JTE HCMUTE
ISSN: 2615-9740

JOURNAL OF TECHNICAL EDUCATION SCIENCE
Ho Chi Minh City University of Technology and Education
Website: https://jte.edu.vn
Email: jte@hcmute.edu.vn

Developing a Solution for Controlling Technical Systems in Airports Based on Flight Information Display System

Thai Hoang Huynh
Ho Chi Minh City University of Technology, VNU-HCM, Vietnam

Corresponding author. *Email: hthoang@hcmut.edu.vn*

ARTICLE INFO

Received:	11/09/2024
Revised:	29/09/2024
Accepted:	08/10/2024
Published:	28/11/2025

KEYWORDS

Building management system;

Flight information display system;

Airport technical system;

Ecostruxe Building Operation;

Ecostruxe Web Service.

ABSTRACT

The paper presents a solution for automatic control of technical equipment in airports, such as electricity, air conditioning, lighting, and baggage conveyor systems… based on the flight information display system to enhance operational efficiency and cost savings according to specific flight schedules. Firstly, the control solution for technical equipment in airports based on building management software is introduced. Next, the paper proposes a solution to update the scheduling database for equipment control based on flight information display system data. A database is designed with three data tables, of which one data table storing device group information and two data tables storing information about arrival and departure flights. A program is developed to read data in the flight information display system and store data in the database. A middleware is employed to retrieve data from the database and integrate it into the building management software to control device groups in the airport following flight schedules. Simulation results demonstrate that technical equipment can be automatically controlled to align with departure and arrival flight schedules. In case of flight delays or cancellations, the equipment control schedule is automatically updated, facilitating energy savings and operational cost reduction.

Doi: https://doi.org/10.54644/jte.2025.1652

1. Introduction

The Building Management System (BMS) is a synchronized system that allows for the control and monitoring of all technical systems in a building, such as the electrical system, water supply pump system, wastewater drainage system, ventilation and air conditioning system, security system, fire alarm and firefighting system, etc. The BMS ensures that the operation of building equipment is accurate, timely, efficient, energy-saving, and cost-effective ([1], [2], [3]). The BMS comprises a central computer with building management software connected to direct digital controllers (DDCs) through a communication network. DDC controllers are connected to field devices, including sensors and actuators, to control and monitor all technical devices in the building. The BMS system has several characteristic features, such as enabling synchronized and accurate operation of building equipment according to the operator's preset schedule. The system can detect and alert to incidents and provide timely warning signals before incidents occur. Additionally, the system aggregates data for storage, monitoring, and reporting information. Currently, BMS systems are widely applied in high-rise buildings, commercial centers, hotels, hospitals, airports, etc. For airports, the BMS system needs to optimally control technical equipment in the departure and arrival terminals according to departure and arrival flight schedules to save energy and minimize operating costs. Many commercial BMS hardware and software products are available from companies such as Schneider, Honeywell, Siemens, Johnson Controls, Distech Control, etc. Commercial BMS software can control devices according to schedules preset by the operator. However, device control schedules cannot be automatically updated based on data from the Flight Information Display System (FIDS). Technical operators must manually update device control schedules, which is time-consuming and can lead to human errors. Whenever a flight is delayed or canceled, operators must update device control schedules, resulting in additional labor costs. To make the airport's BMS system operate more efficiently, an integrated solution is needed to incorporate data from the FIDS into automatically updating the control schedules for technical

JOURNAL OF TECHNICAL EDUCATION SCIENCE
Ho Chi Minh City University of Technology and Education
Website: https://jte.edu.vn
Email: jte@hcmute.edu.vn
ISSN: 2615-9740

equipment at the airport. This is the motivation behind this research. To be able to conduct simulations verifying the integration of FIDS into BMS, this study proposes and implements a solution to integrate data from the FIDS into automatically updating the device control schedules of the Schneider EcoStruxure Building Operation (EBO) building management software. However, it is essential to emphasize that the proposed solution can also be applied to building management software from other vendors.

The rest of the paper is organized as follows. Section 2 presents the structure of the building management system and proposes a solution to integrate data from the flight information display system to update the control schedules of devices in airports automatically. The results of implementing the proposed solution are presented in section 3. Finally, section 4 gives conclusions and discusses future research directions.

2. Solution Development

2.1. Structure of building management system

The building management system studied in this paper is the EBO system ([6]) from Schneider. It is a web-based software platform providing an environment for designing, deploying, and monitoring building management systems connected via communication infrastructure. The EBO system has a hierarchical architecture based on a server-client mechanism, as shown in Figure 1. The Enterprise Central (EC) is the server at the top level in the EBO structure, usually used for multi-site building to manage and operate the system comprehensively. An EC can manage several Enterprise Servers (ES) and consequently manage multiple controllers. As a server at the highest level of the system architecture, the EC allows users to access the system for configuration, control, and search across the entire system. Additionally, the EC collects data across the system for storage, analysis, and reporting. The Enterprise Server (ES) is a server that has an essential role in the system and performs core functions such as logic control and monitoring, data collection and storage, information analysis and report. At the middle level of the system, the direct digital controllers (DDCs) are managed by the servers, and they are responsible for controlling and collecting data from equipment in the building via actuators and sensors. The components in the BMS architecture are connected via a communication network based on various protocols such as BACnet IP, BACnet MS-TP, Modbus TCP, Modbus RTU, and LonTalk.

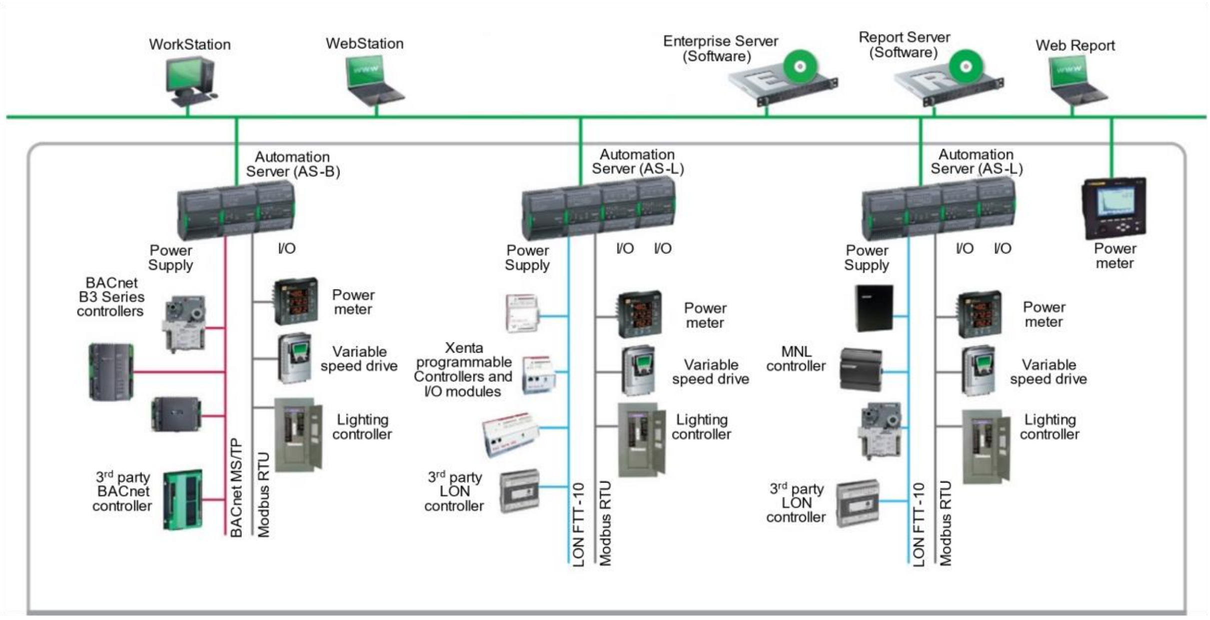

Figure 1. *Structure of building management system*

One of the most commonly used functions of the BMS system is the ability to control devices on or off followed schedules. Figure 2 illustrates the BMS software interface allowing operators to set up device control schedules. In airports, most devices are required to control automatically according to flight schedules to save energy. The fact is that flight schedules are changeable because of many different reasons. If devices are set to operate according to fixed schedules, they may not align with flight operating times and result in a waste of energy. If the operators have to update the device control schedules corresponding to the change of flight schedules, a lot of labor is required, and human mistakes could happen. Therefore, for the BMS system to control devices at the airport more efficiently, automatically updating device control schedules based on information from the Flight Information Display System (FIDS) is necessary.

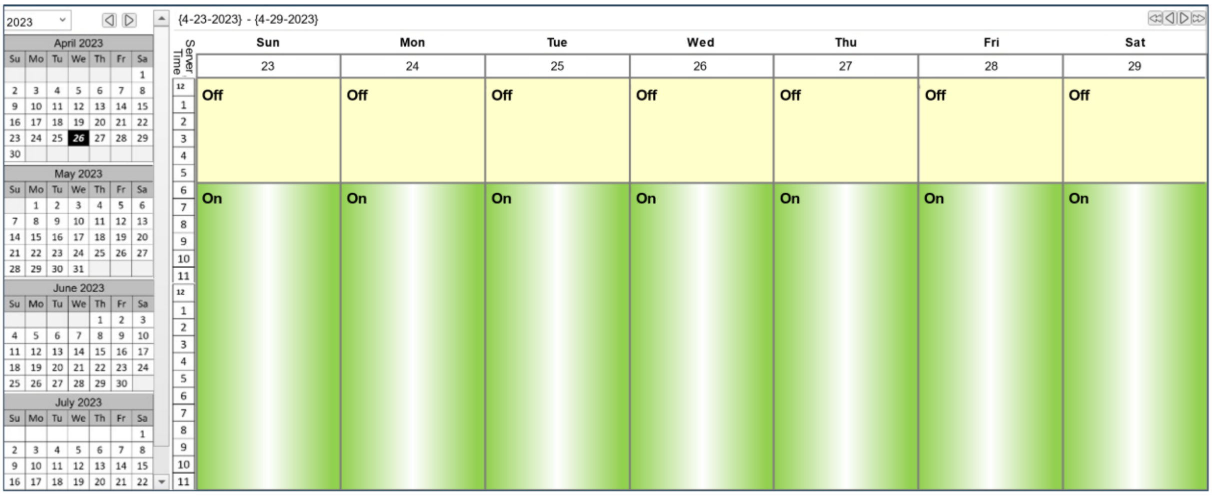

Figure 2. *Schedule setting screen*

2.2. Solution to control technical systems in airports following flight information display system

The Flight Information Display System (FIDS) ([4], [5]) is a system that displays information about flights at the airport. It provides information about the flights, including airlines, flight numbers, departures, destinations, check-in counters, boarding gates, boarding times, and delays. Passengers can view the FIDS electronic boards at the airport or access the website to obtain flight information, see Figure 3.

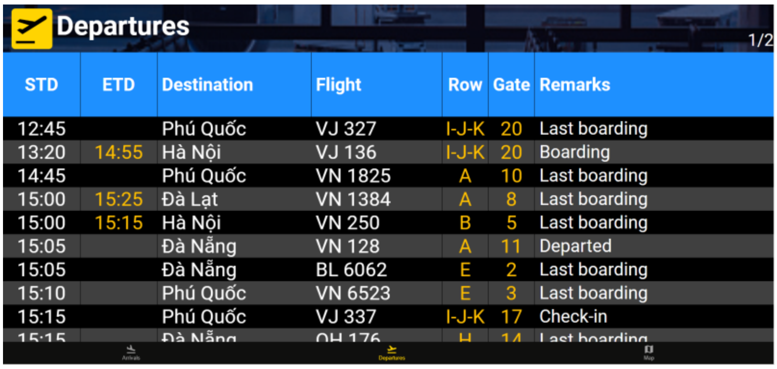

Figure 3. *Flight Information Display System*

The FIDS can communicate with the BMS through high-level interfaces such as OPC, API, or export data to Excel files. In this study, a software application is designed and programmed to automatically read flight information data from FIDS in Excel file format and update the SQL database. Then, the Smart Connector software is used to query flight information stored in SQL and feed it into EBO through EcoStruxure Web Services (EWS) to automatically update the device control schedules, as shown in Figure 4. The Smart Connector is middleware software used to connect Schneider devices to Management Information Systems (MIS) or other systems such as energy management systems, building management systems, and operation and maintenance management systems. EWS is a cloud-based platform providing secure and scalable access to Schneider's IoT-connected products and services. EWS allows users to monitor and control connected devices from anywhere, anytime, via a web browser

JTE HCMUTE
ISSN: 2615-9740
JOURNAL OF TECHNICAL EDUCATION SCIENCE
Ho Chi Minh City University of Technology and Education
Website: https://jte.edu.vn
Email: jte@hcmute.edu.vn

or mobile app. EWS provides a range of features, including real-time data visualization, analysis and alerts, user access management and authorization.

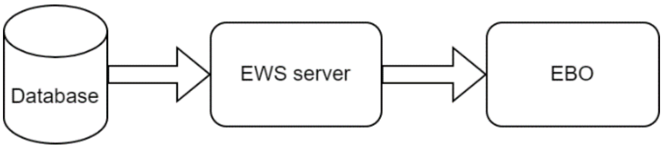

Figure 4. *Get information from the database to EBO via EWS server*

The solution for automatically updating device control schedules at the airport based on information from FIDS is implemented through the following steps: designing the SQL database, designing the EWS server, querying data from SQL using the Smart Connector, receiving and processing data in the EBO software, and automatically updating the SQL database based on information from FIDS. These steps are presented below.

2.2.1. Designing the SQL database

The database consists of 3 tables: 1 table storing device group information and 2 tables storing information about arrival and departure flights.

Table 1. *Structure of data table to store Group of Devices of arrival flight*

Equipment	AG01	AG02	AG03	AG04	AG05	…
Ahu11a	1	1				
Ahu11b			1		1	
Ahu12a				1		
…						
Lgt01	1			1	1	
Lgt02	1	1	1			
Lgt03	1	1	1	1	1	

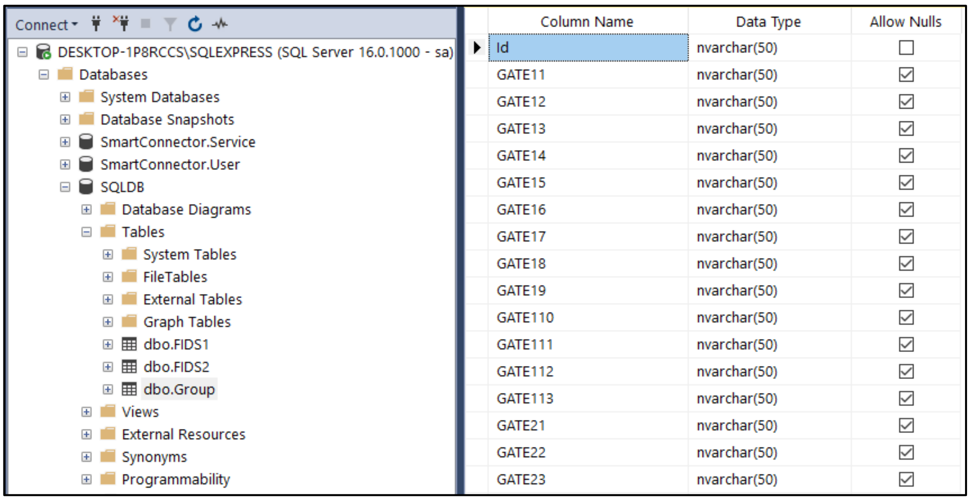

Figure 5. *Data field and data type of Group of Device table*

Device Group Data Table: In airport operations, each flight is associated with specific areas such as check-in counters, departure gates, arrival gates, baggage conveyor belts, etc. Each area is related to several devices, such as air handling units, ventilation fans, lighting lines, etc. Therefore, there is a need

JTE
HCMUTE
ISSN: 2615-9740

JOURNAL OF TECHNICAL EDUCATION SCIENCE
Ho Chi Minh City University of Technology and Education
Website: https://jte.edu.vn
Email: jte@hcmute.edu.vn

for a database to store devices by group, in which devices can belong to one or more groups. A group can contain several devices, and devices within the same group will have the same enabled control signal. If a device belongs to several groups, its enabled control signal will depend on the enabled control signals of these groups. Table 1 illustrates the data table storing devices for arrival flights. For example, Arrival Gate AG01 is associated with devices such as the air handling unit Ahu11a and lighting lines Lgt01, Lgt02, and Lgt03. When there is a flight arriving at gate AG01, all devices related to gate AG01 should be turned on. Light Lgt01 is associated with arrival gates AG01, AG04, and AG05, meaning that if flights arrive at these gates, the lighting line Lgt01 should be turned on. Figure 5 illustrates the fields and data types of the Device Group table after being set up in the SQL database.

Flight Information Data Table: The flight information data table (Figure 6) is designed to store information received from the FIDS. There are various FIDS systems provided by different vendors; however, all FIDS systems provide flight-related information such as airlines, flight number, destination/departure airport, arrival/departure gate, arrival/departure time, baggage conveyor, etc.

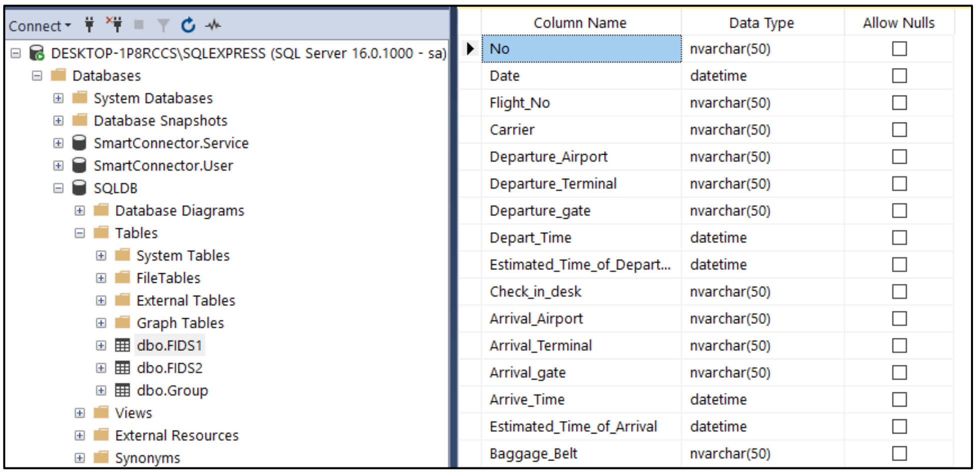

Figure 6. *Data field and data type of FIDS table*

2.2.2. Designing the EWS server

Two EWS servers are created to retrieve data from the SQL database and integrate it into EBO. One is used to update the timing information from the FIDS, and the other is used to update the operation status of device groups. After creation, if the SQL connection string is correctly established, there will be two active EWS servers, as depicted in Figure 7.

Figure 7. *EWS servers running*

To ensure the automatic operation of the EWS server and data updates from the database, operation schedules need to be established so that the server can run automatically after a specified interval. A list of operation schedules with different operating cycles is created, as shown in Figure 8. The purpose is to conveniently make appropriate changes to server running cycles during the simulation and provide more diversity for the users to choose from during actual operations. Subsequently, the created operation schedules are set for each server; as a result, each server will operate after a defined set interval.

	Start Time	Description
	04/25/2023 2:00 PM	1 Hour
	03/07/2023 8:00 PM	1 Minute
	04/25/2023 2:00 PM	10 Minutes
	03/27/2023 11:00 PM	10 Seconds
	04/25/2023 2:00 PM	15 Minutes
	04/25/2023 2:00 PM	2 Minutes
	04/25/2023 2:00 PM	30 Minutes
	03/27/2023 11:00 PM	30 Seconds
	04/25/2023 2:00 PM	5 Minutes

Figure 8. *Setting operation schedules for the EWS servers*

2.2.3. Querying data from the database

SQL queries are created to retrieve data from the database to get the required data, as illustrated in Figure 9. The data includes flight departure and arrival times, as well as information about the device groups, and devices of which groups are to be operated.

Figure 9. *Creating queries to retrieve data from the database*

- For device groups, the data is stored as a string type with the following format: the Id Field represents the device group, and its Value Field contains a list of devices belonging to that group.
- For departure flights, the data is stored as Datetime type and includes the departure gates (DG), check-in areas, etc. For example, when a flight is scheduled for departure, passengers are allocated to check in at check-in counter 1, then the devices belonging to the check-in counter 1 will be turned on.
- For arrival flights, the data is stored as Datetime type and includes the arrival gates (AG), and baggage conveyor areas. For example, when a flight is scheduled to land, passengers are directed to receive luggage at Baggage Conveyor 1, then the devices belonging to the Baggage Conveyor 1 group will be turned on.

Flowcharts for querying the device list in the Device Group Data Table and flight time data in the Flight Information Data Table to update device control schedules are presented in Figures 10.a and Figure 10.b, respectively. In Figure 10.b, the TerminalName could be T1, T2,…; and AreaName could be DGxx, AGxx,… (xx is a number labeling the area).

JTE HCMUTE
ISSN: 2615-9740
JOURNAL OF TECHNICAL EDUCATION SCIENCE
Ho Chi Minh City University of Technology and Education
Website: https://jte.edu.vn
Email: jte@hcmute.edu.vn

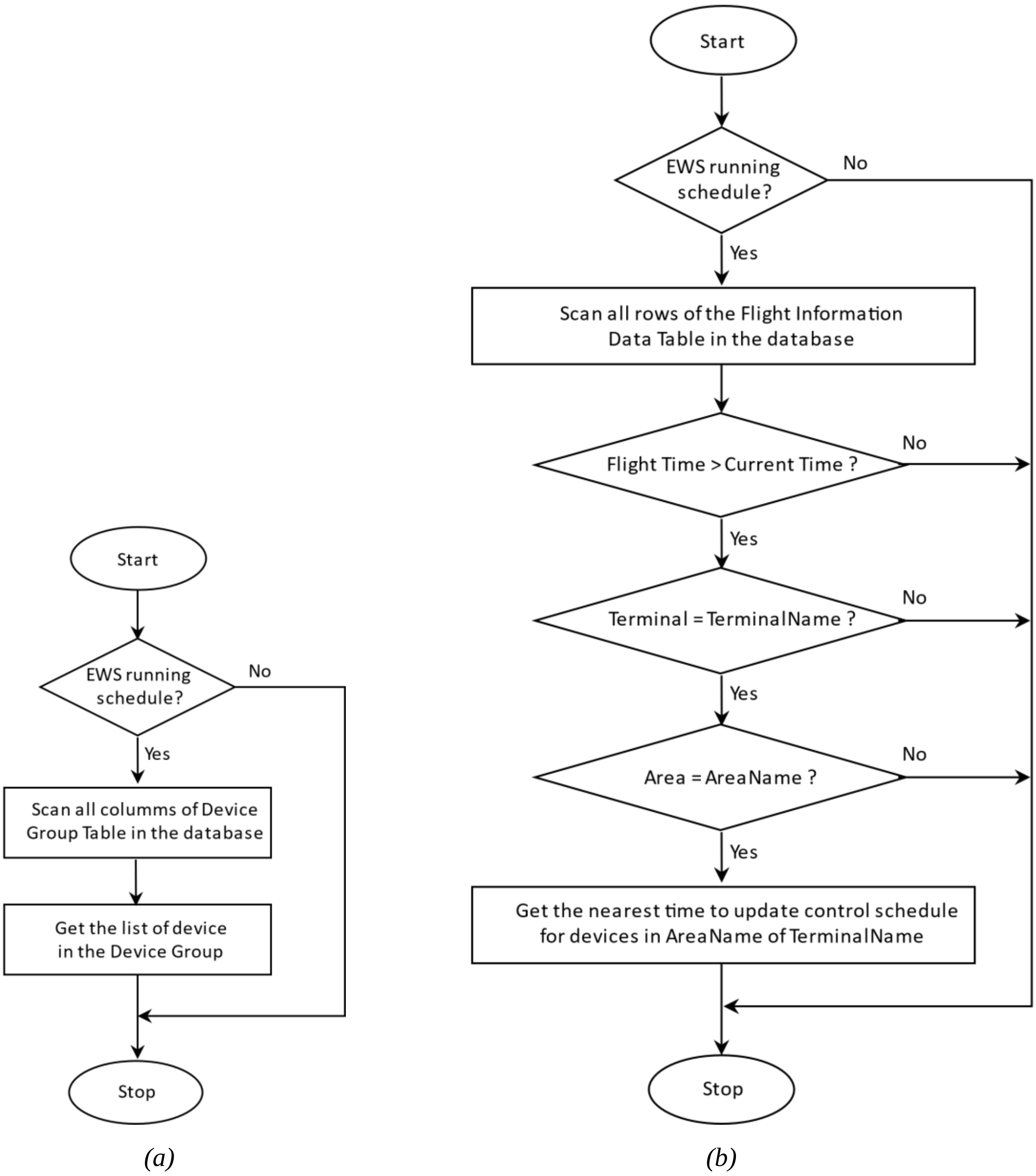

Figure 10. *Flowchart for (a) querying device list from Device Groups table in the database (b) querying flight schedule data from the Flight Information data table in the database*

2.2.4. Receiving and processing data in the EBO software

The data queried by the EWS Server from the database is then sent to the EBO software, which interprets the data as Datetime type. A program is required to compare the flight schedule time read from the database with the current time to generate enabled commands in the form of binary signals to activate the devices. Once we have the enabled signal, we need a program to control each specific device based on that signal. The block diagram of the device control program according to the flight schedule data from the database is depicted in Figure 11.

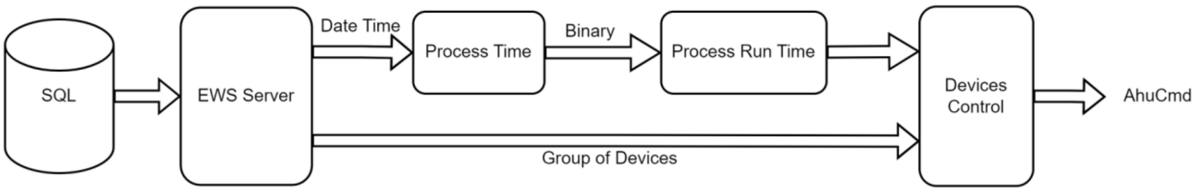

Figure 11. *Block diagram of the program to control devices following flight schedule from the database*

JTE HCMUTE
ISSN: 2615-9740
JOURNAL OF TECHNICAL EDUCATION SCIENCE
Ho Chi Minh City University of Technology and Education
Website: https://jte.edu.vn
Email: jte@hcmute.edu.vn

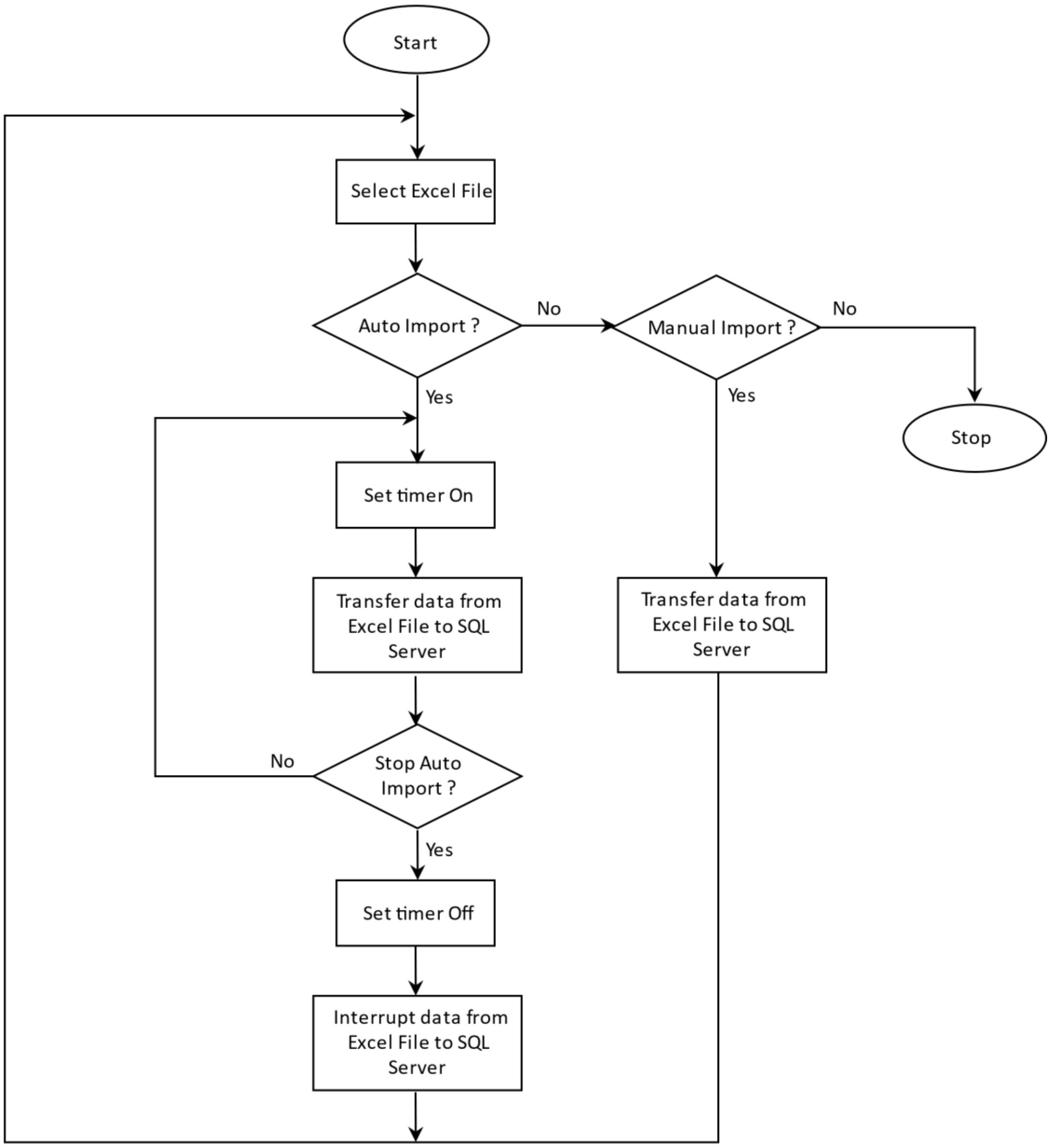

Figure 12. *Flowchart for reading flight information from FIDS and updating the SQL database*

According to the process of serving departure and arrival flights, devices will operate according to different schedules between these two types of flights. Specifically, for departure flights, devices will start operating before the check-in time by a certain period and turn off before the take-off time. Conversely, for arrival flights, devices will start operating after the aircraft has landed, and turn off after a pre-set period. These on/off periods can be set by the operator. Therefore, a program is needed to handle the operating time of devices suitable for each type of flight.

2.2.5. Automatic database update

The system needs to automatically update the flight information database from the FIDS to ensure accurate information that reflects the status of flights, such as adding flights, canceling flights, or delaying flights. Automatic data updates save time for airport technical operators by eliminating the need for manual data entry and searching for flight information, thus reducing the risk of human errors. Figure 12 presents the algorithm flowchart for reading flight information from the FIDS system in Excel format and updating the SQL database. According to the flowchart, data can be updated from FIDS manually by operators or automatically based on a pre-defined time interval. Thanks to continuous

JTE
HCMUTE
ISSN: 2615-9740

JOURNAL OF TECHNICAL EDUCATION SCIENCE
Ho Chi Minh City University of Technology and Education
Website: https://jte.edu.vn
Email: jte@hcmute.edu.vn

database updates with flight information, airport devices can be controlled to turn on or off appropriately based on the flight status.

3. Results

The airport BMS has been implemented using the EBO software (trial version) with an overview interface as shown in Figure 13. The BMS system includes various functions such as monitoring and controlling the electrical system, air conditioning system, elevator system, fire alarm system, and energy management system. This paper focuses on the device control function based on flight information from the FIDS system. Figure 14 is a C# programmed interface that allows operators to set up automatic updates of the SQL database based on flight information exported from the FIDS system in Excel format.

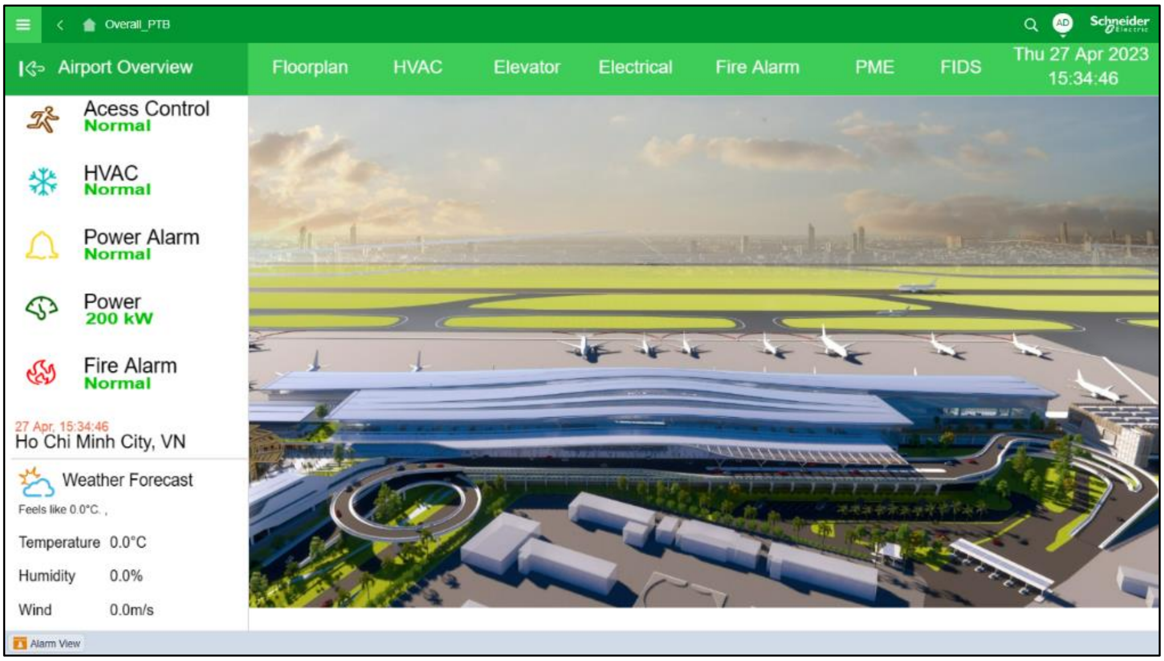

Figure 13. *Overview screen of BMS for controlling and monitoring technical systems in airport*

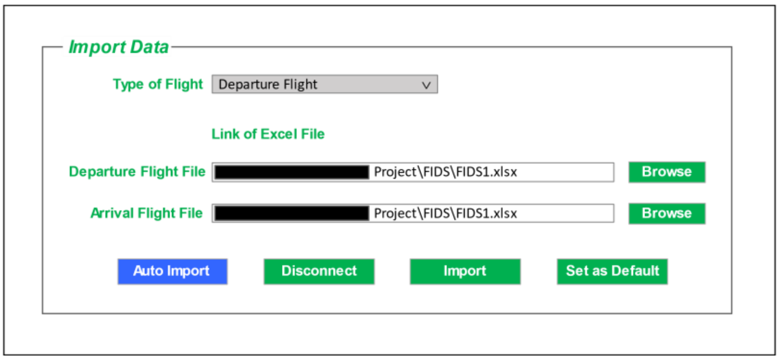

Figure 14. *Graphical interface for setting the update of SQL database from FIDS*

Figure 15 shows monitoring and control interfaces for devices serving departure flights. The interfaces display the start and end times of operations for devices related to flights, such as departure gates, arrival gates, baggage conveyors, etc. Simulation results demonstrate that when flight information from the FIDS system changes, such as adding flights, canceling flights, or delaying flights, the SQL database is automatically updated, and the schedules for controlling airport devices are correspondingly adjusted without any manual intervention. This enhances the efficiency of technical systems' operations at the airport, leading to energy savings and reduced operational costs.

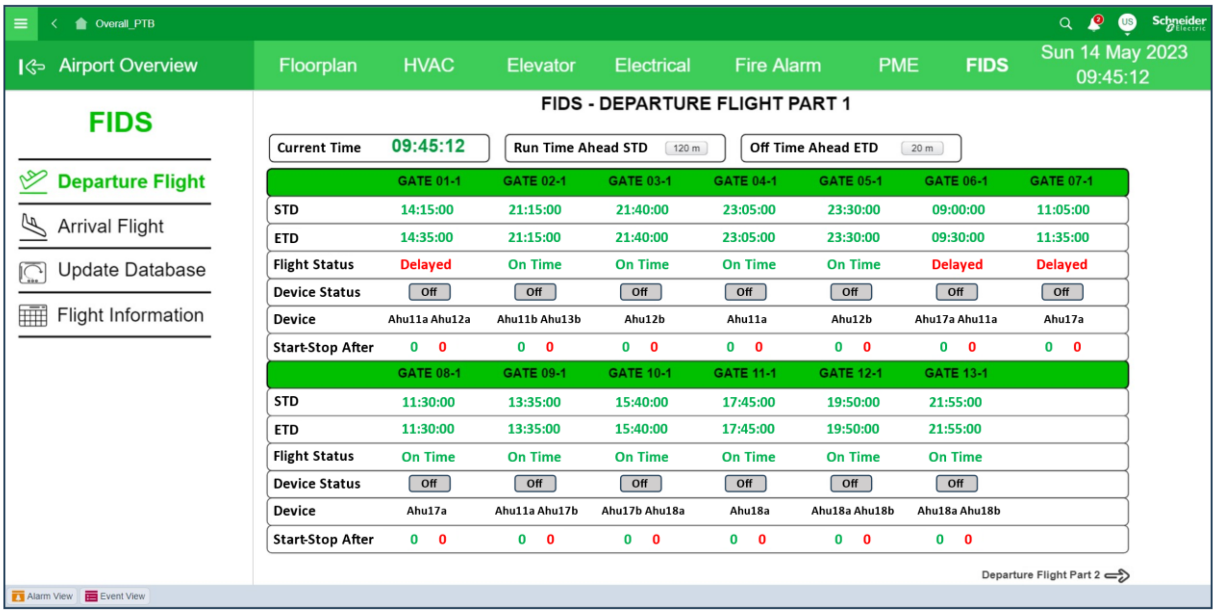

Figure 15. *Graphical interface for controlling and monitoring technical systems following FIDS*

4. Conclusions

This paper has presented a solution for automatically updating the schedule for controlling devices at the airport based on flight information from the FIDS system. The main contribution of this paper is the proposal of a solution and programming for reading flight information from the FIDS system to update the SQL database. Additionally, algorithms and programs were developed to enable the BMS software, specifically EBO, to access information from the database and control devices based on the flight scheduled time stored in the database. This solution improves the efficiency of conventional BMS systems, which typically only control devices based on pre-set schedules. Simulation results show that when flight information changes, such as adding flights, canceling flights, or delaying flights, the schedules for controlling device groups related to flights are automatically updated accordingly. In the future, the proposed solution is expected to be applied to BMS systems for real airports, enhancing the efficiency of controlling technical systems at airports, minimizing errors from operator manual updates, and saving energy and operational costs. Automatically turning off devices that are not serving flights according to the updated schedules also contributes to extending the lifespan of the devices. In this study, flight information from FIDS is read and the SQL database is updated according to pre-set time intervals. The solution can be further improved to update the database only when the Excel file containing flight information from FIDS changes, thus ensuring more timely and efficient data updates.

Acknowledgments

The author sincerely thanks Hai Nam Automation Technology JSC for supporting the equipment, software, and human resources to conduct this research.

Conflict of Interest

The author declares no conflict of interest.

REFERENCES

[1] G. Hayduk, P. Kwasnowski, and Z. Mikoś, "Building management system architecture for large building automation systems," *17th International Carpathian Control Conference (ICCC),* 2016

[2] K. H. Lam, W. M. To, and P. K. C. Lee, "Smart Building Management System (SBMS) for Commercial Buildings—Key Attributes and Usage Intentions from Building Professionals' Perspective," *Sustainability*, 15, 80, 2023

[3] R. Eini, L. Linkous, N. Zohrabi, and S. Abdelwahed, "Smart building management system: Performance specifications and design requirements," *Journal of Building Engineering*, 39, 2021

[4] K. H. Tseng, K. H. Chen, and C. L. Chu, "Design and implementation of flight information management system," *2016 IEEE 11th Conference on Industrial Electronics and Applications (ICIEA)*, Hefei, China, pp. 61-65, 2016

[5] A-ICE, "A-FIDS Flight Information Display System – System Overview." [Online]. Available: http://www.a-ice.aero/

[6] Schneider Electric, "Ecostruxure Building." [Online]. Available: http://se.com/ecostruxure-building

Thai Hoang Huynh obtained his PhD in Automatic Control Engineering from Ho Chi Minh City University of Technology in 2005. He conducted post-doctoral research in 2007 at the University of Haute Alsace (France). In 2013, he was awarded the title of Associate Professor. Currently, he teaches and conducts research in the Department of Control and Automation, Faculty of Electrical-Electronics Engineering, Ho Chi Minh City University of Technology, Vietnam National University. His primary research directions include intelligent control, system identification, and industrial automation.

Email: hthoang@hcmut.edu.vn. ORCID: https://orcid.org/0000-0001-7196-4753