**PLCによるタッチパネル活用技術**



**第１章　タッチパネルの概要**

**１．タッチパネルとは・・・**

液晶パネルのような[表示装置](http://ja.wikipedia.org/wiki/%E3%83%87%E3%82%A3%E3%82%B9%E3%83%97%E3%83%AC%E3%82%A4_(%E3%82%B3%E3%83%B3%E3%83%94%E3%83%A5%E3%83%BC%E3%82%BF))とタッチパッドのような[位置入力装置](http://ja.wikipedia.org/wiki/%E3%83%9D%E3%82%A4%E3%83%B3%E3%83%86%E3%82%A3%E3%83%B3%E3%82%B0%E3%83%87%E3%83%90%E3%82%A4%E3%82%B9)を組み合わせた電子部品であり、画面上の表示を押すことで機器を操作する[入力装置](http://ja.wikipedia.org/wiki/%E5%85%A5%E5%8A%9B%E8%A3%85%E7%BD%AE)である。タッチパネルは表示と入力の2つの機能を備えており、[コンピュータ](http://ja.wikipedia.org/wiki/%E3%82%B3%E3%83%B3%E3%83%94%E3%83%A5%E3%83%BC%E3%82%BF)などの外部から受けた画像情報を[液晶ディスプレイ](http://ja.wikipedia.org/wiki/%E6%B6%B2%E6%99%B6%E3%83%87%E3%82%A3%E3%82%B9%E3%83%97%E3%83%AC%E3%82%A4)などで表示すると共に、操作者がその画面に表示された絵の点または領域に手で触れたり専用のペンや一般のペンで圧力を加える等により、触れられた画面位置の情報を感知して外部へ情報信号として出力する。

**２．使用用途**

　[銀行](http://ja.wikipedia.org/wiki/%E9%8A%80%E8%A1%8C)など[金融機関](http://ja.wikipedia.org/wiki/%E9%87%91%E8%9E%8D%E6%A9%9F%E9%96%A2)の[ATM](http://ja.wikipedia.org/wiki/%E7%8F%BE%E9%87%91%E8%87%AA%E5%8B%95%E9%A0%90%E3%81%91%E6%89%95%E3%81%84%E6%A9%9F)、[自動販売機](http://ja.wikipedia.org/wiki/%E8%87%AA%E5%8B%95%E8%B2%A9%E5%A3%B2%E6%A9%9F)（特に[鉄道駅](http://ja.wikipedia.org/wiki/%E9%89%84%E9%81%93%E9%A7%85)や[レストラン](http://ja.wikipedia.org/wiki/%E3%83%AC%E3%82%B9%E3%83%88%E3%83%A9%E3%83%B3)などの[自動券売機](http://ja.wikipedia.org/wiki/%E8%87%AA%E5%8B%95%E5%88%B8%E5%A3%B2%E6%A9%9F)）のような不特定多数が扱う公共性の高いものをはじめ、[携帯電話](http://ja.wikipedia.org/wiki/%E6%90%BA%E5%B8%AF%E9%9B%BB%E8%A9%B1)、[デジタルオーディオプレーヤー](http://ja.wikipedia.org/wiki/%E3%83%87%E3%82%B8%E3%82%BF%E3%83%AB%E3%82%AA%E3%83%BC%E3%83%87%E3%82%A3%E3%82%AA%E3%83%97%E3%83%AC%E3%83%BC%E3%83%A4%E3%83%BC)、[携帯ゲーム機](http://ja.wikipedia.org/wiki/%E6%90%BA%E5%B8%AF%E3%82%B2%E3%83%BC%E3%83%A0%E6%A9%9F)、[コピー機](http://ja.wikipedia.org/wiki/%E8%A4%87%E5%86%99%E6%A9%9F)、[ファックス](http://ja.wikipedia.org/wiki/%E3%83%95%E3%82%A1%E3%82%AF%E3%82%B7%E3%83%9F%E3%83%AA)、[カーナビ](http://ja.wikipedia.org/wiki/%E3%82%AB%E3%83%BC%E3%83%8A%E3%83%93%E3%82%B2%E3%83%BC%E3%82%B7%E3%83%A7%E3%83%B3)など、デジタル情報機器を中心に多方面で使用されている。

[](http://ja.wikipedia.org/wiki/%E3%83%95%E3%82%A1%E3%82%A4%E3%83%AB:Nintendo_DS_Lite_Ice_Blue.jpg)[](http://ja.wikipedia.org/wiki/%E3%83%95%E3%82%A1%E3%82%A4%E3%83%AB:AONAMI_Line_in_Nagoya_Station_of_Ticket_machine.jpg)[](http://ja.wikipedia.org/wiki/%E3%83%95%E3%82%A1%E3%82%A4%E3%83%AB:Iphone_4G-2.jpg)スマートフォン　　　　　　　　　　　　カーナビゲーション

自動券売機　　　　　　　　　　　　　　　携帯ゲーム機

図１：タッチパネルが使用されている電子機器類

**３．利点と欠点**

◎利点◎

・画面上の操作説明に入力動作が対応するため、直感的に分かりやすく、操作が簡単になる。

・入力装置と表示装置が一体化されるため、装置全体の小型化が期待できる。

・物理的なボタンがなくとも、ソフトウェア次第で多彩な操作性を実現できる。

この装置は使用者が画面の操作手順を見ながら操作できるために、機械操作を誰にでも扱いやすくする事ができるという点で優れている。画面表示を随時切り替えることで、1つの画面上で多様な操作指示に使用できるので、装置全体の小型化が期待できる。

×欠点×

・汚れで画面が見えにくくなることがある。

・一般にキーボードほどの素早く正確な入力は不得意である。

・押しボタンと異なりクリック感がないので、入力動作がぎこちなくなる場合がある。このため、アニメーションによる視覚効果や、音を出したり、振動を与えるなどのフィードバックが工夫される。

**４．タッチパネルの仕組み**

タッチパネルの動作で最低限要求されることとして、**表示が見える状態で、どの位置を、タッチした**かを判別することが必要とな[](http://www.technoveins.co.jp/technical/touchpanel/images/modulelayer.jpg)ります。

タッチパネルは表示装置をタッチしているように見えますが、タッチパネルとLCD（液晶ディスプレイ）などの表示装置は別々の部品であり、2つのモジュール部品を組み合わせ（張り合わせ）、1つのケースに収められて使用します。

図２：タッチパネルの仕組み

タッチした位置を検出する原理に、電気的にタッチを検出するものと、電気を用いないでタッチを検出するものがあります。

・電気的なスイッチやスイッチに相当する部分を他の手段に置き換えたもの

・電気的スイッチを用いずに、物理的な振動や光学的にタッチを検出するもの

・その他、空間的に画像認識や測定を行うもの

などが考えられます。

**第２章　接続機器**

**１．今回主に使用する機器とソフトウェア**

1. タッチパネル

三菱製：GT2508-VTBA

1. 制御用機器

三菱製：シーケンサ（Qシリーズ）　Q03UDVCPU

※主に、入力信号を、あらかじめ組んだプログラムで処理して、意図した通りに出力信号を出す。自動化装置を動かす専用コンピュータです。

1. 画面作成ソフト

三菱製：GT Designer3

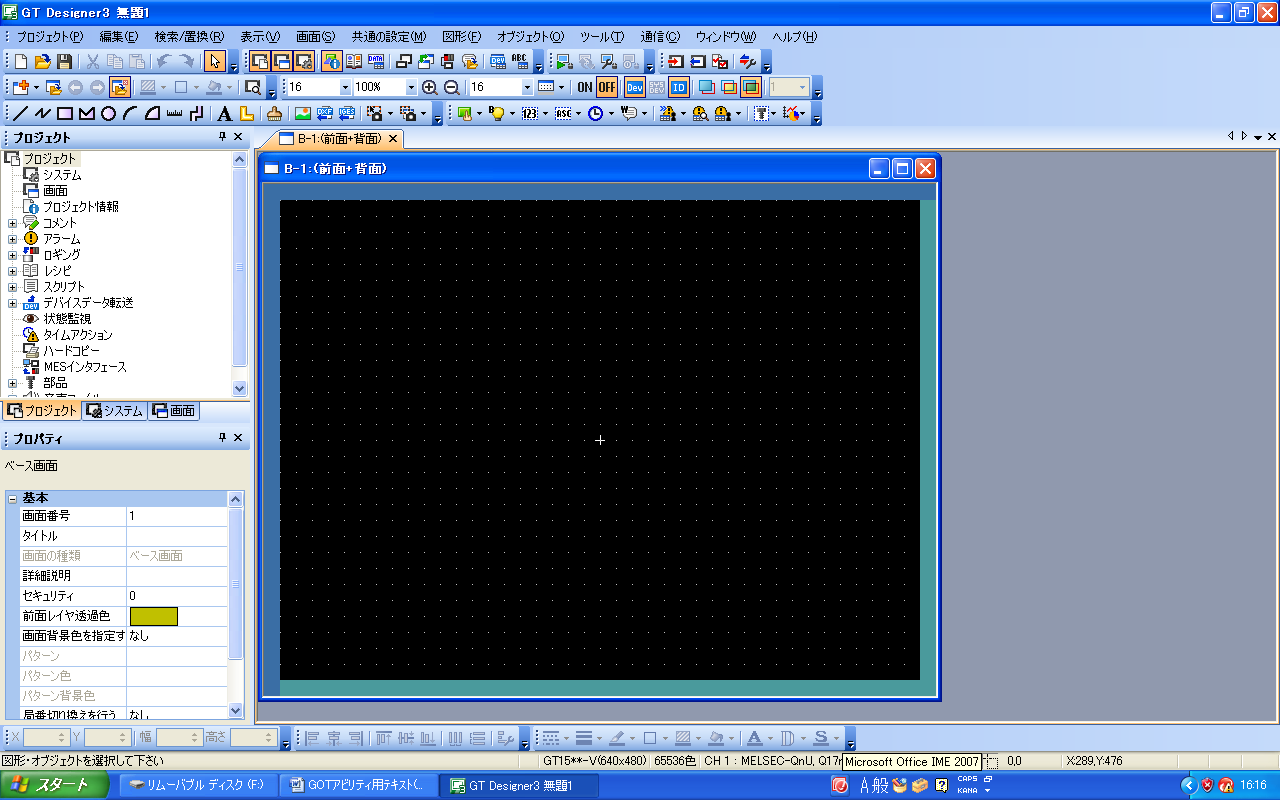
※タッチパネルに表示する画面を作成するソフトです。

1. 制御プログラミングソフト

三菱製：GX Works2

※ラダー図を用いてシーケンスプログラムを作成するソフトです。

**第３章　タッチパネル画面作成**

* **画面説明**

メニューバー

GOTで設定できることを表示している部分

ステータスバー

設定及び座標を確認できる部分

プロパティシート

画面や部品の設定を変更できる部分

描画スペース

GOTに表示させる画面を描く部分

ワークスペース

ベース画面や各種設定画面を開く部分

ツールバー

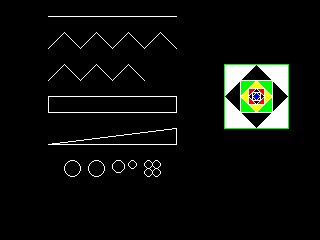
画面を描く際に特に使用する部分

※ ワークスペースやプロパティシートを削除してしまった時には…

　メニューバーの「表示」→「ワークスペース」、「プロパティシート」を選びチェックを付けること。

**２．画面作成**

1. **図形描画**

****

［オブジェクト情報］

の各種アイコンを選択して描画して下さい

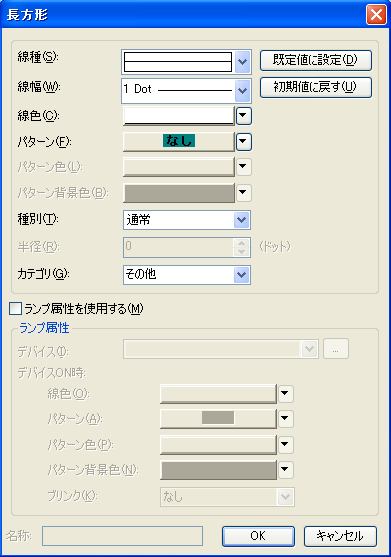
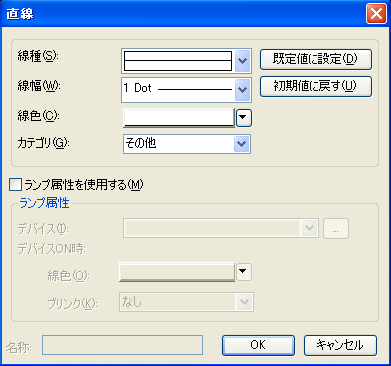
左より『直線』『連続直線（描キー終える時はダブルクリック）』『長方形』『多角形（描キー終える時はダブルクリック）』『円』『円弧』『扇』『目盛』『配管』

※ 設定を変更する場合は…

Ⅰ．描いたオブジェクトを『ダブルクリック』より変更する。

Ⅱ．描いたオブジェクトを『右クリック』→『設定ダイアログを開く』より変更する。

Ⅲ．描いたオブジェクト選択後、『プロパティシート』より変更する。



★**ここで確認**

**グリッド色**

**グリッド間隔**

**ズーム**

**移動量**



**★塗りつぶしを行う**

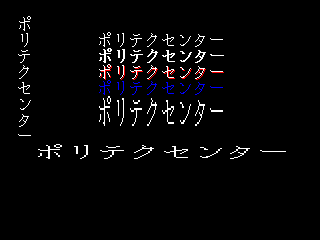
（ペイント）アイコンをクリック後、塗りつぶしたい箇所でクリックするとペイントの設定画面が出てくる。設定完了後、『OK 』ボタンをクリックする。



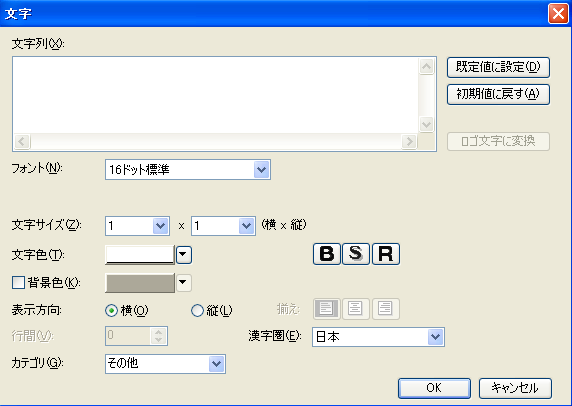
* 境界色は描いた図形の枠線の色に合わせること。
* パターンは塗りつぶし方の設定
* パターン色・背景色は塗りつぶしたい色の設定

※ペイントで作成した色についても１つのオブジェクトとなる為、取り扱いには注意が必要。

1. **文字表示**



※ツールバーの『文字』を選択し、画面上に表示する。



文字飾りを選択

左から『太字』『影』『彫刻』

表示方向を選択

背景色を選択

文字色を選択

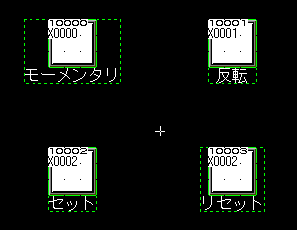
文字サイズを選択

フォントを選択

文字列を入力

1. **タッチスイッチ（ビットスイッチ）**

タッチするとビットデバイスのON/OFFをします。

****

[オブジェクト情報]

**［ビットデバイス］　　　　　　　　　　　　　[動作設定]**

**X0000　　ID10000 ビットモーメンタリ**

**X0001　　ID10001 ビット反転**

**X0002　　ID10002 ビットセット**

**X0002　　ID10003 ビットリセット**

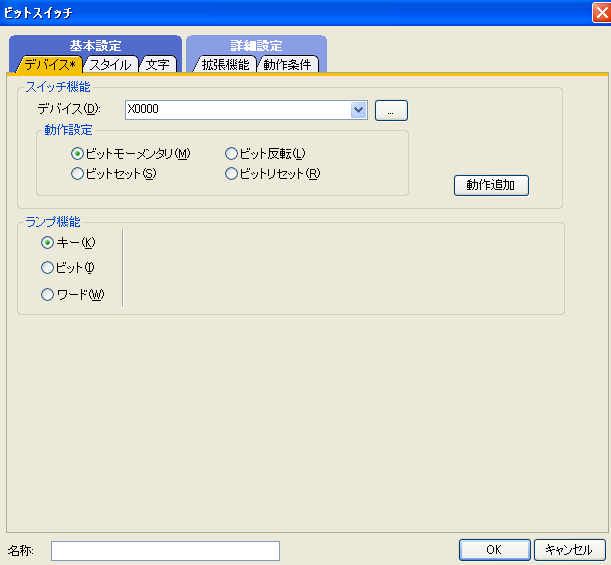
画面上にビットスイッチ４つを配置し、各種設定をおこないます。

【操作】

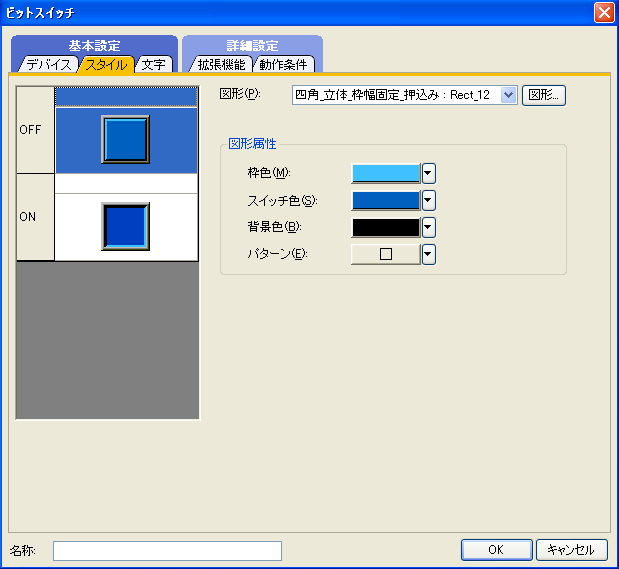
1. ツールバーの『スイッチ』→『ビットスイッチ』を選択し、画面上に表示する。その後設置したスイッチをダブルクリックしてスイッチ設定を行う。



1. 配置したスイッチをダブルクリックします。



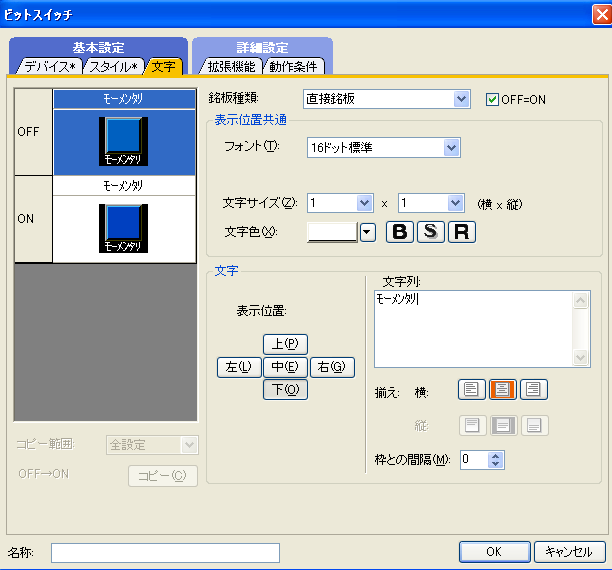
1. デバイスに「X0」と入力します。
2. 動作設定は「ビットモーメンタリ」を選択します。
3. 「スタイル」タブをクリックします。
4. 「図形」ボタンをクリックします。



1. 基本図形の一覧から、「１３２四角\_立体\_枠幅固定\_色変え」を選択し、「10 Rect\_10」をクリックし、「OK」ボタンをクリックします。

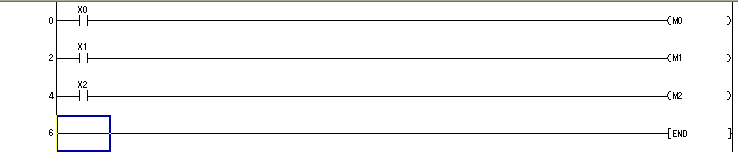


1. 文字を表示させるには、まず表示位置の指定から行います。表示位置の「下」をクリックし、文字列に「ﾓｰﾒﾝﾀﾘ」と入力します。



1. 設定ができたら「OK」ボタンをクリックします。
2. 同様に残りの３つのボタンを作成しましょう。

【動作確認】

以下のラダーを入力し、モニタで動作確認をしましょう。

※スイッチを配置するとどのデバイスが割り付けられているかで確認できる。

Devの方は割り付けたデバイス名及び番号

IDの方はソフトが作った順に割り振る番号

※ランプ機能においてボタンを押した時の表示方法が変更可能！

* キー　：　GOT画面において押している間のみON表示
* ビット：　 GOT画面において《デバイス》タブで設定したビットに応じて

ON/OFF表示がなされる。

* ワード：　ランプ（ワード）にて説明
* **タッチスイッチの動作設定**

　　ビットモーメンタリ　：　タッチ中のみビットをON

　　ビット反転　　　　　：　タッチするごとにビットをON⇔OFF

　　ビットセット　　　　 ：　タッチしたときにビットを強制ON

　　ビットリセット　　　 ：　タッチしたときにビットを強制OFF

* タッチする範囲の変更をする場合には、スイッチのオブジェクトを右クリックし『タッチエリア・枠領域を編集する』を選択する。変更後は必ず、『タッチエリアを表示する』を選択すること。
* **タッチスイッチは重ねることはできないので注意！！**

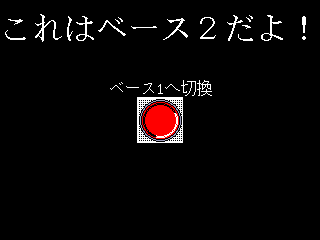
1. **画面切り換え**

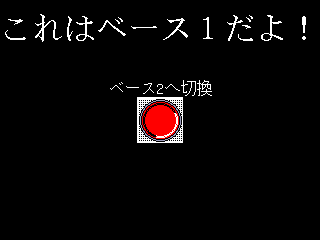
画面の種類

|  |  |  |
| --- | --- | --- |
| ベース画面 | | GOTの基本画面 |
| ウィンドウ画面 | オーバーラップウィンドウ | ベース画面上にポップアップするウィンドウ |
| スーパーインボーズウィンドウ | ベース画面の全面に合成され、１つの画面として表示するウィンドウ |
| キーウィンドウ | 数値などを入力する場合に、ベース画面上にポップアップするウィンドウ |

※GOTではベース画面は最大４０９６画面、ウィンドウ画面は最大1024画面まで表示できます。

**４．１タッチスイッチ（画面切り換え）**

画面切り換えスイッチは、タッチするとベース画面とベース画面の切り換えやベース画面とウィンドウ画面の表示を切り換えます。



ベース2：[オブジェクト情報]

文字

サイズ：2×2

タッチスイッチ

動作：画面切り換え

切り換え先：固定値（１）

ベース1：[オブジェクト情報]

文字

サイズ：2×2

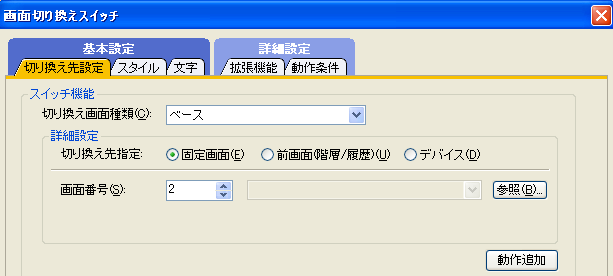
タッチスイッチ

動作：画面切り換え

切り換え先：固定値（2）

【操作】

1. ツールバーの『スイッチ』から『画面切り換えスイッチ』を選択し、画面上に配置する。
2. 配置した画面切り換えスイッチをダブルクリックします。
3. 切り替え画面種類を「ベース」、切り替え先指定「固定画面」、画面番号「２」に設定します。



1. 「スタイル」タブで適宜スタイルを変更します。
2. 「文字」タブをクリックし、ボタンの上に「ベース２へ切り替え」と表示させるように設定します。
3. その他、文字列等を追加し画面の作成をしましょう。
4. 同様にベース画面２も作成します。ベース画面を追加するには、メニューバーの「画面」→「新規作成」→「ベース画面」をクリックします。

切り替え先指定

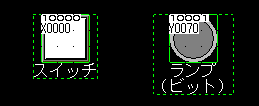
固定値：　選択した番号の画面に切り換える。

前画面：　前回表示した画面に切り換える。

デバイス：デバイスに画面番号を設定し、デバイスの動作に応じて画面を切り替える。

1. **ランプ（ビットランプ）**

ビットデバイスのON/OFF によりランプを点灯/ 消灯させます。



［オブジェクト情報］

ランプ(ビット)　デバイス：Y0070

スイッチ（モーメンタリ）　デバイス：X0000

【操作】

①ツールバーから『ランプ』→『ビットランプ』を選択し、画面上に配置する。

②配置したビットランプをダブルクリックする。

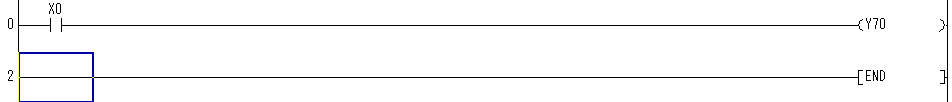
③ランプの種類を「ビット」、デバイスに「Y70」を設定します。また、図形は適宜選択します。



　　　　④同様にビットスイッチも作成しましょう。

【動作確認】

以下のラダーを入力し、動作確認をしましょう。



練習①

[オブジェクト情報]

・ビットスイッチ（モーメンタリ）　X0　（スタート）

・ビットスイッチ（モーメンタリ）　X1　（ストップ）

・ビットランプ　Y70

上記の部品を配置しましょう。

スタートボタンがタッチされるとビットランプが点灯し続け、ストップボタンがタッチされるとランプが消灯するようなプログラムを作成しましょう。

練習②

[オブジェクト情報]

・ビットスイッチ（モーメンタリ）　X0　（スタート）

・ビットランプ　Y70

上記の部品を配置しましょう。

スタートボタンがタッチされた3秒後にランプが消灯するプログラムを作成しましょう。

練習③

[オブジェクト情報]

・ビットスイッチ（モーメンタリ）　X0　（スタート）

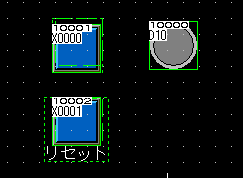
・ビットランプ　Y70

上記の部品を配置しましょう。

スタートボタンがタッチされたら1秒ごとにランプが点灯・消灯を繰り返すプログラムを作成しましょう。

1. **ランプ（ワードランプ）**

ワードデバイスの値によってランプの点灯色を変更させます。

****

［オブジェクト情報］

ランプ(ワード)　デバイス：D10(BIN16)

ビットスイッチ（モーメンタリ）　デバイス：X0000

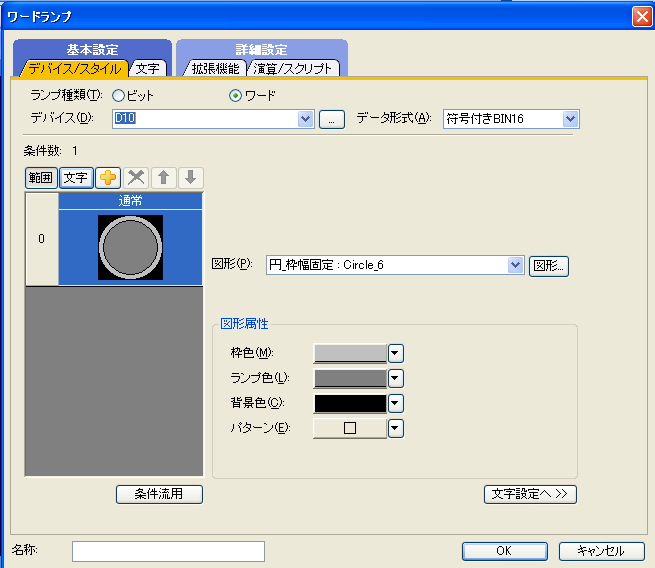
ビットスイッチ（モーメンタリ）　デバイス：X0001　（リセット）

【操作】

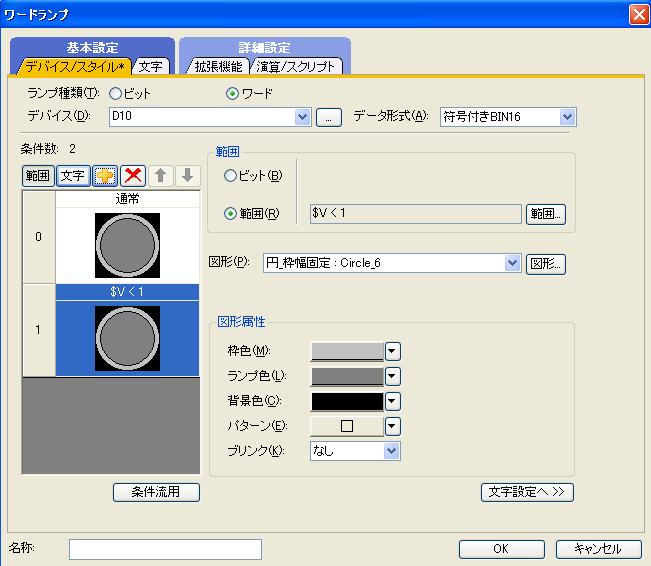
①ツールバーから『ランプ』→『ワードランプ』を選択し、画面上に配置する。

②配置したワードランプをダブルクリックする。

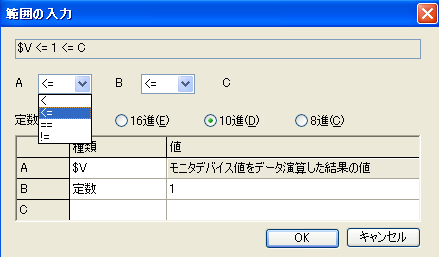
③デバイスに「D10」を設定。ランプ点灯の条件を設定するために、「＋」ボタンをクリックします。



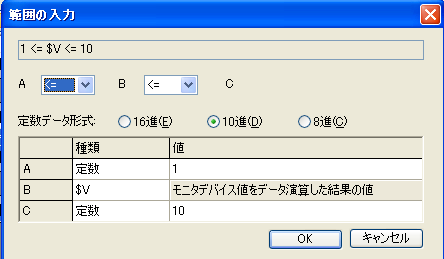
1. 「範囲」ボタンをクリックします。



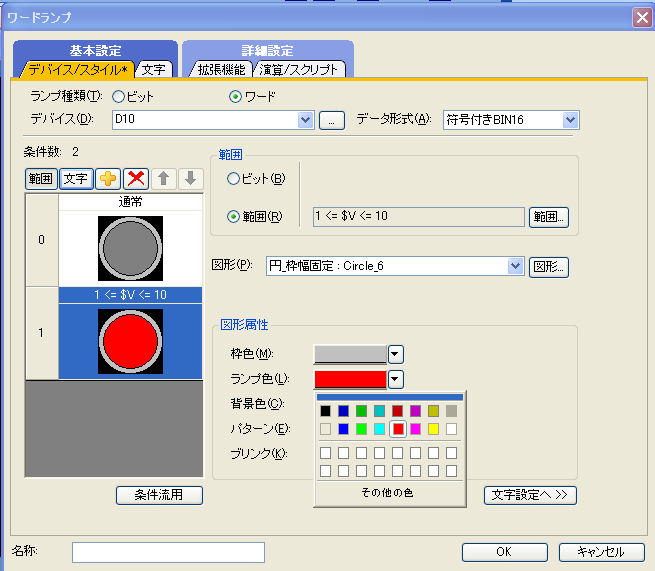
1. 比較演算子を「<=」に設定します。



1. Aの種類を「定数」値を「１」に設定します。Bの種類を「＄V」に設定します。Cの種類を「定数」値を「１０」に設定し、「OK」ボタンをクリックします。



1. ランプ色を「赤」に設定します。



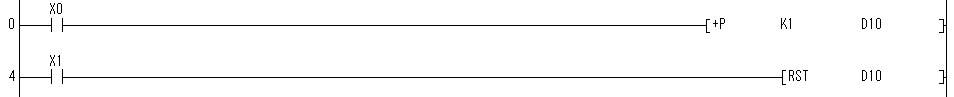
1. 同様に、「＋」ボタンをクリックし、下記のような条件を設定しましょう。

条件２：　11 ＜＝ $V ＜＝ 20　　 （ランプ色：青）

条件３：　21 ＜＝ $V ＜＝ 30　　 （ランプ色：黄）

【動作確認】

以下のラダーを入力し、動作確認をしましょう。

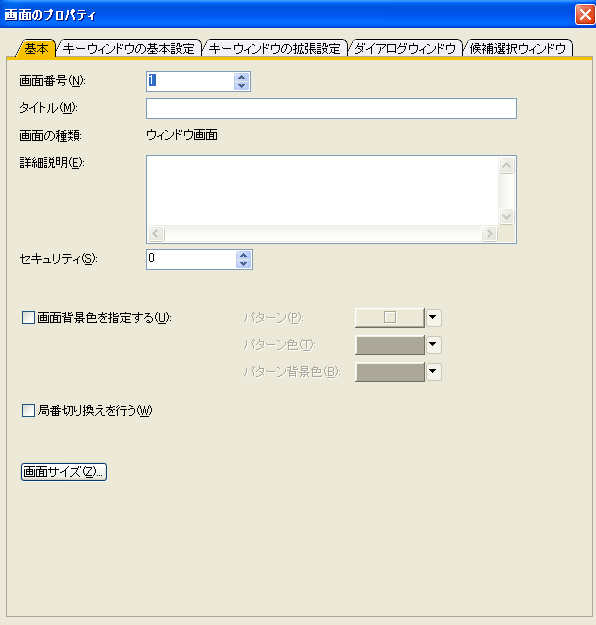


1. **ウィンドウ画面の表示／非表示**

　上記の画面でビットスイッチ（X０）のタッチ回数が30回を超えたときに、メッセージを表示するウィンドウ画面を作成しましょう。

【ウィンドウ画面の作成】

* + 1. メニューバーの「画面」→「新規作成」→「ウィンドウ画面」をクリックします。
    2. 画面番号を確認し、「OK」ボタンをクリックします。



③　ウィンドウ画面に「30回を超えました」という文字列と、ビットスイッチ（モーメンタリ：X2）を1つ配置しましょう。

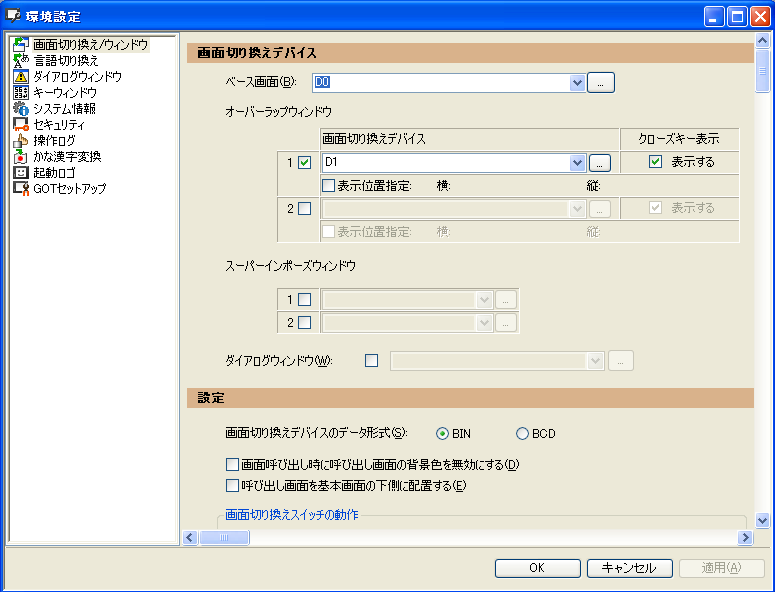


【ウィンドウ画面の表示】

　ウィンドウ画面を表示させるには、画面切り換えスイッチの他にデバイスを使用する方法があります。

1. 設定の確認

メニューバーの「共通の設定」→「GOT環境設定」→「画面切り換え／ウィンドウ」をクリックします。オーバーラップウィンドウの画面切り換えデバイスが「D1」になっていることを確認します。（プロジェクト作成時に自分で設定する項目）



1. デバイスを使用してウィンドウ画面を表示するには、上記のオーバーラップウィンドウの画面切り換えデバイスに指定したD1に表示させたいウィンドウ画面番号を設定します。

（６）ワードランプで書いたラダーに下記を追加しましょう。



プログラム内容：D10が30を超えたら、D1に１（ウィンドウ画面番号）を設定

1. 表示したウィンドウ画面を非表示にする場合には、ウィンドウ画面の右上の×ボタンをタッチするか、画面切り換えデバイスD1に０をセットします。

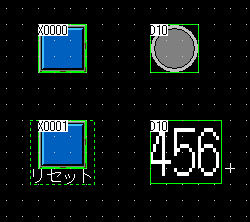


プログラム内容：X2タッチスイッチがタッチされたらD1に０（ウィンドウ非表示）を設定

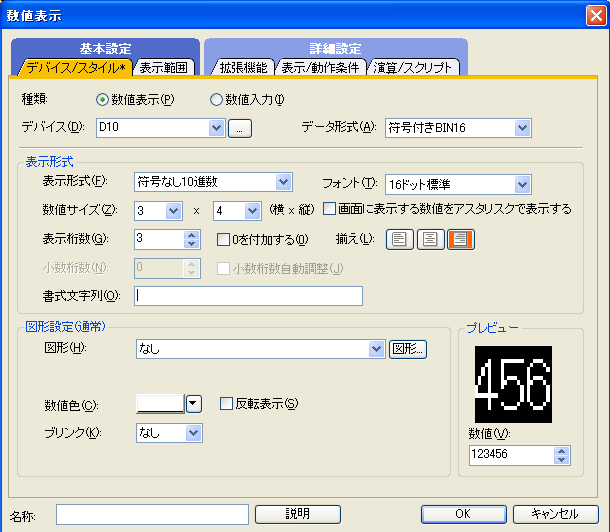
**（８）　数値表示**

数値表示はデバイスの値を各表示形式で画面に表示します。

上記（６）ワードランプで作成した画面に数値表示部品を配置し、動作を確認しましょう。



1. ツールバーから「数値表示／入力」→「数値表示」をクリックして、画面上に配置します。
2. デバイスに「D10」、表示形式「符号なし10進数」、表示桁数「３」に設定し、「OK」をクリックします。



【動作確認】

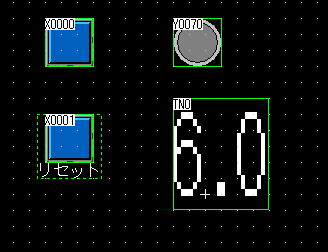
画面をGOTに転送して動作を確認しましょう。ビットスイッチがタッチされた回数が表示されます。

**８－１　数値表示（タイマ現在値表示）**

タイマデバイスは100msecタイマのためミリ秒で表示されます。（1秒　→　10ミリ秒）

ここでは、タイマの現在値が見やすいように秒で表示する方法について確認します。

X0をタッチすると3秒カウントし、カウント後ランプを点灯させます。その時、T0の現在値を0.0～3.0　と表示させます。



[オブジェクト情報]

ビットスイッチ(モーメンタリ)　デバイス：X0

ビットスイッチ（モーメンタリ）　デバイス：X1

ビットランプ　デバイス：Y70

数値表示　デバイス：TN0

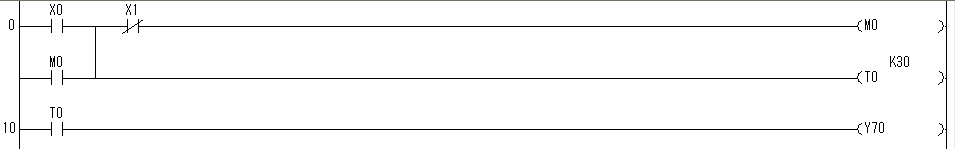
【操作】

1. 配置した数値表示部品をダブルクリックします。
2. デバイスを「TN0」、表示形式「実数」、表示桁数「３」、少数桁数「１」、少数桁数自動調整にチェックを入れ、「OK」ボタンをクリックします。



【動作確認】

以下のラダーを入力し、動作確認をしましょう。



タイマ、カウンタ、積算タイマのデバイスについて

GOTでタイマ、カウンタ、積算タイマを扱う場合は以下のデバイスとなります。

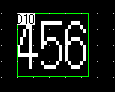
タイマ：現在値　**TN**

カウンタ：現在値　**CN**

積算タイマ：現在値　**SN**

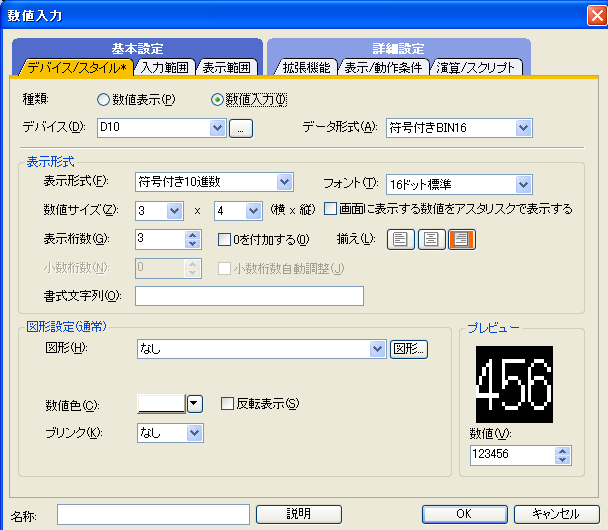
1. **数値入力**

タッチパネルの画面上からキーウィンドウを利用して数値入力することができます。



【操作】

1. ツールバーから「数値入力」を選択し、画面上に配置します。
2. 配置した数値入力をダブルクリックします。
3. デバイス「D10」、表示形式「符号付き10進数」、表示桁数「３」に設定し、「OK」ボタンをクリックします。



【動作確認】

画面を転送し、数値表示をタッチするとキーウィンドウが表示され数値入力ができることを確認しましょう。

**９－１　数値入力（タイマ設定値セット用）**

表示形式の「小数桁数の自動調整」にチェックをいれると、設定した小数桁数に応じて自動的に整数⇔小数点付データ（実数）に変換します。

3.0とGOTに入力

GOT

シーケンサ



小数点桁数が１の時は10倍

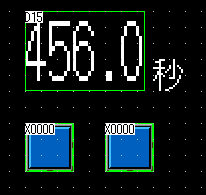
書込み時：3.0×10

30

3.0

読込時：30÷10

数値入力から入力した数値をタイマデバイスの設定値にする場合、GOTでの数値は秒で設定し、シーケンサへの書き込みはミリ秒になるように設定します。（３秒　→　３０ミリ秒）



[オブジェクト情報]

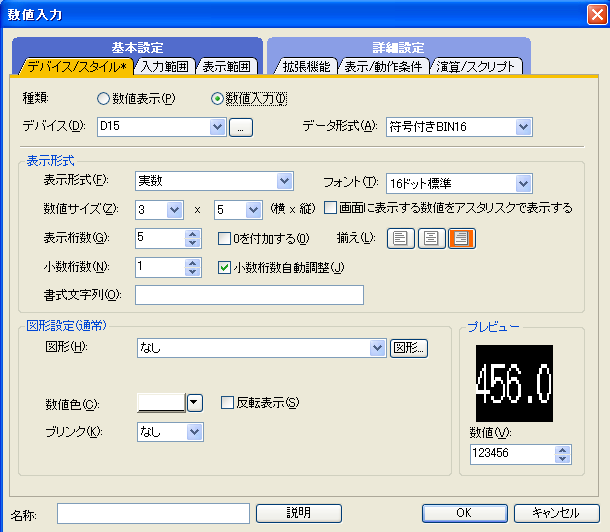
数値入力　デバイス：D15

ビットスイッチ（ビットSET）　デバイス：X0

ビットスイッチ（ビットRST）　デバイス：X0

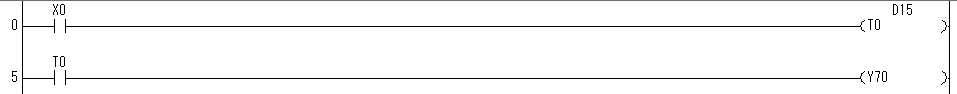
【操作】

1. 上記で作成した数値入力をダブルクリックします。
2. デバイス「D15」、表示形式「実数」、表示桁数「5」、小数桁数「１」、小数桁数自動調整にチェックをいれます。



【動作確認】

以下のラダーを入力し、モニタ機能でD10の値が１０倍されていることを確認しましょう。



**（９）　コメント**

コメントとはGTDesignerの中でユーザが登録した文字列のことです。

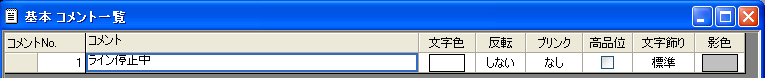
コメントは基本コメントまたはコメントグループに登録することで、複数のオブジェクト機能で表示できます。

**９－１　コメントの登録**

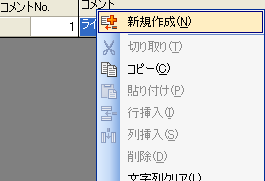
1. 基本コメントを作成するにはプロジェクトウィンドウ→コメント→基本コメントをクリックします。



1. コメントに「ライン停止中」と入力します。



1. 次の行は右クリック→「新規作成」をクリックします。



1. コメントに「ライン運転中」と入力します。

コメントの最大登録数

・基本コメント：３２７６７個

・コメントグループ：最大255個のコメントグループを登録可。１つのコメントグループでは、

　最大３２７６７行×10列

１コメントで登録できる最大文字数

　最大５１２文字

**９－２　コメント表示**

コメントを表示するには2種類の方法があります。

①コメント表示（ビット）

　　ビットデバイスのON/OFFに対応づけて、2種類のコメントを切り替えて表示させます。

　　　　　　　Ｘ0=ON　　　　　　　　　　　　　　　　　　X0=OFF

ライン運転中

ライン停止中

X0がONの時は、「ライン停止中」のコメントを表示。X0がOFFの時は、「ライン運転中」のコメントを表示します。

②コメント表示（ワード）

　　ワードデバイスの値に対応づけて、数種類のコメントを切り替えて表示させます。

　　　　　D100=1　　　　　　　　　　D100=2　　　　　　　　　　D100=3　　　　　　　　　　　D100=4

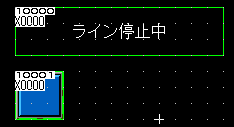
異常停止中

一時停止中

運転中

停止中

**ビットデバイスによるコメント表示**



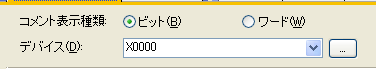
[オブジェクト情報]

ビットコメント　デバイス：X0

ビットスイッチ（ビット反転）　デバイス：X0

【操作】

1. ツールバーから「コメント表示」→「ビットコメント」を選択し、画面上に配置します。
2. 配置したビットコメント部品をダブルクリックします。
3. デバイスに「X0」を設定します。



1. OFF=ONのチェックをはずします。「OFF」をクリックし、コメントNOを「１」にセットします。その後、「ON」をクリックし、コメントNOを「２」にセットし、「OK」をクリックします。



1. ビットスイッチ（ビット反転）を配置し、デバイスを「X0」に設定します。

【動作確認】

X0がONのときとOFFのときのコメント表示を確認しましょう。

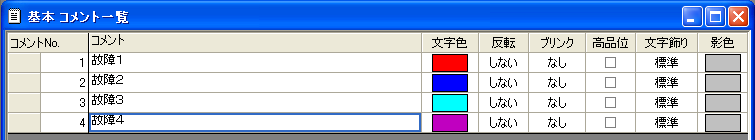
**ワードデバイスによるコメントの表示の設定**

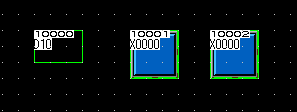
ワードデバイスによるコメントでは、コメントを表示する範囲を決めることができます。

1. 「共通の設定」→「コメント」→「コメント」メニューを選択
2. 「基本コメント」を選択し、ダブルクリック
3. 「基本コメント一覧」ダイアログボックスの「コメント」欄に下記のコメントを入力

|  |  |  |
| --- | --- | --- |
| コメントNO | コメント | 文字色 |
| １ | 故障１ | 赤 |
| ２ | 故障２ | 青 |
| ３ | 故障３ | 水色 |
| ４ | 故障４ | 紫 |

※次の行にコメントを入力する場合は、右クリック新規作成で枠が表示されます。





[オブジェクト情報]

ワードコメント　デバイス：D10

ビットスイッチ（ビットSET）　デバイス：X0

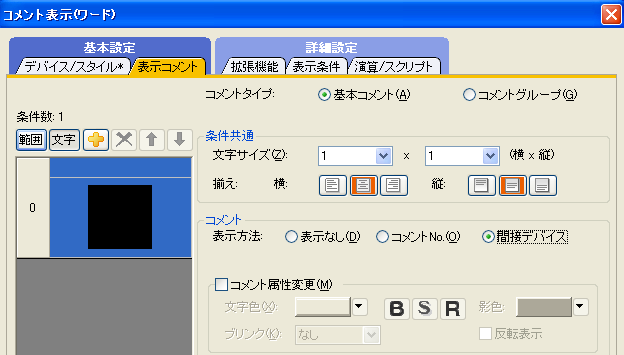
ビットスイッチ（ビットRST）　デバイス：X0

【操作】

1. ツールバーの「コメント表示」→「ワードコメント」をクリックし、配置します
2. 配置したワードコメントをダブルクリックします
3. デバイスをD10に設定

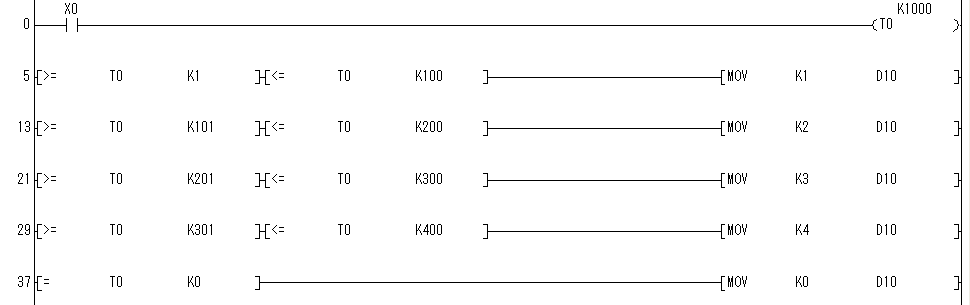


1. 「基本コメント」タブをクリックし、間接デバイスにチェックし、「OK」をクリックします。

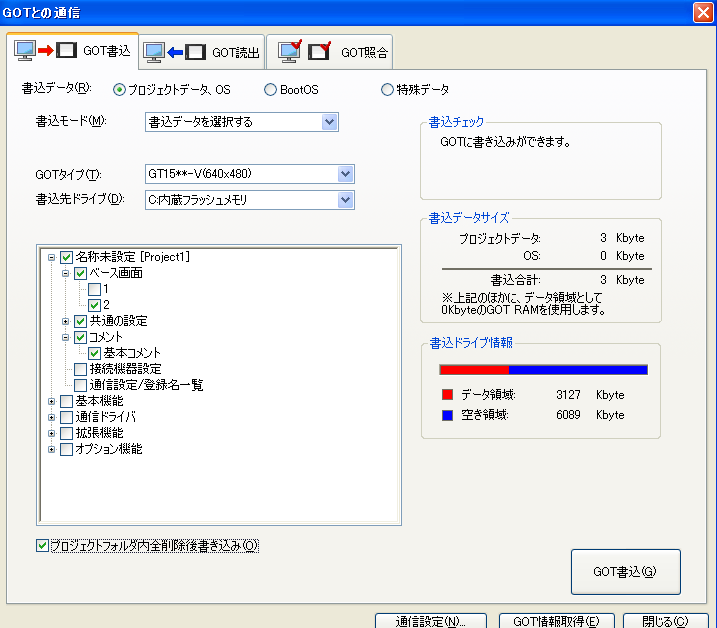


【動作確認】

以下のラダーを入力し、D10の値が1～100のとき、コメントNo1が表示され、101～200のときコメントNo2が表示、201～300のときコメントNo3が表示され、301～400のときコメントNo4が表示されることを確認しましょう。



※画面をダウンロードする際、コメントにもチェックをいれます。

****

**第4章　データの転送**

**（１）データの種類**

|  |  |  |
| --- | --- | --- |
| データの種類 | | 概要 |
| BootOS | | GOTのハードウェア制御やパソコンとGOTの通信時に必要なプログラム。工場出荷時にインストールされているため、通常インストールの必要なし |
| OS | | GOTのモニタ機能、OSや画面データのインストール、OSや画面データの削除 |
|  | 基本機能（必須） | 各インタフェースの制御、OSや画面データのインストール、OSやプロジェクトデータの削除などGOTを動作させるプログラム |
| 通信ドライバ（必須） | PC通信ドライバは、GOTとシーケンサ間の通信を行うドライバ |
| 拡張機能 | 拡張機能を使用する場合にインストールする |
| オプション機能 | オプション機能ボードや多色表示ボードの装着により使用できる機能を使用する場合にインストール |
| プロジェクトデータ（必須） | | 画面データ、コメント、共通設定など、ユーザが作成したモニタ画面用のデータ |

**（２）OSのインストール**

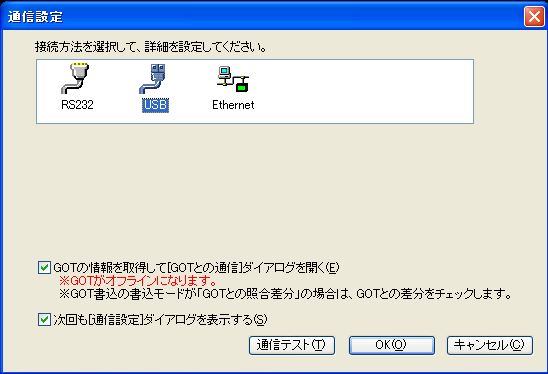
GOTにはモニタするためのOSや通信するためのドライバが入っていません。そのため、初めてモニタをする前にOSのインストールをする必要があります。ただし、OSのバージョンアップや通信方法の変更がある場合は、この操作が必要です。

【操作】

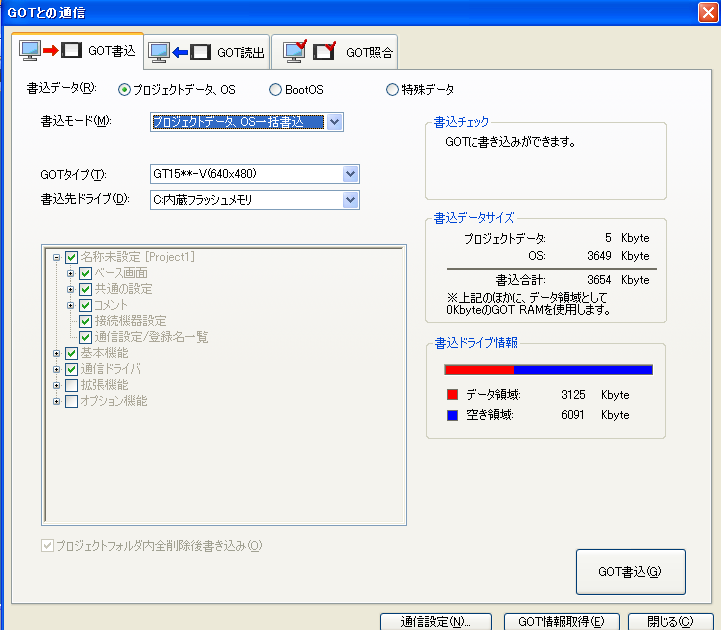
①通信→GOTへの書き込み　をクリック



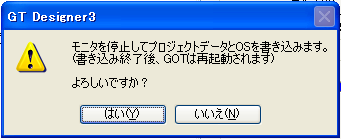
1. USB」を選択し、「OK」をクリック



1. 書込データ「プロジェクトデータ、OS」、書込モード「プロジェクトデータ、OS一括書込」を選択し、「GOT書込」をクリック。



1. 「はい」をクリックすると、プロジェクトデータとOSのインストールが始まります。



**第5章　ラダープログラム作成**

**１．デバイス一覧**

・デバイスとは、シーケンサのCPU内にあるプログラム用のイメージ素子のことであり、プログラムを構成する要素といえます。

**デバイス　＝　デバイス記号　＋　デバイス番号**　で構成されています。

デバイス記号・・・デバイスの種類（入力、出力、etc）を表すもので、アルファベット１文字で表します。

デバイス番号・・・各入力や出力に与えられる数値です。簡単にいえば、家の住所みたいなものです。

|  |  |  |
| --- | --- | --- |
| デバイスの種類 | | 説明 |
| X | 入力 | 押しボタン、デジタルスイッチ等の外部機器によりシーケンサに指令やデータを与えるためのものです。 |
| Y | 出力 | ランプ、デジタル表示器等にプログラムの制御結果として外部へ出力するものです。 |
| M | 内部リレー | 外部へ直接出力することのできないシーケンサ内部の補助リレーです。 |
| T | タイマ | 加算式のタイマで、時間を計測したりする際に使用し、通常時は100msタイマとして使用されます。 |
| C | カウンタ | 加算式で、個数を数えたりする際に使用し、設定数以上はカウントしません。 |
| D | データレジスタ | シーケンサ内のデータを格納するメモリです。 |
| K | 10進定数 | タイマ・カウンタの設定値、基本・応用命令の数値指定に使用します。 |

**２．命令一覧**

1. 基本命令

|  |  |  |
| --- | --- | --- |
| 命令の種類 | | 説明 |
|  | a接点 | 動作させたときにONの状態になる接点（通常時はOFFの状態である） |
|  | b接点 | 動作させたときにOFFの状態になる接点（通常時はONの状態である） |
|  | アウト  （コイル出力） | ON又はOFFの結果を出力させる（使えるデバイス記号はY、M、T、C） |
|  | アウト  （コイル出力以外） | 基本、コイル出力できないデータ値を扱う際に使用する。 |

1. 命令語
2. OUT命令

・OUT命令は入力条件が“ON”すると指定デバイスを“ON”にし、入力条件が“OFF”になった時点で、指定デバイスも“OFF”されます。

プログラム例 タイムチャート

1. タイマの計測

・タイマの接点はコイル励磁後、設定時間だけ遅れて動作します。（オンディレイタイマ）

プログラム例 タイムチャート

1. カウンタの計測

・カウンタは入力信号の立ち上がりでカウントします。設定数カウントアップ後は、入力信号があってもカウントしません。

・一度カウントアップしますと、RST命令が実行されるまで接点の状態、現在値（カウンタのカウント値）は変化しません。

入力条件が“ON”のとき、強制的に指定デバイスを“OFF”にする命令。

プログラム例 タイムチャート

1. 転送命令（データの転送）

★MOV(P)命令

MOV命令は常に入力条件がONであれば転送し続けるが、MOVP命令は入力条件がONになった瞬間のみの値を転送する。

・入力条件が“ON”すると、①で指定されたデバイスのデータを、2進数のままで②で指定したデバイスに転送する命令です。

プログラム例（数値表示を使ってカウンタの現在値を表示させる）

※その他、転送命令には“BIN”“BCD”命令がありますが、今回はタッチパネル内部で自動的に変換されるため使用する必要なし。

“SM400”は常時ONの特殊リレーです。

※カウンタの現在値を“D2”とする。

1. 四則演算命令

★加算　＋(P)

・入力条件が“ON”すると、①で指定したデバイスの内容と、②で指定したデバイスの内容とを加算し、結果を③のデバイスに格納します。

★減算　－(P)

・入力条件が“ON”すると、①で指定したデバイスの内容と、②で指定したデバイスの内容とを減算し、結果を③のデバイスに格納します。

★乗算　＊(P)

・入力条件が“ON”すると、①で指定したデバイスの内容と、②で指定したデバイスの内容とを乗算し、結果を③のデバイスに格納します。

★除算　／(P)

・入力条件が“ON”すると、①で指定したデバイスの内容と、②で指定したデバイスの内容とを除算し、結果を③のデバイスに格納します。また、結果は商のみが格納されるため、余りは③の次のデバイス（③がD10ならばD11）に格納されます。

1. 比較演算命令

・比較命令は2つのデータを比較して、条件が成立すると導通します。故に、条件成立で導通ですから、1個のa接点とみなすことができます。

★　**＝**

・①と②が一致したとき導通します。

★　**＜**

・①が②より小さいとき導通します。

★　**＞**

・①が②より大きいとき導通します。

★　**＜＝**

・①が②より小さいときと一致したとき導通します。

★　**＞＝**

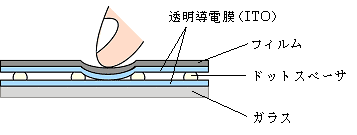
・①が②より大きいときと一致したとき導通します。

★　**＜＞**

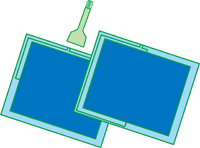
・①と②が不一致のとき導通します。

**参考資料**

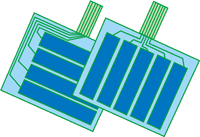
**抵抗膜方式タッチパネルの基本構造**



抵抗膜方式アナログタッチパネルは、透明導電膜(ITO)が設けられた素材(主にフィルムやガラス)を透明導電膜間が向い合う方向に貼り合わせ、指或いはペンで押した時透明導電膜同士が接触することによってタッチパネル入力が行われます。 この上下導電膜間には、ドットスペーサーと呼ばれる絶縁体があり、未入力時の上下導電膜のショートを防止しています。抵抗膜方式タッチパネルは、大きく分けてアナログタイプとマトリックスタイプに分けられます。

**アナログタイプ**

上部、下部共に一面に透明導電膜が形成され、一方でX座標回路、もう一方でY座標回路を構成します。タッチした位置の抵抗の比をアナログ的に検出します。

**マトリックスタイプ**

上部と下部にそれぞれに形成した短冊状の透明導電膜を直交するように構成します。タッチした時のX座標、Y座標により位置を検出します。

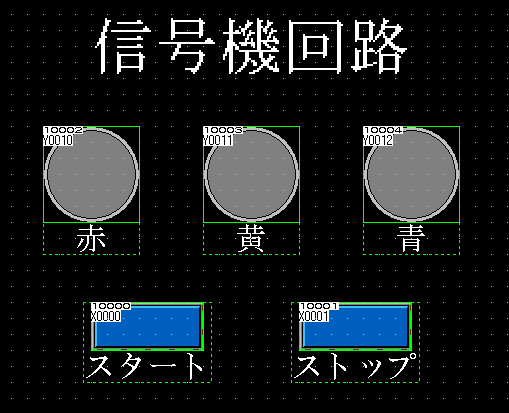
参考URL　 http://www.dmccoltd.com/structure/

課題１　信号機回路

・タッチパネル上に赤、黄、青のランプを配置し、スタートボタンを押すと、赤が5秒点灯→黄が1秒点灯→青が3秒点灯をストップボタンが押されるまで繰り返すような回路を作成しなさい。また、以下に従ってデバイスを割付けなさい。

デバイス割付け表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 部品等 | デバイス | プログラム上の機能 | 部品等 | デバイス | プログラム上の機能 |
| ビットスイッチ１  （モーメンタリ） | X0 | スタート | ビットランプ１ | Y10 | 赤信号 |
| ビットスイッチ２  （モーメンタリ） | X1 | ストップ | ビットランプ２ | Y11 | 黄信号 |
|  |  |  | ビットランプ３ | Y12 | 青信号 |

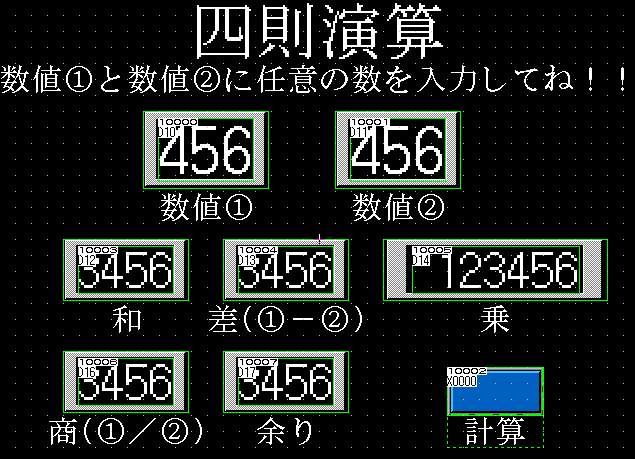
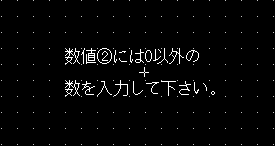


課題２　四則演算結果

・2数（－99～999）を任意で入力し、その2数の和、差（①－②）、乗、商、余り（①／②）を求めるものを作成しましょう！！ただし、数値②は0以外の数にし、0が入力され計算ボタンが押されたら、ウィンドウ画面を表示し、エラー表示させること。また、以下に従ってデバイスを割付けなさい。

デバイス割付け表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 部品等 | デバイス | プログラム上の機能 | 部品等 | デバイス | プログラム上の機能 |
| ビットスイッチ  （モーメンタリ） | X0 | 計算ボタン | 数値表示１ | D12 | 和の結果 |
| 数値入力１ | D10 | 数値① | 数値表示２ | D13 | 差の結果 |
| 数値入力２ | D11 | 数値② | 数値表示３  （符号付BIN32） | D14 | 乗の結果 |
|  |  |  | 数値表示４ | D16 | 商の結果 |
|  |  |  | 数値表示５ | D17 | 余りの結果 |



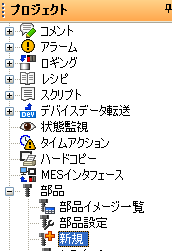
ベース画面 ウィンドウ画面

**スイッチの表示／非表示の切換え**

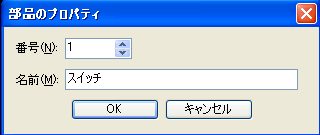
1. 部品の表示／非表示を使用する方法

部品をビットスイッチにして部品の表示／非表示で切り換えます。

* 1. 部品の作成をします。プロジェクトウィンドウ→部品→新規



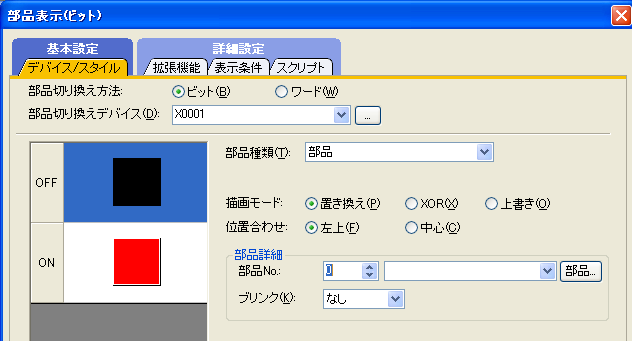
* 1. 部品に名前を付けます。



* 1. 部品を作成します。（例：四角形と塗りつぶし→グループ化）



* 1. メニューバー→オブジェクト→部品表示→ビット部品をクリックします。
  2. 配置した部品をダブルクリックします。
  3. 部品切り換えデバイス：X1　OFFをクリックし、部品Noを「０」に設定します。



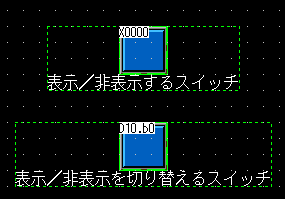
* 1. プログラムからX1をONにするとスイッチが表示され、OFFにするとスイッチが非表示になります。

1. セキュリティレベルを使用する方法

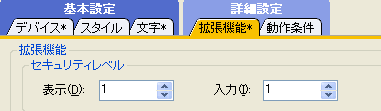
スイッチの表示／非表示をセキュリティレベルの表示で設定します。

* 1. 画面上に表示／非表示されるスイッチと表示／非表示を切り替えるスイッチを配置します。

ビットスイッチ2個（デバイスX0：ビットモーメンタリ／D10：ビット反転）

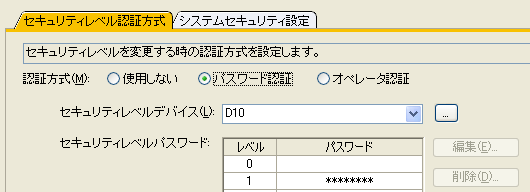


* 1. X0のビットスイッチにセキュリティレベル：１を設定します。



* 1. セキュリティレベル１のパスワード：「１」を設定します。

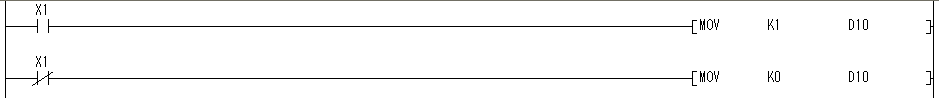
メニューバー→共通の設定→GOT環境設定→セキュリティ　（認証方式：パスワード認証　セキュリティレベルデバイス：D10　セキュリティレベルパスワード：１）



上記の設定をすると、D10ビットスイッチをON／OFFするたびに、X0ビットスイッチが表示／非表示します。（D10：ON→D10：１　／D10：OFF→D10：０）

今回は表示／非表示をビットスイッチで切り替えていますが、上記の設定をプログラムで書き換えることもできます。

例：X1センサがONした時にビットスイッチを表示させる。X1センサがOFFした時にビットスイッチを非表示にする。

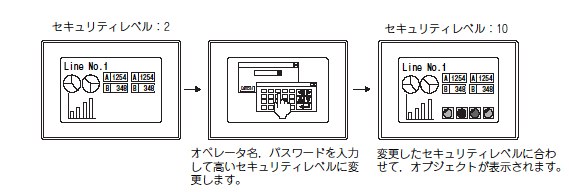


**セキュリティレベルについて**

オブジェクトや画面の表示、操作などにセキュリティを設定できます。セキュリティを設定することで、操作するユーザを制限できます。

●オブジェクトや画面の表示・操作を制限する

オブジェクトや画面ごとにセキュリティレベルを設定することにより、ユーザのセキュリティレベルに合わせた表示内容に変更できます。

****

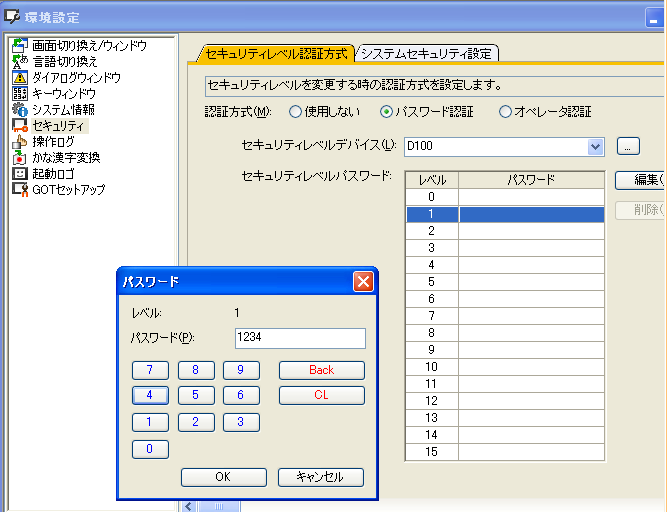
●認証方式

認証方式には、下記の2つがあります。

|  |  |
| --- | --- |
| セキュリティレベル認証 | パスワードのみでセキュリティレベルの変更を行う方式 |
| オペレータ認証 | オペレータ名とパスワードでセキュリティレベルの変更を行う方式 |

【操作】

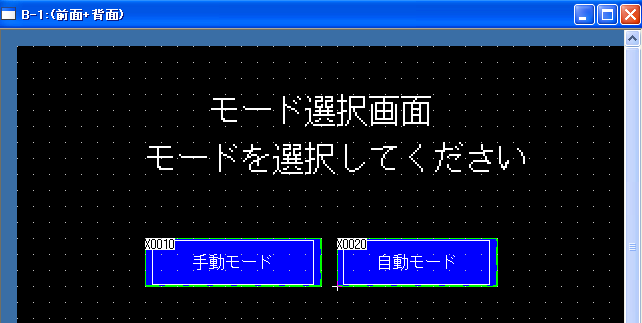
1. セキュリティレベルのパスワードを設定するには、メニューバーの「共通の設定」→「GOTの環境設定」→「セキュリティ」をクリックします。
2. セキュリティレベルデバイスを「D100」に設定し、セキュリティレベルパスワードのレベル１にパスワード「１２３４」を設定します。



【動作確認】

1. 手動操作画面および自動操作画面にセキュリティレベル１を設定し、動作確認をしましょう。
2. 一度セキュリティレベルを解除すると、次に手動操作画面および自動操作画面を表示する際にはパスワードが聞かれなくなります。毎回パスワードの確認画面を出す場合には、以下のように設定します。
   1. モード選択画面の自動モード／手動モード画面切り換えスイッチをスイッチに変更し、画面切り換えとビットスイッチに設定します。

手動モード：X10　自動モード：X20



* 1. 下記のラダーを入力します。



手動モードスイッチ／自動モードスイッチがタッチされたときに、セキュリティレベルを0に設定する。

・登録するパスワードは、最大8文字（半角数字）です。

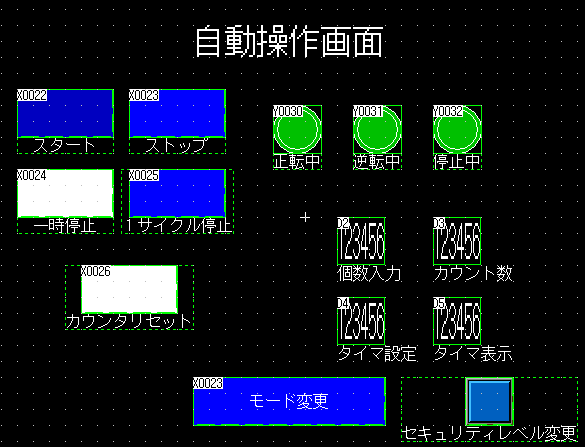
・各セキュリティレベルに設定したパスワードを忘れた場合には、セキュリティ解除ができなくなりますので、注意してください。

練習①

セキュリティレベルパスワード：レベル２にパスワード「4321」を設定しましょう。

練習②

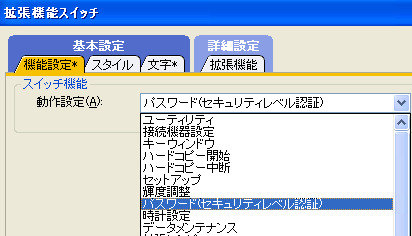
自動操作画面の「個数入力」と「タイマ設定」にセキュリティレベル２を設定しましょう。



【操作】

「個数入力」と「タイマ設定」のセキュリティレベルを変更するために、拡張機能スイッチの設定をします。

1. ツールバーの「スイッチ」→「拡張機能スイッチ」を配置します。
2. 配置した拡張機能スイッチをダブルクリックします。
3. 動作設定の一覧から「パスワード（セキュリティレベル認証）」を選択します。



1. 画面を転送し、セキュリティレベルを変更することで、数値入力ができることを確認しましょう。

コンベア課題

**デバイス割付表**

タッチパネル操作の機能割付

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 部品等 | デバイス | プログラム上  の機能 | 部品等 | デバイス | プログラム上  の機能 |
| ビットスイッチ1  （モーメンタリ） | X20 | 手動／正転 | ビットランプ1 | Y70 | 正転中ランプ |
| ビットスイッチ2  （モーメンタリ） | X21 | 手動／逆転 | ビットランプ2 | Y71 | 逆転中ランプ |
| ビットスイッチ3  （モーメンタリ） | X22 | 自動／スタート | ビットランプ3 | Y72 | 停止中ランプ |
| ビットスイッチ4  （モーメンタリ） | X23 | 自動／ストップ |  | D0 | ベース画面 |
| ビットスイッチ5  （ビットALT） | X24 | 自動／一時停止 |  | D1 | ウィンドウ画面 |
| ビットスイッチ6  （モーメンタリ） | X25 | 自動／1サイクル停止 | 数値入力 | D2 | ワーク個数設定値入力 |
| ビットスイッチ7  （モーメンタリ） | X26 | 自動／カウンタリセット | 数値表示 | D3 | 個数現在値表示 |
|  |  |  | 数値入力 | D4 | タイマ設定値  入力 |
|  |  |  | 数値表示 | D5 | タイマ現在値  表示 |

**画面作成**

1. ベース画面1は『モード選択画面』とし、画面切換えスイッチを用いてベース画面2とベース画面3に移れるようにする。
2. ベース画面2は『手動操作画面』とし、手動操作で必要な部品を配置し、デバイス割付表からデバイスを割りつける。また、画面切換えスイッチを用いてベース画面1に移れるようにする。
3. ベース画面3は『自動操作画面』とし、自動操作で必要な部品を配置し、デバイス割付表からデバイスを割りつける。また、画面切換えスイッチを用いてベース画面1に移れるようにする。
4. 『タイマ設定値』と『タイマ現在値』は秒単位で表示・入力するように、小数桁数の自動調整をする。

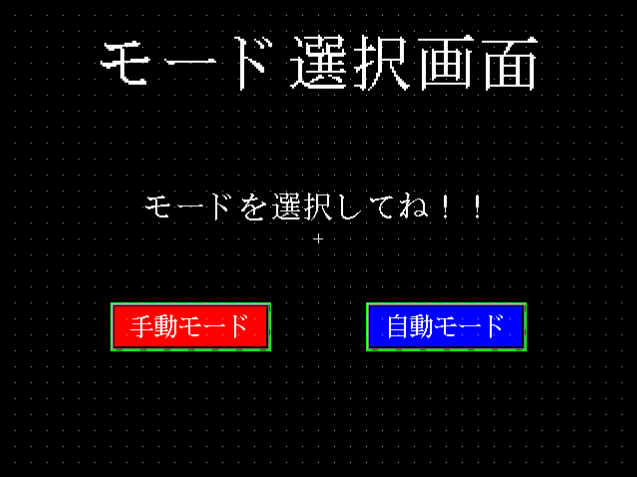
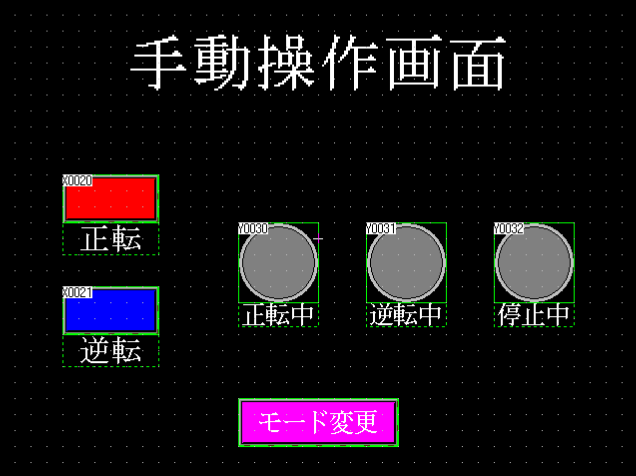


図1：ベース画面1

図2：ベース画面2

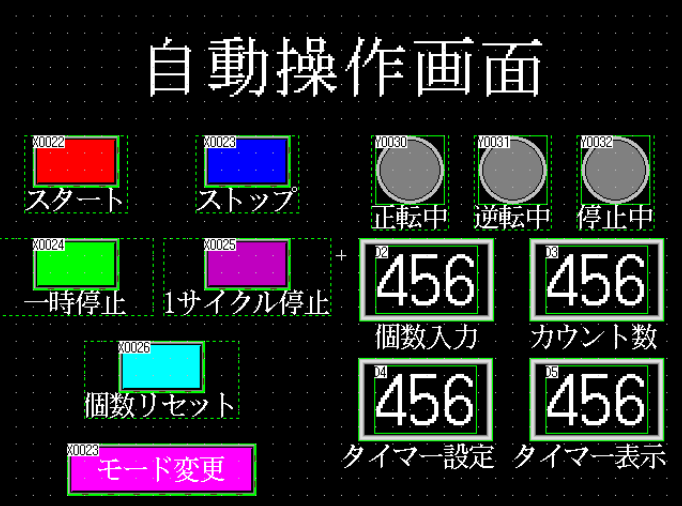
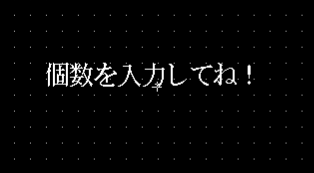


図4：ウィンドウ画面1

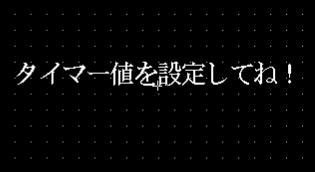


図3：ベース画面3

図5：ウィンドウ画面2

**課題1　手動運転プログラム**

* 『正転』ボタン(X20)がONの間はモータを正転させ、正転方向リミットセンサ(LS2)がONのときは正転しないようにする
* 『逆転』ボタン(X21)がONの間はモータを逆転させ、逆転方向リミットセンサ(LS1)がONのときは逆転しないようにする
* 正転中の時は正転中ランプ（Y70）を逆転中の時は逆転中ランプ（Y71）を停止中の時は停止中ランプ（Y72）をONする

**課題2　自動運転プログラム①**

* 『スタート』ボタン(X22 )がONになったら、モータを正転させワークを移動させる
* 正転方向リミットセンサ（LS2）が反応したら1秒間モータを止め、モータを逆転させる
* 逆転方向(LS1)リミットセンサが反応したら1秒間モータを止め、モータを正転させる
* 『ストップ』ボタン(X23)がONになったらモータは止まる（『モード変更』ボタンがONになった場合も同様に停止させるため、『モード変更』ボタンがONになったらX23をONするようにGOT画面を変更する→ビットスイッチではなくスイッチを使う）
* 正転中の時は正転中ランプ（Y70）を逆転中の時は逆転中ランプ（Y71）を停止中の時は停止中ランプ（Y72）をONする

**課題3 自度運転プログラム② (1サイクルストップ)**

* 『1サイクルストップ』ボタン(X25)がONになったら、ワークをリミットセンサまで移動してから停止させる。（正転中なら正転方向リミットセンサ(LS2)、逆転中なら逆転方向リミットセンサ(LS1)）

**課題4　自動運転プログラム③**

* 『スタート』ボタンがONになった時、個数入力が0の場合はウィンドウ画面1を表示させ、自動運転をスタートさせない
* 中間センサの反応した回数を外部より設定した値までカウントさせ、モータを停止させる。カウンタの現在値は常時表示させる
* 『カウンタリセット』ボタン(X26)がONになったら、カウンタの値をリセットする（『モード変更』ボタンや『ストップ』、『1サイクル停止』ボタンがONになった場合も同様である）

**課題5　自動運転プログラム④**

* 『スタート』ボタンがONになった時、タイマ設定が0の場合はウィンドウ画面2を表示させ、自動運転をスタートさせない
* カウンタがカウントアップした後、リミットセンサまでワークを移動させてからモータを停止する
* 課題２の停止時間1秒を外部より設定した時間（タイマ設定値）に変更し、またタイマの現在値を画面上に表示する。

**課題6　自動運転プログラム⑤ (一時停止)**

* 『一時停止』ボタン(X24)がONのときコンベアの動作を一時停止し、OFFになったら停止前の状態から再開する(正転なら正転、逆転なら逆転から再開する)