TwinCAT EPICS Commands

The available commands are:

- **tcSetAlias:**
  Sets an alias name for a TwinCAT PLC. This name is used to define the info records. The alias name is applied when tcLoadRecords is called. It is reset afterwards.
  Example: tcSetAlias("C1PLC1")
  Sets the alias name to "C1PLC1".
- **tcSetScanRate:**
  Sets the scan rate for the TwinCAT PLC. The first argument is the scan rate in ms for the read or write scanners reading and writing TwinCAT variables. The second number is a multiple which describes the slow down for updating the EPICS read-only channels. The update rate for read/write channels is the same as the TwinCAT scan rate.
  Example: tcSetScanRate(10,5)
  Yields a 10ms TwinCAT update rate, and a 50ms EPICS update rate for read-only channels.
- **tcGenerateList:**
  Generates an additional listings when the records are loaded. Multiple tcList commands can be called in series to produce different listing. The first argument is a output file name. The second argument is a set of options. The lists are generated when tcLoadRecords is called. The list commands are reset afterwards.
  Example 1: tcGenerateList("C:\SlowControls\Target\H1ECATX1\PLC1\PLC1.req","-lb")
  This will generate an autoburt request file.
  Example 2: tcGenerateList("C:\SlowControls\Target\H1ECATX1\PLC1\PLC1.opc.txt","-l -rn -yi -cp")
  This will generate a listing of OPC names.
  Example 3: tcGenerateList("C:\SlowControls\Target\H1ECATX1\PLC1\PLC1.chn.txt","-l")
  This will generate a listing of EPICS names.
  Example 4: tcGenerateList("C:\SlowControls\Target\H1ECATX1\PLC1\PLC1.ini","-l -ns")
  This will generate an EPICS listing without string channels. The available options are listed on page TwinCAT EPICS Options.
- **tcGenerateMacros:**
  Generates ASCII macro lists (aml files) which can be used to generate ADL files for medm. The first argument is a output directory which is used to store the macro files. The second argument is a set of options. The macro files are generated when tcLoadRecords is called. The list commands are reset afterwards.
  Example 1: tcGenerateMacros("C:\SlowControls\Target\H1ECATX1\ADL")
  This will generate a macro files for each encountered structure including both fields and error messages. The resulting files are stored in the ADL subdirectory. Error messages require corresponding exp files (see the coding standard, E1200225.

Example 2: tcGenerateMacros("C:\SlowControls\Target\H1ECATX1\ADL", "-mf")
Generates macro files without error messages.

- **tcLoadRecords:**
Loads a tpy file, then generates and loads the EPICS database. The first argument is the filename to the tpy file. The generated db file will have the same name but with the extension ".db". The second argument is a set of options.
Example: tcLoadRecords("C:\SlowControls\Target\H1ECATX1\PLC1\PLC1.tpy","")
This command will parse the specified tpy file, then generate a db file with the name "C:\SlowControls\Target\H1ECATX1\PLC1\PLC1.db" and the specified options. The available options are listed on page TwinCAT EPICS Options.

The above commands will only be executed before iocInit() is called. Multiple tpy files can be loaded by issuing multiple tcLoadRecords commands. However, tcSetAlias and tcGenerateList need to be specified anew before each tcLoadRecords command. The rate specified with tcSetScanRate will be reused unless a new tcSetScanRate command has been issued.

aLIGO: TwinCAT EPICS Commands (last edited 2015-03-23 20:27:37 by DanielSigg)

TwinCAT EPICS Options

When generating a db file or a listing, a set of options describing the conversion rules is available. Options can be specified either Windows or Unix style. Meaning, both /ea and -ea will produce the same result.

Option	Description
Channel Processing:	
/eo	Only export variables which are marked by an OPC export directive in the tpy file (default)
/ea	Export all variables regardless of the OPC settings in the tpy file
/ys	String variables are processes (default)
/ns	No string variables are processed
/pa	Process all types (default)
/ps	Process only simple types types, e.g., INT, BOOL, DWORD, etc.
/pc	Process only complex types, e.g., STRUCT, ARRAY
Channel Name Conversion:	
/rl	LIGO standard conversion rule (default)
/rv	LIGO rules for initial vacuum channel names (version 1.1)
/rd	Replace dots with underscores in channel names
/rn	Do not apply any special conversion rules
/cp	Preserve case in EPICS channel names
/cu	Force upper case in EPICS channel names (default)
/cl	Force lower case in EPICS channel names
/nd	Eliminate leading dot in channel name (default)
/yd	Leave leading dot in channel name
/yi	Leave array indices in channel names
/ni	Replace array brackets with a single leading underscore (default)
Split File Support:	
/nsio	Do not split database or listing by record type (default)

/ysio	Split database or listing into input only and input/ouput recrods
/sn 'num'	Split database or listing into files with no more than 'num' records
/sn 0	Does not split database or listing into multiple files (default)
Database Generation:	
/devopc	Use OPC name in INPUT/OUTPUT field (default)
/devtc	Use TwinCAT name in INPUT/OUTPUT fields instead of OPC
List Generation:	
/l	Generate a standard listing, name only (default)
/ll	Generate a long listing, name and opc parameters
/lb	Generate an autoburt save/restore file
Macro Generation:	
/ma	Generate a macro file for each structure describing fields and errors (default)
/me	Generate a macro file for each structure describing the error messages
/mf	Generate a macro file for each structure describing all fields

Applicable options are:

Program/Instruction	**Available Options**	**Enforced Options**
tpyinfo	channel processing	
EpicsDbGen	all	
tcLoadRecords	channel processing, channel name conversion	-ps -nsio -sn 0 -devtc
tcGenerateList	channel processing, channel name conversion, list generation	-ps -nsio -sn 0
tcGenerateMacros	macro generation	
infoLoadRecords	channel processing, channel name conversion	-ps -nsio -sn 0 -devtc
infoGenerateList	channel processing, channel name conversion, list generation	-ps -nsio -sn 0

aLIGO: TwinCAT EPICS Options (last edited 2015-03-23 22:12:22 by DanielSigg)

TwinCAT EPICS IOC Source Build

Source tree

Make sure you have subversion installed. Check out the source tree under C:\SlowControls, see SlowControlsSubversion.

Versions

Version	svn revision	Date	Comment
1.0	1406	2013, Sep 10	Initial release
1.1	1974	2014, Nov 10	Alias, vacuum system support

After svn revision 1406 a lot of features have been added which have not undergone sufficient testing. A branch based on version 1 of the code was created in the archive at "branches/maggie". This branch is used for all subsequent version 1.n releases.

Visual Studio

- Install Microsoft Visual Studio 2012. Get it from here.

Building EPICS

Download

- Perl: Strawberry perl, install
- Make: gumake, install
- EPICS: Download EPICS base R3.14.12.3, unzip into C:\SlowCotrols\EPICS\base-3.14.12.3

EPICS build

- Debug build: Edit your base-3.14.12.3/configure/CONFIG_SITE file and set CROSS_COMPILER_TARGET_ARCHS = win32-x86-debug
- Run C:\SlowControls\Scripts\Common\build_epics_base.ps1

Building tcIoc

- Open C:\SlowCotrols\EPICS\Utilities\tcIoc\tcIoc.sln in MSVS.
- Build debug or release version
- Optional: Run C:\SlowControls\Scripts\Common\install_tcioc.ps1 to install new release

binaries into C:\SlowControls\EPICS\Utilities\Bin

Documentation

- Doxygen: Online documentation is created with doxygen. Download from here, install. Then, run with C:\SlowControls\EPICS\Utilities\tcIoc\Doxyfile.

Miscellenaous

- Build expat: The expat libraries are provided in C:\SlowControls\EPICS\Utilities\expat. In case a new built is required, download from http://expat.sourceforge.net/. Build static, multithreaded release and debug versions and save them as libexpatMT.lib and libexpatMTD.lib in the above directory.
- Generate tc device support: The tc device support files are provided and only need to be regenerated when new records are added. Run createTcDeviceSupport.ps1 in C:\SlowControls\EPICS\Utilities\tcIoc\TCatDeviceSupport.
- Generate info device support: The info device support files are provided and only need to be regenerated when new records are added. Run createInfoDeviceSupport.ps1 in C:\SlowControls\EPICS\Utilities\tcIoc\InfoDeviceSupport.

aLIGO: TwinCAT EPICS IOC Source Build (last edited 2014-11-10 19:54:22 by DanielSigg)