

[blog.thecell.eu](https://blog.thecell.eu)

# Weather station

Name \*

3 minutes

On 19. November 2019 by TheCell

At work I started participating in a D.I.Y. club where we will build a weather station. This should reflect the weekly progress and what we are working on. Additionally this should in the end be a tutorial for a complete weather station based on a Raspberry and everything should work with the I2C protocol. Let's start.

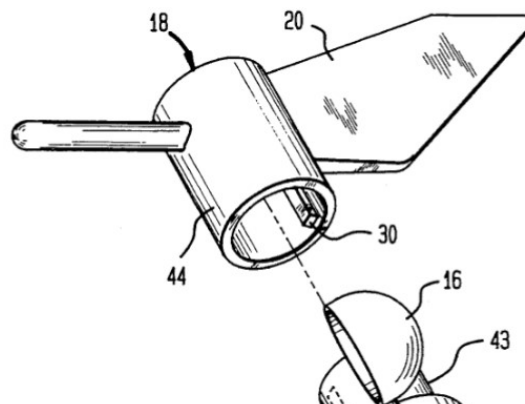
First we got an introduction by Brice Giboudeau on how an anemometer works. There are various options on how they work. [Here](#) is a list on what else is out there, but we looked into one that counts impulses. Basically there is a magnet attached to the rotating top and in the inner part is a reed switch that closes whenever the cup makes a rotation and passes the switch. We will want to count these and calculate the wind speed from there. Because we want to create a low power solution we will not count these with the Raspberry directly and instead have a little circuit that updates the raspberry occasionally.

**U.S. Patent**

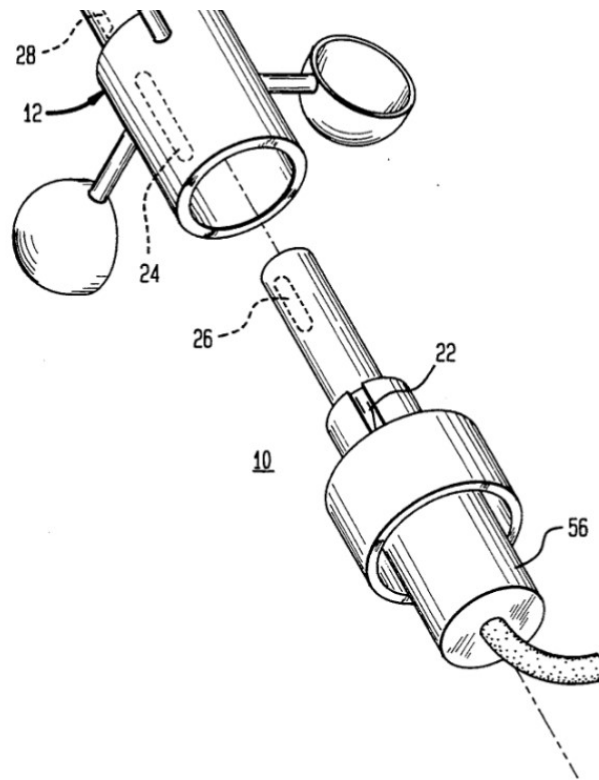
Nov. 8, 1994

Sheet 2 of 6

**5,361,633**



**FIG. 1a**



For the Windspeeds we want to measure average speed and also wind gusts. This we will do via a ripple-carry binary counter: <https://www.google.com/search?q=cd4020> Of course 14 bit is way overkill but we can reset the counter with any bit convenient.

We go for **30 km/h average speed** and want to measure **up to 150 km/h** windspeeds. Resulting in Windspeed  $W_s$ :

$150000\text{m}/3600\text{s} = 41.66\text{ m/s}$ . First we calculate how many impulses this will generate which we have to count:

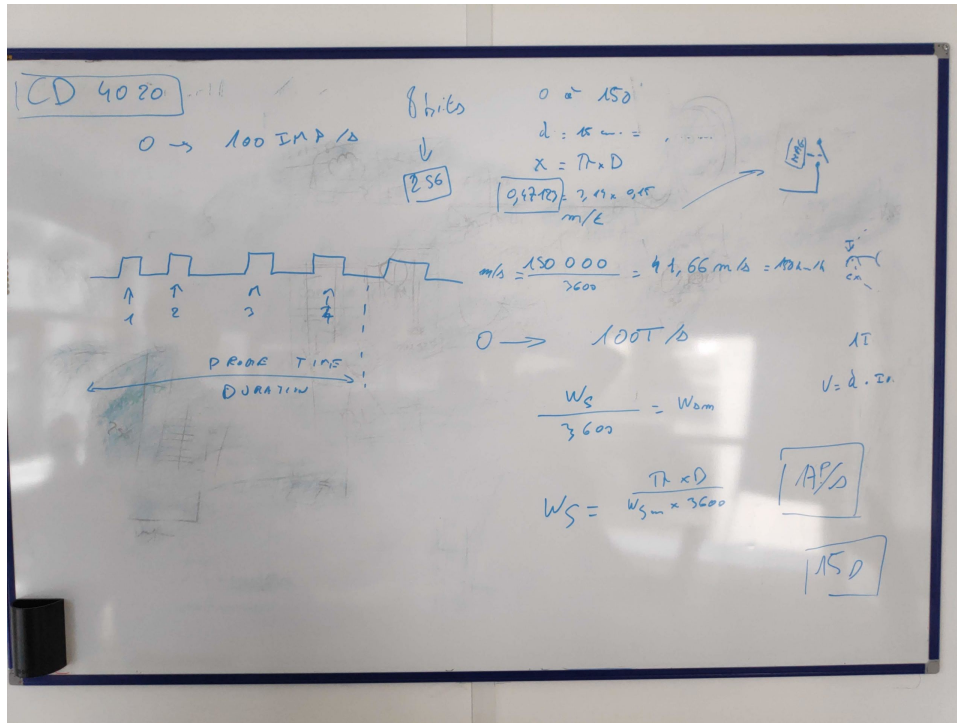
If we have an anemometer with 0.15m diameter and we want to know how many turns it rotates per meter we calculate:

The [circumference](#)  $C$  of our anemometer:  $C = \text{diameter} * \pi = 0.15 * \pi = 0.471239$ . This means our anemometer has a circumference of 0.47 meter.

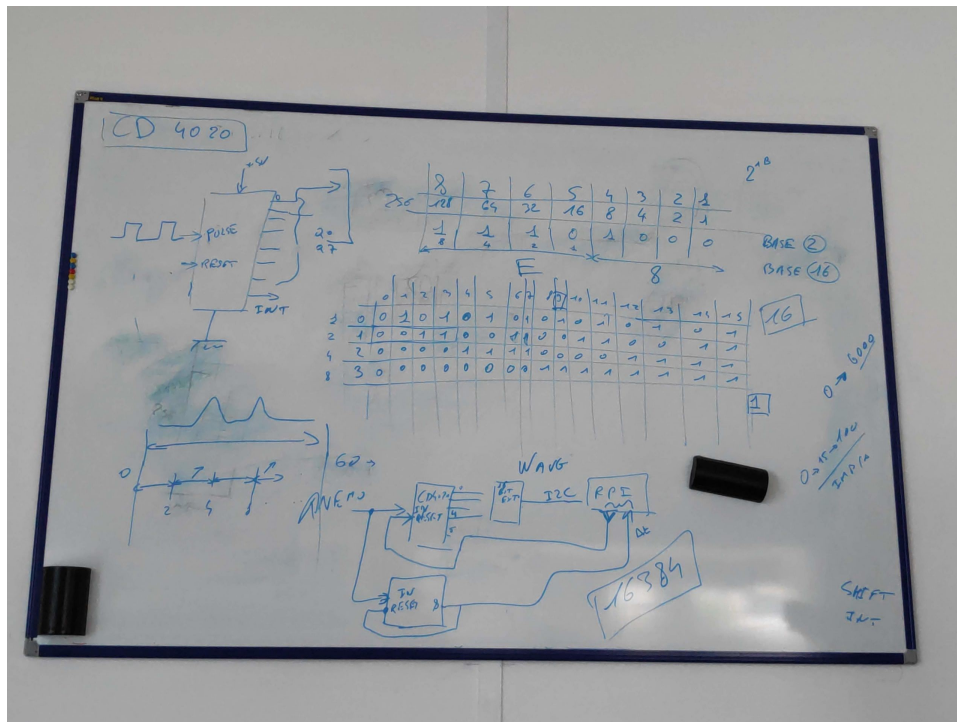
Based on this we can calculate what our circuit must be able to support. [Revolutions](#) (T for turns) per second:  $W_s / C = 41.66 / 0.471239 = 88.6\text{ turns / second}$ . For our project we will calculate with **100 T/s** to be safe.

If we want to pull windspeed data only every minute this means that we need to count up to 6000 Turns per minute. Additionally we will want to display wind gust speeds. We will do this via interrupt, a second CD4020 will update the Raspberry whenever it counts 256

rotations and the Raspberry will track time between interrupts.  
 Luckily our CD4020 is a 14 bit counter resulting in  $2^{14}$  (16384) counts before overflowing.



Whiteboard explanation by Brice Giboudeau



Whiteboard explanation by Brice Giboudeau