

디지털 통신회로 제어를 위한 전문가 시스템

박중식 김재희  
연세대학교 전자공학과

An Expert System for Network Control  
of Digital Communication Systems

Choong-Shik Park Jaihie Kim  
Yonsei University

ABSTRACT

An expert system for the network control of digital communication systems is developed by SALOME(System of Assorted Liberal Object Modelling Environment). SALOME is an expert system building tool which is developed by us, and can processes hybrid knowledge representations by using the object oriented programming technique. SALOME describes the communication network naturally and efficiently, and it can represent heuristic knowledge as well as algorithmic knowledge. As results, required size of database becomes smaller and more flexible controls are possible.

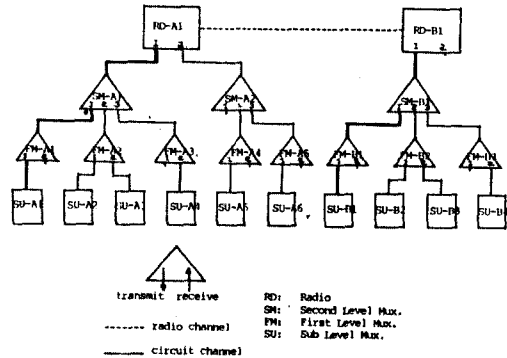
1. 서 론

요즈음의 통신회로망은 구성과 운영이 점차 복잡하여져서 기존의 운영방식으로는 전체 회로망의 상태를 쉽게 이해하고 효율적으로 운용하기 어려워졌다. 본 연구에서는 인공지능(Artificial Intelligence)의 기술을 이용하여 디지털 통신회로의 제어, 운영,자문등을 담당할 수 있는 전문적 지식을 지닌 프로그램 즉, 전문가시스템(Expert System)을 개발하였다. 이러한 전문가시스템을 손쉽게 개발하기 위하여서는 지식표현구조 및 추론기기를 제공하고 있는 전문가시스템 개발도구(Expert System building tool)로써 전문가시스템을 개발할 수 있었는데, 각 개발도구는 그에 적절한 지식표현방식(knowledge representation formalism)과 문제풀이 방식(problem solving method)을 가져야 하므로, 사용에 제한을 받게 된다. 본 연구에서는 여러 문제영역에 적합하도록 자체 개발한 전문가시스템 개발도구인 SALOME(System of Assorted Liberal Object Modelling Environment)로써 디지털통신회로 제어를 위한 전문가시스템을 개발하였다. 먼저 디지털통신회로 제어업무의 이해하고(2장), 그에 사용되는 지식처리의 특징을 생각함으로써 적절한 지식표현과 문제풀이 방법을 고려했다(3장). 자체 개발한 SALOME를 소개하고, SALOME가 이 문제에 적합한 개발도구인지 비교하였다(4장). 마지막으로 SALOME로써 어떻게 디지털통신회로를 위한 전문가시스템이 만들어졌는지 살펴보고자 한다(5장).

2. 디지털통신회로 제어업무의 성격

2. 1. 디지털통신회로의 구성

본 연구에서는 방위용 전용선로시스템(dedicated and non-switching system)을 디지털통신회로망의 모델로 사용하였다. 디지털통신회로는 많은 station으로 구성되며, 각 station은 여러 통신회로 부품들로 이루어져 있다. 각 station에 포함되는 통신회로 최하단위의 단말장치까지



(그림 1) 디지털 통신회로의 간단한 예

신호를 multiplexing 하기 위한 몇개의 포트를 가진 multiplexer들(Second Level Mux.:SM, First Level Mux.:FM, Sublevel Mux.:SU),Encryption device:XG로 이루어진다 [1,2].

이들 부품의 각 입,출력포트는 송,수신이 모두 가능한 두개의 simplex path로 이루어져 있다고 가정한다. 따라서 앞으로 간단히 multiplexer라고 하는 것은 실제로 multiplexer와 demultiplexer 기능을 모두 포함한 것이다. (그림 1)과 같은 두개의 station으로 이루어진 간단한 통신회로망을 생각해보자. 송자와 수자가 결정되어 있는 통신 경로이므로 통신회로의 연결상태가 통신회로의 송자와 수자를 결정한다. station A의 SU-A1과 station B의 SU-B1 사이의 통신경로를 고려할 경우 SU-A1의 송신방향(transmitting)신호는 FM-A1의 1번 포트, SM-A1의 1번포트를 지나 RD-A1에서 microwave로 station B의 RD-B1에 전달되면 역순이지만 같은 레벨의 부품은 각각 송신과 같이 대응되는 포트를 통하여 RD-B1의 1번포트, SM-B1의 1번포트, FM-B1의 1번포트를, SU-B1에 전달되며, 수신방향(receiving) 신호는 그 반대 방향으로 전달된다.

2.2. 디지털 통신회로의 제어업무 (고장검출과 복구, 자문)

통신회로의 제어업무중 가장 중요한 것은 고장부품의 신속한 위치 파악과 격리(fault location/isolation)와 고장으로 파손된 통신경로의 복구(channel reconfiguration)이다. 어떤 station의 한 회로부품이 고장나면 고장난 회로부품 때문에 그 통신경로를 따라 이로부터 지나가는 모든 부품들은 실제고장은 아니지만 실제로 고장난 것처럼 여겨진다. (그림 1)에서 SM-A1의1번포트에 송신방향으로 고장이 발생하면 SM-B1의 1번포트, FM-B1과 SM-B1의 모든 포트가 수신방향으로 고장난 것처럼 간주된다. 각 station은 이러한 실제고장과 가상고장을 모두 중앙통제소에 보고하게 되는데,

보고내용에 있어서는 실제고장과 가상고장의 차이가 없다. 이때 중앙통제소는 이러한 보고서에 실제고장과 가상고장을 빠르고 신속하게 구별하여 각 station에 통보해 주어야 한다 (fault location/isolation). 이러한 실제고장은 하나의 통신경로에 하나의 고장만을 가정한다면 같은 통신경로에서는 고장이라고 보고한 회로부품중에 맨앞에 존재하게 되므로 이의 위치를 찾는 것이 용이할 것으로 생각되지만 실제경우는 통신회로 부품의 위치 파악은 어려워지며, 현재에도 수작업으로 행해지는 많은 시간이 소비되는 작업이다.

통신경로의 복구는 고장검출에 의하여 찾아진 고장회로부품이 영향을 미치는 통신경로를 찾고, 대체할 수 있는 다른 통신경로를 찾아서 계속적인 통신을 가능하도록 하는 것이다. 대체통신경로를 발생시키기 위해서는 다음과 같은 단계의 방법들이 있을 수 있다.

- 1) 어떤 회로부품의 포트가 고장이 발생하면 그 회로부품에서 사용하지 않는 포트를 찾아 그 포트로 통신경로를 복구한다.
- 2) 모든 통신경로는 3가지의 우선순위(priority level)로써 high, middle, low 중에 하나를 갖는다. 어떤 회로부품의 고장포트가 high 순위라면 같은 회로부품의 고장나지 않은 low 순위의 포트로서 통신경로를 복구한다.

여기에서는 이러한 두가지만 고려하였는데, 포트부품이 아닌 부품 자체의 고장인 경우는 예비부품으로 대처하는 것도 고려할 수 있다. 그외에 디지털 통신회로의 사용자가 현재 통신회로의 운용상태 (연결상태, 우선순위상태, 고장상태등)를 알아볼 수 있도록 자문 역할도 하여야 한다.

### 2.3. 디지털통신회로 제어업무와 전문가시스템

디지털통신회로 제어업무를 제대로 컴퓨터 프로그램 한다는 것은 숙련된 통신회로 제어전문가 정도의 기능을 해야한다는 것이다. 그러므로, 제어업무의 컴퓨터 프로그램화에 인공지능기법을 이용한 전문가시스템이 이용되어야 하는 이유는 다음과 같다.

- \* 통신회로 제어전문가는 문제해결을 위해 정형화된 알고리즘 (algorithmic)한 지식뿐만 아니라 과거의 경험으로부터 알아낸 경험적(heuristic)인 지식을 사용한다.
- \* 경험적 지식을 사용하기 때문에 불완전한 정보의 처리도 다루어야 한다.
- \* 비록 통신회로 제어전문가가 경험적 지식을 이용하기는 하지만 엄청난 양의 정보를 처리해야 한다는 부담을 덜어줄 수 있다.
- \* 경험으로부터 축적된 통신회로제어 지식의 유지, 보수, 관리, 이전, 확장이 쉽게 이루어질 수 있다.

### 3. 디지털 통신회로 제어업무를 위한 지식표현과 처리.

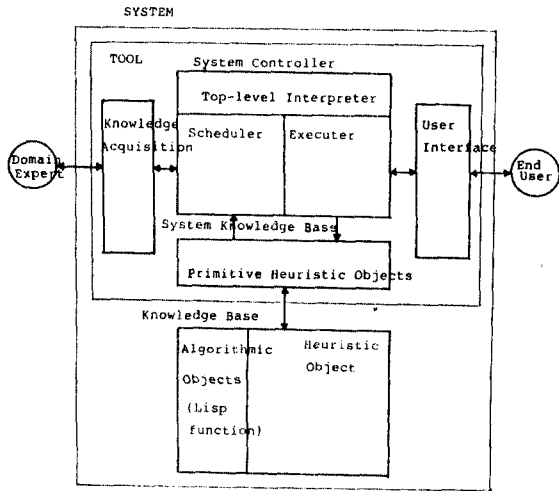
- 1) 디지털 통신회로 제어업무를 위한 전문가시스템을 개발하기 위해서는 인공지능에서 연구되는 적절한 지식표현방식과 문제풀이 방법이 선정되어야 할 것이다. 전문가시스템 개발 도구로써 전문가시스템을 개발할 경우에도 같은 문제가 고려되어야 한다. 전문가시스템 개발도구는 지식표현방식으로써 production rule, logic, semantic nets, frames [3,4,5,6] 를 각각 제공하거나 몇개의 지식표현이 동시에 제공되는 경우도 있다. 지식구성방법으로써는 문제분야로부터 규칙을 추측 함으로써 지식을 묘사하고, 구성하는 Top-down 방식 [7] 과 문제분야에 대하여 모델을 구성함으로써 예측의 생성과 과거의 원인-결과의 관계추적에 의하여 문제해결을 하도록 지식을 구성하는 Model-based 방식 [8], blackboard 라는 global database를 매체로 이용하여 여러개의 프로그램 모듈 (knowledge sources)이 상호보완하여 문제를 처리하도록 지식을 구성하는 Blackboard 방식 [9] 등이 있다.
- 2) 통신회로제어업무를 처리하기 위해서 대두되는 가장 큰 문제는 통신회로망에 형성되어 있는 방대한 양의 통신경로정보를 유지, 관리해야 한다는 것이다. 뿐만 아니라 통신회로망을 이루는 모든 회로부품의 기능을 묘사해야 한다는 것이다. 그러므로 통신회로망을 위하여 회로부품이나 통신경로를 묘사하지 않고 실제 통신회로망처럼 유지할 수 있는 통신회로 모델을 구성함으로써 유지, 관리해야 할 정보의 양이 많이 줄어들 것으로 생각된다. 왜냐하면 송, 수자가 회로부품연결에 의하여 결정된 통신회로이기 때문이다. 또 한가지 고려해야 할 문제는 통신회로 제어업무는 경험적으로 처리해야 할 정보도 있는 반면 정형화된 알고리즘에 의하여 쉽게 해결될 수 있는 문제도 있다. 즉, 고장검출과 같은 작업은 전통적인 알고리즘적인 프로그램에 의하여 쉽게 해결될 수 있으며, 고장회로 복구와 같은 작업은 경험적 지식을 처리해야 하기 때문에 production rule 이나 logic 이 요구된다. 어느

한가지 방법으로 두가지 문제를 해결하는 것은 아무래도 비효율적이다. 그러므로 복합적 지식처리 (hybrid knowledge processing) 가 가능해야 할 것이다.

## 4. SALOME

### 4.1. SALOME 의 특징

SALOME는 object-oriented programming (OOP) [10,11] 기법을 이용하여 model-based 지식구성과 복합지식처리 (hybrid knowledge processing)가 가능하도록 자체 개발한 전문가시스템 개발도구이다. OOP기법은 message 전달방식에 의한 조절기계(control mechanism)를 가지고 있어 프로그램 구성이 용이하며, inheritance 라는 데이터구조 (data structure)를 가지고 있으므로 정보를 계층적으로 묘사할 수 있어 지식을 효율적으로 구성, 이해하기 쉬우며, 데이터 중복을 최대한 억제할 수 있다. 기본적 데이터 단위는 local memory와 local function으로 구성된 object라는 것이며, 어떤 object가 message를 전달하면 local memory를 참조하여 local function에 의하여 만들어진 또다른 message를 다른 object에 보냄으로써 프로그램이 진행된다. 그러므로 object라는 데이터단위로 시뮬레이션하기 쉽기 때문에 model-based 시스템 구축이 가능하며, 다른종류의 정보처리단위를 통합할 수 있다. SALOME 가 OOP 기법을 채용하고 있으나, 여러가지 점에서 기존의 OOP 기법과는 다르며 그중에서도 가장 중요한 것은 object들 간에 relation 이라는 것을 정의하여 어떤 object가 처리한 message 를 또다른 어느 object에 보낼 것인지 정의할 수 있다는 것이다.



(그림 2) SALOME의 구조

### 4.2. SALOME 의 구조

SALOME는 시스템조절기 (system controller), 지식획득부분 (knowledge acquisition), 사용자 인터페이스 (user interface), 시스템 지식베이스 (system knowledge base), 문제분야의 지식베이스 (knowledge base)로 이루어져 있다 (그림 1). 시스템조절기는 지식획득부분이나 사용자 인터페이스에서 사용되는 명령어들을 처리하며, 시스템 구성 모듈간을 연결시켜준다 (top level interpreter). 그리고 프로그램 수행시 지식베이스에서 처리해야 할 object 를 스케줄링 하고 (scheduler), 실행시켜준다 (executer). 시스템 지식베이스에는 production rule이나 logic등을 처리할 수 있는 object가 정의되어 있으며, 유용하다고 생각되는 기능을 수행하는 object 가 정의되어 있다. 이것은 사용자가 문제분야의 지식베이스를 구성할 때 필요하다고 생각되는 object를 시스템이 미리 제공하는 것이다. 본 연구에서는 production rule 을 처리할 수 있도록 production rule engine만 object로 제공하였다. 지식베이스에는 문제분야의 지식이 object 형태로 주입된다.

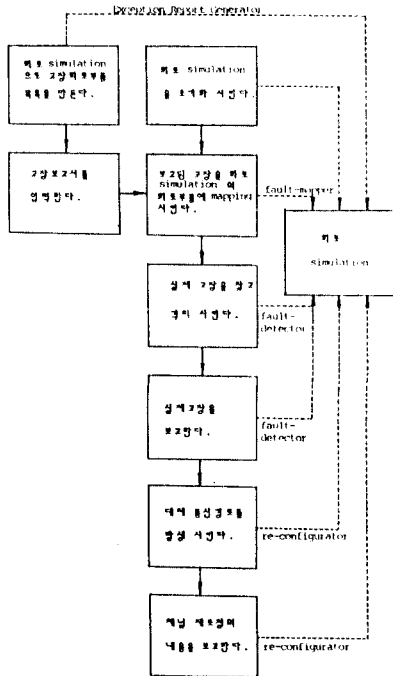
이 때 알고리즘적인 지식처리를 위해서 LISP 함수를 사용할 수 있으며, 경험적 지식을 처리하기 위하여 production rule을 사용하기 위해서는 시스템 지식베이스의 production rule engine object의 subclass로 정의하면 된다.

지식회독부분은 지식베이스에 지식을 주입하기 위하여 간단한 형태의 지식베이스 편집기 (knowledge base editor)를 제공하게 되며, 간단한 명령어 형태로 구성되어 있다. 이러한 명령어는 menu 구성방식에 의하여 제공되어 사용하기에 편리하도록 하였다.

사용자 인터페이스는 사용자로 하여금 SALOME 사용방법을 유도하며, 지식베이스의 내용점검, 프로그램의 수행, 수행시의 시스템상태점검등을 할 수 있도록 한다. 이러한 기능들은 지식회독부분과 마찬가지로 간단한 명령어로 구성된 menu에 의하여 제공된다. SALOME는 스크린상에 시스템의 진행상황을 알려주는 화면, object의 상태를 조사할 수 있는 화면, 그리고 프로그램의 일반적 입출력을 위한 화면등 세개의 화면으로 이루어진다.

### 5. 디지털 통신회로 제어 위한 전문가시스템의 구성

SALOME를 이용하여 일단 통신회로를 시뮬레이션하고, 그 회로 시뮬레이션 자체를 문제풀이의 매체로 사용한다. 통신회로부분의 특성에 기인되는 고장검출 (fault location/isolation)은 알고리즘적인 방법으로 회로 시뮬레이션자체를 이용하여 해결하고, 통신경로의 복구(channel reconfiguration)는 production rule로 처리하여 경험적 지식을 사용할 수 있도록 하였다. 통신회로에 대한 전문도 통신회로의 특성과 구조에 관계되므로 회로시뮬레이션을 이용하여 쉽게 해결할 수 있다. 또한 고장검출의 입력자료로 쓰이는 각 station으로 부터의 고장보고도 실험을 위하여 컴퓨터가 작성하는 시험용 고장보고서 (Exception - Report)로 대체되어야 하는데, 이것 또한 회로시뮬레이션에 의하여 임의의 고장을 가정하여 자유롭게 작성할 수 있다. (그림 3)



(그림 3) SALOME의 디지털 통신회로의 제어업무 순서

이와같은 작업들을 위하여 지식베이스에 회로시뮬레이션을 위한 회로부분 object, 고장보고서 작성을 위한 object, 고장검출을 위한 object, 그리고 통신경로 복구를 위하여 production rule를 가진 object를 정의한다.

#### 5.1. 회로시뮬레이션을 위한 object

OOP기법으로 디지털 통신회로를 시뮬레이션하기 위해서는 각 회로부분을 object로 정의하여야 한다. 즉 회로부분의

입력과 출력, 그리고 유지해야할 내부상태가 규정되어야 하며, 어떤 입력이 들어왔을때 그때의 내부상태를 고려하여 어떤 출력을 내보낼지를 결정하도록 정의되어야 한다

```
(DEF-CLASS
  '(NAME SM)
  '(SUPER EQUIP)
  '(RELATION mo ol o2 o3)
  '(INT-STATES (level 3) (ol 0) (o2 0) (o3 0))
  '(BODY .....))
(INSTANCE-OF 'SM '(SM-A1 SM-B1))
(RELATION 'SM-A1 'mo '(RD-A1)
  'ol '(FM-A1) 'o2 '(FM-A2) 'o3 '(FM-A3))
```

이 시스템에 고려하는 통신회로부분은 RD, SM, FM, SU, KG 등이며 위의 사항은 SM object를 정의한 예이다. 통신회로의 모든 SM의 공통된 성격을 class로 정의한다. 일반적인 SM은 각 포트로 입.출력시킨다고 정의하고, 각 포트의 상태와 주파수처리 속도에 따른 정보처리비율을 내부상태로 보낸다. SM와 다른 통신회로부분의 공통된 성격은 EQUIP class를 SM class의 상위 class로 정의하였다. 또한 SM의 기능은 body에 정의되는데 지면상 생략하였다. 이렇게 정의된 SM class는 실제로 통신회로를 이루는 개별적 SM 들을 instance로 정의함으로써 SM class의 모든 성격을 전수 받도록 한다. 또한 개별적 SM 들이 실제로 어떻게 연결되어 있으며, 어디에 정보를 전달해야 하는지를 relation으로 정의한다.

이렇게 모든 회로부분을 정의하면 회로시뮬레이션이 구성되고 실제로 임의의 회로부분에 신호를 전달하면 신호가 전달되는 상태를 확인할 수 있으며, 통신회로의 연결상태나 통신경로등에 대한 자문을 구할 수 있다 (그림 4, 5)

```
(SRND-MBSSAGE 'STA-A :info)
```

```
>> STATION EQUIPMENT CONNECTION INFORMATION
RADIO
RD-A1:01 => (SM-A1),02 => (SM-A1)
SECOND LBVL MUX.
SM-A1:01 => (FM-A1),02 => (FM-A2),03 => (FM-A3)
SM-A2:01 => NIL ,02 => (FM-A4),03 => (FM-A5)
FIRST LBVL MUX.
FM-A1:01 => (SU-A1),02 => NIL
FM-A2:01 => (SU-A2),02 => (SU-A3)
FM-A3:01 => NIL ,02 => (SU-A4)
FM-A4:01 => (SU-A5),02 => NIL
FM-A5:01 => (SU-A6),02 => NIL
```

(그림 4) station의 부품연결을 알아보기 위한 예

```
(SRND-MBSSAGE 'SU-A1 :path 'ol)
```

```
>> PATH LIST OF SU-A1 01 PORT
(SU-A1 01) <=> (FM-A1 01) <=> (SM-A1 01) <=>
(RD-A1 01) <=> (RD-B1 01) <=> (SM-B1 01) <=>
(FM-B1 01) <=> (SU-B1 01)
```

(그림 5) 회로부분의 통신경로를 알아보기 위한 예

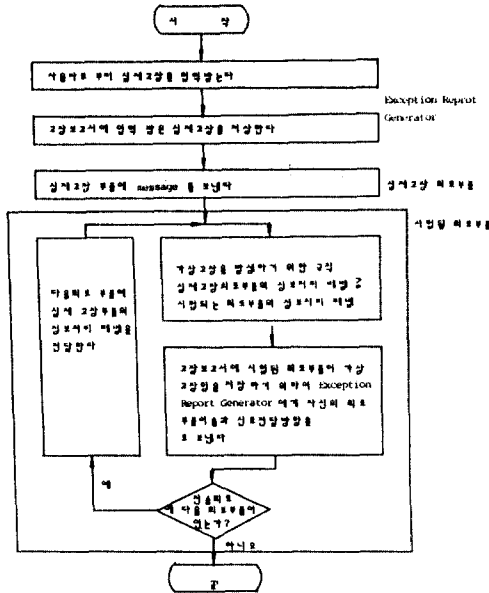
#### 5.2. 시험용 고장보고서 작성을 위한 object (Exception-Report-Generator)

실제고장과 파급된 가상고장으로 이루어진 고장보고서는 고장검출과정에 필수적인 입력자료이다. 사용자가 exception-report-generator 라는 object에 임의의 회로부분에 있는 임의의 포트에 임의의 방향이 실제고장이라고 정해지면 object는 그 부분이 실제고장일 경우에 파급되어 생길 수 있는 가상고장을 발생시켜 실제고장과 파급된 가상고장을 임의의 순서로 정리하여 고장보고서를 작성한다 (그림 6,7). 이 입력자료에 의하여 고장검출과정이 끝난후 찾아낸 실제고장과 가정된 실제고장을 비교해 볼 수 있다.

#### 5.3. 고장검출을 위한 object

(Fault-Mapper, Fault-Detector)

고장보고서를 입력으로 받아들여서 각고장부분의 상태를 회로시뮬레이션의 회로부분에 mapping시킨다(alarm mapping). 이러한 mapping은 회로부분 instance의 port-flag에 표시된다 (예: (01 tx) (02 rx)). 이러한 상태에서 실제고장과 파급된 가상고장을 구별하는 것이다 (고장검출). 어떤 통신경로에 하나의 실제 고장부분이 존재한다고 가정하면, 이 통신경로에 존재하는 여러



(그림 6) Exception Report Generation 순서

```
(SEND-MESSAGE 'EXCEPTION-REPORT-GENERATOR
:real-fault-list
'((SM-A1 (O1 TX))))
```

```
fault-list =>
((SM-A1 (O1 RX))(SM-A1 (O1 TX))
(FM-B1 (O1 RX))(SU-B1 (O1 RX)))
```

(그림 7) Exception Report Generation의 예

고장부품중에서 신호전달 방향의 제일 앞에 있는 것이 실제고장이고 나머지는 모두 이것에 의한 가상고장이다. 위와같은 작업을 하기위하여 fault-detector 와 fault mapper object 를 정의하는데 다음과 같은 방법으로 가능하도록 정의한다.

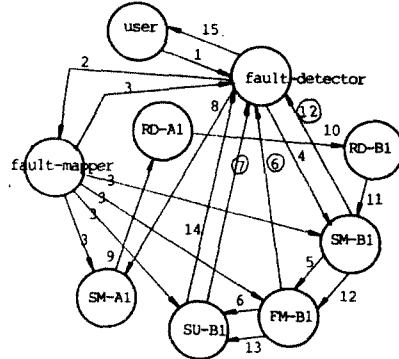
- 1) 사용자가 fault-detector에게 고장보고서를 message 로 보내면 fault-detector는 fault-mapper로 하여금 회로시뮬레이션의 해당 회로부품 instance에 고장표시를 하게 한다.
- 2) fault-mapper가 고장표시가 완료되었음을 알려주면, fault-detector는 고장보고서에서 조사해야할 임의의 고장부품을 골라 해당회로부품 instance 에 신호를 보낸다.
- 3) 신호를 받은 회로부품 instance는 자신의 정보처리레벨을 고장선로 방향에 있는 회로부품 instance 에 보낸다
- 4) 정보처리레벨을 전달받은 회로부품 instance 는 자신이 고장이면서 전달받은 정보처리레벨보다 높거나 같으면 자신을 고장보고서 조사대상에서 제외하도록 fault-detector 에게 message 를 보내고 (가상고장), 그렇지 않으면 message를 보내지 않는다.
- 5) 4)가 끝난 instance는 전달받은 정보처리레벨을 자신의 선로 전달방향으로 보낸다.
- 6) 만약 전달할 곳이 없으면 fault-detector 에게 다른 고장 부품을 조사하도록 message 를 보낸다.
- 7) fault-detector는 고장보고서에서 조사해보지 않은 고장부품이 남아있지 않을 때, 조사해보았지만 없어지지 않고 남아있는 고장부품이 실제고장이므로 이들을 모아서 사용자에게 보고한다.

(그림 1)와 같은 통신회로에서 SM-A1의 이 포트가 송신방향으로 고장이 났을 경우의 예가 (그림 8,9)이다. (그림 9) 의 동그라미 표시한 번호는 가상고장을 보고하는 경우를 의미한다.

```
(SEND-MESSAGE 'FAULT-DETECTOR
: fault-list
'((SM-B1 (O1 RX))(SM-A1 (O1 TX))
(FM-B1 (O1 RX))(SU-B1 (O1 RX))))
```

```
"REAL FAULT LIST"
(SM-A1 (O1 TX))
```

(그림 8) 고장 검출의 예



(그림 9) 고장 검출시의 object-message 순서

#### 5.4. 통신경로 복구를 위한 object (Re-configurator)

어떤 회로부품의 포트가 실제고장임이 발견되면, 그 포트를 지나가는 통신경로를 재조정하여 대체 통신경로를 만들어 주어야한다. 통신경로 복구는 production rule에 의하여 처리하도록 하였으며, rule들이 들어있는 production memory(PM)은 object 의 내부상태로 저장된다.

- 1) 통신경로 복구를 위한 object (re-configurator) 가 먼저 실제고장보고를 받게되면 해당 고장부품으로부터 각 포트의 사용유무, 우선순위에 대한 정보를 요구한다.
- 2) 해당 고장부품은 그러한 정보를 자기 포트에 연결된 회로부품들로부터 모아 re-configurator or 에게 보낸다.
- 3) reconfigurator는 이 정보를 working memory(WM)으로 하고 PM 에 의해서 추론해 나가면서 결론에 도달하면 해당고장부품에게 회로 복구방법을 message 로 보낸다.
- 4) 해당 고장부품은 복구정보에 의해서 포트를 대체하고, 자신의 대칭회로부품에게 다시 복구정보를 보낸다.
- 5) 대칭회로부품도 복구정보를 받아 포트를 대체한 다음, re-configurator 는 사용자에게 어떻게 복구되었는지 알려주고 업무를 끝낸다.

이러한 일을 하기 위한 production rule은 12개가 사용되었는데 예를들면 다음과 같은 것이다.

```
RULE12 : 회로부품 (?y)의 포트(?x) 가 고장이고 다른
포트 (?z)보다 높은 우선순위가면, 회로부품
(?y)의 포트 (?x)는 포트 (?z)를 대체하도록
사용자에게 보고하고 회로부품 (?y) 에게
포트(?x)로 포트(?z)로 대체하도록
message를 전달한다.
```

```
(r12 if ((?x port of ?y is out-of-order)
(In ?x port of ?y is high priority)
(In ?z port of ?y is low priority))
then ((output in ?y, ?x port is replaced by ?z)
(send-m ?y 're-route (list ?x ?z))))
```

(그림 10)은 통신경로복구의 한 예이다.

```
(SEND-MESSAGE 'RR-CONFIGURATOR
:real-fault-list
'(SM-A1 (O1 TX)))
```

```
:REPORT
(IN SM-A1 O1 PORT IS REPLACED BY O3)
```

```
In Symmetric Equipment SM-B1
O1 is replaced by O3
```

(그림 10) 고장 복구의 예

## 6. 결. 론

디지털통신회로제어를 위하여 통신회로에서 고장을 검출하고, 대체통신경로를 발생시키며, 회로운영에 관하여 자문할 수 있는 전문가시스템 개발에 있어서 model-based 방식을 사용하여 회로부품을 시뮬레이션함으로써 기존의 디지털통신 제어시스템과 같이 모든 통신경로를 유지할 필요가 없으므로 저장할 정보의 양을 줄일 수 있었고, 그 회로시뮬레이션을 기반으로 고장검출과 같이 정형화된 업무는 알고리즘적인 방법으로 해결할 수 있었으며, 통신경로의 복구와 같이 경험적 지식을 요하는 업무는 production rule로 처리함으로써 복합지식처리를 사용하여 지식처리에 효율을 기하였다.

앞으로 좀더 복잡한 통신회로 제어업무를 처리할 수 있도록 지식베이스를 강화하는 것이 필요하며, 중앙통제소를 거치지 않고 station 간에 정보를 교환하므로 고장검출과 복구가 가능한 분산처리방식의 연구가 요구된다.

## REFERENCE

1. Report CPMAS 80-01, Jan. 1980, entitled, "Automated Performance Monitorig and Assessment for DCS DigitalSystem," Prepared by GTE Products Group, Sylvania Systems Group, Needham Height. MA. 02194.
2. Yoh-Han Pao and Jaehie Kim, "A linked list database approach to the control of the digital transmission facilities of the Defence Communications System", Proc. of 1982 IEEE Milcom Conf., Oct. 1982, pp(16.4-1)-(16.4-5).
3. D. A. Waterman and F. Hayes-Roth, Pattern-Directed Inference Systems, Academic Press, Inc., Orland, Florida, 1978.
4. W. F. Clocksin and C. S. Mellish, Programming in Prolog, Springer Verlag Berlin Heidelberg, 1981.
5. N. V. Findler, Associative Networks, Academic Press, Inc., London, 1979.
6. Terry Winogord, "Frame Representations and The Declarative/Procedural Controversy", D. G. Bobrow and A. Collins (eds.), Representation and Understanding, PP185-211.
7. P. Harmon and David King, Expert Systems, John Wiley & Sons. Inc., 1985.
8. Yong-Bok Lee, Constraint Propagation In A Pathophysiologic Casual Network, Department of Electrical Engineering and Applied Physics, Case Western Reserve University, Doctor Thesis, August 21, 1986.
9. Robert Balzer et al., "HERASAY-III: A Domain-Independent Framework for Expert Systems," AAAI-80, pp108-110.
10. Brad J. Cos, Object Oriented Programming, Addison Wesley Publishing Co. Inc., 1986.
11. Adele Goldberg and David Robson, Smalltalk-80: The Language and Implementation, Addison Wesley Publishing Co. Inc., 1983.