

CAN bus

- Мрежовата мрежа на контролера (CAN bus) е стандарт за авто bus, проектиран да позволи на микроконтролерите и устройствата да комуникират помежду си в приложения без хост компютър. Това е протокол, базиран на съобщения, проектиран първоначално за мултиплексиране на електрически кабели в автомобилите, за да се спести от мед, но се използва и в много други контексти.
- Разработването на автобуса CAN започна през 1983 г. в Роберт Бош ГмбХ. Протоколът беше официално оповестен през 1986 г. на конференцията на Сдружението на автомобилните инженери (SAE) в Детройт, Мичиган. Първите CAN чипове, произведени от Intel и Philips, са пуснати на пазара през 1987 година. Издаден през 1991 г. Mercedes-Benz W140 е първият производствен автомобил, оборудван с CAN-базирана мултиплекс система

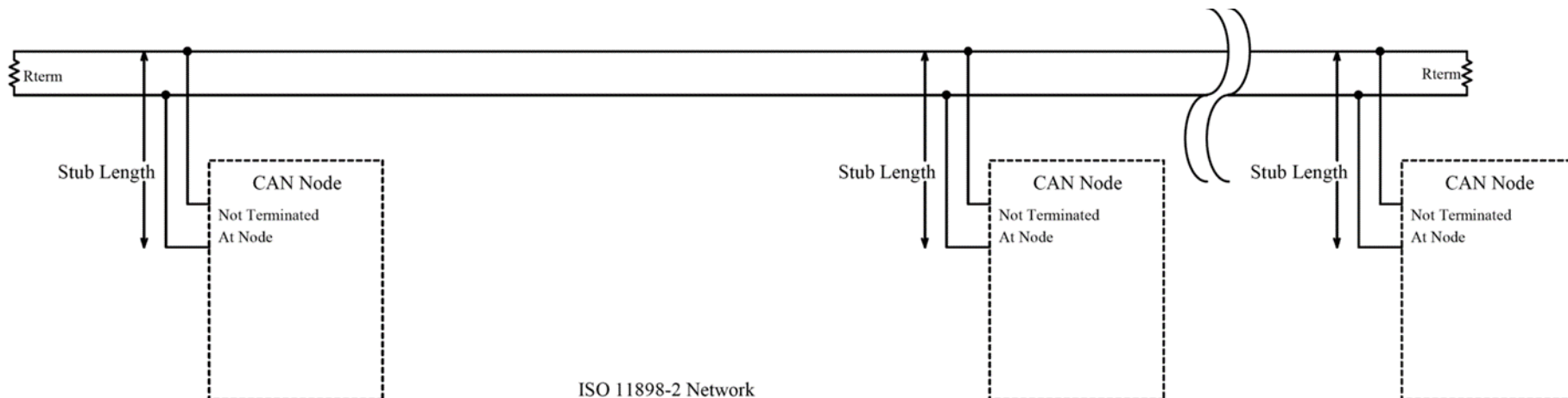
- CAN-шината е един от петте протокола, използвани в стандарта за бордова диагностика (OBD) -II диагностика на превозните средства. Стандартът OBD-II е задължителен за всички автомобили и лекотоварни камиони, продавани в САЩ от 1996 г. насам. Стандартът EOBD е задължителен за всички бензинови автомобили, продавани в Европейския съюз от 2001 г. насам, както и за всички дизелови автомобили от 2004 г. насам.
- Съвременният автомобил може да има до 70 електронни контролни блока (ECU) за различни подсистеми. Обикновено най-големият процесор е блокът за управление на двигателя. Другите се използват за трансмисии, въздушни възглавници, антиблокираща спирачка / ABS, круиз контрол, електрическо управление, аудиосистеми, електрически прозорци, врати, регулиране на огледалата, акумулаторни и презареждащи системи за хибридни автомобили и т.н. Някои от тях формират независими подсистеми, но комуникациите между другото са от съществено значение. Една подсистема може да се наложи да управлява задвижванията или да получава обратна връзка от датчиците. Стандартът CAN е разработен, за да запълни тази нужда. Едно от ключовите предимства е, че взаимосвързаността между различните системи на превозните средства може да позволи използването на широк спектър от функции за безопасност, икономичност и удобство, като се използва само софтуер - функционалност, която би увеличила разходите и сложността, ако тези характеристики бяха "твърди". Примерите включват:

- Автоматичен старт / стоп: Различните сензорни входове от автомобила (сензори за скорост, ъгъл на кормилно управление, вкл. / Изкл. На климатика, температура на двигателя) се събират чрез CAN шина, за да се определи дали двигателят може да бъде изключен, емисии.
- Електрически спирачки на паркирането: функцията за задържане на хълма взема вход от сензора за накланяне на автомобила (използван също от алармата срещу крадци) и сензорите за скорост на пътя (също използван от ABS, управление на двигателя и контрол на теглителната сила) автомобилът е спрял на наклон. По същия начин входовете от сензорите за предпазните колани (част от контролните устройства на въздушните възглавници) се подават от CAN-шината, за да се определи дали са закрепени предпазните колани, така че спирачката за паркиране автоматично да се освободи при движение.
- Системи за подпомагане при паркиране: когато водачът задейства заден ход, управляващият блок може да изпрати сигнал през CAN-шината, за да задейства както системата за паркиране, така и модула за управление на вратите на страничното огледало за пътниците, за да се наклони надолу, на бордюра. Каналът CAN също приема входове от сензора за дъжд, за да задейства задната чистачка на предното стъкло при заден ход.
- Системи за автоматично подпомагане на пътя *avoidance* / за избягване на сблъсък: входовете от сензорите за паркиране се използват също от CAN-шината, за да подават външни данни за близост до системи за помощ на водача, като например предупреждение за напускане на лентата, а в последно време тези сигнали преминават през CAN-шина за задействане на спирачката чрез тел в системи за активно избягване на сблъсък.
- Автоматично избърсване на спирачките: Входът се подава от сензора за дъжд (използван предимно за автоматичните чистачки на предното стъкло) през CAN-шината към ABS-модула, за да се предизвика невъзможно задействане на спирачките, докато се движи, за да се изчисти влагата от спирачните ротори. Някои модели Audi и BMW с висока производителност включват тази функция.

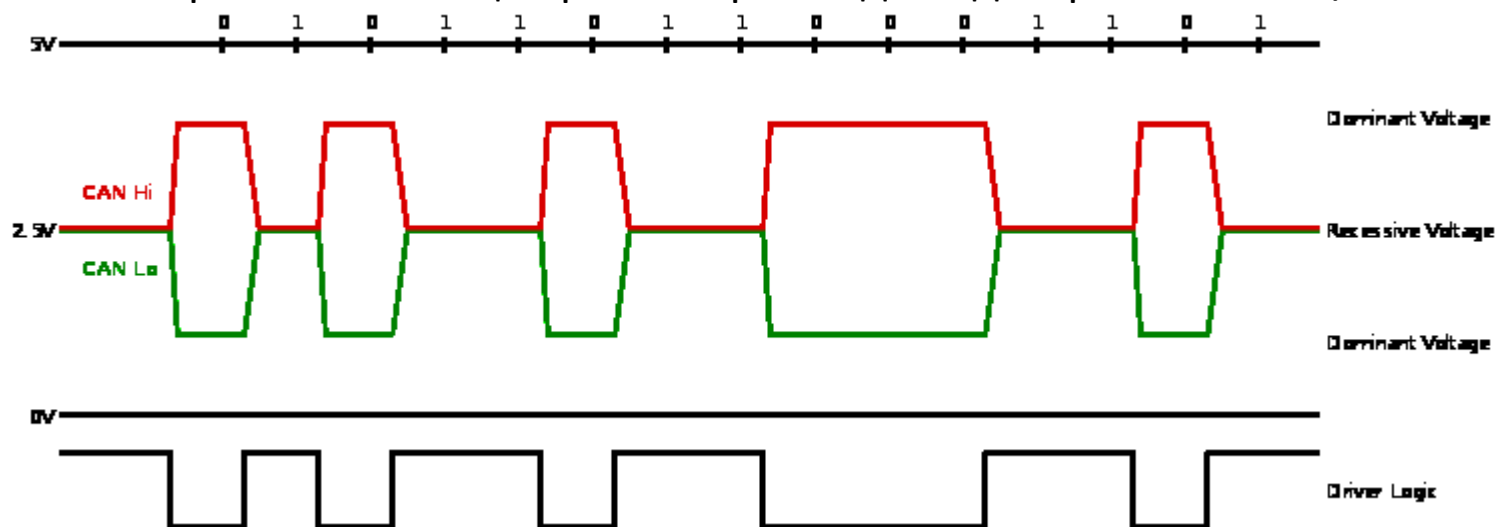
- През последните години беше въведен стандартът за LIN връзка, който допълва CAN за некритични подсистеми като климатизация и инфоразвлекателни услуги, където скоростта и надеждността на предаване на данни са по-малко критични.

Architecture

- CAN е стандартна серийна шина с множество възможности за свързване на електронни контролни блокове (ECUs), известни още като възли. Две или повече възли са необходими на мрежата CAN за комуникация. Сложността на възела може да варира от просто I / O устройство до вграден компютър с CAN интерфейс и сложен софтуер. Възелът може да бъде и шлюз, позволяващ на стандартен компютър да комуникира през USB или Ethernet порт на устройствата в CAN мрежа.
- Всички възли са свързани помежду си чрез двужична шина. Проводниците са усукана двойка с 120 Ω (номинален) характерен импеданс.
- ISO 11898-2, наричана още висока скорост CAN, използва линейна шина, завършена на всеки край с 120 Ω резистори.



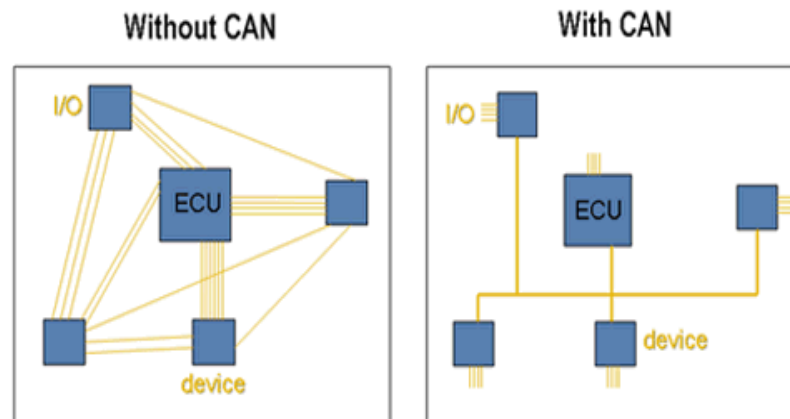
Високоскоростната CAN сигнализация задвижва високия проводник на CAN към 5 V и ниският проводник на CAN към 0 V, когато предава преобладаващо (0), и не задвижва нито един проводник при предаване на рецесивен (1). Доминиращото диференциално напрежение е номинално 2 V. Резисторът за прекратяване пасивно връща двата проводника на номинално диференциално напрежение от 0 V. Доминиращото напрежение в общ режим трябва да бъде в рамките на 1.5 до 3.5 V от обичайното и рецесивното напрежение в общия режим трябва да бъде в рамките на ± 12 от общо.



Introduction to CAN

Какво е CAN и какви са някои от неговите функции?

- Серийна комуникация
- Мулти-мастер протокол
- компактен
 - Връзка с двойка шини
- 1 мегабит в секунда



- **Защо се използва CAN?**

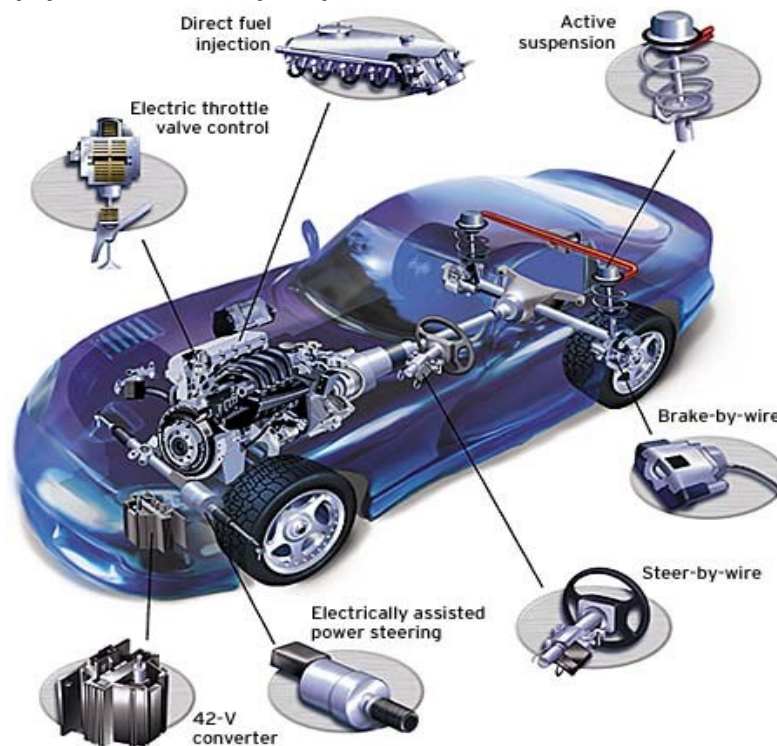
- Здрави в шумни среди
- Настройка на приоритетни сигнали
- Всички устройства в мрежата получават всяка част от информацията, изпратена на шината
- Рентабилен

Protocol	Max. Bit Rate (Mbit/s)	Max. Data Payload (bytes)	Max. Msgs/s	MAC Efficiency	Chip Cost	Development Support
CAN	1	8	8,000	High	\$6	Substantial
LonTalk	1.25	229	560	Moderate	\$5	Substantial
IEEE-1394	400	varies, 2,048 typically	800,000	High	\$30	Moderate

- Какви са някои приложения в реалния свят на CAN?

- Мрежите за контролери са използвани в много различни области, повечето от които са

- Автомобилна индустрия
- Фабрична автоматизация
- Машинен контрол
- Медицинско оборудване и устройства
- И още....



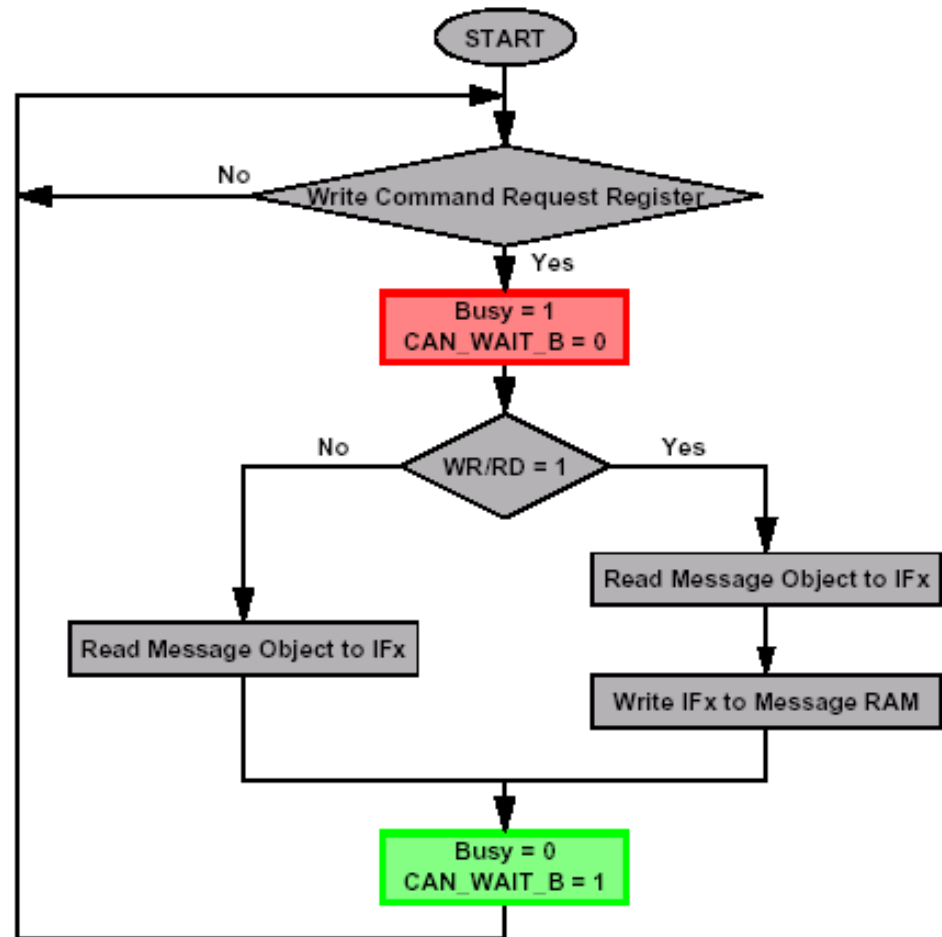
What is transmitted?

Field name	Length (bits)	Purpose
Start-of-frame	1	Denotes the start of frame transmission
Identifier	11	A (unique) identifier for the data
Remote transmission request (RTR)	1	Must be dominant (0)Optional
Identifier extension bit (IDE)	1	Must be dominant (0)Optional
Reserved bit (r0)	1	Reserved bit (it must be set to dominant (0), but accepted as either dominant or recessive)
Data length code (DLC)	4	Number of bytes of data (0-8 bytes)
Data field	0-8 bytes	Data to be transmitted (length dictated by DLC field)
CRC	15	Cyclic redundancy check
CRC delimiter	1	Must be recessive (1)
ACK slot	1	Transmitter sends recessive (1) and any receiver can assert a dominant (0)
ACK delimiter	1	Must be recessive (1)
End-of-frame (EOF)	7	Must be recessive (1)

- All messages sent over a CAN network follows this format. Each bit is used either to verify the validity of the message, or is data itself.

Какъв е процесът на изпращане на съобщение?

- На всяко устройство CAN стартът на бита на рамката съобщава, че се изпраща предаване.
- Идентификаторът бит показва приоритета на съобщението, заедно с определянето на устройството, към което принадлежат данните.



CAN Message Transmission

Basic message frame format

Field name	Length (bits)	Purpose
Start-of-frame	1	Denotes the start of frame transmission
Identifier	11	A (unique) identifier for the data
Remote transmission request (RTR)	1	Must be dominant (0)
Identifier extension bit (IDE)	1	Must be dominant (0)
Reserved bit (r0)	1	Reserved bit (it must be set to dominant (0), but accepted as either dominant or recessive)
Data length code (DLC)	4	Number of bytes of data (0-8 bytes)
Data field	0-8 bytes	Data to be transmitted (length dictated by DLC field)
CRC	15	Cyclic redundancy check
CRC delimiter	1	Must be recessive (1)
ACK slot	1	Transmitter sends recessive (1) and any receiver can assert a dominant (0)
ACK delimiter	1	Must be recessive (1)
End-of-frame (EOF)	7	Must be recessive (1)

Arbitration Field

Message Objects

Обекти на съобщенията

- 32 съобщения
- Конфигурирани за предаване или получаване или и за двете
- Конфигурирани чрез регистрите на интерфейса на обекта за съобщения
- Всеки идентификатор се съхранява в обект на съобщение.
- Номерът на съобщението е приоритетът за получаване / предаване на обектите за съобщения
 - Обект на съобщението 1 има най-висок приоритет, докато съобщението Object 32 има най-нисък приоритет

Message Object Interface register

Message Object												
UMask	Msk28-0	MXtd	MDir	EoB	NewDat		MsgLst	RxIE	TxIE	IntPnd	RmtEn	TxRqst
MsgVal	ID28-0	Xtd	Dir	DLC3-0	Data 0	Data 1	Data 2	Data 3	Data 4	Data 5	Data 6	Data 7

- **ID28-0** Message Identifier
 - ID28 - ID0 29-bit Identifier (“Extended Frame”).
 - ID28 - ID18 11-bit Identifier (“Standard Frame”).
- **Dir** Message Direction
 - one Direction = transmit
 - zero Direction = receive
- **Data 0** 1st data byte of a CAN Data Frame
- **Data 1** 2nd data byte of a CAN Data Frame

Error Field

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
res	res	res	res	res	res	res	res	Test	CCE	DAR	res	EIE	SIE	IE	Init
r	r	r	r	r	r	r	r	rw	rw	rw	r	rw	rw	rw	rw

- Enable test mode to use the modes below

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
res	res	res	res	res	res	res	res	Rx	Tx1	Tx0	LBack	Silent	Basic	res	res
r	r	r	r	r	r	r	r	r	rw	rw	rw	rw	rw	r	r

- Loop Back Mode
- **Silent** Mode
- **Basic** Mode

Status Register

- Transmit successfully
- Receive successfully
- Stuff Error -Form Error
- AckError -Bit1Error
- Bit0Error -CRCError
- Предаване успешно
- Получете успешно
- Грешка на елемента - грешка в формата
- AckError -Bit1Error
- Bit0Error -CRCError

Debugging

- Oscilloscope
 - Useful in Lab 1
 - read the message frame and error bits
 - calculate the frequency of a message frame

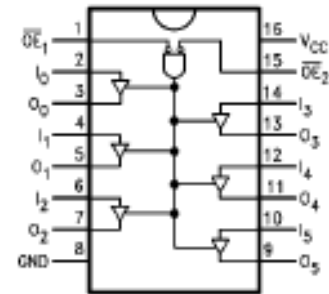
Components for Lab 1

- Slide Switch



- 74F365 Hex Buffer

Connection Diagram



- Bi color LED

