Pressemitteilung

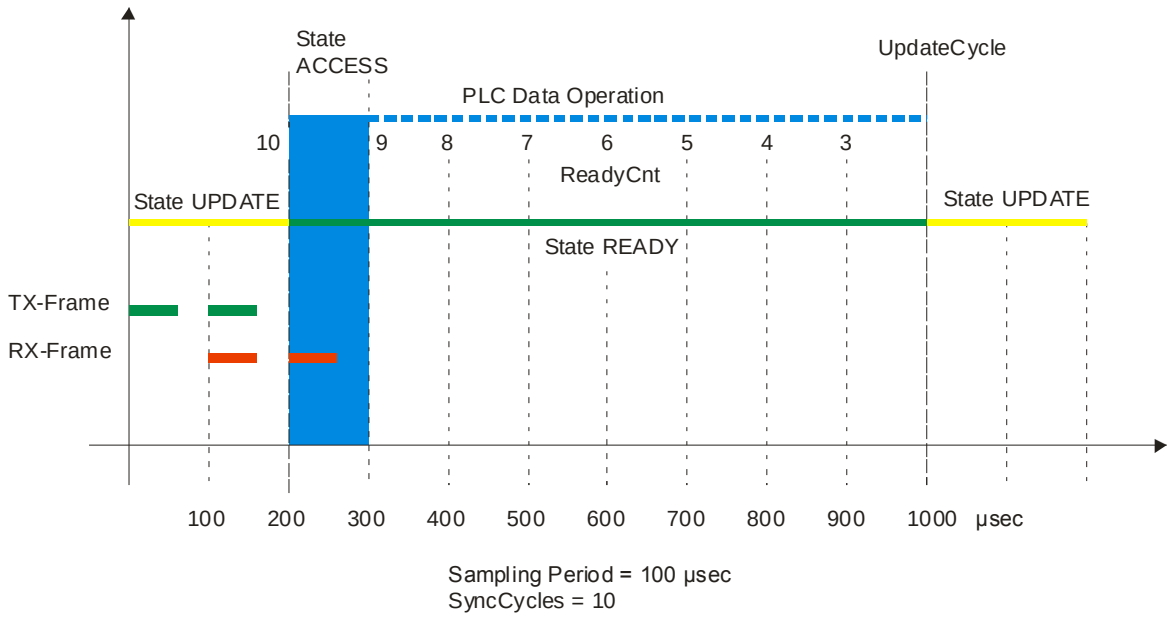
8.1.2018

Das Geheimnis des EtherCAT Zyklus

Der effiziente Umgang mit dem Sybera EtherCAT Master setzt die Kenntnis über die Arbeitsweise des EtherCAT Zyklus voraus. Im EtherCAT Zyklus sind unterschiedliche Mechanismen beteiligt, die Auswirkungen auf den EtherCAT Zyklus verursachen können. Dieser Artikel soll ein besseres Verständnis im Umgang mit dem Sybera EtherCAT Master vermitteln

Die Wrapper States

Die Echtzeit Task des EtherCAT Masters Programms unterscheidet sich grundsätzlich nicht von einer normalen Echtzeittask ohne Master Funktionalität. Die Echtzeittask wird zyklisch aufgerufen und durchlaufen mit dem Zeitraster der vorgegeben Periode. Eine Echtzeittask stellt erst durch die Wrapper-Funktionen ECATENTER und ECATEXIT die Verbindung zum EtherCAT Master her und tauscht mit diesem die Stationsdaten aus.
Der EtherCAT Master fasst nun mehrere Perioden (Zeitslots oder SyncSycles) zu einem Update-Zyklus zusammen und weist bestimmten Perioden Aufgaben zu. Dadurch ergibt sich ein Sampling-Betrieb des Feldbussystems. Grundsätzlich arbeitet der EtherCAT Master mit 4 Tasks RX, TX, APP (funktionale Verarbeitung) und Error), wobei RX, TX und Error-Handling implizit verwaltet werden. Um eine zeitliche Entkopplung zu ermöglichen, werden die Tasks auf verschiedene Zeitslots verteilt. Der EtherCAT Update-Zyklus wird also in funktionale Zeitslots unterteilt, um somit jeder Task (RX, TX, APP und Error) innerhalb des Update-Zyklus einen deterministischen Ausführungszeitpunkt zu ermöglichen.
Die Wrapper-Funktion ECATENTER gibt nun einen Statuswert zurück, um den Zeitpunkt der impliziten Verarbeitung mitzuteilen (ECAT_STATE_UPDATE). Erst wenn die Wrapper-Funktion ECAT_STATE_READY zurückgibt, können die Daten funktional verarbeitet werden.



Sampling Period = 100 µsec
SyncCycles = 10

SYBERA GmbH, Hohenzollernstr. 2, 71088 Holzgerlingen, Tel: 07031-744608 (Fax: -744609), Email: info@sybera.de

ECAT_STATE_UPDATE: hier wird der RX-Task, TX-Task und Error-Task Anteil behandelt, d.h. die App-Task (logische Verarbeitung der Daten) wird relativ schnell verlassen und trägt daher nicht relevant zur Prozessor-Last bei.

ECAT_STATE_ACCESS: Die Übertragung der Ethernet Frames ist vollzogen, und die logische Verarbeitung der Nutzdaten kann erfolgen. Mit nur einem SyncCycle muss jetzt auch die logische die logische Verarbeitung der Daten erfolgen.

ECAT_STATE_READY: Die Übertragung ist vollzogen, und die logische Verarbeitung der Nutzdaten kann erfolgen. Mit einer State-Machine kann die logische Verarbeitung der Daten auf die verbleibenden Zeitslots verteilt werden. Der RX-Task, TX-Task und Error-Task Anteil trägt nun nicht mehr relevant zum Prozessor-Last bei.

In den meisten Fällen wird nur ein Zeitslot für RX, TX und Error Handling benötigt, da die Framegröße relativ klein ist und typischerweise TX und RX in einem Sample abgearbeitet werden können. Um den Systembus zu entlasten, wird jedoch ein zusätzlicher Idle–Slot (ReadyCnt=0) als zeitliche Entkopplung empfohlen.
Im Beispiel (2 Update Slots und 1 Idle Slot) stehen somit 7 Zeitslots für die funktionale Verarbeitung der Daten zur Verfügung. Im Programm können die Daten ab ReadyCnt=1 bis ReadyCnt=n-1 verarbeitet werden (ReadyCnt=0 dient zur Busentlastung).

Prozessor-Last

Die Prozessorlast umfasst alle Aufgaben innerhalb einer Sampling-Periode. Diese Last kann je nach Aufgabe sehr unterschiedlich ausfallen. In der Regel ist die Prozessorlast bei der funktionalen Verarbeitung der Daten (AppTask) am höchsten. Mit dem Programm SYDBG kann die maximale Prozessorlast (MaxProcLoad) ermittelt werden und dient zur optimalen Einstellung der Sampling-Periode.

Es gilt:

Period (optimal) = MaxProcLoad + 20 (Periodenmodulation)

MaxProcLoad = SystemMgmt + RxTask + TxTask + AppTask + ErrTask + MaxJitter

Leider erzeugen unterschiedliche PC-Plattformen auch unterschiedliche Jitter Anteile (hauptsächlich C-States, TurboMode und SpeedStep, aber auch System-Interrupts, DMA und Cache Mechanismen). Bei manchen PC-Plattformen beträgt dieser Jitter > 40µsec.

Somit ergibt sich grob für die AppTask bei 200 µsec eine max. zulässige Ablaufdauer von:

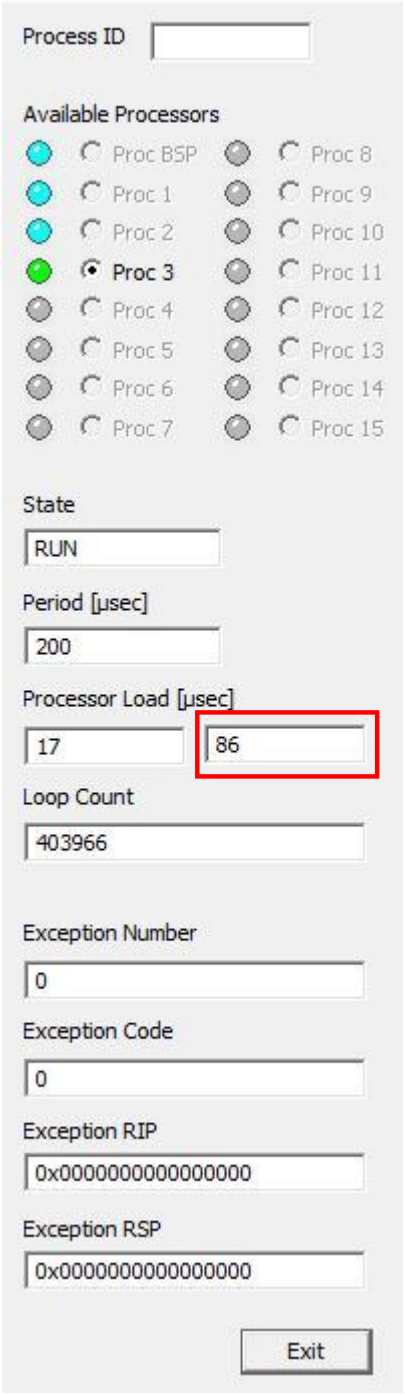
AppTask = 200 (Period) - 20 (Periodenmodulation) - 10 (SystemMgmt)
- 10 (RxTask + TxTask + ErrTask)
- 40 (MaxJitter)
= 120 µsec

Da der JitterAnteil nicht zu vernachlässigen ist, sollte damit auch nicht zu knapp gerechnet werden. Dieser kann über den JitterTest von SYDBG64 ermittelt werden.

Pressemitteilung

8.1.2018

SYDBG64



MaxProcLoad > Period ?

Was passiert nun, wenn MaxProcLoad > Period ist ?
In diesem Fall verlängert die Realtime-Engine die Periode auf MaxProcLoad, wobei dadurch das Echtzeit-Kriterium verletzt wird. Durch die Pulsweitenmodulation wird der Fehler nach ein paar Zyklen wieder ausgeglichen, so dass DC durchaus mit dieser Toleranz arbeiten kann. Diese sporadische Abweichung sollte jedoch nicht größer als 10% sein und keine statische Überlastung der Sampling-Periode darstellen.
Die Verarbeitungszeit der APP-Task kann einfach durch __rdtsc() und __CpuFreq ermittelt werden (siehe Beispiel Jitter-Comp64 aus SHA)

MaxProcLoad

Pressemitteilung

8.1.2018

Periodenmodulation

Mit der neuen EtherCAT Master Version wurde für DistributedClock eine Perioden-Modulation eingeführt, um eine Nachführung der MasterClock zu ermöglichen. Die Periodenmodulation wird für den Drift-Ausgleich zwischen Master-Clock und DC ReferenceClock (typ. erster Slave) benötigt, da sonst kein DC (Distributed Clock) möglich wäre. Hierbei wird die Periode der Master-Clock dynamisch angepasst. Um die neue Funktionalität nutzen zu können, sollte die Umsetzung der gerätespezifischen ESI-Datei nochmals erfolgen, da das Format der ECATDEVICE.PAR mit der Sektion [OPMODE] erweitert wurde.

z.B.

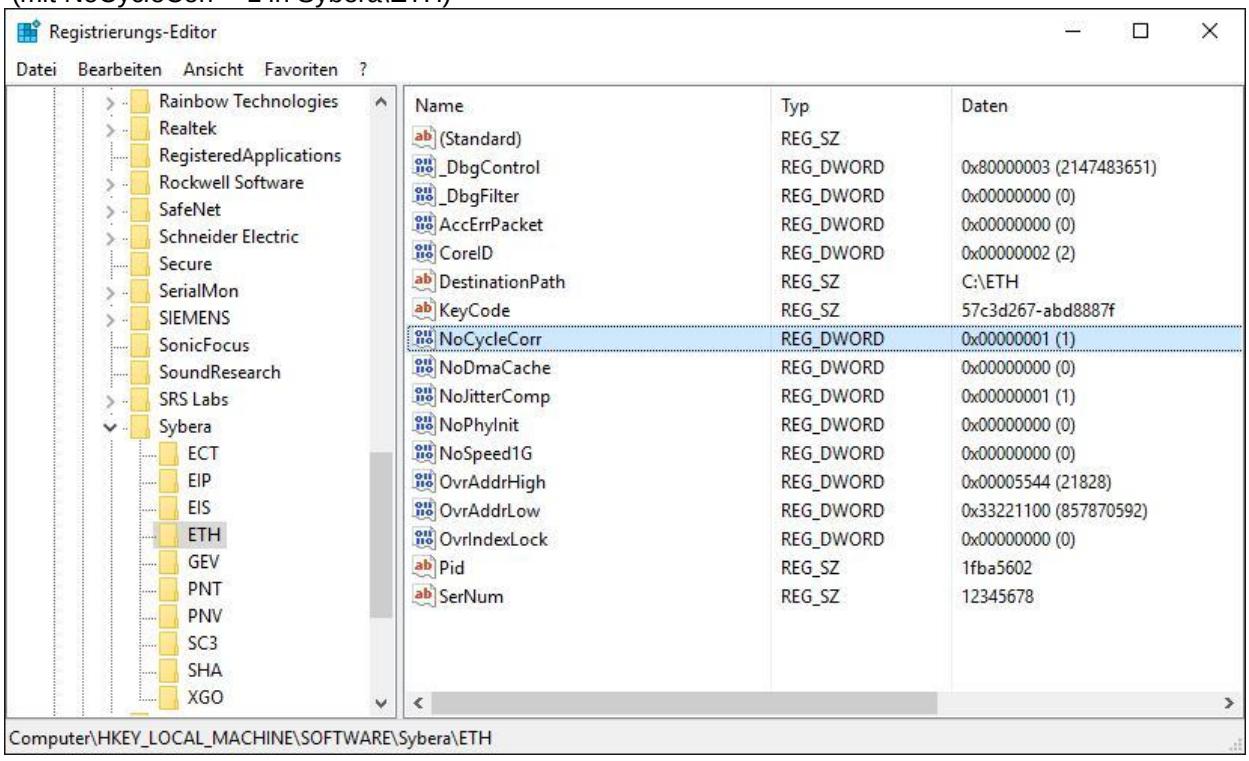
[OPMODE]
00 03 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01 00 00 00

Ebenfalls sollte der SourceCode um einen Befehl erweitert werden:

Ecat64DcControl(); (Siehe Beispiel EL3702TST)

Die Perioden-Modulation ist standardmäßig eingeschaltet und kann in der Registry angepasst werden

Deaktivierung der Perioden-Modulation
(mit NoCycleCorr = 1 in Sybera\ETH)



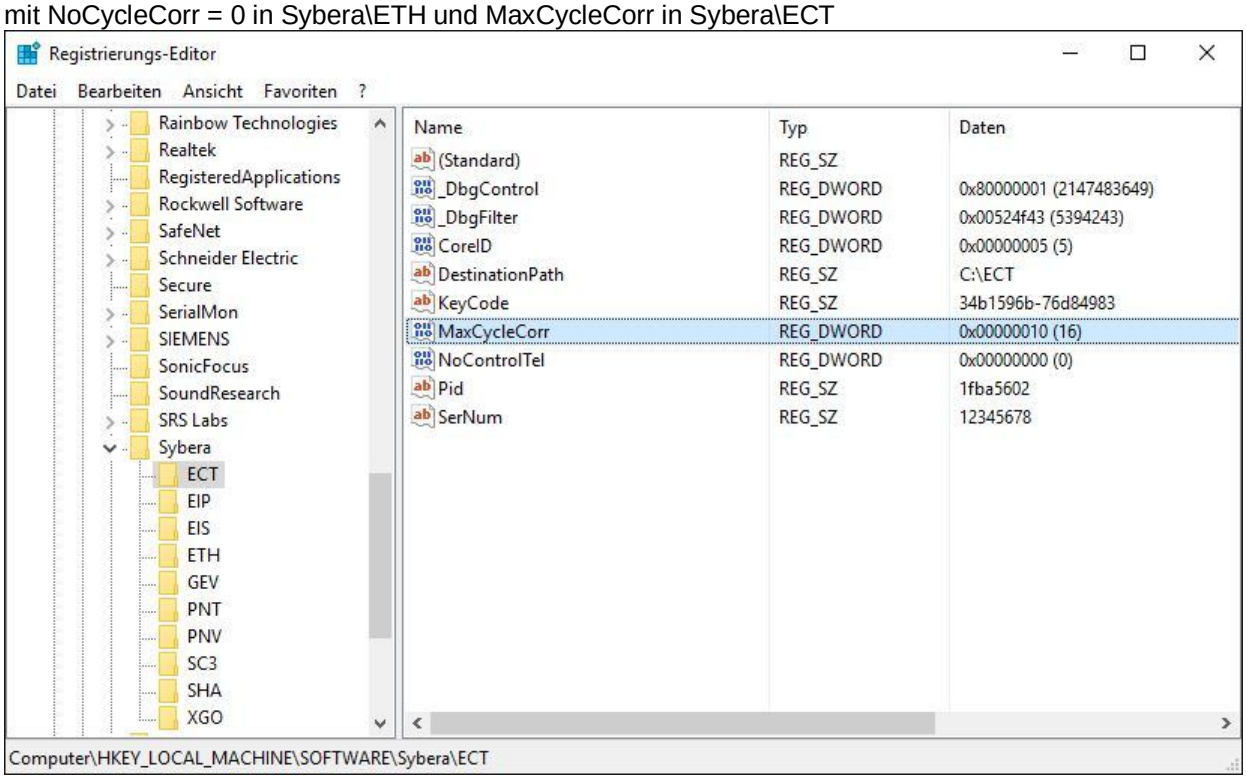
SYBERA GmbH, Hohenzollernstr. 2, 71088 Holzgerlingen, Tel: 07031-744608 (Fax: -744609), Email: info@sybera.de

Pressemitteilung

8.1.2018

Anpassung der Perioden-Modulation

mit NoCycleCorr = 0 in Sybera\ETH und MaxCycleCorr in Sybera\ECT



Bei aktiver Perioden-Modulation kann keine Jitter-Messung mehr erfolgen, da die Periode nun dynamisch nachgeführt wird. Mit der Perioden-Modulation ist es nun auch mit Schrittmotoren möglich, ein stabiles DC-Verhalten zu gewährleisten, selbst bei stark Jitter-behafteten PC's

SYBERA GmbH, Hohenzollernstr. 2, 71088 Holzgerlingen, Tel: 07031-744608 (Fax: -744609), Email: info@sybera.de