Protokol
for seriel kommunikation
Mellem ATC, TC, MSR3 og Havarilog

	Sider:	Udgave:	Dato:
Dansk version	66	01.02	08.07.2002
Engelsk version	66	01.02	08.07.2002

banestyrelsen

		Verificeret		Adresse Banestyrelsen Pakhusvej 10 2100 København Ø	Projektering
		Afløser 01.01			
		Godkendt af Banestyrelsen Allan Orfield			
Tegning:	1. udgave Dato og initialer	Seneste udgave Dato og initialer	Mål	Tegningsnavn Protokol for seriel kommunikation Mellem ATC, TC, MSR3 og Havarilog	
Konstrueret	04.02.2002 jhm	08.07.2003 jhm	Enhed		
Kontrolleret	04.02.2002 hml	26.08.2003 HML			
Godkendt	04.02.2002 nfn	26.08.2003			
	Sprog /da	Udgave 01.02 08.07.2003		Tegningsnr. IN 656 V 1711	Side/af sider 1/133

Protokol
for seriel kommunikation
Mellem ATC, TC, MSR3 og Havarilog

banestyrelsen

		Verificeret		Adresse Banestyrelsen Pakhusvej 10 2100 København Ø	Projektering
		Afløser 01.01			
		Godkendt af Banestyrelsen			
Tegning:	1. udgave Dato og initialer	Seneste udgave Dato og initialer	Mål	Tegningsnavn Protokol for seriel kommunikation Mellem ATC, TC, MSR3 og Havarilog	
Konstrueret	04.02.2002 jhm	08.07.2003 jhm	Enhed		
Kontrolleret	04.02.2002 hml	01.08.2003 hml			
Godkendt	04.02.2002 nfn	01.08.2003 MDJ			
	Sprog /da	Udgave 01.02 08.07.2003		Tegningsnr. IN 656 V 1711	Side/af sider 1/66

Protokol
for seriel kommunikation
Mellem ATC, TC, MSR3 og Havarilog

banestyrelsen

		Verificeret		Adresse Banestyrelsen Pakhusvej 10 2100 København Ø	Projektering
		Afløser 01.01			
		Godkendt af Banestyrelsen			
Tegning:	1. udgave Dato og initialer	Seneste udgave Dato og initialer	Mål	Tegningsnavn Protokol for seriel kommunikation Mellem ATC, TC, MSR3 og Havarilog	
Konstrueret	04.02.2002 jhm	08.07.2003 jhm	Enhed		
Kontrolleret	04.02.2002 hml	01.08.2003 hml			
Godkendt	04.02.2002 nfn	01.08.2003			
	Sprog /da	Udgave **01.02 08.07.2003**		Tegningsnr. IN 656 V 1711	Side/af sider 1/66

Indholdsfortegnelse:

Ændringslog:

Version	Ændring
01.00	Overført fra Rådgivning(dokumentversion 3 nov. 1998) til Banestyrelsen. Tilføjet manglende formatspecifikation for Telegramtype o(lille o) i Bilag 8.1.
01.01	Afs. 7: poll-tid(T0) ændret. Gælder fra og med ATC SW02. Afs. 8 revideret: Fejltelegrammer og førerbordsnøgletelegrammer Afs. 10: TC ikke aktiv overfor ATC.
01.02	DSB (TRIT) rettelser MSR3-TC

1.0 TERMER OG FORKORTELSER

Dette kapitel indeholder en ordforklaring på termer og forkortelser. I hele protokollen skrives de anvendte termer med stort begyndelsesbogstav for at henvise til de definerede termer. Forkortelser skrives enten med stort begyndelsesbogstav eller med versaler.

Gennem hele specifikationen er der brugt diverse termer og forkortelser. De beskrives her.

ATC:	Automatic Train Control (det mobile ATC-anlæg).
Bytecount:	Antallet af bytes, undtagen Startpad og Endpad, i et Telegram. Altså antallet af bytes i et Telegram minus 2. Er 2 ASCII tegn, der angiver et tal fra 1 til 64. 2 bytes.
Checksum:	2's complement af modulo 100 Hex. Beregnet af Bytecount + Telegramtype + Datapackets og udskrevet som ASCII. Er altid et ASCII tal fra 0 til 9 eller et ASCII bogstav fra A til F (30H - 39H og 41H - 46H). Checksummen udregnes i Hex og omskrives til ASCII sådan som eksemplet i "Datapackets" viser. 2 bytes, jf. beskrivelse.
Datafield:	Det område hvori Packettypes og Datapackets befinder sig. Startes efter Telegramløbenr., og afsluttes ved Checksum.
Datapacket:	De bytes, der udgør een type togdata. Jf. skemaet "Det begrænsede ASCII karaktersæt til Datapackets" afsnit 5.3.1. De hexadecimale koder, der skal transmitteres omskrives til ASCII, er sådan, at f.eks. 8BH omskrives til 8 og B (38H og 42H). Dette for at gøre det nemmere at dechifrere kommunikationen, ved en eventuel printerovervågning. Variabelt antal bytes.
DS 2089:	Forkortelse af Dansk Standard nr. 2089, ISO 7 Bitkode, tegnsæt.
Endpad:	Her slutter telegrammet. Er altid et ASCII "CR" (0DH). 1 byte.
Godkendt:	Et godkendt Telegram er et Telegram, hvor Startpad, Bytecount, Checksum, Endpad og paritetscheck er rigtige ved modtagelsen.
Hit:	Er en betegnelse for en forstyrrelse af det elektriske niveau i et telegram, således at de i afsnit 10.2 fastlagte strømniveauer overskrides. De tre stjerner i figurer skal visualisere linieforstyrrelse.
HLOG:	Havarilog (registreringsudstyr, der registrerer hastighedsdata mv. under et køretøjs kørsel).

Liniefejl:	Er enhver anden tilstand end de i denne specifikation godkendte Master- og Slave- tilstande. Master og Slave kan begge internt registrere Liniefejl.
Linieovervågning:	Er den funktion, der ved hjælp af en tidsvagts udløb registrerer Liniefejl.
Linietilstand:	Er en tilstand, som den serielle linie befinder sig i afhængig af tilstanden i Master og i Slave. Linietilstanden afspejler direkte, hvad der skal sendes på linien.
Master:	Er en titel, som tildeles den ene enhed på en seriel linie. Denne enhed har det suveræne herredømme over al kommunikation på linien.
MSR3:	Mobil StrækningsRadio 3. generation.
Packettype:	Angiver hvilke data, der er tale om. Er upper- og lower-case. ASCII bogstaver, (41H -5AH og 61H - 7AH) og ASCII %, (25H). 1 byte.
Retrans:	Angiver en retransmission af et Telegram, som har været sendt før.
Slave:	Er en titel, som tildeles den anden enhed på en seriel linie. Denne enhed er totalt underkastet herredømmet udøvet af Master. Derfor må Slave ikke sende noget på linien uden at Master giver lov.
Startpad:	Er første data i et Telegram. Starter telegrammodtagelse. Er altid et ASCII "LF" (OAH). 1 byte.
TC:	TogComputer.
Telegram:	Den komplette datastreng.
Telegramløbenr.:	Angiver nummeret på et ubesvaret Telegram, set fra Master. Er altid ASCII 1 eller 0. 1 byte.
Telegramtype:	Er en kategorisering af Telegrammer, der opfører sig ensartet på den serielle linie. Hver Telegramtype repræsenterer en bestemt funktion.
Trans:	Betyder opstart af en transmissionssekvens, som kan indeholde retransmissioner. Trans er altid et nyt Telegram.
Vekslingsprocedurer:	Er et regelsæt for, hvorledes Telegramtyper kan udveksles på den serielle linie.

2.0 GENEREL BESKRIVELSE

Dette kapitel indeholder en beskrivelse af transmissionssystemet. Beskrivelsen er en opsummering af, hvorledes systemet skal forstås eller fortolkes.

Afvigende specifikationer vedrørende S-tog er anført i afs. 12. HKT-appendix.

2.1 Princip for Master- og Slavekonceptet

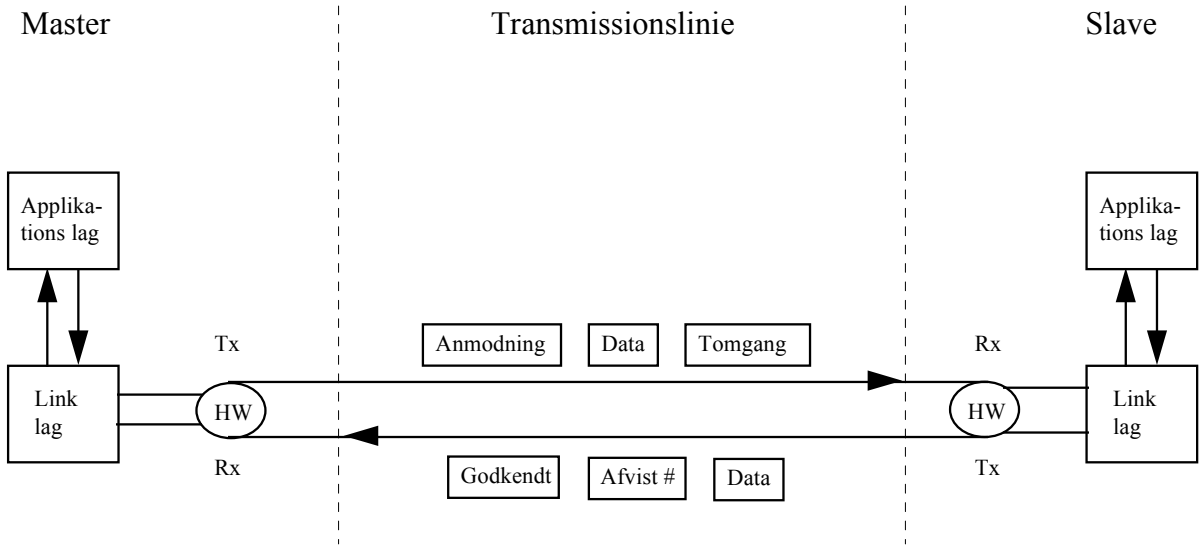


Fig. 2.1 Principtegning af Master- og Slavekonceptet.

2.1.1 Master karakteristika

Master styrer al kommunikation.

Tomgangstelegrammet har laveste prioritet.

Styringen er opdelt i:
A. Anmodning om data.
B. Afsendelse af data.
C. Anfordring om uopfordret data.

Master må ikke starte ny informationsoverførsel, før den forrige er afsluttet.
Afsendelse af nyt Telegram sker tidligst efter, at Slave har svaret.
Master har recieve åben, når og kun når T2 løber.

2.1.2 Slave karakteristika

Slave må kun afsende et Telegram på anfordring fra Master.

Modtagelse af anmodningstelegram eller tomgangstelegram skal af Slave betragtes som anfordring.
Der må kun svares med et svartelegram på en anfordring.

2.2 Logisk opbygning af et transmissionssystem

Transmissionssystemet er opbygget af flg. procedurer:

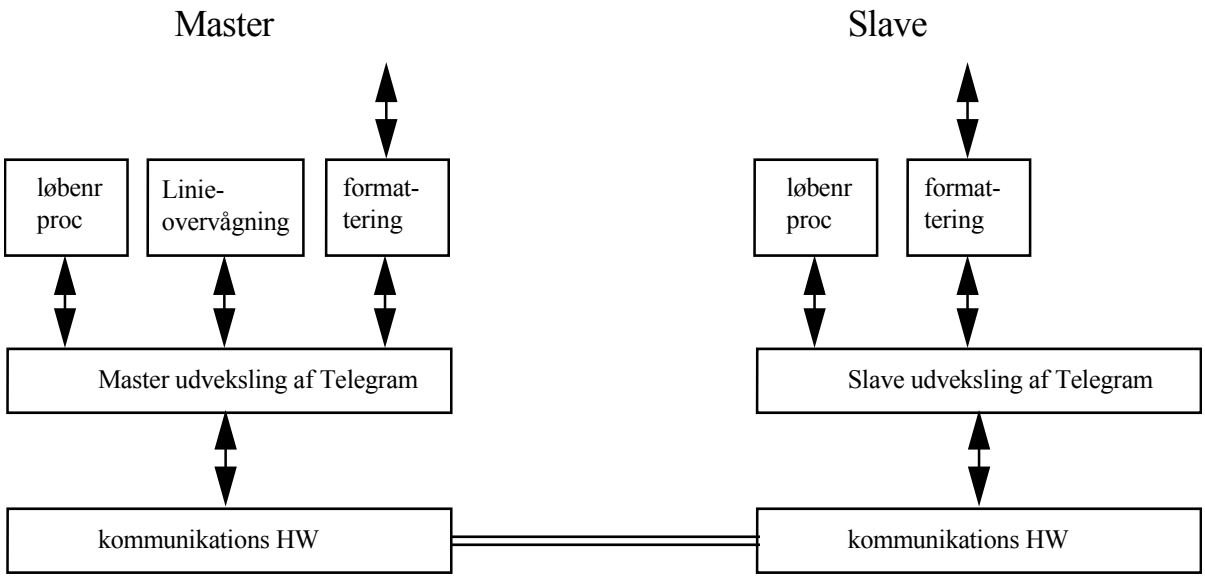


Fig. 2.2 Opbygning af transmissionssystem.

Fig. 2.2 viser grafisk, de procedurer transmissionssystemet er opbygget af:
Nederste lag, kommunikations HW, består af hardware-komponenter f.eks. UART(universal asynchronous receiver transmitter) og eksekverer kontrol og transmission af den enkelte karakter og sikrer at den overholder de karakteristiske data i den enkelte karakter jf. afsnit 10.2.
Mellemste lag er softwareprogrammel, der sikrer transmission af det enkelte telegram.
Øverste lag er programmel, der overvåger at telegrammet har korrekt løbenummer og vha. timerfunktioner 1,2 og 3 (der kan være SW eller HW fremstillede) overvåger at transmissionen forløber korrekt.
Fra øverste lag er forbindelsen ind til udstyrets applikationslag.

3.0 TRANSMISSIONSDEFINITIONER

Dette kapitel indeholder definitioner på de forskellige elementer, som fastlægges for netop denne protokol. Disse er opdelt i:

- Definition af Telegramtyper.
- Definition af timere.
- Definition af tilstande.
- Definition af fejlmuligheder.

3.1 Definition af Telegramtyper

Imellem enhederne på en seriel linie er der defineret et antal telegramnumre. Disse er kategoriseret i 5 Telegramtyper, som hver især behandles på en ensartet måde i udvekslingsprocedurerne, uanset indholdet eller funktionen af telegramnummeret.

De fem Telegramtyper er:
1. Data.
2. Anmodning.
3. Tomgang.
4. Godkendt.
5. Afvist.

3.1.1 Telegramtype *Data*

Funktion: Anvendes til at sende information imellem enhederne på den serielle linie. *Data* kan sendes uopfordret eller som svar på *Anmodning* eller *Tomgang*.

Afsender: Master.

Initiator: Uopfordret.

Svar: *Godkendt, Afvist*(#)

Afsender: Slave.

Initiator: *Anmodning, Tomgang*.

Svar: -

Kommentar: Denne Telegramtype kan sendes både fra Master og fra Slave.

Fra Master sendes *Data* uopfordret.
Fra Slave sendes *Data* aldrig uopfordret, men altid som svar på en *Anmodning* eller *Tomgang* sendt fra Master.

Denne forskel mellem *Data* fra Master og *Data* fra Slave betyder at *Data* fra Master altid er prioriteret højere end *Data* fra Slave.

3.1.2 Telegramtype *Anmodning*

Funktion: Anvendes som forespørgsel efter specifik information, dvs. et på forhånd kendt telegramnummer. Der kan kun forespørges efter ét telegramnummer ad gangen.

Afsender: Master.

Initiator: Uopfordret.

Svar: *Data*, *Afvist* (#).

Kommentar: *Anmodning* er kun defineret til at afsendes fra Master. Som svar kan sendes den ønskede Telegramtype i form af *Data*. Der kan i stedet sendes *Afvist* med en fejlkode, der indikerer grunden til at *Anmodning* er blevet afvist.

3.1.3 Telegramtype *Tomgang*

Funktion: Anvendes af Master til at afgive transmissionsretten til Slave, således at denne får mulighed for at sende.

Afsender: Master.

Initiator: Uopfordret.

Svar: *Data*, *Godkendt*, *Afvist* (#).

Kommentar: Der svares *Data* fra Slave, hvis Slave har uopfordret *Data*, som skal sendes til Master. Har Slave ingenting til Master svares der med *Godkendt* Telegram. I fejlsituationer svares der med *Afvist* og med en fejlkode, der indikerer, hvorfor det modtagne *Tomgang* Telegram er blevet afvist.

3.1.4 Telegramtype *Godkendt*

Funktion: Anvendes af Slave som svar på et Telegram, fejlfrit modtaget fra Master og behandlet korrekt af Slave.

Afsender: Slave.

Initiator: *Data*, *Tomgang*.

Svar: -

Kommentar: Telegramtypen *Godkendt* sendes altid kun fra Slave til Master, som herved får en kvittering for et afsendt Telegram. *Godkendt* afslutter en telegramudveksling.

3.1.5 Telegramtype *Afvist*

Funktion: Anvendes af Slave som svar på et Telegram, modtaget med fejl fra Master. Der medsendes en fejlkode, der indikerer, hvorfor det modtagne Telegram er blevet afvist.
Der kan enten være tale om en transmissionsfejl eller en fejl i behandlingen af telegramindholdet i Slaveenheden.

Afsender: Slave.

Initiator: *Data*, *Anmodning*, *Tomgang*.

Svar: -

Kommentar: Telegramtypen *Afvist* sendes altid kun fra Slave til Master, som herved får en kvittering for et afsendt Telegram.

Modtages et *Afvist* Telegram i Master kan denne vælge at retransmittere eller på anden måde behandle den opståede fejl.
Afvist afslutter således ikke en telegramsekvens.

3.2 Definition af timere

Der er defineret tre timere/tidsvagter i systemet. Disse bruges til overvågning af diverse parametre. Timerne er:

3.2.1 Timer nr. 0

Definition af timerfunktioner i Master og Slave

Timertype: Timer nr. 0 (forkortes T0).

Funktion: Linieovervågningstimer i Master, som bruges til at sikre, at der sker transmission på linien.

Specifikation:
1. T0 er en tidsvagt i Master SW eller HW.
2. T0 har en udløbstid på 20 sek. ± 1 sek.
3. T0 starter, når den kommende Vcc til Master og selvtestprogram er udført eller efter manuel reset.
4. T0 restarter, når Master har afsendt Endpad.
5. Ved udløb af T0 skal Master afsende et Telegram af type: *Tomgang* til Slave.

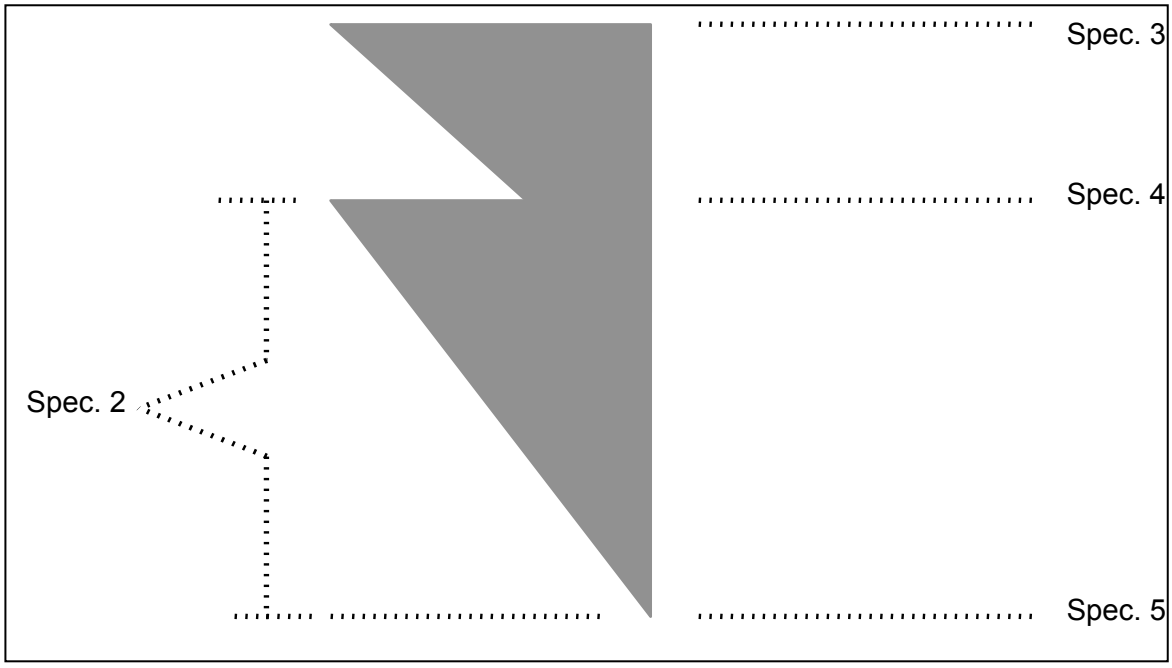


Fig. 3.1 Timerdiagram for T0.

3.2.2 Timer nr. 1

Definition af timerfunktioner i Master og Slave

Timertype: Timer nr. 1 (forkortes T1).

Funktion: Linieovervågningstimer i Slave, som bruges til at kontrollere, at der er transmission på linien.

Specifikation:

1. T1 har en udløbstid på 28 sek. ± 1 sek.
2. T1 er en tidsvagt i Slave SW eller HW.
3. T1 starter ved korrekt modtaget LF, CR, Checksum og ingen transmissionsfejl. Slave skal registrere, at kommunikationen til Master er korrekt.
4. T1 restarter ved korrekt modtaget LF, CR, Checksum og ingen transmissionsfejl.
5. Ved udløb af T1 skal Slave internt registrere, at kommunikation fra Master er ophørt, samt vente på specifikation nr. 3.

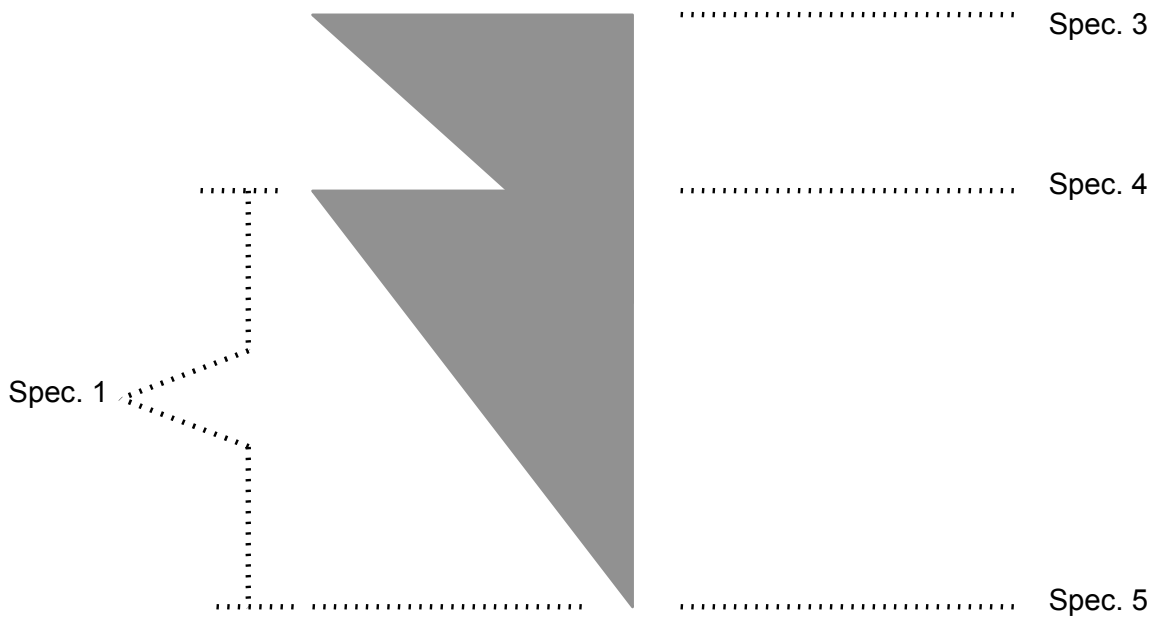


Fig. 3.2 Timerdiagram for T1.

3.2.3 Timer nr. 2

Definition af timerfunktioner i Master og Slave

Timertype: Timer nr. 2 (forkortes T2).

Funktion: Retransmissionstimer i Master.

Specifikation:

1. T2 er en tidsvagt i Master SW eller HW.

2. T2 har en udløbstid på 2 sek. ± 20 ms.
3. T2 starter, når CR er afsendt i Telegram fra Master.
4. T2 stopper og nulstilles ved modtagelse af LF, CR, Checksum og ingen transmissionsfejl.
5. T2 kan ikke retrigges, men kun stoppe og dermed nulstilles.
6. Ved udløb af T2 skal Master registrere, om det er udløb nr. 1, 2 eller 3 i et telegramoverførsel.
7. Ved T2 udløb nr. 1 eller nr. 2 skal Master registrere og udføre retransmission af Telegram.
8. Ved T2 udløb nr. 3 skal Master registrere, at kommunikationen fra Slaven er ophørt.

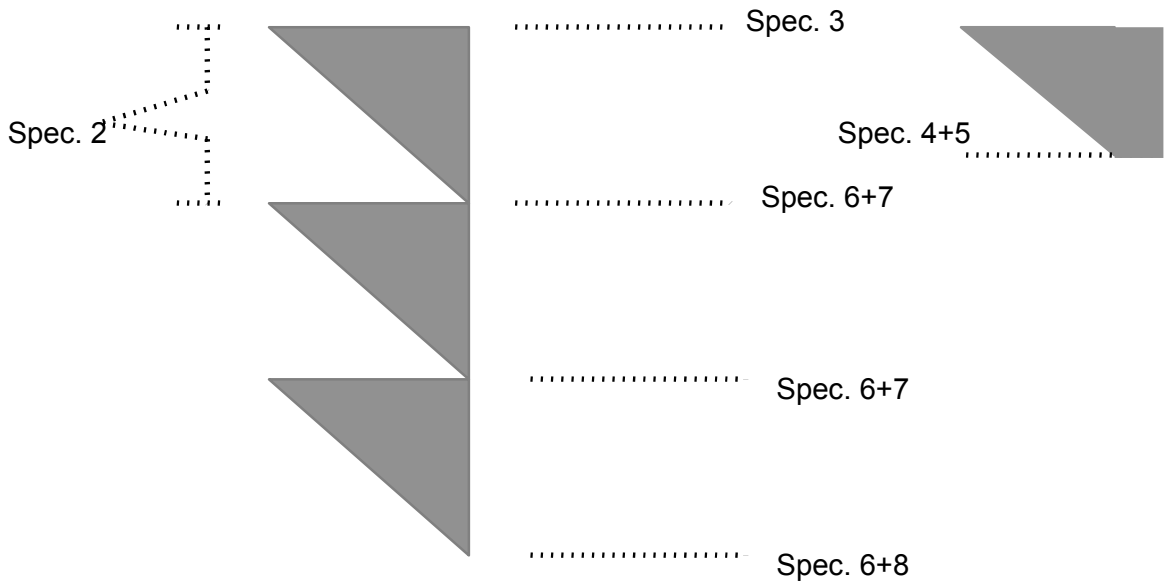


Fig. 3.3 Timerdiagram for T2.

3.3 Definition af tilstande

Transmissionssystemet består af:

1. Master
2. Slave
3. Linie

Disse tre systemelementer kan alle være i selvstændige tilstande, afhængig af udefra kommende påvirkninger. I det følgende er beskrevet hvilke tilstande, der er defineret for de enkelte elementer, og hvorledes de påvirkes.

3.3.1 Mastertilstande

Der er mulighed for fire tilstande på mastersiden. Disse tilstande er:

1. M-Data: Master har data til Slaven.
2. M-Anmodning: Master ønsker at få data fra Slaven.
3. M-Tomgang: Master har hverken data til Slaven eller ønsker at få data fra Slaven.

4. M-Fejl: Master har detekteret Liniefejl, dvs. ingen svar på en Trans +2 Retrans.

3.3.2 Slavetilstande

Der er mulighed for fire tilstande på slavesiden. Disse tilstande er:

1. S-Data: Slaven har data til Master.
2. S-Anmodning: Slaven ønsker at få data fra Master.
3. S-Tomgang: Slaven har hverken data til Master eller ønsker at få data fra Master.
4. S-Fejl: Slaven har detekteret Liniefejl, dvs. at linietidsvagten T1 er udløbet.

Note: S-Anmodning er ikke relevant i den foreliggende specifikation, men er en mulig tilstand, hvis dette måtte ønskes.

3.3.3 Linietilstande

Når Master- Slavetilstande kombineres, giver det mulighed for de linietilstande, som fremgår af følgende tabel:

Master / Slave	M-Data	M-Anmodning	M-Tomgang	M-Fejl
S-Data	L-Master	L-Master	L-Slave	L-Fejl
S-Anmodning	L-Master	L-Master	L-Slave	L-Fejl
S-Tomgang	L-Master	L-Master	L-Tomgang	L-Fejl
S-Fejl	-	-	-	L-Fejl

Fig. 3.4 Mulige linietilstande.

Jf. ovenstående figur er linietilstanden en funktion af Mastertilstanden og Slavetilstanden. Denne sammenhæng giver følgende linietilstande:

1. L-Master: I denne linietilstand sender Master på linien enten M-Anmodning eller M-Data Telegrammer. Slave kan ikke sende S-Data eller S-Anmodning. Dette betyder, at Master har højest prioritet på linien.
2. L-Slave: I denne linietilstand er Master i tomgang, og Slave kan nu få mulighed for at sende.
3. L-Tomgang: Hverken Slave eller Master har information at sende, hvilket betyder, at linien er i tomgang.
4. L-Fejl: Der er opstået en Liniefejl i Master og eventuelt i Slave, hvilket betyder at linien er fejlmeldt.

3.4 Definition af fejlmuligheder

For at muliggøre dataoverførsel (datatransmission) i den fejlsituation, hvor Masters TxD ikke er i forbindelse med Slaves RxD, skal Master fortsætte datatransmissionen også efter Liniefejlen er detekteret.

Fortsættelsen af datatransmission må kun medføre, at Slaven afviser højst ét korrekt modtaget Telegram. Jf. nedenstående figur 3.5.

1. T1 ikke udløbet.

2. Linien igen intakt efter præcis tre trans.

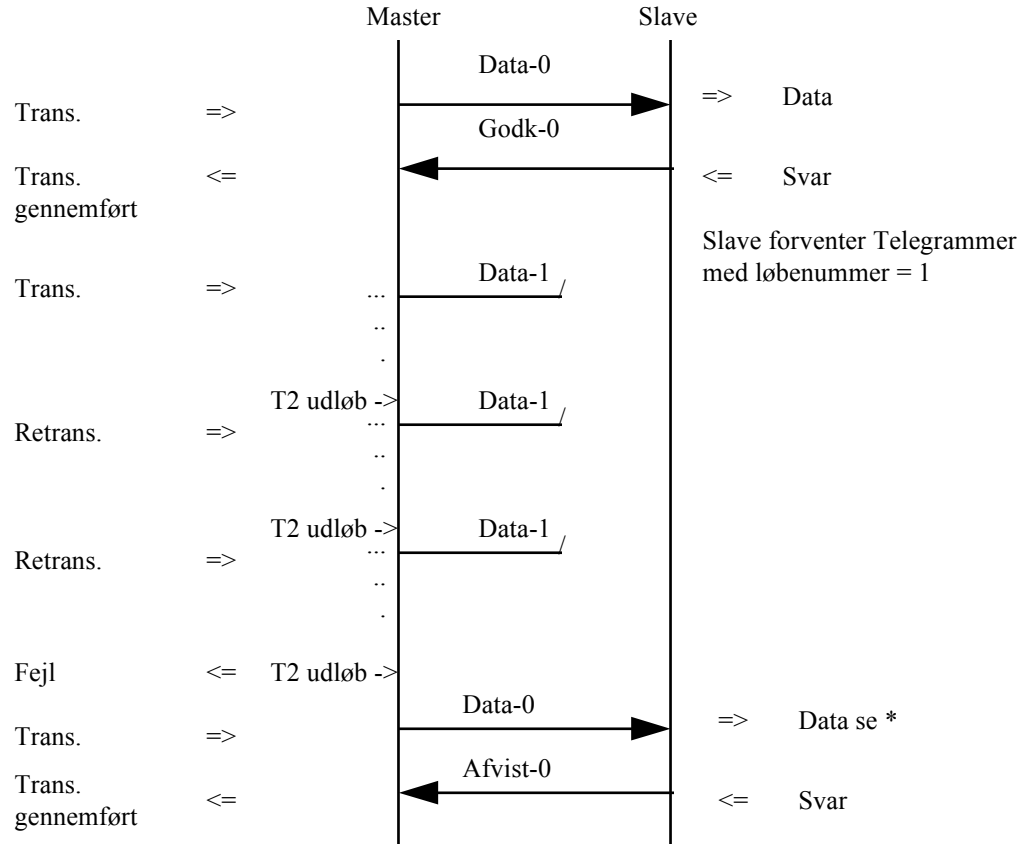


Fig. 3.5 Afvisning af korrekt modtaget Telegram

* Det skønnes at denne fejlsituation, hvor ovenstående to punkter er opfyldt, er tilstrækkelig usandsynlig. Under alle omstændigheder kan Slaven kun ignorere ét korrekt modtaget Telegram.

4.0 TRANSMISSIONSPROCEDURER

Dette kapitel indeholder en beskrivelse af procedurerne, der er specificeret i transmissionssystemet. Procedurerne indeholder en anvendelse af de "elementer", der er defineret i kapitel 3.
I transmissionssystemet er følgende procedurer defineret:

- Udveksling af Telegramtyper.
- Løbenummerering.
- Linieovervågning.

4.1 Udveksling af Telegramtyper

Denne procedure er central i transmissionssystemet, da den viser hvorledes de enkelte Telegramtyper bliver behandlet i forskellige situationer. Proceduren, der her er beskrevet, håndterer alle "normal" situationer, dvs. situationer hvor linien befinder sig i en af følgende tilstande:

- L-Master

- L-Slave
- L-Tomgang

Ovennævnte linietilstande er "normal" tilstande, hvor linien ikke er i fejl, dvs. er fejlmeldt af Master alene eller af både Master og Slave.

Der er anvendt følgende nomenklatur:

Telegram-0: Betyder at et Telegram sendes med løbenummer 0.

T2
.
.
.
* : Betyder at timer T2 løber. Ved stjerne løber timeren ud.

T2
.
.
.
- : Betyder at timer T2 er stoppet før udløb.

T2
.
.
.
-
T2: Genstart af timer T2 før timerudløb.
.
.
.
.

T2
.
.
.
*
T2: Genstart af timer T2 efter timerudløb.
.
.
.
.

***
hit Transmissionsfejl.

4.1.1 Linietilstand L-Master

I dette afsnit behandles linietilstanden L-Master, hvori Master enten kan være i M-Data eller i M-Anmodning tilstanden. I figurerne behandles kun for M-Data tilstanden, altså hvor Master sender Data til Slave. M-Anmodning kan behandles på samme vis.

4.1.1.1 Normal tilfælde

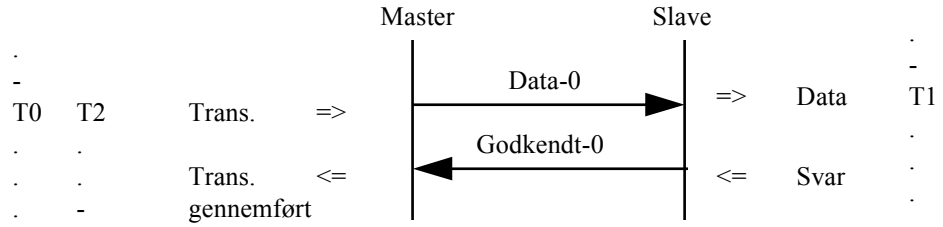


Fig. 4.1 L-Master normal fra Master

4.1.1.2 Transmissionsfejl fra Master

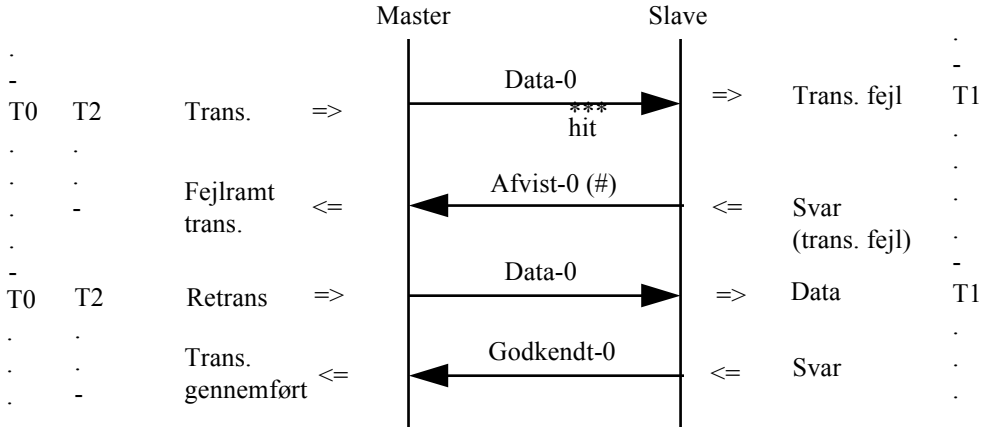


Fig. 4.2 L-Master transmissionsfejl fra Master

4.1.1.3 Transmissionsfejl fra Slave

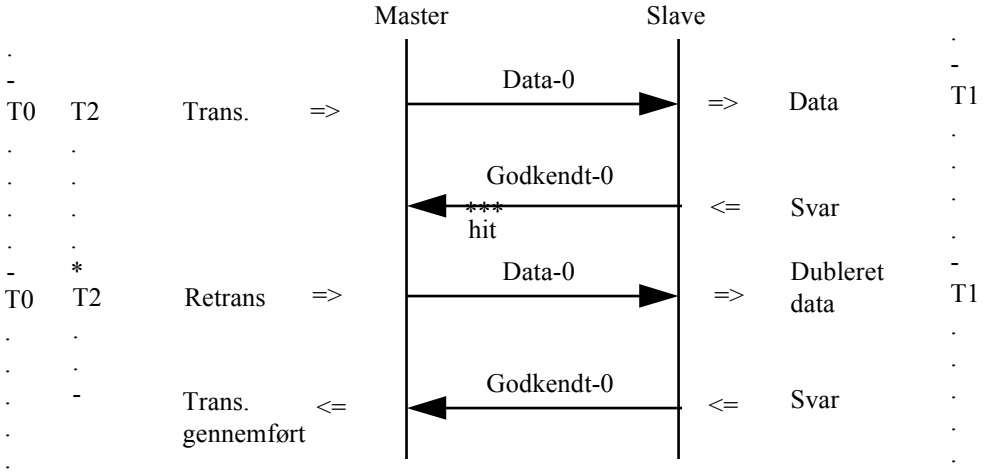


Fig. 4.3 L-Master transmissionsfejl fra Slave

4.1.1.4 Transmission afbrudt til Slave

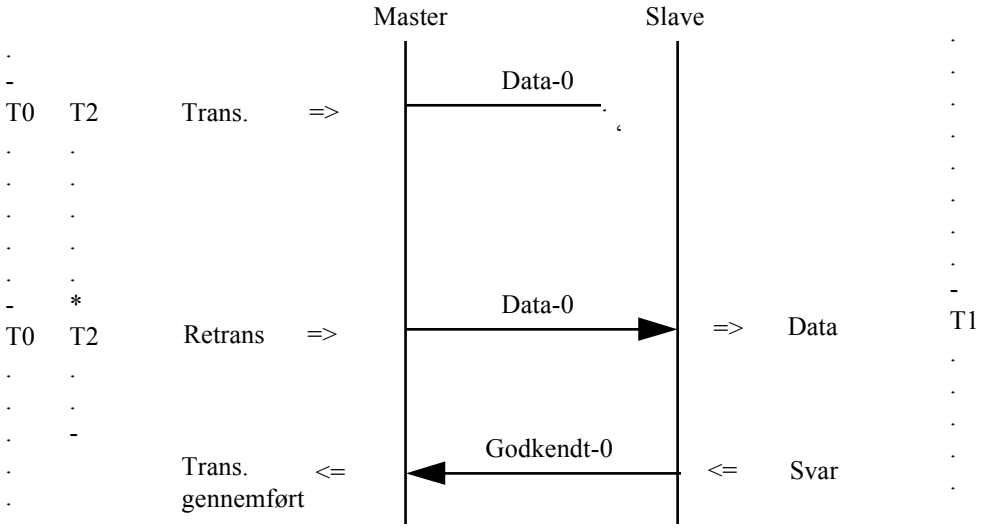


Fig. 4.4 L-Master transmission afbrudt til Slave

4.1.1.5 Transmission afbrudt til Master

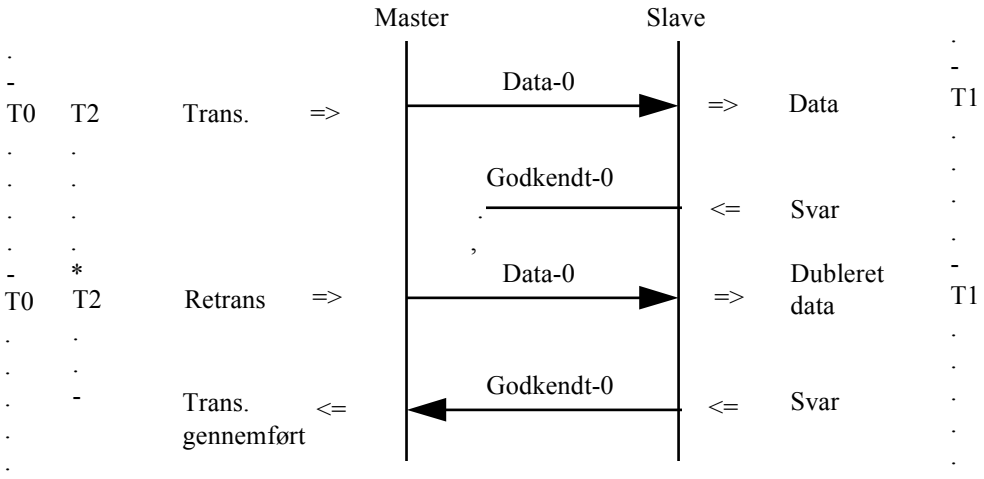


Fig. 4.5 L-Master transmission afbrudt til Master

4.1.1.6 Ingen transmission til Slave

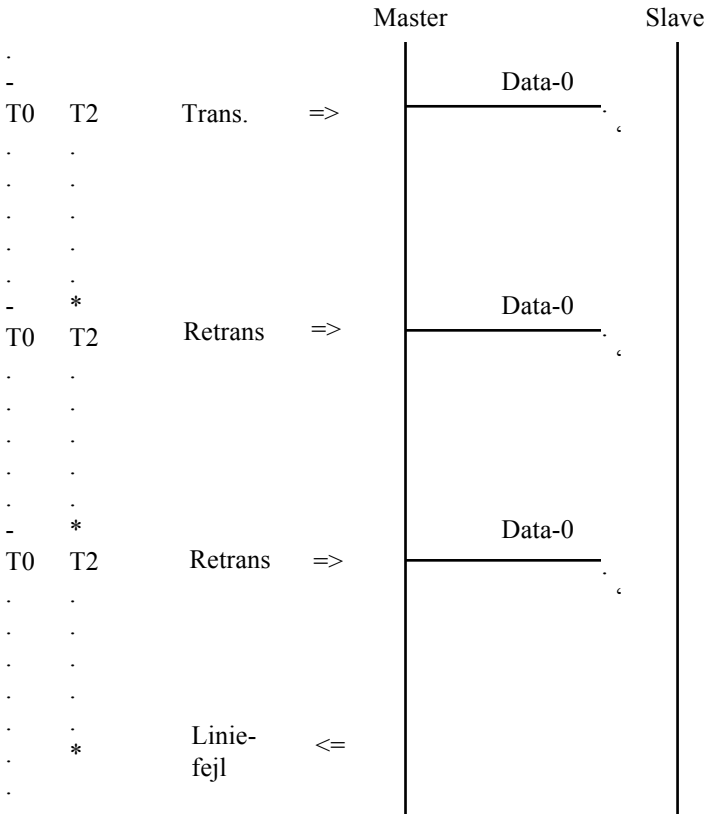


Fig. 4.6 L-Master ingen transmission til Slave

4.1.1.7 Ingen transmission til Master

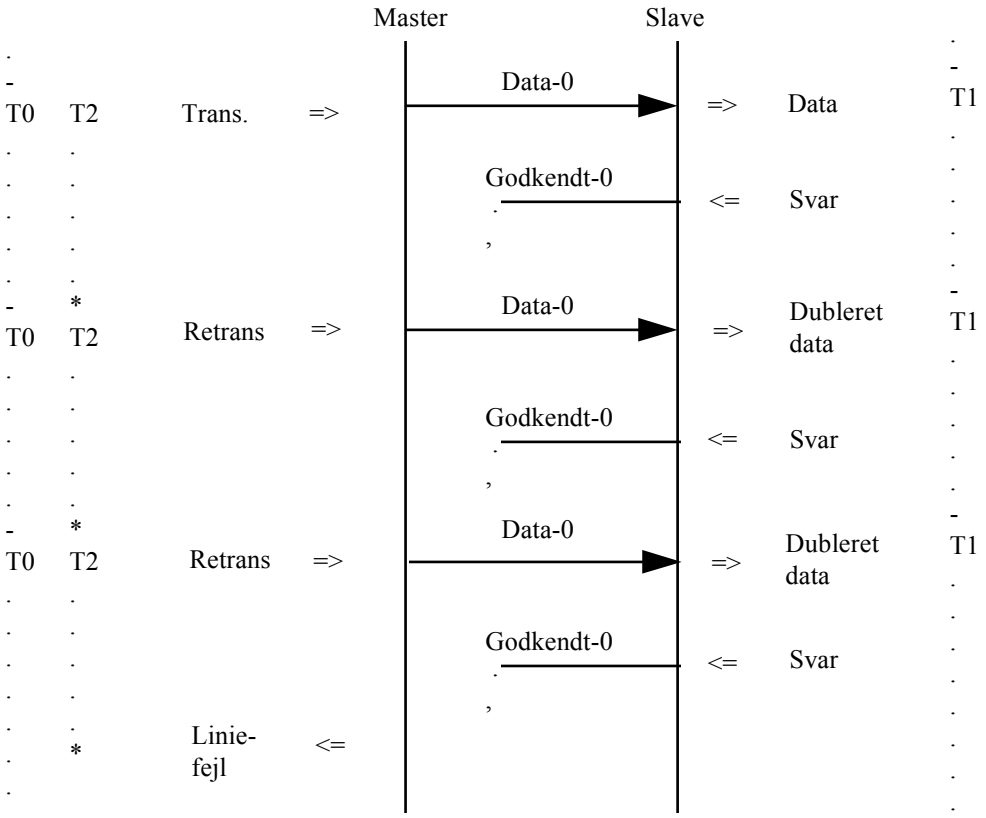


Fig. 4.7 L-Master ingen transmission til Master

4.1.2 Linietilstand L-Slave

I dette afsnit behandles linietilstanden L-Slave, hvori Master er i M-Tomgang tilstanden og Slave er i S-Data. Slave tilstanden S-Anmodning hører også til her, men anvendes ikke i denne protokol.

4.1.2.1 Normal tilfælde

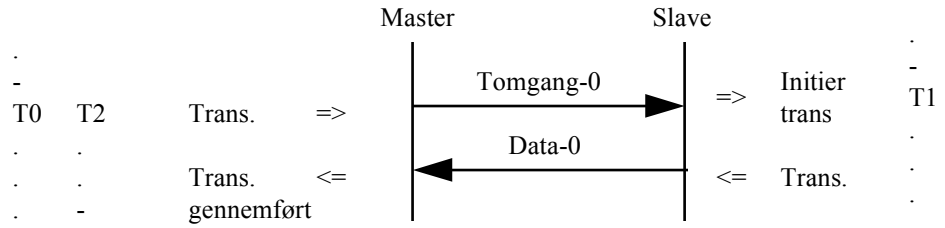


Fig. 4.8 L-Slave normal procedure

4.1.2.2 Transmissionsfejl fra Master

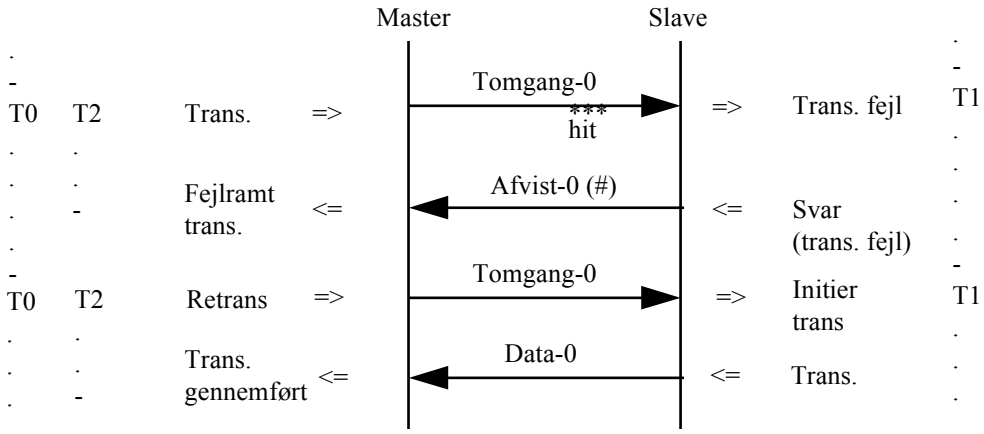


Fig. 4.9 L-Slave transmissionsfejl fra Master

4.1.2.3 Transmissionsfejl fra Slave

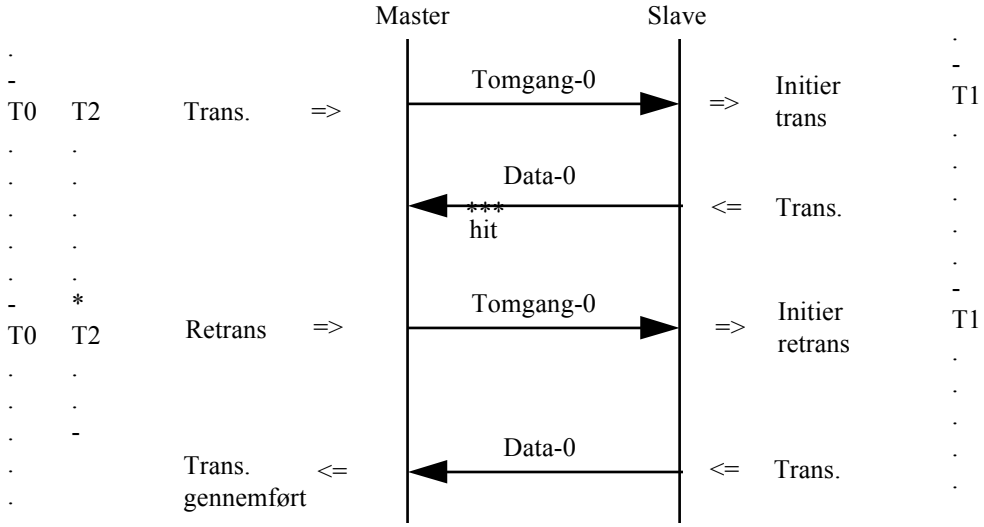


Fig. 4.10 L-Slave transmissionsfejl fra Slave

4.1.2.4 Transmission afbrudt til Slave

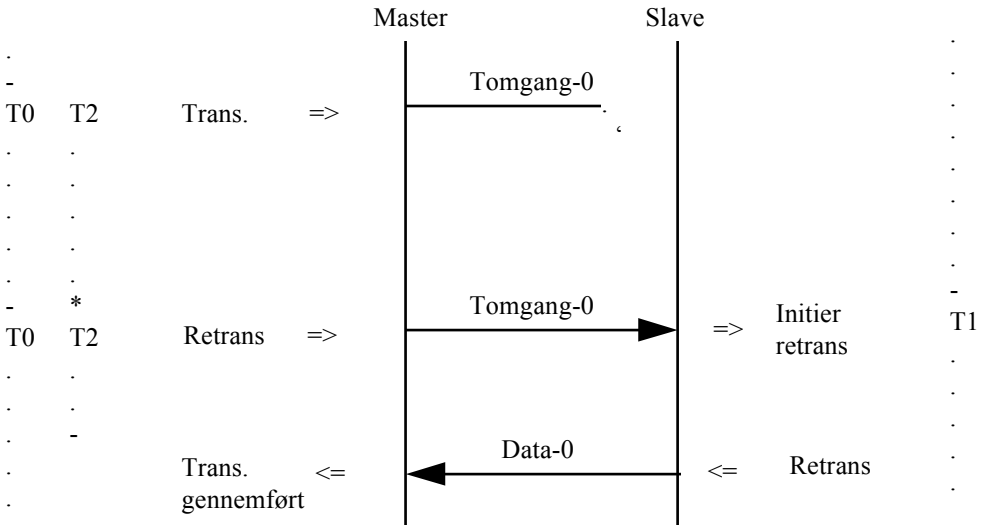


Fig. 4.11 L-Slave transmission afbrudt til Slave

4.1.2.5 Transmission afbrudt til Master

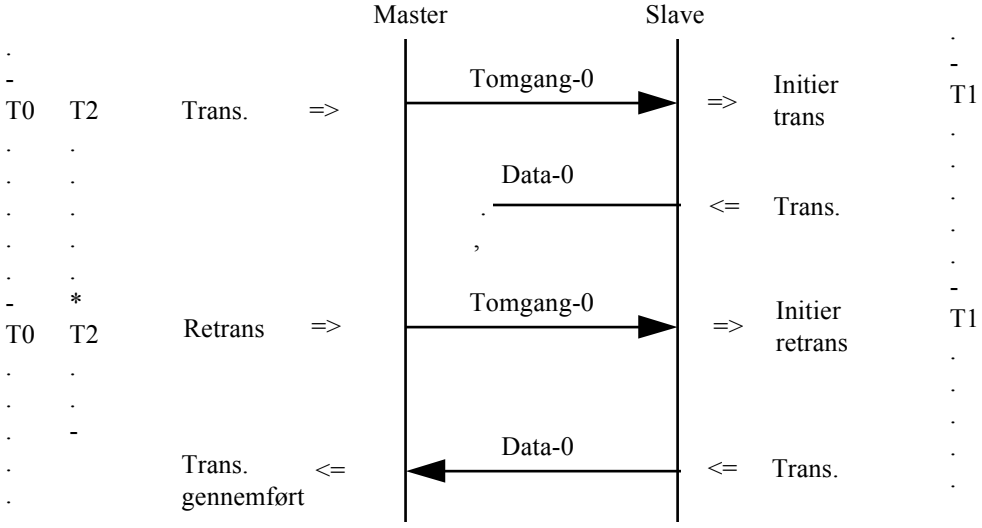


Fig. 4.12 L-Slave transmission afbrudt til Master

4.1.2.6 Ingen transmission til Slave

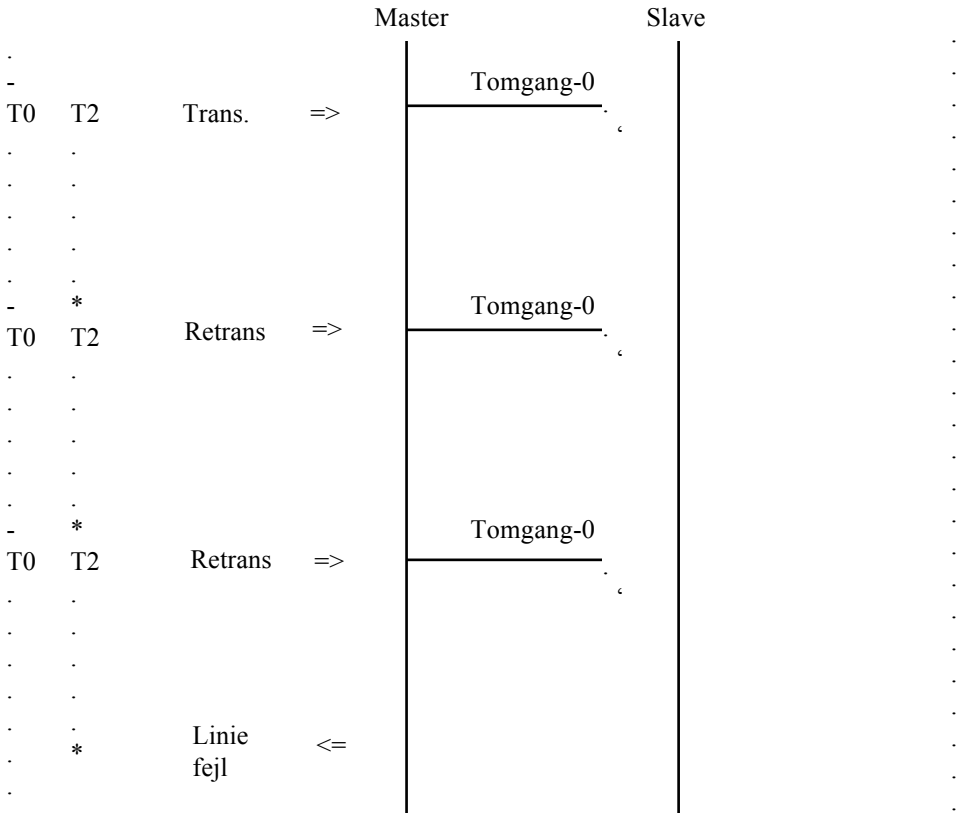


Fig. 4.13 L-Slave ingen transmission til Slave

4.1.2.7 Ingen transmission til Master

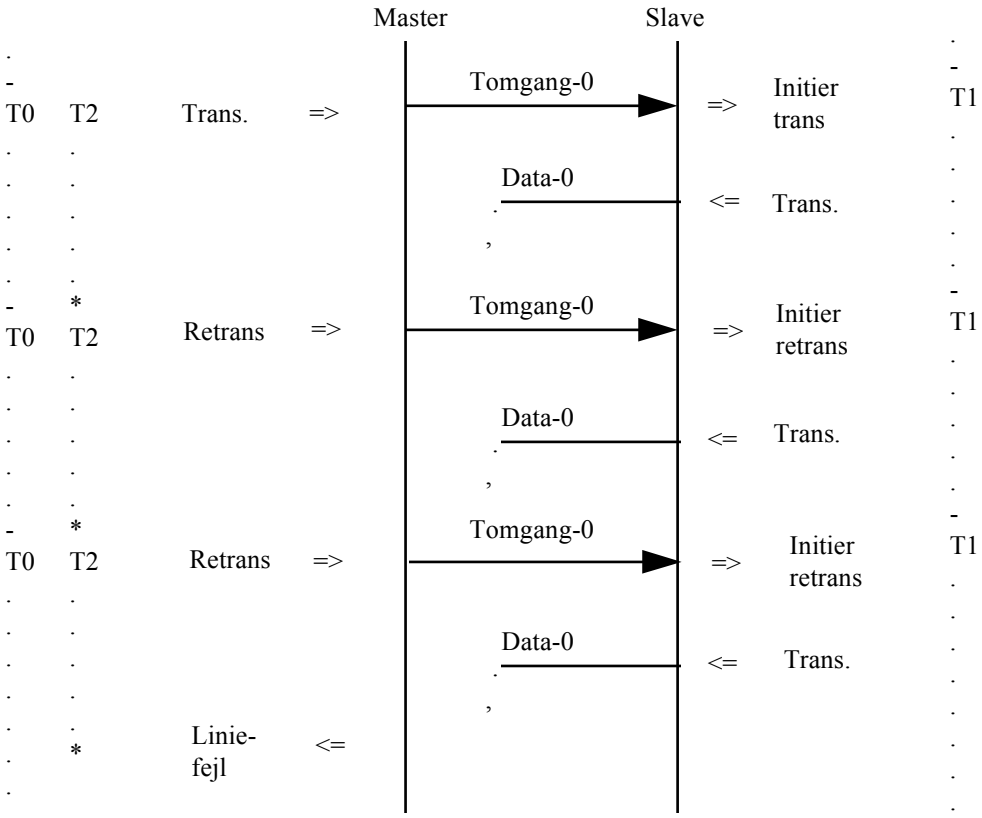


Fig. 4.14 L-Master ingen transmission til Master

4.1.3 Linietilstand L-Tomgang

I dette afsnit behandles Linietilstanden L-Tomgang, hvori Master er i M-Tomgang og Slave er i S-Tomgang tilstanden.

4.1.3.1 Normal tilfælde

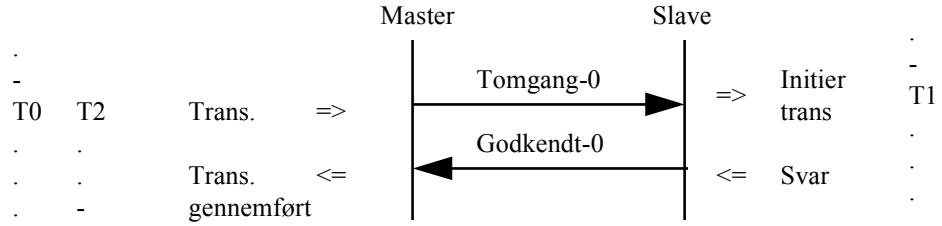


Fig. 4.15 L-Tomgang normal procedure

4.1.3.2 Transmissionsfejl fra Master

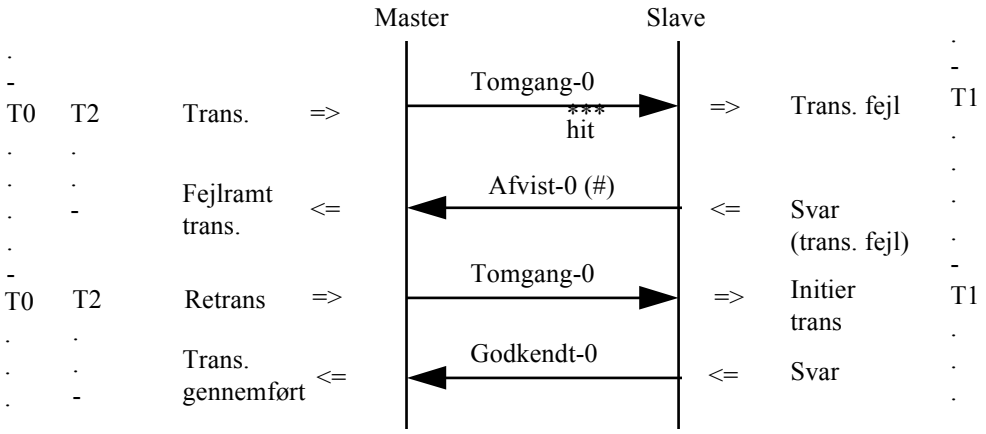


Fig. 4.16 L-Tomgang transmissionsfejl fra Master

4.1.3.3 Transmissionsfejl fra Slave

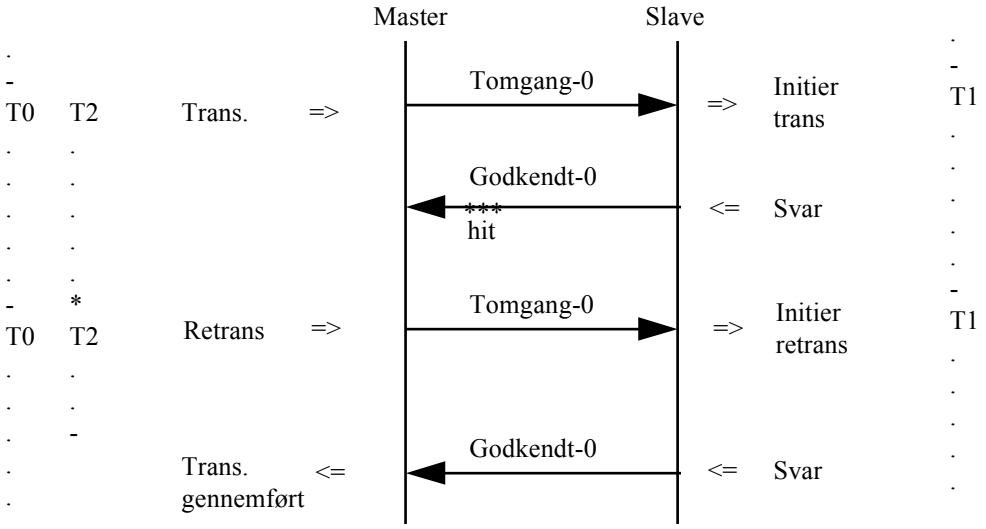


Fig. 4.17 L-Tomgang transmissionsfejl fra Slave

4.1.3.4 Transmission afbrudt til Slave

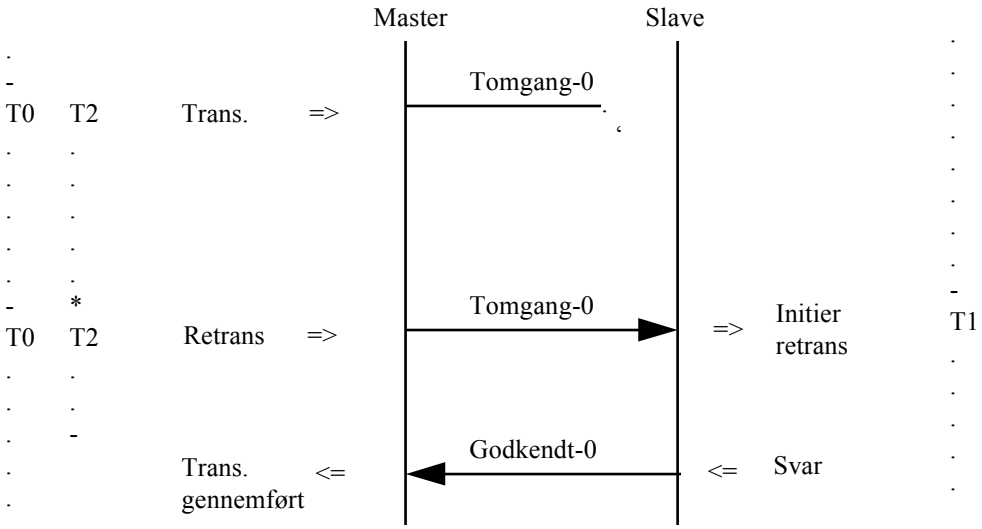


Fig. 4.18 L-Tomgang transmission afbrudt til Slave

4.1.3.5 Transmission afbrudt til Master

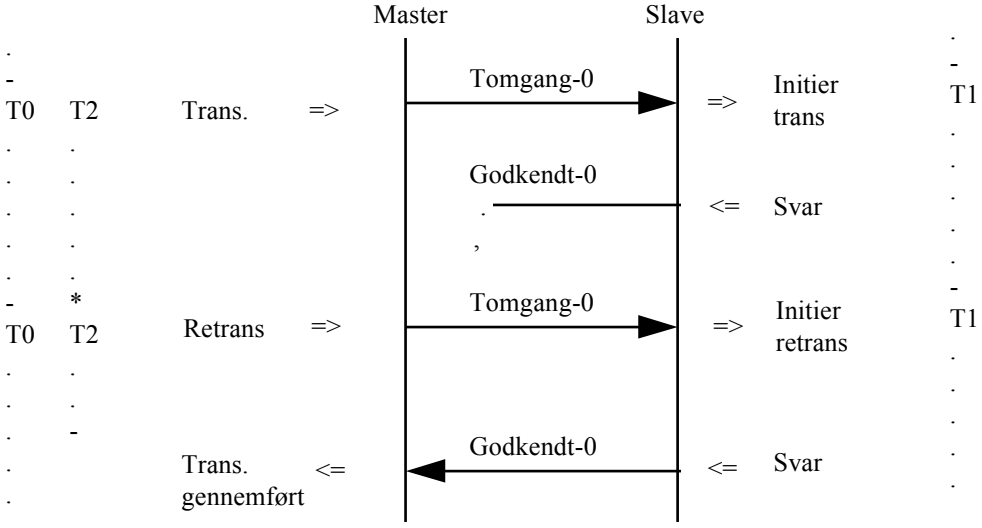


Fig. 4.19 L-Tomgang transmission afbrudt til Master

4.1.3.6 Ingen transmission til Slave

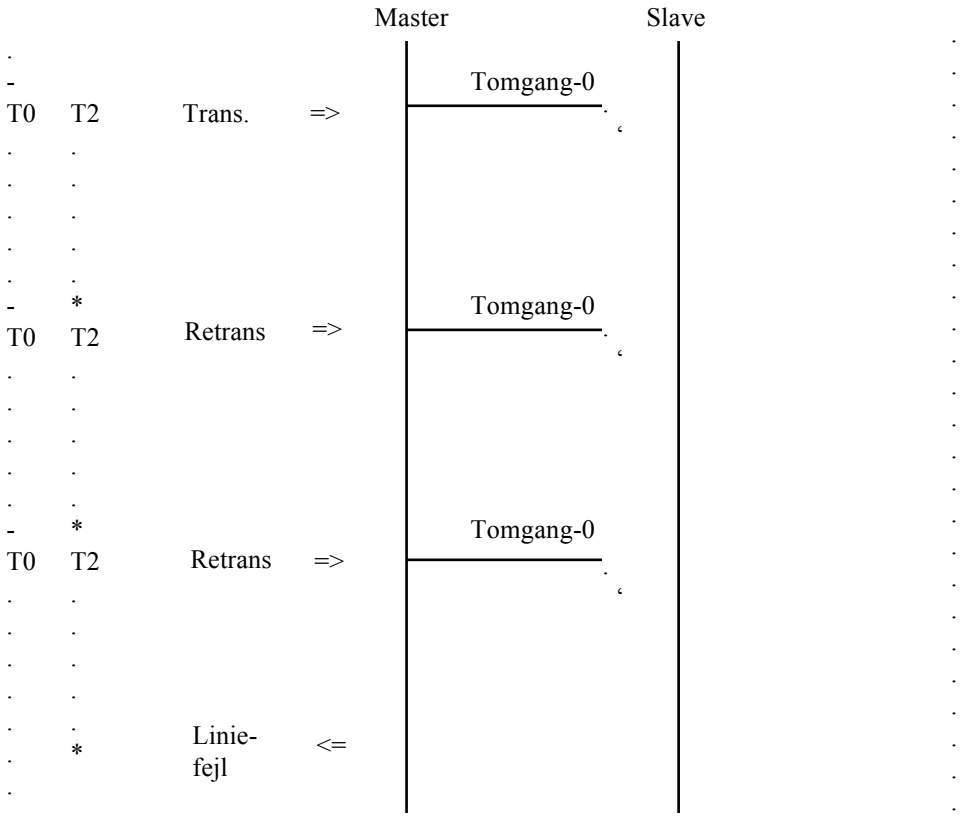


Fig. 4.20 L-Tomgang ingen transmission til Slave

4.1.3.7 Ingen transmission til Master

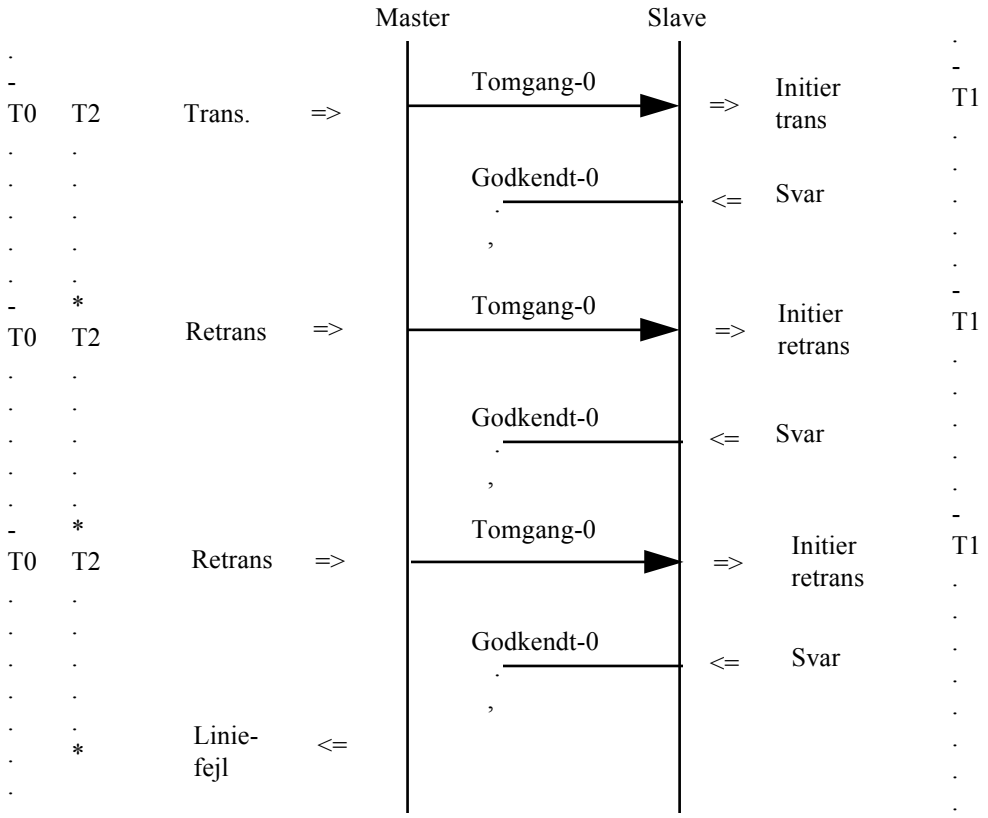


Fig. 4.21 L-Tomgang ingen transmission til Master

4.1.4 Linietilstand L-Fejl

Linietilstanden L-Fejl hænger sammen med Linieovervågningen i Master- Slave-systemet. Linieovervågningsfunktionen involverer alle Telegramtyper, og har til formål at sikre, at såvel Master som Slave er opdateret med den aktuelle tilstand på linien.

Linieovervågningen opererer med fire begreber:

- Master linie fejl.
- Master linie ok.
- Slave linie fejl.
- Slave linie ok.

4.1.4.1 Master linie fejl

Master linie fejl opstår på Master siden, når et Telegram er blevet udsendt tre gange uden, at der er modtaget svar. I figur 3.4 hedder tilstanden i Master M-Fejl, hvilket giver Linietilstanden L-Fejl.

4.1.4.2 Master linie ok

Master linie ok på Master siden opstår, når Master igen modtager korrekt svar på et udsendt Telegram. I figur 3.4 hedder tilstandene enten M-Data, M-Anmodning eller M-Tomgang. Linietilstanden bliver tilsvarende L-Master, L-Slave eller L-Tomgang.

4.1.4.3 Slave Liniefejl

Slave Liniefejl opstår på Slave, når der er gået T1 tid efter, at et Telegram er blevet modtaget korrekt uden, at der er modtaget et nyt korrekt Telegram. I figur 3.4 hedder denne tilstand S-Fejl. Det ses, at denne tilstand ikke påvirker Linietilstanden, idet den er irrelevant, når Master ikke er i M-Fejl.

4.1.4.4 Slave linie ok

Slave linie ok opstår, når Slave igen modtager et korrekt Telegram. Tilstanden på linien kan enten blive L-Master, L-Slave eller L-Tomgang, afhængig af de nye tilstande i henholdsvis Master og Slave.

4.1.4.5 Liniefejl i Master (M-Fejl)

Når Master har detekteret en Liniefejl, registreres fejlen. Master fortsætter uanfægtet sin transmission.

4.2 Løbenummerering

Løbenummeret påføres hvert Telegram og kan antage enten værdien 0 eller 1, idet Master højest kan have ét ukvitteret Telegram ude på linien ad gangen. Løbenummeret bruges af Master til at sikre, at kvitteringen hører til det korrekte afsendte Telegram. Slave bruger løbenummeret til at undgå at modtage dubleret data ved retransmission fra Master.

Følgende regler for løbenummereringen kan listes:

1. Master indeholder "udsendt løbenummer", dvs. det løbenummer, der skal sendes til Slave.
2. Slave indeholder "forventet løbenummer", dvs. det løbenummer, der forventes modtaget fra Master.
3. I L-Fejl sætter Master løbenummeret til en fast værdi (initialværdi).
4. I L-Fejl sætter Slave løbenummeret til den værdi, der modtages i det første korrekte Telegram fra Master.
5. Master skifter udsendte løbenummer for hver ny Trans, der initieres af Master. Retrans skifter således ikke løbenummeret.
6. Slave skifter forventet løbenummer, når det modtagne Telegram er korrekt, og når det modtagne løbenummer er lig med det forventede løbenummer.
7. Slave svarer altid med det modtagne løbenummer.
8. Slave anvender kun Telegrammer ved løbenummer skift. Hvis der ikke sker løbenummer skift antager Slaven, at det er dubleret data og kasserer derfor Telegrammet.

Note: I det følgende er situationerne beskrevet, hvor løbenummeret anvendes aktivt enten af Slave eller af Master. Bemærk at tidsvagterne T0, T1 og T2 er uafhængige af løbenummeret.

4.2.1 Normal drift

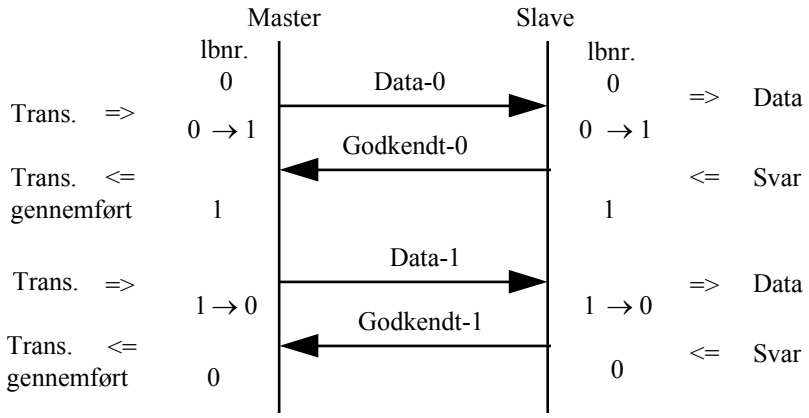


Fig. 4.22 Normal løbenummer procedure

4.2.2 Opstart

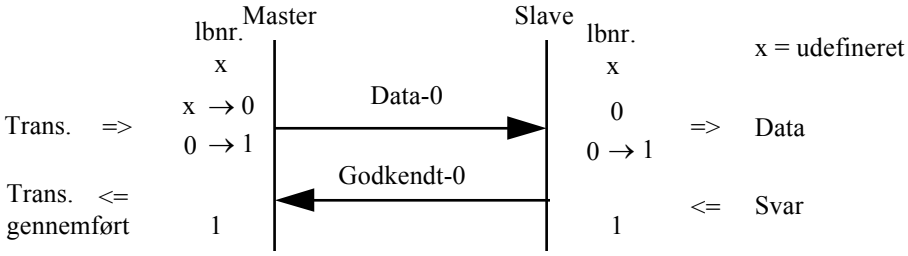


Fig. 4.23 Løbenummerering ved opstart

4.2.3 Liniefejl (L-Fejl)

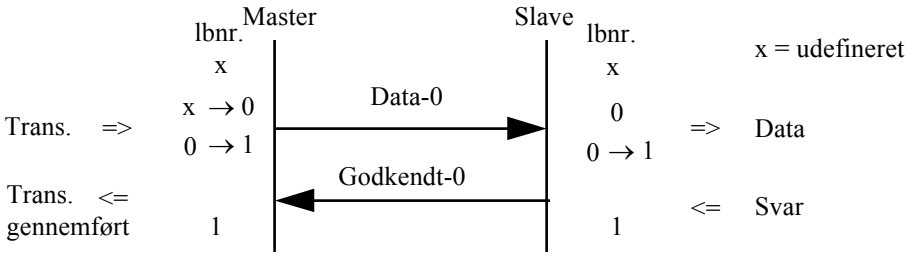


Fig. 4.24 Løbenummerering ved Liniefejl

4.3 Linieovervågning

Denne procedure sikrer, at linien mellem Master og Slave er overvåget hele tiden. Dette betyder, at proceduren er implementeret både i Master og Slave.

4.3.1 Master Linieovervågning

I Master sørger proceduren for at:

4.3.1.1 Udsendelse af telegram inden timerudløb

Der udsendes et Telegram mindst een gang, indenfor tiden T0 (T0 er timer nr. 0, som er defineret i kapitel 3)

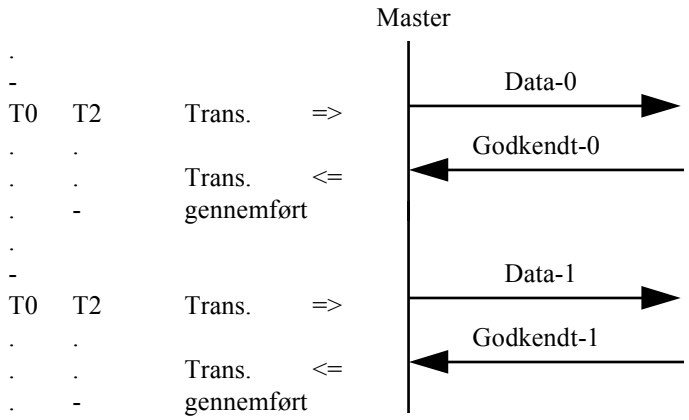


Fig. 4.25 Udsendelse af Telegram inden T0 udløb

4.3.1.2 Polling ved timerudløb

Hvis der ingen Telegrammer sendes fra Master, da udsendes et Tomgangtelegram. Linieovervågningen har således en Poll-funktion overfor Slave.

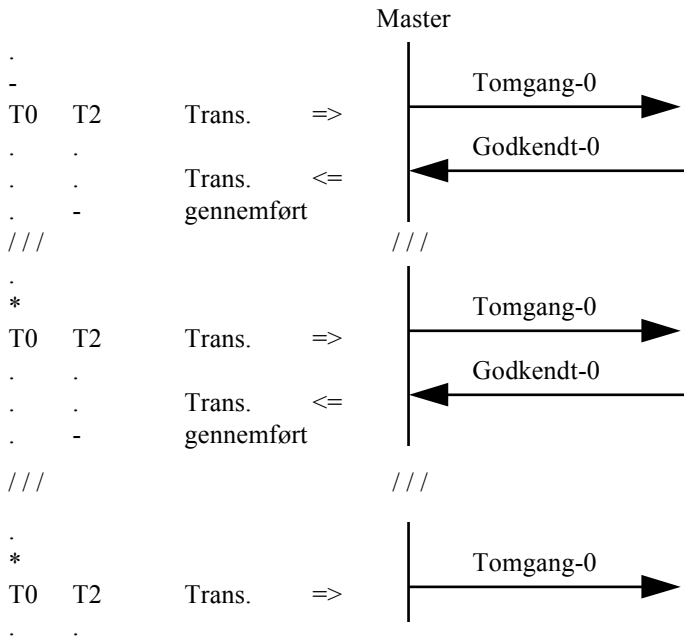


Fig. 4.26 Udsendelse af Telegram ved T0 udløb

4.3.1.3 Melding af liniestatus

At melde Liniefejl eller linie-ok er afhængig af hvilken status, der modtages på de udsendte Telegrammer

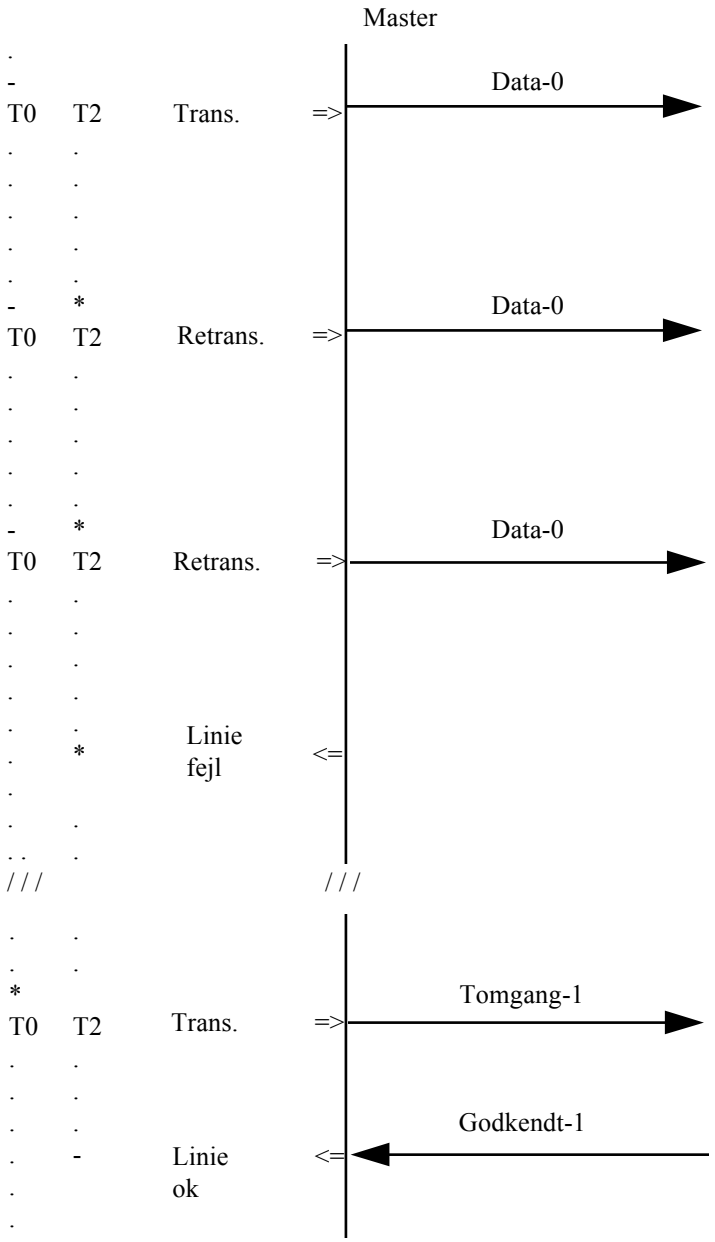


Fig. 4.27 Melding af Liniefejl og linie-ok

4.3.2 Slave overvågning

I Slave sørger proceduren for:

4.3.2.1 Melding af liniestatus

At melde Liniefejl og linie-ok er afhængig af hvilken status der modtages på de modtagne Telegrammer, indenfor tiden T1

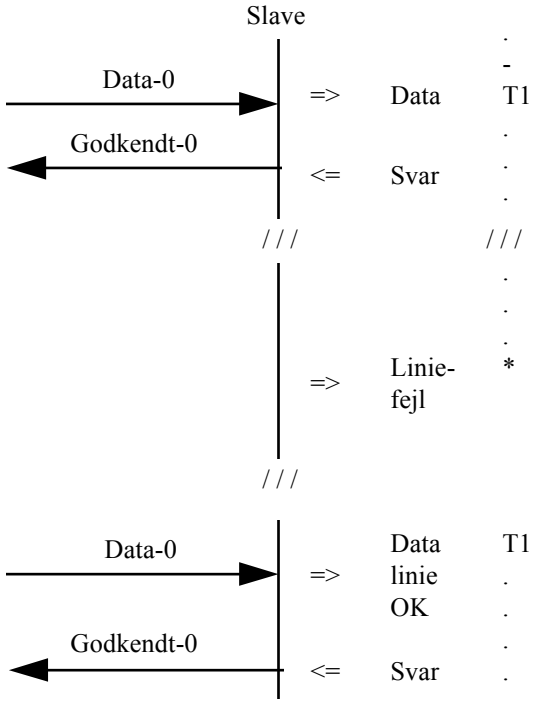


Fig. 4.28 Melding af Liniefejl og linie-ok

4.4 Fremgangsmåde til checksumsberegning

Checksum fremkommer ved løbende (dvs. under transmissionen), at "addere modulo 2 i n'et" til to's kompliment, og denne Checksum udgør da den sidste karakter af den transmitterede blok.

Når blokken retransmitteres "adderes modulo 2^n" de enkelte bytes inklusiv Checksummen i modtageren. Såfremt transmissionen har været korrekt, vil resultatet af den løbende addition være nul.

Eksempel: Checksum på transmitterside.
Vi ønsker at transmittere en blok, bestående af følgende 5 databytes under anvendelse af Checksum mod-256 (100 HEX). Mod-256 fremkommer ved, at vi har 8 databit og modulo beskrives som 2 i n'de er lig med antallet af databit, her 8. Dette giver $2^8 + 1 = 256$:

byte 0: 01 = 0000 0001
byte 1: 10 = 0001 0000
byte 2: A0 = 1010 0000 Hexadecimal
byte 3: 00 = 0000 0000 notation
byte 4: C9 = 1100 1001

"sum mod 2^8" = 0111 1010

1's komplement = 1000 0101
+ 1

Checksum =1000 0110 = 86H (byte 5)

Det vil sige, at vi skal transmittere følgende blok:

01 10 A0 00 C9 86
Databytes Checksum

Eksempel: Checksum på recieverside.

I modtageren foregår løbende en addition mod 2^8:
byte 0: 01 = 0000 0001
byte 1: 10 = 0001 0000
byte 2: A0 = 1010 0000 forudsat korrekt
byte 3: 00 = 0000 0000 overførsel
byte 4: C9 = 1100 1001
byte 5: 86 = 1000 0110
"sum mod 2^8" = 0000 0000 ingen fejl!

Der skal knyttes en kommentar til Checksum beregningen:

Da Telegrammerne holdes i begrænset ASCII, af hensyn til en eventuel printerovervågning, omkodes Checksum til en læsbar ASCII kode.

Det vil sige, at hvis Checksummen bliver f.eks. 8AH omskrives den til 8A ASCII, hvilket vil sige 38H og 41H.

Hvor den før var på 1 byte er den altså nu på 2 bytes.

Den omkodes så til hexadecimal ved modtagelsen, før den sidste "addition mod 256" foregår.

5.0 FORMATSPECIFIKATIONER

Dette kapitel indeholder format for Telegram, specifikation af Telegramtyper, format for Packettype, det begrænsede karaktersæt til Datapackets og specifikation af Packettypes.

5.1 Telegramformat

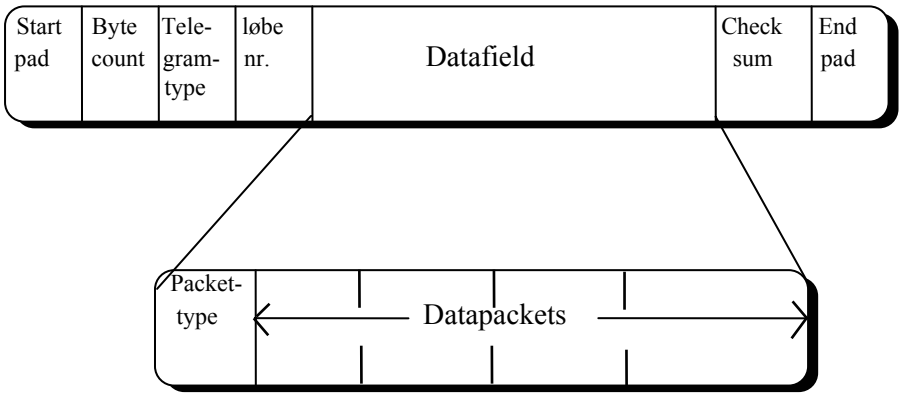


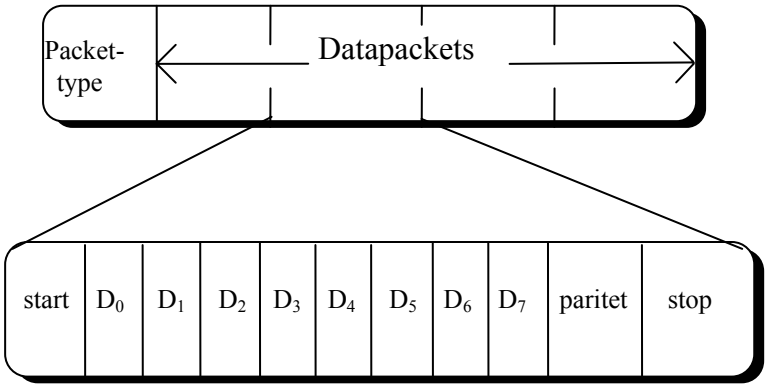
Fig. 5.1 Eksempel på et telegram.

5.2 Specifikation af Telegramtyper

Packettype	Beskrivelse	Beschreibung	ATC-TC	ATC-MSR3	MSR3-TC	ATC-HLOG	Telegram-kategori: Anmodning	Data	Tomgang
1	Aktuelle data fra ATC til MSR3	funkspezifische Tlg.daten		X				X	
2	togdata fra ATC til HLOG	Zugdaten von ATC an HLOG				X		X	
3	hastighedsinfo. fra ATC til HLOG	Geschwindigkeitinfo. von ATC an HLOG				X		X	
4	Baliseinfo. fra ATC til HLOG	Streckentlg. von ATC an HLOG				X		X	
5	nødstop fra MSR3 til ATC	Nothalt von MSR3 an ATC		X				X	
6	Kvittering med hastighed fra HLOG til ATC	Acknowledge von HLOG an ATC				X		X	
7	Forbikørsel af stop fra ATC til HLOG	Signalvorbeifahrt von ATC an HLOG				X		X	
8	disp.	frei							
9	disp.	frei							
A	godkendt	anerkannt	X	X	X	X			
B	aktuelle data fra TC til ATC	Zugdaten von TC an ATC	X					X	
C	reserveret	reserviert		X					
D	reserveret	reserviert		X					
E	aktuelle data fra ATC til TC	Zugdaten von ATC an TC	X					X	
F	Pos./signalnr. fra ATC til MSR3	funkspezifische Tlg.daten		X				X	
G	fejlkode fra ATC til TC og fra ATC til HLOG	Fehlercode von ATC an TC und von ATC an HLOG	X			X		X	
H	HKT hastigheds-information	HKT-geschwindigkeitmeldung				X		X	
I	reserveret	reserviert		X					
J	reserveret	reserviert		X					
K	aktuelle data fra TC til MSR3	funkspezifische Tlg.daten			X			X	
L	reserveret	reserviert		X	X				
M	reserveret	reserviert		X					
N	afvist	abgewiesen	X	X	X	X			

Specifikation af Telegramtyper									
Packettype	Beskrivelse	Beschreibung	ATC-TC	ATC-MSR3	MSR3-TC	ATC-HLOG	Telegram-kategori		
							Anmodning	Data	Tomgang
O	aktuelle data fra MSR3 til TC	funkspezifische Tlg.daten			X			X	
P	disp.	frei							
Q	reserveret	reserviert		X	X				
S	reserveret	reserviert		X					
T	reserveret	reserviert		X					
U	reserveret	reserviert		X					
V	Reserveret	reserviert		X					
W	Reserveret	reserviert		X					
X	Reserveret	reserviert		X					
Y	Reserveret	reserviert		X					
Z	Reserveret	reserviert		X					
a	Driftstelegram fra ATC til HLOG	Betriebs-tlg von ATC an HLOG				X		X	
b	disp.	frei							
c	Test	Test	X			X		X	
d	Reserveret	reserviert	X						
e	Reserveret	reserviert	X						
f	disp.	frei							
g	disp.	frei							
h	disp.	frei							
i	disp.	frei							
j	disp.	frei							
k	disp.	frei							
l	disp.	frei							
m	disp.	frei							
n	Tognummer til TC	Zugnummer zum TC			X			X	
o	Fejlkode MSR3 til TC	Fehlermeldung MSR3 zum TC			X			X	
p	disp.	frei							
q	disp.	frei							
r	Anmodning om specifikke data	Ersuchen spezifischen Daten	X	X	X		X		
s	disp.	frei							
t	Tomgangs data	Leerlaufs Daten	X	X	X				X
u	disp.	frei							
v	disp.	frei							
w	disp.	frei							
x	disp.	frei							
y	disp.	frei							
z	disp.	frei							

5.3 Packettypeformat



5.3.1 Det begrænsede ASCII karaktersæt til Datapackets

Kun følgende karakterer er gyldige:

ASCII Char.	Binær	OCT	HEX	DEC
%	00100101	045	25	37
(	00101000	050	28	40
)	00101001	051	29	41
-	00101101	055	2D	45
:	00111010	072	3A	58
+	00101011	053	2B	43
0	00110000	060	30	48
1	00110001	061	31	49
2	00110010	062	32	50
3	00110011	063	33	51
4	00110100	064	34	52
5	00110101	065	35	53
6	00110110	066	36	54
7	00110111	067	37	55
8	00111000	070	38	56
9	00111001	071	39	57
A	01000001	101	41	65
B	01000010	102	42	66
C	01000011	103	43	67
D	01000100	104	44	68
E	01000101	105	45	69
F	01000110	106	46	70
G	01000111	107	47	71
H	01001000	110	48	72
I	01001001	111	49	73
J	01001010	112	4A	74
K	01001011	113	4B	75
L	01001100	114	4C	76
M	01001101	115	4D	77
N	01001110	116	4E	78
O	01001111	117	4F	79
P	01010000	120	50	80
Q	01010001	121	51	81
R	01010010	122	52	82
S	01010011	123	53	83
T	01010100	124	54	84
U	01010101	125	55	85
W	01010110	126	56	86
V	01010111	127	57	87
X	01011000	130	58	88
Y	01011001	131	59	89
Z	01011010	132	5A	90
space	00100000	040	20	32

ASCII Char.	Binær	OCT	HEX	DEC
a	01100001	141	61	97
b	01100010	142	62	98
c	01100011	143	63	99
d	01100100	144	64	100
e	01100101	145	65	101
f	01100110	146	66	102
g	01100111	147	67	103
h	01101000	150	68	104
i	01101001	151	69	105
j	01101010	152	6A	106
k	01101011	153	6B	107
l	01101100	154	6C	108
m	01101101	155	6D	109
n	01101110	156	6E	110
o	01101111	157	6F	111
p	01110000	160	70	112
q	01110001	161	71	113
r	01110010	162	72	114
s	01110011	163	73	115
t	01110100	164	74	116
u	01110101	165	75	117
v	01110110	166	76	118
w	01110111	167	77	119
x	01111000	170	78	120
y	01111001	171	79	121
z	01111010	172	7A	122
Følgende tolkes som DS 2089				
æ	01111011	173	7B	123
ø	01111100	174	7C	124
å	01111101	175	7D	125
Æ	01011011	133	5B	91
Ø	01011100	134	5C	92
Å	01011101	135	5D	93

5.4 Specifikation af Packettypes

Packettype	Beskrivelse	Beschreibung	ATC-TC	ATC-MSR3	MSR3-TC	ATC-HLOG
A	godkendt	anerkannt	X	X	X	X
B	anmodning om aktuelle togdata	Anruf	X	X	X	
C	C-kanal	C-kanal	X	X	X	
D	D-kanal	D-kanal	X	X	X	
E	fejlkode	Fehlercode	X			X
F	FC-kanal	FC-kanal		X	X	
G	overvågnings-hastighed	Uberwachungsgeschwindigkeit				X
H	max. hastighed i km/h	max. Geschwindigkeit in km/h	X			X
I	driftsfunktion	Betriebsfunktion				X
J	hjuldiameter	Raddurchmesser				X
K	øjeblikshastighed	Istgeschwindigkeit				X
L	toglængde i meter	Zuglänge in Metern	X			X
M	aktuel driftsform	aktuelle Betriebsart		X	X	
N	afvist	abgewiesen	X	X	X	X
O	reserve	Reserve			/	
P	position	Position	X	X	X	
Q	reserve	Reserve			/	
R	ATC-retning	ATC-richtung	X			X
S	forbikørsel af stop	Signalvorbei-fahrt			X	X
T	tognummer	Zugnummer			X	
U	nødstop	Nothalt		X		
V	vagtfunktion	Wachefunktion		X	X	
W	C-effekt	C sendeleistung		X		
X	D-effekt	D sendeleistung		X		
Y	til- og afmelding	Ab- und Anmeldung		X		
Z	landkendings-tal	Landkendungs-zahl		X		
%	bremseprocent	Bremsprozente	X			X
a	balise GK 1	Balise GK 1				X
b	balise GK 2	Balise GK 2				X
c	balise GK 3	Balise GK 3				X
d	balise GK 4	Balise GK 4				X
e	balise reserve	Balise Reserve				X
f	balise reserve	Balise Reserve				X
g	balise reserve	Balise Reserve				X
h	balise driftskontrol	Balise Betriebskontrolle				X
i	reserveret	reserviert		/	/	
j	reserveret	reserviert		/	/	
k	reserveret	reserviert		/	/	
l	reserveret	reserviert		/	/	
m	lampetest	Lampentest	X			X
n	S-/DK-tilst.	S-/DK-Zustand				X*)
o	disp.	frei				
p	disp.	frei				
q	disp.	frei				
r	disp.	frei				
s	disp.	frei				
t	disp.	frei				
u	disp.	frei				

*) Benyttes også til HKT-info i S-tog

6.0 TELEGRAMINDHOLD ATC - TC

Dette kapitel indeholder en beskrivelse af telegramindholdet mellem ATC og TC.

6.1 Telegramindhold mellem ATC og TC

Denne forbindelses formål er primært, at ATC i køretøjer med TC kan hente togets data ved opstart. Derefter kan ATC cyklisk blive opdateret. For en nærmere uddybning af de forskellige Telegramtypers indhold henvises til bilag 1 i dette kapitel.

6.2 Telegramtype r: Anmodning om aktuelle data

Dette Telegram sender ATC til TC, umiddelbart efter opstart, som opfordring til TC om at sende togdata (Telegram B). Derefter sendes Telegrammet cyklisk, og derved virker det som Linieovervågning.

Packettype B: Anmodning

B	*

TC anmodes om at besvare dette Telegram med den Telegramtype, som data angiver. Eks. data = B.

6.3 Telegramtype B: Aktuelle data fra TC til ATC

På opfordring fra ATC med anmodningstelegrammet r og data = B, kvitterer TC ikke med ACK, men med Telegramtype B.

Packettype L: Toglængde

L	*	*	*

Data ligger i intervallet fra 0 til 999 m.

Packettype %: Bremseprocent

%	*	*	*

Bremseprocent i intervallet 0 til 999 %.

Packettype H: Max hastighed

H	*	*	*

Hastighed i intervallet fra 0 til 999 km/t.

Packettype R: ATC retning

R	*	*

Data: 8 8 = Retning A
0 0 = Retning b

Udlægning af ATC-retning ligger udenfor denne protokol. Retning "A" sendes, også når TC ikke kender retningen.

6.4 Telegramtype E: Aktuelle data fra ATC til TC

Udgår.

6.5 Telegramtype F: Position

Dette Telegram sender ATC umiddelbart efter passage af en balise. Positionen er udledt af signalnummeret.

Packettype P: Position

P	*	*	*	*

Område:
1. ciffer efter P kan have følgende værdier 0,1,2,3,4,5,6,7. (SNR1 = 3 bit)
0 → Station
1,2,3,4,5,6,7 → Spor nr. på strækning

De næste 3 cifre dækker decimalt området 0 - 999 (SNR2 = 10 bit). De tre cifre angiver strækningskilometrering. (I området 0 - 99,9 km angiver de tre cifre en afstand i hektometer. I området fra 100 km og opefter angiver de tre cifre en afstand i km).

6.6 Telegramtype G: Fejlkode

ATC's fejlkode sendes til TC umiddelbart efter, at den er detekteret. TC lagrer kun de første to cifre.

Packettype E: Fejlkode

E	*	*	*

Data ligger i intervallet fra 000 til 999.

6.7 Telegramtype b: Bremseprocent fra TC

Udgået.

6.8 Telegramtype c: Test

Dette Telegram tjener til, at alle lamper i førerrumssignalet samt indkodningspanel afprøves. Ved opstart af ATC afprøves disse, dog findes der lamper i førerrumsssignalet, som styres af HLOG og dels TC. Når TC og HLOG modtager dette Telegram, skal de tænde hhv. slukke lamperne.

Packettype m: Lampetest

m	*

Data: S: Sluk lampe
T: Tænd lampe

6.9 Telegramtype t: Tomgang

Indtil videre undertrykkes dette Telegram, idet der er etableret Linieovervågning, ved at ATC sender et Telegram med en fast minimums tidsinterval.

6.10 Telegramtype N: Afvist Telegram

Det er aftalt, at data N bliver følgende:

Data N = 1 betyder Checksumsfejl
2 betyder Timingsfejl
A betyder Applikationsfejl *)

*) Applikationsfejl (datavalideringsfejl) dvs. et fuldt korrekt dekodet Telegram, der ikke giver nogen mening.

Bilag 6.1

Telegramindhold ATC-TC

Telegramtype	A	c	r	t	N
Startpad	LF	LF	LF	LF	LF
Bytecounts	Bytecount 1	Bytecount 1	Bytecount 1	Bytecount 1	Bytecount 1
	Bytecount 2	Bytecount 2	Bytecount 2	Bytecount 2	Bytecount 2
Telegramtype	A	c	r	t	N
	Telegramløbenr.	Telegramløbenr.	Telegramløbenr.	Telegramløbenr.	Telegramløbenr.
Packettype	A	m	B	B	N
Datapacket	data A	data m	data B	data B	data N
Checksum-bytes	Checksum 1	Checksum 1	Checksum 1	Checksum 1	Checksum 1
	Checksum 2	Checksum 2	Checksum 2	Checksum 2	Checksum 2
Endpad	CR	CR	CR	CR	CR
	Godkendt TC til ATC	Anmodning om test ATC til TC	Anmodning om specifikke data ATC til TC	Tomgangsdata ATC til TC	Afvist TC til ATC

Telegramtype	B
Startpad	LF
Bytecounts	Bytecount 1
	Bytecount 2
Telegramtype	B
	Telegramløbenr.
Packettype	L
Datapacket	data L
	data L
	data L
Packettype	%
Datapacket	data %
	data %
	data %
Packettype	H
Datapacket	data H
	data H
	data H
Packettype	R
Datapacket	data R
	data R
Checksum-bytes	Checksum 1
	Checksum 2
Endpad	CR
	Aktuelle data TC til ATC

Telegramtype	F
Startpad	LF
Bytecounts	Bytecount 1
	Bytecount 2
Telegramtype	F
	Telegramløbenr.
Packettype	P
Datapacket	data P
	data P
	data P
	data P
Checksum-bytes	Checksum 1
	Checksum 2
Endpad	CR
	Pos/signal nr. ATC til TC

Telegramtype	G
startpad	LF
Bytecounts	Bytecount 1
	Bytecount 2
Telegramtype	G
	Telegramløbenr.
Packettype	E
Datapacket	data E
	data E
	data E
Checksum-bytes	Checksum 1
	Checksum 2
Endpad	CR
	Fejlkode ATC til TC

7.0 Telegramindhold ATC - MSR3

Dette kapitel indeholder en beskrivelse af indholdet i Telegrammer, der udveksles mellem ATC og MSR3.

7.1 Telegramindhold mellem ATC - MSR3

Denne forbindelses primære formål er automatisk styring af driftsformer og kanalskift for MSR3 samt positionsindikation. Desuden findes der mulighed for udveksling af nødstoptelegram. For det præcise indhold af hver Telegramtype henvises til bilag 1 i dette kapitel.

7.2 Telegramtype 1: Radiobalisetelegram

Telegrammet sendes fra ATC til MSR3 umiddelbart efter, at en radiobalise (GK = 4) er passeret.

Datapacket type F: FC-kanal

F	C	*	*

Område:
2 decimal cifre, 0 - 99 ⇒ 7 bit

eks. Er der i radiobalisen kodet følgende for FC-kanal 0111110, sender ATC følgende til MSR3 efter passage af balisen:

F	C	6	2

Datapacket type D: D-kanal

D	*	*

Område:
2 decimal cifre, 0 - 99 ⇒ 7 bit

eks. Er der i radiobalisen kodet følgende for D-kanal 0010010, sender ATC følgende til MSR3 efter passage af balisen:

D	1	8

Datapacket type C: C-kanal

C	*	*

Område:
2 decimal cifre, 0 - 99 ⇒ 7 bit

eks. Er der i radiobalisen kodet følgende for C-kanal 0110000, sender ATC følgende til MSR3 efter passage af balisen:

C	4	8

Datapacket type M. Aktuel driftsform

M	:	*

Område:
1 ciffer med 8 muligheder ⇒ 3 bit
A : 0 0 0
B : 0 0 1
C : 0 1 0
D : 0 1 1
Ikke def. : 1 0 0
F : 1 0 1
Ikke def. : 1 1 0
Ikke def. : 1 1 1

"F" betyder, at Datapacket skal ignoreres i MSR3.

eks. I balisen er de 3 bit kodet som 0 1 0, følgende vil i givet fald blive sendt fra ATC til MSR3.

M	:	C

Datapacket type V: Vagtfunktion

V	:	*	*

Område:
2 cifre med hver 8 muligheder ⇒ 6 bit

Til hvert ciffer benyttes 3 bit, de kodes efter følgende tabel:

'A' : 0 0 0
'B' : 0 0 1
'C' : 0 1 0
'D' : 0 1 1
Ikke def. : 1 0 0
'F' : 1 0 1
Ikke def. : 1 1 0
' ' ascii 32D : 1 1 1
'F' betyder, at Datapacket skal ignoreres i MSR3.

Eks. I balisen er de 6 bit kodet som 1 1 1 0 1 1, følgende vil i givet fald blive sendt fra ATC til MSR3.

V	:		D

NB. Efter kolon (':') sendes, i det viste eksempel, ascii 32 decimalt = space.

Datapacket type P: Position

P	*	*	*	*

Område:
1. ciffer efter P kan have følgende værdier 0,1,2,3,4,5,6,7. (SNR1 = 3 bit).
0 → Station.
1,2,3,4,5,6,7 → Spor nr på strækning.

De næste 3 cifre dækker decimalt området 0 - 999 (SNR2 = 10 bit). De tre cifre angiver strækningskilometrering. (I området 0 - 99,9 km angiver de tre cifre en afstand i hektometer. I området fra 100 km og opefter angiver de tre cifre en afstand i km).

Datapacket type W: C-effekt

W	*

Område:
8 muligheder ⇒ 3 bit.

F : 0 0 0
1 : 0 0 1
ikke def. : 0 1 0
ikke def. : 0 1 1
ikke def. : 1 0 0
ikke def. : 1 0 1
6 : 1 1 0
ikke def. : 1 1 1

'F' betyder, at Datapacket skal ignoreres i MSR3.

Eks. Er der i balisetelegrammet programmeret 0 0 1, sendes følgende fra ATC til MSR3:

W	1

Datapacket type X: D-effekt

X	*

Område:
8 muligheder ⇒ 3 bit.

F : 0 0 0
1 : 0 0 1
ikke def. : 0 1 0
ikke def. : 0 1 1
ikke def. : 1 0 0
ikke def. : 1 0 1
6 : 1 1 0
ikke def. : 1 1 1

'F' betyder, at Datapacket skal ignoreres i MSR3.

eks. Er der i balisetelegrammet programmeret 0 0 1, sendes følgende fra ATC til MSR3:

X	1

Datapacket type Y: Til-/Afmelding

Y	*

Område:
32 muligheder ⇒ 5 bit.

0	→ Mode 0	16	→ Mode G
1	→ Mode 1	17	→ Mode H
2	→ Mode 2	18	→ Mode I
3	→ Mode 3	19	→ Mode J
4	→ Mode 4	20	→ Mode K
5	→ Mode 5	21	→ Mode L
6	→ Mode 6	22	→ Mode M
7	→ Mode 7	23	→ Mode N
8	→ Mode 8	24	→ Mode O
9	→ Mode 9	25	→ Mode P
10	→ Mode A	26	→ Mode Q
11	→ Mode B	27	→ Mode R
12	→ Mode C	28	→ Mode S
13	→ Mode D	29	→ Mode T
14	→ Mode E	30	→ Mode U
15	→ Mode F	31	→ Mode V

eks. I balisen er de 5 bit kodet som 0 1 0 1 1, følgende vil i givet fald blive sendt fra ATC til MSR3.

Y	B

Datapacket type Z: Lande-kode

Z	*

Område:
8 muligheder ⇒ 3 bit.

0 0 0 - Land 0 (Danmark)
0 0 1 - Land 1 (Tyskland)
0 1 0 - Land 2
0 1 1 - Land 3
1 0 0 - Land 4
1 0 1 - Land 5
1 1 0 - Land 6
1 1 1 - Land 7

eks. Er bit'ene i balisetelegrammet programmeret som 0 0 1 sendes følgende fra ATC til MSR3:

Z	1

7.3 Telegramtype r: Anmodning om specifikke data - Nødstop

ATC spørger cyklisk, om MSR3 har modtaget besked om "Nødstop fra land". MSR3 besvarer ikke dette Telegram med "godkendt", men med Telegram 5 - Nødstop. Dette Telegram virker dermed automatisk som Linieovervågning. I stedet for den generelle T0 anvendes T0=5 sekunder, for at reducere forsinkelsen af et nødstop.

Datapacket type B - Anmodning

B	*

Slaven anmodes om at besvare dette Telegram med den Telegramtype, som data er lig med. Hvis nødstop er data = 5.

7.4 Telegramtype 5: Nødstop

Dette Telegram sendes fra MSR3 til ATC som svar på Telegram r fra ATC.

Packettype U - Nødstop

U	*

Data: + : MSR3 har modtaget nødstop
- : MSR3 har **intet** nødstop

7.5 Telegramtype F: Position

Dette Telegram sender ATC umiddelbart efter passage af en balise. Positionen er udledt af signalnummeret.

Packettype P - Position

P	*	*	*	*

Område:
1. ciffer efter P kan have følgende værdier 0,1,2,3,4,5,6,7 (SNR1 =3 bit).

0 → Station.
1,2,3,4,5,6,7 → Spor nr. på strækning.

De næste 3 cifre dækker decimalt området 0 - 999 (SNR2 = 10 bit). De tre cifre angiver strækningskilometrering. (I området 0 - 99,9 km angiver de tre cifre en afstand i hektometer. I området fra 100 km og opefter angiver de tre cifre en afstand i km).

7.6 Telegramtype N: Afvist Telegram

Det er aftalt, at data N bliver følgende:

Data N = 1 betyder checksumsfejl
2 betyder timingsfejl
A betyder applikationsfejl *)

*) Applikationsfejl (datavalideringsfejl) dvs. et fuldt korrekt dekodet Telegram, der ikke giver nogen mening.

Bilag 7.1

Telegramindhold ATC-MSR3

Telegramtype	A	5	r	t	N
Startpad	LF	LF	LF	LF	LF
Bytecounts	Bytecount 1	Bytecount 1	Bytecount 1	Bytecount 1	Bytecount 1
	Bytecount 2	Bytecount 2	Bytecount 2	Bytecount 2	Bytecount 2
Telegramtype	A	5	r	t	N
	Telegramløbenr.	Telegramløbenr.	Telegramløbenr.	Telegramløbenr.	Telegramløbenr.
Packettype	A	U	B	B	N
Datapacket	data A	data U	data B	data B	data N
Checksum-bytes	Checksum 1	Checksum 1	Checksum 1	Checksum 1	Checksum 1
	Checksum 2	Checksum 2	Checksum 2	Checksum 2	Checksum 2
Endpad	CR	CR	CR	CR	CR
	Godkendt MSR3 til ATC	Nødstop MSR3 til ATC	Anmodning om specifikke data ATC til MSR3	Tomgangsdata ATC til MSR3	Afvist MSR3 til ATC

Telegramtype	1
Startpad	LF
Bytecounts	Bytecount 1
	Bytecount 2
Telegramtype	1
	Telegramløbenr.
Packettype	F
Datapacket	data F
	data F
	data F
Packettype	D
Datapacket	data D
	data D
Packettype	C
Datapacket	data C
	data C
Packettype	M
Datapacket	data M
	data M
Packettype	V
Datapacket	data V
	data V
	data V
Packettype	P
Datapacket	data P
	data P
	data P
	data P
Packettype	W
Datapacket	data W
Packettype	X
Datapacket	data X
Packettype	Y
Datapacket	data Y
Packettype	Z
Datapacket	data Z
Checksum-bytes	Checksum 1
	Checksum 2
Endpad	CR
	Radio data ATC til MSR3

Telegramtype	F
Startpad	LF
Bytecounts	Bytecount 1
	Bytecount 2
Telegramtype	F
	Telegramløbenr.
Packettype	P
Datapacket	data P
	data P
	data P
	data P
Checksum-bytes	Checksum 1
	Checksum 2
Endpad	CR
	Pos/signal nr. ATC til MSR3

8.0 TELEGRAMINDHOLD MSR3 - TC

Dette kapitel indeholder en beskrivelse af telegramindholdet mellem MSR3 og TC.

8.1 Telegramindhold mellem MSR3 - TC

Denne forbindelses primære formål er at udveksle aktuelle data. For det præcise indhold af hver Telegramtype henvises til bilag 1 i dette kapitel.

8.2 Telegramtype K: Aktuelle data, TC til MSR3

Telegrammet sendes fra TC til MSR3 som svar på Telegramtype r.

Datapacket type F: FC-kanal

F	C	*	*

Område:
2 decimal cifre (0..9)
F.eks. FC-kanal 62 er kodet som følger:

F	C	6	2

Datapacket type D: D-kanal

D	*	*

Område:
2 decimal cifre (0..9)
F.eks. D-kanal 18 er kodet som følger:

D	1	8

Datapacket type C: C-kanal

C	*	*

Område:
2 decimal cifre (0..9)
F.eks. C-kanal 48 er kodet som følger:

C	4	8

Datapacket type M: Aktuel driftsform

M	:	*

Område:
1 ciffer med flg. muligheder: A, B, C, D

F.eks. driftsform C er kodet på følgende måde:

M	:	C

Datapacket type V: Vagtfunktion

V	:	*	*

Område:
2 cifre med flg. muligheder:

' ' ASCII 32D, A, B, C, D

F.eks. Vagtfunktion D er kodet som følger:

V	:		D

NB: Efter kolon (':') sendes, i det viste eksempel, ASCII 32 decimalt = space.

Datapacket type P: Position

P	*	*	*	*

Område:

1. ciffer efter P kan have følgende værdier 0,1,2,3,4,5,6,7.

0 → Station.
1,2,3,4,5,6,7 → Spor nr på strækning.

De næste 3 cifre dækker decimalt området 0 - 999.

De tre cifre angiver strækningskilometrering. (I området 0 - 99,9 km angiver de tre cifre en afstand i hektometer. I området fra 100 km og opefter angiver de tre cifre en afstand i km).

8.3 Telegramtype O: Aktuelle data MSR3 til TC

Telegrammet sendes fra MSR3 til TC ved opdatering af radiodata i MSR3.

Datapacket type F: FC-kanal

F	C	*	*

Område:
2 decimal cifre (0..9)
F.eks. FC-kanal 62 er kodet som følger:

F	C	6	2

Datapacket type D: D-kanal

D	*	*

Område:
2 decimal cifre (0..9)

F.eks. D-kanal 18 er kodet som følger:

D	1	8

Datapacket type C: C-kanal

C	*	*

Område:
2 decimal cifre (0..9)
F.eks. C-kanal 48 er kodet som følger:

C	4	8

Datapacket type M: aktuel driftsform

M	:	*

Område:
1 ciffer med flg. muligheder: A, B, C, D
F.eks. driftsform C er kodet på følgende måde:

M	:	C

Datapacket type V: Vagtfunktion

V	:	*	*

Område:
2 cifre med flg. muligheder: " " (ASCII 32D), A, B, C, D
F.eks. Vagtfunktion D er kodet som følger:

V	:		D

NB. Efter kolon (':') sendes, i det viste eksempel, ASCII 32 decimalt = space.

Datapacket type P: Position

P	*	*	*	*

Område:

1. ciffer efter P kan have følgende værdier 0,1,2,3,4,5,6,7.

0 → Station.
1,2,3,4,5,6,7 → Spor nr. på strækning.

De næste 3 cifre dækker decimalt området 0 - 999.

De tre cifre angiver strækningskilometrering. (I området 0 - 99,9 km angiver de tre cifre en afstand i hektometer. I området fra 100 km og opefter, angiver de tre cifre en afstand i km).

Datapacket type T: Tognummer

T	*	*	*	*	*	*

Område: 6 cifre med hver mulighed for 0 til 9.
F.eks. Tognummer 1,2,3,4,5 og 6 er kodet som følger:

T	1	2	3	4	5	6

8.4 Telegramtype r: Anmodning om specifikke data/Aktuelle data

Telegrammet sendes fra MSR3 til TC, når MSR3 anmoder om aktuelle data fra TC.

Datapacket type B: Anmodning

B	K

Slaven anmodes om at besvare dette Telegram, med den Telegramtype som data er lig med. Aktuelle data = K.

8.5 Telegramtype o (lille o) (6F HEX): Fejlkode MSR3 til TC

Ved fejl i MSR3 radioen sendes et fejlkode Telegram til TC.

Datapacket type S

S	*	*

Område: 00-99, med følgende betydning:
01: Kommunikationsfejl mellem CF1 og CL
02: Kommunikationsfejl mellem CF2 og CL
03: Kommunikationsfejl mellem CF3 og CL
04: Kommunikationsfejl mellem CB1 og CL
05: Kommunikationsfejl mellem CB2 og CL
06: Kommunikationsfejl mellem ATC og CL
07: Resetter CF pga. fejl
08: Nødstop modtaget fra ATC

NB!
Der må maximalt sendes 2 ens fejlkoder pr. gang MSR3 er koblet ind.
Dette er indført for ikke at fylde TC'ens fejllog op.

8.6 Telegramtype t: Tomgang

Anvendes til Linieovervågning (MSR3 til TC). Betinget af svaret på Telegramtype A (se 8.8) kan tidsvagt T0 reduceres fra 20 sek. til 10 sek. så længe MSR3 ikke har modtaget besked om førerbordsnøgle i stilling Drift (Telegramtype A data <>X). Når besked om Førerbord i Drift modtages sættes T0 tilbage på 20 sek. Denne funktion er indført for at reducere forsinkelse på opdatering af tognummer i TC.

8.7 Telegramtype N: Afvist Telegram

Telegramtype N sendes som negativ kvittering ved fejl:

Område: 1 betyder checksumsfejl
2 betyder timingsfejl
A betyder applikationsfejl *)

*) Applikationsfejl (datavalideringsfejl) dvs. et fuldt korrekt dekodet Telegram, der ikke giver nogen mening.

8.8 Telegramtype A: Godkendt Telegram

Ved svar på tomgangstelegrammer fra TC, benyttes datapacket A til at overføre information fra TC til MSR3, angående førerbordsnøglens stilling:

Område: 'A': Oprindeligt indhold, opdater ikke TC.
'X': Førerbord i stilling Drift, opdater TC.
'Y': Førerbord i stilling klar, opdater ikke TC.

8.9 Telegramtype n(lille n): Tognummer til TC.

Telegrammet sendes til TC, når der er indtastet et tognummer uanset om nummmeret er ændret eller ej og uanset om MSR-3 modtager svar fra TC (liniefejl). Telegrammet består af et enkelt datapacket med type T, der angiver det indtastede tognummer. Datapacket type T er defineret i afsnit 8.3. Telegrammet er defineret af hensyn til TRIT, der ikke ønsker at modtage tognummeret hver gang MSR-3 sender et telegram 'O' til TC for opdatering af aktuelle radiodata, men kun når tognummeret indtastes på kontrolboksen.
Telegrammet 'n' sendes efter en eventuel udsendelse af telegram 'O' for opdatering af TC på vanlig vis, dvs indtil der modtages 'Godkendt' eller op til 3 gange ved ingen svar eller 'Afvist'.

Bilag 8.1
Telegramindhold MSR3-TC

Telegramtype	A	N	r	t
Startpad	LF	LF	LF	LF
Bytecounts	Bytecount 1	Bytecount 1	Bytecount 1	Bytecount 1
	Bytecount 2	Bytecount 2	bytecount 2	bytecount 2
Telegramtype	A	N	r	t
	Telegramløbenr.	Telegramløbenr.	Telegramløbenr.	Telegramløbenr.
Packettype	A	N	B	B
Datapacket	data A	data N	data B	data B
Checksum-bytes	Checksum 1	Checksum 1	Checksum 1	Checksum 1
	Checksum 2	Checksum 2	Checksum 2	Checksum 2
Endpad	CR	CR	CR	CR
	Godkendt TC til MSR3	Afvist TC til MSR3	Anmodning om specifikke data MSR3 til TC	Tomgangsdata MSR3 til TC

Telegramtype	K
Startpad	LF
Bytecounts	Bytecount 1
	Bytecount 2
Telegramtype	K
	Telegramløbenr.
Packettype	F
Datapacket	data F
	data F
	data F
Packettype	D
Datapacket	data D
	data D
Packettype	C
Datapacket	data C
	data C
Packettype	M
Datapacket	data M
	data M
Packettype	V
Datapacket	data V
	data V
	data V
Packettype	P
Datapacket	data P
	data P
	data P
	data P
Checksum-bytes	Checksum 1
	Checksum 2
Endpad	CR

Aktuelle data

TC til MSR3

Telegramtype	O
Startpad	LF
Bytecounts	Bytecount 1
	Bytecount 2
Telegramtype	O
	Telegramløbenr.
Packettype	F
Datapacket	data F
	data F
	data F
Packettype	D
Datapacket	data D
	data D
Packettype	C
Datapacket	data C
	data C
Packettype	M
Datapacket	data M
	data M
Packettype	V
Datapacket	data V
	data V
	data V
Packettype	P
Datapacket	data P
	data P
	data P
	data P
Packettype	T
Datapacket	data T
	data T
	data T
	data T
	data T
	data T
Checksum-bytes	Checksum 1
	Checksum 2
Endpad	CR

Aktuelle data

MSR3 til TC

Telegramtype	n
Startpad	LF
Bytecount	Bytecount 1
	Bytecount 2
Telegramtype	n
	Telegramløbenr.
Packettype	T
Datapacket	data T
	data T
	data T
	data T
	data T
	data T
Checksum bytes	Checksum 1
	Checksum 2
Endpad	CR

Tognummer

MSR3 til TC

Telegramtype	o
Startpad	LF
Bytecounts	Bytecount 1
	Bytecount 2
Telegramtype	o
	Telegramløbenr.
Packettype	S
Datapacket	data S
Datapacket	Data S
Checksum-bytes	Checksum 1
	Checksum 2
Endpad	CR

Fejlkode

MSR3 til TC

9.0 TELEGRAMINDHOLD ATC - HLOG

Dette kapitel indeholder en beskrivelse af telegramindholdet mellem ATC og HLOG.

9.1 Telegramindhold ATC og HLOG

Denne forbindelses primære formål er at sikre, at de nødvendige data til at analysere et evt. uheld er mulig. Her tænkes specielt på de sidste km før uheldet. For det præcise indhold af hver Telegramtype henvises til bilag 1 i dette kapitel.

9.2 Telegramtype 2: Togdata

Dette Telegram sendes efter godkendt og kvitteret indtastning af togdata. I køretøjer med TC kan togdata tilbydes fra denne, men for disse togdata kvitteres ligeledes af lokomotivfører, og de sendes derefter umiddelbart videre til HLOG.

Packettype L: Toglængde

L	*	*	*

Data ligger i intervallet fra 30 til 960 m med spring på 10 m.

Packettype %: Bremseprocent

%	*	*	*

Data ligger i intervallet fra 030 til 230 %.

Packettype H: Max hastighed

H	*	*	*

Data ligger i intervallet fra 10 km/t til 200 km/t med spring på 10 km/t.

Packettype R: ATC retning

R	*	*

Data: 8 8 = retning A
0 0 = retning b

Packettype J: Hjuldiameter

J	*	*	*	*

Data ligger i intervallet fra 0 mm til 9999 mm.

9.3 Telegramtype 3: Hastighedsinformation

Hastighedstelegrammet indeholder den af ATC registrerede øjeblikshastighed, samt den overvågningshastighed som ATC overvåger på i sendeøjeblikket.

Denne information tjener kun som supplering til den øjeblikshastighed HLOG selv måler, samt som registrering af overvågningshastigheden på sendetidspunktet.

Telegrammet sendes cyklisk, og virker derved som Linieovervågning. Dette giver et tidsafhængigt mønster, med hvilket man kan danne en evt. bremsekurve.

Packettype K: Øjeblikshastighed

K	*	*	*

Data ligger i intervallet fra 000 til 254 km/t.

Packettype G: Overvågningshastighed

G	*	*	*

Data ligger i intervallet fra 000 til 254 km/t.

9.4 Telegramtype 4: Baliseinformation

Dette Telegram sendes umiddelbart efter passage af en balise.

Packettype a til h: Baliseinformation

h	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

Data er kodet efter den mellem Banestyrelsen og Siemens A/S fastlagte specifikation.

9.5 Telegramtype G: Fejlkode

Der sendes ATC's fejlkode til HLOG umiddelbart efter, at den er detekteret.

Packettype E: Fejlkode

E	*	*	*

Data ligger i intervallet fra 000 til 999.

9.6 Telegramtype H: HKT hastighedsinformation

Jf. HKT-Appendix.

9.7 Telegramtype a: Driftstelegram

Der er to former for driftstelegrammer. Enten sendes Telegramtype a med Packettype I eller med Packettype n. Begge driftstelegrammer indeholder 2 databytes.

Packettype I: Driftsfunktion

Denne Packettype indeholder 2 databytes. Driftstelegram sendes med Packettype I, når en betjeningstast er aktiveret.

I	*	*

Data ligger i intervallet fra 01 til 40 og angiver et positionsnummer for en aktiveret ATC-betjeningstast. Betjeningstasternes positionsnumre:

01	:	---	17	:	LØS ATC
02	:	---	18	:	---
03	:	---	19	:	---
04	:	---	20	:	---
05	:	PASS. STOP	21	:	KVITTERING
06	:	TOGLÆNGDE	22	:	---
07	:	BREMSE %	23	:	---
08	:	MAX HASTIGHED	24	:	---
09	:	ATC RETNING	25	:	---
10	:	DRIFTSBREMSE	26	:	---
11	:	NØDBREMSE	27	:	---
12	:	TEST	28	:	---
13	:	---	29	:	---
14	:	RANGER	30	:	---
15	:	YDRE SIGNALER	31	:	---
16	:	---	40	:	UDFØR

Jf. HKT-appendix for alternative positionsnumre i forbindelse med HKT.

Packettype n: Driftstelegram, DK-/S-ATC

n	*	*

Data ligger i intervallet fra 11 til 44 og angiver ATC-driftsvælgerposition og driftstilstand for dansk ATC og sendes hver gang der sker en ændring i driftsvælgerposition eller en ændring i ATC-tilstand.

Data	Driftvælger	ATC-tilstand
11	Kombineret system (S-ATC prioriteret)	Aktiv og initialiseret
12		Overvåger uden strækningsdata
13		Overvåger med strækningsdata
14		F-tilstand
21	Kombineret system (DK-ATC prioriteret)	Aktiv og initialiseret
22		Overvåger uden strækningsdata
23		Overvåger med strækningsdata
24		F-tilstand
31	S-ATC Eksklusivt	Aktiv og initialiseret
32		Overvåger uden strækningsdata
33		Overvåger med strækningsdata
34		F-tilstand
41	DK-ATC Eksklusivt	Aktiv og initialiseret
42		Overvåger uden strækningsdata
43		Overvåger med strækningsdata
44		F-tilstand

Data: 31, 32, 33, 34 og 44 forventes ikke at ville forekomme.

9.8 Telegramtype c: Test

Dette Telegram tjener til at alle lamper i førerrumssignalet samt indkodningspanel afprøves. Ved opstart af ATC afprøves disse, dog findes der lamper i førerrumssignalet, som styres af HLOG og dels af TC. Når TC og HLOG modtager dette Telegram skal de tænde hhv. slukke lamperne.

Packettype m: Lampetest

m	*

Data: S = Sluk lampe, T = Tænd lampe

9.9 Telegramtype N: Afvist Telegram

N	*

Data:
1 = Checksumsfejl
2 = timingsfejl
A = applikationsfejl (datavalideringsfejl) dvs. et fuldt korrekt dekodet Telegram, der ikke giver nogen mening.
Mellemrum (ASCII 32D) = generelt afvist

9.10 Telegramtype 6: Kvittering med hastighed

Telegrammet bruges kun for positiv kvittering (i lighed med Telegramtype A, "godkendt"), og anvender Packettype "K", som også anvendes til at angive ATC's øjeblikshastighed overfor havariloggen, jf. afsnit 9.3. Selve hastigheden består af tre decimale cifre, hvor *Hastighed 1* er det mest betydende ciffer, og *Hastighed 3* er det mindst betydende ciffer. Dette Telegram erstatter ikke det oprindelige kvitteringstelegram af type "A". ATC skal være i stand til at modtage begge Telegrammer.

9.11 Telegramtype 7: Forbikørsel af stop

Ved forbikørsel af et signal, der viser stop, sendes dette Telegram. Signalpositionen identificeres dels ved et enkelt ciffer, og dels ved et trecifret tal, hvor *Signal Nr. 3* er det mindst betydende ciffer.

Packettype S: Forbikørsel af stop

S	*	*	*	*

Data: Efter S'et sendes fire cifre.
1. ciffer angiver station eller spornummeret. Data ligger i intervallet 0 til 9.
2., 3. og 4. ciffer angiver signalnummeret. Data ligger i intervallet 000 til 999.

1. ciffer	**2., 3. og 4. ciffer**
0 Data angiver en station.	Data angiver nummer på billetsalgssted.
1 til 9 Data angiver spornummeret på en strækning.	Kilometrering ≤ 99,9 km: Data angiver kilometrering i hm. Kilometrering > 99,9 km: Data angiver kilometrering i km.

Bilag 9.1

Telegramindhold ATC-HLOG

Telegramtype	A	c	N
Startpad	LF	LF	LF
Bytecounts	Bytecount 1	Bytecount 1	Bytecount 1
	Bytecount 2	Bytecount 2	Bytecount 2
Telegramtype	A	c	N
	Telegramløbenr.	Telegramløbenr.	Telegramløbenr.
Packettype	A	m	N
Datapacket	data A	data m	data N
Checksum-bytes	Checksum 1	Checksum 1	Checksum 1
	Checksum 2	Checksum 2	Checksum 2
Endpad	CR	CR	CR
	Godkendt ATC til HLOG	Anmodning om test ATC til HLOG	Afvist ATC til HLOG

Telegramtype	a
Startpad	LF
Bytecounts	Bytecount 1
	Bytecount 2
Telegramtype	a
	Telegramløbenr.
Packettype	l
Datapacket	data l
	data l
Checksum-bytes	Checksum 1
	Checksum 2
Endpad	CR
	Driftstelegram ATC til HLOG

Telegramtype	a
Startpad	LF
Bytecounts	Bytecount 1
	Bytecount 2
Telegramtype	a
	Telegramløbenr.
Packettype	n
Datapacket	data n
	data n
Checksum-bytes	Checksum 1
	Checksum 2
Endpad	CR
	Driftstelegram ATC til HLOG

Telegramtype	2
Startpad	LF
Bytecounts	Bytecount 1
	Bytecount 2
Telegramtype	2
	Telegramløbenr.
Packettype	L
Datapacket	data L
	data L
	data L
Packettype	%
Datapacket	data %
	data %
	data %
Packettype	H
Datapacket	data H
	data H
	data H
Packettype	R
Datapacket	data R
	data R
Packettype	J
Datapacket	data J
	data J
	data J
	data J
Checksum-bytes	Checksum 1
	Checksum 2
Endpad	CR

Togdata

ATC til HLOG

Telegramtype	4
Startpad	LF
Bytecounts	Bytecount 1
	Bytecount 2
Telegramtype	4
	Telegramløbenr.
Packettype	a-h
Datapacket	data a-h
	data a-h
24 bytes	data a-h
	data a-h
Checksum-bytes	Checksum 1
	Checksum 2
Endpad	CR

Balise information

ATC til HLOG

Telegramtype	3
Startpad	LF
Bytecounts	Bytecount 1
	Bytecount 2
Telegramtype	3
	Telegramløbenr.
Packettype	K
Datapacket	data K
	data K
	data K
Packettype	G
Datapacket	data G
	data G
	data G
Checksum-bytes	Checksum 1
	Checksum 2
Endpad	CR

Hastigheds-information
ATC til HLOG

Telegramtype	G	6
Startpad	LF	LF
Bytecounts	Bytecount 1	Bytecount 1
	Bytecount 2	Bytecount 2
Telegramtype	G	6
	Telegramløbenr.	Telegramløbenr.
Packettype	E	K
Datapacket	data E	data K
	data E	data K
	data E	data K
Checksum-bytes	Checksum 1	Checksum 1
	Checksum 2	Checksum 2
Endpad	CR	CR

Fejlkode

ATC til HLOG

Kvittering med hastighed
HLOG til ATC

Telegramtype	7
Startpad	LF
Bytecounts	Bytecount 1
	Bytecount 2
Telegramtype	7
	Telegramløbenr.
Packettype	S
Datapacket	data S
	data S
	data S
	data S
Checksum-bytes	Checksum 1
	Checksum 2
Endpad	CR

Forbikørsel af stop

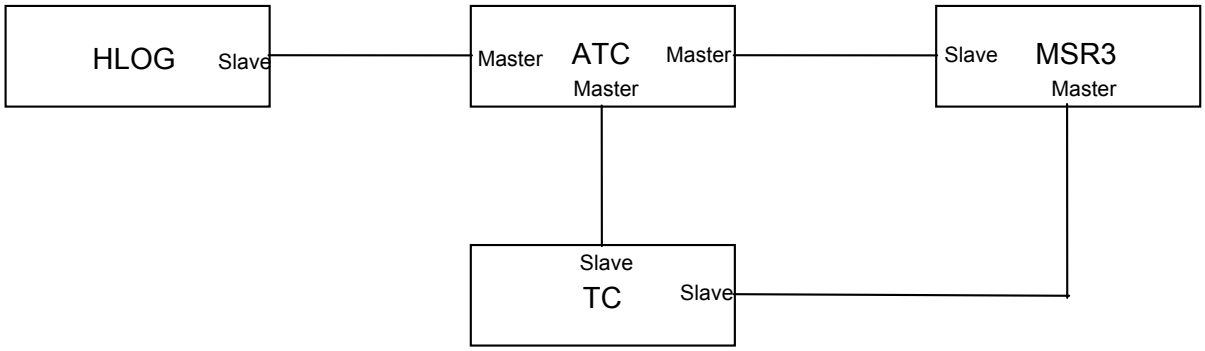
ATC til HLOG

10.0 KOMMUNIKATIONS HARDWARE

Dette kapitel indeholder blokdiagrammer over 20 mA strømsløjfe og en beskrivelse af kommunikationshardwaren mellem ATC - TC - MSR3 - HLOG og MSR3 - TC.

10.1 Blokdiagram over Master - Slave

Blokdiagram over Master/Slave forhold mellem ATC, TC, MSR3, HLOG og MSR3 - TC:



10.2 Beskrivelse af kommunikationshardware mellem ATC - TC - MSR3 - HLOG og MSR3 - TC.

Kommunikationen realiseres via en 20 mA strømsløjfe med følgende karakteristiske data:

1 startbit
8 databit
1 paritetsbit, ODD paritet
1 stopbit

Dette udgør 1 karakter:

start	D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	paritet	stop 1

Kommunikationshastighed: 1200 baud.
Max. afvigelse: ± 3 % pr. bit.

Signalniveau: Det skal bemærkes, at 0 og 20 mA er de tilstræbte nominelle værdier.

Sender aktiv low - 0 $\Rightarrow$ 0 - 2 mA
high - 1 $\Rightarrow$ 18 - 24 mA

Alt der er imellem 2 mA til 18 mA regnes for "ulovligt" niveau.

Modtager low - 0 ⇒ 0 - 6 mA
high - 1 ⇒ 16 - 24 mA

Alt der er imellem 6 mA til 16 mA regnes for "ulovligt" niveau.

"MARK" betragtes som logisk 1
"SPACE" betragtes som logisk 0

Impedans: TxD kan som "aktiv" generere 20 mA over en extern belastning på minimum: 600 ohm.

Slave må højst belaste Master generator med et spændingsfald på 5 Vdc ved 20 mA: 230 ohm i indre impedans.

Forsyningsspænding:
Maximum Vcc til 20 mA strømgeneratoren er: 30 Vdc. Målt over RxD+ og RxD-. Forsyningsspænding til strømgenerator er ikke fastlagt som potentialfri, det er dog den enkelte leverandørs eget ansvar at sikre sin evt. kobling så det funktionsmæssigt kan modstå eventuelt udefra kommende forstyrrelser.

Aktiv/passiv:
ATC regnes som AKTIV overfor MSR3, HLOG og TC, som alle regnes for PASSIVE. TC regnes som AKTIV overfor MSR3, som regnes for PASSIV overfor TC.

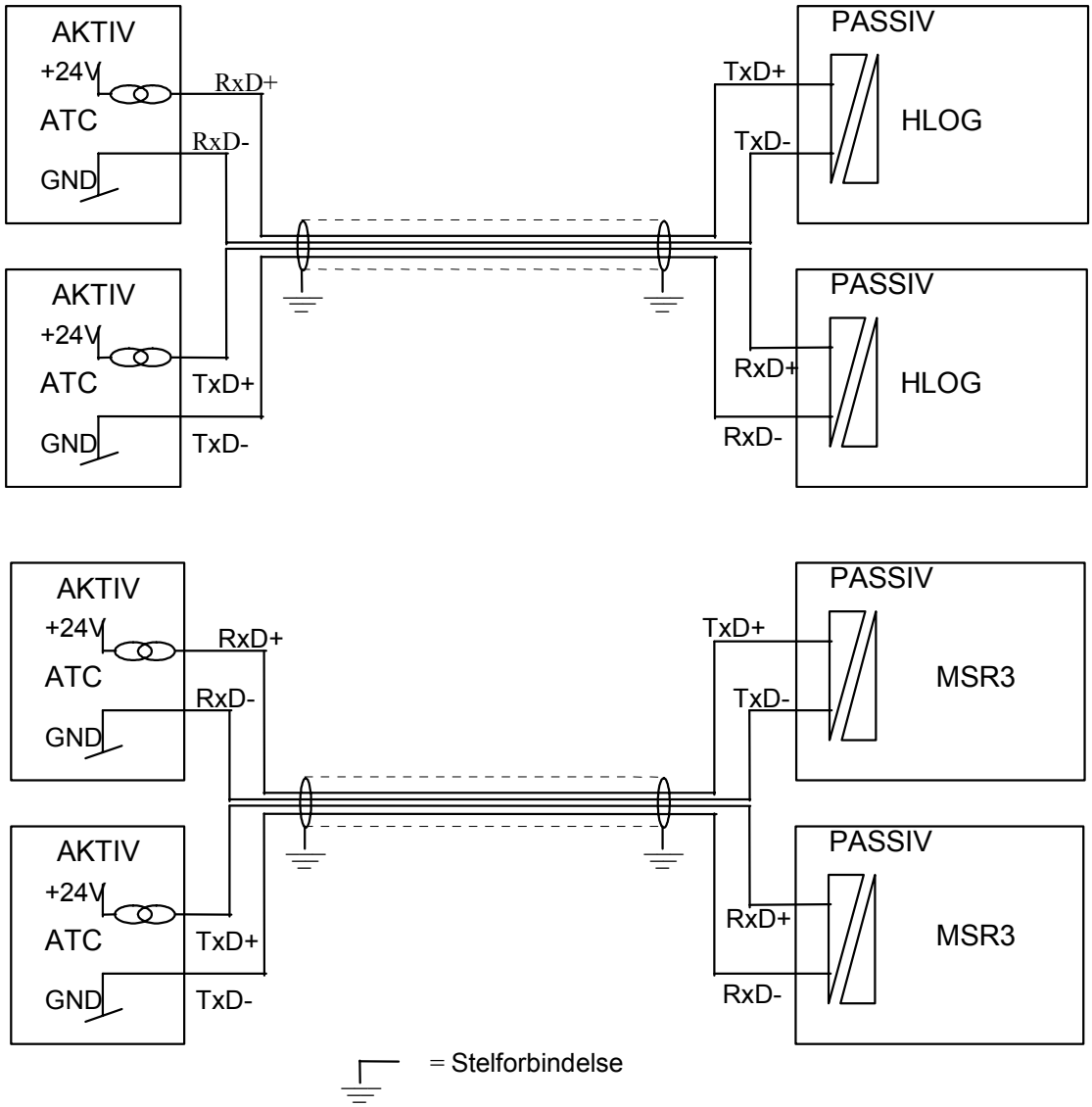
Forbindelse:
Den anvendte strømsløjfe er en punkt til punkt forbindelse, der alene er koblet imellem 2 enheder, jf. afsnit 10.3.

Forbindelsen imellem AKTIV og PASSIV er en firetråds parsnoet kabelforbindelse med skærm.
AKTIV er forbundet med PASSIV som beskrevet i skemaerne: "Blokdiagram over 20 mA strømsløjfe forbindelser". 2 sider.

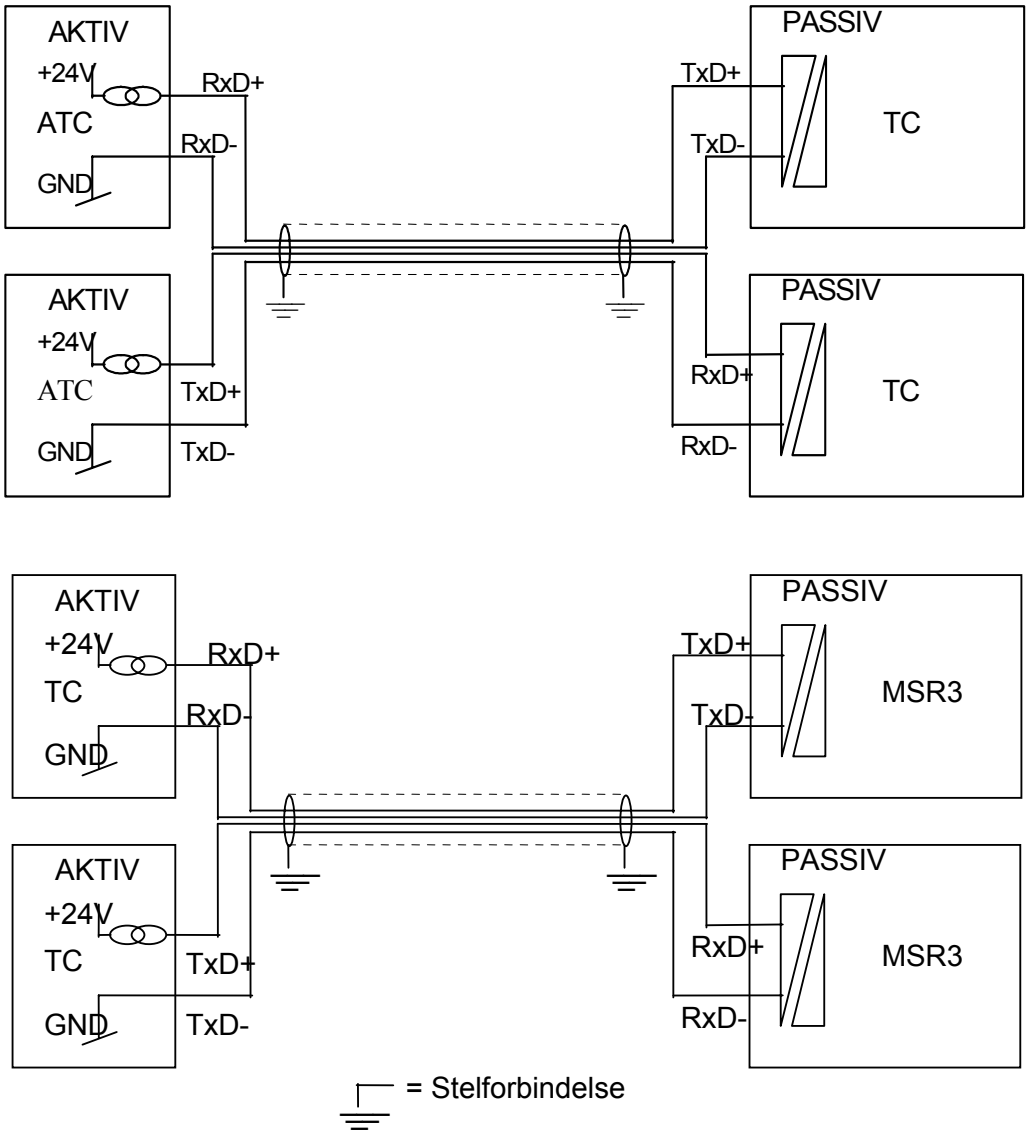
Det forventes, at pin-out benævnelse fra "Blokdiagram over 20 mA strømsløjfe forbindelser" benyttes i den dokumentation, f.eks. brugervejledning, el-diagram osv., der udleveres til BS.

10.3 Blokdiagram over 20 mA strømsløjfeforbindelser

Blokdiagram over 20 mA strømsløjfeforbindelser



Blokdiagram over 20 mA strømsløjfeforbindelser



11.0 ÆNDRINGSAFTALE TIL PROTOKOL

Følgende formular skal benyttes ved behov for ændringer.

<table>
<tr><td colspan="2">Ændringsaftale til protokol</td><td>Ændring nr:</td></tr>
<tr><td rowspan="3">Udfyldes af forslagsstiller</td><td>Ændringens art:</td><td>Medfølgende bilag:</td></tr>
<tr><td>Hvem har fremlagt ændringsforslag:</td><td>Ansvarlig hos fremlægger:</td></tr>
<tr><td colspan="2">Hvilke anlæg/parter berøres af ændringen:
☐ ATC ☐ TC ☐ HLOG ☐ MSR3 ☐ BS' brugere ☐ Operatørers personale</td></tr>
<tr><td>Leverandørgodkendelse</td><td colspan="2">Ændringen godkendt af følgende leverandører: (Stempel, signatur, dato)
ATC: TC: HLOG:
MSR3: Anden leverandør: Anden leverandør:</td></tr>
<tr><td>BS-godkendelse</td><td colspan="2">Ændringen godkendt af Banestyrelsen: (Navn, signatur, dato)
Afdeling: _____
Afdeling: _____
Ændring indført i protokollen: _____</td></tr>
</table>

12.0 TELEGRAMINDHOLD HKT - HLOG (HKT-appendix.)

Dette appendix indeholder en beskrivelse af afvigelser fra protokollens 11 kapitler i forbindelse med anvendelse i S-tog.

Telegramindhold HKT - HLOG
Der anføres i dette appendix afvigelser fra kap. 1 til kap. 11 i forbindelse med seriel kommunikation mellem HKT-anlæg og HLOG. HKT henviser i dette appendix til LZB804/HKT.

12.1 Ad.: 9.6 Telegramtype H: HKT hastighedsinformation

Dette telegram skal alene opfyldes for leverancer til S-tog.
Efter hver ændring af togets HKT hastighedsinformation sendes telegramtype H, packettype n til HLOG.

Packettype n: Hastighedsinformation | n | * |

Data kan antage følgende 16 forskellige værdier, som angiver de tilhørende HKT-hastighedsinformationer:

data n	HKT hastighedsinformation
1	STOP VANDRET
2	80
3	60
4	50
5	Y
6	LA 70
7	100
8	70
9	120
A	LA 50
B	90
C	40
D	LA 30
E	30
F	STOP FALD
0	TRANSMISSIONSUDFALD

12.2 Ad.: 9.7 Telegramtype a: Driftstelegram

Packet type I, som angiver positionsnumre for betjeningstaster er i forbindelse med HKT benyttet på følgende vis:

1. "06, TOGLÆNGDE", "07, BREMSE%", "08, MAX HASTIGHED", og "09, ATC RETNING" er værdier som ikke indtastes i forbindelse med HKT, da der her altid benyttes definerede default-værdier.

2. "17 LØS ATC" hedder i forbindelse med HKT "LØS".

Bilag 12.1

Telegramindhold ATC-HLOG

Telegramtype	**H**
Startpad	LF
Bytecounts	Bytecount 1
	Bytecount 2
Telegramtype	H
	Telegramløbenr.
Packettype	n
Datapacket	data n
Checksum-bytes	Checksum 1
	Checksum 2
Endpad	CR

HKT til HLOG

Protocol
for Serial Communication
between ATC, TC, MSR3 and Event Recorder

banestyrelsen

		Verified		Adress Banestyrelsen Pakhusvej 10 2100 København Ø	Design
		Replaces 01.01			
		Approved by Banestyrelsen			
Drawing:	1. Edition Dato og initialer	Latest edition Date and initials	Measure	Drawing name Protocol for seriel communication between ATC, TC, MSR3 and Eventrecorder	
Made by	04.02.2002 jhm	08.07.2003 jhm	Unit		
Verification	04.02.2002 hml				
Approved	04.02.2002 nfn				
	Language /Eng	Edition **01.02 08.07.2003**		Drawing no. IN 656 V 1711	Page/of pages 1/66

Index:

Amendments Recordings:

Version	Amendment
01.00	Conveyed from Consulting (document version Nov. 3rd, 1998) to Banestyrelsen. Added absent format specification for telegram type o(small letter o) in Appendix 8.1.
01.01	Sect. 7: Poll-time (T0) changed. Valid as from ATC SW02. Sect. 8 revised: Fault telegrams and driver's desk key telegrams. Sect. 10: TC not active towards ATC.
01.02	DSB (TRIT) amendment MSR3-TC

1.0 TERMS AND ABBREVIATIONS

This section lists a definition of terms and abbreviations. In the entire protocol the terms used are written with initial capitals to refer to the terms defined. Abbreviations are written either with initial capitals or capital letters.

Through the entire specification various terms and abbreviations have been used., and they are described as follows:

ATC:	Automatic Train Control (the mobile ATC-installation).
Byte Count:	The number of bytes except Start Pad and End Pad in a Telegram. In other words the number of bytes in a Telegram less 2.2 ASCII characters stating a number from 1 to 64. 2 bytes.
Checksum:	2's complement of module 100 Hex. Calculated from Byte Count + Telegram Type + Data Packets and printed as ASCII. Will always be a number between 0 and 9 or an ASCII letter between A and F (30H - 39H and 41H - 46H). The checksum is calculated in Hex and is converted into ASCII as shown in the example in "Data Packets" 2 bytes, cf. description.
Data Field:	The field where Packet Types and Data Packets are located. To be initiated after Telegram Sequence Number and terminated by Checksum.
Datapacket:	The bytes constituting one type of train data, cf. the table "The limited ASCII set of characters for Data Packets", section 5.3.1. The hexadecimal codes which are to be converted into ASCII have been arranged so that e.g. 8BH is converted into 8 and B (38H and 42H). This to make it easier to decipher the communication in case of a possible printer supervision. Variable number of bytes.
DS 2089:	Abbreviation of Dansk Standard No. 2089, ISO 7 Bit code, set of signs.
Endpad:	Here the Telegram ends. Will always be an ASCII "CR" (0DH). 1 byte.
Approved:	An approved Telegram is a Telegram where Start Pad, Byte Count, Checksum, End Pad and parity check are correct on receipt.
Hit:	Is a designation of an disturbance of the electrical level in a telegram, so that the power levels laid down in section10.2 are exceeded. The three asterisks in figures should visualize line disturbance.

HLOG:	Event Recorder (registration equipment registering pace data etc. during travel).
Line Faults:	Any other situation than the Master- and Slave situations approved in this specification. Master and Slave can both register internally Line Faults.
Line Monitoring:	The function that registers Line Faults by means of the expiry of a time watch.
Line Situation:	A situation in which the serial line is depending on the situation in Master and in Slave. The Line Situation reflects directly what to be sent on the line.
Master:	A title conferred on one unit on a serial line. This unit is in sovereign control of all communication on the line.
MSR3:	Mobile Train Radio, 3rd generation.
Packet Type:	States the data in question. Is upper- and lower-case. ASCII letters, (41H -5AH and 61H - 7AH) and ASCII %, (25H). 1 byte.
Retrans:	States a retransmission of a Telegram which has been sent earlier.
Slave:	A title conferred on the other unit in a serial line. This unit is totally subject to the sovereignty of Master. Consequently Slave is not supposed to send anything in the line without the permission of Master.
Start Pad:	First data in a Telegram. Starts receipt of telegram. Is always an ASCII "LF" (OAH). 1 byte.
TC:	Train Computer.
Telegram:	The complete data sequence.
Telegram Serial No.:	States the number of an unanswered Telegram viewed from Master. Is always ASCII 1 or 0. 1 byte.
Telegram Type:	A classification of Telegrams acting homogeneously in the serial line. Each Telegram Type represents a definite function.
Trans:	Means start of a transmission sequence which may contain retransmissions. Trans is always a new Telegram.
Exchange Procedures:	A set of rules for exchange of Telegram Types in the serial line.

2.0 GENERAL DESCRIPTION

This section comprises a description of the transmission system. The description is a summary of how to understand or construe the system.

Aberrant specifications concerning S-trains are stated in section 12. HKT appendix.

2.1 Principle for the Master- og Slave concept

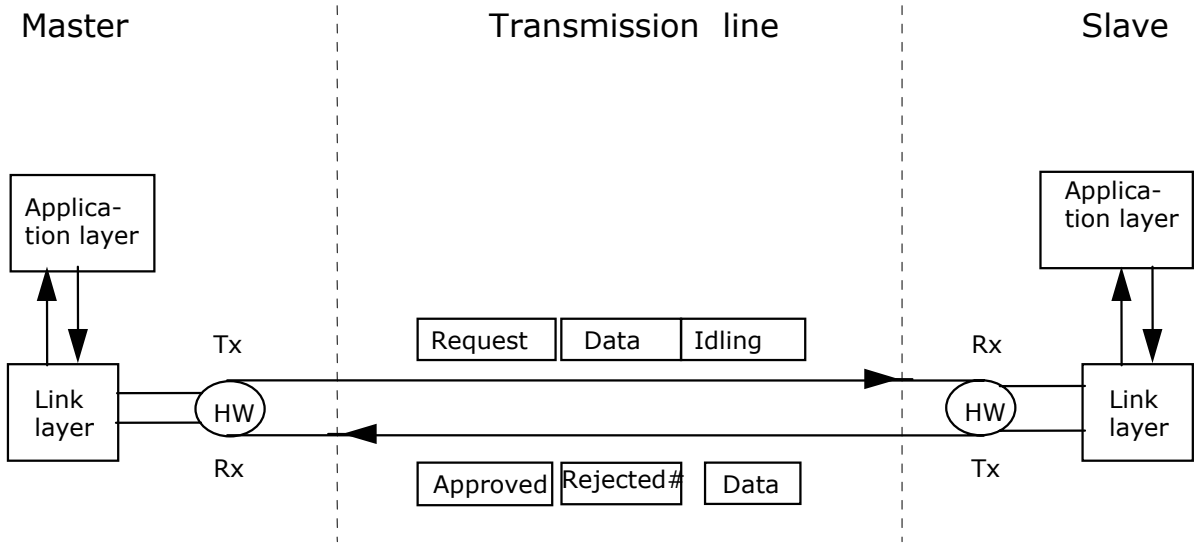


Fig. 2.1 Fundamental drawing of the Master- and Slave concept.

2.1.1 Master Characteristics

Master controls all communication.

The idling telegram has the lowest priority.

The control is divided into:
A. Request for data.
B. Transmission of data.
C. Demand for unsolicited data.

Master should not start a new transmission of information till the previous transmission has been terminated .
Transmission of a new Telegram should not be made till Slave has responded. .
Master has “receive” open when and only when T2 operates.

2.1.2 Slave Characteristics

Slave should transmit a Telegram only on demand from Master.

Slave should consider receipt of a request telegram or an idling telegram a demand.
A demand should be answered only by a telegraphic reply.

2.2 Logical structure of a transmission system

The transmission system consists of the following procedures:

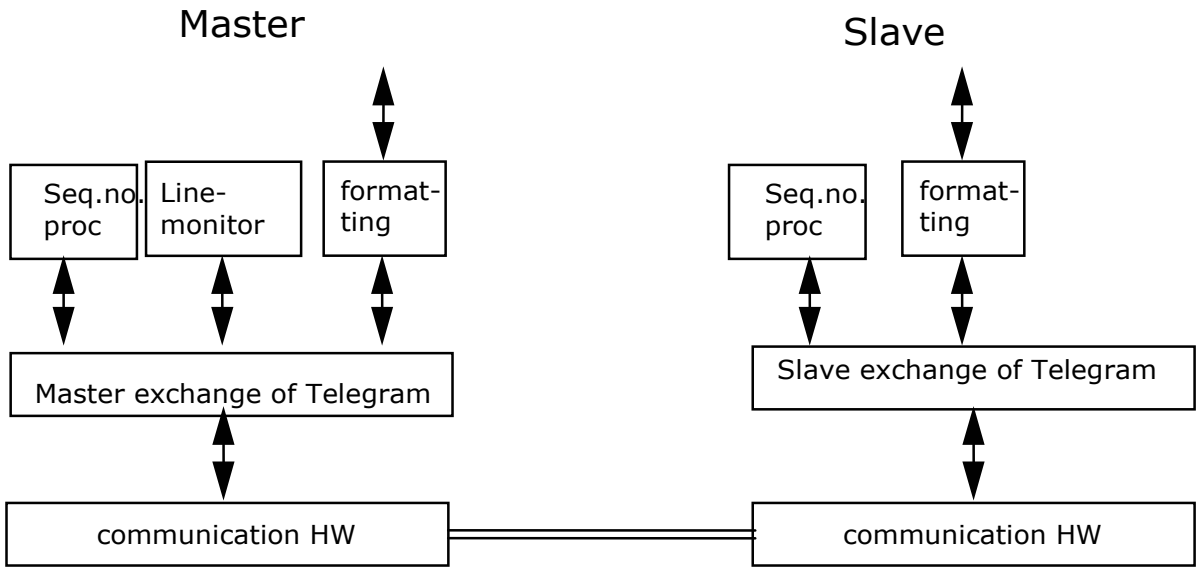


Fig. 2.2 Structure of transmission system.

Fig. 2.2 shows graphically the procedures constituting the transmission system:
The bottom layer, communication HW, consists of hardware components e.g. UART(universal asynchronous receiver transmitter) and carries out control and transmission of the individual character and secures that it complies with the characteristic data in the individual character cf. section 10.2.
The middle layer is software programmel securing transmission of the individual telegram.
The top layer is programmel monitoring the correct Sequence Number of the telegram and by means of timer functions 1, 2 and 3 (which could be SW or HW prepared) monitors the correct progress of the transmission.
The top layer conveys the connection to the application layer of the equipment.

3.0 DEFINITIONS OF TRANSMISSION

This section comprises definitions of the various elements laid down for this very protocol. These definitions have been divided into:

- Definition of Telegram Types
- Definition of timers
- Definition of situations
- Definition of possibilities of faults

3.1 Definition of Telegram Types

Among the units in a serial line a number of telegram numbers are defined. These are categorised in 5 telegram types each being handled in a homogeneous way in

the exchange procedures irrespective of the contents or the function of the telegram number.

De five Telegram Types are:
1. Data.
2. Request
3. Idling.
4. Approved.
5. Rejected.

3.1.1 Telegram Type *Data*

Function: Is used to send information between the units in the serial line. *Data* may be sent uninvited or as an answer to *Request* or *Idling*.

Sender: Master.

Initiator: Uninvited.

Answer: *Approved, Rejected*(#)

Sender: Slave.

Initiator: *Request, Idling*

Answer: -

Comment: This Telegram Type may be sent from Master as well as from Slave.

From Master *data* are sent uninvited.
From Slave *Data* are never sent uninvited but always as an answer to a *Request* or *Idling* sent from Master.

This difference between *Data* from Master and *Data* from Slave means that *Data* from Master always has a higher priority than *Data* from Slave.

3.1.2 Telegram Type *Request*

Function: Is used as an inquiry for specific information, i.e. a telegram number known in advance. Only one telegram number can be inquired for at a time.

Sender: Master.

Initiator: Uninvited.

Answer: *Data*, *Rejected* (#).

Comment: *Request* is defined only to be sent from Master. As an answer the Telegram Type desired may be sent in the form of *Data*. Instead *Rejected* may be sent with a fault code indicating the cause of the rejection of *Request.*

3.1.3 Telegram Type *Idling*

Function: Is used by Master to hand over the right of transmission to Slave, giving Slave the possibility to send.

Sender: Master.

Initiator: Uninvited.

Answer: *Data*, *Approved, Rejected* (#).

Comment: The answer from Slave is *Data* , if Slave has uninvited *Data* to be sent to Master. If Slave has nothing for Master the answer is *Approved* Telegram. In fault situations the answer is *Rejected* with a fault code indicating why the *Idling* Telegram has been rejected.

3.1.4 Telegram Type *Approved*

Function: Is used by Slave as an answer to a Telegram received faultless from Master and handled correctly by Slave

Sender: Slave.

Initiator: *Data, Idling*.

Answer: -

Comment: The Telegram Type *Approved* is always sent only from Slave to Master, which hereby has a receipt for a Telegram sent. *Approved* will terminate an exchange of telegrams.

3.1.5 Telegram Type *Rejected*

Function: Is used by Slave as an answer to a Telegram received with faults from Master. A fault code indicating why the telegram received has been rejected will be enclosed.
It might be a question of a transmission fault or a fault in the handling of the contents of the telegram in the Slave unit.

Sender: Slave.

Initiator: *Data*, *Request, Idling*.

Answer: -

Comment: The Telegram Type *Rejected* is always sent only from Slave to Master which hereby has a receipt for a Telegram sent.

In case a *Rejected* Telegram is received in Master, Master may choose to retransmit or treat the fault occurred in a different way.
Thus *Rejected* does not terminate a telegram sequence.

3.2 Definition of timers

Three timers/timer watches have been defined in the system. These are used to monitor various parameters. The timers are:

3.2.1 Timer No. 0

Definition of timer functions in Master and Slave

Timer Type: Timer No. 0 (abbrev. T0).

Function: Line monitoring timer in Master securing transmission in the line.

Specification:

1. T0 is a time watch in Master SW or HW.
2. T0 has a time of expiry of 20 sec. ± 1 sec.
3. T0 will start when the incoming Vcc to Master and the self testing program have been carried out or after a manual reset.
4. T0 will restart when Master has sent End Pad.
5. At the expiry of TO Master shall send a Telegram of the type; *Idling* to Slave.

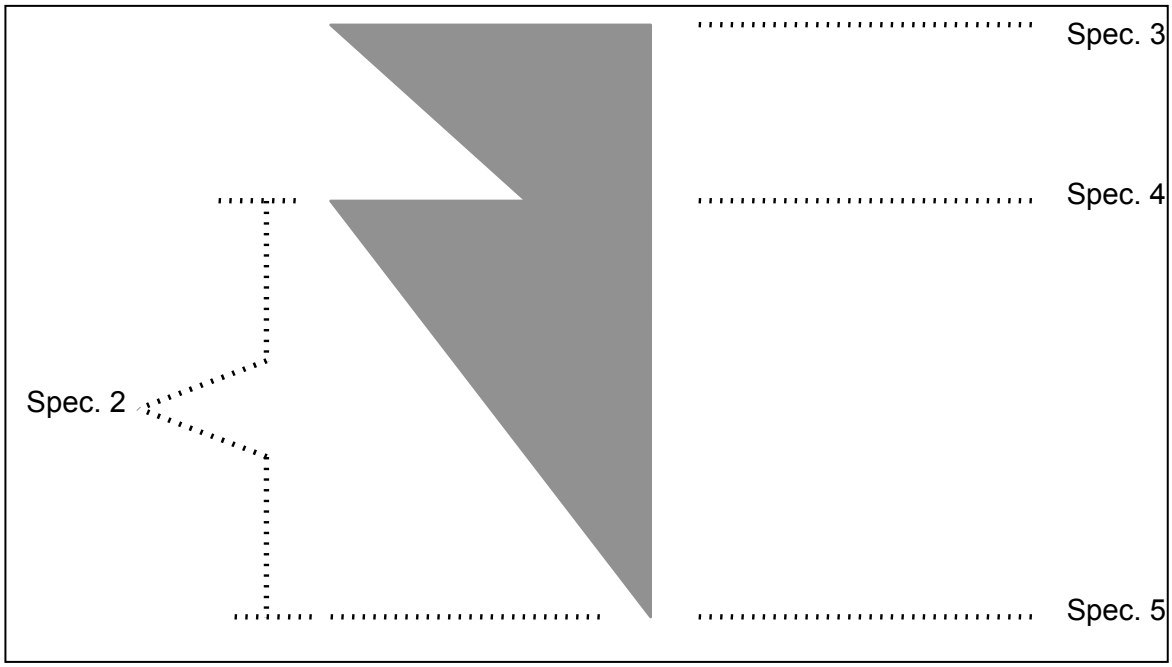


Fig. 3.1 Timer Diagram for T0.

3.2.2 Timer No. 1

Definition of timer functions in Master and Slave

Timer Type: Timer No. (abbrev. T1).

Function: Line monitoring timer in Slave controlling transmission in the line.

Specification:

1. T1 has a time of expiry of 28 sec. ± 1 sec.
2. T1 is a time watch in Slave SW or HW.
3. T1 starts on correct receipt of LF, CR, Checksum and no transmission faults. Slave should register that the communication to Master is correct.
4. T1 restarts on correct receipt of LF, CR, Checksum and no transmission faults.
5. At the expiry of T1 Slave should register internally that the communication from Master is discontinued and wait for specification No. 3.

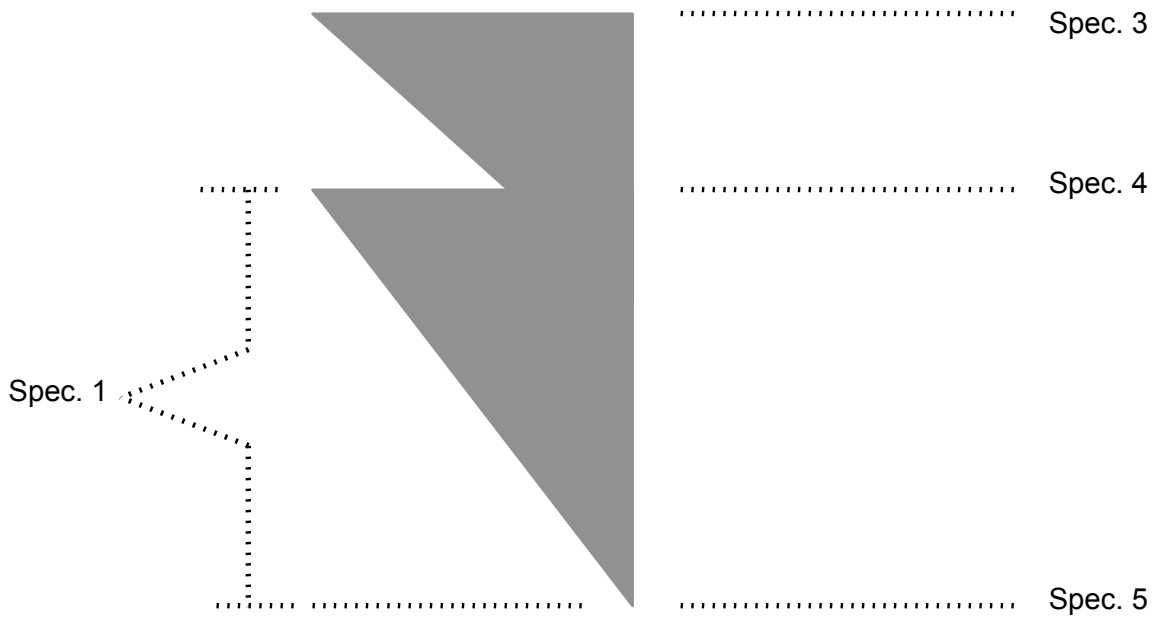


Fig. 3.2 Timer Diagram for T1.

3.2.3 Timer No. 2

Definition of timer functions in Master and Slave

Timer Type: Timer No. 2 (abbrev. T2).

Function: Retransmission timer in Master.

Specification:

1. T2 is a time watch in Master SW or HW.
2. T2 has a time of expiry of 2 sec. ± 20 ms.
3. T2 starts when CR has been sent in Telegram from Master.
4. T2 stops and zeroes on receipt of LF, CR, Checksum and no transmission faults.
5. T2 cannot be retriggered but only stop and zero.
6. At the expiry of T2 Master should register whether it is expiry Nos. 1, 2 or 3 in a telegram transmission.

7. At T2 expiry No. 1 or 2 Master should register and carry out retransmission of Telegram.
8. At T2 expiry No. 3 Master should register that communication from Slave is terminated.

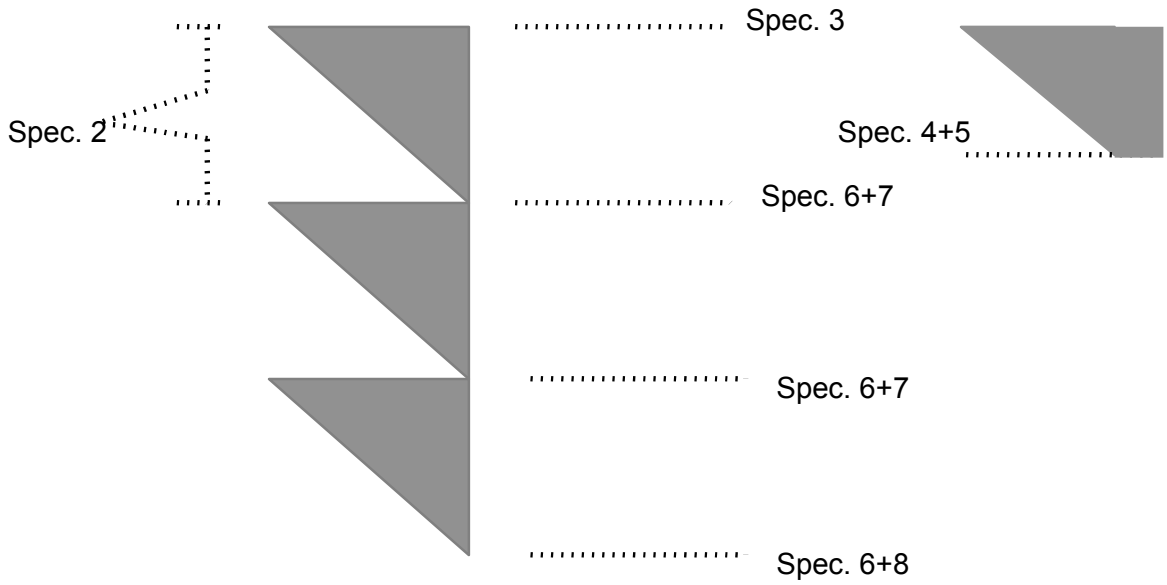


Fig. 3.3 Timer Diagram for T2.

3.3 Definition of situations

The transmission system consists of:

1. Master
2. Slave
3. Line

These three system components may all exist in independent situations dependent on outside influence. In the following is described the situations defined for the individual components, and how they are influenced.

3.3.1 Master situations

Four situations on the master side are possible. These situations are:

1. M-Data: Master has data for the Slave.
2. M-Request: Master requires data from the Slave.
3. M-Idling: Master neither has data for the Slave nor requests data from the Slave.
4. M-Faults: Master has detected Line Faults i.e. no answers to a Trans + 2 Retrans.

3.3.2 Slave situations

Four possibilities on the slave side are possible. These situations are:

1. S-Data: The slave has data for Master.
2. S-Request: The slave requires data from Master.

3. S-Idling: The slave has neither data for Master nor requires data from Master.
4. S-Fault: The slave has detected Line Faults i.e. the line time watch T1 has expired.

Note: S-Request is not relevant in the present specification but is a possible situation if so required.

3.3.3 Line situations

When Master- Slave situations are combined there is a possibility for the line situations which appear from the following table :

Master / Slave	M-Data	M-Request	M-Idling	M-Faults
S-Data	L-Master	L-Master	L-Slave	L-Faults
S-Request	L-Master	L-Master	L-Slave	L-Faults
S-Idling	L-Master	L-Master	L-Idling	L-Faults
S-Faults	-	-	-	L-Faults

Fig. 3.4 Possible line situations.

Cf. The above figure the line situation is a function of the Master situation and the Slave situation. This context gives the following line situations:

1. L-Master: In this line situation Master will send in the line either M-Request or M-Data Telegrams. Slave cannot send S-Data or S-Request. This means that Master has the highest priority in the line.
2. L-Slave: In this line situation Master is idling, and it is now possible for Slave to send.
3. L-Idling: Neither Slave nor Master has information to send meaning that the line is idling.
4. L-Faults: A line fault has occurred in Master and possibly in Slave meaning that the line reports fault.

3.4 Definition of fault possibilities

To facilitate data transmission in the fault situation where the Master TxD is not in connection with the Slave RxD, Master should continue the data transmission, also after detection of the Line Fault.

Continuation of the data transmission should only cause that the Slave rejects max. one correctly received Telegram. Cf. Fig. 3.5 below.

1. T1 not expired
2. Line again intact after exactly three trans.

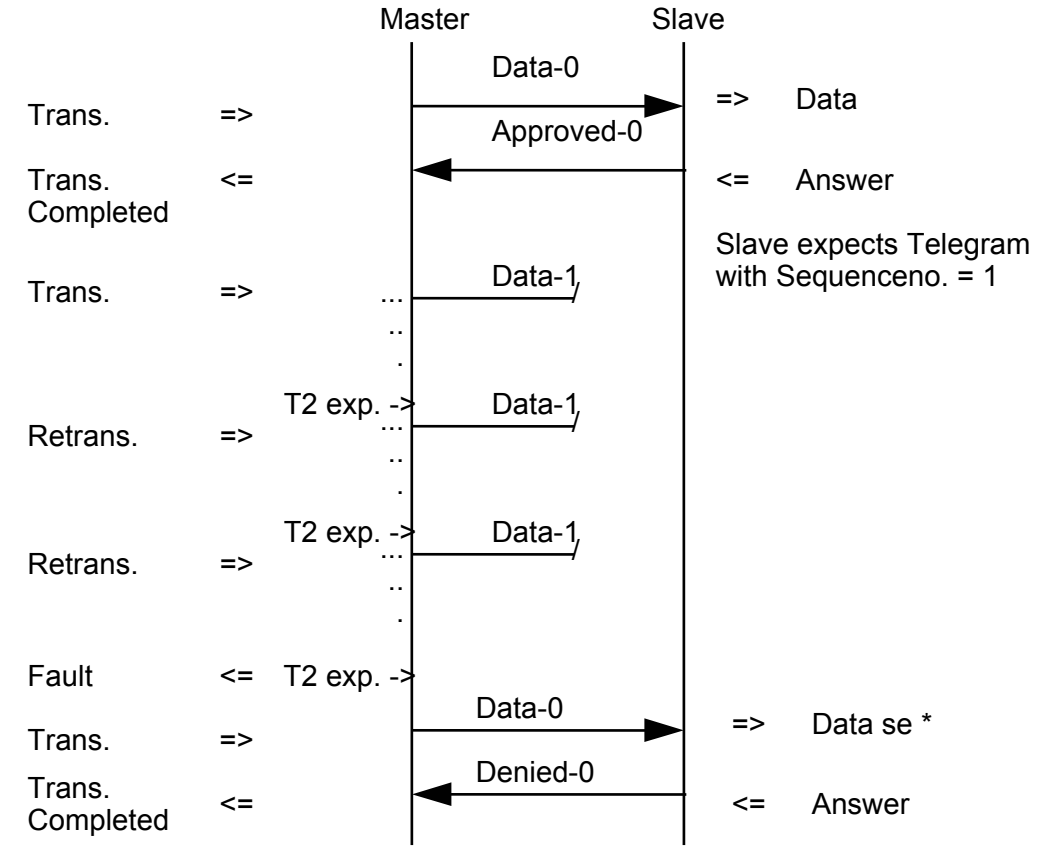


Fig. 3.5 Rejection of correctly received Telegram.

* It is assumed that this fault situation, where the above two points have been fulfilled is sufficiently improbable. At any rate the Slave can ignore one correctly received Telegram only.

4.0 TRANSMISSION PROCEDURES

This section contains a description of the procedures specified in the transmission system. The procedures comprise the use of the “components” defined in section 3.. The following procedures are defined in the transmission system:

- Exchange of Telegram Types.
- Sequence Numbering.
- Line Monitoring.

4.1 Exchange of Telegram Types

This procedure is central in the transmission system as it demonstrates how the various Telegram Types are handled in different situations. The procedure as described here handles all “normal” situations, i.e. situations where the line is in one of the following situations:

- L-Master
- L-Slave
- L-Idling

The above line situations are “normal” situations where the line is not faulty, i.e. faults having been reported by Master alone or by both Master and Slave.

The following nomenclature has been used:

Telegram-0: Means that a Telegram is sent with serial No. 0.

T2
.
.
.
* : Means that timer T2 is operating. At * the timer expires.

T2
.
.
.
- : Means that timer T2 has stopped before expiry.

T2
.
.
.
-
T2: Restart of timer T2 before expiry.
.
.
.
.

T2
.
.
.
*
T2: Restart of timer T2 after expiry.
.
.
.
.

***
hit Transmission Fault.

4.1.1 Line situation L-Master

This section comprises the line situation L-Master, where Master may be either in the M-Data or in the M-Request situation. The figures deal with the M-Data situation only, i.e. where Master sends Data to Slave. M-Request should be dealt with the same way..

4.1.1.1 Normal cases

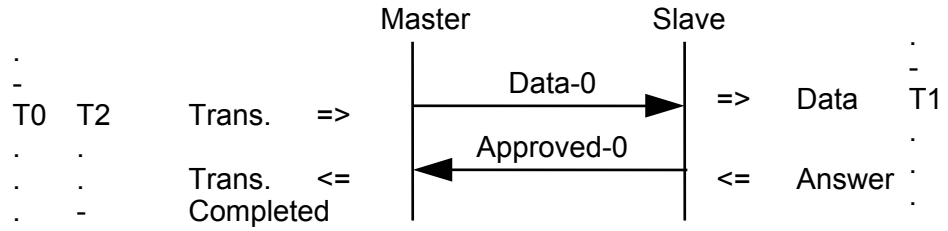


Fig. 4.1 L-Master normal from Master

4.1.1.2 Transmission Faults from Master

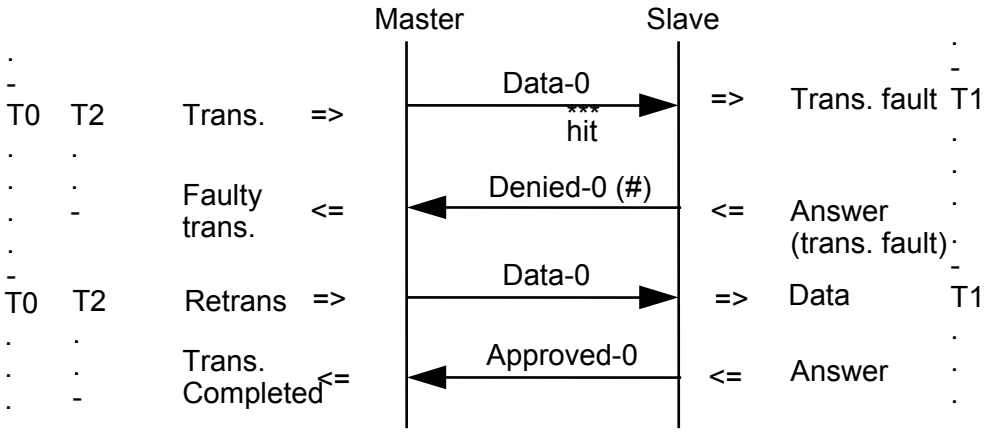


Fig. 4.2 L-Master transmission faults from Master

4.1.1.3 Transmissionsfejl fra Slave

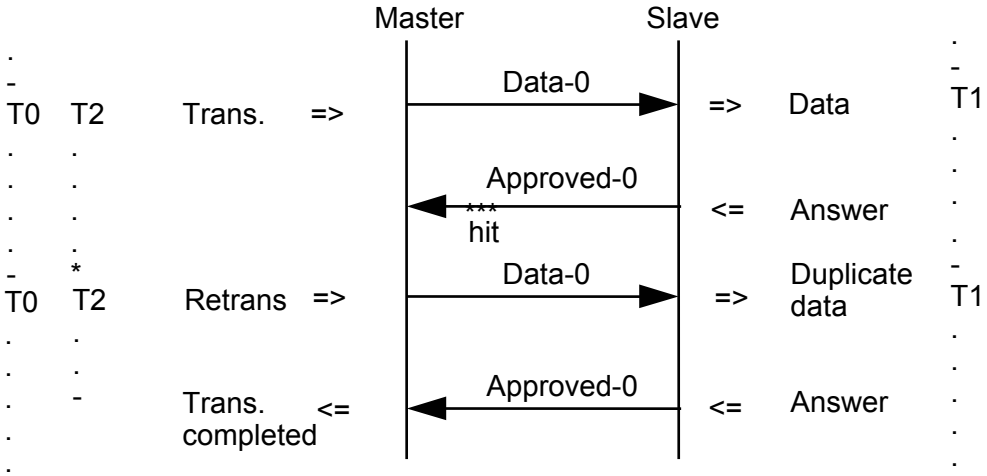


Fig. 4.3 L-Master transmissionsfejl fra Slave

4.1.1.4 Transmission disconnected to Slave

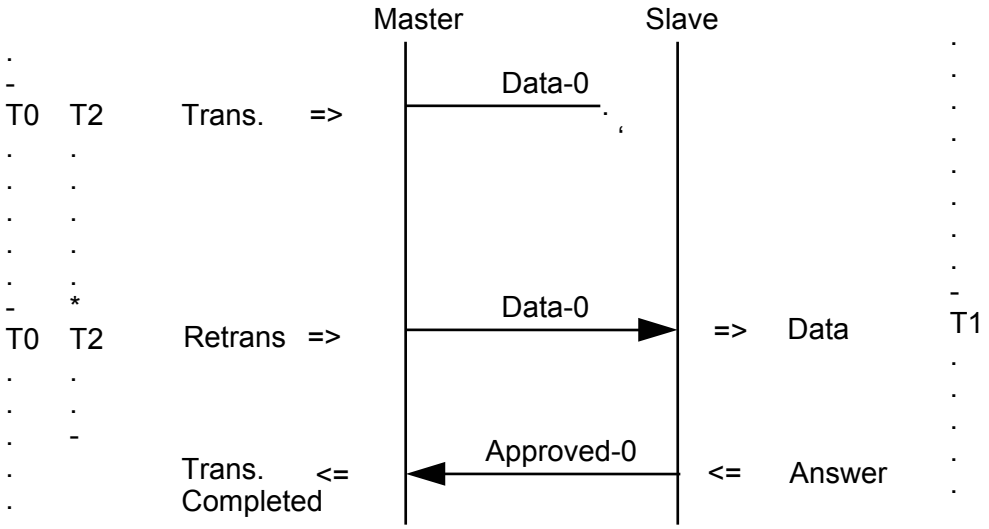


Fig. 4.4 L-Master transmission disconnected to Slave

4.1.1.5 Transmission disconnected to Master

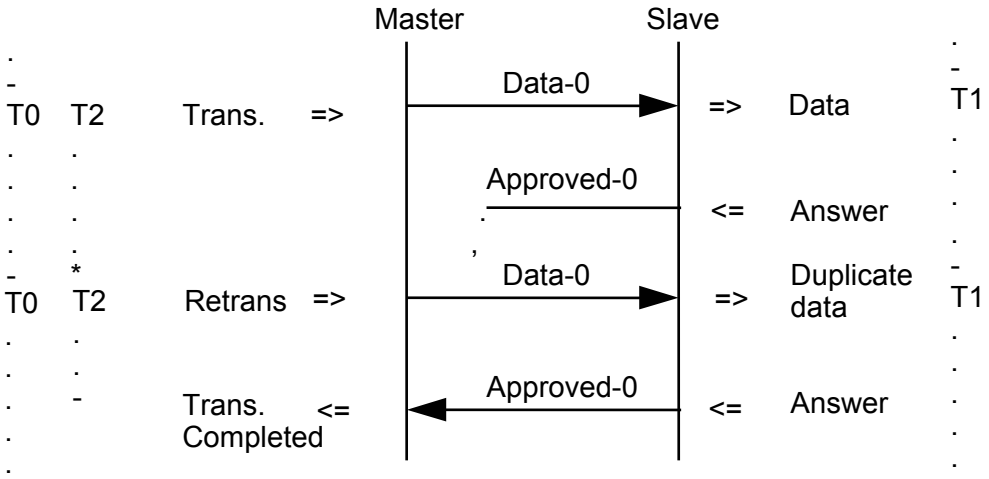


Fig. 4.5 L-Master transmission disconnected to Master

4.1.1.6 No transmission to Slave

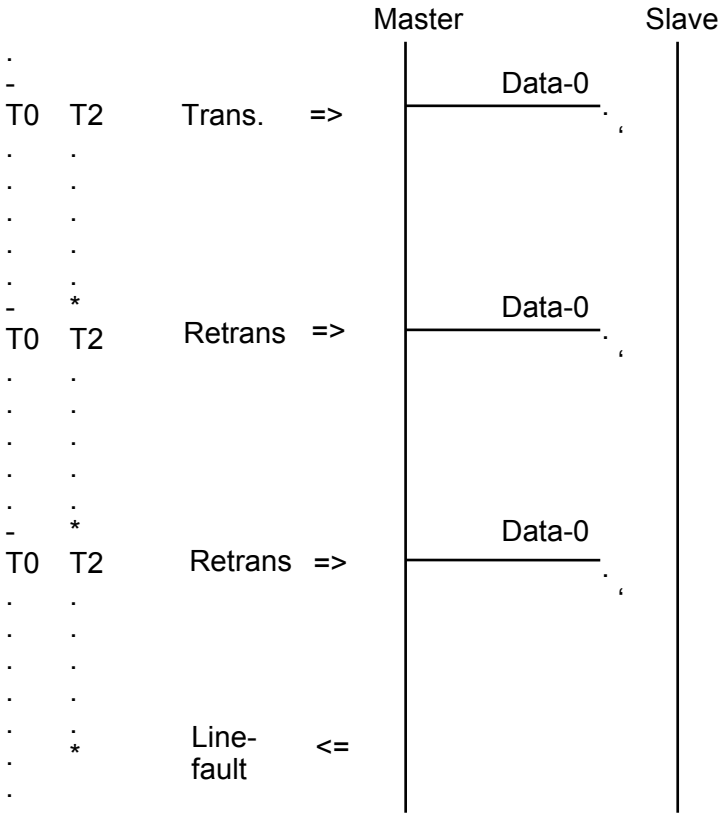


Fig. 4.6 L-Master no transmission to Slave

4.1.1.7 No transmission to Master

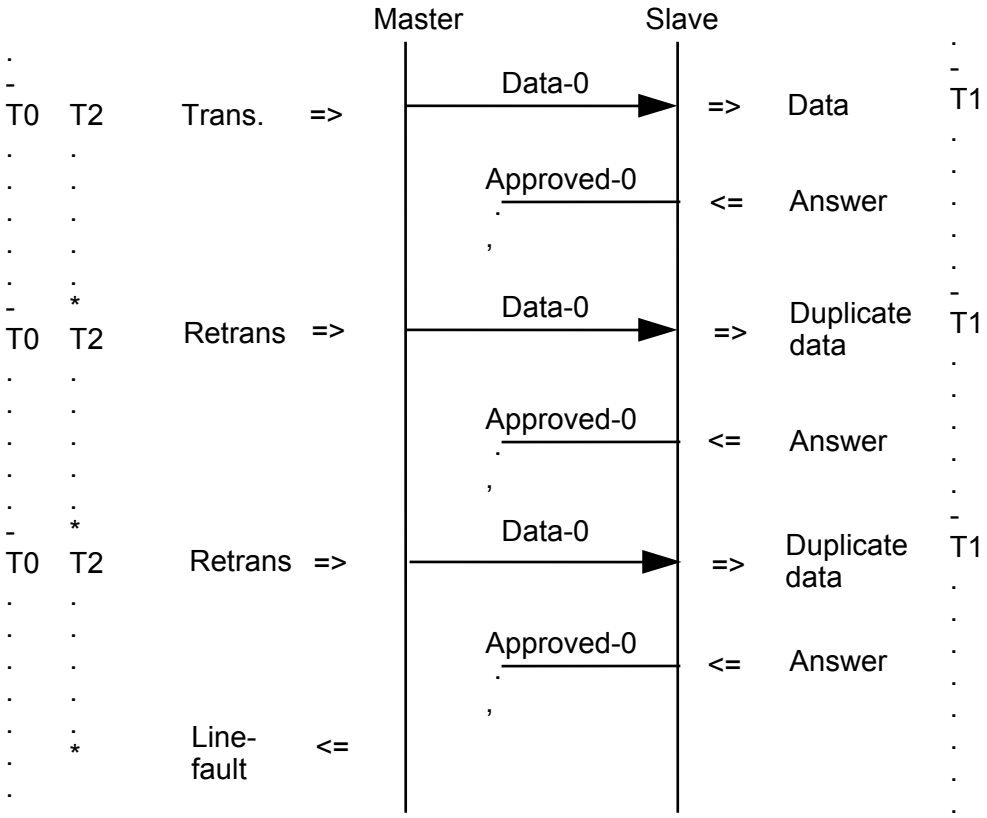


Fig. 4.7 L-Master no transmission to Master

4.1.2 Line Situation L-Slave

This section deals with the line situation L-Slave, where Master is in the M-Idling situation, and Slave is in the S-Data. The Slave situation S-Request also belongs here but is not used in this protocol.

4.1.2.1 Normal cases

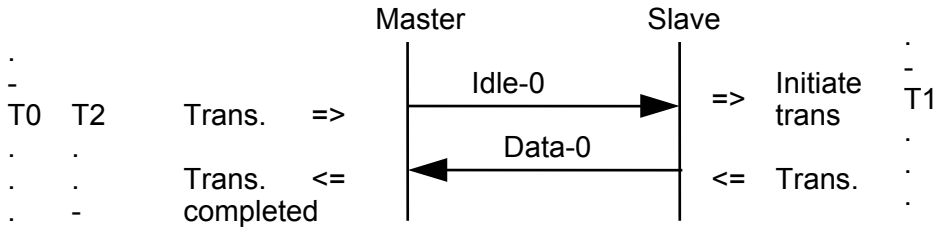


Fig. 4.8 L-Slave normal procedure

4.1.2.2 Transmission fault from Master

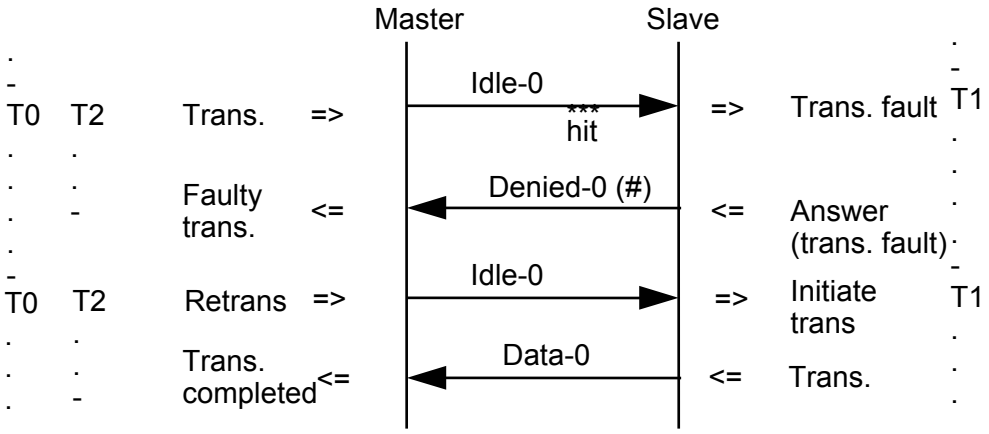


Fig. 4.9 L-Slave transmissions fault from Master

4.1.2.3 Transmission fault from Slave

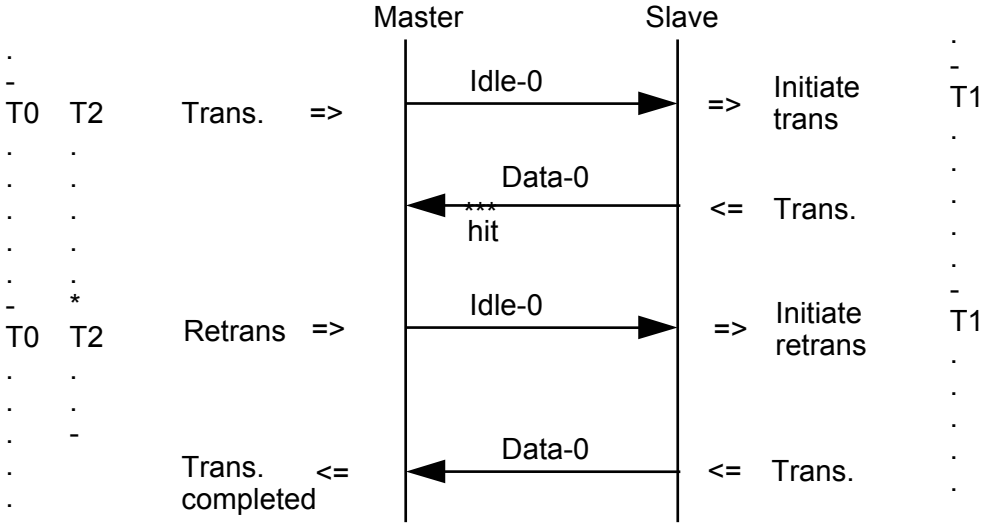


Fig. 4.10 L-Slave transmission fault from Slave

4.1.2.4 Transmission to Slave disconnected

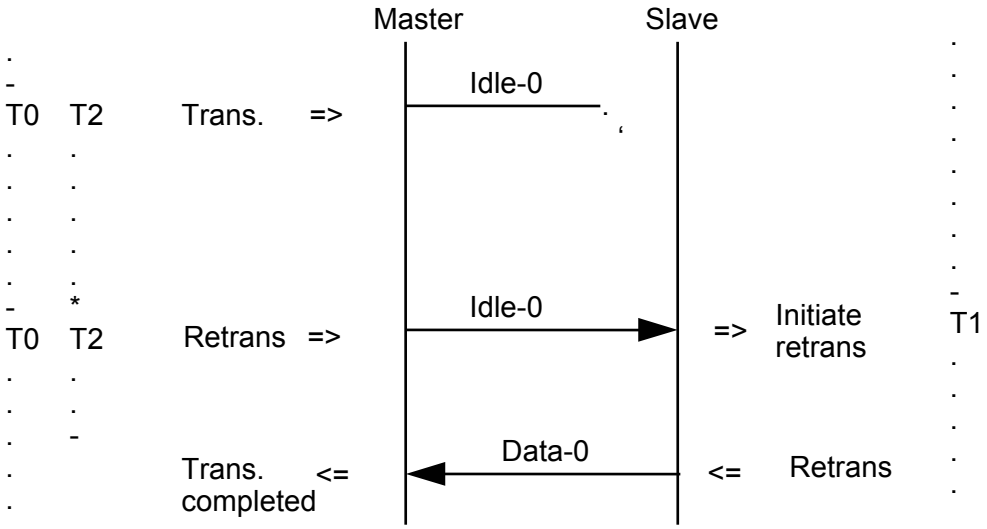


Fig. 4.11 L-Slave transmission to Slave disconnected

4.1.2.5 Transmission to Master disconnected

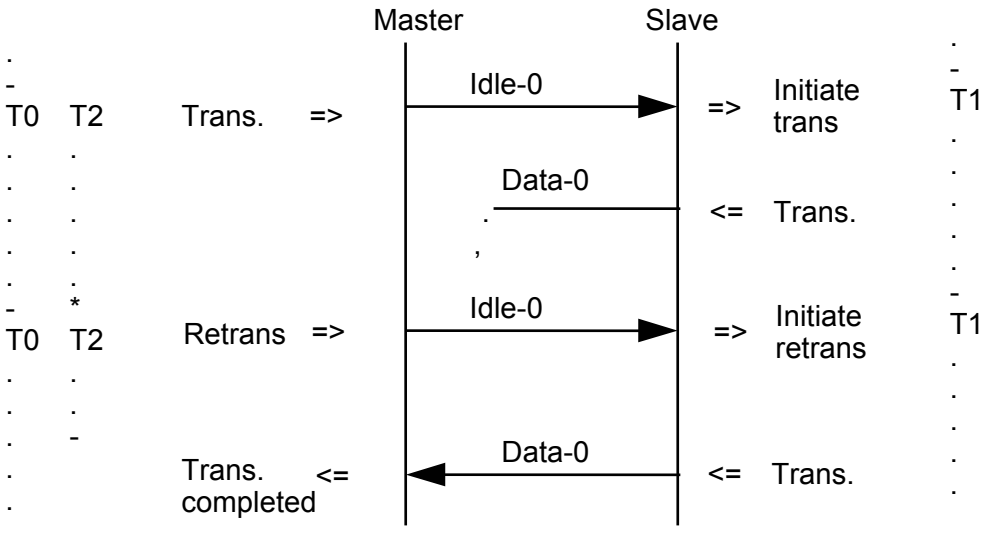


Fig. 4.12 L-Slave transmission to Master disconnected

4.1.2.6 No transmission to Slave

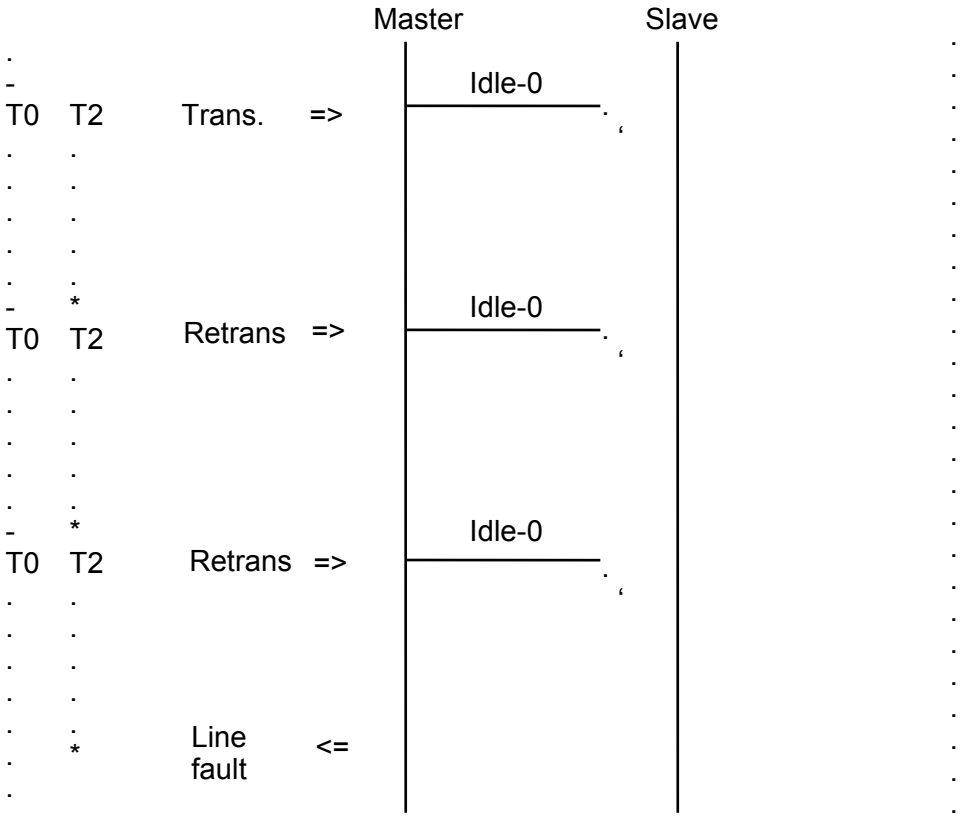


Fig. 4.13 L-Slave no transmission to Slave

4.1.2.7 No transmission to Master

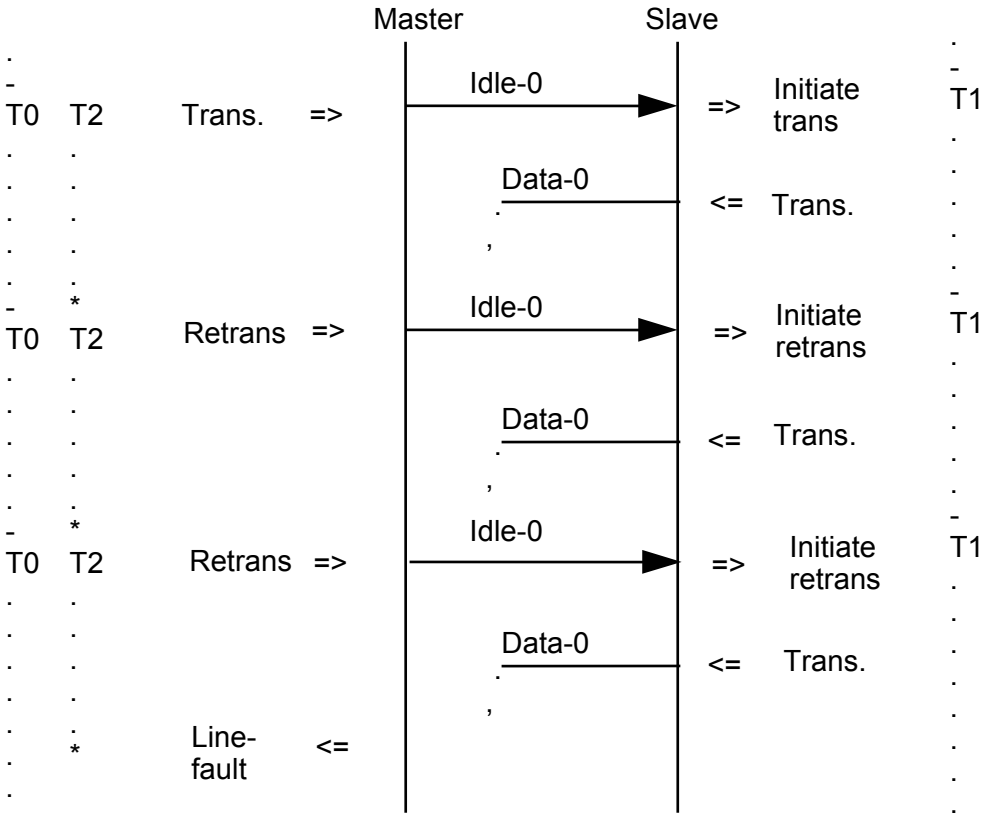


Fig. 4.14 L-Master no transmission to Master

4.1.3 Line situation L-Idling

This section deals with the Line Situation L-Idling where Master is in M-Idling, and Slave is in the S-Idling situation.

4.1.3.1 Normal cases

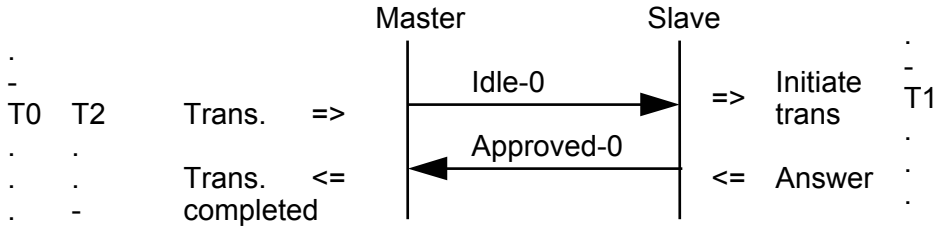


Fig. 4.15 L-Idling normal procedure

4.1.3.2 Transmission faults from Master

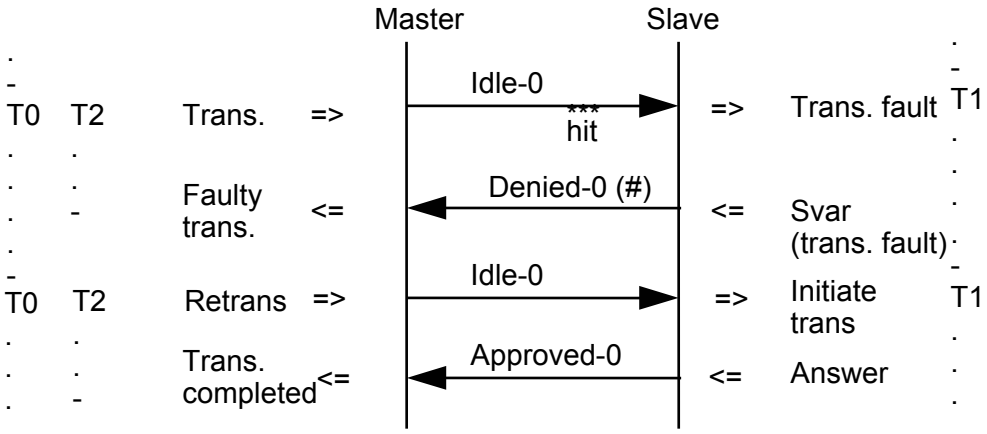


Fig. 4.16 L-Idling transmission faults from Master

4.1.3.3 Transmission faults from Slave

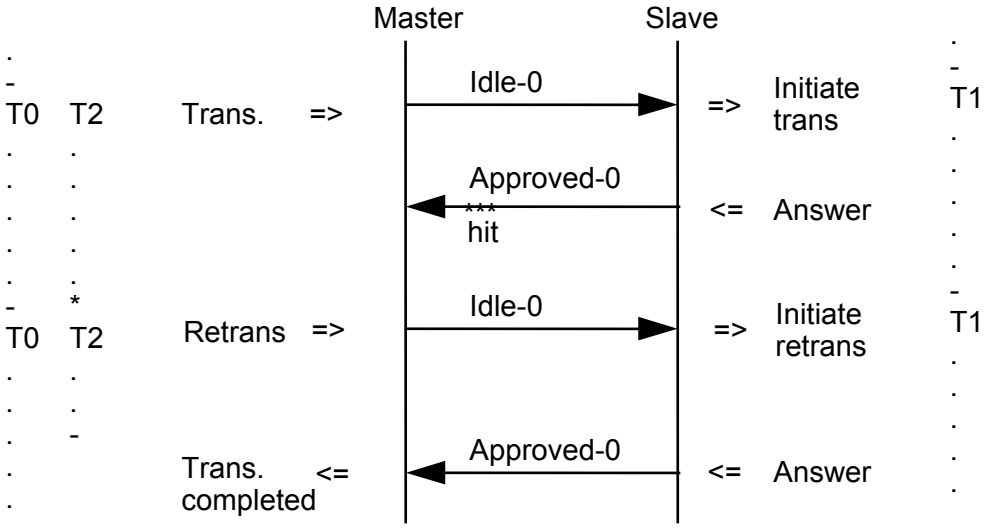


Fig. 4.17 L-Idling transmission faults from Slave

4.1.3.4 Transmission to Slave disconnected

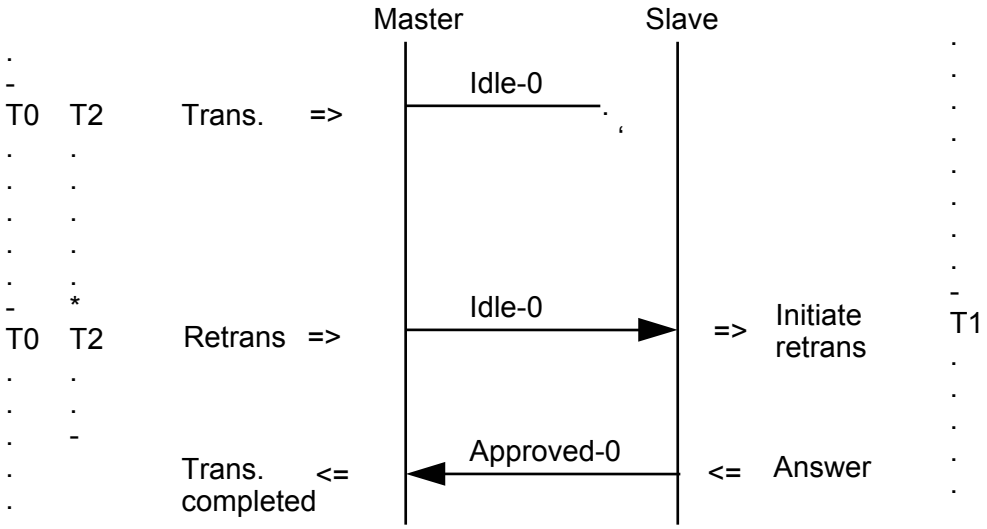


Fig. 4.18 L-Idling transmission to Slave disconnected

4.1.3.5 Transmission to Master disconnected

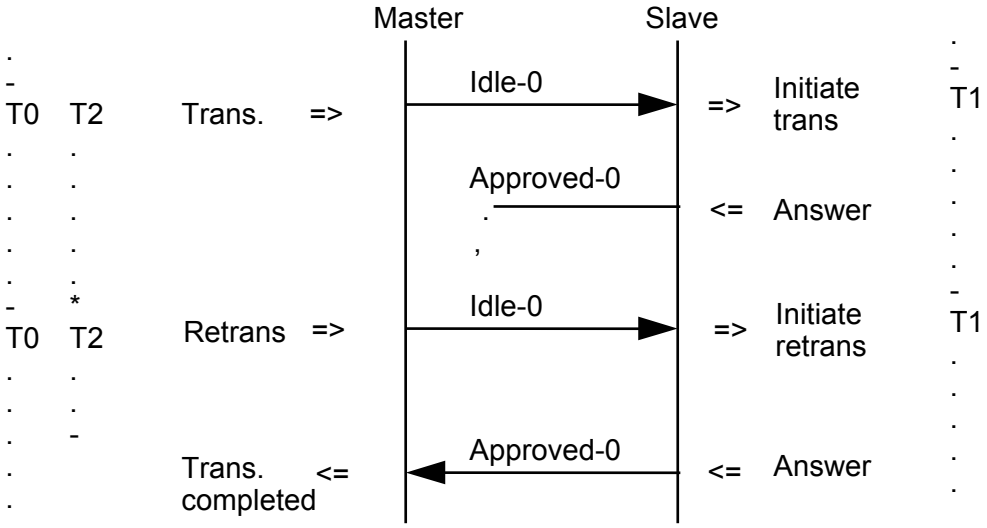


Fig. 4.19 L-Idling transmission to Master disconnected

4.1.3.6 No transmission to Slave

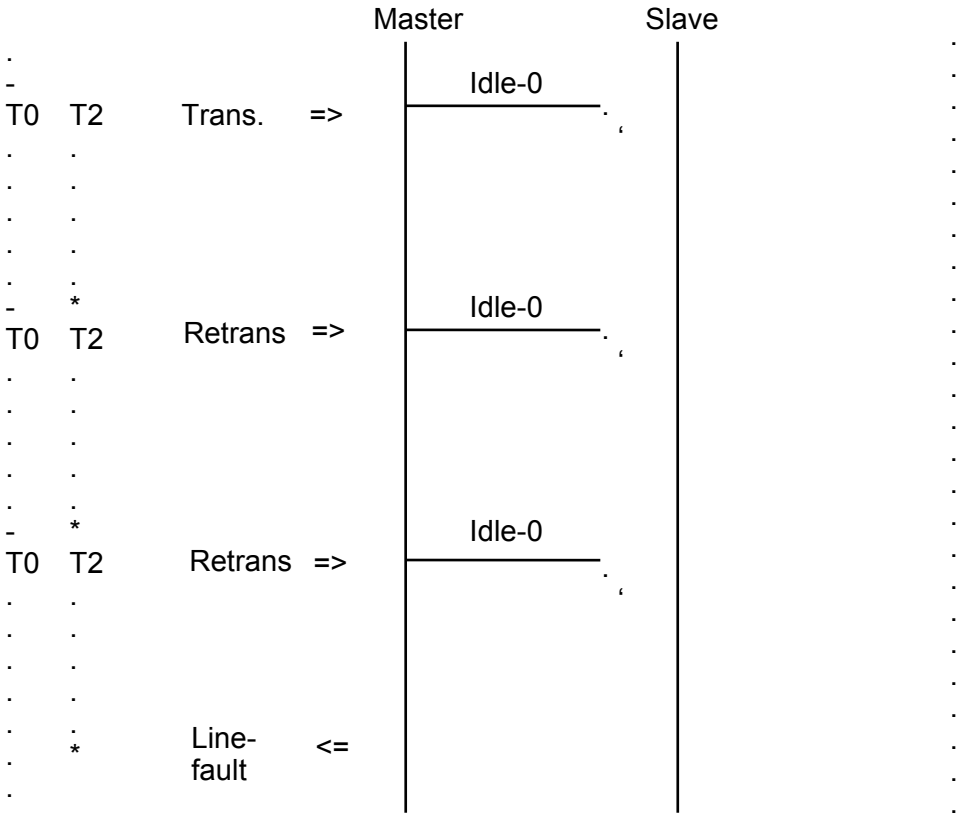


Fig. 4.20 L-Idling no transmission to Slave

4.1.3.7 No transmission to Master

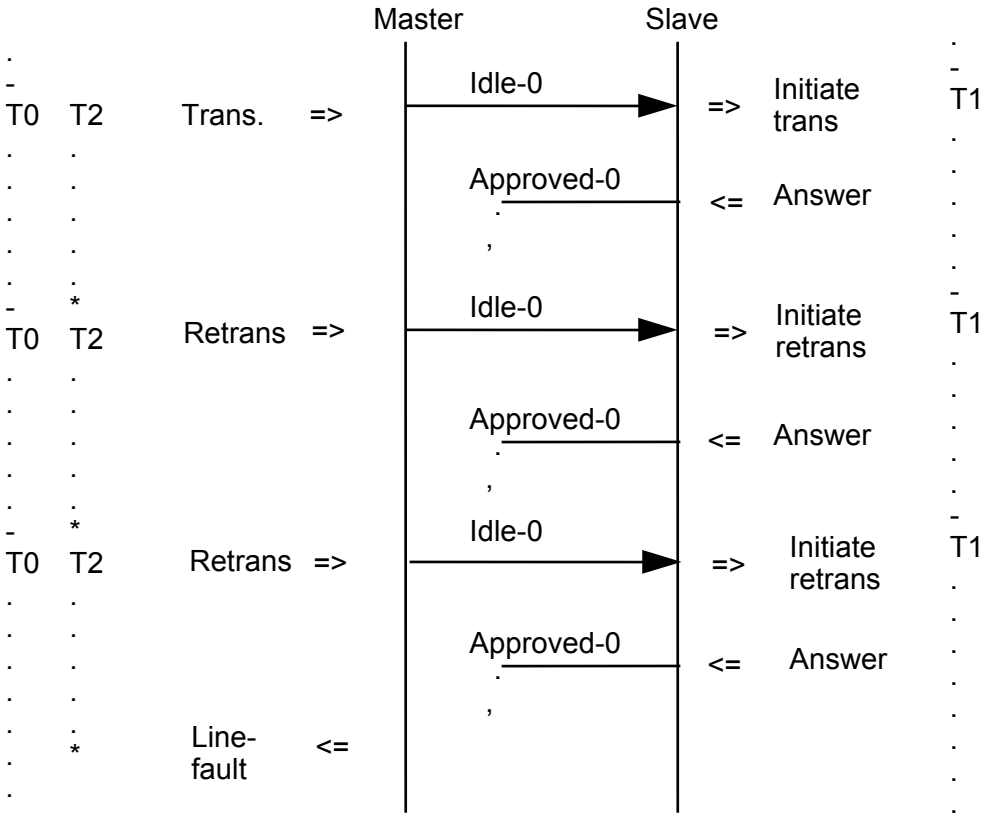


Fig. 4.21 L-Idling to Master

4.1.4 Line situation L-Faults

The Line Situation L-Faults is linked up with the Line Monitoring in the Master-Slave-system. The Line Monitoring Function involves all types of telegrams and its object is to secure that Master as well as Slave are updated with the actual situation in the line.

The Line Monitoring operates with four terms:

- Master line faults.
- Master line in order.
- Slave line faults.
- Slave line in order.

4.1.4.1 Master line faults

Master line faults will occur on the Master side when a Telegram has been sent three times without an answer having been received. In fig. 3.4 the situation in Master is designated M-fault resulting in the Line situation L-fault.

4.1.4.2 Master line in order

Master line in order will occur when Master again receives a correct answer to a transmitted Telegram. In fig. 3.4 the stuations are designated M-Data, M-Request or M-Idling. Correspondingly the Line Situation will be L-Master, L-Slave or L-Idling.

4.1.4.3 Slave Line Faults

Slave Line Faults will occur in Slave, when T1 time has passed after correct receipt of a Telegram without a new correct Telegram having been received. In fig. 3.4 this situation is designated S-Fault. It appears that this situation does not influence the Line Situation as it is irrelevant, when Master is not in M-Fault.

4.1.4.4 Slave line in order

Slave line in order will occur when Slave receives a correct Telegram again. The situation in the line may be either L-Master, L-Slave eller L-Idling dependent on the new situations in Master and Slave resp.

4.1.4.5 Line Fault in Master (M-Fault)

When Master has detected a Line Fault, the fault wil be registered. Master will continue the transmission indifferently.

4.2 Sequence Numbering

The Sequence Number is added to each Telegram and may assume either the values 0 or 1 as Master can have max. 1 unsigned Telegram in the line at a time. The Sequence Number is used by Master to secure that the receipt belongs to the correctly sent Telegram. Slave uses the Sequence Number to avoid receiving doubled data on retransmission from Master.

The following rules of Sequence Numbering may be listed:

1. Master contains “Sequence Number sent”, i.e. the Sequence Number to be sent to Slave.
2. Slave contains “Sequence Number expected”, i.e. the Sequence Number expected from Master.
3. In L-Fault Master will fix the Sequence Number at an established value (initial value)
4. In L-Fault Slave will fix the Sequence Number at the value received in the first correct Telegram from Master.
5. Master will change Sequence Number sent for every new Trans initiated by Master. Hence Retrans will not change the Sequence Number.
6. Slave changes expected Sequence Number when the Telegram is correct and when the Sequence Number corresponds to the expected Sequence Number.
7. Slave will always reply with the Sequence Number received.
8. Slave uses Telegrams only in case of change of Sequence Number. If a change of Sequence Number is not made Slave will assume that it is a question of doubled data and will consequently reject the Telegram.

Note: In the following situations are described where the Sequence Number is used actively either by Slave or Master. Please note that the time watches T0, T1 and T2 are independent of the Sequence Number.

4.2.1 Normal operation

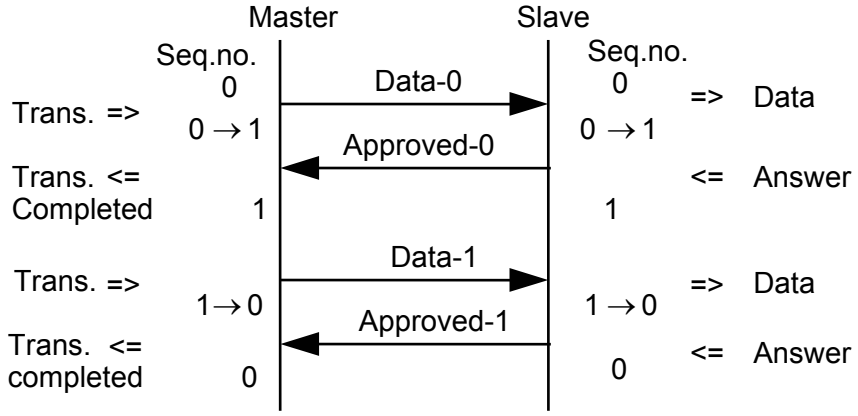


Fig. 4.22 Normal Sequence Number procedure

4.2.2 Starting up

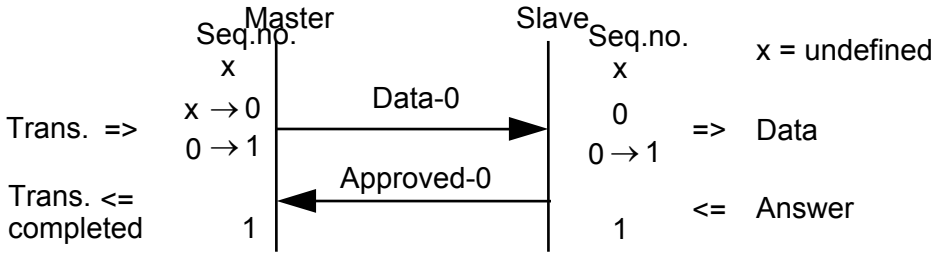


Fig. 4.23 Sequence Numbering at starting up

4.2.3 Line Faults (L-Faults)

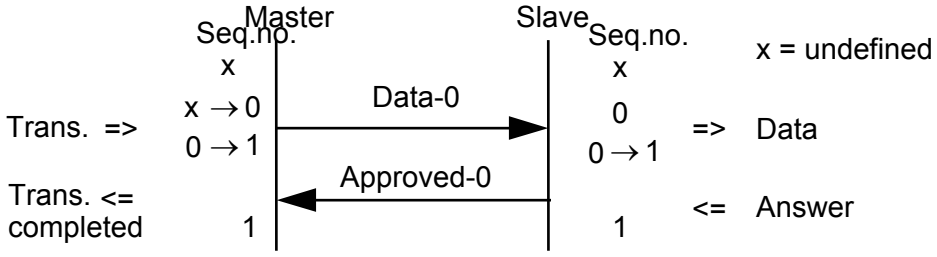


Fig. 4.24 Sequence Numbering in case of Line Faults

4.3 Line Monitoring

This procedure secures that the line between Master and Slave is constantly monitored.. This means that the procedure has been implemented in Master as well as in Slave.

4.3.1 Master Line Monitoring

In Master the procedure will see to it that:

4.3.1.1 Sending of telegram before expiry of timer

A Telegram is sent at least once within the time T0 (T0 is timer No. 0 as defined i section 3)

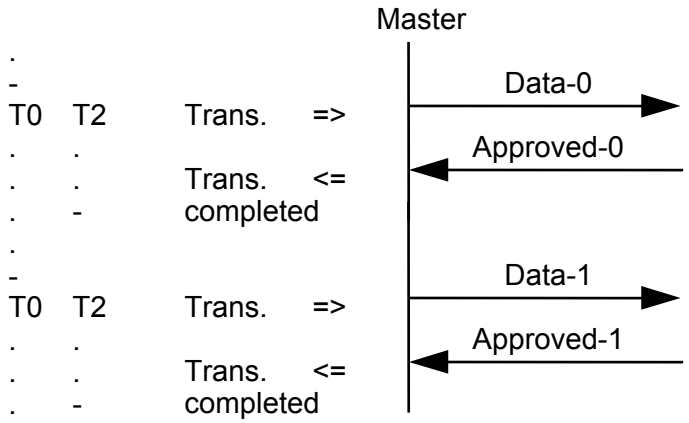


Fig. 4.25 Sending of Telegram before T0 expiry

4.3.1.2 Polling at timer expiry

If no Telegrams are sent from Master, an Idling Telegram will be sent. Hence the Line Monitoring has a Poll-function towards Slave.

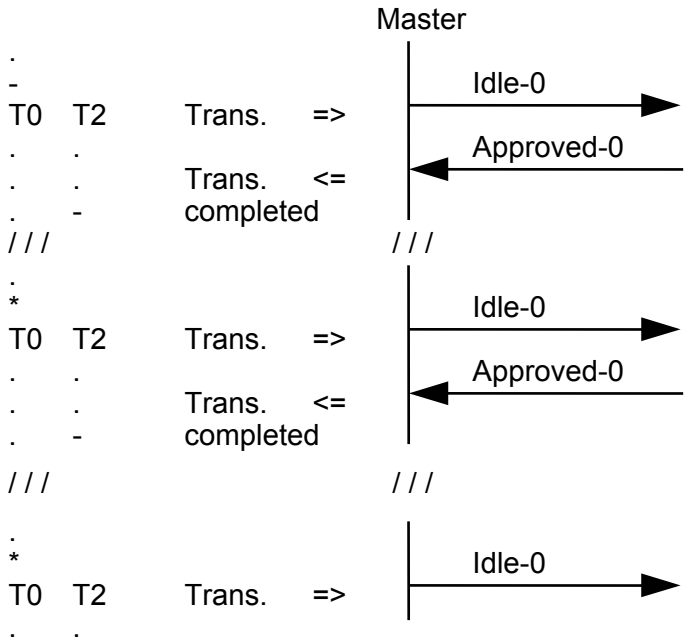


Fig. 4.26 Sending of Telegram at T0 expiry

4.3.1.3 Report of line status

Report of Line Fault or Line in order depends on the status received in the Telegrams sent

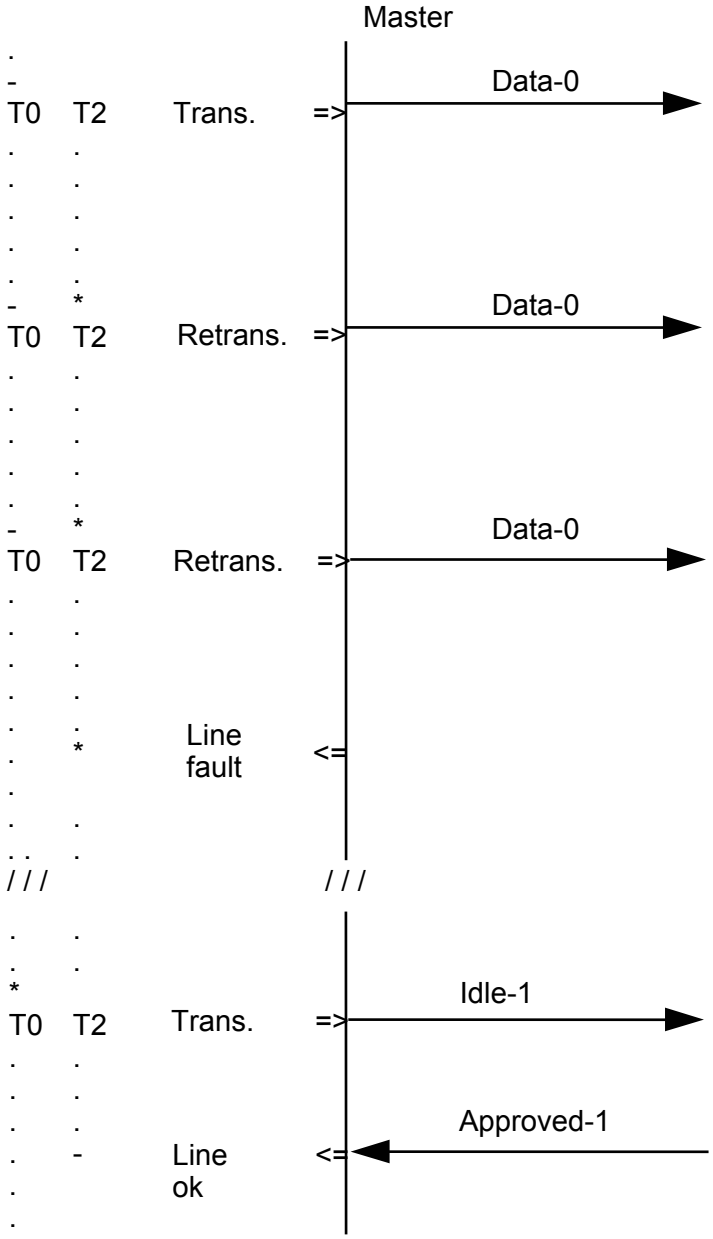


Fig. 4.27 Report of Line Faults and Line in order

4.3.2 Slave monitoring

In Slave the procedure will see to it that:

4.3.2.1 Report of line status

Reporting of Line Faults depend on the status received of the Telegrams received in the Telegrams received within the time T1

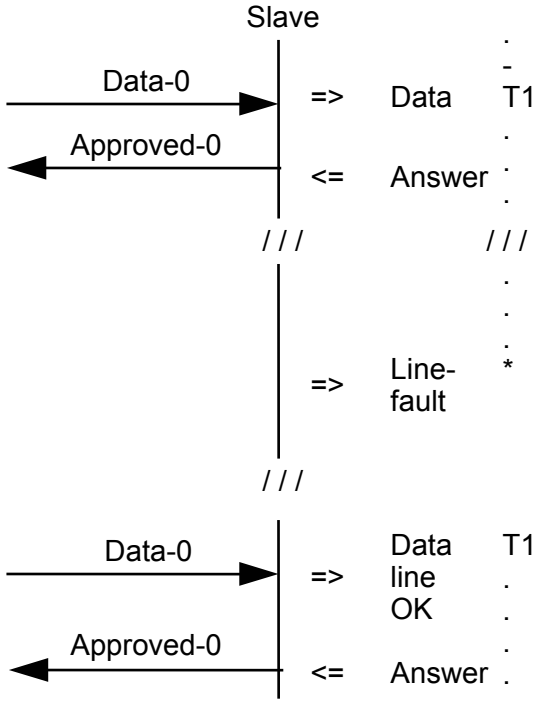


Fig. 4.28 Report of Line Fault and Line in order

4.4 Procedure for calculation of checksum

Checksum results from a current (i.e. during transmission) “addition of modulo 2 in the n’th power” to the co-factor of two, and this Checksum will constitute the last character of the block transmitted.

When the block is retransmitted “modulo 2n is added” to the singular bytes incl. The Checksum in the receiver. If the transmission was correct the result of the current addition will be zero.

Example: Checksum on transmitter side.
We intend to transmit a block consisting of the following 5 data bytes using Checksum mod-256 (100 HEX). Mod-256 results from having 8 data bit, and modulo is described as 2 in the n’th power equal to the numjber of data bit, in this case 8. This gives $2^8 + 1 = 256$:

byte 0: 01 = 0000 0001
byte 1: 10 = 0001 0000
byte 2: A0 = 1010 0000 Hexadecimal
byte 3: 00 = 0000 0000 notation
byte 4: C9 = 1100 1001

"sum mod 2^8" = 0111 1010

1's co-factor = 1000 0101
+ 1

Checksum =1000 0110 = 86H (byte 5)

This means that we are going to transmit the following block:

01 10 A0 00 C9 86
Data bytes Checksum

Example: Checksum on receiver side.

In the receiver a current addition towards 2^8 takes place:
byte 0: 01 = 0000 0001
byte 1: 10 = 0001 0000
byte 2: A0 = 1010 0000 assumed correct
byte 3: 00 = 0000 0000 transmission
byte 4: C9 = 1100 1001
byte 5: 86 = 1000 0110
"sum towards 2^8" = 0000 0000 no faults!

A comment is attached to the checksum calculation:

As the Telegrams are kept in limited ASCII, with a view to a possible printer control, Checksum is recoded to a legible ASCII code.

It means that if the Checksum is e.g. 8AH, it is transcribed to 8A ASCII, meaning 38H and 41H.

Where it was previously of 1 byte it is now of 2 bytes.

It is then recoded to hexadecimal on receipt before the last "addition towards 256" is made.

5.0 FORMAT SPECIFICATIONS

This section comprises format for Telegram, specification of Telegram types, format for Packet type, the limited set of characters for Data packets and specification of Packet types.

5.1 Telegram format

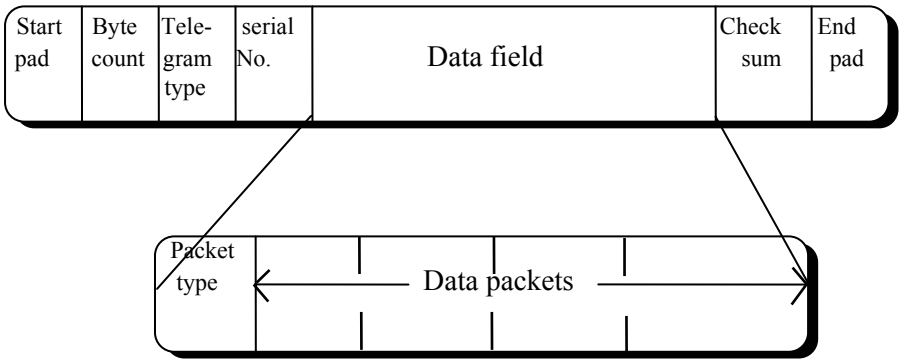


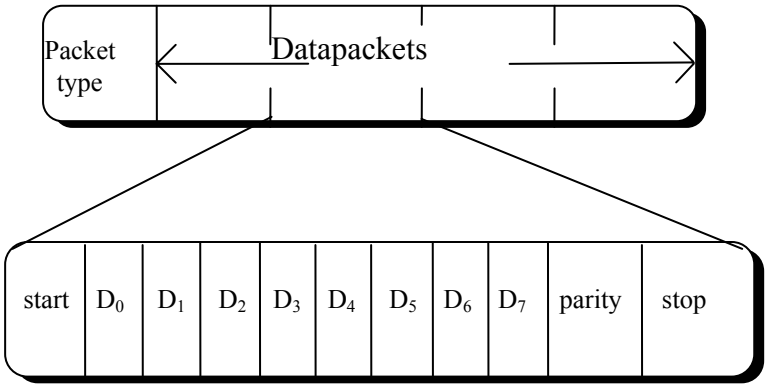
Fig. 5.1 Example of a telegram.

5.2 Specification of Telegram types

Packet type	Description	Beschreibung	ATC-TC	ATC-MSR3	MSR3-TC	ATC-HLOG	Telegram category: Request	Data	Idling
1	Actual data from ATC to MSR3	funkspezifische Tlg.daten		X				X	
2	Train data from from ATC to HLOG	Zugdaten von ATC an HLOG				X		X	
3	Speed info from ATC to HLOG	Geschwindigkeitinfo. von ATC an HLOG				X		X	
4	Transponder info from ATC to HLOG	Streckentlg. von ATC an HLOG				X		X	
5	Emergency stop from MSR3 to ATC	Nothalt von MSR3 an ATC		X				X	
6	Speed receipt from HLOG to ATC	Acknowledge von HLOG an ATC				X		X	
7	Signals passed from ATC to HLOG	Signalvorbeifahrt von ATC an HLOG				X		X	
8	Free	frei							
9	Free	frei							
A	Approved	anerkannt	X	X	X	X			
B	Actual data fromTC to ATC	Zugdaten von TC an ATC	X					X	
C	Reserved	reserviert		X					
D	Reserved	reserviert		X					
E	Actual data from ATC to TC	Zugdaten von ATC an TC	X					X	
F	Pos./sig. No. from ATC to MSR3	funkspezifische Tlg.daten		X				X	
G	Fault code from ATC to TC and from ATC to HLOG	Fehlercode von ATC an TC und von ATC an HLOG	X			X		X	
H	HKT pace information	HKT-geschwindigkeitmeldung				X		X	
I	Reserved	reserviert		X					
J	Reserved	reserviert		X					
K	Actual data from TC to MSR3	funkspezifische Tlg.daten			X			X	
L	Reserveret	reserviert		X	X				
M	Reserveret	reserviert		X					
N	Rejected	abgewiesen	X	X	X	X			

Specification of Telegram types									
Packet type	Description	Beschreibung	ATC-TC	ATC-MSR3	MSR3-TC	ATC-HLOG	Telegram catecory		
							Request	Data	Idling
O	Actual data fra MSR3 to TC	funkspezifische Tlg.daten			X			X	
P	Free	frei							
Q	Reserved	reserviert		X	X				
S	Reserved	reserviert		X					
T	Reserved	reserviert		X					
U	Reserved	reserviert		X					
V	Reserved	reserviert		X					
W	Reserved	reserviert		X					
X	Reserved	reserviert		X					
Y	Reserved	reserviert		X					
Z	Reserved	reserviert		X					
a	Operational telegram from ATC to HLOG	Betriebs-tlg von ATC an HLOG				X		X	
b	Free	frei							
c	Test	Test	X			X		X	
d	Reserved	reserviert	X						
e	Reserved	reserviert	X						
f	Free	frei							
g	Free.	frei							
h	Free.	frei							
i	Free.	frei							
j	Free	frei							
k	Free	frei							
l	Free	Frei							
m	Free	Frei							
n	Trainnumber to TC	Zugnummer zum TC			X			X	
o	Errorcode from MSR3 to TC	Fehlermeldung MSR3 zum TC			X			X	
p	Free	frei							
q	Free	frei							
r	Request for specific data	Ersuchen spezifischen Daten	X	X	X		X		
s	Free	frei							
t	Idling data	Leerlaufs Daten	X	X	X				X
u	Free	frei							
v	Free	frei							
w	Free	frei							
x	Free	frei							
y	Free	frei							
z	Free	frei							

5.3 Packet type format



5.3.1 The limited ASCII set of characters for Data Packets

Only the following characters are valid:

ASCII Char.	Binær	OCT	HEX	DEC
%	0010010	045	25	37
(	0010100	050	28	40
)	0010100	051	29	41
-	0010110	055	2D	45
:	0011101	072	3A	58
+	0010101	053	2B	43
0	0011000	060	30	48
1	0011000	061	31	49
2	0011001	062	32	50
3	0011001	063	33	51
4	0011010	064	34	52
5	0011010	065	35	53
6	0011011	066	36	54
7	0011011	067	37	55
8	0011100	070	38	56
9	0011100	071	39	57
A	0100000	101	41	65
B	0100001	102	42	66
C	0100001	103	43	67
D	0100010	104	44	68
E	0100010	105	45	69
F	0100011	106	46	70
G	0100011	107	47	71
H	0100100	110	48	72
I	0100100	111	49	73
J	0100101	112	4A	74
K	0100101	113	4B	75
L	0100110	114	4C	76
M	0100110	115	4D	77
N	0100111	116	4E	78
O	0100111	117	4F	79
P	0101000	120	50	80
Q	0101000	121	51	81
R	0101001	122	52	82
S	0101001	123	53	83
T	0101010	124	54	84
U	0101010	125	55	85
W	0101011	126	56	86
V	0101011	127	57	87
X	0101100	130	58	88
Y	0101100	131	59	89
Z	0101101	132	5A	90
space	0010000	040	20	32

ASCII Char.	Binær	OCT	HEX	DEC
a	0110000	141	61	97
b	0110001	142	62	98
c	0110001	143	63	99
d	0110010	144	64	100
e	0110010	145	65	101
f	0110011	146	66	102
g	0110011	147	67	103
h	0110100	150	68	104
i	0110100	151	69	105
j	0110101	152	6A	106
k	0110101	153	6B	107
l	0110110	154	6C	108
m	0110110	155	6D	109
n	0110111	156	6E	110
o	0110111	157	6F	111
p	0111000	160	70	112
q	0111000	161	71	113
r	0111001	162	72	114
s	0111001	163	73	115
t	0111010	164	74	116
u	0111010	165	75	117
v	0111011	166	76	118
w	0111011	167	77	119
x	0111100	170	78	120
y	0111100	171	79	121
z	0111101	172	7A	122
	^			
Følgende tolkes som DS				
æ	0111101	173	7B	123
ø	0111110	174	7C	124
å	0111110	175	7D	125
Æ	0101101	133	5B	91
Ø	0101110	134	5C	92
Å	0101110	135	5D	93

5.4 Specification of Packet types

Packet type	Description	Beschreibung	ATC-TC	ATC-MSR3	MSR3-TC	ATC-HLOG
A	Approved	anerkannt	X	X	X	X
B	Request for actual train data	Anruf	X	X	X	
C	C-channel	C-kanal	X	X	X	
D	D-channel	D-kanal	X	X	X	
E	Fault code	Fehlercode	X			X
F	FC-channel	FC-kanal		X	X	
G	Monitoring pace	Uberwachungsgeschwindigkeit				X
H	max. pace in km/h	max. Geschwindigkeit in km/h	X			X
I	Operational function	Betriebsfunktion				X
J	Wheel diameter	Raddurchmesser				X
K	Immediate pace	Istgeschwindigkeit				X
L	Length of train in meters	Zuglänge in Metern	X			X
M	Actual operational form	aktuelle Betriebsart		X	X	
N	Rejected	abgewiesen	X	X	X	X
O	Reserved	Reserve			/	
P	Position	Position	X	X	X	
Q	Reserved	Reserve			/	
R	ATC-direction	ATC-richtung	X			X
S	Stop signal passed	Signalvorbei-fahrt			X	X
T	Train No.	Zugnummer			X	
U	Emergency stop	Nothalt		X		
V	Watch function	Wachefunktion		X	X	
W	C-effect	C sendeleistung		X		
X	D-effect	D sendeleistung		X		
Y	Sign on and sign off	Ab- und Anmeldung		X		
Z	Landfall figures	Landkendungs-zahl		X		
%	Braking percentage	Bremsprozente	X			X
a	Transponder GK 1	Balise GK 1				X
b	Transponder GK 2	Balise GK 2				X
c	Transponder GK 3	Balise GK 3				X
d	Transponder GK 4	Balise GK 4				X
e	Transponder reserve	Balise Reserve				X
f	Transponder reserve	Balise Reserve				X
g	Transponder reserve	Balise Reserve				X
h	Transponder operational control	Balise Betriebskontrolle				X
i	Reserved	reserviert		/	/	
j	Reserved	reserviert		/	/	
k	Reserved	reserviert		/	/	
l	Reserved	reserviert		/	/	
m	Lamp test	Lampentest	X			X
n	S-/DK situation	S-/DK-Zustand				X*)
o	Free	frei				
p	Free	frei				
q	Free	frei				
r	Free	frei				
s	Free	frei				
t	Free	frei				
u	Free	frei				

*) Also used for HKT-info in S-trains

6.0 TELEGRAM CONTENTS ATC - TC

This section comprises a description of the telegram contents between ATC and TC.

6.1 Telegram contents between ATC and TC

Primarily the purpose of this connection is that ATC in vehicles with TC may obtain the train data at starting up. Then ATC can be updated cyclically. A profound study of the contents of the various Telegram Types is referred to in appendix in this section.

6.2 Telegram type r: Request for actual data

This Telegram sends ATC to TC immediately after starting up as a request to TC to send train data (Telegram B). Then the Telegram is sent cyclically and consequently acts as a Line Monitoring.

Packet type B: Request

B	*

TC is requested to answer this Telegram with the Telegram type as stated by data. Example. data = B.

6.3 Telegram type B: Actual data from TC to ATC

At the request from ATC with the request telegram r and data = B, TC shall not acknowledge with ACK but with Telegram type B.

Packet type L: Train Length

L	*	*	*

Data are in the interval from 0 to 999 m.

Packet type %: Braking percentage

%	*	*	*

Braking percentage in the interval 0 to 999 %.

Packet type H: Max. pace

H	*	*	*

Pace in the interval from 0 to 999 km/h.

Packet type R: ATC direction

R	*	*

Data: 8 8 = Direction A
0 0 = Direction b

Expounding of ATC direction is outside this protocol. Direction "A" is sent also when TC does not identify the direction.

6.4 Telegram type E: Actual data from ATC to TC

Omitted.

6.5 Telegram type F: Position

This telegram will send ATC immediately after passage of a transponder. The position is identified by the signal number.

Packet type P: Position

P	*	*	*	*

Area:
1. digit after P may have the following values 0,1,2,3,4,5,6,7. (SNR1 = 3 bit)
0 → Station
1,2,3,4,5,6,7 → Track No. in section

The following 3 digits cover in decimals the area 0 - 999 (SNR2 = 10 bit). The three digits state the sectional kilometer measurement. (In the area 0 - 99,9 km the three digits state a distance in hectometer. In the area from 100 km and up the three digits state a distance in km).

6.6 Telegram type G: Fault code

The ATC fault code is sent to TC immediately after having been detected. TC will store only the first two digits.

Packet type E: Fault code

E	*	*	*

Data is in the interval from 000 to 999.

6.7 Telegram type b: Braking percentage from TC

Omitted

6.8 Telegram type c: Test

This telegram is used to testing all lamps in the driver's cab signal as well as the encoding panel. At the starting up of ATC these are tested, there are, however lamps in the driver's cab signal which are controlled by HLOG and partly by TC. When TC and HLOG receive this telegram they should switch resp. switch off the lamps.

Packet type m: Lamp test

M	*

Data: S: Switch off lamp
T: Switch on lamp

6.9 Telegram type t: Idling

Until further this Telegram will be suppressed, as a Line Monitoring has been established by ATC sending a Telegram with a fixed minimum time interval.

6.10 Telegram type N: Rejected Telegram

It has been agreed that data N will be the following:

Data N = 1 means Checksum fault
2 means Timing fault
A means Application fault *)

*) Application fault (data validation fault) i.e. a fully decoded but meaningless Telegram.

Appendix 6.1

Telegram contents ATC-TC

Telegram type	A	c	r	t	N
Start Pad	LF	LF	LF	LF	LF
Byte Counts	Byte Count 1	Byte Count 1	Byte Count 1	Byte Count 1	Byte Count 1
	Byte Count 2	Byte Count 2	Byte Count 2	Byte Count 2	Byte Count 2
Telegram type	A	c	r	t	N
	Telegram serial No.	Telegram serial No.	Telegram serial No.	Telegram serial No.	Telegram serial No.
Packet type	A	m	B	B	N
Data packet	data A	data m	data B	data B	data N
Checksum bytes	Checksum 1	Checksum 1	Checksum 1	Checksum 1	Checksum 1
	Checksum 2	Checksum 2	Checksum 2	Checksum 2	Checksum 2
End pad	CR	CR	CR	CR	CR
	Approved TC to ATC	Request for test ATC to TC	Request for specific data ATC tol TC	Idling data ATC to TC	Rejected TC to ATC

Telegramtype	B
Start pad	LF
Byte counts	Byte count 1
	Byte count 2
Telegramtype	B
	Telegram serial No.
Packet type	L
Data packet	data L
	data L
	data L
Packet type	%
Data packet	data %
	data %
	data %
Packet type	H
Data packet	data H
	data H
	data H
Packet type	R
Data packet	data R
	data R
Checksum bytes	Checksum 1
	Checksum 2
End pad	CR

Actual data

TC to ATC

Telegram type	F
Star pad	LF
Byte counts	Byte count 1
	Byte count 2
Telegram type	F
	Telegramløbenr.
Packettype	P
Datapacket	data P
	data P
	data P
	data P
Checksum-bytes	Checksum 1
	Checksum 2
End pad	CR

Pos./signal No.

ATC to TC

Telegram type	G
Start pad	LF
Byte counts	Byte count 1
	Bytecount 2
Telegramtype	G
	Telegramløbenr.
Packettype	E
Datapacket	data E
	data E
	data E
Checksum-bytes	Checksum 1
	Checksum 2
End pad	CR

Fault code

ATC to TC

7.0 Telegram contents TC - MSR3

This section comprises a description of the contents of Telegrams exchanged between ATC and MSR3.

7.1 Telegram contents between ATC - MSR3

The primary purpose of this connection is automatic control of operational forms and change of channel for MSR3 as well as indication of position. Moreover there is a possibility to change emergency stop telegrams. For the exact contents of each Telegram type reference is made in Appendix 1 in this section.

7.2 Telegram type 1: Radio transponder telegram

The Telegram is sent from ATC to MSR3 immediately after passage of a radio transponder (GK = 4).

Data packet type F: FC-channel

F	C	*	*

Area:
2 decimal digits, 0 - 99 ⇒ 7 bit

Example: In case that the following has been encoded in the the radio transponder for FC-channel 0111110, ATC will send the following to MSR3 af passage of the transponder.

F	C	6	2

Data packet type D: D-channel

D	*	*

Area:
2 decimal digits , 0 - 99 ⇒ 7 bit

Example:. In case the following has been encoded in the radio transponder for D-channel 0010010, ATC will send the following to MSR3 after passage of the transponder.

D	1	8

Data packet type C: C-channel

C	*	*

Area:
2 decimal digits, 0 - 99 ⇒ 7 bit

Example: In case the following has been encoded in the radio transponder for the C-channel 0110000, ATC will send the following to MSR3 after passage of the transponder.

C	4	8

Data packet type M. Actual operational form

M	:	*

Area:
1 digit with 8 possibilities ⇒ 3 bit
A : 0 0 0
B : 0 0 1
C : 0 1 0
D : 0 1 1
Not defined : 1 0 0
F : 1 0 1
Not defined : 1 1 0
Not defined : 1 1 1

“F” means that Data packet should be ignored in MSR3.

Example: In the transponder the 3 bit has been coded as 0 1 0 in which case the following will be sendt from ATC to MSR3.

M	:	C

Data packet type V: Watch function

V	:	*	*

Area:
2 digits each with 8 possibilities ⇒ 6 bit

For each digit 3 bits are used and are coded as per the following table:

'A' : 0 0 0
'B' : 0 0 1
'C' : 0 1 0
'D' : 0 1 1
Not defined : 1 0 0
'F' : 1 0 1
Not defined : 1 1 0
' ' ascii 32D : 1 1 1
'F' means that Datapacket should be ignored in i MSR3.

Example: In the transponder the 6 bit are coded as 1 1 1 0 1 1 in which case the following wil be sent from ATC to MSR3.

V	:		D

Note: After colon (':') is sent, in the example shown, ascii 32 decimal = space.

Data packet type P: Position

P	*	*	*	*

Area:

1st digit after P may have the following values 0, 1, 2, 3, 4, 5, 6, 7. (SNR1 = 3 bit).
0 → Station.
1,2,3,4,5,6,7 → Track No. in section

The following 3 digits cover with the decimals the area 0 - 999 (SNR2 = 10 bit). The three digits state sectional measurement in kilometers (in the area. 0-99.9 km the three digits state a distance in hectometers. In the area from 100 km and up the three digits state a distance in km).

Data packet type W: C-effect

W	*

Area:
8 possibilities ⇒ 3 bit.

F : 0 0 0
1 : 0 0 1
Not defined : 0 1 0
Not defined : 0 1 1
Not defined : 1 0 0
Not defined : 1 0 1
6 : 1 1 0
Not defined : 1 1 1

'F' means that Data packet should be ignored MSR3.

Eks. In case 0 0 1 has been encoded in the transponder the following will be sent from ATC to MSR3.

W	1

Data packet type X: D-effect

X	*

Area:
8 possibilities ⇒ 3 bit.

F : 0 0 0
1 : 0 0 1
Not defined : 0 1 0
Not defined : 0 1 1
Not defined : 1 0 0
Not defined : 1 0 1
6 : 1 1 0
Not defined : 1 1 1

'F' means that Data packet should be ignored in MSR3.

Example: In case 0 0 1 has been encoded in the transponder telegram, the following will be sent from ATC to MSR3.

X	1

Data packet type Y: Sign-on/Sign-off

Y	*

Area:
32 possibilities ⇒ 5 bit.

0	→ Mode 0	16	→ Mode G
1	→ Mode 1	17	→ Mode H
2	→ Mode 2	18	→ Mode I
3	→ Mode 3	19	→ Mode J
4	→ Mode 4	20	→ Mode K
5	→ Mode 5	21	→ Mode L
6	→ Mode 6	22	→ Mode M
7	→ Mode 7	23	→ Mode N
8	→ Mode 8	24	→ Mode O
9	→ Mode 9	25	→ Mode P
10	→ Mode A	26	→ Mode Q
11	→ Mode B	27	→ Mode R
12	→ Mode C	28	→ Mode S
13	→ Mode D	29	→ Mode T
14	→ Mode E	30	→ Mode U
15	→ Mode F	31	→ Mode V

Example: In the transponder the 5 bit have been coded as 0 1 0 1 1 in which case the following will be sent from ATC to MSR3.

Y	B

Data packet type Z: Country code

Z	*

Area:
8 possibilities ⇒ 3 bit.

0 0 0 - Country 0 (Denmark)
0 0 1 - Country 1 (Germany)
0 1 0 - Country 2
0 1 1 - Country 3
1 0 0 - Country 4
1 0 1 - Country 5
1 1 0 - Country 6
1 1 1 - Country 7

Example: If the bits in the transponder telegram are programmed as 0 0 1 the following will be sent from ATC to MSR3.

Z	1

7.3 Telegram type r: Request for specific data – Emergency stop

ATC asks cyclically if MSR3 has received message concerning “Emergency Stop from traffic controller”. MSR3 will not answer this telegram with “approved” but with Telegram 5 – Emergency stop. Hence this Telegram acts automatically as Line monitor. In stead of the general T0 is used T0=5 seconds to reduce the delay of an emergency stop.

Data packet type B - Request

B	*

The Slave is requested to answer this Telegram with the Telegram type corresponding to data. If emergency stop then data = 5.

7.4 Telegram type 5: Emergency stop

This Telegram is sent from MSR3 to ATC as an answer to Telegram r from ATC.

Packet type U – Emergency stop

U	*

Data: + : MSR3 has received emergency stop
\- : MSR3 has received **no** emergency stop

7.5 Telegram type F: Position

ATC will send this Telegram immediately after passage of a transponder. The position is traced back to the signal number.

Packet type P - Position

P	*	*	*	*

Area:
1st digit after P may have the following values 0,1,2,3,4,5,6,7 (SNR1 =3 bit).
0 → Station.
1,2,3,4,5,6,7 → Track No. in section.

The following 3 digits cover in decimals the area 0 - 999 (SNR2 = 10 bit). The three digits state sectional measurement in kilometer (in the area 0 – 99.9 km the three digits a distance in hectometer. In the area from 100 km and up the three digits state a distance in km).

7.6 Telegram type N: Rejected Telegram

It has been agreed that data N will be the following:

Data N = 1 means checksum faults
2 means timing faults
A means application faults *)

*) Application faults (data validation faults) i.e. a fully correctly decoded Telegram making no sense.

Appendix 7.1

Telegram contents ATC-MSR3

Telegramtype	A	5	r	t	N
Start pad	LF	LF	LF	LF	LF
Byte counts	Byte count 1	Byte count 1	Byte count 1	Byte count 1	Byte count 1
	Byte count 2	Byte count 2	Byte count 2	Byte count 2	Bytecount 2
Telegram type	A	5	r	t	N
	Telegram serial No.	Telegram serial No.	Telegram serial No.	Telegram serial No.	Telegram serial No.
Packet type	A	U	B	B	N
Data packet	data A	data U	data B	data B	data N
Checksum bytes	Checksum 1	Checksum 1	Checksum 1	Checksum 1	Checksum 1
	Checksum 2	Checksum 2	Checksum 2	Checksum 2	Checksum 2
End pad	CR	CR	CR	CR	CR
	Approved	Emergency stop	Request for specific data	Idling data	Rejected
	MSR3 to ATC	MSR3 to ATC	ATC to MSR3	ATC to MSR3	MSR3 to ATC

Telegram type	1
Start pad	LF
Byte counts	Byte count 1
	Byte count 2
Telegramtype	1
	Telegram serial No.
Packet type	F
Data packet	data F
	data F
	data F
Packet type	D
Data packet	data D
	data D
Packet type	C
Data packet	data C
	data C
Packet type	M
Data packet	data M
	data M
Packet type	V
Data packet	data V
	data V
	data V
Packet type	P
Data packet	data P
	data P
	data P
	data P
Packet type	W
Data packet	data W
Packet type	X
Data packet	data X
Packet type	Y
Data packet	data Y
Packet type	Z
Data packet	data Z
Checksum bytes	Checksum 1
	Checksum 2
End pad	CR
	Radio data
	ATC to MSR3

Telegram type	F
Start pad	LF
Byte counts	Byte count 1
	Byte count 2
Telegram type	F
	Telegram serial No.
Packet type	P
Data packet	data P
	data P
	data P
	data P
Checksum bytes	Checksum 1
	Checksum 2
End pad	CR
	Pos/signal No.
	ATC to MSR3

8.0 TELEGRAM CONTENTS MSR3 - TC

This section comprises a description of the telegram contents between MSR3 and TC.

8.1 Telegram contents between MSR3 - TC

The primary purpose is to exchange actual data. For the exact contents of each Telegram type reference is made to Appendix 1 in this section.

8.2 Telegram type K: Actual data, TC to MSR3

The telegram is sent from TC to MSR3 as an answer to Telegram type r.

Data packet type F: FC-channel

F	C	*	*

Area:
2 decimal digits (0..9)
E.g. FC-channel 62 is coded as follows:

F	C	6	2

Data packet type D: D-channel

D	*	*

Area:
2 decimal digits (0..9)
E.g. D-channel 18 is coded as follows:

D	1	8

Data packet type C: C-channel

C	*	*

Area.
2 decimal digits (0..9)
E.g. C-channel 48 is coded as follows:

C	4	8

Data packet type M: Actual operational form

M	:	*

Area:
1 digit with the following possibilitises : A, B, C, D
E.g. operational form C is coded as follows:

M	:	C

Data packet type V: Watch function

V	:	*	*

Area:
2 digits with the following possibilities:

' ' ASCII 32D, A, B, C, D

E.g. Watch function D is coded as follows:

V	:		D

NB: After colon (':') is sent, in the example shown, ASCII 32 decimal = space.

Data packet type P: Position

P	*	*	*	*

Area:

1st digit after P may have the following values 0,1,2,3,4,5,6,7.

0 → Station.
1,2,3,4,5,6,7 → Track No. in section.

The following 3 digits cover in decimals the area 0 - 999.

The three digits state sectional measurement in kilometer. (In the area 0 – 99.0 the three digits a distance in hectometer. In the area from 100 km and up the three digits state a distance in km.)

8.3 Telegram type O: Actual data MSR3 to TC

The telegram is sent from MSR3 to TC at updating of radio data in MSR3.

Data packet type F: FC-channel

F	C	*	*

Area:
2 decimal digits (0..9)
E.g. FC-channel 62 is coded as follows:

F	C	6	2

Data packet type D: D-channel

D	*	*

Area:
2 decimal digits (0..9)

E.g. D-channel 18 is coded as follows:

D	1	8

Data packet type C: C-channel

C	*	*

Area:
2 decimal digits (0..9)
E.g. C-channel 48 is coded as follows:

C	4	8

Data packet type M: Actual operational form

M	:	*

Area:
1 digit with the following possibilities: A, B, C, D
E.g. operational form C is coded as follows:

M	:	C

Data packet type V: Watch function

V	:	*	*

Area:
2 digits with the following possibilities: " " (ASCII 32D), A, B, C, D
E.g. Watch function D is coded as follows:

V	:		D

Note: After colon is sent, in the example shown, ASCII 32 decimal = space.

Data packet type P: Position

P	*	*	*	*

Area:

1st digit after P may have the following values 0,1,2,3,4,5,6,7.

0 → Station.
1,2,3,4,5,6,7 → Track No. in section

The following 3 digits cover in decimals the area 0 - 999.

The three digits state sectional measurement in kilometer. (I the area 0 – 99.9 km the three digits state a distance in hectometer. I the area from 100 km and up the three digits state a distance in km).

Data packet type T: Train number

T	*	*	*	*	*	*

Area: 6 digits each with possibility for 0 to 9.
E.g. Train number 1,2,3,4,5 og 6 is coded as follows:

T	1	2	3	4	5	6

8.4 Telegram type r: Request for specific data/Actual data

The Telegram is sent from MSR3 to TC, when MSR3 requests actual data from TC.

Data packet type B: Request

B	K

The Slave is requested to answer this Telegram with the Telegram type corresponding to data. Actual data = K.

8.5 Telegram type o (small letter o) (6F HEX): Fault code MSR3 to TC

In case of faults in the MSR3 radio a fault code Telegram is sent to TC:

Data packet type S

S	*	*

Area: 00-99 with the following meaning
01: Communication fault between CF1 and CL
02: Communication fault between CF2 and CL
03: Communication fault between CF3 and CL
04: Communication fault between CB1 and CL
05: Communication fault between CB2 and CL
06: Communication fault between ATC and CL
07: Resets CF due to faults
08: Emergency stop received from ATC

Note!!
Max. 2 equal fault codes should be sent each time MRS3 is coupled in.
This has been introduced to avoid filling up the TC fault recorder.

8.6 Telegram type t: Idling

Is used for Line monitoring (MSR3 to TC). Conditional on the answer to Telegram type A (see 8.8) time watch T0 can be reduced from 20 seconds to 10 seconds as long as MSR3 has not received information about driver's desk key i position Operation (Telegram type A data <> X). When information about Driver's desk in Operations received T0 should be reset to 20 seconds. This function has been introduced to reduce delay of updating of train number in TC.

8.7 Telegram type N: Rejected Telegram

Telegram type N is sent as a negative receipt in case of faults:

Area: 1 means checksum faults
2 means timing faults
A means application faults *)

*) Application faults (data validation faults) i.e. a fully correctly decoded Telegram giving no meaning.

8.8 Telegram type A: Approved Telegram

In case of answers to idling telegrams from TC data packet A is used to transmit information from TC to MSR3 re the position of the driver's desk key:

Area: 'A': Original contents, do not update TC.
'X': Driver's desk in position Operation, update TC.
'Y': Driver's desk in position ready, do not update TC.

8.9 Telegram type n(small letter n): Trainnumber to TC.

Telegram is sent to TC, when a trainnumber is input, regardless of change and regardless of receipt of answer from TC(line failure). The Telegramme consist of a single datapacket type T, containing train running number. Datapacket type T is defined in 8.3. Telegramme is used by DSB TRIT, where an receipt of trainnumber every time telegram 'O' is sent to TC is unwanted. An update of TRIT is only wanted when an input is received from the MSR3 Control Box.
Telegramme 'n' is sent after telegramme 'O' updating TC, that is until receipt of "Accepted" or repeated 3 times ("no answer" or "Rejected")

Appendix 8.1
Telegram contents MSR3-TC

Telegram type	A	N	r	t
Start pad	LF	LF	LF	LF
Byte counts	Byte count 1	Byte count 1	Byte count 1	Byte count 1
	Byte count 2	Byte count 2	Byte count 2	Byte count 2
Telegram type	A	N	r	t
	Telegram serial No.	Telegram serial No.	Telegram serial No.	Telegram serial No.
Packet type	A	N	B	B
Data packet	data A	data N	data B	data B
Checksum bytes	Checksum 1	Checksum 1	Checksum 1	Checksum 1
	Checksum 2	Checksum 2	Checksum 2	Checksum 2
End pad	CR	CR	CR	CR
	Approved	Rejected	Request for specific data	Idling data
	TC to MSR3	TC to MSR3	MSR3 to TC	MSR3 to TC

Telegram type	K
Start pad	LF
Byte counts	Byte count 1
	Byte count 2
Telegram type	K
	Telegram serial No.
Packet type	F
Data packet	data F
	data F
	data F
Packet type	D
Data packet	data D
	data D
Packet type	C
Data packet	data C
	data C
Packet type	M
Data packet	data M
	data M
Packet type	V
Data packet	data V
	data V
	data V
Packet type	P
Data packet	data P
	data P
	data P
	data P
Checksum bytes	Checksum 1
	Checksum 2
End pad	CR
	Actual data
	TC to MSR3

Telegram type	O
Start pad	LF
Byte counts	Byte count 1
	Byte count 2
Telegram type	O
	Telegram serial No.
Packet type	F
Data packet	data F
	data F
	data F
Packet type	D
Data packet	data D
	data D
Packet type	C
Data packet	data C
	data C
Packet type	M
Data packet	data M
	data M
Packet type	V
Data packet	data V
	data V
	data V
Packet type	P
Data packet	data P
	data P
	data P
	data P
Packet type	T
Data packet	data T
	data T
	data T
	data T
	data T
Checksum bytes	Checksum 1
	Checksum 2
End pad	CR
	Actual data
	MSR3 to TC

Telegramtype	n
Startpad	LF
Bytecount	Bytecount 1
	Bytecount 2
Telegramtype	n
	Telegram serial No.
Packettype	T
Datapacket	data T
	data T
	data T
	data T
	data T
	data T
Checksum bytes	Checksum 1
	Checksum 2
Endpad	CR
	Trainnumber
	MSR3 to TC

Telegram type	o
Start pad	LF
Byte counts	Byte count 1
	Byte count 2
Telegram type	o
	Telegram serial No.
Packet type	S
Data packet	data S
Data packet	Data S
Checksum bytes	Checksum 1
	Checksum 2
End pad	CR
	Fault code
	MSR3 to TC

9.0 TELEGRAM CONTENTS ATC - HLOG

This section comprises a description of the telegram contents between ATC and HLOG.

9.1 Telegram contents ATC and HLOG

The primary purpose of this connection is to secure that the necessary data to analyze an accident, if any, is possible. Especially the last km before the accident are considered. For the exact contents of each Telegram type reference is made to Appendix 1 in this section.

9.2 Telegram type 2: Train data

Dette Telegram is sent after approved and signed encoding of train data. In vehicles with TC train data can be offered from this one, but for these train data receipt will be made by the driver, and they will be sent on immediately to HLOG.

Packet type L: Train length

L	*	*	*

Data are in the interval from 30 to 960 m with gaps of 10 m.

Packet type %: Braking percentage

%	*	*	*

Data are in the interval from 030 to 230%.

Packet type H: Max. pace

H	*	*	*

Data are in the interval from 10 km/h to 200 km/h with gaps of 10 km/h.

Packet type R: ATC direction

R	*	*

Data: 8 8 = direction A
0 0 = direction b

Packet type J: Wheel diameter

J	*	*	*	*

Data are in the interval from 0 mm to 9999 mm.

9.3 Telegram type 3: Speed information

The Speed telegram contains the momentary speed registered by ATC as well as the monitoring speed monitored by ATC at the moment of transmission.

This information only serves as a supplement to the momentary speed measured by HLOG itself as well as registration of the monitoring speed at the moment of transmission.

The Telegram is sent cyclically and consequently acts as Line monitoring. This gives a time dependent pattern, with which it is possible to produce a braking curve.

Packet type K: Momentary speed

K	*	*	*

Data are in the interval from 000 to 254 km/h.

Packet type G: Monitoring speed

G	*	*	*

Data are in the interval from 000 to 254 km/h.

9.4 Telegram type 4: Transponder information

This Telegram is sent immediately after passage of a transponder.

Packet type a to h: Transponder information

h	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

Data have been coded as per the specification laid down between Banestyrelsen and Siemens A/S.

9.5 Telegram type G: Fault code

ATC fault code is sent to HLOG immediately after detection.

Packet type E: Fault code

E	*	*	*

Data are in the interval from 000 to 999.

9.6 Telegram type H: HKT speed information

Cf. HKT-Appendix.

9.7 Telegram type a: Operational telegram

There are two types of operational telegrams. Either Telegram type a is sent with Packet type I or with Packet type n. Both operational telegrams contain 2 data bytes.

Packet type I: Operational function

This Packet type contains 2 data bytes. Operational telegram is sent with Packet type I, when a key has been activated.

I	*	*

Data are in interval from 01 to 40 and state a position number for an activated ATC key. Key position numbers:

01	:	---	17	:	LOOSE ATC
02	:	---	18	:	---
03	:	---	19	:	---
04	:	---	20	:	---
05	:	PASS. STOP	21	:	RECEIPT
06	:	TRAIN LENGTH	22	:	---
07	:	BRAKING %	23	:	---
08	:	MAX. PACE	24	:	---
09	:	ATC DIRECTION	25	:	---
10	:	OPERATIONAL BRAKE	26	:	---
11	:	EMERGENCY BRAKE	27	:	---
12	:	TEST	28	:	---
13	:	---	29	:	---
14	:	SHUNTING	30	:	---
15	:	EXTERNAL SIGNALS	31	:	---
16	:	---	40	:	EXECUTE

Cf. HKT-appendix for alternative position numbers in connection with HKT.

Packet type n: Operational telegram DK-/S-ATC

n	*	*

Data are in the interval from 11 to 44 and state ATC operational selector position and operational position for Danish ATC and are sent every time a change of operational selector position or a change in the ATC situation is made.

Data	Operational Selector	ATC position
11	Combined system (S-ATC priority)	Active and initialized
12		Monitors without section data
13		Monitors with section data
14		F position
21	Combined system (DK-ATC priority)	Active and initialized
22		Monitors without section data
23		Monitors with section data
24		F position
31	S-ATC Exclusively	Active and initialized
32		Monitors without section data
33		Monitors with section data
34		F position
41	DK-ATC Exclusively	Active and initialized
42		Monitors without section data
43		Monitors with section data
44		F position

Data: 31, 32, 33, 34 og 44 are not expected to occure.

9.8 Telegram type c: Test

The purpose of this telegram to test all lamps in the driver's cab signal as well as the input panel. At starting up of ATC these are tested, there are, however, lamps in the driver's cab signal controlled by HLOG and partly by TC. When TC and HLOG receive this Telegram they are supposed to switch on resp. switch off the lamps.

Packet type m: Lamp test

m	*

Data: S = Switch off , T = Switch on lamp

9.9 Telegram type N: Rejected Telegram

N	*

Data:
1 = Checksum faults
2 = timing faults
A = application faults (data validation faults) i.e. a fully correctly decoded Telegram giving no meaning.
Intervals (ASCII 32D) = generally rejected

9.10 Telegram type 6: Receipt with speed

The Telegram is used only for positive receipt (on the lines of Telegram type A, "approved"), and uses Packet type "K" which is used also to state the momentary speed of ATC towards the event recorder, cf. Section 9.3. The speed itself consists of three decimal digits where Sp*eed 1* is the most significant digit , and Speed *3* the least significant digit. This telegram does not replace the original receipt telegram of type "A". ATC should be able to receive both Telegrams.

9.11 Telegram type 7: Passage of stop

By passage of a signal showing stop this Telegram will be sent. The signal position is identified partly by a single digit and partly by a three-figured digit, where *Signal No. 3* is the least significant digit.

Packet type S: Passage of stop

S	*	*	*	*

Data: After the S four digits are sent.
1st digit states station or track No. Data are in the interval 0 to 9. 2nd, 3rd and 4th digit state the signal No. Data are in the interval 000 to 999.

1st digit	**2nd, 3rd and 4th digit**
0 Data states a station	Data state No. of a booking office.
1 to 9 Data state the track No. in a section.	Km measurement ≤ 99,9 km: Data state measurement in hm. Km measurement > 99,9 km: Data state measurement in km.

Appendix 9.1

Telegram contents ATC-HLOG

Telegram type	A	c	N
Start pad	LF	LF	LF
Byte counts	Byte count 1	Byte count 1	Byte count 1
	Byte count 2	Byte count 2	Byte count 2
Telegram type	A	c	N
	Telegram serial No.	Telegram serial No.	Telegram serial No.
Packet type	A	m	N
Data packet	data A	data m	data N
Checksum bytes	Checksum 1	Checksum 1	Checksum 1
	Checksum 2	Checksum 2	Checksum 2
End pad	CR	CR	CR
	Approved ATC to HLOG	Request for test ATC to HLOG	Rejected ATC to HLOG

Telegram type	a
Start pad	LF
Byte counts	Byte count 1
	Byte count 2
Telegram type	a
	Telegram serial No.
Packet type	I
Data packet	data I
	data I
Checksum bytes	Checksum 1
	Checksum 2
End pad	CR
	Operational telegram ATC to HLOG

Telegram type	a
Start pad	LF
Byte counts	Byte count 1
	Byte count 2
Telegram type	a
	Telegram serial No.
Packet type	n
Data packet	data n
	data n
Checksum bytes	Checksum 1
	Checksum 2
End pad	CR
	Operational telegram ATC to HLOG

Telegram type	2
Start pad	LF
Byte counts	Byte count 1
	Byte count 2
Telegram type	2
	Telegramserial No.
Packet type	L
Data packet	data L
	data L
	data L
Packet type	%
Data packet	data %
	data %
	data %
Packet type	H
Data packet	data H
	data H
	data H
Packet type	R
Data packet	data R
	data R
Packet type	J
Data packet	data J
	data J
	data J
	data J
Checksum bytes	Checksum 1
	Checksum 2
End pad	CR

Train data

ATC to HLOG

Telegram type	4
Start pad	LF
Byte counts	Byte count 1
	Byte count 2
Telegram type	4
	Telegram serial No..
Packet type	a-h
Data packet	data a-h
	data a-h
24 bytes	data a-h
	data a-h
	data a-h
Checksum bytes	Checksum 1
	Checksum 2
End pad	CR

Transponder information

ATC tol HLOG

Telegram type	3
Start pad	LF
Byte counts	Byte count 1
	Byte count 2
Telegram type	3
	Telegram serial No.
Packet type	K
Data packet	data K
	data K
	data K
Packet type	G
Data packet	data G
	data G
	data G
Checksum bytes	Checksum 1
	Checksum 2
End pad	CR

Speed-information

ATC to HLOG

Telegram type	G	6
Start pad	LF	LF
Byte counts	Byte count 1	Byte count 1
	Byte count 2	Byte count 2
Telegram type	G	6
	Telegram serial No.	Telegram serial No.
Packet type	E	K
Data packet	data E	data K
	data E	data K
	data E	data K
Checksum bytes	Checksum 1	Checksum 1
	Checksum 2	Checksum 2
End pad	CR	CR

Fault code

ATC to HLOG

Receipt with speed

HLOG to ATC

Telegram type	7
Start pad	LF
Byte counts	Byte count 1
	Byte count 2
Telegram type	7
	Telegram serial No.
Packettype	S
Datapacket	data S
	data S
	data S
	data S
Checksum bytes	Checksum 1
	Checksum 2
End pad	CR

Passage of stop

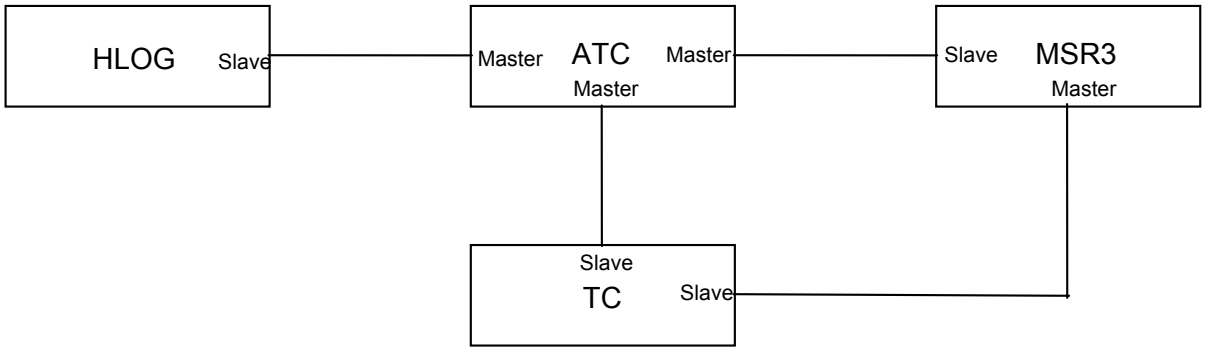
ATC to HLOG

10.0 COMMUNICATION HARDWARE

This section comprises flow charts of 20mA current loop and a description the communication hardware between ATC - TC - MSR3 - HLOG og MSR3 - TC.

10.1 Flow chart of Master - Slave

Flow chart of Master/Slave relation between ATC, TC, MSR3, HLOG og MSR3 - TC:



10.2 Description of communication hardware between ATC - TC - MSR3 - HLOG og MSR3 - TC.

The communication is realized via a 20 mA current loop with the following characteristic data:

1 start bit
8 data bit
1 parity bit, ODD parity
1 stop bit

This makes up 1 character:

start	D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	Parity	stop 1

Communication rate: 1200 baud.
Max. deviation: ± 3 % per. bit.

Signal level: It should be noted that 0 and 20 mA are the nominal values aimed at.

Sender active	low - 0 $\Rightarrow$ 0 - 2 mA
	high- 1 $\Rightarrow$ 18 - 24 mA

All values between 2 mA and 18 mA are considered "illegal" level.

Receiver low - 0 ⇒ 0 - 6 mA
high- 1 ⇒ 16 - 24 mA

All values between 6 mA and 16 mA are considered "illegal" level.

"MARK" is considered logical 1
"SPACE" is considered logical 0

Impedance: As "active" TxD can generate 20 mA over an external loading of minimum: 600 ohm.

Slave should load the Master generator with a voltage drop of max. 5 Vdc at 20 mA: 230 ohm in internal impedance.

Voltage supply:
Maximum Vcc to the 20 mA power generator is: 30 Vdc. Measured over RxD+ and RxD-.
Voltage supply to power generator is not determined as potential free, it is, however, the supplier's responsibility to secure his coupling so that it functionally can resist possible disturbances coming from the outside.

Active/passive:
ATC is considered ACTIVE towards MSR3, HLOG and TC, which are all considered PASSIVE. TC is considered ACTIVE towards MSR3, which is considered PASSIVE towards TC.

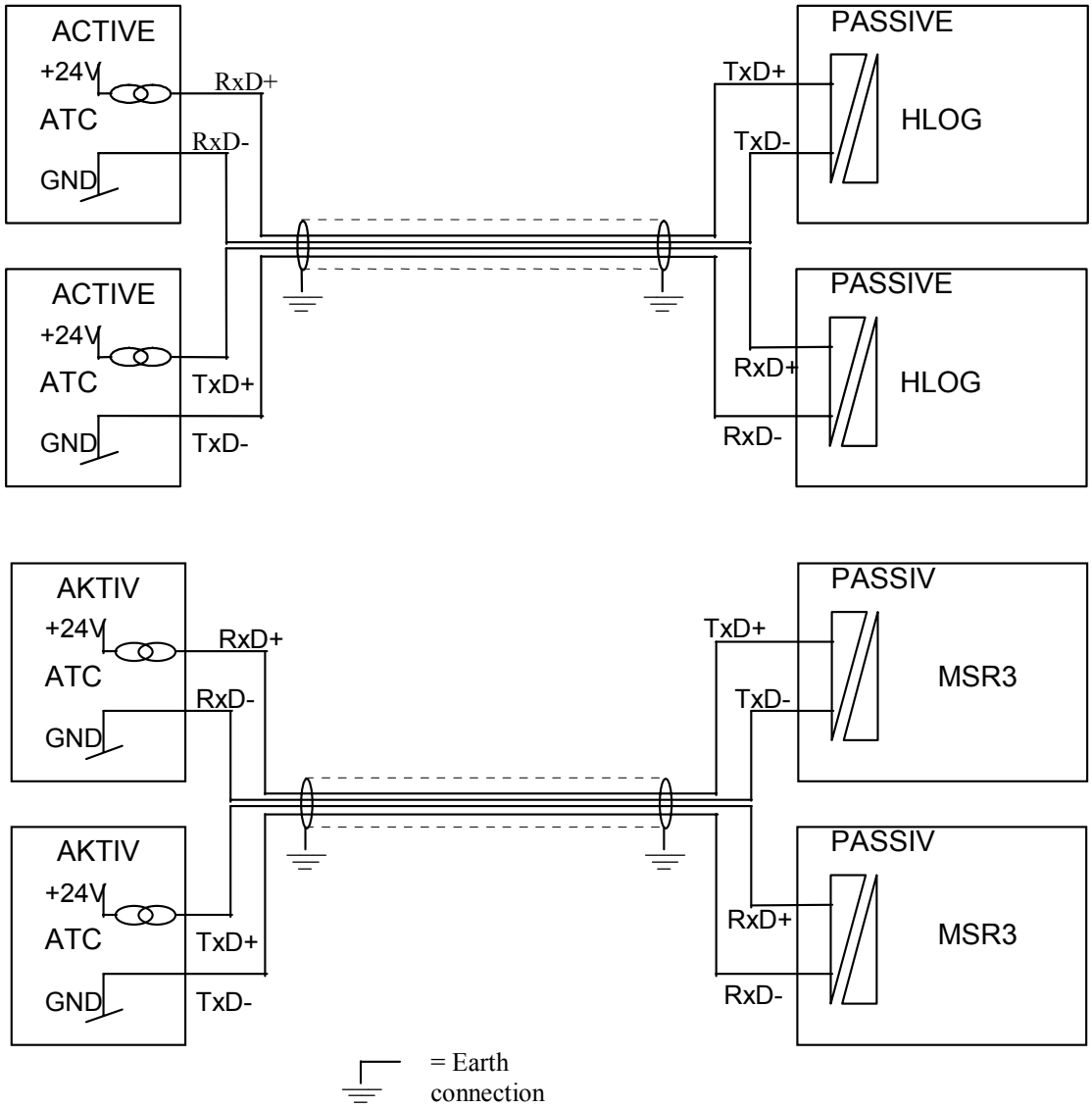
Connection:
The current loop used is a point-to-point connection exclusively coupled between 2 units, cf. Section 10.3.

The connection between ACTIVE and PASSIVE is a four-threaded double-twisted cable connection with a shield.
ACTIVE is connected with PASSIVE as described in the tables: "Flow Chart of 20 mA current loop connections". 2 pages.

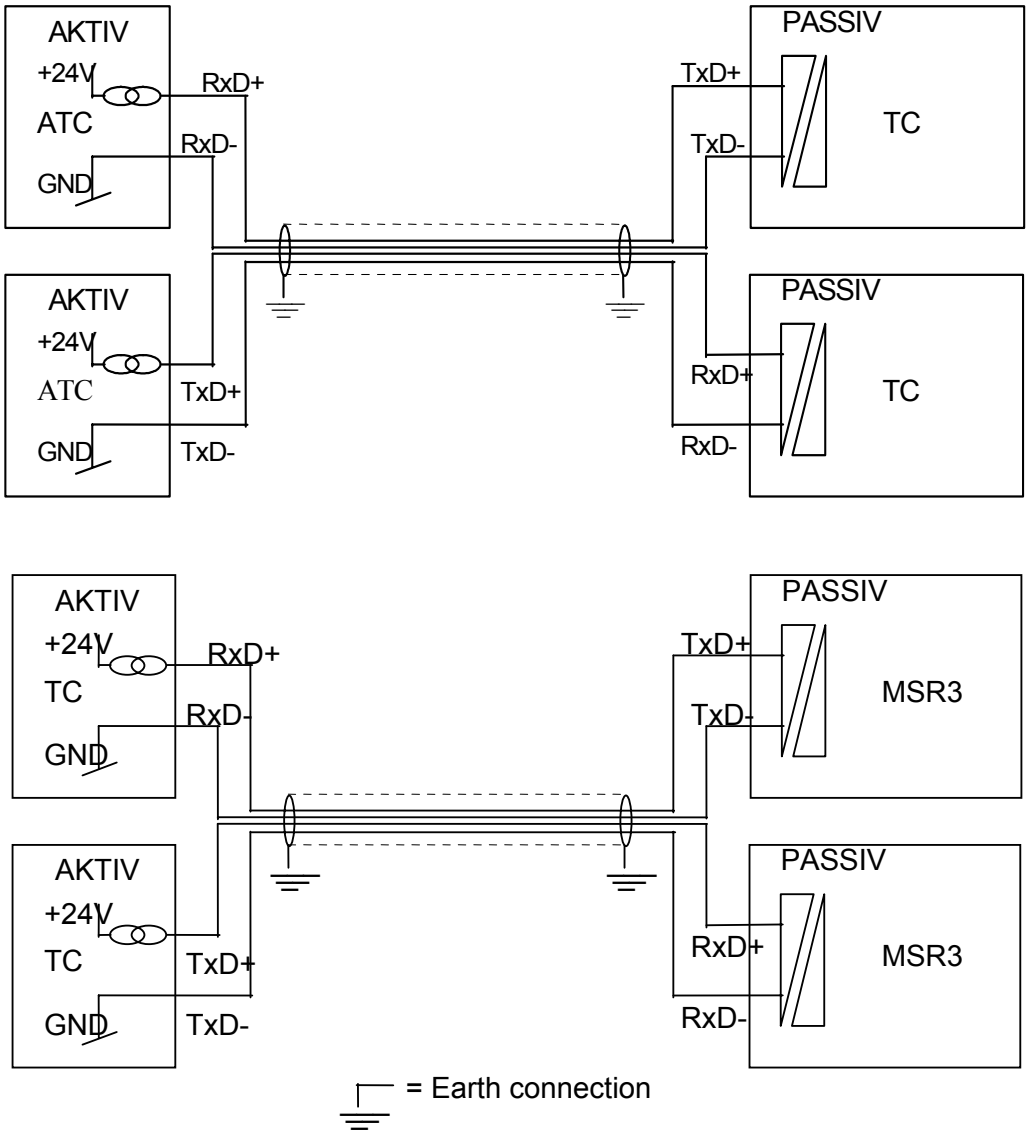
It is expected that the pin-out designation from "Flow Chart of 20 mA current loop connections" are included in the documentation, e.g. directions for use, electric diagram etc. supplied to BS.

10.3 Flow Chart of 20 mA current loop connections

Flow Chart over 20 mA current loop connections



Flow Chart over 20 mA current loop connections



11.0 AGREEMENT ON AMENDMENTS OF PROTOCOL

The following formula should be used in case of need of amendments.

<table>
<tr><td colspan="2">Agreement on amendments of protocol</td><td>Amendment No.:</td></tr>
<tr><td rowspan="3">To be filled out by proposer</td><td>Type of amendment:</td><td>Enclosures:</td></tr>
<tr><td>Name of proposer:</td><td>Responsible with proposer:</td></tr>
<tr><td colspan="2">Which installations are affected by the amendment:
☐ ATC ☐ TC ☐ HLOG ☐ MSR3 ☐ BS' users ☐ Operators' personnel</td></tr>
<tr><td>Approved by supplier</td><td colspan="2">Amendment approved by the following suppliers: (stamp, signature, date)
ATC: | TC: | HLOG:
MSR3: | Other supplier: | Other supplier:</td></tr>
<tr><td>Approved by Banestyrelsaen</td><td colspan="2">Amendment approved by Banestyrelsen: (Name, signature, date)
Department: _____
Department: _____
Amendment entered into protocol: _____</td></tr>
</table>

12.0 TELEGRAM CONTENTS HKT - HLOG (HKT-appendix.)

This appendix comprises a description of deviations from the 11 protocol sections in connection with application in S-trains.

Telegram contents HKT - HLOG

This appendix states deviations from section 1 to section 11 in connection with serial communication between HKT installations and HLOG. In this appendix HKT refers to LZB804/HKT.

12.1 Ad.: 9.6 Telegram type H: HKT speed information

This telegram should alone be fulfilled for supplies to S-trains.
After each change of the HKT speed information of the train telegram type H, packet type n is sent to HLOG.

Packet type n: Speed information

n	*

Data may assume the following 16 different values stating the relevant HKT pace information:

data n	HKT speed information
1	STOP VANDRET
2	80
3	60
4	50
5	Y
6	LA 70
7	100
8	70
9	120
A	LA 50
B	90
C	40
D	LA 30
E	30
F	STOP FALD
0	TRANSMISSION OMISSION

12.2 Ad.: 9.7 Telegram type a: Operational telegram

Packet type I stating position Nos. for keys is in connection with HKT used as follows:

1. "06, TRAIN LENGTH", "07, BRAKING%", "08, MAX,. SPEED", AND "09, ATC RETNING" are values which are not encoded in connection with HKT where defined default values are always used.

2. "17 LØS ATC" is in connection with HKT "LØS".

Appendix 12.1

Telegram contents ATC-HLOG

Telegram type	**H**
Startpad	LF
Byte counts	Byte count 1
	Byte count 2
Telegram type	H
	Telegram serial No.
Packet type	n
Datapacket	data n
Checksum bytes	Checksum 1
	Checksum 2
End pad	CR

HKT to HLOG