THALES

EXSx2 ThreadX application

Example application how to combine SMS with GPIO functionality

Introduction

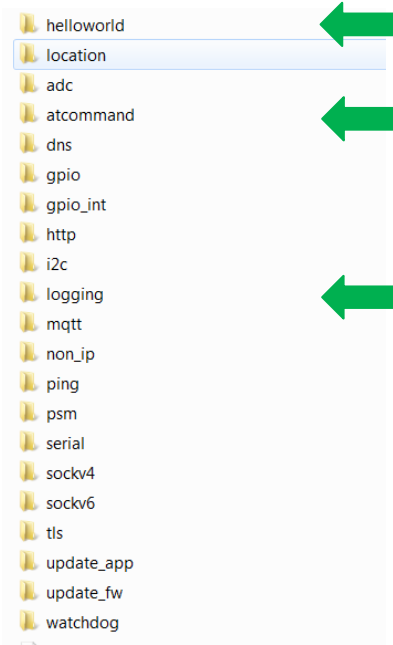
The example ThreadX application showcases how to combine the module's SMS feature with the GPIO functionality

Two scenarios are implemented

- 1st scenario (Remote Reset):
  - Application listens for incoming SMS and toggles after arrival a pre-defined GPIO
  - In case the customer application becomes unresponsive, an SMS can recover/reset units remotely. That is particularly handy in scenarios where units are difficult to access
- 2nd scenario (Theft protection/alarm system):
  - Application sends an emergency via SMS to a pre-defined mobile number in case a specific GPIO is toggled

EXSx2 SDK code examples

The EXSx2 SDK provides heaps of code examples which were utilized to develop the application



…getting familiar with the tooling

…helpful high level functions to talk with the module via AT command interface

…provides heaps of print functions for debugging purposes

SDK/examples

Details 1st scenario

The application retrieves every 5s information if a new SMS has been received. The SMS is expected in the following format:

<device_type>,<action>

<device_type>

- **MCU** : application disregards the message but does not deletes it so that the MCU can process it
- **EP** : application processes the message and deletes it afterwards

<action>

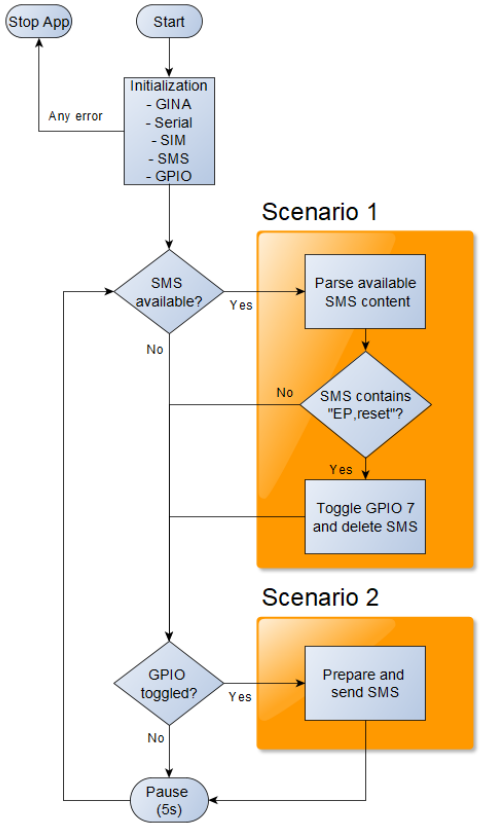
- **reset** : application toggles GPIO7 (for this demo I have connected an LED to this GPIO)
- every other format is disregarded, the SMS will be deleted automatically by the application

Detailed implementation 2nd scenario

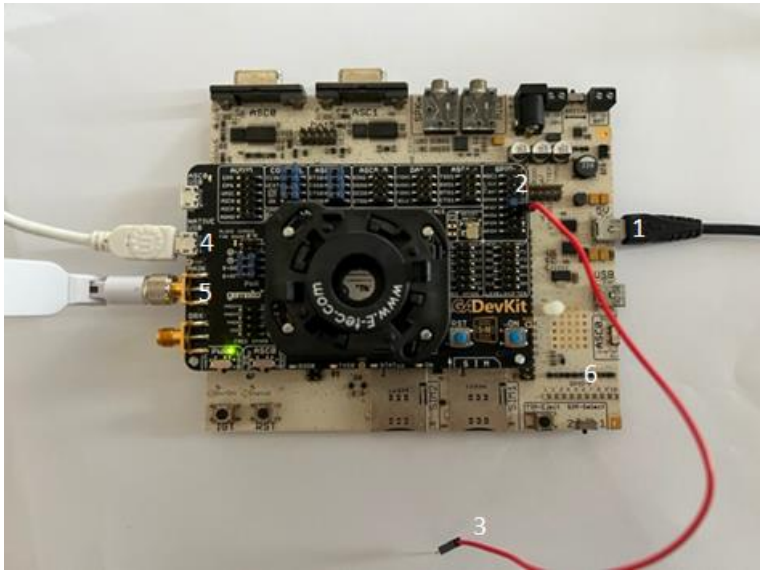
- The application has got a callback function that increments a counter if a rising edge is detected on GPIO6. The application checks the counter value every 5s. For all values greater 0 the module sends an alert via SMS to a pre-defined phone number.
- Important: The application reads in a config file at start up. The file needs to be located in the same folder as the application itself as textfile with the name config.txt
- The config file can be downloaded to the module via a python tool as follows:

```
python fs.py download A:/config.txt
```

Flowchart application



Hardware setup



1. Power supply for DSB mini
2. Jumper for GPIO7 need to be placed
3. Jumper cable connected to GPIO6 (if connected to GND, emergency SMS is sent)
4. Power supply LGA DevKit plus communication interface
5. Main's antenna
6. LED bar – LED connected to GPIO7 will blink

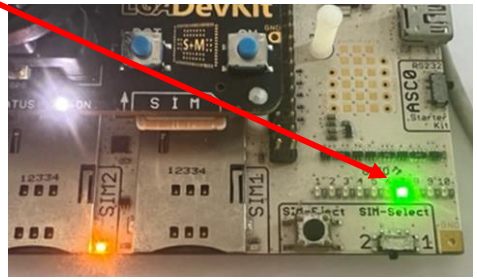
Example run

Alert got triggered, app sends SMS to mobile phone

```
COM21
Cinterion Logging Tool +++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Start logging on dedicated logging port (COM21,115200) at 2021-11-29 17:43:52... (use Control-C to exit)
Start Program!
Initialization completed successfully!
Alert SMS sent!
##################################################
Incoming message 0:
Index: 0
Phone Number: [redacted]
Message: EP:reset
##################################################
##################################################
Incoming message 0:
Index: 0
Phone Number: [redacted]
Message: EP:reset
##################################################
##################################################
Incoming message 0:
Index: 0
Phone Number: [redacted]
Message: MCU:hallo
##################################################
```

SMS with correct format received, GPIO toggled

SMS intended for MCU, SMS is kept



THANK YOU !