

FlexRay vid användningen av multimedia

Björn Ejvald
Handledare: Jerker Björkqvist
Åbo Akademi University
4/4/2011

Referat

Dagens bilar består i allt högre grad av elektroniska moduler och stödsystem. Förutom ökningen av grundläggande moduler ökar också efterfrågan på funktioner i form av audio- och videotillbehör. Denna avhandling undersöker principerna för kommunikationsbussen FlexRay samt om den vore ett bra alternativ för att utnyttja tillsammans med multimedieenheter.

Nyckelord: FlexRay, CAN, Multimedia, MOST, TTEthernet

INNEHÅLL

1. Inledning.....	1
2. Föregångaren till FlexRay	2
2.1. CAN-bussen.....	2
3. FlexRay och dess arkitektur.....	4
3.1. Klusternivå.....	4
3.2. OSI-nivå och användningsområden	5
3.3. Protokollet.....	7
4. FlexRay i användning.....	10
4.1. FlexRay i jämförelse med dess alternativ	10
4.2. Multimedia och dess krav	14
5. Slutsatser	15
Litteraturförteckning.....	16

1. INLEDNING

I dagens bilar finns det en ökande mängd datorer. Upp till 70 datorer med 2500 signaler per sekund skickandes emellan varandra är inte ovanligt. Dessa datorer och moduler behöver vanligtvis kommunicera med varandra under förhållanden med höga störningsfaktorer. Data som skickas är vanligtvis också kritiskt prioriterade: om fel uppgifter gällande hastigheter eller pedaltryck kommer fram till de mottagande enheterna, kan det resultera i katastrofala konsekvenser. Därför krävs särskilda lösningar för en fungerande kommunikation i dylika omgivningar.

Före kommunikationsbussen CAN (Control Area Network), skapades 1986, kopplades komponenter *point-to-point* med varandra; detta resulterade i dyra system med en opraktisk och dyr mängd kablage. CAN-bussen är designad för fordon i vilken den möjliggör moduler att kommunicera med varandra utan en central värd, i hastigheter upp till 1 Mbit/s. Utnyttjandet av CAN-bussen i ett fordon minskar på kabelmängden, komplexiteten, vikten och i slutändan även kostnaden. År 1993 standardiserades CAN-bussen av ISO och används på många håll även idag.

En av efterträdarna till CAN, FlexRay, skapades år 2000 av The FlexRay Consortium med samarbetspartners såsom Volkswagen AG, Bosch, General Motor, BMW Group och DaimlerChrysler. Syftet var att klara av nästa generations bilsystem där *x-by-wire* (se avsnitt 3.1) skulle få en större betydelse. De marknadsförande egenskaperna för FlexRay är dess robusthet, feltolerans och framförallt ökade kapacitet i förhållande till CAN, med hastigheter upp till 20Mbit/s.

Icke-kritiska moduler i bilen börjar också bli allt vanligare. Audiosystem, touch-screens, GPS och även videouppspelning är tillbehör som mer eller mindre börjar tillhöra standardutrustningen. I ett hemmanätverk är det möjligt att använda Ethernet för att dessa apparater skall kunna kommunicera med varandra, men i bilar är störningsfaktorn vanligtvis för hög för att vara praktiskt användbart. Den här kandidatavhandlingen kommer undersöka hur bra FlexRay fungerar tillsammans med multimedia i bilsammanhang och göra en jämförelse med några alternativ.

2. FÖREGÅNGAREN TILL FLEXRAY

2.1.CAN-BUSSEN

Control Area Network eller CAN-bussen utvecklades 1986 av Robert Bosch, då av Mercedes fanns ett behov för ECU:er (*Electronic Control Unit*) att kommunicera med varandra till låg kostnad (1). CAN uppnådde snabbt ISO-standard och är använt av ett stort antal företag än idag. Med ett maximalt meddelandefält på 64 bitar så är antalet paket många och små till storleken, vilket lämpar sig utmärkt för kommunikation inom fordon. CAN har även starka funktioner för felkorrigeringar vilket resulterar i att meddelandena med hög sannolikhet kommer fram med sitt ursprungliga innehåll.



FIGUR 1 STANDARD-CAN MED 11-BIT IDENTIFIERARE

Nätverket i CAN är uppbyggt som en seriebuss där meddelanden skickas till alla noder samtidigt. Meddelandena skickas vid behov och är inte ämnat exklusivt för periodiskt bruk; därför kan man då klassa CAN som ett händelsedrivet protokoll (2). Den första versionen av CAN klarade av hastigheter upp till 125 kbps och är numera känt som låghastighets-CAN. CAN 2.0(A/B) standardiserades av ISO år 1993 och klarar av hastigheter upp till 1Mbps och är därför då känt som höghastighets-CAN (2). En CAN-frame

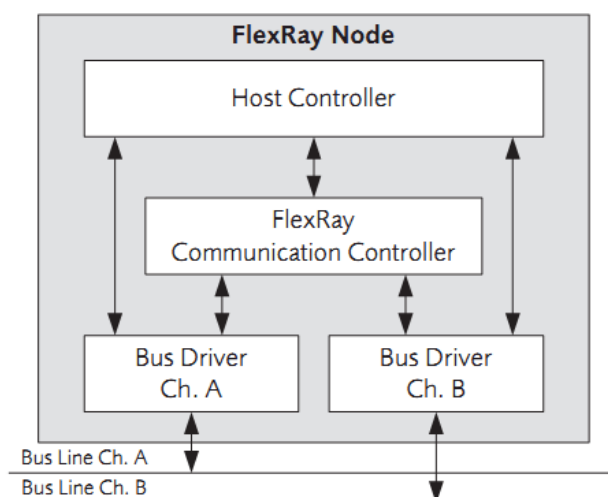
är indelad enligt Figur 1. Identifieraren bestämmer datans prioritet där ett lägre nummer innebär en högre prioritet. I standard-CAN är identifieraren 11 bitar lång medan den i utökade CAN (Extended CAN 2.0B) är 29 bitar lång, vilket ger ungefär 262 000 gånger fler identifierarmöjligheter. Låghastighets-CAN klarar av ett nätverk med 2 till 20 noder, medan höghastighetsvarianten klarar av upp till 30 noder (1). Det är också vanligt att köra CAN i hastigheter under 125 kbit/sek då felkorrigering är möjligt även om bussen lider av skrivfel. Systemet kallas då feltolerant låghastighets-CAN (*Fault Tolerant Low-speed CAN*).

Kommunikationen i CAN-bussen sker via CAN-ramar (se Figur 1). En CAN-ram är uppdelad i 11 distinkta fält. SOF-fältet, *Start-Of-Frame*-fältet, indikerar ramens början och används även för att hålla den mottagande noden synkron. *End-Of-Frame*- eller EOF-fältet indikerar ramens slut och erbjuder även bit-stuffingsfunktionalitet. Datafältet rymmer upp till 64 bitar med information, vilket resulterar i att endast små meddelanden skickas i taget. DLC (*Data-Length-Code*). CRC- (Cyclic Redundancy Check) och ACK-fältet används för att metadata, felkorrigeringar samt för att meddela nätverket om att ramen levererades framgångsrikt (1).

3. FLEXRAY OCH DESS ARKITEKTUR

3.1. KLUSTERNIVÅ

Ur nätverksperspektiv så refereras en elektronisk enhet kopplad till en buss som nod eller stjärna. Alla noder på samma buss tillhör ett kluster. Den grundläggande arkitekturen för en FlexRay-nod (se Figur 2) består först och främst av en *host-controller*, vilken är en mikroprocessor som kör nodens mjukvara. Kontrollern är i sin tur länkad till, alternativt integrerad med, kommunikationskontroller (*communications-controller*) som kör FlexRay-protokollet (3). Beroende på om det är en konfiguration med en enkelkanal eller dubbelkanal, så är de båda enheterna länkade till en eller två *bus-drivers*. Bus-drivern agerar elektriskt gränssnitt till bussen.



FIGUR 2 – ARKITEKTUREN FÖR EN FLEXRAY-NOD

FlexRay stöder ett flertal nätverkstopologier. Den vanligaste användningen är *multi-drop*-bussen som används av CAN och LIN-bussarna (Local-Interconnect-Network, diskuteras ej i denna avhandling), men det finns även möjlighet för stjärntopologier, samt hybrider av de två topologierna (4). I en multi-drop-buss kopplar man flera ECU (Electronic Control Unit) på samma buss, utan enskilda noder emellan. Nätverket blir relativt enkelt att designa och kostnadmässigt billigt, men dock så kan det resultera i långa kablar vilket ökar risken för störningar. I en stjärntopologi agerar en enhet som

aktiv nod och delar ut information till de andra noderna vilket minskar på kabellängderna och därmed även störningar. Med stjärntopologier är det även möjligt att dela upp nätverket så att ifall ett segment bryts så kan de andra fortsätta på egen hand, dock är en stjärntopologi svårare att implementera programmeringsmässigt och ökar därför också på slutkostnaderna. Med hybrider (se Figur 3) kan man få det bästa av de båda världarna: man kan designa långa relativt billiga nät, med stjärntopologier vid de mer kritiska områdena.



FIGUR 3 – EN HYBRID-TOPOLOGI

3.2.OSI-NIVÅ OCH ANVÄNDNINGSSOMRÅDEN

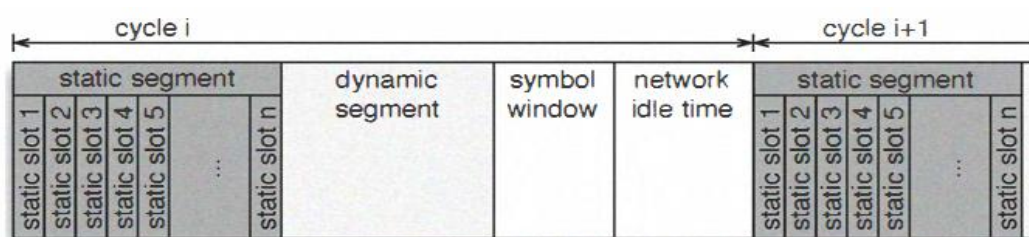
Ur mjukvaruarkitektonisk synvinkel är FlexRay uppdelat i fyra lager. Det högsta lagret '*Application Layer*' eller applikationslagret, används för användarprogrammerade applikationer, '*Presentation Layer*' eller presentationslagret ansvarar för ramstatushantering och även funktioner såsom maskningar och filtreringar. Överföringslagret eller '*Transfer Layer*' tar emot ramar och processerar dem vidare till presentationslagret. Överföringslagret agerar som kärnan för protokollet och hanterar inramningen av meddelanden och fel-detekteringar. Det fysiska lagret (*Physical Layer*) är det lägsta enligt ISO/OSI och definierar hur signaler ska överföras (1). Det fysiska lagret fungerar bra att koppla till partvinnade kablar på ungefär 100Ω , då man vill göra migreringar från tidigare existerande system (3).

Den viktigaste uppgiften för kommunikationsbussen i en bil är att hålla igång de nödvändiga systemen. Vanligtvis delar man in dessa i åtta kategorier: Chassisystem (ESP, ABS etc.), airbag-system, drivlinesystem (motor, kardanaxel etc.), komfortelektronik (fönsterhissar, cruise control etc.), multimediesystem (GPS, video, röststyrning etc.), trådlösa system (anti-stöldfunktioner, GPS), diagnostiska system (underhåll, utsläppsövervakning etc.) och *x-by-wire*. X-by-wire innebär att man byter ut exempelvis hydrauliska system mot elektroniska motsvarigheter, såsom pedalerna vid brake-by-wire eller ratten vid steer-by-wire, vilket börjar ha ökat i popularitet bland biltillverkare under det senaste årtiondet (5).

Dessa kategorier har alla i sin tur egna krav på kommunikationsegenskaper. X-by-wire har exempelvis höga krav på determinism och feltolerans, möts inte kraven kan följderna bli katastrofala (5). Komfortelektroniken å andra sidan har inga krav på någondera, om systemet inte fungerar så blir det i värsta fall ett mindre irritationsmoment. Multimedia kräver inga speciella kritiska krav på varken determinism eller feltolerans.

3.3.PROTOKOLLET

En kommunikationscykel i FlexRay (se Figur 4) är indelad i ett statiskt (ST) segment, ett dynamiskt (DYN) segment, ett symbol-fönster samt inaktivitetstid. Kommunikationscykeln upprepas periodiskt. Varje segment är indelat i luckor (*slots*) där varje lucka består av en FlexRay-ram (*frame*), vilka i sin tur består av FlexRay-protokollets minsta beståndsdel: *macroticks*. Synkroniseringen av macroticks sker på klusternivå så att data tas emot samtidigt i varje nod (4).



FIGUR 4 - EN FLEXRAY-KOMMUNIKATIONSCYKEL

Luckorna i ST-segmentet är av identiska och statiskt bestämda längder d.v.s. ett konstant antal macroticks, till för att överföra information om deterministiska händelser, periodiska händelser samt händelser med hårda deadlines. Enligt denna beskrivning kan man klassa FlexRay som ett tidsdrivet (*time-triggered*) protokoll. Dessa typer av protokoll lämpar sig för periodiska jobb men lider vanligtvis av dåligt utnyttjande av bandbredden p.g.a. antalet synkroniseringsfunktioner. Om en uppgift vill skicka ett meddelande måste den använda sin på förhand bestämda ST-lucka under den tid som systemet bestämt att den har möjlighet. Väljer noden att inte använda sin tur i den nuvarande cykeln blir den tvungen att vänta på nästa cykel för att få överföra informationen; luckan i segmentet förblir tom utan möjlighet att fyllas upp av någon annan nod (6).

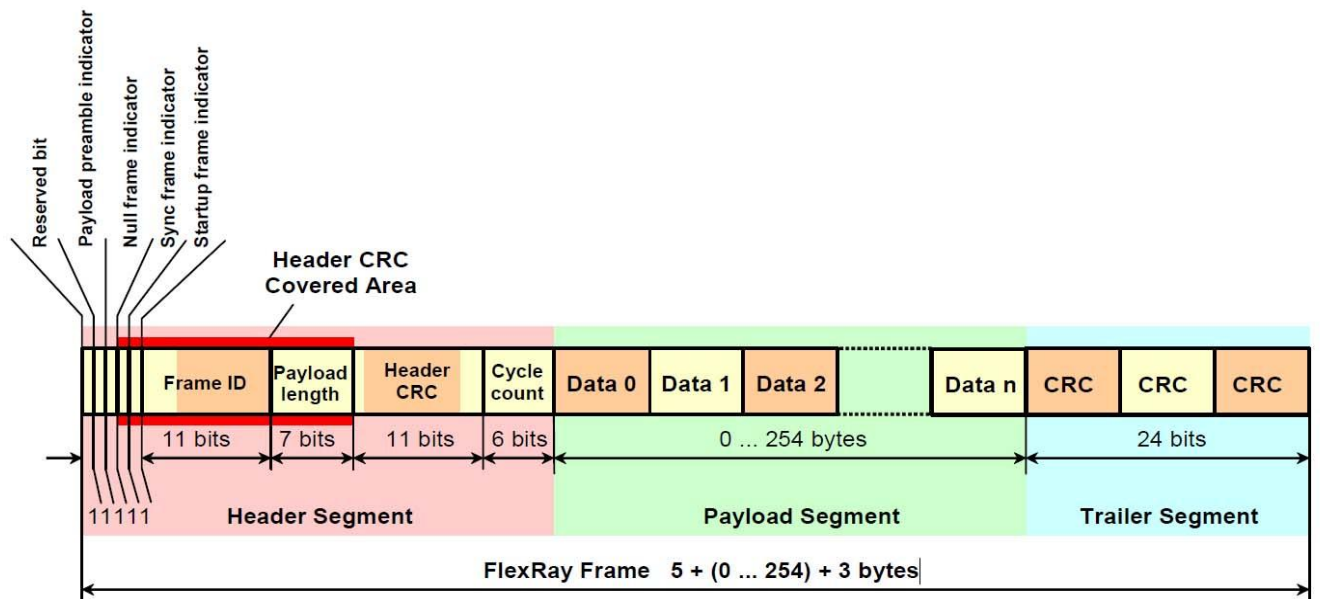
Det dynamiska segmentet är uppdelat i miniluckor, där ett varierande antal miniluckor kan utnyttjas av de kommunicerande noderna. Varje minilucka är av identisk längd ifråga om macroticks. Enheter har endast tillgång till DYN-segmentet om informationen är tillräckligt högt prioriterad. Dock så

prioriteras informationen i det statiska segmentet högre än informationen i det dynamiska gällande tidskritiska sammanhang (6). Det dynamiska segmentet överför tids- och händelsebaserad information vilket resulterar i en hög flexibilitet. Detta ger möjlighet för FlexRay att agera som ett händelsedrivet (*event-triggered*) protokoll likt CAN. Trots denna flexibilitet är de flesta implementationer av FlexRay ändå designade att främst utnyttja ST-segmentet (6).

I symbolfönstret kan en enskild symbol skickas, oftast i underhållssyften eller för att signalera uppstart av nätverket. Inaktivitetsfasen ger systemet tid för underhåll, i form av synkronisationskalkyler, synkronisering samt även felkorrigeringar.

På nivån nedanför kommunikationscykeln finns FlexRay-ramen (se Figur 5) som sköter överföringen i protokollet. En FlexRay-ram är indelad i tre segment: Ett *header*-segment, ett *payload*-segment och ett *trailer*-segment överförda kronologiskt från vänster till höger enligt Figur 5. Header-segmentet är uppdelat i 5 bytes. Den första byten innehåller 5 bitar namngivna enligt följande: *Reserved bit*, *Payload Preamble Indicator*, *Null Frame Indicator*, *Sync Frame Indicator* och *Startup Frame Indicator*. De första två bitarna, *Reserved bit* och *Payload Preamble Indicator*, är för underhållssyften samt för eventuella framtida funktioner. *Null*- och *Sync*-bitarna informerar om ramen ifråga är en tom (*null*) ram eller en synkroniseringsram. *Startup*-biten talar om ifall ramen är en startup-frame vilka används i uppstartsmekanismer då noderna inte ännu är synkroniserade. *Frame ID* består av 11 bitar och informerar om vilken lucka ramen kommer att hamna i. De två sista fälten, *Header CRC* och *Cycle Count* på 11 respektive 6 bitar, används för felkorrigeringar. Den huvudsakliga informationen som överförs i FlexRay-protokollet finns lagrat i *payload*-segmentet även känt som meddelandefältet, vilket är 0 till 254 bytes stort (nästan 32 gånger större än motsvarande CAN) uppdelat i 0 till 127 ord på två bytes styck (4) . *Trailer*-segmentet på 24 bitar innehåller endast *CRC*-information för felkorrigeringar i meddelandefältet.

Det går att se flera likheter med CAN till ramuppbyggnaden, men även många olikheter. FlexRay har satsat på determinism vilket märks i antalet synkronisationsåtgärder och en högre nivå av felkorrigering än CAN är också ett tydligt drag. För multimediebruk är oftast dock en hög bandbredd den mest åtråvärda egenskapen.



FIGUR 5 - EN FLEX RAY-RAM

4. FLEXRAY I ANVÄNDNING

4.1. FLEXRAY I JÄMFÖRELSE MED DESS ALTERNATIV

Ett alternativ som man kunde tänka sig till FlexRay, förutom de vanligaste alternativen LIN och CAN, vore TTEthernet (Time-Triggered Ethernet). Fördelar med TTEthernet är dess ökade överföringskapacitet på upp till 100Mbps, och maximal meddelandestorlek på 1500 bytes, i förhållande till FlexRay:s 20 Mbps kapacitet och meddelandestorlek på maximalt 254 bytes. Dock så är minimilängden på meddelandena för TTEthernet 46 bytes i kontrast till FlexRays 1 byte.

Förutom bandbredden är också latenstiderna av intresse. För FlexRay kan vi beräkna att en ram l_F består av överföringsstartsekvensen l_{TS} , ramstartsekvensen l_{FS} , ett *protocol-overhead* l_{PO} , ramavslutningssekvensen l_{FE} samt meddelandestorleken l_P (7). Om vi därefter antar att varje meddelandebyte kodas i 10 bitar av synkronisationsskäl och att l_{TS} är 15 bitar så går det att beräkna ramstorleken:

$$l_F = l_{TS} + l_{FS} + l_{PO} + l_P * 10 \text{ bit} + l_{FE}$$

$$l_F = 15 \text{ bit} + 1 \text{ bit} + 80 \text{ bit} + l_P * 10 \text{ bit} + 2 \text{ bit}$$

$$l_F = 98 \text{ bit} + l_P * 10 \text{ bit}$$

$$l_{Fmin} = 98 \text{ bit} + (1 * 10 \text{ bit}) = 108 \text{ bit}$$

$$l_{Fmin} = 98 \text{ bit} + (254 * 10 \text{ bit}) = 2638 \text{ bit}$$

Med bithastigheten $t_B = \frac{1}{20 \text{ Mbps}} = 0,05 \frac{\mu\text{s}}{\text{bit}}$ så kan överföringshastigheten beräknas:

$$t_T = l_F * t_B$$

$$t_{Tmin} = 108 \text{ bit} * 0,05 \frac{\mu\text{s}}{\text{bit}} = 5,4 \mu\text{s}$$

$$t_{Tmin} = 2638 \text{ bit} * 0,05 \frac{\mu\text{s}}{\text{bit}} = 131,9 \mu\text{s}$$

Räknar man med en *propagation delay* på $1,42\mu\text{s}$ i ett nätverk på två stjärnor, 72 m kablage, så blir de slutliga latenstiderna för FlexRay:

$$tT_{min} = 5,4 + 1,42 = 6,8\mu\text{s}$$

$$tT_{max} = 131,9 + 1,42 = 133,3\mu\text{s}.$$

Enligt motsvarande TTEthernet med två switchar så ger TTTech latenstiderna $24\mu\text{s}$ till $372\mu\text{s}$, d.v.s. ungefär 3,5 gånger långsammare än FlexRay (7). För tidskritiska system kan detta spela en stor roll, men vinsten för TTEthernet ligger istället i dess ökade överföringskapacitet.

Media Oriented Systems Transport (MOST) är en kommunikationsbuss vilken skapades som en lösning till de ökande kraven från multimedia- och infotainment-system år 1997 (8). MOST har inte samma nivå av robusthet eller feltolerans som FlexRay har, men klarar istället av hastigheter på 25 Mbps respektive 50 Mbps samt har en dedikerad kanal som garanterar 700 Kbps respektive 1400 Kbps asynkron kontrolldata. Även en 150 Mbps MOST-variant, MOST150, är under utveckling (9). MOST klarar av upp till 64 noder (jämför 30 för CAN). Den vanligaste implementerade nätverkstypen för MOST är en ringtopologi med optisk fiber.

TABELL 1- VANLIGA ANVÄNDNINGSMRÅDEN (5)

Användning	CAN	FlexRay	MOST
Chassi	JA	NEJ	NEJ
Airbags	JA	NEJ	NEJ
Drivlina	JA	VISS MÅN	NEJ
Komfort	JA	NEJ	NEJ
X-by-wire	VISS MÅN	JA	NEJ
Multimedia	NEJ	NEJ	JA
Trådlöst/telefoni	NEJ	NEJ	NEJ
Diagnostik	JA	VISS MÅN	VISS MÅN

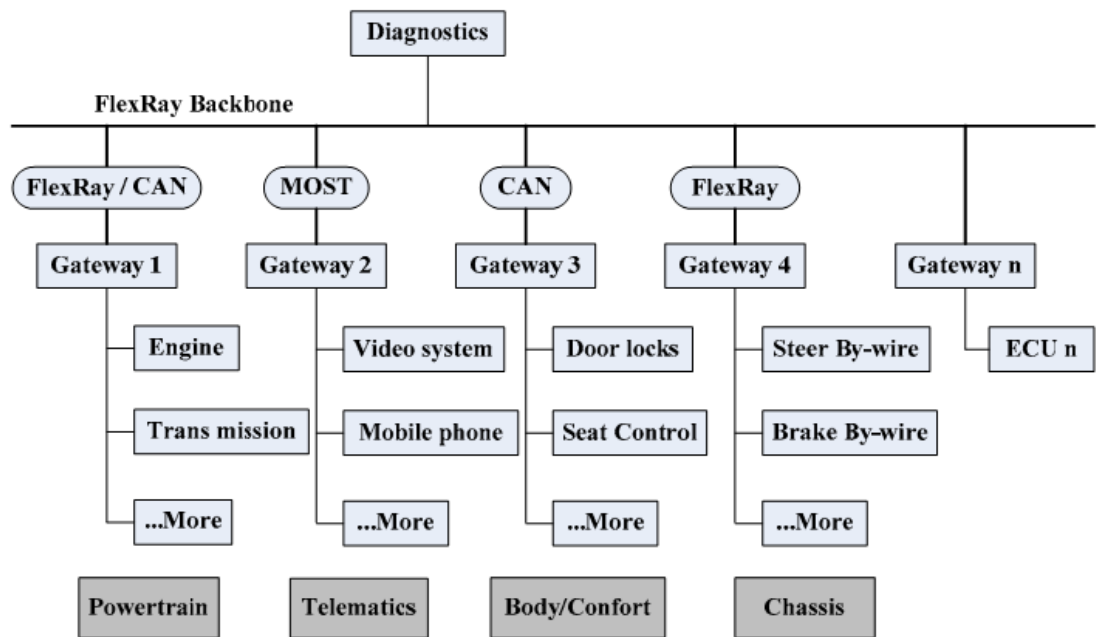
En flaskhals för nätverk i bilar är att en minimal mängd kablage är åtråvärd för att få ner komplexitet och kostnad. Som ett alternativ kan man ersätta standardbussarna med *Power-Line-Communication* (PLC) som utnyttjar de redan existerande strömkablarna för informationsöverföring (10). Nackdelen med detta system är känsligheten för störningar. Man har i en undersökning visat att det går att nå hastigheter upp till 50Mbps enligt ett PLC-modem som använder sig av HomePlug AV-standarden. Dock gällde denna överföringshastighet då bilen ifråga hade motorn, samt samtlig kringutrustning avstängd. Då bilen utnyttjade all elektronik i ett körande tillstånd så sjönk medelhastigheterna till ungefär 20Mbps (10). Systemet i undersökningen var också väldigt känsligt ifråga om kabeldragning, skillnader i kabeldragningarna kunde ge en skillnad på upp till 10Mbps. PLC kan alltså nå högre överföringskapaciteter än FlexRay vilket gör dessa system till ett tänkbart alternativ för multimedieanvändning.

TABELL 2- EGENSKAPER (5)

Krav	CAN	FlexRay	MOST
Feltolerans	VISS MÅN	JA	NEJ
Determinism	JA	JA	VISS MÅN
Bandbredd	VISS MÅN	JA	JA
Flexibilitet	JA	JA	JA
Säkerhet	NEJ	JA	VISS MÅN
Kostnad(Låg/Mellan/Hög)	L/M	M	H

En snabb överblick över de vanligaste använda systemen samt deras egenskaper går att ses i Tabell 1 och Tabell 2. Egenskapen säkerhet syftar på om det finns inbyggda säkerhetsfunktioner mot åtkomst utifrån, exempelvis då mot diagnostikverktyg eller trådlöst. Övriga egenskaper diskuteras i kapitel 3.1.

Ett typiskt upplägg för ett nätverk involverande FlexRay (11) kan ses i Figur 6. FlexRay agerar ryggrad (*backbone*) för hela systemet, med en relativt hög kapacitet och en hög pålitlighet. MOST utnyttjas för multimedieenheter som inte har några specifika krav på determinism o felkorrigeringar, men dock en hög bandbredd. CAN utnyttjas till de mindre datakrävande enheterna istället för FlexRay p.g.a. lägre implementationskostnader. FlexRay agerar alltså inte fullt ut som en ersättare för tidigare system, men fungerar utmärkt att integrera tillsammans med andra.



FIGUR 6 - UPPLÄGG FÖR ETT BILNÄTVERK

4.2. MULTIMEDIA OCH DESS KRAV

Multimediesystem i bilsammanhang syftar på enheter såsom GPS, kameror, touch-screens, audio- och videosystem med mera. En av de mest bandbreddskrävande typerna av multimedia i bilsammanhang är videoöverföring såsom DVD- eller BluRay-spelare. HD-filmer i upplösningen 720p och 1080p är en av de nu mest utbredda videostandarderna i dagens läge. Överföringshastigheterna kan beräknas enligt följande för en 24fps film:

$$\text{Resolution} * \text{Färg per pixel} * \text{Bildfrekvens} = \text{Datamängd}$$

$$720p \text{ Okomprimerad RGB: } 1280 * 720 * 3 * 24 \approx 66 \frac{MB}{sec} \approx 0,53Gbps$$

$$1080p \text{ Okomprimerad RGB: } 1920 * 1080 * 3 * 24 \approx 149 \frac{MB}{sec} \approx 1,19Gbps$$

Okomprimerad data är inte praktiskt möjligt att använda sig av i detta sammanhang. Om man dock använder sig av MPEG-2-komprimering kan man minska ner dessa maximala hastigheter med en faktor av 30-100:1 (12). Använder vi H.264 eller MPEG-4-formatet fås hastigheterna ner till ytterligare ungefär hälften (13). Då blir bitströmmarna approximativt 9Mbps för 720p respektive 20 Mbps för 1080p. Dock så kan man med särskilda tekniker få ner hastigheterna ytterligare; vid exempelvis IPTV kan man beräkna med en genomsnittshastighet på 4-6 Mbps för en 1080i-film (13).

Om man ser till spelindustrin som använder sig av högre resolutioner så vore det nästa logiska steget 1440p. Skillnaden kommer förmodligen inte vara lika drastiskt som steget upp till 720p/1080p och HD-TV men möjligheten finns att tillverkare kommer vilja använda sig av 1440p, om inte annat så för marknadsföring.

$$1440p \text{ Okomprimerad RGB: } 2560 * 1440 * 3 * 24 \approx 265MB/sec \approx 2,1Gbps$$

Datamängden är alltså nästan dubbelt så hög som för föregående resolution. Om vi därefter också gör ett antagande att kompressionen motsvarar de tidigare två fallen så fås en maximal hastighet på ~147 Mbps.

5. SLUTSATSER

The FlexRay Consortium avslutades år 2009 och därmed också utvecklingen av FlexRay-bussen; specifikationer finns dock fortfarande att hitta på The FlexRay Consortium:s hemsida.

Kraven på överföringskapaciteten har ökat närmast explosivt i takt med nya standarder. HDTV har satt nya krav på många områden, även så på bilindustrin. Kraftiga kompressionsalgoritmer motverkar dessa kapacitetskrav men de begränsar istället processorkraften. När man diskuterar multimedieöverföring så är CAN med sin maximala hastighet 1Mbps klart förlegat i förhållande till alternativen som diskuterats. Som det framgick i kapitel 4.1 så existerar redan kommunikationssystemet MOST för att lösa nuvarande krav och i viss mån även framtida krav på multimediebehovet i bilsammanhang. Med en begränsning på 20Mbit/s så klarar FlexRay dock av en ansevärd mängd data, tillräckligt för en tid framöver; men kan inte mäta sig med MOST150 som klarar upp till 7,5ggr högre kapacitet. Överföring av multimedia är möjligt med FlexRay, men tänkbara och möjligen kostnadseffektivare alternativ finns alltså.

För multimedia är det största kravet på nätverket överföringskapaciteten, de övriga uppgifterna sköts i noderna gällande exempelvis dataprocessering och skeduleringsupplägg. Multimedia är inte kritiskt beroende av den nivå av felkorrigering och determinism som FlexRay erbjuder, stabiliteten som kan fås vida överskrider behovet. Men gäller det extrema miljöer med höga nivåer av störning kan man se FlexRay som ett gott alternativ. Speciellt i samband med andra kommunikationsbussar vore FlexRay ett utmärkt alternativ. Kombinationer med MOST är redan vanliga, men även TTEthernet vore ett hållbart alternativ. PLC är också tänkbart, men kräver utförliga tester på störningsmoment utifrån.

LITTERATURFÖRTECKNING

1. *A Gateway System for an Automotive System: LIN, CAN, and FlexRay*. **Kim, Seung-Han, o.a.** Daejeon : IEEE, 2008. The IEEE International Conference on Industrial Informatics. ss. 967-972.
2. *Research on the Controller Area Network*. **Chen, Hanxing and Jun, Tian.** s.l. : IEEE Computer Society, 2009. International Conference on Networking and Digital Society. pp. 251-254. 978-0-7695-3635-4/09.
3. *Approaching the Limits of FlexRay*. **Heller, Christoph, et al.** Cambridge, MA : Network Computing and Applications, 2008. NCA '08. Seventh IEEE International Symposium on , 2008. 978-0-7695-3192-2 .
4. **FlexRay Consortium.** FlexRay_Protocol_Specification_V2.1_Rev.A. [Online] den 15 12 2005. [Citat: den 22 Mars 2011.] <http://www.flexray.com>.
5. *Automotive Communications - Past, Current and Future*. **Nolte, Thomas, Hansson, Hans and Bello, Lucia Lo.** Catania : Emerging Technologies and Factory Automation, 2005. ETFA 2005. 10th IEEE Conference on , 2005. pp. 985-992. 0-7803-9401-1.
6. *Performance Analysis of FlexRay-based ECU-networks*. **Hagiescu, Andrei, et al.** San Diego, CA : Design Automation Conference, 2007. DAC '07. 44th ACM/IEEE, 2007. Design Automation Conference. pp. 284-289. 978-1-59593-627-1.
7. *Comparing Time-Triggered Ethernet with FlexRay: An Evaluation of Competing Approaches to Real-Time for In-Vehicle Networks*. **Steinbach, Till, Korf, Franz and Schmidt, Thomas C.** Nancy : 2010 IEEE International Workshop on Factory Communications Systems, 2010. 978-1-4244-5460-0.
8. **Schoeters, Jurgen, et al.** *In-vehicle Movie Streaming Using an Embedded System with MOST Interface*. Warwick, UK : Automotive Electronics, 2007 3rd Institution of Engineering and Technology Conference, 2007.

9. *In-vehicle Movie Streaming Using an Embedded System with MOST Interface.* **Jurgen, Schoeters, et al.** Warwick, UK : Automotive Electronics, 2007 3rd Institution of Engineering and Technology Conference on , 2007. 0537-9989.
10. **Tanguy, Philippe, Nouvel, Fabienne and Patrice, Maziéro.** *Power Line Communication standards for in-vehicule networks.* Institute for Electronics and Telecommunications of Rennes : IEEE, 2009.
11. *Probabilistic Delay Model of Dynamic Message Frame in FlexRay Protocol.* **Kim, Bonjun and Park, Kiejin.** Las Vegas, NV : Consumer Electronics, 2009. ICCE '09. Digest of Technical Papers International Conference, 2009. 978-1-4244-4701-5.
12. *Multimedia and Multimedia Communication: A Tutorial.* **Wu, Chwan-Hwa and Irwin, J. David.** 1, Auburn : IEEE Transactions on Industrial Electronics, 1998, Vol. 45. 0278-0046.
13. *Performance Analysis of IPTV Traffic in Home Networks.* **Shihab, E., et al.** Washington, DC : Global Telecommunications Conference, 2007. GLOBECOM '07. IEEE, 2007. 978-1-4244-1043-9 .