

## 영광1호기 시뮬레이터 노심계통 연계기술 개발

# Development of the Core-Model Implementation Technology for YGN1 Simulator

홍진혁, 이명수, 이용관, 서인용  
한전전력연구원  
대전광역시 유성구 문지동 103-16

### 요 약

그동안 국내에서 사용된 가압경수로 원전 시뮬레이터 노심모델은 전부 국외 전문회사로부터 도입되었으며, 노심모델 개발당시의 전산기 처리능력의 한계로 인하여 실시간성을 위하여 여러 가지 가정들이나 간략화가 필수적이였다. 이에 기존 시뮬레이터용 노심모델 대신 규제기관(KINS)으로부터 인증받은 핵설계 코드(MASTER)를 기반으로하는 3차원 노심 실시간 시뮬레이션 모델을 개발하고 적용하고자 하는 “원전 시뮬레이터 노심-계통연계 기술개발” 과제를 착수하였으며, 영광1호기 전범위 시뮬레이터를 대상으로 노심해석 분산모델을 개발하고 있는 중에 있다. 현재의 영광1호기 시뮬레이터 서버컴퓨터에서는 컴퓨터 성능의 한계상 MASTER코드를 이용한 노심모델을 구동하는 것은 불가능하므로, 서버와는 별개의 컴퓨터에서 구동시키는 것이 필수적으로 요구되었으며, 본 논문에서는 이러한 분산 노심모델과 시뮬레이터 서버와의 실시간성 보장을 위하여 구현된 통신 규약 및 방법론에 대해 다루고자 한다.

### Abstract

The existing core models for the domestic nuclear power plant simulators for PWRs are entirely imported from the foreign simulator vendor. To solve the time-accuracy problem in the poor capabilities in the computer in the early 1990s, several simplifications and assumptions for the neutronics governing equations were indispensable for the realtime calculations of nuclear phenomena in the core region. To overcome the shortages, a new core model based on the MASTER code certified by the domestic regulatory body (KINS) instead of the existing core models is now being developed especially for the realtime core solver for the YGN-1 simulator. This code is named R-MASTER (Realtime MASTER code). Due to the deficiency of the host computer, it is quitely required to run the R-MASTER code on the separate computer with high performance from the host computer on which all the other models than the core model are running. This paper deals with the applied protocols and procedures to guarantee the realtime communication and calculation of the R-MASTER code.

## I. 서론

TMI (Three Mile Island) 및 체르노빌 원전사고 이후 원자력발전소의 안전성 확보를 위해 운전원의 운전능력 배양을 위한 효과적인 훈련이 강조되어 왔으며 이를 위해 가동중인 원자력 발전소에서는 시뮬레이터를 사용하여 운전원의 훈련에 사용하여 오고 있다.

1992년 시뮬레이터 국산화 계획에 따라 시뮬레이터가 국내 개발되기 이전에는 각기 다른 외국의 전문 업체에서 Turn-Key 베이스로 4기의 운전원 훈련용 시뮬레이터를 도입 운영하고 있었으며 시뮬레이터 구입 후 발전소 노심 주기가 바뀌어도 시뮬레이터의 노심모델은 그대로 변동없이 사용해 왔다. 이러한 현상은 시뮬레이터 국산화 계획에 따라 국내에서 시뮬레이터를 제작한 후에도 계속되었는데 이러한 주 이유는 시뮬레이터에서 사용되는 노심모델의 생성을 위해서는 노심설계코드체계를 이용한 각종 입력자료와 연계자료의 생산이 필수적이며 또한, 시뮬레이터 특성상 실시간에 계산을 하기위해 여러 가지 가정이나 간략화(Simplification)를 통해 야기되는 노심모델의 부정확성으로 노심설계/해석코드 대비 계산결과 오차가 커지므로 이를 보정하기 위한 보정인자의 도입 또는 보정과정 (tuning)이 필요하게 되어 복잡해지기 때문이다.

최근 전산기 중앙처리장치의 발달은 3차원 시공간 중성자 확산방정식의 해를 직접 구하여 노심의 출력 변화와 출력 분포를 직접 계산하는 노심설계와 유사한 수준의 상세계산에서도 실시간에 근접한 계산 속도를 가능하게 하고 있다. 따라서 원전 시뮬레이터의 노심모델에 설계코드와 같은 수준의 실시간 모델이 개발된다면 노심모델의 생산에 필요한 각종 변수의 조정 작업이 생략되거나 최소화됨으로써 주기별 시뮬레이터용 노심모델의 제공이 용이해질 뿐 아니라 원자로 과도현상에 대한 모사계산의 정확도에서도 획기적인 향상을 이룰 수 있다.

본 논문에서는 이러한 목적으로 안전기술원으로 인증받은 핵설계코드인 MASTER를 이용하여 개발중인 실시간 노심모델의 분산 제어 방법론 및 통신 구현 방법에 대해 다루고 있다.

## II. 본론

### 1. 영광1호기 시뮬레이터 환경

영광 1호기 시뮬레이터 환경은 시뮬레이터 소프트웨어 전문 제공업체인 GSE사의 시뮬레이터 환경인 SimSuite Power 3.3 (SSP3.3)을 바탕으로 하고 있다. SSP3.3은 시뮬레이터를 구성하는 모든 소프트웨어 (계통 모델링, P&ID 등)를 개발하고, 실시간 실행하고 디버깅할 수 있는 종합관리 시스템으로, 소프트웨어 구성관리 시스템 (Software Configuration Management System), 논리적 데이터베이스 개념 채택, 다중 소프트웨어 계층 등 계통 모델 개발의 용이성을 도모하였다. 시뮬레이터 서버에서는 SSP 고유의 데이터베이스를 생성, 운영하고 있다. 시뮬레이션 모듈이 수행되면 모든 데이터, 실수, 상수들은 전역공유 메모리 영역(Global Common Memory Partition)에 위치

한다. 다중 프로세서(Multiprocessor)로 구성된 시뮬레이터 컴퓨터의 경우 공유메모리 영역은 모든 프로세서가 공유하는 메모리를 뜻한다. 이 공유메모리 영역은 표준 FORTRAN/C 컴파일러의 전역 공유 메카니즘(Global Common Mechanism) 방법에 의해 생성된다. 데이터 베이스의 내용 및 구조는 DBM (DataBase Manager) 프로그램의 통제하에서 생성 수정되는 MDD(Master Data Dictionary)에 의해 정의된다. 모든 시뮬레이션 데이터(상수,변수)는 MDD에 정의되며, 각 데이터는 다음 사항을 포함한다.

- 31자의 이름
- 72자 이내의 요약
- 세부설명(5줄\*72자/줄)
- Type과 Precision (integer, real, complex 및 logic)
- 배열 및 행렬에 대한 차원 정보(최대 7차원)
- 데이터의 위치 및 영역(Location and Partition)
- 단위
- FORTRAN 출력 형태
- 초기값
- 수정 일자 및 기타 정보
- Cross-reference에 관한 정보

이같은 데이터베이스 구조를 사용함으로써 얻을 수 있는 잇점은

- 모든 심볼을 중앙 데이터베이스에서 관리함으로써 일관성을 유지할 수 있다.
- 모든 변수를 연속된 영역에 보관함으로써 초기 상황 설정(Initial Condition Setup), Snapshot, Backtrack 등을 효율적으로 수행할 수 있다.
- 모든 상수를 데이터베이스에 저장토록 함으로써 데이터의 건전성을 유지할 수 있다.
- MDD에 저장되는 데이터에 대해 단위와 출력 형태를 정의하도록 하여 출력물의 효용을 높일 수 있다.
- 모든 상수를 데이터베이스에 의미있는 부호로 정의하여 Tuning 편의를 도모할 수 있다.

또한 DBM은 모든 시뮬레이션 모듈들에 관한 정보를 보유하고 있는 MSF(Module Specification File)을 생성 유지하는 역할도 수행한다.

MSF에 수록되는 정보는 다음과 같다.

- 시뮬레이션 이름(31자 이내)
- Full Name (10자 이내)
- US3 Type
- Return Type(함수에 한함)

- 인수 형태
- Cross-reference
- 내용(32자 이내)
- 수정 일자 및 기타 정보
- 시스템 인식자
- 다른 모듈과의 일관성(컴파일 여부)

영광 1호기 시뮬레이터의 경우 59개의 전역메모리 공간이 존재하는데, 각 공유메모리마다 연관된 계통이 존재하며, 각기 고유의 공유메모리명이 할당되어 중복되지 않도록 되어있다. 각 전역메모리의 이름 및 의미는 [표.1]과 같다. 이중 노심과 관계한 공유메모리는 global\_crk, global\_crs, global\_crxsc이며, 노심과 연계한 계통과 관계한 공유메모리는 global\_th, global\_disw, global\_dosw, global\_v이다.

[표.1] 영광1호기 시뮬레이터 전역메모리명 및 의미 (계속)

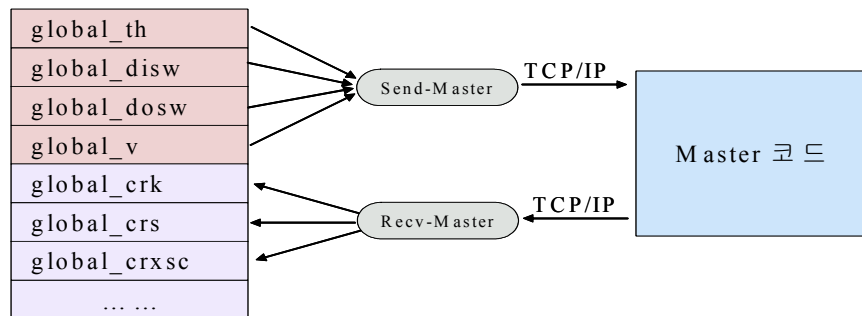
전역 메모리명	의미	크기 (Bytes)
global96	mst communication with mstxfer	2,048
global97	yp instructor station - iomod interface	2,048
global98	Reserved for ISAS	20,480
global99	executive interface global	1,024
global_k	Global for Constants	102,400
global_mstic	ic snap task global	206,848
global_rrio	record/replay task global	3,072
global_v	global for variables	307,200
global_is1	global base	245,760
global_is2	global_is2	16,384
global_aisw	software analog input points	1,024
global_aihw	hardware analog input points	1,024
global_ailp	hardware analog input points	1,024
global_aifl	override flags for analog input points	1,024
global_aiov	override values for analog input points	1,024
global_aosw	analog output points	5,120
global_aohw	analog output points	5,120
global_aolp	analog output points	5,120
global_aofl	analog output points	5,120
global_aoov	analog output points	5,120
global_disw	digital input points	5,120
global_dihw	digital input points	5,120
global_dilp	digital input points	5,120
global_difl	digital input points	5,120

전역 메모리명	의미	크기 (Bytes)
global_diov	digital input points	5,120
global_dosw	digital output points	10,240
global_dohw	digital output points	10,240
global_dolp	digital output points	10,240
global_dofl	digital output points	10,240
global_doov	digital output points	10,240
global_gisw	DCD input points	1,024
global_gihw	digital input points	1,024
global_gilp	DCD input points	1,024
global_gifl	group input points	1,024
global_giov	group input points	1,024
global_gosw	DCD input points	1,024
global_gohw	DCD input points	1,024
global_golp	DCD input points	1,024
global_gofl	group input points	1,024
global_goov	group input points	1,024
global_dihh	digital inputs packed	5,120
global_dohh	digital outputs packed	10,240
global_th	global for YWP(P2500) variables	542,720
global_crs	global for remark	245,760
global_crk	global for remark	12,288
global_crxsc	global for remark	122,880
global_clc	Global for CLASC Var.	30,720
global_tmp	Global for Temporaries	51,200
global_top	TopMeret global	409,600
global_fnet	root point of partition global_fnet	4,096
global_elg	Global for Elegant Variables	614,400
global_gf	internal variable of G-Flow	24,576
global_an	annunciator system variables	10,240
global_yan	AMS dynamics constants	12,288
global_ydh	global for YDH(DEHC) variables	1,024
global_yoa	global for YOA(OACS) variables	3,072
global_yrm	global for YRM(DRMS) variables	1,024
global_ywp	global for YWP(P2500) variables	3,072
global_ao	dummy analog output points for SEIS	1,024

## 2. 영광 1호기 노심모델과 RMASTER 코드와의 대체방법

영광 1호기의 경우, 노심모델은 시뮬레이터 환경 제공사인 GSE의 노심모델인 REMARK (REaltime Mutligroup Advanced Reactor Kinetics)을 사용하였다. REMARK 코드가 담당하는 노심모델을 RMASTER코드로 대체하기 위해서는 기존 REMARK 코드가 계산하는 중요 내부변수 뿐만 아니라, 타계통과의 연계되는 모든 변수를 RMASTER 코드에서 제공해 주어야 한다. 또한, Reset, Snap, Backtrack, Replay 등 기존 시뮬레이터에서 요구하는 각종 제어 기능을 구현하기 위한 제어기능을 RMASTER 코드에서도 역시 구현을 해주어야 한다. 또, 기존 REMARK 코드는 서버 컴퓨터에서 타계통과 함께 실행되는 구조로 되어 있지만, RMASTER 코드의 경우, 기존 서버의 물리적 성능으로는 실시간 실행이 불가능한 점 등의 이유로 분산으로 실행해야만 하므로, TCP/IP를 통하여 RMASTER 코드가 필요로 하는 타계통 변수의 현재값과 타계통이 필요로 하는 노심 변수값을 서로 통신으로 교환해야 하는 등 많은 작업을 요한다. RMASTER코드와 타계통간의 교환해야 할 변수로는 [표.2]와 같다.

또한, TCP/IP를 통하여 통신한 각 변수의 값을 전역 공유메모리에 할당하기 위해서 임시메모리 공간인 Recv\_Master/Send\_Master를 생성하여 TCP/IP를 통하여 매번 RMASTER 코드의 계산결과값이 Recv\_Master 메모리 공간을 통하여 서버의 공유메모리로 이동하며, Send\_Master를 통하여 RMASTER 코드의 입력값으로 제공되는 구조로 되어있다. 시뮬레이터 전역 공유메모리와 RMASTER를 위한 임시 메모리와의 관계는 [그림.1]와 같다.



[그림.1] 시뮬레이터 전역 메모리 공간과 MASTER 코드 결과 공간과의 상관관계

## 3. RMASTER 코드의 연계구조도

RMASTER 코드를 실시간 전범위복제형 시뮬레이터의 노심모델로 적용하기 위해서, 적용 시뮬레이터로는 우선 영광1호기 전범위 시뮬레이터를 대상으로 선정하였다. 현재의 영광1호기 시뮬레이터 서버컴퓨터 (Xeon, 4CPU, 500MHz)에서 RMASTER 코드를 탑재하여 실시간으로 구동시킨

다는 것은 현실적으로 불가능하며, 따라서 노심을 제외한 모든 계통은 현재의 서버에서 구동하는 것으로 하고, RMASTER 코드는 서버와는 별개의 최신 사양의 컴퓨터 (3GHz, Single-CPU)에서 구동되어, 서버와의 데이터를 교환하는 분산방식으로 구동되는 실시간 노심모델을 개발하였다. 이때, 물리적으로 떨어진 RMASTER 모듈이 시뮬레이터 서버에서의 Time-Scheduling을 그대로 동기화하는 논리가 가장 중요하며, 또한 시뮬레이터 서버에서 구동되는 타계통 변수값들과 RMASTER 코드의 변수값들이 서로 실시간으로 통신하는 작업이 또한 핵심적인 작업이다.

노심모델은 Reactor Vessel Thermal-Hydraulic(TH) System, Neutron Instrumentation(NI)/Neutron Monitoring(NM) System, Control Rod Drive(RD) System 등 3개의 계통과 경계한다. 경계하는 각 계통과의 취득 혹은 전달해야 하는 데이터로는 <표.2>과 같다.

[표.2] 타계통과의 연계변수

	INPUT	OUTPUT
TH	- Liquid Density - Vapor/Gas Density - Void Fraction - Liquid Temp. - Fuel Temp. - Boron Conc.	- Normalized Power - total thermal power - power fraction in each heat slab
NI/NM	- In-Core 및 Ex-Core Detector의 측정기 위치정보	- 각 측정기 위치에서의 중성자속 정보
RD	- 제어봉의 삽입길기와 패턴	- (없음)

시뮬레이션 계통 Solver로서의 역할을 담당하는 실시간 분산 시스템을 구현하기 위해서는 다른 계통 모듈과의 실시간 통신 및 시뮬레이터 서버 컴퓨터의 시간 동기화 (Synchronization) 작업이 반드시 이루어져야 한다. 시뮬레이션 서버와 RMASTER와의 소프트웨어 구성도는 [그림.2]과 같다.

서버와 분산모델과의 통신은 비차단 (Nonblocking) 송수신으로 된 연결지향 (Connection-Oriented) 스트림 서비스를 기반으로 한 서버/클라이언트 구조로 되어있다. 영광1호기 시뮬레이터의 경우 약 20개의 계통별 모의프로그램이 각기 실행파일 형태로 존재하며, 시뮬레이터 기동시 이들 모의 프로그램들이 각기 기동됨과 동시에 각 모델에서 필요한 변수값들에 대한 초기화가 이루어진다. 또한, 클라이언트 컴퓨터에서 구동되는 RMASTER 노심분산모델과 서버와의 실시간 통신하기 위한 분산모델 서버 프로그램이 기동되어, 분산된 모델명, 암호 및 각 분산모델에 할당된 공유메모리 크기 등 필요한 정보들을 가지고 클라이언트의 연결요청을 대기한다. 클라이언트인 분산컴퓨터에서는 각 분산모델이 기동되어 서버에 분산모델명, 암호 등을 송신함으로 서버와 연결하게 되어 서버의 제어명령(Run/Freeze/Reset/Snap 등)에 대해 대기상태에 놓이게 된다.

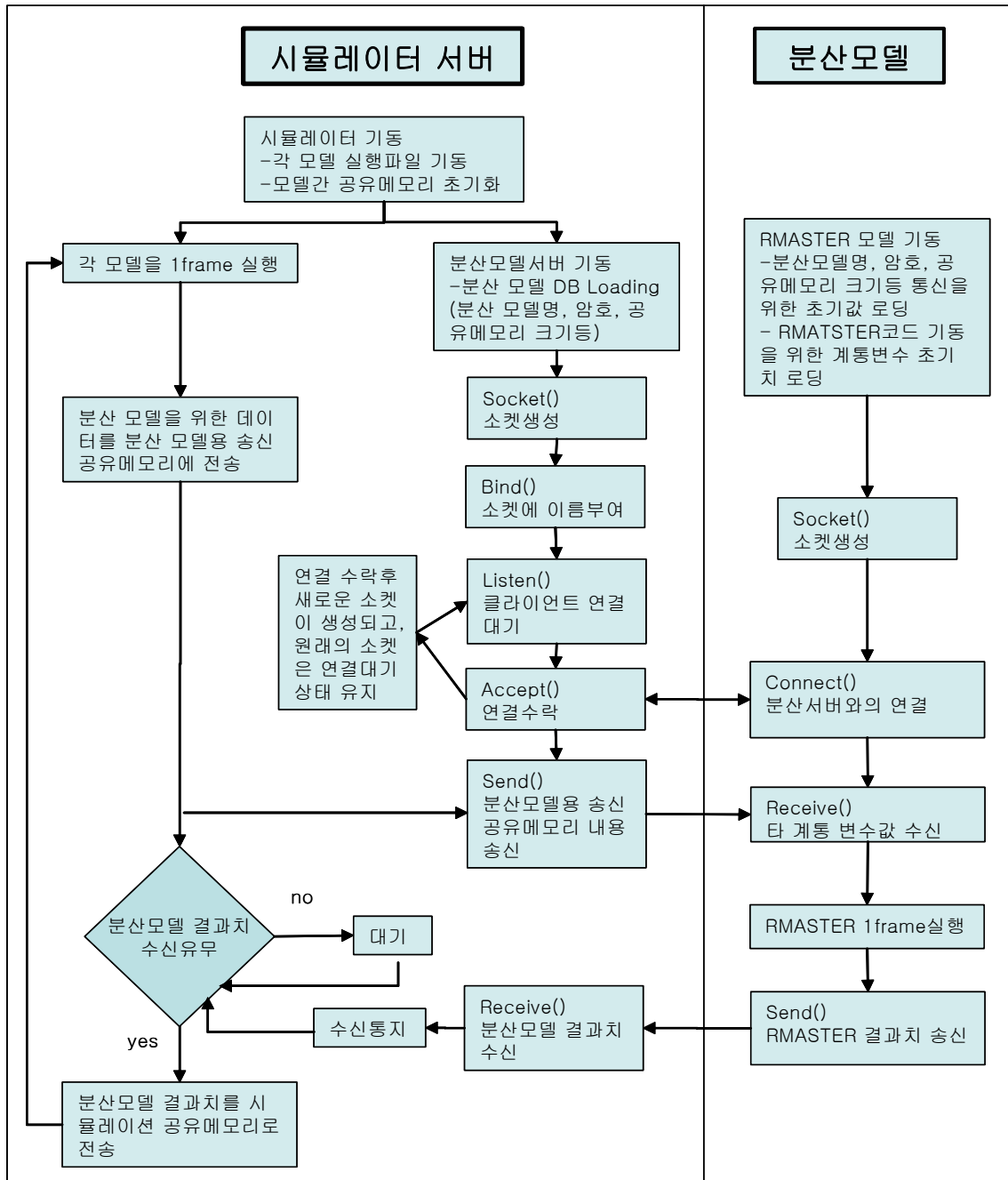
시뮬레이션 서버에서는 약 20개의 계통 모듈을 병렬로 1초에 12번 호출 (12Hz)함으로써 실시간 성능을 유지하게 되는데, 3회 호출당 매처음 호출시 서버측에서는 분산 RMASTER 코드가 구동하

기 위해 필요한 경계 변수값을 송신한 후, RMASTER 코드의 결과값을 기다리기 위한 대기상태 (Waiting Condition)로 있게 된다. 한편, 분산 컴퓨터에서는 수신한 경계 변수값들을 RMASTER 코드의 입력값으로 인식하여, 해당 time-frame에서의 3차원 중성자속 및 출력 정보들을 계산하게 된다. 그리고, 서버에서 구동되는 타계통이 필요로 하는 노심 변수값들을 다시 서버로 송신하게 된다. RMASTER 코드 결과치가 서버에 도달하게 되면, 서버에서는 대기상태에서 벗어나, 다음 time-frame에서의 경계치값들을 RMASTER 코드로 보낼 준비를 하게되는데, 이때 서버에서 분산 컴퓨터로 데이터가 송수신되고 분산 컴퓨터에서 RMASTER 코드가 구동되는 총 시간이 0.25초 (4Hz)이내에 이루어지지 않는다면, 서버측에서는 RMASTER 코드가 실시간 구동에 실패(Fail)한 것으로 간주하게 된다. 타계통 모델은 계산방식 및 로직의 간결성 등의 이유로 12Hz로 구동되지만, RMASTER 코드는 설계코드 수준의 3차원 시공간 중성자 확산방정식의 해를 구하는 고 정밀도와 방대한 연산 로직으로 1초에 4번 구동하는 것이 현재의 컴퓨터 기술로서 실시간성을 유지하기 위한 가장 빠른 주기로 판단했기 때문이다.

또한, 서버/RMASTER간의 통신시에, 서버에서 제공되는 송신 데이터는 패킷이라는 송수신 단위로 TCP/IP를 통하여 나뉘어 보내지게 되는데, 분산모델이 구동되는 클라이언트에서는 수취한 각 Packet들을 취합하여, 입력 데이터로 사용하는 구조로 이루어진다. 통신되는 Packet단위는 다음과 같은 8개의 부분으로 구성되어 있다.

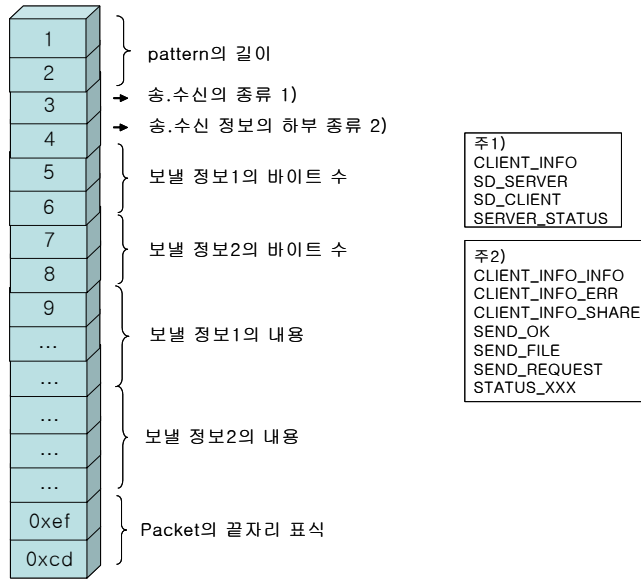
- 1st ~ 2nd 바이트 : 보내고자 하는 Packet의 총 바이트
- 3rd 바이트 : 송신 패킷의 목적을 나타내는 정보 1
- 4th 바이트 : 송신 패킷의 목적을 나타내는 정보 2
- 5th ~ 6th 바이트 : 1번째 보내고자 하는 정보에 대한 바이트수 ( $\equiv N1$ )
- 7th ~ 8th 바이트 : 2번째 보내고자 하는 정보에 대한 바이트수 ( $\equiv N2$ )
- 9th ~ (8+N1)th 바이트 : 1번째 보내고자 하는 정보
- (9+N1)th ~ (8+N1+N2)th 바이트 : 2번째 보내고자 하는 정보
- (9+N1+N2)th ~ (10+N1+N2)th 바이트 : 패킷의 끝을 나타내는 표식 (=0xcdef)



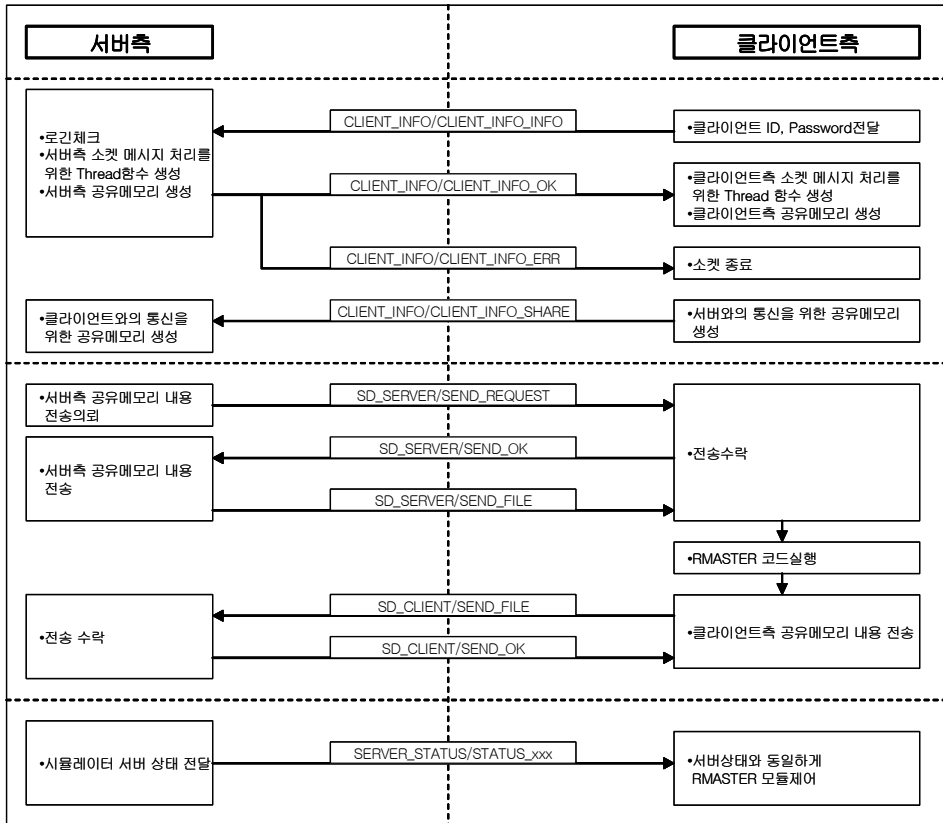


[그림.2] 시뮬레이터 서버 프로그램과 분산모델 (RMASTER)와의 소프트웨어 구성도

[그림.3]은 정의된 패킷의 내용을 요약한 그림이며, [표.3]는 적용된 패킷의 예를 나타낸 것이다. 또한 [그림.4]는 서버와 분산모델 (RMASTER)와의 통신 구조에 대한 요약을 나타내고 있다.



[그림.3] 패킷구조



[그림.4] 서버와 분산모델 (RMASTER)과의 통신 구조

[표.3] 패킷의 적용 사례

Bytes	SD_SERVER/ SEND_REQUEST	SD_SERVER/ SEND_OK	SD_SERVER/ SEND_FILE	SD_CLIENT/ SEND_OK	SD_CLIENT/ SEND_FILE	
2	8+2	8+2	8+2+1027	8+2	8+2+1027	
1	SD_SERVER	SD_SERVER	SD_SERVER	SD_CLIENT	SD_CLIENT	
1	SEND_REQUEST	SEND_OK	SEND_FILE	SEND_OK	SEND_FILE	
2(wFrst)	0	0	0	0	0	
2(wScnd)	0	0	1027	0	1027	
wFrst	-	-	-	-	-	
wScnd	-	-	1	SEND_CENTER /SEND_LAST	1	SEND_CENTER /SEND_LAST
			2	Sending Bytes (MAX:1024)	2	Sending Bytes (MAX:1024)
			1024	Contents	1024	Contents
1	0xef	0xef	0xef	0xef	0xef	
1	0xcd	0xcd	0xcd	0xcd	0xcd	

Bytes	CLIENT_INFO/ CLIENT_INFO_INFO	CLIENT_INFO/ CLIENT_INFO_SHARE	CLIENT_INFO/ CLIENT_INFO_OK	CLIENT_INFO/ CLIENT_INFO_ERR	SERVER_STATUS/ STATUS_XXX <sup>※1</sup>	
2	8+2+24	8+2+4+wFrst+wScnd	8+2	8+2	8+2+4+6	
1	CLIENT_INFO	CLIENT_INFO	CLIENT_INFO	CLIENT_INFO	SERVER_STATUS	
1	CLIENT_INFO_INFO	CLIENT_INFO_SHARE	CLIENT_INFO_OK	CLIENT_INFO_ERR	STATUS_XXX	
2(wFrst)	0	4	0	0	4 (IC number)	
2(wScnd)	24	Length of Shared Memory String	0	0	6 (Failed System Name)	
wFrst	-	Size of Shared Memory	-	-	1	'B'/'I'
					1	0
					1	0
					1	1
wScnd	12	ID String	String of Shared Memory	-	-	6 Characters of System Names
	12	Password				
1	0xef	0xef	0xef	0xef	0xef	
1	0xcd	0xcd	0xcd	0xcd	0xcd	

Symbols	Meaning
STATUS_FREEZE	Normal Freeze Status
STATUS_RUN	Normal Run Status
STATUS_STEP	Timed Run
STATUS_HOLD	No Executive Execution, No IO, Re-Init Counters
STATUS_TERM	Terminate - Everyone Exit
STATUS_FAIL	Failure has occurred, similar to hold
STATUS_SLOF	slow time intermediate frame
STATUS_INIT	freeze equivalent, initialization just completed
STATUS_REMAP	tell mst to attempt io re-initialization
STATUS_CONF	MST Configuration Active
STATUS_SUSP	Suspend Realtime Momentarily, used on exec fork
STATUS_RESU	Resume from Suspend
STATUS_SWCK	Switch Check Active
STATUS_RRIC	Switch Check Complete - Re-read IC
STATUS_IPCR	Pause for Process Computer Reset
STATUS_TBD	start Dort signal
STATUS_DRTS	Start Dort Signal
STATUS_DRTM	Dort Running signal
STATUS_DRTD	Dort Done Signal
STATUS_DRTF	Dort Finished Signal
STATUS_SNAP	Snap Status

주1)

## 5. 노심해석장치 시뮬레이터 통합시험 절차 개발

원전 시뮬레이터에 있어서 운전원의 조작이나 발전소의 자동제어 및 고유 운전특성 등에 대한 시뮬레이션은 정상, 비정상 및 비상시를 포함한 전 운전 범위의 운전에서 질량, 운동량 및 에너지 보존 법칙 등과 같은 자연법칙에 위배되지 않아야 하며, 운전원이 시뮬레이터 제어실에서의 응답 결과와 기준발전소와의 차이를 구분할 수 없도록 검증기준의 범위 이내에서 사실적으로 나타나야 한다. 시뮬레이터의 검증은 발전소 운전자료나 노심해석장치 부착 이전의 시뮬레이터 자료 및 이용 가능한 최적의 자료와의 비교에 의하여 수행되는데 이 자료의 정의는 다음과 같다.

- 실제 기준발전소의 운전자료
- 변환작업, 하드웨어 추가설치 및 소프트웨어 결함보고서에 사용된 자료
- 기술적 분석결과나 설계자료
- 기존 시뮬레이터의 성능자료, 시험자료, 설계자료

이러한 검증은 원자력발전소 훈련 및 면허시험용 전 범위 시뮬레이터의 기능조건을 규정한 ANSI/ANS-3.5-1998을 만족할 수 있도록 하여야 한다.

원전 시뮬레이터의 시험 프로그램은 1단계 시험 (Rehosting Test, Phase I Test), 2단계시험 (Upgrade Test, Phase II Test) 및 3단계 시험 (Site Acceptance Test, Phase III Test)로 분류된다. 1단계 시험은 시뮬레이터의 시스템 변환 시 변환이 완료된 주전산기 및 예비전산기와 부대설비를 기존 시뮬레이터 예비용 전산기와 교체 설치하고 하드웨어 판넬과 연결되는 I/O 하드웨어를 설치한 후 성능을 확인하기 위한 시스템 변환(Rehosting) 검증, 강사조작반 장비 및 사용 기능을 점검하는 강사조작반 시험, 시뮬레이션이 수행되는 각각의 발전소계통에 대한 소프트웨어적인 점검을 수행하는 계통별 시험과 하나 또는 다수의 고장 (malfunction)과 임의조작(override) 기능에서 시뮬레이터의 응답이 유효함을 증명하는 고장 및 임의조작 시험으로 구분된다. 2단계 시험은 추가 설치되거나 변경된 계기등이 (노심 출구온도 지시계, 각종 기록계, 경보창, ASP 등) 1단계 시험이 완료된 시뮬레이터와 연동되는지를 검사하는 하드웨어 구조 및 기능 검증 및 기존 시뮬레이터의 컴퓨터 계통을 성능개선한 부분과 기준 발전소의 설계변경 등으로 인하여 추가로 개발이 필요한 부분 등에 수행하는 컴퓨터 계통 시험으로 구분된다. 3단계 시험은 최적의 이용 가능한 자료와 시뮬레이터의 특정 과도상태나 전상상태에서의 자료 비교에 의하여 시뮬레이터의 성능을 증명하는 것으로 초기조건 평가 (Initial Condition Verification), 정상운전 (Normal Plant Evolution), 과도상태 시험 (Transient Test) 및 100시간 연속운전 시험 등으로 분류된다.

원전 시뮬레이터의 노심주기 계산 성능을 향상시키는 본 과제는 전체 시뮬레이터 중 변경사항이 노심해석 소프트웨어 및 하드웨어에 국한된다. 따라서 본 과제와 관련된 시험 프로그램은 3단계 시험에 국한하여 수행하는 것이 적절하다. 노심해석 소프트웨어 및 하드웨어 변경과 관련하여 컴퓨터 계통 시험은 2단계 시험 사항으로 분류되어 있기는 하나 3단계 시험 중 성능평가만으로도 그 기능을 충분히 입증할 수 있으므로 별도의 2단계 시험은 필요하지 않을 것으로 판단된다. 따라

서, 본 과제에서는 노심해석장치의 시뮬레이터 연계 성능을 검증하기 위하여 3단계 시험인 SAT(Site Acceptance Test)의 성능평가 시험 중 다음 절차서의 초안을 작성하였다.

- 시뮬레이터 종합운전 절차서
- 시뮬레이터 안정상태 시험절차서
- 시뮬레이터 열평형상태 시험절차서
- 시뮬레이터 과도상태 시험절차서

### III. 결론

지금까지, MASTER코드를 기반으로하여 영광1호기 시뮬레이터에 적용하기 위한 실시간 노심모델 개발과 관련하여 실시간 보장 및 타계통 모델과의 동기화 보증을 위한 통신 구약 및 방법론에 대하여 살펴보았다. 추후 원자력발전소 훈련 및 면허시험용 전 범위 시뮬레이터의 기능조건을 규정한 ANSI/ANS-3.5-2002를 만족하는지에 대한 검증 및 시뮬레이터 종합운전절차서, 안정상태 시험절차서, 열평형상태 시험 절차서 및 과도상태 시험절차서 등에 대해서도 실시간 보장 및 계산결과에 대한 검증이 이루어져야 할 것이다. 이러한 검증에서 만족한다면 설계코드 수준의 정확한 실시간 노심 모의 뿐 아니라, 향후 시뮬레이터의 노심주기 성능개선 작업에 용이하게 이용될 수 있을 것으로 기대된다.

### IV. 참고문서

- [1] Song, Jae Seung, Joo. Han Gyu, Zee. Sung-Quun, Noh. Jae Man, "MASTER-2.0 : Multi-purpose Analyzer for Static and Transient Effects of Reactors," Korea atomic Energy Research Institute, January, 1999.
- [2] REMARK, GSE Modeling Technical manual
- [3] ANSI/ANS-3.5 1998, Nuclear Power Plant Simulations for Use in Operator Training.
- [4] 전력연구원 외 2사, "울진표준형 원전 시뮬레이터 개발", 중간보고서, 2001. 9
- [5] 전력연구원 외 2사, "원자력교육원 2호기 시뮬레이터 성능개선", 최종보고서, 2001. 6
- [6] ANSI/ANS3.5 1998 Nuclear Power Plant Simulators for use in Operator Training and Licence Examination
- [7] 이명수 외, 원자력교육원 2호기 시뮬레이터 성능개선, 00-전력연-단165, 한국전력공사 전력연구원, 2000.4.
- [8] 삼성전자, 원자력교육원 #1 시뮬레이터 소프트웨어 납품서, 한국전력공사(1998)
- [9] Myeong-Soo Lee etc, The new research activities of KEPRI for KNPEC-2 Simulator Upgrade project, ASTC-Advanced Simulation Technologies Conference-2001 SCS 2001.
- [10] SimSuite Power SimExec User Guide, GSE