Proceedings of the International Conference on Industrial & Mechanical Engineering and Operations Management
Dhaka, Bangladesh, December 26-27, 2020

Synchronous Control of Maxon EPOS4 Positioning Controller Using NI LabVIEW and NI Industrial Communications for CANopen

Asif Al Zubayer Swapnil and Mohammad Habibur Rahman
Department of Mechanical Engineering
University of Wisconsin-Milwaukee
Milwaukee, WI, USA
aswapnil@uwm.edu, rahmanmh@uwm.edu

Abstract

The EPOS range of digital position controllers for Brushed and Brushless DC (BLDC) motors from Maxon group is popular in industrial and robotics applications. The EPOS4 series provides option for using CANopen, a higher layer protocol, and device profile specifications based on Controller Area Network (CAN) bus for communication with a master controller. CANopen protocol uses two modes of communication, Service Data Objects (SDO) and Process Data Objects (PDO). SDO communication can be used to configure and operate EPOS4 devices in the network in a point-to-point manner over the CAN-bus, making it difficult for use in real-time motion control applications, and impossible for synchronous control of multi-motor systems over the same bus. PDO communication solves the problem by allowing mapped PDO messages and synchronization objects. In this research, a CANopen master is developed in National Instruments' LabVIEW programming environment using NI-Industrial Communications for CANopen module with custom PDO mappings, Network management (NMT) and synchronization objects. This CANopen master is capable of controlling multiple EPOS4 position controllers synchronously in Cyclic Synchronous Torque (CST), and Cyclic Synchronous Position (CSP) mode.

Keywords
CANopen, LabVIEW, Industrial Communication, Positioning Controller, EPOS

1. Introduction
In robotics, servo motors with precise position control is a matter of utmost importance. Historically, the servo motors are driven by Field Oriented Control method using servo amplifiers, and position feedback from encoders mounted on motor shaft is used in a central position controller to precisely control robot motion. In industrial applications, servo drives with integrated position control have become more common. These drives integrate the motor drive, feedback processing, and position control for one motor axis in one integrated package. And multiple drives can be networked with industrial networking and Fieldbus protocols for complicated and distributed motion control with motor drives mounted close to the motors reducing cable length, power loss, and noise. But due to the size and weight constraints, these controllers are not often used in robotics.

In recent times with the development of embedded electronics, lightweight and small-scale integrated position controllers, e.g., Maxon EPOS, and Odrive have become available. The use of these position controllers can lead to weight and energy-saving benefits as well as better performance with local position control. In the BioRobotics Lab at the University of Wisconsin-Milwaukee, we have developed a functional exoskeleton robot and a small manipulator robot which are currently driven by servo amplifiers. A central position controller developed in National Instrument LabVIEW Real-Time is used to synchronously control all joints of these two robots. In this research, we explore a way to get Maxon EPOS4 position controllers with CANopen interface work with LabVIEW in order to evaluate the possibility of using these position controllers to operate these robots in the future.

2. Literature Review

2.1 CAN-bus and CANopen protocol

The Controller Area Network or CAN-bus is a serial bus protocol used in the automotive industry since its introduction in 1986 by Robert Bosch GmbH (Pfeiffer et al. 2008). The CAN-bus specifications cover the lower two layers (physical and datalink layers in the OSI model of communication protocols. In automotive industry different manufacturers have their own proprietary protocols built upon this specification to take over the functionality of higher layers of the OSI model. For use in automation, 'CAN in Automation' of 'CiA' has developed a standard protocol called CANopen (CiA Draft Standard 301) that takes care of the upper 5 layers (Network, Transport, Session, Presentation, and Application) of the OSI model (National Instruments 2020). In the CANopen protocol published in the CiA 301 specifications, the following concepts are used to describe the communication method used by CANopen.

Object Dictionary (OD)

In a CANopen device, an Object Dictionary (OD) must be used to hold all the configuration data and process data in an organized manner. The object dictionary can be conceptualized as a 3-dimensional table with a 16-bit index and 8-bit sub-index to address all of the data contained in the dictionary. CiA 301 standard defines some of these indices (CiA Draft Standard 301) for a predetermined data type, and other ranges of indices are open for manufacturers to implement required process data and configurations. By accessing this object dictionary or each node in the network, the relevant information can be read or written on that device (CiA Draft Standard 301, National Instruments 2020).

CANopen messages

In CANopen message frame is formatted based on CAN data frame format where 11-bit CAN ID is repurposed as a communication object identifier, otherwise known as COB-ID, combining a 4-bit function code for the CANopen message with a 7-bit node-id unique for each node of the network (CiA Draft Standard 301). This specification makes sure the same message from different nodes can be identified uniquely by their COB-ID to avoid conflicts in the network. The 7-bit node-id makes it possible for a CANopen network to contain up to 127 nodes.

SDO and PDO

The two most important types of communication objects used in CANopen are Service Data Objects (SDO) and Process Data Objects (PDO). These are used for transferring data between two or multiple nodes in a CANopen network. Depending on their characteristics, they are used exclusively in different modes of network operation. An SDO is used for peer-to-peer communication in a client-server model as seen in figure 1 (Maxon motor ag. 2019). In SDO communication, any node in the network can act as a server for their object dictionary, and another node acts as a client to request data to be readortwritten from a specific index and sub-index of the server's object dictionary. For each data transaction, the client initiates the SDO communication with a CANopen message with COB-ID formed by combining the CAN ID 600h + node ID of the server. The server with the correct node ID responds with a reply message with COB-ID formed by CAN-ID 580h + node ID of the server. Depending on the type of data transfer, SDO write or SDO read, one message will contain the data, and another will contain an acknowledge message.

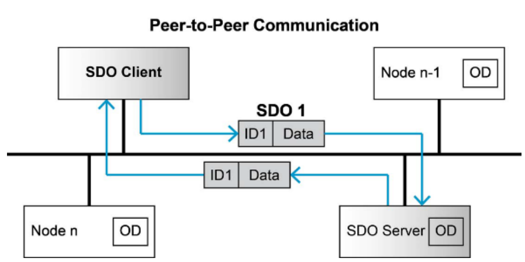

Figure 1. Service Data Object (SDO) [Adapted from Maxon motor ag "EPOS4 Positioning Controllers Communication Guide", Edition 2019-11, Doc ID: rel8759]

For real-time control operations, SDO communication is not suitable as it requires a lot of time and bandwidth for updating single process variable to each node of the network. In those applications, PDO based communication is used, where the network nodes work in a producer-consumer model. In this case, a PDO message is produced by the consumer and based on COB-ID of the message; it is consumed by multiple consumers in the network as seen in figure 2 (Maxon motor ag. 2019). A PDO producer can generate a PDO message based on different triggers. The triggers can be external events, for example, status change of a digital input, it can be a remote request from a node in the network, or it can be periodic updates using timed SYNC messages in the network.

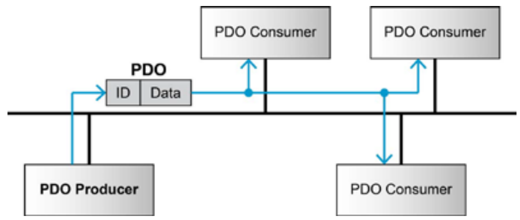

Figure 2. Process Data Object (PDO) [Adapted from Maxon motor ag "EPOS4 Positioning Controllers Communication Guide", Edition 2019-11, Doc ID: rel8759]

PDO Mapping and PDO Linking

In CiA301 specification, the PDO messages are classified in two types, Receive PDO (RPDO) and Transmit PDO (TPDO) (Pfeiffer et. al. 2008, CiA Draft Standard 301). A TPDO is a PDO configured on the producer with its own COB-ID, and an RPDO is a PDO configured in a PDO consumer with a COB-ID set to match the COB-ID of a TPDO in another PDO producer in a network. The process of configuring the RPDO on the consumer to match the correct TPDO on the producer is called PDO linking. To use PDO communication to update process data of different nodes in the network, each PDO messages in the network needs to be mapped to a relevant entry on the node's OD with correct Index and Sub-index. Typically, TPDOs are mapped to process variables that the node updates by itself during operation, and RPDOs are mapped to process variables updated by the central controller.

SYNC and NMT objects

The other two important communication objects used to facilitate the PDO communication in real-time control applications are SYNC (Synchronization) and NMT (Network Management) objects. SYNC objects are produced by a SYNC producer to trigger any event, e.g., sampling of an analog input or transmitting a TPDO on all SYNC consumer nodes in the network. In CANopen, the NMT communication objects create a master/slave configuration where an NMT master controls the operational states of the NMT slaves, as seen in figure 3 (Maxon motor ag. 2019). In the initialization state, the slaves announce their presence and Node-ID in the network. In the Pre-operational state, only SDO communication is possible alongside NMT messages. In the Operational state, SYNC and PDO communications are possible. In a Stopped state, both PDO and SDO communications are blocked, and only NMT communication can take place to change the state again.

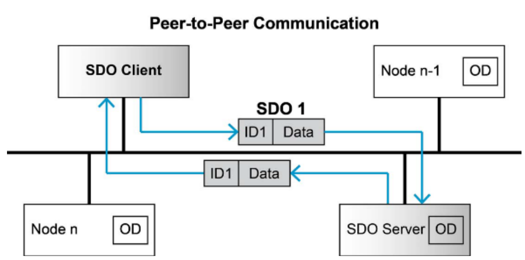

Figure 3. NMT slave states [Adapted from Maxon motor ag "EPOS4 Positioning Controllers Communication Guide", Edition 2019-11, Doc ID: rel8759]

In our application, the synchronous control of the EPOS4 devices in the network was achieved using TPDO messages to update the target position or torque and RPDO messages to read the actual position and torque values, triggered by SYNC messages from the Network Master.

2.2 EPOS4 positioning controller

EPOS4 positioning controllers from Maxon motor ag are compact servo positioning controllers capable of controlling position, velocity or torque of permanent-magnet brushless DC (BLDC) motors in servo operation. The communication interface of EPOS4 controllers conforms to the CiA 301 "CANopen application layer and communication profile" as well as CiA 402 "CANopen device profile for drives and motion control" (CiA Draft Standard 402, Maxon motor ag. 2018a), making it an ideal candidate for real-time control operations for motion application through CANopen.

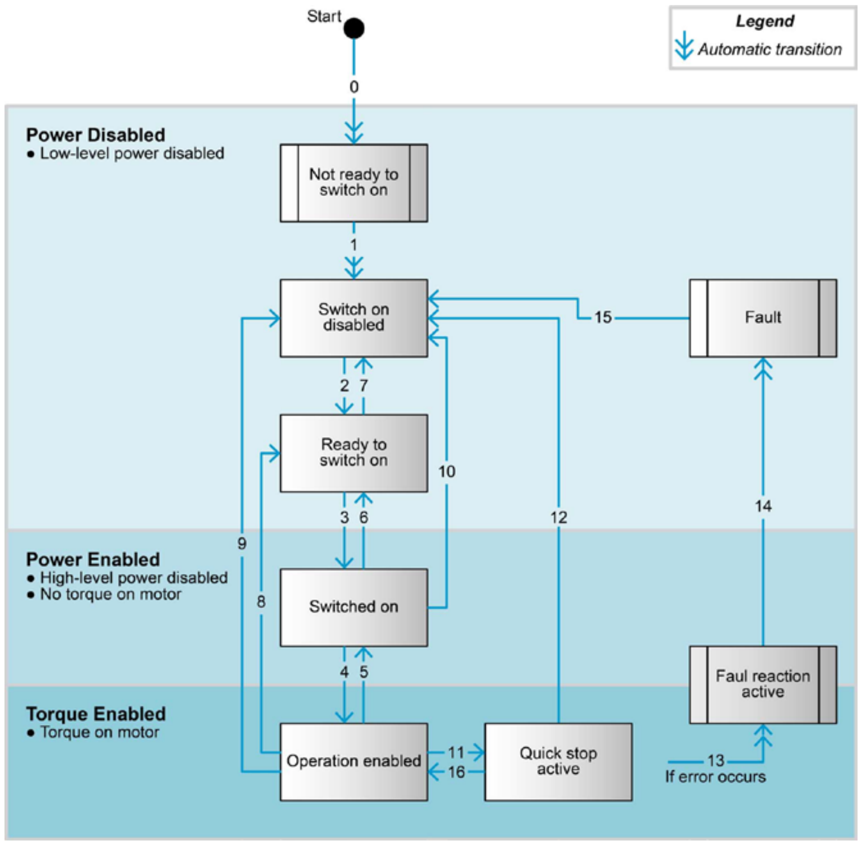

Figure 4. EPOS4 CiA402 Device State Machine [Adapted from Maxon motor ag "EPOS4 Positioning Controllers EPOS4 Firmware Specification", Edition 2018-11, Doc ID: rel8234]

CiA 402 device profile

CiA 402 device specifications are published as a device profile for frequency inverters and motor drives to be used in the CANopen network. This device profile includes operation-specific PDO mappings and an operational-state-machine for motor control operation seen in figure 4 (CiA Draft Standard 402, Maxon motor ag. 2018a).

Operation Modes

Being designed as a versatile position controller for automation and robotics application, the EPOS4 controllers offer multiple modes of operation, each controlling either position, velocity, or torque of a permanent magnet BLDC motor. It offers two profile modes (position and velocity), three cyclic synchronous modes (position, velocity, and torque) and one homing mode. For our robotics operation, Cyclic Synchronous Position (CSP) and Cyclic Synchronous Torque (CST) modes are of particular interest. In CSP mode, the EPOS4 input process variables are 'target position', 'position offset', and 'torque offset', and the output process variables are 'position actual value', 'velocity actual value', and 'torque actual value'. Figure 5 shows the CSP mode controller configuration with the object dictionary indices of each process variable (Maxon motor ag. 2018a).

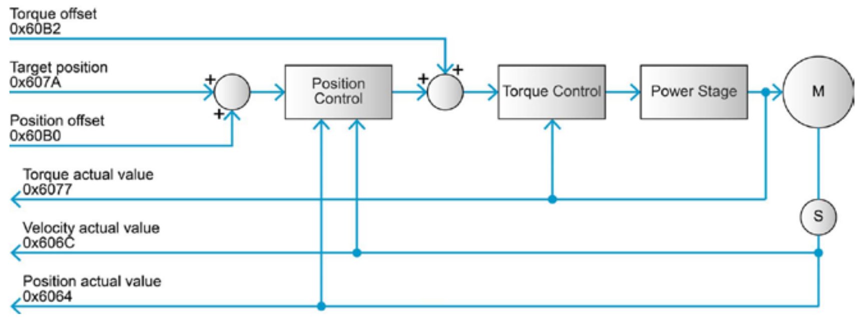

Figure 5. Cyclic Synchronous Position Mode – Overview [Adapted from Maxon motor ag "EPOS4 Positioning Controllers EPOS4 Firmware Specification", Edition 2018-11, Doc ID: rel8234]

In CST mode, only target torque and torque offset are used as input process variables, and actual position, velocity, and torque are again the output process variables, as seen in figure 6 with the OD indices.

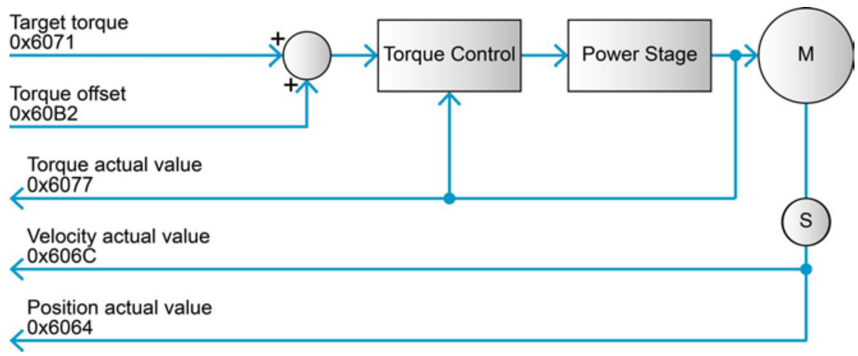

Figure 6. Cyclic Synchronous Torque Mode – Overview [Adapted from Maxon motor ag "EPOS4 Positioning Controllers EPOS4 Firmware Specification", Edition 2018-11, Doc ID: rel8234]

In EPOS4 controllers, all three Cyclic Synchronous modes have their input process variables internally refreshed in 1 millisecond cycle time (Maxon motor ag. 2018b), limiting the fastest effective bandwidth for a central closed-loop controller in the network to 1 kHz.

2.3 NI Industrial Communications for CANopen

National Instruments' (NI) LabVIEW software is a popular programming environment for instrumentation and control prototyping. NI Industrial Communications for CANopen (National Instruments 2020b) is an add-on library adding the functionality of using CANopen communication in LabVIEW programming environment. Paired with a NI real-time controller and a CANopen interface hardware, it is possible to build a CANopen network master capable of controlling multiple CANopen slave nodes in a real-time control network.

3. Methods

In our experimental setup shown in figure 7, a NI PXIe-8135 real-time controller equipped with a NI PXI-8531 CANopen interface module is used to operate multiple EPOS4 test setups in the same CANopen network. A CANopen master application, shown in figure 8, was developed capable of controlling the NMT state of the network, the CiA 402 state of each EPOS4 unit, and synchronously control all the EPOS4 units in the network in CST mode.

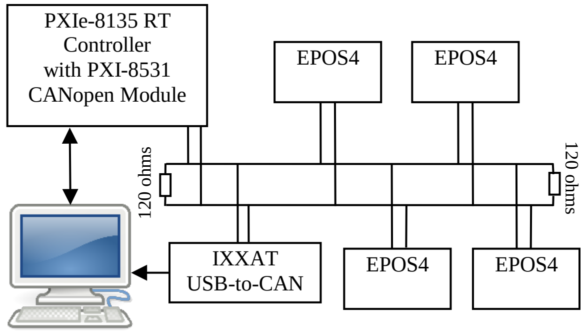

Figure 7. Experimental Setup

An IXXAT 'USB-to-CAN' module was also connected with the 'CAN Analyzer Mini' software to monitor CAN-bus usage. It was observed that with 1-3 motors active in the network, cycle times of 1ms was easily achieved. But with more than three motors active in the network, the highest available CAN-bus baud rate of 1 Mbit/s was being saturated, and the cycle time of the network had to be slowed down with each additional motor.

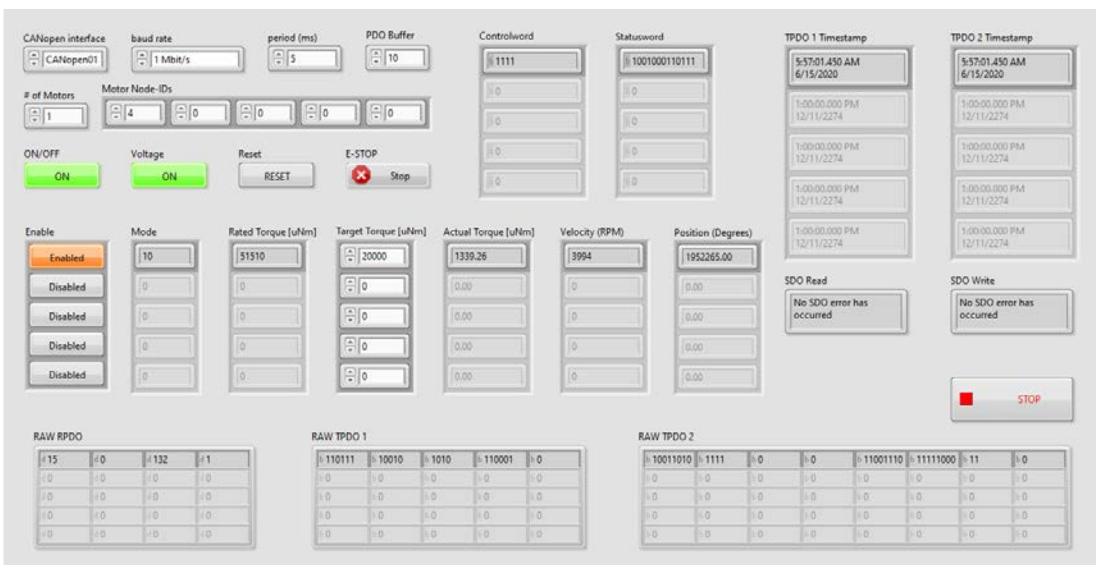

Figure 8. User interface for the LabVIEW program developed for this research

4. Discussion

4.1 Proposed Future Work

We are working towards replacing analog servo amplifiers used in two functional robots at the University of Wisconsin-Milwaukee BioRobotics Lab with EPOS4 controllers in a CANopen network. On completion of that, the robots will be operated with the EPOS4 controllers in CSP as well as in CST mode with a central position controller, and the trajectory tracking performance will be compared with current analog servo amplifier-based setup. With the comparison, any performance change introduced by data transfer delays and bandwidth limitations will be explored.

5. Conclusion

With this research, we have explored a way to control multiple EPOS4 controllers with the CANopen interface from LabVIEW programming environment. This confirms the viability of using these position controllers in our robot. However, the bandwidth and cycle time limitations need to be further explored to verify the position tracking the performance of this configuration compared to the existing analog servo amplifier-based setup.

References

Pfeiffer, O., Ayre, A. & Keydel, C. *Embedded Networking with CAN and CANopen*, Copperhill Technologies Corporation, 2008

"CANopen application layer and communication profile", *CiA Draft Standard 301*, Version 4.02

"The Basics of CANopen". National Instruments. https://www.ni.com/en-us/innovations/white-papers/13/the-basics-of-canopen.html (July 23, 2020)

Maxon motor ag *EPOS4 Positioning Controllers Communication Guide*, Edition 2019-11, Doc ID: rel8759

"CANopen device profile for drives and motion control", *CiA Draft Standard 402*, Version 4.00

Maxon motor ag "EPOS4 Positioning Controllers EPOS4 Firmware Specification", Edition 2018-11, Doc ID: rel8234

Maxon motor ag *EPOS4 Positioning Controllers EPOS4 Feature Chart*, Edition 2018-11, Doc ID: rel8001

"NI-Industrial Communications for CANopen 20.0 Readme". National Instruments. http://www.ni.com/pdf/manuals/375165h.html (July 23, 2020)

Biographies

Asif Al Zubayer Swapnil is a graduate student in the Mechanical Engineering Department of University of Wisconsin-Milwaukee. Currently he is working for his MS the BioRobotics lab under supervision of Dr. Rahman. He received his BSc. Engineering (mechanical) degree from Khulna University of Engineering & Technology, Bangladesh in 2014. He worked as an industrial and mechatronic system designer in product design industries for 4 years. His research interests include mechatronic and robotic systems for rehabilitation and assistance persons with physical disabilities.

Mohammad Habib Rahman is with the Mechanical and Biomedical Engineering Department, University of Wisconsin-Milwaukee, WI, USA. As Director of the BioRobotics Lab at the University of Wisconsin-Milwaukee, he brings the resources and expertise of an interdisciplinary R&D team. For more than 15 years he has been researching bio-mechatronics/bio-robotics with emphasis on the design, development and control of wearable robots to rehabilitate and assist elderly and physically disabled individuals who have lost their upper-limb function or motion due to stroke, cardiovascular disease, trauma, sports injuries, occupational injuries, and spinal cord injuries. He received a BSc Engineering (mechanical) degree from Khulna University of Engineering & Technology, Bangladesh in 2001, a Master of Engineering (bio-robotics) degree from Saga University, Japan in 2005 and a PhD in Engineering (bio-robotics) from École de technologie supérieure (ETS), Université du Québec, Canada in 2012. He worked as a postdoctoral research fellow in the School of Physical & Occupational Therapy, McGill University (2012-2014). His research interests are in bio-robotics, exoskeleton robot, intelligent system and control, mobile robotics, nonlinear control, control using biological signal such as electromyogram signals.