

고속 인쇄기의 레지스터 컨트롤러에 관한 연구

*장 중 학, 이 덕 형, 홍 선 기
호서대학교 정보제어공학과 서보기기 및 제어 연구실

A study for register controller of high speed printing machine

Joong Hack Jang, Duck Hyoung Lee, Sun Ki Hong
Department of Information & Control Eng Hosda University

Abstract - Existing a high-speed printer register controller have used a foreign country goods of the high price. It established the plan to be and progressed to progress research. It find out about former time's a register control system's action, and determined the processor to DSP. It studied the algorithm to consider various situations. It completed finally a simulator board of a register controller of the high capacity.

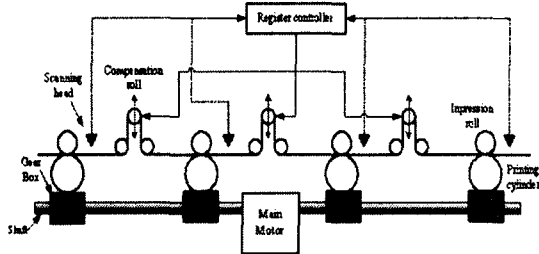
1. 서 론

기존의 인쇄기에서 사용되던 고가의 외국 제품 내에 있는 register controller를 저가로 구현 대체하면서 현재 일반적으로 사용되고 있는 속도보다 빠른 속도의 인쇄기에 적용할 수 있도록 거기에 맞는 register controller의 개발에 중점을 둔다. Scanning head에서 register mark를 인식하고 이 오차신호를 이용하여 register controller는 오차에 해당하는 보정신호를 PPC에 보내주게 된다. 이 신호를 받은 PPC는 각 색선의 register controller에게 보정 명령을 내리고 이러한 과정을 통해서 register error를 보정하게 된다. 여기에서 우리의 연구는 이미 상용화 되어 있는 register mark 인식 센서를 이용하여 register error 신호를 받아 들여서 각 모터를 실제 구동하게 해주는 PPC에게 신호를 주는 register controller를 개발 하는 것이 목표이며, 현재 고성능의 DSP를 이용하여 기존의 register controller에 준하는 성능이나, 혹은 보다 좋은 성능을 내는 컨트롤러가 개발 될 것으로 예상된다.

2. 본 론

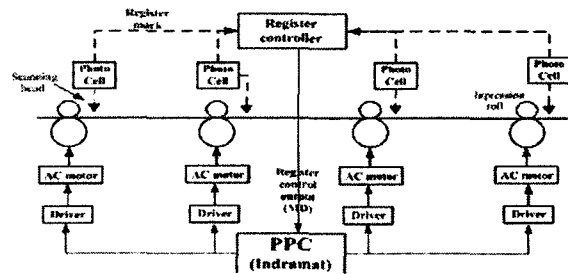
2.1 Register controller의 기본 동작

기존 타입의 register controller logic은 그림1과 같다. 그림의 main motor를 통하여 각 실린더는 하나의 축으로 연결되어 동일한 속도로 구동하게 된다. 이 때 각각의 실린더에 scanning head가 설치되어 각도마다 인쇄 위치를 기록하게 된다. register controller에서는 각 도에서 register mark를 받아 error를 결정한다. 이 때 error는 1도와 2도 혹은 2도와 3도와 같이 인접한 unit에 대하여 계산하게 된다.



<그림 1> 축방향 타입의 register controller 개념도

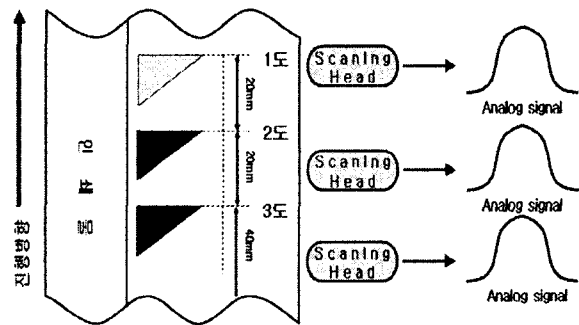
최근의 인쇄기에서는 축방향 타입보다 sectional 타입의 인쇄기가 많이 사용되고 있다. sectional 타입의 인쇄기란 main motor를 축에 물려 동일한 구동방식에서 변화되어 축을 없애고 각 도를 개별적인 motor로 구동하는 방식이다. 또한 각 도마다 개별적인 속도제어가 가능하기 때문에 compensator roll도 없어진다. 각각의 도를 구동하기 위한 motor를 드라이버를 사용하여 구동을 하고 이 motor는 PPC를 통해 제어 된다. 각 도의 1회전 마다 광센서를 통해 원단에 인쇄되는 register mark를 입력 받고 error를 생성후 인접한 도와 비교를 통해 오차를 계산한 다음 network를 통하여 register controller와 PPC로 전송하게 된다. register controller는 전달 받은 오차의 양을 출력하여 인쇄 상황을 파악할 수 있게 하며 PPC는 전달 받은 오차의 양 만큼 각 도의 상을 조절하여 오차를 보정하게 된다. 보정 방식은 register controller의 pulse 형태로 보정파형을 받은 다음 PWM을 이용하여 미리 정해져 있는 각도량 만큼 각 도를 조정하게 된다. 정해져 있는 보정의 양은 센서의 타입이나 성능등을 고려하여 조정 된다.



<그림 2> Sectional driver system의 개념도

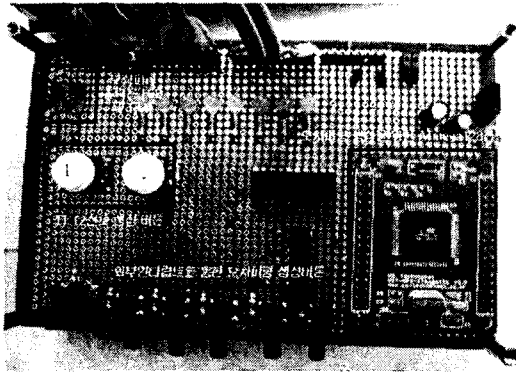
2.2 register mark에 관한 입력 설계

기존의 타입보다 발전된 성능의 register mark를 설계하기 위해서는 기존 방식의 입력되는 센서의 값을 시뮬레이션 하고 그 값에 대한 현재 시스템의 정복이 필요하다. 따라서 register mark에 대한 입력에 관한 시뮬레이션 기능을 하는 새로운 시뮬레이터가 필요하고 입력에 대한 정확한 이해와 정보도 필요하게 된다.

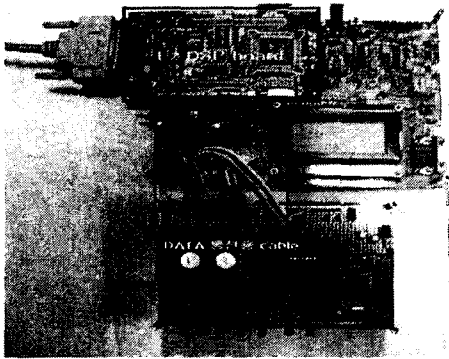


<그림 3> register mark의 입력 및 센서 신호 인식 개념도

그림 3에서 보는 바와 같이 scanning head의 신호를 받는다. 각각의 register mark의 간격은 20mm의 거리로 떨어져 있고 scanning head의 거리도 20mm로 떨어져 있어 동일한 시간대에 신호가 나오면 오차가 없는 것이고 차이가 있으면 오차 값이 발생하는 것이다. 이 신호를 그림에서 보는 바와 같이 analog 신호로 출력을 해주는 방식이다. 이와 같은 파형의 신호 일 지라도 pulse파형으로 바뀌기 때문에 DSP에서 인식하는 것에는 문제가 없게 된다. 버퍼 IC를 이용하여 간단하게 구형파로 바꿀 수 있다. 이 때 인쇄기의 속도를 계산 하고 그 속도에 의해 파형의 폭과 파형과 파형의 간격을 결정하여 실제 시스템에 가장 근접한 신호를 만들게 된다. 이러한 일련의 동작들을 구현하기 위해 확장 보드를 설계하여 실제와 가장 근접한 시뮬레이터를 제작하게 되었다 그림 4는 제작된 시뮬레이터 보드와 실제 보정 파형을 생성해주는 register controller system을 연결해 놓아 실제 상황에 관한 시뮬레이션 및 보정 구현을 할 수 있게 한 것이다. register mark의 입력을 정확하게 처리하기 위해 DATA cable을 구성하여 이 cable을 통하여 신호를 받게 되는 부분을 처리하게 된다. 다시 말해 여러 가지 오차 파형을 받아 오차의 양을 측정하고 그 양을 PPC로 보내주기 위해 register controller를 구성한다. 그 구성을 위해 DSP중 TMS320F2812를 이용해 시스템을 구성하게 된다. 외부 인터페이스 보드 부분을 인쇄기라 가정하고 가상의 시뮬레이션을 구성하여 각각의 속도마다 나오는 오차의 폭과 각각의 오차마다 나오는 차이 등을 고려하여 여러 가지 시뮬레이션을 고려해 보았다. 외부 인터페이스 보드에서는 ATmega128을 이용하여 각 상황에 적합한 일들을 가능하게 하고 경우의 수를 구성하여 scanning signal인 T1과 T2신호를 생성하게 된다. 이러한 준비를 마치고 알고리즘을 정복시켜 실제 상황과 똑같은 상황을 구현함으로써 인쇄기에 바로 적용할 수 있는 상태가 되게 한다.



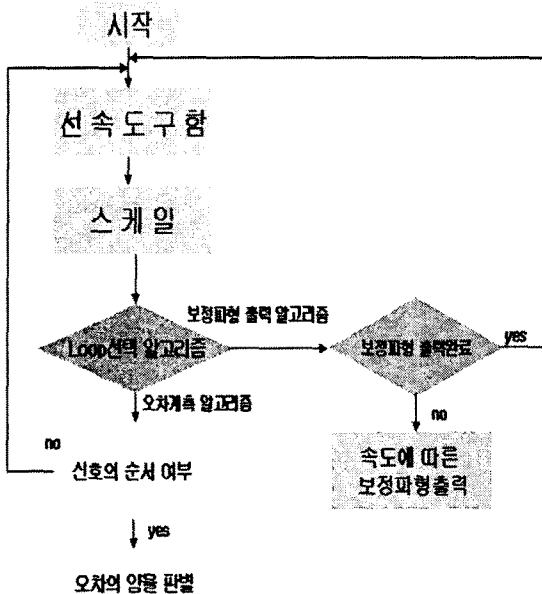
〈그림 4〉 가상의 시뮬레이션 보드



〈그림 5〉 가상의 시뮬레이션 보드와 register controller 연결 모습

외부 인터럽트 1을 이용하여 rising edge를 받아서 GP timer 1을 이용하여 T1과 T2가 들어오는 시간을 인지하고 오차를 알아내어 거기 따른 보정 파형을 출력하는 방식을 채택하였다. GP timer3를 이용하여 보정 신호의 출력파형의 duty비를 정확하게 조절하는 방법을 택하였다. T1과 T2의 신호에서 받은 시간 값을 빼어 오차를 알아내고 거기에 scale을 통한 보정파형 변수를 생성한 다음 GP timer 3를 이용하여 보정파형 변수와 비교를 해서 정확한 duty비를 알아내 GPIO로 출력을 하게 된다.

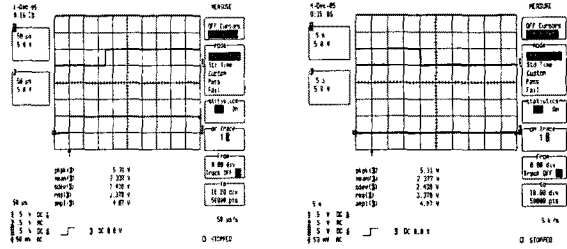
2.3 프로그램 알고리즘



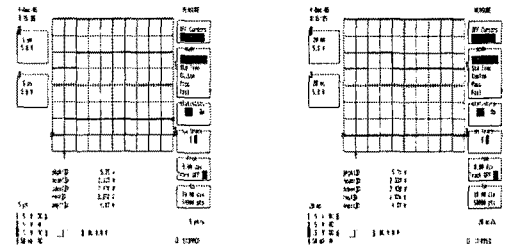
〈그림 6〉 프로그램 알고리즘

위의 알고리즘을 이용하여 여러가지 경우의 수를 따져 오차에 따른 보정 신호 출력에 대해 오실로스코프로 확인을 해보았다. 각각의 시

간 주기는 최소 오차 범위부터 1mm 오차가 낮을 경우 1sec의 보정 신호를 나타내는 것을 기준으로 GPIO신호를 결정 하였으며 초기 오차 신호는 앞에 언급한 바와 같이 ATmega 128을 이용해 버튼 이벤트로 외부 인터럽트를 사용해서 만든 파형이다. 그림 37은 최소 오차 범위인 5us의 오차가 난 경우를 ATmega128로 구현한 것을 스크프로 확인해본 그래프이다. 그림 7의 왼쪽 같은 오차가 발생하였을 경우 main loop 알고리즘을 통해 생성된 파형은 그림 7의 오른쪽에 나타내었다. 5us는 오차 신호의 최소 범위로 5us의 신호를 처리함으로써 그 다음 그보다 더 큰 오차가 발생되었을 경우에도 이상 없이 오차 시간을 읽어와 어느 정도의 오차가 낮았는지 알 수 있게 된다. 오차 파형의 생성은 50ms를 주기로 발생이 되며 오차가 발생한 후 보정이 완료 될 때까지 에러를 검출한다 하더라도 보정 파형은 출력하지 않도록 하였다.



〈그림 7〉 125us 오차 신호 및 보정 출력 파형 그래프



〈그림 8〉 5us 오차 신호 및 보정 출력 파형 그래프

3. 결 론

연구를 진행하기 위해 단계적인 계획을 수립하였고 기존의 레지스터 컨트롤러 시스템의 동작과 전체 시스템의 기본적인 동작에 대해 파악하고 이를 통해서 프로세서를 선정하였다. 선정된 프로세서인 DSP를 빠른 연산능력으로 좀 더 적은 시간에 고성능의 오차 인지 및 보정신호 출력 장치를 개발하는 데 중점을 두었다. 현재의 시스템에서는 DSP의 성능을 심분 발휘하는 것은 아니지만 앞으로 여러 가지 상황등을 고려해서 DSP의 기능을 좀 더 활용하는 방안을 찾고 있다. 시뮬레이터 제작은 현재 완성된 상태로서 기존의 시스템을 부차시켜 서서히 접목시키는 방향으로 하고 있다. 실험상의 편의를 위하여 시스템으로 구성된 DSP보드 및 새로 제작된 외부 인터페이스 보드를 사용하였으나 실제 응용제품을 제작하는 최소화된 DSP모듈을 이용하여 register controller의 크기도 현재보다는 대폭 감소된 보드로 개발을 검토중에 있다. 시뮬레이터 제작과 동시에 여러 가지 신호의 의미와 입출력의 관계를 연구중에 있다. 또한 scanning head 신호의 종류 및 register controller간의 통신 문제도 명확히 규정해야 한다. 앞으로 실제 시뮬레이터의 입출력 관계를 좀 더 명확히 정의하게 되면 본 연구에 사용된 시스템과 접목하여 기존의 register controller의 역할을 대체할 뿐 아니라 좀 더 발전된 성능의 register controller로서 기능을 발휘 할 것으로 예상된다.

[참 고 문 헌]

- [1] 설승기, "전기기기 제어론", 도서출판 브레인 코리아, 2002
- [2] 백종철, "TMS320F28XX CPU 핸드북", 싱크웍스, 2005
- [3] 백종철, "TMS320F24XX- CPU 메모리 인터럽트 MAC의 이해", 싱크웍스, 2004
- [4] 백종철, "TMS320F24XX 계열을 이용한 하드웨어 설계", 싱크웍스, 2004
- [5] Texas Instruments, "TMS320F2810, TMS320F2812 Digital Signal Processor Data Manual, July 2003
- [6] Texas Instruments, "TMS320F28x Event Manager Peripheral Reference Guide, June 2003

본 연구는 성안기계㈜에 의해 지원되었으며, 이에 관계자 제위해 감사 드립니다.