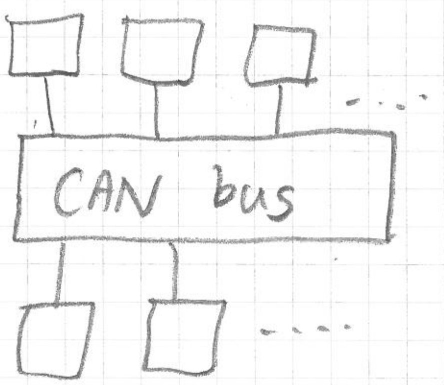
- CAN bus in automotive data communication:

Modern vehicles heavily rely on computing chips (known as ECU, electronic control unit) to monitor and coordinate in-vehicle subsystems such as airbags, safety driving features, wheels, and infotainment.

CAN (Controller Area Network) is designed to enable efficient and robust data communications between these chips.

The CAN bus consists of a single channel that carries bits. Chips connected to a bus can send and receive bits, and they can also monitor the current bit value on the bus.



Each data sent on a bus is considered as a broadcast.

(compare it to the ALOHA protol and review how ALOHA resolves data collisions).

Different from the ALOHA protocol, the CAN bus features an arbitration process that can

1. waste no delay in allowing one of the data to be sent successfully;
2. prioritize data communication.

The priority is encoded in the IDENTIFIER field of a data frame. The field is a string of logical values "dominant" an "recessive". In case of a wired-AND implementation of the bus, the "dominant" level would be represented by a logical "0" and the "recessive" level by a logical "1". Hence, the dominant bit will prevail the recessive bit when concurrently sent on the bus. During arbitration, every data sender compares the level of the bit sent with the level that is monitored on the bus. If both levels are equal, the sender can continue to send; if a "recessive" level is sent but a "dominant" level is monitored, the sender must then withdraw and wait until the next frame. Arbitration starts at MSB.

Example:

	MSB … LSB
d_1	00011100101
d_2	00011100010
d_3	00100110101

then the relative priority:

$d_2 > d_1 > d_3$