철도신호용 통신프로토콜에 대한 검정 및 시험방안에 관한 연구

이재호*, 황종규*, 서미선**, 김성운**
한국철도기술연구원, ** 부경대학교

A Study on Verification and Test Plan of Communication Protocol for Railway Signalling Systems

Lee Jae-Ho*, Hwang Jong-Gyu*, Seo Mi-Seon**, Kim Sung-Un**
* Korea Railroad Research Institute, ** PuKyung National University

Abstract - 프로토콜 명세의 검정과 적합성 시험은 프로토콜 개발 과정에서 가장 중요한 부분으로, 명세에 규정되어진 시스템 기능의 정확성을 향상시키는데 사용되는 상호 보완 기술이다. 본 논문에서는 유한상태 레이블 천이시스템(LTS:Labeled Transition System)으로 명세화된 철도 신호제어용 프로토콜 Type 1 모델의 안전성 및 필연성 특성을 모형검사 기법에 의해 검정하였고, 실제적으로 교착상태의 유무나 초기 상태에서 임의의 상태로 도달 가능한지의 검사를 실험적으로 증명하였다. 구현되어진 형식 검정기는 Modal mu-calculus를 사용하며 Modal 논리로 표현된 특성이 명세에 대해 올바른지 아닌지를 검정할 수 있다. 또한 검정되어진 프로토콜 명세로부터 UIO(Unique Input Output) 방법에 의한 적합성 시험 계열 생성 방법을 제시하였다.

1. 서 론

통신 소프트웨어에 대한 사용자의 요구사항이 복잡화, 다양화, 대형화되어짐에 따라 개발에 따르는 어려움은 더욱 증대되었고, 과거에 사용된 비정형적 방법(Informal method)에 의한 개발은 많은 오류와 비효율성을 내포하고 있다. 따라서 개발에 소요되는 비용 및 시간을 절약하고 심각한 오류를 형식적 방법에 의해 검출하는 기술인 정형기법(Formal method)의 적용이 증가하는 추세이며, 특히 검정, 구현 및 시험 시에 전통적 자연어에 근거한 프로토콜 개발에 비하여 더 체계적인 방법을 제공한다[1].

통신을 위한 프로토콜들이 상호 적절한 기능을 유지하기 위해서는 잠재적 설계에러가 없어야 하며, 사용자 요구사항과 일치하고 다른 프로세서와의 통신이 원활하게 이루어져야 한다. 따라서, 하나의 새로운 통신시스템을 구현하기 위해서는 사용자 요구사항 분석, 프로토콜의 구조적 설계 및 명세화, 검정, 구현, 적합성 시험 등의 단계를 거쳐야 하며, 이러한 프로토콜 개발 단계 중 검정 및 적합성 시험은 가장 핵심이 되는 부분이다.

사용자 요구사항과 명세와의 일치성을 구현 전에 확인하는 검정단계는 모든 프로토콜에 필수적인 정확성(Correctness)을 만족하는가에 대해 프로토콜 명세를 분석하는 과정으로서 모형검사(Model checking)는 이러한 과정을 자동적, 형식적으로 검정하는 기술이며, 일반적으로 LTS를 사용하고 있다.

본 논문에서는 LTS로 명세화된 프로토콜 Type 1 모델의 안전성 및 필연성 특성을 모형검사기법에 의해 검정하기 위해 대수적 명세기법인 Modal mu-calculus를 사용하는 방법과 검정되어진 프로토콜 명세로부터 최적화 기술 기법에 의한 적합성 시험 계열 생성 방법을 제시하는 것이 주요 목적이다.

2장에서는 검정 및 시험할 대상인 철도 신호제어용 프로토콜에 대해 기술하고, 3장에서는 시험 대상의 동작과정을 검정 및 시험하기 위한 중간모델인 유한상태기계(I/O FSM : Input/Output Finite State Machine)과 LTS로 모델링 하였으며, 4장에서는 LTS로 명세화된 철도 신호제어용 프로토콜을 대수적 명세 기법 중 프로토콜의 행위 특성을 가장 강력하게 표현하는 Modal mu-calculus를 사용하고 Model checking 알고리즘인 Solve 알고리즘을 적용하여 프로토콜 명세 특성(안전성, 필연성)을 검정하는 방법을 제시하였다. 5장에서는 적합성 시험에 대해 기술하고 생성된 I/O FSM 중간 모델을 대상으로 UIO 시퀀스 생성 및 시험 계열 생성 방법을 나타냈으며, 6장에서는 본 논문의 결론과 향후 추진사항에 대해 기술한다.

2. 철도 프로토콜 정의 및 특성

철도신호제어용 프로토콜 Type 1은 철도청에서 운용하는 열차집중제어장치(CTC:Centralized Traffic Control)와 전자연동장치(EIS:Electronic Interlocking System)간의 인터페이스를 위한 프로토콜로, 현장 신호기기실에 설치되어 CTC와 EIS간의 정보전송방식을 담당하는 역정보전송장치(LDTS:Local Data Transmission System)와 선로전환기, 신호기 등의 현장신호설비를 제어하고 감시하는 전자연동장치(EIS) 사이의 정보를 전송하는 방식이다. LDTS와 EIS 사이의 최대 전송거리는 100m를 초과하여서는 안되고 인터페이스 링크 구성은 RS-422 방식이며, 메시지 흐름제어는 정지-대기(Stop-and-wait) 제어방식을 사용한다.

2.1 메시지 프레임의 구조

프로토콜 Type 1의 LDTS와 EIS간 전송메시지 프레임 구조는 그림 1과 같다[2].

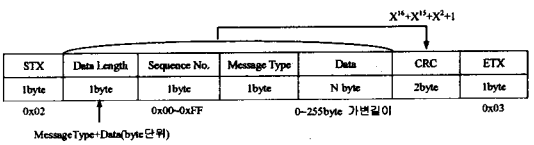

그림 1 Type 1 메시지 프레임

STX(Start of Text)는 메시지 프레임의 시작을 나타내며 0x02로 표기하고, Data Length는 Message Type에서 Data 필드까지의 길이를 byte 단위로 나타낸 것이다. 또한 Sequence Number는 메시지의 전송순서를 나타내기 위해 사용되며 0x00부터 0xFF를 사용해서 한 메시지 프레임의 전송이 성공하면 시퀀스 번호를 하나씩 증가하게 한다. Message Type은 전송되는 메시지의 형식을 의미하며 Data 필드는 실제 전송할 가변 데이터이다. CRC 필드는 Data Length 필드부터 Data 필드까지의 에러 검출을 위해 사용되고 생성다항식은 $X^{16}+X^{15}+X^2+1$을 사용하며, ETX (End of Text)는 메시지 프레임의 마지막을 나타내고 0x03으로 표기한다.

3. FSM 명세 및 LTS 모델링

본 장에서는 2장에서 설명한 철도신호용 프로토콜 Type 1의 행위 특성에 기반해서 I/O FSM과 LTS로 변환한 모델에 대해 기술한다.

3.1 Type 1의 I/O FSM 모델링과 동작분석

일반적으로 I/O FSM은 프로토콜의 제어 부분을 명세화하는데 많이 사용되어져 왔으며 그 정의는 다음과 같다[3][4].

정의3.1 입출력 유한상태기계(I/O FSM)
I/O FSM은 다음의 다섯 가지 요소로 구성된다.
M = <S, so, I, O, tr>
여기서
- S : 한정된 상태들의 집합
- so : 초기 상태
- I : 한정된 입력 알파벳
- O : 한정된 출력 알파벳
- tr : 천이함수,

$$t_r \subseteq \{s\text{-}i/o \rightarrow s' \mid s,s' \in S \wedge i \in I \wedge o \in O\}$$

다음의 4개 요소(p, a, b, q)는 I/O FSM M의 천이(Transition)로 불려지는데 아래와 같이 정의된다.

$$(p, a, b, q) \in S \times I \times O \times t_r$$

위의 정의에서 각 천이는 하나의 입력과 하나의 출력에 의해 일어난다.

LDTS는 EIS로 폴링 메시지와 진로설정이나 선로전환기 혹은 주신호기 등을 제어하는 제어 메시지를 전송하고, EIS는 LDTS로 현장 신호설비들의 상태정보 메시지와 제어 메시지에 대한 응답(ACK)을 전송한다. 그림 2는 Type 1의 흐름 제어를 I/O FSM으로 모델링한 것이다.

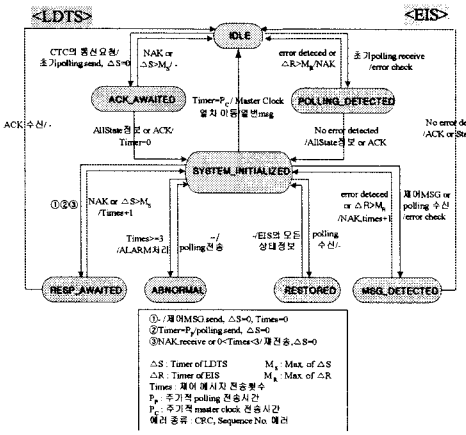

그림 2 Type 1의 I/O FSM 모델링

시스템의 초기화 시, LDTS에 의해 메시지 전송이 시작되며 EIS는 LDTS의 폴링 메시지에 대하여 응답을 함으로써 통신을 시작한다. LDTS로부터 메시지를 수신한 EIS가 에러검출 코드를 검사하고 메시지가 정상일 경우 ACK를, 에러가 검출될 경우에는 NAK 응답을 LDTS로 전송한다. 송신측에서는 전송 메시지 프레임의 끝(ETX)이 전송되면 타이머 △S를 동작시키며, 수신측에서는 수신 메시지 프레임의 시작(STX)이 수신되면 타이머 △R를 동작시켜 메시지가 정상적으로 전송되었는지를 확인한다. LDTS로부터 수신한 메시지에 대해 EIS에서 NAK를 응답하게 되는 경우는, CRC 검사에서 에러가 검출되거나 메시지 시퀀스 번호의 에러가 발생한 경우, 또는 △R이 종료되었음에도 메시지 프레임의 끝을 의미하는 ETX를 수신하지 못하는 경우이다. 송신측에서는 동일한 메시지를 3회까지 재 전송할 수 있는데, 3회 재 전송하였음에도 NAK 또는 어떠한 응답도 받지 못할 경우, 통신이상으로 판단하여 해당 메시지는 삭제시키고 경고(Alarm)처리 한다. 이때 LDTS는 시퀀스 번호 0x00을 갖는 폴링 메시지를 전송하여 LDTS와 EIS 사이의 통신이 정상적으로 복구되는지를 확인한다.

3.2 Type 1의 LTS 모델링

LTS는 프로토콜을 검정하는 정형명세기법 중 프로토콜 정형명세를 위한 의미 모델로서 많이 사용되며 다음과 같이 정의한다[5].

정의3.2 레이블 천이 시스템(LTS)
LTS는 다음 4개의 요소로 구성된 레이블화된 천이시스템 LTS = <S, L, T, so>으로 정의된다.
- S : 상태들의 집합
- L : 관찰 가능한 행위집합
- T ⊆ Sx (Lτ) x S : 천이관계
- So S : 초기 상태

이 정의에서 심볼 "τ"는 비결정형 모델(Non-determinant model)을 위해 이용되는 시스템 내부 혹은 관찰이 가능하지 않은 행위(Internal 혹은 Inobservable action)를 나타낸다.

프로토콜 Type 1을 프로토콜 정형명세를 위한 의미모델로써 많이 사용되는 LTS로 모델링한 것은 그림 3과 같다.

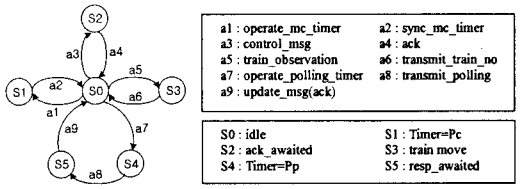

그림 3 프로토콜 Type 1의 LTS 모델링

4. 철도신호용 프로토콜 Type 1 검정 방법

본 장에서는 대수적 명세기법인 Modal mu-calculus를 소개하고 3장에서 제시한 LTS 모델(그림 3)의 안전성(Deadlock 이나 Livelock이 없음)과 필연성(정당한 프로토콜의 특성 즉, 상태 또는 행위가 결국 만족되어짐) 특성을 Modal mu-calculus에 기반을 둔 Model checking 방법을 사용해서 검정하는 완전성 검정 과정을 일 예로 기술한다.

4.1 검정사항

프로토콜 검정은 프로토콜 명세의 정확성, 안전성과 필연성(Liveness) 및 일관성(Consistency) 등을 알아보는 것으로 model checking에서 보다 구체적으로 검정해야 할 프로토콜의 특성이다.

4.2 Modal mu-calculus

시스템 특성 명세 언어인 Modal mu-calculus는 시스템 상태의 부분 집합이 공통적으로 만족하는 논리식인 고정점을 가진 Modal 논리이며, 연산자는 원자명제(Atomic proposition), ∧(논리곱:Conjunction), ∨(논리합 :Disjunction), [ ](필요성: Necessity), < >(가능성: Possibility), ν(최대고정점: Greatest fixed point), μ(최소고정점: Least fixed point)로 구성되어 있으며, 일반화된 Modal mu-calculus의 논리식은 다음과 같다.

$$\Phi ::= tt \mid ff \mid Z \mid \Phi 1 \wedge \Phi 2 \mid \Phi 1 \vee \Phi 2 \mid [k]\Phi \mid <k>\Phi \mid \nu Z.\Phi \mid \mu Z.\Phi \qquad (1)$$

식 (1)에서 tt는 모든 상태에 대해 참인 것을 나타내고 ff는 거짓임을 나타내며 Z는 명제적 변수, k는 천이집합의 원소, Φ는 프로세스 특성을 나타낸 논리식이다. νZ는 시스템의 상태 집합들이 일반적으로 만족하는 특성인 최대고정점 연산자이며, μZ는 시스템의 상태 집합들이 공통적으

로 만족하는 최소고정점 연산자이다[6].

4.3 검정 알고리즘

Modal mu-calculus논리식으로부터 Deadlock, Livelock, Rechability 그리고 Liveness 유무를 검사하기 위해 LTS모델에서 주어진 논리식으로 구성된 비순환적인 블럭들의 닫혀진 집합, ||B||를 R.cleaveland와 B.steffen이 개발한 Model checking 알고리즘인 Solve를 사용하여 구한다[7]. 이 Solve 알고리즘은 Modal mu-calculus의 재귀적 특성을 이용한 것이다.

4.4 검정 적용 예

위의 2절에서 기술한 프로토콜의 안전성과 필연성 특성을 표현하는 Modal mu-calculus 식으로부터 프로토콜 Type 1 LTS의 완전성을 검정하는 방법을 보인다. 예를 들어, Deadlock과 Livelock이 없음을 나타내는 Modal mu-calculus의 논리식은 다음과 같으며, 식이 참이 되기 위해 <-> tt와 [-]Y, [-]Z는 동시에 참이 되어야 한다.

$$\nu Z.(\mu Y.A \vee (<->tt \wedge [-]Y)) \wedge [-]Z \quad 단,\ A=\{S_0\} \qquad (2)$$

위 식 (2)에 Model checking 알고리즘인 Solve 알고리즘을 적용하여 검정결과를 검출한다. 그림 6은 식 (2)에서 최대고정점과 최소고정점을 사용하여 생성된 Max block과 Min block을 나타낸다.

그림 5는 그림 4에서 생성된 Max block과 Min block의 변수 Xi들의 천이 관계를 나타낸 Edge-labeled directed graph G이며, 그림 6은 Solve 알고리즘의 초기화 규칙에 의해 초기화된 Bit-vector, Counter 및 배열을 나타낸다.

$B_1 \equiv \min\{\ X_1 = X_2 \vee X_3,\ X_2 = A,\ X_3 = X_4 \wedge X_5,\ X_4 = [-]X_1,\ X_5 = tt\ \}$

$B_2 \equiv \max\{\ X_7 = X_1 \vee X_8,\ X_8 = [-]X_7\ \}$

그림 4 Max and min block

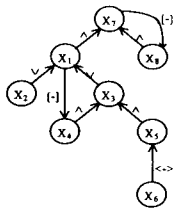

그림 5 Edge-labeled directed graph G

X	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8
S0	0	1	0	0	0	1	1	1
S1	0	0	0	0	0	1	1	1
S2	0	0	0	0	0	1	1	1
S3	0	0	0	0	0	1	1	1
S4	0	0	0	0	0	1	1	1
S5	0	0	0	0	0	1	1	1

C	X_3	X_4
S0	2	4
S1	2	1
S2	2	1
S3	2	1
S4	2	1
S5	2	1

M[1]=<<S0, X2>, <S0, X6>, <S1, X6>, <S2, X6>, <S3, X6>, <S4, X6>, <S5, X6>>

M[2]=< >

그림 6 Bit-vector, counter 및 Array M[i]

그림 7은 초기화된 배열에서 배열 M[i]가 공집합(Empty)이 될 때까지 갱신 알고리즘을 적용하여 Bit-vector와 Counter를 갱신한 결과로, 검정 대상인 Deadlock, Livelock 및 Reachability를 판단할 수 있다.

Deadlock은 Bit-vector의 요소에 의해 판단되는데, 그림 6에서 Bit-vector의 요소는 모두 1로 갱신되어 Deadlock이 발생되지 않았음을 알 수 있다. 또한 Livelock은 관련된 상태의 Counter 요소에 의해 판단되며 위의 결과에서 Counter의 요소는 모두 0으로 갱신됨으로서 Livelock이 검출되지 않았음을 알 수 있다. 즉 위 예시의 모든 요소가 Max block과 Min block을 만족하므로 그림 3의 LTS 모델은 논리식 (2)을 만족하는 완전한 프로토콜 모델이 판단되었다.

X	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8
S0	1	1	1	1	1	1	1	1
S1	1	0	1	1	1	1	1	1
S2	1	0	1	1	1	1	1	1
S3	1	0	1	1	1	1	1	1
S4	1	0	1	1	1	1	1	1
S5	1	0	1	1	1	1	1	1

C	X_3	X_4
S0	0	0
S1	0	0
S2	0	0
S3	0	0
S4	0	0
S5	0	0

M[1]=< > M[2]=< >

그림 7 Bit-vector, Counter and Array M[i]

5. 적합성 시험

적합성 시험의 목적은 어떤 프로토콜의 구현 I(Implementation)가 원래 명세 S(Specification)에 합당하게 구현되었는지를 시험하는 것으로 통신 프로토콜 제품 구현 과정에서 중요한 역할을 한다[8]. 일반적으로 적합성 시험의 정의는 주어진 명세 S를 기초로 하여 생성된 시험 계열(Test cases)로서 구현 I가 명세 S에 대해 프로토콜 행위(Behaviour)와 능력(Capacity)이 일치하는지를 시험하는 것이다.

5.2 UIO방법에 의한 시험 계열 생성

UIO 시퀀스는 시험하는 천이 후에 도착한 상태의 유일한 입력/출력 시퀀스를 시험 계열에 포함시켜 적용한 후, 구현 I/O FSM의 결과 상태를 확인하는 방법이다. I/O FSM으로 표현된 프로토콜 명세로부터 적합성 시험 계열 생성은 명세에 나타나 있는 각 천이에 도착 상태의 UIO 시퀀스를 Concatenation하여 생성하는데 다음과 같은 식 (3)으로 나타낼 수 있다.

$$Ri \cdot \text{short-path}(S0\text{-}Si) \cdot Tij@UIO(Sj) \qquad (3)$$

식 (3)에서 Ri는 시험대상을 초기화 상태로 보내는 심볼이며, short-path(S0-Si)는 초기상태 S0에서 해당 시험천이에 대한 시작 상태까지 최단 경로, Tij는 시험되어야 할, 즉 명세 I/O FSM의 상태 Si에서 Sj로 보내는 천이를 나타내고, UIO(Sj)는 도착 상태의 UIO 시퀀스를 나타낸다. 또 @는 Concatenation 심볼이다. 여기서 UIO 시퀀스는 시험되는 천이에 의해 도착된 상태가 명세에서 원하는 올바른 Sj인가를 시험하는데 사용된다.

표 1 Type 1 상태별 UIO 시퀀스

상태	UIO 시퀀스
IDLE(S0)	CTC통신요청/초기polling.send, △S=0 (A)
ACK AWAITED(S1)	AllState정보 or ACK/Timer=0 (E)
POLLING DETECTED(S2)	Error detected or △R>MR/NAK (C)
SYSTEM INITIALIZED(S3)	-/제어MSG.send, △S=0, Times=0 (I)
RESP AWAITED(S4)	ACK 수신/ - (S)
MSG DETECTED(S5)	Error detected or △R>MR/NAK, Times+1 (Q)
ABNORMAL(S6)	-/Polling전송 (N)
RESTORED(S7)	-/EIS의 모든 상태정보 (O)

간결성을 위해 프로토콜 Type 1의 I/O FSM 모델(그림 2)의 각 천이에 대해 UIC 시퀀스 생성 알고리즘을 적용하여 각 상태별 UIO 시퀀스를 구하면 표 1과 같다. UIO 시퀀스를 사용하여 I/O FSM 모델로부터의 시험 계열 자동 생성을 위한 여러 가지 방법들이 제안되었다[9].

본 논문에서는 구현 I/O FSM 내에 명세 I/O FSM에 명세된 각 천이가 존재하는지를 시험하기 위해 명세 I/O FSM으로부터 단일 시험 계열을 생성하는 방법인 최적화

기술을 사용한다. 표 1과 식 (3)을 적용하면 표 2와 같은 프로토콜 Type 1에 대한 적합성 시험 계열이 생성된다. () 내에 표현된 상태는 하나 이상의 천이에 대한 구별로서 출발상태와 도착상태를 나타낸다.

표 2 Type 1에 대한 적합성 시험 계열 생성

천이		천이에 대한 시험열	천이	천이에 대한 시험열
A		Ri·A·E	J	Ri·A·E·J·S
B		Ri·A·B·A	K	Ri·A·E·K·S
C		Ri·D·C·A	L	Ri·A·E·I·L·I
D		Ri·D·C	M	Ri·A·E·M·N
D		Ri·A·E·I	N	Ri·A·E·M·N·I
F	(S2→S3)	Ri·D·F·I	O	Ri·D·F·P·O·I
	(S5→S0)	Ri·D·F·R·F·A	P	Ri·D·F·P·O
G		Ri·A·E·G·A	Q	Ri·D·F·R·Q·I
H		Ri·A·E·H·A	R	Ri·D·F·R·Q
I		Ri·A·E·I·S	S	Ri·A·E·I·S·A

결과적으로 위에 나타낸 적합성 시험 계열의 입력 부분을 Type 1에 대한 구현이 받아들이고 출력 부분을 생성하여 명세의 출력과 같은지를 판단하면 Type 1의 구현이 Type 1의 명세에 대해 정확하게 구현되었다는 적합성 시험결과를 확인할 수 있다.

7. 결론 및 향후 연구 추진사항

본 논문에서는 LTS로 명세화된 철도 신호제어용 프로토콜 Type 1 모델을 정형기법으로 명세화하고 안전성 및 필연성 특성을 대수적 명세 기법인 modal mu-calculus를 사용하여 검정하였으며, 검정되어진 프로토콜 명세로부터 UIO방법에 의한 적합성 시험 계열 생성 방법을 제시하였다. 그 결과 해당 프로토콜 Type1은 안전성과 필연성을 모두 만족하는 규격임이 검정되었다.

본 논문에서 제시된 검정 방법과 시험 계열 생성 방법은 자연어에 근거한 수동적 검정과 시험 계열 생성에 비해 보다 신뢰성 있는 프로토콜을 개발하는데 사용되어질 것이고, 프로토콜 개발에 소요되는 비용과 시간을 최소화 시킬 것이다.

향후 연구사항으로는 프로토콜 검정 및 적합성 시험 방법을 실제 프로토콜의 정확성을 분석하는데 사용할 수 있도록 프로토콜 검정기와 시험기를 Windows 환경하의 GUI기능에 의해 Window상에 구현되도록 하는 것이다.

[참 고 문 헌]

[1] D.schwabe, "Formal Techniques for the Specification and Verification of Protocol", Ph.D Thesis, Univ. of California Los Angeles, Apr., 1981.

[2] J. G. Hwang, J. H. Lee, "A New Data Link Protocol for Railway Signaling Systems", KIEE Int'l Trans. on EMECS, Vol.3-8, No.4, pp.195-201, 2003.

[3] ISO/IEC 9646-1, "Information Technology-open Systems Interconnection-Conformance testing methodology and framework-Part1": General concepts , 1994

[4] E. M. Clarke, E. A. Emerson and A. P. Sistla, "Automatic Verification of Finite State Concurrent Systems Using Temporal Logic Specifications", ACM TOPLAS 8(2), 1986.

[5] P. V. Koppol and K. C. Tai, "Conformance Testing of Protocol specification as Labeled Transition system", International Workshop on Protocol Test System, IWPTS95, pp.143-158, Evry, France, 1995.

[6] Kenneth L. McMillan, "Symbolic model checking", Kluwer Acacenic Publishers, Model checking, 1996.

[7] R. Cleaveland, B. Steffen, "A Linear-Time Model-Checking Algorithm for the Alternation-Free Modal Mu-Calculus", Formal Methods in System Design 2(2): pp.121-147, 1993

[8] Sung-Un Kim, "INAP protocol conformance Testing", IEEE-AIN97, corolado Springs, USA, May, 1997.

[9] Z.Kohavi, "Switching and Finite Automata Theory", New York, McGraw-Hill, 1978