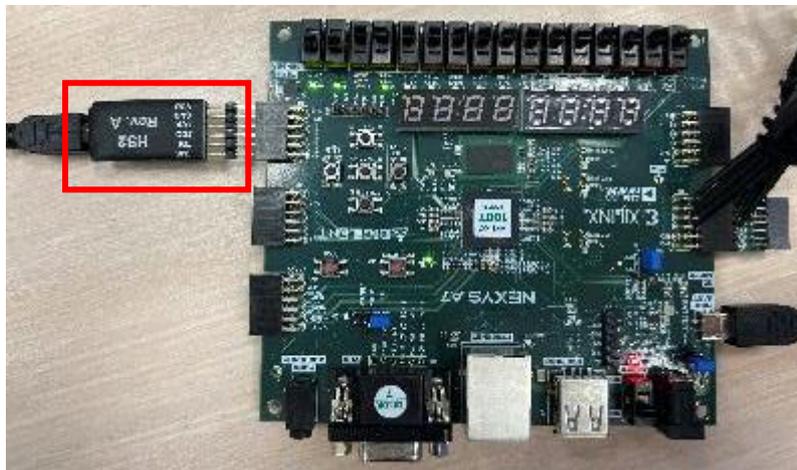
1. JTAG 轉 USB 診斷工具架構：

JTAG 轉 USB 的診斷工具可以方便進行診斷，JTAG 轉 USB 的診斷工具主要是透過轉接板來對 PC 與 FPGA 進行溝通，此 FPGA 是在模擬非量產 chip 或是開發中的 chip，轉接板與整體工具的規格示意圖如下圖所示。PC 與轉接板是用 USB 做連接，轉接板配合工具會將訊號轉換成 JTAG(IEEE1149.1)，再送給 FPGA/IC 進行診斷。



Specification of JTAG-to-USB debugging tool

下圖為實際使用示意圖，紅色框起來的是轉接板，轉接板右側是 FPGA，左側是 PC 端。



2.實際使用:

在 PC 端安裝完 JTAG 轉 USB 的診斷工具後，即可開始使用診斷工具，首先可以要參閱做完整合的 INTEG testbench 中 bist_testing 這個 task，在這個 task 中我們可以找到下圖的 CMD_DATA 資訊，依照下圖資訊即可填寫設定 input binary value 去使用 JTAG(ieee1149.1)的 TDI 做 send_command，即可開始測試。

```
top_default_CMD_DATA = {top_default_DIAG, top_default_ALG,  
                        top_default_SEQ_ID, top_default_GRP_ID,  
                        top_default_MEB_ID, top_default_MEN};
```

訊號解釋:

Controller_name_DIAG：是否要執行 Diagnosis，當設定為時 1 為啟動。

Controller_name_ALG：當 BFL 選項中有開啟 algorithm_selection 時，在 testbench 中就產生 Controller_name_ALG 的指令可以來控制想要測試的 algorithm。

Controller_name_SEQ_ID、Controller_name_GRP_ID、

Controller_name_MEB_ID：都是用來指定目前要做測試的 memory 的 ID。

Controller_name_MEN：Controller BIST enable 的指令，當設定為時 1 為啟動。

JTAG 的 TDO 會產出 capture_commad，而 capture_commad 可參考下圖 INTEG testbench 中的 test_result 的訊號排列，即可解讀 capture_commad 的內容。

```
{top_default_MGO, top_default_MRD,  
top_default_SRD, top_default_LATCH_GO} = top_default_test_result;
```

訊號解釋:

Controller_name_MGO：為 BIST 測試結果，當 BIST 測試 fail 時，為 0。

Controller_name_MRD：為 BIST 測試完成時，會為 1。

Controller_name_SRD：當 Diagnosis Data 已準備好時，會為 1，即可進行 capture Diagnosis Data。

Controller_name_LATCH_GO:此訊號的 width 是依照 START 所產出的 meminfo 檔中的 memory 數量而定，當 LATCH_GO 的每個訊號從 1 變到 0 時，代表此顆 memory 測試 fail。

二、診斷 ATE 工具

設定 START™ v3 (BFL & BII) :

1. 設定 *.bfl :

- A. 開啟設定 diagnosis_support 與 diagnosis_width_info
 - I. diagnosis_support : START™ v3 提供診斷模式並回報診斷資訊。
 - II. diagnosis_width_info :將測試資訊的長度對齊，方便診斷與判讀。
- B. diagnosis_faulty_items : 提供使用者選擇診斷的資訊，以便符合自家的設計，可選擇的資訊包含：演算法、運算單元、演算法元素、記憶體分群位置、記憶體位址、記憶體資料。

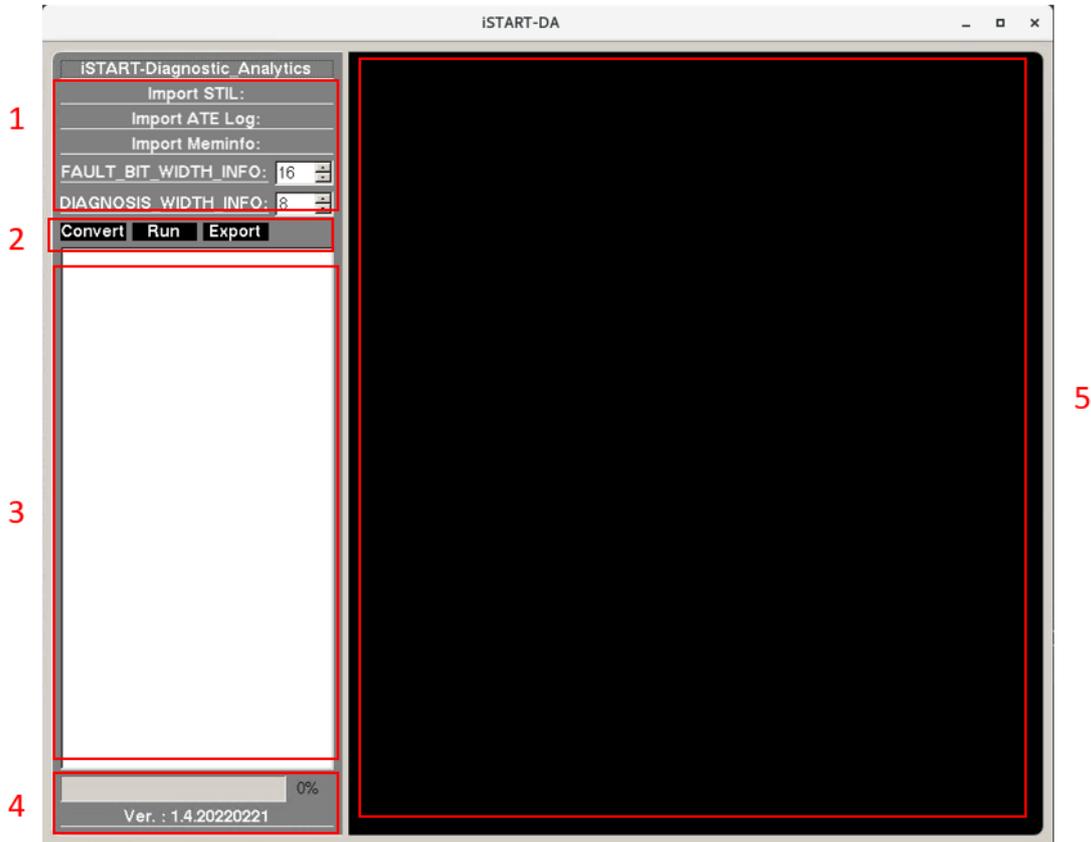
```
define{BIST}
  set diagnosis_support      = yes      # yes, no
  set diagnosis_width_info  = yes      # yes, no
  set diagnosis_faulty_items = algorithm, operation, element, seq_id, grp_id, address, ram_data, rom_data
```

2. 設定 *.bii :

- A. 在 define{Testbench}[INTEG_tb]下，file_format 選擇 STIL，在模擬後產生 memory test pattern 的 STIL 檔案。

```
define{Testbench}[INTEG_tb]
  set pll_wait_cycle      = 10
  set reset_cycle         = 10          # integer bigger than 0
  set file_format         = STIL       # verilog, STIL
```

診斷工具介面介紹：



1. 檔案與設定：

- A. 輸入 ATE 的 STIL 檔：STIL 檔會依照 START™ v3 所開的選項產出對應 Memory Test Pattern(如上述所 START™ v3 設定)
- B. ATE log：ATE 測試完結果的檔案
- C. meminfo 檔：由 START 所產生，其內容會包含所有記憶體的所有資訊。

```
# if the instance is the alais type , The ** is .
[DOMAIN=top_default, cycle=100.0ns]
[CTR] # Hier: top
[SEQ] # No.= 1,InstanceNo= 3,SEQ_max_addr_size= 1024,Hier: top u_t1
[GROUP] # No.=1 1
[SP=1_1_1, byp=no, diag=no, q_pipe=no, repair=no] sram_sp_1024x32
[SP=1_1_2, byp=no, diag=no, q_pipe=no, repair=no] sram_sp_1024x32
[SP=1_1_3, byp=no, diag=no, q_pipe=no, repair=no] sram_sp_1024x32
[SEQ] # No.= 2,InstanceNo= 1,SEQ_max_addr_size= 24,Hier: top u_t1
[GROUP] # No.=2 1
[2P=2_1_1, byp=no, diag=no, q_pipe=no, repair=no] rf_2p_24x28
[SEQ] # No.= 3,InstanceNo= 1,SEQ_max_addr_size= 1024,Hier: top u_t1
[GROUP] # No.=3 1
[DP=3_1_1, byp=no, diag=no, q_pipe=no, repair=no] sram_dp_1024x64

###Total mbist memory instance = 5
###Total SRAM/REGFILE = 3
###Total SRAM_2P = 1
###Total SRAM_DP = 1
###Total ROM = 0
###top_default algorithm is= (March C- @2P,SOLID)(March C-,SOLID)(March C+,SOLID)(March C- @DP,SOLID)

#####Not Group Memory List#####
```

Hierarchy

```
top u_t1 ram_1
top u_t1 ram_2
top u_t1 ram_3

top u_t1 u_2p

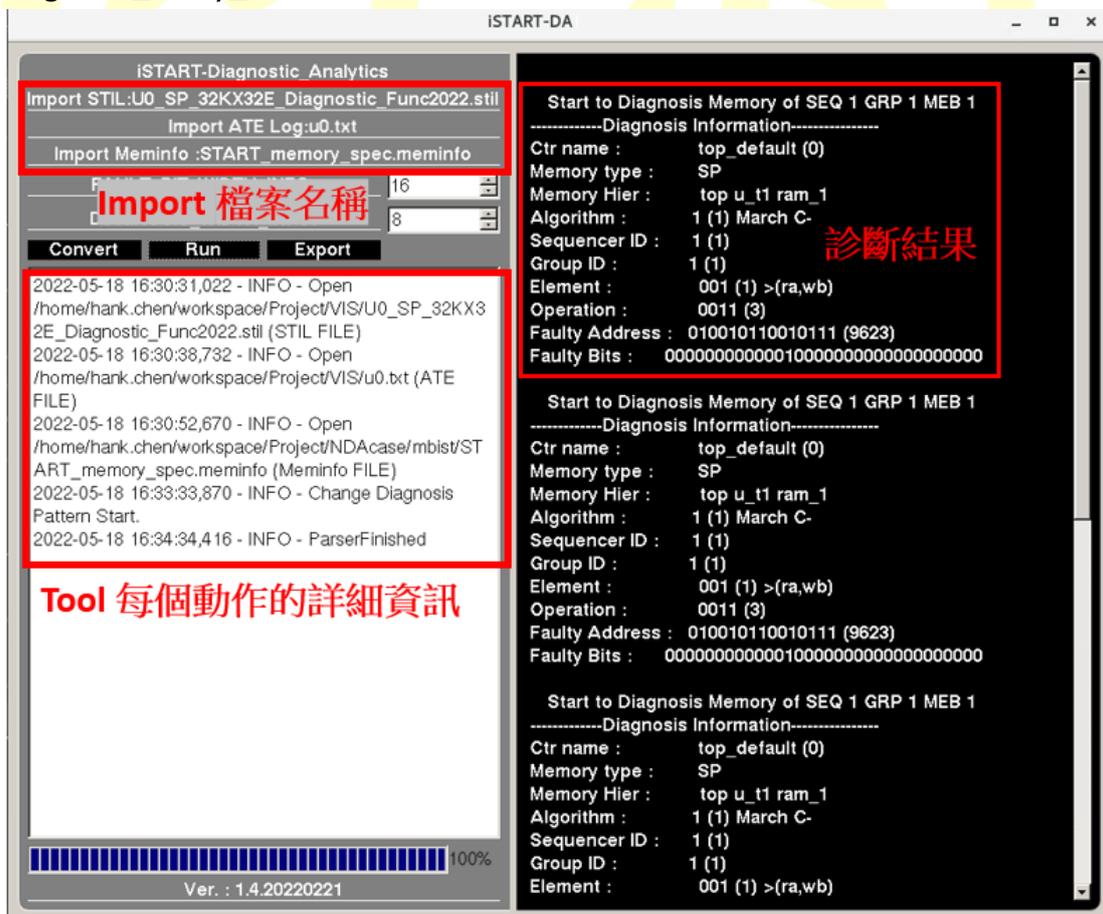
top u_t1 u_dp
```

- D. 設定 FAULT_BIT_WIDTH 與 DAGNOSIS_WIDTH。

2. Convert(資料轉換鍵)：當輸入完所有檔案，即可執行轉換。
Run(執行按鍵)：轉換完成後，即可執行分析。
Export(輸出檔案按鍵)：可將分析結果用 txt 檔做輸出。
3. Tool 執行資訊：Tool 所執行的資訊與狀態都會在此說明
4. 執行的進度條與版本號。
5. Tool 分析結果：此區會產生 Tool 分析的結果。

分析方式：當工具輸入完所有檔案與設定之後，會先分析 ATE 的結果檔，會先找尋在哪一個 Pattern 出錯，並對照 STIL 檔案找出對應的一整段 test command 兩者再做分析，分析完後，要將結果輸出時會對照著 meminfo 檔案中 memory 的順序印出有問題的 memory 資訊。

分析結果：工具分析如下圖，右邊區塊會顯示測試有誤的記憶體之詳細資訊，詳細資訊中包含：控制器、記憶體種類、記憶體階層以及在 START™ v3 的 diagnosis_faulty_items 所設定的資訊。



作者：陳政穎 高級工程師