

원전 제어봉구동장치제어계통 시험설비의 임베디드 제어 프로그래밍 실제

(Practices of Embedded Control Programming for CEDMCSTF of NPP)

장귀숙 · 금종룡 · 박희윤 · 김영국

한국원자력연구소 계측제어 · 인간공학연구부
충남대학교 전기정보통신공학부

{gsjang, jykeum, hypark}@kaeri.re.kr, ykim@cnu.ac.kr

요 약

원전 제어봉구동장치 성능시험을 위한 제어봉구동장치제어계통 시험설비(CEDMCSTF, Control Element Drive Mechanism Control System Test Facility)는 CEDM 성능시험에 필요한 모든 변수에 대해 디지털 신호분석 및 처리를 통해 CRT로 운전원에게 제공하고, 운전원의 제어행위도 소프트웨어방식을 통해 CRT 화면상에서 이루어지도록 하여 기존 CEDMCS의 문제점을 디지털 기술 적용으로 해결하는데 그 목적이 있다. 본 논문에서는 CEDM 운전 및 감시 유니트의 제어봉 삽입, 인출 및 낙하 명령과 CEDM 전력제어유니트에서 검출된 각 CEDM코일의 전류상태를 분석하여, 제어논리신호를 CEDM 전력제어유니트에 출력하는 기능을 수행하는 제어 유니트의 임베디드 제어 프로그램의 실체를 소개한다. 또한 시험을 통해 발견된 논리적인 결함을 수정하여 임베디드 프로그램 수행의 최적화한 결과를 소개한다.

1. 서론

1.1 기존연구

ABB-CE 사의 TF-7 제어봉구동장치(CEDM) 제어계통은 초창기 4 코일(coil) CEDM에 대한 각종 성능시험도구로 사용되었고, 이 후 가동 중 발전소의 서비스 및 영광3,4호기와 울진3,4호기 등 신설발전소의 CEDM 생산시험(production test)에 활용되고 있다. TF-7 CEDM 제어계통의 특징은 아날로그 기반으로 구동전력공급에 따른 CEDM의 기계적 반응 상태에 대한 피

드백을 고려하지 않은 고정식 시간제어방식을 갖는다[1]. 이 방식은 잡음의 영향을 덜 받는다는 점에서 장점이 있으나, 기계적 응답시간과 제어 요구 시간의 불일치가 발생할 경우 예기치 않은 제어봉 낙하 등이 발생할 가능성이 있기 때문에 영광3,4호기 이후의 발전소에서는 마이크로프로세서 기술을 접목하여 전력공급에 따른 기계적 반응을 고려한 피드백 가변구동방식을 채택하고 있다[2]. TF-7 및 영광3,4호기 CEDM 제어계통의 전력제어 방식은 위상제어방식으로서 SCR(Silicon Controlled Rectifier)을

통해 교류전력을 정류하여 고전압/저전압의 형태로 변경시켜 CEDM 코일에 공급하는데, 입력교류의 주파수 변동에 의해 CEDM에 출력되는 전력도 영향을 받는 것으로 밝혀지고 있다[3]. 아울러 스위칭 속도의 제한으로 인해 출력전력의 특성을 개선시키는 데는 한계가 있다.

1.2 연구 목표

본 논문의 CEDMCSTF (Control Element Drive Mechanism Control System Test Facility) 개발전략은 기능적 측면에서는 기존의 CEDM 제어시스템의 기능을 참조로 하여 시험설비 특성에 맞도록 변경하며, 기술적 측면에서는 기존의 장비를 디지털기술이 접목된 새로운 장비로 대체하여 상에서 언급한 기존 CEDM 제어시스템의 부분적인 단점을 보완하는 것이다. 따라서 본 CEDMCSTF는 CEDM 성능시험에 필요한 모든 변수에 대해 디지털 신호분석 및 처리를 통해 CRT로 운전원에게 제공하고, 운전원의 제어행위도 소프트웨어방식을 통해 CRT 화면상에서 이루어지도록 하였다. 구동제어알고리즘은 기존 발전소의 가변 피드백제어방식을 수용하여 본 CEDMCSTF의 개발요건 및 장비 특성에 맞도록 수정한다. 전력제어방식은 기존의 SCR 위상제어방식 대신에 IGBT (Insulated Gate Bipolar Transistor)를 이용한 펄스폭변조(PWM)방식을 채택하여 전력효율, 리플률 등 전력관련 성능을 개선함과 동시에, 입력 주파수의 변동에 따른 출력전압의 영향이 없도록 하고, 전력제어장비 자체를 진단하여 이상 발생 시 조기에 경보하는 능동적 제어개념을 갖도록 한다.

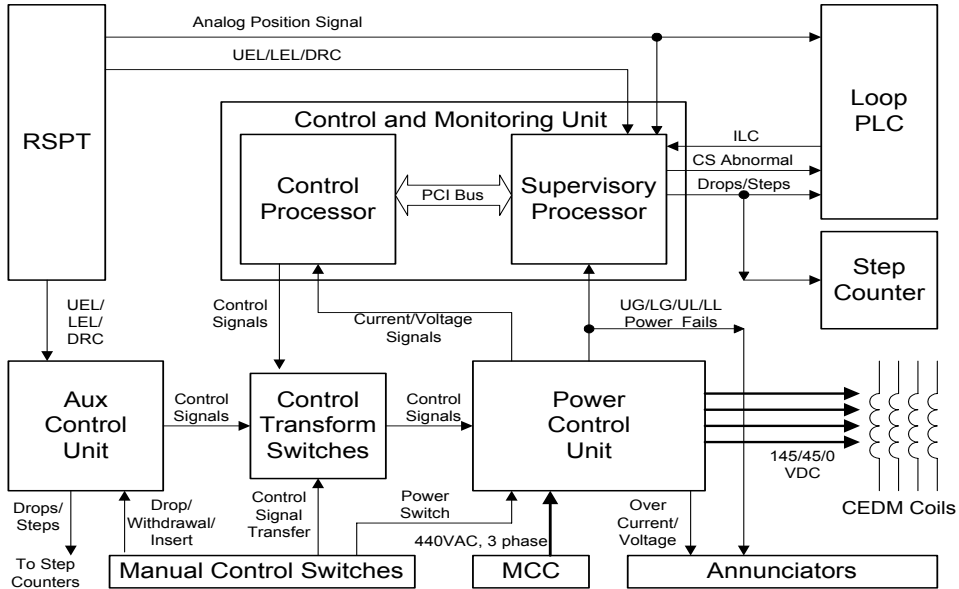
1.3 CEDMCSTF 개요

CEDMCSTF는 CEDM에 전력을 공급하여 CEDM의 구동력(motive power)을 확보하고, CEDM의 운동상태를 분석, 감시함으로써 시험과 관련된 정보를 제공하는데 목적이 있다. 제어기능 요건으로는 CEDM의 구동력과 유지력(holding power)을 제어할 수 있어야 하며, 구동방향 및 속도를 제어할 수 있어야 하고, 가변 시간제어를 위해 CEDM에 흐르는 전류를 측정하여 기계적 거동을 분석하여 다음 제어행위에 피드백되어야 한다.

CEDMCSTF는 CEDM 성능시험에 필요한 모든 변수에 대해 디지털 신호분석 및 처리를 통해 CRT로 운전원에게 제공하고, 운전원의 제어행위도 소프트웨어방식을 통해 CRT 화면상에서 이루어지도록 하여 기존 CEDMCS의 문제점을 디지털 기술 적용으로 해결하는데 목적이 있다[4].

CEDMCSTF는 (그림 1)과 같이 크게 세 유니트로 구분되는데 하나는 제어 및 감시 유니트(Control & Monitoring Unit; CMU)이고 둘째는 전력제어 유니트(Power Control Unit; PCU), 마지막으로 보조제어유니트(Auxiliary Control Unit; ACU)이다.

CMU는 논리제어기능을 담당하는 제어프로세서(Control Processor; CP), 운전원 연계와 각종 감시정보를 처리하는 운전 및 감시 프로세서(Supervisory Processor; SP), 감시 및 운전화면 그리고 이력자료를 저장하기 위한 스텝 계수기(step counter) 등으로 구성된다. 전력제어유니트는 각각의 코일에 대응하는 4개의 Chopping Power Supply Module(CPSM), 절연변압기 등으로 구성된다. 보조제어유니트는 제어기와 입출력 카드 등으로 구성된다.



(그림 1) CEDMCSTF 전체 구성도

CEDMCSTF의 소프트웨어는 제어프로세서에서 수행하는 제어 소프트웨어와 운전 및 감시프로세서 내의 소프트웨어로 나눌 수 있는데, 이 두 종류의 소프트웨어는 각각의 기능을 독립적으로 수행하고, PCI 버스 인터페이스를 기반으로 메일박스와 인터럽트 처리를 통해 통신한다.

본 논문은 CMU 중에서 제어유닛의 임베디드 제어 프로그래밍 사례에 대해 기술한다.

2. 제어 유닛

2.1 기능

제어유닛은 CEDM 운전 및 감시유닛으로부터 CEA 삽입, 인출 및 낙하 신호등을 받은 후 CEDM 전력제어유닛으로부터 피드백되는 각 CEDM 코일의 전류상태를 분석하여 제어논리신호를 CEDM 전력제어유닛에 전달하는 기능을 수행한다. 또한 CEDM

운전 및 감시유닛에 각종 정보, 구동시간, 궤적 등을 제공한다. 제어유닛의 주요기능은 아래와 같다.

- 1) 시험요원 입력, 연동신호 및 피드백 변수를 기반으로 제어봉삽입, 인출, 낙하, 유지 등 구동명령 생성
- 2) CEDM의 기계적 구동을 위한 구동 시퀀스 제어
- 3) CEDM의 4개 코일을 유기적으로 구동할 수 있도록 하기 위한 시간 시퀀스 제어
- 4) 코일에 인가되는 전력량 제어
- 5) 기계적 운동의 적합성을 제어입력에 반영하는 피드백제어
- 6) 감시유닛의 연동신호에 따른 제어 신호 생성
- 7) 이상상태 발생 시 감시유닛으로 경보를 발생하고 홀드(hold)모드로 전환
- 8) 감시를 위한 변수생산

그 세부기능은 아래와 같다.

- 1) 데이터 획득
 - 채널의 전류신호
- 2) 상태분석
 - 자기진단
 - 고전류검출
 - 부적절한 홀딩 전류
 - 출력패턴 점검
 - Engage를 나타내는 Glitch 점
- 3) 제어신호생성
 - 운전원 입력 및 연동신호를 분석하여구동 스킴을 결정
 - 디지털 제어신호 출력

2.2 개발환경

제어유니트는 제어봉 제어를 수행하기 위해 TI TMS320C44 DSP(Digital Signal Processor)를 기반으로 하는 I.I(Innovative Integration)사의

M44보드를 사용한다. M44보드는 OMNIBUS 입출력 기능카드의 집합체를 사용함으로써 데이터 취득 및 제어 작업 수행은 물론 PCI 인터페이스 및 마스터/슬레이브 전송을 통하여 호스트와 데이터 통신을 수행할 수 있고, 또한 모듈라형태로서 입출력 확장이 쉽다. 제어유니트의 개발환경은 (표 1)과 같다.

3. 연계

감시유니트(CSP, 호스트)와 제어유니트(CCP, 타겟)의 통신 및 연계기능을 기술한다.

3.1 통신을 위한 호스트 응용 프로그램 방법

감시유니트와 제어유니트가 서로 상대방의 상태를 파악하고, 정보를 전송하기 위해서 감시유니트는 제어유니트를 제어함은 물론 통신을 수행하기 위한 환경이 아래와

(표 1) 개발환경

항 목	환 경 설 정
하드웨어/운영체제	펜티엄PC/Windows NT 4.5
라이브러리	Zuma 1.5
컴파일러/링커	C3Xtools (TI 5.01버전), C3Xtools (TI-4.7)
에디터	CodeWriter
COFF 파일 실행도구	Terminal Emulator
DSP(Digital Signal Processor)	Texas Instrument's TMS320C44 - 275 MOPS(32bit) - floating-point DSP optimized instruction set for DSP - 40MHz
호스트 PC 연계	PCI 버스 마스터/슬레이브 - 호스트 PC는 인터럽트를 통해 C44 광역SRAM 직접접근
타이머/카운터	○ 1개의 DDS(Directional Digital System) 0.02Hz 해상도 ○ DDS에 의해 구동되는 2개의 16bit 외부 타이머
입출력	모듈라 플러그인을 위한 2대의 OMNIBUS I/O 사이트
개발언어	TI 크로스 개발도구를 사용하는 C 혹은 어셈블리 - 주변기기 라이브러리 - Zuma 도구셋을 통한 윈도우즈 드라이버
소프트웨어	Zuma 도구셋, DLL 등

같이 설정되어 있어야 한다.

- 1) 가상 디바이스 드라이버
 감시유니트의 호스트 디바이스 드라이버는 제어유니트의 인터럽트 신호를 처리 한다.
- 2) 동적 링크 라이브러리(DLL)
 동적 링크 라이브러리는 감시유니트가 다음과 같은 기능을 수행하도록 제어유니트와 상호 동작하는 함수를 제공한다.
 - 기본적인 제어유니트의 타겟보드 제어
 - 제어프로세서 리셋/구동
 - 보드특정 메일박스 레지스터리를 통한 메시지 전송
 - COFF(Common Object File Format) 실행파일다운로드
 - 버스 마스터 메모리잠금 및 접근 제어

3.2 통신방법

CEDMCSTF는 PCI 인터페이스를 통한 광역 SRAM 접근을 이용하여 타겟(CCP)-호스트(CSP)간의 통신을 수행한다. 이와 같은 통신을 수행하기 위해서는 다음과 같은 환경이 설정되어 있어야 한다.

3.2.1 PCI 버스 인터페이스

제어유니트는 감시유니트와의 데이터 전송 및 수신을 위해 고속의 PCI 버스 마스터 /슬레이브 인터페이스를 사용한다.

제어유니트는 감시유니트의 어떤 특정 메모리 어드레스를 직접 접근할 수 있고, DSP의 개입 없이 광역 SRAM으로부터 혹은 광역 SRAM으로 데이터를 전송할 수 있다. 그리고 제어유니트는 제어유니트의 프

로세서(DSP)나 감시유니트의 프로세서의 개입 없이 광역 메모리로 데이터를 쓰거나 혹은 광역메모리로부터 데이터를 읽어오는 버스 마스터 접근을 수행한다. 또한 PCI 인터페이스 하드웨어에는 메일박스 레지스터 셋을 포함하는데, 이것은 한번에 32bit 데이터를 전송하는 기능을 가지는데, 이 메일박스는 호스트 PCI 버스의 I/O 영역으로 배치(map)되어 있다. 메일박스는 일반적으로 긴급사태에 대비하여 특정의 주변장치나 다른 마이크로프로세서에 어드레스된 데이터를 위해서 확보해놓은 공통 RAM 메모리에 있는 기억장소로서 중앙처리장치와 보조 마이크로프로세서가 최소한의 하드웨어를 써서 서로 다른 데이터를 질서 있게 전송하기 위하여 사용하는 기술이다.

3.2.2 PCI버스 인터페이스 레지스터 셋

PCI 버스 인터페이스(V292PBC 디바이스)의 레지스터 집합은 광역 메모리영역에 (시작주소: 0x80800000)에 배치되어 있고, 이 레지스터 집합은 제어유니트가 PCI 버스 인터페이스(버스 마스터 전송, 슬레이브 메모리 배치, 메일박스 레지스터 셋)를 제어하는 데 사용한다.

3.2.3 PCI 버스 마스터링

제어유니트는 PCI 버스를 직접 제어함으로써 감시유니트의 CPU 메모리 영역으로 데이터를 직접 읽거나 써서 데이터를 송수신할 수 있는 버스 마스터링을 활용한다. 따라서 제어유니트의 PCI 버스 인터페이스는 광역 SRAM 메모리를 PCI 메모리 영역에 배치(map)한다. 이 메모리는 감시유니트의 DRAM에 배치된다.

3.2.4 버스 마스터 전송 셋업 및 전송

버스 마스터 전송 셋업은 DMA 제어기 초기화와 유사하다. 먼저 PCI 버스 인터페이스는 버스 마스터 전송을 시작하기 전에 다음과 같은 변수들을 설정한다.

- PCI 메모리 주소
- 제어유니트의 광역 메모리의 오프셋
- 전송할 32bit 워드의 수

또한 전송을 시작하기 위해서 인터페이스에 DMA 제어기를 설정하기 때문에 버스 마스터 제어용 레지스터의 방향비트와 시작비트는 나중에 초기화된다. 버스 인터페이스 하드웨어가 전송을 무사히 끝내는 동안에도 제어유니트의 프로세서가 다른 타스크를 수행할 수 있도록 하기 위해서 버스 마스터 트랜잭션은 인터럽트 드리븐 방법으로 이루어지도록 한다.

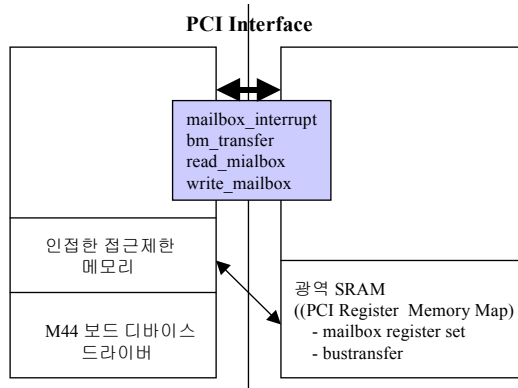
3.2.5 감시유니트의 DRAM 페이지 잠금

제어유니트는 데이터를 전송할 수 있는 감시유니트의 RAM 영역을 알아야 한다. 이 데이터 전송은 감시유니트의 운영체제의 업무와는 비동기적으로 일어나고, 버스 마스터 메카니즘에 사용되는 영역에 데이터를 옮기는 업무에 운영체제는 제한적이어야 하기 때문이다. 그리고 데이터 전송 버퍼를 제공하기 위한 감시유니트의 물리적인 DRAM은 디바이스 드라이버에 의해 자동적으로 잠금된다.

3.2.6 메일박스 접근

(그림 2)와 같은 PCI 버스 인터페이스 메일박스는 4개의 32bit 레지스터를 포함하는데, 이것은 감시유니트와 제어유니트간의 빠른 전달을 요구하지 않는 간단한 정보

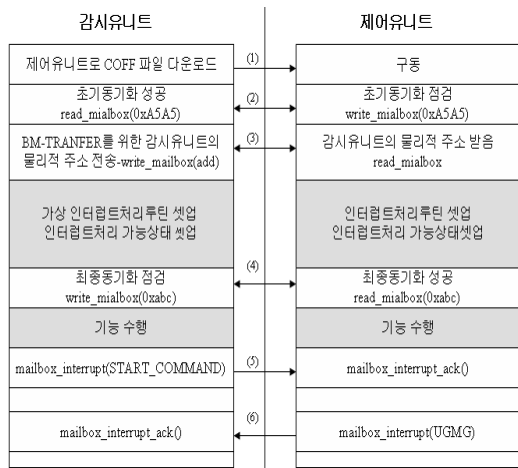
를 전송할 때 사용한다. 인터럽트를 활용하여 이 레지스터를 읽고 쓸 수 있는 기능을 가능하게 하여 사용한다. 메일박스는 핸드셰이크 기능이 없기에 언제라도 다른 프로세서가 어떤 메일박스를 접근할 수 있다.



(그림 2) 메일박스 접근 방법

3.3 연계기반의 구동 방법

제어유니트는 감시유니트의 독립된 보조 프로세서로서 활용되며, CSP가 DSP 메모리로 COFF 파일을 다운로드하여 구동시킬 뿐 만 아니라, 정지 시에도 CSP가 CCP를 정지한다. 구동순서는 (그림 3)과 같다.



(그림 3) 구동방법

- (1) 감시유니트의 프로그램이 구동되면 감시유니트는 제어유니트의 다운로더(downloader)를 호출하여 COFF 파일(***.OUT)을 다운로드한다.
- (2) 통신 동기화를 점검하기 위해서 CCP는 메일박스를 통해 0xA5A5를 CSP로 전송한다.
- (3) CSP는 A5A5를 받아 초기동기화가 성공하면 데이터 전송(bm_transfer)을 위한 감시유니트의 물리적 주소를 제어유니트에 메일박스를 통해 전송한다. 그리고 제어유니트 및 감시유니트는 각자 TASK 실행에 필요한 초기화 및 정의를 수행한다. 만약 서로간의 정의된 순서대로 데이터가 전송되지 않으면 제어유니트는 초기화된다.
- (4) 최종 통신 동기화를 점검하기 위해 감시유니트는 메일박스를 통해 0xabc를 제어유니트에 전송한다.
- (5) 제어유니트는 마지막 동기화 점검이 끝나면 제어유니트의 상태분석 및 데이터 획득 및 필터링 기능을 수행하기 시작하고, 감시유니트로부터 구동요구 및 제어 명령이 인터럽트를 통해 전달되면 정의된 인터럽트 처리 방법에 따라 인터럽트를 우선적으로 처리한다.

4. 임베디드 제어 소프트웨어 모듈

제어소프트웨어는 CEDM 구동에 대한 시간제어 기능, 전류분석을 통한 피드백 및 감시변수생성 기능과 같은 실시간 요건을 만족할 수 있도록 크로스 개발환경에서 ANSIC로 구현하였다. 제어프로세서의 소프트웨어는 크게 4가지 모듈로 분리할 수 있다. 제

어유니트 관리 모듈은 하위 태스크들을 관장하는 모듈로서, 각종 하드웨어를 초기화하고 인터럽트 환경을 설정하여 비주기적으로 발생하는 운전원 제어요구를 분석 및 제어하며, 비정상 상태에 대한 제어를 수행한다.

신호취득 모듈은 4개의 A/D 변환기를 구동하여 얻은 데이터를 디지털 필터링을 통해 잡음을 제거하여 프로세서의 공통 메모리에 저장하는 기능을 수행한다. 입력주기는 4 채널당 200us이며, 5개의 원시데이터를 읽어 디지털 필터링 수행을 통해 1ms 주기로 한 개의 대표값을 생성하여 공통 메모리에 저장함으로써 상태분석 및 구동제어 태스크를 수행하는데 사용된다.

상태분석 모듈은 CEDM 코일에 흐르는 전류상태를 분석하거나, 출력상태를 점검하는 기능을 수행하며 세부적으로는 그리퍼(gripper)의 저전류 점검, CEDM 운전 및 감시 프로세서 제어입력점검, 고전류 점검 등이 있다.

구동제어 모듈은 CEDM 운전 및 감시프로세서로부터 구동요구가 있을 때 요구 형태에 알맞은 구동 시퀀스를 수행하는 기능을 한다. 구동 시퀀스는 구동형태에 따라 결정되는데 낙하, 인출, 삼입으로 나뉜다. 각각의 구동 시퀀스의 최종목표는 적절한 시간간격으로 적절한 출력 패턴을 CEDM 전력제어유니트에 전달하기 위한 것으로 이를 위해 CEDM 전력제어유니트 내의 전류 센서신호를 피드백 받아 분석하여 결정한다. 낙하기능은 모든 코일의 전압을 차단하여 CEA를 자유낙하 시키기 위한 출력패턴을 CEDM 전력제어계통에 공급함으로써 이루어진다. 이 때 낙하지연시간을 코일전류 센서 신호로부터 계산하여 운전 및 감시프로세서로 전송한다. 인출기능과 삼입기능은 구동 시퀀스를 만족하도록 제어출력을 전송하는 기능을 수행한다.

4.1 제어유닛 관리 모듈

4.1.1 메인모듈

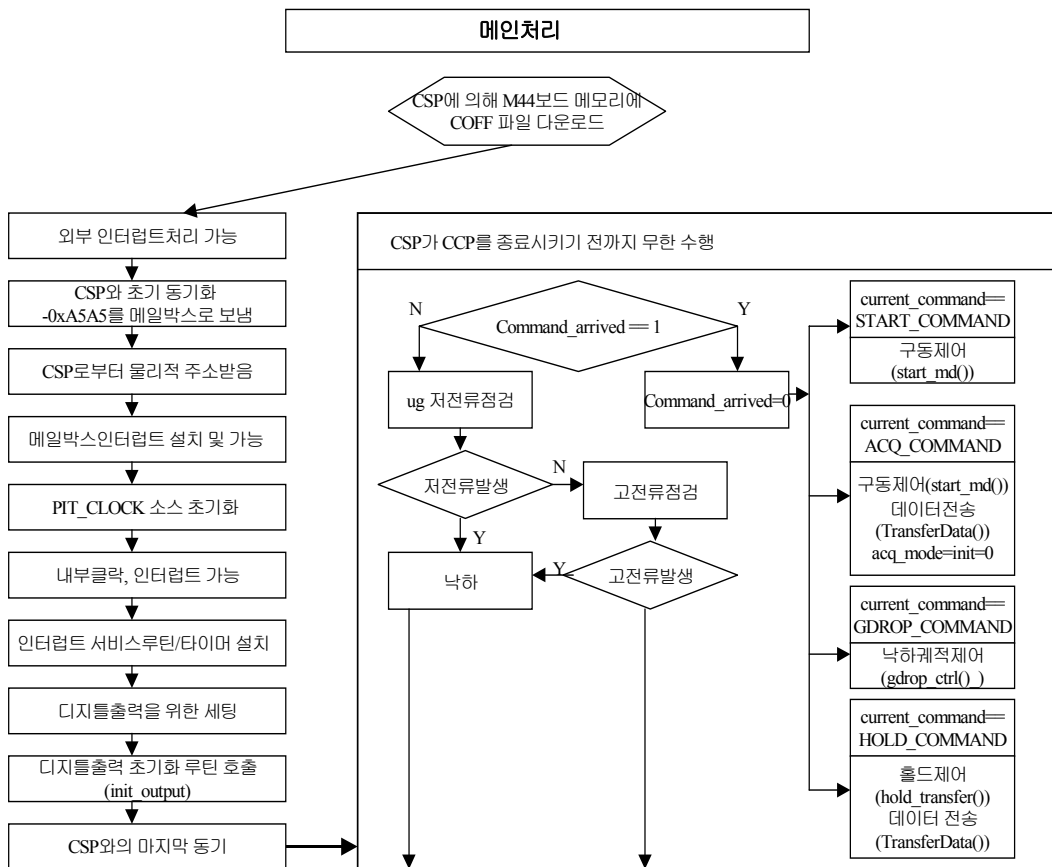
감시유닛이 절차에 따라 COFF 파일을 제어유닛의 특정 메모리에 다운로드하면서 제어유닛의 메인모듈이 실행된다. 메인모듈은 먼저 감시유닛과의 통신을 수행하기 위해서 메일박스를 사용하여 감시유닛에 “A5A5”를 써서 보낸다. 그리고 감시유닛이 “A5A5”를 메일박스를 통해 성공적으로 받으면 다량의 데이터 전송을 위한 메모리의 물리적인 주소를 메일박스를 통해 제어유닛에 전송하여 제어유니

트가 성공적으로 전송받음을 확인함으로써 초기동기화를 점검한다.

또한 타이머, 디지털 출력, 인터럽트 수행 환경을 초기화한 후 데이터 획득 모듈 및 필터링 모듈을 타이머 구동 인터럽트로 설정한다. 그 다음에는 감시유닛이 보낸 마지막 동기 신호를 받음으로써 감시유닛과 제어유닛간의 통신은 건전하다고 판단한다.

감시유닛과 마지막 동기를 시험하기 위해서 감시유닛로부터 데이터 “abc”를 전송받는다. 또한 각 모듈들에 대한 관리자 역할을 수행한다.

마지막 통신 점검이 성공적으로 수행하면



(그림 4) 메인 모듈

감시유니트로부터 전달되는 인터럽트를 감시한다. 인터럽트가 감지되지 않으면 그리퍼의 저전류 및 고전류 점검 등의 상태분석을 수행하고, 인터럽트가 감지되면 구동요구 인터럽트 처리모듈에서 분석된 인터럽트 분석 결과(current_command)에 따라 해당 제어모듈로 분기하여 처리하고 복귀한다. 이 메인모듈은 감시유니트에서 정지시키지 않는 한 계속 수행한다. 메인모듈은 (그림 4)와 같다.

4.1.2 구동요구 인터럽트 처리모듈

인터럽트 서비스 루틴을 메인모듈에서 mailbox_interrupt_install(command_isr), mailbox_interrupt_enable와 같이 설치함으로써, 감시유니트가 아래와 같이 32비트로 묶인 제어정보를 첨부시킨 메일박스 인터럽트를 보내면 제어유니트는 인터럽트 설치에 따라 해당 서비스 루틴인 구동요구 인터럽트 처리루틴(command_isr)을 실행한 후 메인모듈로 복귀한다. 구동요구 인터럽트 처리 모듈은 우선 32비트 제어정보를 분석하여 current_command에 따른 구동제어에 필요한 정보(데이터 전송을 위한 전송할 데이터 갯수(TRANSIZE), 전송 종류(bm_type)와 구동제어 변환을 지시하는 플래그(Stop_init, LEL, UEL, acq_mode, drop_mode, init)들에게 적절한 값을 할당한다.¹⁾

4.1.3 구동제어 모듈

구동요구 인터럽트 처리 모듈을 통해 분석된 구동정보들을 가지고 코일케속요청과 낙하/삽입/인출 제어 등을 수행하기 위한 프로그램 분기를 지시한다. 먼저, ctl_speed에 따른 cycle_time를 결정하고, current_command가 ACQ_COMMAND이면 제어종류(ctl_type),

시간제어(time_type)에 따라 해당 모듈을 호출한다. 그리고 current_command가 START_COAMMD이면, 제어종류(ctl_type), 시간제어(time_type)에 따라 해당 모듈을 호출한다.

4.2 데이터 획득 및 필터링

4.2.1 데이터획득 및 제적요청 처리모듈

4채널의 해당 코일별 아날로그 값을 200us로 A/D변환하여 pair structured buffer에 쌓는다. 그리고 운전원이 코일케속 버튼을 눌렀을 때 피드백/고정, 삽입/인출, 속도에 따라 한 스텝을 구동하면서 전송을 위해 4가지 코일전류 데이터를 큐에 쌓고 한 스텝 완성 후 전송한다.

아날로그 입력을 위해서는 각 채널마다 하드웨어 타이머 기반 트리거 소스를 세팅하고, 게인(gain) 및 멀티플렉스를 맞추어야 한다.

4.2.2 필터링 모듈

디지털 필터링은 채널별로 5개의 데이터(1msec)에 대해 최대, 최소값을 버리고, 나머지 3개의 평균값을 취하여 대표값을 만든다. 또한 채널별 5개 값을 읽어 아래와 같은 계산을 통하여 공학적 단위변환을 수행한다. 16bit 아날로그 입력의 디지털 변환을 통하여 얻은 카운트값(-32768~32768)을 공학적 단위변환 (1 count = 1/32768*10 = 0.0003051)를 통해 전압값으로 변환 한 후 전류값으로 (I=3V) 변환한다.

4.3 상태분석모듈

상태분석모듈은 CEDM 코일에 흐르는 전류상태를 점검하는 루틴으로서 세부적으로는 그리퍼의 저전류 점검, 고전류 점검 등이 있다.

1) LEL(Lower Electrical Limit) UEL(Upper Electrical Limit)

4.3.1 저전류 검사모듈

저전류 검사모듈은 제어봉의 낙하를 방지하기 위해서 홀드 코일의 저전류를 측정한다. 즉 UG(Upper Gripper) 코일과 LG (Lower Gripper)코일에 흐르는 전류상태를 점검하여 둘 중 하나가 설정치 이상이면 정상조건으로 판명하고, 둘 다 설정치 이하이면 경보를 운전 및 감시 유니트에 전달한다. 이 모듈은 리셋, 낙하제어를 제외한 모든 모드에서 수행한다.

4.3.2 고전류 검사모듈

고전류 검사 모듈은 코일의 Burn-out를 방지하기 위해서 수행하는 모듈로서 운전원의 구동요구가 없고, 그리고 유지 상태일 때 4개의 코일에 흐르는 전류데이터 100개(100 msec 동안의 데이터)에 대해 66개 이상의 데이터가 설정치를 넘을 경우 경보를 운전 및 감시유니트에 전달하는 기능을 수행한다. 이 모듈은 리셋, 구동제어 시에는 수행하지 않는다.

4.4 구동제어 모듈

제어에 맞는 시간 스킴을 설정한 후 상태 분석모듈의 분석결과를 바탕으로 전력제어 유니트에 출력신호를 보낸다.

4.4.1 피드백 삽입 구동제어

운전 및 감시프로세서로부터 구동요구가 있을 때 삽입속도를 입력받아 그에 따라 만들어진 출력패턴을 분석 및 제한 시간에 맞추어 생성하여 전력 제어유니트에 전달한다. 그리고 운전원이 정지버튼을 눌렀을 때에는 구동 중이라면 하던 일을 끝내고, 더 이상 진행하지 않고, 다른 외부적인 입력(아날로그 입력), 주기적인 상태점검 작업

은 계속 수행한다. 구동 중에 LEL 인터럽트가 도착하면 남은 구동 횟수만큼 인출구동제어를 수행한다.

4.4.2 피드백 인출 구동제어

운전 및 감시프로세서로부터 구동요구가 있을 때 인출 속도를 입력받아 그에 따라 만들어진 출력패턴을 분석 및 제한 시간에 맞추어 생성하여 전력제어유니트에 전달한다. 그리고 운전원이 stop 버튼을 눌렀을 때에는 구동 중이라면 하던 일을 마저 끝내고, 더 이상 진행하지 않고, 다른 외부적인 입력(아날로그 입력), 주기적인 상태점검 작업은 계속 수행한다. 구동 중에 UEL 인터럽트가 도착하면 남은 구동 횟수만큼 삽입구동제어를 수행한다.

4.4.3 고정삽입구동제어

운전 및 감시프로세서로부터 구동요구가 있을 때 삽입속도를 입력받아 그에 따라 만들어진 출력패턴을 분석 및 제한 시간에 맞추어 생성하여 전력제어유니트에 전달한다. 그리고 운전원이 stop 버튼을 눌렀을 때에는 구동 중이라면 하던 일을 끝내고, 더 이상 진행하지 않고, 다른 외부적인 입력(아날로그 입력), 주기적인 상태점검 작업은 계속 수행한다. 구동 중에 LEL 인터럽트가 도착하면 남은 구동 횟수만큼 인출구동제어를 수행한다.

4.4.4 고정인출구동제어

운전 및 감시프로세서로부터 구동요구가 있을 때 인출 속도를 입력받아 그에 따라 만들어진 출력패턴을 분석 및 제한 시간에 맞추어 생성하여 전력제어유니트에 전달한다. 그리고 운전원이 stop 버튼을 눌렀을 때

에는 구동 중이라면 하던 일을 끝내고, 더 이상 진행하지 않고, 다른 외부적인 입력(아날로그 입력), 주기적인 상태점검 작업은 계속 수행한다. 구동 중에 UEL 인터럽트가 도착하면 남은 구동 횟수만큼 삽입구동제어를 수행한다.

4.5 낙하제어모듈

운전원이 낙하계적 버튼을 눌렀을 때, 피드백 운전모드에서 낙하버튼을 눌렀을 때 그리고 운전원이 고정 운전모드에서 낙하버튼을 눌렀을 때 낙하 제어모듈이 실행된다. 낙하제어를 수행하면서 코일전류 데이터를 큐에 쌓아 제어수행 완료 후 전송한다. 그리고 각 낙하패턴 출력 후 리셋된다.

4.6 디지털 출력제어

제어유니트는 CEA를 구동하기 위해 디지털 출력을 수행함으로써 CEDM 코일에 인가되는 전압을 제어한다. 각 코일에 인가되는 전압은 고전압/저전압/오프의 3가지 상태로서 이를 위해 8bit 디지털 출력으로 구성한다.

4.7 리셋제어모듈

리셋제어모듈은 운전원이 요구 또는 CEDM 코일 전류 분석결과 비정상적인 상황이 발생할 경우 수행된다. 리셋되는 조건은 아래와 같다.

- 그리퍼의 부적절한 전류
- 출력 패턴 이상
- 비정상적인 각 코일 전류 (각 코일에 흐르는 전류가 설정치보다 높음)
- 논리적 제어와 실제 기계적 구동 불일치(기계적 운동 없을 때)
- 낙하 종료 후

5. 시험 고찰

CEDMCSTF의 시험요건을 기반으로 유니트시험과 종합시험을 수행하였다. 유니트 시험은 구현된 각각의 유니트 하드웨어와 소프트웨어가 설계된 하드웨어 및 소프트웨어 사양을 만족하고 있는지 확인하기 위해 수행되었으며, 종합시험은 기능요건 및 설계요건대로 CEDMCSTF가 구현되었는지 확인하고자 수행되었다. 다만 현재 시점에서 CEDMCSTF의 제어 대상물인 CEDM이 설치되지 않아 CEDM 코일의 특성을 나타내는 모의 부하를 제작하여 종합시험을 수행하였으므로 전류분석관련 기능 중 일부 시험절차는 수행하지 못하였다.

제어유니트의 시험결과에 대한 고찰은 다음과 같다.

1) 신호 채터링(chattering)

감시프로그램(CSP)로부터 제어 프로그램(CCP)에 전달되는 인터럽트 신호들 중 제어봉 상한 및 하한 제한치점 도착 시 발생하는 신호의 채터링에 대한 고려를 하지 못했다. 그래서 RSPT (Reed Switch Position Transducer) 연결 불안으로 인한 UEL 및 LEL을 검출하는 센서신호 불안으로 발생하는 검출 신호의 채터링은 결국 감시프로그램에서 제어프로그램으로 계속적인 제한치점 도착 인터럽트를 전송하였다. 결국 제어프로그램에서는 첫 번째 인터럽트 시 방향전환을 수행하여 남은 스텝을 전환된 방향으로 제어를 수행한 후 두 번째, 세 번째로 전송된 인터럽트 신호에 의해 설정된 제한치 플래그에 의해 제일 처음 제어방향에 다음 번 운전원 명령어 시작 시 제어 시작점에서 다시 방향전환 결과를 낳았다.

그래서 CEDM의 실험 중 발생한 CEDMCSTF

외부 환경의 변화에 대한 다음과 같은 고장 허용논리를 삽입하였다.

- LEL 및 UEL 인터럽트가 발생하면 세팅되는 플래그 변수들에 대해 각 발생할 수 있는 상황을 분석하여 같이 발생할 수 없는 인터럽트에 대해 설정되는 인터럽트 플래그를 동시에 클리어한다. 클리어 시점은 플래그 변화를 인지한 다음으로 결정함
 - 동시에 감시프로그램(CSP)에서 제어 프로그램 쪽으로 인터럽트발생 신호들에 대한 검증을 수행하여 여러 조건들을 판단하고 결정하여 전송하기로 함
- 2) 효율적인 메모리 관리 및 사용에 대한 고려 제어 프로그램을 수행하는 DSP 보드의 메모리는 1개의 지역메모리(128k)와 1개의 광역메모리(512k)로 구성되어 있다. 제어프로그램은 비주기적인 인터럽트와 주기적인 인터럽트를 빈번히 수행하기에 문맥교환(Context Switching)이 자주 일어나며 소모되는 메모리 관리 또한 중요하다. 48시간 정도 끊임 없이 실행되는 제어 프로그램은 호출되는 여러 서브루틴들의 실행 및 복귀 그리고 자주 일어나는 문맥교환은 메모리 관리의 한계를 보여 제어프로그램의 무응답 상태를 보여 결과를 보였다. 그래서 서브루틴의 호출과 복귀 지점을 최소화하고, 어떤 제어행위 수행 후 메모리를 초기화함으로써 프로그램 진행의 메모리 관리를 효율적으로 수행하였더니 제어프로그램이 시험요건 시간동안 무리 없이 실행됨을 보였다.

6. 결론

CEDMCSTF는 CEDM 성능시험에 필요한 모든 변수에 대해 디지털 신호분석 및 처리를 통해 CRT로 운전원에게 제공하고, 운전원의 제어행위도 소프트웨어방식을 통해 CRT 화면상에서 이루어지도록 하여 기존 CEDMCS의 문제점을 디지털 기술 적용으로 해결하는데 목적이 있다. 본 논문에서는 CEDM 운전 및 감시 유닛의 제어부 삽입, 인출 및 낙하 명령과 CEDM 전력제어유닛에서 검출된 각 CEDM코일의 전류 상태를 분석하여, 제어논리신호를 CEDM 전력제어유닛에 출력하는 기능을 수행하는 제어 유닛의 임베디드 제어 프로그램의 실체를 소개하였다. 또한 시험을 통해 발견된 논리적인 결함을 수정하여 프로그램 수행에 있어 최적화를 달성한 결과를 소개하였다. 향후 추가적인 연구과제는 디지털 신호분석에 대한 연구를 보완하여 보다 손쉽게 CEDM의 기계적 거동상태를 파악할 수 있도록 하고, 개선된 전력제어방식에 대해서 운전성과 신뢰성을 입증함으로써 향후 실제 발전소에의 적용성 등을 검토할 예정이다.

참 고 문 헌

- [1] R.E.Weber, "Engineering Specification for a Control Element Drive Mechanism Control System for System 80 Standard Design", STS80-ICE-6022, Rev. 04, Combustion Engineering, Inc., 1982.
- [2] R, J, Moreau, Operational Description of the Software and Hardware for UCN 3&4 ACTM, 91691-IC-SX630-2, Rev.00, ABB-CE

- Combustion Engineering, Inc., May 1996
- [3] B. M. Kim, Zero-Cross Detecting Technology in Control Element Drive Mechanism Control System of PWR, KAERI Technical Report, KAERI/TR-689/95(Korean), 1995
- [4] 허섭 외, “제어봉구동장치 성능시험을 위한 제어봉구동장치제어계통 개발”, 한국원자력학회, 춘계학술발표회, 1999



박 희 운
 1984년 경북대학교 전자공학과 학사
 1990년 경북대학교 전자공학과 석사
 2002년 연세대학교 전자공학과

박사

1986년~현재 한국원자력연구소 책임연구원, 과제 책임자

관심분야 : 안전통신망 성능분석 및 망관리



장 귀 속
 1990년 영남대학교 전산공학과 학사
 1992년 경북대학교 컴퓨터공학과 석사
 1992년~현재 한국원자력연구소

선임연구원

관심분야 : 능동 데이터베이스, 실시간 데이터베이스, 원자력 감시시스템 설계



김 영 국
 1985년 서울대학교 계산통계학과 학사
 1987년 서울대학교 계산통계학과 석사
 1995년 버지니아대 컴퓨터공학과

박사

1995년~1996년 핀란드 VTT, 노르웨이 SINTEF DELAB 방문연구원

1996년~현재 충남대학교 전기정보 통신공학부 부교수

2002년 8월~2003년 7월 UC Davis 방문교수

관심분야 : 실시간데이터베이스, 모바일정보시스템, 전자상거래시스템



김 종 용
 1991년 충남대학교 전산학과 학사
 1993년 충남대학교 전산학과 석사
 1993년~현재 한국원자력연구소

선임연구원

관심분야 : 계측제어시스템설계, 정보시스템