

아두이노

김대희

2020년 6월

Contents

1. 아두이노	5
---------	---

1 아두이노	7
1. 아두이노 개요	9
1.1. 상표권 분쟁	9
2. 하드웨어	11
2.1. 호환, 파생 제품들	11
3. 소프트웨어 개발 환경	12
4. 버전	13
5. 구현 사례	16
5.1. 3D 프린터	16
5.2. 드론	16
5.3. 전기 자전거	16
5.4. 사물인터넷(스마트홈)	16
5.5. 스마트 안경(HMD)	16
5.6. 로봇 팔	16
5.7. UAV(프로펠러형)	16
5.8. 스마트 장갑	16
5.9. 인버터	16
5.10. 게임기	16
6. 주의사항	17
7. 한국 내 동향	19
8. 참고문헌	20
9. 아두이노 책	21
9.1. 두근두근 아두이노 공작소	21
9.2. 아두이노 for 인터랙티브 뮤직	22
10. 외부링크	24

List of Tables

Part 1.

아두이노

Table of Contents

1	아두이노	7
---	------	---

CHAPTER 1

아두이노

Contents

1. 아두이노 개요	9
2. 하드웨어	11
3. 소프트웨어 개발 환경	12
4. 버전	13
5. 구현 사례	16
6. 주의사항	17
7. 한국 내 동향	19
8. 참고문헌	20
9. 아두이노 책	21
10. 외부링크	24

1. 아두이노 개요

VR 기반의 마이크로컨트롤러 개발환경인 Wiring에서 파생한 프로젝트다.

영어로 '아두이노', 이탈리아어로 '아르두이노'라고 읽는다. 영어권의 영향이 강한 국내에서 많이 사용되는 명칭은 아두이노. 이탈리아어로 '강력한 친구'라는 뜻이라는 듯. 2005년 이탈리아의 Massimo Banzi와 David Cuartielles가 처음 개발하였다. 개발자 Massimo Banzi가 직접 저술한 <Getting Started with Arduino>(번역명 <손에 잡히는 아두이노>)를 필두로 많은 입문서들이 나와 있다.

비관련자를 위해 쉽게 설명하자면, 간단한 초소형 컴퓨터 기판에 이런저런 기능을 할 수 있도록 프로그래밍을 하여 다양한 기계나 작업, 작품에 써먹는 경우가 많은데, 그것들 중에서도 교육에 특화되어 특히 더 쉬운 사용법을 자랑하는 시스템이라 보면 간단하다. 아두이노와 관련된 작품들을 보고싶다면 이곳을 추천한다. 많은양의 아두이노 작품들을 볼 수 있다.

1.1. 상표권 분쟁

한때 Arduino 브랜드는 미국 안에서만 쓰이고 미국 밖(유럽 등)에서는 옆에 있는 Genuino 브랜드가 쓰인 적이 있는데, 이는 상표권 분쟁에 따른 결과였다.

2015년 7월 경, 해외에서 판매되는 아두이노 제품을 위한 브랜드인 Genuino가 발표되고, Arduino Leonardo 등의 몇몇 제품이 단종되었다. 당시에는 돈을 더 벌기 위한 것이 아니냐는 논란이 있었지만, 사실은 복잡한 사정이 있었다.

아두이노는 2008년 설립한 Arduino LLC가 상표권을 가지고, 실제 생산은 타 업체에서 진행하는 방식으로 제품 판매를 이어왔다. 그런데 2008년 말 창업자 중 한명인 Gianluca Martino의 회사인 Smart Projects에서 아두이노의 상표권을 몰래 이탈리아에서 등록하며 문제가 되기 시작한다. 당시에는 아무도 이 사실을 몰랐으나, 이 사실은 Arduino LLC에서 미국 이외 지역에 상표권 등록을 시도할때 이탈리아에서 이미 상표권이 등록되었다는 것이 밝혀지며 드러난다. Arduino LLC는 즉시

상표권 협상을 진행했으나 결렬되었다. 이후 Gianluca Martino는 회사를 Federico Musto에게 매각했는데 매각 이후 Smart Projects는 로열티를 내는 것을 거부하고 사명도 Arduino SRL로 바꾸기에 이르른다. 결국 Arduino LLC는 Arduino SRL을 고소하나 큰 진전은 없었고, 결국 2015년 7월 Arduino LLC는 미국외 지역 판매를 위해 Genuino라는 브랜드를 등록하게 된다.

이렇게 끝나지 않을 것 같던 분쟁은 Arduino LLC와 Arduino SRL간 합의로 종결된다. 2016년 10월 World Maker Faire에서 양측 회사 대표가 나와 Arduino LLC와 Arduino SRL이 앞으로는 "아두이노 홀딩(Arduino Holding)"이라는 이름으로 합병될 것이며, 소프트웨어 등 지원은 "아두이노 재단(Arduino Foundation)"에서 진행할 예정이라고 발표했다.

이후 아두이노 창업자 4명의 회사인 BMCI가 지분의 49%를, Arduino SRL의 대표인 Musto가 50%를 가진 회사 Arduino AG를 설립하며 아두이노 브랜드에 대한 모든 권리를 가지게 되며, 그와 별도로 아두이노 재단이란 비영리 단체을 설립해 아두이노 IDE의 개발을 맡게 된다.

2017년에는 BMCI가 Arduino AG의 나머지 51%의 지분까지 모두 인수하며 완전히 자회사로 편입시킨다. 그와 동시에 홈페이지 또한 arduino.cc로 통합되었다. arduino.cc측 사이트로 가 보아도 arduino.org쪽 보드들을 판매하고 있다. 이제는 arduino.org에 접속시 영문 리다이렉트 안내 페이지와 함께 몇초 뒤 arduino.cc로 자동으로 리다이렉트하게 된다.

2. 하드웨어

위 사진의 보드는 우노 R3 버전으로 2013년 기준 레퍼런스 보드이자 가장 보편적으로 사용되는 보드다. 사용하는 마이크로컨트롤러는 ATmega328P로 16MHz로 동작하고 프로그램 저장용 플래시 32kb를 내장한 프로세서이다.

주로 Atmel사의 AVR이 사용되는데, 아래 보드들 가운데 사용된 칩 이름 앞에 AT가 들어간 것들이다. 임베디드 개발 경험이 전혀 없는 사람을 위해 개발된 교육용 플랫폼이기 때문에 프로그램을 작성하고 보드에 프로그램을 올리는 과정을 단순화하여 다루기 쉽게 되어 있다. 버전에 따라 조금씩 다르지만 아두이노 보드와 개발 환경은 대개 다음과 같은 모습이다.

자세한 내용은 Arduino/하드웨어 문서 참조.

2.1. 호환, 파생 제품들

아두이노의 설계는 CC BY-SA 2.5에 따라 공개되어 있으므로 누구나 해당 설계도에 따라 제품을 만들 수 있다. 따라서 아두이노 LLC에서 직접 판매하는 정품 [5] 이외에도 중국 등지에서 만든 호환 보드들이 판매되고 있다. 다만 아두이노라는 명칭이나 로고 자체는 함부로 사용할수 없으므로, 이러한 호환 보드에는 주로 로고나 이름 등이 제거된 채 출시된다.

이러한 호환 보드들은 주로 eBay나 Aliexpress 등에서 판매되나 일부 아두이노 호환품은 국내에서도 판매되고 있다. 간단한 선에서는 그냥 아두이노와 100% 동일하나 가격만 저렴하게 만든 중국제들 부터 시작해서 보드에 블루투스, 와이파이 등의 통신모듈을 내장시킨 것들은 흔하며 초소형화된 것들도 볼 수 있다. 대표적으로 라떼판다가 있다.

3. 소프트웨어 개발 환경

4. 버전

■ 2016년 2월 현재 기본형이라고 할 수 있는 Arduino Uno를 비롯한 다양한 변종이 있다. 하드웨어의 회로도까지 오픈 소스라 아두이노와 호환되는 보드를 정품보다 훨씬 저렴한 값에도 구할 수 있다. 모든 소스가 공개되어 능력이 되면 자작도 가능하다.

■ 긴 알파 테스트 기간을 거쳐서 2011년 11월 30일, 1.0 버전이 릴리즈되었다. 2012년 5월 21일 공개된 1.0.1버전부터는 UI의 언어가 다국어지원이 되는데 이 중에는 한글도 있어 초심자들의 접근이 더욱 쉽게 되었다. 2017년 현재 최신버전은 1.8.3버전이다.

■ 2012년 7월 즈음 아두이노 신제품인 아두이노 Leonardo(레오나르도)가 출시되었다. 특징으로 온칩 USB 컨트롤러가 내장된 Atmega32U4를 메인으로 채택해 단가를 줄이고 마우스, 키보드로 인식시킬 수 있어 다양한 활용이 가능하다. 참고로 아두이노 우노를 dfu-program을 이용해서 펌웨어를 업데이트하면 레오나르도처럼 USB입력장치로 사용이 가능하지만 이 경우에는 레오나르도처럼 바로 업로딩 후에 인식이 가능하지 않고, dfu모드 진입과 해제 과정을 거친 후에 HID 장치로 인식이 가능하다. 그래서 한번 펌웨어를 업데이트시킨 상태에서 수정하려면 좀 번거롭다. 그러니 아두이노를 입력장치로 써먹으려면 그냥 레오나르도를 사용하는편이 좋다. 그리고 dfu-program으로 펌웨어를 올리는 과정에서 잘못하다가 아두이노의 UART 컨버터가 벽돌이 될 수 있으니 조심하도록 하자.

■ 2013년 초 ARM Cortex-M3 SAM3E8X(512KB 플래시 메모리, 96KB SRAM, 클럭 84Mhz)를 채택한 Arduino Due가 출시되었다. 기존의 Arduino Mega 2560[13]의 후속모델에 가깝다. ARM 기반이니만큼 프로세서 성능은 훨씬 고성능이다. 다

만 아날로그 입력/PWM 출력이 12핀으로 MEGA보다 약간 적다. DAC 두 개와 CAN 핀 때문에 그렇다. 총 4개가 줄어든 셈. 동작 전압이 다른 보드와 다른 3.3V 이므로 I/O 핀에 5V를 인가하면 핀이 나가는 수가 있으니 주의. 일반적인 아두이노 보드의 동작전압은 5V이다. 3.3V는 FTDI에 내장된 레귤레이터나, UNO 같이 FTDI가 없으면 온보드 레귤레이터로 출력한다.

■ 2013년 10월에는 인텔에서 갈릴레오라는 이름으로 자사 펜티엄 기반의 32nm공정을 사용한 새로운 인텔 퀵크[14] SoC를 탑재한 호환 보드를 발표하였다. 며칠 뒤에는 TI에서 Cortex-A8 기반의 자사 1GHz Sitara SoC와 ATmega32U4를 동시에 내장한 아두이노 Tre라는 보드를 발표하였다. Tre는 Uno와는 100배 이상의 성능 차이가 있다고 하는데 레오나르도에 탑재된 ATmega32U4까지 동시에 내장해서 호환성 문제는 없다. 인텔의 갈릴레오는 2014년 4월경 단종 수순 (핀 호환성이 영 안 좋다는 이야기가 있다. 인텔 퀵크 항목을 참조)이고 대신 갈릴레오2가 출시되었다. 갈릴레오 2 이후 사물인터넷을 위한 Intel Edison 보드가 출시되었다. 아두이노와 꽤 협력적인 관계를 구축한듯. 실제로 아두이노 공식적으로 인증된 타회사제 아두이노 기판은 죄다 인텔제였으나 삼성에서도 나오면서 옛말. Tre는 2019년 현재도 감감무소식이다.

■ 2014년 5월 15일 아두이노 Uno의 후속으로 아두이노 Zero가 발표되었다. 가장 큰 변경점은 MCU가 기존의 ATmega 계열에서 ARM Cortex-M0+ 계열로 변경되면서 연산속도와 메모리공간 등이 늘어난 것.

■ 이외에 아두이노 Yun이라는 리눅스 OS로 구동되는 보드나 브레드보드 같은 곳에 꼽아 사용 가능한 아두이노 Micro, Nano, (Pro)Mini, Yun Mini 같은 물건들도 발매되었다. 리눅스 OS 기반은 갈릴레오같이 에뮬레이션 방식은 아니고 32U4와 리눅스 머신이 같이 내장된다.

■ 현재 최신 보드는 Arduino 101로 인텔과 협작으로 나온 보드다. 특징은 인텔 Curie 칩을 사용하고 BLE와 가속도계, 자이로스코프가 보드에 기본 내장된다. 또 작동 전압이 3.3V이지만 5V를 인가해도 핀이 손상되지 않는다. 가격도 국내에서 4만원 후반 5만원 초중반대의 가격. 라즈베리 파이와 비슷하다. 101은 구조상 프로세서가 2개 붙어있어서 한쪽은 RTOS 서비스를 돌리고 다른쪽 코어는 유저의 코드와 라이브러리를 돌린다. 여기에 둘을 연결하는 메일박스를 두고 양측이 필요한 작업요청을 주고받는 메시지패싱 구조를 가지고 있다. 이때 유저는 OS와 메모리를 나눠 쓰기 때문에 느낌상 메모리가 적어보이게 된다. 그러나 OS가 제공하는 편의성이 괜찮아서 큰 손해는 아니다. 다만 이 OS와 라이브러리는 아직 문서와 예제가 부실해서 문서가 없으면 소스코드를 보면 된다는 정도의 기본기는 있어야 본격적으로 응용을 해볼 수 있는 상태라는게 문제. 이렇게보면 이게 무슨 입문용이나 싶겠지만 어차피 기본문법의 스케치라면 똑같이 돌아가는 것이고, 5V 입력 핀이 달려있어서 태워먹을 위험없이 UNO용 회로들을 시도해볼 수 있으며, 32비트 보드 중에는 가격도 적절해서 입문용이라고 해도 별 문제는 없다[18]. 한편 2017년 들어 인텔의 저전력 하드웨어가 지속적으로 단종되는 가운데 101도 단종 절차에 들어갔다.

5. 구현 사례

5.1. 3D 프린터

5.2. 드론

5.3. 전기 자전거

5.4. 사물인터넷(스마트홈)

5.5. 스마트 안경(HMD)

5.6. 로봇 팔

5.7. UAV(프로펠러형)

5.8. 스마트 장갑

5.9. 인버터

5.10. 게임기

6. 주의사항

■ 8비트 임베디드 시스템 프로그래밍에 있어서는 프로그램 최적화에 매우 신경써야 한다. 마이크로 컨트롤러같이 메모리 리소스가 크게 제한되는 장치에서 프로그래밍을 할 경우 항상 주의해야 하는 부분이다. 아두이노는 성능이 매우 떨어지기 때문에 다른 기계에서는 문제없을만한 부분이 유독 문제를 일으키기도 한다. 컴파일은 잘 해놨는데 정작 보드에서는 장비가 멈춰버린다든지 한다. 개발 과정에서 문제가 생겼다면 마지막으로 추가한 내용들을 살펴보며 코드의 내용을 최대한 다이어트 시켜보자. C에서 함수가 중복 호출될 경우 스택 오버플로를 일으키면서 장비가 정지될 수 있다. 예를 들어, 장비가 정지했는데 개발 도중에 동작 테스트를 위해 추가한 `Serial.print();`문 몇 줄을 지워보니 추가한 코드들이 갑자기 정상적으로 돌아가는 경우도 있다. 이 때문에 deploy판에 디버그용 `print`를 지우고 보내는 것은 기본이다. 하드웨어 시리얼로 블루투스 통신을 한다던지 등 예외적인 경우에만 놔두는 것이 좋다.

■ 아두이노 자체의 성능 제약을 피하려면, 라즈베리 파이 같은 다른 장치를 연결해 연산은 다른 데에서 처리하고, 아두이노는 센서나 액추에이터 등을 관리하는 기계로만 사용하는 게 나을 수도 있다. 아두이노에 Firmata라는 펌웨어를 올리면[19] Firmata 프로토콜을 통해서 외부에서 아두이노를 컨트롤할 수 있게 된다. Processing을 비롯해 꽤 많은 언어를 이용하여 Firmata로 아두이노를 컨트롤할 수 있으니 이쪽을 알아보는 것도 괜찮다. 이걸 잘 활용하면 아두이노에는 최소한의 코드만 올리고 연산 부하가 큰 나머지 부분은 PC나 라즈베리 파이 같은 별도의 장치의 자원을 사용하여 돌리는 식으로 동작시키는 게 가능하다. 아두이노 자체로는 센서값 읽어서 판단하고 트윗올리는 것만 짜넣어도 뽀뽀한 경우가 있으니 판을 크게 벌일 거라면 다른 장치를 사용해 통제하는 걸 고려해보자. 실제로 많은 작업들이 이렇게 제작된다. 판이 크지 않다고 하더라도 외부 컴퓨터-아두이

노 사이의 통신이 필요한 프로젝트인 경우, 시리얼 통신으로 메시지를 주고받는 것보다 자연스러운 형태로 코딩이 가능하므로 Firmata를 적극 사용해보는 것도 괜찮다. 이런 과정이 번거롭다면 내부 플래시와 램 용량이 크게 늘어난 아두이노 두에 혹은 2016년 기준 최신형인 제로[20]나 전술한 101 또는 Primo, Star를 쓰는 것도 좋은 선택.

■ 그리고 또 주의할 점이 있는데, 아두이노 IDE에 내장된 ArduinoISP 예제 [21]를 이용하여 다른 아두이노에 업로드를 할 경우, ArduinoISP로 업로드받은 아두이노는 부트로더가 지워진다! 그렇기 때문에 웬만하면 USB TO UART 컨버터를 이용해 업로드하자.

■ 아두이노 컴파일러에서 소스 코드를 보드에 업로드 할 때, 우분투같은 리눅스로 하는 것이 윈도우로 하는 것 보다 훨씬 빠르다.

7. 한국 내 동향

■ 국내에서도 미디어아트나 취미로 임베디드 프로그래밍을 하는 계층에게 인기를 끌고 있으며 일부 대학에서도 커리큘럼을 개설하고 교육하는 곳이 있다. Processing과 마찬가지로 공학적 지식이 전무한 디자인/예술 전공 학생들에게 충격과 공포를 안겨주는 존재. 다만 경험적으로 보았을 때 디자인/예술 계통 학생들은 썩썩 프로그래밍 학습을 하게 되는 Processing보다는 뭔가 물리적인 리액션을 경험할 수 있는 아두이노 쪽을 재미있어하는 경향이 있다. 하지만 깊게 들어가면 이쪽도 지옥문이 열리기는 마찬가지로 전자공학의 세계에 온 것을 환영하오 나쁜이여 하지만 MCU를 깊숙하게 파고 들어가는 것에 비하면 침대에 누워서 잠자는 수준으로 쉽다. 애초에 디자인/예술 전공 학생들은 자기 전공 하기에 바쁜데 전혀 다른 분야를 알게라도 접하니 어려워 하는 것이 당연하다. 반대로 전자공학 전공자들에게 디자인/예술 개론실습과목 들으라고 해도 카오스가 벌어지듯이...

■ 공대 졸업작품으로도 유용하다.

■ 여기서 더 나아가고 싶다면 AVR MCU 배우기를 추천한다. MCU 프로그래밍은 아두이노에 비해 심각하게 어려운 편은 아니다. 대부분의 전자공학 관련과에 MCU 관련된 강의가 못해도 1개 이상은 있을 거니까 써먹기도 좋다. 하지만 이거도 못해서 교수님들한테 프로그래밍 부탁하는 사람이 꼭 있다. 단, 16진수하고 씨름하고 싶지 않다면 아두이노가 낫다.

8. 참고문헌

9. 아두이노 책

9.1. 두근두근 아두이노 공작소

■ 개요

- 제목 : 두근두근 아두이노 공작소 : 25가지 기발한 실습 예제로 배우는 아두이노 입문
- 지은이 : 마크 게디스
- 옮김 : 이하영
- 출판사 : 인사이트
- 출판일 : 2017년 8월 18일
- 도서관 : 2020.06.19 수정 분관 569 17

■ 목차

- 옮긴이의 글
- 감사의글
- 머리말
- 1. LED
- 2. 소리
- 3. 서브모터
- 4. LCD
- 5. 계수기
- 6. 보안
- 7. 고급 예제

■ 마크 게디스

9.2. 아두이노 for 인터랙티브 뮤직

■ 개요

- 제목 : 아두이노 for 인터랙티브 뮤직
- 지은이 : 채진욱
- 출판사 : 인사이트
- 출판일 : 초판 2011년 3월 10일 2쇄 2012년 6월 20일
- 도서관 : 2020.06.19 수정 분관 569 15

■ 목차

- 추천의 글
- 지은이의 글
- 책에 대해
- 0장 인터랙티브 뮤직의 개요
- 1장 아두이노와 만나다
- 2장 아두이노와 친해지기
- 3장 인터랙티브 뮤직을 위한 준비
- 4장 시리얼 송신 연습
- 5장 MIDI 출력
- 6장 시리얼 수신 연습
- 7장 MIDI 입력
- 8장 아두이노 온 바흐
- 부록 못다한 이야기들
- 맺는말
- 찾아보기

■ 채진욱

10. 외부링크

9 외부 링크