

재사용성을 고려한
항공기 인터콤 오디오 라우팅 처리방안 연구이승목^{1,†}¹Hanwha Systems

A Study on the Audio Routing Processing for Aircraft Intercom Considering Reusability

Seungmok, Lee^{1,†}¹Hanwha Systems

Abstract

The ICS, Intercom is the equipment which mixes and distributes the audio signal from other LRUs and plays the Voice Messages. Henceforth, it is of immense contributory importance to the pilots. Especially, the audio routing, which controls On/Off mode of each audio channel, is significant in executing a pilots' mission. But the audio routing process is quite complicated as it has the interface combination of many control signals. Underthecondition, the exceptional handling becomes difficult, which decreases maintainability and productivity. In the present work, to prevent such a situation, the author suggests amethodology,whichwillhavealower impact when the software is changed and provides high maintainability and productivity for audio routing processing.

초 록

항공기 인터콤 장비는 각종 LRU가 송신한 오디오를 혼합, 분배하고 상황인지용 메시지 재생을 통해 조종사의 원활한 임무 수행에 큰 역할을 담당하는 장비이다. 특히, 수신되는 오디오를 혼합/분배하는 오디오 라우팅의 경우에는 수신되는 오디오 채널에 대해 On/Off 제어를 하고, 연동 LRU에 오디오를 송신하여 임무에 대한 상황전파 및 공유를 통해 임무 수행에 매우 중요한 기능이다. 이러한 오디오 라우팅 처리는 다양한 연동 신호를 수반하고 있어 다양한 조합이 발생해 이에 대한 예외처리가 복잡해지므로 응집도를 낮고 결합도를 높여 유지보수성과 재사용성을 낮춘다. 이를 방지하고자 소프트웨어 변경 시 영향을 최소화하고 재사용성과 유지보수성을 높인 항공기 인터콤용 오디오 라우팅을 효율적으로 처리하는 방안을 제시한다.

Key Words : ICS(인터콤), Audio routing(오디오 라우팅), Maintenance(유지보수), Reusability(재사용성), Coupling(결합도), Strength(강도)

1. 서 론

항공기 인터콤 장비는 각종 LRU에서 수신되는 오디오를 조종사에 또는 동승자에게 송신해 주고, 인터콤과 연동되는 타 장비 간 오디오 신호 및 각종 제어 신호를 송수신하는 장비이다.

Fig. 1의 KUH의 인터콤 시스템 인터페이스[1] 일부를 예를 들면, 각종 헤드셋과 FM1 또는 FM2, AM1 또는 AM2 장비는 PTT(Push To Talk) 신호와 연동해서 인터콤이 오디오 송수신을 제어 할 수 있다. 이렇게 입력된 오디오 신호에 대해서 오디오 출력 채널로 적절히 분배 혹은 제어하는 기능을 오디오 라우팅(Audio Routing)이라고 정의하며 항공기마다 연동되는 장비들이 항공기의 운영개념 및 목적에 따라 각각 다르게 탑재 되기 때문에

Received: Sep. 06, 2017 Revised: Nov. 13, 2017 Accepted: Nov. 16, 2017

† Corresponding Author

Tel: +82-031-8020-7119, E-mail: seungmok23.lee@hanwha.com

© The Society for Aerospace System Engineering

오디오 라우팅은 항공기에 따라 다르게 설정이 된다.

따라서 사전 정의된 오디오 라우팅의 정상 동작이 항공기 조종사 및 동승자에게 정확한 상황을 인지 및 전파에 중요한 영향을 끼친다.

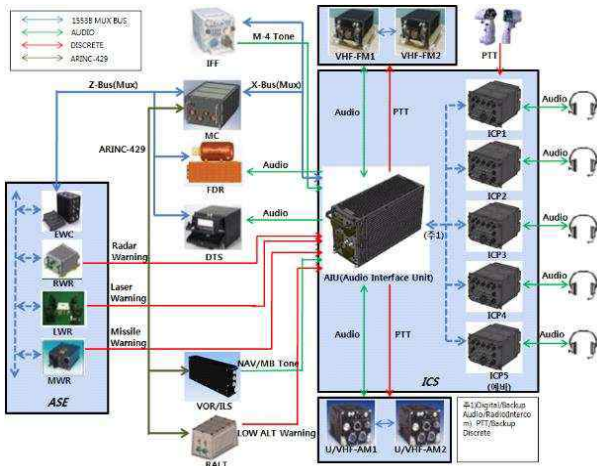


Fig. 1. System Interface diagram of ICS for KUH[1]

하지만 이러한 오디오 라우팅의 동작 및 연동은 다양한 이산(Discrete)신호 및 각종 시리얼 신호를 통한 명령을 통해서 구현되기 때문에 설계 및 구현이 복잡하고 그에 따른 예외처리가 많이 발생된다. 그로 인해 코드 복잡성은 증가하게 되며 강도(Strength)는 낮아지고 결합도(Coupling)는 높아지므로 재사용성은 현저하게 떨어진다. 이렇듯 재사용성이 떨어지면 생산성 저하 및 설계 및 구현 노력이 증가될 뿐만 아니라 유지보수성까지 저하 시키는 결과를 초래한다.

본 논문은 이러한 오디오 라우팅 처리에 대해 예외처리를 최소화하고, 반복적이고 규칙화된 처리로써 재사용성과 유지보수성을 높이고 소프트웨어 변경 시 영향을 최소화한, 항공기 인터콤용 오디오 라우팅을 효율적으로 처리하는 방안을 제시한다. 2장에서는 소프트웨어 재사용성을 높이는 방법을 고찰한 후, 인터콤용 소프트웨어 환경에서 재사용성을 높이는 방법을 확인 및 선정 한다. 3장에서는 2장에서 소프트웨어의 재사용성 향상을 위해 선정한 방법을 이용 설계 및 구현하며 4장에서는 설계 및 구현한 결과를 정량적인 평가를 통해 적용 결과를 검증한 후, 5장에서 결론을 맺는다.

2. 재사용성을 높이기 위한 관련 연구

2.1 배경지식

소프트웨어 재사용은 소프트웨어를 개발하거나 유지보수를 할 때 기존에 사용된 부품을 일부 수정하거나 또는 완전히 다시 사용할 수 있는 개념을 의미한다 [2]. 소프트웨어를 재사용함으로써 얻을 수 있는 이점은 고품질의 소프트웨어를 제작할 수 있고, 개발기간을 단축시켜 생산성을 향상시키며 소프트웨어 개발비용 절감 및 개발인원을 감소시킬 수 있다[3][4]. 따라서 소프트웨어 재사용 기술은 오래 전부터 소프트웨어 개발의 생산성과 품질을 향상시키기 위한 대안으로 제시되어 왔다[5][6][7].

소프트웨어 재사용성을 위해서 적용할 수 있는 기법은 Table 1과 같은 기법이 많이 사용이 된다.

Table 1 The Key method for Software Reusability

적용 시점	재사용성을 위한 주요기법
소프트웨어 개발 전 단계	객체지향방법
	Component Based Development (CBD)
	모듈화
설계 및 구현 단계	디자인패턴

하지만 이러한 소프트웨어 재사용 기법은 항공용 인터콤의 운영환경으로 인해 제약적이다. 인터콤 운영환경은 기본적으로 전력 및 하드웨어의 비용, 장비의 무게는 제약사항이자 요구사항이다. 하지만 통합적인 고성능 연산이나 독립된 다양한 기능을 위한 IMA(Integrated Modular Architecture) 구조는 요구되지 않는다. 따라서 SBC(Single Based Computer)와 같은 고성능의 시스템이 아닌 경량의 임베디드 시스템으로 구성되어야 한다. Table 2와 같이 인터콤 운영의 자원적인 요건을 분석해 볼 때 임베디드 시스템의 구성과 높은 응답성 및 저전력, 저비용, 경량화를 위해서는 실행파일의 크기가 작고, 간단하며, 제어가 빠른 C언어가 필수적이다.

인터콤에 적용되는 C언어는 기본적으로 객체지향 언

어가 아닌 절차지향 언어이므로 Table 3과 같이 주요 재사용 기법 중 C언어 적용 여부를 식별하였다.

Table 2 The analysis of Intercom Operation environment

자원 요건	요구 자원
CPU 사양	중/저사양
응답성	실시간 응답성
전력	저전력
하드웨어 비용	저비용
하드웨어 무게	경량

Table 3 The applicability of the main method for S/W reuse in C Language

재사용성을 위한 주요기법	C언어 적용 여부
객체지향방법	X
Component Based Development (CBD)	O
디자인패턴	X
모듈화	O

Table 3에 명시한대로 객체지향방법은 인터콥 소프트웨어 설계에 적용할 수 없으며 디자인패턴의 경우에도 기본적인 사상은 객체지향기법에서 출발하므로 C언어에서 적용이 어렵다. 반면 CBD와 모듈화는 객체지향, 절차지향과 독립적으로 기능관점의 재사용성 기법이므로 언어와 무관하게 범용적으로 적용할 수 있다. 그러나 CBD의 경우에는 컴포넌트(Component)에 대한 설계 및 구현과 저장소(Repository) 관리 등의 소프트웨어의 거시적인 관점으로 접근하기 때문에 이번 연구에는 개발 및 구현에 즉시 반영 할 수 있는, 모듈화를 이용한 기법을 선정, 적용하고자 한다.

2.2 모듈화를 통한 소프트웨어 재사용기법

모듈화는 프로그램을 분해하고 추상화 하여 소프트웨어의 성능을 향상시키거나, 프로그램의 시험, 통합, 재사용을 용이하게 하는 설계 및 구현 기법을 의미한다.

모듈화는 일종의 추상화를 활용해 단일 기능을 수행

하는 것인데 이를 통해서 모듈을 모듈의 독립성을 높일 수 있어서 타 기능과의 영향성을 최소화 한다.

대표적인 모듈화 기법은 함수화, 컴포넌트화, 전처리 기 등이 있고 이중 함수화는 가장 많이 사용되는 모듈화 기법이다. 최근에 Inline 기법도 많이 사용되는데 이는 실시간성이 요구되는 Real Time 환경의 임베디드에서 많이 사용되는 기법이다.

모듈의 독립성은 강도(Strength)와 결합도(Coupling)의 두 질적 기준으로 측정한다[8]. 모듈의 강도는 내부 요소끼리 어떻게 관련되어 있는지 알 수 있는 척도이다[9]. Table 4는 모듈의 강도를 표현하였는데 총 7개의 강도가 존재하며 Table내 번호 숫자가 작을수록 모듈의 강도가 강하다.

Table 4 The Module Strength[8]

번호	모듈 강도	설명
1	정보적 강도	특정 데이터 구조를 다루는 기능을 모듈화
2	기능적 강도	모듈이 단일 기능을 수행하도록 모듈화
3	통신적 강도	기능과 기능사이 취급하는 데이터를 고려한 모듈화
4	절차적 강도	업무처리 흐름의 차원에서 여러 기능을 하나의 모듈로 처리
5	시간적 강도	처리 시간대 등 시간적으로 연관성으로 여러 기능을 하나의 모듈로 처리
6	논리적 강도	논리적으로 연관이 있는 여러 기능으로 하나의 모듈을 구성
7	동시적 강도	상호 관련성이 적은 복수 개의 기능을 수행

모듈의 강도(Strength)와 더불어 결합도(Coupling)도 모듈화 특성을 위한 중요한 개념이다. 결합도는 소프트웨어 시스템 컴포넌트간 상호의존성의 정도로 정의하며 강도와 마찬가지로 Myers classification에 기반한 6개의 레벨이 있다[10].

Table 5는 Fenton과 Melton에 의한 수정된 Myers

Coupling Levels을 제시해주고 있다[11]

Table 5 Fenton and Melton Modified Definition for Myers Coupling Levels[12]

Type	Definition
Contents	x 는 y 내부를 참조함
Common	x 와 y 는 전역변수를 참조
Control	x 는 y 에게 제어파라미터를 전달
Stamp	x 는 y 에게 기록을 위한 변수를 전달
Data	x 와 y 는 파라미터로 통신
No Coupling	x 와 y 는 비종속적

※ x 와 y 는 컴포넌트(Component)를 의미

결합도의 6개 레벨에서 레벨은 모듈간 연관성 정도를 표현해 주며 No Coupling에 가까울수록 결합도는 낮아진다. 하지만 강도(Strength)와 달리 결합도(Coupling)는 낮을수록 모듈성이 높다고 판단한다.

3. 재사용성을 고려한 오디오 라우팅 구현

3.1 정형화된 테이블을 활용한 재사용 기법 구현

데이터 관점에서 재사용을 높이기 위해 별도의 정형화된 오디오 라우팅 테이블을 이용한다. 오디오 라우팅 테이블은 오디오 입력이 원하는 오디오 출력으로 라우팅이 되기 위한 제어신호의 조합을 정의한 표이다. 이러한 오디오 라우팅 테이블은 논리회로와 같이 제어신호의 High 상태와 Low 상태 그리고 무관(Don't care)상태의 조합으로 이루어진다. 이러한 오디오 라우팅 테이블은 항공기 시스템에 연동되는 제어신호의 종류와 운영개념 그리고 요구사항에 따라 다양하게 정의될 수 있다. 오디오 라우팅 테이블을 통해서 모듈의 강도 중 정보적 강도 즉, 데이터 구조를 다루는 기능을 모듈화 한다.

Index	Audio Input	Audio Output	Discrete 1	Discrete 2	Discrete 3	Discrete 4	...
1	Audio In 1	Audio Out 1	H	L	X	X	...
2	Audio In 2	Audio Out 2	L	H	X	X	...
...

Audio In/Out 정의

제어신호 정의

Fig. 2 The Configuration of Audio Routing Table

Figure 2에서 1번 인덱스의 오디오 라우팅을 살펴보면 Audio 1이 Audio 2로 라우팅을 활성화 시켜주기 위해서는 Discrete 1 신호는 High, Discrete 2 신호는 Low를 유지해야 하고, Discrete 3과 Discrete 4은 어떤 상태이든 제어에 무관한 라우팅으로 해석 할 수 있다.

Figure 3의 KUH의 인터콤 시스템 인터페이스를 예를 들어 오디오 라우팅을 정의해 본다면 KUH 시스템 인터페이스는 조종사 혹은 부조종사 마이크에서 FM1과 FM2 장비로 음성을 전달 할 수 있다. 조종사와 부조종사는 FM1, FM2 각각의 전용 PTT(Push To Talk) 스위치가 있다고 가정할 경우 Fig. 3와 같은 구성으로 오디오 라우팅 테이블을 구성할 수 있다.

오디오 라우팅 테이블 1번 인덱스처럼 조종사가 FM1 라디오를 통해 외부에 음성을 전달하고자 할 때, 오디오 라우팅 테이블에 정의된 제어신호 상태를 살펴보면 조종석 FM1 PTT 상태를 High로 조종석 FM2 PTT 상태를 Low로 유지하도록 한다.

Index	Audio Input	Audio Output	조종석 FM1 PTT	조종석 FM2 PTT	부조종석 FM1 PTT	부조종석 FM2 PTT
1	조종석 마이크	FM1	H	L	X	X
2	조종석 마이크	FM2	L	H	X	X
3	부조종석 마이크	FM1	X	X	H	L
4	부조종석 마이크	FM2	X	X	L	H

Audio In/Out 정의

제어신호 정의

Fig. 3 The Audio Routing example from KUH interface example (1)

부조종석의 FM1, FM2 PTT 상태는 무관(Don't Care)상태이므로 어떤 상태이든 제한적이지 않다. 무관상태가 존재하므로 오디오 라우팅 동작은 인덱스 중 반드시 한 개의 경우만 제어되는 것이 아니라 중복으로 제어가 될 수 있다. 가령 조종석 FM1 PTT를 High, 조종석 FM2 PTT는 Low, 부조종석의 FM1 PTT를 Low, FM2 PTT를 High로 유지시킨다고 한다

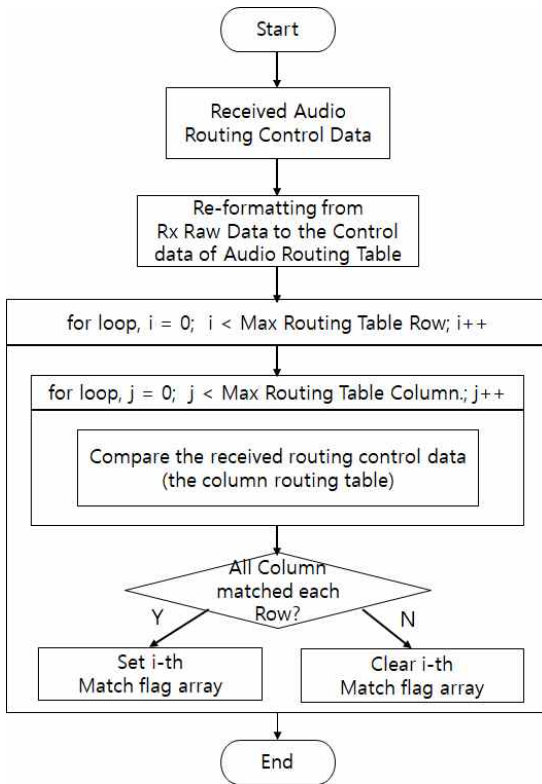


Fig. 6 The Audio routing flow chart to APPLY Software reusability.

```

1  #define COL_MAX    N //The Total Number of Routing control data
2  procedure Non_Reusable_Routing
3  (DiscIn : data structure
4   of Routing control data);
5  var HexVal : integer; //DiscIn converts to this variable
6  MatchingFlag : int_array;
7
8  begin
9  HexVal := (integer)DiscIn;
10 /*
11  In worst case, the number of switch case is
12  2 to the COL_MAX-th power(2^COL_MAX)
13  */
14  switch(HexVal) begin
15  case 0:
16  /* - Performe Routing Control applicable
17   for control data 0
18   - Example : Routing Case 0, 1
19   -> MatchingFlag[0] = MatchingFlag[1] = 1;
20  */
21  break;
22
23  case 1:
24  /* - Performe Routing Control applicable
25   for control data 1
26   - Example : Routing Case 3
27   -> MatchingFlag[3] = 1;
28  */
29  break;
30
31  case 2:
32  /* - Performe Routing Control applicable
33   for control data 2
34   - Example : Routing Case 7, 9
35   -> MatchingFlag[7] = MatchingFlag[9] = 1;
36  */
37  break;
38
39  case 3:
40  /* - Performe Routing Control applicable
41   for control data 3
42   - Example : Routing Case 10
43   -> MatchingFlag[10] = 1;
44  */
45  break;
46
47  //Omitted further cases
48
49  case (MAX-1):
50  /* - Performe Routing Control applicable
51   for control data (MAX-1)
52   - Example : Routing Case 19, 20
53   -> MatchingFlag[19] = MatchingFlag[20] = 1;
54  */
55  break;
56  end; //The end of switch
57  end
  
```

Fig. 7 The pseudo code NOT to APPLY Software reusability.

```

1  #define ROW_MAX    M //The Total Number of Routing cases
2  #define COL_MAX    N //The Total Number of Routing control data
3  procedure Reusable_Routing
4  (DiscIn : data structure
5   of Routing control data);
6  var current : char pointer //H, L, X combination for Control Data
7
8  begin
9  current := (char pointer)DiscIn;
10 CompareTable(current); //Refer to bottom procedure
11 end;
12
13 procedure CompareTable
14 (RoutingTable : data structure
15 of Pre-Defined Routing Table;
16 RouteControlData : char_array);
17
18 var row : integer; //Routing cases or index
19 var column : integer; //Routing Control data index
20 MatchingFlag : int_array;
21
22 begin
23 for row := 0 to ROW_MAX do begin
24 MatchingFlag[row] := 1;
25 for column := 0 to COL_MAX do begin
26 if(RoutingTable[row][column]
27 != RouteControlData[row][column])
28 MatchingFlag[row] := 0;
29 end; //The end of column for
30 end; //The end of row for
31 end;
  
```

Fig. 8 The pseudo code to APPLY Software reusability.

디오 라우팅 구성의 제어신호를 의미한다. 즉, 오디오 라우팅 테이블에서 각 인덱스 별 제어신호를 반복 제어문을 통해 비교한 다음 일치하면 동일한 인덱스의 Matching Flag 배열을 Set하고 불일치면 동일한 인덱스의 Matchg Flag 배열을 Clear 시킨다.

의사코드로 구현한 후 모듈화를 적용의 효과는 더욱 명백해 진다. 단순히 선택 제어문을 이용해서 코드를 작성한 Non_Reusable_Routing은 오디오 라우팅 케이스의 수만개의 case 문 숫자가 생성하게 된다. 하지만 오디오 라우팅 테이블을 활용한 Reusable_Routing은 CompareTable() 함수화 시켰고, 이 함수는 단순 반복 순환문을 통해 간결하게 완성이 된다.

4. 재사용성 기법 적용 결과 평가

4.1 재사용성 기법에 대한 평가 방법

3장에서 구현한 재사용성 기법에 대한 평가는 Table 6과 같이 기술적인 측면과 비용적인 측면으로 수행했다. 기술적인 측면은 모듈화에 대한 정량적인 수치인 결합도 Metric을 활용해서 재사용성이 미반영된 처리방식과 재사용성이 반영된 처리방식간 비교를 통해서 평가를 실시하였다. 비용적인 측면은 새롭게 개발되는 비용과 재사용을 통한 비용간 비교를 위해 산출하는 상대비용과 이를 기반으로 하는 상대생산성을 산출해서 평가를 하였다.

Table 6 The assesment for Reusability method

구분	평가 분류	평가 방법
기술측면	모듈화 수치 비교	결합도 Metric
비용측면	비용 산출	상대비용 계산
	생산성 수치 활용	상대생산성 계산

4.2 결합도 Metric을 활용한 평가

모듈화 적용 결과를 확인할 수 있는 방법은 결합도와 강도이다. 이중 결합도는 수학적인 강도 측정 모델을 이용해 산출이 가능하다. 결합도 측정 모델은 Fenton and Melton Software Coupling Metric[12]을 활용한다. Fenton and Melton Software Metric의 산출은 (1)과 같다.

$$C(x,y) = i + \frac{n}{n+1} \dots (1)$$

여기서 x, y 는 결합관계에 있는 컴퍼넌트를 의미하며, i 는 Table 7의 결합도 레벨, n 은 x, y 간의 연결수를 의미한다. 하지만 n 이 충분히 크다고 가정을 할 경우 $C(x,y)$ 는 (2)과 같이 유도된다.

$$C_{approximate} \cong i + \frac{n}{n} = i \dots (2)$$

따라서 재사용의 반영 전, 후의 결합도 Metric은 결합도 레벨에 의해 결정된다. 재사용성이 미반영된 결합도 Metric을 $C_{NoReuse}$, 재사용성이 반영된 결합도 Metric을 C_{Reuse} 이라고 할 때, 두 수치의 값 비교로 재사용성을 평가한다.

Table 7 The Coupling Levels Value[12]

Type	Level
Contents	5
Common	4
Control	3
Stamp	2
Data	1
No Coupling	0

Fig. 7와 Fig. 8의 의사코드를 이용해서 Table 7의 Coupling Level을 분류할 수 있는데 먼저, 결합도 관계는 재사용성을 반영 전과 후의 오디오 라우팅 처리 모두 DiscIn이라는 입력 변수에 의해 MatchingFlag라는 출력 변수가 결정이 되므로 두 관계가 결합 관계가 성립이 된다. 이 두 변수에 대해서 Coupling Level을 확인하면, 재사용성이 반영되기 전의 오디오 라우팅 처리는 DiscIn이 제어 파라미터로서 전달되므로 이는 Table 7에 따라 Control Type에 해당되므로 Coupling Level $C_{NoReuse}$ 는 3이 된다.

반면, 재사용성이 반영된 오디오 라우팅 처리는 DiscIn이 단순 데이터 파라미터로 전달되므로 이는 Table 7에 명시된 Data Type에 해당되므로 Coupling Level C_{Reuse} 는 1이 된다.

따라서 결합도 Metric은 수치가 낮을수록 모듈화가 좋고 재사용성이 증가되므로 고안된 오디오 라우팅 처리는 재사용성에 효과적이었다고 볼 수 있다.

4.3 상대비용, 상대생산성을 활용한 평가

상대비용(Relative Cost, RC)과 상대생산성(Relative Productivity, RP)은 Bruce Barnes, Thomas Durek, John Gaffney, Arthur Pyster[13]가 정의한 (3)의 식으로 산출한다.

$$RC = (1 - R)1 + Rb = 1 + (b - 1)R \quad \dots (3)$$

$$RP = \frac{1}{RC} = \frac{1}{(b - 1)R + 1} \quad \dots (4)$$

여기에서 b 는 재사용 가능한 부분에서 소요되는 상대비용이며, R 은 프로젝트 내에서 재사용되는 코드의 비율이다.

Fig. 8의 재사용성이 반영된 오디오 라우팅의 의사코드를 참조로 Fig. 7과 비교해 볼 때 입력에 대한 타입 변환의 단순한 노력이 차가 되어 $b = 0$, $R = 1$ 에 가깝다. 하지만 일반적으로 재사용성을 위한 설계 및 구현 시, 인터페이스 노력이 일정량 추가되므로 이를 정성적으로 반영해 $b = 0.5$, $R = 0.5$ 로 설정하면 아래와 같은 결과가 산출 된다.

$$RC = 1 + (0.5 - 1) \times 0.5 = 0.75$$

$$RP = \frac{1}{RC} = \frac{1}{0.75} = 1.33$$

따라서 재사용성이 반영된 오디오 라우팅 처리 이용시 비용은 25% 감소되고 생산성은 33% 증가되는 결과를 도출할 수 있다.

5. 결 론

본 논문에서는 항공용 인터콤에서 중요한 기능 중에 하나인 오디오 라우팅 처리에 대해서 재사용성을 극대화하는 기법에 대해 연구하였다. 항공기용 인터콤이 가지는 운영 환경적 제약으로 인해서 재사용성은 자원적 혹은 개발 환경 제약성이 있었지만 데이터 측면의 정형화된 라우팅 테이블 사용과 개발방법론적인 모듈화 기법을 활용해 오디오 라우팅 처리를 고안 및 구현

했고, 그에 따른 연구 평가를 기술적인 관점과 비용적인 관점에서 정량적인 수치로 수행하였다. 수행결과 모듈화 관점에서 결합도 Metric 값이 낮아져 모듈성은 높아졌고 비용적인 절감과 생산효율성이 향상된 것을 확인하였다.

향후 제안된 기법을 통해 항공용 인터콤의 오디오 라우팅 처리를 공통화하고, 다양한 항공기 체계에 적용하도록 할 계획이다.

References

- [1] Kim, Y.M, et al., "A Study on Voice Communication Quality Improvement of Intercom System for KUH", *J. of The Korean Society for Aeronautical and Space Sciences*, 41(12), 1002-1010
- [2] Kang, H.M, et al., "The Analysis of the productivity using Software Reuse", *KIISE Autumn Congress'97*, 1997, pp. 379-388, 1997.
- [3] Arnold S.P and S.I.Stepoway, "The Reuse System : Cataloging and Retrieval of Reusable Software", *Proceeding of COMPCONS'87*, 1987, pp. 376-379.
- [4] Biggerstaff, T. and C.Richter, "Reusability Framework, Assessment and Direction", *IEEE Software*, Mar. 1987, pp. 41-49.
- [5] T.Biggerstaff, A.J. Perlis, "Software Reusability", Vol. 1: Concepts and Models, Addison-Wesley, 1989.
- [6] P.A.V.Hall, "Overview of Revers Engineering and Reuse Resarch", *Information and Software Technology* 34, pp. 239-249, 1992
- [7] C.W.Krueger, "Software Reuse", *ACM Computing Surveys*, Vol. 24, pp 131~183, 1992.
- [8] Lee, C.W, (2000), "A Study on Securities Application System using a Software Reuse", Master's Thesis, Soonchunhayng Univ., Cheonan, Rep.of Korea, 87 pages
- [9] Myers G. J ., *Composite/ Structured Design*, Van Nostrand Reinhold Company, 1978.
- [10] N. E. Fenton and S. L. Pfleeger, *Software Metrics: A Rigorous & Practical Approach*. 2nd Edition. Reading, 1997.

-
- [11] JARALLAH S. ALGHAMDI, "Measuring Software Coupling", *Proceedings of the 6th WSEAS Int. Conf. on Software Engineering, Parallel and Distributed Systems*, pp. 6-12, February 2007.
- [12] Norman Fenton and Austin Melton, "Deriving Structurally Based Software Measures," *J. System Software*, 1990, Vol. 12, pp 177-187.
- [13] Relative Productivity, RP Bruce Barnes, Thomas Durek, John Gaffney, Arthur Pyster "A Framework and Economic Foundation for Software Reuse", *Proceeding of workshop on Software Reusabiligy and Maintainability*, Oct. 1987