

API-WRAPPER

Wrapper for CANopen and J1939 stacks

The C#-API-Wrapper by Emtas (Germany) is made for the company's CANopen Master/Slave and J1939 stacks. The wrapper allows the use of the stacks in .NET applications in Windows or Linux.

The C# API allows the usage of the CANopen Master/Slave Stack on Windows using the .NET framework. All CANopen services are available. C API functions are wrapped into C# methods and callback functions are mapped to delegates. This solution supports all CANopen services that are supported by the stack itself. Besides the usual PDO, SDO, Heartbeat, Emergency, and SYNC support, the stack also provides a function to handle LSS, MPDOs, SDO routing, and SDO requesting devices.

The same solution exists for the J1939 stack, also provided by Emtas. This J1939 stack supports cyclic and event-driven transmission and reception of J1939 messages and also transfer of larger data using TP or BAM. For both stacks the design of the object dictionary, respectively the set of PGNs, can be done by the graphical tool CAN Device Designer from the same company.

Emtas does not manufacturer its own CAN interfaces but a couple of different CAN interfaces from different manufacturers such as Janz Tec, Peak, Kvaser, or Sys Tec are supported. Potential use cases are the development of customer specific test or service applications. Although the number of use cases might be low, this C# solution is also available for Linux using Mono.

```
// an SDO write indication is called whenever an object in the local object
// dictionary is written. It may reject the new value or not
static CANopen.RET_T sdoWriteInd(bool execute, byte sdoNr, ushort index, byte sub)
{
    // if execute is true, the SDO write access has been finished
    if (execute == true)
    {
        Console.WriteLine("Index 0x{0:X} - {1} written", index, sub);
    }
    return CANopen.RET_T.RET_OK;
}

// main method of application
static void Main(string[] args)
{
    // init CANopen stack it self
    ret = CANopen.coCanOpenStackInit(125);
    if (ret != CANopen.RET_T.RET_OK)
    {
        Console.WriteLine("Error coCanOpenStackInit: {0}", ret);
    }

    // register delegate for SDO write access
    ret = CANopen.coEventRegister_SDO_SERVER_WRITE(sdoWriteInd);
    if (ret != CANopen.RET_T.RET_OK)
    {
        Console.WriteLine("coEventRegister_SDO_SERVER_WRITE failed");
    }

    // register delegate for PDO reception
    ret = CANopen.coEventRegister_PDO(valueHandler.pdoInd);
    if (ret != CANopen.RET_T.RET_OK)
    {
        Console.WriteLine("coEventRegister_PDO failed");
    }

    // enable CAN driver
}
```

The C# wrapper is available for Emtas' CANopen Master/Slave Stack (Photo: Emtas)

[CW](#)