

캐빈 디스플레이 시스템을 위한 원격 제어 소프트웨어의 설계 및 구현

Design and Implementation of Remote Control Software for Cabin Display System

신재권 · 최지환 · 최낙민 · 김정열 · 김준형*

LIG Nex1 항공연구소

Jae-kwon Shin · Ji-hwan Choi · Nak-min Choi · Jeong-yeol Kim · Jun-hyoung Kim*

Aeronautical research institute of LIG Nex1, Daejeon, 34127, Korea

[요약]

최근 항공 산업 경쟁이 심화함에 따라 항공기 인테리어 산업도 첨단 IT (information technology) 요소를 접목하는 등 빠르게 발전하고 있다. 국내에서는 보잉사와 협력하여 항공기용 디스플레이 시스템인 CDS (cabin display system)를 개발 중이며, 본 논문에서는 CDS의 원격 제어 소프트웨어인 CDS RCS (cabin display system remote control software)의 설계 및 구현의 내용을 다룬다. CDS RCS는 ARINC-615A 표준을 응용하여 자체 개발한 항공기용 데이터로드 프로세스와, MQTT (message queuing telemetry transport) 프로토콜을 적용하여 개발하였다. 또한 CDS RCS는 기존의 CDS 제어 소프트웨어와 달리 MQTT client 형태로 설계되어 다른 MQTT client 장치들과 독립적으로 운용할 수 있다는 장점이 있으며, 개인 단말기에 쉽게 설치할 수 있는 안드로이드 기반으로 개발되어 승무원들의 편의성이 크게 향상될 것으로 기대된다.

[Abstract]

In recent years, the aircraft interior industry has been rapidly evolving, incorporating advanced information technology (IT) elements as competition in the aviation industry intensifies. In Korea, a display system for aircraft, called cabin display system (CDS), is being developed in collaboration with Boeing. This paper discusses the design and implementation of cabin display system remote control software (CDS RCS), a remote control software for the CDS. CDS RCS was developed by applying the aircraft data load process developed in-house using the ARINC-615A standard, and the communication protocol between devices applies the MQTT protocol. Unlike existing CDS control software, CDS RCS is designed as an MQTT client and can operate independently of other MQTT clients. It is also developed for android-based devices, which can be easily installed on personal terminals, and is expected to greatly enhance the convenience of flight attendants.

Key word : Smart cabin, Integrated display processing module, Message queuing telemetry transport, Javascript object notation.

<http://dx.doi.org/10.12673/jant.2024.28.5.698>



This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Received 7 October 2024; Revised 26 October 2024

Accepted (Publication) 29 October 2024 (31 October 2024)

*Corresponding Author; Jun-hyoung Kim

Tel: *** - **** - ****

E-mail: junhyoung.kim@lignex1.com

I. 서론

최근 항공사들은 서비스 품질을 향상시키고 경쟁 우위를 확보하기 위해 고객 경험과 인터페이스에 투자하고 있다. 이러한 추세는 팬데믹, 엔데믹을 거치며 여행 수요가 증가함에 따라 더욱 가속화되었다. 이러한 흐름에 맞춰 LIG넥스원은 보잉사와 합작하여 CDS (cabin display system)을 개발하고 있으며, 항공기용 미디어 플레이어인 IDPM (integrated display processing module)과 IDPM 원격 제어용 태블릿 앱을 개발하고 있다[1]-[7].

본 논문은 원격으로 IDPM을 제어하는 안드로이드 앱인 CDS RCS(cabin display system remote control software)의 설계 및 구현에 관한 것이다. 이는 2023년 항공우주학회 춘계학술대회에서 발표된 “항공기 탑재 미디어 플레이어 원격 제어 소프트웨어 설계” 논문[6]의 내용을 확장 및 구체화한 것이다.

논문의 2장에서는 관련 연구로서 CDS에 대한 설명과 CDS 내부 통신 프로토콜인 MQTT(message queuing telemetry transport) 및 항공 장비용 데이터 로드 과정에 대해 설명한다. 3장에서는 CDS RCS의 인터페이스 및 데이터 관리 설계에 대해 설명하며, 4장에서는 CDS RCS 앱 개발 환경 및 구현 결과를 보인다. 마지막으로 5장에서는 결론 및 향후 진행될 연구에 대한 계획을 제시한다.

II. CDS의 개요 및 관련 기술

2-1 CDS

그림 1은 보잉사와 LIG넥스원이 협력하여 개발 중인 CDS를 나타내며, 이는 승객에게 향상된 경험을 제공하기 위해 기존 항공기 객실 인터페이스의 개선을 목표로 한다.

CDS는 flexible OLED(organic light-emitting diode) 패널 어셈블리, transparent OLED 패널 어셈블리, NAS(network attached storage), ethernet switch, WAP(wireless access point), 그리고 승무원용 태블릿으로 구성된다. flexible OLED 및 transparent OLED panel assembly는 OLED panel 조립체와 IDPM으로 구성된다. CDS RCS는 CDS 구성 중 태블릿에 탑재되는 소프트웨어 모듈로, 승무원이 안드로이드 기반의 개인 단말을 통해 원격으로 CDS를 제어할 수 있게 한다[2]-[7].

2-2 MQTT

MQTT는 broker, publisher, subscriber 간 통신 구조로 이루어져 client와 server 간 통신 구조인 UDP(user datagram protocol), TCP(transmission control protocol)와 구조 및 통신 방식이 다르다. MQTT는 필요한 패킷만 선택 수신할 수 있다는 특징이 있으며, 이러한 특징으로 인해 장치 간 통신을 체계적으로 관리할 수 있으며, 저전력 및 최소한의 패킷량으로 통신할 수 있다는 장점이 있어 IoT(internet of things), 항공기 캐빈용 메시지 통신

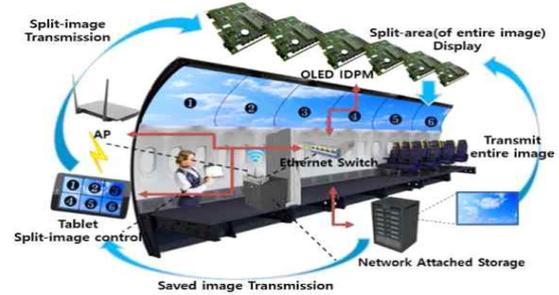


그림 1. cabin display system의 운용 개념
Fig. 1. operational concept of cabin display system

프로토콜로서 적합하며, CDS에서도 장치 간 통신 프로토콜로 MQTT를 채택하고 있다.

MQTT를 구성하는 요소 중 broker는 MQTT 네트워크에 연결된 장치 목록과 메시지 객체인 topic, 각 장치가 등록한 topic 구독 목록을 관리하는 server 역할을 한다.

MQTT의 publisher와 subscriber는 broker에 client로 연결되며 데이터 송신/수신 여부에 따라 하나의 장치가 두 역할을 동시에 수행할 수 있다. 장치가 메시지를 전송하기 위하여 topic을 발행하는 경우에는 publisher로, 구독할 topic을 broker에 등록하고 메시지를 수신할 경우에는 subscriber로 동작한다[8-11].

기존 CDS 제어 소프트웨어는 MQTT server로 설계되어 IDPM이 MQTT client로서 CDS 제어 소프트웨어와의 연결 상태에 의존했다[7]. 그러나 CDS RCS는 MQTT client로 설계되어 IDPM이 CDS RCS와 독립적으로 운영될 수 있도록 하였다.

2-3 항공 장비용 데이터 로드

CDS는 FTP(file transfer protocol) 및 JSON(javascript object notation) 기반 데이터로드 프로세스를 도입하여 ARINC-665 (aeronautical radio, incorporated-665) 및 ARINC-615A 표준에 비해 속도 및 확장성이 향상된 데이터 로딩을 제공한다. 이 새로운 접근 방식은 명령어 전송, 패키지 정보 교환, 파일 리스트 교환, 업/다운로드 수행과 같은 기존 프로세스를 따르지만, 대용량 파일 전송을 위해 FTP를 사용할 수 있도록 설계되었다. 또한, 패키지 정보 및 파일 리스트 교환에 JSON을 사용하여 ARINC-665에 비해 확장성을 높였다[7],[12],[13].

III. 캐빈 디스플레이 시스템 원격 제어 소프트웨어의 설계 및 구현

그림 2는 CDS RCS의 소프트웨어 구조 및 내/외부 통신 구조를 나타낸 그림이다. CDS RCS는 항공기 서버에 있는 MQTT broker와 통신하는 MQTT client로 동작하며, MQTT를 통해 IDPM과 통신한다. CDS RCS는 인터페이스 관리, 데이터 관리, 페이지 시현 관리 CSC(computer software component)로 구성되

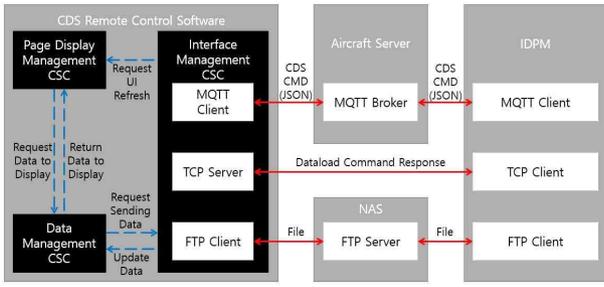


그림 2. CDS RCS SW 및 내/외부 인터페이스
 Fig. 2. CDS RCS SW and internal/external interface.

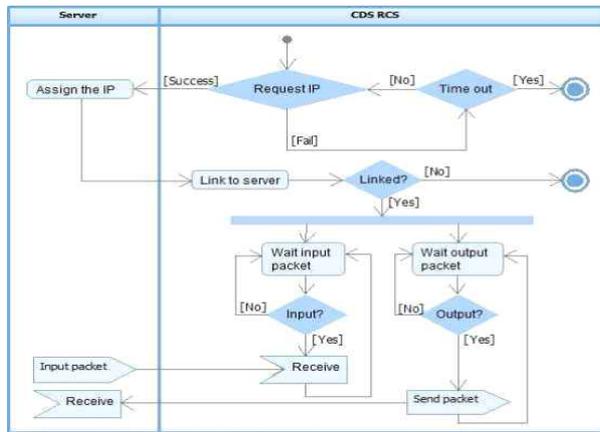


그림 3. 이더넷 통신 기능의 순서도
 Fig. 3. ethernet communication function flowchart.

어 있으며, GUI(graphical user interface)를 통해 사용자로부터 미디어 장치 제어 명령을 받아 IDPM에 전송하고, IDPM으로부터 응답 및 시스템 모니터링 데이터를 받아 내부 database 및 GUI를 업데이트한다. 또한 CDS RCS는 데이터 로드 컨트롤러 역할도 수행한다. TCP를 통해 IDPM과 데이터 로드 명령 및 상태 정보를 송수신하고, FTP를 통해 NAS 및 IDPM으로부터 파일 리스트를 제공받는다.

3-1 인터페이스 설계

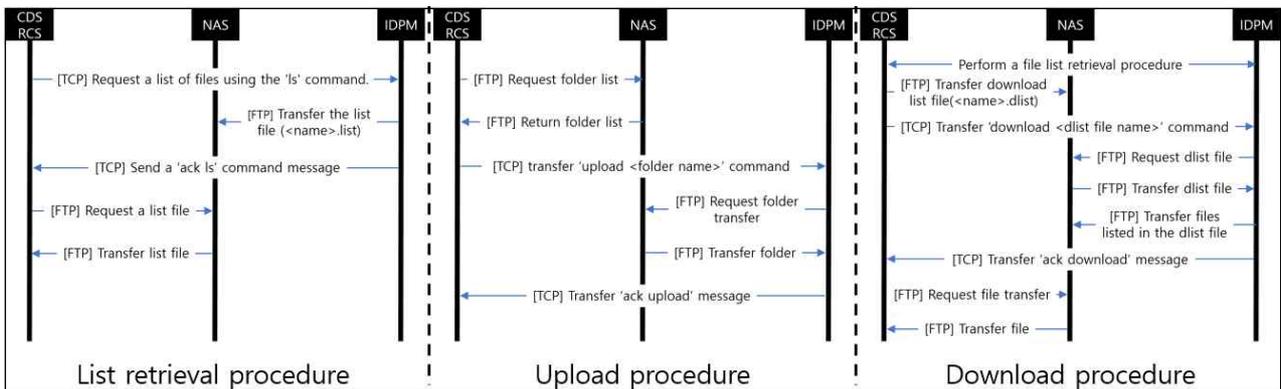


그림 4. 데이터로드(리스트 조회/업로드/다운로드)의 기능 수행 절차
 Fig. 4. procedure of data load(list/upload/download) function.

그림 3은 이더넷 통신 기능의 순서도를 나타낸 그림이다. 이더넷 통신 기능은 MQTT와 TCP, FTP 통신에 동일하게 적용되며, ethernet port를 초기화하고 DHCP (dynamic host configuration protocol) 서버에 IP (internet protocol)주소를 요청하는 초기화 기능과 CDS RCS에서 서버로 명령을 전송하고 요청한 데이터를 수신하는 패킷 송수신 기능을 담당한다.

3-2 데이터로드 기능 설계

데이터로드 기능은 "FTP와 JSON 을 활용한 대용량 미디어의 항공장비용 데이터 로드 프로세스" 논문[7]에서 설계된 데이터 로드 절차를 참고하여 서버의 역할을 분리해 안드로이드 앱으로 포팅하였으며, 조회/업/다운로드 과정은 그림 4와 같다.

1) 리스트 조회

리스트 조회는 사용자가 태블릿의 데이터 로드 페이지에서 refresh 버튼을 누르면 동작하며, CDS RCS가 NAS와 IDPM의 파일 리스트를 요청하고 화면에 파일 트리를 시현하는 기능이다. 그림 4의 리스트 조회 절차를 요약하면 다음과 같다.

- ① [TCP] CDS RCS에서 IDPM으로 "ls" 명령을 전송.
- ② [FTP] IDPM이 NAS로 메타 파일(list 파일)을 전송.
- ③ [TCP] IDPM이 CDS RCS로 ack 메시지를 전송.
- ④ [FTP] CDS RCS가 NAS로 list 파일을 요청.
- ⑤ [FTP] NAS에서 CDS RCS로 list 파일을 전송.

2) 업로드

업로드는 데이터로드 페이지에서 업로드 대상 폴더를 선택 후, upload 버튼을 누르면 동작하며, NAS에서 IDPM으로 폴더 단위의 패키지를 업로드하기 위한 기능이다. 그림 4의 업로드 절차를 요약하면 다음과 같다.

- ① [FTP] CDS RCS가 NAS에게 폴더 리스트를 요청.
- ② [FTP] NAS가 CDS RCS에게 폴더 리스트를 전송.
- ③ [TCP] CDS RCS가 NAS에게서 받은 폴더 리스트 중 업로드할 폴더를 선택하여 IDPM에 업로드 명령을 전송.

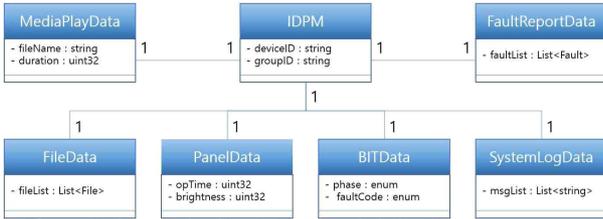


그림 5. 데이터 관리 기능의 클래스 다이어그램
Fig. 5. data management function class diagram.

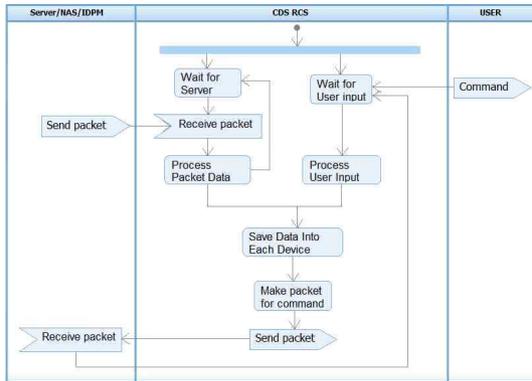


그림 6. 데이터 관리 기능의 순서도
Fig. 6. data management function sequence.

- ④ [FTP] IDPM이 NAS에게 폴더 전송을 요청.
- ⑤ [FTP] NAS가 IDPM에게 폴더를 전송.
- ⑥ [TCP] IDPM이 CDS RCS로 ack 메시지를 전송.

3) 다운로드

다운로드는 사용자가 태블릿의 데이터로드 페이지에서 대상 파일을 선택 후, download 버튼을 누르면 동작하며, IDPM에서 NAS로 파일을 다운로드하기 위한 기능이다. 그림 4의 다운로드 절차를 요약하면 다음과 같다.

- ① 사전에 리스트 조회 절차를 수행.
- ② [FTP] CDS RCS가 다운로드 파일 리스트 메타파일(dlist 파일)을 NAS로 업로드.
- ③ [TCP] CDS RCS가 IDPM에게 다운로드 명령을 전송.
- ④ [FTP] IDPM이 NAS에게 dlist 파일을 요청.
- ⑤ [FTP] NAS가 IDPM에게 dlist 파일을 전송.
- ⑥ [FTP] IDPM이 dlist 파일에 적힌 파일들을 NAS로 전송.
- ⑦ [TCP] IDPM에서 CDS RCS로 ack 메시지를 전송.
- ⑧ [FTP] CDS RCS가 NAS에게 파일 전송을 요청.
- ⑨ [FTP] NAS가 CDS RCS로 파일을 전송.

3-3 데이터 관리 설계

그림 5는 CDS RCS 내 데이터 관리 기능의 클래스 다이어그램이며, 미디어 재생 데이터, 파일 데이터, 패널 데이터,

표 1. CDS RCS 구현 환경

Table 1. CDS RCS implementation environment.

| Item | Specification | |
|------------------------------|------------------------------|------------------------------------|
| Development PC | CPU | Intel i5-9400 @2.90GHz |
| | RAM | 16GB |
| | OS | Windows 10 enterprise 64bit |
| Target device 1 (Galaxy S8+) | CPU | Samsung exynos 8895 @2.3GHz |
| | RAM | 6GB |
| | OS | Android 9.0 |
| Target device 2 (G Pad5) | CPU | Qualcomm snap dragon 821 @2.34 GHz |
| | RAM | 4GB |
| | OS | Android 9.0 |
| MQTT | mqttv5.client-1.2.5 | |
| Tool / language | Android studio 11.0.12/ java | |

BIT(built-in test) 데이터, 시스템 로그 데이터, 결함 보고서 데이터 등 6가지 주요 데이터 카테고리를 포함하고 있다. 이러한 데이터는 IDPM 클래스 내에서 관리되며 각 IDPM 장치별로 구성되어 있다.

그림 6은 데이터 관리 기능의 순서도를 보여주며, 서버로부터 전송받은 데이터는 각 IDPM 장치별로 저장 및 관리되고, 사용자로부터 입력받은 데이터는 처리 후에 MQTT 패킷에 실려 각 IDPM으로 전송된다.

IV. CDS RCS의 구현 환경 및 구현 결과

4-1 구현 환경

본 논문에서 설계한 CDS RCS를 구현하기 위한 환경은 다음의 표 1과 같다.

개발용 PC는 intel i5-9400 CPU와 16GB RAM을 갖추고 있으며, windows 10 enterprise 64bit를 운영체제로 사용하였다. 시험 대상 장치로는 확장성을 염두에 두고 samsung galaxy s8+와 g-pad 5를 사용하였다. galaxy s8+는 exynos 8895 프로세서, 6GB RAM, android 9.0 OS를 탑재하고 있으며, g-pad 5는 qualcomm snap dragon 821 프로세서, 4GB RAM, android 9.0 OS를 탑재하고 있다. 안드로이드 MQTT 패키지로 mqttv5.client-1.2.5가 사용되었고, 개발 도구는 android studio 11.0.12를 사용했으며, 개발 언어는 java를 사용하였다.

4-2 구현 결과

1) 초기 페이지

그림 7은 초기 페이지 화면을 나타낸 것이다. 초기 페이지는 CDS에 연결된 IDPM의 목록을 보여주며, 최초 실행 시 IDPM이 MQTT broker에 등록되는 순으로 번호가 매겨진다. IDPM이 CDS RCS와 정상적으로 데이터를 교환한 이후에 장비의 IP주

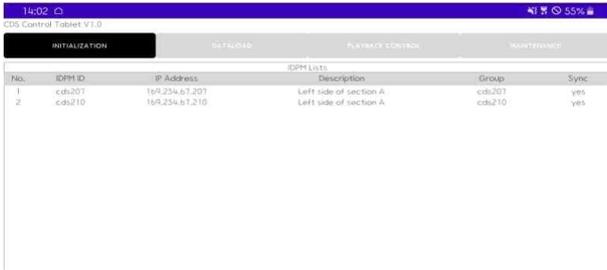


그림 7. 초기화 페이지
Fig. 7. initialization page.

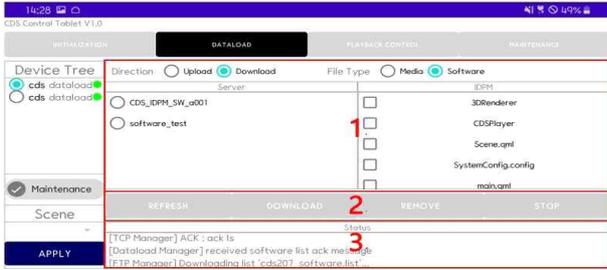


그림 8. 데이터로드 페이지
Fig. 8. dataload page.

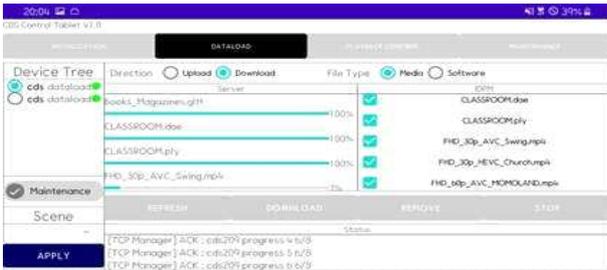


그림 9. 데이터로드가 진행 중일 때의 데이터로드 페이지
Fig. 9. dataload page while dataloading.

소, 설명, 그룹 정보가 출력되며, 동기화 여부를 출력한다. 만약 MQTT broker에는 등록되었지만 CDS RCS의 정보 요청에 응답하지 않은 장치는 IDPM ID(identification)만 표시되고 동기화 여부는 no로 표시되며, 그 외 다른 정보는 '-'로 표시된다.

2) 데이터로드 페이지

그림 8은 데이터 로드 페이지를 나타낸 그림이다. 그림 8에서 1번 영역은 데이터로드 옵션과 전송할 파일을 선택할 수 있는 창으로서 데이터로드 옵션으로 upload 혹은 download 옵션을 선택하여 전송 방향을 정하고 media 혹은 software 타입 중 하나를 선택할 수 있다. 2번 영역은 데이터로드를 제어하기 위한 버튼으로서 1번 영역에서 선택하는 전송 방향 옵션에 따라 버튼의 구성이 달라진다. 업로드 기능을 선택하면 파일 리스트 갱신/업로드/중지 버튼이 활성화되며, 다운로드 기능을 선택하면 파일 리스트 갱신, 다운로드, 삭제, 중지 버튼이 활성화된다.

그림 9는 데이터로드가 진행 중일 때의 화면을 나타낸 것이다.

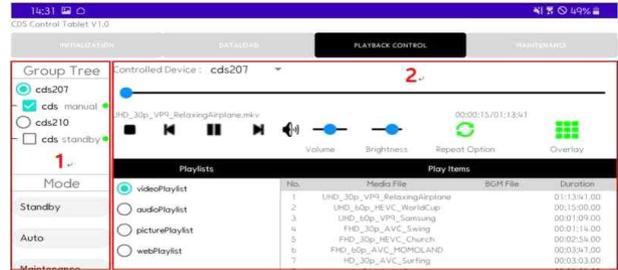


그림 10. 재생 제어 페이지
Fig. 10. playback control page.

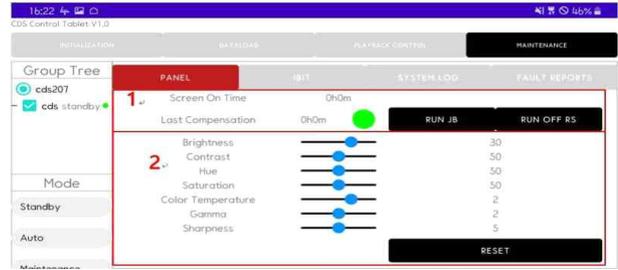


그림 11. 패널 설정 페이지
Fig. 11. panel setting page.

다운로드 시에는 그림 8의 1번 영역의 파일 선택 창 중 좌측 영역이 그림 9와 같이 다운로드 현황을 나타내는 게이지 창으로 변경되고, 업로드 시에는 다운로드 상황과는 반대로 파일 선택 창 중 우측 영역이 IDPM 장치 별 업로드 상황을 나타내는 게이지 창으로 변경된다. 업로드 혹은 다운로드가 완료되면 2초 뒤 그림 8과 같이 초기 데이터로드 화면으로 복귀한다.

3) 재생 제어 페이지

그림 10은 재생 제어 페이지를 보여주며, 이는 IDPM과 그룹의 미디어 재생을 제어할 수 있게 한다. 1번 영역의 상단부는 IDPM/그룹 선택 버튼 영역으로, 한 그룹 내에 있는 장치를 동시에 선택하거나 개별 선택할 수 있다. 각 IDPM의 현재 모드는 텍스트로 표시되며, 결합 여부는 신호등으로 나타낸다. 1번 영역의 하단부는 IDPM에 대기모드, 자동모드, 점검모드를 명령하는 버튼이다. 2번 영역은 미디어 재생을 제어하기 위한 창으로, 화면에 표시할 IDPM을 선택하고 구간탐색, 정지, 전후 탐색, 재생 및 일시정지를 명령할 수 있다. 또한 볼륨 조절, 밝기 조절, 반복 옵션 및 오버레이 출력 여부를 선택할 수 있도록 구성되어 있다. 2번 영역의 하단부는 플레이리스트를 선택할 수 있는 화면으로, 좌측에서 플레이리스트를 선택하면 우측 영역에서 재생 파일, BGM(background music), 재생 시간을 확인할 수 있다.

4) 설정 페이지

그림 11은 panel 설정 페이지를 실행한 화면을 나타낸 것이다. 패널 설정 페이지는 IDPM에 연결된 패널의 현재 상태와 세부

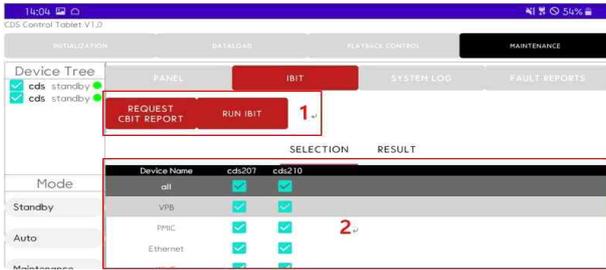


그림 12. BIT 제어 페이지 - BIT 항목 선택
 Fig. 12. BIT control page - BIT item selection.



그림 13. BIT 제어 페이지 - BIT 결과 시현
 Fig. 13. BIT control page - BIT result display.

옵션을 설정할 수 있는 창을 제공하고 타 페이지 이동 메뉴를 시현한다. 그림 11의 1번 영역에서는 패널의 누적 운용 시간 및 마지막 화소 보정 후 경과 시간을 확인할 수 있으며, 화소 보정 기능을 명령할 수 있다. 2번 영역은 패널 설정 창 영역으로 밝기, 대비, 색조 등 패널의 세부 설정 옵션을 제어하는 영역이다. 2번 영역 하단의 reset 버튼을 누르면 입력된 옵션을 초기화할 수 있다.

그림 12는 BIT 항목 선택 및 제어 페이지를 나타낸다. 그림 12의 1번 영역은 CBIT(continuous BIT) 결과 요청 버튼과 IBIT(initiated BIT) 수행 명령 버튼으로 구성되어 있으며, 2번 영역은 BIT 항목 선택 창으로서 IBIT 수행 시에 테스트할 항목을 IDPM 장치 별로 선택할 수 있다.

그림 13은 BIT 결과 페이지를 나타낸 것이다. 1번으로 표시한 영역이 IBIT 결과를 장치 및 BIT 종류별로 시현하는 영역이며, view report 버튼을 눌러 상세 BIT 결과를 확인할 수 있다.

그림 14는 시스템 로그 페이지를 실행한 화면을 나타낸 것이다. 시스템 로그 페이지는 화면 전환, 이벤트 발생, 결함 발생 등 시스템 전반적으로 발생하는 다양한 상황을 기록한, 시스템 로그를 보여준다. 시스템 로그 창을 통해 시간순으로 장치 ID, 중요도, 로그를 시현하는 것을 볼 수 있다.

그림 15는 결함 보고서 페이지를 실행한 화면을 나타낸 것이다. 결함 보고서 페이지는 결함이 발생했을 때의 상황을 기록한, 결함 보고서를 리스트로 보여준다. 결함 설정 시간 순으로 각 결함 별 결함 설정 시간, 결함 해제 시간, 결함 ID, 현 상태, 에러 코드, 결함 레벨, 결함 설명을 나타낸다.

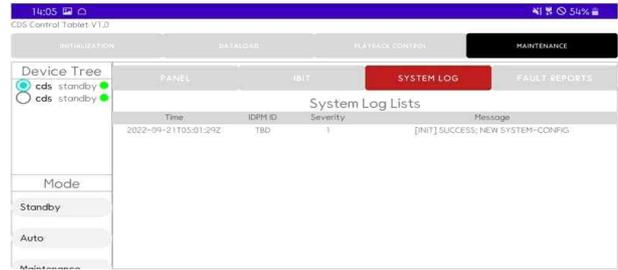


그림 14. 시스템 로그 페이지
 Fig. 14. system log page.

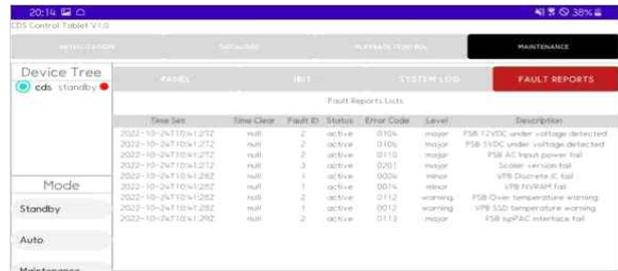


그림 15. 결함 보고서 페이지
 Fig. 15. fault report page.

V. 결론

본 논문에서는 CDS RCS의 설계 및 구현을 다루었다. CDS RCS는 MQTT 기반의 CDS 장치 및 미디어 제어 기능과 ARINC-615A 및 ARINC665를 응용하여 개발된 항공기용 데이터로드 프로세스의 제어 기능을 제공한다.

기존 CDS 제어 소프트웨어는 MQTT server로 설계되어 IDPM 장치들이 MQTT client로서 CDS 제어 소프트웨어와의 연결 상태에 의존하는 단점이 있었지만, CDS RCS는 MQTT client로 설계되어 IDPM 장치들이 CDS 제어 소프트웨어와 독립적으로 운영될 수 있으며, 안드로이드 기반의 개인 단말기에 설치할 수 있어 휴대성과 시인성을 향상시켰다는 장점이 있다.

향후 연구로는 CDS RCS를 IOS 환경으로 포팅하고, 비디오/오디오 스트리밍 기능을 제공할 수 있도록 개발할 예정이다.

Acknowledgments

본 논문은 산업통상자원부 재원으로 한국 산업 기술 평가 관리원(KEIT)의 지원을 받아 수행된 연구 결과입니다. [사업명: 스마트 캐빈 기술개발사업, 과제명: 항공기용 대형 flexible OLED 디스플레이 시스템 개발/과제고유번호: 20011845]

References

[1] G. Torkashvand, *Optimization of cabin design for enhanced*

- passenger experience*, Florida Institute of Technology ProQuest Dissertations Publishing, 2019.28216828, 2019.
- [2] A. O. Moraes, and F. R. D. A. S. Ciaccia, "Smart cabin design concept for regional aircraft: technologies, applications & architecture," *Journal of Aerospace technology and Management* 15, Aug. 2023. DOI: <https://doi.org/10.1590/jatm.v15.1310>.
- [3] B. K. Shin, H. S. Ji, and J. W. Han, "A plan for obtaining technology development certification for the application of a large flexible OLED display system and a smart divider in a single-aisle medium-sized aircraft cabin," in *KSAS 2020 Spring Conference*, pp. 347-348, Jul. 2020. Retrieved from <https://www.dbpia.co.kr/pdf/pdfView.do?nodeId=NODE10442160>.
- [4] D. H. Kim, J. Y. Kim, and J. H. Kim, "Design and test method of smart cabin system for airplane," in *KSAS 2021 Fall Conference*, pp. 447-448, Nov. 2021. Retrieved from https://www.dbpia.co.kr/pdf/pdfView.do?nodeId=NODE11031363&googleIPSandBox=false&mark=0&minRead=5&ipRange=false&b2cLoginYN=false&icstCls=050000&isPDFSizeAllowed=true&accessgl=Y&language=ko_KR&hasTopBanner=true.
- [5] J. K. Shin, J. H. Choi, N. M. Choi, J. H. Kim, and J. Y. Kim, "Multi-division video player design and implementation with standard interface for aircraft in embedded environments," *Journal of The Korea Contents Association*, Vol. 22, No. 11, pp. 80-91, Oct. 2022. DOI: <https://doi.org/10.5392/JKCA.2022.22.11.080>.
- [6] J. K. Shin, J. H. Choi, N. M. Choi, and J. H. Kim. "Aircraft mounted media player remote control software design," in *KSAS 2023 Spring Conference*, pp. 1344-1345, 2023. Retrieved from https://www.dbpia.co.kr/pdf/pdfView.do?nodeId=NODE11439703&googleIPSandBox=false&mark=0&minRead=5&ipRange=false&b2cLoginYN=false&icstCls=050000&isPDFSizeAllowed=true&accessgl=Y&language=ko_KR&hasTopBanner=true.
- [7] J. H. Choi, N. M. Choi, and J. K. Shin, "Data load process of large-sized media for avionics using FTP and JSON," *Journal of Advanced Navigation Technology*, Vol. 27, No. 5, pp.610-620, Oct. 2023. DOI: <https://doi.org/10.12673/jant.2023.27.5.610>.
- [8] N. Naik, "Choice of effective messaging protocols for IoT systems: MQTT, CoAP, AMQP and HTTP," in *2017 IEEE International Systems Engineering Symposium (ISSE)*, pp. 1-7, Oct. 2017. DOI: <https://doi.org/10.1109/SysEng.2017.8088251>.
- [9] ISO/IEC. Information technology message queuing telemetry Transport (MQTT) v3.1.1. [Internet], 20922:2016, Available: <https://www.iso.org/standard/69466.html>.
- [10] U. Hunkeler, H. L. Truong, and A. S. Clark, "MQTT (2008) MQTT-S — A publish / subscribe protocol for wireless sensor networks," in *3rd International Conference on Communication Systems Software and Middleware and Workshops (CMOSWARE '08)*, Bangalore: India, pp. 791-798, Jan. 2008. DOI: 10.1109/COMSWA.2008.4554519.
- [11] OASIS. MQTT Version 5.0 [Internet], Available: <https://docs.oasis-open.org/mqtt/mqtt/v5.0/mqtt-v5.0.html>
- [12] R. Rashmi, R. Srinidhi, S. Shriharsha, K. Ashwani, and R. Byra, "ARINC 615A and 665-3 based data loader for aircrafts," *International Journal of Engineering Trends and Technology*, Vol. 37, No. 5 pp. 260-264, 2016. DOI: 10.14445/22315381/IJETT-V37P244.
- [13] Airlines Electronic Engineering Committee, *Loadable software standards, ARINC report 665-3*, Aeronautical Radio, Inc., 2551 Riva Road, Annapolis, Maryland 21401-7435, 2005.



신 재 권 (Jae-Kwon Shin)

2017년 8월 : 충남대학교 컴퓨터공학과 (공학사)
2020년 2월 : 충남대학교 컴퓨터공학과 (공학석사)
2020년 1월 ~ 현재 : LIG넥스원 항공연구소 선임연구원
※ 관심분야 : 항공전자, 실시간 운영체제, 임베디드 시스템, 스마트캐빈 시스템, 멀티미디어, 안드로이드



최 지 환 (Ji-Hwan Choi)

2020년 8월 : 명지대학교 전자공학과 (공학사)
2020년 8월 ~ 현재 : LIG넥스원 항공연구소 선임연구원
※ 관심분야 : 항공전자, 실시간 운영체제, 임베디드 시스템, 스마트캐빈 시스템, 데이터 통신



최 낙 민 (Nak-Min Choi)

2009년 2월 : 충남대학교 컴퓨터공학과 (공학사)
2011년 2월 : 충남대학교 컴퓨터공학과 (공학석사)
2011년 1월 ~ 현재 : LIG넥스원 항공연구소 수석연구원
※ 관심분야 : 항공전자, 실시간 운영체제, 임베디드 시스템, 스마트캐빈 시스템, 데이터 통신



김 정 열 (Jeong-Yeol Kim)

2005년 2월 : 한밭대학교 컴퓨터공학과(공학사)
2021년 2월 : 충남대학교 컴퓨터공학과(공학석사)
1996년 1월 ~ 현재 : LIG넥스원 항공드론연구소 수석연구원
※ 관심분야 : 항공전자, 임베디드 시스템



김 준 형 (Jun-Hyoung Kim)

2000년 3월 ~ 2007년 2월 : 단국대학교 전기전자컴퓨터공학부 (공학사)
2007년 3월 ~ 2009년 8월 : 한양대학교 전자컴퓨터통신공학과 (공학석사)
2009년 1월 ~ 현재 : LIG 넥스원 항공연구소 수석연구원
※ 관심분야 : 항전시스템, 비행기록장치, 스마트캐빈시스템