Features of CiA 447 Application profile for special-purpose car add-on devices

Olaf Pfeiffer, Embedded Systems Academy GmbH

The trend to add more distributed electronics to applications also applies to taxis, emergency response vehicles, governmental vehicles and cars with special controls for handicapped drivers. However, in modern cars it becomes more and more difficult to add electronics, as build-in electronics and airbags are a closed system and all space around the dashboard is occupied. Access to the internal vehicle networks (IVN) is a delicate matter. For safety reasons, the car manufacturers are reluctant to allow direct access to "everyone".

CiA447 defines an open vehicle network with which car manufacturers provide limited access to their internal networks. As all accesses go through a gateway, the gateway can limit the access to those parameters considered "safe". As an example, where supported, buttons and displays of the car can be used by CiA 447 devices. However, the ultimate control remains within the car. In an emergency situation, the car could still override / overwrite the display with required warnings.

To allow easy and simple network connections, a standard connector, extended plug-and-play mechanisms as well as power-down and wake-up controls are defined.

Overview of the functionality provided by CiA 447

On the physical side, CiA 447 specifies to use ISO11898–2. The CiA 447 devices use an 18-pin VDA interface connector that provides up to 4A power and access to the CAN_L and CAN_H lines. An optional 2-pin 16A power connector is defined for devices requiring more than 4A. Other optional signals include things like ignition (KL15), speed pulse signal, PTT – Push-To-Talk signal and audio signals. The CAN bit rate is set to 125 kbit/s.

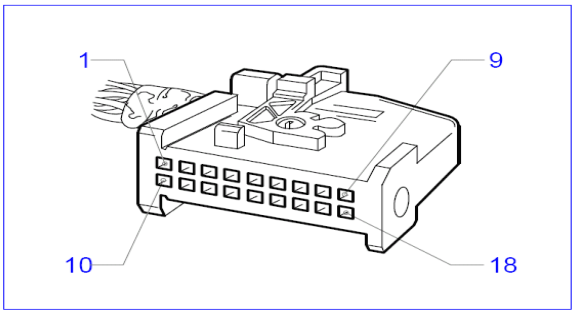

Figure 1: 18-pin VDA interface connector

On the logical side, a CiA447 network is limited up to 16 nodes (versus up to 127 in CiA301 CANopen) allowing for a full SDO mesh implementation with one SDO channel from every node to every node. This requires a custom pre-defined connection set, the CAN message identifiers used for these SDO channels are different from the CiA 301 default. As a result, regular CiA 301 devices such as generic I/O modules do not work in a CiA 447 network, all nodes in a CiA 447 network must be CiA 447 compliant.

The node ID 1 (one) is reserved for and must be used by the gateway. A gateway is mandatory for a CiA447 system to operate, as it is responsible for the network management including detecting and starting nodes. Typically the gateway is provided by the car manufacturer, however, there are also generic gateway solutions that connect to the OBD-II interface of a car to allow the connection of CiA447 compatible devices.

In comparison to "regular" CiA301 CANopen, CiA447 has a few functions, protocols and techniques to satisfy the specific application requirements:

- a high level of plug-and-play functionality

- numerous virtual devices (in V2.0 up to 128)
- signals contain status information
- direct SDO communication between all nodes
- sharing and locking of limited resources, like displays

In the following paragraphs these features are explained.

High level of plug-and-play functionality

One of the early requests for the Taxi application was, that it must be possible to quickly exchange components. Drivers need to be able to do this (like replacing a printer) without requiring the help of a specially trained technician.

This functionality was provided by enhancing the LSS (Layer Setting Services) protocols of CiA305 with the "Fast Scan" protocol. The enhancements allow scanning for unconfigured nodes based on their LSS-ID (128bit device ID from Object Dictionary entry 1018h). Detecting a single node and assigning it a node ID takes less than two seconds using the Fast Scan protocol.

The plug-and-play support in CiA447 is the gateway's responsibility. It detects nodes, assigns node IDs and monitors heartbeat messages to do so.

A known limitation of the plug-and-play support comes from devices with a fixed node ID, which in general are allowed in CiA447. If devices with fixed node IDs are used, the person doing the "plugging" is responsible for ensuring that no other device on the network is currently using a node ID that is already present on the network.

True plug-and-play is therefore only available for those devices supporting the LSS Fast Scan protocols.

Extensive use of Virtual Devices

The concept of using Virtual Devices is essential to CiA447. Initially only up to 32 virtual devices were supported. Since V2.0 of CiA 447 the number of bits used to define virtual devices was extended to support up to 128. These are located in the Object Dictionary at index 6000h at the subindexes one to four.

These entries are one of the first read by communication partners, to determine which node implements what and which resources are available where. As an example, a roof bar controller would check which nodes implement roof bars and from then on focus on the communication with that device, ignoring others.

Virtual device	Type
IVN gateway class 0	Gateway
IVN gateway class 1	Gateway
IVN gateway class 2	Gateway
IVN gateway class 3	Gateway
Fire extinguishing system	Device
Emergency fresh-air system	Device
Power supply	Device
Discrete inputs	Device
Terminal	Device
GPS	Device
Navigation system	Device
Taximeter	Device
Printer	Device
Real time clock (RTC)	Device
Driver identification	Device
Tariff display	Device
Taxi alarm system	Device
Radio	Device
Audio switch	Device
Roof bar light	Device
Roof bar sound	Device
“Blue” light flasher module	Device
Roof bar controller	Controller
Radio controller	Controller
Handicap controller	Controller
Radio hand-free conversation	Device
Tester/tool	Tester
Information signaler	Device
Video	Device
Data recorder	Device
Switch keypad	Generic device
Input output switch	Generic device
Engine control	Device
Active speed control	Device

Figure 2: list of virtual devices

Signals contain status information

In CiA447 signals that normally only have a binary on and off state use two bits to represent a total of four states:

- 00 off
- 01 on
- 10 failure
- 11 signal not available

As a result, consumers of such signals can directly determine if the signal is valid without requiring an extra parameter or mechanism.

A similar setup has been implemented for analog values. In an analog value, all bits set stand for "signal not available" and all bits set minus one for "failure".

For example, object 605Ah defines the displayed vehicle speed using a 16bit value. The value range is from 0 to 0FFEh representing 0.1 local distance unit per hour. Values 0FFFh to FFFDh are reserved, FFFEh stands for failure and FFFFh for signal not available.

Direct node-to-node SDO communication: SDO mesh

In CiA447 each node has 15 or 16 SDO server and 15 SDO clients, allowing each node to send direct SDO requests to any other node. This allows CiA447 devices to independently scan the network for communication partners and get direct access to all their Object Dictionary, not limited to data that is mapped into PDOs.

For the pre-defined connection set (how CAN message identifiers are used for SDOs and PDOs) this means that 480 CAN identifiers (of 2048 available) are used to provide these communication channels. Table 1 and 2 of CiA447-4 show which CAN identifiers are used.

Node-ID	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15	S16
C1	-	241_h	242_h	243_h	244_h	245_h	246_h	247_h	248_h	249_h	$24A_h$	$24B_h$	$24C_h$	$24D_h$	$24E_h$	$24F_h$
C2	250_h	-	252_h	253_h	254_h	255_h	256_h	257_h	258_h	259_h	$25A_h$	$25B_h$	$25C_h$	$25D_h$	$25E_h$	$25F_h$
C3	260_h	261_h	-	263_h	264_h	265_h	266_h	267_h	268_h	269_h	$26A_h$	$26B_h$	$26C_h$	$26D_h$	$26E_h$	$26F_h$
C4	270_h	271_h	272_h	-	274_h	275_h	276_h	277_h	278_h	279_h	$27A_h$	$27B_h$	$27C_h$	$27D_h$	$27E_h$	$27F_h$
C5	340_h	341_h	342_h	343_h	-	345_h	346_h	347_h	348_h	349_h	$34A_h$	$34B_h$	$34C_h$	$34D_h$	$34E_h$	$34F_h$
C6	350_h	351_h	352_h	353_h	354_h	-	356_h	357_h	358_h	359_h	$35A_h$	$35B_h$	$35C_h$	$35D_h$	$35E_h$	$35F_h$
C7	360_h	361_h	362_h	363_h	364_h	365_h	-	367_h	368_h	369_h	$36A_h$	$36B_h$	$36C_h$	$36D_h$	$36E_h$	$36F_h$
C8	370_h	371_h	372_h	373_h	374_h	375_h	376_h	-	378_h	379_h	$37A_h$	$37B_h$	$37C_h$	$37D_h$	$37E_h$	$37F_h$
C9	440_h	441_h	442_h	443_h	444_h	445_h	446_h	447_h	-	449_h	$44A_h$	$44B_h$	$44C_h$	$44D_h$	$44E_h$	$44F_h$
C10	450_h	451_h	452_h	453_h	454_h	455_h	456_h	457_h	458_h	-	$45A_h$	$45B_h$	$45C_h$	$45D_h$	$45E_h$	$45F_h$
C11	460_h	461_h	462_h	463_h	464_h	465_h	466_h	467_h	468_h	469_h	-	$46B_h$	$46C_h$	$46D_h$	$46E_h$	$46F_h$
C12	470_h	471_h	472_h	473_h	474_h	475_h	476_h	477_h	478_h	479_h	$47A_h$	-	$47C_h$	$47D_h$	$47E_h$	$47F_h$
C13	540_h	541_h	542_h	543_h	544_h	545_h	546_h	547_h	548_h	549_h	$54A_h$	$54B_h$	-	$54D_h$	$54E_h$	$54F_h$
C14	550_h	551_h	552_h	553_h	554_h	555_h	556_h	557_h	558_h	559_h	$55A_h$	$55B_h$	$55C_h$	-	$55E_h$	$55F_h$
C15	560_h	561_h	562_h	563_h	564_h	565_h	566_h	567_h	568_h	569_h	$56A_h$	$56B_h$	$56C_h$	$56D_h$	-	$56F_h$
C16	570_h	571_h	572_h	573_h	574_h	575_h	576_h	577_h	578_h	579_h	$57A_h$	$57B_h$	$57C_h$	$57D_h$	$57E_h$	-

Table 1 — SDO CAN-IDs for request client © to server (S)

Node-ID	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15	S16
C1	-	$1C1_h$	$1C2_h$	$1C3_h$	$1C4_h$	$1C5_h$	$1C6_h$	$1C7_h$	$1C8_h$	$1C9_h$	$1CA_h$	$1CB_h$	$1CC_h$	$1CD_h$	$1CE_h$	$1CF_h$
C2	$1D0_h$	-	$1D2_h$	$1D3_h$	$1D4_h$	$1D5_h$	$1D6_h$	$1D7_h$	$1D8_h$	$1D9_h$	$1DA_h$	$1DB_h$	$1DC_h$	$1DD_h$	$1DE_h$	$1DF_h$

C3	$1E0_h$	$1E1_h$	-	$1E3_h$	$1E4_h$	$1E5_h$	$1E6_h$	$1E7_h$	$1E8_h$	$1E9_h$	$1EA_h$	$1EB_h$	$1EC_h$	$1ED_h$	$1EE_h$	$1EF_h$
C4	$1F0_h$	$1F1_h$	$1F2_h$	-	$1F4_h$	$1F5_h$	$1F6_h$	$1F7_h$	$1F8_h$	$1F9_h$	$1FA_h$	$1FB_h$	$1FC_h$	$1FD_h$	$1FE_h$	$1FF_h$
C5	$2C0_h$	$2C1_h$	$2C2_h$	$2C3_h$	-	$2C5_h$	$2C6_h$	$2C7_h$	$2C8_h$	$2C9_h$	$2CA_h$	$2CB_h$	$2CC_h$	$2CD_h$	$2CE_h$	$2CF_h$
C6	$2D0_h$	$2D1_h$	$2D2_h$	$2D3_h$	$2D4_h$	-	$2D6_h$	$2D7_h$	$2D8_h$	$2D9_h$	$2DA_h$	$2DB_h$	$2DC_h$	$2DD_h$	$2DE_h$	$2DF_h$
C7	$2E0_h$	$2E1_h$	$2E2_h$	$2E3_h$	$2E4_h$	$2E5_h$	-	$2E7_h$	$2E8_h$	$2E9_h$	$2EA_h$	$2EB_h$	$2EC_h$	$2ED_h$	$2EE_h$	$2EF_h$
C8	$2F0_h$	$2F1_h$	$2F2_h$	$2F3_h$	$2F4_h$	$2F5_h$	$2F6_h$	-	$2F8_h$	$2F9_h$	$2FA_h$	$2FB_h$	$2FC_h$	$2FD_h$	$2FE_h$	$2FF_h$
C9	$3C0_h$	$3C1_h$	$3C2_h$	$3C3_h$	$3C4_h$	$3C5_h$	$3C6_h$	$3C7_h$	-	$3C9_h$	$3CA_h$	$3CB_h$	$3CC_h$	$3CD_h$	$3CE_h$	$3CF_h$
C10	$3D0_h$	$3D1_h$	$3D2_h$	$3D3_h$	$3D4_h$	$3D5_h$	$3D6_h$	$3D7_h$	$3D8_h$	-	$3DA_h$	$3DB_h$	$3DC_h$	$3DD_h$	$3DE_h$	$3DF_h$

Node-ID	**S1**	**S2**	**S3**	**S4**	**S5**	**S6**	**S7**	**S8**	**S9**	**S10**	**S11**	**S12**	**S13**	**S14**	**S15**	**S16**
C11	$3E0_h$	$3E1_h$	$3E2_h$	$3E3_h$	$3E4_h$	$3E5_h$	$3E6_h$	$3E7_h$	$3E8_h$	$3E9_h$	-	$3EB_h$	$3EC_h$	$3ED_h$	$3EE_h$	$3EF_h$
C12	$3F0_h$	$3F1_h$	$3F2_h$	$3F3_h$	$3F4_h$	$3F5_h$	$3F6_h$	$3F7_h$	$3F8_h$	$3F9_h$	$3FA_h$	-	$3FC_h$	$3FD_h$	$3FE_h$	$3FF_h$
C13	$4C0_h$	$4C1_h$	$4C2_h$	$4C3_h$	$4C4_h$	$4C5_h$	$4C6_h$	$4C7_h$	$4C8_h$	$4C9_h$	$4CA_h$	$4CB_h$	-	$4CD_h$	$4CE_h$	$4CF_h$
C14	$4D0_h$	$4D1_h$	$4D2_h$	$4D3_h$	$4D4_h$	$4D5_h$	$4D6_h$	$4D7_h$	$4D8_h$	$4D9_h$	$4DA_h$	$4DB_h$	$4DC_h$	-	$4DE_h$	$4DF_h$
C15	$4E0_h$	$4E1_h$	$4E2_h$	$4E3_h$	$4E4_h$	$4E5_h$	$4E6_h$	$4E7_h$	$4E8_h$	$4E9_h$	$4EA_h$	$4EB_h$	$4EC_h$	$4ED_h$	-	$4EF_h$
C16	$4F0_h$	$4F1_h$	$4F2_h$	$4F3_h$	$4F4_h$	$4F5_h$	$4F6_h$	$4F7_h$	$4F8_h$	$4F9_h$	$4FA_h$	$4FB_h$	$4FC_h$	$4FD_h$	$4FE_h$	-

Table 2 — SDO CAN-IDs for response server (S) to client (C)

Resource sharing

CiA447 allows sharing a car's resources, such as a dashboard display. In order to provide controlled access, a locking mechanism with confirmation is required to manage the access to such shared resources.

The locking and release mechanism for the "Display 1" is used as an example here to demonstrate how the locking mechanism works.

Object 6098h is an 8bit command for Display 1. For simplicity in this example, Display 1 access is provided by the gateway (node ID 1). Using an SDO write request, any CiA447 node can now write a command to the display control. Writing a value of 1 requests a lock (device wants to access display), writing a value of 2 requests the release of a previously granted lock.

After writing a lock request, the device needs to read the display status. Object 6097h is an 8bit status value for Display 1. Using an SDO read request, any CiA447 node can read 6097h to determine the display status. A value of 2 to 17 means that it is currently locked to node 1 to 16.

So by reading back the status, a node requesting a lock can check if a written lock request was successful. Once a lock is established, the node "owning" the lock can use the resource, in this case writing to the display.

Such locking mechanisms have been defined for all displays and buttons.

Note that it is the responsibility of the resource manager to determine potential priorities, including canceling locks. A dashboard controller will prioritize messages from the car over any other CiA447 request. If an internal warning needs to be displayed, an established lock may be canceled.

Olaf Pfeiffer

Embedded Systems Academy
Bahnhofstr. 17
30890 Barsinghausen
Tel: +49-5105-52 27 89-7
Fax: +4-5105-58407-35
opfeiffer@esacademy.com
www.esacademy.com

Appendix – Selection of gateway signals defined by CiA447

The following is a summary of some of the signals defined for gateways. Most of them are optional, so it is up to the individual car/gateway manufacturer to decide if they want to provide this signal or not.

PDO Data that transmits periodically (typically every 100 ms) or on change-of state

- Car body electronics
  ignition, flaps (doors and hatches), illumination level and car lights, wiper system, central locking system, belt buckles, turn indicator, seat adjustments, accident detection, horn, anti-theft system
- Car power train
  engine and wheel data
- Steering wheel and dashboard inputs
  steering wheel, dashboard and console switches
- HID user terminal
  user terminal (display/switches), function and control keys
- GPS position
  latitude and longitude
- Status of add-on devices

SDO Data that needs to be requested

- Car body electronics
  VIN (vehicle identification number), steering wheel angle, temperatures and climate control
- Car power train
  odometer, displayed speed, tank, fuel consumption, gear

SDO Commands that may be supported

- Car body electronics
  lights and illumination, locking system, windows, doors, hatches, seat adjustment, wiper system, climate control, terminal displays