Name: \_\_\_\_\_\_\_\_\_\_ Class: \_\_\_ Date: \_\_\_\_\_\_\_\_

# Task 1

# Weather station

In this task, you will begin by building the sensor station. Then, we will convert it into a weather station that will transfer your measured data to a web server via MQTT protocol (Message Queuing Telemetry Transport). It will complete a weather forecast and will regularly transmit a camera image, in order to function as a webcam.

## Construction task

Build the IoT station according to the building instructions. Connect the sensors and actuators to the TXT 4.0 according to the wiring diagram.

During the experimental tasks, data will be transferred to a web server using the MQTT protocol, and the transmitted values will be visualised there. To do so, create an account in the fischertechnik cloud at ([www.fischertechnik-cloud.com](http://www.fischertechnik-cloud.com)).

## Programming tasks

**1. Measuring humidity, temperature, and air pressure**

The environmental sensor, a Bosch sensor with type designation BME680, includes a temperature, moisture, and pressure sensor, among other features. The sensor values are transmitted to the TXT via the I²C protocol. You can query the sensor values using the relevant Blockly commands.

Write a Blockly program that turns the IoT station into a weather station which outputs the measured humidity (in %), the temperature (in °C), and the air pressure (in hPa) to the display of the TXT, and updates these once per second.

**2. Barometer**

You can use the measured air pressure to add a forecasting function to the weather station: if the air pressure is dropping, this indicates that precipitation is imminent; if the air pressure is rising, this indicates dry and sunny weather.

At sea level, the air pressure is between around 950 and 1050 hPa; it fluctuates by approx. ±50 hPa around a mean value of about 1000 hPa. If you divide the entire measurement range into three segments, you can use the measured air pressure to make a forecast (“rainy”, “variable”, “dry”).

However, the air pressure drops at higher elevations. The values measured by the sensor, therefore, should be interpreted differently at different elevations. You can take this into account by converting the measured air pressure $p\left(h\right)$ based on the “barometric formula” into a “theoretical air pressure” $p\_{0}$ at sea level [1]:

$$p\_{0}=p\left(h\right)·\left(\frac{T\_{0}}{T(h)}\right)^{5.255}hPa$$

In this formula,

* $p\left(h\right)$ the air pressure measured at the elevation $h$,
* $T(h)$ the temperature measured at this elevation (in Kelvins) and
* $T\_{0}$ the (theoretical) temperature at sea level, which drops approximately 0.0065 K with each meter of elevation:

$$T\_{0}=T\left(h\right)+0.0065·h \frac{K}{m}$$

Determine the elevation $h$ of your weather station above sea level, and use this to calculate the “air pressure at sea level” based on the measured air pressure $p\left(h\right)$ and temperature $T\left(h\right)$ in your Blockly program.

Then, add a weather forecast to your weather station to be shown on the display of the TXT.

## Experimental tasks

**1. Determining temperature using the NTC resistor**

In task 1 for the Base Set, you used the NTC resistor (thermistor) to determine the temperature using the Steinhart-Hart equation (see accompanying materials). To do so, you had to first determine three resistance values at different temperatures using the thermistor. The environmental sensor will now provide these to you with a high level of accuracy.

Connect the NTC resistor to I8, and add a measurement of the NTC resistance to your Blockly program. Enter the results of three measurements into the following table:

|  |  |
| --- | --- |
| Resistance value | Temperature |
|  |  |
|  |  |
|  |  |

*Note*: The environmental sensor will react to temperature changes at a long delay, due to the housing. Therefore, only complete the measurement once the temperature value of the sensor has stabilised.

You can now determine the coefficients of the Steinhart-Hart equation using the accompanying materials and the website indicated in the attachment [4], and also output the temperature calculated from the NTC resistance on the display of the TXT.

**2. Data visualisation on an IoT server**

Now, the measurement data from your weather station should be transmitted to a web server and displayed in a (prepared) dashboard there. To do so, connect your TXT to the account you set up previously in the fischertechnik cloud.

2a. Now, configure the dashboard so that the temperature, humidity, and air pressure are displayed. Click all displays you do not need to deselect them.

2b. The following example program demonstrates how the sensor data is transmitted to the IoT server: First, open the connection using your cloud account (“MQTT connect”); then, the current measured values from the sensor are transmitted at regular intervals in a fixed format (time stamp, temperature, moisture, pressure) to the cloud server (“MQTT publish”).



*IoT\_MQTT.ft*

Add the display on the IoT server to your Blockly program for the barometer.

Now, you can generate measurement series, display the measured values, and download them to the dashboard as a csv file. You can analyse and edit the csv file in a spreadsheet (such as in Excel).

**3. Webcam**

What would a weather station be without a webcam? Expand your program so that it transmits the camera image to the dashboard of the IoT server.

3a. Now, add a display of the camera image to the dashboard.

3b. The following example program shows you how transmission of the camera image works: Each image the camera “takes” is encoded in Base64 format. Then, the current image is transmitted to the IoT server once per second (“MQTT publish”).



*IoT\_Webcam.ft*

Expand your program so that it transmits the camera image to the IoT server.

You can save snapshots in a gallery and download the images individually using the dashboard.

Annex

# Task 1: Weather station

## Required materials

* PC for program development, local or via web interface.
* USB cable or BLE or WiFi connection for transmitting the program to the TXT4.0.
* Example programs “IoT\_MQTT.ft” and “IoT\_Webcam.ft”
* Account in the fischertechnik cloud

## Further information

[1] Wikipedia: [Barometric formula](https://en.wikipedia.org/wiki/Barometric_formula).

[2] Online diagram editor for creating state transition diagrams (drawio format): <https://www.diagrammeditor.de/>

[3] fischertechnik: [*NTC resistance*](https://content.ugfischer.com/cbfiles/fischer/Zulassungen/ft/36437-NTC-resistor.pdf). Data sheet, Art. no. 36437.

[4] Stanford Research Systems (SRS): [*Thermistor Calculator*](https://www.thinksrs.com/downloads/programs/Therm%20Calc/NTCCalibrator/NTCcalculator.htm). V1.1

[5] Dirk Fox: [*“Einmessen” eines digitalen Messgeräts (“Calibrating” a digital measurement device)*](https://ftcommunity.de/ftpedia/2013/2013-1/ftpedia-2013-1.pdf#page=39). ft:pedia 1/2013, p. 39-48.