



Nízkoúrovňové programování

Práce s periferiemi II

Petr Krajča



Katedra informatiky
Univerzita Palackého v Olomouci



- I²C – Inter-Integrated Circuit (I-squared-C)
- jednoduchá sériová sběrnice
- jeden vodič hodinový signál (SCL), druhý datový kanál (SDL)
- jedno zařízení je řídicí (master) – generuje hodinový signál, řídí komunikaci (polling)
- až 128 řízených zařízení (slave)
- je definován základní komunikační protokol pro všechna zařízení
 - START bit – zahájení komunikace
 - 7 bitů – adresa (slave) zařízení
 - 1 bit – režim read/write
 - ACK bit – potvrzení přenosu dat
 - přenášená data (může se opakovat): 8 bitů + ACK (od slave zařízení)
 - STOP bit – uvolnění sběrnice
- viz <https://www.root.cz/clanky/komunikace-po-seriove-sbornici-isup2supc/>



- registry, které umožňují práci (ovládání zařízení)
- tyto registry mohou být přístupné
 - pomocí speciálních instrukcí CPU (např. `in`, `out`)
 - namapovány do paměti
 - protokolem dané sběrnice
- tři základní typy registrů
 - řídicí – zápisem do těchto registrů měníme chování zařízení nebo zadáváme povely (např. reset zařízení, zahájení čtení/zápisu)
 - stavové – slouží k zjištění stavu zařízení (např. je-li možné číst/zapisovat data)
 - datové – slouží k přenosu dat
- registry jsou určeny číslem a mohou nabývat hodnot typicky o velikostech od 8b do 64b
- konkrétní čísla registrů, jejich význam a význam hodnot v nich uložených popisuje *datasheet* zařízení

(a) inicializace zařízení

- 1 softwarový reset zařízení
- 2 nastavení režimu práce se zařízením
- 3 přečtení vlastností zařízení

(b) čtení dat

- 1 nastavení řídicího registru s informací, že budou čtena data
 - 2 testuje se stavový registr, zda jsou data dostupná
 - 3 je přečten datový registr
- kroky 2 a 3 se mohou opakovat, pokud čteme větší množství dat

(c) zápis dat

- 1 nastavení řídicího registru s informací, že budou zapisována data
 - 2 testuje se stavový registr, zda je možné zapisovat
 - 3 data jsou zapsána do datového registru
- kroky 2 a 3 se mohou opakovat, pokud zapisujeme větší množství dat

(d) ukončení práce se zařízením



- Sense HAT má hned několik obvodů schopných měřit teplotu
- nejpresnější je HTS221 (slouží i k měření vlhkosti); zkruseno vyzařovanou teplotou RasPI
- datasheet: <https://www.st.com/resource/en/datasheet/hts221.pdf> (popis registrů od strany 20)

Inicializace zařízení

- 1 přečtením registru WHO_AM_I ověříme, že komunikujeme se správným zařízením (měl by obsahovat hodnotu 0xbc)
- 2 vypneme zařízení (do CTRL_REG1 nastavíme 0x00)
- 3 zapneme zařízení a nastavíme režim práce (do CTRL_REG1 nastavíme pro 0x84 pro jednorázové měření a konzistentní přenos vícebytových dat)



Měření teploty

- 1 vyvoláme požadavek na jednorázové měření teploty (do CTRL_REG2 zapíšeme 0x01)
- 2 testujeme zda STATUS_REG obsahuje hodnotu 0x03, tj. že jsou naměřeny hodnoty teploty a vlhkosti
- 3 přečteme hodnoty vlhkosti a teploty z registrů HUMIDITY_OUT_L+ HUMIDITY_OUT_H a TEMP_OUT_L+TEMP_OUT_H
 - vždy dvě 8bitové hodnoty tvořící jednu 16bitovou hodnotu
 - nejdříve čteme méně významnou část (dolních 8 bitů), protože přečtením horních 8 bitů změníme hodnotu stavového registru
 - hodnota ještě neodpovídá naměřené teplotě!



Konverze hodnot (kalibrace)

- měřidlo by mělo být z výroby kalibrováno
- pro každé měřidlo jsou individuálně nastaveny kalibrační koeficienty
- jsou dány dvě hodnoty $T0_OUT$ a $T1_OUT$, pro které jsou známy teploty ve stupních celsia $T0_DEGC$ a $T1_DEGC$
- skutečnou teplotu získáme lineární interpolací, viz str. 27
- přesněji řečeno: $T0_DEGC$ a $T1_DEGC$ obsahuje skutečné teploty vynásobené osmi, proto je potřeba výsledek ještě vydělit 8
- hodnoty $T0_OUT$, $T0_DEGC$, ... získáme přečtením odpovídajících registrů po inicializaci zařízení



- standardně práci se zařízením by měly řešit ovladače v jádře
- možné implementovat i v uživatelském prostoru
- viz <https://docs.kernel.org/i2c/dev-interface.html>
- je potřeba
 - balíček `libi2c-dev`
 - a linkovat s přepínačem `-li2c`
- otevřeme soubor `/dev/i2c-XY`
- pomocí `ioctl` nastavíme adresu zařízení
- čteme: `i2c_smbus_read_byte_data(fd, WHO_AM_I)`
- zapisujeme: `i2c_smbus_write_byte_data(fd, CTRL_REG1, 0x84)`



- obvod LSM9DS1
- datasheet <https://www.st.com/resource/en/datasheet/lsm9ds1.pdf>
- princip práce podobný jako v případě měření teploty
- komplexnější nastavení režimů práce
- měřené hodnoty a jednotky
 - akcelerometr: zrychlení (g)
 - gyroskop: uhlová rychlost (dps, degree per second)
 - magnetometr: magnetická indukce (gauss)
- vyžadují kontinuální čtení
- šum + chybí kalibrace (+ odchylky měnící se mírně s teplotou a časem)
- nutné řešit ve vlastní režii



- doplňte do vzorového kódu přepočítání relativní vlhkosti
- doplňte do nějaké aplikace gyroskop, příp. akcelerometr nebo magnetometr (při zkoušení se chovejte k počítači ohleduplně)