openQCM Wi2 Firmware Programming Guide

Version 1.2.T4 for Teensy 4.0 Microcontroller

Target Hardware: Teensy 4.0 Microcontroller
Firmware Version: 1.2.T4
Compatible Software: openQCM Wi2 Python Software v2.1+
Date: July 2025
Author: openQCM Team

Overview

This guide provides step-by-step instructions for programming the openQCM Wi2 device firmware version 1.2.T4, specifically designed for Teensy 4.0 microcontroller compatibility with the openQCM Wi2 Python Software.

When Firmware Update is Required

⚠️ CRITICAL COMPATIBILITY CHECK

Firmware update is **required** for devices with:

- **Serial number < 1840**
- **Firmware version < 1.2.T4**

Devices with serial number ≥ 1840 already have compatible firmware and do not require updates.

Firmware 1.2.T4 Key Features

- **Simplified Data Format**: Streamlined frequency,temperature output format
- **Enhanced Serial Communication**: Improved reliability at 115200 baud
- **Device Information Commands**: Serial number and firmware version queries
- **Python Software Compatibility**: Optimized for openQCM Wi2 Python Software v2.1
- **Teensy 4.0 Optimization**: Full compatibility with Teensy 4.0 microcontroller

---

Prerequisites

Hardware Requirements

- openQCM Wi2 device with Teensy 4.0 microcontroller
- USB cable (Type-A to Micro-USB)
- Windows PC with available USB port

Software Requirements

- **TyUploader.exe** (included in firmware package)
- **Firmware file**: `openQCM_Wi2_FW_1_2_T4.ino.hex`
- Windows 10 or later
- USB drivers (automatically installed by Windows)

Download Required Files

1. **Download firmware package**: openQCM_Wi2_firmware_tool_FW_1.2.T4.zip
2. **Extract contents** to a dedicated folder
3. **Verify package contents**:
    - `TyUploader.exe`
    - `openQCM_Wi2_FW_1_2_T4.ino.hex`
    - Documentation files

Firmware Programming Instructions

Method 1: Using TyUploader Application (Recommended)

Step 1: Prepare Hardware

1. **Connect device**: Use USB cable to connect openQCM Wi2 to computer
2. **Verify connection**: Check Windows Device Manager for Teensy device
3. **Close other applications**: Ensure no other software is accessing the device

Step 2: Launch Programming Tool

1. **Navigate** to extracted firmware folder
2. **Run TyUploader.exe** (may require administrator privileges)
3. **Confirm execution** if prompted by Windows security

Step 3: Program Firmware

1. **Click "Upload"** button in TyUploader interface
2. **Select firmware file** when prompted:
    - Navigate to firmware folder
    - Select `openQCM_Wi2_FW_1_2_T4.ino.hex`
    - Click "Open"
3. **Wait for upload completion**:
    - Progress indicator shows upload status
    - Success message appears when complete
    - Process typically takes 30-60 seconds

Step 4: Verify Programming

1. **Disconnect** USB cable from device
2. **Reconnect** USB cable
3. **Test with openQCM Wi2 Software**:
    - Launch openQCM Wi2 Python Software
    - Connect to device
    - Verify firmware version shows "1.2.T4"

Method 2: Using Arduino IDE (Advanced Users)

Prerequisites

- Arduino IDE 1.8.19 or later
- Teensyduino add-on installed
- Experience with Arduino IDE

Programming Steps

1. **Open Arduino IDE**
2. **Install Teensyduino** (if not already installed)
3. **Open firmware source**: `openQCM_Wi2_FW_1_2_T4.ino`
4. **Select board**: Tools → Board → Teensy 4.0
5. **Select port**: Tools → Port → (select appropriate port)
6. **Upload**: Click upload button or Ctrl+U
7. **Verify upload success**

Troubleshooting

Common Issues and Solutions

Device Not Detected

Symptoms: TyUploader doesn't recognize device
Solutions:

- Verify USB cable connection
- Try different USB port
- Check Windows Device Manager for Teensy device
- Restart computer and retry

Upload Fails

Symptoms: Programming process fails or times out
Solutions:

- Ensure device is properly connected
- Close other applications using USB/serial ports
- Try different USB cable
- Run TyUploader as administrator
- Restart device by disconnecting/reconnecting USB

Verification Errors

Symptoms: Firmware appears to upload but device doesn't respond correctly
Solutions:

- Repeat programming process
- Verify correct firmware file selected
- Check for hardware issues
- Contact support if problems persist

Device Recovery

If device becomes unresponsive after programming attempt:

1. **Disconnect** USB cable
2. **Wait 10 seconds**
3. **Reconnect** USB cable
4. **Retry programming** using TyUploader
5. **Contact support** if device remains unresponsive

Post-Programming Verification

Verification Checklist

1. **Physical Connection**
   - [ ] Device powers on (LED indicators active)
   - [ ] USB connection stable
   - [ ] Device recognized by Windows
2. **Software Compatibility**
   - [ ] openQCM Wi2 Python Software detects device
   - [ ] Connection establishes successfully
   - [ ] Firmware version shows "1.2.T4"
   - [ ] Serial number displays correctly
3. **Functional Testing**
   - [ ] Data acquisition starts properly
   - [ ] Frequency and temperature readings appear
   - [ ] Data logging functions normally
   - [ ] All software features operate correctly

Expected Behavior

After successful firmware programming:

- Device should power up with LED indicators
- Software should detect device immediately
- Connection should establish within 2-3 seconds
- Firmware version should display as "1.2.T4"
- Data acquisition should start within 5 seconds

Technical Specifications

Firmware 1.2.T4 Specifications

Parameter	Specification
Target Hardware	Teensy 4.0
Communication	USB Serial 115200 baud
Sampling Rate	1 Hz (1 sample/second)
Data Format	frequency,temperature
Commands	Serial number (1), Firmware version (2)
Gate Time	1,000,000 microseconds
Temperature Sensor	MCP9808 I2C

Communication Protocol

Data Output Format

```
frequency,temperature
```

Example: `10000000,23.45`

Command Interface

- **Command 1**: Returns device serial number
- **Command 2**: Returns firmware version
- **Response Format**: `SERIALNUMBER<value>` or `FIRMWARE<version>`

Hardware Compatibility

Component	Specification
Microcontroller	Teensy 4.0 (ARM Cortex-M7)
Supply Voltage	3.3V
Crystal Oscillator	3.3V supply
Temperature Sensor	MCP9808
Communication	USB 2.0

Support and Contact

Technical Support

Website: https://openqcm.com/
Email: info@openqcm.com

Support Information

When contacting support, provide:

- Device serial number
- Current firmware version (if accessible)
- Programming method attempted
- Error messages or symptoms

- Computer specifications

Warranty and Limitations

Important: Firmware programming carries risk of device damage if performed incorrectly. Follow instructions carefully and contact support if uncertain about any step.

Disclaimer: Programming firmware may void warranty. Proceed only if comfortable with technical procedures or as directed by openQCM support.

Revision History

Version	Date	Changes
1.2.T4	July 2025	Initial release for Python Software compatibility
-	-	Simplified data format implementation
-	-	Enhanced serial communication reliability
-	-	Added device information commands
