Version 0.0.1
Manual Number: 88

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Physikpraktikum für Vorgerückte (VP)
vp.phys.ethz.ch

Instruction Manual

Advanced Physics Lab
Electronics D

Modern Aspect of Data Taking and Processing with a Microcontroller

Pirmin Berger & Michael Reichmann

Abstract

The experiment "Digital Electronic" provides an introduction into modern data taking by operating simple digital circuits utilising an Arduino board. This manual will inform you about the Arduino board, the installation of the required software and the electrical components you will have to use. Basic knowledge on electronics, how to use oscilloscopes, bread boards and power supplies is recommended.

During the experiment you will learn how to build a circuit that measures the temperature, how to operate it using the Arduino board and to modify and improve it using more components.

In case you should already have previous knowledge we will provide many more material and own ideas on implementation are very welcome and can be built consulting the assistants.

ETH Zürich, Switzerland, May 17, 2018

Contents

1 Introduction

Arduino is a computer company, project and user community based on easy-to-use hardware and software, that designs and manufactures single-board microcontrollers and microcontroller kits for building digital devices and interactive objects that can sense and control objects in the physical and digital world. All products are distributed as open-source hardware and software, and it's licences permit the manufacture of Arduino boards and software distribution by anyone. The boards are commercially available in preassembled form, or as do-it-yourself (DIY) kits.

The Arduino project started in 2003 as a program for students without a background in electronics and programming at the Interaction Design Institute in Ivrea (Italy). The aim was to provide a low-cost and easy way for novices and professionals to create devices that interact with their environment using sensors and actuators. The actual name Arduino comes from a bar in Ivrea, where some of the founders of the project used to meet. The bar was named after Arduin of Ivrea, who was the margrave of the March of Ivrea and King of Italy from 1002 to 1014 [5].

In order to work with the Arduino Boards the Arduino programming language, based on Wiring, and the Arduino Software (IDE), based on the Processing are used [1]. Both Wiring and Processing are programming languages using a simplified dialect of features from the programming languages C and C++.

1.1 Arduino Board

The original boards were produced by the Italian company Smart Projects but as of 2018, 22 versions of the Arduino hardware have been commercially produced. The information and specifications of these boards can be found on this website. During this Lab you will work the Arduino Uno shown in Figure 1.

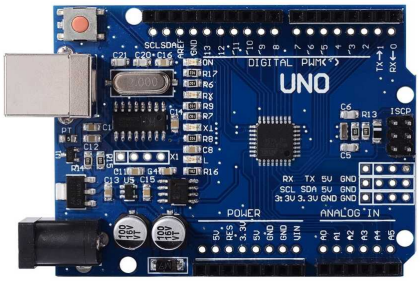


Figure 1: Arduino Uno.

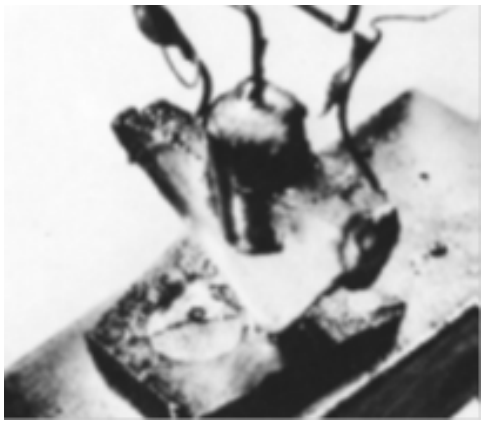
The Arduino Boards use a variety of microprocessors and controllers and are equipped with sets of digital and analogue input/output (I/O) pins that may be interfaced to various expansion boards or Breadboards (shields) and other circuits. The boards feature serial communications interfaces, Universal Serial Bus (USB) on some models, which are also used for loading programs from personal computers.

Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online. You can tell the board what to do by sending a set of instructions to the microcontroller.

1.2 Transistor

The invention of the transistor was announced in 1948 by the American physicists, J. Bardeen and W. H. Brattain as a new type of amplifying device made from semiconducting crystals. At that time almost no one could have foreseen the revolutionary developments that were to follow, developments so important and far-reaching as to change the whole outlook of the

science and technology of electronics. The physical principles of a transistor had been worked out in conjunction with their colleague, W. Shockley. In recognition of their work the three physicists were awarded jointly the Nobel Prize for Physics in 1956.



(a) Point-contact transistor.

(b) Bardeen, Brattain and Shockley.

The term "transistor" is a combination from the words *trans*former and res*istor*, since the device is made from resistor material and transformer action is involved in the operation. In the beginning only point-contact transistors existed, but due to their vulnerability to mechanical shock they were soon replaced by junction transistors which are firmly established now [3].

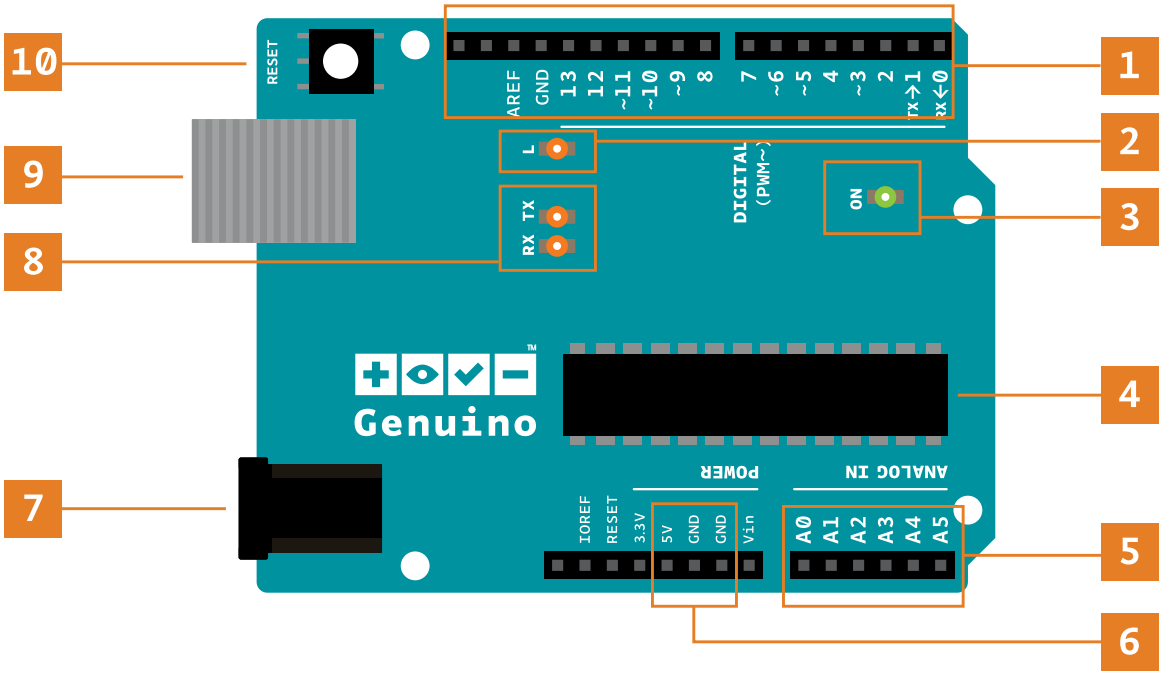
The transistor is the key active component in practically all modern electronics. It is considered as one of the greatest inventions of the 20th century. Its importance in today's society rests on its ability to be mass-produced using a highly automated process that achieves astonishingly low per-transistor costs (10 femto$/transistor) [2].

Although billions of individually packaged (discrete) transistors are produced every year the vast majority of transistors are now produced in integrated circuits (ICs). A logic gate consists of up to about twenty transistors whereas an advanced microprocessor, as of 2009, can use as many as 3 billion transistors. In 2014, about 10 billion transistors were built for each single person on Earth [2].

2 Basics

This section will give you the basic information about the components we are using in this lab.

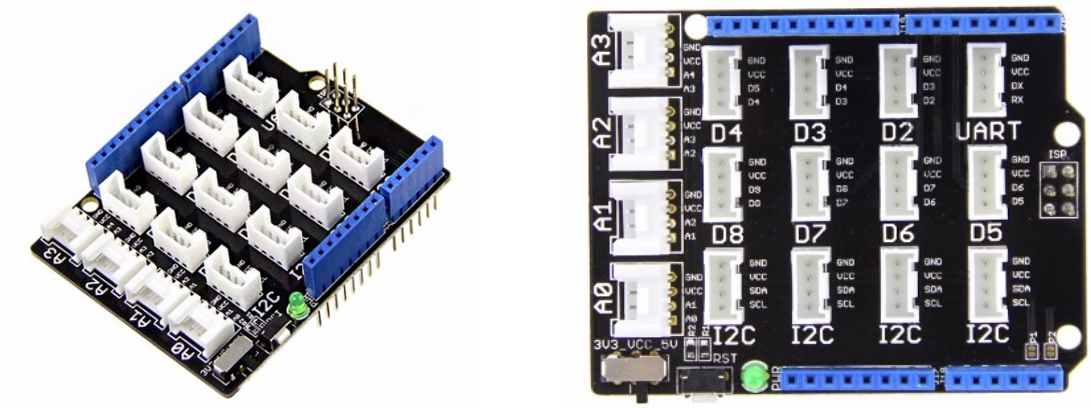
2.1 Arduino Uno



1. **Digital pins:** used with digitalRead(), digitalWrite(), and analogWrite() methods, analogWrite() only works on pins with the PWM symbol
2. **Pin 13 LED:** only built-in actuator
3. **Power LED**
4. **ATmega microcontroller**
5. **Analogue in:** used with analogWrite() method
6. **GND and 5V pins:** provide 5 V power and ground (GND) to the circuits
7. **Power connector:** additional power supply, accepted voltages: $7 \sim 12$ V
8. **TX and RX LEDs:** indicate communication between Arduino and computer
9. **USB port:** used for powering and communication with computer
10. **Reset button:** resets the ATmega microcontroller

2.2 Grove Base Shield

The so called shields are printed circuit expansion boards, which plug into the normally supplied Arduino pin headers. The Grove Base Shield is one example that simplifies projects that require a lot of sensors or LEDs. With the Grove connectors on the base board, one can add all the Grove modules to the Arduino Uno very conveniently.



(a) Side view. (b) Top view.

Figure 3: Grove shield.

There are 16 Grove connectors on the Base Shield which are shown in Table 1. Apart from the connectors the board also consists of a reset (RST) button, a green LED to indicating power status, a toggle switch and four rows of pinouts, which is equivalent to the pinout of the Arduino.

Specification	Name	Quantity
Analog	A0/A1/A2/A3	4
Digital	D2/D3/D4/D5/D6/D7/D8	7
UART	UART	1
I2C	I2C	4

Table 1: Base Shield connectors.

Every Grove connector has four wires, one of which is the voltage common collector (VCC). Since some micro-controller main boards need need different supply voltages the power toggle switch allows you to select the suitable voltage. In the case of the Arduino Uno a voltage of 5 V is required [4].

2.3 The Software

2.3.1 Arduino IDE

All you require to write programs and upload them to your board is the Arduino Software (IDE). There are two options how to use it:

1. Online IDE

- no installation required
  - needs plugin if you want to upload sketches from Linux
- requires you to create account with e-mail verification
- save sketches in cloud (available from all devices)
- always most up-to-date version
- instructions on the website

2. Desktop IDE

- if you want to work offline
- installation usually very straightforward and in has general no dependencies
- if you need help, follow installation instructions depending on your operating system (OS)
  - Linux
  - Mac OS X
  - Windows

2.3.2 Board Drivers

First the Arduino board has to be connected to the computer via the USB cable which will power the board indicated by the green power (PWR) LED. The board drivers should then install automatically in Linux, Mac OS X and Windows. If the board was not properly recognised, follow these instructions.

2.4 Programming

In order to get a feeling for the programming language it is recommended to have a look at the examples first which can be found under: **`File > Examples`**
A list of the most common methods is shown in Table 2. For more information look at the detailed description.

Category	Method Syntax	Description
	digitalRead(pin))	reads the value from the digital pin (HIGH or LOW)
Digital I/O	digitalWrite(pin, value)	writes HIGH or LOW value to the digital pin
	pinMode(pin, mode)	configures the pin as INPUT or OUTPUT
	analogRead(pin)	reads the value (0-1023) from the pin
Analogue I/O	analogWrite(pin, value)	writes an analogue value (PWM wave) to the pin
	analogReference(type)	configures the reference voltage used for analogue input
	tone(pin, f, duration)	generates a square wave of frequency f [Hz] for a duration [ms]
Advanced I/O	pulseIn(pin, value)	returns the time [ms] of a pulse, if value is HIGH: waits until HIGH and stops when LOW
	delay(time)	pauses the program for a time [ms]
Time	micros()	returns time since starting the program [µs]
	millis()	returns time since starting the program [ms]
	constrain(x, a, b)	constrains a number x to be in range [a, b]
	map(x, a, b, c, d)	re-maps x from range [a, b] to range [c, d]
Math	random(a, b)	returns pseudo-random number in range [a, b]
	abs(value)	return the absolute value
	⋮	further general math commands
	Serial.begin(speed)	initialise serial communication at speed [bit/s]
Serial	Serial.print(value)	prints the value to the serial port
	Serial.println(value)	prints the value to the serial port with [\r\n]
	Serial.read()	reads incoming serial data

Table 2: Most common methods for controlling the Arduino board and performing computations.

2.5 Project Management With git

Git is a version control system for tracking changes in computer files and coordinating work on those files among multiple people. It is primarily used for source code management in software development, but it can be used to keep track of changes in any set of files [6].

More explanation?

2.6 Voltage Divider

A voltage divider is a passive linear circuit that produces an output voltage V_{out} that is a fraction of its input voltage V_{in}. Voltage division is the result of distributing the input voltage among the components of the divider. A simple example of a voltage divider is two resistors connected in series, with V_{in} across the resistor pair and V_{out} emerging from the connection between them as shown in Figure 4.

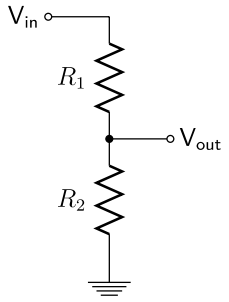


Figure 4: Voltage divider.

Resistor voltage dividers are commonly used to create reference voltages, or to reduce the magnitude of a voltage so it can be measured. Using Ohm's law one can easily derive the formula for V_{out}:

$$V_{\text{out}} = \frac{R_2}{R_1 + R_2} \cdot V_{\text{in}} \tag{1}$$

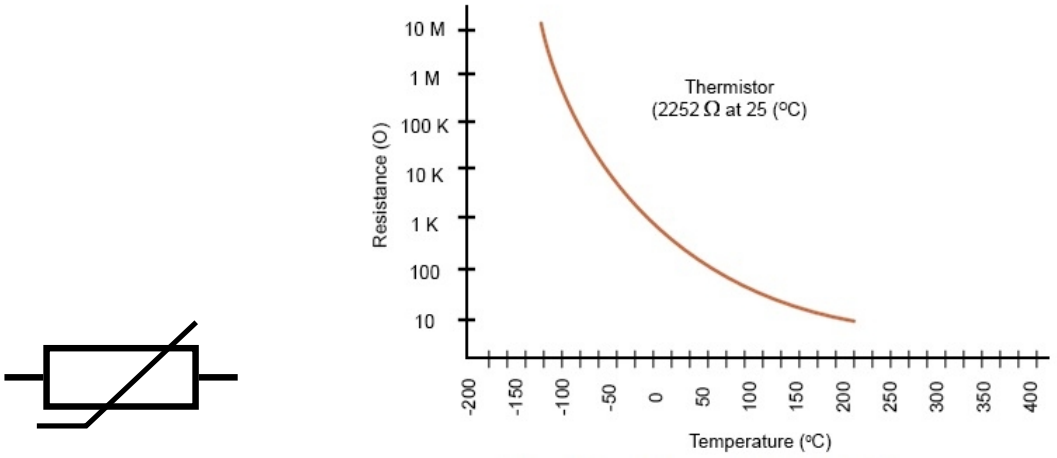
2.7 Thermistor

If, for a given temperature, the current is directly proportional to the applied voltage the electrical component is said to obey Ohm's law. Such components are called linear resistors. If a component does not meet this requirement it is termed a non-linear resistor, which falls into two classes – the temperature-sensitive type and the voltage-sensitive type. The temperature-sensitive types are often known as thermistors and change the resistance very reproducible. The word is a portmanteau of *THERM*ally-sensitive and res*ISTOR*.

They consist of the sintered oxides of manganese and nickel with small amounts of copper, cobalt or iron added to vary the properties and the physical shape is usually a bead, rod or a disc. The electronic symbol is shown in Figure 5a. The resistance is given by

$$R = R_0 \cdot \mathrm{e}^{-B\left(\frac{1}{T_0} - \frac{1}{T}\right)} \tag{2}$$

where R_0 and B are constants depending upon the composition and physical size, T is the temperature in °K, and T_0 the absolute zero. Thermistors can be classified into two types, depending on the classification of B. If B is positive, the resistance increases with increasing temperature, and the device is called a positive temperature coefficient (PTC) thermistor, or posistor. If B is negative, the resistance decreases with increasing temperature, and the device is called a negative temperature coefficient (NTC) thermistor.



(a) Electronic symbol.

(b) NTC resistance/temperature characteristic.

Figure 5: Thermistor properties

2.8 Temperature Sensor

Thermistors have very widespread applications as thermometers. We know that a NTC thermistor varies its resistance as a function of the temperature, but resistance is not the easiest parameter to measure. In our case we want to feed a signal into an analogue-to-digital converter (ADC) to treat it numerically and compute the actual temperature. Now, the ADC of the Arduino requires a voltage at it's input.

An easy solution is to install the NTC in a voltage divider as shown in subsection 2.6. It requires only one additional fixed resistor R_0. V_{in} is the reference voltage used of the ADC and V_{out} the measured voltage. So what you will measure in the end is just a digital 10 bit value corresponding to the divided voltage where the maximum of $2^{10} - 1 = 1023$ corresponds to V_{in}. In order to convert it into the resistance of the thermistor we have to use Equation 1:

$$R = \left(\frac{V_{\text{in}}}{V_{\text{out}}} - 1\right) \cdot R_0 = \left(\frac{1023}{V_{\text{meas}}} - 1\right) \cdot R_0 \tag{3}$$

Now we need to convert the resistance into the temperature using Equation 2:

$$\frac{1}{T} = \frac{\ln\left(\frac{R}{R_0}\right)}{B} + \frac{1}{T_0} \tag{4}$$

Note that this temperature will be in °K!

2.9 Bipolar Junction Transistor

A bipolar junction transistor (BJT) is a type of transistor that uses both electron and hole charge carriers. For their operation, BJTs use two junctions between two semiconductor types, n-type and p-type and thus can be manufactured in two types, NPN and PNP. The basic function of a BJT is to amplify current which allows it to be used as amplifiers or switches, giving them wide applicability in electronic equipment.

(a) NPN (b) PNP

Figure 6: Electronic symbols of the BJT.

2.9.1 Working Principle

Since a transistor consists of two pn junctions within a single crystal, transistor action can be explained with Figure 7. For diagrammatic purposes the base region is shown fairly thick, but in fact the pn junctions are very closely spaced and the active portion of the base is very thin.

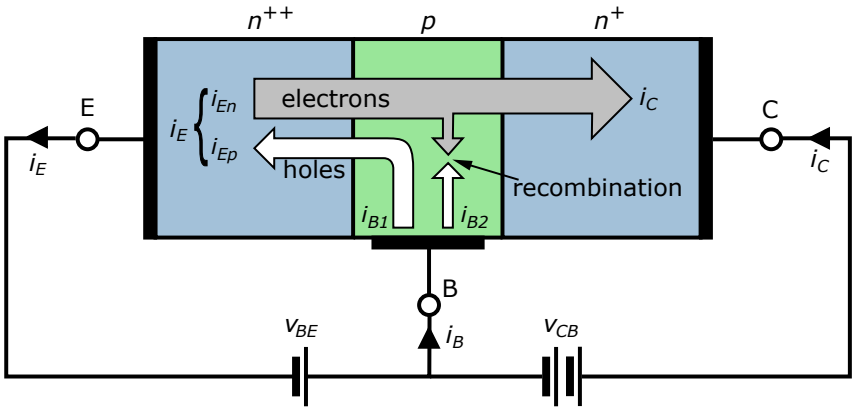


Figure 7: Diagrammatic representation of the amplifying action of a transistor.

The charge flow in a BJT is due to diffusion of charge carriers across a junction between two regions of different charge concentrations. The regions of a BJT are called emitter, collector, and base. Typically, the emitter region is heavily doped compared to the other two layers, whereas the majority charge carrier concentrations in base and collector layers are about the same.

In the absence of any external applied voltages the collector and emitter depletion layers are about the same thickness, the widths depending upon the relative doping of the collector, emitter and base regions. During normal transistor operation the emitter-base junction is forward biased so that current flows easily in the input or signal circuit. The bias voltage V_{BE} is about 200 mV for germanium transistors and about 400 mV for silicon devices. The collector-base junction is reverse-biased by the main supply voltage V_{CB} typically with 4.5 V, 6 V and 9 V. The collector junction is therefore heavily reversed-biased and the depletion layer there is quite thick.

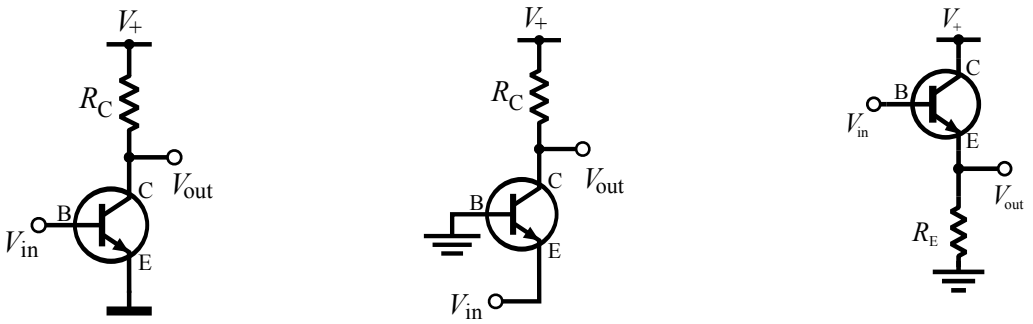
The injection of a hole into the base region by a signal source will now be considered. Once in the base, the hole attracts an electron from the emitter region. The recombination of the hole and electron is not likely to occur however, since the base region is lightly doped compared with the emitter region and so the lifetime of the electron in the base region is

quite long. In addition the base is extremely thin so the electron, instead of combining with the signal hole or with a hole of the p-type base material, diffuses into the collector-base junction. The electron then comes under the influence of the strong field there and is swept into the collector and hence into the load circuit. In a good transistor many electrons pass into the collector region before eventually the signal hole is eliminated by combination with an electron. A small signal current can thus give rise to a large load current i_C, and so current amplification has taken place. In practical transistors for every hole injected into the base 50 to 250 electrons may be influenced to flow into the collector region. The current gain or amplification is therefore 50 to 250. It is usually given by the symbol β.

An PNP transistor behaves in a similar fashion except that electrons are injected into the base and holes flow from the emitter into the collector. To maintain the correct bias conditions the polarity of the external voltages must be reversed.

Figure 8 shows the three basic transistor arrangements. The common emitter mode is the most commonly used arrangement for voltage amplification because very little current is required from the signal source.

The common base mode of operation is also capable of voltage amplification. This is achieved by the use of high values of load resistor. The transistor is able to maintain the current through the load because the device is a good constant current generator [3].



(a) Common emitter. (b) Common base. (c) Common collector.

Figure 8: The three basic amplifier arrangements.

2.9.2 Common Collector

The common collector circuit shown in Figure 8c has a voltage gain of a little less than unity and so is useless as a voltage amplifier. However this circuit has very important impedance matching properties and is typically used as a voltage buffer. In this circuit the base serves as the input, the emitter is the output, and the collector is common to both.

The voltage gain is just a little less than one since the emitter voltage is constrained at the voltage drop over the diode of about 0.6 V (for silicon) below the base. The transistor continuously monitors V_D and adjusts its emitter voltage almost equal (less V_{BEO}) to the input voltage by passing the according collector current through the emitter resistor R_E. As a result, the output voltage follows the input voltage variations from V_{BEO} up to V_+; hence also the name, emitter follower. This circuit is useful because it has a large input impedance, so it will not load down the previous circuit and a small output impedance, so it can drive

low-resistance loads

2.10 Operational Amplifier

Should be covered in AP next semester

3 Setup and Experimental Procedure

3.1 Setting up the Arduino

3.2 Blinking LED on Bread Board

3.3 Grove Temperature Sensor

In this experiment you will work with the Grove - Temperature Sensor V1.2. It uses a NTC thermistor to detect the ambient temperature. The specifications are shown in Table 3.

Specification	Value
Voltage	$3.3 \sim 5.0\,\mathrm{V}$
Zero power resistance	$(100 \pm 1)\,\mathrm{k\Omega}$
Operating temperature range	$-40 \sim +125\,^{\circ}\mathrm{C}$
Nominal B-constant	$4250 \sim 4299\,\mathrm{K}$

Table 3: Grove-Temperature Sensor V1.2 specifications.

3.4 Building Your Own Temperature Sensor

During this lab you will work with the NTC $100\,\mathrm{k\Omega}$ B57164-K104-J thermistor.

3.5 Building a Heating System

3.6 Building a Cooling System

3.7 Read Out the Fan Speed (Advanced)

3.8 Adding a Display (Advanced)

4 Analysis / Protocol

List of Acronyms

IDE integrated development environment

DIY do-it-yourself

USB Universal Serial Bus

I/O input/output

GND ground

RST reset

PWR power

IC integrated circuit

VCC voltage common collector

OS operating system

PTC positive temperature coefficient

NTC negative temperature coefficient

ADC analogue-to-digital converter

BJT bipolar junction transistor

References

[1] Arduino. Introduction to Arduino. https://www.arduino.cc/en/Guide/Introduction, 2018. [Online; accessed 15-May-2018].

[2] Dan Hutcheson. A look at Moore's Law. https://spectrum.ieee.org/computing/hardware/transistor-production-has-reached-astronomical-scales, 2018. [Online; accessed 15-May-2018].

[3] G.H. Olsen. *Electronics: A General Introduction for the Non-Specialist*. Springer US, 2013.

[4] Seeed. Base Shield V2. https://www.seeedstudio.com/Base-Shield-V2-p-1378.html, 2018. [Online; accessed 15-May-2018].

[5] Wikipedia contributors. Arduino — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Arduino&oldid=839287116, 2018. [Online; accessed 15-May-2018].

[6] Wikipedia contributors. Git — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Git&oldid=840401990, 2018. [Online; accessed 16-May-2018].