



گزارش پایانی آزمایشگاه سخت افزار

گروه سه

اعضا:

بهار خدابخشیان 97105906

محمدامین شریفی 97110058

درنا دهقانی 97105939

فهرست

۳	۱. مقدمه...
۳	۲. نحوه انجام پروژه
۳	۳. مأذول ها

۳	Raspberry Pi ۳.۳.۱
۳	LED ۳.۲
۴	۴. سنسور دما ۳.۳
۴	۴. بردبورد ۳.۴
۴	۴. سنسور مکان ۳.۵
۵	۴. معماری سیستم ۴
۵	۵. نحوه ارتباط سنسورها
۵	NEO-7M-C ۵.۱
۶	RW1820 ۵.۲
۶	LED ۵.۳
۷	۶. گرفتن دما
۸	۷. گرفتن مختصات GPS
۱۲	۸. تشخیص جهت تابش آفتاب به کمک مختصات
۱۳	۹. محاسبه جهت حرکت جسم (خودرو)
۱۴	۱۰. صحتسنجی سنسور GPS
۱۵	۱۱. روشن و خاموش کردن خنک کنندها
۱۷	۱۲. بخش اصلی پروژه
۱۹	۱۳. تستها
۲۰	۱۳.۱ لوکیشن: دانشگاه شریف، تهران، ایران (جهت حرکت: شمال شرقی)
۲۰	۱۳.۲ لوکیشن: دانشگاه شریف، تهران، ایران (جهت حرکت: غرب)
۲۲	۱۳.۳ لوکیشن: دانشگاه شریف، تهران، ایران (جهت حرکت: جنوب)
۲۳	۱۳.۴ لوکیشن: جزیره‌ی ماداگاسکار (جهت حرکت: جنوب غربی)
۲۴	۱۳.۵ لوکیشن: جزیره‌ی ماداگاسکار (جهت حرکت: شمال شرقی)
۲۵	۱۳.۶ لوکیشن: رم، ایتالیا (جهت حرکت: شمال)
۲۶	۱۳.۷ لوکیشن: رم، ایتالیا (جهت حرکت: شرق)
۲۷	۱۴. بسته‌بندی
۲۷	۱۵. نتیجه‌گیری
۲۷	۱۶. مراجع

1. مقدمه

هدف از این پروژه، طراحی سیستمی برای متعادل سازی دمای خودرو می‌باشد، به طوریکه به مصرف انرژی به شکلی بهینه کمک کند. از آنجایی که در یک خودرو که در هوای آفتابی در حال حرکت است همواره یک سمت از آن در معرض نور خورشید قرار دارد، این بخش گرمتر از سایر بخش‌ها می‌باشد. اگر برای خنک کردن خودرو، خنک کننده‌ی تمامی قسمتها را روشن کنیم، این کار موجب مصرف بی‌رویه انرژی در خنک کردن بخش‌هایی از خودرو می‌شود که در معرض نور مستقیم نیستند. برای جلوگیری از این مشکل، به کمک GPS و موقعیت خودرو می‌توانیم وضعیت قرارگیری آن نسبت به خورشید را بیابیم تا متوجه شویم کدام سمت آن رو به خورشید است. سپس به کمک حسگرهای دمای بخش‌های مختلف، به مقایسه آنها می‌پردازیم و در صورت تفاوت دما، خنک کننده‌ی آن بخش را روشن می‌کنیم تا دمای خودرو به تعادل برسد. نهایتاً پس از تعادل کامل دمای خودرو، خنک کننده‌ی خودرو خاموش می‌گردد. لازم به ذکر است که بررسی دمای بخش‌های مختلف خودرو باید با هر بار تغییر وضعیت آن نسبت به خورشید انجام گیرد. هم چنین، سیستم خنک کننده در قالب چراغ‌های LED پیاده سازی می‌شوند؛ یعنی اگر بخشی از خودرو نیاز به خنک کننده داشت، LED مربوط به آن روشن می‌گردد و اگر یک خنک کننده باید خاموش می‌شد، LED مربوط به آن خاموش می‌گردد.

2. نحوه انجام پروژه

برای انجام این پروژه، ما از رزبری پای ۳ برای کنترل سیستم استفاده خواهیم کرد. برای بخش بندی خودرو آن را به ۴ قسمت راننده، شاگرد، و ۲ بخش سرنوشتیان عقب تقسیم می‌کنیم که در هر بخش یک حسگر دما قرار می‌گیرد. یک سنسور GPS نیز در خودرو قرار دارد و تمامی این سنسورها به رزبری پای متصل می‌گردند. جهت نمایش هشدارهای روشن و خاموش کردن چراغ‌های خنک کننده نیز از LED‌های متصل به رزبری پای استفاده می‌شود. برای نوشتن برنامه کنترلگر سیستم از زبان پایتون و کتابخانه‌های آن استفاده می‌کنیم. نحوه اجرای پروژه را در [این ویدیو](#) شرح داده‌ایم.

3. ماژول‌ها

Raspberry Pi 3 .3.1

ما از این ماژول به عنوان پردازنده سیستم خود استفاده کردیم. این دستگاه با قابلیت نصب سیستم عامل، اجرای کد‌های پایتون و ارتباط با سنسور‌هایی از قبیل سنسور دما و GPS امکان ارتباط ساده بین اجزای سیستم را ارائه می‌دهد.



Raspberry Pi 3.1 تصویر

LED .3.2

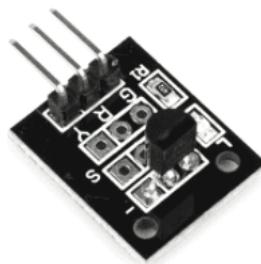
برای نمایش هشدارهای روشن و خاموش کردن خنک کننده در ۴ بخش خودرو، به ۴ عدد LED نیاز داریم.



تصویر 2 LED

3.3. سنسور دما

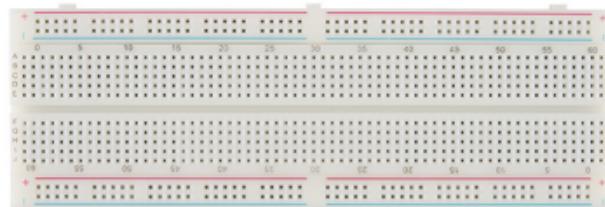
برای اندازه‌گیری دما در خودرو از سنسور دمای RW1820 استفاده خواهد شد. در این پروژه از ۴ عدد از این سنسور برای سنجش دمای ۴ ناحیه خودرو استفاده خواهیم کرد.



تصویر 3 RW1820

3.4. بردبورد

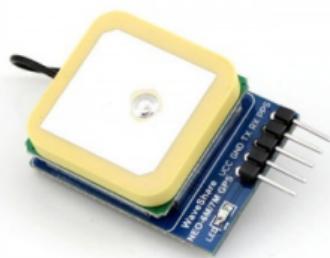
برای اتصال سنسورها و LED ها به رزبری پای به بردبورد احتیاج داریم.



تصویر 4 بردبورد

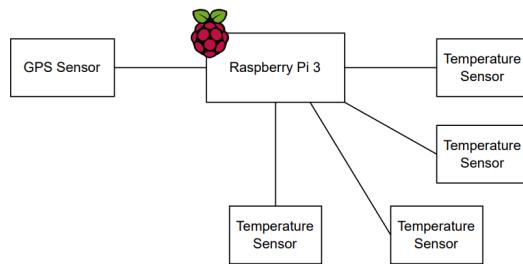
3.5. سنسور مکان

برای بدست آوردن مکان ماشین، از سنسور مکان NEO-7M-C استفاده خواهد شد. این سنسور با ریت ماسکسیم 5Hz این امکان را به ما می دهد تا موقعیت خودرو را بدست آوریم.



تصویر ۵ NEO-7M-C

۴. معماری سیستم

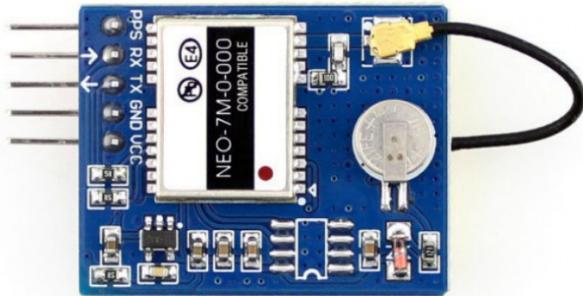


تصویر ۶ معماری سیستم

۵. نحوه ارتباط سنسورها

NEO-7M-C .5.1

این سنسور مکان یاب دارای ۴ پین است که ۲ تا از آنها برای GND و VCC هستند. دو تای دیگر با نام های RX و TX به ترتیب به پین های RXD و TXD در رزبری پای متصل خواهند شد.



تصویر ۷ پین های سنسور مکان یاب

پس از اتصال این پین ها باید config های لازم برای اتصال رزبری و سنسور را انجام دهیم. سپس باید دستورات زیر را برای UART اجرا کنیم.

```
sudo raspi-config → Interfacing Options → Enable Serial
```

سپس برنامه های مورد نیاز برای ارتباط با سنسور را با دستور زیر نصب می کنیم.

```
sudo apt-get install gpsd gpsd-clients
```

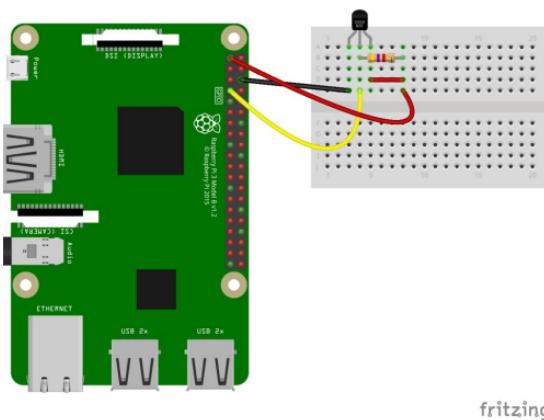
سپس با استفاده از دستور زیر می توان برنامه **gpsd** را اجرا کرد.

```
sudo gpsd /dev/ttyS0 -F /var/run/gpsd.sock
```

در آخر نیز با پکیج **gps** در پایتون می توان داده های لازم مکان را دریافت کرد. در هنگام پیاده سازی متوجه شدیم به دلیل مشکلات موجود در پیاده سازی کتابخانه، قابل استفاده نیست و نهایتاً تصمیم به استفاده از کتابخانه **pynmea2** گرفتیم.

RW1820 .5.2

مراحل اتصال این سنسور نیز مشابه مراحل سنسور قبلی است. این سنسور ۳ پین دارد که دو تا از آنها **VCC** و **GND** هستند. پین سوم که دیتا است باید به پین های **GPIO** در رزبری وصل شود با این تفاوت که نیاز به یک مقاومت نیز هست که با توجه به اینکه ما از مازول این سنسور استفاده می کنیم، نیاز به مقاومت اضافی نیست.



تصویر 8 نحوه اتصال سنسور دما با رزبری پای

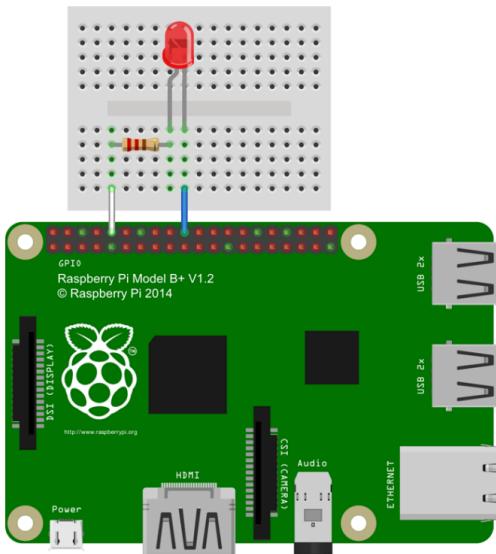
سپس مانند سنسور قبلی باید رابط با رزبری پای را تنظیم کنیم که با دستور زیر قابل انجام است.

```
sudo raspi-config → Interfacing Options → Enable 1-Wire
```

در انتهای نیز با کتابخانه **W1ThermSensor** می توان اطلاعات دما را از سنسور دریافت کرد.

LED .5.3

در این پروژه قرار بر این است که از **LED** برای نشان دادن روشن بودن خنک کننده هر قسمت از ماشین استفاده کنیم. **LED** را از طریق بردبورد به رزبری پای وصل می کنیم تا بتوانیم آن را با استفاده از برنامه پایتون و پکیج **GPIO.RPi** کنترل (روشن و خاموش) کنیم.



تصویر ۹ نحوه اتصال LED به رزبری پای

6. گرفتن دما

برای این بخش از پروژه مژول DS18B20 که شامل سنسور RW1820 می‌باشد را انتخاب کرده بودیم که به شکل مژول به ما داده شد. در این مژول خروجی‌های G، R و 7 به ترتیب باید به پین‌های VCC، GND و GPIO 4 (GPCLK0) متصل شوند. پین‌های General Purpose Clock یا [GPCLK](#) بدون نیاز به کنترل نرم‌افزاری ایجاد می‌کنند.

ابتدا می‌خواهیم نحوه نمایش دما توسط این سنسور را ببینیم. برای تنظیمات اولیه و ایجاد رابط یک طرفه بین مژول و رزبری پای، پس از روشن کردن رزبری پای با زدن دستور `sudo nano /boot/config.txt` در انتهای فایل باز شده عبارت `dtoverlay=w1-gpio` را اضافه می‌کنیم. سپس رزبری پای را `reboot` می‌کنیم. پس از روشن شدن مجدد و زدن دو دستور `sudo modprobe w1-gpio` و `sudo modprobe w1-therm` در دستور `cd /sys/bus/w1/devices` دایرکتوری را تغییر می‌دهیم. به کمک `ls` می‌توانیم لیست دایرکتوری‌های موجود در آن را مشاهده کنیم. دایرکتوری که اسم با فرمت XXXXXXXXXXXXXXXX-28 دارد شامل اطلاعات دما هستند. با `cat w1_slave` وارد آن شده و با دستور `t=XXXXXX` داده شده است و نشان می‌دهد دما XXX.XXX درجه سانتیگراد است. هر کلاک را مشاهده می‌کنیم که به شکل `w1_slave` در فایل این فایل، اطلاعات دمای گرفته شده در آن کار کافیست هر 1 یا 2 ردیف از بردبورد را به یکی از پین‌های رزبری پای اختصاص دهیم و پین‌های هر یک از چهار مژول دما به آن ردیف‌ها متصل کنیم.

پس از اتصال صحیح هر چهار مژول دما، در پوشه `devices` که پیش‌تر تنها یک دایرکتوری که با فرمت اسم -28 XXXXXXXXXXXXXXXX داشت، اکنون چهار دایرکتوری با این مشخصات قابل مشاهده است که فایل `w1_slave` در هر دایرکتوری اطلاعات دمای گرفته شده توسط آن مژول را نمایش می‌دهد.

```
mmd@raspberrypi: ~/Desktop/project/hw_project-group-3/Code
Sensor 1 => 25.5 (C) - 77.9 (F)
Sensor 2 => 25.375 (C) - 77.675 (F)
Sensor 3 => 25.062 (C) - 77.11160000000001 (F)
Sensor 4 => 27.562 (C) - 81.61160000000001 (F)
Sensor 1 => 25.562 (C) - 78.0116 (F)
Sensor 2 => 25.375 (C) - 77.675 (F)
Sensor 3 => 25.0 (C) - 77.0 (F)
Sensor 4 => 26.062 (C) - 78.9116 (F)
Sensor 1 => 25.5 (C) - 77.9 (F)
Sensor 2 => 25.375 (C) - 77.675 (F)
Sensor 3 => 25.0 (C) - 77.0 (F)
Sensor 4 => 25.75 (C) - 78.35 (F)
Sensor 1 => 25.562 (C) - 78.0116 (F)
Sensor 2 => 25.375 (C) - 77.675 (F)
Sensor 3 => 25.062 (C) - 77.11160000000001 (F)
Sensor 4 => 25.625 (C) - 78.125 (F)
Sensor 1 => 25.562 (C) - 78.0116 (F)
Sensor 2 => 25.375 (C) - 77.675 (F)
Sensor 3 => 25.062 (C) - 77.11160000000001 (F)
Sensor 4 => 25.0 (C) - 77.0 (F)
```

تصویر 10 نمایش دمای چهار سنسور دما

در نهایت، برای استفاده از این دماها برای مراحل آتی از کد پایتون استفاده می‌کنیم که دما را از فایل‌های گفته شده می‌خواند. این کد در فایل `temp_sensor.py` قابل مشاهده است.

اطلاعات ذکر شده در این بخش عمدتاً از [این لینک](#) گرفته شده است.

برای تشخیص اینکه هر سنسور دما بیانگر دمای چه بخشی از خودروست، به طور دستی به تغییر دمای محیط اطراف هر یک می‌پردازیم و سنسور دماها را به ترتیب 0 (راننده)، 1 (سرنشین عقب پشت راننده)، 2 (شاگرد راننده) و 3 (سرنشین عقب پشت شاگرد راننده) نامگذاری می‌کنیم.

7. گرفتن مختصات GPS

برای دریافت مختصات خودرو از سنسور Neo-7m-C استفاده کردیم که 4 پین VCC، GND، TX و RX آن به ترتیب به پین‌های 3.3 یا 5 ولت، GPIO 14 (TX) و GPIO 15 (RX) متصل می‌شوند. برای برقراری اتصال بین سنسور و رزبری پای لازم است تنظیماتی انجام گرددند. پس از روشن کردن رزبری پای با زدن دستور `sudo nano /boot/config.txt`، در انتهای فایل باز شده عبارات

```
dtparam=spi=on
dtoverlay=pi3-disable-bt
core_freq=250
enable_uart=1
force_turbo=1
```

را اضافه کرده و ذخیره می‌کنیم. از آنجایی که سیستم رزبری از UART به عنوان یک کنسول سریال استفاده می‌کند، برای خاموش کردن این عملکرد باید فایل `/boot/cmdline.txt` را تغییر دهیم. برای این کار ابتدا از نسخه اصلی آن با دستور `sudo cp /boot/cmdline.txt /boot/cmdline_backup.txt` یک فایل پشتیبان تهیه می‌کنیم. سپس با دستور `sudo nano /boot/cmdline.txt` آن را باز کرده و محتوای آن را با عبارت زیر جایگزین کرده و ذخیره می‌کنیم.

```
dwc_otg.lpm_enable=0 console=tty1 root=/dev/mmcblk0p2
rootfstype=ext4 elevator=deadline fsck.repair=yes rootwait quiet
splash plymouth.ignore-serial-consoles
```

سپس رزبری پای را `reboot` می‌کنیم.

پس از این تنظیمات می‌توانیم کار کرد سنسور را مشاهده می‌کنیم. پس از اتصال سنسور به رزبری پای، یک LED قمز رنگ روی آن روشن می‌شود که بیانگر بودن سنسور است. پس از مدتی نسبتاً طولانی، یک LED سبز رنگ شروع به چشمک زدن می‌کند که بیانگر دریافت مختصات است. اگر این زمان از 5 دقیقه بیشتر شد باید کار کرد آن در هوای آزاد یا نزدیک پنجره بررسی شود. پس از شروع چشمک زدن، با اجرای دستور `sudo cat /dev/ttyAMA0` اطلاعات مختصات دریافت شده توسط سنسور مطابق تصویر زیر نمایش داده می‌شود:

```
mmd@raspberrypi:/ $ sudo cat /dev/ttyAMA0
$GPRMC,,V,,,,,,,,,N*53
$GPVTG,,,,,,N*30
$GPGGA,,,0,00,99.99,,,,,*48
$GPGSA,A,1,,,,,,,,,,99.99,99.99,99.99*30
$GPGLL,,,,V,N*64
$GPRMC,,V,,,,,,,,,N*53
$GPVTG,,,,,,N*30
$GPGGA,,,0,00,99.99,,,,,*48
$GPGSA,A,1,,,,,,,,,,99.99,99.99,99.99*30
$GPGLL,,,,V,N*64
$GPRMC,,V,,,,,,,,,N*53
$GPVTG,,,,,,N*30
$GPGGA,,,0,00,99.99,,,,,*48
$GPGSA,A,1,,,,,,,,,,99.99,99.99,99.99*30
$GPGLL,,,,V,N*64
$GPRMC,,V,,,,,,,,,N*53
$GPVTG,,,,,,N*30
```

تصویر 11 مختصات خام دریافتی از سنسور GPS

مطابق بخش قبل، برای استفاده از مختصات دریافتی سنسور باید از کد پایتون کمک بگیریم، اما پیش از آن نیاز به چند تنظیم دیگر داریم، چون رزبری پای به طور پیش فرض از پورت سریال برای کنسول لاجین استفاده می‌کند و ما به این پورت برای دریافت اطلاعات موقعیت مکانی نیاز داریم؛ پس باید کنسول لاجین را غیرفعال کنیم. در رزبری پای دو پورت سریال `serial0` و `serial1` وجود دارد که `serial0` به `GPIO 14` و `GPIO 15` متصل است. برای مشاهده پورت‌های متصل به `serial0` از دستور `ls -l /dev` استفاده می‌کنیم.

برای غیرفعال کردن کنسول از دستورات زیر استفاده می‌کنیم:

```
sudo systemctl stop serial-getty@ttyAMA0.service
sudo systemctl disable serial-getty@ttyAMA0.service
```

نهایتاً به کمک کد پایتون در فایل `gps.py` می‌توانیم اطلاعات موقعیت مکانی را خوانده و در بخش‌های بعدی از آن استفاده کنیم. اطلاعات ذکر شده در این بخش عمدتاً از [این لینک](#) گرفته شده است. سنسور Neo-7m-C اطلاعات زیادی در خود دارد و یک نمونه از خروجی آن در تصویر زیر قابل مشاهده است، که البته به علت دوری از هوای آزاد خروجی دقیقی نیست:

```
mmd@raspberrypi:~/Desktop/project/hw_project-group-3/Code $ python gps.py
$GPRMC,105506.00,V,,,,,,181222,,,N*70
{'azimuth': -0.5518667376714196, 'altitude': 1.0859745784966284}
Latitude=0.0and Longitude=0.0
$GPRMC,105507.00,V,,,,,,181222,,,N*71
{'azimuth': -0.5517680835033895, 'altitude': 1.0860065621303725}
Latitude=0.0and Longitude=0.0
$GPRMC,105508.00,V,,,,,,181222,,,N*7E
{'azimuth': -0.551650420273802, 'altitude': 1.0860446960081322}
Latitude=0.0and Longitude=0.0
$GPRMC,105509.00,V,,,,,,181222,,,N*7F
{'azimuth': -0.5515328416997615, 'altitude': 1.0860827888870161}
Latitude=0.0and Longitude=0.0
$GPRMC,105510.00,V,,,,,,181222,,,N*77
{'azimuth': -0.5514152910748177, 'altitude': 1.0861208591628513}
Latitude=0.0and Longitude=0.0
$GPRMC,105511.00,V,,,,,,181222,,,N*76
{'azimuth': -0.5512976548651984, 'altitude': 1.0861589435997345}
Latitude=0.0and Longitude=0.0
$GPRMC,105512.00,V,,,,,,181222,,,N*75
{'azimuth': -0.551161113894121, 'altitude': 1.0862031314075655}
Latitude=0.0and Longitude=0.0
$GPRMC,105513.00,V,,,,,,181222,,,N*74
{'azimuth': -0.5510623858246105, 'altitude': 1.086235070714714}
Latitude=0.0and Longitude=0.0
$GPRMC,105514.00,V,,,,,,181222,,,N*73
```

تصویر 12 خروجی سنسور و کد GPS

این اطلاعات به ترتیب عبارتند از:

نمايش شروع خط شامل اطلاعات	زمان	Navigation receiver status	تاریخ	North/South	اعتبارسنجی داده با XOR بین کاراکترهای * و \$
\$GPRMC	HHMMSS.SSS	V	DDMMYY	N	checksum

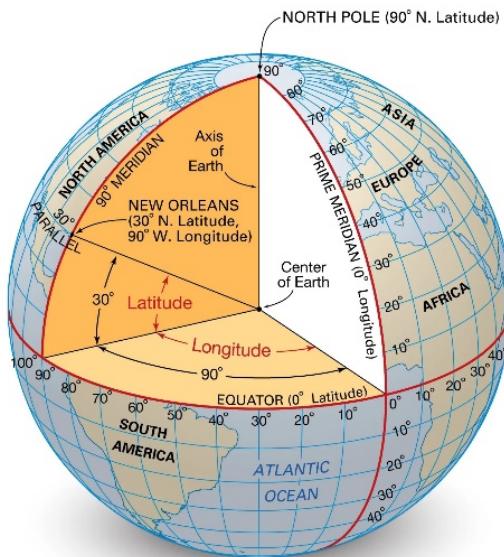
و همچنین `.longitude` و `latitude`

در ادامه با اجرای کد دریافت مکان خورشید، محل آن به کمک `altitude` و `azimuth` داده می‌شود. تعاریف این مفاهیم عبارتند از:

Longitude: طول جغرافیایی. بصورت زاویه‌ای بین منفی 180 تا 180 بیان می‌شود و فاصله یک نقطه روی زمین تا نصف‌النهار مبدأ را نشان می‌دهد.

Latitude: عرض جغرافیایی. بصورت زاویه‌ای بین منفی 90 تا 90 بیان می‌شود و فاصله یک نقطه روی زمین تا خط استوا را نشان می‌دهد.

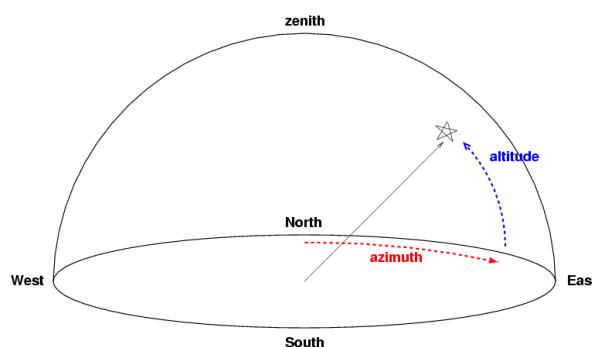
به کمک دو مقدار فوق، مختصات یک نقطه روی سطح زمین بدست می‌آید.



© Encyclopædia Britannica, Inc.

تصویر 13 نمایش عرض و طول جغرافیایی

Azimuth: سمت. فاصله‌ی زاویه‌ای ستاره بر روی افق از شمال، که با حرکت در جهت شرق از ۰ تا ۳۶۰ درجه افزایش می‌یابد.
Altitude: ارتفاع. فاصله‌ی زاویه‌ای برحسب درجه از افق که از ۰ تا ۹۰ درجه تغییر می‌کند.
به کمک دو مقدار فوق، محل یک نقطه در فضا بدست می‌آید.



تصویر 14 نمایش دستگاه مختصات افقی

همانطور که در گزارش‌های پیشین گفته شد برای تشخیص جهت حرکت خودرو به مختصات آن در دو زمان نیاز است. برای این کار، پس از دریافت مختصات در یک لحظه، پس از ۵ ثانیه مجددًا مختصات جدیدی دریافت می‌گردد تا به کمک این دو مقدار جهت حرکت خودرو تعیین گردد. مقدار زمان ۵ ثانیه به این علت انتخاب شد که هم نیاز است خودرو در این مدت زمان مسافت قابل توجهی حرکت کرده باشد، و هم نباید آنقدر زیاد باشد که در این بین خودرو تغییر جهت داده باشد. همچنین اگر به هر علته، از جمله ضعیف بودن سیگنال GPS، داده‌ی دوم پس از ۲۰ ثانیه از سنسور گرفته نشد، با هشدار Timeout مواجه می‌شویم و مجددًا تلاش به دریافت این دو مختصات می‌کنیم.

خروجی این سنسور در هوای آزاد در دانشگاه به شکل زیر است:

```
mmd@raspberrypi:~/Desktop/project/hw_project-group-3/Code $ nano gps.py
mmd@raspberrypi:~/Desktop/project/hw_project-group-3/Code $ python gps.py
$GPGRMC,111217,00,A,3542.09769,N,05121.16434,E,1.789,,261222,,,A*73 $GPGRMC,111222.00,A,3542.10127,N,05121.16710,E,3.129,35.02,261222,,,A*54
35.701628166666666 51.352785 35.70168783333333 51.352785 0.0 0.0
Latitude=35.70168783333333and Longitude=51.352785
Azimuth=-3.002263500535479,Sun Angle=3.002263500535479
$GPGRMC,111223.00,A,3542.10236,N,05121.16800,E,3.098,27.04,261222,,,A*56 $GPGRMC,111229.00,A,3542.10737,N,05121.17271,E,3.582,27.90,261222,,,A*56
35.701706 51.3528785 35.7017895 51.3528785 0.0 0.0
Latitude=35.7017895and Longitude=51.3528785
Azimuth=-3.000094510698154,Sun Angle=0.000094510698154
$GPGRMC,111230.00,A,3542.10836,N,05121.17317,E,2.541,22.52,261222,,,A*54 $GPGRMC,111236.00,A,3542.11435,N,05121.17364,E,2.751,9.40,261222,,,A*61
35.701806 51.352894 35.701905833333335 51.352894 0.0 0.0
Latitude=35.70190583333335and Longitude=51.352894
Azimuth=-2.997937916586526,Sun Angle=-2.997937916586526
$GPGRMC,111237.00,A,3542.11532,E,2.398,8.56,261222,,,A*60 $GPGRMC,111242.00,A,3542.12017,N,05121.17084,E,2.505,317.13,261222,,,A*61
35.701922 51.35284733333334 35.70200283333333 51.3528473333334 0.0 0.0
Latitude=35.70200283333333and Longitude=51.3528473333334
Azimuth=-2.9960879922471624,Sun Angle=-2.9960879922471624
$GPGRMC,111243.00,A,3542.12058,N,05121.16994,E,2.593,310.12,261222,,,A*68 $GPGRMC,111249.00,A,3542.12220,N,05121.16553,E,2.267,,261222,,,A*78
35.70200966666667 51.35275883333335 35.702036666666665 51.35275883333335 0.0 0.0
Latitude=35.70203666666665and Longitude=51.35275883333335
Azimuth=-2.99393876542296,Sun Angle=-2.99393876542296
$GPGRMC,111250.00,A,3542.12216,N,05121.16470,E,2.725,,261222,,,A*76 $GPGRMC,111255.00,A,3542.12278,N,05121.16098,E,2.386,271.32,261222,,,A*6F
35.702035 51.352683 35.702046333333335 51.352683 0.0 0.0
Latitude=35.702046333333335and Longitude=51.352683
Azimuth=-2.9920934211147117,Sun Angle=-2.9920934211147117
$GPGRMC,111256.00,A,3542.12281,N,05121.16027,E,2.274,273.98,261222,,,A*60 $GPGRMC,111302.00,A,3542.12365,N,05121.15535,E,2.712,272.93,261222,,,A*61
35.70206833333334 51.35258916666667 35.70206833333334 51.35258916666667 0.0 0.0
Latitude=35.70206833333334and Longitude=51.35258916666667
Azimuth=-2.989945649140728,Sun Angle=-2.989945649140728
$GPGRMC,111303.00,A,3542.12377,N,05121.15455,E,2.288,277.81,261222,,,A*6C $GPGRMC,111309.00,A,3542.12373,N,05121.14949,E,2.354,,261222,,,A*7E
35.702062833333336 51.3524915 35.702062166666664 51.3524915 -0.0 -0.0
Latitude=35.70206216666664and Longitude=51.3524915
Azimuth=-2.987799180812416,Sun Angle=-2.987799180812416
Azimuth=-2.987799180812416
```

تصویر 15 خروجی سنسور GPS در دانشگاه

```
Azimuth=-2.963629385627599,Sun Angle=2.963629385627599
$GPGRMC,111429.00,A,3542.13705,N,05121.09754,E,0.136,,261222,,,A*79 $GPGRMC,111434.00,A,3542.13744,N,05121.09730,E,0.641,,261222,,,A*79
35.70228416666667 51.3516216666667 35.70229066666667 51.3516216666667 0.0 0.0
Latitude=35.70229066666667and Longitude=51.3516216666667
Azimuth=-2.961795034894503,Sun Angle=-2.961795034894503
$GPGRMC,111435.00,A,3542.13746,N,05121.09739,E,0.044,,261222,,,A*70 $GPGRMC,111441.00,A,3542.13698,N,05121.09702,E,0.174,,261222,,,A*7B
35.702291 51.351617 35.702283 51.351617 -0.0 -0.0
Latitude=35.702283and Longitude=51.351617
Azimuth=-2.959656317221534,Sun Angle=-2.959656317221534
$GPGRMC,111442.00,A,3542.13746,N,05121.09694,E,0.109,,261222,,,A*75 $GPGRMC,111448.00,A,3542.13670,N,05121.09548,E,0.341,,261222,,,A*7C
35.70228183333333 51.35159133333333 35.70227833333333 51.35159133333333 -0.0 -0.0
Latitude=35.70227833333333and Longitude=51.35159133333333
Azimuth=-2.95752097427525,Sun Angle=-2.95752097427525
$GPGRMC,111449.00,A,3542.13659,N,05121.09540,E,0.654,,261222,,,A*7F $GPGRMC,111454.00,A,3542.13841,N,05121.09466,E,2.552,341.91,261222,,,A*66
35.702276499999964 51.35157766666664 35.7023068333333 51.35157766666664 0.0 0.0
Latitude=35.70230683333333and Longitude=51.35157766666664
Azimuth=-2.955689490899079,Sun Angle=-2.955689490899079
$GPGRMC,111455.00,A,3542.13901,N,05121.09463,E,2.559,339.99,261222,,,A*6B $GPGRMC,111501.00,A,3542.14318,N,05121.09407,E,2.438,,261222,,,A*7D
35.70231683333333 51.35156783333333 35.70238633333333 51.35156783333333 0.0 0.0
Latitude=35.70238633333333and Longitude=51.35156783333333
Azimuth=-2.953558962019554,Sun Angle=-2.953558962019554
$GPGRMC,111502.00,A,3542.14378,N,05121.09373,E,2.656,348.45,261222,,,A*66 $GPGRMC,111508.00,A,3542.14645,N,05121.09253,E,2.214,,261222,,,A*76
35.70239633333333 51.35157766666664 35.70240833333333 51.35157766666664 0.0 0.0
Latitude=35.70240833333333and Longitude=51.35157766666664
Azimuth=-2.9514295024119547,Sun Angle=-2.9514295024119547
$GPGRMC,111509.00,A,3542.14677,N,05121.09177,E,1.881,347.66,261222,,,A*68 $GPGRMC,111515.00,A,3542.14740,N,05121.08998,E,2.658,302.13,261222,,,A*62
35.70244616666667 51.35149966666667 35.70245666666667 51.35149966666667 0.0 0.0
Latitude=35.702456666666667and Longitude=51.35149966666667
Azimuth=-2.94929921804463,Sun Angle=-2.94929921804463
$GPGRMC,111516.00,A,3542.14746,N,05121.08942,E,2.301,299.06,261222,,,A*6E $GPGRMC,111522.00,A,3542.14355,N,05121.08777,E,4.751,213.14,261222,,,A*61
35.70245766666667 51.35146283333333 35.702395 51.35146283333333 -0.0 -0.0
Latitude=35.702395and Longitude=51.35146283333333
Azimuth=-2.9471709014548595,Sun Angle=-2.9471709014548595
$GPGRMC,111523.00,A,3542.14221,N,05121.08870,E,5.405,,261222,,,A*72 $GPGRMC,111529.00,A,3542.14016,N,05121.09280,E,3.323,,261222,,,A*7F
35.70237016666667 51.351546666666664 35.702333 51.351546666666664 -0.0 -0.0
Latitude=35.702333and Longitude=51.351546666666664
Azimuth=-2.9450373886078425,Sun Angle=-2.9450373886078425
$GPGRMC,111530.00,A,3542.14066,N,05121.09439,E,4.207,,261222,,,A*74 $GPGRMC,111536.00,V,,,,261222,,,N*7B
35.70234343333333 51.35146283333333 35.702395 51.35146283333333 -0.0 -0.0
Latitude=0.0and Longitude=0.0
Azimuth=-1.047448844983487,Sun Angle=-1.047448844983487
*Craceback (most recent call last):
  File "/home/mmd/Desktop/project/hw_project-group-3/Code/gps.py", line 30, in <module>
    newdata = ser.readline()
  File "/usr/lib/python3/dist-packages/serial/serialposix.py", line 547, in read
    ready, _, _ = select.select([self.fd, self.pipe_abort_read_r], [], [], timeout.time_left())
KeyboardInterrupt
```

تصویر 16 خروجی سنسور GPS در دانشگاه 2

با مقایسه این خروجی با مقدار دقیق مختصات آن نقطه، متوجه شدیم خروجی GPS کاملاً صحیح است و فقط با مقداری تاخیر تغییر می کند.

8. تشخیص جهت تابش آفتاب به کمک مختصات

در نهایت، به کمک خروجی گرفته شده در بخش قبل می خواهیم زاویه تابش آفتاب را بیابیم. بدین منظور، علاوه بر اطلاعات مذکور به تاریخ و زمان نیز نیاز داریم. سپس به کمک تابع `get_position` کتابخانه `suncalc` در پایتون می توانیم زاویه تابش آفتاب را بیابیم. این تابع با گرفتن 3 ورودی، به ترتیب تاریخ (در قالب `datetime` پایتون)، که شامل تاریخ و زمان است، و `longitude` (به

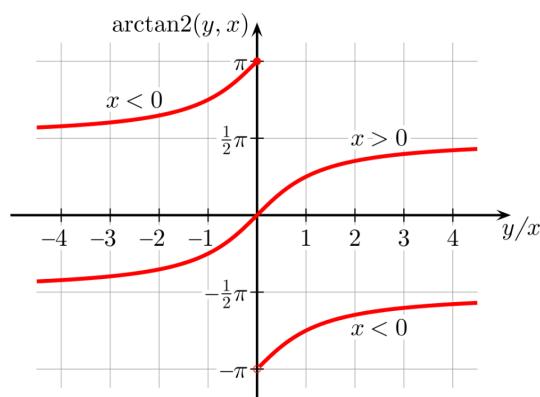
شکل درجه در قالب `float` پایتون) و `latitude` (به شکل درجه در قالب `float` پایتون) خروجی را در قالب یک دیکشنری می‌دهد، که به ترتیب شامل دو `key` می‌باشد: `altitude` و `azimuth`. محاسبات این کتابخانه بطور خلاصه به کمک ورودی‌های گفته شده، حرکت زمین به دور خورشید که با سرعت ثابتی نیست و زاویه بین محور چرخش زمین و صفحه مدار آن، صورت می‌گیرد. قطعه کد لازم برای این بخش در فایل `gps.py` قابل مشاهده است. برای پیاده‌سازی این بخش از [این لینک](#) کمک گرفتیم.

9. محاسبه جهت حرکت جسم (خودرو)

برای تشخیص محلی از خودرو که در برابر خورشید است نیاز به دانستن جهت حرکت آن داریم. اگر فرض کنیم یک خودروی در حال حرکت در یک لحظه در مختصات $\begin{bmatrix} \text{lng1} \\ \text{lat1} \end{bmatrix}$ و در لحظه بعد پس از حرکت بدون تغییر جهت در مختصات $\begin{bmatrix} \text{lng2} \\ \text{lat2} \end{bmatrix}$ قرار دارد، مسافتی که در جهت شمال-جنوب طی کرده است از فرمول $2R \sin\left(\frac{\text{lat2}-\text{lat1}}{2}\right)$ و مسافتی که در جهت شرق-غرب طی کرده است از فرمول $2R \sin\left(\frac{(\text{lng2}-\text{lng1})}{2}\right)$ بدست می‌آیند و R شاعر کره زمین است. در نتیجه، جهت حرکت خودرو نسبت به محور شمال-جنوب از فرمول زیر بدست خواهد آمد:

$$\tan^{-1} \frac{\sin\left(\frac{(\text{lng2}-\text{lng1})}{2}\right)}{\sin\left(\frac{(\text{lat2}-\text{lat1})}{2}\right)}$$

در نهایت و به کمک تشخیص جهت قرارگیری خودرو و محل خورشید در آسمان، می‌توان محلی از خودرو که در برابر نور خورشید قرار دارد را شناسایی کرد. لازم به ذکر است که برای محاسبه معکوس تانژانت ابتدا از تابع `arctan` در پایتون استفاده می‌کردیم که به دلیل محدود بودن خروجی آن بین منفی 90 تا مثبت 90 درجه، این تابع را به `arctan2` تغییر دادیم که خروجی آن بین منفی 180 تا مثبت 180 درجه است.



تصویر 17 نمودار تابع `atan2`

نهایتاً به کمک قطعه کد زیر تشخیص می‌دهیم کدام بخش خودرو در معرض نور خورشید است. بخش‌های 0 تا 3 خودرو مطابق بخش‌ها گفته شده در بخش [گرفتن دما](#) هستند.

```

if sun_angle < np.pi / 2:
    exposed_sensor = 2
elif sun_angle < np.pi:
    exposed_sensor = 3
elif sun_angle < 3 * np.pi / 2:
    exposed_sensor = 1
else:
    exposed_sensor = 0

```

تصویر 18 تشخیص بخش در معرض آفتاب

10. صحتسنجی سنسور GPS

برای بررسی کارکرد این سنسور به شرایط خاصی نیاز داشتیم. به دلیل ضعیف بودن سیگنال‌های GPS نمی‌توانستیم در محیط بسته آن را تست کنیم و ناچار بودیم در محیط آزاد مانند حیاط دانشگاه به تست آن بپردازیم. هم‌چنین برای بررسی تغییر مختصات GPS باید حرکت می‌کردیم، اما سرعت حرکت با پیاده‌روی بسیار پایین بود و تغییر چشمگیری در مختصات دیده نمی‌شد. به همین دلایل، پس از اطمینان از خروجی صحیح این سنسور، برای تسهیل تست سایر بخش‌های پروژه از یکتابع به نام `test` برای generate کردن مختصات استفاده کردیم. در این تابع مختصات یک نقطه ابتدایی (مثلًاً نقطه خاصی از دانشگاه) در قالب `longitude` و `latitude`، جهتی که می‌خواهیم حرکت را در آن تست کنیم به شکل `N, E, S, W`، و یک مقدار ثابت گام برای تغییر مختصات، بطور پیش‌فرض `0.001`، به عنوان ورودی داده می‌شود و با اضافه یا کم کردن گام از طول و عرض جغرافیایی بر حسب جهت گفته شده، مختصات‌های جدید generate می‌گردند. با این کار می‌توانیم برای صحتسنجی جهت تابش آفتاب در خودرو و روشن کردن LED مربوطه، به راحتی داده تولید کنیم. به عنوان مثال، برای بررسی کد بخش جهت تابش آفتاب بر اساس جهت حرکت خودرو از این داده‌سازی استفاده کردیم و خروجی آن در تصاویر زیر قابل مشاهده است:

```

mmd@raspberrypi:~/Desktop/project/hw_project-group-3/Code $ nano gps.py
mmd@raspberrypi:~/Desktop/project/hw_project-group-3/Code $ python gps.py
35.703129 51.351671 35.70412899999995 51.351671 0.0 0.0009999983331011
Movement Directio: 0.0 ,Sun Azimuth: 1.22794032588571 ,Sun Angle: 1.22794032588571
Sensor 2 is close to the Sun.
35.70412899999995 51.351671 35.70512899999999 51.351671 0.0 0.0009999983331011
Movement Directio: 0.0 ,Sun Azimuth: 1.2281352125737597 ,Sun Angle: 1.2281352125737597
Sensor 2 is close to the Sun.
^CTraceback (most recent call last):
  File "/home/mmd/Desktop/project/hw_project-group-3/Code/gps.py", line 107, in <module>
    manual_main()
  File "/home/mmd/Desktop/project/hw_project-group-3/Code/gps.py", line 102, in manual_main
    time.sleep(5)
KeyboardInterrupt

```

تصویر 19 تشخیص محل تابش آفتاب در صورت حرکت به سمت شمال

```
mmd@raspberrypi:~/Desktop/project/hw_project-group-3/Code $ nano gps.py
mmd@raspberrypi:~/Desktop/project/hw_project-group-3/Code $ python gps.py
35.703129 51.351671 35.703129 51.352671 0.00099999833331011 0.0
Movement Directio: 1.5707963267948966 ,Sun Azimuth: 1.231415676012904 ,Sun Angle: 5.943804656397594
Sensor 0 is close to the Sun.
35.703129 51.352671 35.703129 51.353671 0.00099999833331011 0.0
Movement Directio: 1.5707963267948966 ,Sun Azimuth: 1.2316159058045542 ,Sun Angle: 5.944004886189244
Sensor 0 is close to the Sun.
^CTraceback (most recent call last):
  File "/home/mmd/Desktop/project/hw_project-group-3/Code/gps.py", line 107, in <module>
    manual_main()
  File "/home/mmd/Desktop/project/hw_project-group-3/Code/gps.py", line 102, in manual_main
    time.sleep(5)
KeyboardInterrupt
```

تصویر 20 تشخیص محل تابش آفتاب در صورت حرکت به سمت شرق

```
mmd@raspberrypi:~/Desktop/project/hw_project-group-3/Code $ nano gps.py
mmd@raspberrypi:~/Desktop/project/hw_project-group-3/Code $ python gps.py
35.703129 51.351671 35.702129 51.351671 0.0 -0.00099999833331011
Movement Directio: 3.141592653589793 ,Sun Azimuth: 1.2324272859094225 ,Sun Angle: 4.374019939499216
Sensor 1 is close to the Sun.
35.702129 51.351671 35.701129 51.351671 0.0 -0.00099999833331011
Movement Directio: 3.141592653589793 ,Sun Azimuth: 1.2326146470065626 ,Sun Angle: 4.374207300596356
Sensor 1 is close to the Sun.
^CTraceback (most recent call last):
  File "/home/mmd/Desktop/project/hw_project-group-3/Code/gps.py", line 107, in <module>
    manual_main()
  File "/home/mmd/Desktop/project/hw_project-group-3/Code/gps.py", line 102, in manual_main
    time.sleep(5)
KeyboardInterrupt
```

تصویر 21 تشخیص محل تابش آفتاب در صورت حرکت به سمت جنوب

```
mmd@raspberrypi:~/Desktop/project/hw_project-group-3/Code $ nano gps.py
mmd@raspberrypi:~/Desktop/project/hw_project-group-3/Code $ python gps.py
35.703129 51.351671 35.703129 51.350671000000006 -0.00099999833331011 0.0
Movement Directio: -1.5707963267948966 ,Sun Azimuth: 1.233371821130741 ,Sun Angle: 2.8041681479256377
Sensor 3 is close to the Sun.
35.703129 51.350671000000006 35.703129 51.34967100000001 -0.00099999833331011 0.0
Movement Directio: -1.5707963267948966 ,Sun Azimuth: 1.2335535684602756 ,Sun Angle: 2.8043498952551724
Sensor 3 is close to the Sun.
^CTraceback (most recent call last):
  File "/home/mmd/Desktop/project/hw_project-group-3/Code/gps.py", line 107, in <module>
    manual_main()
  File "/home/mmd/Desktop/project/hw_project-group-3/Code/gps.py", line 102, in manual_main
    time.sleep(5)
KeyboardInterrupt
```

تصویر 22 تشخیص محل تابش آفتاب در صورت حرکت به سمت غرب

11. روشن و خاموش کردن خنک کننده‌ها

همانطور که گفته شد، به جای خنک کننده از LED استفاده می‌کنیم. برای 4 بخش خودرو به 4 LED نیاز داریم و هر یک باید با یک سنسور دما هماهنگ باشد، بطوریکه پس از تشخیص جهت تابش آفتاب، با اندازه‌گیری دمای 4 بخش خودرو، در صورت گرم بودن هر بخش چراغ LED آن روشن شود. برای این کار 4 LED را به 4 پین GPIO مجزا از رزبری پای متصل می‌کنیم. با این کار به هر دماسنج یک خنک کننده تخصیص می‌دهیم. سپس در هر ثانیه دمای یک سنسور را دریافت می‌کنیم و در نتیجه، پس از 4 ثانیه 4 دمای جدید سنسورها گرفته می‌شود. اکنون میانگین آنها را محاسبه کرده و بخش‌هایی از خودرو که دمایی 1 درجه بیش از میانگین دارند را به عنوان بخش‌هایی که باید خنک شوند در نظر می‌گیریم و چراغ بخش مربوطه را روشن می‌کنیم. LED‌های بخش‌های 0، 1، 2 و 3 خودرو به ترتیب به GPIO‌های 5، 6، 13 و 19 رزبری پای متصلند. در پیاده‌سازی این بخش به چند تابع ساده محاسبه میانگین و ... نیاز داریم که در فایل `math_functions.py` قرار دارد. کار کرد این بخش از کد تست شده و در تصاویر زیر قابل مشاهده است:

```
mmd@raspberrypi:~/Desktop/project/hw_project-group-3/Code $ python temp_sensor.py
Sensor 1 => 23.625 (C) - 74.525 (F)
Sensor 2 => 23.625 (C) - 74.525 (F)
Sensor 3 => 23.812 (C) - 74.86160000000001 (F)
Sensor 4 => 23.562 (C) - 74.4116 (F)
LED 0 off
LED 1 off
LED 2 off
LED 3 off
Sensor 1 => 23.562 (C) - 74.4116 (F)
Sensor 2 => 23.625 (C) - 74.525 (F)
Sensor 3 => 23.75 (C) - 74.75 (F)
Sensor 4 => 23.562 (C) - 74.4116 (F)
LED 0 off
LED 1 off
LED 2 off
LED 3 off
Sensor 1 => 28.062 (C) - 82.5116 (F)
Sensor 2 => 23.625 (C) - 74.525 (F)
Sensor 3 => 23.812 (C) - 74.86160000000001 (F)
Sensor 4 => 23.625 (C) - 74.525 (F)
LED 0 on
LED 1 off
LED 2 off
LED 3 off
Sensor 1 => 29.25 (C) - 84.65 (F)
Sensor 2 => 23.75 (C) - 74.75 (F)
Sensor 3 => 23.937 (C) - 75.0866 (F)
Sensor 4 => 23.687 (C) - 74.6366 (F)
LED 0 on
LED 1 off
LED 2 off
```

تصویر 23 صحبت سنجی تطابق سنسور دما و LED

```
LED 2 off
LED 3 off
Sensor 1 => 29.625 (C) - 85.325 (F)
Sensor 2 => 23.75 (C) - 74.75 (F)
Sensor 3 => 24.0 (C) - 75.2 (F)
Sensor 4 => 23.687 (C) - 74.6366 (F)
LED 0 on
LED 1 off
LED 2 off
LED 3 off
Sensor 1 => 29.812 (C) - 85.66159999999999 (F)
Sensor 2 => 23.75 (C) - 74.75 (F)
Sensor 3 => 24.0 (C) - 75.2 (F)
Sensor 4 => 23.687 (C) - 74.6366 (F)
LED 0 on
LED 1 off
LED 2 off
LED 3 off
Sensor 1 => 29.937 (C) - 85.8866 (F)
Sensor 2 => 23.75 (C) - 74.75 (F)
Sensor 3 => 24.0 (C) - 75.2 (F)
Sensor 4 => 23.75 (C) - 74.75 (F)
LED 0 on
LED 1 off
LED 2 off
LED 3 off
Sensor 1 => 30.0 (C) - 86.0 (F)
Sensor 2 => 28.312 (C) - 82.9616 (F)
Sensor 3 => 24.0 (C) - 75.2 (F)
Sensor 4 => 23.75 (C) - 74.75 (F)
LED 0 on
LED 1 on
LED 2 off
LED 3 off
Sensor 1 => 30.0 (C) - 86.0 (F)
Sensor 2 => 28.812 (C) - 83.8616 (F)
Sensor 3 => 24.062 (C) - 75.3116 (F)
Sensor 4 => 23.75 (C) - 74.75 (F)
LED 0 on
LED 1 on
LED 2 off
LED 3 off
Sensor 1 => 28.625 (C) - 83.525 (F)
Sensor 2 => 27.0 (C) - 80.6 (F)
Sensor 3 => 24.062 (C) - 75.3116 (F)
Sensor 4 => 23.75 (C) - 74.75 (F)
LED 0 on
LED 1 on
LED 2 off
```

تصویر 24 صحبت سنجی تطابق سنسور دما و 2 LED

```

LED 2 off
LED 3 off
Sensor 1 => 27.125 (C) - 80.825 (F)
Sensor 2 => 25.937 (C) - 78.6866 (F)
Sensor 3 => 24.062 (C) - 75.3116 (F)
Sensor 4 => 23.75 (C) - 74.75 (F)
LED 0 on
LED 1 off
LED 2 off
LED 3 off
Sensor 1 => 26.187 (C) - 79.1366 (F)
Sensor 2 => 25.25 (C) - 77.45 (F)
Sensor 3 => 24.0 (C) - 75.2 (F)
Sensor 4 => 23.687 (C) - 74.6366 (F)
LED 0 on
LED 1 off
LED 2 off
LED 3 off
Sensor 1 => 25.625 (C) - 78.125 (F)
Sensor 2 => 24.687 (C) - 76.4366 (F)
Sensor 3 => 23.937 (C) - 75.0866 (F)
Sensor 4 => 23.625 (C) - 74.525 (F)
LED 0 on
LED 1 off
LED 2 off
LED 3 off
Sensor 1 => 25.187 (C) - 77.3366 (F)
Sensor 2 => 24.5 (C) - 76.1 (F)
Sensor 3 => 23.937 (C) - 75.0866 (F)
Sensor 4 => 23.687 (C) - 74.6366 (F)
LED 0 off
LED 1 off
LED 2 off
LED 3 off
Sensor 1 => 24.937 (C) - 76.8866 (F)
Sensor 2 => 24.375 (C) - 75.875 (F)
Sensor 3 => 23.937 (C) - 75.0866 (F)
Sensor 4 => 23.687 (C) - 74.6366 (F)
LED 0 off
LED 1 off
LED 2 off
LED 3 off
Sensor 1 => 24.687 (C) - 76.4366 (F)
Sensor 2 => 24.25 (C) - 75.65 (F)
Sensor 3 => 23.937 (C) - 75.0866 (F)
Sensor 4 => 23.625 (C) - 74.525 (F)
LED 0 off
LED 1 off
LED 2 off

```

تصویر 25 صحبت‌سنگی تطبیق سنسور دما و LED

12. بخش اصلی پروژه

نهایتاً در قطعه کد زیر در فایل `temp_sensor.py` به اجرای پروژه می‌پردازیم. طبق این کد، در بخش‌هایی از خودرو که دمایی بیش از یک درجه بالای میانگین دمای همه بخش‌های خودرو دارند، در صورت در معرض نور آفتاب بودن، خنک‌کننده روشن می‌شود؛ و در غیر اینصورت خاموش می‌گردد.

```

for sensor_id in hot_sensors_ids:
    if sensor_id == exposed_sensor:
        turn_on_led(led_temp_sensor[sensor_id])
    else:
        turn_off_led(led_temp_sensor[sensor_id])

```

تصویر 26 کد ترکیب شرط دما و تابش آفتاب

```

mmd@raspberrypi:~/Desktop/project/hw_project-group-3/Code$ python gps.py
35.703129 51.351671 35.703129 51.352671 0.0009999983331011 0.0
Movement Direction: 1.5707963267948966 ,Sun Azimuth: -0.893199518930662 ,Sun Angle: 3.8191894619916233
Sensor 1 is close to the Sun.
Sensor 0 => 23.375 (C) - 74.075 (F)
Sensor 1 => 23.025 (C) - 74.525 (F)
Sensor 2 => 26.0 (C) - 78.8 (F)
Sensor 3 => 23.937 (C) - 75.0866 (F)
LED 2 off
LED 0 off
LED 2 off
LED 3 off
35.703129 51.352671 35.703129 51.353671 0.0009999983331011 0.0
Movement Direction: 1.5707963267948966 ,Sun Azimuth: -0.8928022786913962 ,Sun Angle: 3.8195867016932934
Sensor 1 is close to the Sun.
Sensor 0 => 23.375 (C) - 74.075 (F)
Sensor 1 => 24.312 (C) - 75.7616 (F)
Sensor 2 => 25.625 (C) - 78.125 (F)
Sensor 3 => 23.937 (C) - 75.0866 (F)
LED 2 off
LED 0 off
LED 1 off
LED 3 off
35.703129 51.353671 35.703129 51.35467099999996 0.0009999983331011 0.0
Movement Direction: 1.5707963267948966 ,Sun Azimuth: -0.8924007498123694 ,Sun Angle: 3.8199882305723203
Sensor 1 is close to the Sun.
Sensor 0 => 23.375 (C) - 74.075 (F)
Sensor 1 => 27.75 (C) - 81.95 (F)
Sensor 2 => 25.375 (C) - 77.675 (F)
Sensor 3 => 23.937 (C) - 75.0866 (F)
LED 1 on
LED 0 off
LED 2 off
LED 3 off
35.703129 51.35467099999996 35.703129 51.35567099999994 0.0009999983331011 0.0
Movement Direction: 1.5707963267948966 ,Sun Azimuth: -0.8920070276487032 ,Sun Angle: 3.8203819527359864
Sensor 1 is close to the Sun.
Sensor 0 => 23.437 (C) - 74.1866 (F)

```

تصویر 27 نتیجه نهایی اجرای کد

```

mmd@raspberrypi:~/Desktop/project/hw_project-group-3/Code$ python gps.py
35.703129 51.351671 35.703129 51.352671 0.0009999983331011 0.0
Movement Direction: 1.5707963267948966 ,Sun Azimuth: -0.8888704147979894 ,Sun Angle: 3.8235185655867947
Sensor 1 is close to the Sun.
Sensor 0 => 23.562 (C) - 74.4116 (F)
Sensor 1 => 24.562 (C) - 76.2116 (F)
Sensor 2 => 24.125 (C) - 75.425 (F)
Sensor 3 => 24.862 (C) - 75.3118 (F)
LED 0 off
LED 1 off
LED 2 off
LED 3 off
35.703129 51.352671 35.703129 51.353671 0.0009999983331011 0.0
Movement Direction: 1.5707963267948966 ,Sun Azimuth: -0.8888731403490301 ,Sun Angle: 3.8239158400356597
Sensor 1 is close to the Sun.
Sensor 0 => 23.562 (C) - 74.4116 (F)
Sensor 1 => 24.5 (C) - 76.1 (F)
Sensor 2 => 24.062 (C) - 75.3116 (F)
Sensor 3 => 26.562 (C) - 79.8116 (F)
LED 3 off
LED 0 off
LED 1 off
LED 2 off
35.703129 51.353671 35.703129 51.35467099999996 0.0009999983331011 0.0
Movement Direction: 1.5707963267948966 ,Sun Azimuth: -0.8888741964563484 ,Sun Angle: 3.824314783934341
Sensor 1 is close to the Sun.
Sensor 0 => 23.625 (C) - 74.525 (F)
Sensor 1 => 28.375 (C) - 83.075 (F)
Sensor 2 => 24.0 (C) - 75.2 (F)
Sensor 3 => 28.562 (C) - 83.41159999999999 (F)
LED 0 on
LED 3 off
LED 0 off
LED 2 off
35.703129 51.35467099999996 35.703129 51.35567099999994 0.0009999983331011 0.0
Movement Direction: 1.5707963267948966 ,Sun Azimuth: -0.8876709325765141 ,Sun Angle: 3.8247180478081755
Sensor 1 is close to the Sun.
Sensor 0 => 23.625 (C) - 74.525 (F)
Sensor 1 => 29.062 (C) - 84.3116 (F)
Sensor 2 => 24.0 (C) - 75.2 (F)
Sensor 3 => 29.062 (C) - 84.3116 (F)
LED 1 on
LED 0 off
LED 3 off
LED 0 off
35.703129 51.35567099999994 35.703129 51.35667099999994 0.0009999983331011 0.0
Movement Direction: 1.5707963267948966 ,Sun Azimuth: -0.887275542863866 ,Sun Angle: 3.825113438098303
Sensor 1 is close to the Sun.

```

تصویر 28 نتیجه نهایی اجرای کد 2

```

Movement Direction: 1.5707963267948966 ,Sun Azimuth: -0.8775706106309099 ,Sun Angle: 3.83481836975378
Sensor 1 is close to the Sun.
Sensor 0 => 23.025 (C) - 74.525 (F)
Sensor 1 => 26.7 (C) - 79.7 (F)
Sensor 2 => 23.5 (C) - 74.3 (F)
Sensor 3 => 24.375 (C) - 75.875 (F)
LED 1 on
LED 0 off
LED 2 off
LED 3 off
35.703129 51.355670999999994 35.703129 51.35667099999999 0.0009999983331011 0.0
Movement Direction: 1.5707963267948966 ,Sun Azimuth: -0.8771635478395572 ,Sun Angle: 3.8352254325451325
Sensor 1 is close to the Sun.
Sensor 0 => 23.025 (C) - 74.525 (F)
Sensor 1 => 25.75 (C) - 78.35 (F)
Sensor 2 => 23.562 (C) - 74.4116 (F)
Sensor 3 => 24.375 (C) - 75.875 (F)
LED 1 on
LED 0 off
LED 2 off
LED 3 off
35.703129 51.35667099999999 35.703129 51.35670999999999 0.0009999983331011 0.0
Movement Direction: 1.5707963267948966 ,Sun Azimuth: -0.8767644300983511 ,Sun Angle: 3.8356245502863384
Sensor 1 is close to the Sun.
Sensor 0 => 23.025 (C) - 74.525 (F)
Sensor 1 => 25.25 (C) - 77.45 (F)
Sensor 2 => 23.562 (C) - 74.4116 (F)
Sensor 3 => 24.375 (C) - 75.875 (F)
LED 1 on
LED 0 off
LED 2 off
LED 3 off
35.703129 51.35670999999999 35.703129 51.35867099999999 0.0009999983331011 0.0
Movement Direction: 1.5707963267948966 ,Sun Azimuth: -0.8763651513804178 ,Sun Angle: 3.836023829004272
Sensor 1 is close to the Sun.
Sensor 0 => 23.562 (C) - 74.4116 (F)
Sensor 1 => 25.0 (C) - 77.0 (F)
Sensor 2 => 23.025 (C) - 74.525 (F)
Sensor 3 => 24.375 (C) - 75.875 (F)
LED 0 off
LED 1 off
LED 2 off
LED 3 off
35.703129 51.35867099999999 35.703129 51.359670999999985 0.0009999983331011 0.0
Movement Direction: 1.5707963267948966 ,Sun Azimuth: -0.8759576487935229 ,Sun Angle: 3.8364313315911667
Sensor 1 is close to the Sun
^CTraceback (most recent call last):
  File "/home/mmd/Desktop/project/hw_project-group-3/Code/gps.py", line 120, in <module>

```

تصویر 29 نتیجه نهایی اجرای کد 3

طبق نتایج فوق می‌بینیم که در تصویر 28 ابتدا تمامی سنسورها نسبتاً دمای یکسانی دارند. سپس سنسور بخش 3 خودرو با اینکه دمای بالایی را دریافت می‌کند، به علت در معرض تابش آفتاب نبودن، خنک‌کننده‌ی خود را روشن نمی‌کند. در همین تصویر مشاهده می‌شود که پس از مدتی دمای بخش 1 بالا رود و چون در معرض خورشید نیز قرار دارد، چراغ آن روشن می‌گردد. پس از مدتی، با پایین آمدن دمای این بخش، در انتهای تصویر 29 مشاهده می‌شود که با وجود در معرض نور خورشید بودن بخش 1، چون دمای پایین‌تری دارد، دیگر احتیاجی به خنک‌کننده نداشته و آن را خاموش می‌کند.

سنسورهای قرار داده شده در ۴ قسمت ماشین به صورت زیر است:

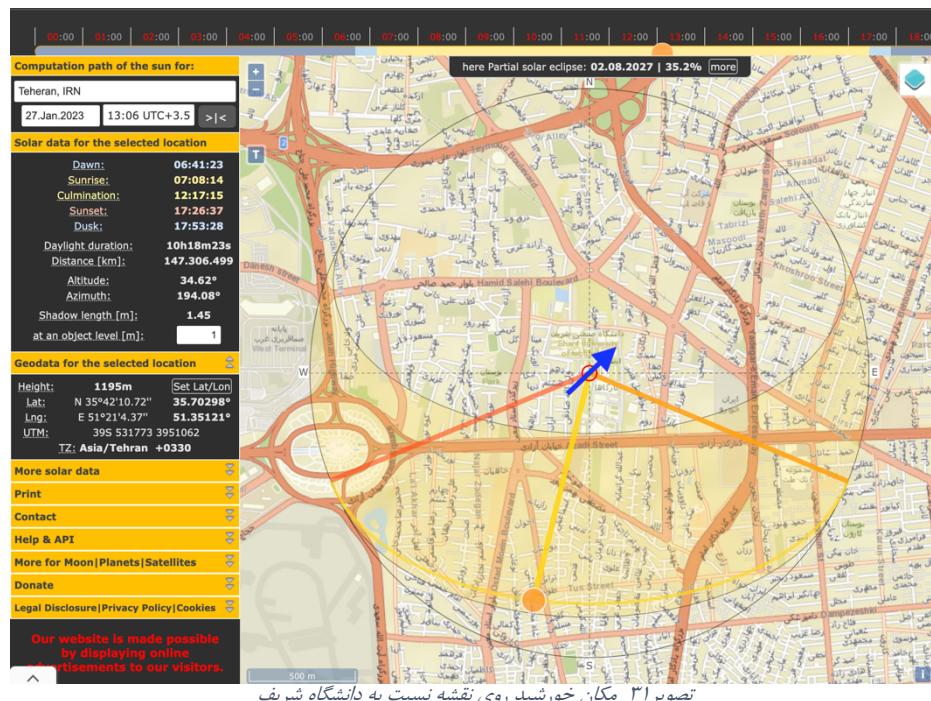


۱۳. تست‌ها

برای تست دستی محصول، ۳ لوكیشن متفاوت را انتخاب کردیم و زاویه‌ی تابش آفتاب در یک ساعت مشخص بدست آوردیم و با توجه به این ورودی‌ها، شماره‌ی سنسوری را که باید روشن شود، به عنوان خروجی نشان می‌دهیم.

تصویر ۳۰ نحوه‌ی اختصاص دادن سنسورها

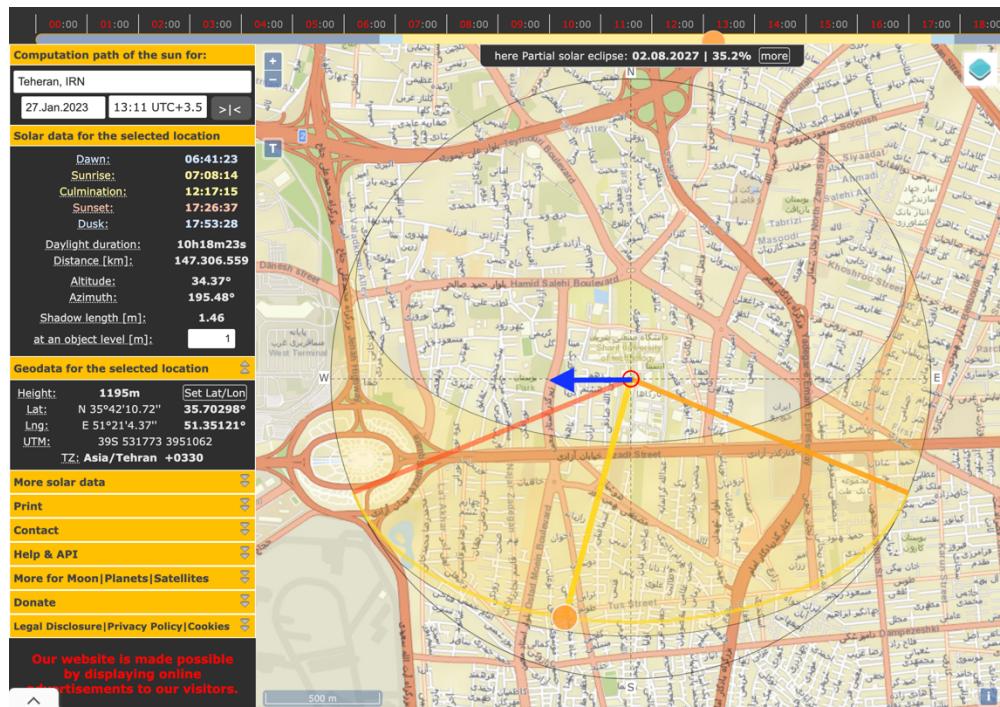
۱۳.۱ لوکیشن: دانشگاه شریف، تهران، ایران (جهت حرکت: شمال شرقی)



```
mmd@raspberrypi:~/Desktop/project/hw_project-group-3/Code $ python gps.py
35.703129 51.351671 35.70412899999999 51.352671 0.0009999983331011 0.0009999983331011
Movement Direction: 0.7853981633974483 ,Sun Azimuth: 3.4100387482567838 ,Sun Angle: 2.6246405848593355
Sensor 3 is close to the Sun.
Sensor 0 => 16.875 (C) - 62.375 (F)
Sensor 1 => 16.75 (C) - 62.15 (F)
Sensor 2 => 16.625 (C) - 61.925 (F)
Sensor 3 => 17.062 (C) - 62.711600000000004 (F)
LED 0 off
LED 1 off
LED 2 off
LED 3 off
35.70412899999999 51.352671 35.70512899999999 51.353671 0.0009999983331011 0.0009999983331011
Movement Direction: 0.7853981633974483 ,Sun Azimuth: 3.410678705874469 ,Sun Angle: 2.6252805424770207
Sensor 3 is close to the Sun.
Sensor 0 => 16.875 (C) - 62.375 (F)
Sensor 1 => 16.875 (C) - 62.375 (F)
Sensor 2 => 16.625 (C) - 61.925 (F)
Sensor 3 => 17.062 (C) - 62.711600000000004 (F)
LED 0 off
LED 1 off
LED 2 off
LED 3 off
35.70512899999999 51.353671 35.70612899999999 51.35467099999996 0.0009999983331011 0.0009999983331011
Movement Direction: 0.7853981633974483 ,Sun Azimuth: 3.4113316200660755 ,Sun Angle: 2.625933456668627
Sensor 3 is close to the Sun.
Sensor 0 => 16.937 (C) - 62.4866 (F)
^XSensor 1 => 16.875 (C) - 62.375 (F)
Sensor 2 => 16.687 (C) - 62.03660000000001 (F)
^XSensor 3 => 17.125 (C) - 62.825 (F)
LED 0 off
LED 1 off
LED 2 off
LED 3 off
```

تصویر ۱۴ خروجی برنامه، سنسور ۳

۱۳.۲ لوکیشن: دانشگاه شریف، تهران، ایران (جهت حرکت: غرب)

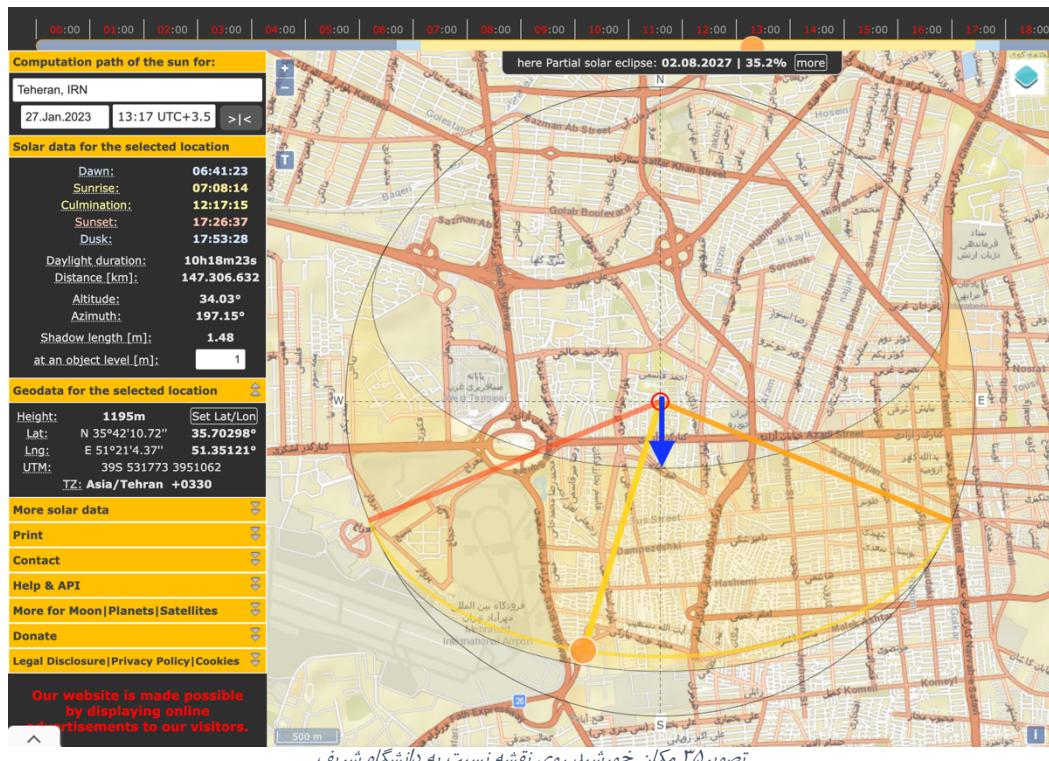


تصویر ۳۴ مکان خورشید روی نقشه نسبت به دانشگاه شریف

```
mmd@raspberrypi:~/Desktop/project/hw_project-group-3/Code $ python gps.py
35.703129 51.351671 35.703129 51.35067100000006 -0.000999999833331011 0.0
Movement Direction: -1.5707963267948966 ,Sun Azimuth: 3.4290313305760307 ,Sun Angle: 4.999827657370927
Sensor 0 is close to the Sun.
Sensor 0 => 16.625 (C) - 61.925 (F)
Sensor 1 => 16.437 (C) - 61.586600000000004 (F)
Sensor 2 => 16.812 (C) - 62.2616 (F)
Sensor 3 => 16.812 (C) - 62.2616 (F)
LED 0 off
LED 1 off
LED 2 off
LED 3 off
35.703129 51.35067100000006 35.703129 51.3496710000001 -0.000999999833331011 0.0
Movement Direction: -1.5707963267948966 ,Sun Azimuth: 3.429640663382943 ,Sun Angle: 5.000436990177839
Sensor 0 is close to the Sun.
Sensor 0 => 16.75 (C) - 62.15 (F)
Sensor 1 => 16.437 (C) - 61.586600000000004 (F)
Sensor 2 => 16.875 (C) - 62.375 (F)
Sensor 3 => 16.875 (C) - 62.375 (F)
LED 0 off
LED 1 off
LED 2 off
LED 3 off
35.703129 51.3496710000001 35.703129 51.3486710000001 -0.000999999833331011 0.0
Movement Direction: -1.5707963267948966 ,Sun Azimuth: 3.4302409206807396 ,Sun Angle: 5.001037247475637
Sensor 0 is close to the Sun.
Sensor 0 => 16.812 (C) - 62.2616 (F)
Sensor 1 => 16.562 (C) - 61.811600000000006 (F)
Sensor 2 => 16.875 (C) - 62.375 (F)
Sensor 3 => 16.937 (C) - 62.4866 (F)
LED 0 off
LED 1 off
LED 2 off
LED 3 off
```

تصویر ۳۵ خروجی برنامه، سنسور *

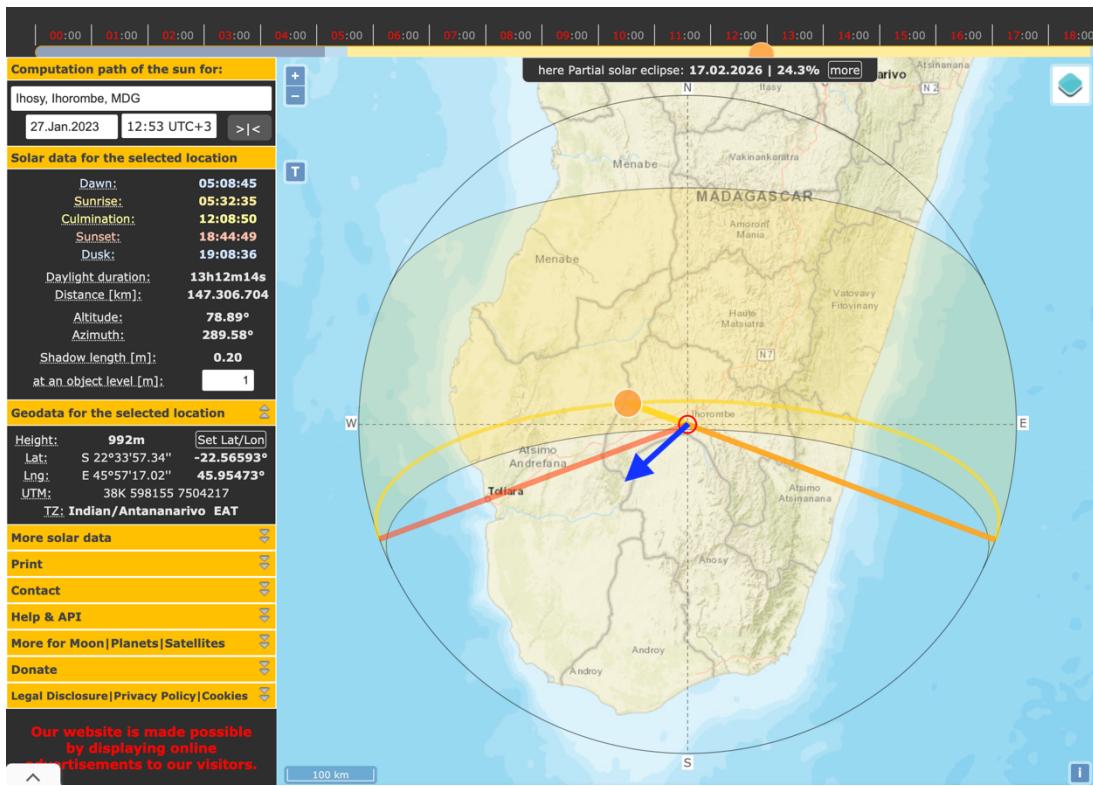
۱۳.۳ لوکیشن: دانشگاه شریف، تهران، ایران (جهت حرکت: جنوب)



```
mmd@raspberrypi:~/Desktop/project/hw_project-group-3/Code $ python gps.py
35.703129 51.351671 35.702129 51.351671 0.0 -0.00099999983331011
Movement Direction: 3.141592653589793 ,Sun Azimuth: 3.4444972056398444 ,Sun Angle: 0.3029045520500513
Sensor 2 is close to the Sun.
Sensor 0 => 17.312 (C) - 63.16160000000001 (F)
Sensor 1 => 17.5 (C) - 63.5 (F)
Sensor 2 => 17.125 (C) - 62.825 (F)
Sensor 3 => 17.437 (C) - 63.3866 (F)
LED 0 off
LED 1 off
LED 2 off
LED 3 off
35.702129 51.351671 35.701129 51.351671 0.0 -0.00099999983331011
Movement Direction: 3.141592653589793 ,Sun Azimuth: 3.445129176795125 ,Sun Angle: 0.30353652320533175
Sensor 2 is close to the Sun.
Sensor 0 => 17.312 (C) - 63.16160000000001 (F)
Sensor 1 => 17.5 (C) - 63.5 (F)
Sensor 2 => 17.125 (C) - 62.825 (F)
Sensor 3 => 17.437 (C) - 63.3866 (F)
LED 0 off
LED 1 off
LED 2 off
LED 3 off
35.701129 51.351671 35.700129000000004 51.351671 0.0 -0.00099999983331011
Movement Direction: 3.141592653589793 ,Sun Azimuth: 3.4457488425559406 ,Sun Angle: 0.3041561889661475
Sensor 2 is close to the Sun.
Sensor 0 => 17.312 (C) - 63.16160000000001 (F)
Sensor 1 => 17.5 (C) - 63.5 (F)
Sensor 2 => 17.062 (C) - 62.711600000000004 (F)
Sensor 3 => 17.5 (C) - 63.5 (F)
LED 0 off
LED 1 off
LED 2 off
LED 3 off
```

تصویر ۳۶ خروجی برنامه، سنسور ۲

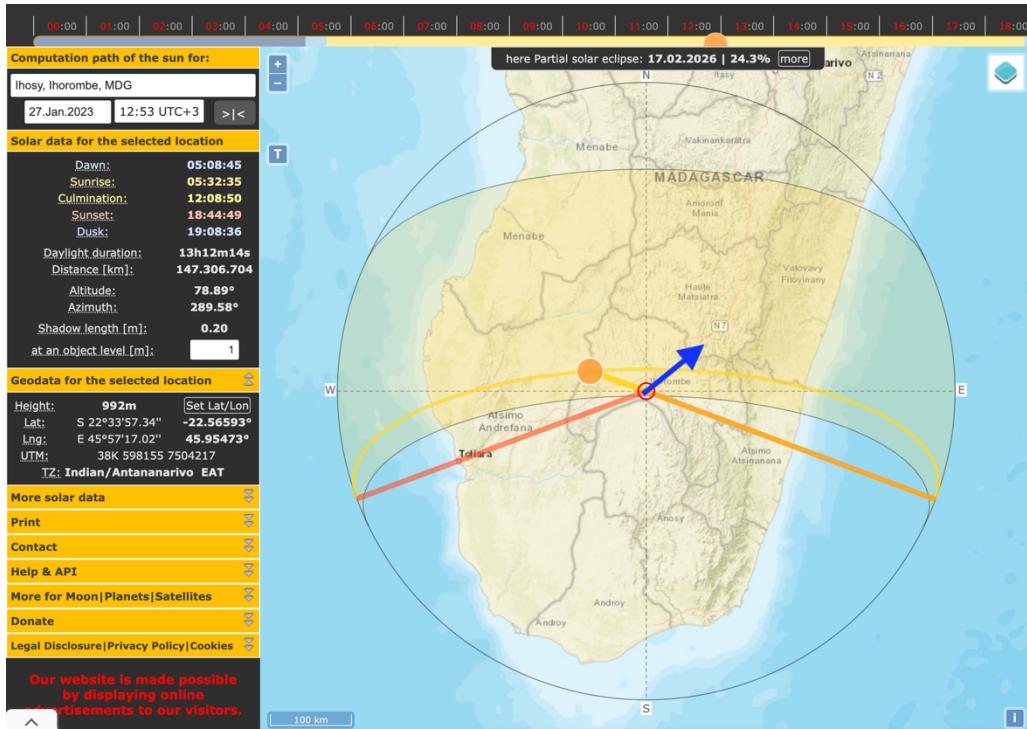
۱۳.۴ لوکیشن: جزیره‌ی ماداگاسکار (جهت حرکت: جنوب غربی)



```
lmid@raspberrypi:~/Desktop/project/hw_project-group-3/Code $ python gps.py
-22.56593 45.95473 -22.566930000000003 45.95373 -0.00099999833331011 -0.000999998333345638
Movement Direction: -2.356194490194121 ,Sun Azimuth: 5.009569523022366 ,Sun Angle: 1.0825787060369017
Sensor 2 is close to the Sun.
Sensor 0 => 17.625 (C) - 63.725 (F)
Sensor 1 => 17.875 (C) - 64.175 (F)
Sensor 2 => 17.5 (C) - 63.5 (F)
Sensor 3 => 17.812 (C) - 64.0616 (F)
LED 0 off
LED 1 off
LED 2 off
LED 3 off
-22.566930000000003 45.95373 -22.567930000000004 45.95273 -0.00099999833331011 -0.000999998333345638
Movement Direction: -2.356194490194121 ,Sun Azimuth: 5.008753207599567 ,Sun Angle: 1.0817623906141023
Sensor 2 is close to the Sun.
Sensor 0 => 17.625 (C) - 63.725 (F)
Sensor 1 => 17.875 (C) - 64.175 (F)
Sensor 2 => 17.5 (C) - 63.5 (F)
Sensor 3 => 17.812 (C) - 64.0616 (F)
LED 0 off
LED 1 off
LED 2 off
LED 3 off
-22.567930000000004 45.95273 -22.568930000000005 45.951730000000005 -0.00099999833331011 -0.000999998333345638
Movement Direction: -2.356194490194121 ,Sun Azimuth: 5.007948187612913 ,Sun Angle: 1.0809573706274485
Sensor 2 is close to the Sun.
Sensor 0 => 17.625 (C) - 63.725 (F)
```

تصویر ۳۶ خروجی برآمده، سنسور ۲

۱۳.۵ لوکیشن: جزیره‌ی ماداگاسکار (جهت حرکت: شمال شرقی)

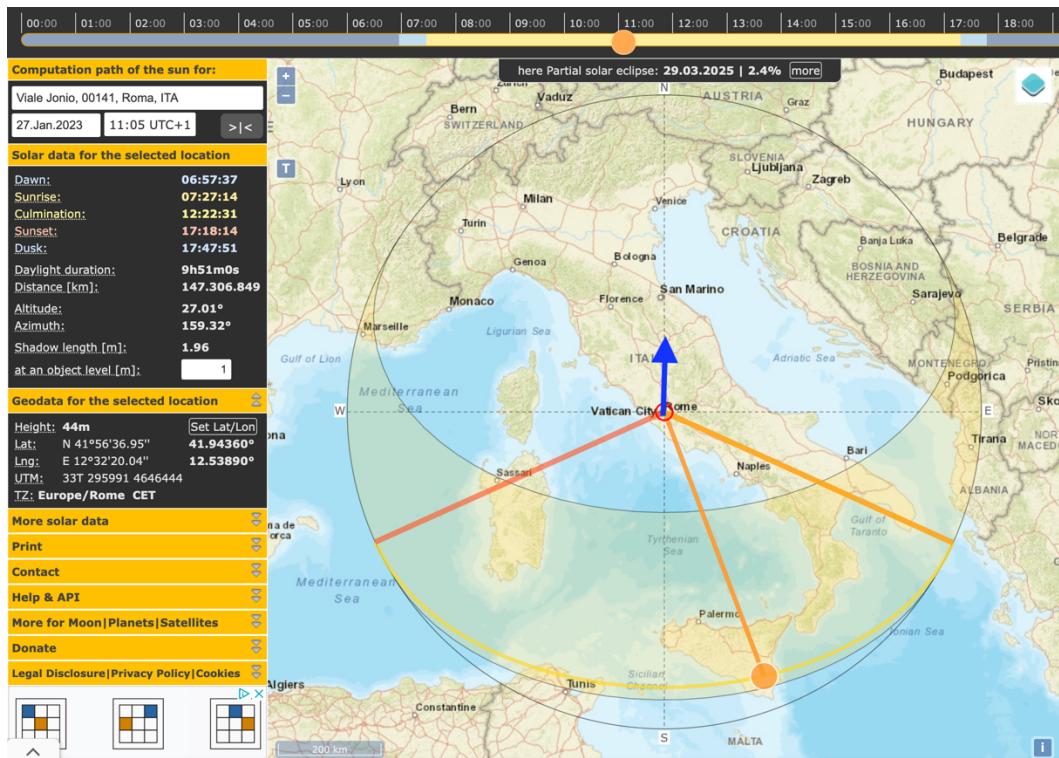


تصویر ۳۹ مکان خورشید روی نقشه نسبت به جزیره ماداگاسکار

```
mmd@raspberrypi:~/Desktop/project/hw_project-group-3/Code $ python gps.py
-22.56593 45.95473 -22.56493 45.95572999999996 0.000499999791655016 0.00049999979167278
Movement Direction: 0.7853981633956719 ,Sun Azimuth: 4.986064194783283 ,Sun Angle: 4.200666031387611
Sensor 1 is close to the Sun.
Sensor 0 => 17.812 (C) - 64.0616 (F)
Sensor 1 => 17.5 (C) - 63.5 (F)
Sensor 2 => 17.75 (C) - 63.95 (F)
Sensor 3 => 17.687 (C) - 63.8366000000000004 (F)
LED 0 off
LED 1 off
LED 2 off
LED 3 off
-22.56493 45.95572999999996 -22.56393 45.95672999999999 0.000499999791655016 0.00049999979167278
Movement Direction: 0.7853981633956719 ,Sun Azimuth: 4.985141383119534 ,Sun Angle: 4.199743219723862
Sensor 1 is close to the Sun.
Sensor 0 => 17.812 (C) - 64.0616 (F)
Sensor 1 => 17.5 (C) - 63.5 (F)
Sensor 2 => 17.75 (C) - 63.95 (F)
Sensor 3 => 17.687 (C) - 63.8366000000000004 (F)
LED 0 off
LED 1 off
LED 2 off
LED 3 off
-22.56393 45.95672999999999 -22.56292999999998 45.95772999999999 0.000499999791655016 0.00049999979167278
Movement Direction: 0.7853981633956719 ,Sun Azimuth: 4.98420795737542 ,Sun Angle: 4.198809793979748
Sensor 1 is close to the Sun.
Sensor 0 => 17.875 (C) - 64.175 (F)
Sensor 1 => 17.562 (C) - 63.6116 (F)
Sensor 2 => 17.687 (C) - 63.8366000000000004 (F)
Sensor 3 => 17.687 (C) - 63.8366000000000004 (F)
LED 0 off
LED 1 off
LED 2 off
LED 3 off
```

تصویر ۴۰ خروجی برنامه، سنسور ۱

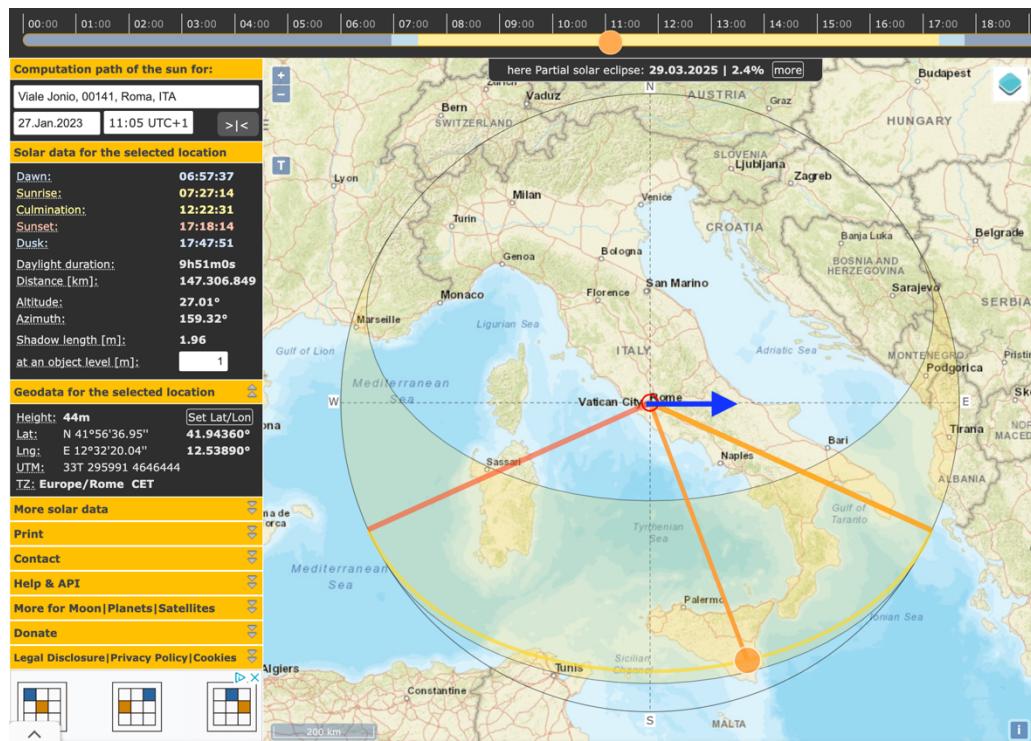
۱۳.۶ لوکیشن: رم، ایتالیا (جهت حرکت: شمال)



```
mmd@raspberrypi:~/Desktop/project/hw_project-group-3/Code $ python gps.py
41.9436 12.5389 41.9446 12.5389 0.0 0.000499999791655016
Movement Direction: 0.0 ,Sun Azimuth: 2.805242626859979 ,Sun Angle: 2.805242626859979
Sensor 3 is close to the Sun.
Sensor 0 => 17.687 (C) - 63.83660000000000 (F)
Sensor 1 => 16.937 (C) - 62.4866 (F)
Sensor 2 => 17.937 (C) - 64.2866 (F)
Sensor 3 => 17.25 (C) - 63.05 (F)
LED 0 off
LED 1 off
LED 2 off
LED 3 off
41.9446 12.5389 41.9456 12.5389 0.0 0.000499999791655016
Movement Direction: 0.0 ,Sun Azimuth: 2.8058223449541373 ,Sun Angle: 2.8058223449541373
Sensor 3 is close to the Sun.
Sensor 0 => 17.812 (C) - 64.0616 (F)
Sensor 1 => 17.062 (C) - 62.71160000000000 (F)
Sensor 2 => 18.062 (C) - 63.4616 (F)
Sensor 3 => 17.312 (C) - 62.2116 (F)
```

تصویر ۳۸ خروجی برنامه، سنسور ۳

۱۳.۷ لوکیشن: رم، ایتالیا (جهت حرکت: شرق)



```
mmd@raspberrypi:~/Desktop/project/hw_project-group-3/Code $ python gps.py
41.9436 12.5389 41.9436 12.5399 0.0004999999791663898 0.0
Movement Direction: 1.5707963267948966 ,Sun Azimuth: 2.79756005880986 ,Sun Angle: 1.2267637320149634
Sensor 2 is close to the Sun.
Sensor 0 => 17.5 (C) - 63.5 (F)
Sensor 1 => 17.312 (C) - 63.16160000000001 (F)
Sensor 2 => 17.812 (C) - 64.0616 (F)
Sensor 3 => 17.437 (C) - 63.3866 (F)
LED 0 off
LED 1 off
LED 2 off
LED 3 off
41.9436 12.5399 41.9436 12.54089999999999 0.0004999999791663898 0.0
Movement Direction: 1.5707963267948966 ,Sun Azimuth: 2.7981652934294226 ,Sun Angle: 1.2273689666345255
Sensor 2 is close to the Sun.
Sensor 0 => 17.625 (C) - 63.725 (F)
Sensor 1 => 17.25 (C) - 63.05 (F)
Sensor 2 => 17.812 (C) - 64.0616 (F)
Sensor 3 => 17.5 (C) - 63.5 (F)
LED 0 off
LED 1 off
LED 2 off
LED 3 off
41.9436 12.54089999999999 41.9436 12.54189999999998 0.0004999999791663898 0.0
Movement Direction: 1.5707963267948966 ,Sun Azimuth: 2.798762837484175 ,Sun Angle: 1.2279665106892779
Sensor 2 is close to the Sun.
Sensor 0 => 17.687 (C) - 63.83660000000004 (F)
Sensor 1 => 17.312 (C) - 63.16160000000001 (F)
Sensor 2 => 17.812 (C) - 64.0616 (F)
Sensor 3 => 17.5 (C) - 63.5 (F)
LED 0 off
LED 1 off
LED 2 off
LED 3 off
```

تصویر ۴۴ خروجی برنامه، سنسور ۲

14. بسته‌بندی

برای پیاده‌سازی بسته‌بندی این محصول به‌طوریکه در خودرو قابل استفاده باشد، نیاز به ابزارهای دیگری داریم. اولین مسئله در پیاده‌سازی، سنسور GPS است. سنسوری که ما برای این محصول استفاده کردیم Neo-7m-C بود که با وجود دقت بالا، فقط قابل استفاده در محیط آزاد بود. پس اگر جنس بدنه خودرو به گونه‌ای باشد که منجر به عدم رسیدن سیگنال‌های GPS به این سنسور گردد، قابلیت استفاده از آن را نخواهیم داشت و نیاز به سنسور جایگزین داریم. در غیر اینصورت، هنگام بسته‌بندی نیز باید توجه داشته باشیم که از محصولاتی استفاده شود که در این زمینه عایق نباشند و سیگنال GPS را از خود عبور دهنده تا کارکرد دستگاه دچار مشکل نشود.

مورد دیگری که باید به آن توجه شود، مجزا کردن 4 بخش ذکر شده است. در پیاده‌سازی فعلی، هر 4 سنسور دما در یک بردبورد قرار دارند که گرچه از لحاظ فضا بهینه است، اما با واقعیت خودرو تطبیق ندارد؛ چراکه نیاز داریم در خودرو سنسورها را دور از هم داشته باشیم تا تغییر دما در یک بخش روی بخشی دیگر تاثیر نگذارد. در پیاده‌سازی کنونی، به علت نزدیکی سنسور دماها، این مشکل وجود دارد که دماهای گرفته شده بسیار به هم نزدیک خواهند بود و در دقت محصول تاثیر خواهد گذاشت. برای رفع این مشکل، نیاز داریم هر بخش را در یک بردبورد پیاده‌سازی کنیم. یعنی در هر بردبورد، یک سنسور دما و یک LED قرار دهیم تا قرارگیری آن در خودرو ممکن شود. هم‌چنین، برای ارتباط بردبورد ها با یکدیگر به سیم‌های طولانی تری نیاز خواهیم داشت.

مسئله نهایی که باید مورد توجه قرار دهیم، منبع تغذیه است. از آنجایی که برای پیاده‌سازی این محصول از رزیزی پای استفاده کردیم، نیاز به فراهم کردن منبع تغذیه برای آن خواهیم داشت. هنگامیکه به تست محصول خارج از خودرو می‌پرداختیم از لپتاپ به منظور این کار استفاده می‌کردیم. این روش در خودرو نیز قابل استفاده است، اما باید توجه داشته باشیم که از لحاظ زمانی محدود بوده و نیاز به منبع تغذیه جایگزین خواهیم داشت.

15. نتیجه‌گیری

با توجه به توضیحات ذکر شده، سیستم خنک‌کننده خودرو با توجه به جهت تابش خورشید به خودرو و دمای بخش‌های مختلف آن، پیاده‌سازی شد. در این محصول ابتدا جهت حرکت خودرو تشخیص داده می‌شود، سپس زاویه تابش خورشید نسبت به شمال بدست می‌آید و نهایتاً با ترکیب این دو، تشخیص می‌دهیم کدام یک از چهار بخش خودرو در معرض آفتاب است. سپس دمای این چهار بخش اندازه‌گیری شده و خنک‌کننده بخش در معرض آفتاب اگر دمای بالاتری داشت روشن می‌شود. هم‌چنین پس از متعادل‌سازی دمای خودرو، به جهت بهینه‌سازی مصرف انرژی این خنک‌کننده خاموش می‌گردد.

وروودی	دماهای خودرو، مختصات خودرو
خرسچه	روشن یا خاموش شدن LED خنک‌کننده
دماهای کار	منفی 10 تا مثبت 85 درجه سانتیگراد
دققت دما	0.5 درجه سانتیگراد
ولتاژ مجاز	1.6 تا 5 ولت

جدول 1 دیتاشیت محصول

16. مراجع

1. <https://www.suncalc.org/>
2. <https://www.circuitbasics.com/raspberry-pi-ds18b20-temperature-sensor-tutorial/>
3. <https://sparklers-the-makers.github.io/blog/robotics/use-neo-6m-module-with-raspberry-pi/>
4. <https://github.com/kylebarron/suncalc-py>
5. <https://pinout.xyz/>