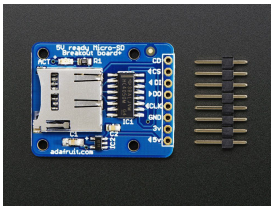
**Evaporometer PCB Pin Configuration**

| **Feather 32u4** | P13 | P12 | P11 | P10 | P7 | P6 | P5 | SCL | SDA |
|---|---|---|---|---|---|---|---|---|---|
| **Connection** | HX711_ SCK (load cell) | HX711 _D0 (load cell) | DS3231_S WQ/INT (RTC) | Tare_ button | [reserved for battery input read voltage] | [open] | [open] | All SCL signals | All SDA signals |

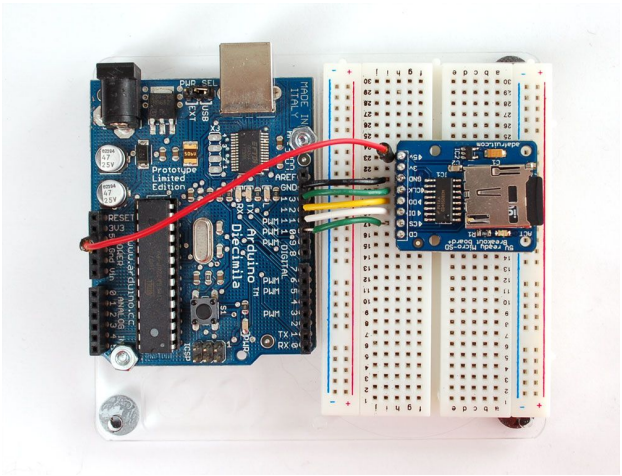
---

**MicroSD card breakout board + (External data storage)**



- Takes both 3V and 5V power source, has an onboard voltage regulator to shift 5V down to 3V.
- Keep wires short, avoid using resistor dividers for the 3.3V logic lines.
- Interface: SPI Mode “lower speed”, *use this*, requires four pins
- Format using FAT16 or FAT32 filesystem
- Require a lot of data transfer, best performance when connected up to the hardware SPI pins on a microcontroller.
- Use the Arduino IDE's SD library which supports FAT and FAT32 SD cards

**CD** (Card Detect Pin) = Shorts to ground when a card is inserted. Connect a pull up resistor (10K or so) and wire this to another pin if you want to detect when a card is inserted.



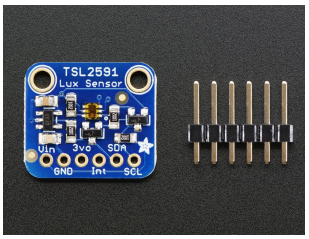
(to Feather 32u4, **NOT** shown)
**CS** to #9
**DI** to MOSI
**DO** to MISO
**CLK** to pin SCK
**GND** to GND
**5V** to 3V

**the MicroSD is typically only used for the Evaporometer Hub, but included in case this is used on the Evaporometer itself**
**the Evaporometer PCB does not currently include connections for the SD card breakout, so would need to be manually connected to the Feather 32u4**

*Resources:* https://www.adafruit.com/product/254

---

**Adafruit TSL2591 High Dynamic Range Digital Light Sensor**



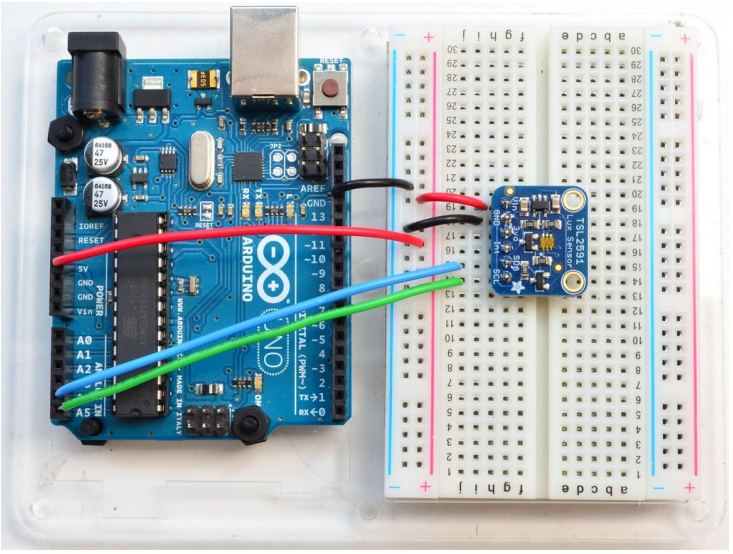

as

- Digital light sensor (use four outside pins)
- 12C sensor - uses the two I2C data/clock wires available on most microcontrollers, can share those pins with other sensors long as they don't have an address collision. The I2C address is 0x29.
- Can use with any microcontroller

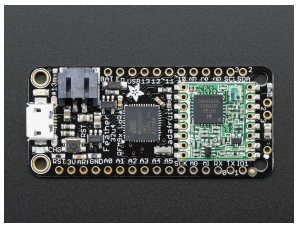
**Vin** - Power pin. The chip uses 3 VDC, a voltage regulator can take 3-5V and safely down. To power the board, give it the same power as the logic level of your microcontroller - e.g. for a 5V micro like Arduino, use 5V
**3vo** - The 3.3V output from the voltage regulator, can take up to 100mA from this.
**GND** - common ground for power and logic
**SCL** - I2C clock pin, connect to your microcontrollers I2C clock line.
**SDA** - I2C data pin, connect to your microcontrollers I2C data line.
**INT** - INTerrupt pin from the sensor. Can be programmed to do a couple different things with the i2c registers. For example trigger when a conversion is done, or when the light level has changed a lot.

<u>(to Uno, shown)</u>
**Vin** to 5V
**GND** to GND
**SDA** to A4
**SCL** to A5

<u>(to Feather 32u4, **NOT** shown)</u>
**Vin** to 3V
**GND** to GND
**SDA** to SDA
**SCL** to SCL



*Resources:* https://www.adafruit.com/product/1980



**Adafruit Feather 32u4 RFM95 LoRa (Long Range) Radio**

- 8 MHz and 3.3V logic
- 32k of flash, 2k of RAM

Power Pins

- GND - to ground
- BAT - positive voltage to/from the JST jack for the optional battery
- USB - positive voltage to/from the USB jack if connected
- EN - 3.3V regulator's enable pin. Pulled up, so connect to ground to disable the 3.3V regulator
- 3V - output from the 3.3V regulator, can supply 500mA peak

Logic Pins

- #0 / RX - GPIO #0, also receive (input) pin for Serial1 and Interrupt #2
- #1 / TX - GPIO #1, also transmit (output) pin for Serial1 and Interrupt #3
- #2 / SDA - GPIO #2, also the I2C (Wire) data pin. No pull up on this pin by default, when using with I2C, you may need a 2.2K-10K pullup. Also Interrupt #1
- #3 / SCL - GPIO #3, also the I2C (Wire) clock pin. No pull up on this pin by default, when using with I2C, you may need a 2.2K-10K pullup. Can also do PWM output and act as Interrupt #0.
- #5 - GPIO #5, can also do PWM output
- #6 - GPIO #6, can also do PWM output and analog input A7
- #9 - GPIO #9, also analog input A9 and can do PWM output. This analog input is connected to a voltage divider for the lipoly battery so be aware that this pin naturally 'sits' at around 2VDC due to the resistor divider
- #10 - GPIO #10, also analog input A10 and can do PWM output.
- #11 - GPIO #11, can do PWM output.
- #12 - GPIO #12, also analog input A11 and can do PWM output.
- #13 - GPIO #13, can do PWM output and is connected to the red LED next to the USB jack
- A0 thru A5 - These are each analog input as well as digital I/O pins.
- SCK/MOSI/MISO - These are the hardware SPI pins, used by the RFM radio module. Should be kept free as they should be available for the radio module. Also used to reprogram the chip with an AVR programmer if you need.

Radio Pins

- #8 - used as the radio CS (chip select) pin
- #7 - used as the radio GPIO0 / IRQ (interrupt request) pin.
- #4 - used as the radio Reset pin

Other

- RST - Reset pin, tie to ground to manually reset the AVR or launch the bootloader manually
- ARef - Analog reference pin. Normally the reference voltage is the same as the chip logic voltage (3.3V) but if you need an alternative analog reference, connect it to this pin and select the external AREF in your firmware. Can't go higher than 3.3V!

*Resources:* https://www.adafruit.com/product/3078

---

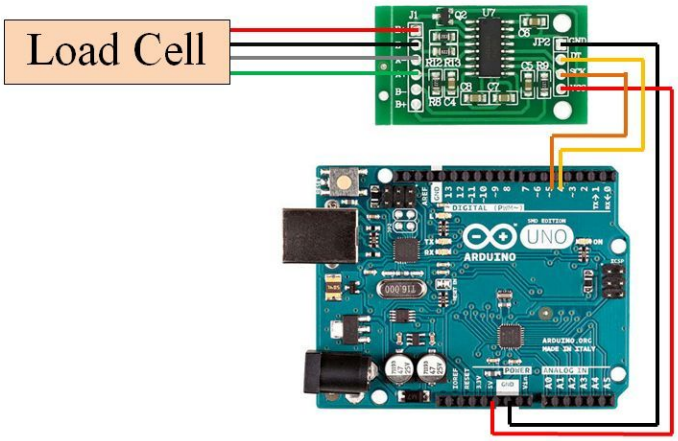
**XFW-HX711 Load Cell (0 - 5 kg)**

- 24 high-precision A/D converter, two analog input channels
- Operation supply voltage range 4.8 ~ 5.5V

(to Load Cell)
**E+** Excitation+ or VCC
**E-** Excitation- or ground
**A-** Amplifier-
**A+** Amplifier+

(to Uno, shown)
**VCC** to 5V
**GND** to GND
**SCK** to D5
**DT** to D6

(to Feather 32u4, **NOT** shown)
**VCC** to 3V
**GND** to GND
**SCK** to #13
**DT** to #12

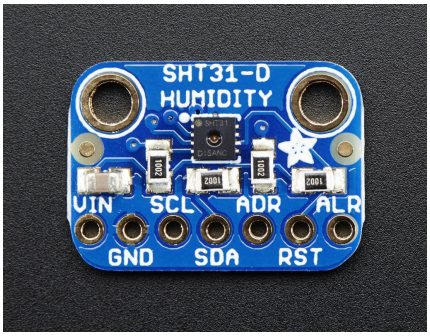



*Resources:* https://circuits4you.com/2016/11/25/hx711-arduino-load-cell/

---

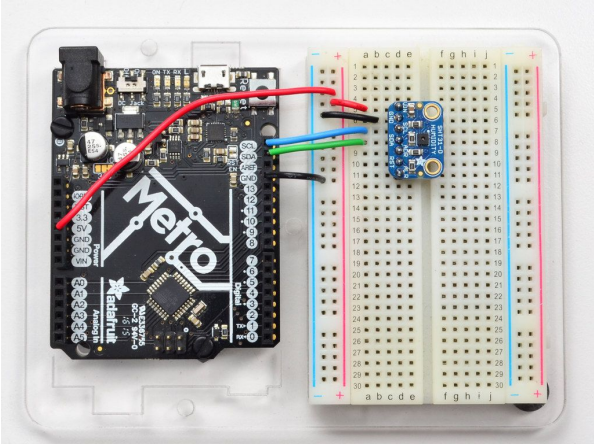
**Adafruit Sensiron SHT31-D Temperature & Humidity Sensor Breakout**

- Two I2C data/clock wires can share those pins with other sensors. Default I2C address is 0x44.




**VIN -** Power pin, can use 2.5-5VDC for power. Give it the same power as the logic level of your microcontroller - e.g. for a 5V micro like Arduino, use 5V. For a 3.3V controller like a Raspbery Pi, connect to 3.3V
**GND** - Ground for power and logic.
**SCL** - I2C clock pin, connect to your microcontrollers I2C clock line. This pin has a 10K pullup resistor to Vin.
**SDA** - I2C data pin, connect to your microcontrollers I2C data line. This pin has a 10K pullup resistor to Vin

**ADR** - The I2C address selection pin. This pin has a 10K pull down resistor to make the default I2C address 0x44.
**RST** - Hardware reset pint. Has a 10K pullup on it to make the chip active by default. Connect to ground to do a hardware reset.
**ALR** - Alert/Interrupt output. Can alert when an event has occurred.



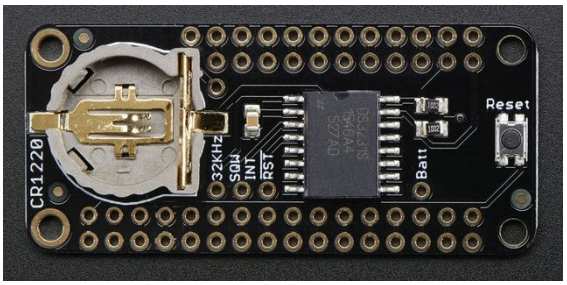
(to Metro, shown)
**VIN** to 5V
**GND** to GND
**SCL** to SCL
**SDA** to SDA

(to Feather 32u4, **NOT** shown)
**VIN** to 3V
**GND** to GND
**SCL** to SCL
**SDA** to SDA

*Resources:* https://www.adafruit.com/product/2857

---

**DS3231 Precision Real Time Clock (RTC) FeatherWing**




- Internal 32kHz timing crystal with temperature sensor that compensates for frequency changes
- Easily connects to Feather boards
- Needs a CR1220 coin cell to use battery-backup capabilities

- **SCL** - I2C clock pin, connect to your microcontrollers I2C clock line. This pin has a 10K pullup resistor to 3.3V
- **SDA** - I2C data pin, connect to your microcontrollers I2C data line. This pin has a 10K pullup resistor to 3.3V
- **BATT** - To use the battery for powering something else connect to the **BATT** breakout
- **32K** - 32KHz oscillator output. Open drain, need to attach a pullup to read this signal from a microcontroller pin (or use a microcontroller that can turn on it's internal pullup)
- **SQW** - Optional square wave or interrupt output. Open drain, need to attach a pullup to read this signal from a microcontroller pin (or use a microcontroller that can turn on it's internal pullup) **(**signal goes to Feather Digital Pin 11 through the Evap Shield**)**

- **RST** - Used to reset an external device or indicate when main power is lost. Open drain, but has an internal 50K pullup. The pullup keeps this pin voltage high as long as Vin is present. When Vin drops and the chip switches to battery backup, the pin goes low.

(to Feather 32u4)
The FeatherWing directly stacks on to the Feather 32u4, just make sure all the pins are correctly lined up.
**The interrupt pin on the PCB goes to P11 on the Feather 32u4**
****Only the SQW (interrupt) signal needs to be directly wired from the SQW pin on the RTC PCB to the RTC INT pin on the OPEnS lab custom Evap Shield****

*Resources:*
https://learn.adafruit.com/ds3231-precision-rtc-featherwing/pinouts
https://open-sensing.squarespace.com/lorablog/2017/10/23/low-power-rtc-and-sleeping-processors-a-guided-walkthrough

---

**Tare Button**
Taring occurs in the initial setup under the load cell calibration section. The scale.tare() function resets the the scale to zero at the start of the code, so when the evaporometer is collecting load cell values from precipitation it does not take into account the weight of the container or the fibers. Will add a tare button in the future. (or can be done remotely?)

- P10 on PCB board