**E**lectric **M**otor **T**est**B**ench

Starting point:

GitHub code from Robert Hahn (IFX colleague)

* Python and C++ Application using a C-coded DLL
* Runs on 32-Bit machines only
* Communication Error after turning on IndraDrive must be deleted via IndraWorks GUI (Bosch Rexroth software)
  + Open IndraWorks SW
  + Connect to IndraDrive
  + Delete error
  + Close IndraWorks SW
  + Run Python Application in PyCharm (Inicio 1.1.4.0 used)

Our actions:

Adaption of the Python Application so it can run on 64 Bit machine

* Rebuild the Application for 64 Bit (using Microsoft Visual Studio 2017)
* Change of Python code to convert DLL-32-bit specific data types to 64-bit data types (in the \_\_init\_\_ function)
* Communication Error is cleared via the MotorTestBench in the beginning (no IndraWorks SW/license needed anymore)

HowTo use the Testbench:

1. Install Inicio (newest version)
   1. If drivers are already installed, choose profile "Lab Automation (no drivers)"
   2. If not, choose profile "Lab Automation (with drivers)"
2. When the driver subinstaller is executed, only choose the runtimes to be installed (NI VISA Runtime, IVI Compliance Package)
3. Install PCAN driver (google for download of latest pcan.dll)
4. Install MCC DAC driver
5. Checkout MotorTestBench from SVN
6. Configure */src/config/setting.py* and adapt the values accordingly:
   1. Check the COM port in which the IndraDrive-USB-plug is inserted (Device Manager > Ports (COM&LPT))
      1. Adapt the port number: IndraDrive\_COMPORT = "COM8"
   2. Check the VISA address of the Yokogawa Poweranalyzer (in NI VISA GUI) and adapt if necessary:
      1. YKWT3000\_VISA\_ADDRESS = "USB0::0x0B21::0x0025::39314D363136363434::INSTR"

**OFFLINE MODE**

The offline mode allows to start the MotorTestBench without having a connection to the instruments.

> To enable the OFFLINE MODE set the variable OFFLINE\_MODE in */src/config/setting.py* to True.

> Features:

* Dummy CAN messages are encoded
* Motor Control Window can be used normally and played around with
* WT1800 configuration/measurement setups can be loaded/saved but not applied/queried of course

> Following can be adapted in the PreLoop-method:

* WT1800 settings:
  + Torque/Speed measurement setup
  + ETA measurement setup (e.g. self.wt.interface.vi\_write(':MEAS:EFF:ETA1 PA,P4'))
  + User defined functions (e.g. self.wt.Measurements.ConfigureUserDefinedFunction(1, "UPP", "V", "UPPK(E2)\*1000"))
  + CAN settings: Message ID filter
* Configure the peripheral instruments in the lab & start the MotorTestBench: double-click on StartTestBench.bat
  + ! Only turn the red switch on the IndraDrive !
  + ! Green button must be off and “Reglerfreigabe” switched to “Aus” !
* When message pops up in the command line, press on the IndraDrive’s green button and switch “Reglerfreigabe” to “Ein”