

다중 SoC를 지원하는 JTAG Writer에 관한 연구

박영리*, 노영섭*

*서울벤처정보대학원대학교 임베디드시스템학과
e-mail: ysroh@suv.ac.kr

A Study on JTAG Writer for multiple SoCs

Ling-Li Piao*, Young-Sup Roh*

*Dept of Embedded Systems, Seoul University of Venture & Information

요 약

본 논문에서 연구하고 구현된 JTAG(Joint Test Action Group) Writer는 하나의 SoC(System On a Chip)만 지원하도록 설계된 기존 제품의 단점을 보완할 수 있도록 각 SoC의 제조 회사에서 제공하는 BSDL(Boundary Scan Description Language)을 이용하여 여러 가지 SoC에 쉽게 사용할 수 있도록 모듈화 했다. 그리고 기존 제품들이 사용하고 있는 직렬 포트나 병렬 포트 대신 안정적이고 편리한 USB(Universal Serial Bus) 접속규격을 지원하도록 개선했다.

1. 서론

JTAG 접속규격은 BSC(Boundary Scan Chain)을 이용하여 소자의 상태와 관계없이 소자가 제공하는 모든 신호를 제어하거나 읽을 수 있다. 따라서 JTAG를 이용하면 소자에서 제공하는 신호들의 제어를 통하여 회로의 전기적 연결 상태에 대한 시험, 연결된 소자의 동작 확인, 제품의 초기 프로그램의 적재 등에 사용할 수 있다.

대부분의 개발환경에서 사용되고 있는 JTAG Writer는 보통 하나의 SoC만을 지원하고 있어 개발환경이 바뀔 때마다 새로운 JTAG Writer를 사용해야만 한다. 따라서 본 논문에서는 여러 가지의 SoC에 사용될 수 있도록 JTAG Writer의 구조를 개선하여 모듈화 하였다. 또한, 개발된 JTAG Writer는 JTAG 표준 접속 규격을 적용하였기 때문에 JTAG 접속을 사용할 수 있는 SoC(System On a Chip)를 사용하여 설계하는 모든 제품의 생산 또는 개발 초기에 유용하게 사용될 수 있다.

2. 관련연구

JTAG는 IEEE 1149.1에 규격이 정의되어 있으며, 이를 지원하는 모든 칩은 칩 내부에 바운더리 셀(boundary cell)이란 것을 설치하여 외부의 핀을 개별적으로 제어할 수 있도록 구성되어 있다. 따라서 JTAG 접속을 이용하면 프로세서가 할 수 있는 모든 동작을 인위적으로 제어할 수 있어 여러 가지 하드웨어 시험이나 연결 상태의 검사 등에 활용할 수 있다.

JTAG 블록은 4개의 레지스터와 이들을 적절하게 TDI 또는 TDO에 연결해 데이터를 주고받을 수 있도록 제어해주는 TAP 제어기로 구성되어 있다.

그림 1의 칩 구조에서 명령 레지스터(Instruction

register)는 입력된 명령을 기반으로 내부의 기능을 제어한다. 통과 레지스터(Bypass register)는 TDI(Test Data In)와 TDO(Test Data Out)을 연결하여 입력된 데이터를 바로 출력으로 연결한다. ID 레지스터(Identification register)는 소자를 구별하기 위한 레지스터이다. BSC는 입출력 핀을 제어하거나 상태를 읽어 들일 수 있는 레지스터이다.

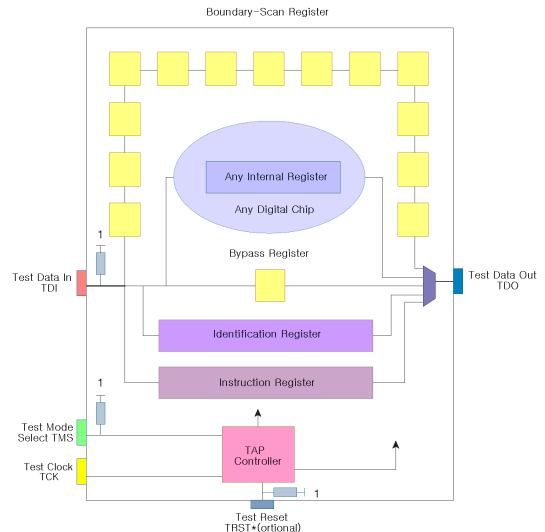


그림 1 . IEEE 1149.1 칩 구조[5]

3. JTAG Writer 설계 및 구현

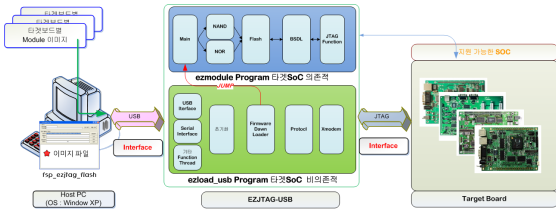
3.1 다중 SoC를 지원하기위한 방안

일반적인 JTAG Writer는 대부분 한 가지의 SoC만 지

원되도록 구현되어 있어 다양한 개발환경에 적용하기에 어려움이 있기 때문에, 본 논문에서는 JTAG Writer에 독립적인 SoC를 사용하여 타겟보드의 접속과 제어를 담당하도록 구성하고, SoC의 제조회사에서 제공하고 있는 BSDL을 사용하여 각 SoC에 쉽게 대응할 수 있도록 내부 구성을 타겟 비의존적인 부분과 타겟 의존적 부분으로 나누어 그림 2과 같이 설계했다.

타겟 비의존적인 부분은 JTAG Writer의 부트로드 (boot loader)에 해당하는 부분으로서 호스트 컴퓨터와 접속을 담당하는 USB 접속 프로그램과 직렬 접속 프로그램, 윈도우용 프로그램과 통신을 담당하여 SoC별 모듈 프로그램 다운로드하는 X-modem 프로그램, 통신 프로토콜을 제어하는 프로그램, 최초 전원이 인가될 때 동작하는 초기화 프로그램을 포함하고 있다.

타겟 의존적인 부분은 모듈프로그램의 전체 흐름을 제어하는 프로그램, 낸드(NAND) 플래시와 노어(NOR) 플래시의 제어, 읽기, 쓰기의 프로토콜을 담당하는 프로그램, BSDL을 해석하는 프로그램, JTAG 접속 신호의 제어를 담당하는 프로그램 등으로 구성했다. 여기서 다양한 SoC를 지원하도록 타겟 의존적인 부분을 최적화하고 모듈형태로 구성하여 새로운 SoC를 지원하기 쉽도록 구성했다.

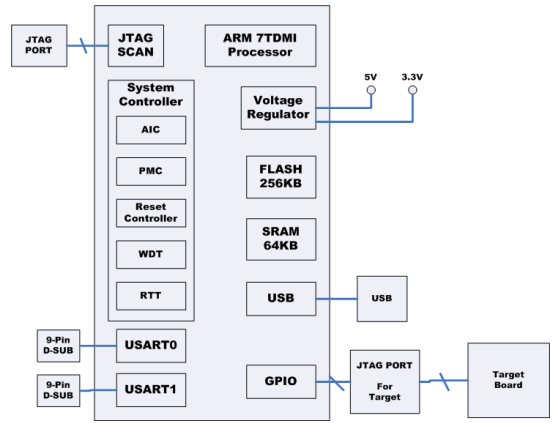


(그림 2) JTAG Writer 모듈 프로그램 구현방법

3.2 JTAG Writer 하드웨어 설계

그림 3은 AT91SAM7S256을 이용한 JTAG Writer의 하드웨어 구성도이다. AT91SAM7S256 SoC는 256KB의 플래시 메모리, 64KB의 SRAM, 시스템 클럭 및 파워관리를 위한 전력 제어기, 타이머, 인터럽트 컨트롤러, USB 포트, 그 밖에 모니터링을 위한 UART 등의 인터페이스를 가지고 있다.

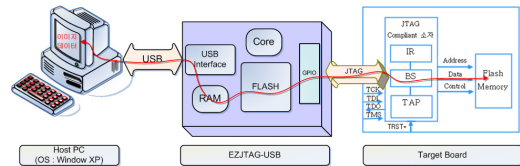
JTAG Writer는 AT91SAM7S256 SoC를 바탕으로 설계되었으며, AT91SAM7S256 안에 있는 ARM7TDMI를 접근하기 위한 JTAG 포트와 모니터링 부분을 위한 UART, 시스템 전원을 위한 전압조정기, 컴퓨터와의 데이터 전송을 위한 USB포트, 그리고 JTAG Writer로 제어할 타겟 보드(Target Board)와 JTAG 접속을 연결하기 위한 GPIO(General Purpose Input Output) 블록으로 이루어져 있다.



(그림 3) JTAG Writer 시스템 구성도

3.3 JTAG Writer 소프트웨어 설계

타겟보드의 플래시 메모리에 이미지 파일을 다운로드하려면 우선 호스트 컴퓨터에서 이미지 데이터를 JTAG Writer의 RAM에 다운로드하고 JTAG 접속을 제어하는 프로그램을 구동하여 GPIO를 제어함으로써 타겟보드와 접속 가능하게 하여 RAM에 저장된 이미지를 타겟보드의 원하는 플래시 주소에 쓰도록 해야 한다. 즉 그 흐름은 아래 그림 4과 같이 표현할 수 있다.

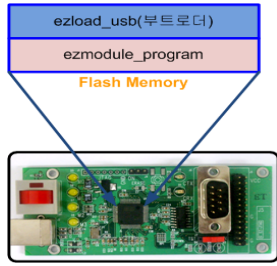


(그림 4) 타겟 보드 플래시 쓰기 시스템

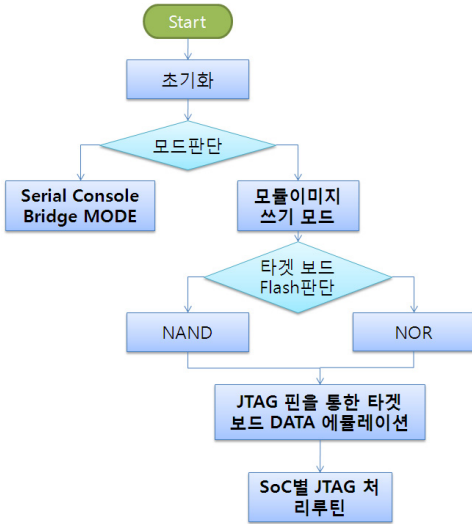
그림 5에서 ezload_usb 프로그램은 JTAG Writer의 부트로드 프로그램이라 할 수 있으며 이는 주로 타겟보드 비의존적 소스를 포함하고 있다. 즉 타겟이 달라도 이 부분의 프로그램은 변하지 않는다. ezload_usb 프로그램은 모듈 이미지 쓰기 및 시리얼 콘솔 브릿지로 구성되어 있다.

모듈 이미지 쓰기는 JTAG Writer 플래시 메모리 특정 공간에 다음 실행할 모듈 프로그램을 다운로드 받는 기능과 다운 받은 모듈 프로그램으로 제어권을 넘기는 기능을 수행한다.

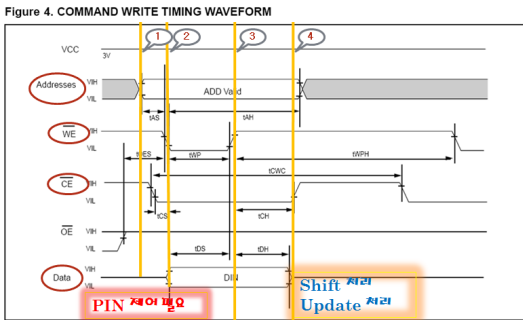
시리얼 콘솔 브릿지는 일반 직렬포트를 USB로 변환하는 장치와 같은 동작을 지원한다. 타겟보드의 콘솔장치 즉 시리얼 인터페이스를 USB로 연결 시키는 기능을 제공하여 호스트 컴퓨터에서 터미널 프로그램(시리얼)을 이용할 수 있다.



(그림 5) JTAG Writer 프로그램 구성



(그림 6) JTAG Writer의 프로그램 흐름도



(그림 7) 플래시 쓰기 타이밍

그림 5에서 ezmodule_program은 호스트 컴퓨터와 JTAG Writer를 USB 또는 Serial 인터페이스를 이용하여 호스트 컴퓨터에 있는 모듈 이미지를 Xmodem(전송 프로토콜)을 이용하여 수신한다. 그 후 JTAG Writer는 전송 받은 이미지를 플래시 메모리의 특정 위치에 저장하고 부트로더의 제어권을 모듈 프로그램에게 넘긴다.

JTAG 기능을 구현한 프로그램은 타겟 보드의 BSDL

파일에 의한 핀을 제어한다. 즉 타겟 플래시의 원하는 주소, 데이터, 컨트롤 핀에 대한 바운더리 레지스터 값을 설정한다. JTAG Writer의 TAG 포트 접속은 기본적인 TAP 접속 즉, 컴퓨터와 JTAG 포트 사이의 입출력을 지정하고 더불어 외부 핀들의 상태를 지정 또는 읽어 오는 기능들이 구현한다. 메모리제어 블록은 JTAG 포트 인터페이스를 통하여 외부의 메모리를 제어하기 위해 읽기 또는 쓰기 동작을 하기 위해서 그때 마다 필요한 각각의 주소 및 데이터 그리고 제어 신호들을 규정하여 구현한다. 마지막으로 플래시 제어 블록은 플래시 메모리와 관련된 기능들을 구현하게 된다. 그림 6은 JTAG Writer에 내장된 프로그램의 흐름도이다.

본 논문에서는 타겟보드의 주소, 데이터, 컨트롤 신호를 제어하여 그림 7과 같은 플래시 쓰기 타이밍에 맞도록 쓰기 알고리즘을 구성하였다.

3.4 윈도우 응용 프로그램

윈도우에서 작성된 응용 프로그램은 각각의 SoC에 맞도록 타겟보드를 설정하고, 전송하려고 하는 이미지 파일을 찾아 JTAG Writer에 전송한 후 결과를 받아 사용자에게 알려 주는 기능을 한다.

3.5 연구의 구현결과

이상과 같이 구현된 JTAG Writer는 목표로 하였던 S3C2440, S3C2410, AU-1200, 그리고 PXA255을 사용한 타겟보드에서 적절히 동작함을 확인했다. 현재까지 지원 가능한 SoC의 목록은 표 1과 같다.

<표 1> JTAG Writer로 제어 가능한 SoC 목록

SoC	Target Board
PXA255(ARM Core)	EZ-X5
AU-1200(MIPS Core)	EZ-AU120
S3C2410A(ARM Core)	EZ-S2410
S3C2440A(ARM Core)	EZ-S2440

4. 결론

본 연구에서 구현한 JTAG Writer는 USB 인터페이스를 사용하여 기존의 직렬 포트를 사용한 제품에 비하여 동작 속도와 안정성을 월등히 높였으며, 내부 구성을 S3C2410, S3C2440, PXA255, AU-1200등의 여러 가지 SoC에 사용 가능하도록 모듈방식으로 설계 및 기능을 구현하였다. 구현된 JTAG Writer는 각 SoC가 탑재된 타겟 보드를 사용하여 직접 동작을 통하여 구현 결과를 검증하였다.

참고문헌

- [1] 김재성, “JTAG Port를 이용한 On-Board Flash Fusing JTAG의 이해와 구성”, 임베디드 월드, pp 84~91, 2003
- [2] 김재성, “JTAG Port를 이용한 On-Board Flash Fusing JTAG의 동작원리”, 임베디드 월드, pp 50~60, 2003
- [3] 유영창, “JTAG의 소개 및 원리”, FA-LINUX, pp 1~22, 2002
- [4] 박철, “JTAG_How_To”, KELP, pp 3 ~ 16, 2001
- [5] IEEE, “IEEE Standard Test Access Port and Boundary-Scan Architecture”, IEEE Standard 1149.1, pp 1~41, 2001
- [6] ARM, “ARM7TDMI Technical Reference Manual”, pp 1~286, 2004