KU LEUVEN

Software for Real-Time and Embedded Systems | B-KUL-H04L2a-2021

Project: Time-Synchronized Embedded Device

Arne Wouters, r0632965

November 12, 2020

Contents

1 The system

The system consists of 4 tasks:

- LoRaReceiver
- DatabaseManager
- LoRaSender
- CommandManager

There are 2 queues, the *DatabaseQueue* and the *LoRaSenderQueue*. We also have a semaphore that is required for every access to EEPROM.

1.1 LoRaReceiver

This task receives and reads incoming beacosn. Packets with a size smaller than 5 will be ignored because we assume the message of the beacon has a length of atleast 5 characters. The system can handle packets with more characters as long as they follow the format, that the first 4 characters are the gateway ID and other characters are the time until the next beacon transmission. Once a beacon is received, the time until the next beacon is put in the *DatabaseQueue* and the *firstPackageReceived* flag is set to *true* if it was false. If low-power operation mode is enabled, the *DatabaseManager* task is resumed and the *LoRaReceiver* will go to sleep for x seconds, where x is the time until the next beacon. When the *LoRaReceiver* processed 20 beacons, it will put the system in ultra low-power mode.

1.2 DatabaseManager

This task reads values from the *DatabaseQueue*. If low-power operation mode is enabled, it will suspend itself after every loop and only resume when the function *vTaskResume()* is called in the *LoRaReceiver*. The item in the queue gets grabbed and the *DatabaseManager* takes or waits for the semaphore. We get the temperature from the temperature sensor and then we write the temperature and the data from the queue to the database. After the write we update the address that refers to the address behind the last write in the EEPROM. We also store this value in the first 2 bytes of the EEPROM so we can resume adding to the database at the end without overwriting the old data when the arduino is retarted. When the EEPROM is full, we reset and start writing from the first address again. After writing to the EEPROM, the semaphore is returned. The temperature is put into the *LoRaSenderQueue* and the *LoRaSender* task is resumed.

1.3 LoRaSender

This task reads values from the *LoRaSenderQueue*. If low-power operation mode is enabled, it will suspend itself after every loop and only resume when the function *vTaskResume()* is called in the *DatabaseManager*. This task reads the temperature value from the queue, makes a packet and sends it back to the gateway.

1.4 CommandManager

This task reads commands from the serial port and prints output to the serial port at a rate of 9600 baud. There are 4 commands supported (the input for these functions is just the numbers 1, 2, 3 and 4):

1. read the latest temperature value and beacon details from database and print the output to the serial port
2. read all temperature values and beacon details from database and print the output to the serial port
3. enable low power operation mode
4. reset the database

2 Varying real-time constraint

3 Synchronization

4 Power consumption