ISO9141-2 and J1939 Protocols on OBDII

黃其泮

彰化師範大學電子工程系

Outlines

- OBDII Introduction
- ISO9141-2 on OBDII
  - The Physical Layer and Data Link Layer
    - The data rate and signal level of K-Line
    - Data frame and packet structures of ISO 9141-2
  - The Mode and PID Definitions
  - The DTC Definitions
  - Application: In-Vehicle Data Logger

Outlines

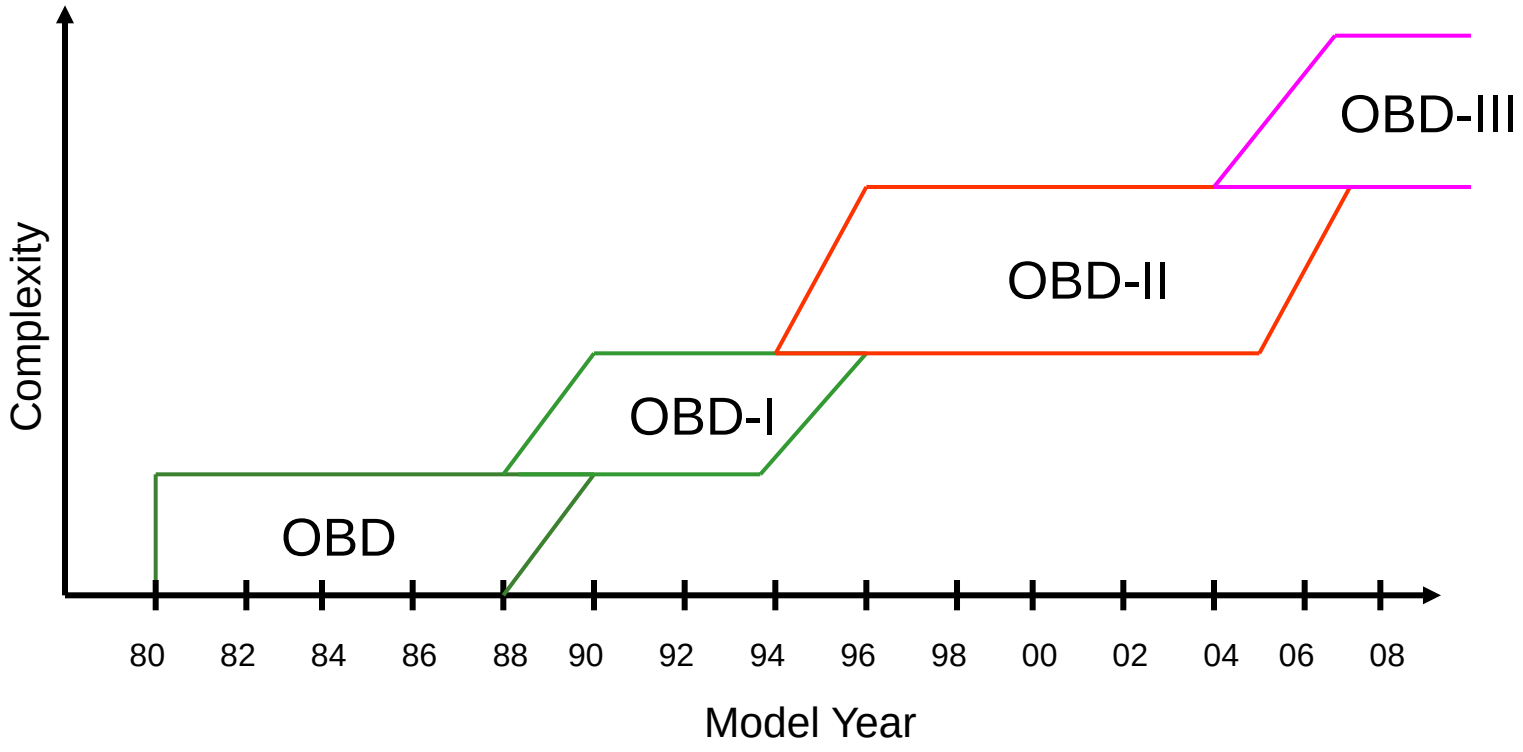
- J1939 on OBDII
  - The Physical and Data Link Layer
    - The data rates and signal levels
    - Extended identifier messages
  - The PGN (Parameter Group Number) Definitions
  - Transport Protocols
  - The DTC (Diagnostic Trouble Code) Definitions
  - Application: Parameters and Trouble Codes Monitoring and Display

OBDII Introduction

- OBD stands for On-Board Diagnostics
  - OBDII is the latest industry standard (introduced on 1996 vehicle models)
  - It provides *monitoring* of nearly all *engine controls*, and also some other parts of the vehicle (*chassis*, *body*, etc.)
  - The most useful part of OBD is the ability to *read information and trouble codes* from the vehicle

OBDII Introduction

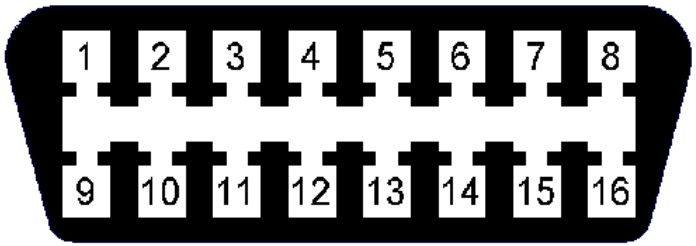
- On-Board Diagnostics (OBD) U.S. timetable



OBDII Introduction

- Pin definitions of OBDII connector

OBDII Connector



Pin 2 - J1850 Bus+
Pin 4 - Chassis Ground
Pin 5 - Signal Ground
Pin 6 - CAN High (J-2284)
Pin 7 - ISO 9141-2 K Line
Pin 10 - J1850 Bus
Pin 14 - CAN Low (J-2284)
Pin 15 - ISO 9141-2 L Line
Pin 16 - Battery Power

ISO9141-2 on OBDII

- The ISO9141 standard
  - It specifies the requirements for setting up the *interchange of digital information* between *an on board ECU* and *a suitable diagnostic tester* to facilitate *inspection*, *test*, *diagnosis* and *adjustment* of vehicle's ECUs

ISO9141-2 on OBDII

- The ECU must have one (**K**) or two (**K** and **L**) communication connections
- Connecting lines **K** or **L** from *one or more ECUs together* results in a *bus system*
- Line **K** is defined as the line that provides information in a *serial digital form* from the *ECU to the diagnostic tester*
  - Line **K** may also be *bi-directional*, in which case it may *carry commands* or *data* form the diagnostic tester to the ECU

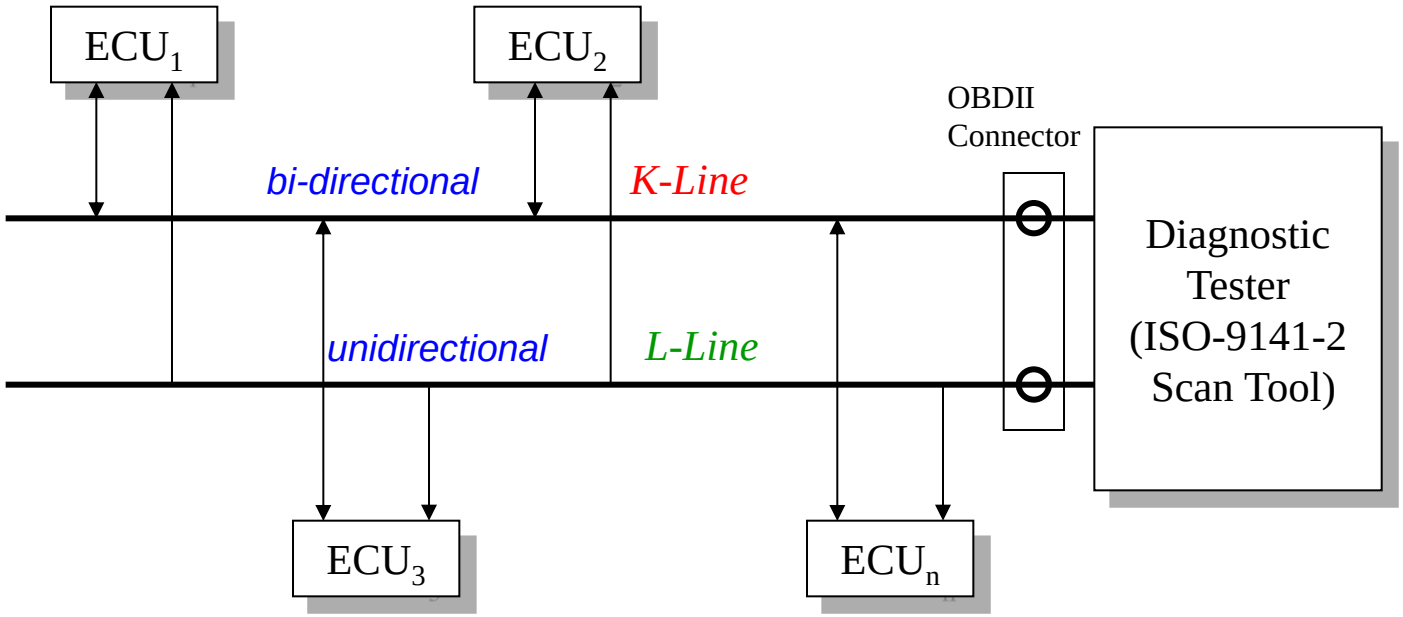
ISO9141-2 on OBDII

- Line **K** may also be used to *initialize* the serial communication
- The standard **K**-line is ISO9141 compatible for baud rates up to *250kbps*

- Line **L** is a *unidirectional* line from the diagnostic tester to the ECU
  - It may be used to *initialize* the serial communication and/or to carry commands and/or data

ISO9141-2 on OBDII

- ISO-9141-2 configuration (bus system)



ISO9141-2 on OBDII

- Signal levels of OBDII protocols

	K-Line(ISO-9141)	CAN Bus(J1939)	J1708/J1587
Bit rate	10400bps	250k/500k/1Mbps	9600bps
Logical level	Single-end: 0 ~ V_{BAT}	Differential: 0 ~ 5V	Differential: 0 ~ 5V
Wiring	Single wire	Twist wire	Twist wire

ISO9141-2 on OBDII

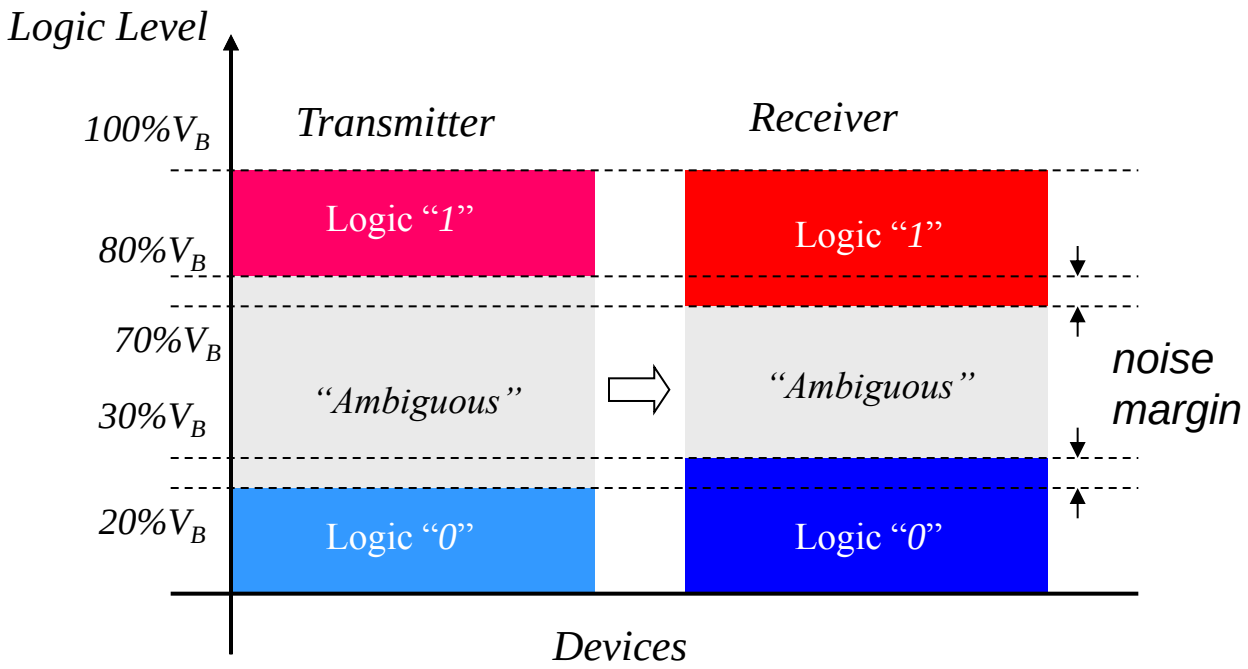
- ISO 9141-2 describes a subset of ISO9141
  - It specifies the requirements for setting-up the interchange of digital information between *on-board emission-related ECUs* or *road vehicles* and the *SAE OBDII scan tool*
  - It is limited to vehicles with nominal *12 V* supply voltage

ISO9141-2 on OBDII

- Characteristics of ISO 9141-2
  - Complies with OSI 7-Layer model
    - Data Link Layer
      - No arbitration
      - **5 baud** initialization
    - Physical Layer
      - **NRZ** @ 10.4kbps
      - **K-Line** or **K**&**L**-Line
      - **12V** systems only

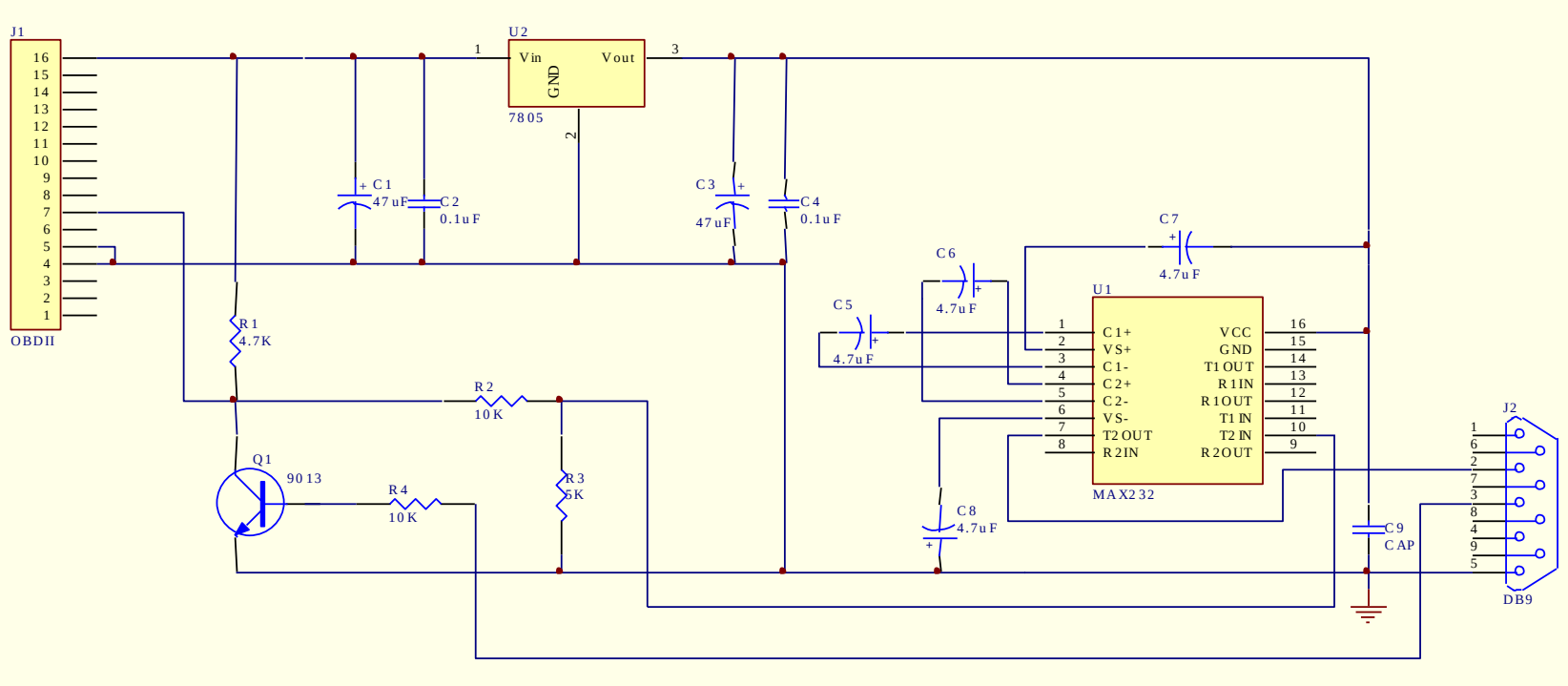
ISO9141-2 on OBDII

- ISO 9141: Physical layer
  - Signal levels of **K**-Line



ISO9141-2 on OBDII

- Interface circuit (**K**-Line-to-**RS**232)

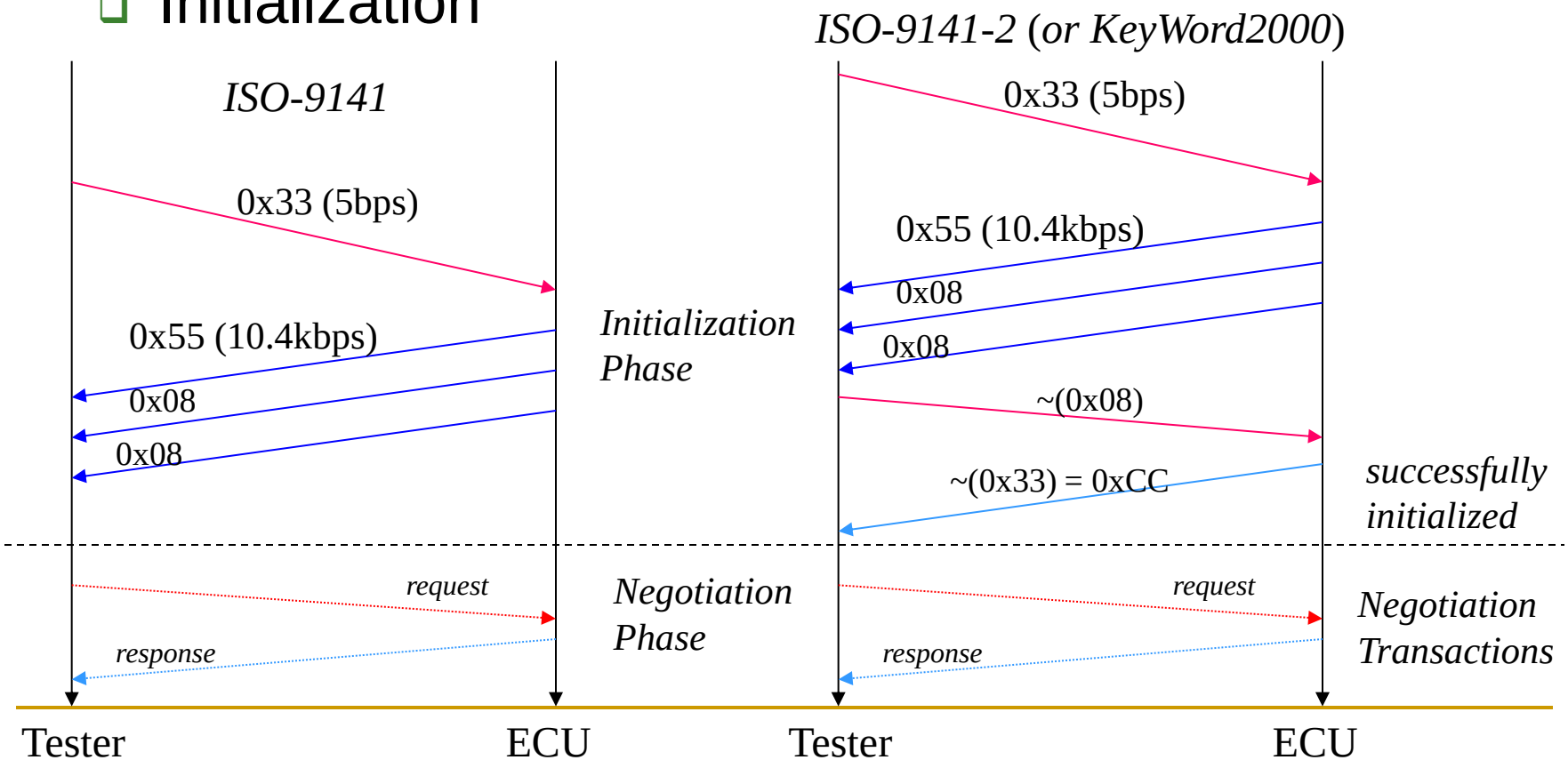


ISO9141-2 on OBDII

- Data rates
  - Wake-up (initialization) : 5 bps
  - Normal operation: 10.4kbps

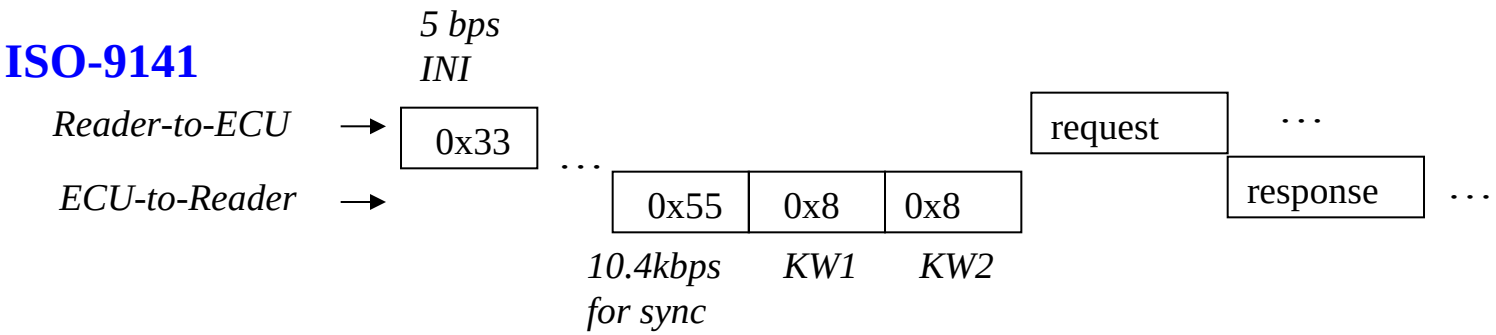
ISO9141-2 on OBDII

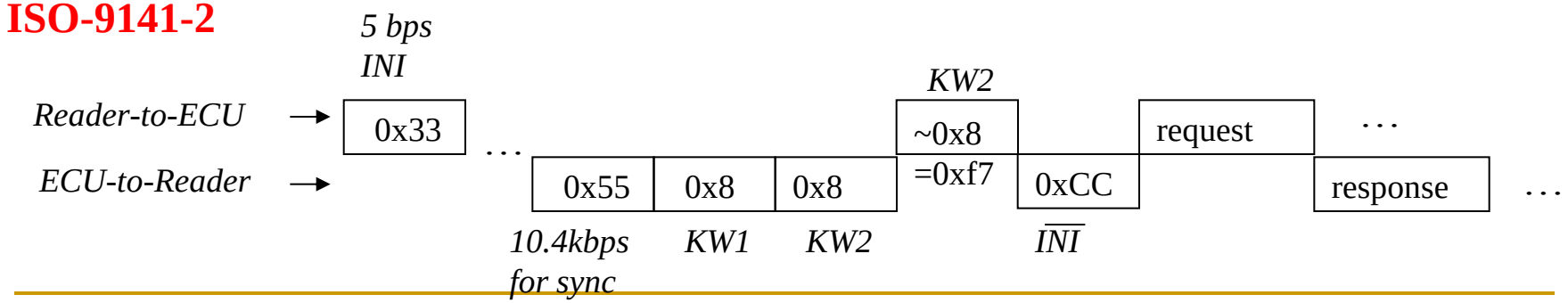
- Transactions of data link layer
  - Initialization



ISO9141-2 on OBDII

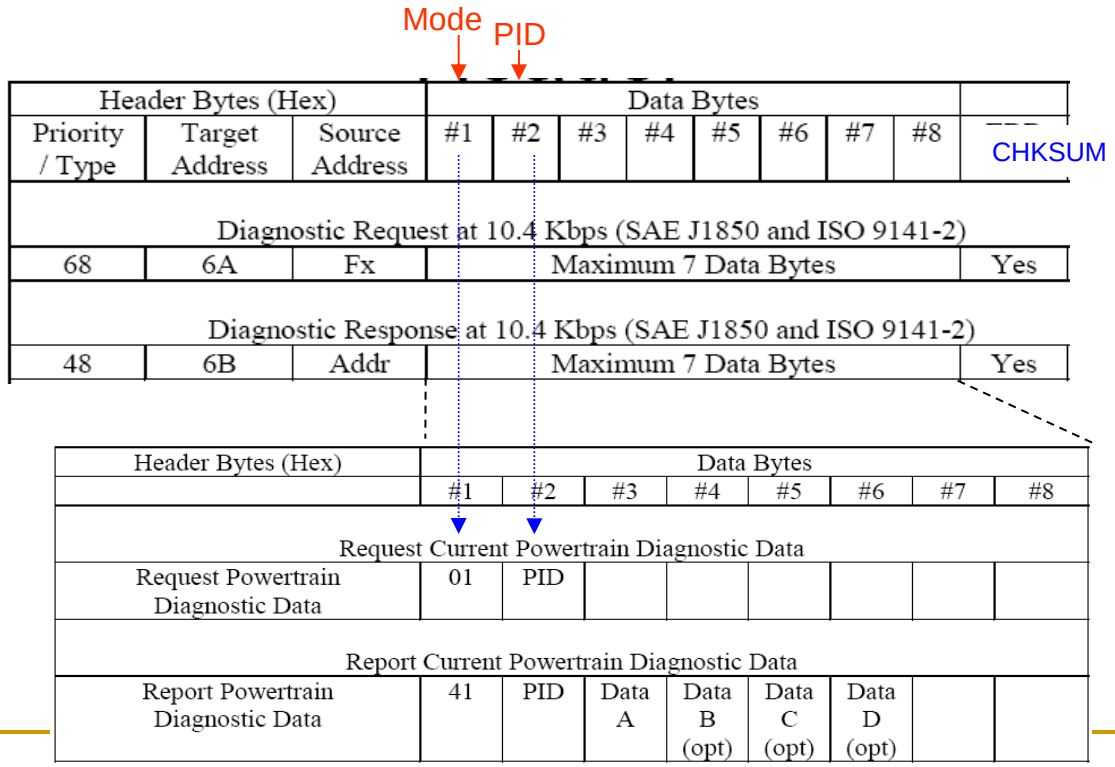
- Transaction timing





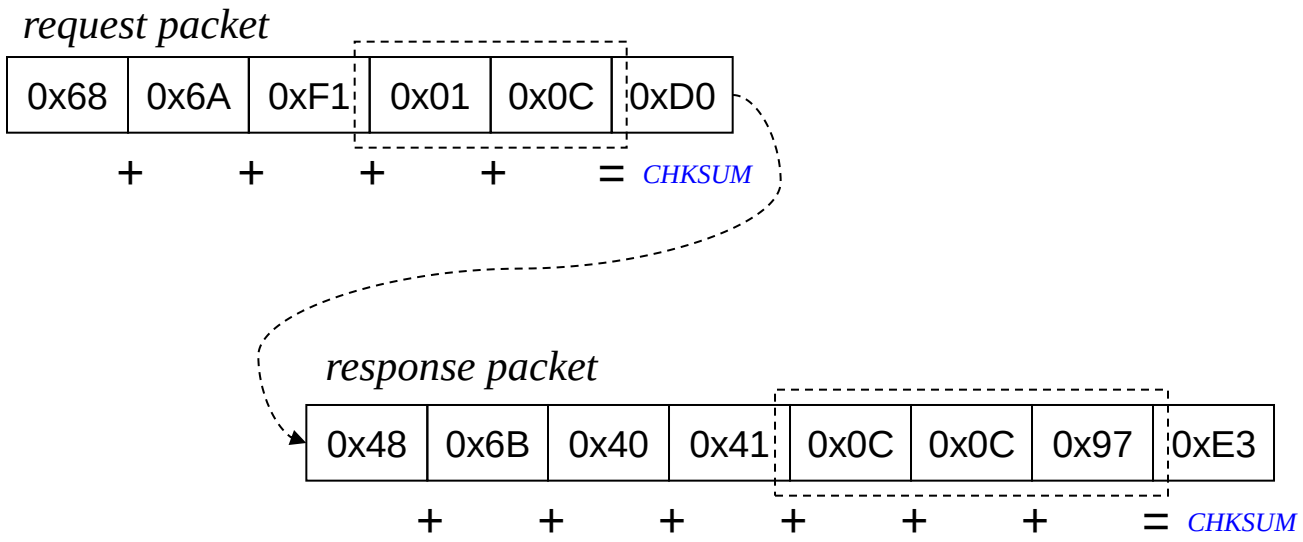
ISO9141-2 on OBDII

- Negotiation phase
  - Request/Response packet formats



ISO9141-2 on OBDII

- Example: *request Engine RPM*



ISO9141-2 on OBDII

- The Mode and PID definitions
  - Modes of operation in the ISO9141-2
    1. Show current data
    2. Show freeze frame data
    3. Show stored Diagnostic Trouble Codes
    4. Clear Diagnostic Trouble Codes and stored values

ISO9141-2 on OBDII

5. Test results, oxygen sensor monitoring
6. Test results, other component/system monitoring
7. Show pending Diagnostic Trouble Codes
8. Control operation of on-board component/system
9. Request vehicle information

- Vehicle manufactures are not required to support all modes
- Each manufacturer may define *additional* modes above mode-9

ISO9141-2 on OBDII

- **PID**s (Parameter IDs) in the ISO9141-2
  - The **PID** gives access to the *real time operational parameters* as well as *flagged DTCs*
  - The specific **PID** in **Mode** 1 was to request the *currently expected vehicle parameters*
  - The specific **PID** in **Mode** 2 requests a snapshot of vehicle parameters taken at the point when the last DTC (Diagnostic Trouble Code) is set

ISO9141-2 on OBDII

- Each **PID** was translated into meaningful parameter the predefined formula

  e.g., Engine RPM PID = 0x0C,

  2 bytes (A and B) are returned

  The predefined formula is

  $((A * 256) + B) / 4$

  Units : rpm

ISO9141-2 on OBDII

- Not all vehicles will support all PIDs
  - Mode1, PID = 0 requests the supported PIDs on the specific vehicle

e.g., in *Chrysler Voyager*, the response of PID=0 is

0xBF 0xBE 0xB9 0x91

1011 1111 1011 1110 1011 1001 1001 0001

ISO9141-2 on OBDII

Example: Supported PIDs on *Chrysler Voyager* (*http://en.wikipedia.org/wiki/OBD-II_PIDs*)

Mode (hex)	PID (hex)	Data bytes returned	Description	Min value	Max value	Units	Formula
01	00	4	PIDs supported				Bit encoded [A7..D0] == [PID 0x01..PID 0x20]
01	01	4	Number of trouble codes and I/M info				Bit encoded. See below.
01	02	8	Freeze DTC				
01	03	2	Fuel system status				Bit encoded. See below.
01	04	1	Calculated engine load value	0	100	%	A*100/255
01	05	1	Engine coolant temperature	-40	215	°C	A-40
01	06	1	Short term fuel % trim—Bank 1	-100 (Rich)	99.22 (Lean)	%	0.7812 * (A-128)
01	07	1	Long term fuel % trim—Bank 1	-100 (Rich)	99.22 (Lean)	%	0.7812 * (A-128)
01	08	1	Short term fuel % trim—Bank 2	-100 (Rich)	99.22 (Lean)	%	0.7812 * (A-128)
01	09	1	Long term fuel % trim—Bank 2	-100 (Rich)	99.22 (Lean)	%	0.7812 * (A-128)
01	0A	1	Fuel pressure	0	765	kPa (gauge)	A*3
01	0B	1	Intake manifold pressure	0	255	kPa (absolute)	A

ISO9141-2 on OBDII

01	0C	2	Engine RPM	0	16,383.75	rpm	((A*256)+B)/4
01	0D	1	Vehicle speed	0	255	km/h	A
01	0E	1	Timing advance	-64	63.5	° relative to #1 cylinder	A/2 - 64
01	0F	1	Intake air temperature	-40	215	°C	A-40
01	10	2	MAF air flow rate	0	655.35	g/s	((256*A)+B) / 100
01	11	1	Throttle position	0	100	%	A*100/255
01	12	1	Sec.(?) air status				Bit encoded. See below.
01	13	1	Oxygen sensors present				[A0..A3] == Bank 1, Sensors 1-4. [A4..A7] == Bank 2...
01	14	2	Bank 1, Sensor 1: Oxygen sensor voltage, Short term fuel trim	0 0	1.275 99.2	Volts %	A * 0.005 (B-128) * 0.7812 (if B==0xFF, sensor is not used in trim calc)
01	15	2	Bank 1, Sensor 2: Oxygen sensor voltage, Short term fuel trim	0 0	1.275 99.2	Volts %	A * 0.005 (B-128) * 0.7812 (if B==0xFF, sensor is not used in trim calc)
01	16	2	Bank 1, Sensor 3: Oxygen sensor voltage, Short term fuel trim	0 0	1.275 99.2	Volts %	A * 0.005 (B-128) * 0.7812 (if B==0xFF, sensor is not used in trim calc)
01	17	2	Bank 1, Sensor 4: Oxygen sensor voltage, Short term fuel trim	0 0	1.275 99.2	Volts %	A * 0.005 (B-128) * 0.7812 (if B==0xFF, sensor is not used in trim calc)

ISO9141-2 on OBDII

01	18	2	Bank 2, Sensor 1: Oxygen sensor voltage, Short term fuel trim	0 0	1.275 99.2	Volts %	A * 0.005 (B-128) * 0.7812 (if B==0xFF, sensor is not used in trim calc)
01	19	2	Bank 2, Sensor 2: Oxygen sensor voltage, Short term fuel trim	0 0	1.275 99.2	Volts %	A * 0.005 (B-128) * 0.7812 (if B==0xFF, sensor is not used in trim calc)
01	1A	2	Bank 2, Sensor 3: Oxygen sensor voltage, Short term fuel trim	0 0	1.275 99.2	Volts %	A * 0.005 (B-128) * 0.7812 (if B==0xFF, sensor is not used in trim calc)
01	1B	2	Bank 2, Sensor 4: Oxygen sensor voltage, Short term fuel trim	0 0	1.275 99.2	Volts %	A * 0.005 (B-128) * 0.7812 (if B==0xFF, sensor is not used in trim calc)
01	1C	1	OBD standards this vehicle conforms to				Bit encoded. See below.
01	1D	1	Oxygen sensors present				Similar to PID 13, but [A0..A7] == [B1S1, B1S2, B2S1, B2S2, B3S1, B3S2, B4S1, B4S2]
01	1E	1	Auxiliary input status				A0 == Power Take Off (PTO) status (1 == active) [A1..A7] not used
01	1F	2	Run time since engine start	0	65,535	seconds	(A*256)+B
01	20	4	PIDs supported 21-40				Bit encoded [A7..D0] == [PID 0x21..PID 0x40]

ISO9141-2 on OBDII

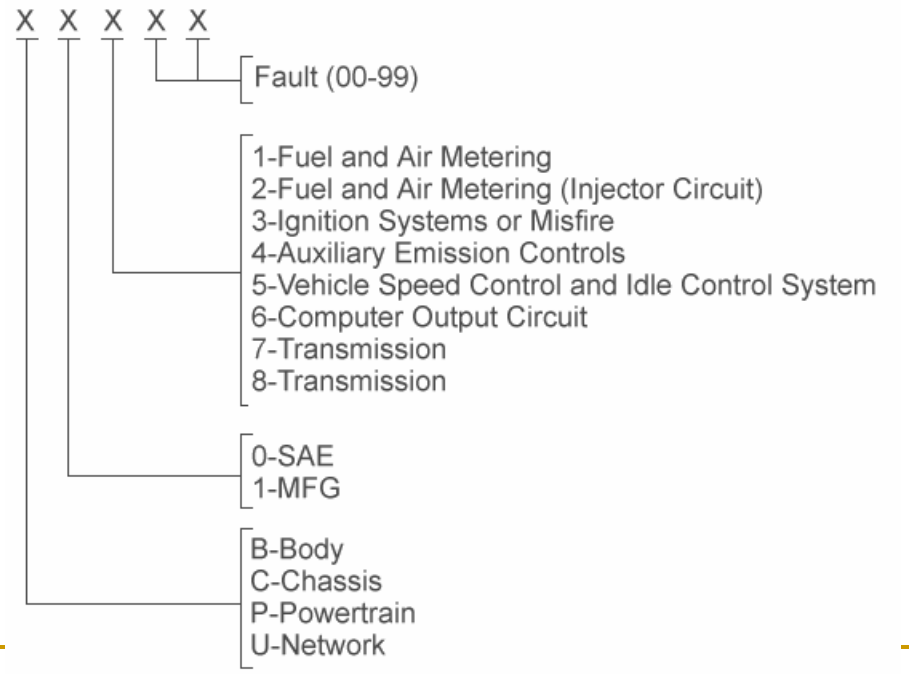
- The DTC (Diagnostic Trouble Code) definitions
  - Mode 3 (no PID required) requests the DTCs that have been set
  - The response will be an integer number of packets each containing 6 data bytes (A, B, C, D, E, F)

ISO9141-2 on OBDII

- Each trouble code requires 2 bytes to describe (AB, CD, EF)
- A trouble code can be decoded from each pair of data bytes

ISO9141-2 on OBDII

- Anatomy of the DTC
  - A DTC is made up of 5 digits
  - The figure below demonstrates the composition of a DTC



ISO9141-2 on OBDII

- Interpretation of DTC's data bytes
  - Each DTC consists of five parts within a pair of data bytes (AB, CD, EF)
    - **First Character** — System

A7	A6	First DTC character
0	0	P - Powertrain
0	1	C - Chassis
1	0	B - Body
1	1	U - Network

ISO9141-2 on OBDII

- **Second Digit** — Code Type

A5	A4	Second DTC digit
0	0	Generic
0	1	Enhanced (manufacturer specific)
1	0	undefined
1	1	undefined

ISO9141-2 on OBDII

- **Third Digit** — Sub-System

A3	A2	A1	A0	Third DTC digit
0	0	0	0	SAE Reserved
0	0	0	1	Emission Management (Fuel or Air)
0	0	1	0	Injector Circuit (Fuel or Air)
0	0	1	1	Ignition or Misfire
0	1	0	0	Emission Control
0	1	0	1	Vehicle Speed & Idle Control
0	1	1	0	Computer & Output Circuit
0	1	1	1	Transmission
1	0	0	0	Transmission
1	0	0	1	SAE Reserved

ISO9141-2 on OBDII

- **Fourth and Fifth** — Digits are variable, and relate to a particular problem

e.g., the response of Mode 3 request

0x04 0x20 0x00 0x00 0x00 0x00

A B C D E F

00 00 0100 0010 0000

P 0 4 2 0

P0420:
Catalyst System Efficiency Below Threshold (Bank 1)

P = Powertrain

Generic

Emission Control

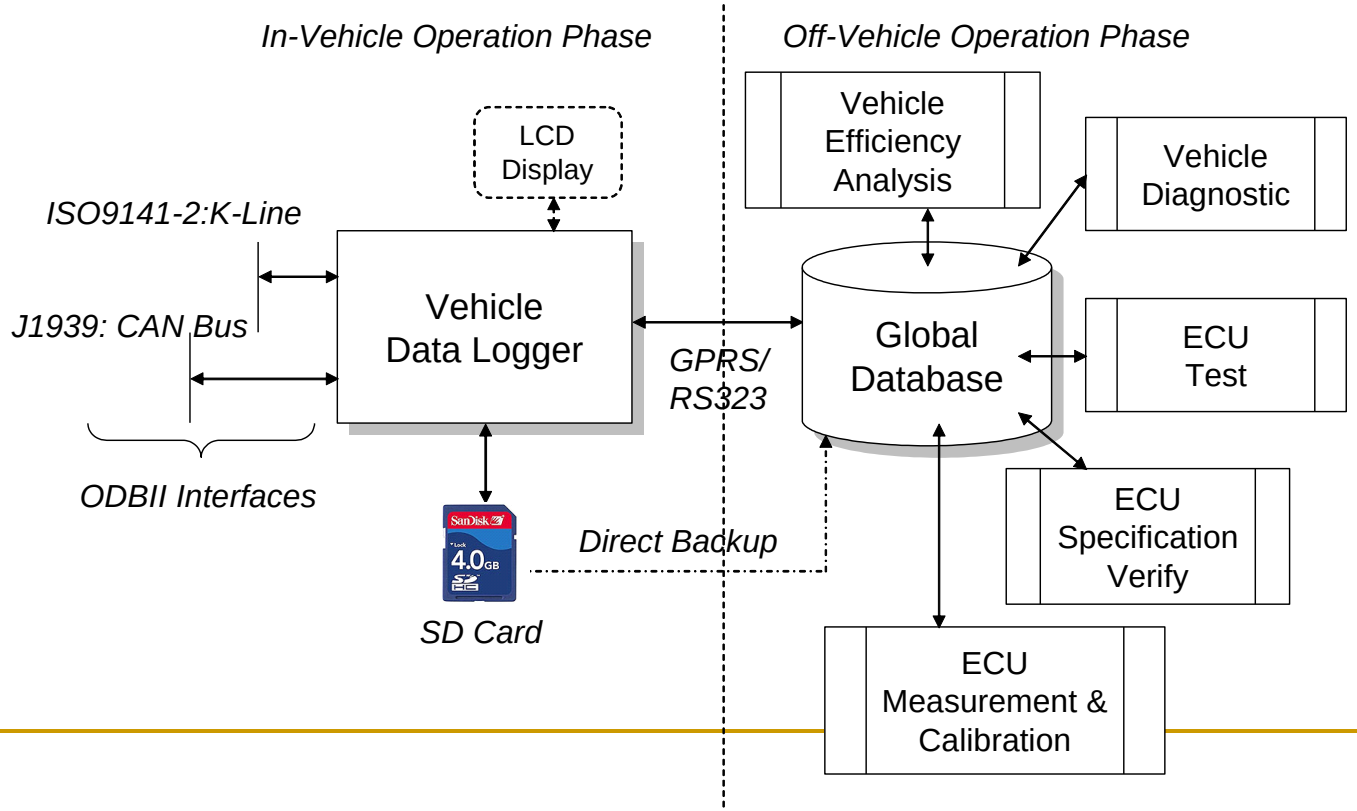
ISO9141-2 on OBDII

- Example: Generic DTCs of Powertrain (*http://www.myscantool.com/dtc/*)

P0080	Exhaust Valve Control Solenoid Circuit High
P0081	Intake Valve Control Solenoid Circuit
P0082	Intake Valve Control Solenoid Circuit Low
P0083	Intake Valve Control Solenoid Circuit High
P0084	Exhaust Valve Control Solenoid Circuit
P0085	Exhaust Valve Control Solenoid Circuit Low
P0086	Exhaust Valve Control Solenoid Circuit High
P0087	Fuel Rail/System Pressure - Too Low
P0088	Fuel Rail/System Pressure - Too High
P0089	Fuel Pressure Regulator 1 Performance
P0090	Fuel Pressure Regulator 1 Control Circuit
P0091	Fuel Pressure Regulator 1 Control Circuit Low
P0092	Fuel Pressure Regulator 1 Control Circuit High
P0093	Fuel System Leak Detected - Large Leak
P0094	Fuel System Leak Detected - Small Leak

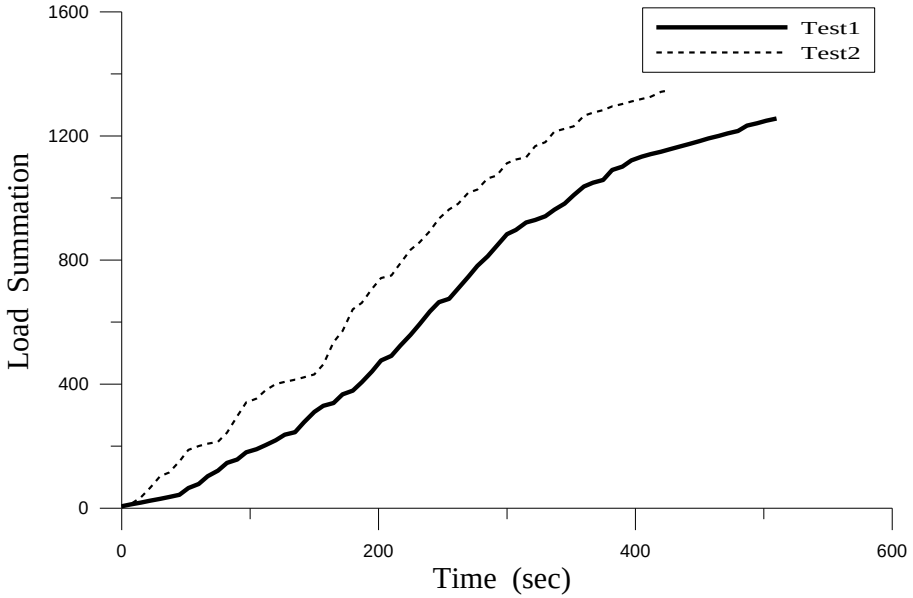
ISO9141-2 on OBDII

- Application
  - In-Vehicle Data Logger (System Architecture)



ISO9141-2 on OBDII

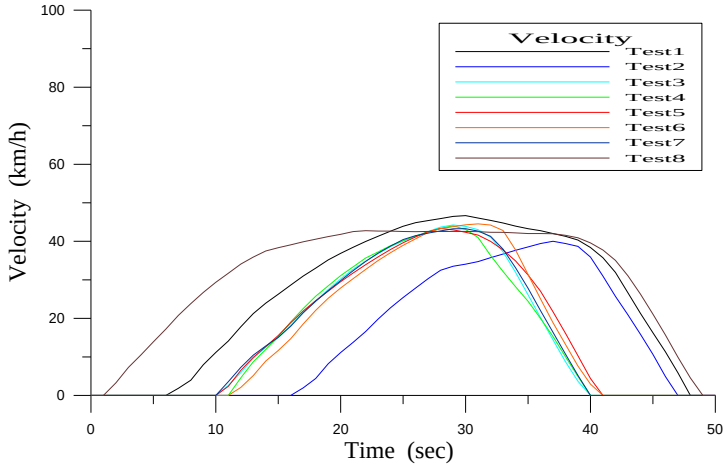
- Analysis: Energy Efficiency Comparison

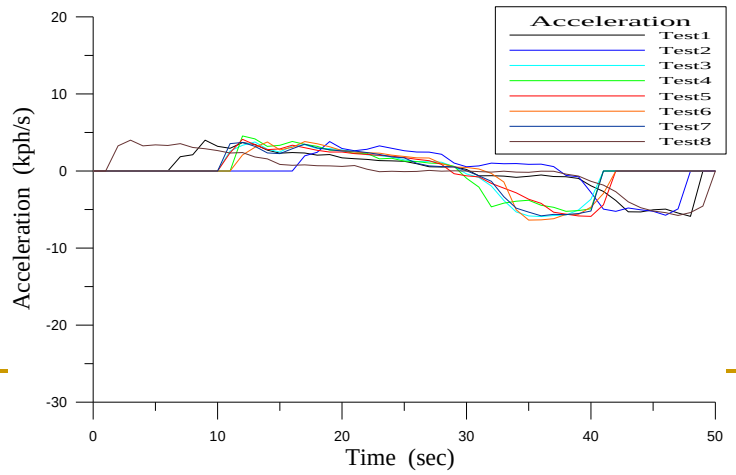


較溫和駕駛行為		較激烈駕駛行為	
行駛時間	510 sec	行駛時間	427 sec
行駛距離	6116 m	行駛距離	6000 m
平均速度	42.54 km/s	平均速度	49.67 km/s
累積負載	1256	累積負載	1348

ISO9141-2 on OBDII

- Analysis: Driving Behavior

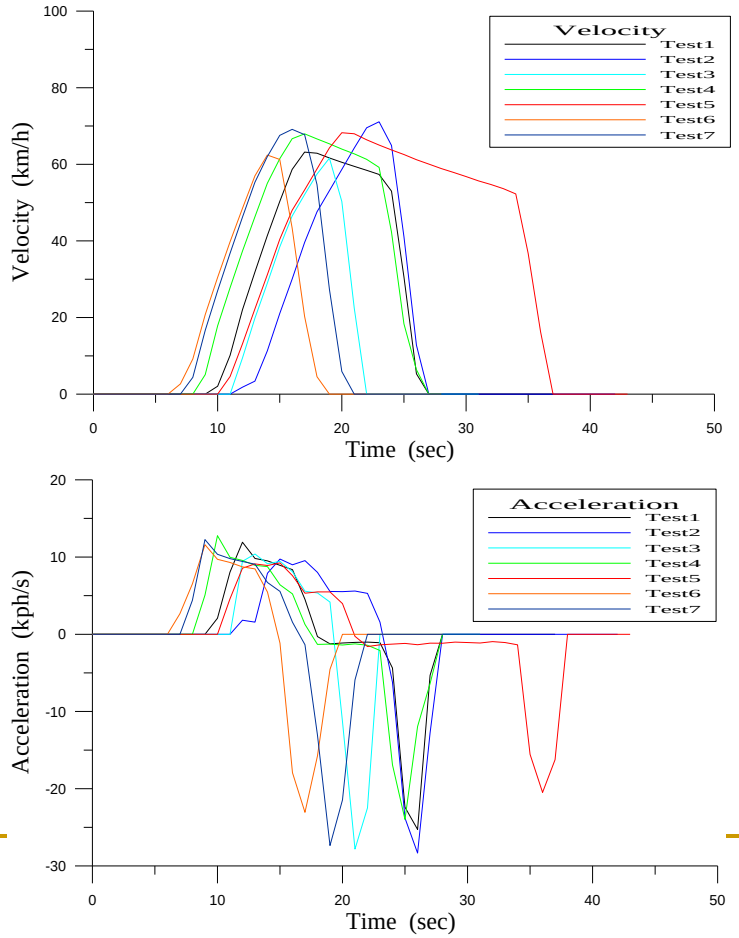




輕度加減速平均值	
平均加速度	2.53 kph/s
平均最大加速度	3.88 kph/s
平均減速度	4.41 kph/s
平均最大減速度	5.76 kph/s

ISO9141-2 on OBDII

- Analysis: Driving Behavior (Cont.)



重度加減速平均值	
平均加速度	7.56 kph/s
平均最大加速度	11.12 kph/s
平均減速度	16.74 kph/s
平均最大減速度	25.18 kph/s

J1939 on OBDII

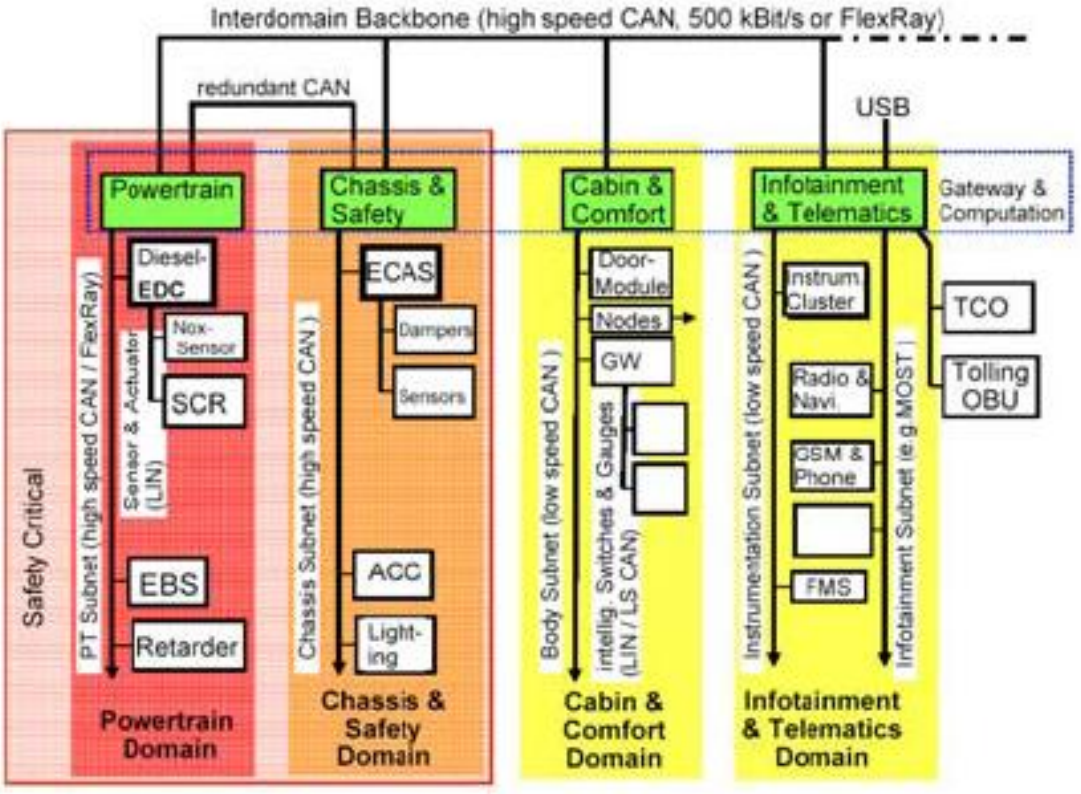
- J1939 protocol
  - SAE J1939 is the *open standard* for networking and communication in the commercial vehicle sector, e.g.; *power train*, *chassis*, *OBD*
  - J1939 is the use of *CAN technology* for networking and communication as well as manufacturer-spanning interoperability

J1939 on OBDII

- The J1939 protocol comes from the Society of Automotive Engineers (SAE) and works on the physical layer with CAN-high speed according to ISO11898 (CAN2.0B)

J1939 on OBDII

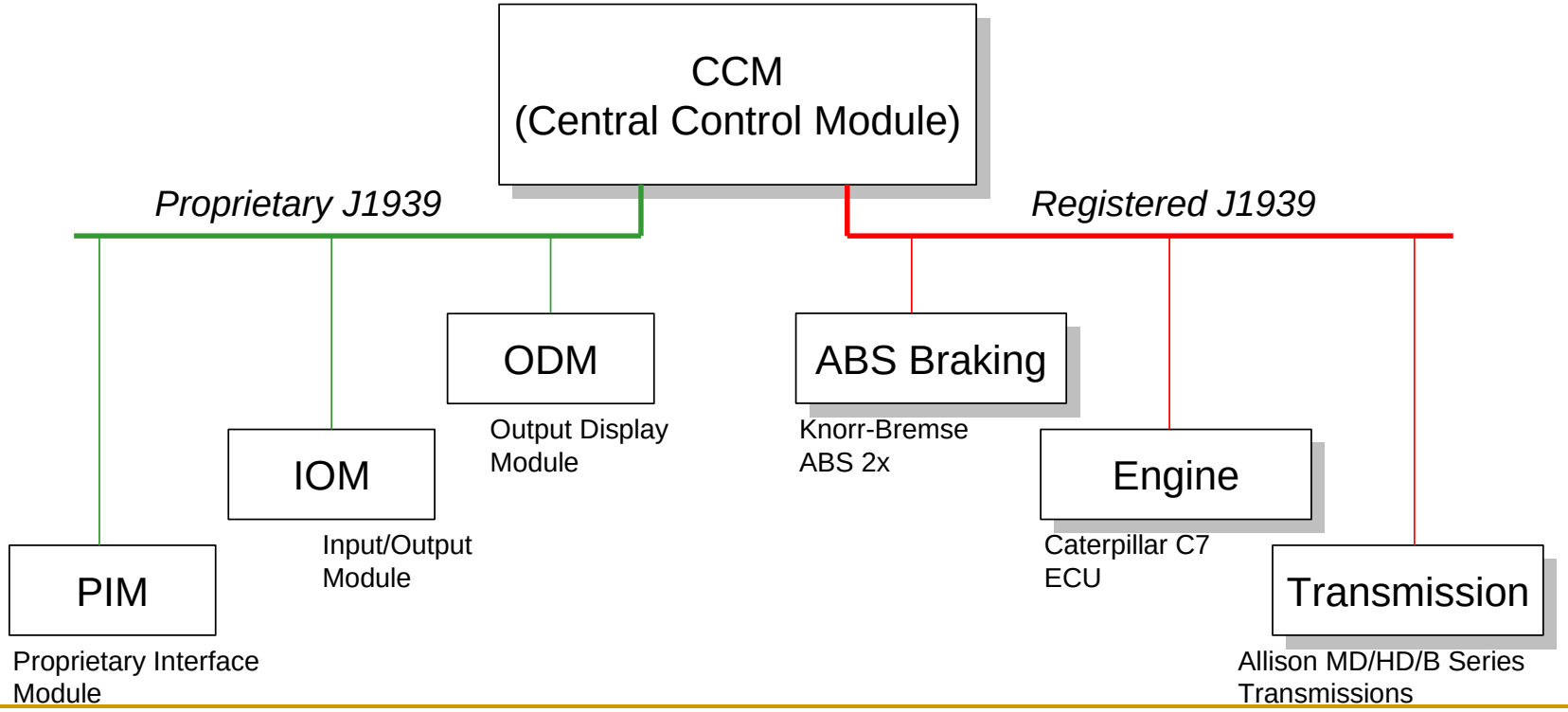
- SAE J1939-*conforming*, *decentral* CAN network architecture



from http://www.vector-worldwide.com/portal/medien/cmc/press/PON/CANimNfz_CiA_200509_PressArticle_EN.pdf

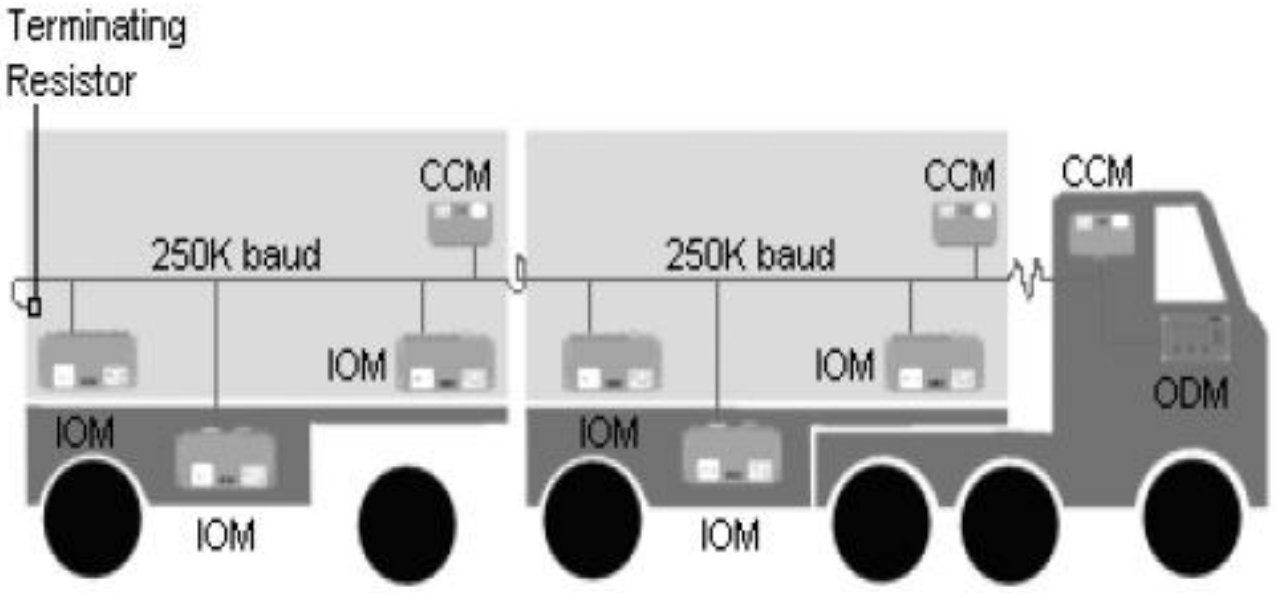
J1939 on OBDII

- ECUs on the Registered and Proprietary J1939 Buses



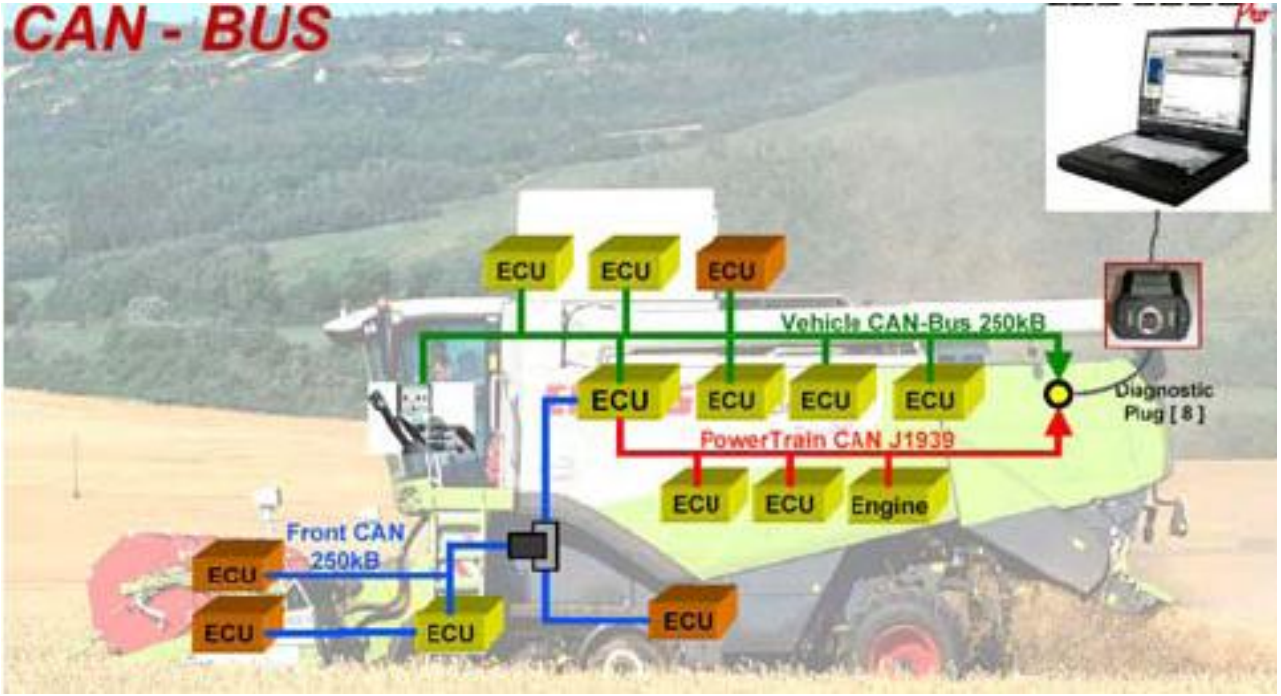
J1939 on OBDII

- Tractor trailer application



J1939 on OBDII

- Diagnosis via CAN in agrarian technology

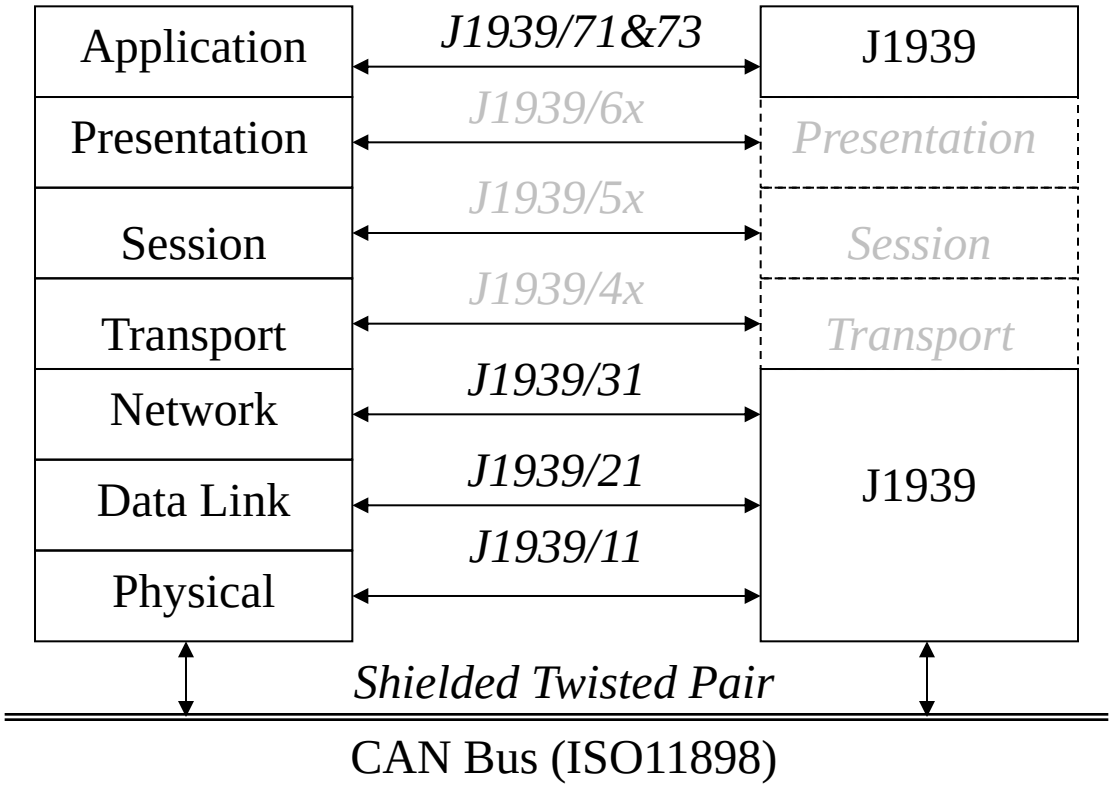


J1939 on OBDII

- The particular characteristics of J1939
  - 29-bit identifier (extended-identifier message)
  - Peer-to-peer and broadcast communication
  - Transport protocols for up to 1785 data bytes (255 packets)
  - Network management
  - Definition of parameter groups

J1939 on OBDII

- OSI 7-Layer model for J1939

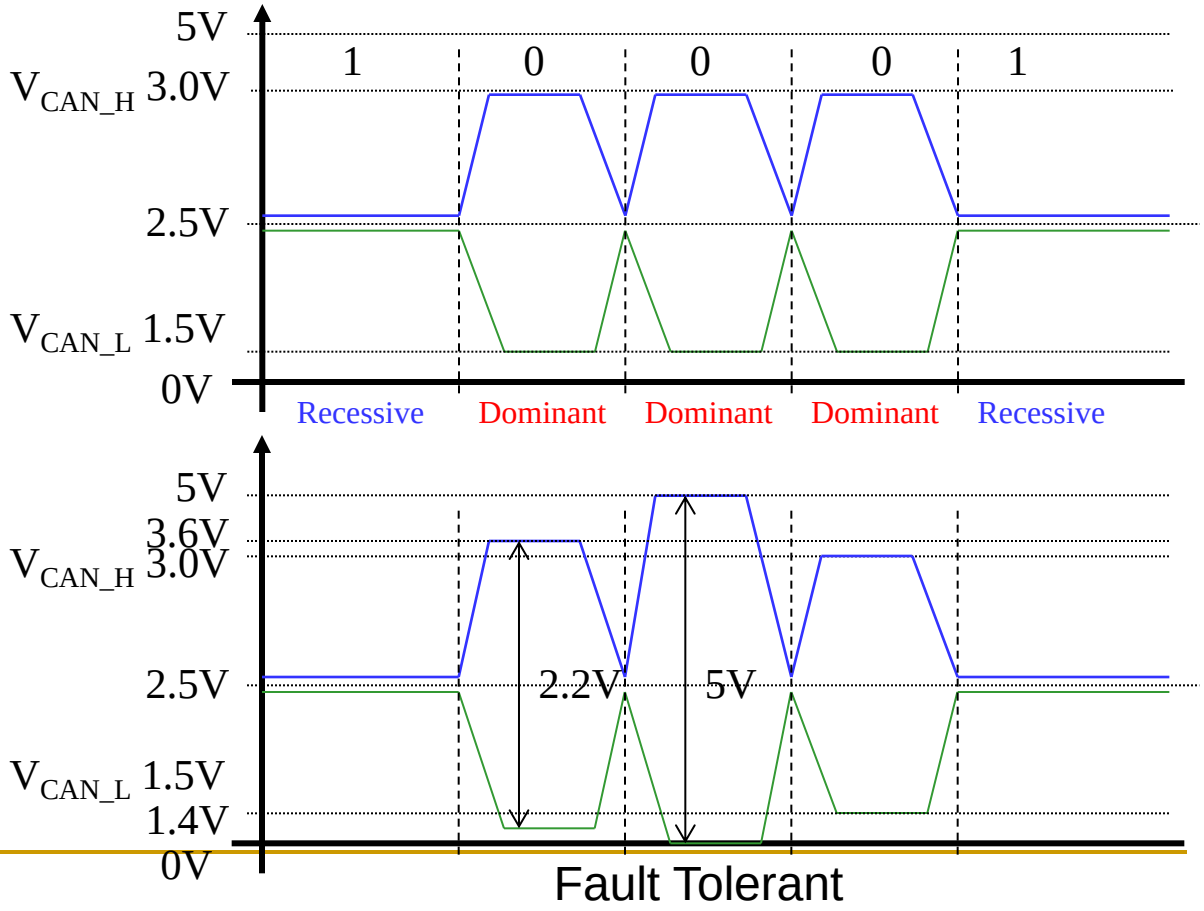


J1939 on OBDII

- J1939 main documents
  - SAE J1939 - Recommended Practice for a Serial Control & Communications
  - SAE J1939/11 - Physical Layer
  - SAE J1939/21 - Data Link Layer
  - SAE J1939/31 - Network Layer
  - SAE J1939/71 - Vehicle Application Layer
  - SAE J1939/73 - Application Layer Diagnostics – OBD
  - SAE J1939/81 - Network management Protocol

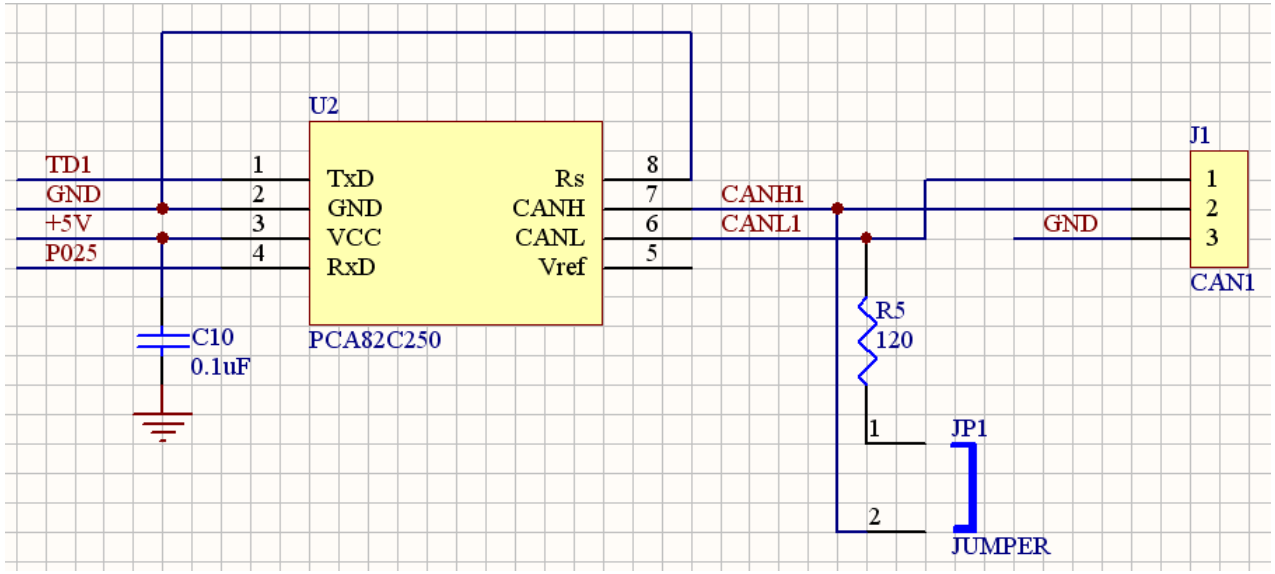
J1939 on OBDII

- The physical and data link layer
  - Signal levels



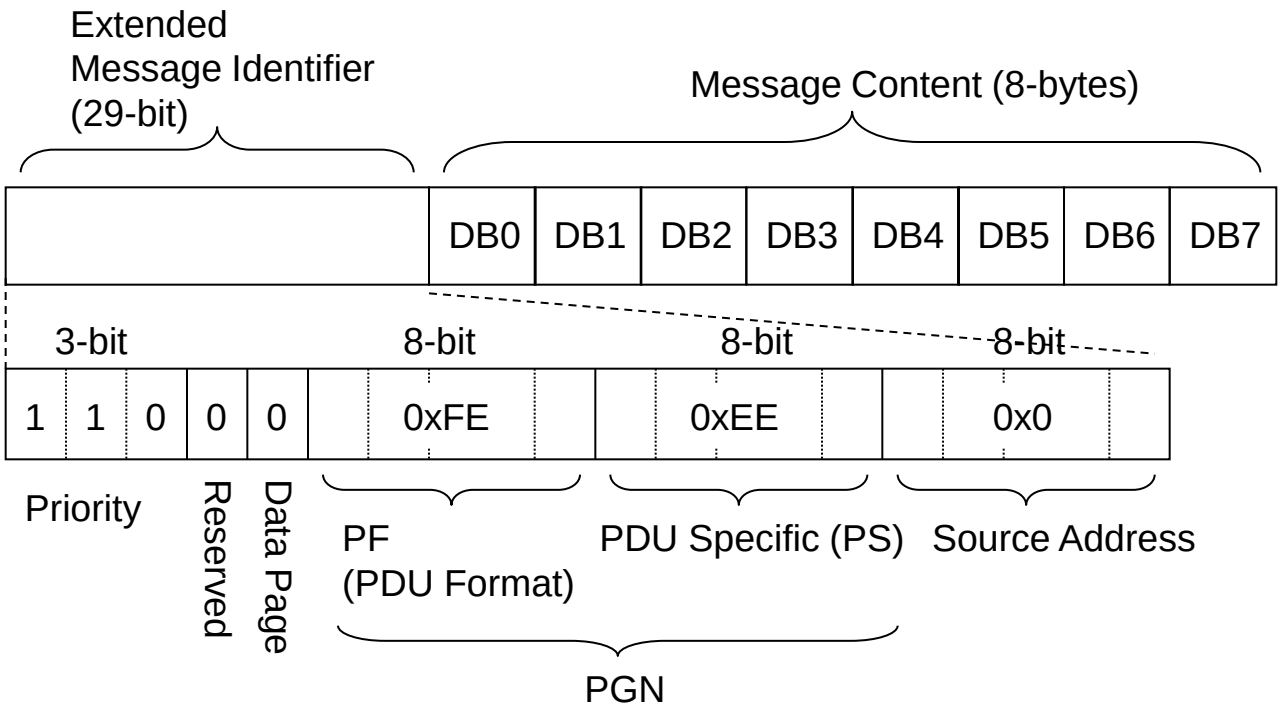
J1939 on OBDII

- Interface circuit



J1939 on OBDII

- CAN Message format on J1939 (extend-identifier messages)



J1939 on OBDII

- The PGN (Parameter Group Number) definitions
  - Parameter groups combine similar or associated *signals*
    - In the specification SAE J1939-71 the parameter groups are defined with the *signals* they contain
  - In addition, some *manufacturer-specific* parameter groups can be used

J1939 on OBDII

- Parameter groups with up to *8 data bytes* are transmitted in a CAN message
  - With more that 8 bytes, a *transport protocol* is used
- Each *parameter group* is addressed uniquely via a number (**PGN**)
  - For this number, a *16-bit value* is used that is composed of the *PDU format* and *PDU specific*

J1939 on OBDII

- There are two types of parameter group numbers:
  - **Global PGNs** for parameter groups that are sent to all (broadcast)
    - Here all 16 bits of the PGN are used ≥ *0xF0*
  - **Specific PGNs** for the parameter groups that are sent to particular devices (peer-to-peer)
    - With these PGNs, only the higher-value 8-bit (PDU format) are valid and the value must be smaller than 240 (< *0xF0*)
    - The lower value byte (PDU specific) is always 0

J1939 on OBDII

- Sample of parameter group definition

Name:		Engine temperature (5.3.28)	
Transmission rate:		1s	
Data length:		8 bytes	
Data page:		0	
PDU format:		254 (0xFE)	
PDU specific:		238 (0xEE)	
Default priority:	6		
PGN number:		65262 (0xFEEE) → CAN Message ID = 0x18FEEE00	
Description of data:			
Byte:	**1**	Engine coolant temperature	(5.2.5.5)
	2	Fuel temperature	(5.2.5.14)
	3,4	Engine oil temperature	(5.2.5.15)
	5,6	Turbo oil temperature	(5.2.5.16)
	7	Engine intercooler temperature	(5.2.5.6)
	8	Engine intercooler thermostat opening	(5.2.5.242)

J1939 on OBDII

5.2.5.5 Engine coolant temperature –

Temperature of liquid found in engine cooling system

Data length:	1 byte
Resolution:	1 °C/bit gain, –40°C offset
Data range:	–40 to +210 °C (–40 to 410 °F)
Type:	Measured
Suspect parameter number:	110
Reference:	5.3.28

J1939 on OBDII

PGN $F004 (61,444) **Electronic Engine Controller #1 : EEC#1**

Source Address: $0 (0) Priority: 3
Repetition rate: 15 ms Data length: 8 Bytes
Data Page: 0 On request service: Yes
SAE section: 5.3.7 (SAE – J1939-71)

Byte	Bits	Parameter
3	8-1	Actual engine – percent torque (5.2.1.5) Resolution: 1%/Bit gain, -125% Offset Data Range: -125 – 125% $FF Not used
4-5	8-1	Engine speed (5.2.1.9) Resolution: 0.125 rpm/Bit gain, 0 rpm offset Data range: 0 – 8031.875 rpm $FFFF Not used

```
case 0xf004: // Actual engine percent torque and Engine speed
            Actual_Engine_Torque = receivedata[rx_head].MsgData.Data[2];
            Engine_Speed = receivedata[rx_head].MsgData.Data[4] * 0x100 + receivedata[rx_head].MsgData.Data[3];
```

J1939 on OBDII

- `struct` definition on C language for PGN

```
struct J1939_FIELDS {
        unsigned int SRC: 8;
        unsigned int PS : 8;
        unsigned int PF : 8;
        unsigned int DP : 1;
        unsigned int R : 1;
        unsigned int PRI : 3;
        unsigned int :3;
}; // little-endian
```

J1939 on OBDII

- Transport protocols (for Specific PGNs)
  - Parameter groups that contain more than 8 data bytes are transmitted with a transport protocol
  - There are two special parameter groups available
    - *TP.CM* for connection management
    - *TP.DT* for the transmission of the data

J1939 on OBDII

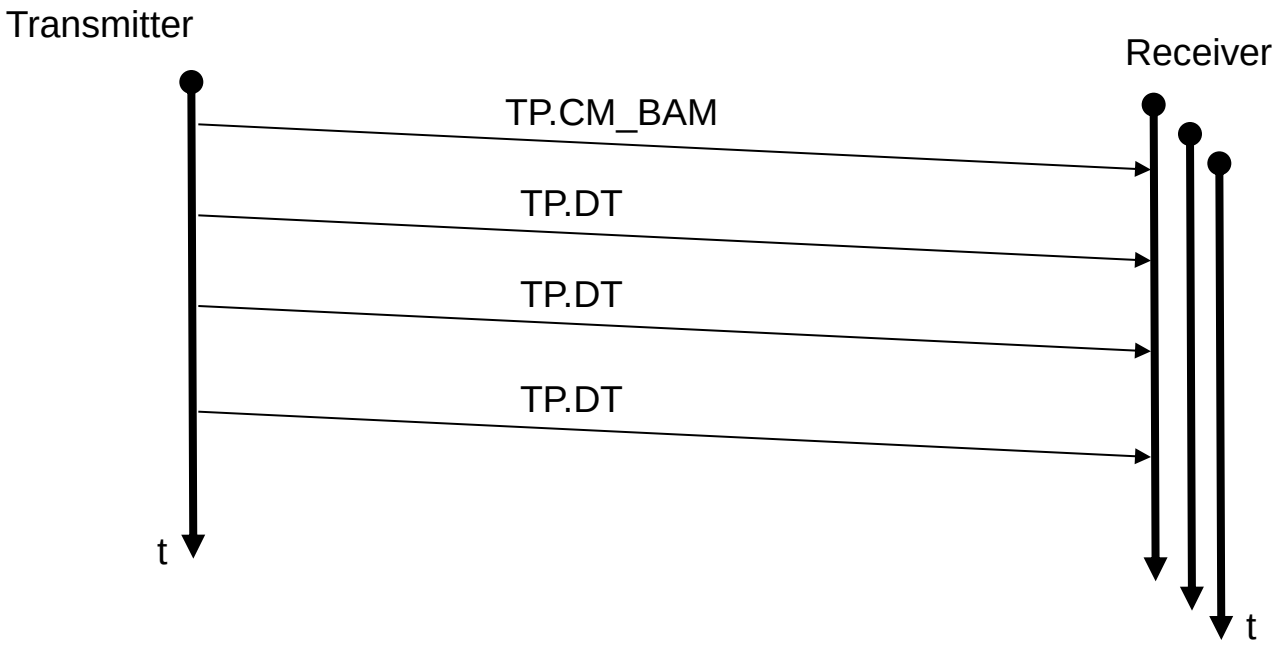
- There are two different protocols
  - Peer-to-peer and
  - Broadcast transmission

J1939 on OBDII

- Broadcast transmission
  - For broadcast transmission, the **BAM** (Broadcast Announce Message) protocol (*TP.CM_BAM*) is used
  - Here, after a BAM-PGN (0xEC00) the transmitter sends all data PGs at a *minimum interval of 50ms*

J1939 on OBDII

- Transactions of broadcast transmission

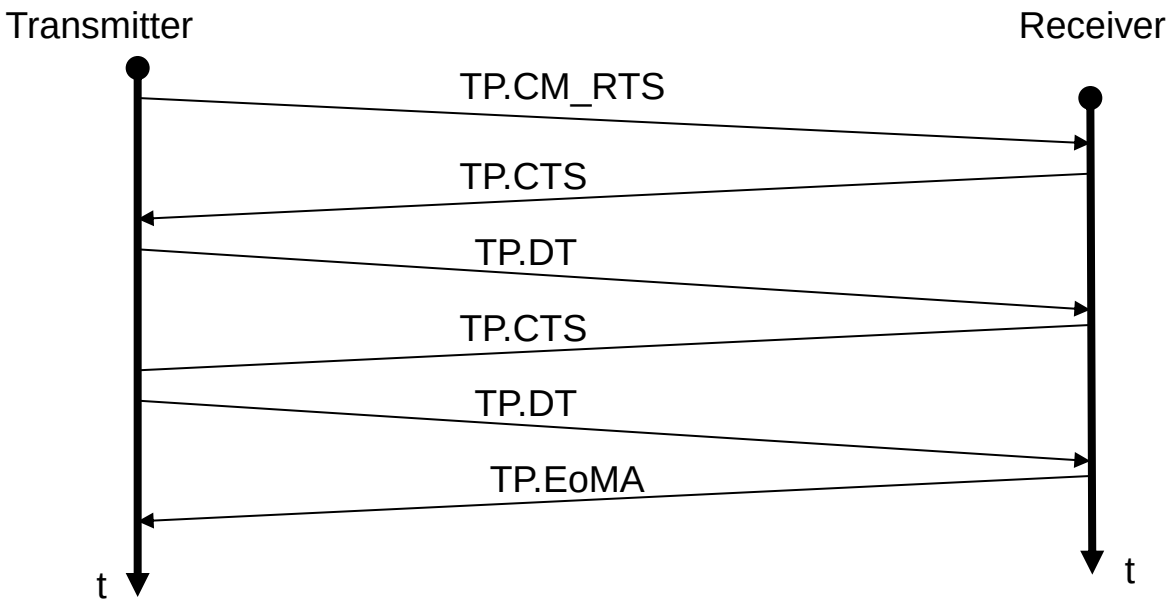


J1939 on OBDII

- Peer-to-peer transmission
  - With the peer-to-peer transmission, the transmitter initiates the connection with a "request to send" (*TP.RTS*) message
  - The receiver then controls the transport protocol with "clear to send" (*TP.CTS*) and "end of message acknowledge" (*TP.EoMA*)

J1939 on OBDII

- Transactions of peer-to-peer transmission



J1939 on OBDII

- The DTC (Diagnostic Trouble Code) definitions (SAE J1939-73)
  - A DTC is made up of 4 elements
    - Suspect Parameter Number (SPN) 19 bits
    - Failure Mode Identifier (FMI) 5 bits
    - Occurrence Count (OC) 7 bits
    - SPN Conversion Method (CM) 1 bit

J1939 on OBDII

DTC																													
Byte 3 (Least Significant Byte of SPN)								Byte 4 (Second Byte of SPN)								Byte 5 (3 Most Significant Bits of SPN)								Byte 6							
SPN																			FMI					CM	OC						
8	7	6	5	4	3	2	1	8	7	6	5	4	3	2	1	8	7	6	5	4	3	2	1	8	7	6	5	4	3	2	1
1	0	1	1	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	1	0	0	0	0	1	0	1	0

SPN 1208 : Pre-filter oil pressure

SPN = 000 00000100 10111000 = 0x0040B8 = 1208
FMI = 00011 = 0x3 = 3
OC = 0001010 = 0xA = 10
CM = 0

```
tmp = (BAM[i+2] >> 5);
spn = tmp * 0x10000 + BAM[i+1] * 0x100 + BAM[i];
fmi = BAM[i+2] & 0x1f;
oc =  BAM[i+3] & 0x7f;
```

J1939 on OBDII

- **DM1**: Active Diagnostic Trouble Codes
  - The information communicated is limited to *the currently active DTC*

Data length:		variable
Data page:		0
PDU format:		254 (0xFE)
PDU specific:		202 (0xCA)
Default priority:		6
PGN number:		65226 (0xFECA)
Description of data:		
Byte:	**1 & 2**	Malfunction Indicator Lamp Status
	3,4,5,6	SPN, FMI, CM, and OC

J1939 on OBDII

- Multiple DTCs
  - Broadcast Announce Message (*TP.CM_BAM*)

Data length: 8 bytes
Data page: 0
PDU format: 236 (0xEC)
PDU specific: Destination Address
Default priority: 7
PGN number: 60416 (0xEC00) →0xECFF in **Caterpillar C7**
Description of data:

Byte:	**1**	Control byte = 32
	2&3	Total message size (# Bytes)
	4	Total number of packets
	5	Reserved
	6-8	PGN of the packeted message (abcdbcdbcd…)

J1939 on OBDII

- Data Transfer (*TP.DT*)

Data length:		8 bytes
Data page:		0
PDU format:		235 (0xEC)
PDU specific:		Destination Address (Global = 0xFF)
Default priority:		7
PGN number:		60160 (0xEB00) →0xEBFF in **Caterpillar C7**
Description of data:		
Byte:	**1**	Sequence number
	2-8	Packeted data (extra byte be filled 0xFF)

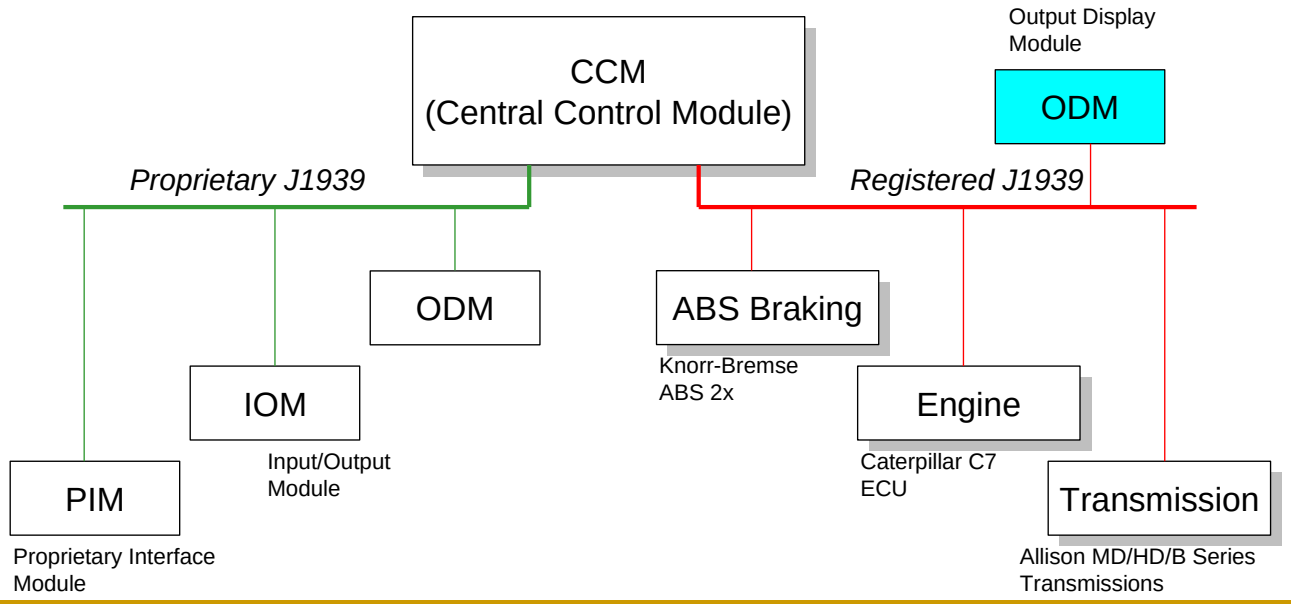
J1939 on OBDII

- Some SPN codes in *Caterpillar C7* engine

SPN-FMI	Diagnostic Code Description
100-03	Oil Pressure Sensor Open Circuit
100-04	Oil Pressure Sensor Short Circuit
100-11	Very Low Oil Pressure
102-01	Boost Pressure Reading Stuck Low
102-02	Erratic Boost Pressure
102-03	Boost Pressure Sensor Open Circuit
102-04	Boost Pressure Sensor Short Circuit
102-07	Excessive Boost Pressure

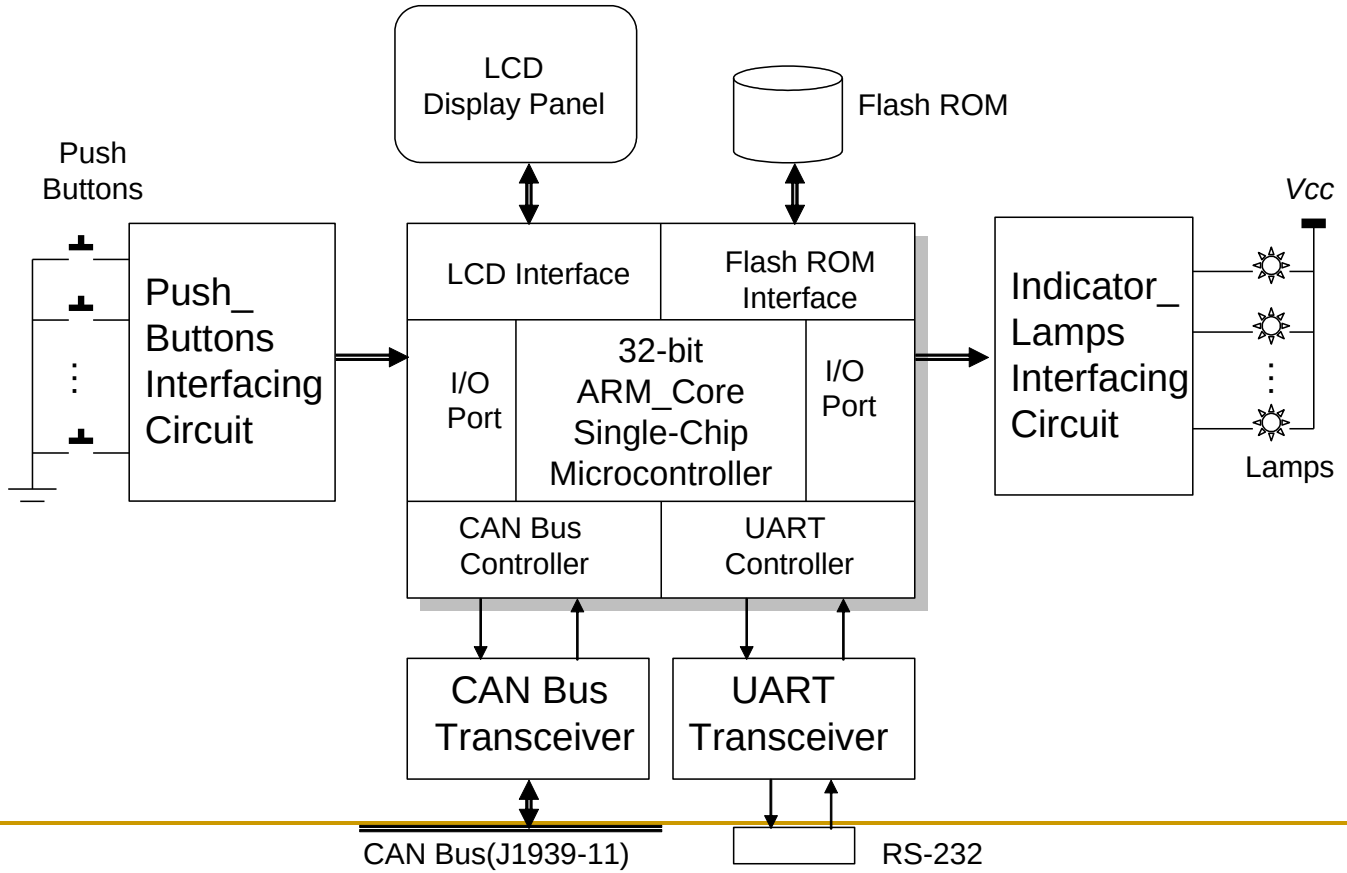
J1939 on OBDII

- Application:
  - Parameters and Trouble Codes Monitoring and Display



J1939 on OBDII

- System architecture



簡報結束

敬請指教