

JTAG를 이용한 철도 종합제어 장치의 DEBUGGING

송용수, 이수길, 신승권, 한성호
한국철도기술연구원

Debugging Of TCMS(Train Control and Monitoring System) In Use JTAG

Song Yongsoo, Lee Sugil, Shin Seungkwon, Han Seongho
Conventional Rail Engineering Corps

Abstract - ARM CORE용 칩으로 철도 종합 제어 부분에 이용될 수 있는 main processor 부분을 설계하고, JTAG 기술을 이용하여 그 안정성과 토폴링 기술에 이용될 수 있는 process를 사전 단계에서 bebugging 해보고, 이에 따른 신호 및 성능을 JTAG(Boundary Scan)을 이용하여 시스템의 신호와 과형을 시험 평가한다. 또한 예비 단계로의 JTAG 검증 가능성 여부를 알아보고자 한다. 철도 종합 제어 시스템의 신호 및 정확성을 측정해 보기 위한 선행 연구라 할 수 있다.

Keyword: JTAG(Boundary Scan), 토폴링 기술, 철도 종합 제어 시스템

1. 서 론

JTAG(Joint Test Access Group)은 표준 IEEE Std 1149.1-1990, IEEE Standard Test Access Port and Boundary-scan Architecture의 표준이다. 요즘에 나오는 모든 하드웨어에는 JTAG port가 들어가 있다. 그러나 보통은 host board와 target board의 독립적인 통신을 통해 프로그램을 로딩 하는 수준이다. 시스템이 완성되면 전원에 인가된 부분을 먼저 검사해서 우선 라 디바이스별 전원의 공급 선이 이상이 없는가를 확인한다. 물론 테스트나 점퍼 테스터를 이용하여 하나씩 핀들의 연결 관계를 시험하고, 여기서 단자들끼리 납땜이 잘 되어 있나 전원 공급이 잘 되었나를 일일이 확인 해야한다. 그러나 8비트 이전에 나온 칩들은 핀 수가 작아서 몇 번의 시행착오를 거치면 되지만 200핀 정도의 ARM은 무리한 일이다. 이에 따라 인터페이스를 내장하게 되는데 이게 바로 JTAG이다. 본 연구는 철도 종합제어 시스템에 이용될 수 있는 시스템을 설계하고, 이 시스템을 JTAG를 이용하여 선들의 배선 여부와 Chip의 Identification을 확인해 보았다. 본 논문에서는 이에 따라 철도에 이용되는 종합 제어 장치를 JTAG으로 설계시 오차나 칩들의 배치 여부와 동작 작동여 부를 확인해 보았다. 이는 향후 완성될 시스템의 선행 단계라 할 수 있다.

2. 본 론

2.1 Boundary 스캔의 정의

개발자가 하드웨어를 테스트하다 보면 문제의 원인을 알아내기 위해서 특정 단자의 상태를 임의로 설정해 줄 필요가 있다. 그런데 각 디바이스는 PCB상에서 이미 다른 디바이스와 연결되어 있어서 강제로 인가하려면 핀의 연결을 끊고 강제로 인가하는 수밖에 없다. 그런데

여러 조건을 시험하려면 이렇게 하는 것이 그리 쉽지 않은 않다. PCB나 BGA타입이라면 PCB 기판을 강제로 끊어 버리는 것이 쉬운 일은 아니다. 그래서 입력 핀들과 LSI 로직 사이에 셀을 넣어서 로직을 강제로 인허가 하는 것이 boundary scan의 원리이다. 만약 boundary 스캔 셀이 없다고 가정하면 아래 그림1-1을 보면 LSI 1-1에 입력이 인가되면 LSI 1-2를 통해서 신호는 LSI 1-3으로 나가게 됩니다. 만약 회로에 이상이 발생하게 되면 LSI 1-2 와 LSI 1-1 사이를 강제로 연결을 제거시키고 회로를 검사 해야한다. 그러나 boundary-scan을 이용하게 되면, 다시 말하자면 중간에 인위 적으로 그 관계들 인허가 하는 셀이 존재한다면 아주 편하게 debugging 할 수 있다. 그러나 이런 셀 또한 200핀 짜리 arm_core 안에서는 편마다 각각의 셀을 단다면 아주 비효율적이다. 그래서 보통Boundary 스캔 셀은 슈프트 레지스터 형식으로 되어 있다.

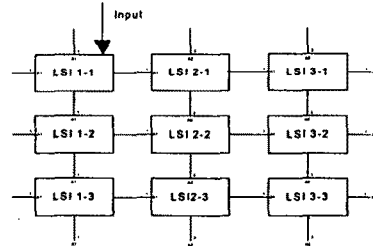


그림 1-1 Boundary_scan의 내부 구조

2.2 Boundary_scan의 내부구조론

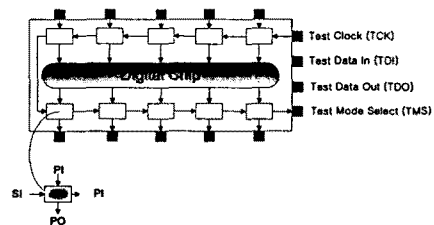


그림 2 Scan 셀의 내부 구조

그림2 은 Scan_cell의 내부 구조도 이다. 이 그림에서 PI(Parallel Input)으로 외부 신호 입력 또는 디지털 로직에서 나오는 출력으로 셀 입장에서 입력으로 볼 수 있

다. PO(Parallel Output)는 외부 신호 출력 또는 디지털 로직에 인가되는 입력으로 Cell입장에서는 출력으로 볼 수 있다. SI(Scan Input)는 직렬 입력단자이고, SO(Scan Output)는 직렬 출력단자이다.

2.2.1 Boundary_cell의 각 기능

Boundary_cell의 기능을 확인해 보면 다음과 같다.

- PI의 신호를 캡처하는 기능.
- PI에 들어 있는 신호를 메모리에 기억시키는 기능
- 메모리에 있는 내용을 PO에 인가하는 기능
- PI의 신호를 PO에 바로 넘기는 기능
- SI의 입력을 메모리에 기억 및 메모리의 내용을 SO에 옮겨오는 기능

이런 기능을 이용하여 각 CELL의 내용을 외부에서 읽어 볼 수 있고 특정 상태를 내부에 인가할 수 있다. 이런 기능들은 외부에 있는 래지스터에 의해 제어 받게 된다.

하나의 boundary scan cell을 확대하면 다음과 같습니다.

Cell은 앞과 뒤에 멀티플렉서를 갖는 두개의 D-Flipflop으로 구성되어 있다.

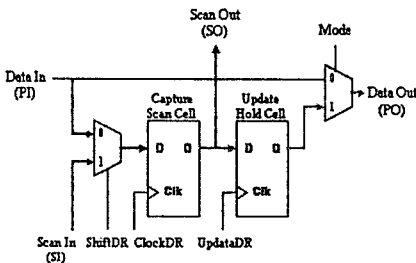


그림 3 Scan_cell의 확대

2.2.2 Boundary_cell의 원리

(1) Normal Mode 일 때, Data In(PI)는 Data Out(PO)로 바로 전달되고, 즉, Mode가 0이면 PI신호가 PO에 바로 전달이 된다.

(2) Update Mode 일 때, Update Hold Cell의 내용이 Data Out(PO)로 전달되고, 즉, Mode가 1이고, UpdateDR에 한 클럭을 주면, 메모리에 있는 내용용에 인가한다.

(3) Capture Mode 일때, Data In 신호는 Capture Scan cell의 입력으로 전달되고, 그 값은 다음 ClockDR에 의해 Capture 된다. 즉, Data In(PI)신호는 ShiftDR을 0으로 하고, ClockDR에 한 클럭을 주면 Scan Out(SO)로 인가된 상태로 PI신호를 Capture 한다.

(4) Shift Mode 일때, 하나의 Capture Scan Cell의 Scan Out은 배선의 경로를 통하여 다음 Capture Scan Cell의 Scan In으로 전달 된다. 즉, SI의 입력을 메모리에 기억시키고, 메모리의 내용을 SO에 옮겨놓는다.

capture와 shift 동작은 병렬 입력 단자에서 병렬 출력 단자로의 데이터 전달을 간섭하지 않는다는 점을 유의해야 한다. 따라서, 동작 중인 값들을 capture하고, 검사의 목적으로 이러한 값들을 간섭없이 shift 시킬 수 있다.

2.3 하드웨어 설계

2.3.1 하드웨어 사양과 기능

본 논문에서 제안된 제어 시스템은 열차 내의 통신 또는 제어 시스템의 응용에서 이용될 수 있다. 다음은 본 논문에서 구성한 하드웨어의 세부 사양이다.

(1)Processor - Cirrus Logic : EP7312

ARM core를 기반으로 하는 칩(SOC: System On Chip)은 다양하다. 하지만, 각각의 칩이 그 나름대로의 특성을 가지고 있다. EP7312는 Communication과 Multimedia Functioning에 최적화되어 있는 칩이다.

Processor는 Embedded System에 있어서 가장 핵심이 되는 것으로, 이 칩의 성능, 주변 장치와의 interface 등을 고려해야 한다. 따라서 본 논문에서는 Cirrus Logic의 EP7312는 멀티미디어(특히 Digital Audio File의 decoding과 LCD interface) 기능과 Auto Control 구현에 매우 적합한 구조를 갖추고 있다. ARM7TDMI core의 강력한 기능과 함께 LCD controller내장, memory interface, digital audio interface 등을 갖춘 최적화된 칩이라 할 수 있다. 다음 그림은 프로세서의 내부 블록도이다.

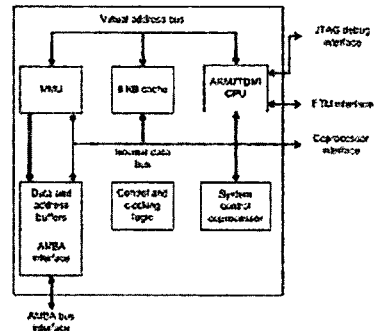


그림 4 Processor 내부 블록도

이 프로세서는 48KB SRAM을 가지고 있고 또한 SRAM은 고속의 동작을 위해 필요한 것으로, 개발자의 설정에 따라 Cache나 SRAM 그 자체로의 사용이 가능하다. 최대 90MHz의 clock speed : RISC Chip의 경우 동작이 간단하기 때문에, 하나의 동작을 수행하는데 많은 시간을 필요로 하지 않는다. 또한, pipelining 기능을 봉하면 거의 1clock에 1operation의 수행이 가능하다. 여기에서 90MHz의 clock speed는 열차 제어에 이용할 수 있는 Control적으로 decoding하기에 충분한 clock speed이다.

(2)LCD controller

LCD panel을 구동하기 위해서는 controller와 driver IC가 필요하다. Driver IC는 직접적으로 LCD panel을 구동시키는 부분이며, controller는 processor등의 user level controller와의 interface라고 생각하면 된다. 즉 controller에 따라 firmware programming이 달라질 수 있다. 일반적으로 LCD module에는 driver IC가 내장되어 있기 때문에, 개발자는 controller만을 따로 선택하여

board를 설계하면 되는데, EP7312에서는 chip내부에 LCD controller가 내장되어 있으므로, 단지 firmware programming만 하면 된다.

(3) JTAG and Embedded ICE

JTAG은 보통 chip의 상태를 점검하기 위해 chip내부와 통신을 할 수 있도록 구현되어 있는, 일종의 규약이라고 생각할 수 있다. 이 protocol을 이용하여 ICE(In Circuit Emulator)의 구동도 가능하게 된다. 이 JTAG을 이용하여 Flash memory Programming등 여러 가지 기능을 구현할 수 있다.

(4) NOR Flash Memory

NOR Flash는 NAND Flash와 달리, Random Access에 적합하며, 빠른 응답성능을 보인다. 그렇지만, 제조 원가가 높기 때문에, 가격이 비교적 낮게 책정된다. 이러한 이유로 NOR Flash는 주로 Program code를 저장하고 실행하는 용도로 사용된다.

본 board에서는 128Mbits(16MB)*2를 사용하고 있다. 이는, 단순히 Control등을 구현하기에는 과도한 용량으로 생각될 수 있으나, eCOS나 embedded LINUX등의 embedded OS를 적재하기 위해서는 최소한 이 정도의 용량은 필요할 것으로 생각되어 채택하고 있다.

(5) SDRAM

SRAM보다는 느리지만 용량이 크고 비교적 빠른 편이다. SDRAM(혹은 DRAM)은 processor가 여러 가지 연산을 하기 위해 임시 memory공간으로 사용할 수 있다. 또한, MMU에서의 page table등으로도 사용될 수 있다. 뿐만 아니라 LCD의 픽셀에 대한 정보(on/off 혹은 색 정보)를 담고 있는 LCD 비피로씨도 사용될 수 있다. 본 논문에서는 이것을 2개 사용하여 32MB를 구현하고 있다.

(6) ADC

ADC는 digital recording을 위한 것이다. 보통 소량의 데이터만 처리하는 것이라면, 낮은 sampling주파수만 지원해도 되고, 분해능도 작아도 될 것이다. 하지만, 본 보드에서는, 향후 고성능의 연차 제어와 여러 가지 센서들의 입력을 처리할 계획으로 비교적 고성능 ADC를 사용하고 있다. ADC의 특징은 다음과 같다.

- 24bit conversion, 96khz sample rates,
- 1.8V~3.3V supply

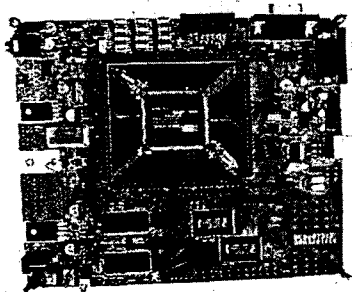


그림 5 완성된 하드웨어

2.4 JTAG Debugging

본 논문에서는 JTAG을 이용하여 메인 프로세서 ID를 Checking 하였고 또한 Flash 메모리의 크기 및 남아있는 용량과 다운로드 하려는 파일 크기, 비피와 블록 크기 등을 알아보았다. 그림6은 도스 상에서 JTAG 프로그램 실행시킨 모습입니다.

```
D:\FARM_Book\program\BEBoS_JTAG_FINAL\Debug>behos_jtag test.bin
ID OK!
EP7312 ID = 00001111000011110000111100001111
0x890889
0x170017
Flash ID = 0x89178917

File size = 44 bytes
max erase time = 16 seconds
device size = 8388608 bytes
write buffer size = 16(*16) bits
number of blocks = 64 ea.

Block erase operation is started.
Erasing block 0
Erasing block 1
Erasing block 2
Erasing block 3
Selected Block is erased.

Programming operation is started.
Flash is Programmed.
```

그림 6 JTAG 프로그램 실행 모습

3. 결 론

3.1 실험 결과 및 분석

보통 TCMS은 주요 장치의 제어와 각 기기들간의 상태를 모니터링 하여 운전자 및 검수원에게 지원하고 열차의 최대 성능을 유도하는 중요한 핵심 장치이다. 본 논문에서는 전동차 종합 제어 장치(Train Control and Monitoring System)에 들어 갈 수 있는 메인보드의 개발 초기 단계에서 제기 될 수 있는 기기들간의 인터페이스의 효율성을 증대시키고 검사하는 목적에서 시행되었다. 검사 결과 현재 모든 프로세서의 존재하는 JTAG을 이용하여 다른 하드웨어들간의 인터페이스 문제를 초기에 발견할 수 있음을 보여 주었고, 더 나아가 프로세서의 기술이 발전하여 프로세서의 종류가 변화하더라도 JTAG의 기술을 인지하고 이용하면 프로세서의 종류에 관계없이 debugging 할 수 있음을 보여 주었다.

[참 고 문 헌]

- [1] www.cirrus.com
- [2] 종합 제어장치 연구 개발 보고서, 1997. 한국철도기술연구원, 건설 교통부
- [3] Vernazza, G.; Zunino, R.: "A distributed intelligence methodology for railway traffic control" Vehicular Technology, IEEE Transactions on , Volume: 263-270: Issue: 3, Aug 1990.
- [4] Goundan, P.R.; Jhunjhunwala, A.: "Evaluation of relative merits/demerits of different signaling schemes used at a railway station for an approaching train in Indian Railways today Vehicular Technology", IEEE Transactions on, Page(s): 561-566, Volume: 51 Issue: 3, May 2002
- [4] Oued, T.; Weiss, S.: "Embedded instruction memory in automotive engine controllers"; Vehicular Technology, IEEE Transactions on , Volume: 52, Page(s): 173-183 Issue: 1, Jan 2003