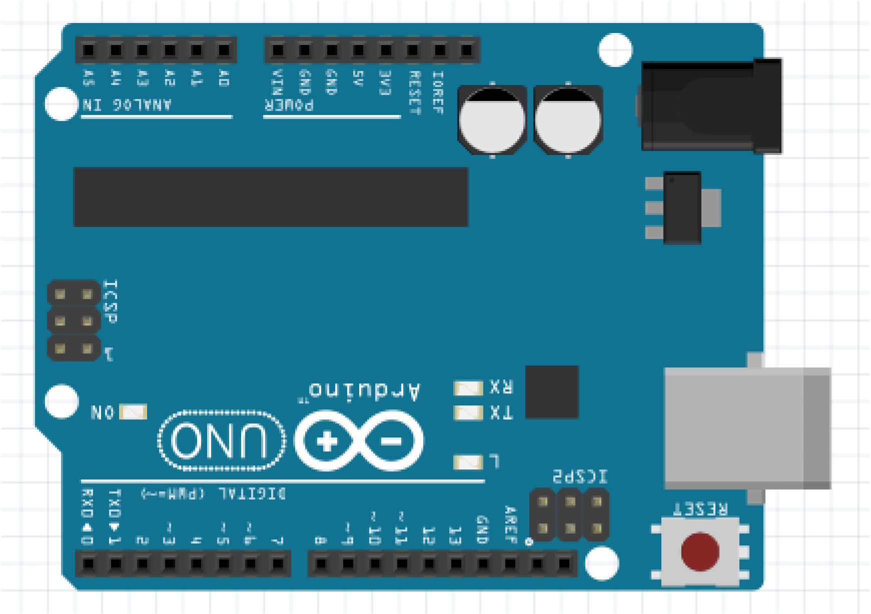
C_ARDUINO – GPS Receiver

https://github.com/teaksoon/c_arduino_comp



HARDWARE
1x Computer with Arduino IDE Software
1x USB 2.0 Type A/B Data Cable
1x Arduino Uno Board
1x Solderless Breadboard
Nx Jumper wires

1x VK16E GPS Receiver

VK16E Red Wire to Arduino Uno 5V
VK16E Black Wire to Arduino Uno Gnd
VK16E Blue Wire(TX) to Arduino Pin 2
VK16E Green Wire(RX) to Arduino Pin 3

VK16E GPS Receiver is just one of the many GPS Receivers in the market. Most of them works similar. Some are more sensitive, some are less sensitive

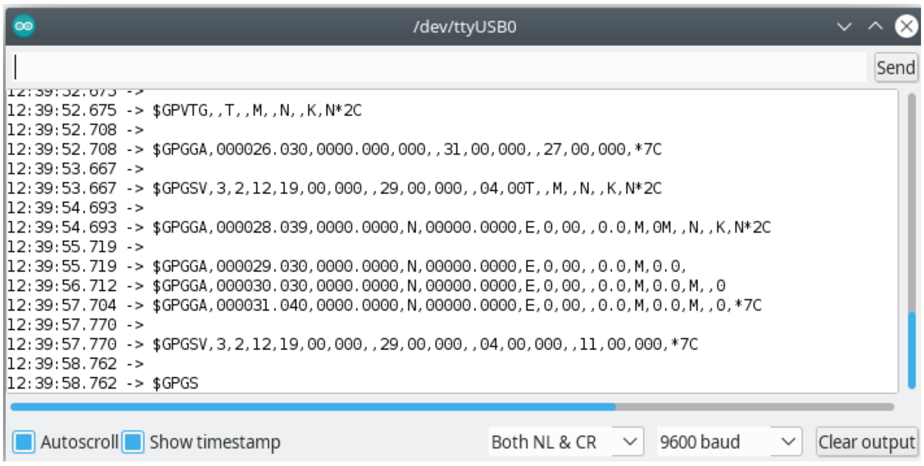
Source code: **gps_receiver_vk16e_raw_serial**
Download from:
https://github.com/teaksoon/c_arduino_comp/blob/main/2022_01_13_gps_receiver.zip
Upload PROGRAM, use the Serial Monitor to see output

```
#include <SoftwareSerial.h>

#define PIN_SS_RX   2
#define PIN_SS_TX   3
SoftwareSerial ss(PIN_SS_RX, PIN_SS_TX);

void setup(){
  Serial.begin(9600);
  ss.begin(9600);  // VK16E default baud rate is 9600
  Serial.print("\n\nRaw data from VK16E\n\n");

  delay(1000);
}
void loop(){
char ch;
  while (ss.available() > 0) {
    ch = ss.read();
    if (ch=='$') Serial.print("\n");
    Serial.print(ch);
  }
  delay(1000);
}
```



```
12:39:52.675 ->
12:39:52.675 -> $GPVTG,,T,,M,,N,,K,N*2C
12:39:52.708 ->
12:39:52.708 -> $GPGGA,000026.030,0000.000,000,,31,00,000,,27,00,000,*7C
12:39:53.667 ->
12:39:53.667 -> $GPGSV,3,2,12,19,00,000,,29,00,000,,04,00T,,M,,N,,K,N*2C
12:39:54.693 ->
12:39:54.693 -> $GPGGA,000028.039,0000.0000,N,00000.0000,E,0,00,,0.0,M,0M,,N,,K,N*2C
12:39:55.719 ->
12:39:55.719 -> $GPGGA,000029.030,0000.0000,N,00000.0000,E,0,00,,0.0,M,0.0,
12:39:56.712 -> $GPGGA,000030.030,0000.0000,N,00000.0000,E,0,00,,0.0,M,0.0,M,,0
12:39:57.704 -> $GPGGA,000031.040,0000.0000,N,00000.0000,E,0,00,,0.0,M,0.0,M,,0,*7C
12:39:57.770 ->
12:39:57.770 -> $GPGSV,3,2,12,19,00,000,,29,00,000,,04,00,000,,11,00,000,*7C
12:39:58.762 ->
12:39:58.762 -> $GPGS
```

This PROGRAM is currently running indoor, we wont get much data from the VK16E as the satellite data cant go through so many layers of our ceilings. If you look at the displayed data on the Serial Monitor, it does not look very good. The best is take this device outdoor for a few minutes, it should be able to load quite alot of data for testing.

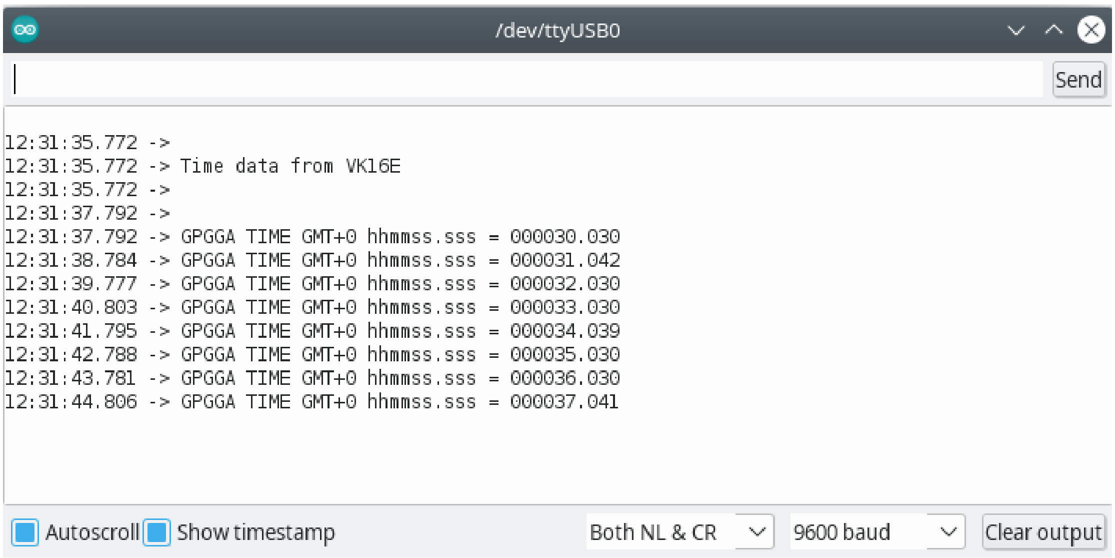
NOTE:
There is only one hardware UART (Pin 0-RX and Pin 1-TX) available in the ATMEGA328 micro-controller. The only hardware UART in the ATMEGA328 micro-controller is used by the Arduino IDE Software Upload and Serial Monitor

This GPS Receiver also uses UART (RX/TX) to receive/transmit data. If we want to use our Serial Monitor to display output, the only hardware UART will not be available for the GPS Receiver.

We will have to use regular I/O Pins to do data transfer for our GPS Receiver. Fortunately, the Arduino Library provide us with the PROGRAM code “SoftwareSerial” to perform UART on the regular I/O Pins (there is nothing stopping us from coding it if we wish, we just use what has been provided for now)

Source code: **gps_receiver_vk16e_time_serial**
Download from:
https://github.com/teaksoon/c_arduino_comp/blob/main/2022_01_13_gps_receiver.zip
Upload PROGRAM, use the Serial Monitor to see output

Source Codes are long, wil not be posted here. Please download to see



```
12:31:35.772 ->
12:31:35.772 -> Time data from VK16E
12:31:35.772 ->
12:31:37.792 ->
12:31:37.792 -> GPGGA TIME GMT+0 hhmmss.sss = 000030.030
12:31:38.784 -> GPGGA TIME GMT+0 hhmmss.sss = 000031.042
12:31:39.777 -> GPGGA TIME GMT+0 hhmmss.sss = 000032.030
12:31:40.803 -> GPGGA TIME GMT+0 hhmmss.sss = 000033.030
12:31:41.795 -> GPGGA TIME GMT+0 hhmmss.sss = 000034.039
12:31:42.788 -> GPGGA TIME GMT+0 hhmmss.sss = 000035.030
12:31:43.781 -> GPGGA TIME GMT+0 hhmmss.sss = 000036.030
12:31:44.806 -> GPGGA TIME GMT+0 hhmmss.sss = 000037.041
```

☑ Autoscroll ☑ Show timestamp   Both NL & CR   9600 baud

In this PROGRAM, we filter out all the unwanted data. In this example, we are only interested in the time transmitted by the satellite. The time from the satellite is in the second column of the $GPGGA data. This time is at GMT +0. Depending on where our GPS Receiver is. For example: Malaysia is GMT +8, so we need to plus 8 hours to it. Data is in hhmmss.sss 24hours format

Note: the Serial Monitor shows data starting 000030.030, that is 30 seconds when I started off my GPS Receiver (I am still indoor). Try bring the GPS Receiver outdoor with clear sky and switch it on for a while and you will get the real satellite time

There are any other data inside the $GPGGA data, for example: the GPS latitude (N/S) and longitude (E/W) position of the GPS Receiver. With a small calculation on the raw data, we can feed the position into the Google Map to show the exact location of the GPS Receiver on the Google map.

Apart from $GPGGA there are few other message headers from the satellite for different use, please refer to the datasheet for more details.