

DSH-PLCSim PLC シミュレータ

操作説明書

2008年6月

株式会社データマップ

文書番号 DSHPLCSIM-08-30305-00



[取り扱い注意]

- ・ この資料ならびにソフトウェアの一部または全部を無断で使用、複製することはできません。
- ・ 本説明書に記述されている内容は予告なしで変更される可能性があります。
- Windows は米国 Microsoft Corporation の登録商標です。
- ・ MELSEC は三菱電機株式会社殿の登録商標です。
- ・ ユーザーが本ソフトウェアの使用によって生じた遺失履歴、(株)データマップの予見の有無を問わず発生 した特別損害、付随的損害、間接損害およびその他の拡大損害に対して責任を負いません。

改訂履歴

番号	改訂日付	項目	概 略
1.	2008.6	初版	
2.			
3.			
4.			



1. 概要	1
[関連文書]	1
[必要なソフトウェア]	1
1. 1 DSH-PLCSimの目的	2
1. 2 構成	3
1. 3 機能概要	4
(1) I/0 デバイス定義機能	4
(2)入出力制御機能	4
(3) キャリア I Dリーダー, スロットマップリーダーと処理データ入力機能	5
(4)簡易シーケンスプログラム画面による自動シーケンスの実行機能	5
1. 4 操作の流れ	6
(1) I/0 デバイス定義ファイルを作成する。	6
(2)シーケンス制御プログラムを作成する。	6
(3) デバイス制御をマニュアル操作で行なう。	6
(4) デバイス制御をシーケンスプログラムを使って自動で行なう。	6
2. PLC シミュレータの起動 とメイン画面	7
(1) I/O 定義ファイル編集 ボタン	7
(2) 1/0 操作開始 ボタン	7
3. I/0 デバイス定義と定義ファイル作成/編集操作	8
3.1 I/0 デバイス定義の開始操作	8
(1) ファイル(F)メニューから開始する方法	8
(2) ツールバーの I/O 定義ファイル編集 ボタンのクリックで開始する方法	8
3. 2 入出力デバイスのアドレス空間と画面	10
3.3 I/0 デバイス定義の操作	11
3. 4 定義情報のファイル保存操作と定義操作終了	12
3. 4. 1 保存操作	12
 3.4.2 I/0 デバイス定義操作の終了 	12
4. I/0 デバイス制御操作	13
4.1 I/0 デバイス制御の開始手順と操作画面	13
(1) I/0 定義ファイルの選択	13
(2)開始	13
(3)操作ボタン	14
4. 2 デバイスの 0n/0ff 操作手順	15
(1) デバイス名の選択	15
(2) 0n/0ff 操作	15
(3) 全デバイスの 0n/0ff	15
5. シーケンスプログラムによる自動制御のための操作	16
5. 1 開始とシーケンスプログラム画面	16
5. 1. 1 開始操作	16
5. 1. 2 プログラムの新規作成/開く/開きなおす	17
(1)新規作成	17
(2)開く	17
(3)開きなおす	17
5.1.3 画面と操作ボタンの説明と操作	18
(1)プログラム設定グリッド部	18
(2)入出力デバイス名選択コンボボックス	19



(3)制御コマンドボタン	19
(4) 設定操作例	20
5. 2 編集と保存	21
(1)編集	21
(2)保存形式と保存	21
5.3 シーケンスプログラムの実行と停止	22
(1) 実行	22
(2)停止	22
6. 入出力デバイスの制御プログラムサンプル	23
6.1 I/0 デバイス定義と制御シーケンス	23
6. 2 プログラム作成	24
6. 3 実行操作	25
7. キャリア ID 画面操作とプログラミング	27
(1)Carrier ID - キャリア ID 画面とプログラム	27
(2) スロットマップ画面操作とプログラミング	28
(3) プロセスデータ	28
8. その他の操作	29
8.1 ログ表示とログファイルについて	29
8. 1. 1 画面のログ表示	29
8.1.2 ログファイル	29
8. 2 EQCON との接続情報の設定	30
(1) DSH-SIMPlc での通信設定の方法	30
(2)EQCON との接続タイミング	30
(3) 切断のタイミング	30
8.3 I/0 デバイス定義画面の文字フォントとセルサイズの変更	31
(1)フォントの変更設定操作	31
(2) I/0 編集画面の Cell サイズ設定	31



1.概要

本説明書は、DSH-PLCSim PLCシミュレータに関する操作について詳しく説明します。

本シミュレータがサポートする PLC は、三菱電機(株)製 MELSEC シリーズの PLC であり、バス・インタフェース関数が使用できるものを対象としています。

[関連文書]

DSH-PLCSim 関連文書一覧表

#	文書番号	文書名	注釈
1	DSHPLCSIM-08-30304-00	DSH-PLCSim ユーサ゛ース゛・カ゛イト゛	機能と操作概要説明書
2	DSHPLCSIM-08-30305-00	DSH-PLCSim 操作マニュアル	この文書です。
3	DSHPLCSIM-08-30320-00	DSH-PLCSim インタフェース関数説明書	装置コントローラ側が使用できる
			インタフェース関数
4	DSHPLCSIM-08-30390-00	DSH-PLCSim PLC 制御デモ・プログラムに	VB6 のデモプログラムの説明書で
		ついて (VB6.0 言語版)	す。
5	DSHPLCSIM-08-30391-00	DSH-PLCSim PLC 制御デモ・プログラムに	Visual C++のデモプログラムの説
		ついて(VC++6.0 言語版)	明書です。

[必要なソフトウェア]

ソフトウェア	
OS	Windows-2000、Windows-XP またはWindows-Vista
DSHPLCSIM 関連	dshplcsim.exe - シミュレータ本体プログラムファイル dshplc.dll - PLC 通信プログラムライブラリファイル



1.1 DSH-PLCSim の目的

DSH-PLCSimの目的は、半導体製造工場内に設置される半導体製造装置の制御プログラムの開発をトータルシステム として総合的にシミュレーションできるようにするためのソフトウェアツールとして、きめ細かにその役割を果たす ことです。

弊社は、半導体製造装置コントローラ(以下、EQCON と呼びます)のソフトウェア開発システムとして以下のソフトウェアツールならびにコンポーネントとしてのライブラリを提供しています。



このようなシステムの中で、EQCON に接続される PLC (Programmable Logic Controller)、キャリア ID リーダー、スロットマップリーダーの代わりを果たすシミュレータとしての機能を提供します。

すなわち、DSH-PLCSimは、PLC、キャリア ID リーダー、スロットマップリーダーの役割と機能を論理的に受け持ち、 EQCON のプログラムテストを基本的なテストから総合的なシーケンスのテストまでのプログラム処理を、バグが無く なるまで繰り返して簡単に実行できるようにするための環境を提供します。



1.2 構成

DSH-PLCSimにおけるシステム構成は大体以下のようになります。

[DSH-PLCSimの構成]



シーケンスプログラム実行画面を含めた画面は次のようになります。





1.3 機能概要

PLC シミュレータ機能として、以下の機能があります。

(1) I/0 デバイス定義機能

入出力接点デバイス番号の定義を I/O マップ画面の操作でおこないます。 各デバイスには、名前とアドレスが与えられます。アドレスは入力、出力それぞれ 0~17FF(16 進)まで の空間まで定義できます。 入力デバイスの名前には**"X_"**を、出力デバイスの名前には**"Y_"**を付けます。

また、定義された I/O デバイス名とそれに与えられたデバイス番号が対応付けられた C, VB6 用ソースファ イルが生成されます。EQCON のプログラムをヘダーファイルとして使用することができます。 入力デバイス定義画面例

🔚 Input	t Bit Device Definitio	n Editor – C:¥DshGen	1Lib¥PlcSim¥	Sample1.	pls
X-addr	0	1	2		3
0000					
001 0	X_Port1 RdyToLoad	X_Port1 StartLoad	X_Port1LoadEn	d	X_Port1 C
0020	X_Port2RdyToLoad	X_Port2StartLoad	X_Port2LoadEn	d	X_Port2C
0030	X_Port1 RdyToUnload	X_Port1 StartUnload	X_Port1UnloadE	Ind	
0040	X_Port2RdyToUnload	X_Port2StartUnload	X_Port2UnloadE	Ind	
0050	X_ProcessStart	X_ProcessEnd			
0060					
]	FQCON	
	C用hea VB6用	l. bas file		制御7° 使用	ログラムで 可

(2)入出力制御機能

(1) で定義した I/0 デバイスの I/0 マップ制御画面上での操作であり、デバイスの On/Off の設定とその 情報をシミュレータに接続されている装置コントローラへ送信します。EQCON からはデバイス読込み関数 を使ってデバイスの On/Off 状態を読み出すことができます。これらの操作はマウス操作で簡単にできます。

1/0定義	7>-17.4	5	65		C¥D	shGe	mLibi	VP1c3	Sim¥	Sampl	le1.pl	ls –	1/0)操作(17 HS	_	10	60710156	V	0腰1	制料了		1/6	定网	7731	新 集	17	0774	1.67.9	(終了	1	R258	法	Log
16 接点	入力()n/(Dff 1	(9)	7												×	<mark>℃</mark> 接点	出力()n/()HI 🗟	197												
ON	OF	F	翻	X_Po	et1B	dyTo	Load			۲	A	llOn	A	lion		司期	E	ON	OF	E já	訳	(_P1	Star	rt				٠	A	llOn	A	llOff		司期
X-addr 0000 0010 0020 0020 0020 0020 0020 0050 005	• • • •		-	.0	- 0	-	.0	-					C					Y-34dr 0000 0019 0020 0030 0040 0050 0050 0050 0050 0050 005	0 0 - -	-	-	-	-	-	-	-		9	A		C		C	
0080 0000 0000 0010 0100 0110 0120 0130 013																		0080 0000 0000 0000 0000 0100 0110 0120 0130 0140 0150 0160 0170																

これらのデバイスは、(4)で述べる簡易シーケンスプログラムによってシナリオに基づく制御も可能です。



(3) キャリア I D リーダー, スロットマップリーダーと処理データ入力機能

画面上にキャリアID、スロットマップ、プロセスデータの3種類の情報を設定できます。これらの情報は、EQCONから読み込むことができます。

٩c	キャリアID. スロット	77 💶 🗙
_Ca	arrier ID	Slotmap
	CARID_02	🗖 (slot-01)
1	CARID_01	IV slot-02
2	CARID_02	IV slot-04
3	CARID_03	🔽 slot-05
4	CARID_04	IV slot-06
5	CARID_05	IV slot-07 IV slot-08
6	CARID_06	▼ slot-09
7	CARID_07	▼ slot-10
8	CARID_08	I slot−11
9	CARID_09	IV slot-12
10	CARID_10	▼ slot-14
		🔲 slot-15
⊏7°	ロセス データー・・・・	🔽 slot-16
1	AAAAA	I slot-17
,	BBBBB	IV slot-18
2	333.3	Slot-19
4	444.4	▼ slot-21
5	555.5	▼ slot-22
6	666.6	IV slot-23
7	XXXXX	slot-24
Ľ		,
	確認	更新

(4) 簡易シーケンスプログラム画面による自動シーケンスの実行機能

簡単なコマンドを使って、表の中にデバイスの 0n/0ff 条件と設定を並べたり、ステップのジャンプ、時間 遅延などの制御を行ないながら、制御シーケンスを作り、それを実行させることができます。 実行すると、自動的に表上に組まれたプログラムを実行してくれます。これで一通りの搬入、処理、搬出 までのシナリオ制御を自動的に行なうことが可能です。

<mark>れ</mark> シーケン	<mark>₩</mark> シーケンス・プログラミング - C:¥DshGemLib¥PlcSim¥seq_300.prg											
-7ァイル	: 1				行ってもろし	tt ≤ 75						
新祝作成												
##本 X_Port1RdyToLoad ▼ 入力設定 Y_PlcStart ▼ 出力設定 行挿入 7ァイル挿入 行削除 Beep Off ▼												
-1771	ステッフ°	設定対象信号名	On/Off	On/Off条件になる信号名	On/Off	 אנאב 						
ALLX	1	\$ALLX	Off			全Input signal = Off 📃						
ALLY	2	SALLY	Off			全Output signal = Off						
	3	\$SLEEP	3									
GOTO	4	404DID										
IFGO	0	DCARID	1									
	7	X Port 1BdyToLoad	0n	Y Port1Rdvlamp	0n	We CENCOOP の過点 Y Port1Rdvl amp = Op 待ち						
DELAT	8				on	Optortation for the start of th						
PAUSE	9	X Port1StartLoad	On									
WAIT	10	Y_Port1RdyLoadLamp	Off									
	11											
Carid	12	\$DELAY	2									
	13		-									
	14	X_Port1Clamped	On									
SlotMap	10	DELAT	4									
1 🗸	17	A_POPUISUARULOAD	UTT									
ProcData	18	\$WAIT	0n	Y_Port1FoupOpen	On							
	19											
	20	X_Port1FoupOpened	On			•						



1.4 操作の流れ

操作は、以下の順に進めます。

(1) I/0 デバイス定義ファイルを作成する。

入力情報、ファイル	操作	出力されるファイル
(1) I/Oデバイス割付仕様書	I/0 デバイス番号割付仕様書に従って I/0 デ	 I/Oデバイス定義ファイル
	バイス定義操作画面上で定義し、定義ファイル	.pls 7711
	を作成します。	(2) C/C++, VB6 用
	(画面で修正もできます)	デバイス名定義やり
	=>3章で説明しています。	.h, .bas
		装置コントローラプログラムで
		使用可能

(2) シーケンス制御プログラムを作成する。

入力情報、ファイル	操作	出力されるファイル
(1) I/0デバイス定義ファイル	対装置コントローラプログラムのシナリオに	(1) シーケンス制御プ ログ ラムファイル
.pls	基づき入出力制御 PLC 側のシーケンスプログ	.prg 771N
	ラムを作成します。	
(2) 装置コントローラがPLCに対し		
て行なう制御ンーケンス(シナリ	=>5章で説明しています。	
れ) 仕様書		

(3) デバイス制御をマニュアル操作で行なう。

入力情報、ファイル	操作	出力されるファイル
(1) I/Oデバイス定義ファイル	対装置コントローラプログラムのシナリオに	(1) デバイスの制御りブファイル
.pls	基づき PLC 側のデバイスの On/Off 制御を画面	dshplc.log
	上でマニュアル操作で行ないます。	
(2) 装置コントローラが PLC に対し	装置コントローラからの出力デバイスの変化	
て行なう制御シーケンス(シナリ	を見ながら操作する。	
オ) 仕様書	==> 4章で説明しています。	

(4) デバイス制御をシーケンスプログラムを使って自動で行なう。

入力情報、ファイル	操作	出力されるファイル
(1) I/0デバイス定義ファイル	対装置コントローラプログラムのシナリオに	(1) デバイスの制御ログファイル
.pls	基づき PLC 側のデバイスの On/Off 制御をシー	dshplc.log
	ケンスプログラムを使って自動的に行なう。	
(2) シーケンス制御プ ログ ラムファイル		
.prg	=> 6 章で説明しています。	



2. PLC シミュレータの起動 とメイン画面

dshplcsim.exe プログラムファイルを実行します。

インストール時、通常、dshplcsim.exe は、c:¥Program Files¥DSHPLCSIM のフォルダーにインストールされます。 そのフォルダーを開き、dshplcsim.exe ファイルをダブルクリックすることによって起動することができます。

起動すると次のメイン画面が表示されます。

え DSH-PlcSim-2008-1 Trial S/N: 1212630205-T - effective 180 days after 2008-6-5 PLC Side 株式会社
アイル(を) 編集(を) 表示(な) 通信設定(な) ウインドウ(分) ログファイル名(な) ヘルブ(分)
1/0定義7748名 ●編 C¥DshGenLb¥PkSim¥Sample1.pls 1/0操作開始 東 550710152 1/0操作時7 1/0定義7748編集 1/07748編集時7 四5論者 Los On 💽
↓ し な し な し い の た 、 に 関 結 し ま す 」 ↓ し い で が 「 し い の た い に 、 能 の は 、 な の は て 、 な の た い に 、 な 、 な の は て 、 な い の た い に 、 な い の た い に 、 な い の た い に 、 な い の た い い れ 、 な の は 、 な 、 な 、 な 、 な 、 な 、 な 、 な 、 な 、 な 、
Agence Operation Los ログ 表示画面

PLC シミュレータの主な操作は、ツールバー上にある2つのボタンのクリックから始まります。

1/0操作開始	SEQ7個がラム	1/0操作终了	1/0定義ファイル編集	1/07ァイル編集終了
---------	----------	---------	-------------	-------------

(1) I/O 定義ファイル編集 ボタン

PLCの I/0 デバイス定義画面が表示されます。入力、出力それぞれ別の画面で、アドレス (デバイス番号) マップ上でデバイス名を定義します。

定義操作の結果として、I/Oデバイス定義ファイルと、EQCONで使用することができるプログラムファイルが作成されます。

I/O デバイス定義画面は、7740(F)メニューのファイルを開くなどのタブのクリックによっても開始することができます。

(2) I/0 操作開始 ボタン

(1)の I/O 定義ファイル編集の画面で作成した I/O デバイス定義ファイル上に定義された I/O デバイスの On/Off 制御のための操作画面が表示されます。

この I/O 操作画面では、キャリア ID、スロットマップ、プロセスデータ設定画面も表示されます。 また、シーケンスプログラム画面による自動制御も実行できます。



3. I/O デバイス定義と定義ファイル作成 / 編集操作

I/0 デバイス定義は、入力(X)、出力(Y)それぞれのアドレスに割り付けられたデバイス名を入力、決定することです。

定義された情報は、I/0 デバイス定義ファイルに保存されます。そして、4. で説明する I/0 制御の操作の対象デバイスになります。

3.1 I/O デバイス定義の開始操作

ファイルメニューから開始する方法と、ツールバー上の I/O 定義ファイル編集 ボタンから開始する方法があります。

(1) ファイル(F)メニューから開始する方法

次に示す3つのメニュータブのクリックで開始します。

<mark>የ</mark> DSH-	H-PlcSim-2008-1	Trial S/N : 1213
ファイル(E)	E) 編集(E) 表示(2) 通信設定(<u>S</u>)
I/O定義; I/O定義; I/O定義; I/O定義; I/O定義;	義ファイルー新規作成() 義ファイルー開きなおす() 義ファイルー開く(Q) 義ファイルー上書き保存 義ファイルー名前を付け)) Ctrl+N () Ctrl+R Ctrl+O (<u>S</u>) Ctrl+S (保存位) Ctrl+A
履歴ファイ 画面情報	ァイルの初期化の 韩報をテウォルトに戻す())
シミュレーター	タ-終了⊗	

新規作成の場合は、定義が全く行われていない空の定義画面が表示されます。

開きなおす場合は、もし以前開いたファイルがあれば、その一覧表の画面がポップアップされますので、 その中から選択します。選択されたファイルの内容を読み出し、定義画面に内容を表示します。この場合、 既存のファイルの編集になります。

開く場合は、ファイルを開くためのダイアログ画面が表示されますので、そこで開きたい I/0 デバイス 定義ファイルを指定します。指定されたファイルの内容を読み出し、定義画面に内容を表示します。この 場合も既存のファイルの編集になります。

(2) ツールバーの I/0 定義ファイル編集 ボタンのクリックで開始する方法

ボタンがクリックされると、画面の I/0 定義ファイル名の表示部にファイル名が表示されているかどうかによって次のようになります。

1/0定義ファイル名	参照	
	22.88	J

①ファイル表示部がブランクである場合は新規の定義ファイルの作成になります。

②ファイル表示部に定義ファイル名が表示されている場合は、ここに表示されているファイル名の定義 ファイルが開かれることになります。

参照ボタンを使って、ファイルを開くダイアログ画面で、ファイルを選択し、それをファイル表示部に表示さることもできます。

次ページに、新規画面と開きなおしによる場合の I/0 デバイス定義画面の表示内容を示します。



[新規で作成する場合の定義画面]

1/0定着7>	as 💷		1/0陳伊間台	580710954 1/01	#作許了 1/0定前7x43	回回 1/07r-估議集終了	DITAR Log On		
<mark>¶ç</mark> İnput B	3it Device Definitio	n Editor -							
X-addr 0		1	ż	3	4	5	6	7	8
0000									
0020									
0000									
0040	-								
0050	スカ側・	未定義たの	で全地が空日	白にたります	-				
0000		小儿我ない			0				
0000									
0090									
00A0									
0080									
0000									
0060									
00F0									
01.00									
Y-addy 0		1	2	3	4	5	6	7	
0000									
0 100									
0000									
0040									
0050									
0060									
0070	出力側:	未定義なの)で全地が空	白になります	F_				
0000		1001200		H. 0.9 0.9	0				
0040									
0000									
0000									
0000									
0060									
01 00									
1									

[開く、開きなおしでファイルを読み出した場合の定義画面]

1/028	NALS OR CHO	shGemL&#PicSim#sample</th><th>pla 1/014/17285</th><th>EE075154 1/0</th><th>#作許字 1/0定稿7x4</th><th>1/07+33編集件了</th><th>Diag Lot Or</th><th>2</th><th></th></tr><tr><td>Ac Input</td><td>Bit Device Definitio</td><td>n Editor - C.VDshGer</td><td>nLibYPIcSim¥sample.p</td><td>ła</td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>X-addr</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>1</td></tr><tr><td>0000 0010 0020 0040 0040 0050 0050</td><td>XPortERdyToLoad XPortERdyToLoad XPortERdyToLlabed XPortERdyToLlabed XProcessStart</td><td>X_PortStarLoad X_PortStarLoad X_PortStarUnload X_PortStarUnload X_ProcessEnd</td><td>X.PortI LoadEnd X.PortI LoadEnd X.PortI UnicadEnd X.PortI UnicadEnd</td><td>X.Port1 Clanged X.Port2Clanged</td><td>X/PortEFoxpOpened X/PortEFoxpOpened</td><td>X,PortIF oupClosed X,PortIF oupClosed</td><td>X,Port1MovedCarWork X,Port2MovedCarWork</td><td>X.Port1Move0OarOst X.Port2Move0OarOst</td><td>X/PortSLoadReg X/PortSLoadReg</td></tr><tr><td>0090 0090 0040</td><td>入力側:</td><td>:定義済みの</td><td>Dアドレスには</td><td>Mに入力デバ</td><td>収名が表示</td><td>されます。</td><td></td><td></td><td></td></tr><tr><td>0080 0000 0000</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>00F0</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>Y-addr</td><td>0</td><td>1</td><td>2</td><td>э</td><td>4</td><td>5</td><td>6</td><td>7</td><td>1</td></tr><tr><td>0000 0010 0020 0000 0040 0050 0050</td><td>Y/PortERdyLoadLamp Y/PortERdyLoadLamp Y/PortERdyUnloadLamp Y/PortERdyUnloadLamp Y/PoceasStartRdy</td><td>Y/PotTRdyToLoad Y/PortTRdyToLoad Y/PortTRdyToLoad</td><td>Y/Porti ToDodk Y/Porti ToDodk Y/Porti ToDodk</td><td>Y/Port1 Clamp Y/Port2Clamp</td><td>Y/Fort1FoxpOpen Y/Fort2FoxpOpen</td><td>Y/FortEF oupClose Y/FortEF oupClose</td><td>Y/Port1MoveCarWork Y/Port2MoveCarWork</td><td>Y/Port1MowDarDat Y/Port2MowDarDat</td><td></td></tr><tr><td>0090</td><td>出力側:</td><td>:定義済みの</td><td>Dアト レスにはt</td><td>ルこ出力デバ</td><td>(れ名が表示)</td><td>されます。</td><td></td><td></td><td></td></tr><tr><td>0040 0080 0000 0000 0000 0000 0000 0000</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>100</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr></tbody></table>
-------	-------------	---



3.2 入出力デバイスのアドレス空間と画面

入出力デバイスの定義の内容は、デバイスのアドレス(番号)と名前、デフォルト 0n/0ff 値ならびにそのタイトル 名です。アドレス1個に1ビットのデバイスが割当てられます。

デバイスの名前の頭に、入力デバイスに対しては "X_" を、出力には "Y_" を付けることにします。

デバイスのアドレスは、4桁の16進数で表現し、個々のデバイスについて、入力はX-アドレス、出力はY-アドレスの空間に割当てられます。

入力の空間は、 X-0000 から 最大 X-73FF まで、また、出力の空間は、Y-0000 から最大 Y-73FF まであります。

アドレス空間の画面での表現は、横軸に 0~F までの 16 アドレス (ビット) そして、縦軸に 16 単位のアドレスとす る平面のグリッドになります。例えば、入力については下表のようになります。

X-addr	0	1	2	3	F
0000					
0010		X_Valve_10			
0020					
0030					
0040					
0050					
0060					
73E0					
73F0					

上の表の X_Valve_10 のアドレスは、16 進で X-0011 番地になります。(番地はデバイス番号と同じ意味です。)

実際の定義操作では、仕様で定められた各デバイス名をデバイス番号に相当するアドレスに設定していくことになります。

デバイス番号に名前を付けることによって、EQCONのPLC制御プログラムは、この定義操作の結果として生成されるプログラム用ファイルをそのまま使い、デバイス番号をここで決めた名前で参照することができます。

例えば、X_Valve_10の入力デバイスの状態を読み出す場合、次のようにプログラミングすることができます。

C/C++ 言語

result = QBF_X_In_BitEx(&path, sFlg, X_Valve_10, &data);

VB6 言語

result = QBF_X_In_BitEx(path, sFlg, X_Valve_10, data)



3.3 I/O デバイス定義の操作

- 3. 2の例で、X-0011 番地を X_Valve_10 の名前として定義するための操作を順に説明します。
 - (1) まず、画面上の0011番地に当る画面上のセルをマウスでダブルクリックします。

ትሮ Input Bit Device Definition Editor –						
X-addr	0	1	2			
0000						
001.0		I 👗				
0020						
0030						
0040						

次のようにデバイス定義の入力画面がポップアップされます。

Ac Input Signal S	etting	×
アドレス(Dev No.)	11	
名前(半角英数字)	X	
初期状態(On/Off)	Off 💌	
コメント		
	ОК	キャンセル

- 以下、この画面で入力、選択を行ないます。
- (2)名前を"X_Valve10" とキー入力します。(頭の X_ は、最初から表示されています。)
- (3) 初期状態、On/Off かを選択します。
- (4) コメント欄には、デバイスの意味合いなどのコメントを入力します。(空白でも構いません。) 例えば、"Chamber Gus-A Valve-10"のように入力します。
- (5)入力が終った後、OKボタンのクリックで内部に取り込まれ、この入力画面が閉じます。
- (6) 一旦入力画面が終了した後、再び、0011 番地のセルをダブルクリックすると、(1) で入力した内容が表示されます。

Ac Input Signal S	etting	×
アドレス(Dev No.)	11	
名前(半角英数字)	K_Valve10	_
初期状態(On/Off)	Off 💌	
	Chamber Gus-A Valve-10	
	OK	キャンセル

以下、順に定義された入力デバイスを設定していきます。

出力デバイスについても同様に行ないます。



3.4 定義情報のファイル保存操作と定義操作終了

3.4.1 保存操作

定義した I/0 デバイス定義情報は、774%(F)メニューの次の2つのタブのクリックで保存できます。

上書き保存では、1/0 定義ファイル名表示部に表示されている名前のファイルに上書きされます。 また、名前を付けて保存の場合、保存した後、1/0 定義ファイル名表示部に保存したファイル名が表示されます。

(1) I/0 デバイス定義情報は、拡張子 ".pls"のテキスト形式ファイルとして保存されます。

3. 3で定義した X_Valve10 の入力デバイスを test.pls に保存した場合、ファイル内では次のように保存されます。

[test.plsファイル]

//----- Input Device Bit Definition def_in_bit X_Valve10{ addr: 0x11 nominal: 0 caption: "Chamber Gus-A Valve-10"

(2)保存時に、EQCON制御プログラム用ファイルが生成されます。

[test.hファイル]

/*			*/
/* C:¥D	shGemLib¥PlcSim¥test.h -	PlcSim Addr Definition	n */
/*			*/
#define	X_Valve10	0x0011	// 17 Chamber Gus-A Valve-10

[test.bas ファイル]

Attribute VB_Name = "test.bas"

test.bas - PlcSim Addr definition

Public Const X_Valve10 = 17 '0x00000011 Chamber Gus-A Valve-10

3.4.2 I/O デバイス定義操作の終了

メイン画面のツールバー上の 1/077小編集終了ボタンのクリックで定義画面が終了します。



4. I/O デバイス制御操作

3章で作成したI/0デバイス定義ファイルに定義された入出力デバイスのON/OFF制御を画面上で行なうことができます。

また、入出力デバイスの制御の他に、キャリア ID リーダー、スロットマップリーダーならびにプロセスデータの設 定操作も制御の1部として実行することができます。これらについては7章で説明します。

4.1 I/O デバイス制御の開始手順と操作画面

まず、メイン画面上の参照ボタンを使って、操作したい I/0 デバイス情報が登録されている I/0 定義ファイル名を 選択します。そして、その後、ツールバーの **I/0 操作開始ボタン**をクリックして、制御画面を表示します。

以下、既存の I/O 定義ファイル sample.pls についての操作を順に説明します。

(1) I/0 定義ファイルの選択

★照ボタンをクリックし、ファイルを開くダイアログ画面を開いてⅠ/0 定義ファイルを選択します。 選択されたファイル名が表示部に表示されます。

1/0定義ファイル名	参照	C:¥DSHPIcSim¥sample.pls	1/0操作開始	SEQ7 [®] ロがラム

(2)開始

次に I/0 操作開始ボタン をクリックします。

I/O 画面が表示されます。(キャリア ID リーダーの画面も出てきますが、7章で説明します。) この画面で、入力、出力デバイスをコンボボックスまたはデバイスアドレスマップ上の位置のクリックで 選択しながら 0N/0FF の制御操作を行なうことになります。

I/O定義ファイル名 参照 C¥DSHPIcSim¥samp	le.pls 1/0操作I	開始 SEQ 7 17 5	4 I/O B	操作終了	1/0定義7ァイ/	1/0	ファイル編身	烏終了	ロケジ首		; Or
<mark>ใ</mark> ழ接点入力On/Off マ ッ プ			出力0n/	/0ff マップ						_ 🗆 ×	4
ON OFF 選択 X_Port1RdyToLoad	AllOn AllOff	f 同期 ON	OFF	選択 Y_P1	cStart		• A	llOn	AllOff	同期	
X-addr 0 1 2 3 4 5 6 7	8 9 A B C D	E F Y-addr	0 1	2 3	4 5 6	78	9 A	BC	; D	<u>E F</u>	
	-	- 0010				-					-
<u>0020</u> O - O	-	0020	- 0			-					
0040		0040	-								
0060		0060			.0						
0070 マップ 画面		0070		7	ップ 画面						
<u>0090</u> 入力デバイスの状	能	0090		出	力デバイス	の状態					
0080 - · Off 计台		0080			- · 0ff	中能					
<u>0000</u> - · · · · · · · · · · · · · · · · · ·		0000			• 011	1八亿					
● O : On 状態		00E0			O: 0n	犬態					
		0100			X : (≠	ト 接続 お	(熊)				
0110		0110		<u> </u>							
0130		0130					-				
0140 入力デバイスマッ	7°	0140			出力デバー	イスマップ					
0160		<u>0160</u> ▼ 0170								_	-
X Port2FoupClosed = ON		Y_Port1	⊣ RdyToLoa	d = OFF	<u> </u>						1
		1,-	1								~1
	<u> </u>										
現在選択され	ているデバク名と	: On/Off 状態か	表示る	されま	す。						
(7)7 西西	トで選択されたデッ	いれをですう			Ĩ						
(197) 回山_		· 1/2日 く 9 o /									



(3) 操作ボタン

入力、出力デバイス別々に操作ボタンがあります。ボタンは両方とも同じ名前で、同じ機能になります。





4.2 デバイスの On/Off 操作手順

デバイス単位の On/Off 操作手順について説明しますが、操作されたデバイスと状態情報は、EQCON と接続中であれば、その入力デバイス情報については EQCON にも送信されます。

(1) デバイス名の選択

二通りの方法があります。

 ①1つ目の方法は、デバイス名コンボボックス上のデバイス名をマウスで選択します。例えば、 X_Port1RdyLoad、デバイス番号 = 0x0011 を選択すると次のようになります。

n∕Off ∃	マップ						
選択	X_Port1	RdyToI	.oad		•	A	
1 2	X_Port1	RdyToI	.oad		-		
	X_Port1 X Port1	StartI LoadEr	.oad 1d				\mathbb{r}
	X_Port1 X_Port1	Clampe FourDr	ed Demed				_
	X_Port1	FoupCl	osed	1-			
-	X_Port1 X_Port1	movea(Moved(.arwo CarDs	rĸ t	-		

<mark>℃</mark> 接点)	<mark>ใழ</mark> 接点入力On/Off マ ッ プ										
ON	OF	OFF 選択 K_Port1StartLoad									
X-addr	0	1	2	3	4	5	6	7			
0000											
0010	-	-	-	-	-	-	-	-			
0020	-	-	-	0	-	0	-	-			
0030	-	-	-								
0040	-	-	-								
0050	-	-									
0060											
0070											

②2つ目の方法は、デバイスマップのデバイスセルを直接クリックします。
 選択したいデバイスのデバイス番号(番地)が予め分かっている場合に行ないます。この場合、ヒント表示(ポップアップされる)でデバイス名が表示されます。

(上の右側の X-addr の 0011 の位置をクリックします。)

③選択されたデバイス名は、1/0マップ画面の下のステータスバー上に表示されます。

0150			
0160			
0170			
X Port1Star	tLoad	-	ステータスハー

(2) 0n/0ff 操作

(1) でデバイス名を選択した後、画面左側の ON または OFF ボタンをクリックします。

クリックされたボタンによって、画面の選択された位置の表示も変化します。

ON : O が表示されます。

OFF : - が表示されます。

(3) 全デバイスの 0n/0ff

全デバイスを 0N または OFF にしたい場合は、AllOn または AllOff ボタンをクリックします。 定義されている全デバイスについて状態が 0n または Off にセットされます。 このときも入力デバイスの情報は EQCON と接続中であれば、EQCON に送信されます。



5.シーケンスプログラムによる自動制御のための操作

4章では、個々のデバイスの状態を制御する操作について説明しましたが、ここでは、画面上でシーケンスプログラムを作成し、自動的に EQCON とのデバイスのやりとりをしながら装置の制御を自動でシミュレーションするための 操作について説明します。

プログラミングに使用するコマンドは簡易型で、ほとんどの操作はマウスとキー操作で行なうことができます。

保存されるシーケンスプログラムファイル名の拡張子は、".prg"です、

本機能は、I/0デバイス制御画面とともに実現されます。

5.1 開始とシーケンスプログラム画面

まず、5章の I/0 デバイス操作画面を開いた後に SEQ プログラムボタンをクリックすると始まります。

5.1.1 開始操作

(1) SEQ プ い 74 ボタンのクリックしてプログラム画面を開いて始めます。

1/0操作開始 SEQ70からム 1/0操作終了

、一/ 1/0定義ファイル名	参照	C:¥Ds	hGemLi	5¥PlcS	o. Sim¥sa	ample-	1.pls	I/C	操作開	1站	SE	EQ.7°07'54	1/0	影作終了	7	1/0定	競ファイル	編集	1/07	アイルギ	扁集談	×7	D.	が消去	Lot	; On
<mark>化</mark> 接点入力0	n/Off マゥ	ナ									×	<mark>℃</mark> 接点	出力の	∿/Off	マップ										_ []	×
ON OFF	🗌 選択 🔼	LoadEnd				•	AllOn	A	llOff	同期		ON	OFF	」選択	Y_Lo	adStar	t			•	AI	lOn	All	Off	同期	
X-addr 0	1 2 3	3 4	5 6	7	8	9	A B	C	D	E F		Y-addr	0	1 2	3	4	56	7	8	9	A	В	C	D	E F	ī.
0010					-	-						0010				_	_									-
0020				_			_					0020			_			_					_			-
0040												0040														
0050							_					0050		_	_										_	- 1
0070												0070											_			
0080							-					0080														- 1
0040												0040														
0000												0000														
0000							_					0000														
00F0	り シーク	シス・プ	ログラミン	אר															_ [⊐l ×	1					
0100	771I			1		. 1							実行						_							
0120	新規作	ūί	開く		引きなる	5 ड	上書的	¥存	別名	で保存		踩 了	Top[2]	5	実行	77:	20	OOms	• ß	产止						
0140	新乐 X_LoadE	nd		•	নম	力設定		LoadSt	art			▼ 出力器	定	行挿入	72	(ル挿入)	行削除	≩ Bee	ep Of	f 🖵						
0150	-37VM	1 27-57°	制御対	ー 象デル	。 *れ名			Dn/Off	目条件	になる	$\hat{\tau}^* h^* i$	 (1名	10n/	'Off (עב					1.	iH	_		_	_	
	ALLX	1																			1					I
<u>,</u>	ALLY	2																		_						_//_
	GOTO	4							_				_													
		5																								
	IFGO	6																								
	DELAY	<u>-</u> 7							_				_							_						
	PAUSE							-	1° H	े स्ट	i zi															
	WAIT	10)	u)	/4世	山田															
	Orvit	11						_のĪ	町面	上に	プロク	ブラミング	しま	す。						_						
	Garid	13																		-						
	<u> ' -</u>	14																								
	SlotMap	15																		_						
	1 💌	16							_				-							-						
	ProcData	18																								
	1 -	19																								
		20																			1					

(2) 画面は、次のようになります。



5.1.2 プログラムの新規作成/開く/開きなおす

プログラムの編集あるいは実行操作は、ファイル関連ボタンブロックの新規作成、開く、開きなおすボタンのクリ ックから始めます。



プログラムファイル名が確定している場合、プログラム画面のタイトルバーに開かれているファイル名が表示されます。

(1)新規作成

新規作成ボタンをクリックすると、そのとき表示されているプログラムの内容は全てクリアされ、入力設定位置がステップ-1 の1列目になります。

このあと、5.1.3に説明されている操作方法に沿ってプログラムの入力を行います。

そして、必要なタイミングで、別名で保存ボタンを使って画面上のプログラムをファイルに保存します。

(2)開く

開くボタンをクリックすると、ファイルを開くダイアログ画面がポップアップされ、".prg"のファイル名 が表示されます。

その中の1つのファイル名を選択して、開くボタンをクリックすると、選択されたファイルに保存されて いるプログラムがロードされプログラム設定グリッド上に表示されます。 この後、プログラムを編集または実行することができます。

(3) 開きなおす

開きなおすボタンをクリックすると、それまで開いたことがあるプログラムファイルのリストがポップア ップされた画面に表示されます。

リスト上に表示されているファイルの1つを選択し、**OK**ボタンをクリックすると、(2)の**開く**ボタン の場合と同様に選択されたファイルに保存されていたプログラムがプログラム画面に表示されます。 この後、プログラムの編集または実行をすることができます。

具体的な操作については、6章のサンプルプログラムについての説明を参照してください。



5.1.3 画面と操作ボタンの説明と操作

プログラム画面は次の通りです。

<mark>flc</mark> シーケ	ンス・プログラミング			
-7ァイル	も 聞く 聞きかおす	- 上書保友 別タで保友	実行 終了 I Topl: 東行	ろ 実行 7〒47º 2000ms - 位正
福集	6 1991 X 1993 C 4 4 6 3 3			
X_LoadEr	d 🔽 入力	設定 Y_LoadStart	▼ 出力設定 1	行挿入 ファイル挿入 行削除 Beep Off 🖃
-1771	ステップ 制御対象デバイス名	On/Off 条件になるテ	^{**} バイス名 On/C	אנאב) ff
ALLX				
ALLY	3			
GOTO	4			
IFGO	5			
DELAT	8	プログラム設定グリッド部		
PAUSE	9			
WAIT	10			
Carid	12			
	13			
	14			
SlotMap	16			
	17			
ProcData	18			
1 🖵	20			

(1)プログラム設定グリッド部

画面の表には、6列のフィールドがありますが、各フィールドの用途は次の通りです。

フィールト	見出し	意 味
1	ステップ	プログラムの実行ンーケンスのステップです。実行は、正順に進みます。
		ステップ番号は\$GOTO, \$IFGO のジャンプ先にもなります。
2	制御対象デバイス名	On/Off したいデバ れ名または制御コマンドを設定します。
		入力(出力)デバイス名選択コンボボックスと入力設定(出力設定)ボタンを使って設定
		します。
		制御コマント、の場合、左側のコマント、ボタンのクリックで設定されます。
3	0n/0ff	制御対象デバイスの 0n/0ff または制御コマンドのパラメータを設定します。
		デバイスの On/Off はこのthをゲブ ルクリックすることで On/Off を反転することが
		できます。
		\$DELAY コマンドでは時間値になります。
		\$GOTO, \$IFGO コマント の場合はジャンプ先ステップ番号になります。
4	条件になるデバイス名	制御対象デバイスを 0n/0ff にする際に他のデバイスの 0n/0ff 条件になるデバイス
		名を指定することができます。
		制御コマント、の場合にはパラメータに使用されます。
		また、\$WAIT コマンドの場合は待ち対象デバイス名設定に使用されます。
5	0n/0ff	条件になるデバイスの 0n/0ff 条件を設定します。
		On/Off 条件の場合は、ダブルクリックで反転させることができます。
6		注釈の欄です。プログラム実行上の意味はありません。



(2)入出力デバイス名選択コンボボックス

3. 1で説明した I/0 デバイス定義ファイルに登録された入出力デバイス名が入力、出力それぞれにコン ボボックスのリストに表示されます。

コンボボックスをプルダウンし、コマンドとして設定したいデバイス名を選択します。そして、入力設定 または出力設定ボタンを使って所望のステップと列にデバイス名を設定します。

細來 X_Port1RdyToLoad	-	入力設定 Y_PlcStart 🔍 出力設定	
X_Port1RdyToLoad X_Port1StartLoad	-	2 On/Off On/Off条件になる信号名	0
X_Port1LoadLnd X_Port1Clamped X_Port1FounOpered			
X_Port1FoupClosed X_Port1MovedCarWork			
X ⁻ Port1MovedCarDst	•		

[設定例]

ステップ	制御対象デバイス名	0n/0ff	条件になるデバイス名	0n/0ff	
1					
2	X_Port1StartLoad	On	Y_Port1RdyToLoad	On	
3			Y_Port1RdyLoadLamp	On	
4					
5	\$DELAY	5			
6	Y_Port1RdyToLoad	Off			

上の例は、Y_Port1RdyToLoad=On で、かつ、Y_Port1RdyLoadLamp=On の状態になったら X_Port1StartLoad=On にし、ステップが5 に進み、5 秒間遅延の後、ステップ-6 に進みます。 因みに、通常、条件になる出力デバイスの制御はEQCON が行ないます。 (ブランクのステップは無処理(no operation)となり、次のステップに進みます。)

(3) 制御コマンドボタン

制御コマンドの設定は画面左側に配置されている制御ボタンを使って行ないます。 コマンド名は、制御対象デバイス名の列に設定されます。 そして、コマンドパラメータはコマンドによって所定の列に設定することになります。 コマンド名は、大文字で、ボタンの表示名の頭に**\$**を付けたものになります。

制御対象デバイス名	On/Off	条件になるデバイス名	0n/0ff	機能
\$ALLX	0n/0ff			全入力デバイスを 0n/0ff する。
\$ALLY	0n/0ff			全出力デバイスを 0n/0ff する。
\$DELAY	$\langle t \rangle$			<t>秒間遅延します。</t>
\$GOTO	<step></step>			<step>位置に無条件ジャンプします。</step>
\$IFGO	<step></step>	<i 0="" デバイス名=""></i>	0n/0ff	<i 0デバイス名="">の 0n/0ff 条件で<step>シジャンプします</step></i>
\$WAIT		<i 0="" デバイス名=""></i>	0n/0ff	<i 0="" デバイス名="">の 0n/0ff 条件になるまで待機しま</i>
				す。
\$PAUSE				一旦停止します。
\$CARID	<index></index>			<index>(1~10)のキャリア ID を選択します。</index>
\$SLOTMAP	<index></index>		0n/0ff	<index>のスロットマップのチェックを変更します。</index>
\$PROCDATA	<index></index>	〈データ値〉		<index>(1~7)の値を<データ値>にセットします。</index>

(注) <データ値>は文字列データです。



(4) 設定操作例

たとえばプログラムの一部分になりますが、 X_{Fy7} -1から、順に、以下の制御を行なうことにします。 出力デバイス Y_Port1RdyLoadLamp1 と Y_Port1RdyToLoad が両方とも ON になったら、入力デバイス X_Port1StartLoad を ON し、10 秒間だけ DELAY するためのプログラムの操作は次のように行ないます。

①最初にマウスで、現在の入力位置をステップー1の制御対象デバイス名の欄にします。

編集 X_Port1R	dyToLo:	ad 💌	[2] 力設定]	Y	PlcStar	t	•	出力設定		行挿
-1771	ステッフ°	制御対象デルイ	7,2		On/Off	条件になるデバ	(7名		0n/	Off
ALLX	1	,	/							
	2	×	•							

②制御対象デバイス名 X_PortStartload を入力デバイス名のコンボボックス上で選択し、入力設定ボタンを クリックし、この入力デバイス名を現在の入力位置に設定します。

X_Port1R	dyToLo:	ad	▼ (入力設定)	Y	_PlcStar	·t	T	出力設定	(行挿)
-1721*	ステッフ゜	制御対象が	が収名		On/Off	条件になるデバ	(7)名		On/Off
ALLX	1	X_Port1Rds	/ToLoad		On				
	2								

3列目の 0n/0ff 欄には 0n が表示されます。

③次に、Y_Port1RdyLoadLamp=On が条件になるデバイス名に設定するために設定位置を同じステップの 条件になるデバイス欄にし、Y_Port1RdyLoadLampの出力デバイス名としてコンボボックス上で選択し、出 力設定ボタンをクリックします。すると、次のようになります。

X_Port1R	dyToLo	ad 💌	入力設定 Y	_Port1Rd	yLoadLamp	▼ 出力設定] 行挿)
-1721*	えテッフ゜	制御対象デバイ	ゐ	On/Off	条件になるデバイ	伝名	On/Off
ALLX	1	X_Port1RdyToL	oad	On	Y_Port1RdyLoadL	amp	On
	2						

④今度は、X_Port1StartLoad=Onの条件になる Y_Port1RdyToLoad=Onの条件を次のステップ-2 に設定します。

X_Port1R	dyToLo:	ad 💽 入力設定 Y_	Port1Rd	yToLoad 🗾 🖬 刀滚落	[] 行挿)
-1721*	ステッフ。	制御対象デバム名	On/Off	条件になるデバイス名	On/Off
ALLX	1	X_Port1RdyToLoad	On	Y_Port1RdyLoadLamp	On
	2			Y_Port1RdyToLoad	On
ALLY	3				

(注) これで、Y_Port1RdLoadLamp=On と Y_Port1RdyToLoad=On になったら X_Port1StartLoad=On にするためのプログラムの入力が終りました。

⑤そして、ステップ-3の制御対象デバイス名の位置をクリックし、DELAY ボタンをクリックします。

ステップ 制御対象デバイス名 On/Off 条件になるデバイス名 On/Off ALLX 1 X_Port1RdyToLoad On Y_Port1RdyLoadLamp On 2 Y_Port1RdyToLoad On Y_Port1RdyToLoad On	編集 X_Port1R	dyToLo	ad 💌 入力設定 🛛	_Port1Rd	lyToLoad 🗾 💌	出力設定	(行挿)
ALLX 1 X_Port1RdyToLoad On Y_Port1RdyLoadLamp On 2	-1721*	ステッフ°	制御対象デルに公名	On/Off	条件になるデバイス名		On/Off
Y_Port1RdyToLoad On	ALLX	1	X_Port1RdyToLoad	On	Y_Port1RdyLoadLamp		On
		2			Y_Port1RdyToLoad		On
ALL I 3 \$DELAY 1	ALLY	3	\$DELAY	1			

この後、入力位置が時間地入力位置(3列目)になります。この値"1"を キー操作で"10" に変えます。

これで例題プログラム入力操作の終了です。



5.2 編集と保存

(1) 編集

編集ボタンブロックを使って操作します。

١	*E.K.						
	X_Port1RdyToLoad	▼ (人力設定)	Y_PlcStart	▼ 出力設定	行挿入	7月11挿入	行削除

4. 1のプログラミング操作の中で、コマンド行の挿入、削除などの編集作業が必要になります。

コマンド行の挿入は、行挿入ボタン、コマンド行の削除は、行削除ボタンのクリックで行ないます。行挿 入ではブランク行が挿入されます。また、行削除では、そのとき選択されているステップのコマンド行を 削除し、後ろのコマンド群を前方向にプルアップします。

この際、\$GOT0, \$IFG0 コマンドのジャンプ先は、挿入、削除によって変わる可能性がありますが、そのジャンプ先の調整はシミュレータによって自動的に行われます。

既に作成されているファイルを行の間に挿入したい場合は、7744挿入ボタンを使用します。この場合の挿入によって影響を受ける\$Go, \$IFGOのジャンプ先ステップの調整も自動的に行なわれます。

(2)保存形式と保存

ファイルボタンブロックのボタンを使用します。

-79-11。 | 棄現作成 | 関く | 関きなおす | 上書保存 | 別名で保存 | 終了 |

画面上にプログラミングされたプログラムをファイルに保存することができます。そして、保存したプロ グラムを再度開いて使用することができます。

プログラムファイルは、拡張子.prg で保存されます。

形式は、プログラム画面の表に表示されているステップ順に、2列目から6列目までのセルの内容をカンマ切りされて保存されます。

下に、サンプルを示します。

\$ALLX,	0ff,	,	,	全Input signal = Off,
\$ALLY,	0ff,	,	,	全Output signal = Off,
\$SLEEP,	3,	,	,	,
2	,	,	,	,
\$CARID,	1,	,	,	キャリア ID 先頭(CARID_01),
,	,	,	,	** ここがLoop の起点,
X_Port1RdyToLoad,	0n,	Y_Port1Rdylamp,	0n,	Y_Port1RdyLamp = On 待ち,
,	,	,	,	On になったら X_Port1StartLoad=On,
X_Port1StartLoad,	0n,	,	,	,
Y_Port1RdyLoadLamp,	0ff,	,	,	,
,	,	,	,	,
\$DELAY,	2,	,	,	,
,	,	,	,	,
X_Port1Clamped,	0n,	,	,	,
\$DELAY,	4,	,	,	,
X_Port1StartLoad,	0ff,	,	,	,
,	,	,	,	,
\$WAIT,	0n,	Y_Port1Foup0pen,	0n,	,
,	,	,	,	,
X_Port1Foup0pened,	0n,	,	,	,
Y_Port1Foup0pen,	0ff,	,	,	,
1				



5.3 シーケンスプログラムの実行と停止

実行はボタンブロックのボタンをクリックして行ないます。

| 実行 | Topに戻る|| 実行 | ステップ* | 2000ms | ズ 停止 | /ンターハ・ル

(1) 実行

実行開始は、画面上の開始したいステップを選択し、実行ボタンのクリックで開始します。 実行中は、コマンドとコマンドの実行にインターバルをとることができます。 ステップボタンの右横にある インターバルコンボボックス(単位は ms)の値を選択して実行します。。このことによって、実行進行状態を目で追従、確認しながらテストすることができます。

ステップ ボタンを使うとコマンドを1個ずつステップ順に実行できます。

EQCON との通信が必要なコマンドの実行の条件として、DSH-PLCSim と EQCON との間で TCP/IP 通信接続が確立していることが必要です。接続しているかどうかの確認は、メニューバーの下のランプで確認できます。



(2) 停止

プログラムの実行停止は、一旦停止も含め、次の1つのことによって停止します。

①プログラムが終端に達したとき。
 ②停止ボタンがクリックされたとき。
 ③\$PAUSE コマンドに達したとき。



6.入出力デバイスの制御プログラムサンプル

実際のプログラミング操作について簡単な例を挙げて説明します。

6.1 I/O デバイス定義と制御シーケンス

この制御例についてはプログラムサンプルであって、制御内容については特に意味はありません。

(1) I/0 デバイスの定義を次のようにします。

X-addr	入力デバイス名	Y-addr	出力デバイス名
0000	X_LoadEnd	0000	Y_LoadStart
0001	X_MovedSrc	0001	Y_MoveSrc
0002	X_FoupOpened	0002	Y_OpenFoup
0003	X_MovedWork	0003	Y_MoveWork
0004	X_ProcessStarted	0004	Y_StartProcess
0005	X_ProcessEnd	0005	Y_MoveDst
0006	X_MovedDst	0006	Y_CloseFoup
0007	X_FoupClosed	0007	Y_UnloadStart
0008	X_RdyUnload	0008	
0009	X_UnloadEnd	0009	

(2) EQCON との間の制御シーケンス

PLC	デバれのやり取り	EQCON	注釈
Delay(3)	•	Y_LoadStart = ON	Load 開始
X_LoadEnd = ON			Load 完了
Y_LoadStart = OFF			
Delay(3)	•	$Y_MoveSrc = ON$	Src へ移動
X_MovedSrc = ON			同完了
	←	Y_OpenFoup = ON	Foup 開
$X_FoupOpened = ON$	▶		同完了
	←	Y_MoveWork = ON	Workへ移動
X_MovedWork = ON	▶		同完了
Delay(5)	•	Y_StartProcess = ON	処理開始
X_ProcessEnd = ON			処理終了
Delay(3)	•	$Y_MoveDst = ON$	Dst へ移動
$X_MovedDst = ON$	├ ─── ▶		同完了
		$Y_CloseFoup = ON$	Foup 閉
$X_FoupClosed = ON$		•	同完了
X_RdyUnload = ON	►		
Delay(3)	<	Y_UnloadStart = ON	Unload 開始
X_UnloadEnd = ON			同完了
PAUSE			
全入力デバイス = OFF	┣────►		
全出力デバイス = 0FF			



6.2 プログラム作成

プログラム入力操作の前に、以下の準備操作をしてください。

- (1)一旦、I/0操作終了ボタンをクリックして、メイン画面を初期状態に戻します。
- (2) 6. 1の(1)のデバイス定義情報を3章で説明した操作によって I/0 定義ファイルに保存します。 このファイル名を sample-1.pls にしてください。
- (3) メイン画面に戻り、 参照 ボタン I/0 定義ファイル名の選択を sample-1.pls にした上で、I/0 操作開始ボタン で I/0 制御画面にしてください。
- (4)次に、SEQアログラムボタンのクリックで、シーケンスプログラム画面を表示します。
 そして、新規作成ボタンをクリックしてください。
 5.1.3で説明したプログラム設定操作の方法に従って、6.1の(2)の制御シーケンスのプログラム内容を画面上に設定します。
 全て、入力すると画面は次のようになります。

<mark>れこ</mark> シーケン	パウ	コグラミング - C:¥DshGemLib	¥PlcSim	n¥sample−1.prg			
-7711- 新想作时		四// (2015-2015-2017-201-1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-1	現方	- 実行 別々で保方 実行	년 에드로 지	₩ /元 75 × 12 2000ms - //	÷
- 新加工FA	<u> </u>		17417		picærø		
X_Port1R	dyToLo:	ad 🔍 入力設定 Y	Port1Rd	lyToLoad 🚽 出力設定	: 行推	●入 ファイル挿入 行削除 Beep Off	F 💶
	22.20		10-/066	<u></u>	10- /066		
	<u>λ/97</u> 1	መባወቅኦነው// // 1አላጋ ቁሠለተተ		V LoadStart	05	→//r V LoadStart 待ち	
	2	\$DELAY	3		on		
ALLY	3		0n				
<u>дото</u>	4	\$WAIT		Y MoveSrc	0n		
	5	\$DELAY	3				
IFGO	6	X_MoveSrc	On				
DELAY	7	\$WAIT		Y_OpenFoup	On		
DAUGE	8	X_FoupOpened	On				
PAUSE	9	\$WAIT		Y_MoveWork	On		
WAIT	10	\$DELAY	5				
	11	X_MovedWork	On		-		
Carid	12	\$WAIT		Y_StartProcess	On		_
1 -	13	X_ProcessStarted	Un				_
	14	SDELAY	5				_
SlotMap	10	A_ProcessEnd	Un	V. Maus Dat	0-		_
1 🚽	10	ΦΝΑΤΙ ΦΝΑΙΙ	0	T_MOVEDSt	UN		_
Durabati	17	WovedDat	0 00				-
ProcData	19	\$WAIT	on	Y CloseFour	0n		-
	20	X FoupClosed	On				
	21	X RdyUnioad	0n				
	22	\$WAIT		Y_UnloadStart	On		
	23	\$DELAY	3				
	24	X_UnioadEnd	On				
	25	\$PAUSE				一旦停止	
	26	\$ALLX	Off			全入力Off	
	27	\$ALLY	Off			全出力Off	
	28	\$GOTO	1				
	29						-

(5) 画面上でのプログラミングが終ったら、それを**別名で保存**ボタンをクリックし、sample-1. prg として保存 します。(途中で保存しながら入力することを奨めます)



6.3 実行操作

前節で入力したシーケンスプログラムを実行させます。

ここでは、このサンプルプログラムの相手をする EQCON(装置コントローラ)がありませんので、その代わりに出力 デバイスの ON/OFF 操作を I/O 操作画面上でマウスを使ってマニュアルで行なうことにします。

<mark>℃</mark> 接点出力On/Off	マップ										_			
ON OFF 選拐	Y_LoadS	Start				-	AI	l0n	AI	IOff	[ī	司期		
Y-addr 0 1 2 0000 - 0010 0020 0030 0040	3 4 2 ON ま クリックしま	5 たは ます。	6 – OFF	7 - * 92	8	9	A	B	に ① を述	D 0N/0 選択	E)FF しま	F したい す。	デバイ	风名

Y-addr 0000~0007 まで定義されています。サンプルプログラムでは便宜のため、シーケンスでは、0000 から順に 0007 まで ON にしながら進行することになっています。 (入力側ではプログラムが 0000 から順に ON にセットしていきます。)

- (1)まず、マウスで開始位置を ステップ-1 にします。(Top に戻るボタンを使うこともできます。)
- (2) 次に実行ボタンをクリックします。 これによって、ステップー1の\$WAIT Y_LoadStart=0Nが実行され、Y_LoadStartが0Nになるのを待機します。
- (3) 出力デバイス画面のデバイス名コンボボックスから、Y_LoadStart を選択し、ON ボタンをクリックします。 このとき、Y_LoadStart のデバイス番号が = 0000 ですので、そのデバイスに対応した位置が ON 表示(O) になります。

ON	OF	F_」词	劉 沢	Y_Lo:	adSta	art				-	A	lOn	A	llOff	F [司期	
Y-addr	0	1	2	3	4	5	6	7	8	9	A	В	C	D	E	F	
0000	0	-	-	-	-	-	-	-									-

そして、プログラムは Y_LoadStart=ON (○表示)なったので、ステップ−2 に進み、\$DELAY 3 (3 秒間遅延)を 実行します。

(4) 3秒経過後、ステップ-3の X_LoadEnd = 0N を実行します。入力デバイス側の画面も次のようになります。

<mark>化(</mark> 接点)	ኢታ	0n/(Off ₹	マップ											_		×
ON	OF	F 浅	¥択 [X_Lo	adEn	l				•	A	llOn	8	not	6	司期	
X-addr	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
0000	0	-	-	-	-	-	-	-	-	-							
0010																	_

X_LoadEnd のデバイス番号は 0000 です。その位置が 0N (○)表示になります。そしてプログラムはステップ -4の\$WAIT Y_MoveSrc=0N を待機します。

(5)以下、(3)同様の操作方法で実行を進めます。

ステップ-25 に達したら、\$PAUSE コマンドで実行が停止します。 そのときの I/0 デバイスの状態はそれぞれ次のようになります。

<mark>凡c</mark> 接点)	ኢታ	0n/(Off ₹	マップ											J	_ 🗆	×
ON	OF	F_] រ៉	¥択	X_Lo	adEnd	1				-	A	llOn	A	llOff		同期	
X-addr	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	i 🔺
0000	0	0	0	0	0	0	0	0	0	0							
0010																	



<mark>℃</mark> 接点!	出力	0n/(Off ₹	マップ											_		×
ON	OF	F_kä	閨択│	Y_Lo	adSt	art				-	A	llOn	A	llOf	E F	司期	1
Y-addr	0	1	2	3	4	5	6	7	8	9	A	В	C	D	E	F	Ē
0000	0	0	0	0	0	0	0	0									Г
0010																	ш

また、プログラム画面は、ステップー26(\$PAUSEの次)が次の実行ステップになります。

<mark>れ</mark> シーケン	ノス・プロ	コグラミング - C:¥DshGemLib¥	∮PlcSim	¥sample-1.prg	_		×
-7ァイル 新規作成		聞く 聞きなおす 上書	保在	男子で保存 第1 第1	に戻る	宝行 (27777751 2000ms - 使	ıF.
编集			121-1-2				
X_Port1R	dyToLo	ad 🔻 入力設定 Y_	Port1Rd	lyToLoad ▼ 出力設定	行揖	i入 ファイル挿入 行削除 Beep Off	.
-172N	77.7%		05/066		105/0ff		
	1	መጣመዶጎ380/ // 1840 \$WAIT	017011	V LoadStart		↓ JoadStart结ち	⊢
	2	\$DELAY	3		on		┼──┚
ALLY	3	X LoadEnd	° On				-
сото I	4	\$WAIT		Y MoveSrc	On		-
	5	\$DELAY	3				
IFGO	6	X_MoveSrc	On				-
DELAY	7	\$WAIT		Y_OpenFoup	On		
DAUCE	8	X_FoupOpened	On				
FAUSE	9	\$WAIT		Y_MoveWork	On		
WAIT	10	\$DELAY	5				
	11	X_MovedWork	On				_
Carid		\$WAIT		Y_StartProcess	On		-
1 -	13	X_ProcessStarted	Un				-
	14	SVELAT V. Ducesse Feed	5 0-				-
SlotMap	10	A_PROCESSENC	UN	V. MeusDet	0-		-
1 🖵	10		0	T_MOVEDSt	UN		-
Prophetal	18	WovedDst	o On				-
FrocData	19	\$WAIT	011	Y CloseFour	Ωn		-
	20	X FoupClosed	On				-
	21	X RdyUnload	On				
	22	\$WAIT		Y_UnloadStart	On		1
	23	\$DELAY	3				1
	24	X_UnioadEnd	On				
	25	\$PAUSE				一旦停止	
	26	\$ALLX	Off			全入力Off	
	27	\$ALLY	Off			全出力Off	
	28	\$GOTO	1				

(6) この状態で、実行ボタンをクリックすると\$ALLX Off、\$ALLY Off を実行した後、ステップ-1 に分岐します。
 \$ALLX Off、\$ALLY Off は定義されている全デバイスの状態を Off にするためのコマンドです。
 以下、(3)の操作を繰り返します。



7.キャリア ID 画面操作とプログラミング

I/Oデバイス制御開始時、キャリア ID リーダー画面が同時に表示されます。 Carrier ID、Slotmap、ProcessData の3種類の情報が表示される画面があります。



(1) Carrier ID - キャリア ID 画面とプログラム

Carrier ID は仮想キャリア ID リーダの役割をします。 キャリア ID 候補の ID 変更と選択の方法は上図の説明の通りです。 このキャリア ID は、EQCON から読出し関数を使って読むことができます。 UPLC_CarldRead () 関数を使います。

また、シミュレータ側では、シーケンスプログラムの中で、\$CARID コマンドを使ってキャリア ID の候補から1個選択し、トップ位置の選択 ID 位置に表示させることができます。

\$CARID 〈候補番号(1~10)〉





(2) スロットマップ画面操作とプログラミング

先の画面左に説明したとおりですが、25スロット分のスロットマップ情報についてもプログラムで設定、 読出しすることができます。

EQCON からは、UPLC_SlotmapRead() 関数を使って、一度に25スロットのウェハーの在荷マップ情報を読み出すことができます。

シーケンスプログラムにおいては \$SLOTMAP コマンドを使ってスロット単位でウェハ有無のチェックを設定することができます。



(3) プロセスデータ

ProcessData については、7個の文字列データをキーまたはシーケンスプログラムにより設定できます。 そして、それを EQCON から UPLC_ProcDataRead() 関数を使って読み出すことができます。

シーケンスプログラムでは、\$PROCDATA コマンドを7個の中の1個に対し、データを設定変更することができます。

\$PROCDATA-i <文字列データ>

i番目のプロセスデータに文字列データ>を設定 (i=1~7)





8. その他の操作

DSH-PLCSim についてここまでまだ説明していないログ画面などの全般的な操作について説明します。

8.1 ログ表示とログファイルについて

ログ表示は、PLC Operation Log 画面(い) 画面)に行なわれますが、主に以下の操作、デバイス制御情報が表示されます。

- 1. DSH-PLCS im の起動情報
- 2. I/0 制御画面のデバイス On/Off 操作情報
- 3. シーケンスプログラム実行情報
- 4. EQCON との通信関連ログ情報

画面に表示された情報は、同時にログファイルの中にもタイムスタンプ付きで記録されます。

8.1.1 画面のログ表示

ログ画面に関連する操作として以下のものがあります。

(1) ログ画面へのログメッセージ表示の消去と表示 On/Off 操作

メイン画面のツールボタンバーの右端にあるボタンとLogOn/Off コンボボックスを使って行ないます。

- ・
 の
 が
 消去
 ボタンを
 クリック
 すると、
 ログ
 ・
 画面に
 表示
 されている
 内容を
 全て消去します。
- ・LogOn/Off コンボボックスの LogOff を選択すると、その後、ログ表示は一切行なわれなくなります。 ただし、ログファイルには記録されます。LogOn に戻すと再びログメッセージは表示されます。

8.1.2 ログファイル

ログ情報は画面に表示されるとともに、ログファイル上にもタイムスタンプ付きで記録されます。 タイムスタンプは YY-MM-DD hh:mm:ss の形式です。YY: 西暦下2桁、MM:月、DD:日、hh:時、mm:分、ss:秒です。

(1) ログファイル名

デフォルトのログファイル名はDSHP1cSim. exe が保存されているディレクトリに dshp1c. log の名前で作成されます。

メインメニューの**ログファイル名(L)**タブのクリックで、名前を付けて保存のダイアログ画面が表示されますので、その画面上で、別のログファイル名を指定することができます。

(2) ログファイルのサイズと世代ファイル管理

ログファイルに際限なくログ情報を書込む訳にいかないので一定の行数に達したら、ひとつ古いファイル の名前に変えて保存し、正規のファイルを一旦空にし、また先頭から書込むようにします。

ログファイルのサイズ : 約100、000行

残しておく過去のファイル : 3世代

過去の世代ファイル名には、正規のログファイル名の後ろに "_i"(i=1,2,3 世代番号)を付けて保存しま す。世代番号は若い番号の方がより新しいファイルになります。

例えば、正規のファイル名が dshplc.log の場合、dshplc_3.log が最も古い過去のファイルになります。



8.2 EQCON との接続情報の設定

シミュレータと装置コントローラとの接続と通信は、Ethernet を使った TCP/IP プロトコルで行ないます。

装置コントローラ側がサーバーで、シミュレータがクライアントの立場で接続します。対抗でもいいですし、LAN ネットワークの中での接続も可能です。



通信接続情報として、DSH-PLCSimは、EQCONのIPとポート番号が必要になり、EQCON側では、ポート番号が必要になります。

例えば、TCP ポート番号が 5930 で、EQCON の IP が 192.168.1.30 の場合には、シミュレータ側では、この情報を設 定する必要があります。

(1) DSH-SIMPlc での通信設定の方法

メインメニューの通信設定(S) タブのクリックで設定画面を開いて設定することができます。

<mark>የር</mark> ፐርቦ/	/IP通信の設定	
IP	192.168.1.2	
port	5930	
	設定	閉じる

この画面で、EQCONのIPとポート番号をキー入力して設定ボタンのクリックするだけです。

(2) EQCON との接続タイミング

EQCON との接続は、I/O 操作開始ボタンが最初にクリックされたときに行なわれます。

接続が完了すると、I/0操作開始ボタンの右側のみランプが緑色に点灯します。

1/0操作開始	SEQ7"ቢን"ትሬ	1/0操作終了
wis 11 - Maxim	/ H/ /M	

同時に接続された旨のメッセージがログ画面に表示されます。

(3) 切断のタイミング

I/O 操作終了ボタンがクリックされたときに EQCON との間の通信を切断します。このとき緑のランプが消えます。



8.3 I/O デバイス定義画面の文字フォントとセルサイズの変更

I/0 デバイス定義画面の定義画面上の文字フォント、サイズ、色を変更することができます。 また、I/0 デバイス名を設定するセルのサイズも必要に応じて変更することができます。 この操作は、I/0 デバイス定義画面が開かれているときだけできます。

(1) フォントの変更設定操作

メインメニューの表示(E)の中の編集画面の7パン設定(E)をクリックします。

<mark>ሺር</mark> DSH-	PlcSim-2	2008–1 Tr	ial S/N : 121	2630205-T	- effective	e 180 days (
ファイル(E)	編集(E)	表示(<u>D</u>)	通信設定(S)	ウィンドウ(W)	ログファイル名	ふし ヘルプひ	
1/0定義7;	加名 📕	編集画 1/0編集	面のフォント設定(画面のCellサイ)	E) X設定(S)	1.pls	□操作開始	
				- 100 E			1
<mark>ी (</mark> Input	t Bit Dev	ice Defin	ition Editor -	- C:¥DshGei	mLib¥PlcSi	m¥sample-1	
X-addr	0		1		2		ĺ
0000	X_LoadEnd		X_MoveSrc		XFoupOpe	ned	
001.0							

次のフォント設定画面がポップアップしますので、ここで、望みのフォントを選択し、OK ボタンをクリックします。

フォント			<u>? x</u>
フォント名(E): MS P明朝 MS Sans Serif MS Serif TM MS UI Gothic TM MS UI Gothic TM MS UI MS UI MS UI MS UI MS UI MS UI MS UI MV Boli	スタイル\℃: 標準 標準 斜体 太字 太字 斜体	サイズ(S): 8 9 10 11 12 14 16 ▼	OK キャンセル
文字師り □ 取り消し線(k) □ 下線(U) 色(<u>c</u>): ■ 黒 ■	-サンブル	P	

(2) I/0編集画面のCell サイズ設定

上の表示メニューの I/O 編集画面の Cell サイズ 設定 タブをクリックします。

┡╻。編集画面のセル(₩、Η)サイス* 🔲 🗖 🗙
編集をルの高さ、幅の設定
幅と"ウセル)
高さとりセル) 16
,
確定 キャンセル

1 個のデバイス名入力用セルの縦横のサイズをピクセル単位で設定し、確定ボタンで即時画面のセルサイズが変わります。