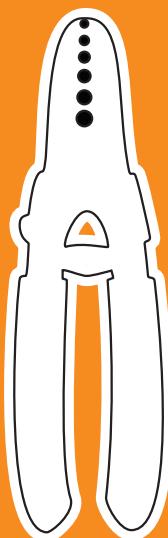
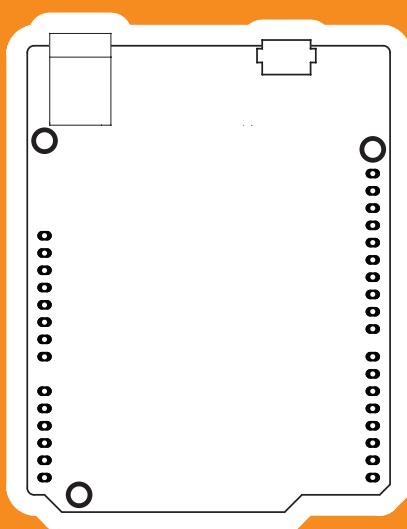


1

무엇이 필요할까

Required materials



2 말하고 듣고 대화하는 Physical Computing 아두이노

자료 1

아두이노, 브레드보드, 컴퓨터와 소프트웨어

이 책의 예제들을 작업하기 위해서는 아두이노 마이크로프로세서, 브레드보드, 컴퓨터, 소프트웨어 등이 필요하다.

아두이노 Arduino

아두이노는 마이크로프로세서의 한 종류로, 초보자도 비교적 쉽게 이용할 수 있다. 아두이노에는 여러 종류가 있다. 이 책의 모든 예제에서는 아두이노 레오나르도 [Arduino Leonardo](#)를 사용할 것이다. 아두이노 레오나르도는 프로그래밍과 코드 실행에 필요한 회로 두 가지 모두를 갖추고 있다. 이 분야를 전공하고 있는 학생들이라면 마이크로프로세서로 회로를 직접 설계해보는 작업을 한 적이 있을 것이다. 이러한 작업들은 그다지 어렵지 않다. 앞으로 이 책에서 여러분만의 프로젝트를 만들기 위해 알아야 하는 기초 회로 설계와 코드 작성법에 대해 하나씩 배울 것이다. 자신만의 프로젝트를 만들기에 앞서 무엇보다 중요한 것은 기초를 제대로 다지는 일이라는 것을 잊지 않기를 바란다

아두이노 레오나르도의 기본 구조는 그림 1.1과 같다.

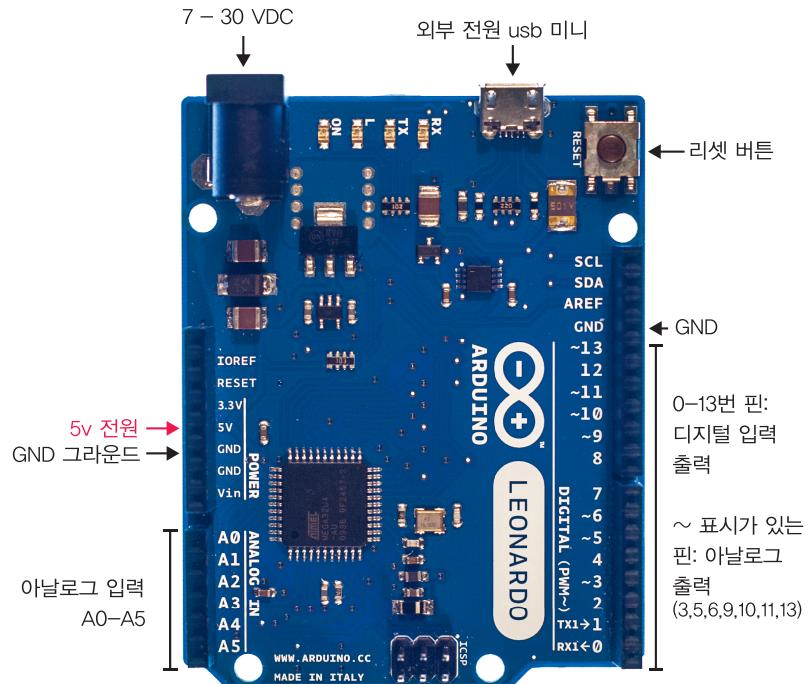


그림 1.1

브레드보드 Breadboard

브레드보드는 프로토타이핑 보드 prototyping board라고도 부르며 회로를 만드는 장소다. 다음에 나오는 그림 1.2는 브레드보드의 내부 구조를 보여주고 있다. 브레드보드의 뒷면을 열어 안을 들여다보면 어떤 구멍끼리는 서로 연결되어 있는 반면 어떤 구멍끼리는 연결되어 있지 않은 것을 볼 수 있다. 이 구조를 파악하는 것은 매우 중요한데, 그렇지 않으면 실수를 하기 십상이기 때문이다. 그런 일을 피하기 위해서 다음 그림 1.2의 브레드보드 설명을 보며 천천히 배워보도록 하자.

4 말하고 듣고 대화하는 Physical Computing 아두이노

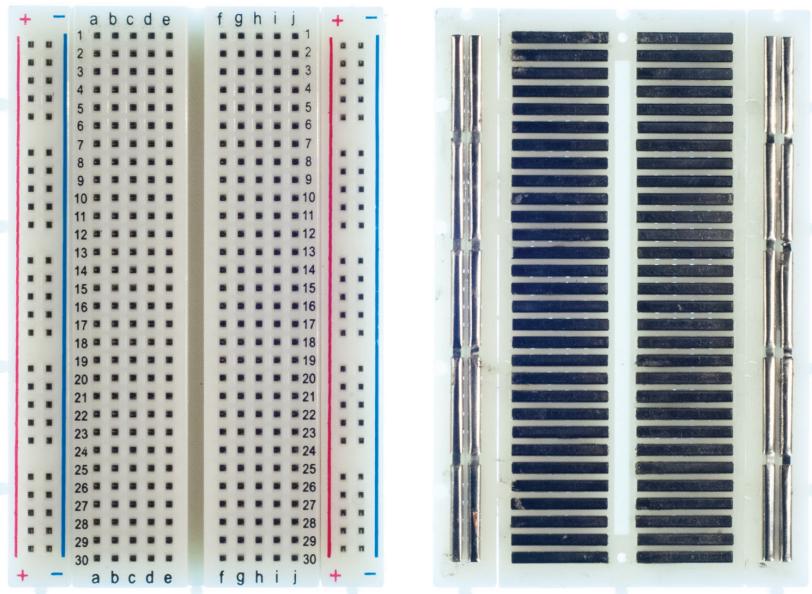


그림 1.2

브레드보드는 가로줄과 세로줄로 이루어져 있다. 브레드보드의 왼쪽, 오른쪽 양 끝에는 각각 두 개씩 총 네 개의 세로줄이 있다. 이 네 개의 세로줄은 서로 연결되어 있지 않으며, 하나의 세로줄 안에 있는 구멍들은 그림 1.3에서 표시한 노란줄끼리 연결되어 있다. 빨간색 줄 옆에 있는 세로줄을 우리는 파워 버스 power bus라 부른다. 파란줄 옆에 있는 줄은 그라운드 버스 ground bus라 부른다. 브레드보드의 중앙에는 가로 한 줄에 다섯 개씩 구멍이 있다. 이 가로 방향의 다섯 개의 구멍은 서로 연결되어 있지만, 윗줄과 아랫줄, 즉 세로방향으로는 서로 연결되어 있지 않다. 브레드보드는 정중앙을 중심으로 크게 왼쪽과 오른쪽으로 구분된다. 중앙을 중심으로 왼쪽 보드와 오른쪽 보드는 서로 전혀 다른 구역이다. 왼쪽편에 있는 구멍들은 오른쪽편에 있는 구멍들과 연결되어 있지 않다.

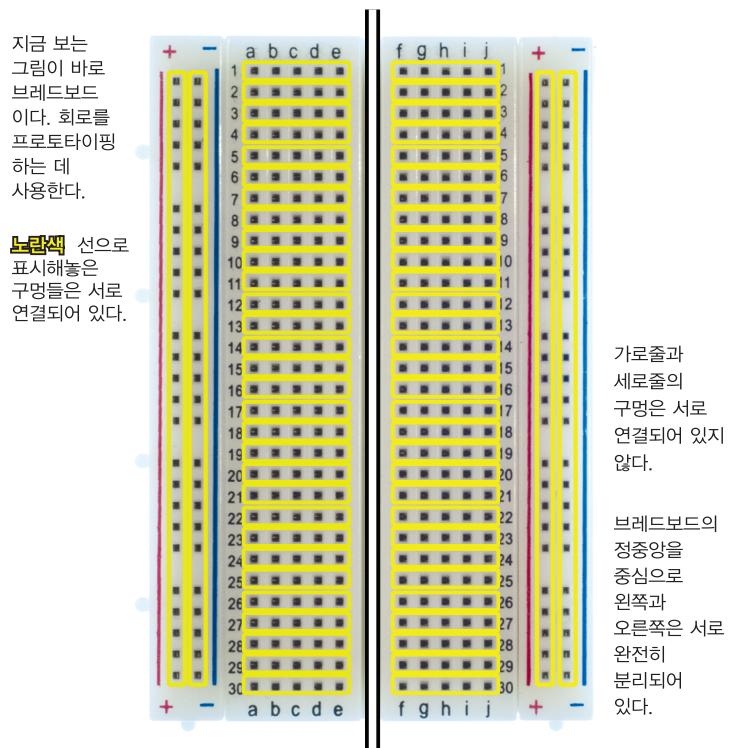


그림 1.3

브레드보드의 어떤 구멍들이 서로 연결되어 있고, 어떤 구멍들이 연결되어 있지 않은지 그림 1.3을 통해 다시 한 번 확인해라. 이렇게 이 부분에 시간을 많이 들이는 이유는 전기합선을 방지하기 위해서이다. 연결된 구멍과 단절된 구멍을 무시한 채 부품을 꽂을 경우 합선으로 인한 화재의 위험이 있으므로 주의해야 한다.

6 말하고 듣고 대화하는 Physical Computing 아두이노

컴퓨터와 소프트웨어

아두이노

아두이노와 아두이노 소프트웨어는 윈도우, OS X, 리눅스 등, 어느 운영체제에서나 작동 가능하다. 인터넷에서 아래 주소를 입력해라.

<http://arduino.cc/kr/Main/Software>

각자의 운영체제에 맞는 소프트웨어 패키지를 클릭해서 다운받는다. 다운로드가 완료되면 자동으로 설치창이 나타난다. 만약 설치법을 잘 모르겠다면, 아래 주소로 들어가서 설치하는 방법을 참조하길 바란다.

<http://arduino.cc/kr/Guide/HomePage>

아두이노의 모델과 여러분이 사용하는 운영체제에 따라 설치 방법이 조금씩 다르므로 주의하길 바란다.

그럼 1.4가 바로 아두이노 통합개발환경이다. 통합개발환경을 약자로 IDE라고도 한다.

아두이노의 IDE는 다음과 같이 구성되어 있다.

- Verify button 확인 버튼 – 코드에 오류가 있는지 검사해주는 기능
- Upload button 업로드 버튼 – 코드를 아두이노 보드에 업로드할 때 사용
- Open, New, and Save buttons 열기, 새로 만들기, 저장 버튼
- Serial Monitor button 시리얼 모니터 버튼 – 시리얼 모니터 창을 띄울 때 사용
- Option button 옵션 버튼 – 여러 개의 탭을 띄워놓고 작업할 때 사용
- Code window 코드 윈도우 – 코드를 작성하는 창
- Message window 메시지 윈도우 – 아두이노 프로그램의 메시지를 표시하는 창

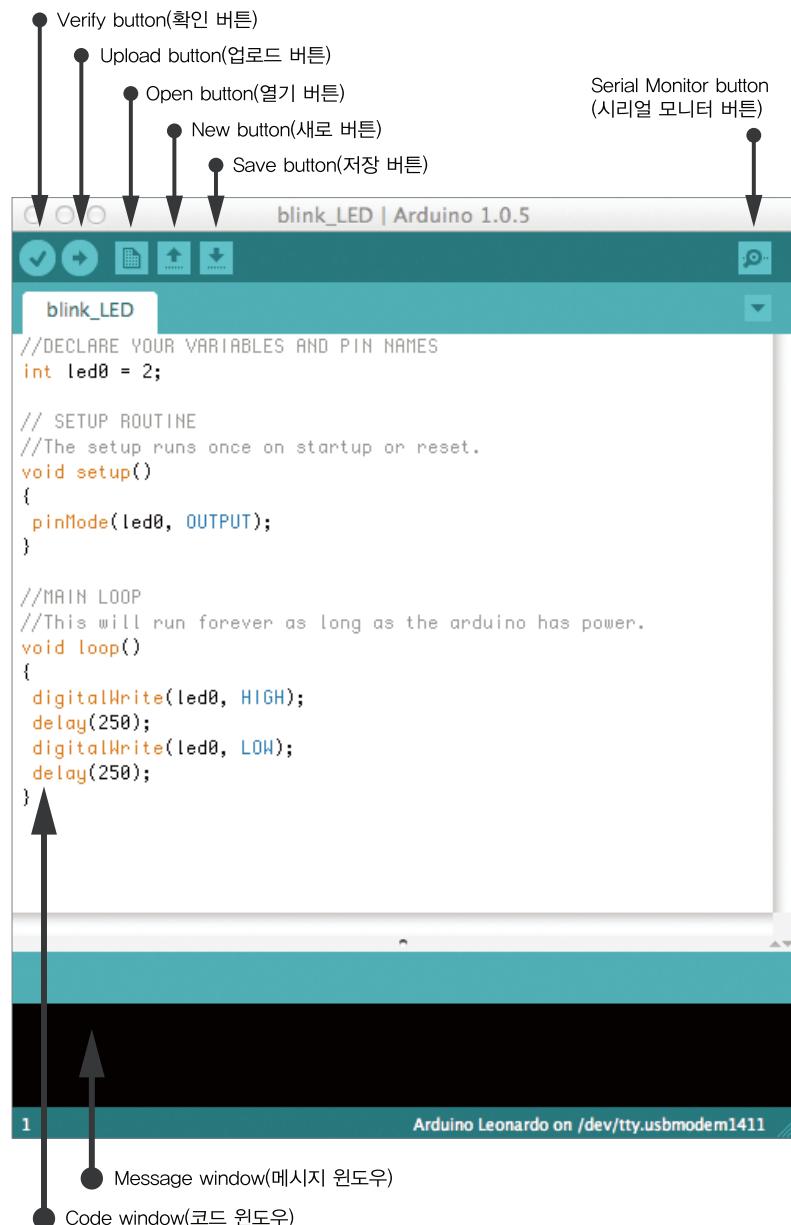


그림 1.4

8 말하고 듣고 대화하는 Physical Computing 아두이노

프로세싱 Processing

우리가 사용할 또 다른 응용프로그램은 바로 프로세싱이다. 프로세싱은 스크린을 기반으로 한 코딩 프로그램이다. 시리얼 포트를 이용하여 아두이노와 프로세싱을 연동할 수 있으며, 이 방법을 통해 더 많은 인터랙티브 작업이 가능하다. 프로세싱을 다운받으려면 아래 주소로 접속해라.

<http://processing.org/download/>

프로세싱과 프로세싱 사용 환경에 대해 더 자세히 배우고 싶다면 아래의 사이트를 참조하기 바란다.

<http://processing.org/learning/gettingstarted/>

그럼 1.5가 바로 프로세싱 통합개발환경이다. 역시 줄여서 프로세싱 IDE라고도 한다.

프로세싱 IDE의 구성은 다음과 같다.

- Run button 런 버튼 – 버튼을 누르면 현재 스케치를 재생
- Stop button 스탑 버튼 – 스케치 재생을 멈춤
- Open, New and Save buttons 열기, 새로 만들기, 저장 버튼
- Option button 옵션 버튼 – 여러 개의 탭으로 작업할 때 사용
- Code window 코드 윈도우 – 코드를 작성하는 창
- Message window 메시지 윈도우 – 프로세싱 IDE 프로그램의 메시지를 표시하는 창

아두이노와 프로세싱의 IDE는 매우 비슷하다. 현재 작업하고 있는 소프트웨어가 아두이노인지 프로세싱인지 항상 확인하기 바란다.

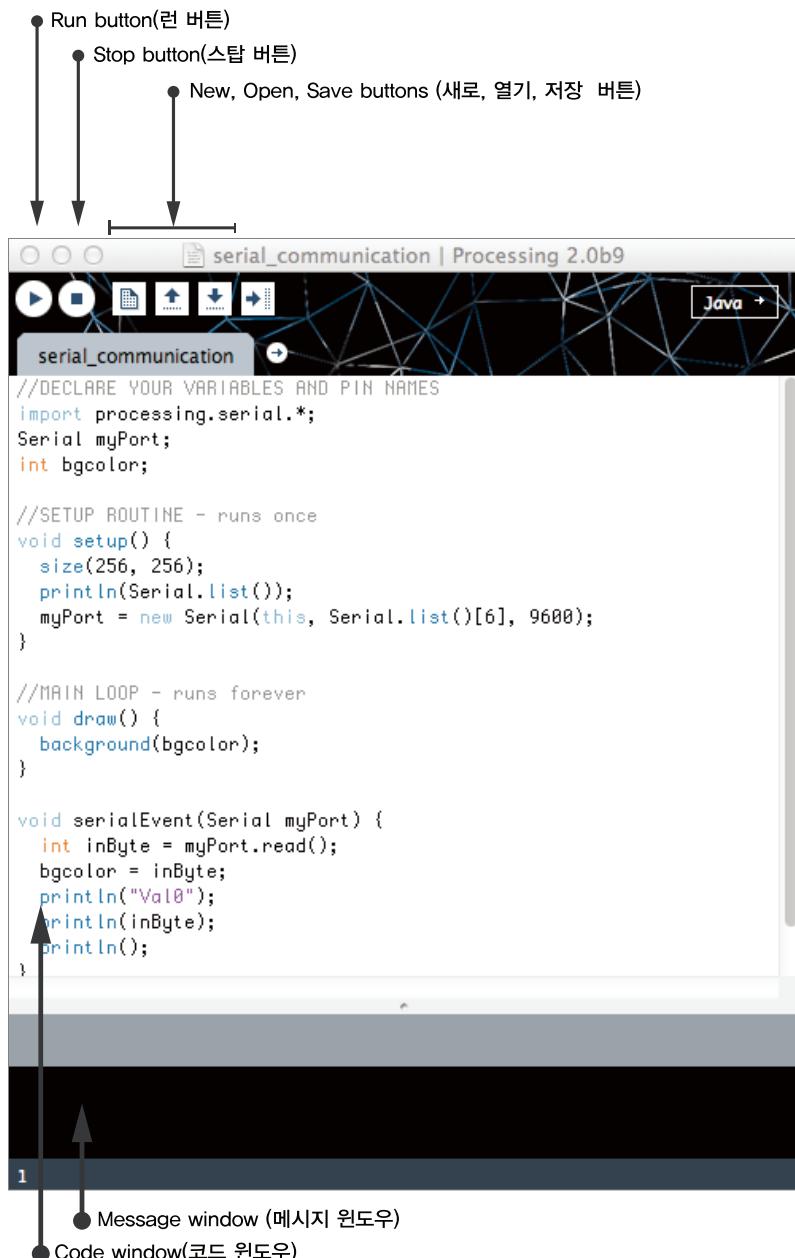


그림 1.5

자료 2

전자 부품

예제들로 기초를 다진 뒤 여러분만의 프로젝트를 만드는 것이 이 책의 목표이다. 이 책에 수록된 예제들을 작업하기 위해서는 다음과 같은 전자부품이 필요하다.

그림 1.6

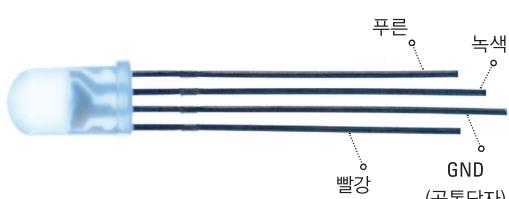


LED 발광다이오드 여덟 개

LED는 다이오드의 일종이다. 다이오드는 한 방향으로만 전류가 흐른다.

LED는 긴 다리[애노드](#)와 짧은 다리[캐소드](#)로 이루어져 있다. 애노드는 파워에 연결하며 캐소드는 그라운드에 연결한다. 반대로 연결하게 되면 LED에 불이 들어오지 않는다.

빨간색, 노란색, 초록색 LED 정도만 준비하면 충분하지만 좀 더 예쁘게 만들고 싶다면 다양한 색을 사용해도 상관없다(그림 1.6).



RGB LED 한 개 – 공통 캐소드(선택사항)

RGB LED는 LED 하나에서 세 가지 색상의 불이 모두 들어오므로 세 개의 LED를 포함하고 있는 것이라 보면 된다. 다른 네 개가 있으며, 그중 가장 긴 다리를 공통단자[common ground](#)라 한다(그림 1.7).

저항 Resistor

저항은 다양한 값으로 출시되며, 저항값의 단위는 ‘옴’을 사용한다. 이 책의 예제에서 우리는 주로 아래 두 가지 크기의 저항을 사용할 것이다(그림 1.8).

저항 값은 곁에 표시된 색띠로 구별한다.

	220Ω
	빨간색, 빨간색, 갈색 띠
	10kΩ
	갈색, 검은색, 주황색 띠



그림 1.8

10k옴 포텐시오미터 potentiometer 세 개

포텐시오미터 역시 다양한 값으로 출시가 된다. 우리는 10k옴을 사용할 것이다.

포텐시오미터는 다른 말로 가변저항기라고도 한다. 위에서 봤던 저항과 다르게, 10k옴 포텐시오미터는 저항 값이 정해져 있는 것이 아니라 0옴부터 10k옴까지 범위 내의 저항을 자유자재로 만들어 낼 수 있다(그림 1.9).

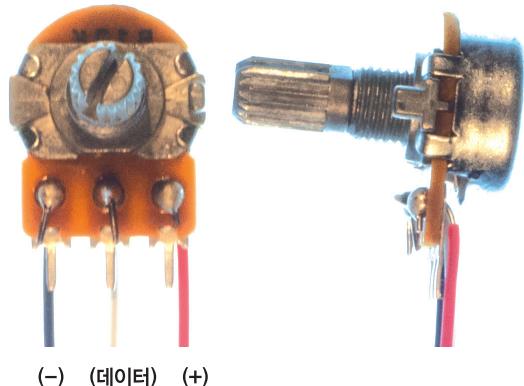


그림 1.9

순간 스위치 momentary switches 세 개

스위치는 버튼을 누르기 전까지는 그저 잘려진 전선들이 연결되어 있는 것에 불과하다. 우리는 앞으로 두 종류의 스위치를 사용할 것이다.

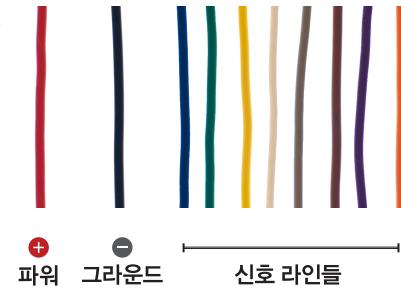
순간 스위치는 스위치를 누를 때만 전선이 순간적으로 연결되는 스위치다(누르면 연결이 되고 누르지 않으면 연결되지 않음 – 그림 1.10).



그림 1.10

12 말하고 듣고 대화하는 Physical Computing 아두이노

그림 1.11



솔리드 코어 전선 22-28 gauge solid core wire

어떤 색을 사용하든 상관없지만 기본적으로 빨간색, 검은색, 그외 다른 색 하나, 총 세 가지 정도를 준비해라(그림 1.11).

TIP

전자부품의 핀과 핀 사이의 간격을 피치(pitch)라 한다. 여기서는 5.08mm 피치의 단자판과 2.54mm 피치의 헤더를 사용할 것이다. 부품의 피치가 다를 경우 브래드보드에 맞지 않으므로 부품의 피치가 맞는지 반드시 확인하기 바란다.

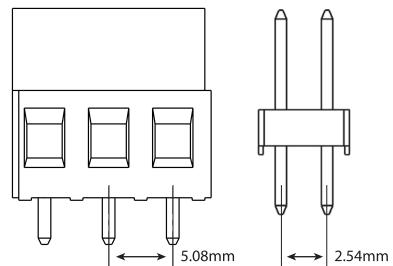


그림 1-12 단자판 Terminal Blocks 과 헤더 Headers

그림 1-13 3-5볼트 DC 모터



그림 1.14

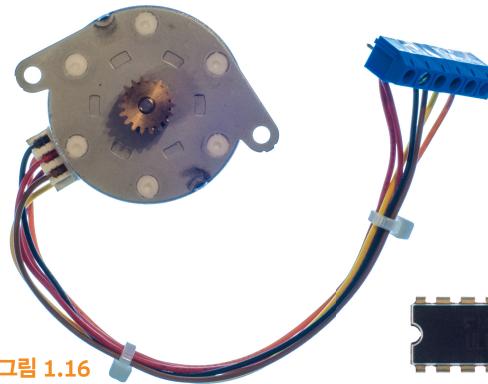
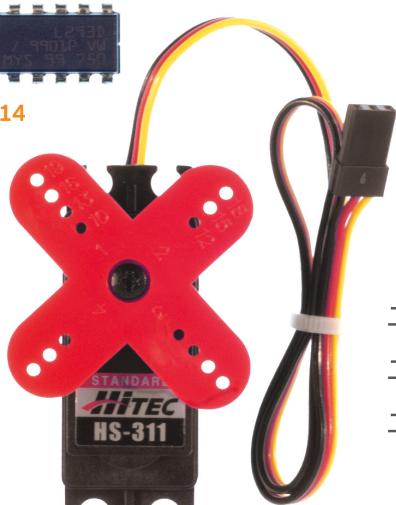


그림 1.17

그림 1-14 L293 하프브릿지 모터드라이버 half-bridge motor driver

그림 1-15 180도 회전 서보 모터 A 180 degree servo motor

그림 1-16 스텝 모터-바이폴라 또는 유니폴라

A stepper motor – either bipolar or unipolar

그림 1.15

그림 1-17 ULN2004 트랜지스터 어레이 Transistor Array

재료 3

도구

앞으로 소개될 예제들을 작업하기 위해서는 전기부품뿐만 아니라 도구(연장) 역시 필요하다.

전선 스트립퍼 wire strippers

언뜻 보기에는 가위나 플라이어(펜치)와 비슷하게 생겼지만, 한쪽 끝에 톱니 모양 날이 달려 있어서 전선의 피복을 벗길 수 있도록 되어 있다는 점이 다르다(그림 1.18).



그림 1.18

작은 플랫헤드 & 필립스 헤드 스크류드라이버

A Small Screwdriver - flat and philips head

작은 스크류드라이버는 가지고 있으면 언제든지 유용하게 사용할 수 있다(그림 1.19).

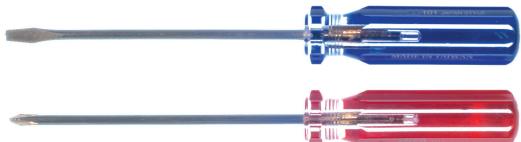


그림 1.19

14 말하고 듣고 대화하는 Physical Computing 아두이노



그림 1.20

니퍼 Nipper 와 바늘코 플라이어 Needle nose pliers

스냅스는 일반 가위로는 자르기 힘든 전선이나 집타이^{zip ties}를 자를 때에 사용하는 아주 강력한 가위다. 끝이 뾰족한 특이한 종류의 플라이어로, 작은 전선이나 금속을 다룰 때 사용한다(그림 1.20).



그림 1.21

납땜기 Soldering Iron/Solder (선택사항)

부품에 납땜을 할 때는 납땜기를 사용하며 가격은 저렴한 편이다. 대신 납땜기를 사용할 때에는 안전에 각별히 유의해야 한다(그림 1.21).