Universität
Bremen

MaMBA Project: Measuring gas concentrations in an extraterrestrial habitat

Softwareconcept for measuring and sending from and to different peripherials using LoRa

1 General idea

The software thats to be implemented should be able to receive measurement values from different peripherials and send those values to a given network. Selected components are not disscussed here, where only the requirements for the software are disscussed. However a short briefing of relevant components of the system should be given.
The central unit on the measuring side is a microcontroller (MCU = **M**icro**C**ontrolling **U**nit) containing an ESP32. Connected to the MCU are several peripherials via an I^{2}C bussystem, an SPI bus or via the one-wire-protocol. For communication is a LoRa (**L**ong**R**ange) transceiver used, with an onboard SX1276 chip. In general the MCU requests data or a measurements from the different peripherials, receives the requestet data and stores it. To publish the data into the previous mentioned given network the MCU sends data via the SPI bus to the IC with the onboard SX 1276, which then sends out data to a single channel LoRa gateway. This gateway publishes with use of the MQTT protocol to a MQTT broker, who is integrated in the given network. The network stores, displays and forwards received data as implemented, the exact process is not relevant for the purpose of disscussing the system until the MQTT broker. For programming *Micropython* is used.

2 Tasks

After the short overview of the existing system some tasks that are to be implemented are already quiet clear: Requesting, receiving and sending data are the main tasks of the MCU. In addittion a comprehensive error handling should be implemented.

2.1 measuring and error handling

Starting with the boot of the MCU, garbage collection is to be enabled. Later in the process the garbage collection keeps the memory allocation clean and prevents any memory overflows. After that a connection to all the peripherials connected via I^2C or SPI should be established. As explained later the peripherials connected via the one-wire-protocoll only need to be called when data is requested. In the different libraries the exact measuring methods are defined (libraries are explained from page 2). Hence the peripherials only need to be instatiated with an object and the corresponding measurement/instantiation methods have to be called. For constant measuring these measurement methods are to be called in an infinite loop. The received measurement values from each iteration can be send or kept in a local memory connected to the MCU.

Problems can occur if one of the peripherials gets damaged or is disconnected from the system. In that case the MCU has to check in each iteration if there is a replacement or reconnection (which is the same from a software point of view) and start requesting measurement values again. If the connection to the transceiver fails, the data is to be stored in the local memory, so that access later on is secured. As for the communication, measuring with the peripherials those are all error cases handled. The whole LoRa communication is disscussed later (page 3).

2.2 libraries

In this section the peripherials are disscussed in more detail. The O_2 and CO sensor contain an I^2C Interface and a ROM. Measurement is started if the master generates the corresponding starting condition, the addresses are hardwired by the manufacturer. The measurement value received by the MCU is in steps, which has to be converted into huma readable decimal numbers, the library takes care of that.
The MCU is using proprietary libraries for measuring pressure and CO_2, exact specifications can be obtained from the datasheets. Both the pressure and CO $_2$ Sensor contain an I^2C Interface and Registers. The temperature and humidity Sensors contain a one-wire-protocol interface. In order to receive measurement values the MCU has to send the corresponding order and can the read the values. In Micropython there is already a *DHT22* (**D**igital **H**umidity and **T**emperature) library implemented, which can be used with the Sensors.

3 LoRa