Anemometer

Calculating wind speed

I have a three cup anemometer with half spherical cups of 80mm diameter and the cup canters are 155mm from rotation point. if the device turns and 60 rpm, what is the resulting wind speed?

To calculate the wind speed based on the given information for your three-cup anemometer, you can use the formula:

Wind Speed (in meters per second) = (π * D * RPM) / (2 * C)

Where:

- D is the diameter of the cups (in meters)
- RPM is the revolutions per minute
- C is the distance from the cup center to the rotation point (in meters)

In your case:

- D = 80mm = 0.08m
- RPM = 60 (revolutions per minute)
- C = 155mm = 0.155m

Plugging in the values into the formula:

Wind Speed = (π * 0.08 * 60) / (2 * 0.155)

Simplifying the equation:

Wind Speed ≈ 7.307 m/s

Therefore, with a rotation speed of 60 RPM, the resulting wind speed would be approximately 7.307 meters per second.

interrupt-driven counter

To create an interrupt-driven counter on a Raspberry Pi using GPIO pins, you can follow these steps:

Import the required libraries

```
import RPi.GPIO as GPIO
import time
```

Set up the GPIO mode and initialize the counter variable

```python GPIO.setmode(GPIO.BCM) counter = 0 ```

Set up the GPIO pin for input and configure the interrupt

```
GPIO.setup(pin_number, GPIO.IN, pull_up_down=GPIO.PUD_UP)
GPIO.add_event_detect(pin_number, GPIO.FALLING, callback=increment_counter)
</cpde>

**Note:** Replace ''pin_number'' with the GPIO pin number you want to use.

==== Define the callback function to increment the counter ====

<code python>
def increment_counter(channel):
    global counter
    counter += 1
```

Create a loop to keep the program running

```
try:
    while True:
        time.sleep(1)
except KeyboardInterrupt:
    GPIO.cleanup()
```

This code sets up the GPIO pin as an input with a pull-up resistor and detects falling edge interrupts. When the interrupt is triggered, the `increment_counter` function is called, which increments the counter variable. The program then enters an infinite loop, waiting for interrupts.

Remember to replace `pin_number` with the desired GPIO pin number for your application.

Important notes about reed switches

The sensitivity of a REED switch is characterized by the value of the magnetic field which is required to change its state. The magnetic field is expressed in Ampere-turns (or ampere/meter). In order to close a normally open switch, for instance, a certain number of Ampere-turns, called **CAT**, are required (**c**losing **A**mpere-**t**urns). To open the same switch the magnetic field must be reduced down to an **o**pening value in **A**mpere-**t**urns **OAT**).

OAT<CAT is always true

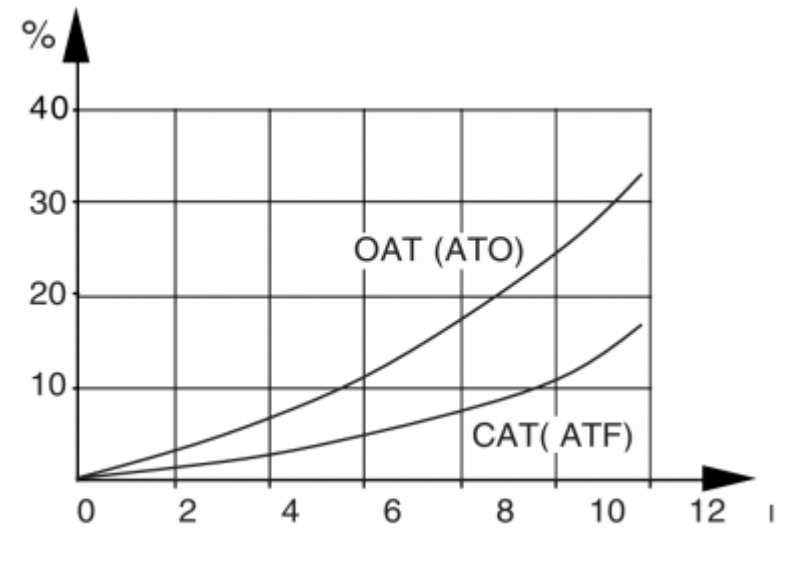
Order of magnitude OAT-CAT :
```

For standard switches	
CAT	12 and 50 Ampere-turns
OAT	5 and 30 Ampere-turns
For standard or mercury wetted switches	
CAT	35 and 150 Ampere-turns
OAT	10 and 80

NOTE : if the leads on a switch are cut, the CAT and OAT values will increase

Bending modifies the CAT and OAT values of a switch Cutting the leads on a REED switch has the effect of increasing the CAT and OAT ratings in a significant manner.

The figure below illustrates this variation.



- ap-interen.pdf
- notes_reed_switch_modifications.pdf