Harmonizing CANopen bootloaders

CAN in Automation (CiA) has almost finalized the CiA 710 document specifying a generic CANopen bootloader function. The bootloader enables secure managing of the device's firmware updates via CAN. It is suitable for CANopen CC (classic) and CANopen FD devices.

(Source: Adobe Stock)

The CAN-based higher-layer protocol CANopen is used in embedded control networks in many application fields. An essential aspect of modern embedded networking is the bootloader, which facilitates firmware updates for embedded devices, e.g. within a CANopen network. CAN in Automation has almost finalized the development of a harmonized handling strategy for bootloader applications in a CAN/CANopen environment, known as CiA 710. The intent of the specification is to allow software tools from any manufacturer to update any CAN/CANopen device's firmware by means of the same basic principles. Of course, among others, this approach considers cybersecurity aspects.

What is a bootloader?

A bootloader is a small program that is, by using a minimum of resources, responsible for initializing the hardware and loading the main firmware of a device during its startup. In the context of CANopen, the CANopen bootloader is responsible for managing firmware updates via CAN. CANopen bootloaders ensure that devices can receive new firmware versions via CAN, and can validate and install them. Supporting a CANopen bootloader enhances flexibility for system maintainers. A new version of device firmware may remove identified security weaknesses or may add new application-related functions. Thus, the bootloader extends the longevity of devices.

Prior to the start-up of the system, tools or CANopen host controllers double-check the configuration as well as the software version of the CANopen devices. During that task they may identify that an update of the configuration or the entire CANopen device's firmware is required. With the aim to simplify and generalize the way of running a firmware update via CAN, the CiA TF (task force) Generic bootloader has specified a harmonized procedure.

Key features of CANopen bootloader

To allow a harmonized control of the firmware update via CAN, the CiA 710 introduces a finite state automation (FSA) for embedded devices. The FSA defines a predictable device behavior for all FSA states. Thus, configuration tools or host controllers switching the FSA states of CANopen devices can expect a given device behavior. The FSA specified in CiA 710 (see Figure 1) differentiates between the basic modes of operation, a bootloader mode (BM) and an application mode (AM). When an embedded device is switched on, at first, it transits to the "bootloader initializing" state. The existence of a valid application program to be started is checked by means of the autostart information, referred to by data object $1F59_h$. If no valid program is found or starting a program is not intended, the device switches to the bootloader mode by default. As usual for CANopen devices, then the device enters the "Initialization" state, where the setup operations, including CAN controller initialization and bit rate configuration are performed, in accordance to CiA 1301. In bootloader mode (BM), the device expects and verifies the user authentication. Legal tools or CANopen host controllers have therefore the possibility to identify themselves to the CANopen device in bootloader mode. The successful identification initiates the CANopen device being ready to accept new application programs or configuration data. Therefore, the CANopen device enters the "BM allow application download" state, where it waits to receive a new application program. Prior to the data transfer of the firmware via CAN, tools or CANopen host controllers can learn the CANopen device's attributes such as flash status and flash operation times. This enables them to adapt their behavior – in particular their internal time-outs – accordingly. After a successful program transfer, tools or CANopen host controllers force the CANopen device to leave the bootloader mode, and (typically) to start the new application program. When the new application program has been started, the device runs in application mode (AM).

If a transition back to the bootloader mode is required (e.g., to modify the configuration or the entire application program) tools or host controllers have to pass the security check again, prior to initiate switching back to the bootloader mode. In addition, the status of the currently running device's application has to be taken into account. Of course, switching to BM is only possible, in case the device's application ▷

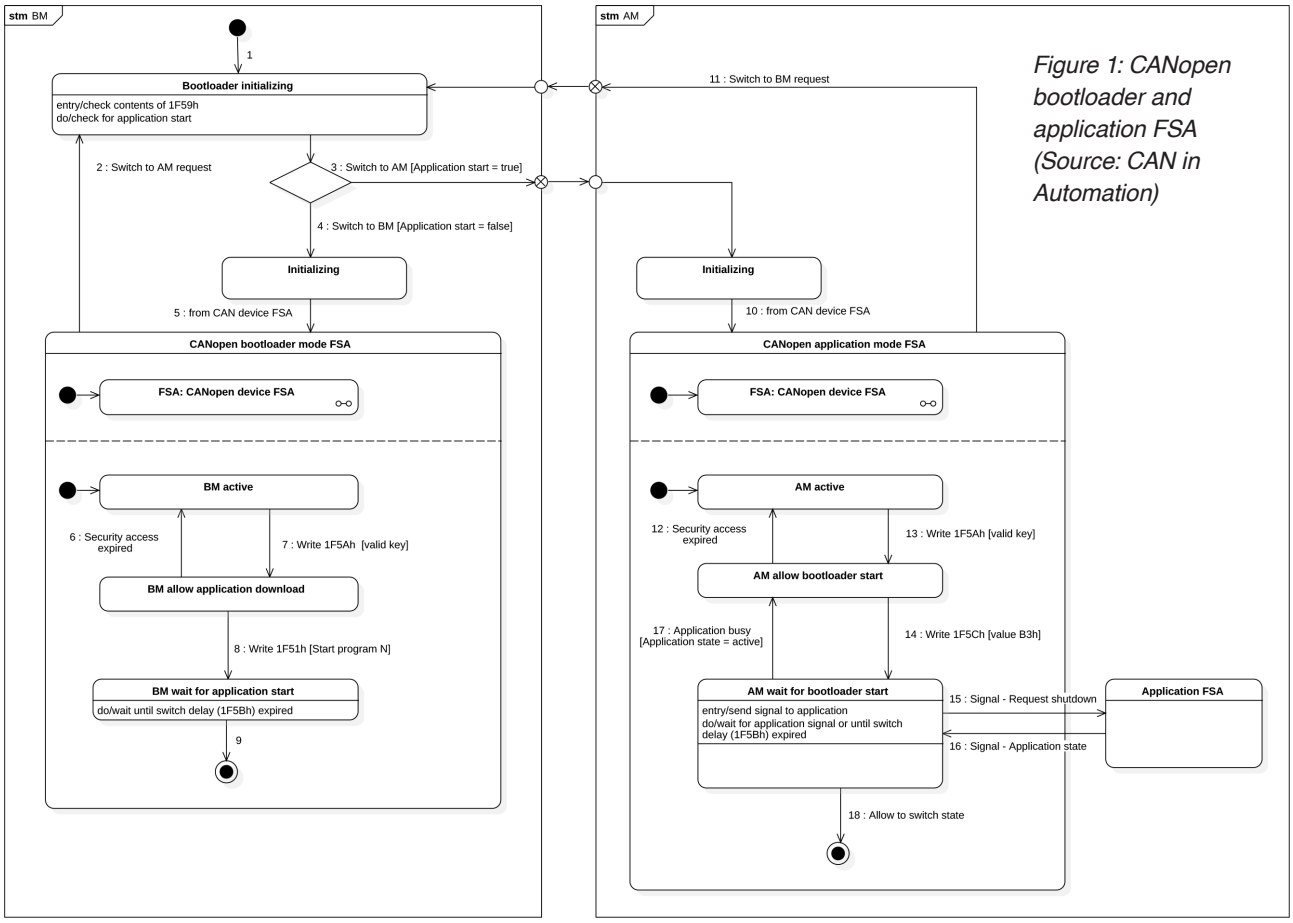

Figure 1: CANopen bootloader and application FSA (Source: CAN in Automation)

state allows a safe mode switching from AM to BM. To avoid "loosing of device" due to a non-successful firmware transfer, a roll-back respectively recovery function has been considered in CiA 710. In case of an error scenario during the firmware update, the device is not lost but will start with a pre-configured default application program. Overall, CiA 710 enables tools and CANopen host controllers to orchestrate the operating states of the device, to ensure secure firmware updates, and to maintain the integrity and reliability of the system operation.

CiA 710 pre-implementation

Microcontrol (Germany) has pre-implemented the generic CANopen bootloader specification (CiA 710) in its CANopen bootloader protocol stack. It is used to securely update software on devices in a CANopen (FD) network without having to remove them from their environment. Versatile configuration options enable individual customization to a target product. The bootloader has been implemented to meet low storage requirements. It comprises a reduced object dictionary and supports NMT (network management), SDO (service data object), emergency, heartbeat as well as LSS (layer setting services) functionality. Up to four separate sections for storage of programs and data can be defined. A defined API (application programming interface) facilitates adjustment to a flash memory of the target hardware. Flash drivers for various controllers (e.g. STM32 series) are part of the CANpie driver. Ready-to-run examples for different demo boards are in the scope of delivery. The solution has been introduced at the Embedded World 2024 trade show in Nuremberg (Germany). *of*

Conclusion

A bootloader is a fundamental function in modern embedded control systems, offering a flexible method to react on modified system/device requirements, by means of a device's firmware update over CAN. The CiA 710 specification represents a significant step forward in this field, providing a harmonized handling that supports interoperability, security, and devices' operational efficiency. CiA 710 considers various use cases such as firmware update at end-of-line production, diagnostic scenarios in laboratories, or updates over the air in the field, via a CANopen gateway that is not necessarily embedded in the network's CANopen host controller. As the industry evolves, adhering to CiA 710 maintains the reliability and performance of embedded systems across many applications. CiA 710 has almost been finalized. A CiA-internal release of CiA 710 as DSP (draft standard proposal) is expected in the second half of 2024.

Author

Reiner Zitzmann
CAN in Automation
headquarters@can-cia.org
www.can-cia.org