Water Leak Alarm

Concept Document

*By: Bob Glicksman; 01/19/17*

# Problem.

A leaking washing machine, overflowing tub or sink, or leaky pipes or toilet can cause considerable damage to a home if not detected and shut off early. Years ago, my washing machine water valve stuck open while washing clothes late at night. We awoke in the middle of the night to find a flood in the washer/dryer vestibule resulting in $12,000 in water damage. Since that time, I have been reluctant to run the washing machine at night or at any time when we are not home to watch over it. This is not always convenient and it would help to have a solution that can detect a water leak and signal us via a local alarm (for when we are at home) and/or a smartphone notification (for when we are out).

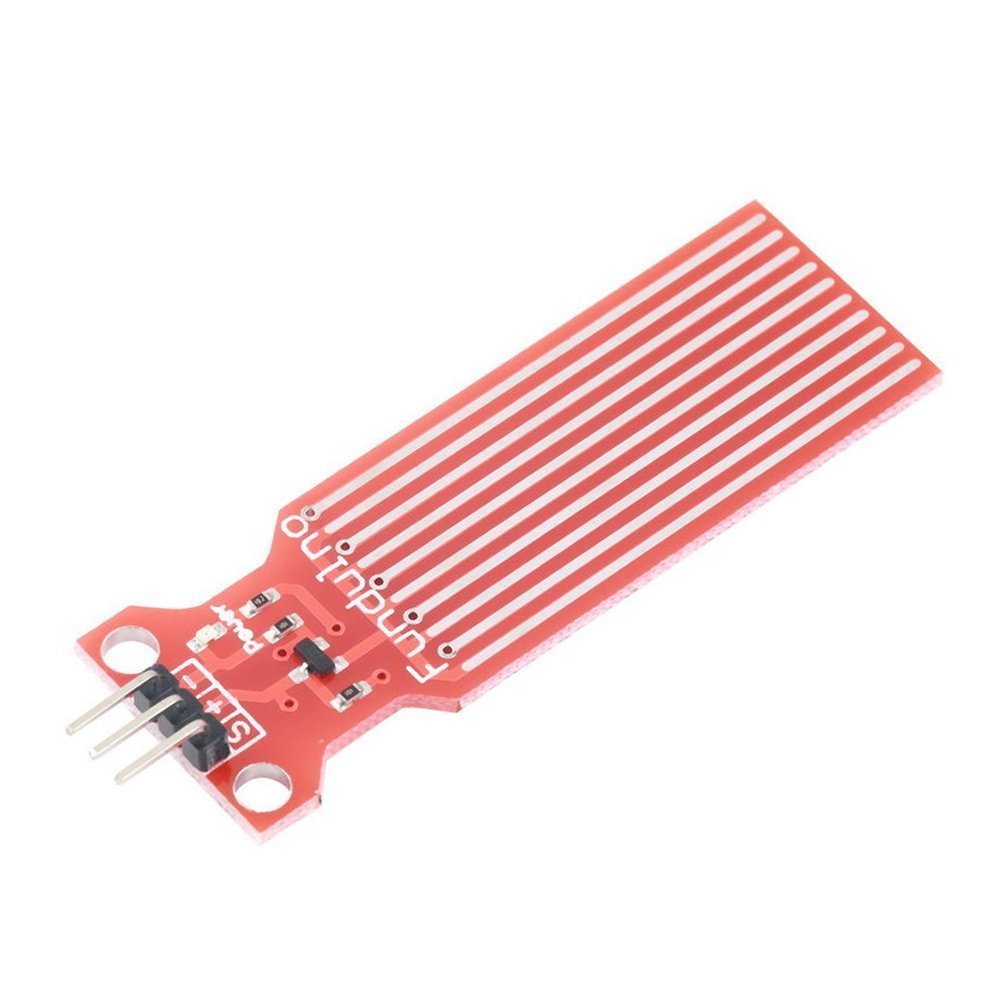
Leaky or humid basements present a similar problem. Adding a second water leak detection sensor, as well as temperature and humidity measurement to the project, would cover all of these bases.

# Proposed Solution.

There are inexpensive, analog water detection sensors, such as:

<https://www.amazon.com/Solu-Detection-Arduino-Sensitivity-Surface/dp/B00UJ0DVN4/ref=sr_1_5?ie=UTF8&qid=1484328763&sr=8-5&keywords=water+level+sensor+arduino>

This sensor looks as follows:



*Water Leak Sensor.*

The sensor consists of a bunch of parallel conductors that have a very high DC resistance when the sensor is dry. However, when water shorts out adjacent conductors, the resistance falls. Analysis and testing of this device was performed and is documented in:

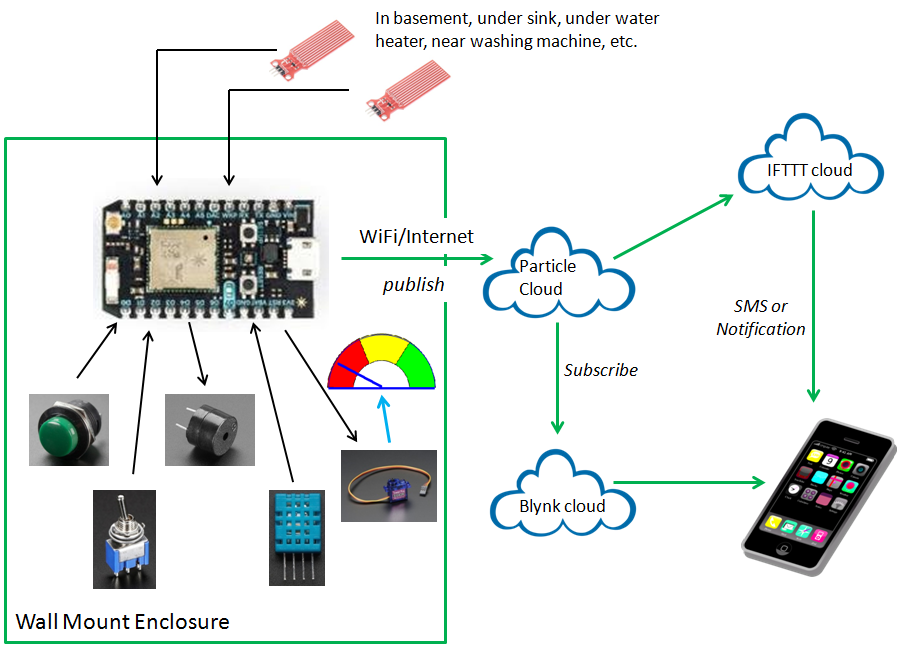
<https://drive.google.com/open?id=1c0LSk9H2GjFn0QsQXAi1s4Fm1Fzk0K3_fwEfErgecaM>

The idea is to use a Particle Photon (e.g. <https://www.adafruit.com/products/2721>) to sense the presence and amount of water via an analog input from this sensor. Based upon the analysis and test results for the water level sensor, the sensor’s analog output should be thresholded to produce an alarm condition when the voltage output goes from 0 volts (dry) to a relatively low value (e.g. 0.5 volts). Tests of the sensor have determined that a pool of water about 1 mm deep will cause this level of change. The Photon will sound a loud, pulsing, audible alarm made from an inexpensive piezo buzzer (e.g. <https://www.adafruit.com/products/1536>) whenever the threshold is exceeded. An LED will also flash when the water level threshold is exceeded. A backlit pushbutton switch (e.g. <https://www.adafruit.com/products/1440>) can provide both the LED indicator and a means to mute the audible alarm when pressed.

Temperature and humidity can both be measured by one, inexpensive (DHT11) sensor (e.g. <https://www.adafruit.com/products/386>). An inexpensive servo (e.g. <https://www.adafruit.com/products/169>) can be used to provide a visual “meter” for temperature or humidity, based upon a toggle switch selection (e.g. <https://www.adafruit.com/products/3221>).

The WiFi capability of the Photon will be used to provide a smartphone notification or SMS message of an alarm condition, using IFTTT. The Blynk app (<http://www.blynk.cc/>) may optionally be used to remotely display alarm status, temperature and humidiy readings on a smartphone, over the Internet.

The concept is shown, schematically, below:



*Water Leak Alarm Concept.*

# Operational Details.

The project will support 2 water level sensors. Each water level sensor may be placed under or near a sink, toilet, washing machine, water heater, or basement wall to be monitored. The sensors should be positioned vertically so that the maximum number of conductors are shorted when the water level rises to the bottom of the sensor and above.

The two sensors will be cabled up to the Photon which, along with the DHT11, toggle switch, pushbutton switch, piezo buzzer and servo, will be wall mounted in a plastic box. The location of the plastic box will be somewhere visible, so that the temperature/humidity indication can clearly be seen and the LED/mute button accessed. Analysis of the water level sensor has determined that it has an output impedance of about 100 ohms; thus, a relatively long cable (e.g. 10 feet) can be used between the sensors and the electronics enclosure, particularly if several threshold readings are required to trigger an alarm.

The firmware in the Photon will continuously monitor the amount of water on the sensors, based upon their analog signal value. The sensors will be manually calibrated (via experimentation) so that a reasonable threshold for “flood alert” can be determined. When the “flood alert” threshold is exceeded for some number of measurements (e.g. five), the piezo buzzer will be pulsed to create an easily heard audible alarm. The alarm will latch so that it sounds continuously until someone presses the pushbutton to silence it. The alarm will also publish a message to IFTTT which will be used to create an SMS text or cellphone notification of the flood alarm condition. The alarm will reset when some number (e.g. five) of measurements falls below the threshold value.

The Photon will also will read out temperature and humidity values from the DHT11 sensor every few seconds. The servo “meter” will read out either temperature or humidity based upon the toggle switch selection. The current temperature and humidity readings and the water level alarm status will be published to the Blynk cloud and displayed on the Blynk app on a smartphone.

# Possible Enhancements.

Alarm thresholds and scale factors are presently envisioned as being hardcoded as named constants, but they could be made programmable over the Internet (from the Blynk app or at least from the SIS Debug Client). While this is easy to implement, it raises the issue of the need for non-volatile storage of this user-defined data. We could add in an I2C EEPROM to do this (raising the parts count - is it worth it?).

A 433 Mhz OOK receiver might be added to the electronics. SIS compatible firmware in the Photon could read received trip codes from motion or other types of alarm sensors, whose information could be integrated into the IFTTT notification service and/or Blynk app.