

제조업체 사무자동화를 위한 RPA 소프트웨어 구현에 대한 연구: 지역 제조업체 사례를 중심으로

A Study on the Implementation of RPA Software for the Manufacturer Automation: Focusing on the Case of a Local Manufacturer

정성욱^{1*}

Sung-Wook Chung^{1*}

〈Abstract〉

Robot Process Automation (RPA) is a computer technology called Robotic Process Automation, a form of business process automation based on the concept of software robots or artificial intelligence (AI) walkers. In general, in traditional workflow automation tools, software developers design software that creates a set of actions to automate tasks and interfaces for the back-end systems using internal APIs or dedicated script languages. However, in RPA software, automation can be implemented by configuring an operating processor as if the general user is directly performing the task of the application. In other words, it can be said that it is a suitable development method for automating simply repetitive tasks rather than developing specific programs in which all necessary functions are implemented, as in general software development. Thus, this is more appropriate for configuring and automating RPA software in traditional manufacturing companies that are not easy to develop and apply smart factories or high-end AI software. Therefore, this research aims to analyze the requirements required at the actual manufacturing companies, focusing on the manufacturer's case in Changwon, Gyeongsangnam-do, called SinceWin Co., Ltd., and to examine the possibility of RPA software in the manufacturing companies by implementing actual RPA software that supports office automation. Through the research, it was confirmed that the actually implemented RPA software met the requirements of the company and helped manufacturer practice significantly by automating the parts that were worked error-prone and manually periodically.

Keywords : *Robotic Process Automation (RPA), RPA Software, Manufacturing Automation, Case Study, Local Manufacturer Automation*

^{1*} 정회원, 주저자, 창원대학교 컴퓨터공학과 부교수
E-mail: swchung@changwon.ac.kr

^{1*} Associate Professor, Dept. of Computer Engineering,
Changwon National University

1. 서론

컴퓨터 기술의 발달과 다양한 유무선 네트워크 기술 및 단말기술의 발전은 다양한 분야에서의 컴퓨터 기술의 적용 및 활용으로 나타나고 있다. 일반적으로 컴퓨터기술은 컴퓨터자체의 성능향상이나 알고리즘 등의 기술을 탐구하고 연구하는 분야가 있고, 컴퓨터를 활용하여 여러분야에 그 기술을 응용하는 분야가 있다. 이를 통해, 일상생활에서의 누구나 손쉽게 컴퓨터기술의 활용이 가능해지고 조선이나 제조산업 분야 등 직접적인 컴퓨터 분야가 아닌곳에서도 컴퓨터활용이 보편화되어가고 있다.

특히, 제조산업 분야에서는 다양한 제조공정에서 생산성의 향상 및 자동화를 위한 컴퓨터 기술의 적용이 가능하다. 예를 들어, CAD/CAM (Computer-Aided Design/Computer -Aided Manufacturing) 등을 활용하여 제조하고자 하는 산출물의 디자인 및 설계를 지원하고, ERP (Enterprise Resource Planning) 시스템 등을 활용하여 인사, 예산, 재무, 행정 등을 자동화하고, 재고물품의 데이터베이스화를 통한 실시간 기자재 현황 모니터링, 그리고 IoT (Internet of Things) 기술을 활용하여 제조공정에서의 실시간 현황 관리 등이 가능하다. 특히, 4차산업기술에서 부각되는 자동화 시스템 및 스마트디바이스를 활용하는 전 제조환경 및 공정에서의 스마트 팩토리 (Smart Factory) 기술의 적용이 가능하다.

하지만, 일반적인 제조환경에서 이러한 전체적인 스마트 팩토리 환경을 구성하여 바로 적용하는 것은 쉬운 일이 아니다. 따라서, 현실적으로는 전체 제조공정에서 적용가능한 부분부터 하나씩 컴퓨터자동화를 채택하여 적용할 수 있는 기법의 필요성이 대두된다.

RPA (Robot Process Automation)는 로보틱

처리 자동화라는 컴퓨터 기술로써, 소프트웨어 봇 또는 인공지능(AI) 워커(worker)의 개념에 기반을 둔 만들어진 비즈니스 프로세스 자동화 기술 형태이다. 일반적으로 전통적인 워크플로 자동화 도구에서 소프트웨어 개발자는 내부 API 또는 전용 스크립트 언어를 사용하여 백엔드 시스템에 대한 태스크와 인터페이스를 자동화하기 위한 일련의 동작을 생성하는 소프트웨어를 개발한다. 하지만, RPA 소프트웨어에서는 마치 일반 사용자가 애플리케이션의 태스크를 직접 수행하는 것처럼 작동 프로세서를 구성하여 자동화를 구현할 수 있다.

즉, 일반적인 소프트웨어 개발에서처럼 필요한 모든 기능이 구현된 구체적인 프로그램을 개발한 다기 보다는 단순 반복적인 작업을 자동화하여 수행하기에 적합한 개발 방법이라고 할 수 있다. 이는 스마트 팩토리 또는 고사양 AI 소프트웨어를 개발하여 적용하기 쉽지 않은 전통적인 제조기업에서 RPA 소프트웨어를 구성하여 자동화하기에 더욱 적합하다고 할 수 있다. 예를들어 Fig. 1에서처럼 그동안 사람에 의해 수행되었던 매일 반복되는 작업을 RPA 소프트웨어로 구현한다면 제조기업에 보다 적용하기 쉽고 업무능률 향상 및 에러를 감소 등의 효율을 높이는 것이 가능하다.



Fig. 1 RPA applicable-feasible areas

본 논문에서는 제조기업에서 RPA 소프트웨어의 적용 가능성을 확인해보기 위하여 실제 제조기업의 요구사항을 바탕으로 RPA 소프트웨어를 디자인하여 구현해보고 해당 제조기업에서 활용해보고자 한다. 즉, 전통적인 제조기업에서의 RPA 적용의 가능성을 살펴봄으로써 그동안 주기적으로 매뉴얼하게 수행했던 반복적인 업무 및 휴먼에러(Human Error)의 가능성이 높았던 업무의 RPA 자동화를 확인해보고자 한다.

이를 위해, 1991년에 경남 창원에서 설립되어 에어컨/냉장고 컴프레서 부품, 자동차 엔진부품 등을 제조하여 납품하여 2020년 기준으로 연매출 153억을 기록하고 있는 경남 지역 제조업체인 (주)신스윈(사업자등록번호: 609-81-55562, <https://sincewin.co.kr>)과 협업하여 제품 제조과정에서 RPA 소프트웨어 적용가능 부분을 살펴보고 실제 구현하여 제조기업에서 RPA의 효과를 살펴보고자 한다.

본 논문은 다음과 같이 구성되었다. 1장에서는 RPA의 필요성 및 제조기업에서의 적용가능성을 설명하고 2장에서는 RPA와 관련한 기존 연구들을 살펴본다. 3장에서는 제조기업에서의 요구사항을 분석하며 4장에서는 해당 RPA 소프트웨어를 구현한다. 그리고 5장에서는 구현된 SW와 그 결과를 분석하며 마지막으로 6장에서는 본 논문을 마무리한다.

2. 관련연구

RPA에 대한 기본개념과 RPA의 적용성 및 활용가능성은 [1][2][3]에서 논의되었다. 뿐만 아니라, RPA의 트렌드 및 융합방향은 [4][5][6]에서 설명되었으며, [7][8]에서는 일반적인 비즈니스에서 RPA의 접목방법 및 생산성향상 방법을 제시하였다. 그리고, 실제 제조산업에서 RPA를 응용한 반

도체공정 및 데이터 관리방법등은 [9][10]에서 논의되었다.

3. RPA개발 요구분석 및 고려사항

(주)신스윈은 2002년에 설립된 경남소재 중소기업으로 에어컨, 컴프레서 부품, 냉장고 컴프레서 부품, 자동차 엔진 부품 등을 생산하여 대기업 등 주요 고객사에 납품하는 제조회사이다. (주)신스윈은 해당 부품들을 주문생산하기 위해 고객사들로부터 주문서를 수령 하여 해당 부품을 제작하여 납품하는데, 각 주문서의 양식 및 부품의 명세, 그리고 재고의 현황, 납품 기일, 납품 완료 여부 등을 엑셀 파일 형태로 사람이 수동으로 관리를 하고 있다.

3.1 개발 요구사항

(주)신스윈은 수동으로 처리하는 작업에 대한 문제점을 인지하였고 문제점은 다음과 같다. 1) '사람이 수작업을 통해 데이터를 입력함으로써 인한 계산 오차' 2) '엑셀 파일 관리를 위해 걸리는 시간이 많아 다른 업무 처리가 힘들' 3) '정형화된 엑셀 형태가 존재하지 않아 유지보수가 힘들' 3가지가 존재한다.

이에 본 논문에서는 RPA를 적용한 사무 자동화 기술을 접목해 전 단락에서 제시한 (주)신스윈이 가지고 있는 3가지의 문제점을 해결하고자 한다. 즉, 주문서의 표준화, 재고를 고려한 주문사항 반영, 주문서의 자동화 등을 고려하여 전체 RPA 소프트웨어에 대한 모델을 제시하고 이를 개발하고자 한다. Fig. 2는 (주)신스윈의 전체 작업과정으로 네모박스로 강조된 부분이 RPA 소프트웨어를 구현하여 RPA 자동화를 제공하여 실제 제조현장에 적용할 부분이다.



Fig. 2 RPA S/W requirements for sincewin Co. Ltd

3.2 RPA 개발모델

본 논문에서 제시하는 RPA 소프트웨어의 개발 모델은 Fig. 3과 같다. 개발모델에서 RPA 소프트웨어와 외부 시스템에 대한 인터페이스를 명확히 하고 실제로 기술개발을 통해 구현해야 하는 부분과 각 컴포넌트 간의 인터페이스 및 입출력을 명확히 한다.

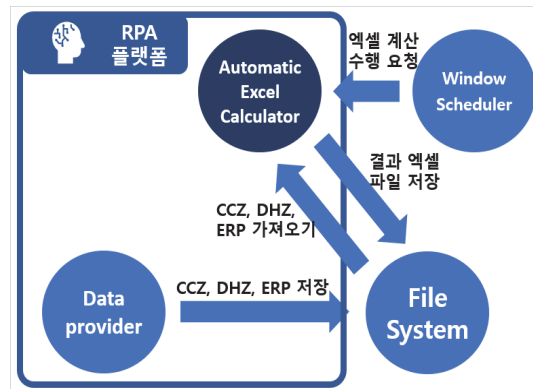


Fig. 3 Development design for RPA S/W

4. RPA시스템 소프트웨어 구현

RPA 소프트웨어는 Fig. 3의 개발모델에 따라, Data Provider와 Automatic Excel Calculator로 구성되며, 외부의 File System과 Window Scheduler와 연동된다.

4.1 Data Provider

Data Provider는 전체 RPA 시스템의 한 부분으로 주요 역할은 외부 시스템으로부터 Table 1에서 제시

Table 1. Input files, mid-output, result files for automatic excel calculator

Input Files	Mid-Output		Result Files	Run-Period
CCZ_Receiving_yyyymmdd.xlsx	CCZ Sheet	Merged Receiving Workbook	출하대비 입고실적 비교.xlsx	Week
DHZ_Receiving_yyyymmdd.xlsx	DHZ Sheet			
ERP_Receiving_yyyymmdd.xlsx	ERP Sheet			
Receiving_base.xlsx				
CCZ_price_yyyymmdd.xlsx	PU_SCC sheet	Merged Price Workbook	단가변경확인.xlsx	Month
DHZ_price_yyyymmdd.xlsx				
ERP_price_yyyymmdd.xlsx				
Price_base.xlsx				

된 ‘입력 파일’을 Window의 File System 내의 특정 경로(디렉토리)에 저장한다. 디렉토리의 위치는 ‘입력 파일’에 따라 2가지로 구분되며 각각 Receiving과 Price라는 이름을 가진다. 디폴트(Default)값으로 Receiving 관련 파일은 매주 1개씩 생성되며, Price 관련 파일은 매달 1개 씩 생성된다. 결과적으로 생성되는 날짜를 기준으로 새로운 경로(디렉토리)가 생성되며 ‘입력 파일’이 저장되게 된다.

예를 들어 Receiving 관련 ‘입력 파일’이 2021년 10월 11일에 생성될 경우, Receiving 디렉토리 내에 20211011이라는 이름을 가진 디렉토리가 새로 생성되며 해당 디렉토리 안에 아래 3가지의 엑셀 파일이 저장된다.

- CCZ_Receiving_20211011.xlsx
- DHZ_Receiving_20211011.xlsx
- ERP_Receiving_20211011.xlsx

4.2 Window Scheduler

Window Scheduler는 Window 운영체제에서 기본적으로 제공하는 기능으로, Table 1에서와 같이 ‘결과 파일’은 Weekly 혹은 Monthly로 주기적으로 생성되어야 한다. 따라서 RPA 시스템을 일정 주기로 실행해야 할 있으며, 이를 위해 구현하는 RPA 소프트웨어는 Window scheduler를 이용하여 주기적으로 수행할 수 있도록 구현하였다. 물론, 수동으로 “실행” 버튼을 통하여 필요할 때 바로 직접수행도 가능하다.

4.3 Automatic Excel Calculator

Automatic Excel Calculator (AEC)는 전체 RPA 시스템의 한 부분으로 본 논문에서 실제로 구현한 부분이다. AEC의 전체 시스템 구성도는 Fig. 4와 같으며, Window Scheduler에 설정된

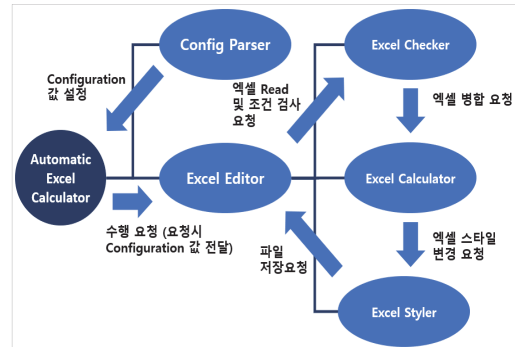


Fig. 4 Automatic excel calculator (AEC)

값의 일정 주기마다 실행되어 Data Provider에서 생성한 ‘입력 파일’을 받아 엑셀 파일을 계산하여 ‘중간 산출물’을 만든 후 이를 이용해 ‘결과 파일’을 생성해 Window의 file System 내의 특정 경로(디렉토리)에 저장하는 역할을 한다.

4.3.1 Config Parser

Config Parser는 사용자가 메모장에 설정한 Configuration 값을 Parsing 해서 AEC에 알려주는 역할을 하고 Configuration 값은 메모장에 태그 형태로 저장된다. 여기서 사용되는 Configuration 값은 ‘입력 파일’과 ‘결과 파일’에 대한 위치 정보가 저장된다.

예를 들어 Receiving의 ‘입력 파일’ 경로에 대한 정보를 위해 Path 태그를 구성하며, 해당값을 얻기 위해서 순서대로 Path(경로 Flag) → Receiving (파일 종류 Flag) → In(입력 or 결과 파일 Flag)을 Config Parser에게 제공하면 Receiving 디렉토리의 위치 정보를 반환하여 AEC에 그 값을 전달하게 된다. Fig. 5는 Configuration 태그구성 예시를 나타낸다.

```
<path>
  <receiving>
    <in> D:\Users\신스원\자료\Receiving </in>
    <out> D:\Users\신스원\결과 </out>
  </receiving>
  <price>
    <in> D:\Users\신스원\자료\Price </in>
    <out> D:\Users\신스원\결과 </out>
  </price>
</path>
```

Fig. 5 Tag congiguration for config parser

(a) Excel Checker

Excel Checker는 Scheduler로부터 AEC의 실행 요청이 들어오고 Config Parser에 의해 Configuration 값이 설정되었을 때, Excel Editor가 실제로 ‘결과 파일’ 생성을 위한 수행을 해야 할지 판단하며, 만약 다음에 해당하는 조건들이 부합하지 않는다면, Binary Back-off n시간(n=사용자가 설정한 대기 시간, 단위=sec) 동안 AEC를 Wait 한 후, 다시 점검을 시작한다. 여기서, 실행을 위한 조건은 1) 디렉토리 정보가 유효한가?, 2) ‘엑셀 파일의 I/O가 올바르게 실행되었는가?’ 에 대한 2가지 조건이 모두 충족하며 했을 경우 Receiving 관련 연산을 요청하기 위해 ‘Receiving 관련 연산이 1주일 이내에 실행되었는가?’를 검사하고 1주일 이내에 실행된 기록이 없으면 Receiving 연산을 요청한다. 또한, Price 관련 연산을 요청하기 위해 ‘Price 관련 연산이 1달 이내에 실행되었는가?’를 검사하고 1달 이내에 실행된 기록이 없으면 Price 관련 연산을 요청한다. Fig. 6은 Excel Checker에 대한 Sudo Code를 나타낸다.

```

TRY:
    Excel = 디렉토리 위치에 엑셀 파일 READ
CATCH: 디렉토리 정보가 유효한가?
    PRINT(경로 에러)
CATCH: I/O가 올바르게 실행되었는가?
    PRINT(I/O 에러)

IF Excel == 'Receiving' 그리고
1주일 이내 실행 안됨:
    RETURN Excel, Excel Calculator 요청
ELSE IF Excel == 'Price' 그리고
1달 이내 실행 안됨:
    RETURN Excel, Excel Calculator 요청
EISE:
    Binary Back-off 동안 대기
    Excel Checker 다시실행
    
```

Fig. 6 Excel checker algorithm

(b) Excel Calculator

Excel Calculator는 Excel Checker에 의해 요청되며 Fig. 7과 같은 Receiving 관련 결과 파일을 생성한다. Excel Calculator는 Excel Checker

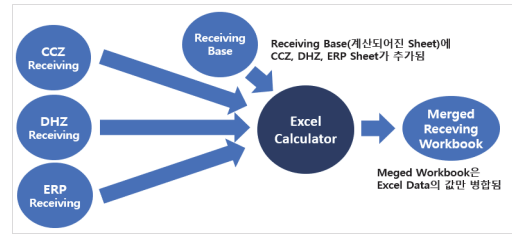


Fig. 7 Merged receiving workbook

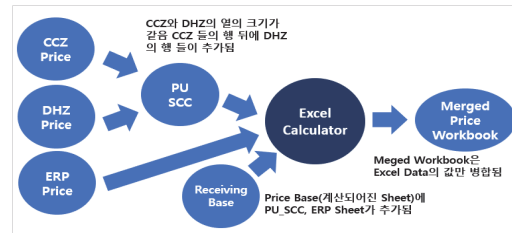


Fig. 8 Merged price workbook

에서 Read된 CCZ_Receiving, DHZ_Receiving, ERP_Receiving, 파일에서 각각 CCZ, DHZ, ERP sheet를 뽑아내어 Receiving_base에 붙여 넣는다. 그리고 Receiving_base의 출하대비 입고실적을 비교하기 위한 요청된 함수식을 계산한 후 그 값을 적용하여 중간 산출물인 Merged Receiving Workbook을 생성한다.

그 후, Fig. 8과 같이 Price 관련 결과 파일을 생성한다. Excel Calculator는 Excel Checker에서 Read된 CCZ_price, DHZ_price, ERP_price, 파일에서 각각 CCZ, DHZ, EPR sheet를 뽑아내서 낸 후, CCZ와 DHZ를 합쳐 PU_SCC라는 sheet를 만들고 EPR와 PUSCC sheet를 Price_base에 붙여 넣는다. 그리고 price_base에 단가변경을 확인하기 위해 요청된 함수식을 계산한 후 그 값을 적용하여 중간 산출물인 Merged Price Workbook을 생성하게 된다.

(c) Excel Styler

Excel Styler은 Excel Calculator에 의해 생성된

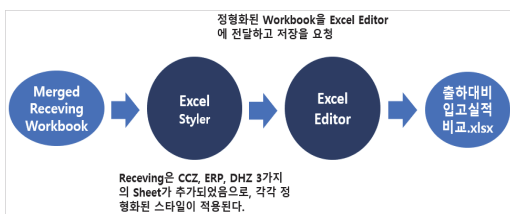


Fig. 9 Excel styler for MRW

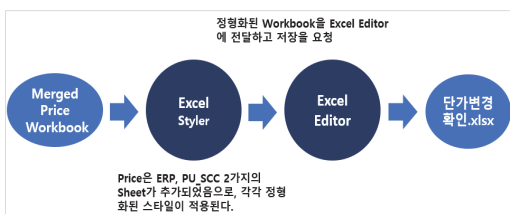


Fig. 10 Excel styler for MPW

Merged Receiving Workbook (MRW)과 Merged Price Workbook (MPW)을 사용하여 보기좋은 결과출력을 지원하여 구성하는 역할을 한다. 즉, MRW와 MPW는 Excel Calculator가 생성한 ‘중간 산출물’들의 단순 데이터 병합으로 구성되기 때문에 Fig. 9와 Fig. 10과 같이 MRW와 MPW는 각각의 Sheet 별로 정형화된 형태로 스타일을 재구성되며, 최종적으로 Excel Editor에 ‘결과 파일’ 저장을 요청하게 된다. Excel Editor는 요청받은 ‘결과 파일’을 File System의 특정 디렉토리에 저장하여 수행된 최종결과를 완성한다.

5. RPA 소프트웨어 개발 결과

본 연구를 통해 (주)신스윈의 요구사항을 반영하여 제한한 RPA 소프트웨어 디자인을 바탕으로 실제 구현하였으며, 개발 언어, 개발환경, 사용 API는 아래와 같다.

- 개발 언어: Python

- 개발 IDE: PyCharm, jupyter, Anaconda
- 사용 API: numpy, openpyxl, panda,
- OS: Window 10 Hone
- CPU: Intel Core i7-4790, 8GB Memory

5.1 RPA 소프트웨어 소스코드 및 UI

(주)신스윈의 요구사항을 반영하고, 개발을 위한 세부 구현디자인의 반영한 소스코드 Fig. 11과 같으며 해당 소스를 컴파일하여 수행한 RPA 소프트웨어의 UI (User Interface) 모습은 Fig. 12와 같다. 개발된 RPA 소프트웨어는 입력화일 및 출력화일의 디렉토리 위치를 지정할수있으며 수행과정을 표출하고 수행완료시 결과 Excel 파일을 지정된 디렉토리에 저장하게 된다.

data	2021-01-20 오후 9:13	파일 폴더
dist	2021-01-20 오후 9:16	파일 폴더
venv	2021-01-20 오후 9:13	파일 폴더
config_parser.py	2021-01-20 오전 11:48	Python File
excel_calculator.py	2021-01-20 오후 12:57	Python File
excel_checker.py	2021-01-20 오전 11:48	Python File
excel_editor.py	2021-01-20 오전 11:48	Python File
excel_styler.py	2021-01-20 오전 11:02	Python File
main.py	2021-01-20 오후 12:57	Python File
sincewin.ico	2021-01-20 오전 10:40	아이콘

Fig. 11 Developed source files for RPA S/W



Fig. 12 RPA software for sinceWin Co. Ltd.

5.2 RPA 소프트웨어 입력 및 출력파일

RPA 소프트웨어에 입력되는 파일은 Fig. 13과 같이 Price, Base, Receiving 파일들이다. 해당 입력파일들의 디렉토리를 개발된 RPA 소프트웨어에 입력으로 설정하고 해당 소프트웨어를 수행하면, Fig. 14와 같은 결과 Excel화일이 구성되어 아웃풋으로 출력되게 된다. 해당 결과화일은 (주)신스윈에서 요구했던 사항들이 반영된 내용들로 구성되며, 실제 부품생산시 원처에서 발주된 부품별 요청사항과 현재 회사의 재고를 고려하여 필요한 추가 제작 부품들을 생산하도록 기존의 양식에 맞추어 자동으로 구성되어 생산라인으로 전달되게 된다.

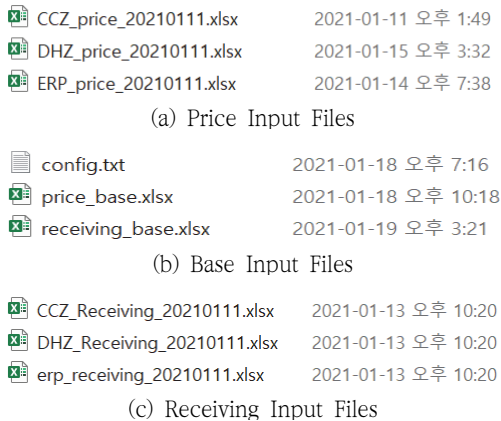


Fig. 13 Input files for RPA S/W



Fig. 14 Final result files produced by RPA S/W

6. 결론

RPA 자동화 기법은 AI기법을 활용하는 Smart Factory나 전체적인 공정과정의 수정없이 단순 반

복된 업무나 필요한 부분을 자동화하기에 적합한 방식이다. 본 연구에서는 RPA 소프트웨어의 적용 가능성을 확인하기 위하여 RPA 기법을 경남 창원 소재의 제조업체인 (주)신스윈에 실제 RPA 소프트웨어를 구현하여 적용하였다. 해당 RPA 소프트웨어를 구현하기 위해서는 먼저 해당 업체의 전체 제조과정 중 어느 부분에 RPA 기법이 적용가능한지 분석하고, 해당 부분을 구현함에 있어 해당업체의 요구사항을 반영하였다. 실제 구현된 RPA 소프트웨어는 주기적으로 수동으로 관리되었던 주문 발주 및 생산공정에서 재고를 반영한 생산량 계산을 위한 Excel화일의 자동화를 구현하였다. 이를 통해 인력의 변경에 따른 유지보수의 어려움, 수동 입력으로 인한 휴먼에러 가능성을 낮출 수 있었으며, 실제 제조현장에서도 RPA 자동화의 적용가능성을 확인할 수 있었다. 따라서 앞으로 다양한 분야에서의 RPA 자동화가 접목 및 운영이 기대된다.

사 사

이 논문은 2021~2022년도 창원대학교 자율연구과제 연구비 지원으로 수행된 연구결과임.

참고문헌

- [1] Wil M. P. van der Aalst, Martin Bichler, and Armin Heinzl, "Robotic Process Automation," Business & Information System Engineering, vol. 60, no. 4, pp. 269-272, (2018).
- [2] Somayya Madakam, Rajesh M. Holmukhe, and Durgesh Kumar Jaiswal, "The Future Digital Work Force: Robotic Process Automation (RPA)," Journal of Information Systems and Technology

- Management, vol. 16, pp. 1-17, (2019).
- [3] Jaewook Choi and Myungsik Yoo, "A study on the introduction of RPA (Robotic Process Automation) and improvement of efficiency," Proceedings of Symposium of the Korean Institute of communications and Information Sciences, pp. 378-380, (2021).
- [4] Ji-won Kang, Hee-kyung Kim, Min-Gi Choi, Hun Choi, and Seong-Yeol Yoo, "RPA Analysis and Implications in the Era of the 4th Industrial Revolution," The Korea Institute of Information and Communication Engineering, vol. 25, no. 1, pp. 317-319, (2021).
- [5] Y. Hyun and J. Lee., "Trends Analysis and Future Direction of Business Process Automation, RPA(Robotic Process Automation) in the Times of Convergence," Journal of Digital Convergence, vol. 16, no. 11, pp. 313-327, (2018).
- [6] Ki-Bong Kim, "A Study of Convergence Technology in Robotic Process Automation for Task Automation," Journal of Convergence for Information Technology, vol. 9, no. 7, pp. 8-13, (2019).
- [7] Santiago Aguirre and Alejandro Rodriguez, "Automation of a Business Process Using Robotic Process Automation (RPA): A Case Study," Proceedings of Applied Computer Sciences in Engineering, pp. 65-71, (2017).
- [8] Young Geun Hyun¹, Joo Yeoun Lee^{2*}, "A Study On The Application of RPA(Robotic Process Automation) For Productivity Of Business Documents," Journal of Digital Convergence, vol. 17, no. 9, pp. 199-212, (2019).
- [9] Ssu Chieh Lin, Lian Hua Shih, Damon Yang, James Lin, and Ji Fu Kung, "Apply RPA (Robotic Process Automation) in Semiconductor Smart Manufacturing," Proceedings of e-Manufacturing & Design Collaboration Symposium, pp. 1-3, (2018).
- [10] Andreas M. Radke, Minh Trang Dang, Albert Tan, "Using Rorotic Process Automation (RPA) to Enhance Item Master Data Maintenance Process," LogForum, vol. 16, no. 1, pp. 129-140, (2020).