

## ソフトウェアに自信を持つ

ユニットテストと呼ばれる、繰返しテストを行なう事でソフトの品質を高める手法が、ソフトウェア業界で注目されています。この手法をPLC(プログラマブルシーケンサ)に応用しようというのがナントダ<sup>1</sup>です。

パソコンの中だけで完結するソフトであれば、外部との影響をほとんど考える事なく、自由に何度も実行させながら動作を確認する事が出来ます。

しかし、PLCでは様々な装置を制御するので、間違っただけで機械を壊してしまう事にもなりかねず、自由に動作させる事が出来ません。

実機を用いてテストを行なう場合どのような障害があるのでしょうか。

### 時間がかかる

PLCで制御する大半の動作は、時間待ちの動作です。ワークが届くまで、動作が完了するまで、センサーが入るまでといった何かを待つ動作が大半を占めます。この時間をなくす事が出来れば、今までと比較にならない回数のテストが可能になります。

### 条件作成が困難

特にエラー処理の確認は、困難を伴います。シリンダーの動作で異常が出た場合に、ブザーを鳴らす仕様の確認はどの様にしましょうか。力づくでシリンダーが動かない様にしますか。それともせきかく調整したセンサーをいじったり、エアを抜いて異常の状態を作りますか。この後には、それを元に戻す作業が待っています。センサーの調整が微妙な箇所では、テストにかかった時間以上に調整に時間を取られてしまうかもしれません。ほとんど発生する事のない仕様のために多大な時間が費やされてしまいます。

### 単純ミス

エラー処理確認の為にソフトを一部変更して対応する場合があります。この時に元に戻す事を忘れてしまったという事はないでしょうか。そのまま出荷されて、客先で本来動くべき所が動かないという事になったら大変です。

### 副作用

ソフトウェアを変更した場合、知らないうちに他の箇所に影響を与えてしまい、弊害を伴う事が良くあります。実機では様々な要因によりテストが十分が出来ず、このような副作用を見つける事が出来ない場合があります。もしも十分にテスト出来れば副作用の発見が出来たかもしれません。

この様に実機では、様々な弊害により十分なテストが出来ない場合が多く、あまり自信が持てないまま出荷したりしてないでしょうか。1回動くのを確認したというより10回確認して10回とも問題なかったらどうでしょうか。100回、1000回と回数が増えれば増える程、ソフトの品質に自信を持つ事が出来ます。

## 仮想入出力

ナントダではPLC内に、仮想的な入出力を持たせる<sup>2</sup>事で実際の入力と関係なく、入力条件を与える事が出来、実際の出力に影響を与えない様にテストする事が可能です。

本来はPLCに仮想入出力の機能があれば良いのですが、残念ながらこの機能を持ったPLCはまだないので、PLCソフト(ラダー)で作ります。

実機での確認方法  
実際に入力を操作し出力を確認する

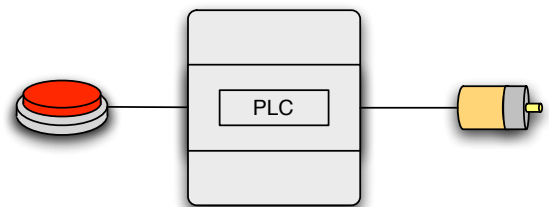


図1

<sup>1</sup> 試作品で今回は参考出展。

<sup>2</sup> 特許出願中。

図2はキーエンス社のKVシリーズ<sup>3</sup>の場合の仮想入出力回路です。17500のコイルがOFFの場合は実際の入力(0000)が仮想入力(1000)として機能し、ONの場合は17000が仮想入力(1000)として機能します。ナントダから17500をONにし、17000をON/OFFさせる事で自由に入力条件を作る出す事が可能です。出力も同様で、17500がONの場合は実際の出力(0500)に出力<sup>4</sup>されません。

PLCソフト制御部を作成する場合は、実際の入力である0000~0415の代わりに1000~1415に、出力である0500~0915の代わりに1500~1915に置き換えるだけで、今までと同様に作成する事が出来ます。今まで行って来た事はそのまま、仮想入出力回路部<sup>5</sup>を加えるだけです。

## 実機との分離

PLCに仮想入出力機能を持たせた事で、仮想入力を自由に操作する事が可能になりました。ナントダから仮想入力を操作し、仮想出力がどうなっているかを確認する事でPLCソフトが意図した通りに動作しているかテストを行います。つまり、図3の様に実際の入出力(実機)と分離してテストする事が可能になった事になります。

実機と分離し、自由にテストを行なえるようになると、どの様な事が出来るでしょうか。

### 時間を省力化

理論上はワークが届くまで、といった時間を0秒にする事が出来、大幅な時間節約になります。また実機では、動作させる前の段取りや、終了させるための手順があり、途中で止める事が出来ない事がよくあります。

全てのワークを排出しないと終了出来ないという手順であれば、簡単なテストのために数分単位で時間が費やされてしまいます。

しかし、実機と分離した場合は実際の出力を変化させません。確認したい所だけ確認したら中断する事が出来、大幅な時間短縮につながります。

### 条件作成が容易

ナントダから仮想入力を操作するのは通信でコマンドを送るだけなので、実機では難しい条件も自由に作り出す事が可能です。

### 単純ミスの回避

一時的にソフトを変更しても、変更前の正規の条件でテストが作られていれば、ナントダのテストに失敗します。最後にナントダで一通り全てのテストをする事で、単純ミスを発見する事が出来ます。

### 副作用

副作用がある場合は、ナントダで一通り全てのテストを実行する事で、洗い出す事が可能になります<sup>6</sup>。

### 人員の削減

実機では、操作スイッチ、センサー、出力デバイスが離れた所に配置されている場合がほとんどです。この様な状況でテストするには、数人でそれぞれのデバイスに配置し、場合によっては無線機で連絡を取りながら行う必要があります。ナントダでは一人で簡単にテストができます。

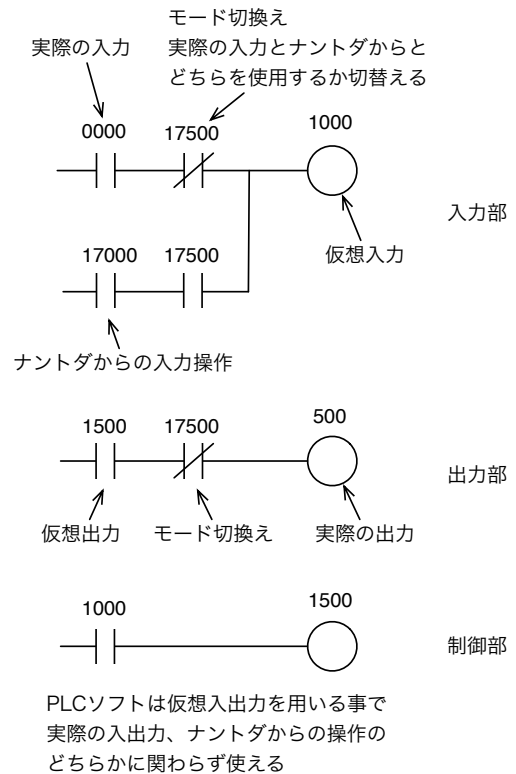


図2

ナントダでの確認方法  
PLC内部の仮想的な入力を操作し仮想的な出力を確認する

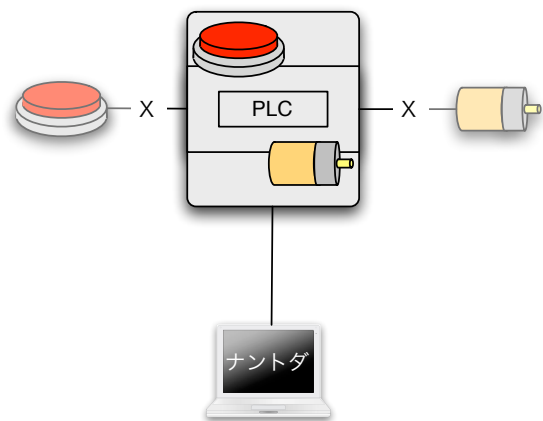


図3

<sup>3</sup> 現在はキーエンス社のKVシリーズのみの対応。

<sup>4</sup> 通常ONになっている出力は、テストしている間はOFFになります。

<sup>5</sup> 自動生成させる事を予定しています。PLCソフト上でコピー&ペーストで簡単に追加出来る様になります。

<sup>6</sup> テストとして組み込まれていない項目は検出出来ません。

# テストの作成

ナントダのテストは簡単なコマンドを組み合わせることで一つのテストシーケンスを作ります。例えば図2の制御部ではスイッチ(0000)を押している間モーター(0500)が動作します。これをテストするテストシーケンスは以下のようになります。括弧書きはナントダでの表現です。

1. 0000をONにする。 (Set 0000)
2. 0500がONになっている事。 (Evaluate 0500 ON)
3. 0000をOFFにする。 (Reset 0000)
4. 0500がOFFになっている事。 (Evaluate 0500 OFF)

ここで、2、4のEvaluate(評価)で失敗するとテスト失敗となり、その後のコマンドは実行されません。このテストの最初にEvaluateコマンドを追加し、開始時にモーターが停止している事を確認する様にします。

- 1) 0500がOFFになっている事。 (Evaluate 0500 OFF)
- 2) 0000をONにする。 (Set 0000)
- 3) 0500がONになっている事。 (Evaluate 0500 ON)
- 4) 0000をOFFにする。 (Reset 0000)
- 5) 0500がOFFになっている事。 (Evaluate 0500 OFF)

残念ながらこのテストは失敗する事があります。1で0000がどうなっているかわかりません。もしONになっていたら失敗してしまいます。ナントダでは全体的な前処理(Root Before Sequence)と全体的な後処理(Root After Sequence)があり、1つ1つのテストシーケンスの前後に必ず実行されます。全体的な前処理では初期値を与え、後処理では次のテストに支障がない様にします。全体的な前処理に0000をOFFにするコマンドを入れることでテストが成功する様になります。

また、シーケンス毎の前処理(Before Sequence)と後処理(After Sequence)があります。例えば原点復帰が完了している事を前提としたテストシーケンスの場合は、原点復帰させる手順を前処理に作成します。一つのテストシーケンスを実行すると次の様に実行されます。

全体的な前処理 -> シーケンス毎の前処理 -> テストシーケンス -> シーケンス毎の後処理 -> 全体的な後処理

# テストの実行

テストの実行はとても簡単でRUNボタンを押すだけです。RUNボタンを押すと、全てのテストシーケンスが実行され、全てが成功すると進捗を示すバーが緑になり(図4)、1つでも失敗があると赤で表示されます(図5)。赤のバーになってしまった場合は、PLCソフトを修正し、緑のバーになるまで繰り返してテストして下さい。ナントダは何回でも正確に繰り返してテスト出来ます。

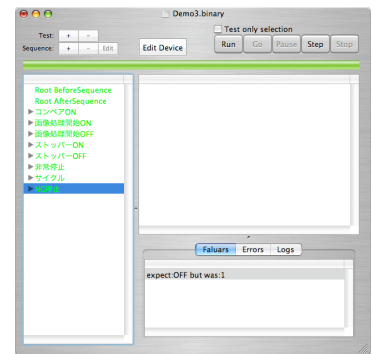


図4

# テストのポイントと仕様としての役割

PLCソフトでは入力に対する出力を制御します。テストを作成する時には表1の様に、出力デバイス毎に前提条件、ON条件、OFF条件に注目します。上で作成したテストの1)が前提条件に該当し、2)、3)、はON条件の確認、4)、5)はOFF条件の確認をしています。PLCの内部でどのようなソフトを組んでも、この条件を満たしていれば仕様を満足していると言えます。

テストは言い換えると仕様を記述する事になります。文章による仕様書ではなかなか伝わらない場合がありますが、テストでは明確に記述する事が出来ます。また、出来ないテスト出来ない事になります。

テストはソフトを制作する側の仕事ですが、発注する側で仕様書として提示し、要求している事が満足しているかどうかの確認の手法ともなり得ます。

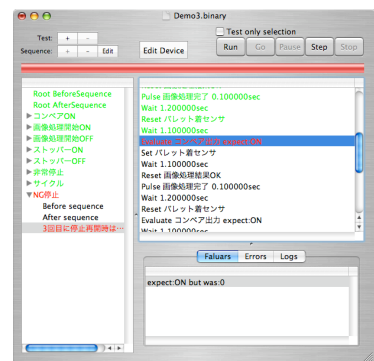
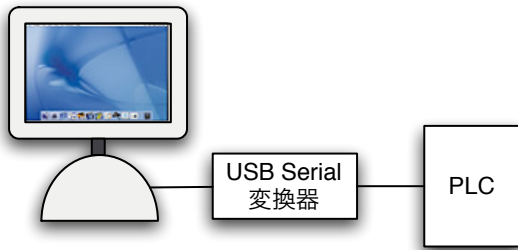


図5

出力デバイス	前提条件	ON条件	OFF条件
モーター	モーター停止している	スイッチをONにする	スイッチをOFFにする。

表1

## 構成



	条件
Mac	OS:Mac OS X 10.4(Tiger)以降
USB Serial変換器	Mac OS X対応の物。Keyspan社の製品は動作確認済み
ソフト	ナントダ
PLC	キーエンス社 KVシリーズ

## 今後の展開について

ナントダは試作品で今回は参考出展となっております。将来的には製品化<sup>7</sup>を目指しています。

今後の構想としては、仮想入出力回路の自動生成、コマンドの充実、KVシリーズ以外のPLCへの対応などを考えております。また、GUIの構成、操作性も今後の課題となっております。

商業的には、Windowsでの開発が望まれると思いますが、アイデアを形にするにはMac OS Xの方が迅速に行えるため、Mac OS Xで開発しています。仕様がある程度固まった段階でWindowsへ移植するかどうか検討する事になります。

2010年9月3日 作成

<sup>7</sup> 弊社は現在受託開発を主としていて、仕事の合間に開発しているため、なかなか製品化にたどり着けない状態にあります。