TLC / Vendor specific data Frames

Vendor specific data frames for communication with TLC signale modules
Revision 1.13 / 03.08.2022
0007-000020

Electronic Equipment for Traffic Systems

EETS GmbH
Chaltenbodenstrasse 6E
8834 Schindellegi
SWITZERLAND

TLC / Vendor specific data Frames

Vendor specific data frames for communication with TLC signale modules

Revision 1.13

Table of contents

1 Overview

TLS (Technischer Leitfaden für Streckenstationen) is a standard from the German BASt (Bundesanstalt für Strassenwesen, Federal Highway Research Institute). The present document is meant as extension to this document. It describes vendor specific data frames defined and used by EETS GmbH to communicate with its Traffic Light Controller TLC.

These vendor specific data frames will add or extend features on the OSI layer 7 for the TLS function groups 4 and 254 respectively.

2 Definitions and document conventions

2.1 Multibyte values

To transfer values that are larger then one byte, i.e. 2 bytes (16 bit) or 4 bytes (32 bit) we use the following abbreviations for the different components of the value:

Abbreviation	Description
LSB	least significant byte of a word (bits 0 – 7)
MSB	most significant byte of a word (bits 8 – 15)
LSW	least significant word of a double word (bits 0 – 15)
MSW	most significant word of a double word (bits 16 – 31)

For 32 bit values (double word values) the following combinations of the abbreviations shown in the previous table are used:

Abbreviation	Description
LSB/LSW	least significant byte of least significant word (bits 0 – 7)
MSB/LSW	most significant byte of least significant word (bits 8 – 15)
LSB/MSW	least significant byte of most significant word (bits 16 – 23)
MSB/MSW	most significant byte of most significant word (bits 24 – 31)

All multibyte values are transferred in Little-Endian format.

3 Data Frames

The description of the frames follows the style from the TLS 2012 standard and should therefore facilitate the integration and mixture with existing data frames.

3.1 DE block structure of type 130 “Channel-Current”

Used with ID 2 (status message) on response and ID 18 (status message) on request respectively.

Generally a TLC module will drive a traffic light containing 3 LED modules. Each of these 3 LED modules (usually red, yellow and green) is driven by a dedicated power switch. The “Channel-Current” data frame reports the currents consumed by each of the 3 LED modules. These values can be used to configure the TLC module.

The “Channel-Current” data frame is never sent spontaneously by the TLC module. The “Channel-Current” data frame is only sent upon request. The response data frame contains the 3 measured currents.

Position	*Description*	*Comment*
byte 1	length of DE block	[2 | 8]
byte 2	data terminal ID (DE)	[1 .. 254]
byte 3	type of DE data	[130]
byte 4	current channel 0 (red), LSB	*see blow*
byte 5	current channel 0 (red), MSB	*see blow*
byte 6	current channel 0 (yellow), LSB	*see blow*
byte 7	current channel 0 (yellow), MSB	*see blow*
byte 8	current channel 0 (green), LSB	*see blow*
byte 9	current channel 0 (green), MSB	*see blow*

On request the data frames length is 2, i.e. it contains no current values.

On response the current of every LED module is reported by using 2 bytes. These bytes report the current in entities of 100µA. The current of one channel is calculated according to the following formula: $I_{tot} = I_{LSB} + 256 \cdot I_{MSB}$

3.2 DE block structure of type 131 “Flasher-Synchronization”

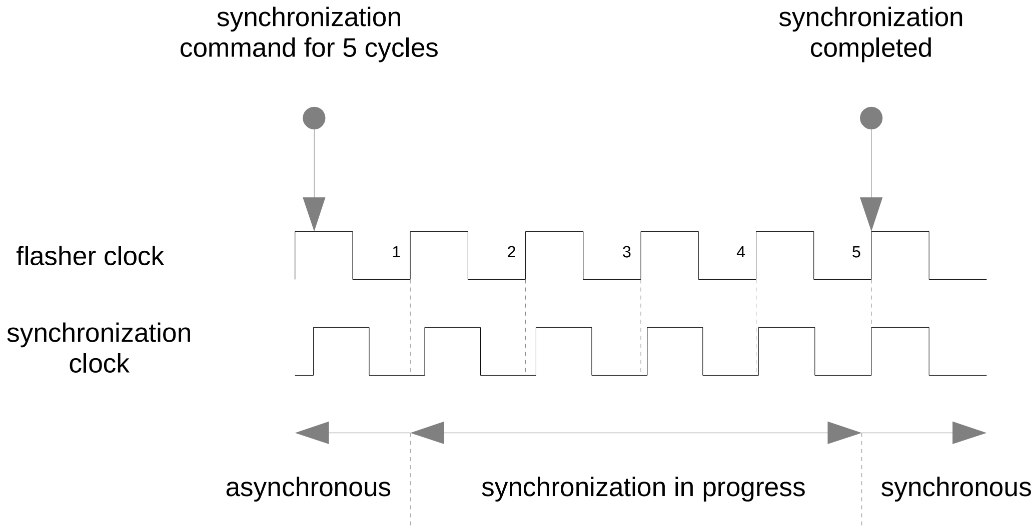
Used with ID 5 (command) on request.

TLC modules can be configured to create their proper flasher clock for driving yellow flashers. The data frame “Flasher-Synchronization” allows to synchronize this flasher clock amongst multiple TLC modules. All synchronized TLC modules will then be flashing in a consistent manner. It is self evident that all TLC modules to be synchronized must have been configured to generate the same flasher clock (frequency and duty cycle).

Position	*Description*	*Comment*
byte 1	length of DE block	[4]
byte 2	data terminal ID (DE)	[1 .. 254 | 255]
byte 3	type of DE data	[131]
byte 4	clock count for the synchronization	*see blow*
byte 5	location in synchronization queue	*see blow*

The data frame can be sent to the OSI2 broadcast address 255 too. This will synchronize all TLC modules connected to the bus on which the broadcast is emitted. In case of a broadcast a negative acknowledgment will never be emitted by any TLC module.

To prevent road users from taking note of the synchronization process the flashers are usually not hard synchronized by just resetting the flasher clock upon reception of the “Flasher-Synchronization” data frame. Instead the synchronization is spread over a certain period of time (multiple flashing clock phases). Hereby the flasher clock is accelerated or delayed continuously until it reaches the synchronization point. Bytes 4 and 5 of the data frame are used to configure this process. The process is shown in the following figure:



Data frame byte 4: Clock count for synchronization

The synchronization process will be spread across the number of clock cycles defined in this byte. Valid values are 0 .. 255. The value 0 will result in a hard synchronization, i.e. the flasher clock will be reset immediately. Generally this will result in an extended on-phase of the current flasher cycle which will most likely be noticed by the road user. The more clock cycles the synchronization process is spread across the less it will be visible by the road user.

Data frame byte 5: location in synchronization queue

Valid values are 0 .. 255. Generally flasher synchronization of TLC modules connected to same bus takes place by sending the data frame to the broadcast address. This ensures that the synchronization process is started at exactly the same time on all modules. If synchronization takes place by means of the broadcast address and the synchronization does not spread across multiple buses, the value of the data frame byte *location in synchronization queue* should be set to 0. If however the TLC modules are addressed individually or the synchronization spreads across multiple buses the data frame byte *location in synchronization queue* allows for an additional delay. If for example 6 TLC modules are to be addressed individually, the **first** should be passed a *location in synchronization queue* value of **6**, the **second** a value of **5** etc. until the **last** which should receive a value of **1**. The values are therefore assigned in the opposite way the TLC modules are addressed.

Timing recommendations for best synchronization results

To make even smallest clock skews between different TLC modules invisible for the road user we recommend to synchronize approximately every 1 hour. When ever possible this should be done using a broadcast. The synchronization is best spread across 20 to 30 flasher cycles.

3.3 DE block structure of type 132 “OSI2-Address”

Used with ID 3 (parameter) on request.

TLC modules do not have any means (such as DIP switches) to modify their OSI2 address. However it is imperative that all TLC modules connected to the same bus have their dedicated and unique OSI2 address. The factory default OSI2 address of all TLC modules is 1.

The data frame “OSI2-Address” allows to modify the OSI2 address of a TLC module.

Position	*Description*	*Comment*
byte 1	length of DE block	[3]
byte 2	data terminal ID (DE)	[255]
byte 3	type of DE data	[132]
byte 4	new OSI2 address	[1 .. 199]

This data frame has to be sent to the *system control* function group (FG254).

WARNING: Immediately after successful reception of this data frame the TLC module will reset itself. After the reset the TLC module will only be accessible by its new OSI2 Address of course.

3.4 DE block structure of type 133 “Version”

Used with ID 3 (parameter) on response and ID 19 (parameter) on request respectively.

The data frame “Version” allows to request the software as well as the hardware version of a TLC module. The hardware version number is used to exactly describe the TLC modules hardware including the assembly variant.

Position	*Description*	*Comment*
byte 1	length of DE block	[2 | 8]
byte 2	data terminal ID (DE)	[255]
byte 3	type of DE data	[133]
byte 4	hardware version number (major)	[0 .. 255]
byte 5	hardware version number (minor)	[0 .. 255]
byte 6	hardware version number (revision)	[0 .. 255]
byte 7	software version number (major)	[0 .. 255]
byte 8	software version number (minor)	[0 .. 255]
byte 9	software version number (revision)	[0 .. 255]

On request the data frames length is 2 and has to be sent to the *system control* function group (FG254). On response the data frames length is 8.

3.5 DE block structure of type 134 “Statistics”

Used with ID 3 (parameter) on response and ID 19 (parameter) on request respectively.

The data frame “Statistics” allows to retrieve statistical information collected by the TLC during its up time. To retain as much information as possible the counters are usually multibyte, i.e. 16 or even 32 bit in size.

Position	*Description*	*Comment*
byte 1	length of DE block	[2 | 72]
byte 2	data terminal ID (DE)	[255]
byte 3	type of DE data	[134]
byte 4	total bytes received (RX)	LSB/LSW
byte 5	total bytes received (RX)	MSB/LSW
byte 6	total bytes received (RX)	LSB/MSW

Position	*Description*	*Comment*
byte 7	total bytes received (RX)	MSB/MSB
byte 8	total bytes transmitted (TX)	LSB/LSW
byte 9	total bytes transmitted (TX)	MSB/LSW
byte 10	total bytes transmitted (TX)	LSB/MSW
byte 11	total bytes transmitted (TX)	MSB/MSB
byte 12	bit framing errors	LSB/LSW
byte 13	bit framing errors	MSB/LSW
byte 14	bit framing errors	LSB/MSW
byte 15	bit framing errors	MSB/MSW
byte 16	bit parity errors	LSB/LSW
byte 17	bit parity errors	MSB/LSW
byte 18	bit parity errors	LSB/MSW
byte 19	bit parity errors	MSB/MSW
byte 20	bit overrun errors	LSB/LSW
byte 21	bit overrun errors	MSB/LSW
byte 22	bit overrun errors	LSB/MSW
byte 23	bit overrun errors	MSB/MSW
byte 24	OSI-2 long frames received (RX)	LSB/LSW
byte 25	OSI-2 long frames received (RX)	MSB/LSW
byte 26	OSI-2 long frames received (RX)	LSB/MSW
byte 27	OSI-2 long frames received (RX)	MSB/MSW
byte 28	OSI-2 short frames received (RX)	LSB/LSW
byte 29	OSI-2 short frames received (RX)	MSB/LSW
byte 30	OSI-2 short frames received (RX)	LSB/MSW
byte 31	OSI-2 short frames received (RX)	MSB/MSW
byte 32	OSI-2 acknowledges received (RX)	LSB/LSW
byte 33	OSI-2 acknowledges received (RX)	MSB/LSW
byte 34	OSI-2 acknowledges received (RX)	LSB/MSW
byte 35	OSI-2 acknowledges received (RX)	MSB/MSW
byte 36	OSI-2 broadcast frames received (RX)	LSB/LSW
byte 37	OSI-2 broadcast frames received (RX)	MSB/LSW
byte 38	OSI-2 broadcast frames received (RX)	LSB/MSW

Position	*Description*	*Comment*
byte 39	OSI-2 broadcast frames received (RX)	MSB/MSW
byte 40	OSI-2 rejected frames (RX)	LSB/LSW
byte 41	OSI-2 rejected frames (RX)	MSB/LSW
byte 42	OSI-2 rejected frames (RX)	LSB/MSW
byte 43	OSI-2 rejected frames (RX)	MSB/MSW
byte 44	OSI-2 long frames transmitted (TX)	LSB/LSW
byte 45	OSI-2 long frames transmitted (TX)	MSB/LSW
byte 46	OSI-2 long frames transmitted (TX)	LSB/MSW
byte 47	OSI-2 long frames transmitted (TX)	MSB/MSW
byte 48	OSI-2 short frames transmitted (TX)	LSB/LSW
byte 49	OSI-2 short frames transmitted (TX)	MSB/LSW
byte 50	OSI-2 short frames transmitted (TX)	LSB/MSW
byte 51	OSI-2 short frames transmitted (TX)	MSB/MSW
byte 52	OSI-2 acknowledges transmitted (TX)	LSB/LSW
byte 53	OSI-2 acknowledges transmitted (TX)	MSB/LSW
byte 54	OSI-2 acknowledges transmitted (TX)	LSB/MSW
byte 55	OSI-2 acknowledges transmitted (TX)	MSB/MSW
byte 56	OSI-7 frames received (RX)	LSB/LSW
byte 57	OSI-7 frames received (RX)	MSB/LSW
byte 58	OSI-7 frames received (RX)	LSB/MSW
byte 59	OSI-7 frames received (RX)	MSB/MSW
byte 60	OSI-7 broadcast frames received (RX)	LSB/LSW
byte 61	OSI-7 broadcast frames received (RX)	MSB/LSW
byte 62	OSI-7 broadcast frames received (RX)	LSB/MSW
byte 63	OSI-7 broadcast frames received (RX)	MSB/MSW
byte 64	OSI-7 rejected frames (RX)	LSB/LSW
byte 65	OSI-7 rejected frames (RX)	MSB/LSW
byte 66	OSI-7 rejected frames (RX)	LSB/MSW
byte 67	OSI-7 rejected frames (RX)	MSB/MSW
byte 68	OSI-7 frames transmitted (TX)	LSB/LSW
byte 69	OSI-7 frames transmitted (TX)	MSB/LSW
byte 70	OSI-7 frames transmitted (TX)	LSB/MSW

Position	*Description*	*Comment*
byte 71	OSI-7 frames transmitted (TX)	MSB/MSW
byte 72	Operating hours	LSB
byte 73	Operating hours	MSB

On request the data frames length is 2 and has to be sent to the *system control* function group (FG254). On response the data frames length is 72.

All communication related statistics will we reset to zero each time the TLC module is restarted.

3.6 DE block structure of type 135 “Serial Number”

Used with ID 3 (parameter) on response and ID 19 (parameter) on request respectively.

The data frame “Serial Number” allows to request the serial number of the TLC module.

Position	*Description*	*Comment*
byte 1	length of DE block	[2 | 10 | 14]
byte 2	data terminal ID (DE)	[255]
byte 3	type of DE data	[135]
byte 4	serial number byte 1	[0 .. 255]
byte 5	serial number byte 2	[0 .. 255]
byte 6	serial number byte 3	[0 .. 255]
byte 7	serial number byte 4	[0 .. 255]
byte 8	serial number byte 5	[0 .. 255]
byte 9	serial number byte 6	[0 .. 255]
byte 10	serial number byte 7	[0 .. 255]
byte 11	serial number byte 8	[0 .. 255]
byte 12	serial number byte 9	[0 .. 255]
byte 13	serial number byte 10	[0 .. 255]
byte 14	serial number byte 11	[0 .. 255]
byte 15	serial number byte 12	[0 .. 255]

On request the data frame length is 2 and has to be sent to the *system control* function group (FG254). On response the data frame length is 10 or 14 depending on the format of the serial number.

Format: EUI-48/EUI-64, 64-bit serial number (8 byte)
with data frame length 10

Byte1 to byte 3 describes the Organizationally Unique Identifier (OUI)

Byte 4 to byte 8 represents the Extension Identifier (EI).

Consider that always the 8 byte EUI-64 format is transmitted.

Note: EUI-48 to EUI-64 address encapsulation convention

EUI-48: 00-04-A3-12-34-56
EUI-64: 00-04-A3-**FF**-**FE**-12-34-56

Format: STM MCU unique device identifier UID-96, 96-bit serial number (12 byte)
with data frame length 14

The 96-bit unique device identifier provides a reference number which is unique for any device and in any context. These bits cannot be altered by the user.

UID-96 structure according to STM32 MCU registers:
(STMicroelectronics, STM32F0x referenz manual RM0091, chapter 33.1 Unique device ID register)

UID[31:0]: X and Y coordinates on the wafer expressed in BCD format

UID[39:32]: WAF_NUM[7:0] Wafer number (8-bit unsigned number)

UID[63:40]: LOT_NUM[23:0] Lot number (ASCII encoded)

UID[95:64]: LOT_NUM[55:24] Lot number (ASCII encoded)

Consider that in type 135 frame the UID-96 serial number is transmitted byte-wise in straight bit order UID[0]...UID[96].

3.7 DE block structure of type 252 and type 254

This data frames are exclusively used by manufacturer/vendor for internal or vendor-specific purpose!

4 List of revisions

Revision	Date	Comment
1.00	30.11.2010	first draft
1.01	11.03.2012	data frame type 131
1.02	25.05.2012	fixed data frame lengths of type 130 and 131
1.03	29.05.2012	typos
1.04	30.05.2012	added data frames 132 and 133, TLC-001 is now simply TLC
1.05	13.06.2012	fixed request identifier for data frames 130 and 133
1.06	27.08.2012	added timing recommendations for flasher synchronization
1.07	05.09.2012	clarified the broadcast address type for data frame 131
1.08	30.10.2012	added data frame 134
1.09	08.04.2019	added data frame 135, fixed wrong FG of control function group
1.10	05.01.2021	Extended data frame 135 to longer UID-96
1.11	06.12.2021	Explanatory Supplement for data frame 135
1.12	07.03.2022	added note on internal data frame 252 and 254
1.13	02.08.2022	Modification on internal data frame 252