

임베디드 리눅스 기반의 서버와 웹 어플리케이션을 이용한 곡물 선별 모니터링 시스템

박세현¹ · 김영욱² · 김현재^{2*}

Monitoring system for grain sorting using embedded Linux-based servers and Web applications

Se-hyun Park¹ · Young-wook Geum² · Hyun-jae Kim^{2*}

¹Department of Electronic Engineering, Andong National University, Andong 36729, Korea

^{2*}College of Engineering 1st, 106, Andong National University, Andong 36729, Korea

요 약

본 논문에서는 고속 FPGA 및 임베디드 리눅스를 사용하여 곡물 선별 모니터링 시스템을 구현하였다. 기존의 곡물 선별 모니터링 시스템은 독립형 모드에 기반으로 설계하였는데 비해 제안된 시스템은 웹 서버와 웹 응용 프로그램 기반으로 설계하였다. 제안된 곡물 선별 시스템은 웹 서버 상에서 FPGA의 고속 하드웨어 인터페이스를 설계하였다. 제안된 시스템은 리눅스 웹 서버의 멀티태스킹의 장점과 FPGA의 고속 하드웨어의 장점을 갖는다. 고속 레이트 라인 스캔 CCD 카메라의 제어 로직, 무게 중심점 추출 방법, 그리고 HSL 디코딩 및 웹 서버의 인터페이스는 FPGA로 구현하였다. 구현된 모니터링 시스템은 웹 어플리케이션에 의해 곡물 선별 모니터링과 시스템 고장 및 복구를 원격으로 제어할 수 있다는 장점을 갖는다. 그 결과, 기존의 시스템에 비해 곡물 선별 성능을 업그레이드할 수 있었다.

ABSTRACT

In this paper, we implement monitoring system for grain sorting using a high-speed FPGA and embedded LINUX. The proposed system is designed by base on web server and web-based applications while existing system was designed by base on stand-alone mode. The interface the Web server with high speed hardware of FPGA is designed on the implemented monitoring system. The proposed system has the advantages of multi-tasking on Linux web server and real-time high speed on FPGA also. The control logic of a high speed rate line-scan CCD camera, the method of center of gravity, HSL decoding and the interface on the Web server are implemented in FPGA. The implemented monitoring system has the advantage of being able to control the grain monitoring, system failure and recovery remotely by web application. As a result, we can upgrade the performance of sorting quality compared by existing system.

키워드 : FPGA, HSL 디코딩, 웹 어플리케이션, 임베디드 리눅스

Key word : FPGA, HSL Decoding, Web Application, Embedded Linux

Received 05 August 2016, Revised 29 August 2016, Accepted 12 September 2016

* Corresponding Author Hyun-Jae Kim(E-mail:guswo1209@gmail.com, Tel:+82-54-820-7945)

College of Engineering 1st, 106, Andong National University, Andong 36729, Korea

Open Access <http://doi.org/10.6109/jkice.2016.20.12.2341>

print ISSN: 2234-4772 online ISSN: 2288-4165

©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.
Copyright © The Korea Institute of Information and Communication Engineering.

I. 서 론

곡물 처리는 건조, 도정 및 선별의 과정을 거친다. 마지막 단계인 선별은 원료를 양품과 불량으로 구분하는 공정을 말한다. 이 과정에서는 원료 내 불량 곡물만 제거하는 작업이 필요한데 이러한 작업을 위해 색채 선별기를 사용하고 있다[1,2].

색채 선별기는 불량 곡물만 선별적으로 제거하는 것을 특징으로 가지며 적용하는 대표적인 곡물로는 쌀, 찹쌀, 현미 등이 있으며 최근에는 땅콩과 녹차와 같은 다양한 원료로 적용이 확대되어 가고 있다.

현재 색채 선별기를 개발하는 대표적인 업체로는 buhler, Satake, Toyo, 아이디알 시스템, 대원 GSI, 에이맥스 등이 있다.

곡물 선별기의 성능을 개선하는 연구가 다양하게 진행되고 있는데 주로 영상처리 성능 향상과 이를 모니터링 하는 기술 개발로 나뉜다[3-5].

기존에 제품화 되고 있는 색채 선별기는 임베디드 시스템으로 구현되어 있으며 FPGA에 의한 고속 선별과 산업용 PC를 이용한 RTOS 기반 모니터링 방식으로 되어 있다[6,7].

본 논문에서는 영상처리 성능 향상을 위해 기존의 RGB 선별 방식에서 벗어나 HSL Decoding 방식을 개발하였으며, 모니터링 기술 방식은 기존의 RTOS 방식을 떠나 웹 기반의 모니터링 시스템으로 개발하였다.

기존의 시스템은 선별기가 설치되어 있는 환경적(복합 공정 플랜트) 또는 위치적(해외) 제약으로 인해 제품의 수리, 제어 및 업그레이드의 경우 오프라인으로 진행하게 되어 선별기의 유지보수 비용을 상승시키는 주된 문제를 지니고 있었다.

본 논문에서는 기존 시스템의 주된 문제인 환경적 또는 위치적 제약에서 벗어나기 위해 임베디드 웹 서버 및 웹 기반 어플리케이션 개발을 통한 웹 기반의 모니터링 시스템을 제안하고자 한다.

또한 임베디드 웹에 의해 구동되는 고속 하드웨어 인터페이스를 구축하고 곡물 선별을 위한 HSL Decoding 과 곡물의 무게 중심점 검출을 FPGA로 설계하고자 한다.

제안된 시스템은 선별 감도를 원격에서 제어가 되는 것은 물론 선별기 운용 상태를 다수의 사용자가 위치에 무관하게 접근해서 모니터링하고 아울러 시스템의 고

장 및 보수를 판단 할 수 있는 장점을 가지게 한다.

II. 곡물 선별 모니터링 시스템

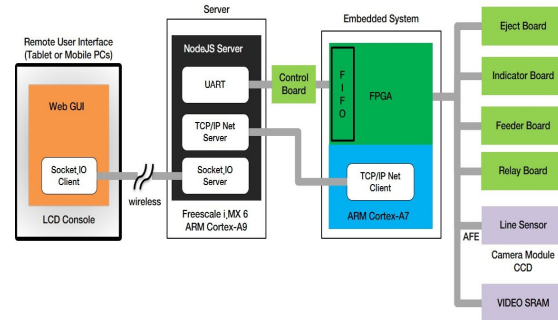


Fig. 1 Monitoring system for Grain sorting

본 논문에서 제안한 곡물 선별 시스템은 라인 센서부, FPGA부, 임베디드 시스템 구동부 그리고 웹 서버 시스템 및 LCD 콘솔부로 구성 된다.

그림 1은 곡물 선별 모니터링 시스템을 구성도로 나타낸 것이다. 곡물 선별 모니터링 시스템은 사용자의 편의를 위해 원격에서 접속한다. 이를 위해 서버 시스템인 ARM Cortex-A9은 nodejs를 사용하여 웹 서버를 구축하였고 웹 서버의 소켓 인터페이스는 사용자 UI와 socket.io로 통신할 수 있게 구축하였다.

곡물 선별 시스템의 각 하드웨어 모듈인 불량 제거 구동 보드, 임베디드 시스템의 카메라 모듈, 피더, 릴레이 보드 등은 각각 임베디드 시스템의 TCP/IP 통신을 사용하여 상태 데이터를 임베디드 웹 서버인 nodejs의 net socket 모듈로 전송한다[8].

net socket 모듈로 전송된 데이터는 socket.io를 통하여 non-blocking 방식으로 웹 기반 어플리케이션의 UI로 표시된다. 같은 방식으로 웹 기반 어플리케이션의 UI에 의해 설정된 곡물 제어 데이터는 socket.io 통하여 net socket 모듈로 데이터를 전송하며 net socket 모듈은 FPGA 하드웨어를 제어하게 한다.

모니터링 시스템과 사용자 인터페이스 서버 간의 통신은 그림 2와 같은 프로토콜에 의해서 통신이 이루어진다. 제어주소와 데이터의 관계는 JSON(JavaScript Object Notation) 방식을 사용한다.

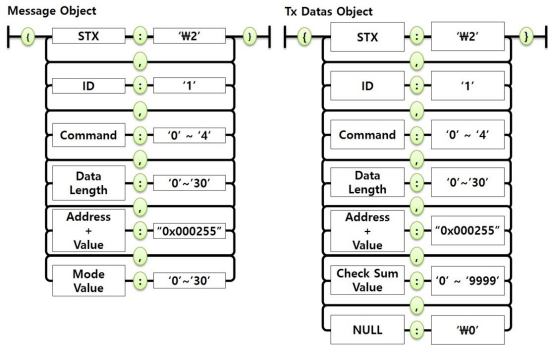


Fig. 2 Protocol of the user UI and the server

그림 3은 각각 서버와 사용자 프로그램의 초기화 과정을 나타낸 것이다.

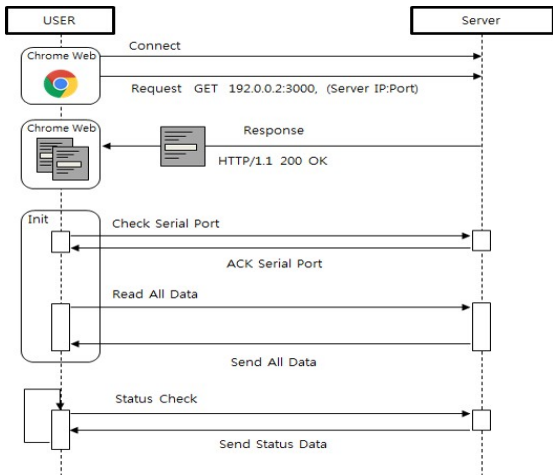


Fig. 3 Initialization process of the server and the user program

사용자 UI 프로토콜과 임베디드 시스템의 프로토콜은 setInterval 함수를 이용하여 시간 지연에 따른 전송 에러를 검출하며 패킷의 마지막에는 CRC Check로 최종 에러를 검출 한다.

III. 곡물 선별 제어기 설계

그림 4는 곡물 선별기의 제어기 하드웨어 구성을 나타낸 것이다. 제어기는 FPGA Altera Cyclone IV를 사용하였고 120MHz의 메인 클럭으로 구동 된다.

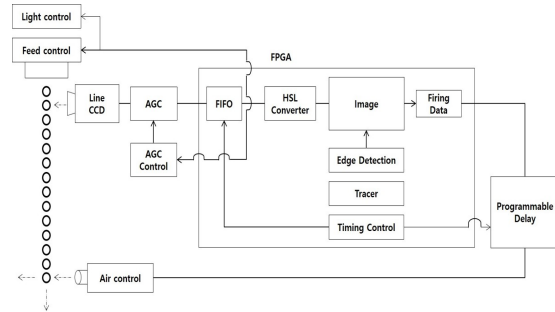


Fig. 4 The configuration of the grain sorting controller

곡물 선별기의 제어기 하드웨어를 설명하면 다음과 같다. Line CCD Camera를 통하여 Gain과 Offset을 조정 후 FIFO를 통하여 이미지를 구성한다.

Line CCD Camera를 통하여 데이터를 받을 때, 곡물의 색깔과 종류에 따라 피더의 속도와 빛의 밝기 등을 제어하게 되는데 이는 AGC를 통하여 필터를 거친 후 자동 조절된다.

이때 RGB 데이터는 HSL Decoding을 통하여 주변 밝기의 영향을 제거한 실시간 데이터의 이미지가 되며 n*n의 라인으로 구성된다. n*n의 이미지는 외곽선 추출 회로와 곡물의 무게 중심을 기준으로 필터를 통하여 곡물의 양품 혹은 불량품을 선별하게 된다. 선별 데이터는 곡물의 자유 낙하 속도에 맞추어 타이밍 컨트롤에 의하여 계산되며 미세조정을 위해 프로그래머블 딜레이를 통하여 에어건을 작동 시킨다.

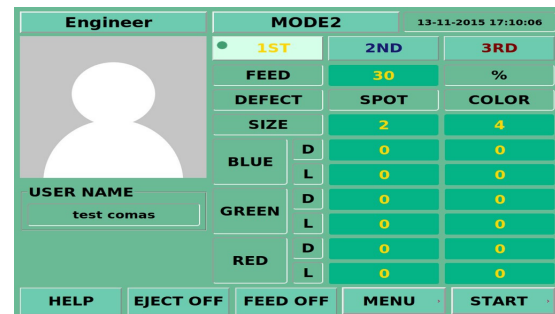


Fig. 5 The main screen of the user interface

그림 5는 사용자 인터페이스의 메인 화면이며 기존의 RTOS에서 Stand-alone 방식의 모니터링이 아닌 웹 기반 모니터링 시스템으로 사용자가 기기를 떠나 어느 위치에 있더라도 원격으로 스마트폰이나 태블릿 PC를

사용하여 접근이 가능하다. 또한 웹 서버가 구축 되어 있으므로 원격에서 데이터의 이동이나 프로그램을 갱신할 수가 있다.

그림 6은 대표적인 사용자 인터페이스 화면의 예시를 나타낸다. 그림 6의 각각의 화면은 그림 4의 각 부분 회로를 설정하기 위한 유저 인터페이스이다.

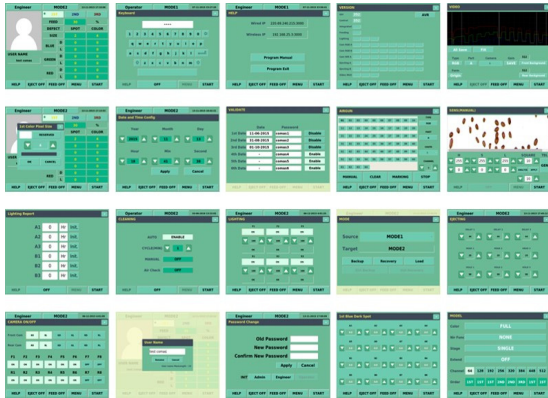


Fig. 6 A typical user interface screen

그림 7은 곡물의 무게 중심 검출 방법과 필터를 거친 $n \times n$ 이미지이다.

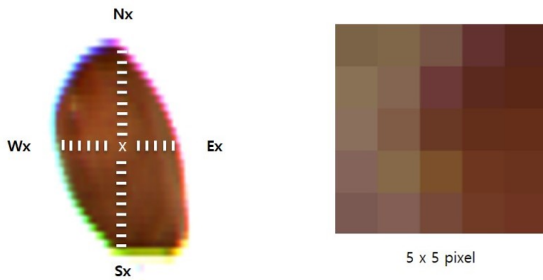


Fig. 7 The center of gravity and Image of $n \times n$ pixels

곡물의 무게 중심을 구하는 방법은 그림 7과 그림 8의 흐름도와 같다. 무게 중심은 고속으로 계산되어야 하기 때문에 외곽선 추출과 더불어 매 픽셀마다 중심점을 FPGA로 실시간으로 계산한다.

그림 8에서 N_x 와 S_x 그리고 W_x 와 E_x 가 같은 거리가 될 때 이를 무게 중심으로 판단하고 $n \times n$ 의 픽셀에 양품과 불량률의 개수가 일정 이상의 값이 될 때 이를 에어건 구동 회로로 보내지게 된다.

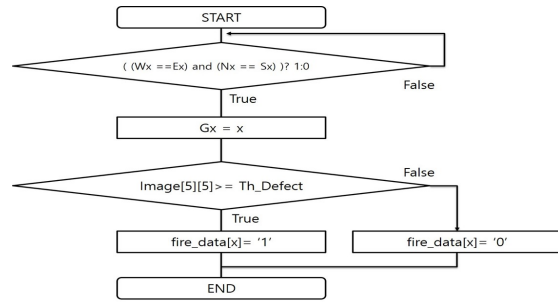


Fig. 8 Flowchart for grain sorting

IV. 실험 및 결과

그림 9는 웹에서 보는 Line CCD 카메라의 RGB Data 테스트 과정을 보여준다. 그림 9의 파형은 라인센서 R, G, B 값의 출력 파형을 각각 나타내고 있다[9].



Fig. 9 RGB Data of Line CCD Camera

그림 10은 곡물 선별을 위한 웹 제어 화면으로 HSL의 상한과 하한 값을 설정하며 선택 영역 크기를 조절하여 외곽선을 추출한다.

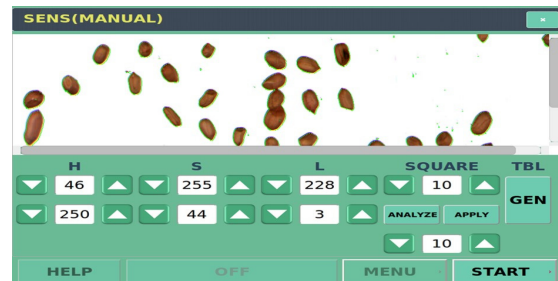


Fig. 10 Grain sorting web control screen

그림 11은 HSL의 히스토그램 분포를 나타내 본 것이며 mean 필터를 거친 화면이다.

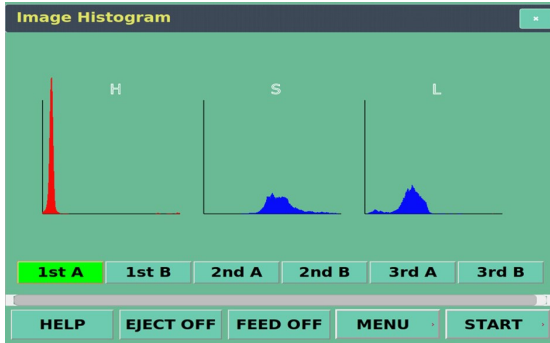


Fig. 11 Histogram distribution of HSL

기존의 곡물 선별 방법은 히스토그램에 의한 빈도 수 기반 처리 방법으로 수행되었다. 본 논문은 곡물의 무게 중심을 기점으로 HSL Decoding 방법을 사용하여 수행된 결과 표 1과 같은 결과가 도출되었다. 총 에러 개수 2728개에 대해서 정상 에러와 비정상 에러 개수를 기존의 방식(frequency-based processing method)과 본 논문에서 구현한 방식(HSL Decoding method)과 곡물의 무게 중심점 검출한 곡물 선별)과 비교 해 보았다.

그 결과 기존 방식에서의 비정상 에러는 763개인데 비해 제안된 방식의 경우 비정상 에러가 403개로 낮게 나타났다. 그리고 정상 에러에 대한 검출 능력을 비교 하면 기존의 방식은 1877개인데 제안된 방식은 2169개로 향상된 것을 알 수 있다.

Table. 1 Table comparing the frequency-based processing method and HSL Decoding method

The frequency-based processing method		
Error	Normal Error	Non-normal Error
2728	1877	763
HSL Decoding method		
Error	Normal Error	Non-normal Error
2728	2169	403

그림 12는 곡물 선별기를 제어하는 하드웨어 보드를 나타낸 것이다.



Fig. 12 Grain sorter hardware board

그림 13은 본 논문에서 제안된 방식을 적용한 64-Channel 곡물 선별기이다.



Fig. 13 64-Channel grain sorting machine

V. 결론

본 논문은 웹 기반의 모니터링 선별기를 개발하였다. 개발된 웹 기반의 모니터링 선별기는 영상 처리 성능 향상을 위해 기존의 RGB 선별 방식에서 벗어나 HSL Decoding 방식을 적용하여 개발하였다.

본 논문에서 제안한 웹 기반의 모니터링 시스템은 웹 서버구축을 위해 nodejs를 사용하였다. 이것은 기존의 선별 시스템에서 적용하지 않은 방식으로 선별기가 설치되어 있는 환경적 또는 위치적 제약에서 벗어나게 하는 장점이 있다. 또한 제안된 시스템은 임베디드 웹 서버에 의해 구동되는 고속 하드웨어 인터페이스를 구축

하고 기존의 RGB 빈도수 기반 처리 방식이 아닌 HSL Decoding과 곡물의 무게 중심점 검출을 사용하여 선별 하였다.

제안된 곡물 선별 시스템은 원격에서 제어가 되는 것은 물론 선별기 운용 상태를 다수의 사용자가 위치에 무관하게 선별 제어, 모니터링 그리고 시스템의 고장 및 보수를 판단 할 수 있는 장점을 지닌다.

본 논문에서 제안한 곡물 선별 시스템을 적용해본 결과 기존의 시스템보다 향상된 모니터링과 선별이 됨을 알 수 있었다.

ACKNOWLEDGMENTS

This work was supported by a 2015 Research Grant of Andong National University.

REFERENCES

[1] B. Lim, K. Jung, T. Khil and Y. Yoon, "Droplet size measurement using image processing method," *Journal of the Korean Society of Visualization*, vol. 2, no. 1, pp. 25-31, Aug. 2004.

[2] F. Takeda, H. Uchida, T. Tsuzuki, H. Kadota, and S. Shimanouchi, "A proposal of grading system for fallen rice

using neural network," in *Proceedings of the 2002 International Joint Conference on Neural Networks*, Honolulu:HI, pp. 709-714, May 2002.

[3] L. Zhang, A. Dehqhani, Z. Su, T. King, B. Greenwood and M. Levesley, "Development of a mechatronic sorting system for removing contaminants from wool," *IEEE/ASME Trans. Mechatronics*, vol. 10, no. 3, pp. 297-304, June 2005.

[4] S. H. Kim, K. H. Lee, H. B. Im, J. T. Chung, "Structural Modification for the Performance Improvement of a Grain Sorting Machine," *Transactions of the Korean Society for Noise and Vibration Engineering*, vol. 18, no. 2, pp. 208-214, Feb. 2008.

[5] K. S. Kim, H. M. Son, Y. M. Kim, C. G. Tak, S. S. Park, S. K. Lim, J. S. Ha, M. J. Jeong, Y. J. Lee, "Development of Embedded Controller Based Color-Sorter System," *IEMEK Journal of embedded systems and applications*, vol. 5, no. 2, pp. 84-92, June 2010.

[6] L. Yan, C. Y. Lee, G. Zhao, S. R. Lee, and K. M. Bae, "Image acquisition and processing for falling objects with line CCD sensor," in *Proceedings of the 6th IEEE International Conference on Industrial Informatics*, Daejeon: KOR, pp. 1257-1261, July 2008.

[7] H. Tayara, D. J. Lee, K. T. Chong, "Designing of FPGA-Based Real Time Spatial Filter of Image Processing," *Journal of Mechanical Science and Technology*, vol. 2014, no. 4, pp. 99-102, April 2014.

[8] L. Robert, *Linux System Programming*, 2nd ed, Sebastopol, CA:O'Reilly Media, 2013.

[9] S. Fulton, J. Fulton, *HTML5 Canvas*, 2nd ed, Sebastopol, CA:O'Reilly Media, 2013.



박세현(Se-Hyun Park)

한국정보통신학회논문지 제7권 제3호 참조
현재 안동대학교 전자공학과 교수
※관심분야 : 디지털시스템, 임베디드 리눅스



금영욱(Young-Wook Geum)

2002년 안동대학교 전기전자공학교육과 학사
2013년 안동대학교 전자공학과 석사수료
2010년 ~ 현재 ㈜코마스 연구원
※관심분야 : 위성통신, 마이크로파 회로해석 및 설계, 계측제어



김현재(Hyun-Jae Kim)

2014년 안동대학교 전자공학과 학사
2016년 안동대학교 전자공학과 석사수료
2014년 ~ 현재 (주)코마스 연구원
※관심분야 : IoT, 디지털시스템, 임베디드 리눅스