

江端さんのDIY奮闘記 EtherCATでホームセキュリティシステムを作る(9):

制御の世界の“黒船”、TwinCATでメイドを動かす

<http://eetimes.jp/ee/articles/1603/31/news037.html>

SOEMに限界を感じていた私は、ベッコフオートメーションのソフトウェアPLC「TwinCAT3」に手を出しました。そう、文字通り“手を出してしまった”のです。今回は、制御の世界に「黒船」を持ち込んできたTwinCAT3を使ってメイドを動かすまでの、私の苦闘と孤闘の全容をご覧くださいと思います。

2016年03月31日 10時30分 更新

[江端智一, EE Times Japan]



FA(ファクトリオートメーション)を支える「EtherCAT」。この超高度なネットワークを、無謀にも個人の“ホームセキュリティシステム”向けに応用するプロジェクトに挑みます……!! ⇒「[江端さんのDIY奮闘記 EtherCATでホームセキュリティシステムを作る](#)」連載一覧

週末エンジニアである私にとって、年末年始、ゴールデンウィーク、お盆に取れる1週間程度の休みは、普段できない比較的厄介なことを「やっつける」ための、めったにないチャンスです。

一人暮らしをしている父の実家に、PC、EtherCATのスレーブ(私のメイドたち”Rio”, ”Yuko”, ”Mao”)を持ち込んで、実家のキッチンテーブルの上に、EtherCATシステムを作って、ツールをインストールしたり、プログラムを書いたりしています。

実家のキッチンテーブルでは、LEDがピカピカと点滅し続けるEtherCATのスレーブ(メイド)と、すごい勢いでログが表示されるPCの間に、茶碗や皿が配置されて、2人で黙々と食事をする——というのが日常となっています(ごはん食べつつ、ディスプレイの表示を見ながら、Ctrl+c[強制終了ショートカットキー]を押下することもある)。

ま、それはさておき。

2015年の正月に、実家で父の食事を作りながら、ある会社のライブラリを使ってEtherCATの実験をやっていたのですが、思い通りに動かすことができませんでした(この辺の事情は[こちら](#)をご覧ください)。

いろいろと理由はあるのですが、はっきりいって「マニュアルの説明が不親切だった」に尽きると思います。特に、エンジニア向けのマニュアルは、ほぼ例外なく不親切です。

しかし、掃除機、洗濯機、冷蔵庫などの家電製品に対して、マニュアルを一度も読まないままで、「動かん」と怒り出す人は、世の中にはたくさんいます。そのようなユーザーに対して、メ

一カーが「マニュアルを読まないアンタが悪い」という態度を取ったら、そのメーカーの信用は地に落ちるでしょう。

そもそも人間は「機械」が大嫌いで、「マニュアルを読む」という行為は、もっと嫌いなのです。

そんな中にあっても、エンジニア向けの専用装置やソフトウェアのマニュアルは、恐ろしく分かりにくく書かれています。「分かってもらおう」という姿勢が絶無に近く、悪意を感じさせるほどです。

なぜ、こうなるのか？

それは、製品の開発者自らが、マニュアルを作るからです。彼らは「[パレートの法則](#)」に忠実に従ったマニュアルを作ります。

一度も使うことがない20%の機能の説明が、マニュアルの全体の80%を占めるのです。

しかも、その開発者兼マニュアルライターは、最も簡単な使用方法「だけ」を説明することができません。全ての機能を漏らさず記載しようとするので、その結果、最も簡単な動作方法すら分からない、最低にして最悪のマニュアルが完成する、というわけです。

なぜ、そのようなことが起こるのか？

制御向けの専用装置のユーザー数が少ないからです。WindowsやOfficeのように、全世界に数千万人もいるユーザーがいるわけではなく、日本国内に100人もいないだろうユーザーに対して、素晴らしいマニュアルを作る手間暇をかけるくらいならば、クレーム対応した方が、トータルコストは、はるかに安くなるでしょう。

それに「エンジニアなんだから、自分でなんとかできるだろう？」という、上から目線の傲慢（ごうまん）な態度も垣間見られます。だから、エンジニア向けの専用装置やソフトウェアのマニュアルは、恐ろしく分かりにくくなるのです。

加えて製作元のFAQも当てになりません。開発者でもある彼らは、その製品を「初めて使う人間」の思考が理解できません（開発者なんですから、製品については100%理解している）。例えば、私はWindowsのOSですら、トラブルになった時、マイクロソフトの公式FAQサイトで助けられたという記憶は、一度もありません）。

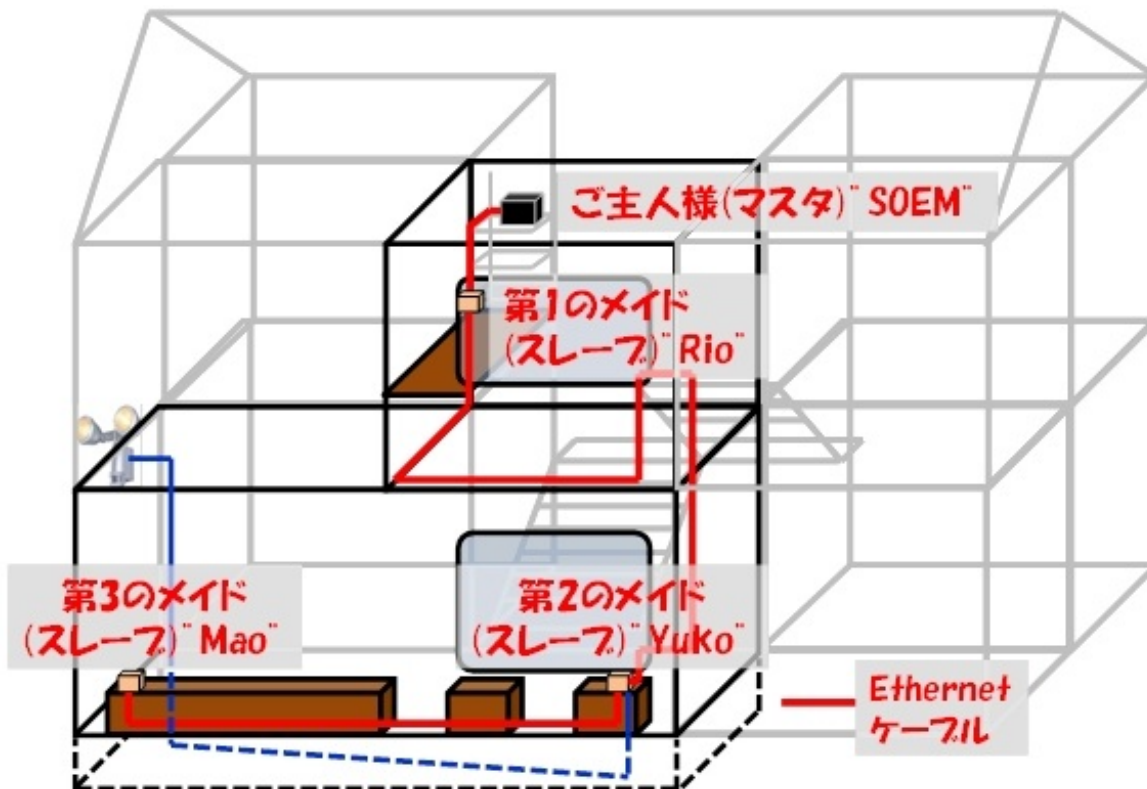
いつだって、頼りになるのは「同じ問題で困ったエンジニアが、この問題に対して、（その怒りやイライラな気持ちを込めて）Webに書き残しておいてくれた『メモ』だけ」です。

それはさておき。

「TwinCAT3」を使うチャンスが……！

[前回](#)は、SOEM(Simple Open EtherCAT Master)を使った、EtherCATベースの江端家ホ

ームセキュリティシステムについてお話をしました。



ところが、マスタであるSOEM(PCベースのご主人様)は、PCのメモリマップの配置などを自分で読み取って、その情報をSOEM用のプログラムにベタ書きする必要があります。これがかなり面倒なのです。

それに、これはEtherCATスレーブ(メイド)が追加される度に、そのプログラムの書き直しをしなければならず、こちらも厄介です。

「そんなことは、最初から分かっていたことだろうが!」と言われればそれまでなのですが、SOEMって、フリー(無料)のオープンソースなんですよ。EtherCATを勉強したいけど、金もコネもない週末エンジニアが、SOEMにたどり着くのは仕方がないことでしょう。

□

—— というようなお話を、先日(2016年1月)ベッコフオートメーションの川野社長にお会いした際に、愚痴っていたところ、川野社長から、「ぜひ、弊社の『TwinCAT3』を試してください」と強く勧められました。

江端:「いや、あのですね、週末エンジニアの私に、御社のPC用のマスタを購入する金なんて……」

川野さん:「大丈夫です。TwinCAT3は無料の試用期間があります」

江端:「その試用期間が終了したら、江端家のホームセキュリティシステムが止まってしまいます。そんなものは使えません」

川野さん:「江端さんがホームセキュリティシステムを連続で起動させるのは、ご実家に帰省される1週間程度ですよね。それなら、大丈夫です。帰省される度に、無料ライセンスを更新し続けられればいいのですよ」

——正直だなー、この人……というより、TwinCAT3の販売元の社長として、大丈夫か？

しかし、自社製品への絶対の自信と、その限りない慈愛を感じました。大切に育ててきた製品を、1人でも多くの人に試してもらいたいという熱い思いが、ビンビンと伝わってきました。

「どうやら、この人(ベッコフ日本法人社長、[川野俊充さん](#)、根っこはこっち側(エンジニア側)だ」ということが、私にも分かってきました。



川野俊充(かわの・としみつ) ベッコフオートメーション 代表取締役社長

東京大学理学部 物理学科 卒業、カリフォルニア大学バークレー校 ハース経営大学院経営学修士、慶應義塾大学SFC研究所 上席所員(訪問)。「EtherCAT」開発元のベッコフオートメーションにて、ソフトウェアPLC/NC/CNCのTwinCATによるPC制御ソリューションの普及に努めている。

なぜ、PCをEtherCATのマスタにするのか

こんにちは。江端智一です。

今回は、2016年1月にベッコフに訪問して、社長の川野俊充さんと、ソリューション・アプリケーション マネージャ 小幡正規さんにインタビューさせていただいた内容の一部(インタビューの全容は来月の最終回を予定)と絡ませながら、TwinCAT3のインストールから、制御プログラムの作り方まで、一通りご説明したいと思います。

インタビューの際に私が作成した資料は、[こっち*](#)です。資料の文言は丁寧ですが、つまるところ「EtherCATをデイスる内容」になっていることが分かると思います。

*) 参考: [公開用 ベッコフオートメーション株式会社 EtherCAT連載インタビュー\(1回目\)](#)

これ、この私が、EtherCAT以外の制御LANの団体の方を訪問してかき集めたEtherCATへの悪口の集大成でもあります*)。

*) なぜ、わざわざそんなことをしたかと言うと、この連載が面白い内容になると思ったからです(私は、ちょうちん記事は書きません)。

結果として、この「デイスる」「悪口」に対して、川野さんと小幡さんは、完全に私を論破(しかも数値や実例の証拠付きで)してしまうのですが —— それは、次回の最終回で。

□

江端:「SOEMもTwinCAT3もそうですが、なんで、PCをEtherCATのマスタにする、などという発想が出てきたのでしょうか？ 専用機のマスタを作って売っている方が、もうかるんじゃないのですか？」

川野さん:「そうですね。加えて、ユーザーは、PCのマスタや新しいスレーブに対して、強烈な拒否反応もあります。現状と同じ構成のマスタやスレーブをASICで作り直そうとしている会社もありますし、静電容量が変わるかもしれないから、今のスレーブを触るな！ 掃除もするな！（ホコリを取り除くと静電容量が変わるかもしれない）と言われたことすらあります」



「掃除はダメよ」

江端:「ああ、その話分かります。私も、お客さんから『10Mのイーサネットのイエローケーブル』と『20年前と完全同一のコントローラーを作って持ってこい』と、言われたことがあります」

川野さん:「それですね、わが社が、TwinCAT3のようなPCのマスタ(ご主人様)を、強く押し進めている理由は2つあります。1つ目は、『ラダーによる記述の限界』です。ラダーというのは、機械屋の発想からできたものですが、最近のPLCのプログラムの複雑化で、ラダー記述では、限界が見えてきたのですよ。



こんなやつですね

2つ目は、プログラムの再利用性ですね。ラダー表記に比べれば、オブジェクト指向のプログラムの拡張性や再利用性は文句なしに優れていますし、プロファイルを使うことで、プログラムの使い回しも、簡単に可能となりますしね」

江端:「それと、制御の世界のリアルタイム性能を、簡単に実現できるくらい、PCの性能が上がってきたから……ですね？」

川野さん:「それは、もちろん大きな前提の1つです」

全然「楽々インストール」じゃない!!

『そこまで川野社長が勧めるTwinCAT3なら、実際に試してみるか』と思い、実際にインストールしてみたのですが —— 口が割けたって、「楽々インストール」などということはありませんでした(繰り返しますが、私は、ちょうちん記事は書きません)。

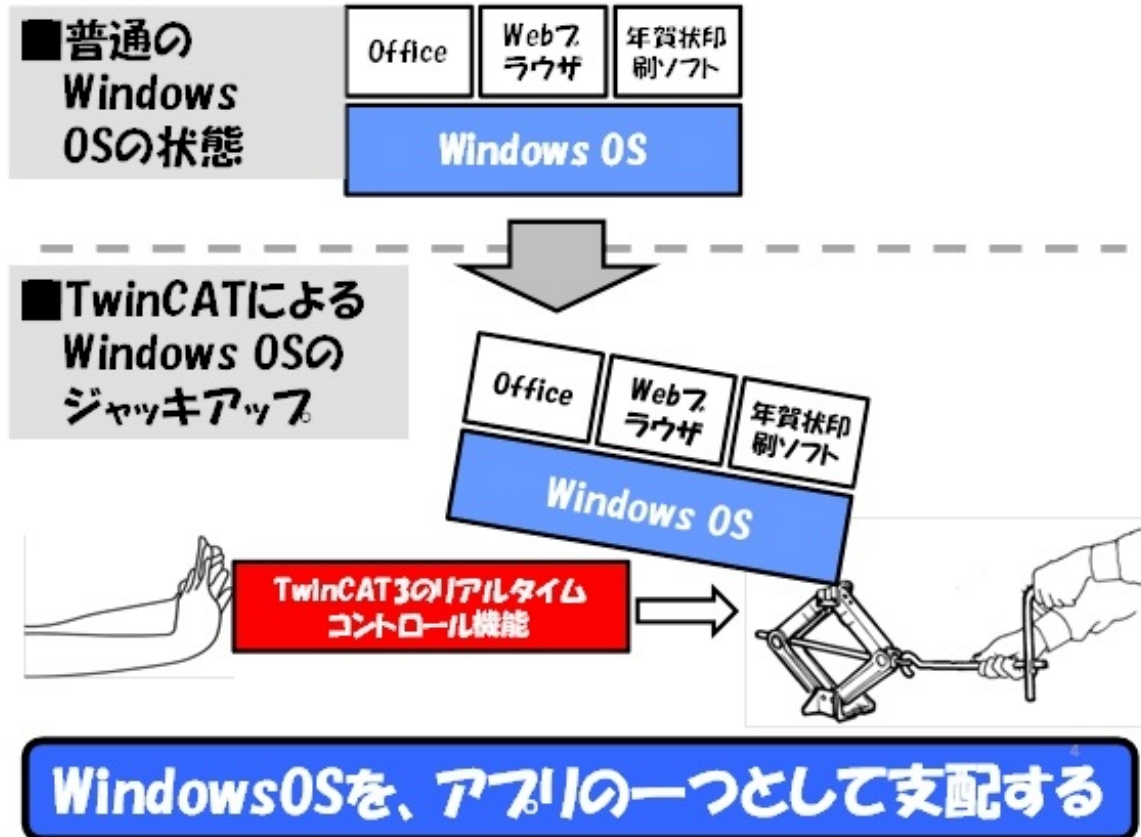
そもそも、TwinCAT3は、Windowsにインストールして使う「リアルタイム性能を保証する」EtherCATのマスタ(ご主人様)のソフトウェアですが、そもそもWindowsというOSが、リアルタイム性能を保証しないのです。

リアルタイムを保証しないOSの上に、リアルタイムのアプリが載るということは、論理的にありえないことです。

ですから、TwinCAT3は、すごいことをします。

例えるのであれば「WindowsOSをジャッキで持ち上げて、そこに寝っころがって、力づくでWindowsOSの下に潜り込む」のです。そして、WindowsOSを、TwinCAT3のアプリケーションのように取り扱ってしまうのです。

TwinCAT3が、そこらのアプリと違う点



もはや、これは“OSの下剋上(げこくじょう)”といっても良いでしょう。

で、こういうすごいことをするソフトウェアは、多くの場合、インストールに失敗すれば、回復不能の絶望的なトラブルを起こします。

私も一度TwinCAT2の時にインストールに失敗して、WindowsPC、1台を丸ごとパーにしたことがありますし、この連載の読者の方からも、同じような話を教えてもらったことがあります(私は、ちようちん記事は(以下省略))。

TwinCAT3のインストールは、「お正月の年賀状作成ソフト」のインストールとは、訳が違います。インストールに失敗すれば、あなたのシステムを全て破壊する可能性が極めて高いことを認識しておいてください(可能なら、[Acronis](#)のようなバックアップソフトで、OSイメージ丸ごと、外部のHDDなどに退避しておくことを強くお勧めします)。

□

散々脅かしましたが、――では、私の自宅セキュリティシステムを例として、TwinCAT3のインストール方法と、極めて簡単なプログラミングを説明させていただきます。

接続しているスレーブ(メイド)やマスタの配置は、前回のコラムの内容と同じものを使います([こちらを参照してください](#))。

私の場合は、ラダープログラム的一种であるST言語を勉強する気力はなかったので、最初からC/C++を使うことにしました。

目標は「アナログ入力のスレーブに接続されている光センサーに懐中電灯を向けると、デジタル出力のスレーブに接続されているLEDが点滅するようにする」としました。

では、次ページより、インストールの手順を紹介します。

【Step 1】PCにネットワークインタフェースカード (Network Interface Card: NIC) をもう1枚差せるPCIスロットがあるかを確認

TwinCAT3は、リアルタイム性能を保证するために、NICの種類に「うるさい」です。

私の場合、ノートPCのEthernetポートでは動いてくれませんでした。USB-Ethernet変換器なんぞ全くダメ。面倒を避けたければ、比較的性能の良い拡張PCIスロットのあるデスクトップPCを選ぶことをお勧めします。

【Step 2】EthernetのNICを1枚購入



原則Intelの純正NICなら大丈夫のようです。一応、[こちら](#)に、対応するIntel製ネットワークチップの一覧が掲載されています。

私のPCでは、インテル Gigabit CT Desktop Adapter EXPI9301CTが、動いています。

ま、ここはケチケチせずに、Intel純正のNICを購入しましょう。TwinCAT3をインストールした後で、NICが対応していないことが分かって手遅れです。

【Step 3】Windowsシステムのフルバックアップ

本当にしつこいですが、再起動しないPCの



前で青冷めるのって、本当に怖いですよ。完全に元に戻せるという安心感を持って、余裕のある作業をしましょう。



著者が持っているバックアップソフト

【Step 4】「Visual Studio (VS) 2010／2012／2013」のProfessional／Ultimate／Premiumのいずれかをインストール

これがないと、TwinCAT3でC++のプログラムが組めません。

そして、必ずTwinCAT3のインストールの前にインストールしてください。

ちなみに、私、これをTwinCAT3のインストール前に、インストールしなかったために、安定稼働中のTwinCAT3を、泣く泣くアンインストールしました(血の涙が出そうでした)。



私は深く考えずに、VS2012のPremiumをインストールしました。

Visual Studioは有償なのですが、試用期間(30日)中であれば、TwinCATで問題なく使えます。

「C／C++なんぞは使わん、[ST言語](#)だけで十分」という方は、このステップは飛ばしてください。

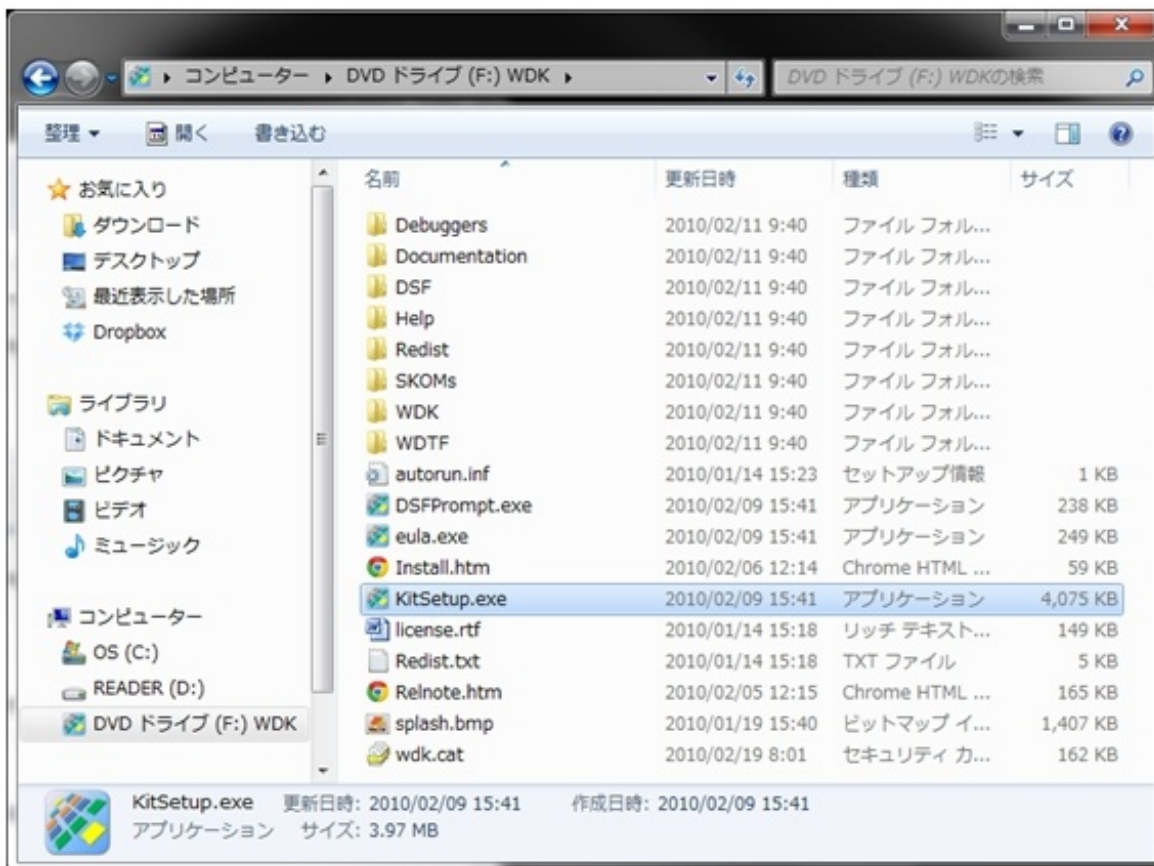
【Step 5】「Windows Driver Kit (WDK)」のインストール

TwinCAT 3でC／C++のリアルタイム制御を行うため必要なのだそうです。

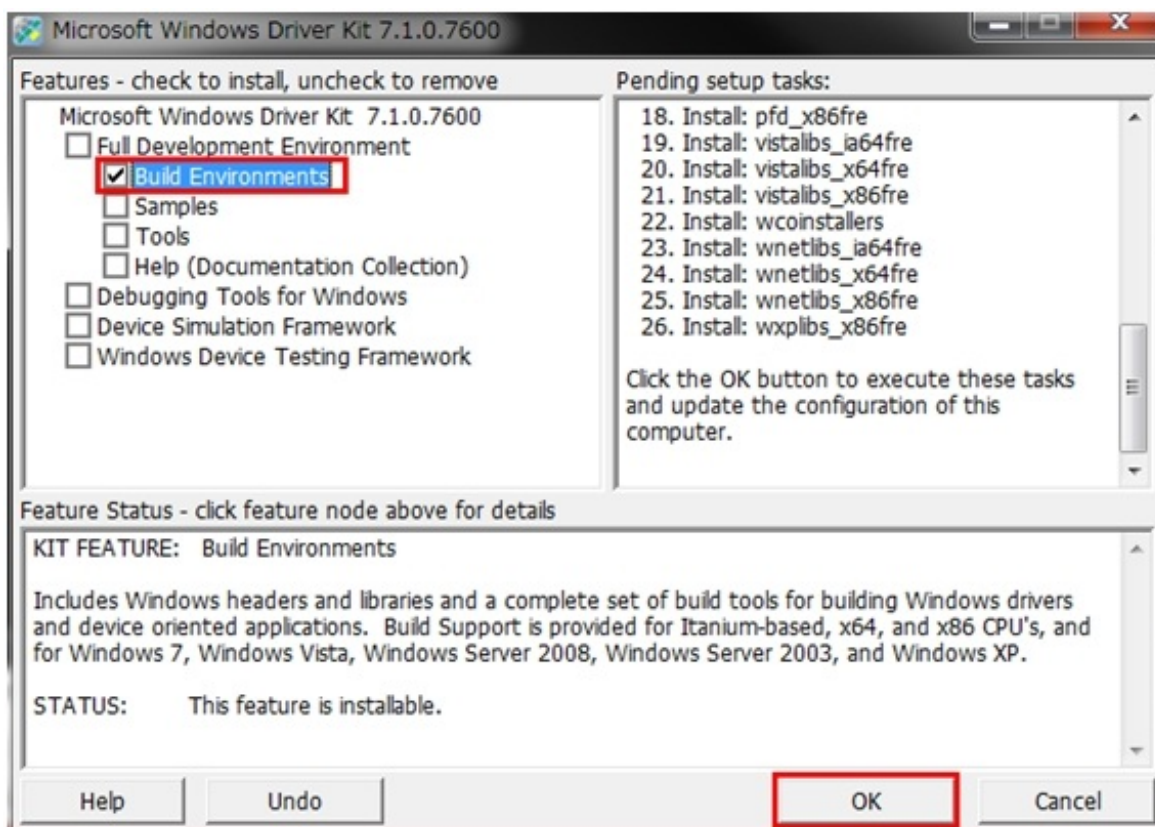
MicrosoftのWebサイトから「Windows Driver Kit Version 7.1.0」を[ダウンロード](#)してください。

ですが、これがまた大変なのです。ダウンロードファイルはISOイメージなので、ファイルからCDまたはDVDを作成する必要があります(ISOファイルをいじっているとWindowsが「ああせい、こうせい」と、いろいろ言うので、その通りやっていたらCD／DVDを作ることができます。運よく私はできました)。

で、そのCD/DVDの中にある、「KitSetup.exe」をダブルクリックして起動します。



TwinCAT 3 C/C++には「Build Environments」だけが必要です。項目をチェックし、「OK」をクリックしてください。



【Step 6】TwinCAT3のインストール

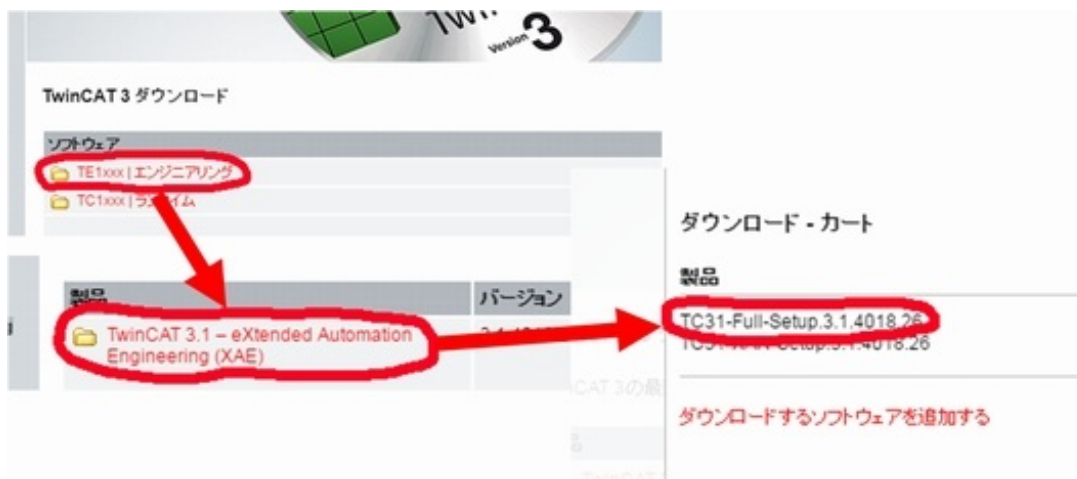
よー——やく、ここまできて、TwinCAT3のインストールを開始できます。

TwinCAT 3はベッコフの[Webサイト](#)から無料でダウンロードできます。

ですが、ここで私はダウンロードするものを間違えて、どつぽにはまりました。

選ぶのは、「TC31-Full-Setup.3.1.40xx.0.exe」です。

このダウンロードの際に、ユーザー登録などが要求されますので、ここは（無駄な抵抗をせずに）ユーザー登録してください（ただ、このユーザー登録、なんか調子が悪い時が多いようで、3回くらい登録した記憶があります）。



クリックで拡大

これをダブルクリックするとzipファイルのダウンロードが始まります。その後、zipファイルを解凍すると「TC31-Full-Setup.3.1.40xx.0.exe」ができていますので、それをダブルクリックしてインストールを実行してください。



「I accept the terms in the license agreement」のボタンをクリックしてライセンスに同意し、「Next>」をクリックしてください。

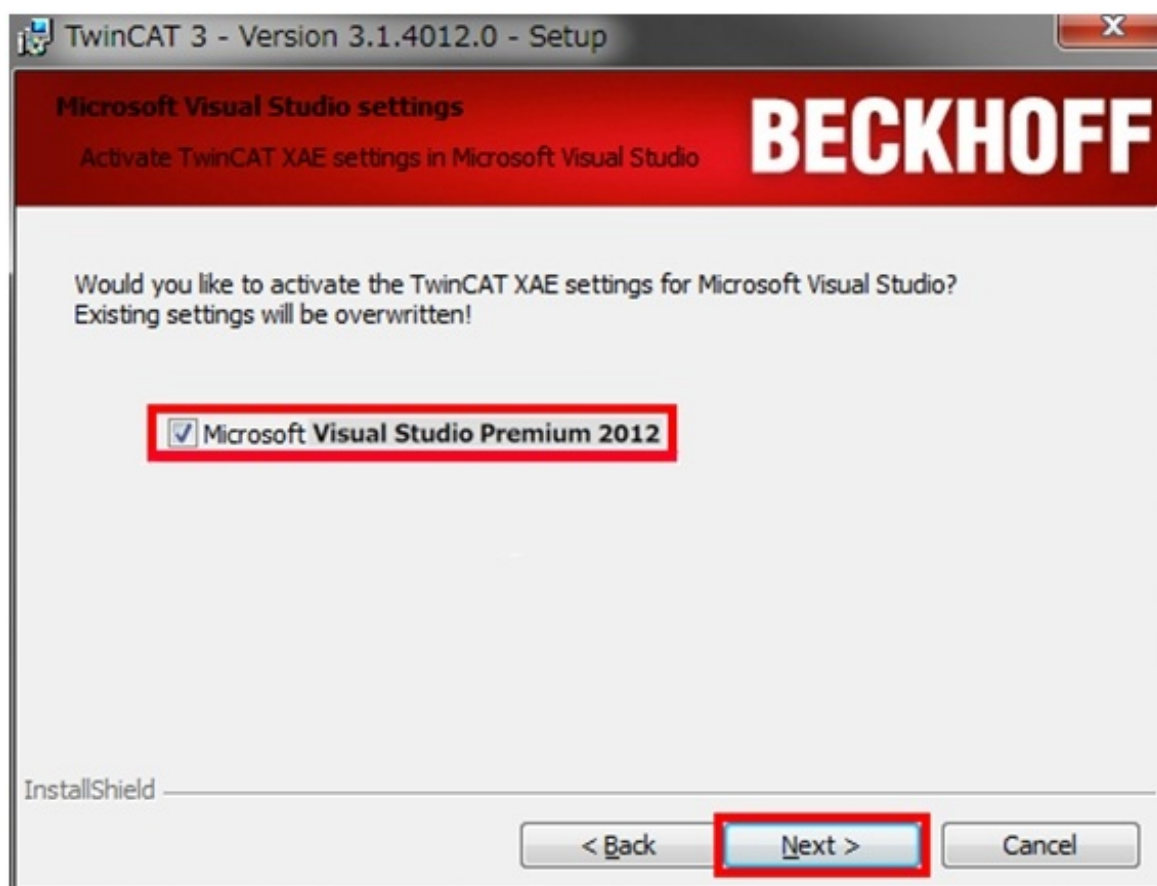
さらに、登録するユーザー名と会社名を記入し、「Next>」をクリックしてください。次にインストールのタイプを聞かれてきますが、何も考えずに「Complete」を選択し、「Next>」をクリックしてください。

VS 2010／2012／2013のProfessional／Premium／Ultimateがインストール済みの

場合、TwinCAT 3を統合するVSの選択画面が表示されます。

私の場合、厄介なことに、昔インストールした？ のかもしれないVSの選択肢が出てきたせいで選択を間違えてしまい、ここでもドツボにはまりました。

面倒を避けるためにも、ここは先ほどインストールしたVS1種類だけ(私の場合VS2012のPremiumだけ)をチョイスしておくことをお勧めします。



インストールには10分程度かかります。最後に、「Finish」をクリックすると、再起動するかと聞いてきますので、再起動に成功することを真剣に祈りながら、「Yes」を押下してください。

繰り返しますが、TwinCAT3のインストールは、「お正月の年賀状作成ソフト」のインストールとは、訳が違います。心の底からの真摯な祈りと愛が必要とされるのです([参考:『技術』は『愛』というプラットフォームの上でしか機能しない](#))。

【Step 7】EthernetのNICをTwinCAT専用NICにする

さて、ここから、いよいよNICの設定を始めます。

前述した通り、EtherCATのNICは、フツウのNICであることは許されません。

厳密に「1000分の1秒レベルの周期」、かつ、「百万分の1秒(1 μ s)のジッタ(ゆらぎの誤差)*)」で、正確にEtherCATのフレームを送出し続けるNICに改造しなければならないからです。

*) 1 μ sとは、1秒間に地球を7周半回ることのできる光が、3mだけ進める時間です。

そのためには、TwinCAT3を立ち上げなければならないのですが、驚くことに、TwinCAT3は、スタートボタンにないのです*)。全く理由が分かりませんが、Windows7の右下のタスクバーからでないと、立ち上げることができないのです。

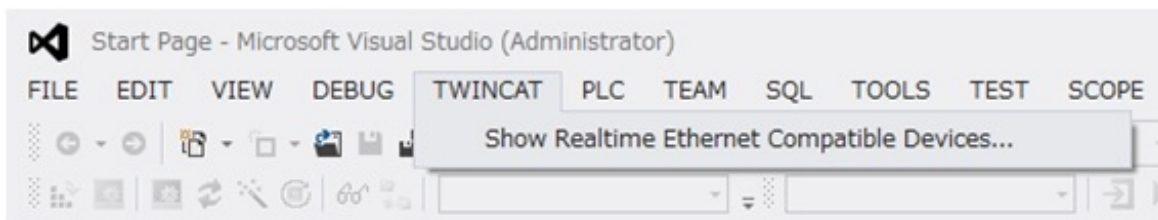
―― 一体何考えているんだろう？

多分、多くの人が困惑すると思うので、起動時のイメージを撮影しておきました。

*)これ、私の完全な勘違いでした。[スタートメニュー]→[すべてのプログラム]→[Beckhoff]→[TwinCAT3]から立ち上げできる旨、小幡さんからご指摘を受けました(江端も確認済み)。でも、同じ勘違いをする人は絶対にいるハズだと踏んで、上記の文章を残すことにしました。

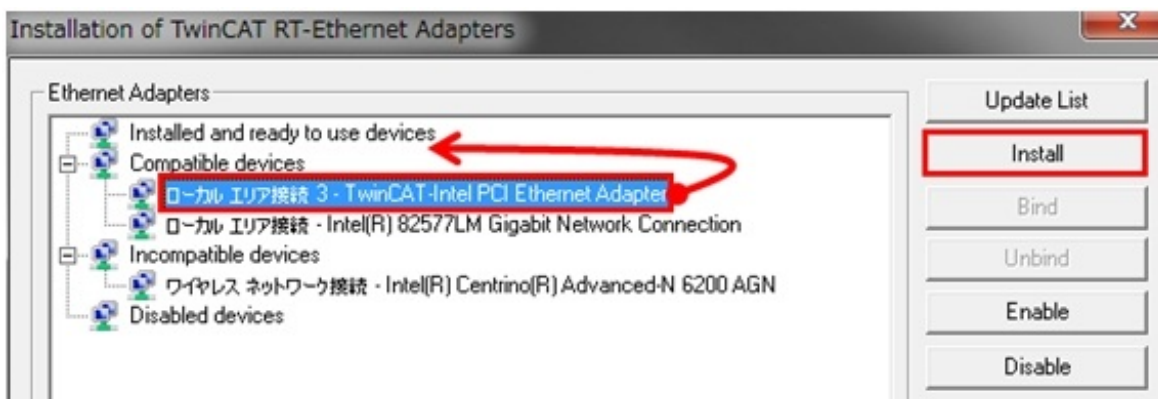
この立ち上がったツールは、"TwinCAT XAE (eXtended Automation Engineering)"と呼ばれるようですが、ここでは簡単に「TwinCAT3」と呼ぶことにします。

次に、立ち上がったTwinCAT3のメニューバーで「TwinCAT→Show Realtime Ethernet Compatible Devices」を選択してください。



Compatible devicesに表示されているNICからドライバをインストールするものを選択し、「Install」ボタンをクリックしてください。

すると、そのNICは、Installed and ready to use devicesの方に移動して、TwinCAT3の専用のNICに改造されます。



気を付けていただきたいのは、「PCのどのNICが、TwinCAT3に割り当てられたか分からなくなることがある」という点です(私のPCには、5つもNICが付いていたので、該当するNICを見つけるためだけに、Wiresharkをインストールして動かすなどという、間抜けなことをやることになってしまいました)。

事前にNICの名前などを記載したふせんを、PCのNICのポートに張りつけておくなどの対処をしておいた方が良いでしょう。

【Step 8】ESIファイルのインストール

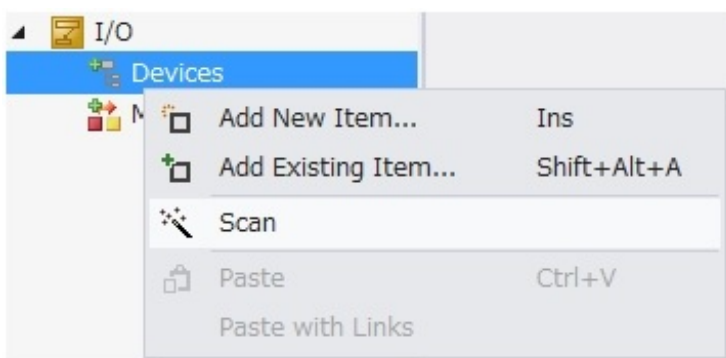
ESIファイルとは、EtherCATスレーブ情報(EtherCAT Slave Information:ESI)ファイルはEtherCATスレーブのメーカーや型番などの認識情報、設定や通信に必要な情報が記述されたXMLファイルです。まあ、メイドの個人情報とっていただければ良いでしょう。

ベッコフのスレーブ(メイド)を使う場合は、特に何もしなくて良いです(インストール済み)が、それ以外の会社のスレーブ(メイド)を使う時は、その会社がリリースしている(はず)のESIファイルを、「C:\¥TwinCAT¥3.1¥Config¥Io¥EtherCAT」に放り込んでおくだけで大丈夫のはずです。

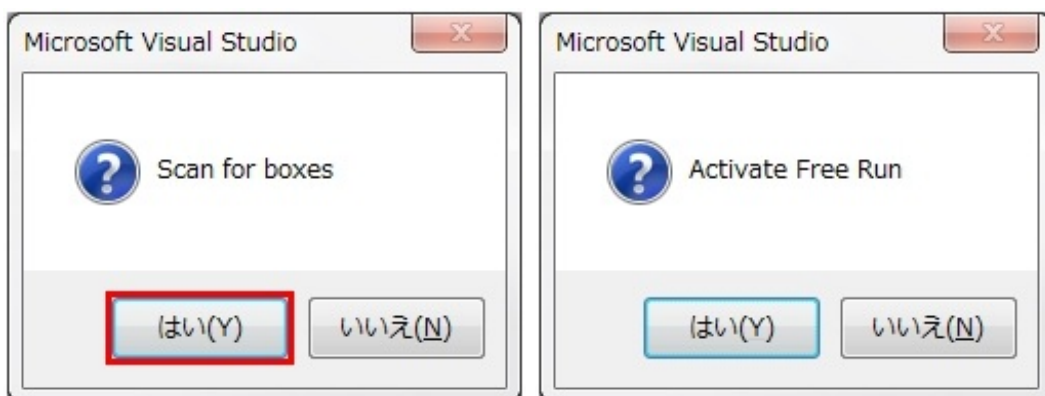
【Step 9】スレーブ(メイド)たちの接続

では、ここから、EtherCATの十八番(おはこ)である、自動構築のマジックをご覧くださいませ。

TwinCAT3の左側にあるツリーの「I/O→Devices」右クリックし、「Scan」を選択してください。



次のダイアログで「OK」を押すと、



てな画面がでてきますので、「はい」「はい」と押し進めると、現在接続されているスレーブ（メイドたち）が、TwinCAT3の左側のツリーに、全員、どどどどとラインアップされます。

—— 壮観

うん、うん、やっぱり、EtherCATの自動構築はいいなあ。

ここで、よーーーーやく、TwinCAT3によるEtherCATの構築が一通り完了です。

でもね、TwinCAT3のインストールや構築、全然ラクじゃないでしょう？ まあ、制御LANの世界でラクしようというのが、どだい無理な話なのかもしれませんが。

□

では、せっかく、スレーブ（メイド）たちをコントロールできる状態になったのですから、ちょっと、メイドたちと遊んでみましょう。

まずは手動で、LEDライトを付けたり、センサー値を読み取ったりしてみます。

まず、メニューバーから「TWINCAT→Toggle Free Run State」を選択してFree Runモードにしてください（"Free Run"とは何ぞや、てなことは考えなくても良いです）。



クリックで拡大

そして前回のコラムで紹介したメイド"Rio"のLEDを点滅させます。

TwinCAT3の左側に表示されているツリーから"Rio"のデバイスを選択して、そのデジタル出力ポートの信号を手動でオンにしてみます。

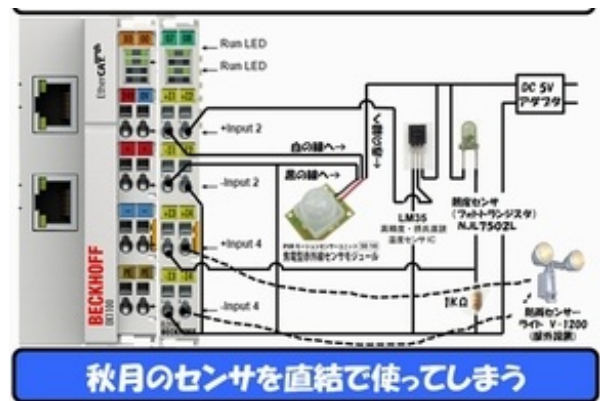
その映像がこちらです。

では、今度は"Yuko"のセンサー値の取得をり

第2のメイド "Yuko" の配線

アルタイムで見てみましょう。

4つのセンサー値がリアルタイムで動いているのが見えます。その映像はこちらです。



LED点滅用のスイッチも、センサー値を表示する画面も全部作って待っていてくれる。これが、ESI (Ethernet Slave Information) とEtherCATツール (TwinCAT3) の連携によるマジックです。

もし、SOEMで同じようなことをやろうとすれば、これ、全部私が1人で画面を作らなければなりません、はっきりいってそんなことは不可能です。

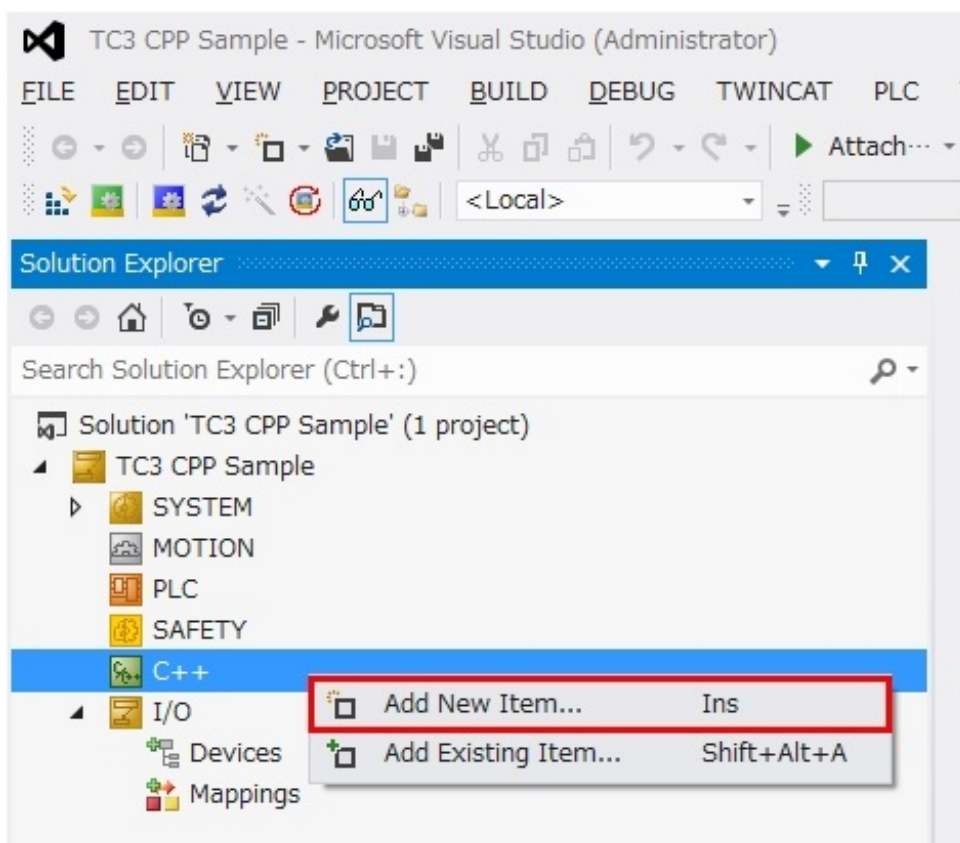
TwinCAT3の「ラスボス」

では最後に、このTwinCAT3の“ラスボス”であるプログラミング方法を紹介しましょう。

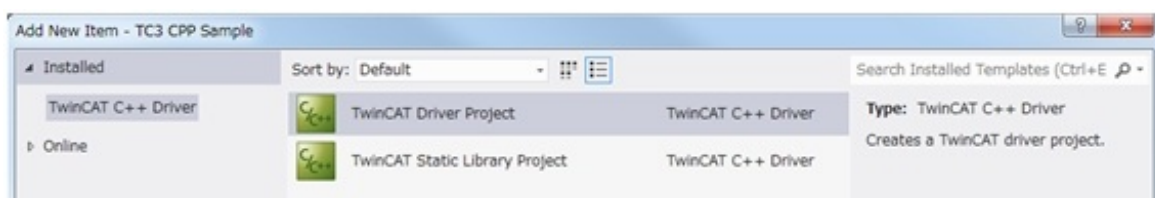
「アナログ入力のスレーブに接続されている光センサーに懐中電灯を向けると、デジタル出力のスレーブに接続されているLEDが点滅するようにする」

を「C/C++言語」で作ります。

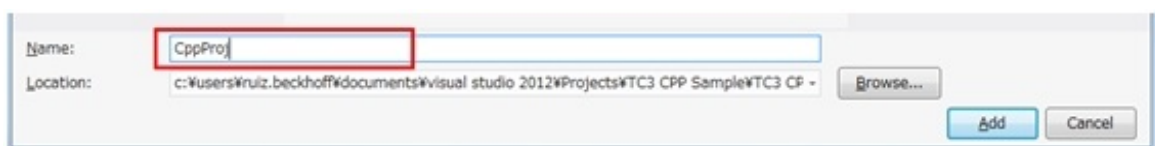
まず、TwinCAT3の左側にあるツリーの「C++」(C++設定)を右クリックして「Add New Item...」(新しいアイテムの追加)を選択してください。



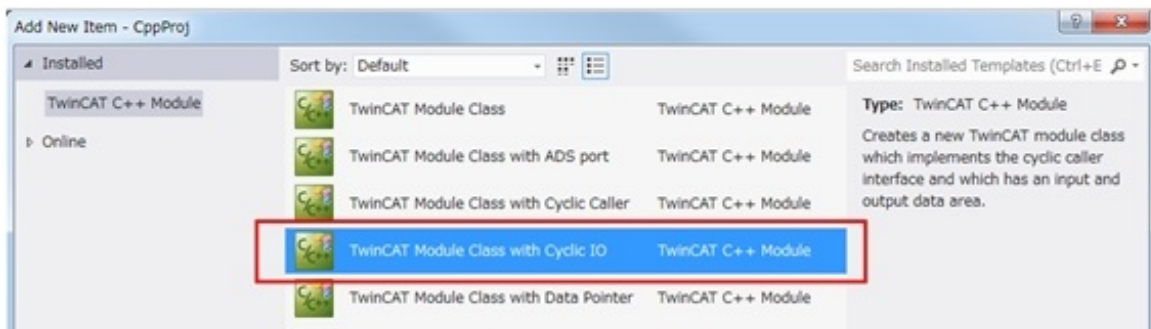
「TwinCAT Driver Project」(TwinCATドライバプロジェクト)を選択し、プロジェクト名を入力後、「OK」をクリックしてください。



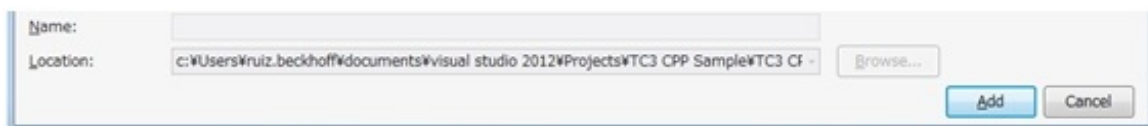
画面下のプロジェクト名は、CppProjとしておいてください。



「TwinCAT Module Class with Cyclic I/O」(サイクリックIO付きTwinCATモジュールクラス)を選択し、「OK」をクリックしてください。



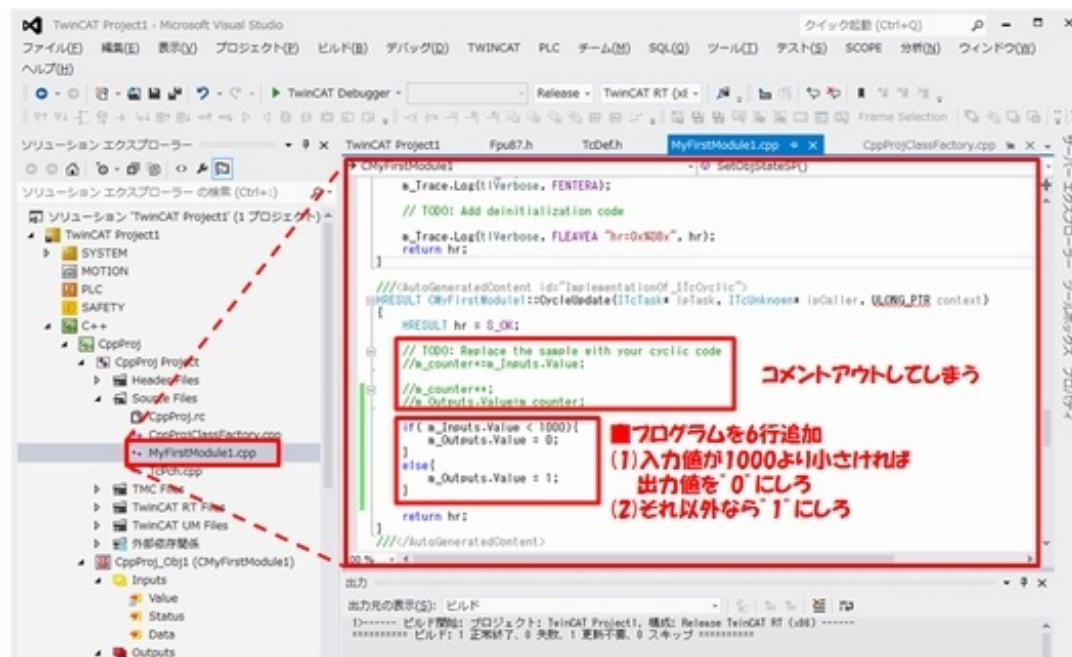
で、画面下のAddボタンを押してください。



自動生成されたソースコード(MyFirstModule1.cpp)を開きます。

で、ソースファイルの169行にある、周期的に勝手に呼び出される関数CycleUpdate()の内容を、以下のように変更します。

```
if (m_Inputs.Value < 1000) {
    m_Outputs.Value = 0;
}
else{
    m_Outputs.Value = 1;
}
```



照度センサの数値の大小だけで、LEDのON/OFFを決める(単純な)プログラム

クリックで拡大

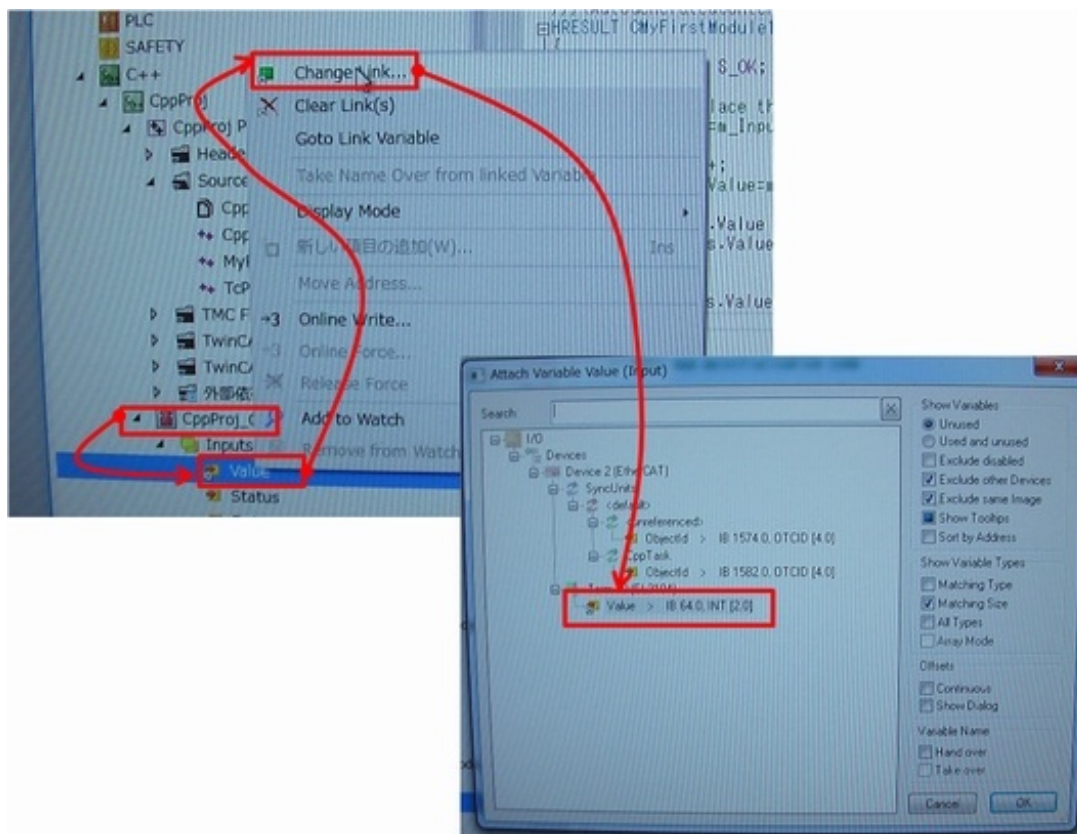
これは、m_Outputs.Valueの値が1000を超えたら、m_Outputs.Valueの値を変更するというプログラムです(もちろん、この"1000"という値は、私の使っているセンサーの値です。環境に応じていろいろ変更させる必要があります)。

しかし、この時点で、m_Inputs.Valueやら、m_Outputs.Valueやらが、どこにも定義されていません。

TwinCAT3の本領発揮

で、ここからが、TwinCAT3の本領発揮なのです。このTwinCAT3というヤツは、既にSCANされているスレーブ(メイド)たちのポートの値を、簡単にプログラムにリンクしてしまうのですよ。

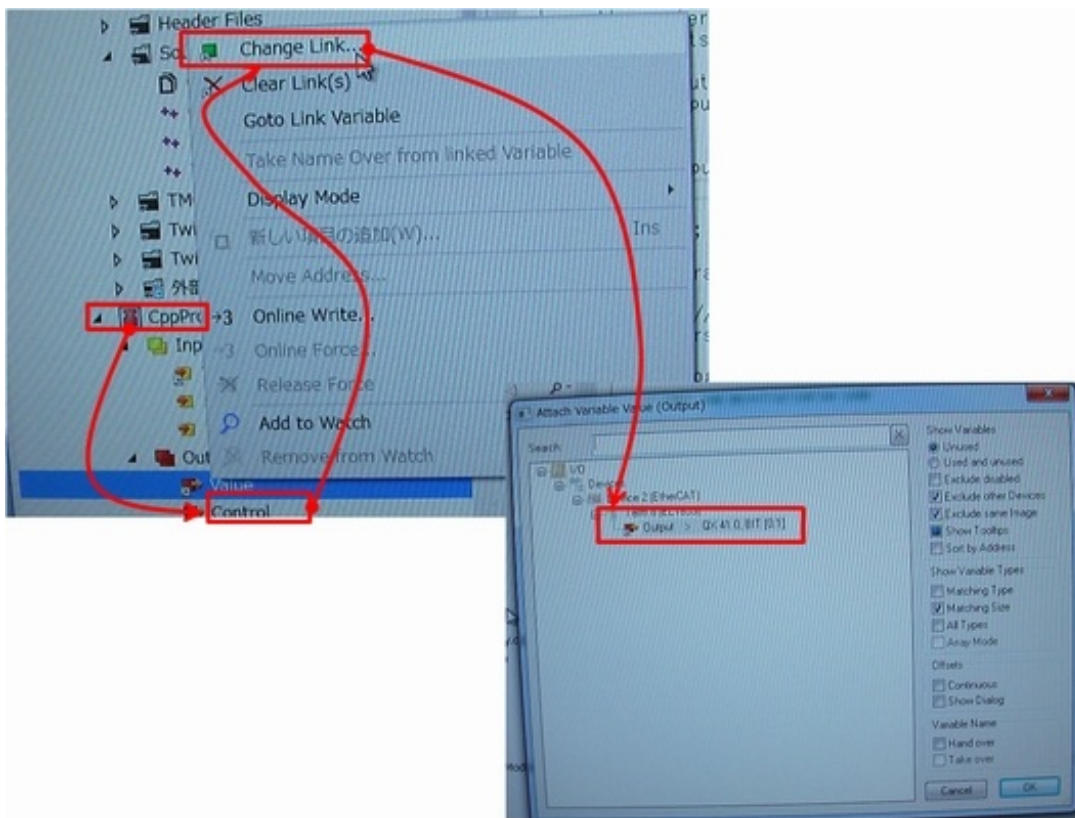
まず、m_Inputs.Valueのリンクを、以下の図の順番通りに行ってください。



クリックで拡大

これで、m_Inputs.Valueは、スレーブ(メイド"Yuko")の照度センサー用に割り当てたアナログポートの値になります。

次に、m_Outputs.Valueのリンクを、以下の図の順番通りに行ってください。



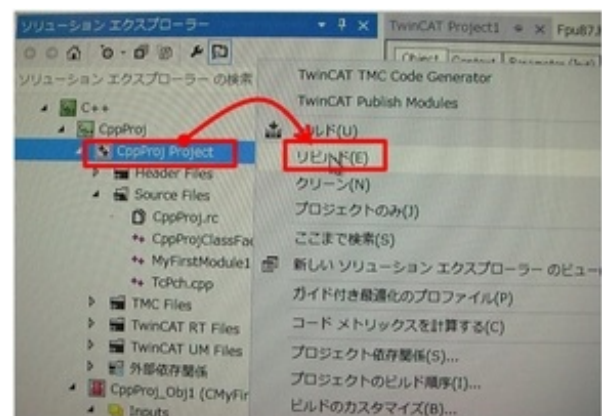
クリックで拡大

これで、m_Outputs.Valueは、スレーブ（メイド"Rio"）のLEDライトのデジタル出力ポートの値になります。

最後にリビルドを実施します。

まあ、初回はいろいろエラーが出てくると思いますが——頑張ってください。（私もあちこちいじっている内に、いつの間にかコンパイルが通りました）。

最後に、TwinCAT3のメニューの「TWINCAT」から「Activate Configuration」を選択するとプログラムが動き出します（動き出すまで、数秒のタイムラグがあるかな、という感じがします）。



クリックで拡大

実験結果は、こんな感じになりました。

これで、SOEMだけでなく、TwinCAT3でも、江端家のホームセキュリティシステムを稼働させられるメドができました。

TwinCAT3を使ってみて

さて、ここまでのTwinCAT3に対する私の感想をまとめてみたいと思います。

(1) TwinCAT3のインストールや構築が「楽々」とは、口が割けても言えません。大抵どこかで失敗して、再インストールすることになるので、最初から何回か失敗するつもりで取り組んだ方が、精神安定上、良いと思います。

(2) TwinCAT3のインタフェースも、そんなに分かりやすいとは思いませんでした。正直、まだ、プログラムの起動、停止のタイミングも良く分からない^{*)}し、私の頭の中では、いまだに各モードの違いも整理できていません。

*) 小幡さんコメント:「TwinCAT3 C/C++タスク(C/C++プログラム)はRUNモードに移行すると自動的に起動します。プログラムを停止させるには、PCのシャットダウンまたはCONFIGモードに移行してください」。

TwinCAT3も、やはり硬派の制御用ツールの1つで、

「貴様、制御LANを担うエンジニアの誇りがあるなら、ひたすらTwinCAT3を使い込め！」
「押忍！」

という、体育会系の空気がするのは否定できません。

(3) 現時点で、TwinCAT3の入門書や日本語の助けは、絶望的に期待できません。

そもそも制御LANという世界自体がニッチな世界ですから、そこに、先人のメモを期待すると自体がムチャというものかもしれません。ここに「日本語」というフィルターが入れると、もう絶望的に情報が消えます。

現時点で、親切なマニュアルや入門書は、全世界のどこにも見つけられません。「初めてのTwinCAT3」とか「この1冊で分かるTwinCAT3」などという本は存在しませんし、今後も出版されないでしょう。

しかし、それでもTwinCAT3はすごいのです。

(4) スレーブをつなぐだけで、スレーブ情報がキレイに表示される。コンフィギュアードアドレスやら、なんやらのアドレスを気にする必要一切なし。もちろん、マスタのメモリ配置も全然考える必要なく、リアルタイムでスレーブの状態を直接見ることができる。

正規化されたセンサーデータの値が、同時に4つ、リアルタイムでドタバタ動いている画面を見て、私は感激の涙を抑えられませんでした。

(5) プログラムに、表示されているスレーブのポートをツールの上でリンクするだけで、制御プログラミングが可能 ―― これは、プログラマーにとっては一種のマジックキングダムです。

当然、マスタのどのアドレスの何ビット目とかリトルエンディアンなどという、面倒なことを考える必要一切なし、というのも本当に助かるものであることは、制御プログラムを作っている人

には、容易に理解できると思います。

(6) (今回紹介しませんでした)、特にすごいと思ったのは、スレーブのトポロジーがGUIで表示されるところです。スレーブを取り違えることって、結構あるので、こういう機能は本当にうれしいです。

まあ、そもそもオープンソースでフリーのSOEMと比較するのは、酷というのですが、メモリの値を、「この私」が直接読み込み、ビット解析をして、プログラミングしなければならないSOEMに比べれば、TwinCAT3は、そりゃもう「夢と魔法の王国」ですよ。

嫌な予感

しかし、その一方で、私は嫌な予感がしています。

EtherCAT + TwinCAT3 のような装置やツールによって、IT業界の人間が、大挙してドバッと制御の世界に乱入してくる可能性は高いです。

IT業界の人間が全て「意識高い系」みたいな奴とはいいません。

しかし、「お互いリスペクトできるパートナーシップを築いて、シナジー効果を生んでいって、ブレインストーミングをしながらロジカルシンキングで考えるべきだ」というような、内容がスカスカの用語を頻用するITベンチャーの社長とその社員たちが、

油の臭いのする工場のラインで、現場のエンジニアと大声で怒鳴りながら、泥臭く仕事をする、そういう私たちの世界に乱入してくる —— 考えるだけでもおぞましいですが —— そういう時代がやってくる。

「通信フレームを、百万分の1秒(1 μ s)以内の揺らぎで、毎秒8000個送出する」という超高精度なリアルタイム通信を、EtherCATやTwinCAT3に押しつけて、おいしいところだけ食いにくる奴らがやってくる。

オブジェクト指向とプロファイルとコピペだけで、鼻歌まじりに、制御プログラムを書き上げるヤツらがやってくる。

—— 嫌だ

そんな奴らに、こっちに来てほしくない。

「情報と制御の融合」「IoT」「Industrie4.0」「Industrial Internet Consortium」なんて、私にはどうでも良いことです。

もちろん、制御の情報の融合は素晴らしいことです。IoTは誰にでも制御できて、使えるようになるものであることが望ましいと思う。世界のためにはそれが一番いい。

でも、私（江端）のためには、制御の世界はクローズで構わん。

世界がどうだろうが、私の知ったことか。

「私はあまり世界にコミットしたくないんです。自分だけの世界でぬくぬくしていたいんです」
—— と、心からそう思っているのです。

そう考えると、—— うん、TwinCAT3の構築やインタフェースは、今のまま難しくていい。
TwinCAT3は、「お正月用の年賀状印刷ソフト」のように簡単になる必要はない。

ヤツらがこっちの世界に参入してくることを阻むためにも、TwinCAT3に丁寧な日文翻訳のマニュアルは不要です。「楽々TwinCAT3」の出版も断固阻止です。



しかし、残念ながら、世界は私の思うようには動いてくれないだろう、とも思うのです。

EtherCATは、リアルタイム制御という、制御の世界の最高峰の技術を、「サバのみそ煮定食」くらいの普遍性を持って、この世界に殴り込みをかけてきました。

近い未来に、制御の世界までもが、私の嫌いなTwitterやFacebookのようなSNSの世界のようになるのは、もはや時間の問題だろうと思えます。

本当に、「EtherCATとTwinCAT3の連合軍は、私が最も望んでいなかったパラダイムを、制御の世界に持ち込んできたのだ」と、今、私は深いため息をついています。

特別協力:

本連載では、スレーブの提供などで [ベッコフオートメーション](#) にご協力いただいております。



Profile

江端智一（えばた ともいち）

日本の大手総合電機メーカーの主任研究員。1991年に入社。「サンマとサバ」を2種類のセンサーだけで判別するという電子レンジの食品自動判別アルゴリズムの発明を皮切りに、エンジン制御からネットワーク監視、無線ネットワーク、屋内GPS、鉄道システムまで幅広い分野の研究開発に携わる。

意外な視点から繰り出される特許発明には定評が高く、特許権に関して強いこだわりを持つ。特に熾烈（しれつ）を極めた海外特許庁との戦いにおいて、審査官を交代させるまで戦い抜いて特許査定を奪取した話は、今なお伝説として「本人」が語り継いでいる。共同研究のために赴任した米国での2年間の生活では、会話の1割の単語だけを拾って残りの9割を推測し、相手の言っている内容を理解しないで会話を強行するという希少な能力を獲得し、凱旋帰国。

私生活においては、辛辣（しんらつ）な切り口で語られるエッセイをWebサイト「[こぼれネット](#)」で発表し続け、カルト的なファンから圧倒的な支持を得ている。また週末には、LANを敷設するために自宅の庭に穴を掘り、侵入検知センサーを設置し、24時間体制のホームセキュリティシステムを構築することを趣味としている。このシステムは現在も拡張を続けており、その完成形態は「本人」も知らない。

本連載の内容は、個人の意見および見解であり、所属する組織を代表したものではありません。

関連記事



[電力という不思議なインフラ\(前編\)～太陽光発電だけで生きていけるか?～](#)

太陽光発電のみで生活する――。これが現実になれば、私たちはもう原発やら電気代やらを心配することなく、夢のような生活を送ることができるでしょう。市販のソーラーパネルの「発電出力」だけを見れば、あながち不可能ではない気もしてしまいます。ですが、太陽光発電には大きな“落とし穴”があるのです。



[トラブル遭遇時の初動方針は、「とにかく逃げる!」](#)

どれだけ周到に準備をしたとしても完全には回避できない――。悲しいかな、トラブルとはそういうものです。悪天候でフライトがキャンセルされたり、怖い兄ちゃんが地下鉄に乗り込んできたり、“昼の”歓楽街でネーチャンにまともりつかれたり……こういうものは、はっきり言って不可抗力です。実践編(海外出張準備編)の後編となる今回は、万が一トラブルに遭遇した場合の初動方針についてお話しします。



[ルネサス 自動運転車の頭脳となる次世代SoC発表](#)

ルネサス エレクトロニクスは2015年12月2日、車載情報システム向けSoC「R-Carシリーズ」の第3世代品を発表した。2018年以降に市販される自動車への搭載を見込んだ製品群。第1弾製品として同日、サンプル出荷を開始した「R-Car H3」は“自動運転時代のSoC”と位置付けたハイエンド品で、最先端となるTSMCの16nm世代FinFET+プロセスを採用し、高性能な処理能力を盛り込んだ。



[人間の脳が握る、デバイス低消費電力化の鍵](#)

ウェアラブル機器に欠かせない要件の1つに、低消費電力がある。「第2回 ウェアラブルEXPO」のセミナーに



登壇した日本IBMは、超低消費電力のコンピュータとして、人間の”脳”を挙げ、IBMが開発中の「超低消費電力脳型デバイス」について語った。

Copyright © 2016 ITmedia, Inc. All Rights Reserved.

