

## RC4:

Intenté sacar la temperatura de varias maneras:

Primero intente instalando OpenHardwareMonitor, pero este necesitaba el uso de C#

The screenshot shows the Open Hardware Monitor website and its application interface. The website header includes 'Open Hardware Monitor' with a logo, and navigation links: 'Home', 'Downloads', 'Screenshots', 'Documentation', 'Support', and 'License'. There are 'Login' and 'Register' buttons in the top right. The application window is titled 'Open Hardware Monitor' and shows a tree view of hardware sensors. The 'Sensors' list includes:

- Windows
  - Gigabyte X570 AORUS MASTER
    - ITE IT8688E
      - Voltages
        - CPU VCore: 0.420 V, 0.216 V, 1.476 V
      - Temperatures
        - System 1: 36.0 °C, 36.0 °C, 36.0 °C
        - CPU: 42.0 °C, 41.0 °C, 68.0 °C
      - Fans
        - CPU Fan: 301 RPM, 0 RPM, 431 RPM
  - AMD Ryzen 9 3950X
    - Temperatures
      - System 2: 38.0 °C, 38.0 °C, 39.0 °C
  - Load
    - CPU Package: 41.8 °C, 41.3 °C, 68.1 °C
    - CPU CCD #1: 41.0 °C, 39.3 °C, 61.0 °C
    - CPU CCD #2: 41.0 °C, 39.5 °C, 55.5 °C
  - Powers
    - CPU Total: 0.0 %, 0.0 %, 7.2 %

- NVIDIA GeForce RTX 2070 SUPER

The application window also displays a table with columns 'Sensor', 'Value', 'Min', and 'Max'. To the right of the application window, there is a description of the software and a 'Download Open Hardware Monitor 0.9.5' button.

OpenHardwareMonitor manda los atributos a WMI, pero este necesita la cabecera wbemidl.h, pero no pude llegar a entender completamente como incluirlo o descargarlo, vi en un foro como necesitaba descargar SDK y vs6 (<http://forums.codeguru.com/showthread.php?369894-Cannot-open-include-file-Wbemidl-h-No-such-file-or-directory>).

## Initializing COM for a WMI Application

05/31/2018 • 2 minutes to read •

The first step in connecting to WMI is setting up the COM calls to **CoInitializeEx** and **CoInitializeSecurity**.

The code examples in this topic require the following references and `#include` statements to compile correctly.

```
C++  
  
#define _WIN32_DCOM  
#include <iostream>  
using namespace std;  
#include <wbemidl.h>  
#pragma comment(lib, "wbemuuid.lib")
```

The following procedure describes how to initialize COM from a client application.

También intente entrar a la memoria directamente, pero me parece que Windows no permite entrar a un espacio de memoria por dirección, además del hecho que no comprendí como direccionarlo con la información de la siguiente imagen:

**4.2 Thermal (THM)**

- The thermal block contains all the features related to temperature sensing, control, and reporting. It includes:
- Temperature collection and calculation using TCON (digital control logic) and TMON and Remote Diode Interface macros.
  - Fan speed control for off-chip fans.
  - Temperature reporting through the SMBUS interface.

**4.2.1 Registers**

**SMUTHMx00000000 (SMU::THM::THM\_TCON\_CUR\_TMP)**

Read-write. Reset: 0000 0000h.	
aliasSMN; SMUTHMx00000000; SMUTHM=0005 9800h	
Bits	Description
31:21	<b>CUR_TEMP.</b> Read-write. Reset: 000h. Provides current control temperature.
20	Reserved.
19	<b>CUR_TEMP_RANGE_SEL.</b> Read-write. Reset: 0. 0=Report on 0C to 225C scale range. 1=Report on -49C to 206C scale range.
18:0	Reserved.

Seguiré investigando.