Controller Area Network

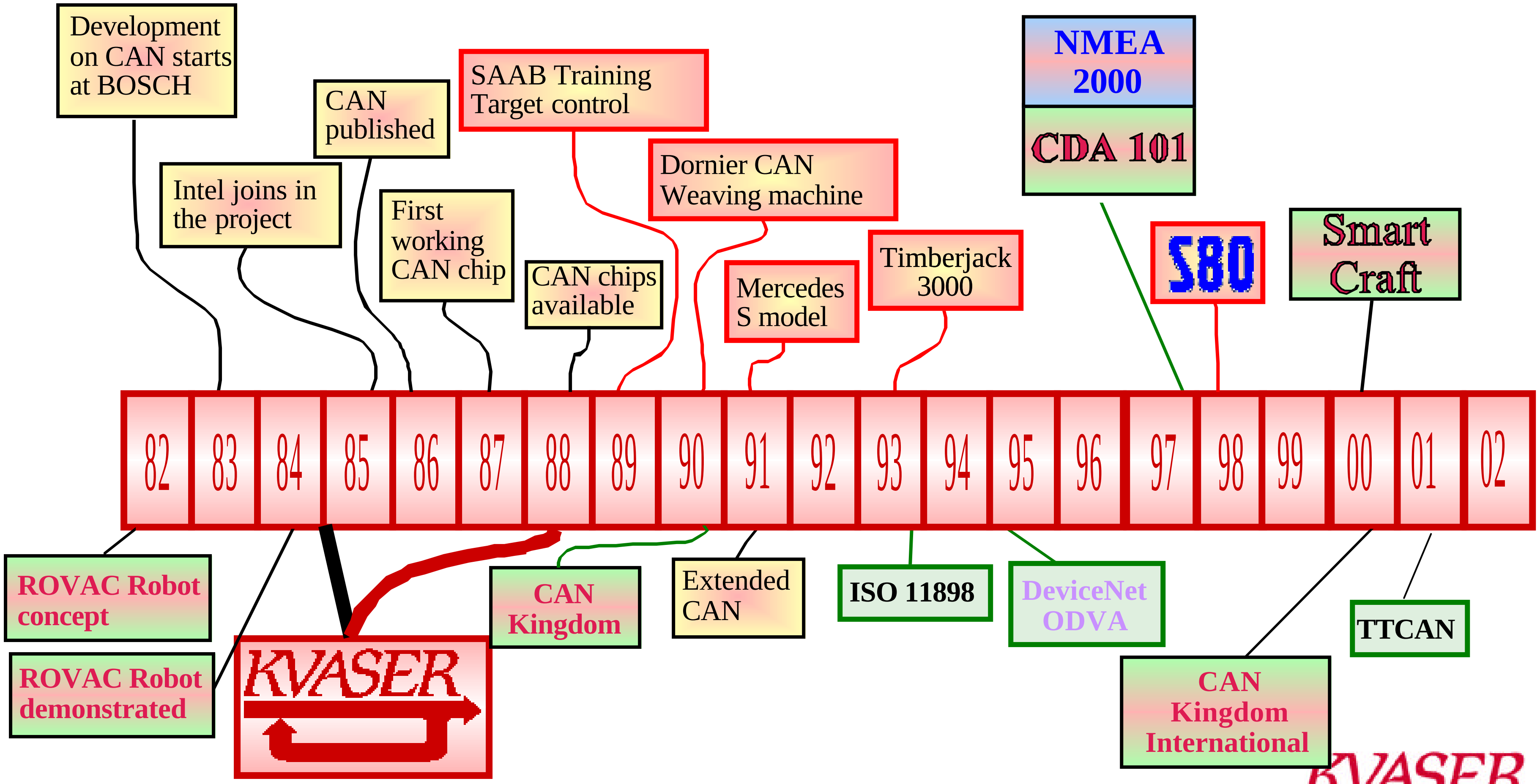
CAN

overview

KVASER

Advanced CAN Solutions

Some CAN Milestones



Development on CAN starts at BOSCH

Intel joins in the project

CAN published

First working CAN chip

CAN chips available

SAAB Training Target control

Dornier CAN Weaving machine

Mercedes S model

Timberjack 3000

NMEA 2000

CDA 101

S80

Smart Craft

ROVAC Robot concept

ROVAC Robot demonstrated

KVASER

CAN Kingdom

Extended CAN

ISO 11898

DeviceNet ODVA

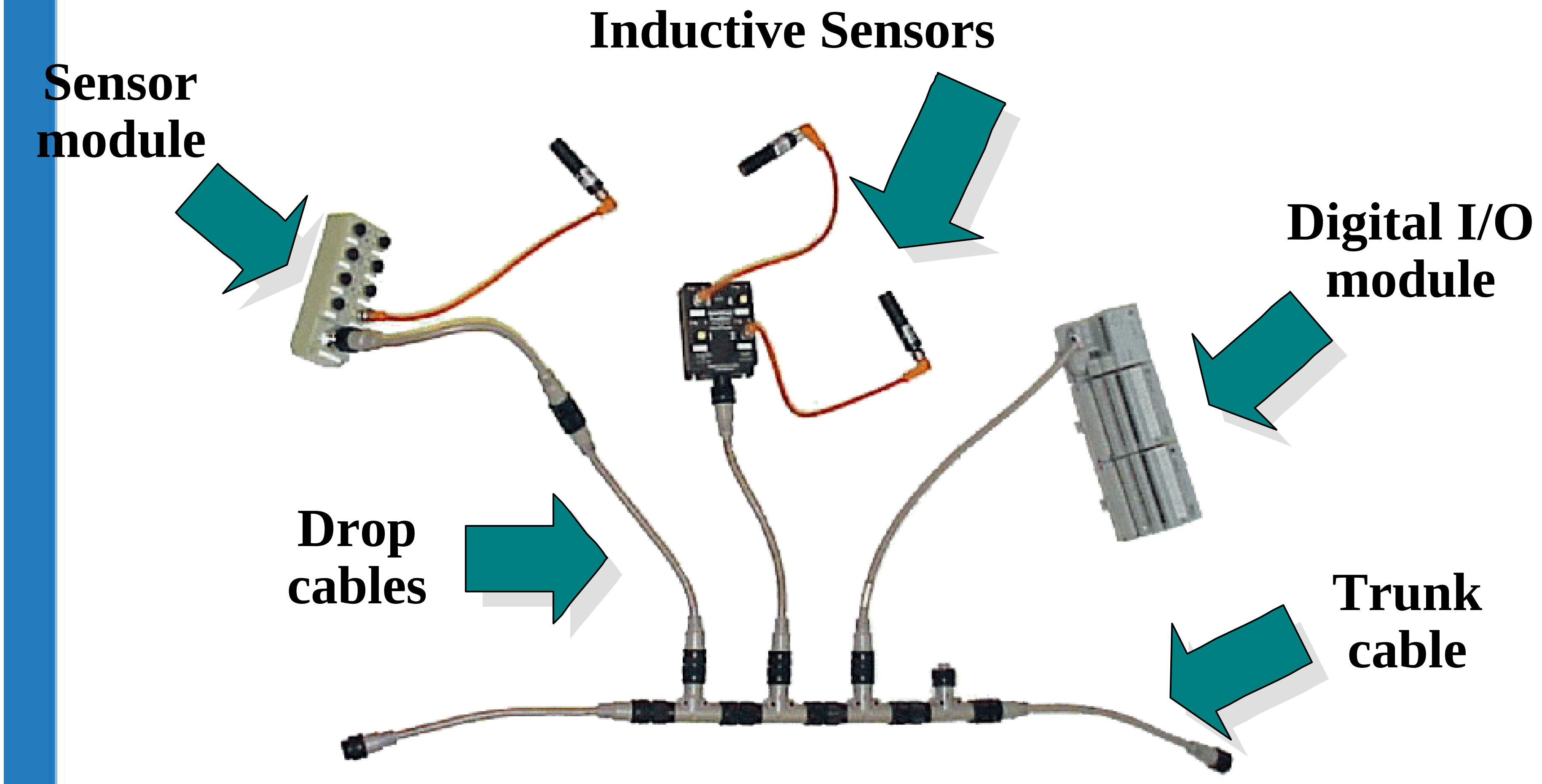
CAN Kingdom International

TTCAN

KVASER

Advanced CAN Solutions

Example of a CAN system

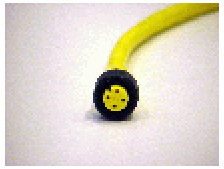


(A segment on a DeviceNet CAN bus)

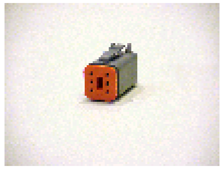
Some CAN cables and modules

9-pole DSUB

DN C-type connector



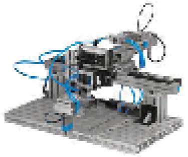
DN mini-style connector



CANHUG connector

I/O module

Sensor module



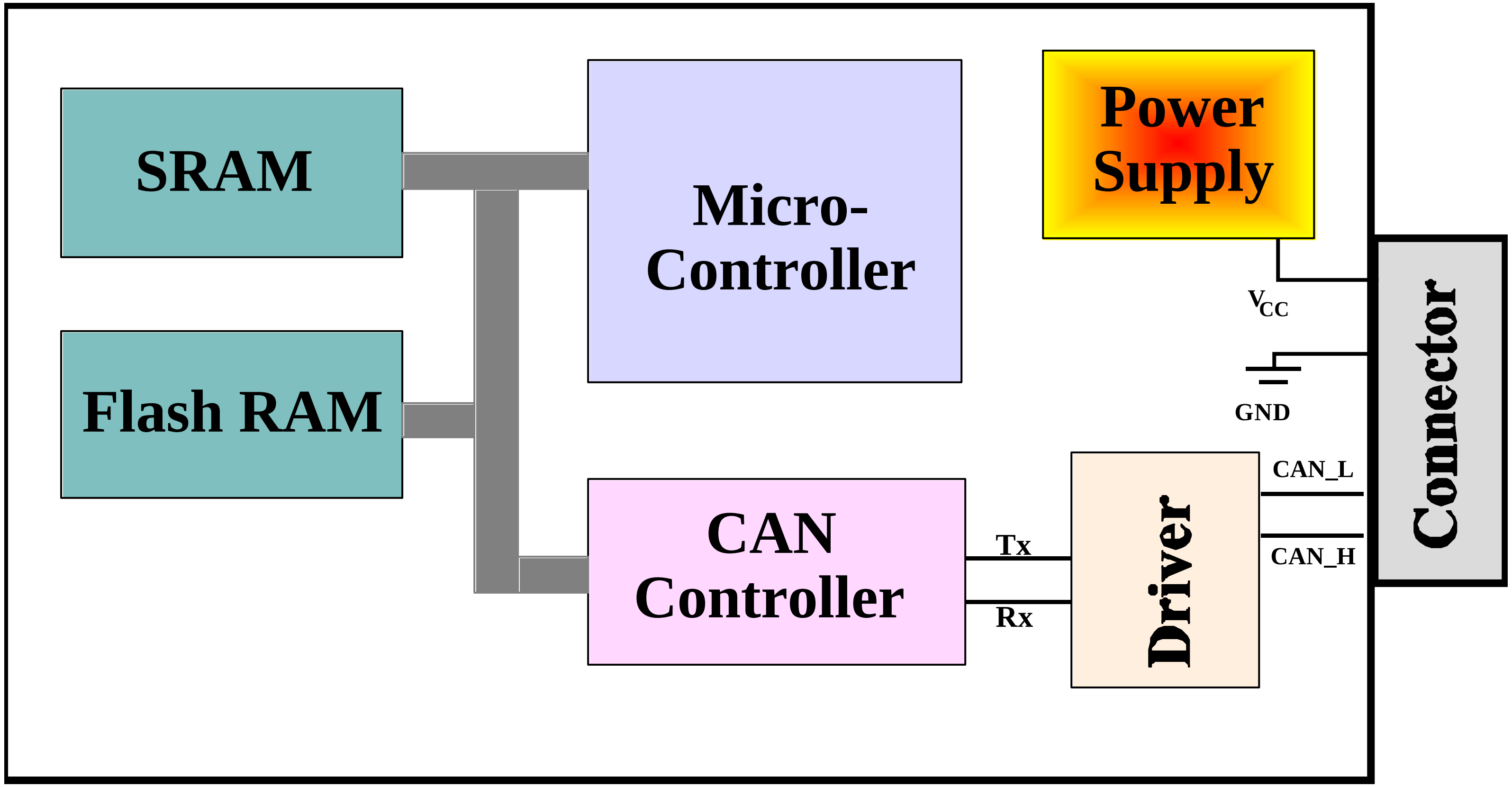
Pneumatic control

Motion Control

KVASER

Advanced CAN Solutions

Schematic of a CAN module



Advanced CAN Solutions

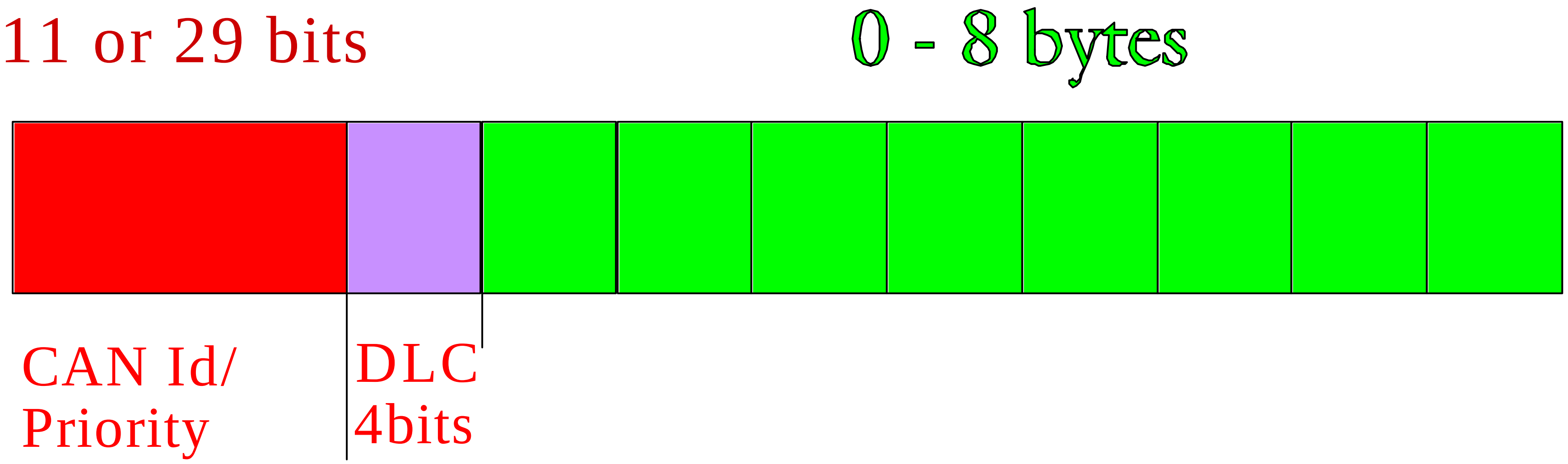
Message Transmission

The message will be sent to all modules by HW.

The transmitter will send messages bit by bit according to the CAN protocol.

- All modules, including the transmitter, will:
  - Be active in all bus activity.
  - Check for errors.
  - Force retransmit of an erroneous message.
- All modules, except for the transmitter, will:
  - Acknowledge a correct message reception.
  - Have a copy of a correct message.

CAN message



Data Frame

The Philosophy of CAN

If a module has information that is needed elsewhere in a system, make it available on the CAN bus

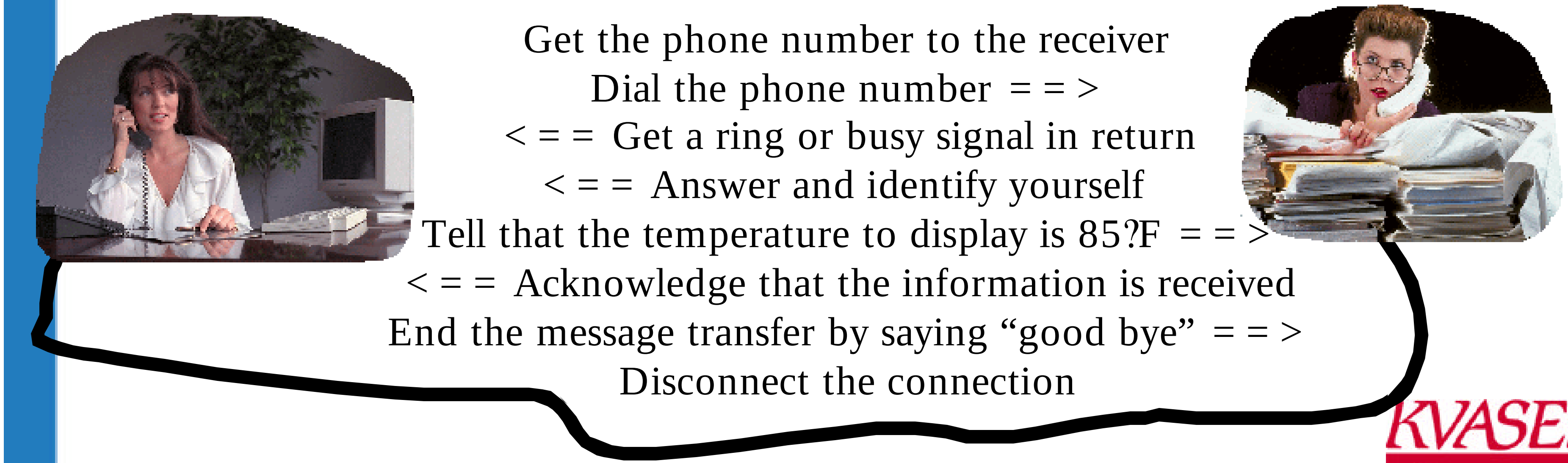
- New information => send it on the CAN bus
- All modules will have identical information
- The receiving modules will store all needed data

In this way you will get a global database, where the same information is available in all modules.

The source and destination of the information are of no importance.

A Telephone Network

- Who is the client?
- Who is the server?
- If the communication fails, what will happen?
- On errors, what can be done?



Get the phone number to the receiver

Dial the phone number = = >

< = = Get a ring or busy signal in return

< = = Answer and identify yourself

Tell that the temperature to display is 85?F = = >

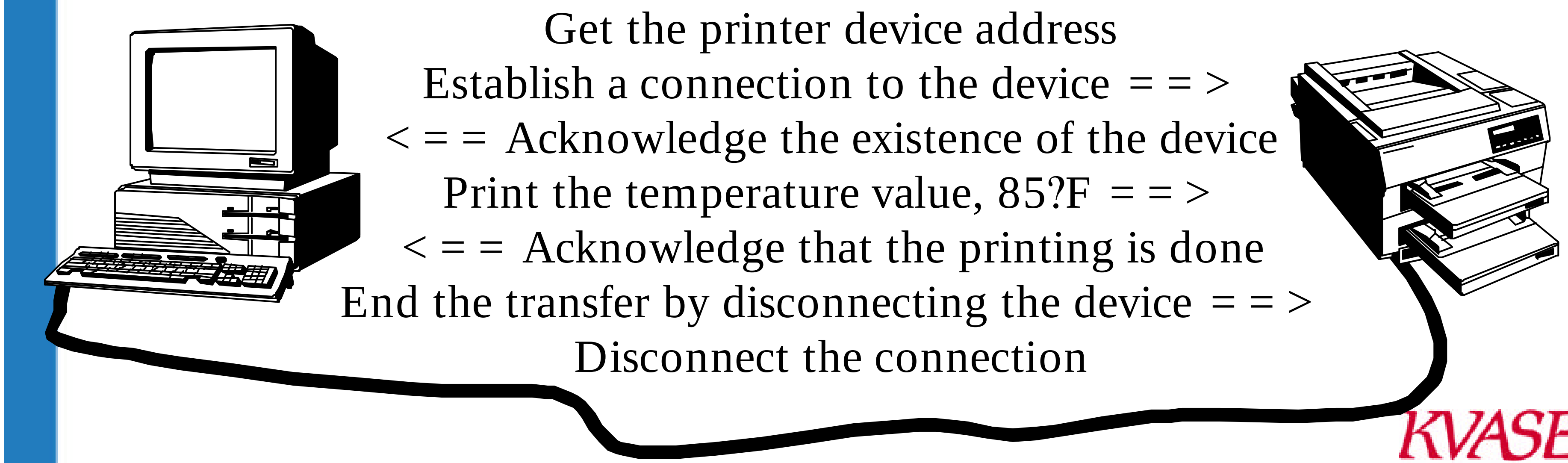
< = = Acknowledge that the information is received

End the message transfer by saying “good bye” = = >

Disconnect the connection

A Computer Network

- Who is the client?
- Who is the server?
- If the communication fails, what will happen?
- On errors, what can be done?



Get the printer device address

Establish a connection to the device = = >

< = = Acknowledge the existence of the device

Print the temperature value, 85?F = = >

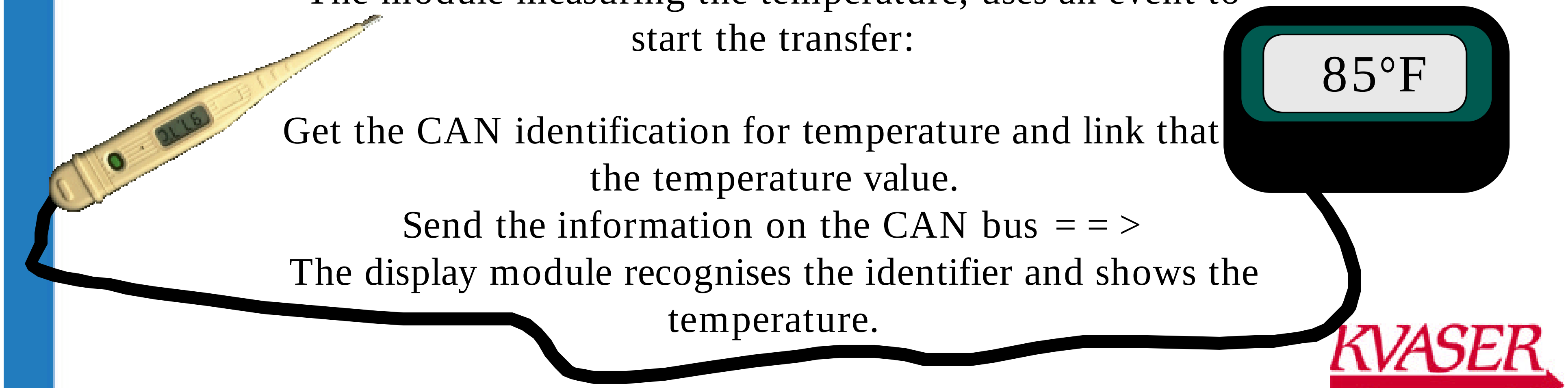
< = = Acknowledge that the printing is done

End the transfer by disconnecting the device = = >

Disconnect the connection

How this is done in a CAN System!

- Who is the client?
- Who is the server?
- If the communication fails, what will happen?
- On errors, what can be done?



The module measuring the temperature, uses an event to start the transfer:

Get the CAN identification for temperature and link that the temperature value.
Send the information on the CAN bus ==>
The display module recognises the identifier and shows the temperature.

The Remote Frame

The CAN hardware of some CAN controllers supports a certain kind of events: The remote frame

- An identifier with request RTR will force the same identifier with data to be sent.
- The remote frame can be transmitted by any module, possible at the same time.
- The DLC (Data Length Code) has to have the same value in the remote frame as in the data frame, but will have no data bytes.

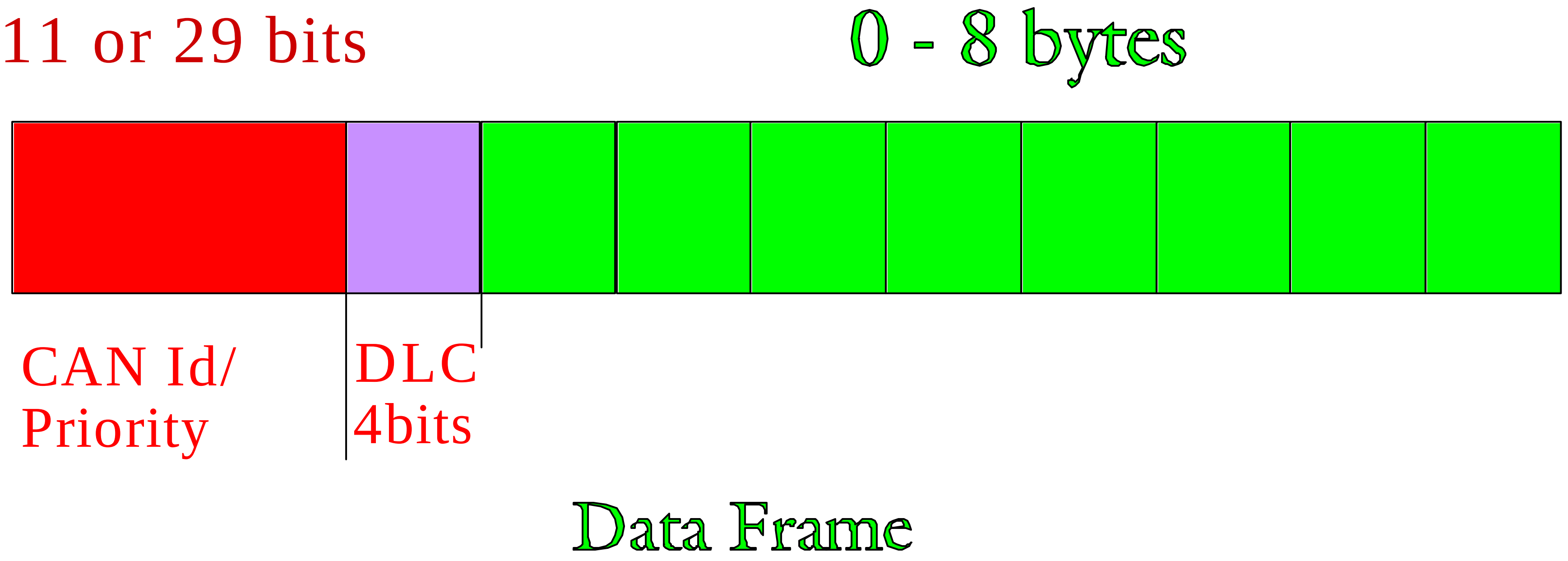
Filtering and Selection

The filtering and selection in the receiver will be according to the information available in the message. The information in a CAN message can belongs to any of three main parts:

- PRIO
  - priority / identifier
- DLC
  - Data Length Code [0..8]
- Data
  - Number of bytes according to the DLC

Filtering and selection can be made in all three parts.

CAN message



KVASER
Advanced CAN Solutions

Filtering and Selection

- Filtration by CAN controllers
  - All CAN controllers support some kind of filtration in the priority / identifier.
  - Some CAN controllers support filtration in some of the data bytes.
- Filtration by module software
  - All kinds of filtration, but the above mentioned, have to be taken care of by module software.
  - The three parts – PRIO, DLC, the data bytes – are available for filtration handled by module software.

KVASER

Advanced CAN Solutions