

NAV projekt 2019/2020

Demonstrační aplikace s použitím Hexiwear a Raspberry Pi

Autor : Pavel Koupý

1. Zadaní

S využitím dostupných knihoven a nezbytných vývojových nástrojů vytvořte jednoduchou demonstrační aplikaci na platformě Hexiwear. Mezi klíčové prvky této vývojové platformy patří mikrokontrolér NXP MK64F na bázi jádra Cortex-M4, bezdrátový komunikační SoC modul NXP MKW40Z pro standard BLE v4.1 a pestrá škála různých senzorů (např. tlakoměr, gyroskop, akcelerometr, srdeční tep či intenzita osvětlení).

Cílem projektu je tedy demonstrační aplikace s použitím platformy Hexiwear. V rámci projektu jsem využil další platformy a to Raspberry Pi. Tento mikropočítač má integrované rozhraní Bluetooth a Wi-Fi a je ho tak použito k vytvoření MQTT brokera a jako gateway pro hodinky Hexiwear, které umožňují komunikaci pouze pomocí Bluetooth. Tedy Raspberry Pi tvoří most z BT do MQTT, který umožňuje čtení údajů např. Pomocí Wifi nebo Ethernetu. Navíc Raspberry Pi vytváří statistiky z údajů dostupných z platformy Hexiwear a tyto hodnoty umožňuje snadněji interpretovat v rámci času za pomoci grafů a historie údajů dostupných ve formě jednoduchého webové rozhraní.

2. Implementace

V rámci implementace je použito Raspberry Pi 3 B s malým dotykovým TFT displejem, který slouží jako brána pro Hexiwear platformu a umožňuje vizualizaci a archivaci dat z hodinek Hexiwear.



Obrázek 1: Demonstrační aplikace

2.1. Přemostění Bluetooth komunikace

K přemostění komunikace mezi Bluetooth BLE protokolem a MQTT protokolem umožňuje rozšířit dostupnost Hexiwearu v rámci jakékoliv sítě, například Wi-Fi či Ethernetu.

```
BT_UUID_HEXIWEAR_BAT_SERVICE_LEVEL      = "00002A19-0000-1000-8000-00805f9b34fb";
BT_UUID_HEXIWEAR_APP_SERVICE_APPMODE     = "00002041-0000-1000-8000-00805f9b34fb";
BT_UUID_HEXIWEAR_MOTION_SERVICE_ACC       = "00002001-0000-1000-8000-00805f9b34fb";
BT_UUID_HEXIWEAR_MOTION_SERVICE_GYRO      = "00002002-0000-1000-8000-00805f9b34fb";
BT_UUID_HEXIWEAR_MOTION_SERVICE_MAG       = "00002003-0000-1000-8000-00805f9b34fb";
BT_UUID_HEXIWEAR_WEATHER_SERVICE_AMB      = "00002011-0000-1000-8000-00805f9b34fb";
BT_UUID_HEXIWEAR_WEATHER_SERVICE_TEMP     = "00002012-0000-1000-8000-00805f9b34fb";
BT_UUID_HEXIWEAR_WEATHER_SERVICE_HUMID    = "00002013-0000-1000-8000-00805f9b34fb";
BT_UUID_HEXIWEAR_WEATHER_SERVICE_PRESS    = "00002014-0000-1000-8000-00805f9b34fb";
BT_UUID_HEXIWEAR_HEALTH_SERVICE_HR        = "00002021-0000-1000-8000-00805f9b34fb";
BT_UUID_HEXIWEAR_HEALTH_SERVICE_STEPS     = "00002022-0000-1000-8000-00805f9b34fb";
BT_UUID_HEXIWEAR_HEALTH_SERVICE_CAL       = "00002023-0000-1000-8000-00805f9b34fb";
BT_UUID_HEXIWEAR_ALERT_SERVICE_IN         = "00002031-0000-1000-8000-00805f9b34fb";
BT_UUID_HEXIWEAR_ALERT_SERVICE_OUT        = "00002032-0000-1000-8000-00805f9b34fb";
```

Obrázek 2: UUID jednotlivých datových položek

K účelu převodu mezi těmito protokoly slouží aplikace v C++, která využívá GATTLIB knihovny pro komunikaci s BT modulem přes DBUS a MQTT klienta pomocí, kterého publikuje data z BT. V dokumentaci k Hexiwear platformě je možné dohledat potřebné UUID k jednotlivým datovým položkám. Přemostění je sestaveno pomocí CMAKE a po jeho spuštění je zapotřebí zapnout archivaci do DB, popis této části je uveden dále.

2.2. MQTT Broker

Na operačním systému Raspbian Buster, který je nasitalovaný na Raspberry Pi, je nutné pro správnou funkci nainstalovat balíčky Mosquitopp, Gattlib a MQTT broker. Po spuštění programu pro přemostění se začnou posílat data a vytvoří se v obrázku níže uvedené topic, ze kterých je možné číst data ve formě řetězce s hexadecimálním číslem.

```
MQTT_HEXIWEAR_BAT_SERVICE_LEVEL          = "hexiwear/2A19/battery_service/battery_level";
MQTT_HEXIWEAR_APP_SERVICE_APPMODE         = "hexiwear/2041/appmode_service/app";
MQTT_HEXIWEAR_MOTION_SERVICE_ACC           = "hexiwear/2001/motion_service/accelerometr";
MQTT_HEXIWEAR_MOTION_SERVICE_GYRO          = "hexiwear/2002/motion_service/gyroscope";
MQTT_HEXIWEAR_MOTION_SERVICE_MAG           = "hexiwear/2003/motion_service/magnetometr";
MQTT_HEXIWEAR_WEATHER_SERVICE_AMB          = "hexiwear/2011/weather_service/ambient_light";
MQTT_HEXIWEAR_WEATHER_SERVICE_TEMP         = "hexiwear/2012/weather_service/temperature";
MQTT_HEXIWEAR_WEATHER_SERVICE_HUMID        = "hexiwear/2013/weather_service/humidity";
MQTT_HEXIWEAR_WEATHER_SERVICE_PRESS        = "hexiwear/2014/weather_service/pressure";
MQTT_HEXIWEAR_HEALTH_SERVICE_HR            = "hexiwear/2021/health_service/heart_rate";
MQTT_HEXIWEAR_HEALTH_SERVICE_STEPS         = "hexiwear/2022/health_service/steps";
MQTT_HEXIWEAR_HEALTH_SERVICE_CAL           = "hexiwear/2023/health_service/calories";
MQTT_HEXIWEAR_ALERT_SERVICE_IN             = "hexiwear/2031/alert_service/alert_in";
MQTT_HEXIWEAR_ALERT_SERVICE_OUT            = "hexiwear/2032/alert_service/alert_out";
```

Obrázek 3: MQTT topics

2.3. Archivace údajů v databázi

K archivaci údajů z MQTT topiců slouží, skript napsaný v pythonu, který je převzat z projektu do SEN 2018/19, který vyčítá zadané topic a vkládá je do MySQL databáze MariaDB. Data jsou

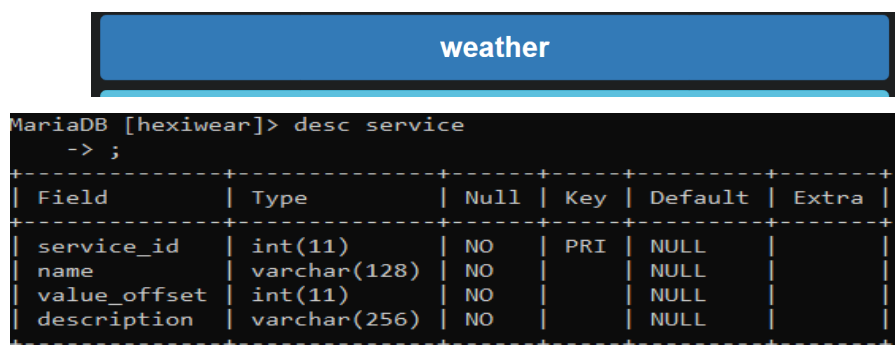
```
MariaDB [hexiwear]> desc data;
+-----+-----+-----+-----+-----+-----+
| Field | Type   | Null | Key | Default | Extra           |
+-----+-----+-----+-----+-----+-----+
| id     | int(11) | NO   | PRI | NULL    | auto_increment |
| device_id | int(11) | NO   |     | NULL    |                 |
| service_id | int(11) | NO   | MUL | NULL    |                 |
| time   | timestamp | YES  |     | current_timestamp() |                 |
| value  | varchar(128) | YES  |     | NULL    |                 |
+-----+-----+-----+-----+-----+-----+
```

Obrázek 4: Struktura tabulky s daty

ukládána jednotně do tabulky data, kde typ dat rozhoduje tabulka services, která předepisuje jednotlivé datové položky pomocí jejich UUID , a také obsahuje popis datové položky, offset položky pokud je definován. Tabulka dat tak obsahuje záznamy z jednotlivých services v intervalu 30 vteřin , kdy je jsou vyčtena data z Hexiwearu a jsou publikována na MQTT topiců.

2.4. Vizualizace

Vizualizace je tvořena jednoduchým dashboardem vytvořeným pomocí bootstrap snippetu spuštěným na webovém serveru na Raspberry Pi. Jednotlivé aplikace nabízí konkrétní servis Hexiwear dle jmenovky na tlačítku v úvodním menu.

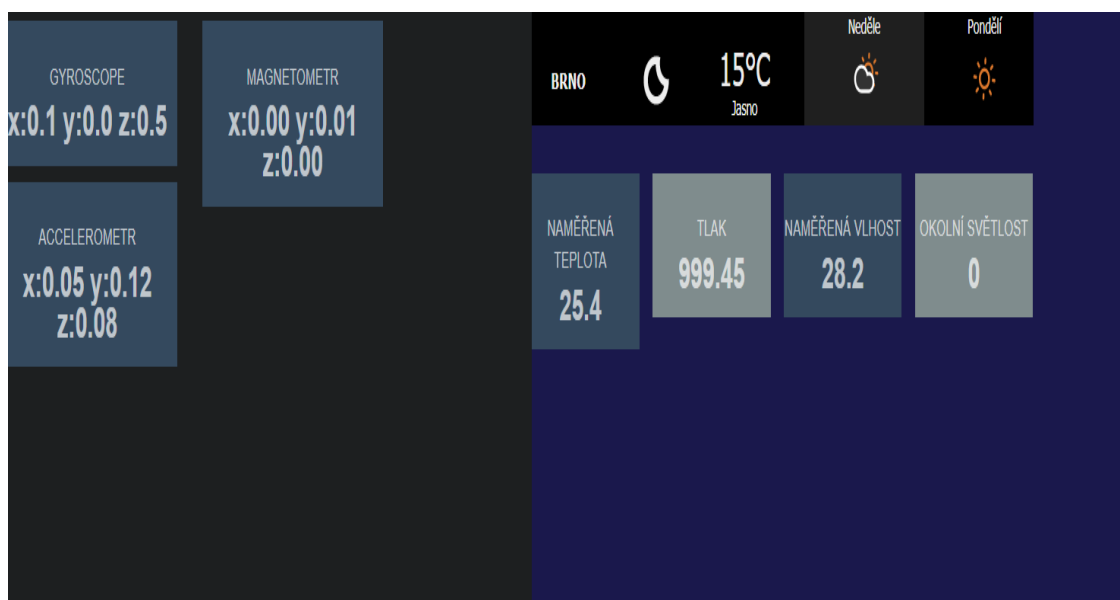


```
MariaDB [hexiwear]> desc service
-> ;
```

Field	Type	Null	Key	Default	Extra
service_id	int(11)	NO	PRI	NULL	
name	varchar(128)	NO		NULL	
value_offset	int(11)	NO		NULL	
description	varchar(256)	NO		NULL	

Obrázek 5: Struktura tabulky se servisy

Servis s počasím zobrazuje data ze senzorů v hodinkách a banner s předpovědí. Data jsou načítána z MQTT brokera a jsou tedy periodicky aktualizována. V rámci některých datových položek jako jsou údaje o teplotě a vlhkosti je nutné zapnout mód „sensor tag“, který začne vysílat údaje z většiny senzorů na jednotlivá UUID.



Obrázek 7: Webová aplikace - údaje ze senzoru teploty a akcelerometru...

Další aplikace jsou obdobné ve čtení dat. Data jsou čteny z MQTT jako již bylo zmíněno, ale též z DB, pokud je potřeba data v rámci časové osy. Malou nevýhodou je skutečnost, že například pro čtení srdečního pulzu je nutné zapnout stejnojmennou aplikaci na hodinkách, aby se začali přes BT začala vysílat požadovaná datová položka.

2.5. Nedostatky

V rámci projektu není plně dokončena vizualizace a není zprovozněna demmostrace některých hodnot, ale vše je ukládáno do DB a data jsou k dispozici v DB nebo na MQTT.

3. Závěr

Projekt je demonstrací práce s vestavěnou platformou Hexiwear a mikropočítačem Raspberry Pi. Mikropočítač umožňuje přemostění komunikace z Bluetooth protokolu do MQTT protokolu a tato část je implementována kompletně. Další kompletní částí je archivace údajů v databázi pomocí skriptu mqtt2db.py napsaného v Pythonu, který slouží jako datová základna pro vizualizaci dat. Vizualizace dat není dokončená a lepší variantou by bylo použití nějakého frameworku pro IoT platformy jako home assistant a podobných, který má vizualizaci již velmi dobře zpracovanou.

4. Literatura

1. Projekt SEN 2018/19 – senzor pro měření hladiny vody, Pavel Koupý

2. <https://github.com/MikroElektronika/HEXIWEAR/blob/master/documentation/HEXIWEAR%20Bluetooth%20Specifications.pdf> – Specifikace komunikačního protokolu
3. <https://git.kernel.org/pub/scm/bluetooth/bluez.git/tree/doc>