아두이노 보드에 대한 소프트웨어 테스트 방법 연구

경민기*, 민덕기*[1]
*건국대학교 컴퓨터공학과
e-mail : moonend@konkuk.ac.kr
dkmin@konkuk.ac.kr

Study of Embedded Software Test Method on Arduino Board

MinGi Kyung*, Dugki Min*
* School of Computer Engineering, KonKuk University

요 약

아두이노 보드는 Atmel 8 비트 RISC 마이크로프로세서를 이용해서 제작된 임베디드 보드이다. 마이크로프로세서 내부에 프로그램을 쓸 수 있다는 기능과 하드웨어 설계도가 오픈소스로 제작된다는 점, 일반 사용자들을 위해서 제공되는 쉬운 개발 언어 및 개발 환경을 제공한다. 본 논문에서는 아두이노 보드 위에서 동작하는 임베디드 소프트웨어에 대해 테스트하는 방법과 앞으로의 테스트 방법에 대한 개발방향에 대하여 논한다.

1. 서론

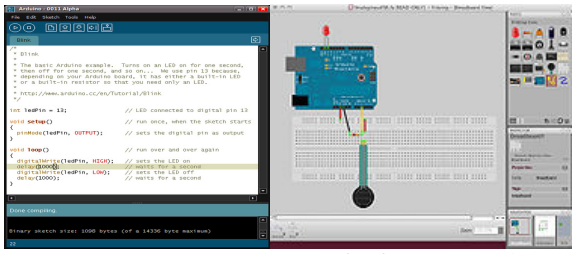
아두이노 보드(Aduino Board)는 Atmel 사에서 제작한 8 비트 RISC 마이크로프로세서인 AVR 마이크로 컨트롤러를 이용해서 제작된 임베디드 보드이다. 아두이노 보드는 내부에 플래시 메모리가 내장되어 있기 때문에 마이크로프로세서 내에 임베디드프로그램을 써 넣을 수 있다.[1, 2]

아두이노 보드는 하드웨어의 회로도가 공개되어 있으며, 오픈소스 소프트웨어를 통해서 단순화된 개발 언어, 간단하게 쓸 수 있는 라이브러리 등을 제공하고 있다. 하드웨어 설계도가 공개되어 있기 때문에 GPS, 무선 통신 모듈, Ethernet 모듈, 모터 모듈들을 사용자들이 직접 제작해서 추가할 수 있다. 이러한 편리성 때문에 미디어 아티스트들이 본 보드를 이용해서 많은 작품들을 만들어내고 있으며, 임베디드 디바이스 개발 교육에도 이용되고 있다. 또한 2009 년 7 월에 우주 공간에 있는 국제 우주 스테이션 ISS(International Space Station)에서 실시된 ANDE(Atmospheric Neutral Density Experiment) 실험 프로젝트[3]의 디바이스 구성에는 아두이노 기반의 측정 디바이스와 프로그램이 탑재되었다.

본 논문에서는 아두이노 보드의 임베디드 소프트웨어를 테스트하는 방법에 대해서, 그리고 앞으로의 아두이노 보드에 대한 테스트 방법의 개선 및 발전 방안에 대해서 소개하기로 한다.

2. 아두이노 보드의 제반 환경

아두이노 보드의 하드웨어 특징 및 설계도가 아두이노 홈페이지[4]를 통해서 공개되어 있다. 아두이노 보드에서 쓰이는 Atmega 모델의 마이크로 프로세서는 ISP(In-System Programming)이라는 방법을 이용해서 별도의 기록장비의 도움 없이 임베디드 프로그램을 기록하는 롬 라이팅(Rom Writing) 작업을 할 수 있다. 또한 센서 등 하드웨어에 센서를 추가하는 등의 회로 디자인 작업을 수행하는 오픈소스 프로그램 Fritzing [5]가 공개되어 있다.



(그림 1) Arduino IDE(좌)와 Fritzing(우)

아두이노 보드의 임베디드 소프트웨어 제작에는 Arduino IDE 도구에서 아두이노 프로그래밍 언어(Arduino Programming Language)가 쓰인다. 아두이노 프로그래밍 언어는 AVR-C 언어의 복잡한 라이브러리를 단순화시킨 Wiring 언어 및 라이브러리를 기반으로

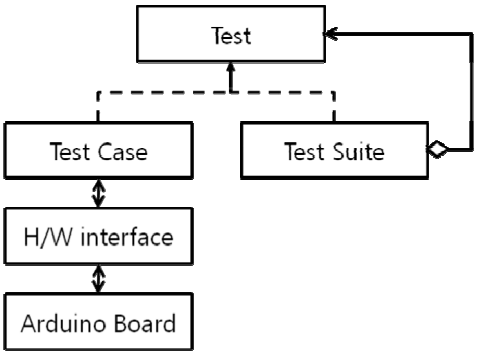
[1] 교신저자: 민덕기 (dkmin@konkuk.ac.kr)

* 본 연구는 지식경제부 및 정보통신산업진흥원의 대학 IT 연구센터 지원 사업의 연구결과로 수행되었음 (NIPA-2009-C1090-0902-0026)
* 본 연구는 지식경제부의 IT 성장동력기술개발사업의 일환으로 수행하였음.
[2008-S-007-01, 차량 전장용 통합제어 SW 플랫폼 개발]

제작되었으며, 처음으로 아두이노 보드를 접하는 사용자들이 임베디드 프로그램을 쉽게 제작할 수 있도록 명령어의 숫자와 용법이 Wiring 언어보다 더욱 단순화된 프로그래밍 언어이다.
또한 AVR 칩에 대한 시뮬레이션 환경은 Avrora 프로젝트[6] 등이 존재한다. 하지만 기존의 AVR 칩에 대한 시뮬레이션 프로젝트들은 AVR 칩에 대한 시뮬레이션 기능만을 지원하고 센서 및 디지털/아날로그 I/O 의 기능이 제대로 구현되어 있지 않기 때문에 주변 환경에 대한 시뮬레이션 작업 등을 수행할 수 없다. 또한 아두이노 보드에 대한 arduEmu 시뮬레이션 프로젝트는 아직 개발 단계에 있기 때문에 본 논문의 테스트 방법에서는 이용할 수 없었다.

3. 아두이노 보드 테스트 방법

아두이노 보드는 AVR 보드의 일종이지만, 하드웨어의 구조가 정해져 있기 때문에 주변장치에 대한 표준화된 인터페이스를 구성할 수 있고, AVR-C 언어가 아닌 좀 더 단순화된 아두이노 프로그래밍 언어를 사용하고 있기 때문에 테스트를 수행하기 위한 다른 방법이 필요하다.
본 논문에서는 Arduino Diecimila 보드와 Arduino IDE 를 이용해서 테스트하기 위한 환경을 구축하였다. 또한 아두이노 보드 기반의 임베디드 소프트웨어에 대한 테스트 작업을 수행하기 위해서 현재 소프트웨어 테스트에 이용되는 대표적인 유닛 테스트 도구인 JUnit 과 유사한 테스트 코드를 이용한 테스트 방법을 이용하였다.
테스트 케이스가 실제로 H/W 인터페이스 코드를 이용해서 아두이노 임베디드 보드에 입력 값을 전송해서 결과값을 받아오고, 그 값이 정확한지를 비교하는 역할을 수행한다. 그리고 테스트 케이스들을 묶어서 테스트 슈트를 생성하고 이용한다.



(그림 2) Arduino Testing Method

테스트 과정은 다음과 같은 순서로 이루어졌다. 테스트 케이스는 아두이노 보드에 테스트를 위한 입력 값을 전송하고, 아두이노 보드로부터 결과값을 받아, 제한 시간 안에 정확한 결과값을 보내주는지를 확인하였다. 테스트 코드는 자바 언어를 이용해서 제작되었고, USB 포트를 이용한 가상시리얼포트 제어를 위해서 RXTX 라이브러리[7]를 이용하였다.

4. 향후 개발 방안

하드웨어를 구하거나 만들기 쉽고, 임베디드 소프트웨어를 만들기 쉽다는 특징을 가지고 있기 때문에 다양한 임베디드를 필요로 하는 다양한 분야와의 융합 프로젝트에 이용되고 있다. 아두이노를 이용한 프로젝트는 센서 네트워크나 임베디드 컴퓨팅 분야를 넘어서서 아닌 예술 분야에서도 이루어지고 있으며, 특히 IR multitouch 의 구현, Wearable Computer 및 새로운 센서들과 그 변형에 대한 구현 등과 같은 새로운 개념의 지원 및 구현에 많은 도움을 주고 있다. [8, 9, 10, 11, 12, 13]
본 논문에서는 아두이노 보드에 대한 테스트 방법을 이용해서 아두이노 보드에 올려진 임베디드 소프트웨어의 테스트를 자동화하는 것이 가능하다. 또한 테스트 방법을 이용해서 임베디드 소프트웨어의 에러를 효과적으로 측정할 수 있으며, 에러나 기능에 대한 정량화 작업을 수행할 수 있다.
앞으로 아두이노 보드에 대한 테스트 방법에서 추가적으로 필요한 테스트 기능은 실제로 센서 등을 시뮬레이션 모듈을 제작, 아두이노 보드를 테스트하는 방법에 대한 기능과 하드웨어 보드 설계, 임베디드 소프트웨어 제작 및 제작한 임베디드 소프트웨어에 대한 테스트 케이스들을 더욱 유기적으로 실행, 관리하기 위한 프레임워크의 구축이 필요하다.

참고문헌

[1] 윤덕용, AVR 마이크로콘트롤러의 특성 및 응용기술, 전력전자공학회지, 2004
[2] ARDUINO 보드를 소개하며, http://imaso.co.kr/?doc=bbs/gnuboard.php&bo_table=article&wr_id=32219
[3] Atmospheric Neutral Density Experiment, http://eludium.stensat.org.nyud.net/ANDE/Main.html
[4] Arduino Homepage, http://arduino.cc/
[5] Fritzing http://fritzing.org/
[6] Avrora – The AVR Simulation and Analysis Framework http://compilers.cs.ucla.edu/avrora/
[7] RXTX: serial and parallel I/O libraries http://rxtx.org
[8] Santiago J. Barro Torres, Carlos J. Escudero Cascón, Embedded Sensor System for Early Pathology Detection in Building Construction, IJCSI International Journal of Computer Science Issues, Vol. 1, 2009
[9] Martin Dybdal , Troels Henriksen and Jesper Reenberg , Fladuino: Controlling embedded devices with functional reactive programming, 2009
[10] Andrew Schmeder, Adrian Freed, A Low-level Embedded Service Architecture for Rapid DIY Design of Real-time Musical Instruments, New Interfaces for Musical Expression, 2009
[11] IR Multitouch, http://arduino.cc/blog/?p=317
[12] Leah Buechley, Mike Eisenberg, Jaime Catchen, Ali Crockett, The LilyPad Arduino: using computational textiles to investigate engagement, aesthetics, and diversity in computer science education, Conference on Human Factors in Computing Systems, 2008