RTC-CANopen is a device profile for RT (robotic technology) components, which consists of modularized robotics functional elements, to run on a CANopen environment.

Traditionally robots often were called “industrial robots”, mainly used in production and manufacturing. But in a changed social environment with advanced robotics technologies, practical applications of “service robots”, which perform their operations nearer to human beings, will become acceptable. Non-manufacturing business such as medical and healthcare fields, and personal fields such as home automation are the targets for such service robots. However, environment for human life is very complicated, so service robots need to recognize various objects stably and respond accordingly to dynamically changing environments or situations in order to work at places near humans. Furthermore, high safety is also required not to cause adjusting people harm. Required functions will be expected to be of great variety, since targeted places and users range much wider.

Because of the background described above, it is considered that traditional development methods to make out complete robots individually will meet a limitation when applied for service robots. Therefore a new approach for robotics development has been proposed. In this approach, robots are divided into various robotics elements as components and those elements are combined to be a complete robotics system.

RT middleware is also being developed as a platform on which RT functional elements (RT components) run.

Since RT middleware enables reuse and switch of RT components, it becomes possible to respond to various requirements from users by combination of RT components. And it also becomes possible to develop parts individually and make development projects smaller. In the result, each vendor can concentrate on those parts they are good at.

It is defined that an RT component is a robotics component, which provides some meaningful functionality. Thus not only devices such as servomotors, various sensors, cameras, and so on are called RT components, but also combinations of these devices such as motor-driven carts and arms. In addition, RT components are not limited to modules embedded in hardware, but software modules such as different control algorithms become RT components as well.

What RT components should provide, for example interface specifications and component models, is defined in the standard specification (Robotic Technology Component Specification) at OMG (Object Management Group). OMG is an international nonprofit standardization consortium standarding object oriented technologies. However, this standard specification defines only functionality and state transitions, which RT components should provide. Details about communication techniques are not described.

RTC-CANopen is a profile for using CANopen in order to communicate among RT components. This profile describes necessary information to make existing CANopen devices work as RT components. RTC-CANopen enables mutual collaboration between various devices connected to native buses set up for CANopen, and RT components running on general-purpose networks. It becomes easy to combine them and build a robotic system. A system built on RTC-CANopen consists of DeviceRTCs running on an embedded MPU (microprocessor unit), ProxyRTCs managing DeviceRTCs on RT middleware, and RTC-CANopen servers managing the whole system. ProxyRTCs and DeviceRTCs have correspondence relations. Using such relations, when a ProxyRTC connects to another ProxyRTC, their corresponding DeviceRTCs transmit data directly via a native bus.

RTC-CANopen profile is defined by adding necessary information for devices to run as RT components, to the existing CANopen profile. Therefore, it is compatible with existing CANopen compliant products, and makes extension for robots easier. A set of tools is also provided, which generates robotics system definition files from definition files of CANopen (DCF files) automatically. RTC-CANopen enables improvement of robotic system development efficiency and realization of various requirements from users.

RTC-CANopen has following key features:

- Lightweight: Implements light RT middleware providing only necessary functions for embedded devices.
- Robustness: Improves system robustness introducing monitoring methods for operations on native buses.
- Real-time: Implement fast and reliable communication between components with the aid of advantages of native buses.
- Reusability: Improves reusability of each element, since both hardware and software can be dealt as components.
- Flexibility: It is possible to change system configurations flexibly, switching parts or changing combinations.