User Manual

# List of materials and Parts

Full model with labels

# Set up Procedure

1. Align both parts of the frame together as shown below and secure them together.

(Insert picture)

2. Place boards for the track onto the frame. (insert picture)

3. Put track together as shown below. Observe that three pieces must always remain together permanently. (insert picture)

4. Snap track into the slots created on the board pieces. (insert picture)

5. Place cart on the track, ensure that it is on the track properly. (insert picture)

6. Connect the track to power and now it is ready to be turned on (when the rest of the machinery is in place as instructed by the ME team).

# Take Down Procedure

1. Turn apparatus off.

2. Remove the cart from the track.

3. Remove track from the boards (should snap out easily).

4. Take track apart. CAUTION: Three pieces including the switch must always remain together permanently! Do not take these apart. (insert picture)

5. Separate the frame.

6. Clean any parts as needed and pack up.

# Cleaning

## Track

The track is stainless steel and can be cleaned using a damp cloth dipped in a mixture of warm water and mild dish detergent. Do not submerge the track in water. The track should be able to be clean by wiping it down.

Parts to be wary off: anything with electronics, such as the switch and...

Cleaning products that should NOT be used: Windex

## Platform

The platform is made of coated wood and can be wiped off using a damp cloth.

Cleaning products that should NOT be used: Windex

## Cart

DO NOT SUBMERGE IN WATER. Be wary of the electronics.

Can be wiped down with a damp towel (NOT SOAKING WET. Be cautious of the motors underneath the cart).

Cleaning products that should NOT be used: Windex

The brushes on the cart can be unclipped and cleaned.

# Troubleshooting Guide

|  |  |  |
| --- | --- | --- |
| Problem | Potential causes | Potential Solutions |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# Technical Reference

## Introduction

The aim of the technical reference is to provide details about how everything works from the EP side of the project. This includes an overview of the code, as well as details of circuits and operation of sensors. This section may prove helpful if basic troubleshooting does not resolve problems. It also serves as a means of passing on knowledge to any future individuals that may need to alter the current design.

## System Overview

What each box is, how they are connected, power, communication

## I2C and the Master Arduino

Some code explanation and pull resistor description

## Train and Sensor Arduino

What code is doing generally

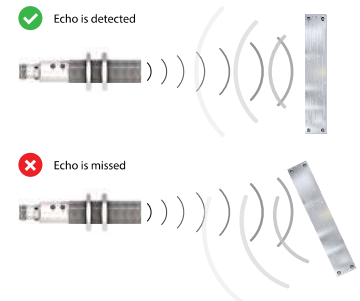
## Train, Tracks, and Switch

controls, digital pot description of operation, power circuit, switch operation

## Ultrasonic Sensors

There are two RU130U-M18E-LU8X2-H1151 ultrasonic sensors from Turck in the Cookie Lab, each detecting the position of the cart as it passes under the dispensers. Ultrasonic sensors work by sending out pulses of ultra-high frequency sound (above the limits of human hearing) and listening for the echoes of objects in its operating range.

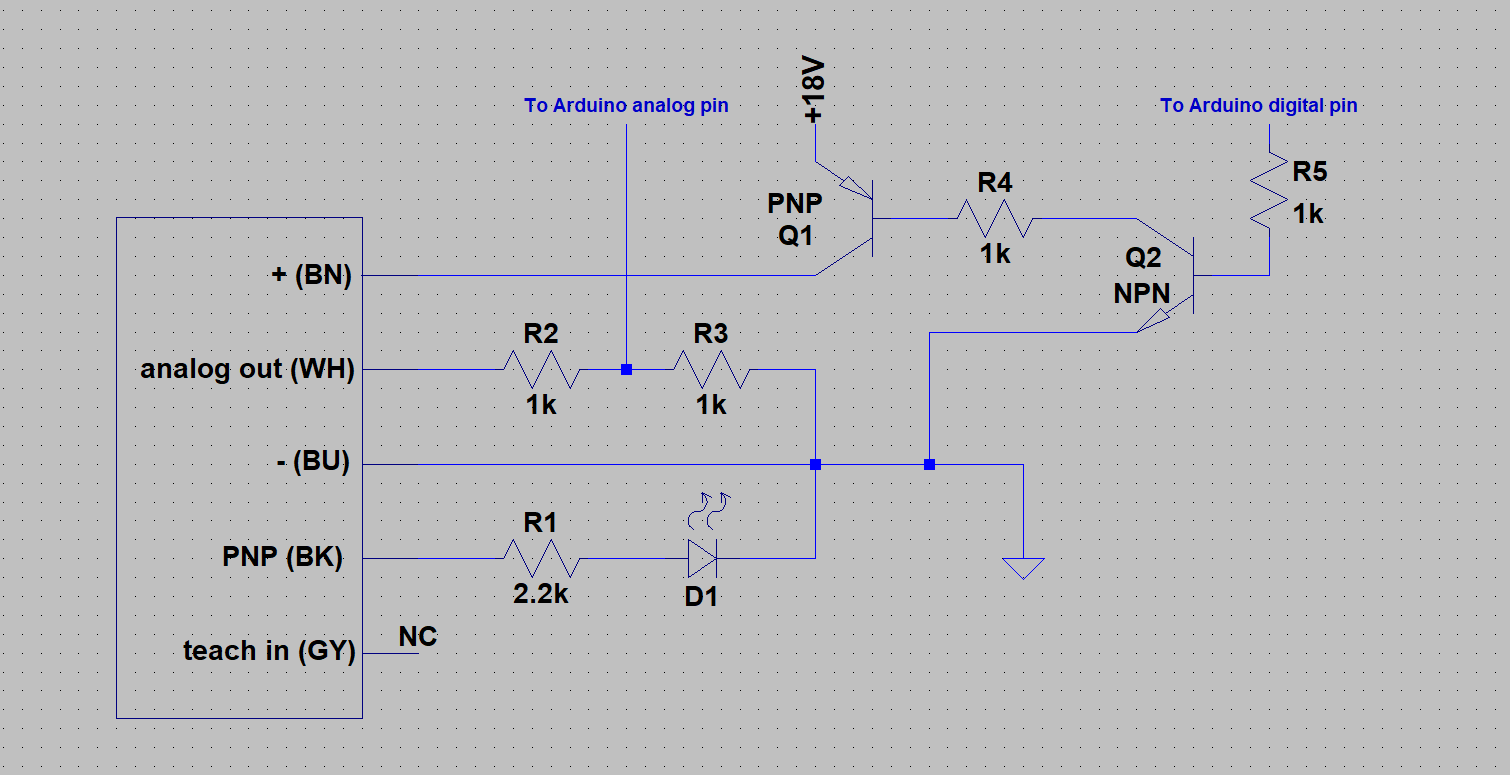
Generally, the sensors can sense things within a cone spreading out from their face. This means that, while the sensor is running, anything in the general vicinity of the track, not just the train, could be sensed and lead to an erroneous reading. However, since the sensor rejects any reflections that are too weak, it does not sense the dispenser frame. The best reflections come from large flat objects perpendicular to the propagation direction of the sound (hence the flat target on the cart). The sensing regions of the sonic cone for different target sizes are shown in the attached data sheets for the sensor.



The sensor also prefers reflections from things that are closer to it than things that are further away. This means that if it senses something in front of the cart it will output that reading instead of the cart further away. If the sensor isn’t picking up the cart movement but has a static output, then it might be sensing something else closer to the sensor. Additionally, if both sensors are turned on at the same time, they may be picking up interference from each other, so each sensor should be turned off when the other is turned on. The current code already handles this.

The range of the sensors is 150-1300 mm. There are two types of outputs on the sensor: an analog output, and a PNP switching output. The analog output is 0-10 V and depends upon the position of the target in the sensing range. For example, if the target is in the exact middle of the sensing range the output would be 5 V. The first 150 mm is a dead zone that the sensor cannot accurately measure. The analog output of the sensor in the dead zone, as well as anything past the max range is around 0 V. The PNP output acts like a simple switch connected to the input power. Whenever there is anything in the sensing range, the PNP output will be at the same voltage as the input power. If the sensor doesn’t sense anything, the PNP output will be floating (usually it is pulled to ground).

The following diagram is the circuit connected to the ultrasonic sensor. In the diagram, BN is referring to the brown wire, WH is the white wire, BU is the blue wire, BK is the black wire and GY is the grey wire.



The wires are connected as follows:

**+:** This is the input power. It is connected to two transistors that are used to control power to the sensor. The transistors act as switches. The PNP transistor allows current to flow from the 18 V supply when a ground is connected to its base (at resistor R4). The NPN transistor allows current to flow from the base of the PNP to ground when it receives 5 V at its base (at resistor R5). The Arduino supplies the 5 V needed to switch the NPN, which connects the PNP to ground allowing 18 V to go to the sensor. The two resistors are there to limit current.

**analog out:** This is the analog output of the sensor. Since the Arduino can only read 0-5 V on its analog pins, two resistors are used as a voltage divider to cut 0-10 V down to 0-5V. The resistors are equal values so that they each drop half of the voltage.

**-:** This is the ground wire of the sensor. This ground is kept common with the Arduino ground so that the voltage readings are accurate.

**PNP:** This is the PNP output. The output is hooked up to a current limiting resistor and an LED. The output will be 18 V when something is in the sensing range.

**teach in:** The teach in pin is used to set a decreased range for the sensor, create a window function, or to flip the analog output. The sensor is already taught correctly and should not be altered. The full range of the sensor is needed. Do not flip the output. This will make the default analog value around 6 V (12 V actual) instead of 0 V. This will damage the Arduino. If the sensor must be taught for some reason, short the teach wire with the power input to start the process. More details on how to teach the sensor are in the data sheets.

## Capacitive Sensor

## Other Arduinos