Table of Contents

Introduction 3
  Overview 3
  Product Description 3
  Additional Resources 4

Technical Specifications 5
  Block Diagram 5
  Pinout 6
    Table 2.2.1 Shield Default Pinout 6

Getting Started 7
  Operating Modes 7
    UART Configuration 7
    Sensor Connections 7
  Required Hardware 8
  Mounting the Shield 8
  Skywire Placement 10
  Programming the chipKIT board 11
  Viewing the Device Data 12
  Troubleshooting 12
1. Introduction

1.1 Overview

This document is the User Manual for the Skywire® Sensor Shield when used with chipKIT WF32 development kit. Throughout the document Skywire Sensor Shield will often be referred to as 'the shield', although it may be referred to by its full name.

The NimbeLink Skywire modem is available with bundled data plans from leading cellular carriers. The Skywire cellular modem and antennas are sold separately. Make sure you check NimbeLink's Skywire Sensor Shield product page for the most up to date information.

1.2 Product Description

Connect the Microchip WF32 development platform to the Internet of Things (IoT) quickly and easily with NimbeLink's Skywire Sensor Shield. Whether you are a hobbyist or a developer preparing a product for launch, NimbeLink's shield and your choice of NimbeLink Skywire plug-in cellular modems will provide the cellular connectivity you need. With Cellular connectivity, several included sensors, and example code you can have your proof of concept up and running in minutes with a clear path to production.

- Skywire socket that supports 2G, 3G, and 4G modems
- NimbeLink’s certified modems and bundled data plans can have you connected to the network faster than any other cellular solution
- Four sensors to demonstrate your IoT solution (Temperature, Acceleration, Pressure, and Humidity)
- Example code to upload sensor data to Verizon Thingspace.io dashboards
### 1.3 Orderable Parts

<table>
<thead>
<tr>
<th>Orderable Device</th>
<th>Description</th>
<th>Manufacturer</th>
<th>Carrier</th>
<th>Network Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>NL-AB-ST-NCL</td>
<td>Skywire Sensor Shield (includes antenna and power supply)</td>
<td>NimbeLink</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NL-SW-1XRTT-V</td>
<td>Skywire 1xRTT Cellular Modem without data plan. 2G Modem.</td>
<td>NimbeLink</td>
<td>Verizon</td>
<td>2G CDMA 1xRTT</td>
</tr>
<tr>
<td>NL-SW-EVDO-V</td>
<td>Skywire EVDO Cellular Modem without data plan. 3G Modem.</td>
<td>NimbeLink</td>
<td>Verizon</td>
<td>3G CDMA EVDO</td>
</tr>
<tr>
<td>TG.30.8113</td>
<td>Cellular Antenna</td>
<td>Taoglas</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TDGL021</td>
<td>chipKIT WF32 IoT development kit</td>
<td>Microchip</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### 1.4 Additional Resources

- NimbeLink's [Skywire Sensor Shield Product Page](#)
- NimbeLink's [Skywire Sensor Shield Schematic](#)
- NimbeLink's [Skywire 2G 1xRTT Product Page](#)
- NimbeLink's [Skywire 3G GSM/HSPA+ Product Page](#)
- NimbeLink's [Skywire 3G CDMA/EVDO Product Page](#)
- Microchip [chipKIT WF32 Platform Page](#)
2. Technical Specifications

2.1 Block Diagram
## 2.2 Pinout

Table 2.2.1 Shield Default Pinout

<table>
<thead>
<tr>
<th>Pin Name</th>
<th>Header</th>
<th>Shield Connection</th>
</tr>
</thead>
<tbody>
<tr>
<td>IO0</td>
<td>IOL</td>
<td>Skywire DOUT</td>
</tr>
<tr>
<td>IO1</td>
<td>IOL</td>
<td>Skywire DIN</td>
</tr>
<tr>
<td>IO2</td>
<td>IOL</td>
<td>No Connect</td>
</tr>
<tr>
<td>IO3</td>
<td>IOL</td>
<td>Button 1</td>
</tr>
<tr>
<td>IO4</td>
<td>IOL</td>
<td>LM75 INT</td>
</tr>
<tr>
<td>IO5</td>
<td>IOL</td>
<td>LIS3DH INT1</td>
</tr>
<tr>
<td>IO6</td>
<td>IOL</td>
<td>HTS221 DRDY</td>
</tr>
<tr>
<td>IO7</td>
<td>IOL</td>
<td>LPS331 INT1</td>
</tr>
<tr>
<td>IO8</td>
<td>IOH</td>
<td>No Connect</td>
</tr>
<tr>
<td>IO9</td>
<td>IOH</td>
<td>Skywire Reset</td>
</tr>
<tr>
<td>IO10</td>
<td>IOH</td>
<td>Skywire DTR</td>
</tr>
<tr>
<td>IO11</td>
<td>IOH</td>
<td>Skywire RTS</td>
</tr>
<tr>
<td>IO12</td>
<td>IOH</td>
<td>Skywire ON/OFF</td>
</tr>
<tr>
<td>IO13</td>
<td>IOH</td>
<td>Skywire CTS</td>
</tr>
<tr>
<td>GND</td>
<td>IOH</td>
<td>GND</td>
</tr>
<tr>
<td>AREF</td>
<td>IOH</td>
<td>No Connect</td>
</tr>
<tr>
<td>SDA</td>
<td>IOH</td>
<td>I2C_SDA</td>
</tr>
<tr>
<td>SCL</td>
<td>IOH</td>
<td>I2C_SCL</td>
</tr>
<tr>
<td>All Pins</td>
<td>AD</td>
<td>LEDs/Analog Sensors</td>
</tr>
<tr>
<td>VIN</td>
<td>POWER</td>
<td>12 V Input</td>
</tr>
<tr>
<td>GND</td>
<td>POWER</td>
<td>GND</td>
</tr>
<tr>
<td>5V</td>
<td>POWER</td>
<td>5V Input (unused)</td>
</tr>
<tr>
<td>3.3V</td>
<td>POWER</td>
<td>3.3V Input</td>
</tr>
<tr>
<td>RESET</td>
<td>POWER</td>
<td>No Connect</td>
</tr>
<tr>
<td>IOREF</td>
<td>POWER</td>
<td>Skywire Vref</td>
</tr>
</tbody>
</table>
3. Getting Started

3.1 Operating Modes

NimbeLink’s Skywire Sensor Shield is designed to be compatible with the Microchip chipKIT WF32 board. The shield also provides headers to plug in additional shields to further the platform’s functionality. The user must ensure the pins on expansion shields do not interfere with the pins of the Skywire Sensor Shield.

3.1.1 UART Configuration

By default the UART connection between the Skywire and the chipKIT board connects to UART1. The default debug interface also uses UART1 so it is important to avoid using this line for both communications. It is recommended that UART4 (Pins 39/40 on the chipKIT WF32 Board) be used for the debug UART in order to avoid interference. This can be done with an external Serial to USB cable, for example.

The Skywire defaults to an 115200 BAUD rate, 8 bit data, no parity, and 1 stop bit. The Data Terminal Ready (DTR) and Ready To Send (RTS) lines must be driven low from the WF32 to the modem.

3.1.2 Sensor Connections

The shield has four sensors connected to Analog/IO pins that can be polled by the chipKIT board.

The shield also has four sensors connected to the I2C interface coming from the chipKIT board. The shield provides 4.75K pullup resistors on the I2C interface. On the chipKIT board used in this startup guide, the I2C interface is on pins 45 (SCL) and 46 (SDA).

See the table below for details on each sensor.

<table>
<thead>
<tr>
<th>Sensor</th>
<th>Function</th>
<th>Pin</th>
</tr>
</thead>
<tbody>
<tr>
<td>PhotoTrans</td>
<td>Returns the amount of surrounding light, value = 0-1023</td>
<td>AD0</td>
</tr>
<tr>
<td>Button 1</td>
<td>A digital read of the pin returns a 1 for closed, a 0 for open</td>
<td>IO3</td>
</tr>
<tr>
<td>Button 2</td>
<td>A digital read of the pin returns a 1 for closed, a 0 for open</td>
<td>AD4</td>
</tr>
<tr>
<td>Potentiometer</td>
<td>Adjustable voltage divider changes output by rotating, value = 0-1023</td>
<td>AD5</td>
</tr>
<tr>
<td>HTS221</td>
<td>Humidity 0-100% RH</td>
<td>I2C</td>
</tr>
<tr>
<td>LIS3DH</td>
<td>3-Axis Accelerometer 2g/4g/8g/16g</td>
<td>I2C</td>
</tr>
</tbody>
</table>
3.2 Required Hardware

This guide assumes the user has access to the following hardware:

- Skywire chipKIT Cellular Sensor Shield
- chipKIT WF32 Demo Board (or equivalent)
- 12 V power supply
- USB A to mini-B connector (for programming the chipKIT board)
- Cellular antenna
- Optional: USB-to-Serial cable (for debugging)

3.3 Mounting the Shield

Before mounting the shield, verify that the following jumpers in particular are in the correct positions as shown in the figure below:

![Correct Jumper Positions](image)

Figure 3-1: correct jumper positions for chipKIT board.

J15: jumper should be in the ‘EXT’ position. This allows the shield to be powered off of the chipKIT board’s external 12 V supply port instead of the USB port.

JP4/JP5: jumpers should be in the ‘AN’ positions to disconnect the I2C bus from the analog sensor ports.
The Shield is designed to interface with the Arduino compatible headers on the chipKIT development board. To mount the shield, orient the pins correctly over the Arduino headers and press down on the shield until the shield is seated fully. There should be no unconnected pins below the shield.

Figure 3-2: chipKIT board and Skywire Shield proper orientation

Figure 3-3: Skywire Shield mounted on chipKIT board
3.4 Skywire Placement

The Skywire Cellular modem is designed to be placed as shown below.

To mount your Skywire Cellular modem follow these steps:

1. Gather the following:
   a. Skywire Sensor Shield
   b. Skywire Cellular Modem
   c. U.FL extractor tool (Always use a U.FL extractor tool when placing or removing U.FL cables on the Skywire modem to avoid damaging the U.FL connectors).

2. Line up the Skywire cellular U.FL connector(s) with the circles inside the shield’s Skywire socket footprint. Depending on the type of Skywire Modem there may be one or two U.FL connections.

3. Carefully seat your Skywire into the shield’s Skywire socket (J6). Take care to ensure that the pins are correctly aligned. Failure to properly align the pins may damage your Skywire!

4. Attach the U.FL cable to the U.FL connector marked X1 on the Skywire modem.
3.5 Programming the chipKIT board

To program the chipKIT board while the Skywire Shield is mounted, remove the jumpers from J3 and J4 on the shield to disconnect the Skywire modem from the serial communication lines used for programming. Then, download the chipKIT board software, MPIDE, from the chipKIT website:

http://chipkit.net/mpide

and follow the steps below to program the chipKIT board:

- Follow the MPIDE installation guide referenced on the MPIDE download page (Windows, Mac, Linux)
- Extract the zip file to the MPIDE sketchbook folder, located by default in your “My Documents” directory
- Within MPIDE, open the extracted pde file: ‘File -> Open -> Skywire_chipKIT_Dweet -> Skywire_chipKIT_Dweet.pde’
- Remove jumpers from J3 and J4 on the shield
- Power the chipKIT board and shield using the external 12 V power jack
- Connect the chipKIT board to your computer via USB
- Within MPIDE: Select ‘Tools -> Serial Port -> COM#’ (where # corresponds with the COM port assigned to your chipKIT board)
- Select ‘Tools -> Board -> chipKIT board -> chipKIT WF32’
- Select the ‘Upload’ button
- Wait for the upload to finish. The LD1 and LD2 LEDs should be flashing during this process

Before running:

- Disconnect USB cable from the chipKIT board
- Unplug the power cord from the chipKIT board and shield
- Reconnect jumpers on J3 and J4
3.6 Viewing the Device Data

Once your chipKIT board is programmed and has a mounted shield, the following steps can be used to view the data being posted to the cloud by your device:

- Make an account at https://freeboard.io/ if you do not already have one
- Go to https://freeboard.io/board/6Xxpac and make a copy of the dashboard by clicking on the "Clone" button in the lower right corner
- Open the cloned dashboard and click on "skywire" under "Datasources"
- Replace the "Thing Name" field with your modem's MEID or IMEI and select "Save"
- Power on your chipKIT board and shield and wait for the green LED to start flashing on the shield

At this point, your dashboard widgets should begin updating in real-time as your device uploads its sensor data to dweet.io.

3.7 Troubleshooting

If you are unable to view your device's data on freeboard.io after going through the steps in the previous sections, try the following troubleshooting options:
• Make sure the shield has jumpers on J3 and J4 to allow the chipKIT board to send serial commands to the modem.

• If the shield's green LED is flashing, verify that the data is being uploaded to dweet.io by going to https://dweet.io/get/latest/dweet/for/<thingname>, where <thingname> is replaced by the MEID or IMEI of your modem. If your device’s data is displayed there, check that the correct thing name is entered under the "Data Sources" section of your freeboard.io dashboard.

• **The following requires a USB-to-UART Serial cable:** If the green LED is not flashing, power off the chipKIT board and remove the shield. Then, place a wire in pin 40 on the chipKIT board and re-mount the shield. Connect the "GND" wire of the USB-to-Serial cable to J4 on the chipKIT board (ground) and the "RX" pin to the wire in pin 40 of the chipKIT board. Power on the chipKIT board and shield. Using a terminal program such as Tera Term or PuTTY, open a serial connection with 115200 BAUD, 8-bit, no parity, and 1 stop bit to view the debugging trace statements from the chipKIT board.

• **The following requires a Skywire Development Kit:** Follow the instructions at


  to verify that your modem is activated and is able to connect to the cellular network.