

호출기 자동 생산 제어 프로그램 언어 구현

전세중*, 이건영
광운대학교 전기공학과

Implementation of a process control language for pager manufacturing

Se Jung Jeon, Keon Young Yi
Dept. of Electrical Eng. Kwangwoon University

Abstract - Implementation of a process control language for a pager manufacturing is dealt in this paper. The pager manufacturing process is consisted of a tune and an exam part. In the tune part, three capacitor trimmers are tuned to optimize the state of the pager. In the exam part, message receiving status with the arbitrary calling is tested using a vision system. The program has the ability to reuse the address discarded in the exam part when the pager testing is not working properly even though there exist the processing gaps between two processes.

The system is composed of a personal computer(PC586) and TEM-cells, Flex pager testers, an oscilloscope, a camera, and actuators. Visual Basic running on the Windows 95 is used to implement the control software which has the GUI to make an operator convenient. The validity of applying the completed program to practical machines, which are developing in the KITECH, is shown by using the virtual machine.

1. 서론

본 논문은 호출기 자동 생산 공정을 제어하기 위한 프로그램 언어의 구현을 목표로 한다. 기존의 공정에서는 호출기의 신호 파형을 분석할 수 있는 전문가가 직접 공정을 관리하여야 하며, 호출기의 라이프사이클이 짧은 데 반하여 모델 변화에 따라 작업자를 재교육해야 하는 문제로 모델 변화에 신속한 대응이 어려웠다. 따라서 본 연구에서는 기존의 생산 공정을 분석하여 전체 공정을 자동화 할 수 있는 공정 제어 언어를 정의하고, 개발된 기계 장치에 적용될 수 있는 호출기 생산 공정 제어기 및 공정제어 언어를 개발하고자 하며, 비 숙련자도 간단한 교육만으로 쉽게 작업할 수 있도록 하여 생산성 증대 및 불량률 감소를 꾀하고자 한다.

제어 프로그램 언어를 구현하는데 있어서 중요한 요소는 사용자가 하나의 작업 명령을 쉽게 작성 또는 수정할 수 있으며, 확장성을 가지고 있게 하여야 한다는 것이다[1],[2]. 즉, 전문적인 프로그래머나 시스템 사용자가 아닌 일반 사용자도 작업 지시를 위한 프로그램 편집이 가능하도록 하는 것은 매우 중요한 문제이다. 이를 위해 메뉴 방식 또는 사용자 접속(GUI)방식의 프로그래밍 기법이 많이 이용되고 있다[3].

본 연구에서도 윈도우 95와 비주얼베이직에서 제공하는 사용자 접속 방식을 사용하여 이를 구현하였다. 즉, 공정 프로그램이 텍스트 형식의 프로그램 언어로 구성되어 있어 호출기의 모델 변화에 유연하게 대처할 수 있고, 각 언어에서 필요한 파라미터의 선정이 메뉴 방식으로 되어 있어 공정 프로그램의 편집 또한 간편하게 할

수 있다.

본 논문에서는 전체 시스템과 공정 제어 언어의 구성 그리고 실제 프로그램을 작성하는 과정 및 결과를 설명한다. 마지막으로, 개발된 프로그램의 특징과 개선 사항을 논의한다.

2. 본론

2.1 호출기의 자동 생산 시스템 구성

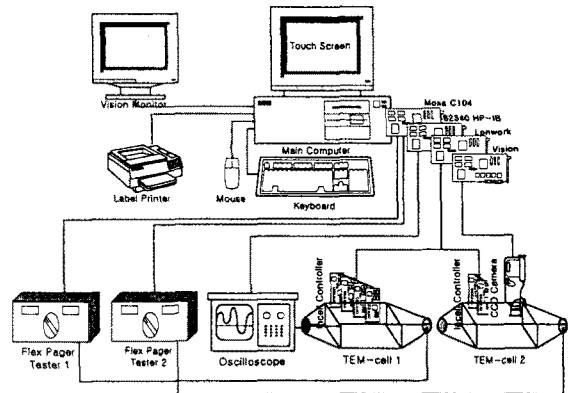


그림 1. 시스템 구성도

Fig. 1 The block diagram of the system

호출기 자동 생산을 위한 전체 시스템은 그림1과 같다. 제어 프로그램을 관장하는 주 컴퓨터 내에는 4개의 인터페이스 카드가 있어 각 계측 장치 및 기계 장치와 실시간으로 데이터를 처리한다. 외부의 RF신호로부터 차폐된 조정 및 검사 공간을 제공하는 2대의 템셀(TEM-cell)이 있다. 템셀 1의 내부에는 론웍(Lonwork)에 연결된 Incell-Controller가 호출기의 버튼을 누르는 공압 실린더 또는 캐패시터 트리머를 조절하는 모터를 구동한다. 템셀 2에는 카메라가 장착되어 비전 시스템 카드를 통하여 호출기 화면을 분석한다.

Flex pager tester는 주 컴퓨터의 제어 명령에 의해 RF신호를 발생하여 템셀 내부에 있는 호출기에 메시지를 전달하며, 오실로스코프는 메시지를 받은 호출기의 테스트 포인트의 신호 파형을 측정한다. 기타 외부 장치로는 불량품 목록을 출력하는 레이블 프린터, 작업자에게 유용한 인터페이스를 제공하는 터치 스크린, 호출기의 메시지 수신상태를 보여주는 비전 모니터가 있다.

2.2 공정 제어 프로그램 언어

이 장에서는 자동 조정과 검사 공정으로 나누어 설계한 공정 제어 프로그램 언어의 구성에 대해서 소개하고, 두 공정사이에서 발생 할 수 있는 어드레스의 불일치 및 불량품에 의한 불량 어드레스의 재사용 문제와 그 해결 방법에 대해서 설명한다.

2.2.1 자동 조정 공정

자동 조정 공정의 주요 기능으로는 "VCO조정", "ANT조정", "Local조정"이 있다. 이 기능은 호출기 테스트 포인트에 최적의 전압 파형이 출력되도록 캐패시터 트리머를 조절하는 것이다. 그리고, 기타 기능으로는 호출기의 진동 상태 및 음량 상태 검사, 전원 절약 모드 검사, 그리고 기구 동작 명령이 있으며, 명령어의 목록은 표1과 같다.

표 1. 자동 조정 공정 관련 명령어
Table. 1 Commands relating with Tune

명령어	작업 내용
"초기화#"	조정 시스템에 연결된 장치를 초기화
"실린더동작:L.F#"	잠금/픽스처 실린더를 잠금
"실린더해제:L.F#"	잠금/픽스처 실린더를 해제
"조정봉설치:V.L.A#"	캐패시터 트리머의 조정봉을 설치
"전원공급:N#"	호출기에 1.5V전원 공급
"전원차단#"	호출기에 공급중인 전원을 차단
"RF발생:324.475:-40#"	Radio Frequency를 발생(Flex)
"RF차단#"	Radio Frequency를 차단(Flex)
"PROG입력:PAGER#"	호출기에 프로그램을 입력
"버튼동작:R:한번(H)#"	호출기의 버튼(R)을 누름/해제
"진동검사:0.9#"	호출기의 진동 상태를 점검
"음량검사:1.5#"	호출기의 음량 출력 상태를 점검
"스윙검사:1.5:0.02:5#"	스윙 폭을 검사
"VCO조정:1.5:0.02:3#"	VCO를 측정하여 조정함
"VCO확인:1.5:0.02#"	VCO의 값을 검사
"AGC확인:1.5:0.02#"	AGC의 값을 검사
"Local조정:1.5:0.02:5#"	Local의 값을 측정하여 조정함
"ANT조정:1.5:0.02:5#"	ANT의 값을 측정하여 조정함
"BS확인:0.5#"	호출기의 소비 전류치를 측정
"일단정지#"	시스템을 일시 정지시킴
"종료#"	자동 조정 공정을 마침

위의 명령어들의 문법 표현은 " "사이에 실행문과 인수로 나누어 기술하고 있다. 명령어의 앞부분에 해당하는 실행문은 자연어와 거의 유사한 형식을 가지고 있고, 명령어의 뒷부분에 해당하는 인수는 호출기의 최적상태 및 불량품 판정의 기준이 된다.

위 명령어 중에서 자동 조정 공정의 주요 기능의 하나인 "Local조정"의 기능을 예로 들어 설명하면 다음과 같다. 주 컴퓨터는 콘솔에 "Local조정"명령을 전달하여 명령의 해당 트리머 조절을 준비시키며, GPIB 명령을 이용하여 오실로스코프로부터 해당 측정 신호의 상승 시간과 하강 시간을 읽어낸다. 읽어낸 상승 및 하강시간은 콘솔에 전달되며, 이를 기준으로 캐패시터 트리머가 조정된다. 이어서, 주 컴퓨터는 콘솔로부터 트리머 조절후의 값을 전달받고, 이를 설정 값과 비교하여, 그 결과가 설정된 한계값을 만족하지 않은 경우에는 앞서의 명령을 계속 반복한다. 만일 "Local조정"명령의 마지막 인수인 5초 내에 조정이 완료되지 않으면 타임아웃이 발생하여 작업자에게 작업 취소 여부를 묻는다.

```
"TProject File - 9232837"
"초기화#"
"실린더동작:L.F#"
"조정봉설치:V.L.A#"
"전원공급:N#"
"RF발생:324.475:-40#"
"PROG입력:PAGER#"
"Local조정:1.5:0.02:5#"
"PROG입력:FINAL#"
"RF차단#"
"전원차단#"
"실린더해제:L.F#"
"종료#"
```

그림 2. Local조정 프로젝트 파일
Fig. 2 The project file of Local tuning

그림 2는 "Local조정" 작업을 수행하기 위해 작성된 프로젝트 파일이다. 이 프로젝트 파일은 그래픽 사용자 메뉴 방식의 프로젝트 작성 마법사에서 파라미터를 설정

함으로써 쉽게 작성할 수 있으나, 호출기 생산 공정의 흐름을 아는 전문가가 메모장을 이용하여 직접 작성할 수도 있다. 작성한 프로젝트파일의 실행은 2.3절에서 설명한다.

위의 예제 파일에서는 "Local조정"작업 하나만을 나타내었지만 생산 공정에 따라 "VCO조정", "ANT조정", 그리고 기타의 적절한 명령어를 추가로 사용하여 프로젝트 파일을 작성하여야 한다.

2.2.2 검사 공정

검사 공정의 주요 기능으로는 "화면비교", "Address 검사"가 있고, 기타 기능으로는 "라이트검사", "LED검사", "LABEL출력" 등이 있다.

"화면비교"의 기능에서는 그림1의 Flex pager tester 2에서 RF(Radio Frequency)를 발생한 후, 탭셀 2에 장착된 카메라를 이용하여 메시지 출력 이미지를 얻는다. 현재의 수신 이미지와 정상적으로 동작한 이미지(사전에 저장하여야 함)를 비전 처리하여 일치하지 않으면 불량으로 판정한다.

위에서 설명한 검사 공정을 수행하기 위해 필요한 명령어의 목록은 표2와 같다. 표에 나타난 명령어들은 탭셀 2에 연결된 장치에 전달된다는 사실 외에는 자동 조정 공정의 명령어와 동일한 형식으로 구성되었다.

표 2. 검사 공정 관련 명령어
Table. 2 Commands relating with Exam

명령어	작업 내용
"초기화#"	검사 시스템에 연결된 장치의 초기화
"실린더동작#"	잠금 실린더의 동작
"실린더해제#"	잠금 실린더의 해제
"RF발생:324.475:-80#"	324.475MHz, -80dBm의 RF발생
"RF차단#"	Radio Frequency를 차단
"버튼동작:S:한번(H)#"	호출기의 버튼(S)을 누름/해제
"화면읽음:3#"	호출기 수신 화면 촬영(카메라 이용)
"화면비교:Half패턴:3#"	호출기의 메시지 수신 화면을 비교
"화면저장:Half패턴:3#"	현재의 수신 화면을 디스크에 저장
"Address검사:2#"	호출기의 어드레스를 검사
"라이트검사:3#"	호출기의 라이트 상태를 검사
"LED검사:5#"	호출기의 LED 상태를 검사
"LABEL출력:3#"	불량품 목록을 레이블로 출력
"일단정지#"	시스템을 일시 정지시킴
"종료#"	검사 공정을 마침

위의 명령어 중에서 검사 공정의 주요 역할을 하는 "화면비교"의 기능을 예로 들어 설명하면 다음과 같다. 주 컴퓨터는 비전 시스템에 "화면비교"명령을 전달하고, 비전 시스템에서 이전 명령어(화면읽음)에 의해 얻어낸 이미지와 인수로 전달받은 이미지(사전에 디스크에 저장되어야 함)를 비교한 결과를 리턴한다. 비전 시스템에서 3초내에(명령어 인수 2) 응답이 없는 경우에는 타임아웃이 발생하여 작업자에게 작업취소 여부를 묻는다.

2.2.3 두 공정간의 인터페이스

두 공정이 하나의 시스템에서 서로 독립적으로 진행됨으로써 발생하는 어드레스의 불일치, 검사 공정에서 불량품이 발생할 때 앞 공정에서 프로그램된 어드레스를 재 사용해야 하는 문제점이 나타난다. 이 문제는 두 개의 링 버퍼 큐(Queue)와 큐 Handler에 의해 처리한다.

그림3에 나타난 것처럼 조정 공정을 시작 할 때 초기화된 현재의 어드레스가 조정 공정을 마치면 현재큐(Cur_Queue)에 저장되는데, 만일 현재큐가 모두 찬 경우에는 조정 공정을 일시 중지하며 검사 공정이 완료되어 큐에 여유가 생기도록 한다. 그리고, 다음 조정 공정을 위한 새 어드레스를 위하여 불량큐(Bad_Queue)에서 어드레스 하나를 가져와 다음 어드레스로 정한다. (없는 경우는 자동 산출함)

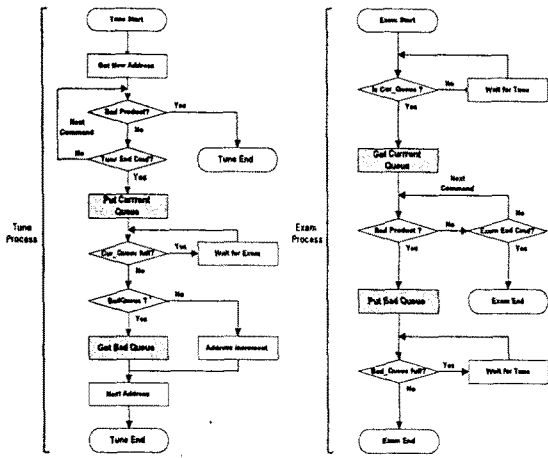


그림 3. 큐 및 큐 핸들러
Fig. 3 Queue and Queue Handler

검사 공정이 시작되면 현재큐에서 어드레스 하나를 가져와 검사를 시작하는데, 만일 현재큐가 비어있는 경우에는 검사 공정을 일시 중지하며 조정 공정이 완료되기를 기다린다. 검사공정에서 불량 발생하면 불량큐에 어드레스를 저장하는데, 만일 불량큐가 모두 찬 경우 검사 공정을 일시 중지하며 조정 공정이 완료되어 불량큐에 여유가 생기도록 한다.

2.3 실험 및 결론

그림 4는 구현된 프로그램의 주 화면이며, 좌우로 나누어진 시스템은 다시 작업상태, 설정, 작업기록, 그리고 프로젝트로 나누어진다. 작업 상태에서는 진행중인 작업상태를 텍스트 형식으로 작업자에게 제공하며, 설정 부분에서는 작업을 수행하는데 필요한 어드레스나 불량함 목록을 설정한다. 작업 기록은 작업 수행 중 발생하는 불량품 발생시간 및 발생 항목에 대한 정보를 가지고 있다. 기타의 기능은 그림 5와 같이 구성되어 있다.

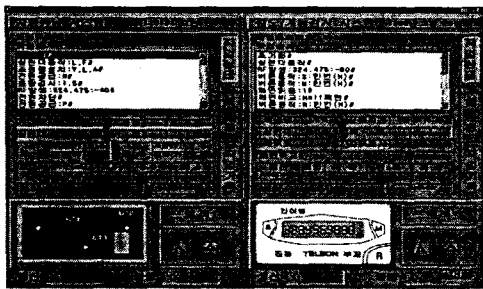


그림 4. 프로그램의 주메뉴
Fig. 4 Main menu of the program

시작 버튼에 의하여 진행되는 공정은 아래와 같이 진행된다.

- 수 작업: 작업자는 조립된 호출기를 템셀 1(조정)에 넣는다.
- step 1: 그림4의 중앙 하단에 있는 "프로젝트"버튼을 눌러서 기존의 프로젝트 파일을 불러오거나, 프로젝트 작성 마법사를 실행하여 프로젝트 파일을 작성한다.
 - step 2: 작업자는 주 화면의 중앙 하단에 있는 프로젝트(자동 조정) "시작"버튼을 누른다.
 - step 3: 그림4의 좌측 상단에 있는 작업 표시창의 명령어를 행 단위로 계속 장치나 기계장치에 전달한 후 응답 대기 상태로 들어간다. 이때, "시작"버튼은 "중지"버튼으로 바뀐다.
 - step 4: 장치의 응답에 따라 적절한 조치(트리머 조정, 불량품 판정)가 취해지고, 다음 명령으로 분기한다. 설정 시간 내에 장치에서 응답이 없는 경우, 타임아웃을 발생하고 작업자에게 작업취소 여부를 요구한다.
 - step 5: 프로젝트 파일의 "종료"명령을 처리할 때까지 작업 창에서 한 스텝씩 증가하여 step 3-4를 반복한다. 만일 조정 불량이 발생하면 작업자에게 불량처리 여부를 요구한다.
 - step 6: "종료"가 실행되면 작업 표시 창은 다시 처음으로 되돌아가 반전 상태에 있으며, 작업자에게 작업 종료로 알려준다.
- 수 작업: 작업자는 템셀 1내에 있는 호출기를 꺼내어 템셀 2 또는 버퍼에 넣는다.

검사공정의 진행도 위와 동일한 과정으로 진행되며, 공정이 이루어지는 템셀과 명령이 전달되는 장치만 다르다.

3. 결 론

본 논문에서는 비주얼베이직을 기반으로 하여 호출기 자동 생산을 제어하는 프로그램 언어를 설계하였다. 개발된 프로그램 언어는 구현 시간이 짧으며, 모델의 변화에 유연하게 대처할 수 있는 장점을 가진다.

개발된 공정 제어 프로그램은 다중 처리 프로그램 방식을 택하였고, 공정이 서로 독립적으로 수행되기 때문에 발생하는 어드레스 부여의 문제는 Queue Handler를 이용하여 해결하였다.

개발된 프로그램은 KITECH에서 개발중인 장치에 적용하기 위한 단계에 있으며, 앞으로 프로그램과 장치를 조립하는 것에 관한 연구가 진행중이다.

(참 고 문 헌)

- [1] S. Bonner and Kang G. Shin, "A Comparative Study of Robot Languages," IEEE Computer, pp. 82-96, 1982
- [2] 여희주, "지능형 로봇 시스템을 위한 다중 센서 융합에 의한 Task-Level Robot Language의 설계 및 구현," 대한전기학회 논문지, 45권, 6호, pp. 899-909, 1996
- [3] Richard A. Volz, "Report of the Robot Programming Language Working Group: NATO Workshop on Robot Programming Languages," IEEE J. Robotics and Automation, Vol. 4, No. 1, pp. 86-90, February 1988

호출기 자동 생산 제어 프로그램

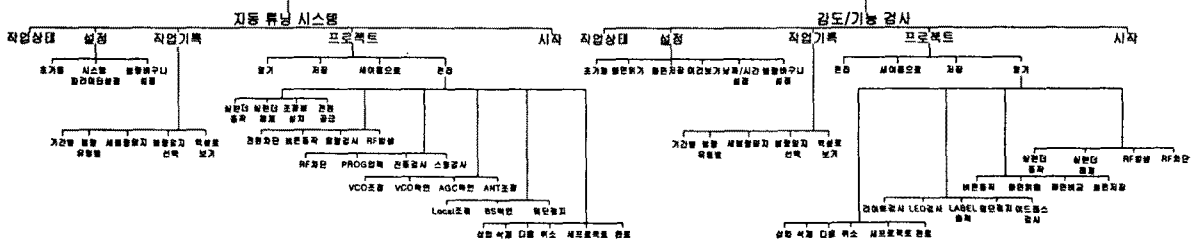


그림 5. 공정 제어 프로그램 구조도
Fig. 5 The structure of the program