

Beschreibung
Description

MULTIPLEX SENSOR BUS



Version 2

1. Index

| | |
|--|-----------|
| 1. INDEX | 2 |
| 2. HISTORY | 3 |
| 3. MULTIPLEX SENSOR BUS [DEUTSCH] | 4 |
| 3.1. Vorwort | 4 |
| 3.2. Schnittstelle | 4 |
| 3.3. Kenndaten | 4 |
| 3.4. Idle Line Multiprozessor Protokoll | 5 |
| 3.5. Adressverwaltung | 5 |
| 3.6. Datenanforderung | 6 |
| 3.7. Werteklassen und -bereiche | 7 |
| 3.8. Adressen / Befehle | 7 |
| 3.9. Sonderdaten Werteklasse 0 | 8 |
| 1.1.1. ECU Status und Fehlermeldungen | 8 |
| 3.10. Beispiel | 9 |
| 3.11. MSB-Logo | 10 |
| 4. MULTIPLEX SENSOR BUS [ENGLISH] | 11 |
| 4.1. Preface | 11 |
| 4.2. Interface | 11 |
| 4.3. Characteristics | 11 |
| 4.4. Idle line multiprocessor protocol | 12 |
| 4.5. Address management | 12 |
| 4.6. Data request | 13 |
| 4.7. Value classes and ranges | 14 |
| 4.8. Addresses / commands | 14 |
| 4.9. Special data value class 0 | 15 |
| 1.1.2. ECU status- und error messages | 15 |
| 4.10. Example | 16 |
| 4.11. MSB logo | 17 |

2. History

| | |
|-----------|---|
| Version 2 | <ul style="list-style-type: none">- Neues Kapitel 3.3 Kenndaten- Kapitel 3.5: Adressen müssen einstellbar / deaktivierbar sein- Kapitel 3.7: undefinierte Klassen dürfen nicht benutzt werden- Kapitel 3.7: Erweiterung der Klasse 5- Neues Kapitel 3.9 Sonderdaten Werteklasse 0- <i>New chapter 4.3 Characteristics</i>- <i>Chapter 4.5: Address have to be adjustable / deactivatable</i>- <i>Chapter 4.7: It is not allowed to use undefined value classes</i>- <i>Chapter 4.7: Add on to value class 5</i>- <i>New chapter 4.9 Special data value class 0</i> |
| Version 1 | <ul style="list-style-type: none">- Erste zweisprachige Version mit MSB-Logo.- <i>First bilingual version with MSB-logo.</i> |

3. MULTIPLEX Sensor Bus [Deutsch]

3.1. Vorwort

Dieses Dokument beschreibt den Multiplex Sensor Bus (MSB) in Bezug auf den Bus-Betrieb (Messbetrieb). Die enthaltenen Informationen ermöglichen es, eine beliebige Sensorhardware an den Bus anzukoppeln, ohne dass Konflikte mit anderen Sensoren oder dem Abfragegerät (Master) entstehen.

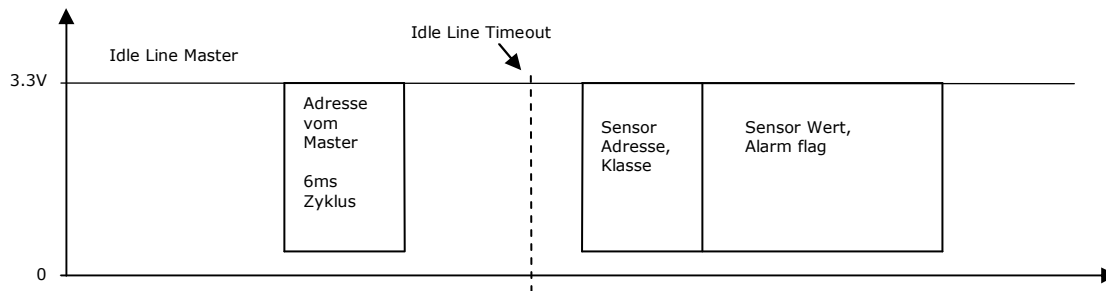
3.2. Schnittstelle

| | |
|--|---|
| Schnittstelle Hardware | Die Sensoren werden über Uni-Steckverbinder an den Bus angeschlossen. Die Belegung ist identisch mit einem Servostecker: GND – UBat – Daten |
| Schnittstellenparameter | 38400, n, 8, 1 Eindrahtschnittstelle halbduplex |
| Protokoll: Idle Line Multiprozessor Protokoll | Der Sensor ist Slave. Er empfängt solange, bis ein Idle Line Timeout auftritt. Anschließend werden die Daten ausgewertet und entsprechend reagiert. Siehe Grafik 3.4 Idle Line Multiprozessor Protokoll |
| Datenabfrage | Der Master schickt ca. alle 6ms eine Adresse auf den Bus (0x00 bis 0x0f rotierend). Erkennt der Sensor die Adresse, antwortet dieser. |
| Antwort vom Sensor | Die Antwort besteht aus 3 Bytes. 1. Byte: Adresse (Echo) + Werteklasse (= Einheit) 2.+3. Byte: Sensorwert und Alarmflag (siehe Kapitel 3.6 Datenanforderung) |

3.3. Kenndaten

| Parameter | Min | Typ | Max | Einheit |
|-----------------------------|-----|-----|-----|---------|
| Batteriespannung | 3,5 | | 9 | V |
| Idle Line Timeout | 256 | | 560 | µs |
| Low level Eingangsspannung | | | 0,8 | V |
| Low level Ausgangsspannung | | | 0,4 | V |
| High level Eingangsspannung | 2,0 | | | V |
| High level Ausgangsspannung | 2,4 | | | V |

3.4. Idle Line Multiprozessor Protokoll



Der Sensor ist Slave. Er empfängt so lange Bytes, bis ein Idle Line Timeout auftritt. Nach dem Timeout interpretiert der Sensor die Anzahl und Inhalt der empfangenen Bytes. Hat er genau ein Byte empfangen und ist dies eine im Sensor eingetragene Adresse oder ein Broadcast-Befehl, reagiert der Sensor und sendet seine Antwort, bzw. reagiert auf den Befehl (siehe Kapitel 3.8 Adressen / Befehle).

Ist der Sensor nicht an die Sensorkette angeschlossen, sind auch andere Befehle, z.B. zur Parametrierung, möglich. Diese dürfen sich allerdings nicht mit den Befehlen des MSB überschneiden.

3.5. Adressverwaltung

Die Adressen der Sensoren müssen einstellbar / deaktivierbar sein, da über die Adressen die Darstellung der Sensorwerte auf dem Senderdisplay gesteuert wird. An einem Sensorbus darf jede Adresse nur einmal vergeben sein, sonst entstehen Kollisionen.

Ein Sensor kann mehrere Adressen enthalten. Diese können innerhalb des Sensors z.B. auch für berechnete Werte vergeben werden (Mittelwert / Maximalwert / Minimalwert usw.)

Im Auslieferungszustand hat jeder Sensor bestimmte Adressen voreingestellt (Default / Werkseinstellungen). Diese Voreinstellung sollte vom Sensorhersteller mit der Multiplex Modellsport GmbH & Co. KG abgesprochen werden, um Konflikte zwischen Sensoren unterschiedlicher Hersteller zu vermeiden.

3.6. Datenanforderung

Um einen Wert in der Sensorkette abzufragen, sendet der Master eine Adresse. Die Adresse wird von allen Sensoren empfangen. Nur der eine Sensor, der diese Adresse hinterlegt hat, antwortet. Die Antwort enthält im ersten Byte im Highnibble eben diese Adresse und im Low-Nibble die Einheit (Werteklasse). Die beiden weiteren Bytes enthalten den Wert als signed 15 Bit in Zweierkomplementdarstellung, sowie ein Alarmflag. Der Wert ist um ein Bit in Richtung MSB verschoben. Das LSB wird für das Alarmflag verwendet. Übertragungsreihenfolge ist Low byte / High byte

Sensorantwort auf eine Werteanforderung:

Byte 1

| | |
|-------------------|-----------------------|
| Adresse (0 .. 15) | Werteklasse (1 .. 15) |
|-------------------|-----------------------|

Byte 2 (Lowbyte)

| | | | | | | | |
|--|--|--|--|--|-----|----------|-----------|
| | | | | | ... | Wert LSB | Alarmflag |
|--|--|--|--|--|-----|----------|-----------|

Byte 3 (Highbyte)

| | | | | | | | |
|------------|----------|-----|--|--|--|--|--|
| Vorzeichen | Wert MSB | ... | | | | | |
|------------|----------|-----|--|--|--|--|--|

Wertebereiche:

Der übertragbare Wertebereich ist durch die 15 Bit signed und das Alarmflag begrenzt auf -16384 bis +16383 (0xC000 bis 0x3FFF). Die Nutzung dieses Wertebereichs ist abhängig von der Sensorklasse. Z.B. Sensorklasse Spannung: -600 bis +600 entspricht -60,0 bis +60,0 Volt. Siehe Kapitel 3.7 Werteklassen und -bereiche.

Alarmflag:

Das Alarmflag kann vom Sensor gesetzt werden, wenn der Wert eine voreingestellte Schwelle unter- oder überschritten hat. Im Senderdisplay wird dieser Wert dann schwarz hinterlegt und der Benutzer durch einen Alarmton aufmerksam gemacht.

Ungültige Meßwerte:

Wenn der Sensor die Adresse erkennt, aber keinen gültigen Meßwert liefern kann (z.B. Startup-Zeit noch nicht beendet), sendet er als Wert 0x8000 zurück. Auf dem Senderdisplay wird dann "-.- Einheit" angezeigt.

3.7. Werteklassen und -bereiche

Eine Werteklasse repräsentiert die Einheit des Wertes. Der Bereich für Werteklassen ist auf 4 Bit beschränkt, sodass Adresse und Klasse eines Werts in ein Byte zusammengepackt werden können. So können 15 verschiedene Maßeinheiten verwendet werden. Nicht definierte Werteklassen dürfen nicht benutzt werden.

| Klasse | Beschreibung | Auflösung | Min Wert | Max Wert |
|--------|-----------------------------|--------------|----------|----------|
| 0 | Sonderdaten siehe Kapitel 0 | | | |
| 1 | Spannung | 0,1V | -600 | +600 |
| 2 | Strom | 0,1A | -1000 | +1000 |
| 3 | Steigen/Sinken | 0,1m/s | -500 | +500 |
| 4 | Geschwindigkeit | 0,1km/h | 0 | +6000 |
| 5 | Drehzahl | 100 1/min | 0 | +500 |
| | | 10 1/min | 0 | -5000 |
| 6 | Temperatur | 0,1°C | -250 | 7000 |
| 7 | Richtung | 0,1 Grad (°) | 0 | 3600 |
| 8 | Höhe/kurze Distanz | 1m | -500 | 2000 |
| 9 | Füllstand Tank | 1% Tank | 0 | +100 |
| 10 | Link Quality Indicator | 1% LQI | 0 | +100 |
| 11 | Stromverbrauch | 1mAh | -16000 | 16000 |
| 12 | Flüssigkeiten | 1mL | 0 | 16000 |
| 13 | lange Distanz | 0,1km | 0 | 16000 |
| 14 | | | | |
| 15 | | | | |

3.8. Adressen / Befehle

Der Adressraum für die Sensoren beträgt 4 Bit. Damit sind Adressen zwischen 0 und 15 möglich. Zusätzlich sind im Busbetrieb weitere Befehle möglich, die außerhalb des Adressraums liegen. Die Sensoren reagieren individuell darauf. Einige Befehle sind von Multiplex für bestimmte Funktionen vordefiniert. Externe Sensorhersteller dürfen diese Befehle nur für die hinterlegten Funktionen verwenden. Der Bereich 0x80-0x8f ist für Erweiterungen des MSB vorgesehen, es darf demnach nicht auf Anforderungen reagiert werden, welche mit Bytes aus diesem Bereich beginnen.

| Adressen / Befehle Busbetrieb | Anzahl Bytes | Funktion |
|-------------------------------|--------------|---|
| 0x00 - 0x0f | 1 | frei belegbar für Sensorwerte |
| 0x5A | 1 | Rücksetzen Befehl: die Sensoren setzen alle gespeicherten bzw. kumulierten Werte zurück (z.B. Min, Max, Durchschnitt, Ladungszähler etc.) |
| 0x80 - 0x8f | n | reserviert für spätere Erweiterungen des MSB |

Die Adressabfrage und der Rücksetzen-Befehl sind 1-Byte Befehle. Den Befehlen 0x80-0x8f kann eine spezifische Anzahl von Bytes mit weiteren Informationen zu diesem Befehl folgen.

3.9. Sonderdaten Werteklasse 0

Die Werteklasse 0 dient zur Übertragung von Sonderdaten und ist unterteilt in Unterklassen. Die Unterklassen werden durch Byte 3 unterschieden. Nichtdefinierte Unterklassen und Bereiche dürfen nicht benutzt werden.

Aufbau der Sensorantwort bei Klasse 0

Byte 1

| | |
|-------------------|-----------------|
| Adresse (0 .. 15) | Werteklasse = 0 |
|-------------------|-----------------|

Byte 2 (Lowbyte)

| |
|--|
| |
|--|

Byte 3 (Highbyte)

| |
|-------------|
| Unterklasse |
|-------------|

Unterklassen

| Unterklasse (Byte 3) | Funktion |
|----------------------|---------------------------------|
| 0x01 | ECU Status- und Fehlermeldungen |
| | |

1.1.1. ECU Status und Fehlermeldungen

Mit der Unterklasse „ECU Status- und Fehlermeldungen“ können Meldungen auf das Senderdisplay geschrieben werden. Die unterschiedlichen Meldungen sind im Sender gespeichert und werden durch die oberen sieben Bit von Byte 2 Adressiert. Bit 0 dient als Alarmflag und kann z.B. bei Fehlermeldungen gesetzt werden.

| | |
|----|--------------|
| 0 | -OFF- |
| 1 | Stby/START |
| 2 | Ignite... |
| 3 | acceler. |
| 4 | Stabilise |
| 5 | LearnHI |
| 6 | LearnLO |
| 7 | RUN... |
| 8 | SlowDown |
| 9 | Manual |
| 10 | SwitchOff |
| 11 | RUN (reg.) |
| 12 | AccelrDly |
| 13 | SpeedCtrl |
| 14 | Rpm2Ctrl |
| 15 | PreHeat1 |
| 16 | PreHeat2 |
| 17 | MainFStrt |
| 18 | --- |
| 19 | Keros.FullOn |

| | |
|----|-----------|
| 20 | ---- |
| 21 | RC-Off |
| 22 | OverTemp |
| 23 | IgnTimOut |
| 24 | AccTimOut |
| 25 | Acc. Slow |
| 26 | Over-Rpm |
| 27 | Low-Rpm |
| 28 | BattryLow |
| 29 | Auto-Off |
| 30 | LowTemp |
| 31 | HiTempOff |
| 32 | GlowPlug! |
| 33 | WatchDog |
| 34 | FailSafe |
| 35 | Manual |
| 36 | PowerFail |
| 37 | TempFail |
| 38 | FuelFail |
| 39 | Rpm2Fail |

| | |
|----|-----------|
| 40 | 2nd EngF |
| 41 | 2nd Diff |
| 42 | 2nd Comm |
| 43 | No-OIL |
| 44 | OverCurr |
| 45 | No Pump! |
| 46 | WrongPmp |
| 47 | - ON - |
| 48 | Enabled |
| 49 | Disabled |
| 50 | Prop-Fail |
| 51 | Cooling |
| | |
| | |
| | |
| | |
| | |
| | |

3.10. Beispiel

Es sind zwei Multiplex Sensoren an den Bus angeschlossen, ein Spannungssensor mit zwei Kanälen und ein Stromsensor. Die Kanäle auf den Sensoren sind wie folgt parametrier:

Spannungssensor Kanal 1:

Adresse Spannung 1 = 3
 Alarm max = aus (16383 = $0x3FFF = 2^{14}-1$)
 Alarm min = 46 = 4,6V

Spannungssensor 2:

Adresse Spannung 2 = 4
 Alarm max = aus (16383)
 Alarm min = 114 = 11,4 V

Stromsensor:

Adresse Strom = 5
 Alarm max = 20 A
 Alarm min = aus (-16384 = $0xC000$)

Abfrage der Daten:

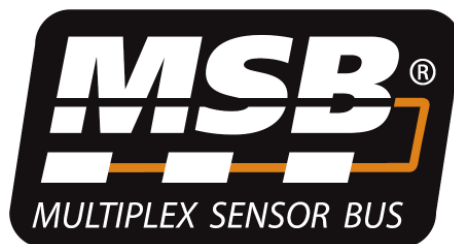
| Master sendet | Sensor: gemessener Wert | Sensor antwortet |
|---------------|-------------------------|--|
| 0 | | - |
| 1 | | - |
| 2 | | - |
| 3 | 4,5 V | 0x31 = Adr. 3 + Werteklasse 1 "Spannung" 0x005B = $45*2+1 = 4,5$ Volt + Alarmflag |
| 4 | 12,5 V | 0x41 = Adr. 4 + Werteklasse 1 "Spannung" 0x00FA = $125*2+0 = 12,5$ Volt, kein Alarmflag |
| 5 | 18,3 A | 0x52 = Adr. 5 + Werteklasse 2 "Strom" 0x016E = $183*2+0$ |
| 6 | | - |
| 7 | | - |
| 8 | | - |
| 9 | | - |
| 9 | | - |
| 10 | | - |
| 11 | | - |
| 12 | | - |
| 13 | | - |
| 14 | | - |
| 15 | | - |

3.11. MSB-Logo

Für Geräte, welche kompatibel zum MULTIPLEX Sensor Bus sind kann nach Rücksprache mit der Firma MULTIPLEX das MSB-Logo benutzt werden. Da das Logo von MULTIPLEX geschützt ist, ist hierfür die schriftliche Genehmigung der Firma MULTIPLEX erforderlich.

Das Logo kann mit der Genehmigung sowohl in langer, als auch in kurzer Form benutzt werden.

Lange Version:



Kurze Version:



4. MULTIPLEX Sensor Bus [English]

4.1. Preface

This document describes the Multiplex Sensor Bus (MSB) interface regarding to bus (measuring) operation. With this document it is possible to connect any sensor hardware to the Multiplex Sensor Bus without causing interference to other bus hardware (sensors, RF-receiver / master)

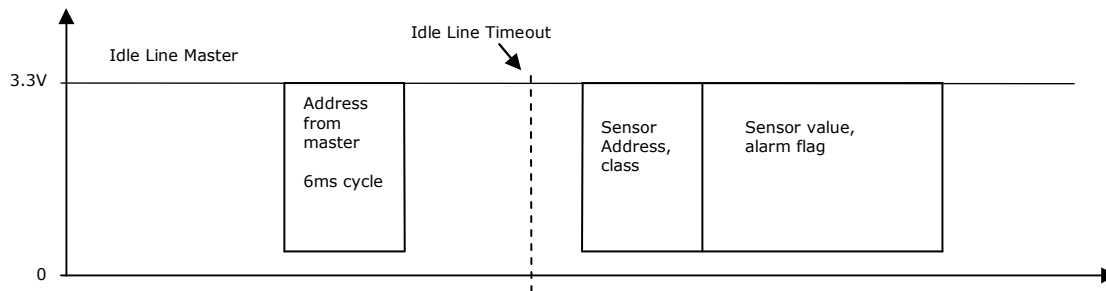
4.2. Interface

| | |
|--|--|
| Interface hardware | Sensors are connected to the master via Uni-connector. Pin assignment ist identical to servo cable: GND – Battery – Data |
| Interface parameter | 38400, n, 8, 1 single wire half duplex |
| Protocol: Idle line multiprocessor protocol | The sensor is slave. It receives data until an idle line timeout occurs. Then the data will be interpreted and the sensor reacts. See diagram 4.4 Idle line multiprocessor protocol |
| Data request | The master sends an address (0x00 to 0x0f) to the bus every 6ms. The sensor responds, if it recognizes this address. |
| Sensor response | The sensor response consists of 3 bytes: 1st Byte: address (echo) + value class (= unit) 2nd+3rd Byte: sensor value + alarm flag (see chapter 4.6 Data request) |

4.3. Characteristics

| Parameter | Min | Typ | Max | Unit |
|---------------------------|-----|-----|-----|------|
| Battery voltage | 3,5 | | 9 | V |
| Idle Line Timeout | 256 | | 560 | µs |
| Low level input voltage | | | 0,8 | V |
| Low level output voltage | | | 0,4 | V |
| High level input voltage | 2,0 | | | V |
| High level output voltage | 2,4 | | | V |

4.4. Idle line multiprocessor protocol



The sensor is slave. It receives bytes continuously until an idle line timeout occurs. After the timeout, the sensor interprets number and content of the received bytes. If the number of bytes is 1 (one) and this is a valid address or a broadcast command, the sensor responds with data or reacts to the command (see chapter 4.8 Addresses / commands).

If the sensor is not connected to the bus, other commands are possible e.g. for changing sensor setup. It must not overlap with MSB commands.

4.5. Address management

The sensor addresses have to be adjustable and deactivatable, because the display presentation of the measured values is controlled by the addresses. Each address may be assigned one time only at the bus. Otherwise data collision will occur.

A sensor can contain multiple addresses. They can be assigned to measured values or to calculated values like minimum / average / maximum or others.

In factory default settings, specific addresses are assigned to the different values, the sensor provides. Sensor developers should arrange these default addresses together with Multiplex Modellsport GmbH & Co.KG. This will avoid conflicts with sensors from different manufacturers, when they are used with factory default settings.

4.6. Data request

To get a value from the Multiplex Sensor Bus, the master sends an address. Any sensor at the bus receives this address but only one sensor will respond with the associated value. The first byte of the answer contains address (high nibble) and unit (value class, low nibble). The next two bytes contain the value as signed 15 Bit in two's complement plus alarm flag. The value is left-shifted one time, LSB is used as alarm flag. Transmit order is low byte / high byte.

Sensor response on a data request:

Byte 1

| | |
|-------------------|-----------------------|
| address (0 .. 15) | value class (1 .. 15) |
|-------------------|-----------------------|

Byte 2 (Lowbyte)

| | | | | | | | |
|--|--|--|--|--|-----|-----------|------------|
| | | | | | ... | value LSB | alarm flag |
|--|--|--|--|--|-----|-----------|------------|

Byte 3 (Highbyte)

| | | | | | | | |
|----------|-----------|-----|--|--|--|--|--|
| sign bit | value MSB | ... | | | | | |
|----------|-----------|-----|--|--|--|--|--|

Value ranges:

The maximum range of signed 15 bit is -16384 to +16383 (0xC000 to 0x3FFF). The usage of this range depends on the value class. E.g. value class voltage: -600 to +600 matches -60,0 to +60,0 Volt. See chapter 4.7 Value classes and ranges.

Alarm flag:

The alarm flag can be set by the sensor, if the value exceeds or runs below a specific threshold. The Multiplex Royal Pro will blacken this value in the display and indicate the alarm by a sound.

Non-valid values:

If the sensor recognizes an address, but is not able to deliver a valid measurement (e.g. startup time is still running, measure error occurred...), it responds with 0x8000. The transmitter will display "-.- unit".

4.7. Value classes and ranges

A value class represents the unit of the measured value. The range is limited to four bits, so that address and class can be packed in one byte. By this means, 15 units can be encoded. It is not allowed to use undefined value classes.

| Class | Description | resolution | Min value | Max value | |
|-------|-------------------------------|----------------|-----------|-----------|--|
| 0 | Special data see chapter 0 | | | | |
| 1 | Voltage | 0,1V | -600 | +600 | |
| 2 | Current | 0,1A | -1000 | +1000 | |
| 3 | Rising / sinking (variometer) | 0,1m/s | -500 | +500 | |
| 4 | Speed | 0,1km/h | 0 | +6000 | |
| 5 | Revolution | 100 1/min | 0 | +500 | |
| | | 10 1/min | 0 | -5000 | |
| 6 | Temperature | 0,1°C | -250 | 7000 | |
| 7 | Direction | 0,1 degree (°) | 0 | 3600 | |
| 8 | Height / short distance | 1m | -500 | 2000 | |
| 9 | Tank level | 1% Tank | 0 | +100 | |
| 10 | Link Quality Indicator | 1% LQI | 0 | +100 | |
| 11 | Charge counter | 1mAh | -16000 | 16000 | |
| 12 | Fluid | 1mL | 0 | 16000 | |
| 13 | Long distance | 0,1km | 0 | 16000 | |
| 14 | | | | | |
| 15 | | | | | |

4.8. Addresses / commands

The address space for sensors is 4 Bit. This means addresses between 0 and 15 are possible. Additional commands are possible, which are out of the address space. The sensors will react uniquely on such special commands. Some commands are already defined by Multiplex sensors in bus operation mode. Other sensor manufacturers may use these commands only for the predefined function. The range 0x80-0x8f is reserved for future extensions of MSB. It must not react on a request, which start with these bytes.

| Addresses / commands | No. of Bytes | Function |
|----------------------|--------------|--|
| 0x00 - 0x0f | 1 | addresses, free in use |
| 0x5A | 1 | "Clear command": charge counter, min, avg, max values will be reset when this command appears on the bus |
| 0x80 - 0x8f | n | reserved for future extensions of MSB |

Address query and clear command are 1-Byte commands. The commands 0x80-0x8f can be followed by a specific number of bytes with additional command information.

4.9. Special data value class 0

Value class 0 is used for special data and is separated in sub classes. The sub class is identified by byte 3. It is not allowed to use undefined sub classes.

Sensor response at class 0

Byte 1

| | |
|-------------------|-----------------------|
| address (0 .. 15) | value class (1 .. 15) |
|-------------------|-----------------------|

Byte 2 (Lowbyte)

| |
|--|
| |
|--|

Byte 3 (Highbyte)

| |
|-----------|
| Sub class |
|-----------|

Sub class

| Sub class (Byte 3) | Function |
|--------------------|--------------------------------|
| 0x01 | ECU status- and error messages |
| | |

1.1.2. ECU status- und error messages

Sub class "ECU status- and error messages" it is used to write messages to telemetry display on radio. The different messages are stored inside the radio and will be addressed by the upper 7 bit of byte 2. Bit 0 is used as alarm flag. It could be set in case of an error message for example.

| | |
|----|--------------|
| 0 | -OFF- |
| 1 | Stby/START |
| 2 | Ignite... |
| 3 | acceler. |
| 4 | Stabilise |
| 5 | LearnHI |
| 6 | LearnLO |
| 7 | RUN... |
| 8 | SlowDown |
| 9 | Manual |
| 10 | SwitchOff |
| 11 | RUN (reg.) |
| 12 | AccelrDly |
| 13 | SpeedCtrl |
| 14 | Rpm2Ctrl |
| 15 | PreHeat1 |
| 16 | PreHeat2 |
| 17 | MainFStrt |
| 18 | --- |
| 19 | Keros.FullOn |

| | |
|----|-----------|
| 20 | ----- |
| 21 | RC-Off |
| 22 | OverTemp |
| 23 | IgnTimOut |
| 24 | AccTimOut |
| 25 | Acc. Slow |
| 26 | Over-Rpm |
| 27 | Low-Rpm |
| 28 | BattryLow |
| 29 | Auto-Off |
| 30 | LowTemp |
| 31 | HiTempOff |
| 32 | GlowPlug! |
| 33 | WatchDog |
| 34 | FailSafe |
| 35 | Manual |
| 36 | PowerFail |
| 37 | TempFail |
| 38 | FuelFail |
| 39 | Rpm2Fail |

| | |
|----|-----------|
| 40 | 2nd EngF |
| 41 | 2nd Diff |
| 42 | 2nd Comm |
| 43 | No-OIL |
| 44 | OverCurr |
| 45 | No Pump! |
| 46 | WrongPmp |
| 47 | - ON - |
| 48 | Enabled |
| 49 | Disabled |
| 50 | Prop-Fail |
| 51 | Cooling |
| | |
| | |
| | |
| | |
| | |
| | |
| | |

4.10. Example

Two Multiplex sensors are connected to the bus, a two-channel voltage sensor and a current sensor. The sensor channels are set as follows:

Voltage sensor channel 1:

Address voltage 1 = 3

Alarm max = off (16383 = $0x3FFF = 2^{14}-1$)

Alarm min = 46 = 4,6V

Voltage sensor channel 2:

Address voltage 2 = 4

Alarm max = off (16383)

Alarm min = 114 = 11,4 V

Current sensor:

Address = 5

Alarm max = 200 = 20 A

Alarm min = aus (-16384 = $0xC000$)

Data request

| Master requests | Sensor: measured value | Sensor responds |
|-----------------|------------------------|---|
| 0 | | - |
| 1 | | - |
| 2 | | - |
| 3 | 4,5 V | 0x31 = Addr. 3 + value class 1 "Voltage" 0x005B = $45*2+1 = 4,5$ Volt + alarm flag |
| 4 | 12,5 V | 0x41 = Addr. 4 + value class 1 "Voltage" 0x00FA = $125*2+0 = 12,5$ Volt, no alarm flag |
| 5 | 18,3 A | 0x52 = Addr. 5 + value class 2 "Current" 0x016E = $183*2+0$ |
| 6 | | - |
| 7 | | - |
| 8 | | - |
| 9 | | - |
| 9 | | - |
| 10 | | - |
| 11 | | - |
| 12 | | - |
| 13 | | - |
| 14 | | - |
| 15 | | - |

4.11. MSB logo

For devices which are compatible with the MULTIPLEX Sensor Bus it is possible to use MSB-Logo, after consultation with MULTIPLEX. The MSB logo is trademarked by MULTIPLEX, so it is necessary to get a written permission.

After permission it is possible to use long and/or short version of MSB-logo.

Long version:



Short version:

