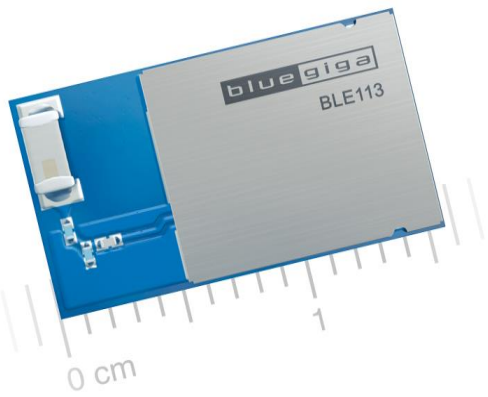
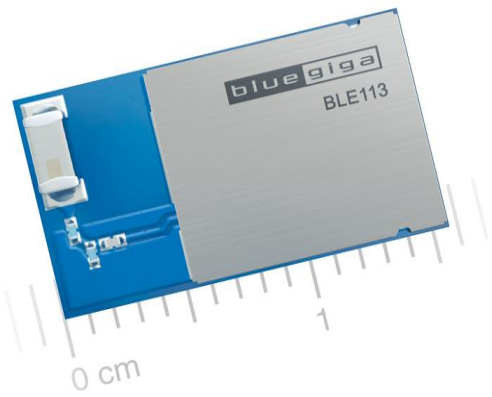
# BLE113 *Bluetooth*® Smart Module

# Table of Contents

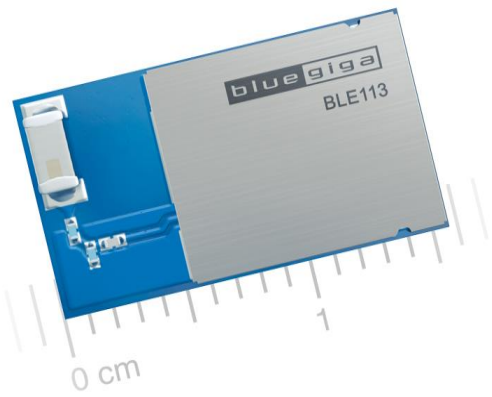


# Key Features



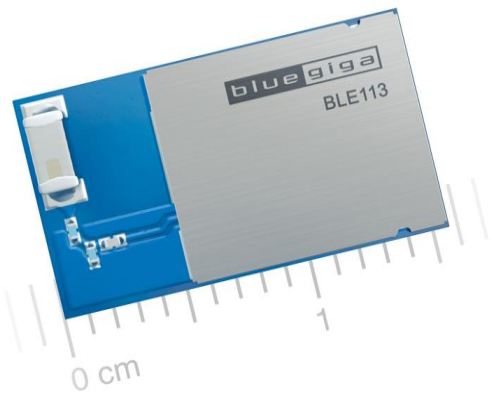

- ***Bluetooth* v.4.0, single mode compliant**
  - Supports master and slave modes
  - Up to 8 connections
- **Integrated *Bluetooth* Smart stack**
  - GAP, GATT, L2CAP and SMP
  - *Bluetooth* Smart profiles
- **Radio performance**
  - Transmit power : +0 dBm
  - Receiver sensitivity: -93 dBm
- **Ultra low current consumption**
  - Transmit: 18 mA (0 dBm)
  - Sleep mode 3: 0.5 uA
- **Flexible peripheral interfaces**
  - UART or SPI
  - I2C
  - PWM, GPIO
  - 12-bit ADC
- **Host interfaces**
  - UART
- **Programmable 8051 processor for stand-alone operation**
- ***Bluetooth*, CE, FCC, IC, South-Korea and Japan qualified**

# Benefits



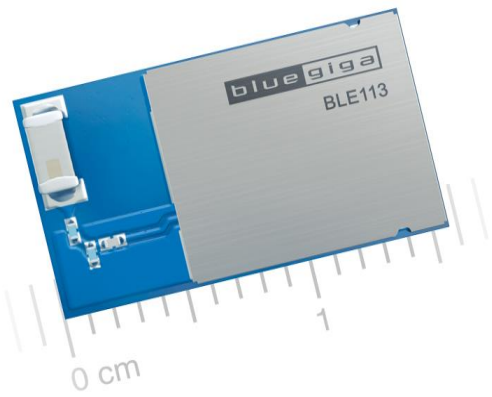

- **Fully integrated *Bluetooth* Smart solution**
  - Integrated *Bluetooth* Radio, micro controller and software stack
  - Fast time to market
  - Low development risks
- **Application hosting capabilities**
  - All application code can be executed on the BLE113
  - No need for external micro controller
  - Lower cost and smaller physical size
- **Flash based**
  - Firmware is field upgradable
  - Application data can be stored on the flash
  - Settings can be stored on the flash
- **Compact size:**
  - Dimensions: 15.75 x 9.15 x 2.1 mm
- ***Bluetooth*, CE, FCC, IC, Japan and South Korea qualified**
  - Proven interoperability
  - Minimal qualification costs

# BLE113 Overview




- ***Bluetooth* low energy radio**
  - Frequency: 2402 – 2480 MHz
  - TX power: +0 dBm
    RX sensitivity: -93 dBm
  - Modulation: GFSK
  - Symbol rate: 1 Mbps
- **Antenna**
  - Integrated ceramic chip
- **Typical line of sight range:**
  - +0 dbm: 100+ meters
  - -20 dBm: ~5 meters

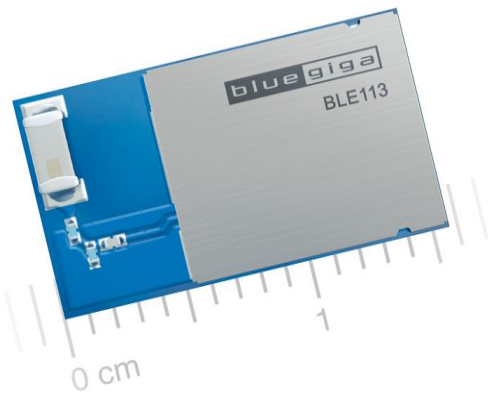
# BLE113 Overview




**A total of 21 general purpose I/O pins**

- **USART0**
  - SPI master/slave or UART 1Mbps
  - Hadware flow control
- **USART1**
  - SPI master/slave or UART 1Mbps
  - Hadware flow control
- **ADC**
  - 7 x ADC, 7-12-bit resolution
  - Internal temperature sensor
  - Internal battery monitor
- **I2C**
  - Low power, full speed I2C
- **GPIO**
  - Software programmable GPIO
- **PWM**
  - Up to 4 channel PWM

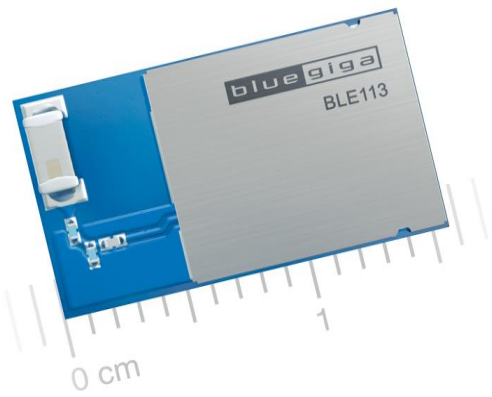
# BLE113 Overview




**A programmable 8051 microcontroller**

- **Architecture**
  - 8-bit, 8051 architecture
- **SRAM**
  - 8 kB
- **Flash**
  - 128kB
  - or 256kB (BLE113-A-256)

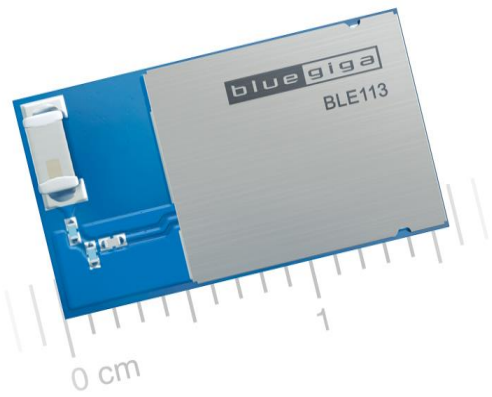
# BLE113 Overview




## Power supply and power consumption

- **General**
  - TX/RX can be as low as 14.7mA
  - Low MCU current consumption (~250uA/MHz)
  - Extremely low power sleep modes – as low as 0.5uA
- **Optimized for coin cell CR2032**
  - Quick start-up – minimize duration of peak current consumption
  - Minimum operating voltage of 2.0 V provides good resistance to dips in voltage supply
  - Architecture allows 8051 core to operate independently from the radio keeping peak current as small as possible
- **Good for alkaline as well**
  - Operating voltage range of 2.0 – 3.6 V matches dual AA

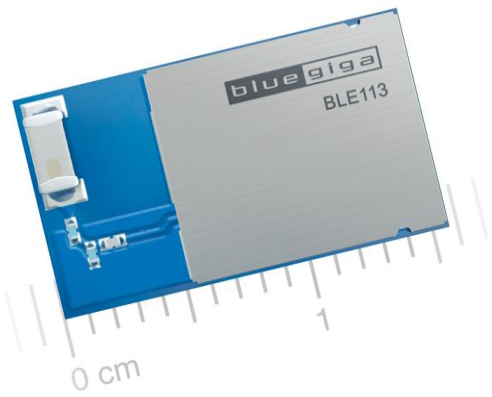
# BLE113 Overview




## BLE113 current consumption

- **TX peak**
  - 18.2 mA (0 dBm)
  - 14.3 mA (with DC/DC)
- **RX peak**
  - 17.9 mA
  - 14.7 mA (with DC/DC)
- **Sleep modes:**
  - 270 uA (power mode 1)
  - 1 uA (power mode 2)
  - 0.5 uA (power mode 3)

# BLE112 vs. BLE113

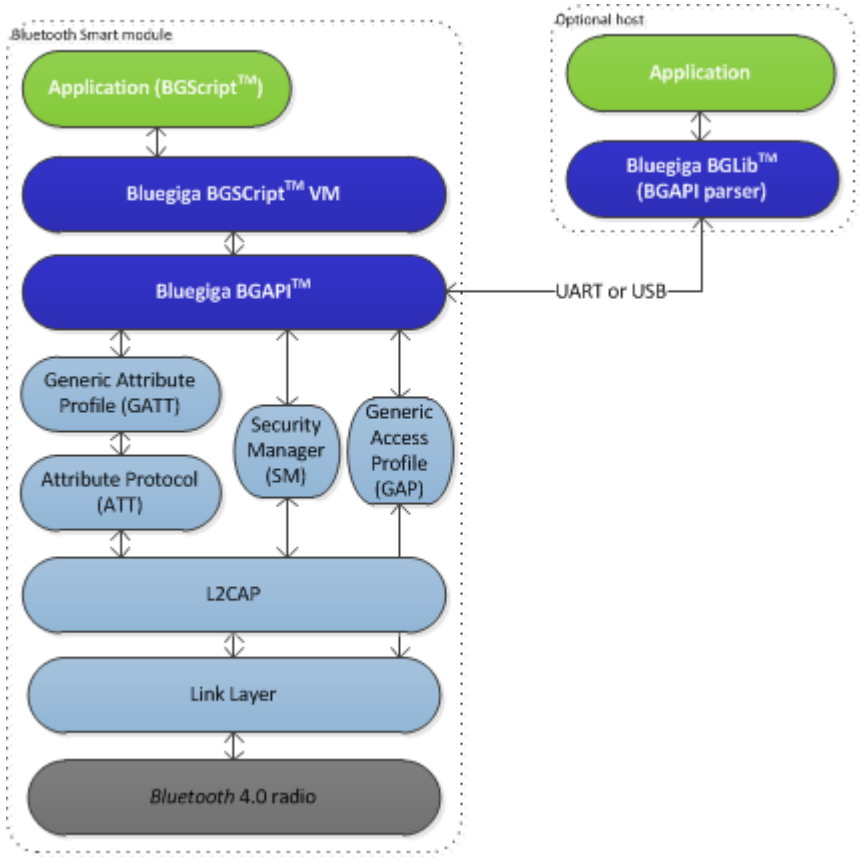



- **TX power**
  - BLE112 3 dBm
  - BLE113 0 dBm
- **Current consumption**
  - BLE112 30 mA (-2 dBm)
  - BLE113 18.2 mA (0 dBm)
- **Physical size**
  - BLE112 18 x 12 x 2.3 mm
  - BLE113 15.75 x 9.15 x 2.1 mm
- **Interfaces**
  - BLE113 lacks USB, but has a hardware I2C instead

# *Bluetooth*® Smart Software

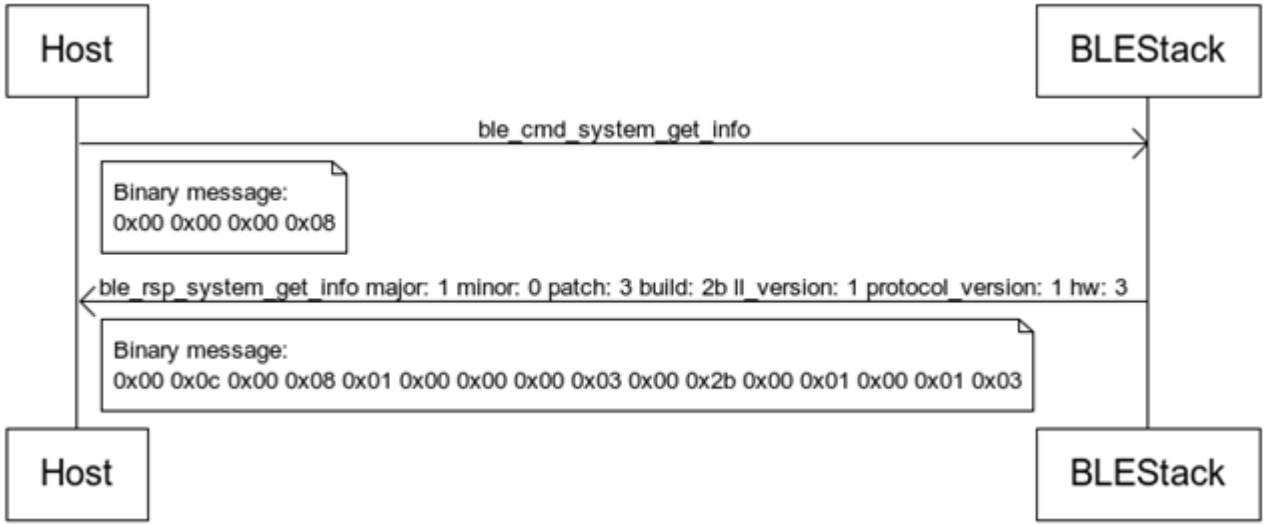
# *Bluetooth* Smart Software

- ***Bluetooth v.4.0*, single mode compliant**
  - Supports master and slave modes
  - Up to 8 simultaneous connections
- **Implements all *Bluetooth* Smart functionality**
  - GAP, L2CAP, ATT, GATT
  - Security manager: bonding, encryption
  - *Bluetooth* Smart profiles
- **Simple API for external host processors**
  - BGAPI™ : A simple protocol over UART or USB interfaces
  - BGLib™ : A C library for host processors implementing BGAPI
- **Supports standalone applications as well**
  - BGScript™ : A simple scripting language for writing applications
  - Native C application development with IAR Embedded Workbench
  - **No separate host needed**
- **Over-the-Air firmware upgrade support**
- **Blutoooth Smart Profile Toolkit™**
  - XML based development tool for Bluetooth Smat profiles
  - Fast and simple profile development
- **Small memory requirements**
  - ~4-6 kB RAM
  - ~60-90 kB flash (depending of used features/profiles)
- ***Bluetooth* qualified**




Bluegiga *Bluetooth*® Smart Software

- **BGAPI™ protocol** : A simple binary command, response and event protocol between the host and the stack
  - Used when a separate host (MCU) is used to control BLE113 over UART or USB
  - Very small memory requirements size requirement and low implementation overhead



# *Bluetooth* Smart Software

- **BGLib™ library** : A portable ANSI C library, which implements the BGAPI protocol
  - Easy to port to various architectures such as : ARM Cortex, PIC16/32 etc.
  - Uses fuction–call back architecture

**C Functions**

```
/* Function */
void ble_cmd_gap_connect_direct(
    bd_addr address ,
    uint8 addr_type ,
    uint16 conn_interval_min ,
    uint16 conn_interval_max ,
    uint16 timeout
);

/* Callback */
void ble_rsp_gap_connect_direct(
    uint16 result ,
    uint8 conn
);
```

# *Bluetooth* Smart Software

- **BGScript™ scripting language** : A very simple BASIC-like application scripting language
  - Used when applications are implemented on the BLE113's 8051 controller
  - Enables very fast application development and allows programs to be executed directly on the BLE113 without the need of an external MCU

```
# System boot event listener : Executed when BLE112 is started
event system_boot(major ,minor ,patch ,build ,ll_version ,protocol_version ,hw )


    # Configure ADV interval to 1000ms and start advertisements an all channels
    call gap_set_adv_parameters(1600, 1600, 7)

    # Start generic advertisement and enable connections
    call gap_set_mode(2,2)

   #Start a continuous software timer, which generates interrupts every 1000ms
   call hardware_set_soft_timer(32768, 1, 0)
end
```

# *Bluetooth* Smart Software

- **Why to use BGScript™?**

- **Very simple to use**
  - Fast development of simple *Bluetooth* Smart applications
  - Examples: Pairing, simple user interfaces, simple sensors

- **Free software development tools**
  - Code developed with any text or source code editor
  - Code compiled with Bluegiga's free compiler

- **Several example scripts available**
  - Heart Rate sensor
  - Proximity reporter
  - FindMe tag
  - Medical devices such as blood glucose

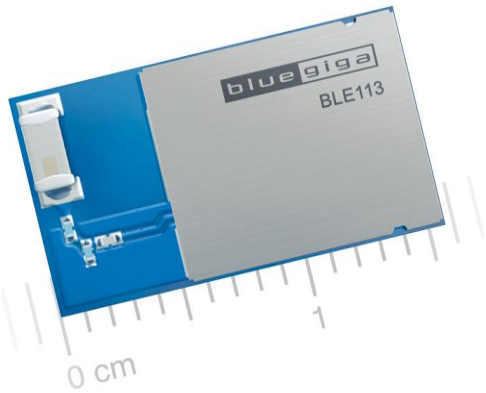
- **Cuts out the need for external MCU**
  - Reduced product eBoM
  - Smaller footprint
  - Faster time-to-market

# *Bluetooth* Smart Software

- ***Bluetooth* Smart Profile Toolkit™**: A tool for creating *Bluetooth* Smart profiles
  - *Bluetooth* Smart profiles are very simple
  - Can be describes with a single file of XML
  - Profile toolkit is a Simple description language of *Bluetooth* Smart Profiles
- **Several example profiles and services available**
  - Heart Rate Sensor
  - Proximity Reporter
  - FindMe
  - Blood glucose

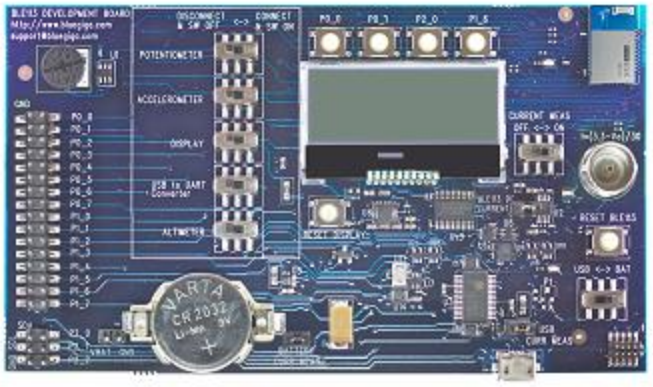
```
<?xml version="1.0" encoding="UTF-8" ?>
- <configuration>
  + <service>
  - <service>
      <uuid>3a00</uuid>
      <description>Heartrate Service</description>
    - <characteristic id="heartrate">
      - <properties>
          <read />
          <notify />
        </properties>
        <uuid>3a01</uuid>
        <value type="UINT8" />
        <description>Beats per minute</description>
      </characteristic>
    - <characteristic id="rr_interval">
      + <properties>
        <uuid>3a02</uuid>
        <value type="UINT16" />
        <description>R-R Interval</description>
      </characteristic>
    - <characteristic>
        <uuid>3a03</uuid>
      + <properties>
        <value type="SFLOAT" unit="kJ" />
        <description>Energy Expended</description>
      </characteristic>
    - <characteristic>
        <uuid>3a04</uuid>
      + <properties>
        <value type="UINT8" />
        <description>Sensor Status</description>
      </characteristic>
    + <characteristic type="aggregate">
    </service>
</configuration>
```

# Certifications




- ***Bluetooth* 4.0**
  - BLE113: Controller subsytem
  - Software : Host subsystem
- **CE**
  - EN300328
  - EN301489-1/17
  - EN60950-1
- **FCC**
  - Part 15C modular approval
- **Industry Canada**
  - IC modular certification
- **South Korea**
  - KCC certification
- **Japan**
  - ARIB-STD-66

# Development Tools



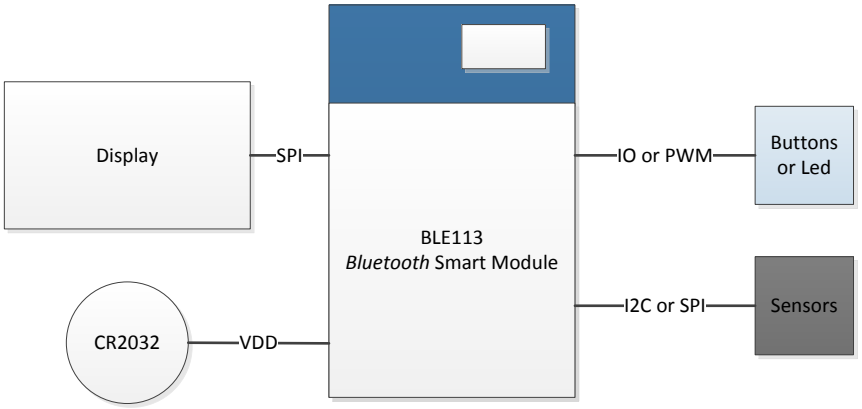
- **BLE113 Development Kit**
  - BLE113-A
  - Display
  - On-board accelerometer
  - On-board altimeter
  - Potentiometer
  - CR2032 battery holder
  - USB and RS232 interfaces
  - On-board firmware programming
  - Current measurement point
  - External DC/DC converter
  - I/O headers

  + External SPI flash board (for OTA)
  + BLED112 USB dongle
  + 2 x BLE113-A modules

- ***Bluetooth* Smart SDK**
  - BGAPI™ documentation
  - BGScript™ development tools
  - BGLib™ source code
  - Profile Toolkit™
  - BGScript and BGLib examples
  - Profile examples
  - Documentation
  - iOS and Android example applications

# Use Cases

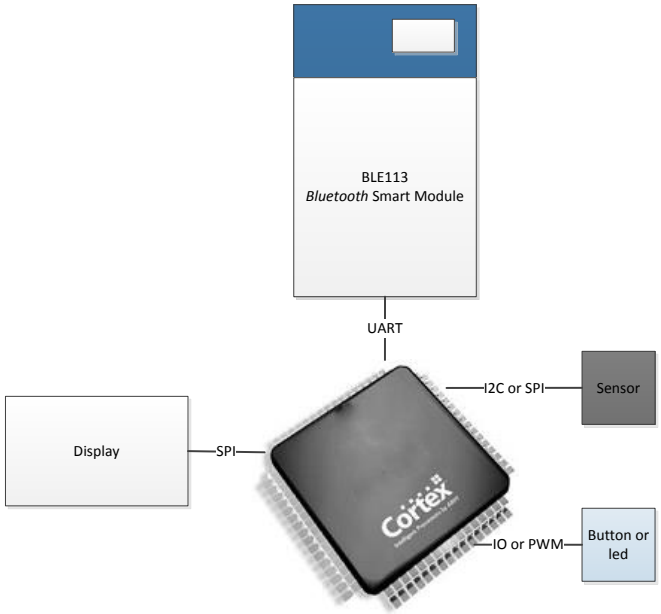
- **Standalone architecture**: No separate host processor
  - Sensors and peripherals are directly connected to the BLE113 via the IO interfaces
  - Application executed on the on-board 8051
  - Application developed with BGScript™ or ANSI C and services and profiles with Profile Toolkit™




**Applications:** sport and fitness, medical and health care, smart energy, home automation, security, proximity and precence etc.

# Use Cases

- **Hosted architecture**: A separate MCU is used
  - Sensors and peripherals are directly connected to the MCU via the IO interfaces
  - BLE113 connected to the MCU via UART or USB
  - Application developerd to the MCU and interfacing to BLE113 done using BGAPI™ protocol (BGLib™ can be used on the host)
  - Profile developed with Profile Toolkit™



# Thank You