MIKROE



GNSS 7 CLICK

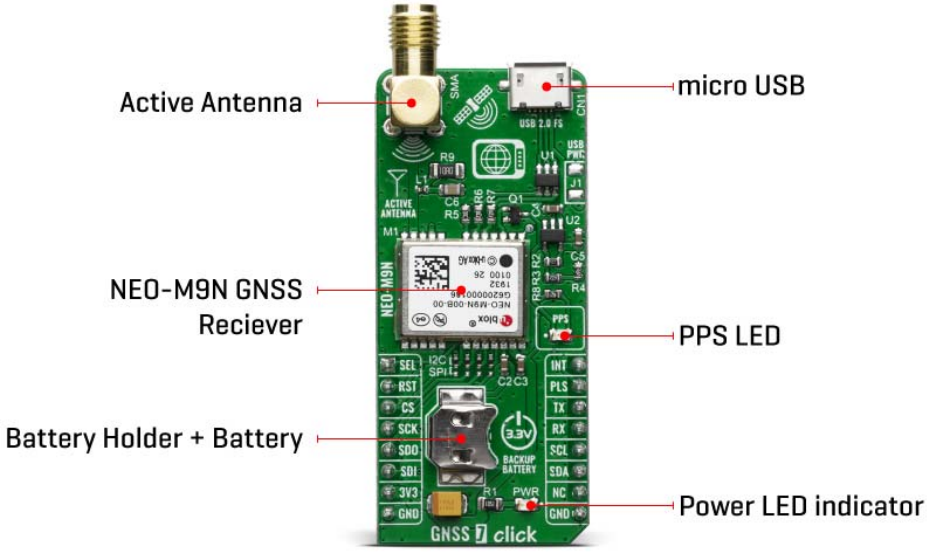
PID: MIKROE-3922
Weight: 25 g

Determine your current position with GNSS 7 click. It carries the NEO-M9N GNSS receiver module from u-blox. Thanks to u-blox's advanced algorithms and complete GNSS solution, the GNSS 7 click meets even the most stringent requirements in versatile industrial and consumer applications, such as UAVs, vehicles and assets tracking, navigation devices based on GPS, GLONASS, Beidou and Galileo, road navigation devices, public transport, wearable devices, and many more.

GNSS 7 click is supported by a mikroSDK compliant library, which includes functions that simplify software development. This Click board™ comes as a fully tested product, ready to be used on a system equipped with the mikroBUS™ socket.

HOW IT WORKS?

GNSS 7 click carries the NEO-M9N an ultra-robust meter-level GNSS positioning receiver module from U-blox. The mentioned module features onboard serial flash memory, message integrity protection, anti-jamming, anti-spoofing, and many more, making this Click board™ meet even the most stringent requirements in versatile industrial and consumer applications, such as UAVs, vehicles and assets tracking. For RF optimization, the NEO-M9N module features Advanced filtering algorithms mitigate the impact of RF interference and jamming, thus enabling the product to operate as intended. NEO-M9N click is designed mainly for using with NSS/GLONASS compatible active antennas. The NEO-M9 series utilizes concurrent reception of up to four GNSS (GPS, GLONASS, BeiDou, Galileo), recognizes multiple constellations simultaneously and provides outstanding positioning accuracy in scenarios where urban canyon or weak signals are involved.



Note: Button battery 1.55V is provided with the GNSS 7 click

The u-blox NEO-M9 modules can also benefit from the u-blox AssistNow assistance service. The Online service provides GNNS broadcast parameters, e.g. ephemeris, almanac plus time or rough position to reduce the receiver's time to first fix significantly and improve acquisition sensitivity.

Hardware Backup Mode - If the main supply voltage fails, and a battery is connected to V_BCKP, parts of the receiver switch off, but the RTC still runs providing a timing reference for the receiver. This operating mode enables all relevant data to be saved in the backup RAM to allow a hot or warm start later.

The GNSS 7 click supports both SPI and I2C/UART communication protocol configurations. Therefore, this Click board™ have communication interface selection jumpers, in order to allow the user to set whether to use SPI communication or

combination. A USB interface (micro USB port), which is compatible with the USB version 2.0 FS (Full Speed, 12 Mbit/s), can be used for communication as an alternative to the UART. The USB port can be used as a power supply as well if you need the click board™ to be a standalone device.

The Click board™ can operate with 3.3V MCUs only, it is set to work over the I2C and UART by default, and it is already equipped with the pull-up resistors. It is ready to be used as soon as it is inserted into a mikroBUS™ socket of the development system.

SPECIFICATIONS

Type	GPS+GNSS
Applications	Asset tracking, for navigation devices based on GPS and GLONASS, road navigation devices, public transport, wearable devices, etc.
On-board modules	The NEO-M9N GNSS receiver module from u-blox with Active antena
Key Features	NEO-M9 series utilizes concurrent reception of up to four GNSS (GPS, GLONASS, BeiDou, Galileo)
Interface	I2C,UART,SPI,GPIO
Compatibility	mikroBUS
Click board size	L (57.15 x 25.4 mm)
Input Voltage	3.3V

PINOUT DIAGRAM

This table shows how the pinout on GNSS 7 click corresponds to the pinout on the mikroBUS™ socket (the latter shown in the two middle columns).

Notes	Pin	mikroBUS			Pin	Notes
Interface Select	SEL	1	AN	PWM	16	EXI	External Interrupt
Reset Module	RST	2	RST	INT	15	PPS	Pulse-Per-Second (Time Pulse)
SPI Chip Select	CS	3	CS	RX	14	RX	UART Receive
SPI Clock	SCK	4	SCK	TX	13	TX	UART Transmit
SPI Data OUT	SDO	5	MISO	SCL	12	SCL	SCL I2C Clock
SPI Data IN	SDI	6	MOSI	SDA	11	SDA	SDA I2C Data
Power Supply	3.3V	7	3.3V	5V	10	NC	
Ground	GND	8	GND	GND	9	GND	Ground

ONBOARD SETTINGS AND INDICATORS

Label	Name	Default	Description
LD1	PWR	LED	Power LED Indicator
LD2	PPS	LED	Pulse Per Second (by default) when the GNSS receiver has established a connection with the satellite constellation the LED starts blinking

Note: That there is four SMD jumpers and all of them should be in the same position.

SOFTWARE SUPPORT

We provide a library for the GNSS7 Click on our LibStock page, as well as a demo application (example), developed using MikroElektronika compilers. The demo can run on all the main MikroElektronika development boards.

Library Description

Library carries generic command parser adopted for GPS-MNEA command based modules. Generic parser

Key functions:

- `gnss7_cmdSingle` - Sends provided command to the module
- `gnss7_setHandle` - Handler assignation to the provied command
- `gnss7_modulePower` - Turn on module

Examples description

The application is composed of three sections :

- System Initialization - Initializes all necessary GPIO pins, UART used for the communcation with GSM module and UART used for infromation logging
- Application Initialization
- Initializes driver, power on module and sends few command for the default module configuration
- Application Task - running in parallel core state machine and checks for call flag. If call detected parser will hang up call.

```
void application_task ( )
{
    char *p_lat;
    char *p_long;
    char *p_alt;
    char rsp_com[ 50 ] = {0};

    gnss7_process( );

    if ( timer_cnt > 5000 )
    {
        p_flag++;
        if ( p_flag > 2 )
        {
            p_flag = 0;
```

```
        }
        timer_cnt = 0;
        disp_flag = 1;
    }

    if ( ( p_flag == 0 ) && ( disp_flag == 1) )
    {
        mikrobus_logWrite( " ---------------------------------------- ", _LOG_LINE );
        p_lat = gnss7_gps_parser( &demo_buf[ 0 ], &demo_cmd[ 0 ], 2 );

        if ( p_lat == 0 )
        {
            mikrobus_logWrite( " Latitude : No data available!", _LOG_LINE );
        }
        else
        {
            strcpy( &rsp_com[ 0 ], p_lat );
            mikrobus_logWrite( " Latitude : ", _LOG_TEXT );
            mikrobus_logWrite( rsp_com, _LOG_LINE );
        }
        disp_flag = 0;
        mikrobus_logWrite( " ---------------------------------------- ", _LOG_LINE );
    }

    if ( ( p_flag == 2 ) && ( disp_flag == 1) )
    {
        mikrobus_logWrite( " ---------------------------------------- ", _LOG_LINE );
        p_alt = gnss7_gps_parser( &demo_buf[ 0 ], &demo_cmd[ 0 ], 9 );
        if ( p_alt == 0 )
        {
            mikrobus_logWrite( " Altitude : No data available!", _LOG_LINE );
        }
        else
        {
            strcpy( &rsp_com[ 0 ], p_alt );
            mikrobus_logWrite( " Altitude : ", _LOG_TEXT );
            mikrobus_logWrite( rsp_com, _LOG_LINE );
```

```
        }
        disp_flag = 0;
        mikrobus_logWrite( " ---------------------------------------- ", _LOG_LINE );
    }

    if ( ( p_flag == 1 ) && ( disp_flag == 1 ) )
    {
        mikrobus_logWrite( " ---------------------------------------- ", _LOG_LINE );
        p_long = gnss7_gps_parser( &demo_buf[ 0 ], &demo_cmd[ 0 ], 4 );
        if ( p_long == 0 )
        {
            mikrobus_logWrite( " Longitude : No data available!", _LOG_LINE );
        }
        else
        {
            strcpy( &rsp_com[ 0 ], p_long );
            mikrobus_logWrite( " Longitude : ", _LOG_TEXT );
            mikrobus_logWrite( rsp_com, _LOG_LINE );
        }
        disp_flag = 0;
        mikrobus_logWrite( " ---------------------------------------- ", _LOG_LINE );
    }
}
```

All additional functions such as timer initialization and default handler.

Notes GPS:

- The GPS module returns data in the form of NMEA responses
- For example, we used the GGA response

For parsing, use the GPS Parser function to send the following form of arguments: The name of the NMEA response that you want to parse, the position of the data that you need. As a response - you will get a separate buffer with the requested data

The full application code, and ready to use projects can be found on our LibStock page. Other mikroE Libraries used in the example:

- UART
- String
- Conversions

Additional notes and informations

Depending on the development board you are using, you may need USB UART click, USB UART 2 click or RS232 click to connect to your PC, for development systems with no UART to USB interface available on the board. The terminal available in all MikroElektronika compilers, or any other terminal application of your choice, can be used to read the message.

MIKROSDK

This Click board™ is supported with mikroSDK - MikroElektronika Software Development Kit. To ensure proper operation of mikroSDK compliant Click board™ demo applications, mikroSDK should be downloaded from the LibStock and installed for the compiler you are using.

For more information about mikroSDK, visit the official page.