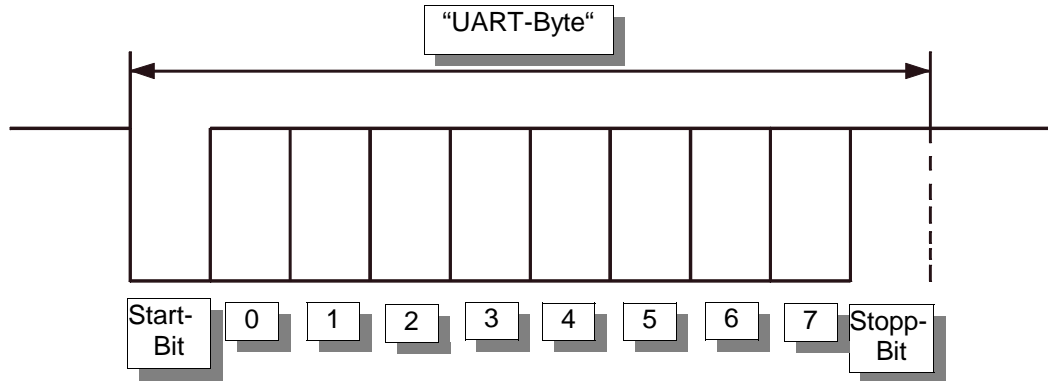


**5.3 Nachrichtenformate beim LIN**

**5.3.1 Aufbau der LIN-Botschaft**

Die LIN-Botschaft beruht auf einem seriellen Kommunikationsprotokoll, das sich mit einer Ausnahme (dem Synchronisationsbreak) auf das UART-Protokoll (Universal Asynchronous Receiver Transmitter) mit zehn Bits bezieht. Das „UART-Byte“ bestehen aus einem Startbit, acht Datenbits und einem Stopbit.



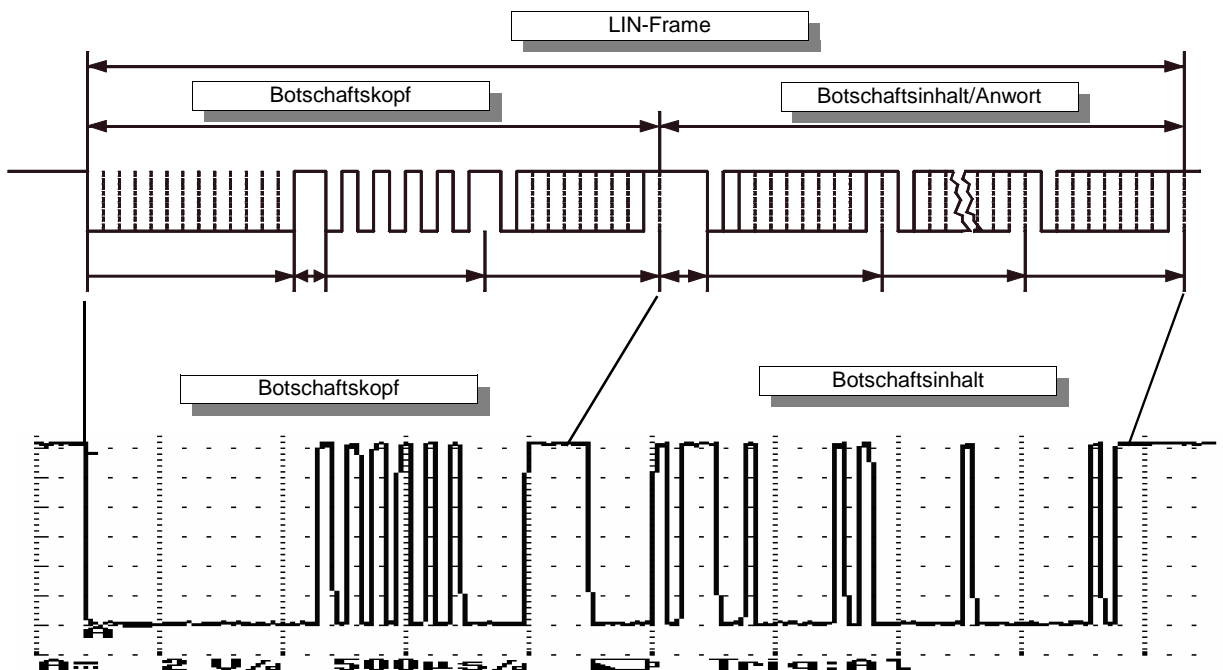
**UART-Protokoll**

LIN eignet sich speziell zum Aufbau einfacher Netzwerke, in der ein Master, wie z.B. eine elektrische Steuereinheit mit CAN-Anschluss (Fahrzeugtechnik), einen oder mehrere Slaves steuert. Ein Slave darf nur senden, wenn es die Berechtigung durch den Master bekommt. LIN unterstützt sowohl den Master – Slave – Dialog als auch eine Slave – Slave – Kommunikation. Im Gegensatz zu klassischen Master-Slave-Bussen, die mit Teilnehmeradressen arbeiten, verwendet man beim LIN eine Nachrichtenadressierung (vergleiche CAN).

**5.3.2 LIN-Frame (LIN-Botschaft)**

Ein kompletter LIN-Frame besteht aus dem Header (Botschaftskopf) und dem Response (Antwortfeld oder Botschaftsinhalt). Den grundsätzlichen Aufbau zeigt die folgende Abbildung.

Ergänzen Sie in der folgenden Abbildung die fehlenden Bezeichnungen: LIN-Frame, Botschaftskopf, Botschaftsinhalt.



**LIN-Botschaft**

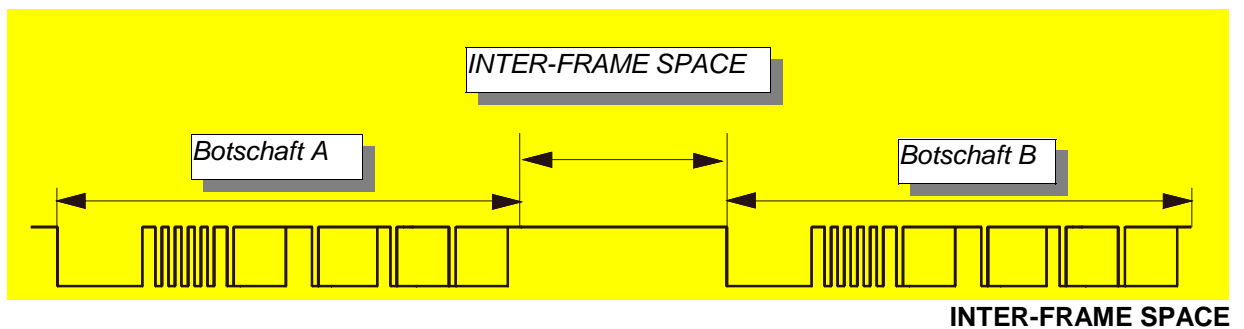
Jede Botschaft wird vom Master gestartet und beginnt mit dem Header, der aus einem Synchronisationsbrake, einem Synchronisationsbyte und einem Message-Identifizier besteht. Der Header ist der Botschaftskopf des zu übertragenden Datenpaketes. Er enthält keine Nutzdaten. Nach dem Identifizier beginnt das Response-Feld mit den eigentlichen Nutzdaten in den Datenfeldern (Daten-Field) mit 1 bis 8 Datenbytes und der Checksumme im Checksum Field. Das Response kann von einem der Slaves oder vom Master selbst ausgesandt werden.

### 5.3.3 Aufbau der einzelnen Botschaftsabschnitte

#### 5.3.3.1 INTER-FRAME SPACE

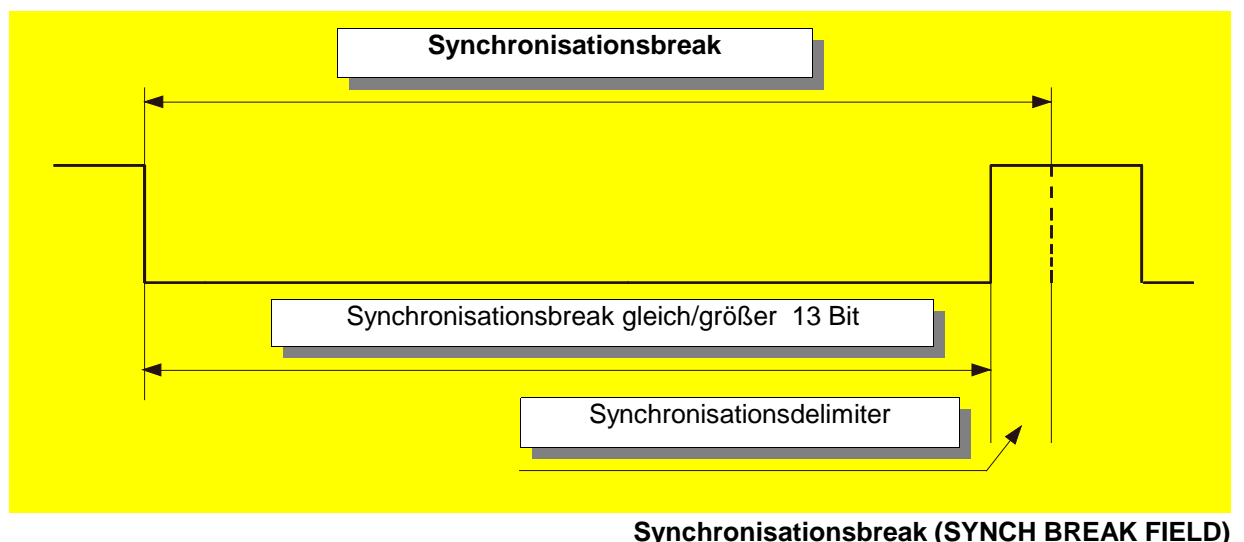
Der INTER-FRAME SPACE Abschnitt bezeichnet die Pause zwischen den Botschaften. Die Zeitdauer ist nicht festgelegt und kann vom Systementwickler bestimmt werden. Dies ist sinnvoll, weil in einigen Systemen die Knoten (Master und Slaves) mehr Zeit für die Ausführung bestimmter Aufgaben benötigen.

Ergänzen Sie in der folgenden Abbildung die Bezeichnungen: INTER-FRAME SPACE, Botschaft A, Botschaft B.



#### 5.3.3.2 Synchronisationsbreak (SYNCH BREAK FIELD)

Nach der Pause beginnt die neue Botschaft immer mit dem Synchronisationsfeld, das den Anfang einer Nachricht kennzeichnet. Es besteht aus einem Synchronisationsbreak (synchronisations Abschnitt) und dem Synchronisationsdelimiter (Synchronisationsbegrenzer). Die Länge des Synchronisationsbreak ist mindestens 13 Taktzeiten (13 Bits) lang. Dieser lange Zeitabschnitt ist notwendig, damit Slaves mit einer sehr ungenauen Taktbasis (es ist eine Abweichung von 14% erlaubt) den Synchronisationsbreak eindeutig erkennen. Die mindestens 13 dominanten Bits werden von mindestens einem rezessiven Bit, dem Delimiter, begrenzt.



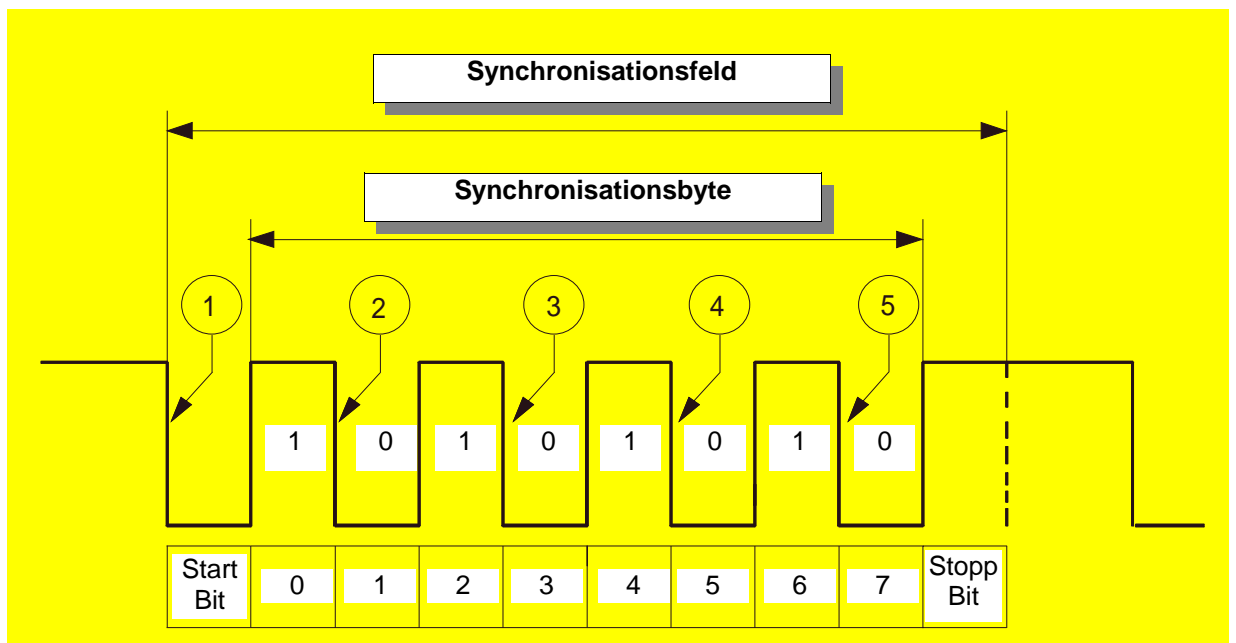
Synchronisationsbreak (SYNCH BREAK FIELD)

### 5.3.3.3 Synchronisationsfeld

Nachdem die Slaves 11 dominante Bits im Synchronisationsbreak registriert haben, schalten diese auf den Synchronisationsvorgang um. Er beginnt mit dem Synchronisationsfeld. In ihm sendet der Master immer das Byte 55h. Zusammen mit dem Startbit, dem Synchronisationsbyte und dem rezessiven Stopp-Bit oder Synchronisationsdelimiter für die Synchronisationsbegrenzung zeigt sich die in der Abbildung gezeigte Bitfolge.

Die Slaves können den Systemtakt des Masters (MASTER CLOCK) über ihre Timer-Elektronik eindeutig bestimmen und ihre eigene Zeitbasis bzw. Uhr mit dem Takt des Masters in Einklang bringen bzw. synchronisieren. Der eben beschriebene Vorgang ermöglicht den Einsatz kostengünstiger Slave-Elektronik mit RC-Oszillatoren anstelle von teuren Quarz-Oszillatoren.

Nachdem die Slaves den Bezug zur Taktrate des Busses hergestellt haben, darf sich ihre Zeitbasis während der Übertragung nur um maximal 2% ändern, um eine sichere Übertragung zu gewährleisten.



Synchronisationsfeld

#### Aufgaben 1:

Wandeln Sie die Zahl 55h in eine Dualzahl um und vergleichen Sie das Bitmuster mit dem oben gezeichneten Signal.

Was stellen Sie fest?

Die Zahl 55h als Dualzahl : 0 1 0 1 0 1 0 1

Vergleicht man das Bitmuster der Dualzahl mit der Bitfolge in der Botschaft, stellt man fest, dass das niederwertigste Bit zuerst gesendet wird.

#### Aufgabe 2:

Vergleichen Sie das Sendeverfahren des LIN mit dem des CAN. Erläutern Sie Ihre Beobachtung.

Beim CAN-Bus erfolgt die Übertragung des niederwertigen Bits nicht an erster, sondern an letzter Stelle

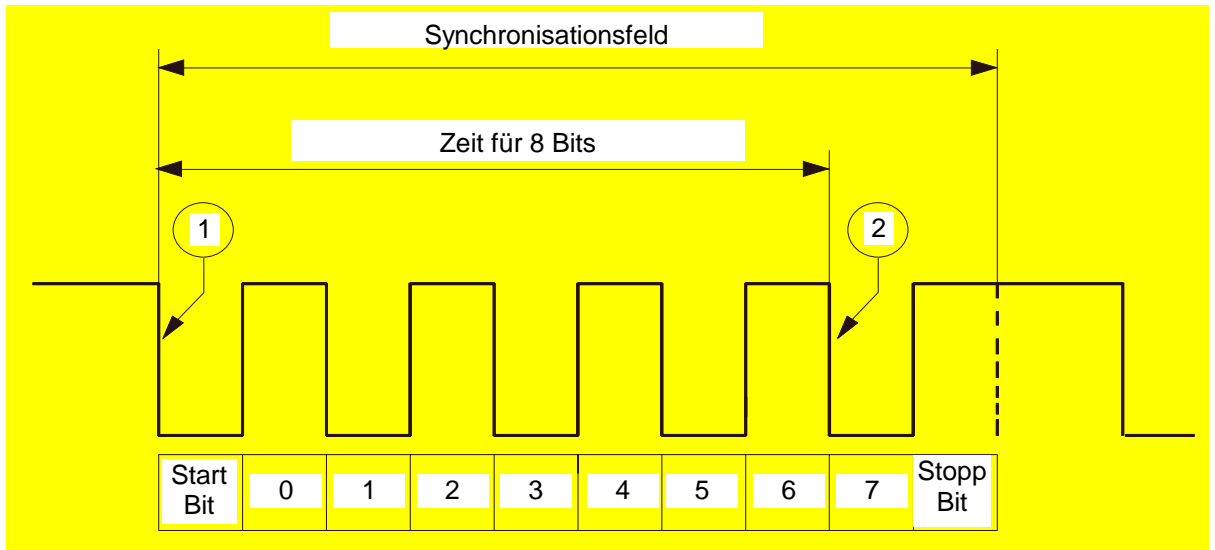
#### Aufgabe 3:

Ordnen Sie dem Synchronisationsbyte (in der Abbildung) das Bitmuster der Hexadezimal 55h zu.

**Ablauf der Synchronisierung**

Mit der fallenden Flanke des Start-Bits beginnen die Slaves mit der Bestimmung der Zeitspanne für 8 Bits. Die registrierte Zeit wird durch 8 geteilt und liefert die einzustellende Baudrate.

Grundsätzlich lässt die ISO 9141 beim LIN eine Übertragungsrate zwischen 20 und 1000 kBaud zu. In der Praxis sind in Europa Baudraten von 9,6 oder 19,2 kBit/s und in Amerika 10,4 kBit/s üblich.



**Synchronisationsfeld mit den zu vermessenden Flanken**

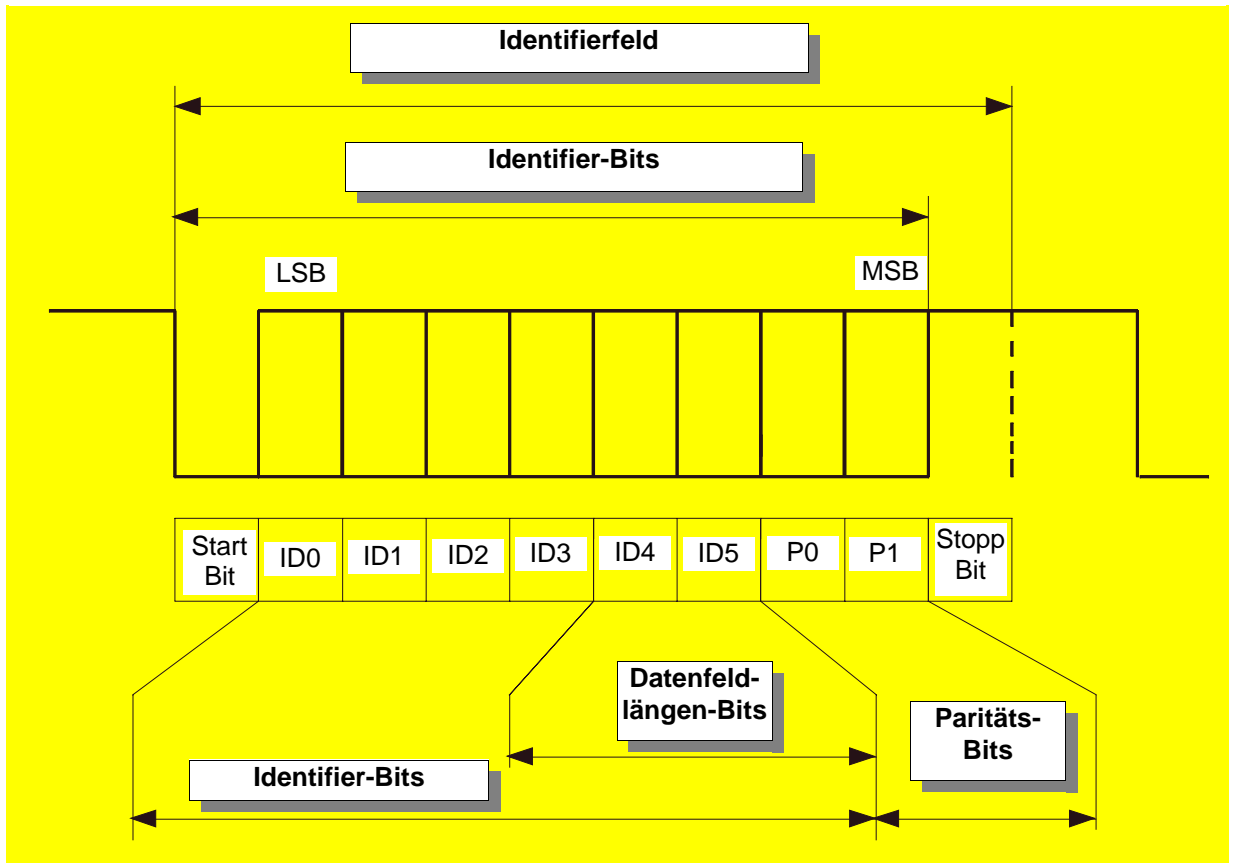
Beim Messvorgang dürfen nur die fallenden Flanken verwendet werden, da diese einen deutlich steileren Verlauf aufweisen als die steigenden. Die Unterschiede lassen sich durch die elektrische Schaltung des Transceivers erklären, da der Transistor den Buspegel schneller auf Null-Potential ziehen kann als der Pull-Up-Widerstand den Low-Pegel auf High.

**Aufgabe:**

Oszilloskopieren Sie eine LIN-Botschaft und vergrößern Sie das Oszillogramm so weit, dass die Steilheit der fallenden und steigenden Flanken zu bestimmen ist.

### 5.3.3.4 Identifizierfeld (IDENTIFIER FIELD)

Das Identifizierfeld ist das dritte und letzte Byte des Headers. Es besteht aus einem Start-Bit, sechs Identifier-Bits, zwei Paritäts-Bits und einem Stopp-Bit. Mit den Identifier-Bits ID0 bis ID5 können bis zu 64 verschiedene Botschaften mit einer Kennung versehen werden. Die zwei Paritätsbits P0 und P1 dienen der Datensicherheit.



**IDENTIFIER FIELD**

In der LIN-Spezifikation bis V1.2 geben die Bits ID4 und ID5 (Datenfeldlängen-Bits) die Anzahl der Datenbytes im RESPONSE fest vor. Ab der Version 1.3 ist die Kodierung an dieser Stelle nicht mehr verbindlich. Die Datenfeldlänge ist dann bei der Nachrichtendefinition im LIN Description File anzugeben.

ID5	ID4	Anzahl der Datenbytes im RESPONSE
0	0	2
0	1	2
1	0	4
1	1	8

**Anzahl der übertragbaren Datenbytes in Abhängigkeit von den Datenfeldlängen-Bits**

Von den 64 möglichen Identifiern stehen nicht alle zur Nachrichtenadressierung zur Verfügung, da bestimmte Kennungen reserviert sind.

Die Identifier 3Ch und 3Dh sind für Command und die Identifier 3Eh und 3Fh für Extended-Frames reserviert. Die Command Telegramme dienen Servicezwecken und Extended-Frames sind für zukünftige LIN-Erweiterungen vorgesehen.

**Paritäts-Bits**

Zur Sicherheit wird der Identifier auf Bitfehler untersucht und einer Paritätsprüfung unterzogen. Dabei wird die Summe der dominanten Identifier-Bits wie folgt miteinander verknüpft.

$$P0 = ID0 + ID1 + ID2 + ID4 \text{ (even)}$$

$$P1 = ID1 + ID3 + ID4 + ID5 \text{ (odd)}$$

Beim even-parity-Bit P0 wird die Anzahl der in der vorgegebenen Bitfolge auftretenden dominanten Bits durch das Paritätsbit zu einer geraden Anzahl ergänzt. Beim odd-parity-Bit wird dagegen durch das Hinzufügen des Paritätsbits eine ungerade Anzahl dominanter Bits erzeugt.

**Aufgabe 1:**

Ergänzen Sie zu den Beispielen den jeweiligen Zustand des Paritätsbits.

P0	ID0	ID1	ID2	ID4
1	1	1	0	1
0	0	1	0	1

P1	ID1	ID3	ID4	ID5
0	1	1	0	1
1	0	1	0	1

**Aufgabe 2:**

Wieviel Datenbytes werden bei dem folgenden Identifier-Bits in der Botschaftsantwort enthalten sein und welchen Zustand nehmen die Paritätsbits ein?

ID0	ID1	ID2	ID3	ID4	ID5	P0	P1
1	0	1	1	0	1	0	1

**Aufgabe 3:**

Geben Sie die Botschaftskennung (den Identifier) als Hexadezimalzahl und Dezimalzahl an.

P1	P0	ID5	ID4	ID3	ID2	ID1	ID0
1	0	1	0	1	1	0	1

2h	Dh
----	----

Lösung:

Die Hexadezimalzahl ist: 2Dh , Die Dezimalzahl ist:  $2 \cdot 16^1 = 2 \cdot 16 = 32$

$$13 \cdot 16^0 = 13 \cdot 1 = \frac{13}{45}$$

**Aufgabe 4:**

Welchen hexadezimalen- und dezimalen Wert besitzt das ID-FIELD? Lösung: Adh , 173d

Tabelle der verfügbaren Identifier

ID[0..5]		P0= ID0⊕ID1⊕ID2⊕ID4	P1= ID1⊕ID3⊕ID4⊕ID5	ID-Field 7 6 5 4 3 2 1 0	ID-Field		# of Data Bytes
Dec	Hex				Dec	Hex	
0	0x00	0	1	1 0 0 0 0 0 0 0	128	0x80	2
1	0x01	1	1	1 1 0 0 0 0 0 0	193	0xC1	2
2	0x02	1	0	0 1 0 0 0 0 0 1	66	0x42	2
3	0x03	0	0	0 0 0 0 0 0 0 1	3	0x03	2
4	0x04	1	1	1 1 0 0 0 0 1 0	196	0xC4	2
5	0x05	0	1	1 0 0 0 0 0 1 0	133	0x85	2
6	0x06	0	0	0 0 0 0 0 0 1 1	6	0x06	2
7	0x07	1	0	0 1 0 0 0 0 1 1	71	0x47	2
8	0x08	0	0	0 0 0 0 0 1 0 0	8	0x08	2
9	0x09	1	0	0 1 0 0 0 1 0 0	73	0x49	2
10	0x0A	1	1	1 1 0 0 0 1 0 1	202	0xCA	2
11	0x0B	0	1	1 0 0 0 0 1 0 1	139	0x8B	2
12	0x0C	1	0	0 1 0 0 0 1 1 0	76	0x4C	2
13	0x0D	0	0	0 0 0 0 0 1 1 0	13	0x0D	2
14	0x0E	0	1	1 0 0 0 0 1 1 1	142	0x8E	2
15	0x0F	1	1	1 1 0 0 0 1 1 1	207	0xCF	2
16	0x10	1	0	0 1 0 1 0 0 0 0	80	0x50	2
17	0x11	0	0	0 0 0 1 0 0 0 1	17	0x11	2
18	0x12	0	1	1 0 0 1 0 0 1 0	146	0x92	2
19	0x13	1	1	1 1 0 1 0 0 1 1	211	0xD3	2
20	0x14	0	0	0 0 0 1 0 1 0 0	20	0x14	2
21	0x15	1	0	0 1 0 1 0 1 0 1	85	0x55	2
22	0x16	1	1	1 1 0 1 0 1 1 0	214	0xD6	2
23	0x17	0	1	1 0 0 1 0 1 1 1	151	0x97	2
24	0x18	1	1	1 1 0 1 1 0 0 0	216	0xD8	2
25	0x19	0	1	1 0 0 1 1 0 0 1	153	0x99	2
26	0x1A	0	0	0 0 0 1 1 0 1 0	26	0x1A	2
27	0x1B	1	0	0 1 0 1 1 0 1 1	91	0x5B	2
28	0x1C	0	1	1 0 0 1 1 1 0 0	156	0x9C	2
29	0x1D	1	1	1 1 0 1 1 1 0 1	221	0xDD	2
30	0x1E	1	0	0 1 0 1 1 1 1 0	94	0x5E	2
31	0x1F	0	0	0 0 0 1 1 1 1 1	31	0x1F	2
32	0x20	0	0	0 0 1 0 0 0 0 0	32	0x20	4
33	0x21	1	0	0 1 1 0 0 0 0 1	97	0x61	4
34	0x22	1	1	1 1 1 0 0 0 1 0	226	0xE2	4

ID[0..5]		P0= ID0⊕ID1⊕ID2⊕ID4	P1= ID1⊕ID3⊕ID4⊕ID5	ID-Field 7 6 5 4 3 2 1 0	ID-Field		# of Data Bytes
Dec	Hex				Dec	Hex	
35	0x23	0	1	1 0 1 0 0 0 1 1	163	0xA3	4
36	0x24	1	0	0 1 1 0 0 1 0 0	100	0x64	4
37	0x25	0	0	0 0 1 0 0 1 0 1	37	0x25	4
38	0x26	0	1	1 0 1 0 0 1 1 0	166	0xA6	4
39	0x27	1	1	1 1 1 0 0 1 1 1	231	0xE7	4
40	0x28	0	1	1 0 1 0 1 0 0 0	168	0xA8	4
41	0x29	1	1	1 1 1 0 1 0 0 1	233	0xE9	4
42	0x2A	1	0	0 1 1 0 1 0 1 0	106	0x6A	4
43	0x2B	0	0	0 0 1 0 1 0 1 1	43	0x2B	4
44	0x2C	1	1	1 1 1 0 1 1 0 0	236	0xEC	4
45	0x2D	0	1	1 0 1 0 1 1 0 1	173	0xAD	4
46	0x2E	0	0	0 0 1 0 1 1 1 0	46	0x2E	4
47	0x2F	1	0	0 1 1 0 1 1 1 1	111	0x6F	4
48	0x30	1	1	1 1 1 1 0 0 0 0	240	0xF0	8
49	0x31	0	1	1 0 1 1 0 0 0 1	177	0xB1	8
50	0x32	0	0	0 0 1 1 0 0 1 0	50	0x32	8
51	0x33	1	0	0 1 1 1 0 0 1 1	115	0x73	8
52	0x34	0	1	1 0 1 1 0 1 0 0	180	0xB4	8
53	0x35	1	1	1 1 1 1 0 1 0 1	245	0xF5	8
54	0x36	1	0	0 1 1 1 0 1 1 0	118	0x76	8
55	0x37	0	0	0 0 1 1 0 1 1 1	55	0x37	8
56	0x38	1	0	0 1 1 1 1 0 0 0	120	0x78	8
57	0x39	0	0	0 0 1 1 1 0 0 1	57	0x39	8
58	0x3A	0	1	1 0 1 1 1 0 1 0	186	0xBA	8
59	0x3B	1	1	1 1 1 1 1 0 1 1	251	0xFB	8
60 <sup>a</sup>	0x3C	0	0	0 0 1 1 1 1 0 0	60	0x3C	8
61 <sup>b</sup>	0x3D	1	0	0 1 1 1 1 1 0 1	125	0x7D	8
62 <sup>c</sup>	0x3E	1	1	1 1 1 1 1 1 1 0	254	0xFE	8
63 <sup>d</sup>	0x3F	0	1	1 0 1 1 1 1 1 1	191	0xBF	8

Quelle: Motorola GmbH

**a**

Identifizier 60d (3Ch) ist für das MASTER REQUEST COMMAMND FRAME reserviert.

**b**

Identifizier 61d (3Dh) ist für das SLAVE REPONSE COMMAND FRAME belegt.

**c**

Identifizier 62d (3Eh) ist für zukünftige Botschaftserweiterungen reserviert. Es kann für eigene Telegramme verwendet werden.

**d**

Identifizier 63d (3Fh) ist für zukünftige LIN Erweiterungen vorgesehen.



### 5.3.3.5 Botschaftskopf (HEADER)

Der HEADER wird vom LIN-Master regelmäßig gesendet. Er lässt sich in die in den vorigen Abschnitten erläuterten Bereiche einteilen.

#### Aufgabe 1:

Nennen Sie die Bereiche, in die der Botschaftskopf oder Header unterteilt werden kann.

Nach der Pause zwischen den Botschaften (dem INTER-FRAME-SPACE) folgen die Bereiche:

*Synchronisationsbreak*

*Synchronisationsfeld*

*Identifizierfeld*

#### Aufgabe 2:

Die Abbildung auf der nächsten Seite zeigt das Oszillogramm eines LIN-Master-Steuergerätes, an dem keine Slaves angeschlossen sind.

Erläutern Sie, warum im Oszillogramm nur der Botschaftskopf zu sehen ist.

*Da die Slaves nicht angeschlossen sind, gibt es in dieser Botschaft auch keinen Antwortteil (Response). Die Botschaft wird vom Master immer wiederholt, da er keine Antwort erhält.*

#### Aufgabe 3:

Ergänzen Sie im abgebildeten Oszillogramm (folgende Seite) des Botschaftskopfes (Header)

- a) die unter 1 genannten Bereiche
- b) die Bit-Bezeichnungen (z.B. ID1)
- c) den Bit-Zustand (high oder low)
- d) die Stopp- und Startbits

Hinweis:

Das Oszillogramm einer Botschaft ohne Response kann z.B. beobachtet werden, wenn Steuergeräte für Sonderausstattungen nicht verbaut wurden. Das Signal hat keine Auswirkungen auf die Funktion des Systems.

Aufgabe 4:

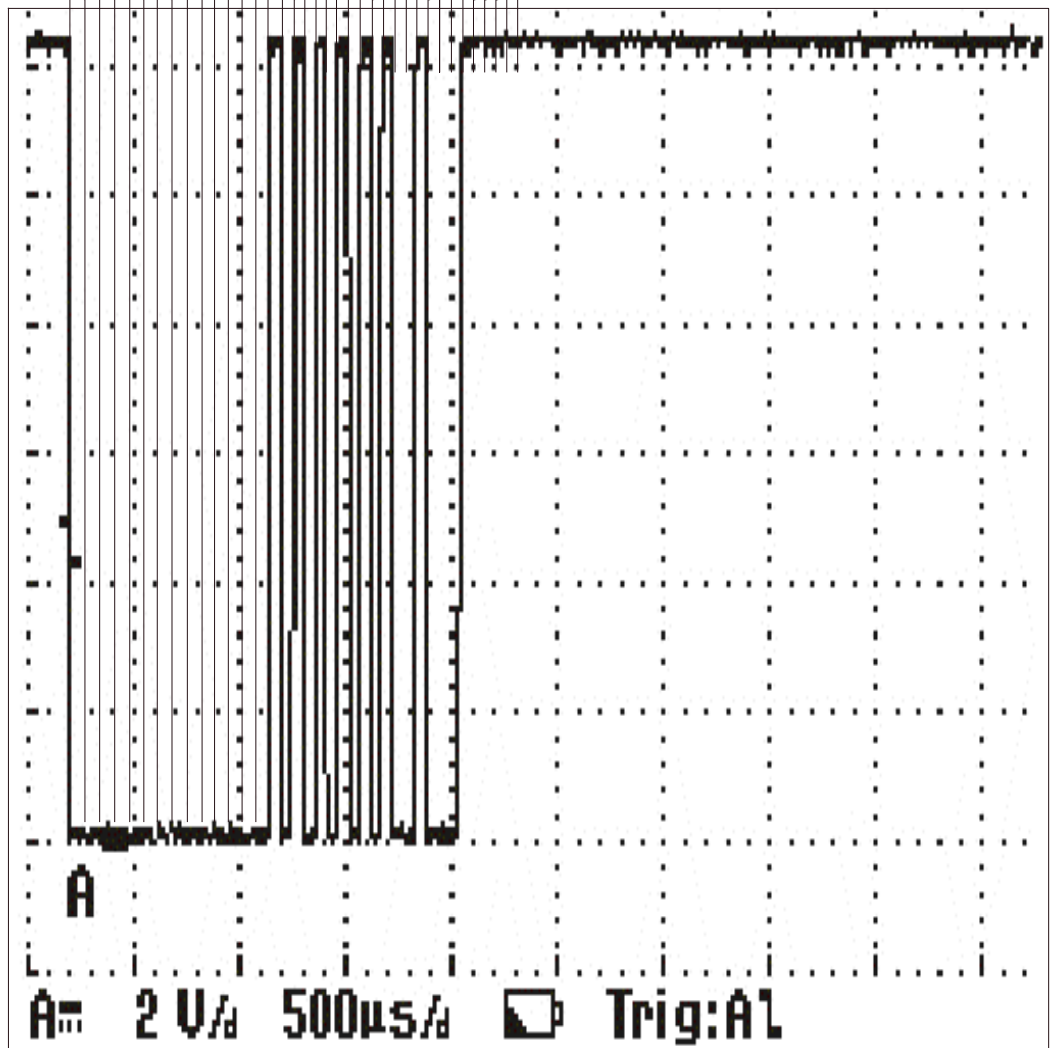
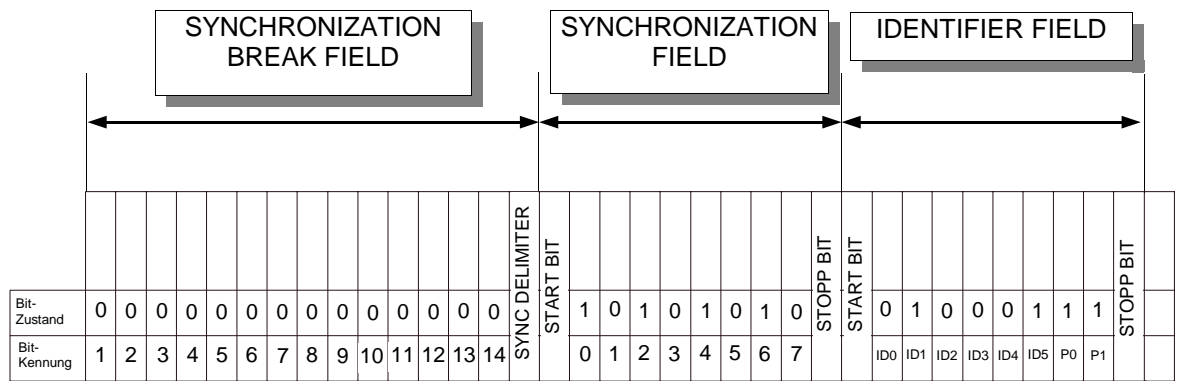
Bestimmen Sie die Übertragungsrate des Systems.

9 Bit	in	490 $\mu$ s		$0,00005444s$	—————	1 Bit
1 Bit	in	54,44 $\mu$ s		1s	—————	X Bit

$$X = 18\,382 \text{ Bit/s} = \text{ca } 18 \text{ kBit/s}$$

*Der LIN-Bus arbeitet mit 20kbit/s*

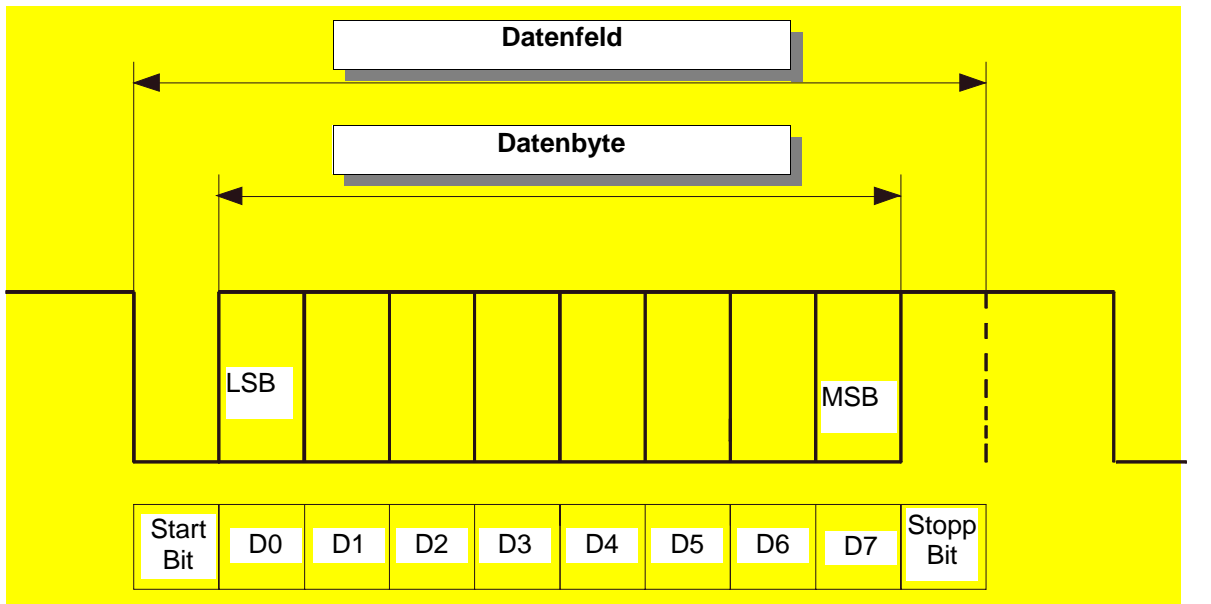
**Botschaftskopf (HEADER)**



**5.3.3.6 Datenfeld**

Nachdem der Master das Identifier Feld als letztes im Header gesendet hat , können nach dem Stoppsbit des Identifierfeldes und dem IN-FRAME-RESPONSE SPACE die angesprochenen Slaves bzw. der Master selbst die Botschaftsantwort (Message Response) absetzen. Diese beginnt mit 2, 4 oder maximal 8 Datenfeldern (bis LIN 1.2).

Die Anzahl der Datenfelder und damit die Länge der Botschaft sind im Identifierfeld durch die Bits ID4 und ID5 zu bestimmen und fest mit dem Identifier verbunden. Jedes Datenfeld besteht aus einem Start-Bit, dem Datenbyte und einem Stopp-Bit. Die Start- und Stopp-Bits ermöglichen den Slaves eine Nachsynchronisation.



**Datenfeld**

**Aufgabe 1:**

Tragen Sie die Nutzdaten 4Fh in das unten vorgegebene Datenfeld ein.

**Umwandlung der Hexadezimalzahl in eine Dualzahl:**

4h = 

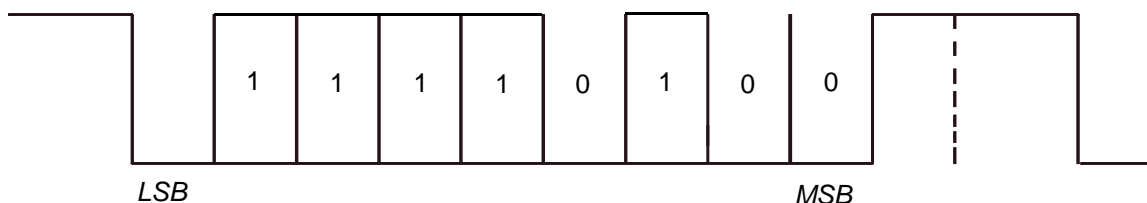
0	1	0	0
---	---	---	---

Fh = 

1	1	1	1
---	---	---	---

4Fh = 

=	0	1	0	0	1	1	1	1
---	---	---	---	---	---	---	---	---



**Aufgabe 2:**

Beim Übertragen der Nutzdaten in das Datenfeld muss etwas beachtet werden. Was ist es? Und wie wird in den Abbildungen darauf hingewiesen? Tragen Sie den Hinweis in das Datenfeld ein.

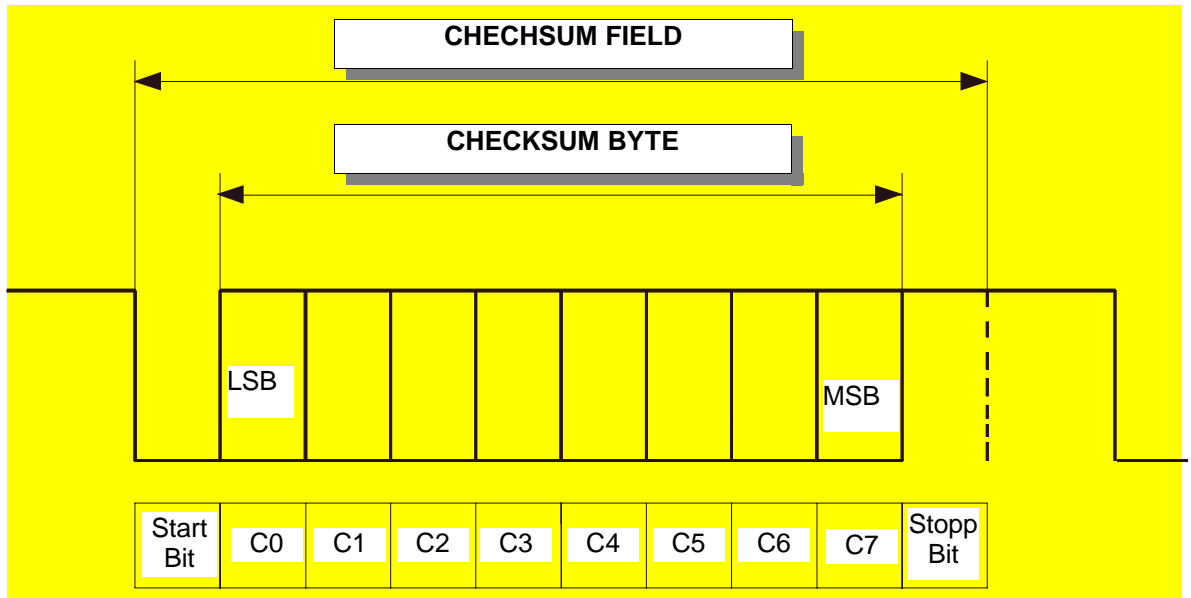
*Da beim LIN die niederwertigen Bits zuerst gesendet werden, ist das Bitmuster der Nutzdaten in entgegengesetzter Reihenfolge einzutragen. Dies wird durch das Eintragen der Abkürzungen LSB und MSB in den Abbildungen deutlich gemacht.*

*LSB steht für: least significant bit (niederwertiges Bit)*

*MSB steht für: most significant bit (höherwertiges Bit)*

### 5.3.3.7 Checksummen Feld (CHECKSUM FIELD)

Zur Erkennung von Übertragungsfehlern im Bereich der Datenfelder wird an diese das CHECKSUM FIELD angehängt und die Botschaft beendet.  
Das CHECKSUM FIELD wird immer von dem Busteilnehmer (Master oder SLAVE) berechnet und gesendet, der auch die Datenfelder auf den Bus gelegt hat.



CHECKSUM FIELD

### 5.3.3.8 INTERBYTE SPACE

Als INTERBYTE SPACE bezeichnet man die Pausen zwischen den Datenfeldern. In der LIN\_Spezifikation ist ihre Dauer nicht festgelegt, aber durch die festgesetzte maximale Botschaftslänge begrenzt. Die kürzeste Zeit ist 0 Sekunden.

### 5.3.3.9 IN-FRAME RESPONSE SPACE

Die Pause zwischen dem Header und Response wird als IN-FRAME RESPONSE SPACE bezeichnet.

Als minimale Länge ist wie beim INTERBYTE SPACE ein Wert von 0 Sekunden möglich. Die maximale mögliche Länge wird ebenfalls durch die maximale Botschaftslänge begrenzt.

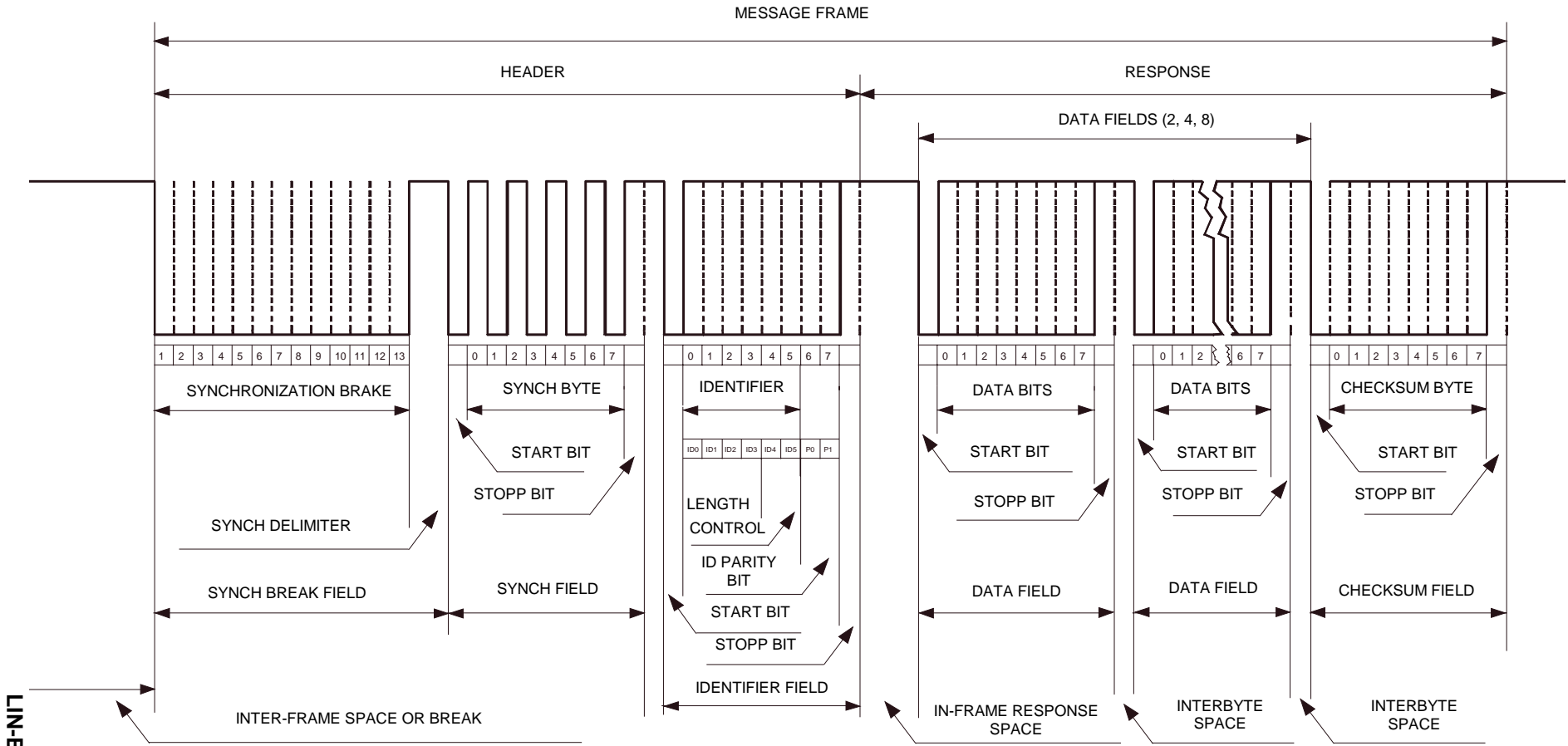
### 5.3.3.10 MESSAGE FRAME

Auf der folgenden Seite ist eine komplette LIN-Botschaft (MESSAGE FRAME) dargestellt. Ergänzen Sie die fehlenden Bezeichnungen.

Übertragen Sie die Bezeichnungen auf das Oszillogramm der realen LIN-Botschaft. Tragen Sie danach die Bitbezeichnungen sowie die Bitzustände ein. Ermitteln Sie den ID der Botschaft und die übertragenen Datenpakete. Kontrollieren Sie die Paritätsbits.

Erläutern Sie das Zustandekommen der unterschiedlichen Spannungspegel.

*Die dominanten Pegel können durch unterschiedliche Ausführungen der Transceiver entstehen. Dies hat keinen Einfluss auf die Datenübertragung.*



LIN-Botschaft

LIN-Botschaft

