

 **pvvx / ATC_MiThermometer** Public

forked from [atc1441/ATC_MiThermometer](#)

Custom firmware for the Xiaomi Thermometers and Telink Flasher via USB to Serial converter

 github.com/pvvx/pvvx.github.io/tree/master/atc_mithermometer

 View license


☆ **915** stars  **286** forks

☆ Star

 Notifications

<> **Code**

⦿ Issues **48**


 Pull requests

▶ Actions

 Projects

 Wiki

...

 master ▾

Go to file

This branch is up to date with master.

 Contribute ▾



pvvx https://github.com/pvvx/pvvx.github.io/tree/master/atc_mithermometer ...

2 days ago  **358**

[View code](#)

☰ **README.md**

Xiaomi Mijia LYWSD03MMC HW: B1.4, B1.6, B1.7, B1.9, B2.0 & Xiaomi Miaomiaoce MHO-C401 & CGG1 Mijia Bluetooth Thermometer Firmware

This repository contains custom firmware for several Bluetooth Thermometer & Hygrometer devices.

The custom firmware can be flashed *via a modern browser* and *over-the-air (OTA)* without opening the device and unlocks several customization options. You can go back to the original firmware at any time. OTA and configuration main page - [TelinkMiFlasher.html](#).

Key features

- Supports [connections using PIN-code](#) and encrypted **bindkey** beacon.
- **3 LCD Display Screens** (Looping): Temperature & Humidity & Comfort, Temperature & Battery Level, Clock
- **Measurement values recording** & Charting. See [Reading Measurements from Flash](#)
- **Adjustable correction offsets** and **Comfort zones**
- Improved battery life
- Concurrent support for Xiaomi, ATC and Custom Bluetooth Advertisement format
- Adjustable RF TX Power & Bluetooth advertising interval



Xiaomi Mijia (LYWSD03MMC)	Xiaomi Miaomiaoce (MHO-C401)	Qingping Temp & RH Monitor (CGG1-Mijia)
		

Table of content

- [Getting Started](#)
 - [Flashing or Updating the Firmware \(OTA\)](#)
 - [Configuration](#)
- [Firmware](#)
 - [Firmware Binaries](#)
 - [Firmware version history](#)
- [Applications](#)
 - [Reading Measurements from Flash](#)
 - [Reading Measurements in Connected Mode](#)
 - [Reading Measurements in Advertising Mode](#)
- [Technical specifications](#)
 - [Average power consumption](#)
 - [Bluetooth Advertising Formats](#)
 - [Bluetooth Connection Mode](#)
 - [Temperature or humidity trigger on GPIO PA5 \(label on the "reset" pin\)](#)

- [Reed Switch on GPIO PA6 \(label on the "P8" pin\)](#)
- [Interface for receiving and displaying data on the LCD.](#)
- [The USB-COM adapter writes the firmware in explorer. Web version.](#)
- [Related Work](#)
- [Resources & Links](#)
 - [CJMCU-2557 BQ25570](#)

🔗 Getting Started

You can conveniently flash, update and configure the bluetooth thermometers remotely using a bluetooth connection and a modern web browser.

🔗 Flashing or Updating the Firmware (OTA)

To flash or update the firmware, use a Google Chrome, Microsoft Edge or Opera Browser.

1. Go to the [Over-the-air Webupdater Page](#) [TelinkMiFlasher.html](#) *
 2. If using Linux: Ensure you enabled "experimental web platform features". Therefore copy the according link (i.e. `chrome://flags/#enable-experimental-web-platform-features` for Chrome), open a new browser tab, paste the copied URL. Now sten the *Experimental Web Platform features* flag to *Enabled*. Then **restart the browser**.
 3. In the Telink Flasher Page: Press `Connect` : The browser should open a popup with visible Bluetooth devices. Choose the according target device (i.e. `LYWSD03MMC`) to pair.
 4. After connection is established a *Do Activation* button appears. Press this button to start the decryption key process.
 5. Now you can press the *Custom Firmware ver x.x* button to directly flash the custom firmware. Alternatively you can choose a specific firmware binary (i.e. the original firmware) via the file chooser
 6. Press *Start Flashing*.
- Use [TelinkOTA](#) to flash old or alternative versions ([ATC1441](#)). This is a program for OTA projects with Telink SDK, no third-party (mijia) protections.

🔗 Configuration

After you have flashed the firmware, the device has changed it's bluetooth name to something like `ATC_F02AED` . Using the [TelinkMiFlasher.html](#) you have various configuration options.

General process:

1. Press *Connect*
2. Select the according device to connect/pair (i.e. `ATC_F02AED`)
3. Several configuration options appear
4. Choose *Send Config* to send changed value to the device. Or press *Set default* and then *Send config* to revert to the Defaults

Option	Description
<i>Temperature and Humidity offset</i>	Enter a value to correct the offset of the Temperature or Humidity displayed: For example <code>-1.4</code> will decrease the Temperature by 1.4°
<i>Smiley or Comfort</i>	Choose a static smiley or check the "Comfort" Radio box to change the smiley depending on current Temperature and Humidity.
<i>Comfort Parameters</i>	Defines the Lower (Lo) and Upper (Hi) Range for Temperature and Humidity interpreted as comfort zone. In the default configuration a smiley will appear.
<i>Advertising Type</i>	Type of supported Bluetooth Advertising Formats .
<i>Set time</i>	sends the current time to the device
<i>Comfort, Show batt, Clock</i>	Ticking the according boxes you can enable interval rotation between different LCD screens. See the example video below.

Example of LCD display modes

You can configure different LCD Display modes using *Comfort*, *Show batt* and *Clock* configuration checkboxes. The enabled LCD Display Modes will appear one-by-one in a loop.



Battery and clock display are enabled in the settings. The rest of the settings is kept default. The video contains 2 cycles.

1. Temperature and humidity
2. Temperature and % of battery
3. Temperature and humidity
4. Hours and minutes

🔗 Firmware

🔗 Firmware Binaries

You can directly update/flash the firmware without downloading the binaries below.

Custom Firmware Versions:

- [LYWSD03MMC Custom Firmware Version 3.5](#) (HW: B1.4, B1.6, B1.7, B1.9)
- [MHO-C401 Custom Firmware Version 3.5](#) (manufacturer Xiaomi, not [Azarton!](#))
- [CGG1 Custom Firmware Version 3.5](#) (CGG1-M)

Original Manufacturer Firmware Version

In case you want to go back to the original firmware, you can download them here:

- [Xiaomi LYWSD03MMC Original Firmware v1.0.0_0109](#) (HW: B1.4 only)

- [Xiaomi LYWSD03MMC Original Firmware v1.0.0_0130](#) (HW: B1.4..B1.9)
- [Xiaomi MHO-C401 Original Firmware v1.0.0_0010](#)
- [Azarton MHO_C401 Original Firmware v1.0.0_0016](#)
- [Qingping CGG1-M Original Firmware v1.0.1_0093](#)
- [Qingping CGPR1 Original Firmware v1.1.1_0228](#)

🔗 Firmware version history

Version	Changes
1.2	Bind, Set Pin-code, Support MHO-C401
1.3	Get/Set comfort parameters
1.4	Get/Set device name, Get/Set MAC
1.5	Add Standard Device Information Characteristics
1.6	Fix alternation of Advertising in mi mode
1.7	Authorization and encryption in permissions to access GAP ATT attributes, if pin code is enabled
1.8	Time display (instead of a blinking smile)
1.9	Recording measurements to flash memory (cyclic buffer for 19632 measurements)
2.0	Recording measurements with averaging to flash memory
2.1	Periodic display refresh for MHO-C401 'Erase mi-keys' option to return to original firmware
2.2	Added parameter "Encrypted Mi Beacon"
2.3	Added "Delete all records"
2.4	Added parameter "Clock time step"
2.5	Remove TRG/FLG errors, minor optimization
2.6	Expanding the ranges of threshold parameters (TRG)
2.7	Reducing power consumption of MHO-C401 (EDP update), adding version for CGG1

Version	Changes
2.8	Added saving bindkey to EEP if mi-keys are erased, reduced TX power to 0 dB for defaults
2.9	Added additional id flags to advertising packages
3.0	Added toggle support for advertising package structures for third-party software
3.1	Fix security attributes (for pincode)
3.2	Added new encrypted beacon formats , reed switch maintenance
3.3	Added autodetection for LYWSD03MMC hardware versions B1.6 and B1.9 . For CGG1-M and MHO-C401 - autodetection of SHTV3 or SHT4x sensors
3.4	Correct Hardware Version Setting for LYWSD03MMC B1.7
3.5	Correction of moisture readings for SHT4x sensors. Rounding off sensor values on display .

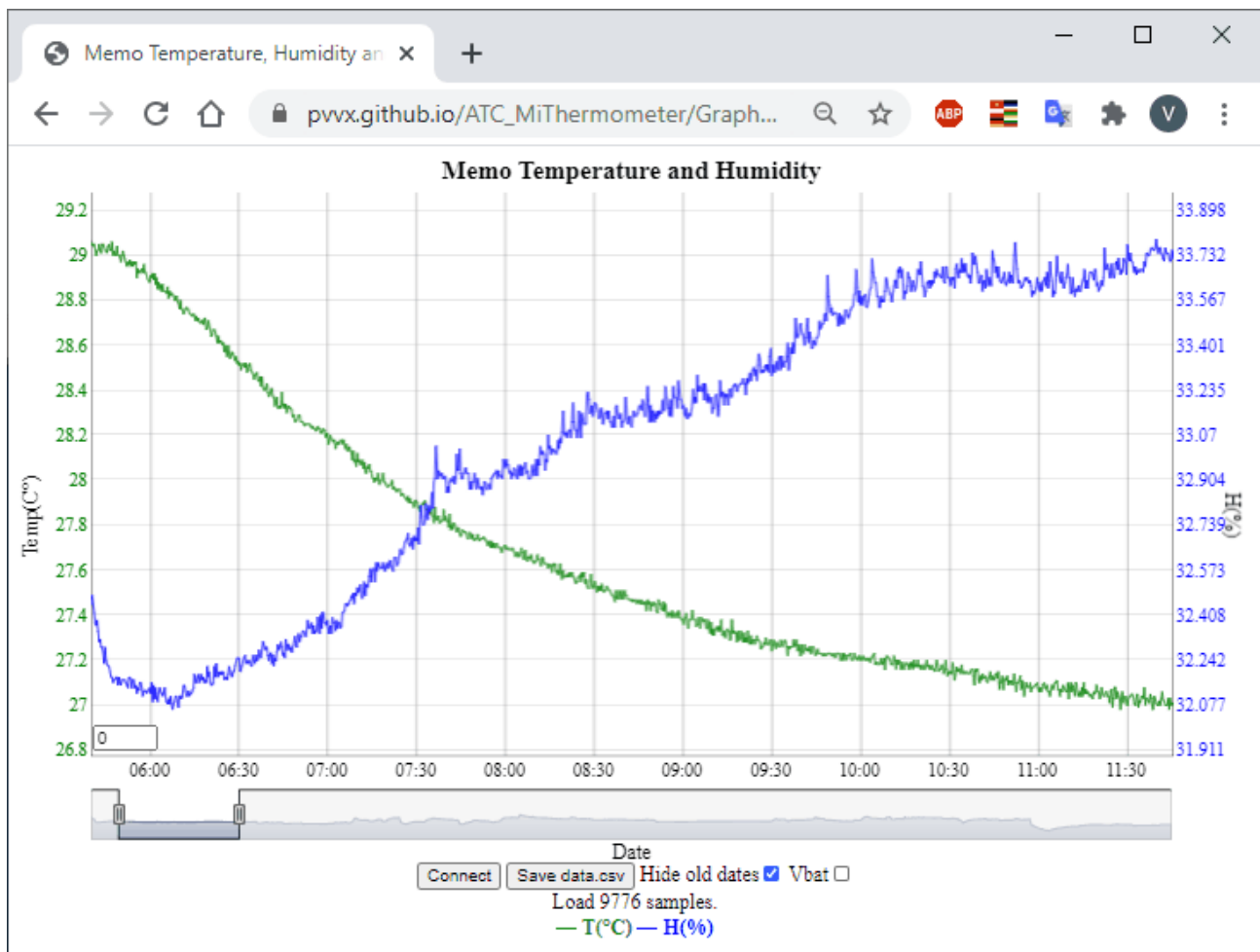
🔗 Applications

🔗 Reading Measurements from Flash

[GraphMemo.html](#)

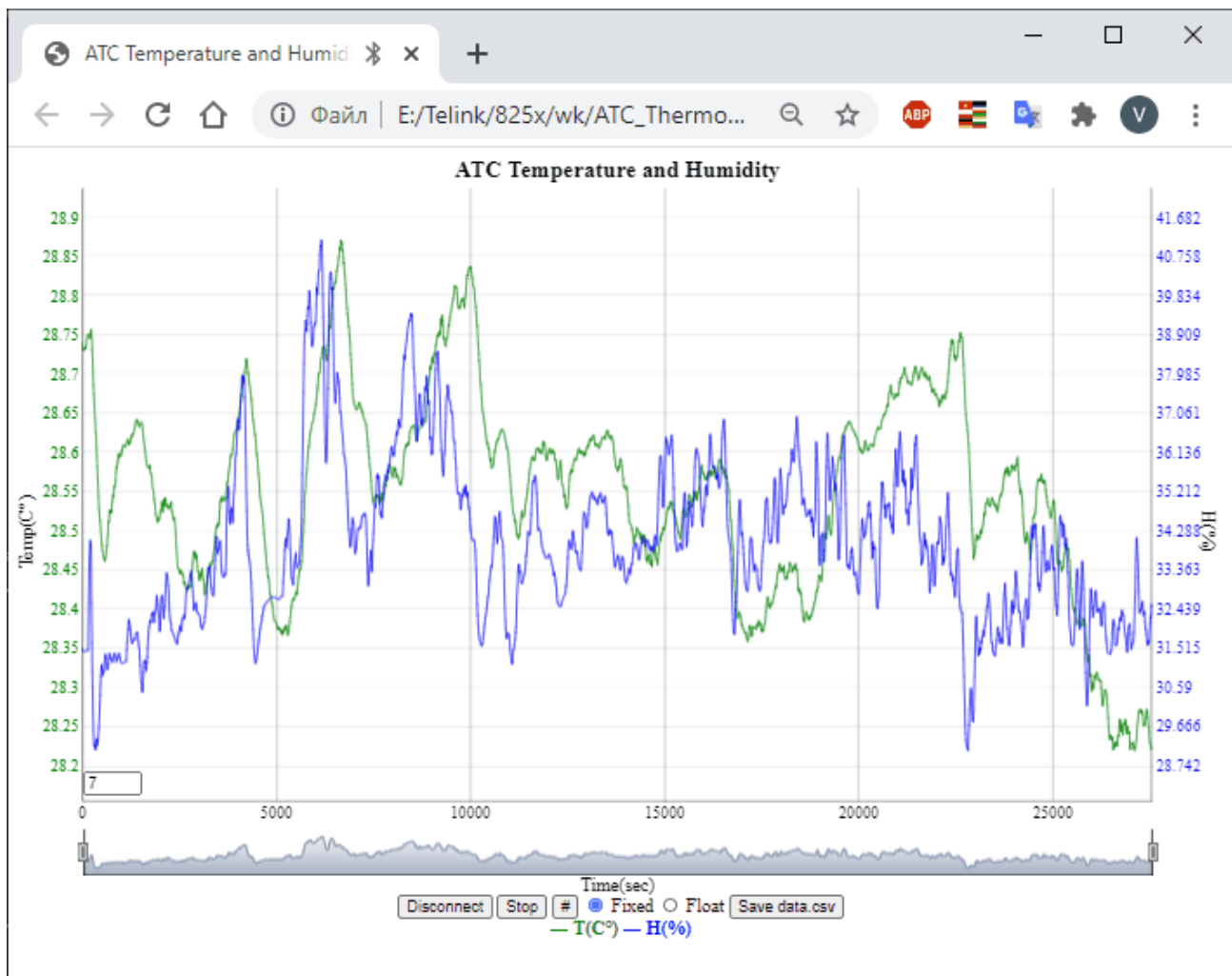
To record the measurement results, a cyclic buffer in the flash memory for 19632 measurements is used. The recording step interval is set in the interface. With the default settings, the recording step is 10 minutes, which gives a recording depth of 4 and a half months. During the step period, the sensor data and battery voltage are averaged, time stamped, and written to flash memory.

Setting the value to 0 disable logging to internal storage.



🔗 Reading Measurements in Connected Mode

[GraphAtc.html](#) [GraphAtc1.html](#) [GraphAtc2.html](#) [DevPoint.html](#)



🔗 Reading Measurements in Advertising Mode

This requires the *Experimental Web Platform Features* flag enabled in your browser. See [Flashing or Updating the Firmware \(OTA\)](#).

[Advertising.html](#)

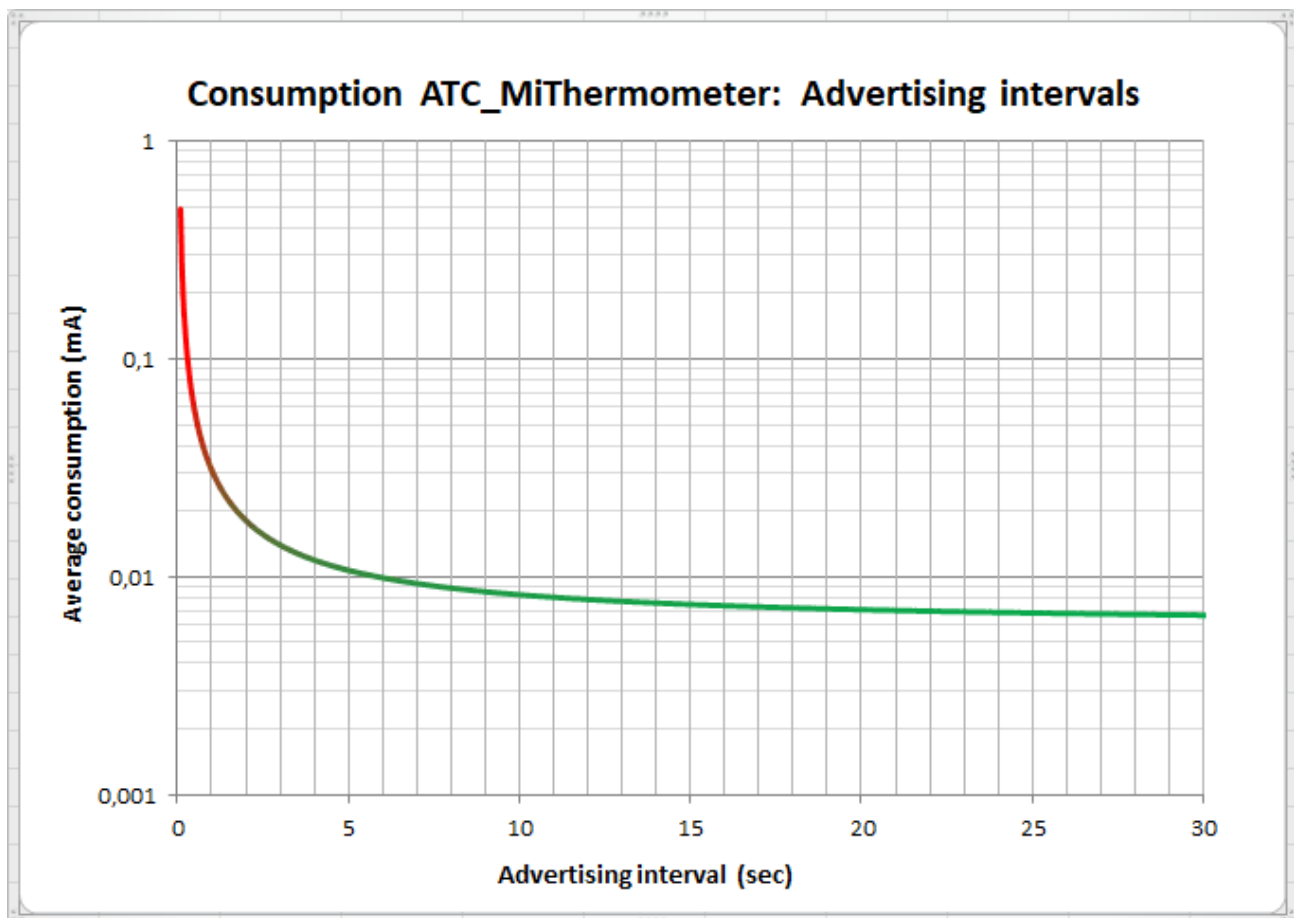
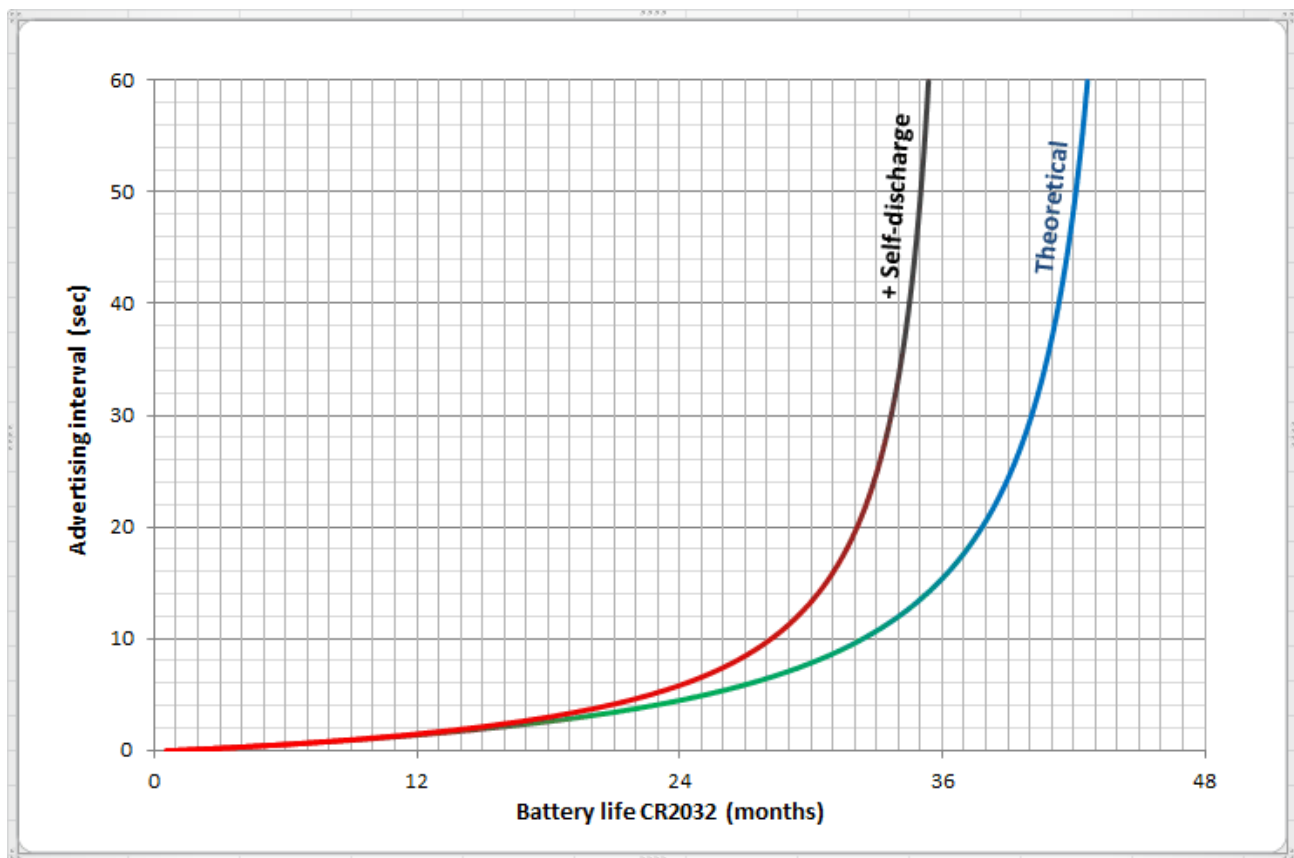
🔗 Technical specifications

🔗 Average power consumption

Xiaomi Mijia (LYWSD03MMC)

Using the default settings for advertising interval of 2.5 seconds and measurement interval of 10 seconds:

- Bluetooth Advertisement: 14..20 uA 3.3V ([CR2032 over 6 months](#))
- Bluetooth Connection: 14..25 uA 3.3V ([CR2032 over 6 months](#))



Xiaomi Miaomiaoce (MHO-C401)

Using the default settings for advertising interval of 2.5 seconds and measurement interval of 20 seconds:

- Bluetooth Advertisement: 12..30 uA 3.3V ([depends on the amount of temperature or humidity changes over time to display](#))
- Bluetooth Connection: 15..30 uA 3.3V (depends on the amount of temperature or humidity changes over time to display)

🔗 Bluetooth Advertising Formats

The Firmware can be configured to support one of four different Bluetooth advertisements data formats. Supports bindkey beacon encryption.

You can also configure to transferring everything in turn (round-robin)

🔗 atc1441 format:

UUID 0x181A - size 16: [atc1441 format](#)

🔗 Custom format (all data little-endian):

UUID 0x181A - size 19: Custom format (all data little-endian):

```
uint8_t    size;    // = 19
uint8_t    uid;     // = 0x16, 16-bit UUID
uint16_t   UUID;    // = 0x181A, GATT Service 0x181A Environmental Sensing
uint8_t    MAC[6];  // [0] - lo, .. [6] - hi digits
int16_t    temperature; // x 0.01 degree
uint16_t   humidity; // x 0.01 %
uint16_t   battery_mv; // mV
uint8_t    battery_level; // 0..100 %
uint8_t    counter; // measurement count
uint8_t    flags; // GPIO_TRG pin (marking "reset" on circuit board) flags:
                // bit0: Reed Switch, input
                // bit1: GPIO_TRG pin output value (pull Up/Down)
                // bit2: Output GPIO_TRG pin is controlled according to t
                // bit3: Temperature trigger event
                // bit4: Humidity trigger event
```

🔗 Encrypted beacon formats (uses bindkey):

- [Mijia standard format](#)
- [Energy efficient format \(low security level\)](#)

🔗 Xiaomi Battery Charge

3. UUID 0xFE95 - 0x0A: [Xiaomi](#) - battery charge level 0..100%, battery voltage in mV

🔗 Xiaomi Temperature & Humidity

4. UUID 0xFE95 - 0x0D: [Xiaomi](#) - temperature x0.1C, humidity x0.1%

🔗 Xiaomi Encrypted (bindkey enable)

5. UUID 0xFE95 - 0x0A, 0x04, 0x06 [Xiaomi](#) - battery charge level 0..100%, temperature x0.1C, humidity x0.1% (All data are averaged over a period of 16 measurements)

🔗 Bluetooth Connection Mode

- Primary Service - Environmental Sensing Service (0x181A):
 - Characteristic UUID [0x2A1F](#) - Notify temperature x0.1C
 - Characteristic UUID [0x2A6E](#) - Notify temperature x0.01C
 - Characteristic UUID [0x2A6F](#) - Notify about humidity x0.01%
- Primary Service - Battery Service (0x180F):
 - Characteristic UUID [0x2A19](#) - Notify the battery charge level 0..99%
- Primary Service (0x1F10):
 - Characteristic UUID [0x1F1F](#) - Notify, frame id 0x33 (configuring or making a request): temperature x0.01C, humidity x0.01%, battery charge level 0..100%, battery voltage in mV, GPIO-pin flags and triggers.

🔗 Temperature or humidity trigger on GPIO PA5 (label on the "reset" pin)

Xiaomi LYWSD03MMC has a PCB soldering test point. Available without disassembling the case and [marked "Reset"](#). Use as a trigger output with adjustable temperature and / or humidity hysteresis. Hysteresis and thresholds are set in [TelinkMiFlasher.html](#). The output status is displayed in the advertising package.

Management GPIO_PA5 (mark "reset"):

Temperature hysteresis: °, Humidity hysteresis: %

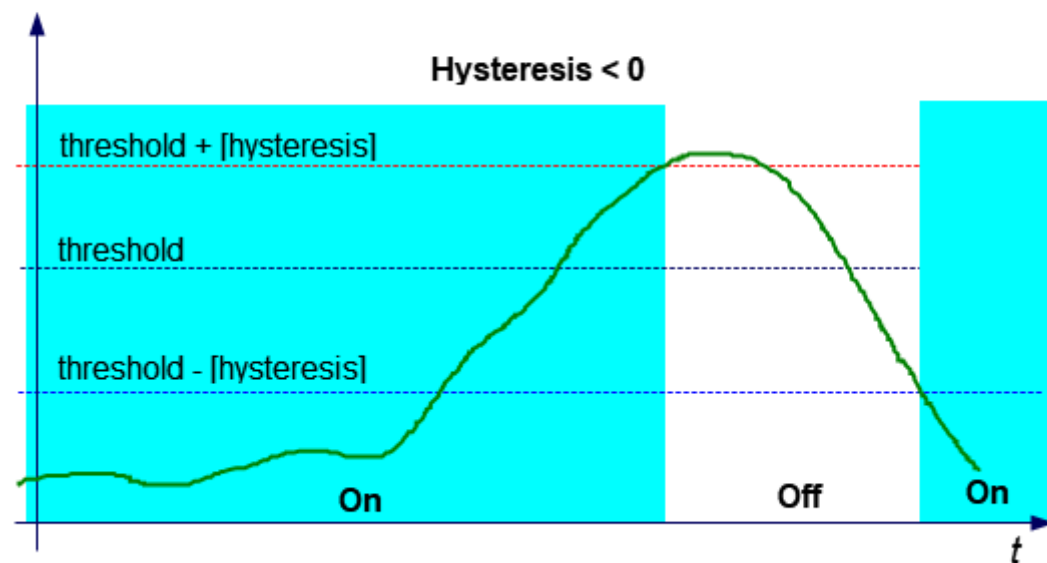
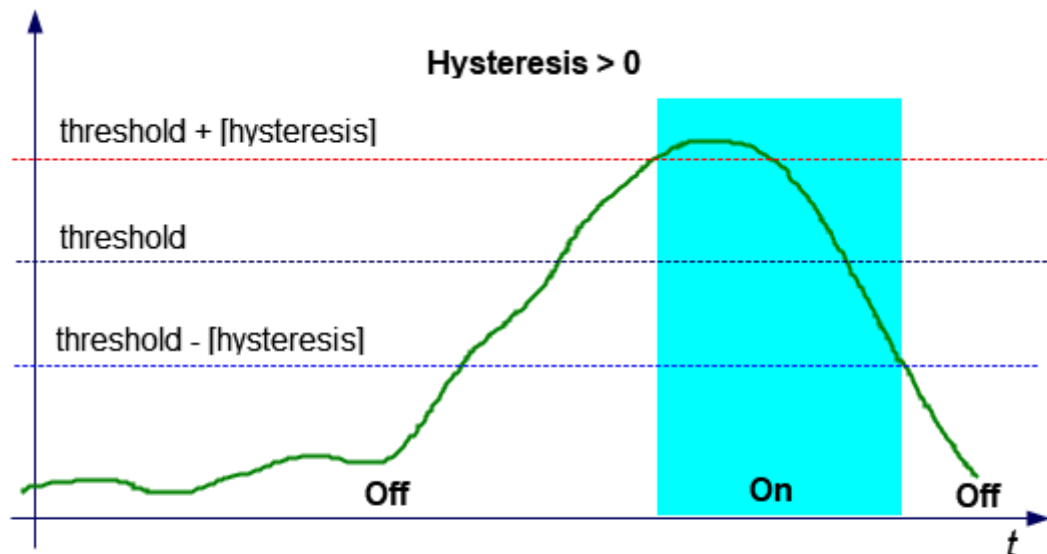
Temperature threshold: °, Humidity threshold: %

Get TRG

Set TRG

Set pin to "1"

Set pin to "0"

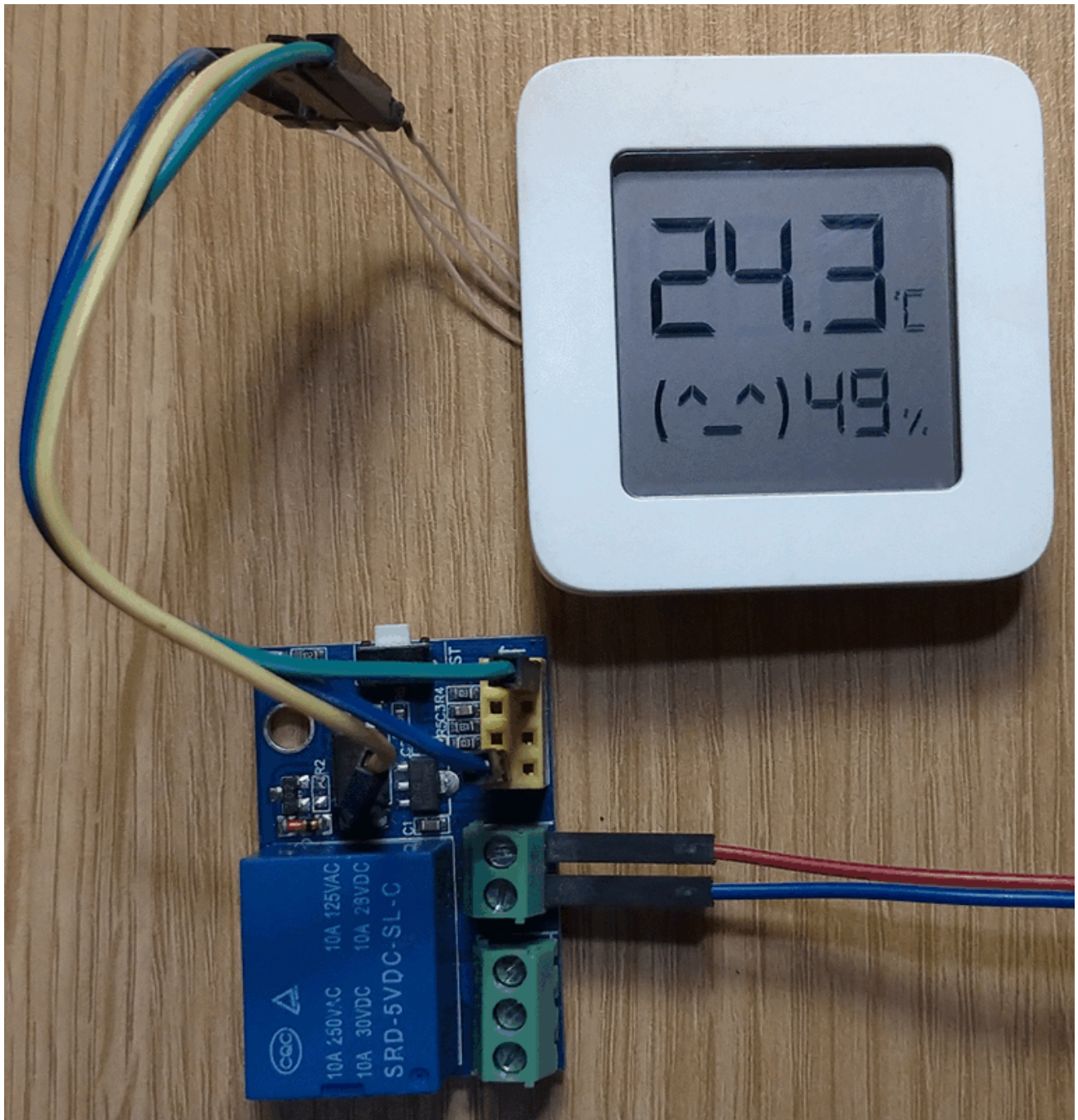


Hysteresis:

- =0 off,
- if less than zero - activation on decrease,
- if more than zero - activation on excess

Setting the pin to "1" or "0" works if both hysteresis are set to zero (TRG off).





Trigger and reed switch status receiver on JDY-10 module (\$1).

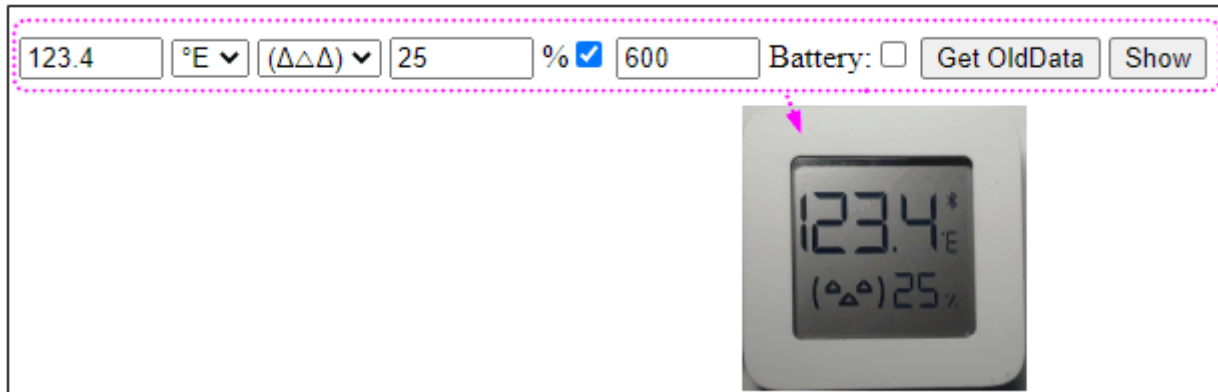
🔗 Reed Switch on GPIO PA6 (label on the "P8" pin)

It is possible to solder a reed switch on the LYWSD03MMC board to the pins marked "P8" and GND. The state of the reed switch is transferred to the advertising package. For CGG1 - GPIO_PD3, MHO_C401 - no free pins.

🔗 Interface for receiving and displaying data on the LCD.

- LCD shows:
- Big number: -99.5..1999.5

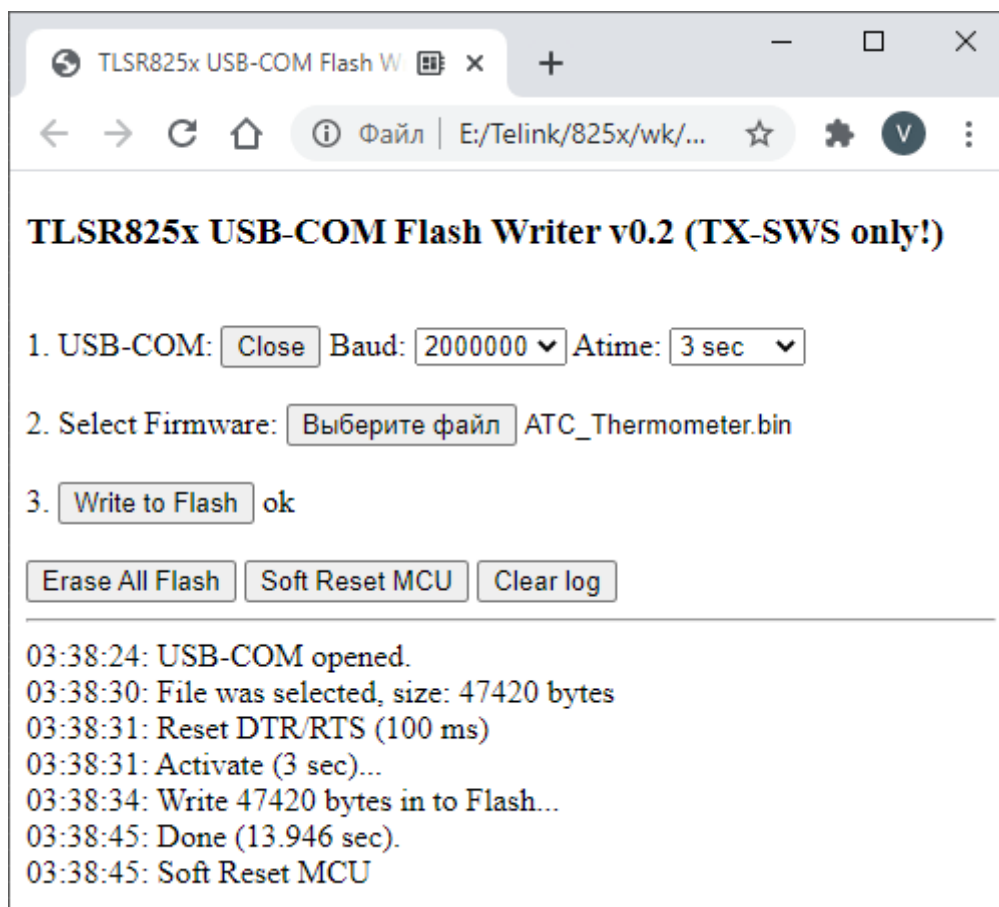
- Small number: -9..99
- Smiley, battery, degrees
- Setting the display time limit in sec



🔗 The USB-COM adapter writes the firmware in explorer. Web version.

- Connect only TX-SWS and GND wires.

[USBCOMFlashTx.html](#)



🔗 Chipset LYWSD03MMC HW:B1.4

- TLSR8251F512ET24 (TLSR8258 in 24-pin TQFN). SoC: TC32 32-bit MCU 48Mhz, 64 KiB SRAM, 512 KiB Flash (GD25LE40C), Bluetooth 5.0: Mesh, 6LoWPAN, Zigbee, RF4CE, HomeKit, Long Range, Operating temperature: -40°C to +85°C, Power supply: 1.8V to 3.6V.
- SHTV3 sensor. Measurement range: Temperature -40°C to +125°C, Humidity 0 to 100 %RH. Power supply: 1.8V to 3.6V
- IST3055NA0 LCD controller

LYWSD03MMC B1.4 BoardPinout

Chip pin	Mark	Name
1	P7	SPI_CK/I2S_BCK/7816_TRX(UART_TX)/PD<7>
2	Reset	DM/PA<5>
3	P8	DP(SWS)/PA<6>
4	P14	SWS/UART_RTS/PA<7>
12	P10	SDM_P1/SPI_DI/UART_RTS/lc_comp_ain<6>/sar_aio<6>/PB<6>
13	B1	SDM_N1/SPI_DO/UART_RX/lc_comp_ain<7>/sar_aio<7>/PB<7>
15	P13	PWM0/7816_TRX(UART_TX)/I2C_SDA/XC32K_O/PGA_P1/PC<2>
16	P15	PWM1/UART_RX/I2C_SCK/XC32K_I/PGA_N1/PC<3>
17	P9	PWM2/UART_CTS/PWM0_N/ sar_aio<8>/PC<4>
24		SPI_CN/I2S_LR/PWM3/PD<2>

🔗 Building the firmware

1. Go to wiki.telink-semi.cn and get the IDE for TLSR8 Chips.
2. Clone https://github.com/Ai-Thinker-Open/Telink_825X_SDK
3. Install the IDE and import the 'ATC_MiThermometer' project
4. Change 'Linked resource' and 'C/C++ Build/Build command'.
5. Compile the project

🔗 Related Work

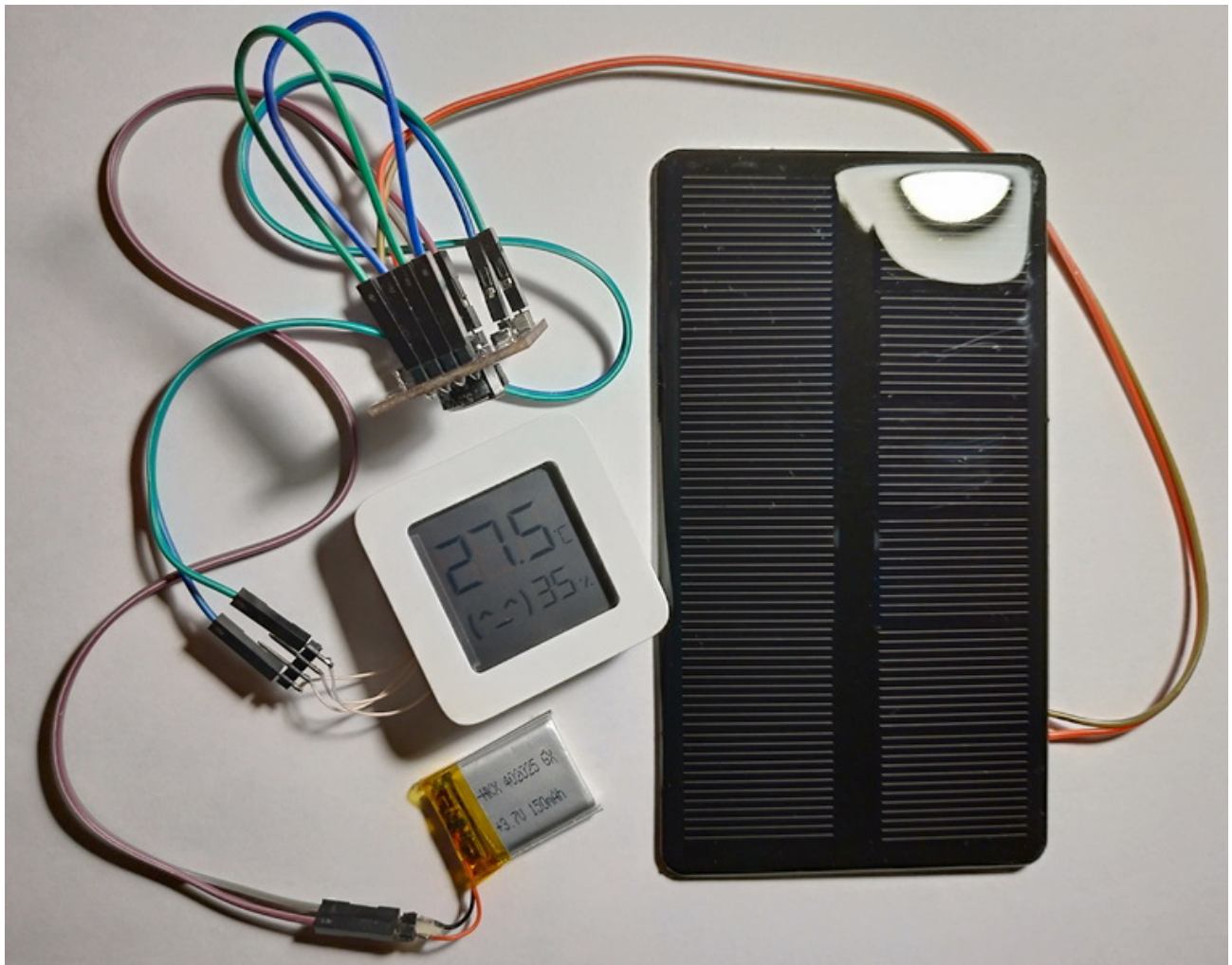
ATC_MiThermometer is based on the original work of [@atc1441](#), who developed the [initial custom firmware version and the web-based OTA flasher \(Source\)](#).

🔗 Resources & Links

- [HTML files](#)
- [Telink Linux Tool Chain 2020-06-26](#)
- [MHO-C401 Info](#)

- [CGG1-Mijia Info](#)
- [Build Telink EVK on E104-BT10-G/N module \(TLSR8269\)](#)
- [Alternative programmer TLSR SWire on module TB-04/03F or E104-BT10](#)
- [Additional information on the format Advertising for Xiaomi LYWSD03MMC](#)
- [Reading and recovering Mi-Home keys](#)
- [LYWSD03MMC Forum link \(russian\)](#)
- [MHO-C401 Forum link \(russian\)](#)

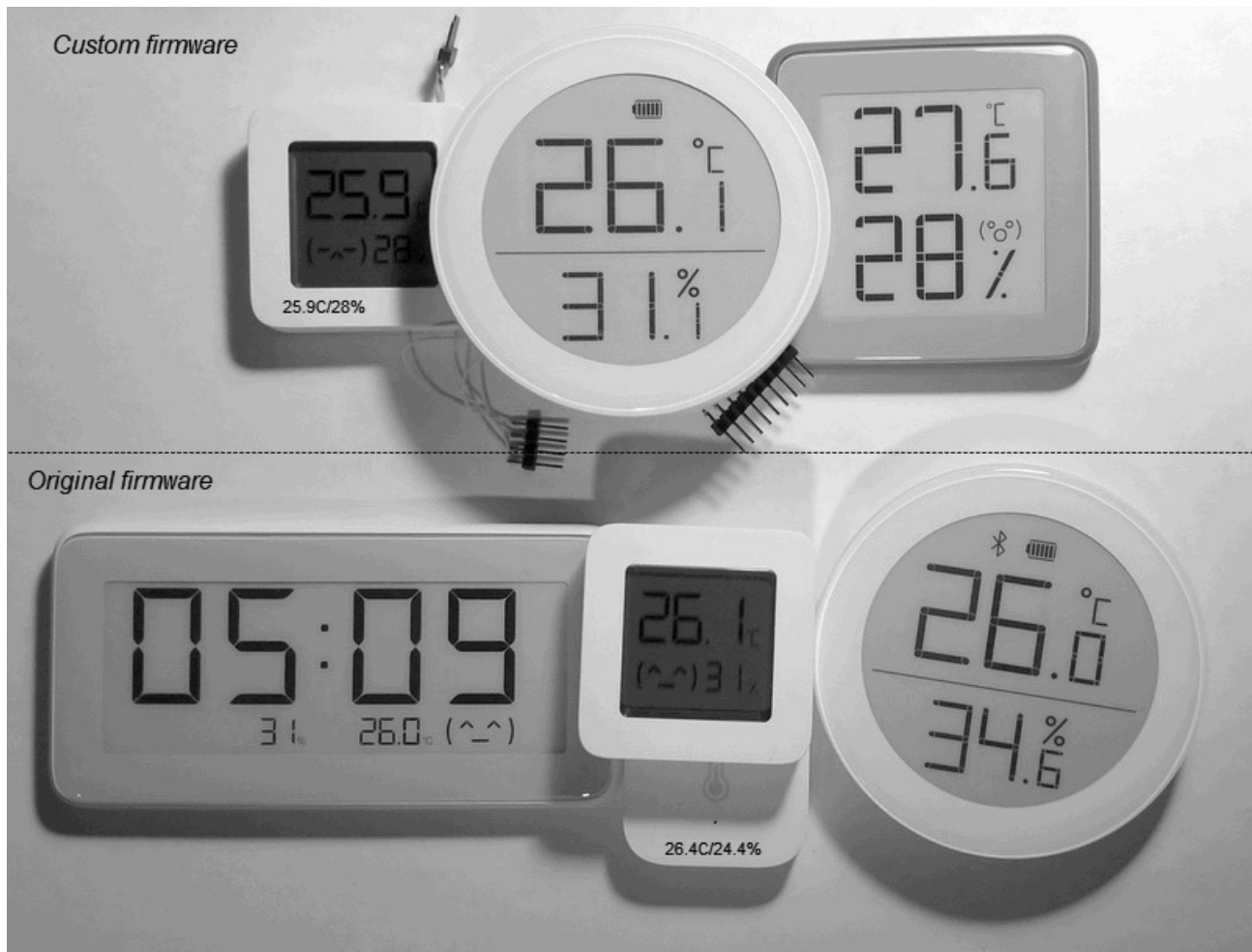
🔗 **CJMCU-2557 BQ25570**



🔗 **Control function ID when connected**

🔗 **Primary Service UUID 0x1F10, Characteristic UUID 0x1F1F**

ID	Command
0x01	Get/Set device name
0x10	Get/Set MAC
0x11	Get/Set Mi key: DevNameID
0x12	Get/Set Mi keys: Token & Bind
0x15	Get all Mi keys
0x16	Restore prev mi token & bindkeys
0x17	Delete all Mi keys
0x18	Get/set binkey in EEP
0x20	Get/Set comfort parameters
0x22	Get/Set show LCD ext.data
0x23	Get/Set Time
0x24	Get/set adjust time clock delta
0x33	Start/Stop notify measures in connection mode
0x35	Read memory measures
0x36	Clear memory measures
0x44	Get/Set TRG config
0x45	Set TRG output pin
0x4A	Get/Set TRG data (not save to Flash)
0x55	Get/Set device config
0x56	Set default device config
0x5A	Get/Set device config (not save to Flash)
0x60	Get/Set LCD buffer
0x61	Start/Stop notify LCD buffer
0x70	Set PinCode
0x71	Request Mtu Size Exchange



Releases 9

 **v3.4** Latest
on Nov 27, 2021

[+ 8 releases](#)

Packages

No packages published

Languages

