ClickCloud

QUICK START GUIDE

VERSION 2.0

MIKROE

Document Scope

Intention of this document is to describe steps which should be taken in order to connect your hardware by using any of the Go to Cloud Click boards™ and Web app to the Click Cloud.

To be able to use Click Cloud services, you need to have:

- **Click Cloud Account** - a cloud hosting service, with the WEB-based user interface
- **Go to Cloud Click** - a hardware device used to connect to the Click Cloud service
- **USB to UART converter** or microcontroller (MCU) based board that have mikroBUS™ socket.

This document describes a basic usage of the Click Cloud services/application and it encompasses setup of all three components mentioned above by following next steps to get started:

STEP 1 / CLICK CLOUD ACCOUNT SETUP

STEP 2 / A DEVICE CREATION ON CLICK CLOUD

STEP 3 / HARDWARE SETUP

STEP 4 / CONFIGURING AND CONNECTING

STEP 5 / SENDING DATA TO THE CLOUD

STEP 6 / ACQUIRE DATA FROM THE CLOUD

There are two versions of the Go to Cloud Click boards available:

Go to Cloud (G2C) Click
which uses WiFi network for connection to the Internet.

Go to Cloud (G2C) 3G Click
which uses 3G UMTS/HSPA network for connection to the Internet.

STEP 1 / CLICK CLOUD ACCOUNT SETUP

Click Cloud IoT Tool Web App is free-to-use application which enables access your devices' data from your computer with the purpose of enabling visualization of real-time data acquired by Click Cloud IoT Platform, monitoring significant events and controlling actuators connected to the Platform.

As the first step it is required to create a ***Click Cloud Account*** if you do not have one. If you already have an account, you can just ***Login*** and create a new product on your Click Cloud or Hexiwear account.

For more information and step-by-step guide on how to manage your Click Cloud account please visit our Click Cloud ***User Guide*** page.

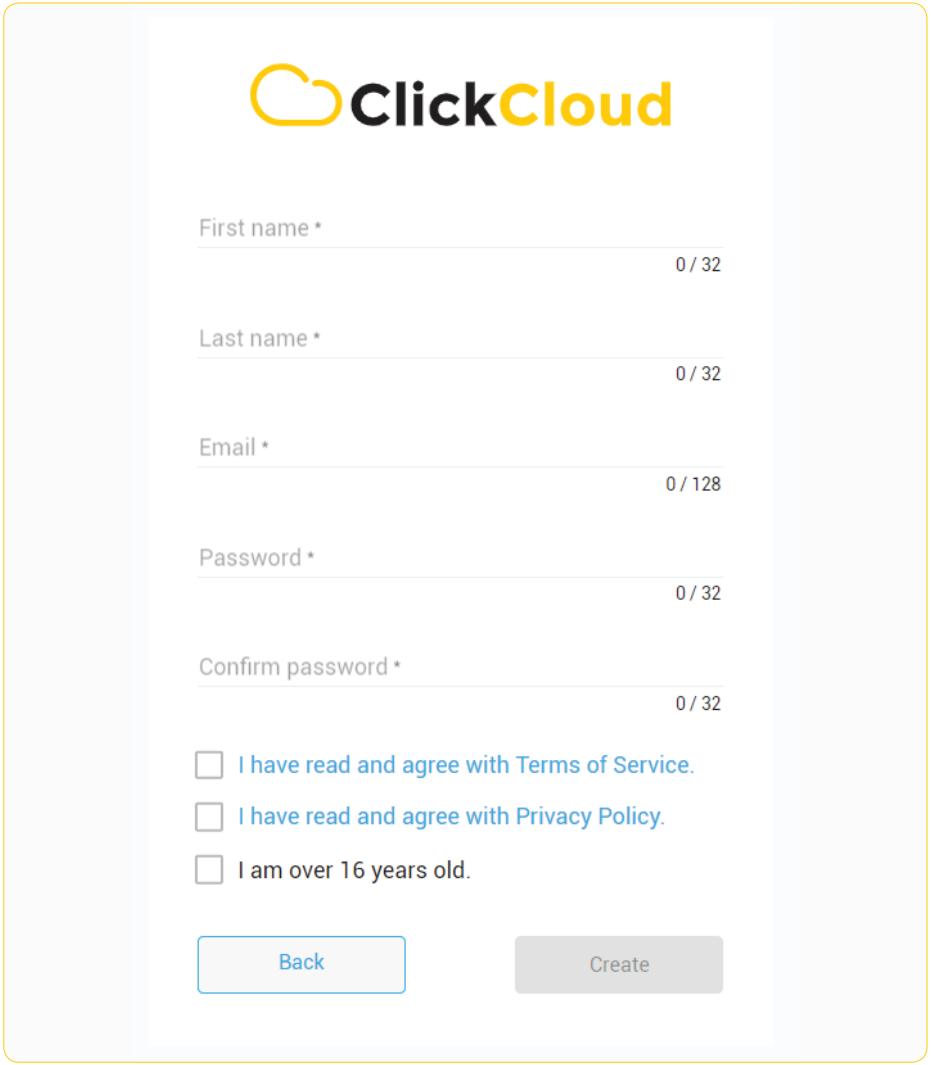

Figure 1
Registration window

STEP 2 / Device Creation on Click Cloud

After a successful ***Login*** to the Click Cloud service via a web browser, click on the **Devices** icon on the sidebar. We are going to create a new device, which will contain the existing TEST device manifest.

1. Devices > *Add device* – Then **Add device** on ⊕ in the upper left corner of your Device window.

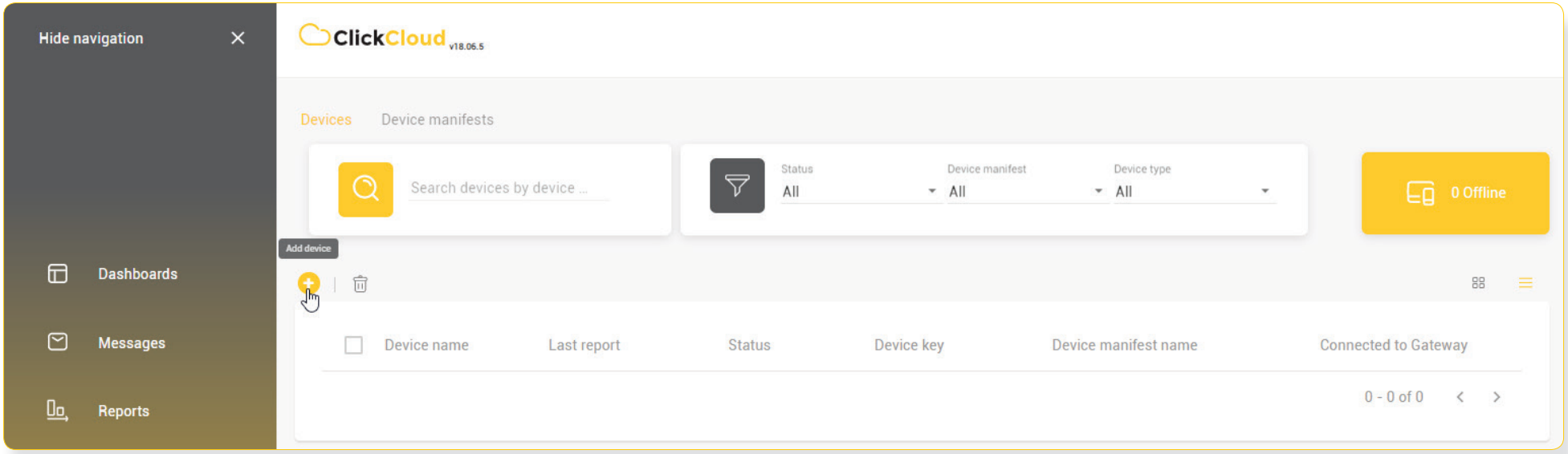

Figure 2 – Devices window

2. Select **Go To Cloud click – TEST** manifest, which contains both Sensor and Actuator settings, from the drop-down list and press **Next step.**

Manifest represents a contract between a physical device and ClickCloud IoT Platform. For more information visit this ***page***.

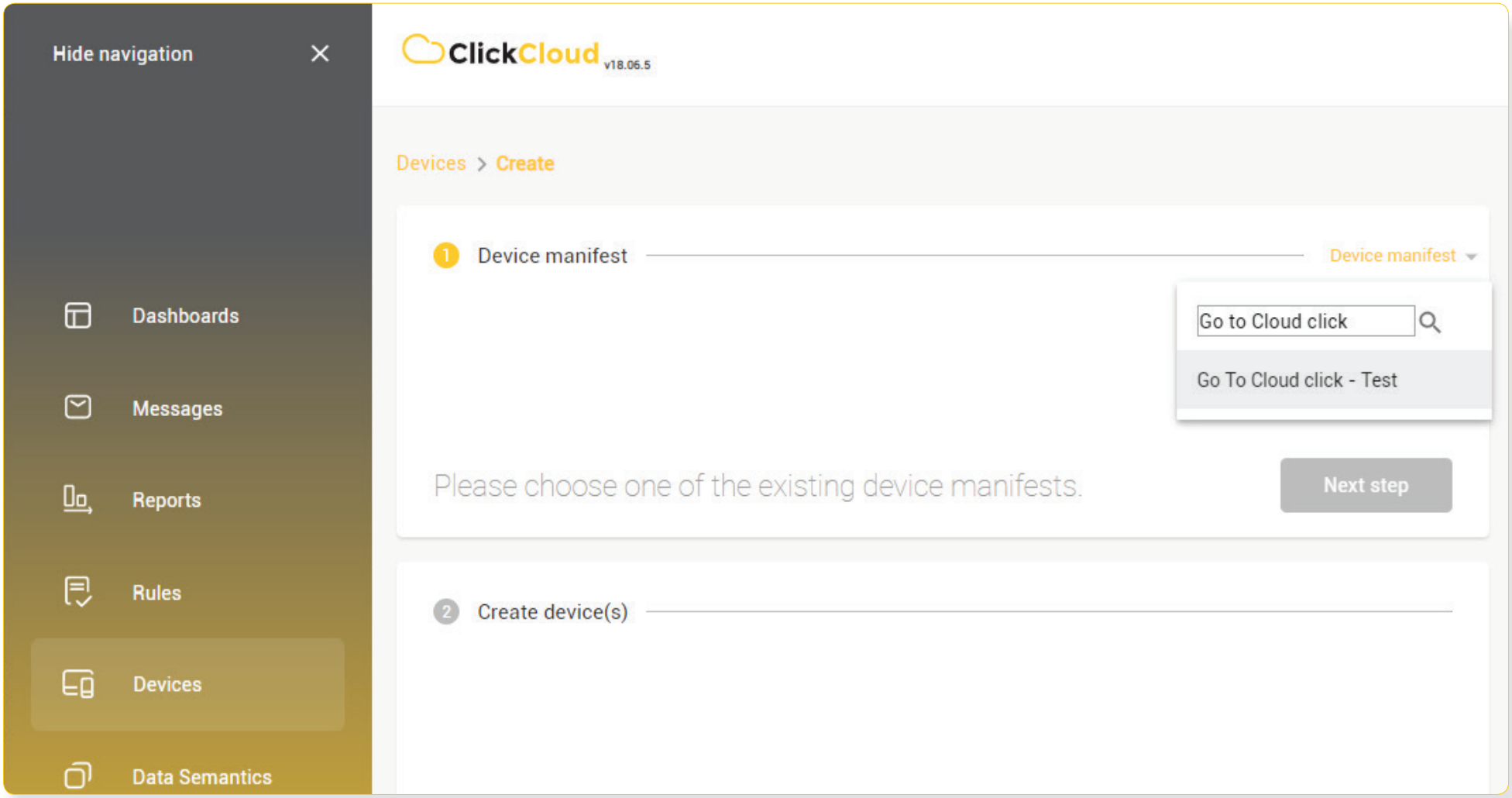

Figure 3 – Device > Create window

3. Write **Device name > Save** [Leave the Existing device Key field blank since it will be created in the next step]

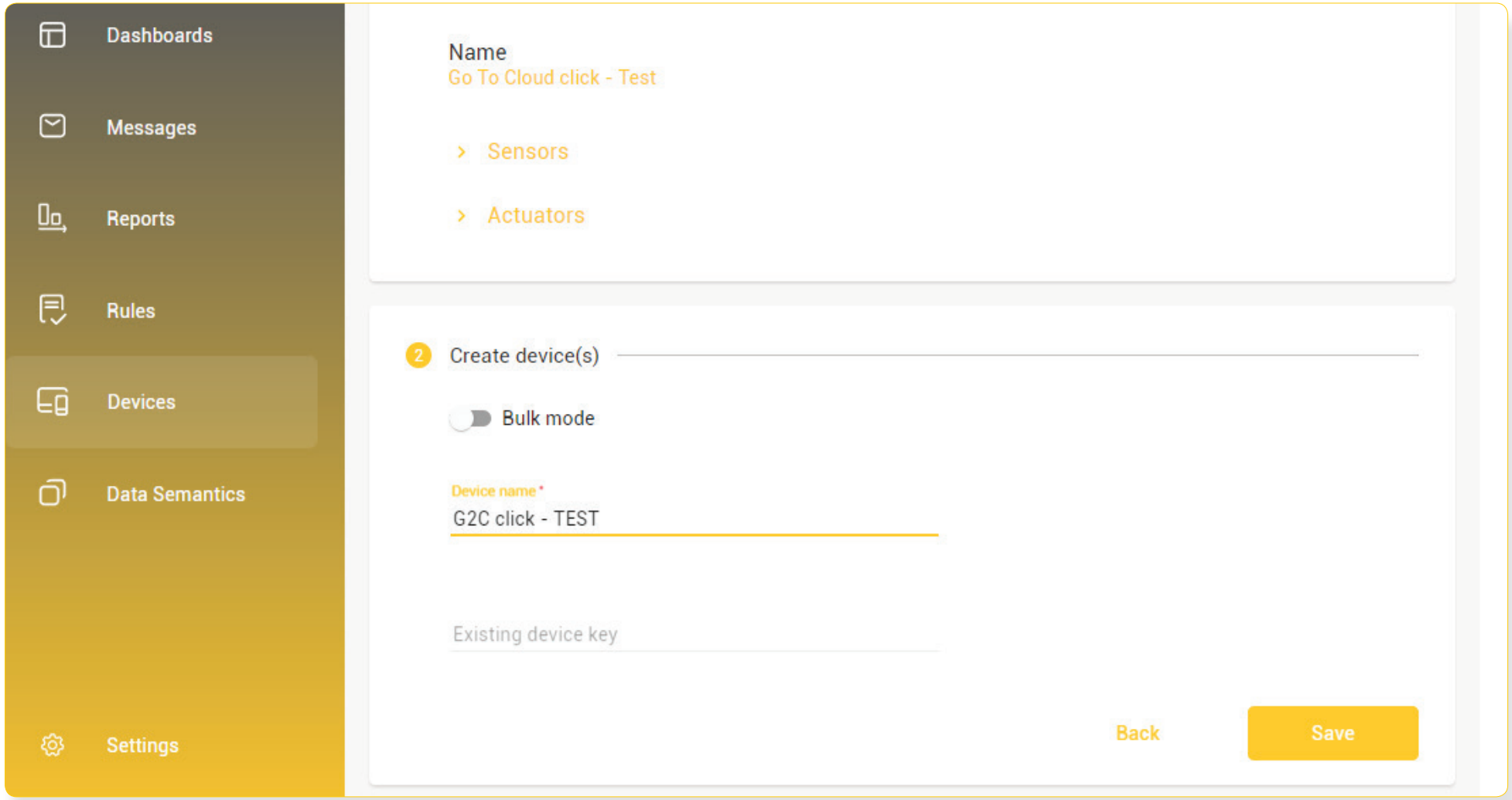

Figure 4 – Device name window

4. Save the provided device key and password by downloading them or sending them to email. Same Device Key needs to be used when configuring your hardware.

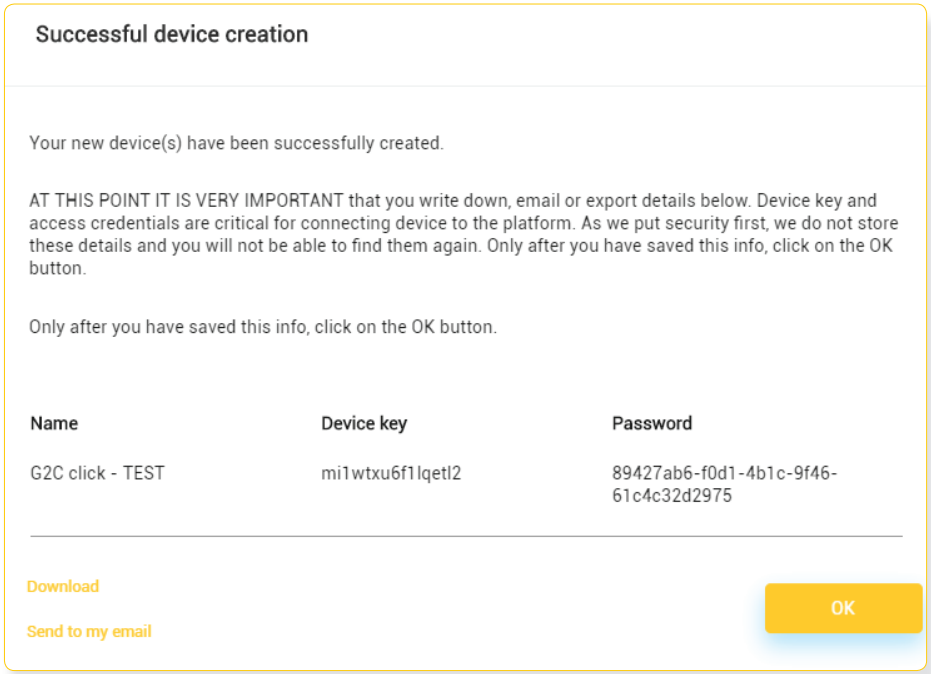

Figure 5 – Successful device creation window

After you have ran through these steps, the Click Cloud web application will return you to the **Devices** window, where you can see the new device you have just created.

It is important to note that only one **Device** is required for each board, however you can make several devices from one board.

STEP 3 / Hardware Setup

In order to connect to the Click Cloud platform you'll only need one of our Go to Cloud Click's – the hardware which allows physical connection to this service over internet. Considering that Go to Cloud Click is a board with UART interface, you will also need additional board to make it work. Go to Cloud Click should be configured by connecting it as shown on the *Figure 6*.

There are several ways that you can put in use your Go to Cloud Click boards:

Option 1 - PC controlled by using terminal application on a personal computer (for connecting the Go to Cloud Click and PC you will need to have an USB - to - UART converter such as ***Click USB adapter***).

Option 2 - MCU controlled by sending the UART data directly from the microcontroller (MCU) of your choice by using some development board with mikroBUS™ socket such as ***Clicker board***.

UART interface – configuration parameters

Baud rate: 57600	**Parity: NO**
Data bits: 8	**Stop bit: 1**

Send AT commands through the USART terminal to the board for testing it or directly from the main board. The Go to Cloud Click firmware accepts AT commands, which can be sent over the UART interface pins of the mikroBUS™ or through COM port on a personal computer. When transmitting the AT command string, a timing interval between consecutive characters should not exceed 5 seconds.

The complete documentation with in-depth explanation of each AT command and its response can be found in the ***AT Command Manual***.

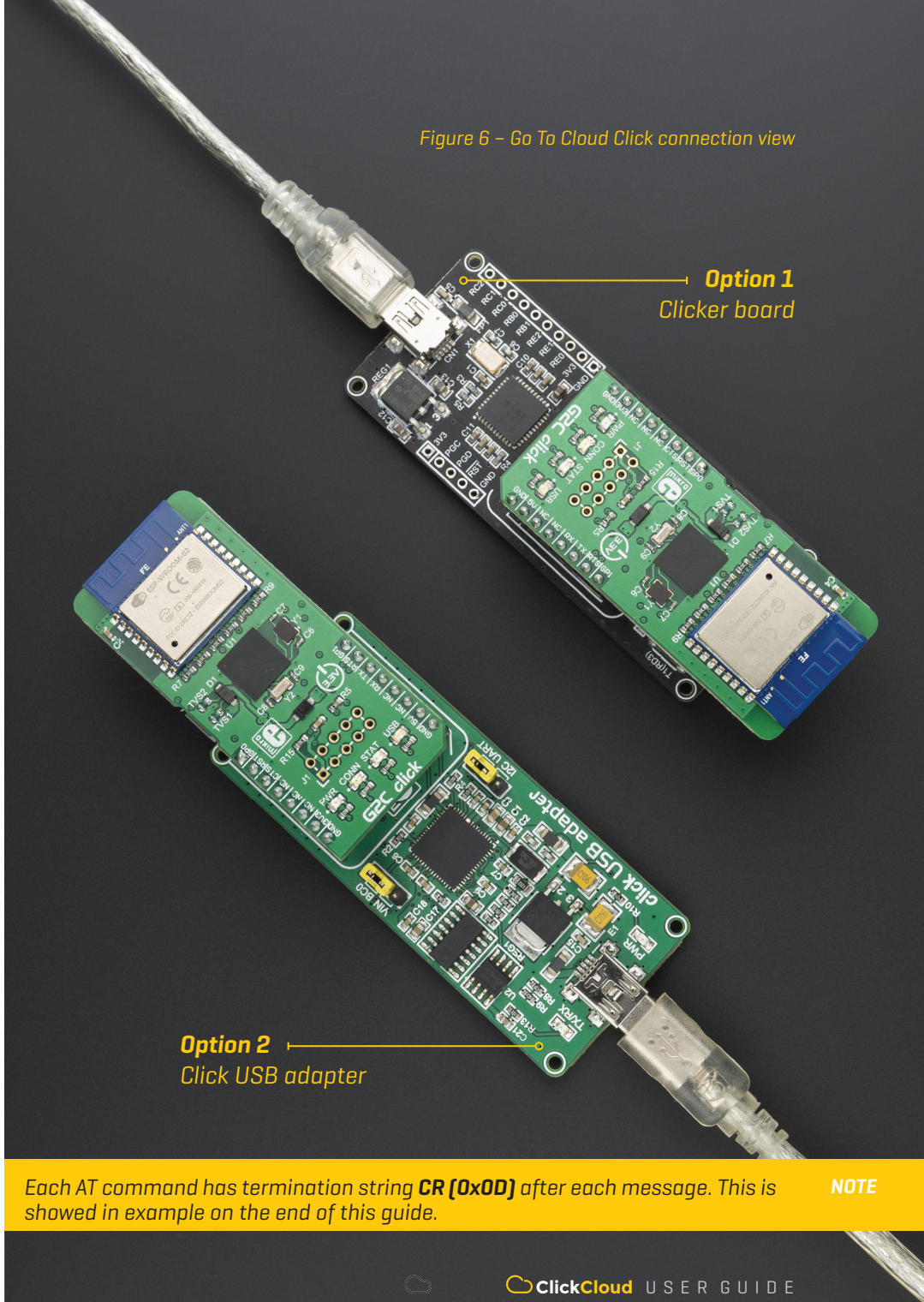

Figure 6 – Go To Cloud Click connection view

NOTE

*Each AT command has termination string **CR (0x0D)** after each message. This is showed in example on the end of this guide.*

STEP 4 / Configuring and Connecting

After selecting your hardware setup, the next step in to establish connection between the Go to Cloud Click and the Click Cloud service. It can be achieved by using the following command sequence sent via terminal application for appropriate hardware:

G2C click:

```
AT+CEN=1
AT+NWP=1
AT+NWCR="MyNetwork","MyPassword"
AT+NWC=1
```

The successful Internet connection will be indicated by the **STAT LED** on the Click board™, which will be turned on.

NOTE *"MyNetwork" and "MyPassword" strings should match SSID and password settings of your local network.*

G2C 3G click:

```
AT+CEN=1
AT+NWCR="APN","username","password"
AT+NWC=1
```

The successful Internet connection will be indicated by the **STAT LED** on the Click board™, which will be turned ON.

NOTE *"APN", "username" and "Password" strings should match your local GSM operator which you want to use.*

Final step is to connect the Go to Cloud Click to the MQTT broker service and the application server itself.

```
AT+BRCR="DeviceKey","DevicePass"
AT+BRC=1
```

After the successful connection, the **CONN LED** on the Click board™ will be turned ON. You will also notice that the device status displayed on the web interface of the Click Cloud, is changed to **Connected**.

NOTE *"DeviceKey" and "DevicePass" strings should match the DeviceID and the password which were generated during the Click Cloud device creation step (section 1.4.)*

STEP 5 / Sending Data to the Cloud

When the Go to Cloud click is connected with the service, the publication of the data is done by executing the following AT command sequence:

```
AT+DSET="G2C_SENSOR","56.8"
AT+PUB
```

By clicking on the ***Device > Details > Data*** tab, in the sensors list, we can see that the value is changed to 56.8 which we have sent using the previous command sequence.

Access to the Click Cloud and visualization of the data can be made by using ***Web*** or ***mobile app***. The Click Cloud App is available on Apple Store and Android Play for downloading.

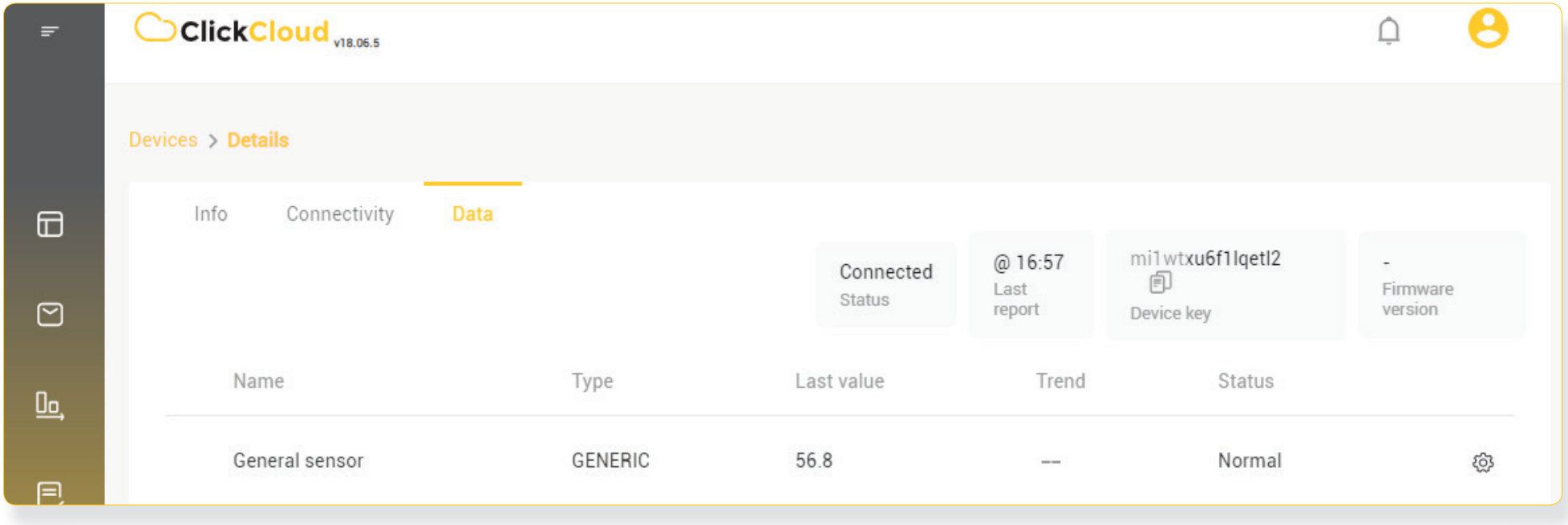

Figure 7 – Devices > Details window

STEP 6 / Sending Data from the Cloud

To demonstrate sending data from the Click Cloud to the board, we will use the simple "Hello G2C" message sent from the Click Cloud dashboard terminal. To do that, create a new Dashboard and add new widget.

1. ***Dashboards>Create dashboard*** (if you do not have it already)

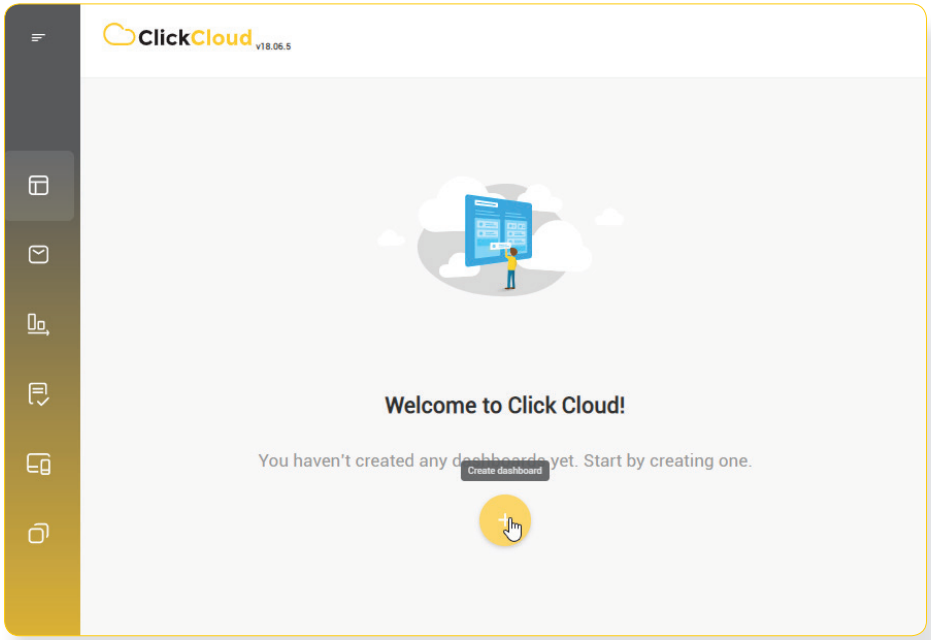

Figure 8 – Dashboard window

2. Then ***Add new widget*** on ⊕ in the lower right corner of your Dashboard page.

3. In Choose widget window select ***Actuators*** and select ***General ACT-S*** like on the page below, then press ***Save.***

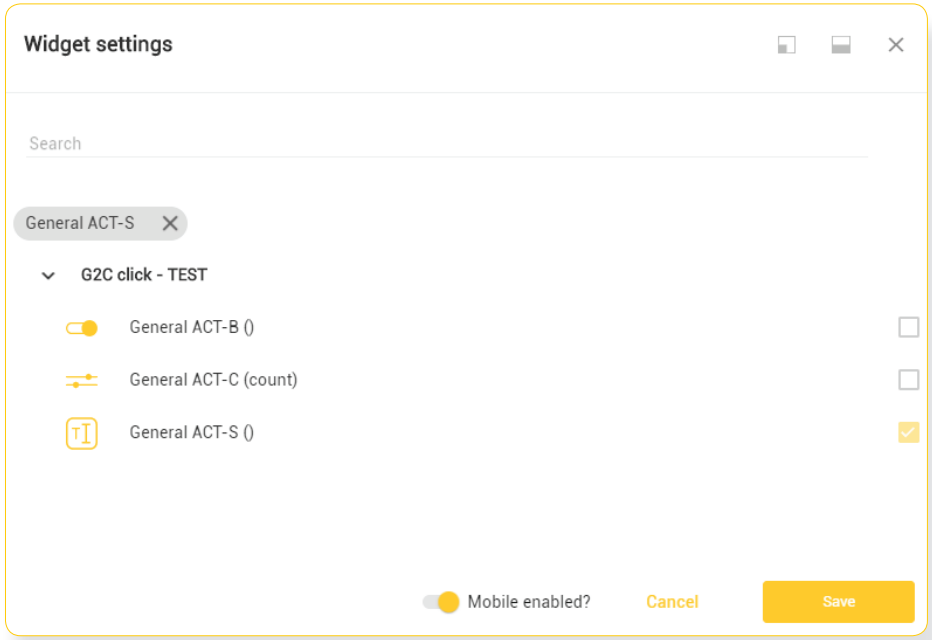

Figure 9 – Widget settings window

4. This step will create new widget on your Dashboard, exactly like this one:

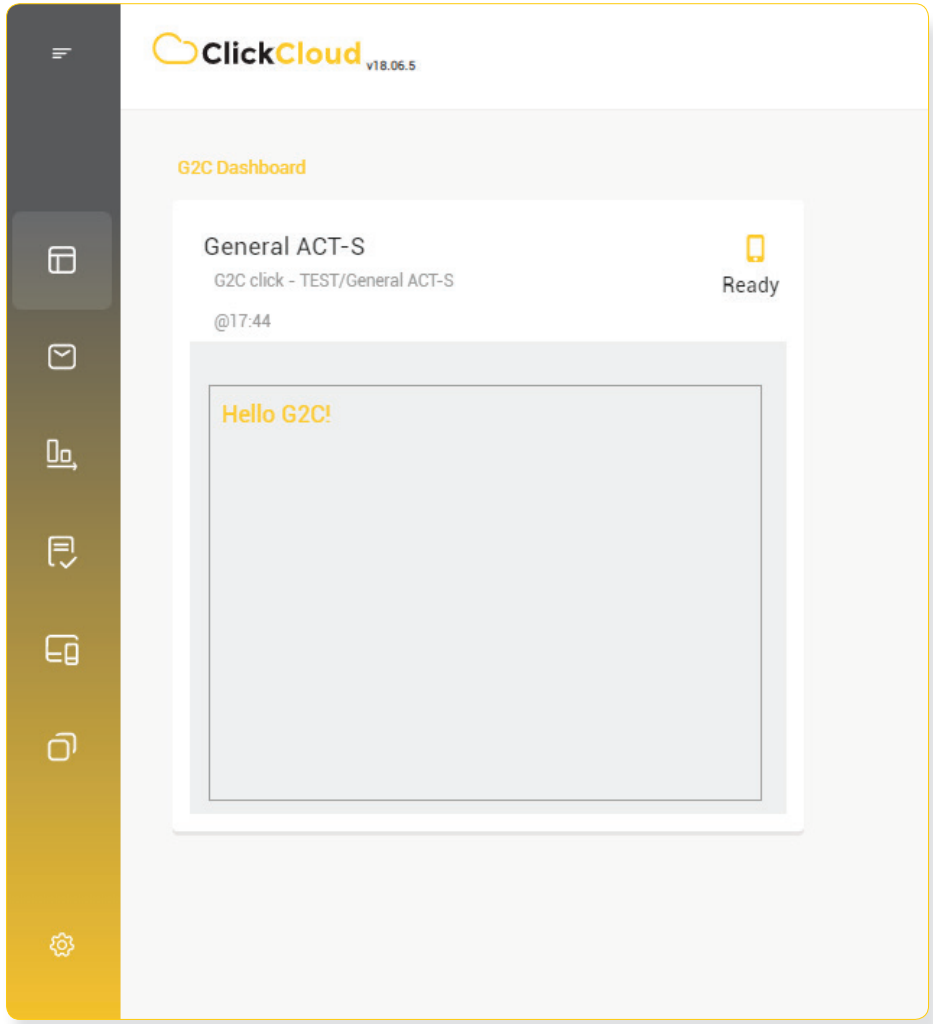

Figure 10 – Terminal Dashboard window

5. Here you can type text strings which will appear in your UART terminal after saving it, as shown in the ***Figure 11.***

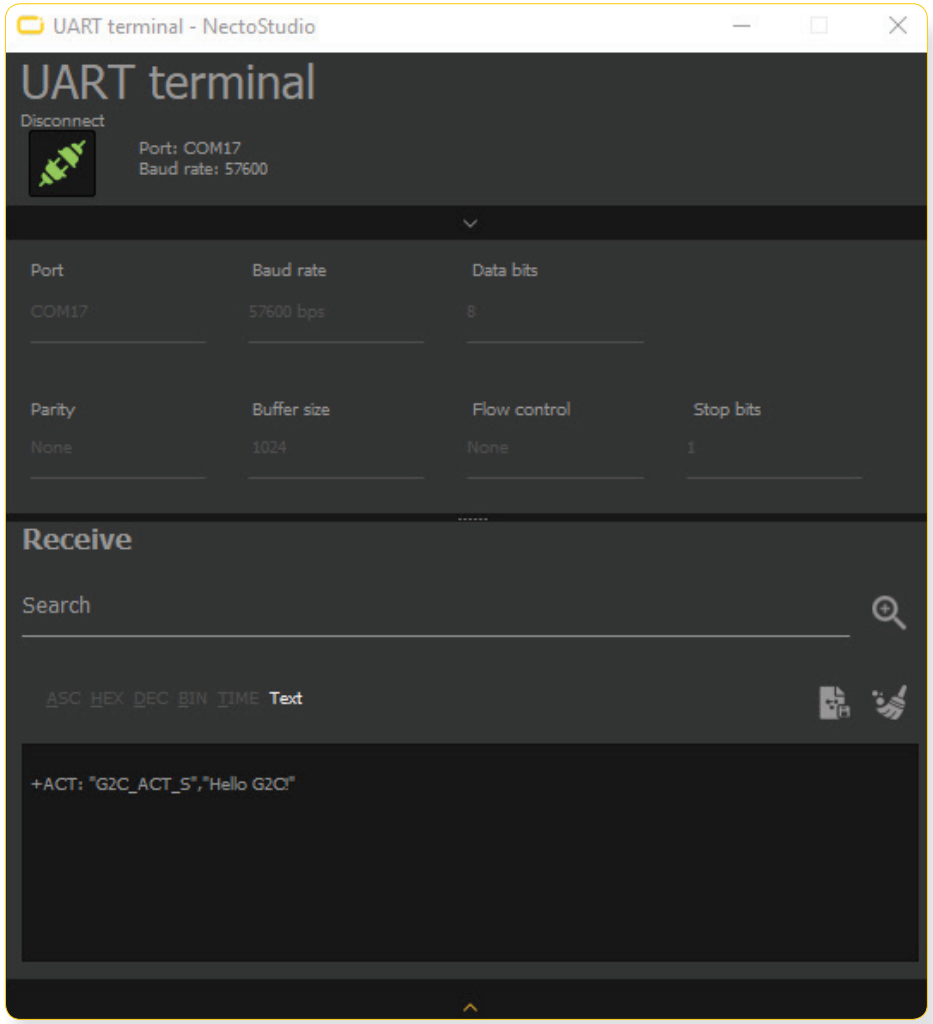

Figure 11 – UART terminal window

SIMPLE PUBLISHING EXAMPLE CODE

```
/*!
 * Device: G2C click v103
 * Board: Fusion for ARM v8
 * MCU: STM32F407ZG / 160Mhz
 *
 * LOG: USB UART (UART 6)
 * DEV_COM: MIKROBUS 1 (UART 1)
 *
 * Device key: [ vzcdgzakoco0cpfq ]
 * Device password: [ a13748a4-d19c-4ec2-bbce-d7cf15d32d04 ]
 *
*/

#include “stdint.h”

#define LOG_WRITE( x ) UART6_Write( x );
#define LOG_WRITE_TEXT( x )\
        UART6_Write_Text( x );\
        UART6_Write( 13 );\
        UART6_Write( 10 );
#define DEVICE_SEND_CMD( x )\
        UART1_Write_Text( x );\
        UART1_Write( 0x0D );

sbit DEV_RST_PIN at GPIOE_ODR.B11;
sbit DEV_CS_PIN at GPIOA_ODR.B4;

void device_init( void )
{
    // Uart init
    UART1_Init_Advanced( 57600,
                         _UART_8_BIT_DATA,
                         _UART_NOPARITY,
                         _UART_ONE_STOPBIT,
                         &_GPIO_MODULE_USART1_PB67 );

    // Interrupt init
    RXNEIE_USART1_CR1_bit = 1;
    NVIC_IntEnable( IVT_INT_USART1 );
```

```
/*!
 * Device: G2C 3G click v101
 * Board: Fusion for ARM v8
 * MCU: STM32F407ZG / 160Mhz
 *
 * LOG: USB UART (UART 6)
 * DEV_COM: MIKROBUS 1 (UART 1)
 *
 * Device key: [ vzcdgzakoco0cpfq ]
 * Device password: [ a13748a4-d19c-4ec2-bbce-d7cf15d32d04 ]
 * Operator settings: [APN]-[username]-[password]
 *
*/

#include “stdint.h”

#define LOG_WRITE( x ) UART6_Write( x );
#define LOG_WRITE_TEXT( x )\
        UART6_Write_Text( x );\
        UART6_Write( 13 );\
        UART6_Write( 10 );
#define DEVICE_SEND_CMD( x )\
        UART1_Write_Text( x );\
        UART1_Write( 0x0D );

sbit DEV_RST_PIN at GPIOE_ODR.B11;
sbit DEV_CONN_LED at GPIOA_IDR.B3;
sbit DEV_NET_LED at GPIOD_IDR.B12;

void device_init( void )
{
    // Uart init
    UART1_Init_Advanced( 57600,
                         _UART_8_BIT_DATA,
                         _UART_NOPARITY,
                         _UART_ONE_STOPBIT,
                         &_GPIO_MODULE_USART1_PB67 );
```

```
    EnableInterrupts( );

    // PIN init
    GPIO_Digital_Output( &GPIOE_BASE, _GPIO_PINMASK_11 );// RST
    GPIO_Digital_Input( &GPIOA_BASE, _GPIO_PINMASK_4 );  // CS
}

void wake_up_module( void )
{
    DEV_CS_PIN = 1;
    Delay_ms( 8000 );
}

void device_reset( void )
{
    DEV_RST_PIN = 1;
    Delay_100ms( );
    DEV_RST_PIN = 0;
    Delay_100ms( );
    Delay_100ms( );
    Delay_100ms( );
    DEV_RST_PIN = 1;
    Delay_1sec();
}

void log_init( void )
{
    UART6_Init_Advanced( 57600,
                         _UART_8_BIT_DATA,
                         _UART_NOPARITY,
                         _UART_ONE_STOPBIT,
                         &_GPIO_MODULE_USART6_PC67 );

    LOG_WRITE_TEXT( ">> System INIT <<" );
    Delay_ms( 1000 );
}

void main()
{
    Delay_ms( 500 );
    log_init( );
    device_init( );
    wake_up_module( );
    device_reset( );
    Delay_ms( 2000 );
    LOG_WRITE_TEXT( ">> Device START <<" );
    DEVICE_SEND_CMD( "AT" );
```

```
    // Interrupt init
    RXNEIE_USART1_CR1_bit = 1;
    NVIC_IntEnable( IVT_INT_USART1 );
    EnableInterrupts( );

    // PIN init
    GPIO_Digital_Output( &GPIOE_BASE, _GPIO_PINMASK_11); // RST
    GPIO_Digital_Input( &GPIOA_BASE, _GPIO_PINMASK_3); // CONN LED
    GPIO_Digital_Input( &GPIOD_BASE, _GPIO_PINMASK_12 ); //NET LED
}

void device_reset( void )
{
    DEV_RST_PIN = 1;
    Delay_100ms( );
    DEV_RST_PIN = 0;
    Delay_100ms( );
    Delay_100ms( );
    Delay_100ms( );
    DEV_RST_PIN = 1;
    Delay_ms( 8000 );
}

void log_init( void )
{
    UART6_Init_Advanced( 57600,
                         _UART_8_BIT_DATA,
                         _UART_NOPARITY,
                         _UART_ONE_STOPBIT,
                         &_GPIO_MODULE_USART6_PC67 );
    LOG_WRITE_TEXT( ">> System INIT <<" );
    Delay_ms( 1000 );
}

void main()
{
    Delay_ms( 500 );
    log_init( );
    device_init( );
    device_reset( );
    Delay_ms( 2000 );

    while( ( DEV_CONN_LED != 0 ) && ( DEV_NET_LED != 0 ) );
    LOG_WRITE_TEXT( ">> Device START <<" );
    DEVICE_SEND_CMD( "AT" );
    Delay_ms( 8000 );
    DEVICE_SEND_CMD( "AT+GMR" );
```

```
    Delay_ms( 8000 );
    DEVICE_SEND_CMD( “ATE0” );
    Delay_ms( 8000 );
    DEVICE_SEND_CMD( “AT+CEN=1” );
    Delay_ms( 8000 );
    DEVICE_SEND_CMD( “AT+NWCR=\”network_name\”,\”network_pass\””);
    Delay_ms( 8000 );
    DEVICE_SEND_CMD( “AT+NWC=1” );
    Delay_ms( 8000 );
    DEVICE_SEND_CMD( “AT+BRCR=\”device_key\”,\”device_pass\”” );
    Delay_ms( 8000 );
    DEVICE_SEND_CMD( “AT+BRC=1” );
    Delay_ms( 8000 );
    LOG_WRITE_TEXT( “>> CONNECT TO CLICK CLOUD <<” );
    DEVICE_SEND_CMD( “AT+DSET=\”G2C_SENSOR\”,\”33.3\”” );
    Delay_ms( 8000 );
    DEVICE_SEND_CMD( “AT+PUB” );
    Delay_ms( 8000 );
    LOG_WRITE_TEXT( “>> FINISH <<” );

    for ( ; ; )
    {
    }
}

void UART_RX_ISR( ) iv IVT_INT_USART1 ics ICS_AUTO
{
    char tmp;
    if( RXNE_USART1_SR_bit )

    {
        tmp = USART1_DR;
        LOG_WRITE( tmp );
    }
}
```

```
    Delay_ms( 8000 );
    DEVICE_SEND_CMD( “AT+CEN=1” );
    Delay_ms( 8000 );
    DEVICE_SEND_CMD( “AT+NWCR=\”internet\”,\”carrier\”,\”gprs\””);
    Delay_ms( 8000 );
    DEVICE_SEND_CMD( “AT+NWC=1” );
    Delay_ms( 8000 );

    while( DEV_NET_LED != 1 );
    Delay_ms( 2000 );
    DEVICE_SEND_CMD( “AT+BRCR=\”device_key\”,\”device_pass\”” );
    Delay_ms( 8000 );
    DEVICE_SEND_CMD( “AT+BRC=1” );
    Delay_ms( 8000 );

    while( DEV_CONN_LED != 1 );
    Delay_ms( 2000 );
    LOG_WRITE_TEXT( “>> CONNECT TO CLICK CLOUD <<” );
    DEVICE_SEND_CMD( “AT+DSET=\”G2C_SENSOR\”,\”33.3\”” );
    Delay_ms( 8000 );
    DEVICE_SEND_CMD( “AT+PUB” );
    Delay_ms( 8000 );
    LOG_WRITE_TEXT( “>> FINISH <<” );

    for ( ; ; )
    {
    }
}

void UART_RX_ISR( ) iv IVT_INT_USART1 ics ICS_AUTO
{
    char tmp;
    if( RXNE_USART1_SR_bit )
    {
        tmp = USART1_DR;
        LOG_WRITE( tmp );
    }
}
```

DISCLAIMER

TRADEMARKS

Click Cloud GUIDE

VERSION 2.0

Publisher
MikroElektronika Ltd.
www.mikroe.com

January, 2020.