



FMS OEM CHIP V3+

Obsah:

ZÁKLADNÍ POPIS	2
INTEGRACE FMS CHIPU DO SYSTÉMU	3
SPI INTERFACE	5
PŘÍKAZ SETTINGS	8
PŘÍKAZ READ	8
READ: FIRMWARE	8
READ: CAN BUS DATA	9
READ: TACHOGRAPH DATA	12
READ: SETTINGS	13
PŘÍKAZ SETTINGS U NEJČASTĚJI MONITOROVANÝCH VOZIDEL.	15

Ing. David Španěl

Mgr. Vítězslav Rejda

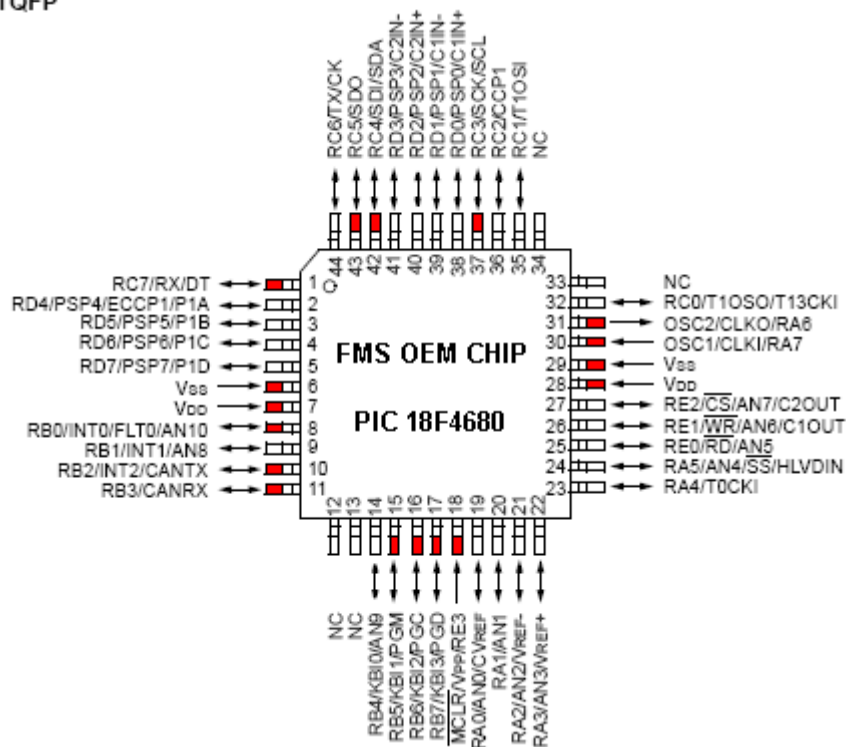
CANLAB s.r.o.

Základní popis

FMS OEM CHIP je předprogramovaný microcontroller PIC 18F4680 v pouzdře 44-pin TQFP, určený k integraci do systému pro sledování vozidel, systémů pro správu vozového parku apod. Chip provádí předzpracování dat z vozidlové sběrnice CAN a případně i z digitálního tachografu DTCO 1381, ke kterému je připojen prostřednictvím jeho info interface. Současná verze firmware podporuje zpracování dat ze sběrnice CAN u vozidel, která využívají protokolu SAE J1939 a dále zpracování dat ze sběrnice CAN u části vozidel koncernu VW (včetně Škoda), Mercedes Sprinter, Nissan atd.. Tato předzpracovaná data poskytuje do nadřazeného systému prostřednictvím sběrnice SPI, kdy pracuje v režimu SPI slave.

Do firmware lze na přání zákazníka zapracovat jiný vůz vybavený sběrnici CAN. Podmínkou je krátké zapůjčení tohoto vozidla pro sérii měření, jejichž úkolem je identifikovat požadovaná data na sběrnici CAN tohoto vozidla.

44-Pin TQFP



Obr. 1: Pouzdro FMS OEM CHIPu

Integrace FMS chipu do systému

Obvod vyžaduje pro svoji činnost napájecí napětí 5V a zdroj hodinového kmitočtu 10MHz. Aby byla možné provádět update firmware obvodu, je doporučeno vybavit jej programovacím konektorem.

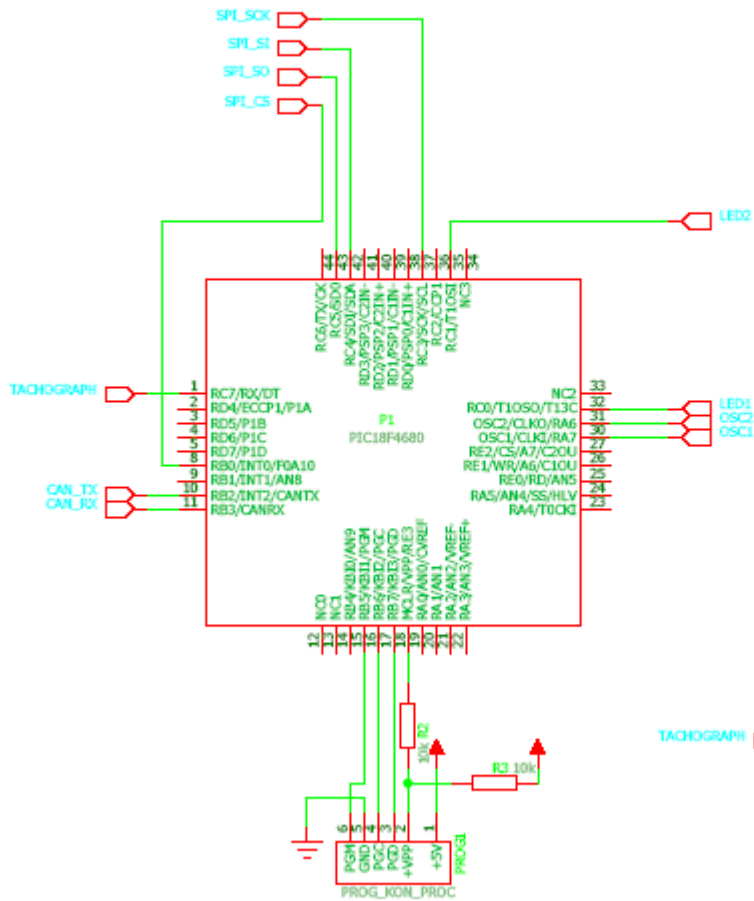
Dále je vhodné při návrhu počítat s volitelným zapnutím/vypnutím zakončovacího odporu CAN sběrnice o velikosti 120 ohmu. Tento zakončovací odpor se využívá při připojení na tzv. FMS bránu u nákladních vozidel. CAN bus je obvykle zakončen na každé straně zakončovacím odporem 120 ohmu (mezi CAN H a CAN L se naměří odpor 60 ohmu).

Signál MCLR je vhodné zapojit jako externí reset obvodu z nadřazeného výstupu. Signál SDO je v době, kdy je CS neaktivní přepnut jako vstup.

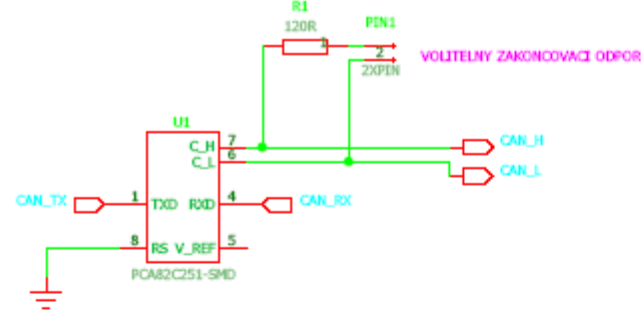
K chipu je možné připojit 2 indikační LED. LED připojená k pinu RC0 indikuje činnost CANu, při příjmu zprávy, která nese data o rychlosti vozidla, výstup změní stav. LED připojená k pinu RC1 pak indikuje činnost tachografu. Při příjmu celé validní zprávy z tachografu výstup změní stav.

Rozhraní CAN je po zapnutí neaktivní dokud chip neobdrží nastavení příkazem SETTINGS. Tachografový vstup je aktivní a zpracovává data ihned po připojení napájecího napětí a vnitřní inicializaci chipu (do 100ms dle varianty firmware).

SPI INTERFACE

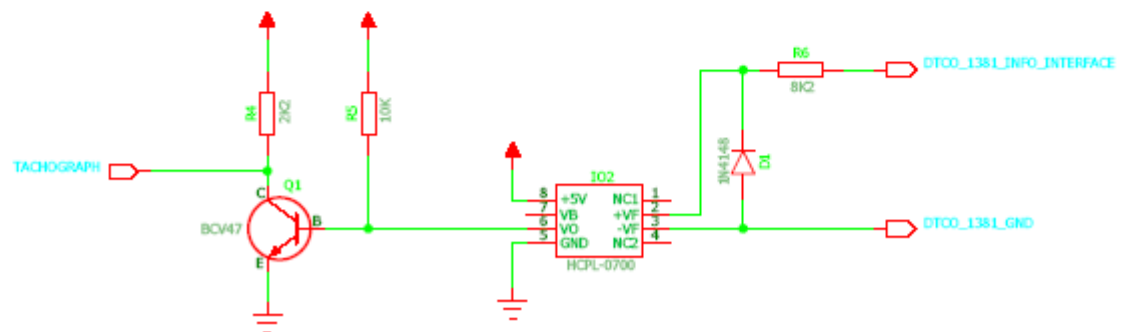
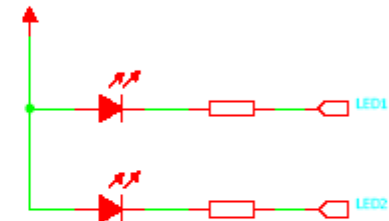
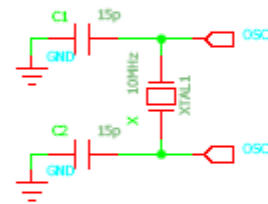


PROGRAMOVACI KONEKTOR



PRIKLAD ROZHRANI CAN

INDIKACNI LED



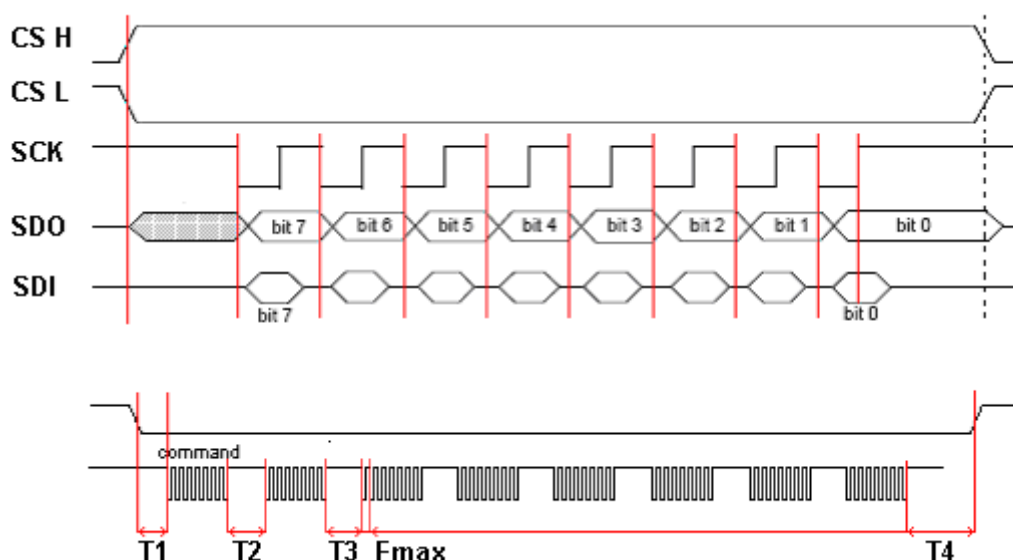
PRIKLAD PRIPOJENI K DIGITALNIMU TACHOGRAFU

Obr. 2: Typické zapojení obvodu.

SPI interface

FMS OEM CHIP poskytuje data prostřednictvím rozhraní SPI. Rozhraní pracuje v SPI módu 01. Obvod lze dodat i ve variantě s aktivní úrovní signálu CS ve stavu H. Maximální rychlost SPI (F_{SCKMAX}) je 2 MHz. Při komunikaci po SPI je nutno dodržet tato pravidla:

- $F_{SCK} \leq 2.25\text{MHz}$
- čas mezi přechodem signálu CS do aktivní úrovně (logická 0) a odesláním-čtením prvního bajtu ($T1$) musí být $\geq 12\mu\text{s}$.
- čas mezi ukončením přenosu bajtu command a počátkem čtení prvního datového bajtu přenosu ($T2$) $\geq 12\mu\text{s}$.
- čas mezi ukončením přenosu datového bajtu a čtením dalšího datového bajtu ($T3$) $\geq 1\mu\text{s}$.
- čas po přenosu posledního datového bajtu a deaktivaci CS ($T4$) $\geq 10\mu\text{s}$.
- aby mohla být zaručena rychlost zápisu dat do interního registru SPI v režimu SPI slave, je v okamžiku aktivního signálu CS je pozastaveno zpracování dat z CAN sběrnice. Příjem dat do HW registrů integrovaného CAN řadiče je i nadále aktivní, tudíž v praxi nedochází k významné ztrátě dat. Důležitá data na CAN sběrnici se periodicky opakují a tedy maximálně dojde k jejich pozdější aktualizaci. Jeho délka je závislá na délce aktivace signálu CS.



CS H – Varianta firmware s aktivním Signálem CHIP SELECT v logické 1.
 CS L – Varianta firmware s aktivním Signálem CHIP SELECT v logické 0.
 Variantu signálu CS si volí zákazník dle jeho potřeby.

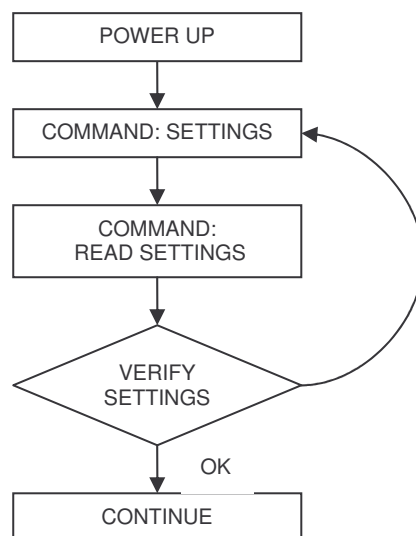
Obr. 3: Rozhraní SPI

Přenos na SPI zahajuje master zasláním bajtu „command“. Tento bajt specifikuje další činnost. Za tímto bajtem může následovat pole dat čtených z FMS OEM CHIPu. Jednotlivé příkazy jsou specifikovány v následující tabulce:

Název příkazu	Hodnota příkazového bajtu (binárně)	Popis
READ	0000 0XXX	XXX: 000 – Verze firmware 001 – CAN bus data 010 – Tachograph data 111 – Settings
SETTINGS	01ZY XXXX	Z: 0 – Normální mód CANu 1 – Listen only mód CANu Y: 0 – Standardní 11 bitový CAN ID 1 – Rozšířený 29 bitový CAN ID XXXX : Rychlost CAN sběrnice 0001 – 20k 0010 – 33.3k 0011 – 50k 0100 – 62.5k 0101 – 83.3k 0110 – 100k 0111 – 125k 1000 – 250k 1001 – 500k 1010 – 1M 1111 – autodetekce *
SETTINGS2	1XXX XXXX	Přídavná informace pro výběr typu vozidla. Tato informace je vyžadována v případech, kdy u 2 různých typů vozidel nese CAN zpráva se stejným identifikátorem stejnou informaci. 000 0000 -SAE_1939 001 0000 -BasicVW 011 0000 -Basic Mercedes 100 0000 -Basic Nisan 101 0000 -Basic Opel 110 0000 -Basic Toyota

* Funkce zatím není implementována. Její funkce vyžaduje aby kromě FMS OEM CHIPu byly na sběrnici CAN alespoň 2 další CANovská zařízení, která spolu komunikují.

Příkaz SETTINGS



Příkaz READ

Příkaz READ je určen k blokovému čtení dat z FMS OEM CHIPu. Master odešle první datový bajt s příkazem READ, a přijme hodnotu 0. Dalším čtením z SPI přijímá data. Čtení je ukončeno po přečtení celé velikosti struktury dat, může však být ukončeno i dříve vrácením signálu CS do neaktivní úrovně (logická 1).

READ: FIRMWARE

Tento příkaz je určen ke čtení verze firmware FMS OEM CHIPu. Délka čtených dat je 13 bajtů. Celý přenos je tedy dlouhý 14 bajtů, 1 bajt pro příkaz READ FIRMWARE a 13 datových bajtů. Vrácená hodnota je textový řetězec s označením verze firmware, například „FMSOEM003CT__“. Řetězec není ukončen žádným zakončovacím znakem.

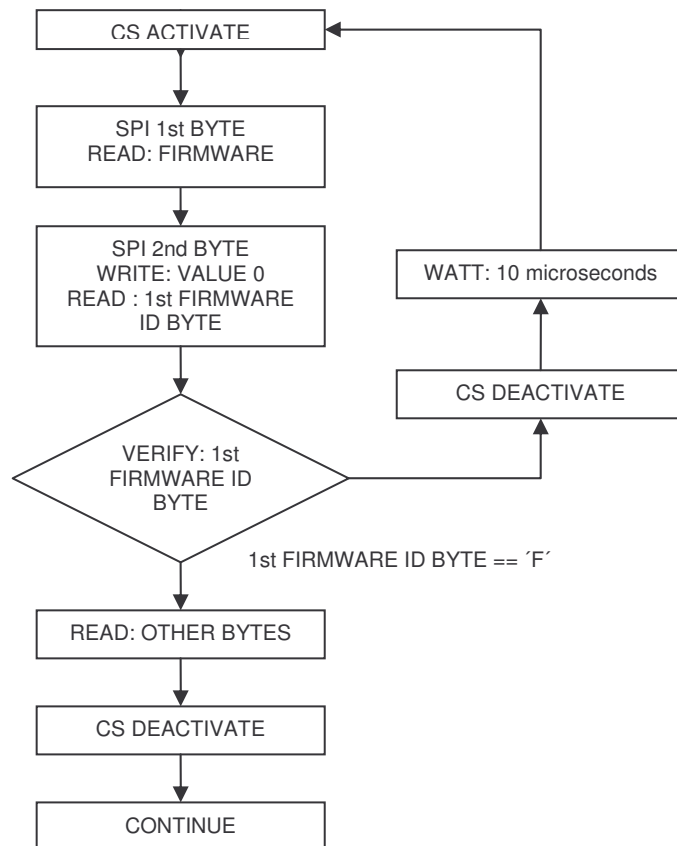
FMSOEM003CT__ (xxxxxyyyab__)

yyy - verze firmware

a - C-podporuje CAN, X-nepodporuje CAN

b - T-podporuje digitální tachograf DTCO 1381, X-nepodporuje tachograf

__ - rezerva



READ: CAN BUS DATA

Tento příkaz je určen ke čtení dat z rozhraní CAN. Jsou čtena data, která tvoří následující strukturu:

```

#define BYTE      unsigned char      // 8 bitový datový typ bez znaménka
#define UINT      unsigned __int16   // 16 bitový datový typ bez znaménka
#define SINT      signed   __int16   // 16 bitový datový typ se znaménkem
#define ULONG     unsigned __int32   // 32 bitový datový typ bez znaménka
  
```

```

typedef struct _FMS {
    BYTE  begin_check;           // sizeof(FMS) = 0x52
    UINT  rpm;
    UINT  speed;
    BYTE  acc_pedal;
    BYTE  brake_pedal;
    ULONG total_fuel_used;
    ULONG total_engine_hours;
    BYTE  fuel_level;
    UINT  fuel_consumption;
    BYTE  axle_weight_captured[12];
    BYTE  axle_weight_location[12];
    UINT  axle_weight[12];
    ULONG total_vehicle_distance;
    UINT  daily_vehicle_distance;
    SINT  service_distance;
    BYTE  engine_coolant_temperature;
  
```

```

        UINT tachograph_speed;
        BYTE tachograph[4];
        BYTE end_check; // 0xAA
    } FMS;

```

Platná hodnota položek `begin_check` a `end_check` označuje platná data. Pokud tyto 2 položky nemají správné hodnoty, je třeba data z vozidla ignorovat. Jsou-li všechny byty položek nastaveny na `0xFF`, je indikováno nepřijetí těchto dat z CANu.

Data ze struktury je možné převést na skutečné hodnoty pomocí této tabulky:

Data	Počet bitů	Váha 1 bitu	Offset
Rychlost	16	1/256 km/h	0
Poloha pedálu akcelerace	8	0.4 %	0
Poloha brzdového pedálu	8	0.4 %	0
Celkově spotřebované palivo	32	0.5 litru	0
Stav palivové nádrže	8	0.4 %	0
Otáčky motoru	16	0.125 otáčky	0
Zatížení nápravy	16	0,5 kg	0
Celkový počet motohodin	32	0,05 hod.	0
Celkově najeté kilometry	32	5 m	0
Vzdálenost v kilometrech do servisní prohlídky	16	5 km	-160 635
Teplota chladící kapaliny.	8	1 °C	-40
Průměrná spotřeba.	16	1/512 km/L	0

Axle weight

Položka `axle_weight_location[x]` udává lokaci hodnoty zatížení nápravy v položce `axle_weight[x]`. Hodnota `axle_weight_captured[x] = 0` udává, že položka neobsahuje žádnou (platnou) hodnotu, `axle_weight_captured[x] = 1` udává, že položka obsahuje platnou hodnotu.

V položce `axle_weight_location[x]` je zakódována informace o čísle měřené nápravy a kole této nápravy. Dolní 4 bity udávají index kola, horní 4 bity udávají index nápravy. Jsou-li všechny 4 bity nastaveny na 1, lokace není známa.

Položka tachograph[4]

Tato položka obsahuje informace, které je možno dekodovat dle následujícího popisu:

tachograph[0]

Bit 2..0 :Driver 1 working state

- 000 = Rest
- 001 = Driver available
- 010 = Work
- 011 = Drive
- 110 = Error
- 111 =

Bit 5..3 :Driver 2 working state

- 000 = Rest
- 001 = Driver available
- 010 = Work

011 = Drive
110 = Error
111 = not available

Bit 7..6 :Drive recognize

00 = Vehicle motion not detected
01 = vehicle motion

tachograph[1]

Bit 3..0 : Driver 1 time rel states

0000 = normal
0001 = 15 min bef. 4 ½ h
0010 = 4 ½ h reached
0011 = 15 min bef. 9 h
0100 = 9 h reached
0101 = 15 min bef. 16 h
0110 = 16h reached
1110 = Error
1111 = not available

Bit 5..4 :Driver 1 card

00 = Card not present
01= Card present

Bit 7..6 :Overspeed

00 = No overspeed
01 = Overspeed

tachograph[2]

Bit 3..0 : Driver 2 time rel states

0000 = normal
0001 = 15 min bef. 4 ½ h
0010 = 4 ½ h reached
0011 = 15 min bef. 9 h
0100 = 9 h reached
0101 = 15 min bef. 16 h
0110 = 16h reached
1110 = Error
1111 = not available

Bit 5..4 :Driver 2 card

00 = Card not present
01= Card present

Bit 7..6 :Not used

tachograph[3]

Bit 0..1 :System event

00 = no tachogr. Event
01 = tachogr. Event

Bit 2..3 :Handling information

00 = no handling information
01 = handling information

Bit 5..4 :Tachgraph performance

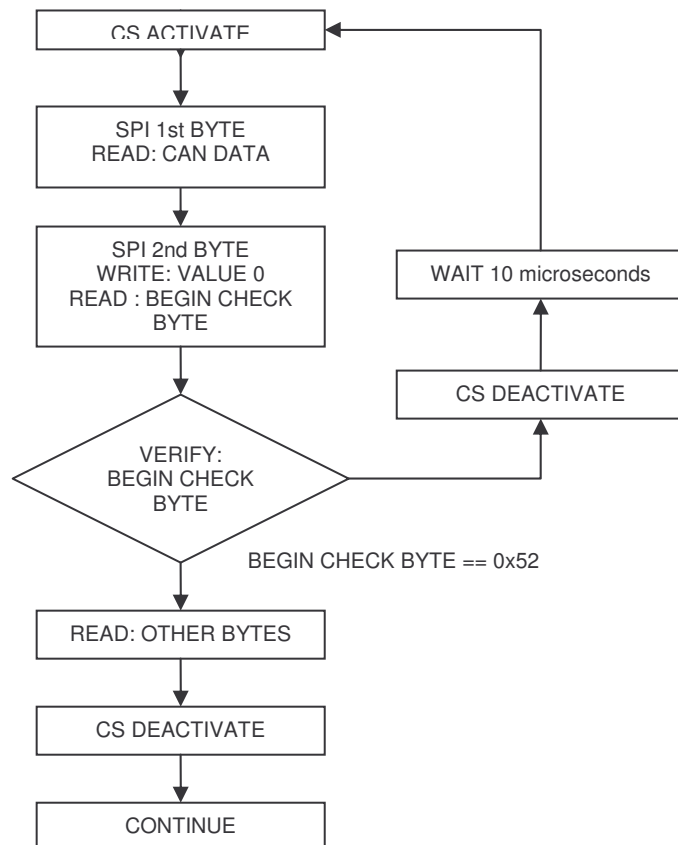
00 = Normal performance

01 = Performance

Bit 7..6 :Direction indicator

00 = Forward

01 = Reverse

**READ: TACHOGRAPH DATA**

```

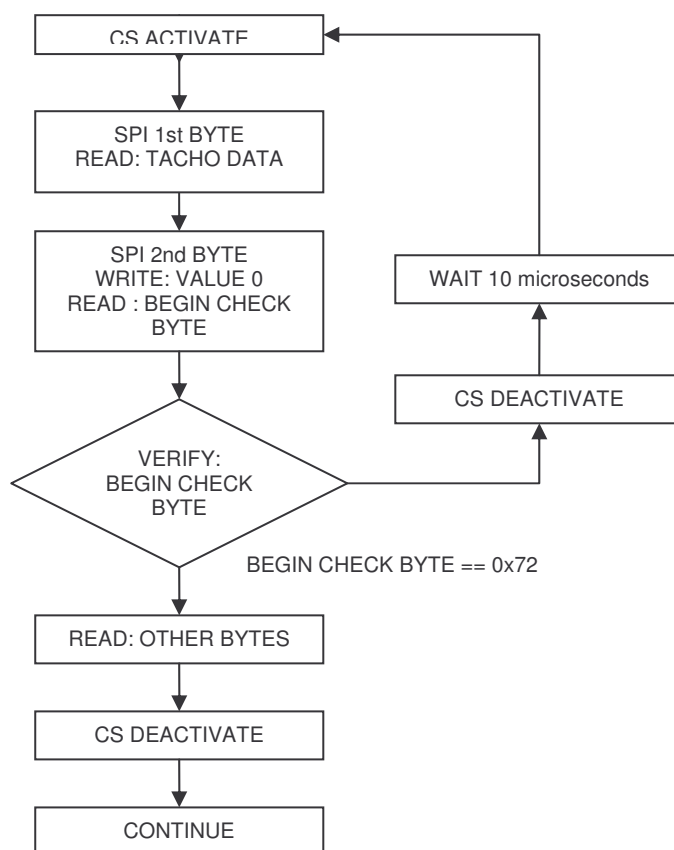
typedef struct _DTI{
    BYTE    begin_check;           // sizeof(DTI) = 0x72
    BYTE    seconds;
    BYTE    minutes;
    BYTE    hours;
    BYTE    month;
    BYTE    day;
    BYTE    year;
    BYTE    local_minute_offset;
    BYTE    local_hour_offset;
    BYTE    work_states;
    BYTE    driver_1_states;
    BYTE    driver_2_states;
    BYTE    tachograph_status;
    UINT    tachograph_vehicle_speed;
    LONG    total_vehicle_distance;
    LONG    trip_distance;
    INT     k_factor;
}
  
```

```

INT      engine_speed;
INT      additional_information;
BYTE     vehicle_id_len;
BYTE     vehicle_id[20];
BYTE     vehicle_reg_len;
BYTE     vehicle_reg[20];
BYTE     driver_1_len;
BYTE     driver_1[20];
BYTE     driver_2_len;
BYTE     driver_2[20];
BYTE     end_check;           // 0xAA

} DTI;

```

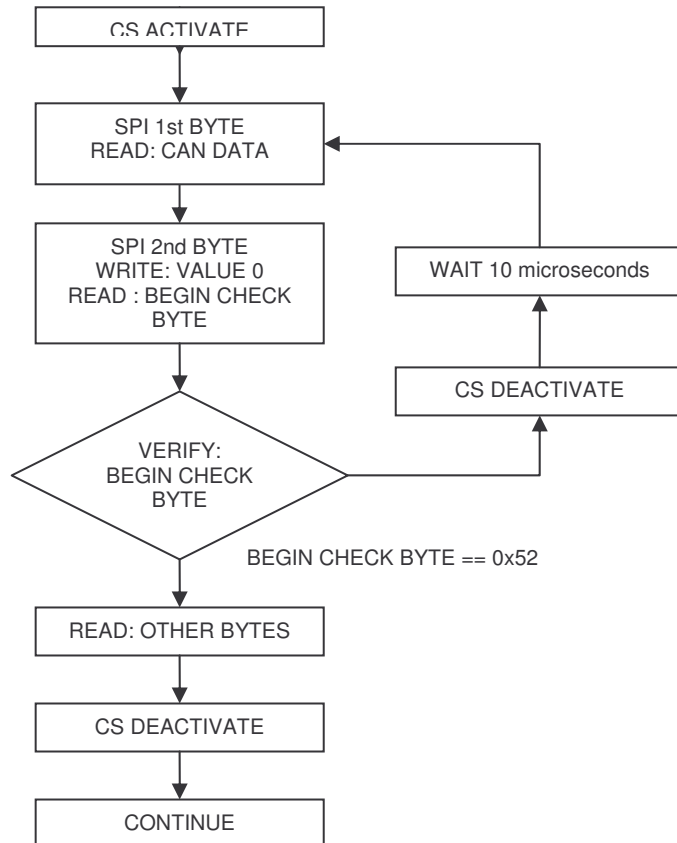


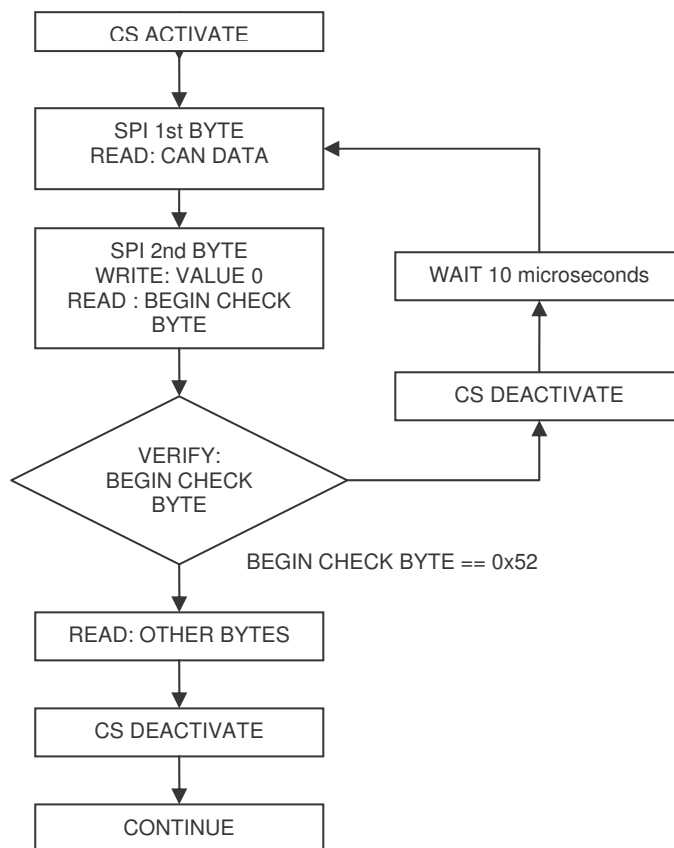
READ: SETTINGS

Tento příkaz je určen ke zpětnému čtení aktuálního nastavení. Odpovědí je jeden datový bajt, jehož bity mají následující význam:

AAZY XXXX
AA: Určeno pro testy firmware.
Z: 0 – Normální mód CANu
1 – Listen only mód CANu
Y: 0 – Standardní 11 bitový CAN ID
1 – Rozšířený 29 bitový CAN ID
XXXX : Rychlost CAN sběrnice
0001 – 20k
0010 – 33.3k

- 0011 – 50k
- 0100 – 62.5k
- 0101 – 83.3k
- 0110 – 100k
- 0111 – 125k
- 1000 – 250k
- 1001 – 500k
- 1010 – 1M
- 1111 – autodetekce * (zatím nepodporováno)





Příkaz SETTINGS u nejčastěji monitorovaných vozidel.

Nákladní vozidla – páteřní CAN bus

- listen only
- rozšířený CAN ID
- rychlost 250k

Nákladní vozidla – FMS gateway (FMS brána)

- normální mód
- rozšířený CAN ID
- rychlost 250k

Vozidla Škoda/VW, motorový CAN bus

- listen only
- standardní CAN ID
- rychlost 500k