

EA0028: 임베디드 소프트웨어 I 실습

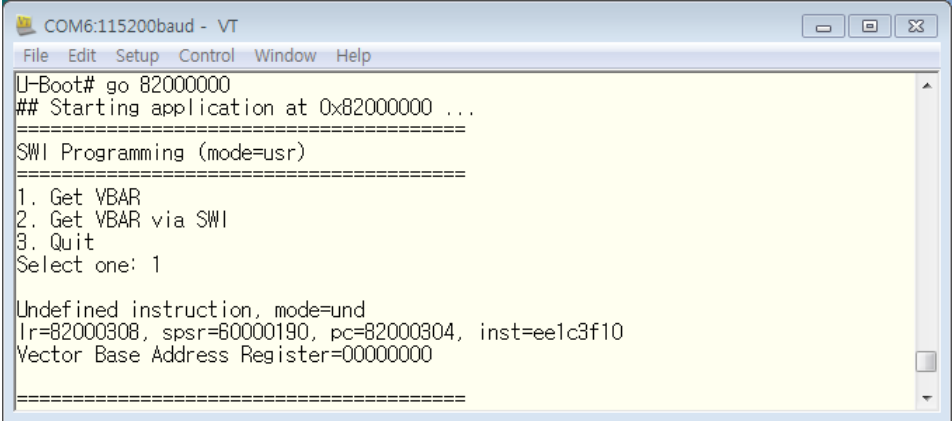
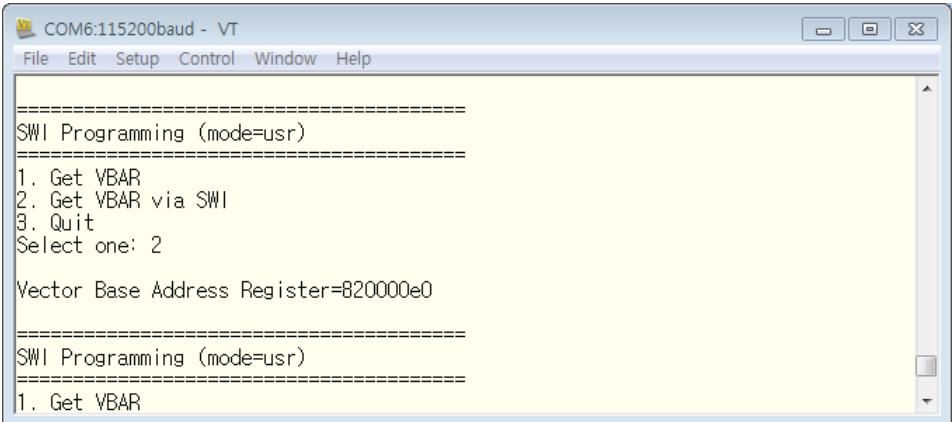
상명대학교 컴퓨터과학과

2018년 1학기

실습 번호	06	실습 점수	/7
실습 날짜	2018년 월 일	실습 폴더	~/es1/lab06
학생 이름		학번	
실습 제목	SWI Programming		
참고 자료	1. 신동하, 6 Exception Programming, 임베디드 소프트웨어 I 강의 자료, 2018. 2. ARM Limited, ARM® Architecture Reference Manual, ARM DDI 0100I, July 2005. (Chapter A2.6) 3. ARM Limited, ARM® Architecture Reference Manual ARM®v7-A and ARM®v7-R Edition, ARM DDI 0406B, July 2009. (Chapter B1.6)		

실습 번호	1	점수	3						
실습 내용	<p>본 실습 프로그램은 ARM processor의 usr mode에서 수행된다. 본 실습은 usr mode에서 수행하면 undefined exception이 발생하거나 부정확한 결과를 주는 "privileged instruction"을 SWI instruction을 통하여 svc mode에서 수행하여 정확한 결과를 줄 수 있게 프로그래밍하는 실습이다. 본 실습을 위하여 아래 표에 설명한 프로그램 파일을 작성하라.</p>								
	<table><tr><th>파일</th><th>프로그램 설명</th></tr><tr><td>start.S</td><td>이 부분은 이전 실습 05의 Exception Programming에서 작성한 프로그램 파일 start.S와 같이 undefined exception 및 swi exception이 순조롭게 처리될 수 있도록 exception vector를 프로그래밍 한다.</td></tr><tr><td>swi.c</td><td><p>이 파일의 함수 main()은 coprocessor 15에 있는 register 12(=VBAR register)를 읽는 coprocessor instruction "MRC"를 메뉴 1에서는 usr mode에서 수행하고, 메뉴 2에서는 swi instruction을 통한 svc mode에서 수행한다. User mode에서 instruction MRC를 수행하면 결과 값도 부정확하지만 undefined exception이 발생한다. 함수 main()은 제공된다. 함수 svc_to_usr_mode() 및 get_and_print_vbar()은 아래와 같이 프로그래밍 한다. (주: 이들 함수의 일부는 ARM asselbler로 프로그래밍하여야 한다.)</p><ul style="list-style-type: none">함수 svc_to_usr_mode(): processor의 mode가 reset 후 svc에 있다. 이를 usr mode로 변경한다함수 get_and_print_vbar(): coprocessor 15의 register 12인 VBAR를 읽어서 화면에 출력한다.</td></tr></table>			파일	프로그램 설명	start.S	이 부분은 이전 실습 05의 Exception Programming에서 작성한 프로그램 파일 start.S와 같이 undefined exception 및 swi exception이 순조롭게 처리될 수 있도록 exception vector를 프로그래밍 한다.	swi.c	<p>이 파일의 함수 main()은 coprocessor 15에 있는 register 12(=VBAR register)를 읽는 coprocessor instruction "MRC"를 메뉴 1에서는 usr mode에서 수행하고, 메뉴 2에서는 swi instruction을 통한 svc mode에서 수행한다. User mode에서 instruction MRC를 수행하면 결과 값도 부정확하지만 undefined exception이 발생한다. 함수 main()은 제공된다. 함수 svc_to_usr_mode() 및 get_and_print_vbar()은 아래와 같이 프로그래밍 한다. (주: 이들 함수의 일부는 ARM asselbler로 프로그래밍하여야 한다.)</p> <ul style="list-style-type: none">함수 svc_to_usr_mode(): processor의 mode가 reset 후 svc에 있다. 이를 usr mode로 변경한다함수 get_and_print_vbar(): coprocessor 15의 register 12인 VBAR를 읽어서 화면에 출력한다.
	파일	프로그램 설명							
	start.S	이 부분은 이전 실습 05의 Exception Programming에서 작성한 프로그램 파일 start.S와 같이 undefined exception 및 swi exception이 순조롭게 처리될 수 있도록 exception vector를 프로그래밍 한다.							
swi.c	<p>이 파일의 함수 main()은 coprocessor 15에 있는 register 12(=VBAR register)를 읽는 coprocessor instruction "MRC"를 메뉴 1에서는 usr mode에서 수행하고, 메뉴 2에서는 swi instruction을 통한 svc mode에서 수행한다. User mode에서 instruction MRC를 수행하면 결과 값도 부정확하지만 undefined exception이 발생한다. 함수 main()은 제공된다. 함수 svc_to_usr_mode() 및 get_and_print_vbar()은 아래와 같이 프로그래밍 한다. (주: 이들 함수의 일부는 ARM asselbler로 프로그래밍하여야 한다.)</p> <ul style="list-style-type: none">함수 svc_to_usr_mode(): processor의 mode가 reset 후 svc에 있다. 이를 usr mode로 변경한다함수 get_and_print_vbar(): coprocessor 15의 register 12인 VBAR를 읽어서 화면에 출력한다.								

	handlers.c	이 부분은 이전 실습 05의 Exception Programming에서 작성한 프로그램 파일 handlers.c와 유사하게 swi 및 undefined exception의 handler 함수로서 학생이 프로그래밍 하여야 하는 부분이다.
실습 결과	1.1. 작성한 프로그램 파일 start.S, swi.c 및 handlers.c를 출력하고 출력물 우측 여백에 연필로 직접 프로그램의 설명을 적어서 제출하라.	

실습 번호	2	점수	4
실습 내용	<p>앞 실습에서 작성한 프로그램 파일 start.S, swi.c 및 handlers.c를 컴파일하여 swi.bin을 생성하라. 이 프로그램을 수행하고 화면에 나타나는 메뉴 1 - 2를 순서적으로 선택하여 아래와 같은 화면이 나타나는 지를 확인하라.</p> <p>1) 메뉴 1번 선택</p>  <p>2) 메뉴 2번 선택</p> 		
실습 결과	<p>2.1. 위 그림과 같이 프로그램 swi.bin을 BeagleBone 상에서 수행한 두 화면을 출력하여 제출하라.</p> <p>2.2. 수행한 두 화면 상에 출력된 내용의 의미를 자세히 설명하라.</p>		