```
from machine import Pin, UART, I2C

#Import utime library to implement delay

import utime, time


#_____________________

from ssd1306 import SSD1306_I2C

#https://github.com/stlehmann/micropython-ssd1306

#_____________________

from micropyGPS import MicropyGPS

#https://github.com/inmcm/micropyGPS

#_____________________


############################################################

#Oled I2C connection

i2c=I2C(0, sda=Pin(8), scl=Pin(9), freq=400000)

oled = SSD1306_I2C(128, 64, i2c)

############################################################


############################################################

#GPS Module UART Connection

gps_module = UART(1, baudrate=9600, tx=Pin(4), rx=Pin(5))

############################################################



############################################################

TIMEZONE = 5.30

my_gps = MicropyGPS(TIMEZONE)
```

```
############################################################

############################################################
def convert(parts):
    if (parts[0] == 0):
        return None

    data = parts[0]+(parts[1]/60.0)
    # parts[2] contain 'E' or 'W' or 'N' or 'S'
    if (parts[2] == 'S'):
        data = -data
    if (parts[2] == 'W'):
        data = -data

    data = '{0:.6f}'.format(data) # to 6 decimal places
    return str(data)
############################################################

############################################################
while True:
    #_____________________
    #print(i2c.scan())
    length = gps_module.any()
    if length>0:
        b = gps_module.read(length)
        for x in b:
            msg = my_gps.update(chr(x))
```

```
#____________________

latitude = convert(my_gps.latitude)

longitude = convert(my_gps.longitude)

#____________________

if (latitude == None and latitude == None):

   oled.fill(0)

   oled.text("No Data", 0, 0)

   oled.show()

   continue

#____________________

t = my_gps.timestamp

#t[0] => hours : t[1] => minutes : t[2] => seconds

gpsTime = '{:02}:{:02}:{:02}'.format(t[0], t[1], t[2])


gpsdate = my_gps.date_string('long')

speed = my_gps.speed_string('kph') #'kph' or 'mph' or 'knot'

#____________________

print('Lat:', latitude)

print('Lng:', longitude)

#print('time:', gpsTime)

#print('Date:', gpsdate)

#print('speed:', speed)

#____________________

oled.fill(0)

oled.text('Lat:'+ latitude, 0, 0)

oled.text('Lng:'+ longitude, 0, 12)

oled.text('Speed:'+ speed, 0, 24)
```

```
    oled.text('Time:'+ gpsTime, 0, 36)
    oled.text(gpsdate, 0, 48)
    oled.show()
    #____________________
```

##