# IoT-Monitor

# 一取扱説明書-

# Rev. 1.91

IoT-Monitor Ver.1.20 対応

# 目次

1.	IoT-Monitorとは	3
2.	動作環境	5
3.	画面説明	6
4.	ファンクションの設定	8
	1.カウント	. 10
	2.インジケータ	12
	3.ストップウォッチ	. 13
	4.ログ保存	14
	5.IO 一括ログ保存	. 16
	6.メール	. 18
	7.ランプ	. 20
	8.警告画面	21
	9.サウンド	. 22
	10.クラウドカメラ連携	. 23
	11.写真	25
	12.ビデオ	26
	13.ファイル実行	27
	14.PDF 表示	28
	15.パワーポイント	. 29
	16.マクロ	. 30
	17.コントローラ→エクセル出力	32
	18.コントローラ→共有メモリ出力	. 34
	19.カレンダー指定	. 36
	20.ボタンスイッチ	. 37
	21.トグルスイッチ	. 38
	22.バーコード	. 39
	23.コントローラ連携	. 41
	24.エクセル→コントローラ出力	42
	25.共有メモリからコントローラ出力	. 43
5.	固定テキスト	. 45
6.	オプション	. 55
	1.自動起動	. 55
	2.タイムアウト設定	. 56
	3.言語設定	. 56
	4.通信切断時のメール送信設定	. 57
7.	改版履歴	. 58

# I IoT-Monitor とは

IoT-Monitor は SiOt シリーズや SiO-X、MiO 等のコントローラと、信号のやり取りをする PC ソフトです。Ethernet 通信を行い、PC とコントローラを連携できます。

コントローラの IO 状態を PC に保存したり、PC からコントローラのランプを光らせたりすること が可能です。

利用できる効果(ファンクション)は以下の通りです。

- >> □ コントローラから PC へ
- ・接続されたコントローラの IO 状態を画面表示
- ・特定の IO 状態になった際に、PC でデータ保存などをおこなう
   例:ワークがセンサを通過した時間を、PC で記録する



- << □ PC からコントローラへ
- ・特定のタイミングで、PC からコントローラに信号を送信 例:バーコードリーダで特定のバーコードをスキャンすると、棚のランプを光らせる



■SiO-Programmer/MiO-Programmer とは

SiO(MiO)-Programmer は、コントローラにプログラムを登録する PC ソフトです。 条件を選択するだけで、SiO(MiO)のプログラミングが可能です。 ※別途ダウンロードが必要です。

コントローラ本体の IP アドレスなどの設定も SiO(MiO)-Programmer で行います。 IoT-Monitor から送信される Etherflg は、SiO(MiO)のプログラム内で 「条件」として設定することができます。

・SiO-Programmer の場合

70:	ジェクト名 Ne	ewPrj					C	in	<b>D</b>		
出力	条件設定> [	プログラム初期化					3		Pre	DĆ	JI
	山 山 山 のMの条件										
	11/1	1			2			壮	忧 態		
	(191)	IN1	ON	さらに	IN2	OFF		すると	直接值	3.)	0 秒
	OUT1	Ether1 (ポタン1)	ON	または		_		ಕると	直接値	0.0	* *
	OUT2		-	-	-	IN	•	すると	直接値	0.0	Å
	OUT3	- ]	-	-	-	OUT		ಕನ೭	直接値	0.0	Å
	OUT4	- ]	-	-	-	複数選択		すると	直接値	0.0	Å
	OUT5		-	-	-	RUN		ಕತ೭	直接値	0.0	Å
	atuo.	<u> </u>				Ether	•	Ether1(オ	タン1)	l	
	FLAG1	- ]		-			_	Ether2(개	タン2)		Å
	FLAG2	-	-	-	-	-		Ether3(M	"ルホ"タン1)		×

#### ・MiO-Programmer の場合

		Luiei14
MiO		Ether15
▼出力条件設定	Ether16	
		Ether17
出力		Ether18
	1 2	Ether19
G-FLAG1		Ether20
		Ether21
G-FLAG2	G-FLAG 🕨 – –	Ether22
	MiO-IO	Ether23
G-FLAG3	RUN •	Ether24
G-FLAG4	複数選択	Ether25
	Ether → 1~8 →	Ether26
G-FLAG5	1P_1N ▶ 9~64 ▶	Ether27
	ID.1 •	Ether28
C-ELAC6		

詳細は SiO(MiO)-Programmer の取扱説明書を参照ください。

2 動作環境

IoT-Monitor を動作させるためには、以下の環境が必要です。

# ■対応するパソコン機種

本ソフトが動作する事を確認した機種は以下の通りです。

Windows 7	(32 / 64Bit 版)
Windows 8	(32 / 64Bit 版)
Windows 8.1	(32 / 64Bit 版)
Windows 10	(32 / 64Bit 版)
Windows 11	

が動作する機種

※ 上記 OS であっても、機種によっては正常に動作しない場合もあります。
 ※ Windows は、米国 Microsoft Corporation の米国及びその他の国における登録商標です。

# ■CPU&メモリ

**800MHz** 以上の CPU、512MB 以上のシステムメモリを推奨 拡張メモリ 512MB 以上を推奨

# ■ハードディスク空き容量

空き容量 100MB 以上

■ディスプレイ

解像度 1280×768 以上 カラー256 色以上

# ■その他

・LAN ポート LAN ポートがない場合、USB-LAN 変換アダプタを使用してください。

・MiO コントローラを接続する場合、 コントローラのバージョンが 1.30 以上である必要があります。

# 3 画面説明

以下の手順で通信を行います。

- 1.「②コントローラ登録」でコントローラを追加し、「⑧コントローラ設定」でコントローラの 情報を設定する。
- 2.「⑨ファンクション追加」でファンクションを追加し、設定する。
- 3.「③接続」ボタンをクリックし、通信を行う。
- 4.「⑬通信状況表示」を見ながら、正しく接続されているか確認する。

■初期画面

📎 IoT Programmer v130β		
ファイル オブション コントローフ登録 2010 1988 1988		(上) 更新時間 0 ms (4) (5)
	6	
۵۶ 👼	$\boxed{7}$	۲ ۲

1)		すべてのコントローラとの通信とファンクション処理にかかった時間					
	更新時間	です。値が大きい場合、回線が安定していない可能性があります。					
		PCとSiOtを直接接続した場合、10ms以内に応答があります。					
2		コントローラを追加します。現在のネットワークから検索する方法					
	コントローラ登録	と、アドレスやポートなどを直接指定する方法があります。					
		※コントローラの最大接続台数は 50 台以下を推奨しています。					
3	接続	設定されたコントローラで通信を開始します。					
4	一括設定	カウンタやインジケータ、ログ保存などの設定ができます。					
5	固定テキスト	固定テキスト機能を使用できます。→p.44					
6	タコントロニラ乳空	「②コントローラ登録」でコントローラを追加し、ここで設定を行い					
	谷コンドローノ設定	ます。					
$\overline{7}$	ыĂ	ログ記録のメッセージやコントローラ接続エラーなどのログを表示し					
	L 2	ます。					

	■コントローラ設定画面									
	NoT Pro	ngrammer v170								
8	) Si001	機器ダイブ Sio コントローラ名 SOO1 PPアドレス 192.186	10.100 #-HNo. (4001) 武臣 新臻 -场石							
9	④ + ○ NUI ・ の状態をジンプ表示 ON 赤 ・ OFF グレー・ ランプ名 表示  ※ NUI ・ の状態をジンプ表示 ON 赤 ・ OFF グレー・ ランプ名 表示									
	<u>الم</u>									
	8	コントローラ乳ウ	コントローラの機器タイプ、名前、アドレス、ポート No							
		コントローノ設定	などを設定します。							
	9	ファンクション追加	ファンクションを追加します。							
	10	ファンクション一覧	追加したファンクションを設定します。							
	11		sio ファイル(SiO-Programmer で生成)や mio ファイル							
		メモファイル読み/削除	(MiO-Programmer で生成)を読込み、メモ情報を反映させ							
			ることができます。読込んだ状態でファイル保存(.iot)した							
			場合、メモ情報も含めて保存されます。							
	12		一部のファンクション(ランプ/カウント/インジケータ/ス							
		一括モニタ	トップウォッチ/ボタンスイッチ/トグルスイッチ)で表示							
	される画面を、コントローラごとに一括で表示できます									

#### ■通信接続時の画面

📎 IoT Pr	ogrammer v160β	RE-Fille して「ファイル最後年」		
ファイル コントロー			<b>()</b> 更新時間 10 ms	
S S	001 概器 111 トローラ名 SiO01 1	РРКИХ 192.168.0.100 Ж-FNo.	4001 【読】【読】 70-E3-D5-1 E-D0-00	
Ę	): [IN01   * の状態をランプ表示 ON 赤	● OFF グレー ▼ ランプ名		
د ق ال	]			
			*	
13	5	接続動作中	コントローラとの通信動作を行っています。 接続できているかは、個タブのアイコンで	, 判
			断します。	
14)		通信状況表示	各コントローラとの通信が正常であるかど	う
	🛜 SiO01 🗙 SiO02		か、谷タフのアイコンで判断します。 	-
			👕 囲宿中 🛛 木 通信矢敗・接続リトフィー	4

SUS Corporation

# ↓ ファンクションの設定

>> □ コントローラから PC へ

ファンクションは、すべてのコントローラの設定を合わせて512まで追加できます。



No.	アイコン	機能	内容
1	562	カウント	コントローラの IO 状態を、PC 画面上でカウント表示します。 例 : センサ 3 が ON になった回数を、PC 画面に表示
2		インジケータ	コントローラの IO 状態をインジケータ形式でカウントします。 例:ボタン1が ON になった回数を、PC 画面上にインジケータ形式で表示
3	Q	ストップ ウォッチ	コントローラが条件を満たしている時間を計測します。 例:センサがONになっている秒数を計測し、csvファイルに保存
4		ログ保存	コントローラの IO 状態を、csv 形式で保存します。 例:センサ1が ON になった日時を、csv ファイルで出力
5	Elle	IO 一括保存	コントローラが条件を満たすと、IO状態を一括保存します。 例:非常停止ボタンが押された前後のIO状態を、csvファイルで保存
6	$\square$	メール送信	コントローラが条件を満たすと、メールを送信します。 例:非常停止スイッチが ON すると「sus@example.com」 宛てにメールを送信
7	Ŵ	ランプ	コントローラの IO 状態を、PC 画面上にランプで表します。 例:入ロランプの ONOFF 状態を、PC の画面上で表示
8	:	警告画面	コントローラが条件を満たすと、PC上に警告画面を表示します。 例:非常停止スイッチが押されたら、PC画面上に警告画面を表示
9		サウンド	コントローラが条件を満たすと、PC でサウンドを鳴らします。 例:呼び出しボタンが押されたら、PC で電話の音を鳴らす
10	<b>F</b> safie	クラウド カメラ	コントローラが条件を満たすと、Safie のクラウドカメラに アクセスできる URL の情報を csv 形式で保存します。 例:チョコ停が発生した時間を記録し、csv ファイルからカメラ映像を確認する
11	<u> </u>	写真	条件を満たす前後の画像を PC に接続されたカメラで撮影します。 例:ボタンを押すと、作業スペースの画像を保存
12		ビデオ	条件を満たす前後の映像を PC に接続されたカメラで撮影します。 例:非常停止スイッチが押される前後の数秒間を録画し、ビデオファイルに保存
13		ファイル 実行	コントローラが条件を満たすと、ファイルを実行します。 例:スイッチ1がON すると、取扱説明書の pdf を表示

No.	アイコン	機能	内容
14	PDF	<b>PDF</b> 表示	コントローラが条件を満たすと、PDFを操作します。 例:ボタン1をONすると、PDFが次へ進む
15	P	パワー ポイント	コントローラが条件を満たすと、パワーポイントを操作します。 例:ボタン1をONすると、スライドが次へ進む
16	ox I	マクロ	コントローラが条件を満たすと、Excel マクロ関数を操作します。 例:ボタン1をONすると、エクセルファイルにカウント1の値が入力される
17	×	コントローラ →エクセル 出力	コントローラが条件を満たすと、Excelのセルに値を出力します。 例:ボタン1をONすると、エクセルファイルのA1セルに1が入力される
18		共有メモリ へ出力	コントローラが条件を満たすと、共有メモリに出力します。 例:ボタン1のONOFF状態を、共有メモリ1に出力します。

<< □ PC からコントローラへ

No.	アイコン	機能	内容
19	2	カレンダー 指定	PC の時計が設定された日時になると、EtherFlg/OUT を出力しま す。 例:毎朝9時になったら、EtherFlg でコントローラの動作開始信号を送信
20		ボタン スイッチ	PC 上のボタンスイッチで、EtherFlg/OUT を ONOFF します。 例: PC 画面に表示されたボタンを押して、EtherFlg で電動ストッパを動作させる
21		トグル スイッチ	PC 上のトグルスイッチで、EtherFlg/OUT を ONOFF します。 例: PC 画面に表示されたトグルスイッチを ONOFF し、EtherFlg でパトライトラン プを点灯・消灯させる
22		バーコード	PC に接続されたバーコードリーダで、設定されたバーコードを読み取ると、EtherFlg/OUT を出力します。 例:「PARTS001」というバーコードを読込むと、EtherFlg でランプ1が点灯
23		エクセル→ コントローラ 出力	エクセルのセルが特定の値になると、EtherFlg/OUT を ONOFF します。 例: エクセルファイルのセル A1の値が1になると、EtherFlg でランプ1が点灯
24	-[	コントローラ 連携	接続された他のコントローラの IO 条件を満たすと、 EtherFlg/OUT を ONOFF します。 例: コントローラ1に接続されたセンサ1が ON になると、 コントローラ2に接続されたランプ2が点灯
25		共有メモリ から出力	共有メモリの条件が一致すると、EtherFlg/OUTを出力します。 例:共有メモリが ON になったら、EtherFlgで動作停止信号を送信

※ PowerPoint、Excel は、米国 Microsoft Corporation の米国及びその他の国における登録商標です。 ※セーフィーおよび Safie はセーフィー株式会社の登録商標です。

# 1.カウント

指定した IO 条件を満たした回数をカウントして、PC の画面上に表示します。



カウンタ1~カウンタ16が選択できます。 カウントアップ(1増える)か、カウントダウン(1減る)を選択できます。 下記の設定では、IN1をONすると値が増え、IN2をONすると値が減ります。

562	IN01	· ON	🔹 ಕると	カウンタ1 ・	<b>口</b> 表示	を	カウントアップ・
562	IN02	- ON	🔹 ಕると	カウンタ1 ・	<b>本</b> 表示	を	カウントダウン・

ウンタ/ファイ	ル設定															
カウンタ	インジケ	一タ	ログノ	(計測)	ァイル		ストップ	ウォッチ	阔	衍写真						
1 2	34	5	6	7	8	9	10	11	12	13	14	15	16			
・カウン	タ名 カウ	291														
• <b>==</b> ==`//	511+7215															
		= <i>Л</i> .,		1 N 14	+ = 0.4		ere ( diffuient a									
	77778	- 59,7E	. D	VD9~	- 28X)	εU	05/2801	10		291291						
	リセット®	守(こ口?	7保存													
	保存項	8	指定し	,たファ	伊レー		•	参照								
						1 -	-+	المحاد								
	Luke La	±(	8/875			1	9.950	UPF								
	92919	4603	/1*1 <del>/</del>													_
	保存項	Ħ	指定し	ידרשת	111		· ·	参照								
・カウン	ト値の出力	ΤΞ														
・初期低	<b>ē</b> 0			-	•											
. + + \.	-=	しちが	,													
-75721	×//s⊃ ⊵	] 181.20	,													
	定カウントて	出力	する													
0		5		ታ ተ	ットする	52				<b>√ ¢</b>	1秒間	1		- 出力	ಕನ	
		_		/3/.		~				_		4				
			OK		1							****`d	711.	1		
			01									-1721	_/v			

歯車マーク、歯	ーク、歯車ボタンをクリックすると、設定画面が開きます。								
562 IN01	▼ ON ▼ すると カウンタ1 ▼ 200 表示 を カウントアップ ▼								
カウンク表示 カウンタ1 リセット 日時設定 なし Distr なし 0 出力 以政定 なし リセット									
カウンタ名	カウンタの名前を変更できます。								
	カレンダー リセット     設定した日付・時刻になるとリセットします。       リセット     1000000000000000000000000000000000000								
カウンタ リセット	条件一致 リセット SiO02 IN02 ・ ON ・ するとリセット								
	リセット条件を満たした際に、csvにログ保存することができます。 設定の詳細は→p.14 ログ保存を参照ください。 ※Ver1.70にて、保存項目「ミリ秒」は削除されました。 「時分秒」を選択すると「○時:○分:○:○秒」と保存されます。 ※メモ読み込み(→P.7)をした状態で「信号(I0)名」を保存した場合、 I0名(メモ名)で保存されます。								
カウント値の 出力	カウンタの値を、固定テキストに設定します。 実際のカウント値をcsvファイルに保存したり、メール送信したりすることができ ます。詳細は→p.45を参照ください。								
初期値	カウンタの初期値を設定します。初期値は0です。 カウンタリセット後にもこの値になります。								
カウントボタン	カワンタリセット後にもこの値になります。 カウントボタンの表示非表示を設定します。カウントボタンは、条件に関係な く、クリックすると値を増減できるボタンです。 ・チェック有 ・チェックなし 2017年08月10日(CV) 0900 1002/14/08月10日(CV) 0900 1002/14/08 10								
ー定カウント で出力	<ul> <li>条件が一致すると、コントローラに信号を出力します。</li> <li>Etherflg 接続するコントローラによって、EtherFlgの最大点数は 異なります。</li> <li>&lt;例&gt;SiOt バージョン3.00はEther1~Ether8 SiOt バージョン3.10以降はEther1~Ether64</li> <li>OUT コントローラのRUNスイッチがOFFになっている場合のみ 出力できます。</li> <li>コントローラがRUN状態の場合、以下のマークが表示されます。</li> </ul>								

# 2.インジケータ

指定した IO 条件を満たした回数をカウントして、インジケータ表示で PC の画面上に表示します。



インジケータ1~インジケータ16が選択できます。 カウントアップ(1増える)かカウントダウン(1減る)を選択できます。

歯車マーク、歯車ボタンをクリックすると、設定画面が開きます。



SUS Corporation

# 3.ストップウォッチ

コントローラが条件を満たしている時間を計測します。計測時間は csv ファイルに保存できます。

$\langle$	3	IN01	- ON	・の時間	を ス	トップウォッチ1	- 1	表示	で計測	
						ストップウォッチ1 スパ 平均からん のの sec 平均かん いたか	197099931 0.0 Sec 使業分化。 0.0 sec D設定 なし 即にる			
Č	<b>)</b> IN	101 💌 ON	の時	間を ストッ	ップウォッチ1	T 🎝	表示で語	計測		
出		ーク、歯車ボタ <sup>ウォッチ1</sup> 219799971 のの sec #H3944 の sec 00 sec 00 sec 00 sec 00 sec 00 sec 00 sec 00 sec 00 sec		00000000000000000000000000000000000000	と、設 2020/ 2014 1 D D D D D D D D 1 D D D D D D D 1 D D D D		きます。			
	ス	トップウォッチ	名 ストッ	・プウォ	ッチの	名前です。				
		表示単位 ⑦ 分 <b>③</b> 秒	ストッ 分/秒(	・プウォ のどちら	ッチ画 っで表え	面で表示す 示するかを	「る際、 設定しまう	す。		
		保存項目	csvフ ※Ver1 「時分	アイルに 70にて低 7秒」を通	こ保存す 尿存項目 選択する	する項目をす  「ミリ秒」  うと「○時:(	選択しまう は削除され ○分:○:	す。 ました。 ○秒」と保	存されます	0
	化	呆存ファイル名	csvフ	アイルの	つ名前を	を設定しま	す。			
		表示単位	分/秒 ここで 反映さ	のどち 設定し れます	らかを た単位 。	設定します は、csv保ィ	-。 存や固定す	テキスト出	力にも	
		標準タイム	標準タ 示され	イムを ます。	設定し	ます。設定	ごした値を	計測値が_	上回ると、	赤く表
	平均	肉タイムリセッ	ト条件が	「一致す	ると、	平均タイム	をリセッ	トします。		
	最	終計測タイムの 出力	) 最終計	・測タイ	ムを固	定テキスト	・に出力し	ます。		
	• /		-> -> -> -> -> -> -> -> -> -> -> -> -> -		> A /		,		``	

※一括モニタ画面(→p.7)では、リセット条件や標準タイムの設定はできません。 設定画面で設定してください。

# 4.ログ保存

指定した IO 条件を満たした際に、csv ファイルに保存します。 ※ csv ファイルを開いた状態のまま書き込みを行うと、エラーが発生します。

IN01 - ON	▼ すると 保存項目 ログ出	力指定	ミしたファイル ▼ 参照 C:¥20240402
保存項目 出力項目設定			csv モニタ→P.23
<ul> <li>新規TEDX時に項目3</li> <li>✓ 日付</li> <li>✓ 時分秒</li> <li>✓ SiO名</li> </ul>	名を入れる 固定テキスト 丁三 一 テキスト01		
 ▼ MACアドレス ▼ IPアドレス ▼ ポート番号 ▼ 信号(IO)名	<ul> <li>□ テキスト02</li> <li>□ テキスト03</li> <li>□ テキスト04</li> <li>□ テキスト05</li> </ul>	II	※メモ読み込み(→P.7)をした状態で「信号(IO)名」を 設定した場合、「IO 名(メモ名)」で保存されます。 例: IN1(スイッチ 1)
ОК	テキスト06     テキストの     チャンセル	•	※Ver1.70にて、保存項目「ミリ秒」は削除されました。 「時分秒」を選択すると「○時:○分:○秒:○」と 保存されます。

「新規作成時に項目名を入れる」をクリックすることで、

ファイルを新規に作成した際、csvファイルに項目名を保存することができます。

・チェックしない場合

2021/8/10	9:36:42.7	SiO02	70-B3-D5-42-28-0B	192.168.0.109	40001	IN01	ON	

・チェックした場合

日付	時分秒	SiO名	MACアドレス	IPアドレス	ボート 番号	信号名	ONOFF
2021/8/10	9:36:42.7	SiO02	70-B3-D5-42-28-0B	192.168.0.109	40001	IN01	ON

固定テキストの設定方法は→p45を参照ください。

■ファイル指定

指定したファイルか、ファイル1~ファイル16を選択することができます。

・指定したファイル

「参照」ボタンをクリックし、ファイル名を直接指定します。

指定したファイル・	参照	C:¥20210810093612.csv

・ファイル1~ファイル16

ファイル1~ファイル16設定を利用することができます。→p.14

	7ฅๅ๎ル1 ◄		Ô	1週間単位で新規生成	C:¥¥log1_xxx.csv
Ľ		_			

<ファイル設定>

csvファイルの保存では、ファイル保存で使用できる設定を1~16まで設定できます。

	7711/1	Ø	1週間単位で新規生成 C:¥¥log1_xxx.csv
--	--------	---	-----------------------------

ファイル1~ファイル16を選択して歯車ボタンをクリックすると、設定画面が表示されます。

カウンタ/ファイ	(ル設定											
カウンター	インジケー	-y [	ログ/計測	77711	写真							
1 2 3	34	5	6 7	8 9	10	11	12	13	14	15	16	
ファイル1		ファイル1										
・ファイル生	E成	1週間単	単位で新述	現生成	-							
•保存フォ	มรี	参照	C:¥									
・ファイルネ	2 [	log 1			*	xx xxx(z)(z)	X.CSV 日付が反	硬				
	0	к	]					[	キャンセ	n I		
			J									

ファイル(1~16)	ファイル設定の名前です。
ファイル生成	タイミングに応じて新規にファイルを作成します。 日付毎に新しいファイルを作りたい場合は「1日単位」で生成します。 一時間 / 一日 / 一週間 / 一ヶ月 / 一年 から選択できます。 ※一週間の場合、日曜日から新規ファイルになります。
保存フォルダ	ファイルを保存するフォルダを選択します。
ファイル名	保存されるファイル名を入力します。 実際のファイル名には、ファイル生成のタイミングに応じた数字が 付与されます。

# 5.IO 一括ログ保存

コントローラが条件を満たすと、指定した秒数の IO 状態を保存します。 ※ csv ファイルを開いた状態のまま書き込みを行うと、エラーが発生します。

IN01 -	ON 🔻 する 約 3 🗘 秒前~約	3 🔄 秒後までを 保存項目 🚺	10 をログ出力 指定	したファイル 🔹 参	照 C:¥Log.csv	
■保存項目			∎I/O		(	$csv$ $\tau = \beta \rightarrow P.23$
<ul> <li>出力項目設定</li> <li>♥ 日付</li> <li>♥ 時分秒</li> <li>SiO名</li> <li>MACアドレス</li> <li>IPアドレス</li> <li>ポート番号</li> <li>♥ 信号(IO)名</li> </ul>	固定テキスト 一 テキスト01 一 テキスト02 一 テキスト03 一 テキスト03 一 テキスト04 一 テキスト05 一 テキスト06 ー ーーーーー		1/0 1/0 1/0 1/1 1/2 1/2 1/2 1/2 1/2 1/2 1/2	OUT OUT1 OUT2 OUT3 OUT3 OUT3 OUT5 OUT5 OUT5 OUT9 OUT10 OUT10 OUT11 OUT12 OUT13 OUT15 OUT16	FLAG FLAG FLAG3 FLAG3 FLAG4 FLAG5 FLAG6 FLAG6 FLAG7 FLAG8 FLAG9 FLAG1 FLAG10 FLAG10 FLAG11 FLAG12 FLAG13 FLAG15 FLAG15 FLAG15 FLAG17	

時間設定	記録する時間を設定します。 記録できるのは、条件が一致した前後の数秒間です。 最大時間はそれぞれ60秒で、計120秒間です。
保存項目	csvファイルに保存する際のヘッダーを指定します。 ヘッダーは記録ごとに保存されます。 固定テキストの設定方法は→p.45を参照ください。
	※Ver1.70にて保存項目「ミリ秒」は削除されました。 「時分秒」を選択すると「○時:○分:○秒:○」と保存されます。
I/0	記録するI/0を設定します。 指定したI0の0N/0FF状態のどれかに変化があると、指定したすべてのI0状 態を保存します。
ファイル	保存するファイルを設定します。 指定したファイルか、ファイル1~ファイル16を選択できます。 ファイルの設定は→p.15を参照ください。
csvモニタ	csvファイルを表示する画面を開きます。 詳細は→P.23を参照ください。
ID	設定IDです。記載されたNo.が、csvファイルに保存されます。 番号は通信開始時に更新されます。

※保存中に再度条件が一致した際は、ログデータを保存できません。

別の設定の保存タイミングが重なった場合は保存できます。

<Infomation> 22/06/22 8:54:05 ファイル保存先:C:¥ <Warning> 22/06/22 8:54:05 コントローラ名:SiO01 #1 IO一括データ保存中は受付できません。

csv へIO一括ログ保存開始

■例:「IN1 ON」を条件(トリガー)として、前後3秒間の IN2/IN3/IN4 の IO 状態を記録する。

・設定

▶ [N0] · ON · する約3 ÷ 1	り前~約 3 🔶 秒後までを 保存項目 10 をD	7出力 指定したファー	(ル   • 参照	C:¥Users¥Osh	idaC¥Desktop¥20220	621153300.csv <b>#</b> 1						
・IO 設定	・csvファイル	・csv ファイル例										
I/O	日付	時分秒	ミリ秒	IN02	IN03	IN04						
	2022/6/21	15:33:05	157	OFF	OFF	OFF						
	2022/6/21	15:33:07	336	ON	OFF	OFF						
IN1	2022/6/21	15:33:07	636	ON	ON	OFF						
📝 IN2	2022/6/21	15:33:07	897	ON	ON	ON						
📝 IN3	2022/6/21	15:33:08	157	TRIGGE	R #1(IN01	ON)						
<b>IN4</b>	2022/6/21	15:33:08	396	OFF	ON	ON						
IN5	2022/6/21	15:33:08	756	OFF	OFF	ON						
IN6	2022/6/21	15:33:08	956	OFF	OFF	OFF						

■設定ごとに「#〇」と ID が割り振られ、保存するファイルで「TRIGGER #〇」と記録さ<u>れ</u>ます。

£	IN01 -	ON	💌 する 約 3 🐥 秒前~約 3 🐥 秒後までを	保存項目	IO	をログ出力	指定したファイル・	参照	C:¥Users¥OshidaC¥Desktop¥20220621145030.csv	#1
ß	IN05 -	ON	💌 する 約 10 🕀 秒前~約 10 🔆 秒後まです	保存項目	IO	をログ出力	指定したファイル・	参照	C:¥Users¥OshidaC¥Desktop¥20220621145030.csv	# 2

・csv ファイル例

日付	時分秒	ミリ秒	IN02	IN03	IN04
2022/6/21	15:16:15	544	OFF	OFF	OFF
2022/6/21	15:16:18	242	ON	ON	ON
2022/6/21	15:16:18	544	TRIGGER #	1)	
2022/6/21	15:16:18	842	OFF	ON	ON
2022/6/21	15:16:19	423	OFF	OFF	OFF
日付	時分秒	ミリ秒	IN06	IN07	IN08
2022/6/21	15:16:10	683	OFF	OFF	OFF
2022/6/21	15:16:20	403	ON	ON	ON
2022/6/21	15:16:20	683	TRIGGER #	1)	
2022/6/21	15:16:21	83	OFF	ON	ON

#### 6.メール

条件を満たすと、設定したアドレスにメールを送信します。

- ✓ ---- ▼ すると メールサーバ設定 sus@example.com
   送信内容
   体名
   本文
   文
  - メールサーバを指定し、そのサーバの「メール送信する機能」を使ってメールを送信します。 (この機能は SMTP と呼ばれます)。

SMTP を使用するためには、利用するメールサービスのユーザ登録をしておく必要があります。

- 以下は、IoT-Monitorのメール機能の仕組みを示したものです。
- ① あらかじめ Yahoo!のユーザ「Tanaka」と、メールアドレス「Tanaka@yahoo.co.jp」 を取得しておきます。
- ② 通信を開始し、条件を満たすと、IoT-Monitorからメール送信のリクエストが送られます。
- ③ Yahoo!サーバの SMTP 機能を利用し、Yahoo!のユーザ「Tanaka」のメールアドレスである「Tanaka@yahoo.co.jp」からメールを送信します。
- ④ メールは B 社の社員宛に届いています。



※メール送信機能を利用するためには、インターネットに接続しておく必要があります。
 ※サービスによっては、外部からのアプリケーション(IoT-Monitor)から SMTP が

# 使用できないよう設定されています。外部アプリケーションを許可するよう設定してください。

#### ■使用前の設定

使用するメールサーバ(アドレス)を決めます。アカウントがない場合は作成し、 メールアドレスを取得してください。

Yahoo!メール、Gmail、Hotmailを使用すると簡単です。

- ※「Yahoo!」は、米国 Oath Inc.の登録商標または商標です。
- ※「Gmail」は、米国 Google LLC の登録商標または商標です。
- ※「Hotmail」は、米国 Microsoft Corporation の米国及びその他の国における登録商標です。

#### IoT-Monitor



設定が完了したら、「接続テスト」をクリックし、正しく接続が行えるか確認してください。

② 「送信内容」ボタンをクリックし、「メール内容設定」を開きます。

	ample.com		▲文 (本文
メール内容設定       件名       件名       本文	件名 内容 追加項目 固定 テキスト	メールの件名 <sup>-</sup> メールの内容 <sup>-</sup> メールの内容 <sup>-</sup> メールの内容に、 詳細は固定テキ	です。 です。 こ情報を追加できます。 , 固定テキストを追加できます。 スト→p.45を参照ください。
追加項目 マ 日付 固定テキスト TE マ 時分秒 テキスト01 ・ マ 50名 テキスト02 E マ MACアドレス テキスト03 マ IPアドレス テキスト03 マ ポート番号 テキスト04	沃什	メールに添付す; 「+」アイコン 最大ファイル 容量 ファイル	るファイルを指定します。 で添付ファイルを追加します。 最大ファイル容量を設定します。 添付したデータがこの容量を超えた場合、 データが削減された状態で添付されます。 ファイルを指定します。
✓ 信号(10)名 〒キスト05 添付ファイル + 最大ファイルサイズ 30 ● MB OK キャンセル	「小竹 ファイル	フォルダ ログ/ 計測ファイル 写真データ	フォルダを指定します。 送信時はzipに圧縮されます。 ログ保存→p.14や ストップウォッチ機能→p.13 で記録したcsvファイルを指定できます。 写真→p.25で記録した 画像ファイルを指定できます。

メールで送信する内容や、添付ファイルを設定します。

# 7.ランプ

指定した IO の状態を、PC の画面上で表示します。 ON の色と OFF の色を、それぞれ7色から選択できます。

; C;	IN01 -	の状態をランプ表示	ON 赤 💽	OFF グレー 💌	動作中ランプ	表示
					動作中ランプ	

条件が一致すると、ランプの色が変化します。

■色の選択

赤	
禄	
黄色	
青	
橙	
水色	
紫	
グレー	

# 8.警告画面

条件を満たすと、PCの画面上に警告画面を表示します。

NO1 ▼ ON ▼ すると 警告 ▼ 画面 第一工場	A工程でチョコ停が発生しています!	IV C閉じる
実行条件		閉じる条件
警告画面		
警告表示		
第一工場		
A工程でチョコ停が発生し	ています!	
24/04/02 16:32:54	<b>開</b> じる	

アラーム画面が表示されているときに、 文字の大きさを変更できます。

■画面の選択



# 9.サウンド

条件を満たすと、PC 上でサウンドを鳴らします。



上記の設定では、IN1 が ON になるたびにサウンドを鳴らします。

▶ ボタンで音色の確認ができます。

サウンドは規定の7つの音色から選択できます。

# 10.クラウドカメラ連携

指定した IO 条件を満たした際に、csv ファイルに保存します。 条件を満たした時間の Safie クラウドカメラへアクセスできる URL を保存します。

					·	4
実行条件	時間指定	カメラ ID	保存項目	ファイル指定		Csv モニタ

「保存項目」「ファイル指定」については→P.14~P.15 を参照ください。

カメラID	使用しているクラウドカメラのIDです。 設定したカメラIDをもとに、アクセスできるURLを生成します。
-------	--

※セーフィーおよび Safie はセーフィー株式会社の登録商標です。

URL は以下の形式で生成されます。

https://safie.link/app/streaming/カメラデバイス ID?timestamp=UNIT TIME

条件が一致すると、以下のようにリンクを含めたデータが生成されます。

日付	時分秒	SiO名	IPアドレス	ポート番号	信号(IO)名	ONOFF	リンク
2024/4/3	14:01:09:0	SiO01	192.168.0.100	40001	IN01	ON	https://safie.link/app/streaming/?timestamp=1712120469000
2024/4/3	14:01:56:9	Si001	192.168.0.100	40001	IN01	ON	https://safie.link/app/streaming/?timestamp=1712120516000
2024/4/3	14:02:08:4	SiO01	192.168.0.100	40001	IN01	ON	https://safie.link/app/streaming/?timestamp=1712120528000
2024/4/3	14:03:11:1	Si001	192.168.0.100	40001	IN01	ON	https://safie.link/app/streaming/?timestamp=1712120591000
2024/4/3	14:03:19:9	SiO01	192.168.0.100	40001	IN01	ON	https://safie.link/app/streaming/?timestamp=1712120599000

<csv モニタ>

csv モニタボタンをクリックすると、モニタ画面が表示されます。 csv ファイルのデータを読み出し、画面に表示させることができます。

<b>F</b>	IN01		- ON	-	すると カメラID	example		ۍ	保存項目	をログ出力	指定した	:วราน 💌	参照	C:¥WebCame	raLog				Q
																			J
SV€19							- 0	×											
\$81	C#Users#Oshida	aC¥Desktop¥	20240402171207csv				更新		·····	171007	0.001	00.15.01	05 100.10		7110		011	1.11 11 12	
24/04/02	17:12:37:7	SiO01	8C-1F-64-65 1	92.168.0.100	40001 EN	01 ON	https://safiel.	1	247/047/02	17:1287:	SIUUT	8C-1F-04-	65 192.10	8.0.100 40001			UN	https://sam	<u></u>
24/04/02	17:12:38:4	Si001	8C-1F-64-65 1	192.168.0.100	40001 IN	J1 ON	https://safie.L_								<u>(</u>				
24/04/02	17:12:38:8	Si001	8C-1F-64-65 1	192.168.0.100	40001 IN	J1 ON	https://safie.l_												
24/04/02	17:12:39:1	SiO01	8C-1F-64-65 1	192.168.0.100	40001 IN	J1 ON	https://safie.l												
24/04/02	17:12:39:3	Si001	8C-1F-64-65 1	192.168.0.100	40001 IN	J1 ON	https://safie.l							~					
24/04/02	17:12:39:5	Si001	8C-1F-64-65 1	192.168.0.100	40001 IN	J1 ON	https://safie.l						$\sim$						
24/04/02	17:12:39:7	Si001	8C-1F-64-65 1	192.168.0.100	40001 IN	J1 ON	https://safie.l												
24/04/02	17:12:39:9	Si001	8C-1F-64-65 1	192.168.0.100	40001 IN	J1 ON	https://safiel_												
24/04/02	17:12:40:1	Si001	8C-1F-64-65 1	92.168.0.100	40001 IN	J1 ON	https://safiel_			🗇 🌹 Safie	(セーフィー)	× +							
24/04/02	17:12:40:3	Si001	8C-1F-64-65 1	192.168.0.100	40001 IN	J1 ON	https://safie.l.				0	O A E Marile	fin link Janua Jatan						
24/04/02	17:12:40:6	Si001	8C-1F-64-65 1	192.168.0.100	40001 IN	J1 ON	https://safie.l.				ш.	0 0 4 mapat/2	incarne upp/sec	in ingreampre anneza	110-1112040001000				
24/04/02	17:12:40:8	SiO01	8C-1F-64-65 1	92.168.0.100	40001 IN	J1 ON	https://safie.l.			-									
24/04/02	17:12:41:1	Si001	8C-1F-64-65 1	92.168.0.100	40001 IN	J1 ON	https://safieJ			🖉 safie									
24/04/02	17:12:41:3	Si001	8C-1F-64-65 1	92.168.0.100	40001 IN	J1 ON	https://safieJ			•									
24/04/02	17:12:41:5	SiO01	8C-1F-64-65 1	92.168.0.100	40001 IN	J1 ON	https://safiel_			副乳 カメラー	E >	+							
24/04/02	17:1241:7	SiOUT	8G-1F-64-65 1	.92.168.0.100	40001 IN	JI ON	https://satiel.												
24/04/02	12:12:41:9	SiOUT	8G-1F-64-65 1	92.168.0.100	40001 IN		https://satiel.			■ メディア	クリップ >								
24/04/02	17.124222	51001	80-11-64-65	.92.168.0.100	40001		nttps://satie.l.	l l											
										81 9								-	
											UDI ふ	祝友々	47	いる	行たと	11.	,カオ	-スレ	
										<b>₩</b> 53	UKL /J	MAC	AUC	v · · J	1.9 \	シン	// 9	JC,	•
													- 18				- 1.	, ,	
											規定の	)ブラウ	ァザー	ごひら	くこん	レガー	できき	ET.	
															, _ ,	_ /*			

「参照」でファイルを選択し、「更新」ボタンをクリックすることで表示を更新できます。 ※表示された後に変更された csv ファイルの内容は、自動で反映されません。 ■連携できるサービスについて Safie のクラウドカメラサービスのみ連携が可能です。 対応したカメラ、クラウド録画サービスの契約が必要です。 詳細は Safie 社のホームページを参照ください。 <u>https://safie.jp/</u>

#### 【利用例】

現場をカメラで録画します。 ワークの詰まりが発生したら SiO でランプを光らせ、クラウドカメラ連携ファンクションで csv ファイルに保存します。 その後 csv ファイルから、チョコ停が発生したときのカメラの映像を確認できます。



# 11.写真

条件を満たしたとき、写真で撮影します。

						, .					
ſ	<u>)</u>	IN01 -	ON	すると	HD camera	- で撮影	指定したフォルダ	-	参照	C:¥	へ保存

カメラは PC に内蔵されたものや、USB 等で PC に接続されているものを選択することができます。 ※作成されたファイルの解像度は下がります。

カメラ選択	PCに接続されているカメラから選択します。 表示されない場合、カメラが正しく接続・認識されているかを 確認してください。
保存フォルダ	「参照」を選択し、jpgファイルを保存するフォルダを選択します。 条件が一致し、録画されるたびに新規のjpgファイルが作成されます。

### 12.ビデオ

条件を満たした前後を、ビデオで録画します。

		(						(		C 11 1				
0	9 71	IN01 -	ON	-	する 約 3	🔷 秒前と 約 3	🔷 秒後を	HD camera	- 〇 で録画	マイク (USB オー ディオ デバイフ)	で録音	参照	C¥Video¥	へ保存
	- 14									1212271/2/				

PC に内蔵されたカメラや、USB 等で接続されているカメラを選択することができます。 作成された動画ファイルの解像度は自動で 640\*480 程度に縮小されます。

時間設定	録画する時間を設定します。 録画できるのは、条件が一致した前後の数秒間です。 最大録画時間はカメラやPCの性能によって変化しますが、 おおよそ合計15秒程度です。
カメラ選択	PCに接続されているカメラから選択します。 表示されない場合、カメラが正しく接続・認識されているかを 確認してください。
マイク選択	PCに接続されているマイクから選択します。 表示されない場合、マイクデバイスが正しく接続・認識されているかを 確認してください。 動画に音声を含めない場合は「なし」を選択します。 ※音声を含める場合、動画の生成にかかる時間が長くなります。
保存フォルダ	「参照」を選択し、aviファイルを保存するフォルダを選択します。条件 が一致し、録画されるたびに新規のaviファイルが作成されます。

<例>IN1 が ON すると、その5秒前から2秒後までを動画保存します。

IN01 -	ON -	する	約 5	🗧 秒前と	約 2	HD came

条件が一致すると、動画が保存されます。

تو 💼	
<information> 2021/08/10 11:09:54 ファイル保存先:C:¥ へ動画保存</information>	開始…
<information> 2021/08/10 11:09:58 ファイル保存先:C:¥ へ動画保存</information>	完了

動画の保存を行っている間に再度条件が一致した場合、対応できません。 録音を行う場合は動作の生成に時間がかかるため、ご注意ください。

ט 💼	
<ul> <li><infomation> 2021/08/10 11:11:06 ファイル保存先に¥ へ動画保存開始…</infomation></li> <li><warning> 2021/08/10 11:11:07 コントローラ名:SiO02 タスクID:10 動画データ生成中は受け</warning></li> </ul>	付できません。

# 13.ファイル実行

コントローラが実行条件を満たすと、指定したファイルを実行します。 終了条件を満たすと、実行したプロセスを終了します。

IN01		ON	•	ಕると 🛛	参照	C:¥Sample.exe	を実	美行 IN02	- ON	▼ で終了
	実行条	件							終了条件	

※ファイルは実行時に関連付けられたアプリケーションで動作します。

※実行条件を二回満たした場合、一回目のアプケーションが終了してから再起動します。
 ※終了条件一致時は強制終了するため、ファイル内容の変更を保存することはできません。
 ※ブラウザでのインターネットショートカットなど一部のアプケーションは実行できますが、
 終了できません。

# 14.PDF 表示

コントローラが条件を満たすと、PDF ファイルを操作します。

	1								
רו		TND1	- 11	ON	-	オスト	DDC主子 ·	<u>++.97</u>	
PDF		11401		014		9.9C	LDLSOLC .	-29°.X.K	0.+#/10017#/0% =+3i0 put
			_		_				

※複数の PDF を同時に表示することはできません。

使用する PC にビューアをインストールしておく必要はありません。

PDF表示	指定したPDFを開始します。							
次へ	ページを1つ進めます。							
戻る	ページが1つ戻ります。							
初めに戻る	ページの冒頭に戻ります。							
ジャンプ	指定したページへジャンプします。							
終了	表示を終了します。							

15.パワーポイント

コントローラが条件を満たすと、PC のパワーポイントを操作します。

P	IN01	-	ON	*	すると	スライド開始	•	参照	C¥発表用pptx
				_					

※複数のパワーポイントを同時に表示することはできません。 使用する PC にパワーポイントがインストールされている必要があります。

スライド開始	指定したパワーポイントを開始します。
次へ	スライドが1つ進めます。
戻る	スライドが1つ戻ります。
初めに戻る	スライドの冒頭に戻ります。
ジャンプ	指定したスライドNoヘジャンプします。
終了	スライドを終了します。

#### 16.マクロ

コントローラが条件を満たすと、Excelのマクロ関数を実行します。

x	IN01	ON	-	すると	参照	C¥Samplexism	メソッド名 Sample1	引数	を実行	更新データを表示・		テスト	$\mathbf{x}$
-											4 U		1.00

使用するマクロは標準モジュール内に記述してください。

エクセルファイルのファイル名に記号が含まれていると、エラーが発生する場合があります。 使用する PC にエクセルがインストールされている必要があります。

ファイル	マクロを使用するエクセルファイルを選択します。
メソッド名	実行する関数名を指定します。
引数	メソッドに使用する引数を指定します。 引数は複数指定することができ、1, 2…の順番どおりに設定されます。 変数はstring/Int32/Byte/double/long/bool/日付(string)から選択できま す。引数は固定テキストを指定します。固定テキストの設定方法はp.39を 参照ください。
動作の設定	バックグラウンドで実行: 実行するマクロファイルが開かれていない場合、ファイルを開かずに実行 します。開かれている場合は、そのままマクロを実行します。 更新データを表示: ファイルを開いたままマクロを実行します。 開かれていない場合、ファイルを開いてからマクロを実行します。
テスト	マクロを実行し、動作を確認できます。

例:「Sample.xlsm」で以下のマクロを実行する場合

・セル A1 に「引数 1」を、セル A2 に「引数 2 + 引数 3」の値を代入する関数「Sample1」

```
Sub Sample1(str As String, uil As Integer, ui2 As Integer)
Range("A1") = str
Range("A2") = uil + ui2
End Sub|
```

・設定

ファイル	参照	C:¥SamplexIsm	メソッド名	Sample1	を実行

・引数

•	固定テキス	arepsilon

り指定	Ē		_			
引数						
1	string	-	テキスト01_1	-	ታスト	$\mathbf{\times}$
2	Int32	-	テキスト02_1	-	5	$\mathbf{\times}$
з	Int32	-	テキスト03_1	-	10	×
			ОК		キャンセル	

固定テキスト編集	PTALLATERS.	
	1	
ታキストዕነ 📝	<u>,</u>	
テキスト02 📝	5	
ታキスト୦ଓ 📝	10 ,	
テキスト04 🖍		

→ 実行すると、

・セル A1 に引数1の「テスト」、

・セル A2 に引数 2 と引数 3 の「5」と「10」を足した「15」

が入力される。

	А	В
1	テスト	
2	15	
3		
4		
5		

17.コントローラ→エクセル出力

コントローラが条件を満たすと、Excelのセルに指定した値を出力します。

×	IN01 -	ON -	ಕると	IotPro_ExcelData_1	開く	のセル	A1 -	(2	直接値	1	を出力
---	--------	------	-----	--------------------	----	-----	------	----	-----	---	-----

作成されたエクセルファイルに、指定した値を出力します。 使用する PC にエクセルがインストールされている必要があります。

※エクセルファイルのファイル名に記号が含まれていると、エラーが発生する場合があります。 ※IoT-Monitor で作成したエクセルファイルのみ使用できます。

※必ず指定フォルダ内に保存されている必要があります。「開く」をクリックすることで、 フォルダを参照できます。

※.xlsm ファイルに対応した Excel のみ使用できます。共有にはマクロを使用するため、 マクロの実行を許可しておく必要があります。

ファイル	出力先であるエクセルファイルを選択します。 新規に作成する場合は「新規作成」からファイル名を設定してください。 ※ファイルの生成は直接行わず、必ずこの機能から行ってください。 コピーなどで作成した場合、正しく動作しない可能性があります。
セル	出力先のセルを選択します。 出力できるセルは、A1~A512の中から選択できます。
出力する値	「直接値」または「固定テキスト」を選択します。 直接値の場合は、出力する値を設定してください。 固定テキストの場合は、設定した固定テキストが出力されるため、 カウンタ値や読込んだバーコードなどを出力することができます。

<例 1>IN1 が ON すると「1」を、OFF なら「0」を、

エクセルファイル「IotPro\_ExcelData\_1.xlsm」のセル「A1」に出力する

×	IN01	- ON	📩 ಕると	IotPro_ExcelData_1	闌	のセル	A1 -	] (z	直接値	1	を出力
×	IN01	- OFF	💽 ಕると	IotPro_ExcelData_1	猒	のセル	A1 -	3	直接値	0	を出力

<例 2> IN2 が 0N すると、カウンタ 1 のカウンタ値を エクセルファイル「IotPro ExcelData 1. x1sm」のセル「A2」に出力する

	JotPro_ExcelData_1			· · · · · · · · · · · · · · · · · · ·
カウンタ/ファイル設定 ガウンタ インジカータ ログ(計測ファイル ストップウォッチ 添 1 2 3 4 5 6 7 8 9 10 11 12 ・ガウンタ名 ガウンタ1 ・ガウンタ125ト				
	B定テキスト編集 「テキストOT ♪ [/カウンタ]カウンタ1 カウント取 テキストOZ ♪ 「オンパロンタ】 「オンパロンタ」 「オロンタ」 「コーム 「 「 「 「 「 「 「 「 「 「 「 「 「	2	※カウンタ値を固定テキ するためには、p.45 を	ストに出力 ・参照ください。

〈エクセルの設定〉

「新規作成」より、以下のようなエクセルファイルが生成されます。 このテンプレートに機能を追加していき、作成してください。

🕼 🗄 5- 0	IotPro_ExcelData_1.xlsm	- Excel	? 🗈 – 🗆 X
ファイル ホーム 挿入 ページレイ	(アウト 数式 データ 校開	表示 開発 チーム	OshidaC + 🔍
MS         P∃>yh         11           Bib)dib         B         I         U         A         A           W         S         I         U         ×         A         A	<ul> <li>■ ■ ■</li> <li>■ ■</li> <li>■ ■</li> <li>■ ■</li> <li>■ ■</li> <li>■</li> <li>■</li></ul>	<ul> <li>□□□ 条件付き書式 *</li> <li>□□□ テーブルとして書式設定 *</li> <li>□□□ セルのスタイル *</li> </ul>	器 挿入 ↓ ▲
クリップボード ら フォント	G 配置 G 数值	ュ スタイル	セル・
11 $\cdot$ : $\times \checkmark f_x$			~
A	В	D E	F G 🔺
1			
2		共有開始	
4			
5			
6			
8			
9			
10			
12			
13			
14			
15			
16			
18			
19			
20			
21			
22			
24			
25			
26			
27			
20			
Sheet1 (+)			Þ
準備元了 副		₩ ■ ₩ -	+ 100%

各セルの設定内容は以下の通りです。

A1~A512:コントローラ→Excel ファンクション専用のセル B1~B512:Excel→コントローラファンクション専用のセル C1、C2:連携に必要なパラメータが入力されているため、削除・変更しないでください。 共有開始・共有停止ボタン:IoT-Monitor との共有を開始/停止できます。

A/B/Cのセルは削除・挿入できません。

D以降のセルを自由に変更することができます。

以下のように、共有させたい値を A1~B512 に連動させる形でご使用ください。



※共有にはマクロを使用するため、マクロの実行を許可しておく必要があります。 ※シート名「Sheet1\_IoTShare」は変更しないでください。 18.コントローラ→共有メモリ出力

指定した IO の状態を、指定された共有メモリへ保存します。

(	1				
IN01 -	の状態を	共有外刊名	IotProgrammer_memory	Index 1	に保存

※一つの Index に対して、一つの IO を設定してください。 複数の IO を設定すると、いずれかの最後の動作が反映されます。

「共有メモリ」は、PCメモリ内に指定する記憶領域です。ここに保存された情報は、 他のPCソフトからアクセスすることができます。 共有メモリを使用することで、IOのONOFF状態を他のPCソフトと共有できます。

以下の設定で連携してください。

共有メモリ名	共有メモリの名前を指定できます。 共有先のソフトでも、同名の名前を指定してください。
使用可能 メモリサイズ	256 バイト
データ型	バイト 1以上 = ON 0 = OFF
共有メモリ生成	このファンクションでは、共有メモリがない場合は生成、 生成されている場合はオープンを行います。

■共有メモリの使用例



■例: IN1の ONOFF 状態を、自作アプリケーション A で取得

1. コントローラの IN1 の ONOFF 状態を IoT-Monitor が読込み、 共有メモリ「Index1」に書込む





 アプリケーションAで共有メモリ「Index1」を読込む 「1」なら ON、「0」なら OFF

```
【アプリケーションA側】メモリ読込例(C#)
```

∥ 共有メモリを開く

MemoryMappedFile mapped = MemoryMappedFile.CreateOrOpen ("IotProgrammer\_memory", 256); MemoryMappedViewAccessor accessor = mapped.CreateViewAccessor();

// 共有メモリを読込む

int index = 0; // Index1:0 Index2:1 Index256:255 byte data = accessor.ReadByte(index); //dataにIndex1のデータを格納

// 開放 accessor.Dispose();

# 19.カレンダー指定

指定した日付になると、PC からコントローラへ信号を送信します。

さ カレンダー設行	<b>毎日 11:30</b>		EtherFlg01 をONする
	カレンダー設定	teriti - teriti	
	<ul> <li>日付で指定</li> <li>曜日で指定</li> <li>第×曜日で</li> </ul>	指定 💿 毎日	
	2021年08月10日(火)	カレンダー <ul> <li>オレンダー</li> <li>4 2021年 8月</li> </ul>	
	11 <sup>(</sup> ) 時 30 <sup>(</sup> ) 分 一 毎時	日月火水木金土	
	·第/曜日 第 1   • 日曜日   •	22 23 24 25 26 27 28 29 30 31 1 2 3 4	
	1曜日	🔄 毎年 🔄 毎月	
		水曜日──太曜日──金曜日──土曜日	
	ОК	キャンセル	

	信号をONにする時間を設定します。 ONになっている時間は、条件が一致した1分間です。								
	<ul> <li>日付で指定</li> <li>2000</li> <li>2000</li></ul>								
	毎日 カレンダー クロ21年 8月 ト								
カレンダー設定	11 ↓     時     30 ↓     分     11 ↓     日月火水木金土       25 26 27 28 29 30 31     1 2 3 4 5 6 7       一 毎時     8 9 10 11 12 13 14       15 16 17 12 13 14								
	第34章 第1111日間目111日111日111日111日111日1111日111日1111日1111日1111								
	<例>上記の場合、毎日11時30分に信号が0Nになり、 11時31分に0FFになります。								
	送信する信号を設定します。								
	接続するコントローラによって、EtherFlgの最大点数は 思わります								
	英なりより。 <例>SiOt バージョン3.00はEther1~Ether8								
EtherFlg	SiOt バージョン3.10以降はEther1~Ether64								
OUT	• OUT								
	コントローフのRUNスイッナかOFFになっている場合のみ 出力できます								
	コントローラがRUN状態の場合、以下のマークが表示されます。								
	カレンダー設定 毎日 11:30								

20.ボタンスイッチ

PC 上のボタンスイッチを押すと、コントローラの EtherFlg/OUT を ONOFF します。 ボタンは、<u>押している間だけ</u> ON になります。





■色の選択

赤	Ether1ボタン
禄	Ether1ボタン
黄色	Ether1ボタン
青	Ether1ポタン
橙	Ether1ボタン
水色	Ether1ボタン
紫	Ether1ボタン
グレー	Ether1ボタン

# 21.トグルスイッチ

PC 上のトグルスイッチを押すと、コントローラの EtherFlg/OUT を ONOFF します。 ボタンは、クリックするたびに ON/OFF が<u>切り替わります</u>。

Ether 1スイッチ	表示	を押すと	EtherFlg01	- <b>t</b>	≹ONする
		Ether 1ス	19 <del>5</del> 😣		
			OFF		

	光信ナス信日を訊点します
	达信 9 <b>の</b> 信 万 を
	• Etherflg
	接続するコントローラによって、EtherFlgの最大点数は
	異なります。
	<例>SiOt バージョン3.00はEther1~Ether8
EtherFlg	SiOt バージョン3.10以降はEther1~Ether64
OUT	• OUT
	コントローラのRUNスイッチがOFFになっている場合のみ
	出力できます。
	コントローラがRUN状態の場合、以下のマークが表示されます。
	<b>OUT1スイッチ</b>

# 22.バーコード

指定したバーコードを読み取ると、PCからコントローラへ信号を送信します。 バーコードリーダは一台のみ接続可能です。

■ 製品A バーコードが Parts001 と 一致すると ▼ EtherFig/OUT を 次の読込まで ▼ SONする 出力 デキ	011 - 読町	読取通知 EtherFlg02 -

タイトル	この設定のタイトルです。
	読み取る文字列を設定します。最大で250文字を設定できます。スペースや 半角/全角なども正しく入力してください。実際にバーコードリーダーで バーコードを読込み、直接文字を入力するのがお勧めです。
バーコード	<ul> <li>※使用できる文字は半角数字・半角アルファベット・ 半角スペース・ 記号(/)です。</li> <li>※アルファベットは大文字と小文字を区別しません。 Iot-Monitorで「abc」と設定し、バーコード「ABC」を読み取ると 一致します。</li> <li>※バーコードリーダーの終端文字は「Enter」「Tab」に設定してください。</li> </ul>
	一致すると 文字列に一致すると、信号を出力します。
一致すると/ 正規表現で 一致すると	正規表現を使用できます。 <例>「Parts001.*」の場合 Parts001* ○「Parts001001」 ○「Parts001002」 ×「Parts002」
EtherF1g OUT	送信する信号を設定します。 • Etherflg 接続するコントローラによって、EtherFlgの最大点数は 異なります。 <例>SiOt バージョン3.00はEther1~Ether8 SiOt バージョン3.10以降はEther1~Ether64 • OUT コントローラのRUNスイッチがOFFになっている場合のみ 出力できます。 コントローラがRUN状態の場合、以下のマークが表示されます。 WHE バーコードが 5555555 と 一致すると
ON時間	バーコードが一致したとき、信号がONになる時間を設定します。
出力先	バーコードが一致したとき、スキャンしたバーコードの値を固定テキストに 出力できます。設定すると、固定テキストにはこの設定のタイトルが表示さ れます。 保存されたバーコードは、ソフトが終了するまで保持されます。 使用事例はp.53を参照ください。
読取通知	一致、不一致に関係なく、バーコードリーダで読み取りを行った際に、 指定した信号を0.5秒間送信することができます。 複数のバーコードファンクションで設定した場合、すべてが出力されます。

<バーコードリーダとは>

パソコンの周辺機器として、バーコードリーダが販売されています。 USB や RS-232C など様々なインタフェースがあります。

PCと接続したバーコードリーダは、バーコードを読み取ると、PCに文字を入力します。 メモ帳などを起動し、バーコードを読み取ると、その文字がキーボードのように入力されます。



バーコードやバーコードリーダにより、使用できる文字が制限されている場合があります。 お使いのバーコードリーダの取扱説明書をよく参照の上、ご使用ください。

※バーコードスキャンが認識されない場合、

終端文字が Enter(改行)または Tab に設定されているか、 読み取り言語が正しく設定されているかを確認してください。

### 23.コントローラ連携

条件を満たすとコントローラの信号を ON します。満たさなくなると OFF になります。 条件には、**追加されているすべてのコントローラ**の IO を指定することができます。



上記ではコントローラ A の IN1 を ON すると信号が ON になり、 IN1 が OFF になると信号も OFF になります。

条件	条件には他のコントローラのIOも指定できます。 <例>コントローラBのIN1がONすると、コントローラAに信号を送信します。
	送信する信号を設定します。
EtherFlg OUT	<ul> <li>Etherflg 接続するコントローラによって、EtherFlgの最大点数は 異なります。</li> <li>&lt;例&gt;SiOt バージョン3.00はEther1~Ether8 SiOt バージョン3.10以降はEther1~Ether64</li> <li>OUT コントローラのRUNスイッチがOFFになっている場合のみ 出力できます。</li> <li>コントローラがRUN状態の場合、以下のマークが表示されます。</li> </ul>

24.エクセル→コントローラ出力

Excel のセルに指定した値が入力されると、PC からコントローラへ信号を送信します。

	IotPro_ExcelData_1	-	厭	のセル	B1	•	の値が	直接値	-	1	と一致すると	EtherFlg/OUT	をONする
--	--------------------	---	---	-----	----	---	-----	-----	---	---	--------	--------------	-------

作成されたエクセルファイルのセルが指定した値になると、信号を送信します。 使用する PC にエクセルがインストールされている必要があります。

※エクセルファイルのファイル名に記号が含まれていると、エラーが発生する場合があります。 ※IoT-Monitor で作成したエクセルファイルのみ使用できます。 ※必ず指定フォルダ内に保存されている必要があります。「開く」でフォルダを参照できます。

※.xlsm ファイルに対応した Excel のみ使用できます。共有にはマクロを使用するため、

マクロの実行を許可しておく必要があります。

※エクセルファイルの利用方法については、p.33を参照ください。

ファイル	セルを読込むエクセルファイルを選択します。 新規に作成する場合は「新規作成」からファイル名を設定してください。
セル	読込むセルを選択します。 読込み可能なセルは、B1~B512の中から選択できます。
比較する値	エクセルから読込んだ値が、ここで設定された値と一致する場合に コントローラへ出力します。 「直接値」または「固定テキスト」を選択します。 直接値の場合は、比較する値を設定してください。 固定テキストの場合は、設定した固定テキストと参照します。 カウンタ値や読込んだバーコードなどを出力することができます。

<例 1>エクセルファイル「IotPro\_ExcelData\_1.xlsm」のセル「B1」の値が「1」なら Ether1 を、 「2」なら Ether2 を ON する

	IotPro_ExcelData_1	▼ 開<	のセル 🖪	31	🚽 の値が	直接値	- 1		と一致すると	EtherFlg/OUT	をONする
L L	IotPro_ExcelData_1	▼ 開<	Ø₽₩ [	31	🔹 の値が	直接値	• 2		と一致すると	EtherFlg/OUT	をONする
					EtherFlg/C	UT		×	EtherFlg/OUT		
					EtherFle	t .	OUT		EtherFlg	OUT	r
					V E	therflg01	<b>^</b> 0	UT01	Etherf	le01	OUT01
					E	therflg02	0	UT02	👿 Etherf	lg02	OUT02

<例 2>エクセルファイル「IotPro\_ExcelData\_1.xlsm」のセル「B2」の値が、 カウンタ1のカウンタ値と一致した場合に Ether1 に出力する

IotPro_ExcelData_1 ・ 開く のセル B	2 ・ の値が テキスト01_1 ・ [わウンタ]カウンタ1 と一致すると EtherFlg/OUT をONする
2000/04/26     2000/07.30/06/2     100/07.00/07.20/07.	※カウンタ値を固定テキストに出力 するためには、p.48 を参照ください。

25.共有メモリからコントローラ出力

共有メモリが条件を満たすと、PCからコントローラへ信号を送信します。

「共有メモリ」は、PC内に設定された記憶領域です。ここに保存された情報は、他のPCソフトからアクセスすることができます。

共有メモリを使用することで、他の PC ソフトからコントローラを出力させることができます。

以下の設定で連携してください。

共有メモリ名	共有メモリの名前を指定できます。 共有先のソフトでも、同名の名前を指定してください。
使用可能 メモリサイズ	256 バイト
データ型	バイト 1以上 = ON 0 = OFF
共有メモリ生成	このファンクションではメモリのオープンのみを行っています。 自作アプリケーション側で共有メモリを生成してください。

■共有メモリの使用例



●例:自作ソフト「アプリケーションA」から、コントローラのランプを光らせる
1. アプリケーションAで、共有メモリ「Index1」に「1」を書込む
【アプリケーションA側】メモリ書込み例(C#)
// 共有メモリを開く
MemoryMappedFile mapped = MemoryMappedFile.CreateOrOpen("IotProgrammer\_memory", 256);
MemoryMappedViewAccessor accessor = mapped.CreateViewAccessor0;
int index = 0; // Index1:0 Index2:1 Index256:255
byte data = 1; // ON:1 OFF:0
// 共有メモリに書込む
accessor.Write(index, data);
// 開放
accessor.Dispose0;

2. IoT-Monitor で「Index1」の ON(1)を読込み、コントローラに出力

【IoT-Monitor 側】設定



# 5 固定テキスト

固定テキストは、メールや csv 保存で文字を保存できる機能です。通信接続中でも変更できます。 ここでは、以下4通りの使用事例を説明します。

#### 1. 同じ文字を、メールや csv 保存などで繰り返し使用する (→p.46)

同じ文字を、メールやログなどで何度も繰り返し使用できます。

日付	時分秒	SiO名	MACアドレス	IPアドレス	ポート番号	信号名	ONOFF	製品
2021/12/24	9:38:15.2	Si001	70-B3-D5-42-29-19	192.168.0.100	40001	IN01	ON	製品A
2021/12/24	9:38:16.5	SiO01	70-B3-D5-42-29-19	192.168.0.100	40001	IN01	ON	製品A
2021/12/24	9:38:17.4	Si001	70-B3-D5-42-29-19	192.168.0.100	40001	IN01	ON	製品A

2. カウンタやインジケータのカウント値を、csv ファイルに出力する (→p.48)

カウンタ/インジケータファンクションのカウンタ値を、メールやログに出力できます。

日付	時分秒	SiO名	MACアドレス	IPアドレス	ポート番号	信号名	ONOFF	製品カウント
2021/12/24	9:43:26.2	SiO01	70-B3-D5-42-29-19	192.168.0.100	40001	IN01	ON	21
2021/12/24	9:43:32.1	SiO01	70-B3-D5-42-29-19	192.168.0.100	40001	IN01	ON	26
2021/12/24	9:42:36.7	Si001	70-B3-D5-42-29-19	192.168.0.100	40001	IN01	ON	41

3. 作業中の担当者の名前を、csv ファイルに出力する (→p.49)

固定テキストは、通信中も変更することができます。これを利用し、固定テキストを csv 保存するようにしておき、担当者名を入力し担当者の名前を csv ファイルに記録できます。\_\_\_\_\_

日付(計測開始)	時分秒(計測開始)	日付(計測終了)	時分秒(計測終了)	信号(IO)名	ラップ	min/sec	担当者
2023/1/18	44:42.8	2023/1/18	44:46.9	IN01	4.044	sec	山田花子
2023/1/18	44:47.4	2023/1/18	44:51.0	IN01	3.599	sec	山田花子
2023/1/18	44:51.4	2023/1/18	44:56.3	IN01	4.919	sec	山田花子
2023/1/18	44:56.7	2023/1/18	45:00.1	IN01	3.49	sec	山田花子
2023/1/18	45:00.4	2023/1/18	45:04.4	IN01	3.999	sec	田中一郎
2023/1/18	45:04.8	2023/1/18	45:09.0	IN01	4.249	sec	田中一郎
2023/1/18	45:09.3	2023/1/18	45:13.8	IN01	4.459	sec	田中一郎

4. 読み取ったバーコードの値を、csv ファイルに出力する (→p.51)

バーコードファンクションで読み取ったバーコードの値を、メールやログに出力できます。

日付	時分秒	信号(IO)名	ONOFF	バーコード	
2023/5/16	10:50:32.3	IN01	ON	SUC-001	001
2023/5/16	10:50:38.0	IN01	ON	SUC-002	001
2023/5/16	10:50:43.6	IN01	ON	SUC-003	001
2023/5/16	10:50:49.0	IN01	ON	SUC-005	002
2023/5/16	10:51:05.9	IN01	ON	SUC-006	002

1.同じ文字を、メールや csv 保存などで繰り返し使用する

① 固定テキストアイコンをクリックし、「固定テキスト編集」を開きます。

No IoT Programmer for SiOt v122β			
ファイル(E) オプション			
	0 ms		
	固定テキスト編集	success prime Raise	18 18 81 81
		1 2	3
	7#21F01 💉 🛛		
	7キスト02 🖋		
	テキスト08 🧪		
	テキスト04 🖋		
	テキスト05 🖋		
	742106 🖉		
	テキスト07 🖍		
	7421-08 🖍		
	742100 🖉		
	テキスト10 🖋		
	742111 🖉	).	,
	テキスト12 🖍		
	テキスト13 🖋		
	テキスト14 🖋	),	,
	テキスト15 🖍	).	,
	テキスト16 🖍	,	,
		間はる	

② 「テキスト 01」右にある鉛筆アイコンをクリックし、好きなタイトルを入力します。 ここでは「タイトル」と入力しています。

固定テキスト編集				
	1	2		
テキス 01 📝				
<del>7</del> ≠λトα	固定テキスト タイ	トル編集		
ታキストα 🧪				
74240	テキスト01	タイトル		
7+210				
7#2108 🖉	ОК		閉じる	

③「タイトル」の1に文字を入力します。ここでは「サンプルテキスト」と入力しています。

固定テキスト編集	
1	2
֍イトル 🖉 サンプルテキスト	,
テキスト02 🖉	,
テキスト03 🖉	

④ コントローラを追加し、ログ保存のファンクションを追加します。

👌 IoT Programmer for SiOt v122β	and the second s	and the second second second	
ファイル(E) オプション			
コントローラ登録			
Si001			
	ポート番号 40001		
17 C			
	指定したつってい. ・ 金昭	コッイルが設定されていません	
	JEAC OCCYTIAN	571703%DEC10C0%E208	

⑤ ログ保存ファンクションの「保存項目」をクリックし、「固定テキスト」から 先ほど編集した「タイトル」をチェックし、「OK」をクリックします。

ton → da	保存項目 きつび出力	指定したファイル・	参照 ファイルが設定されていません。
		出力項目設定	
		新規作成時に項目:	名を入れる
		▼日付	固定テキスト 丁三
		♥ №7779 ▼ SiO名	▼ \$41 FJ
		<ul> <li>✓ MACアドレス</li> <li>✓ IPアドレス</li> </ul>	□ テキスト03
		☑ ポート番号	デキスト04
	$\rightarrow$	▼信号(IO)名	📰 ታትスト05
			🕅 テキスト06
			The second secon
		ОК	キャンセル
			]

メールの場合、「送信内容」から固定テキストを選択します。

メール内容設定	
·件名 件名	
·内容本文	
·追加項目	
🔽 日付	固定テキスト 丁三
📝 時分秒	
三 3.1秒	デキスト02 E
▼ SiO名	デキスト03
<ul> <li>✓ MAGPドレス</li> <li>✓ IPアドレス</li> </ul>	🕅 テキスト04

⑥条件、ファイル設定を行います。



⑦ 通信を開始します。

ログ保存では、生成された csv ファイルにタイトルとテキストが記録されています。

日付	時分秒	SiO名	MACアドレス	IPアドレス	ポート番号	信号(IO)名	ONOFF	タイトル
2021/8/10	15:36:30.4	SiO01	70-B3-D5-42-29-19	192.168.0.100	40001	IN01	ON	サンプルテキスト

メールの場合、送信されたメールにタイトルとテキストが書かれています。

************************************	)
タイトル:サンブルテキス	⊦,
****	

2. カウンタやインジケータの値を、csv ファイルに出力する			
① カウンタ/インジケータのファンクションを追加し、設定を行う。			
IN01     V     ON     すると     カウンタ1     マ     表示	を	カウントアップ	-

② 設定画面を開き、「カウント値の出力」を選択。

ワンタ		111	設定	5	- 10	(=1.284	- / ·		~±						
	220	12	>	5	6	at /29	ر 771 ر د	۷ ۵	ラ県 10	11	10	12	14	15	16
	ったい たいしょう たいしょう	。 2名 11/17ッ	י שלים א	。 לעלי	I				10		12	10		10	10
		カレ: 	ンダー	設定	] שני   י		-を設) 	定して   ▼	下さい。 ] すると	IJセット	. (	ロセット			
	かうント	·値のと 	出力	ΤΞ		•	]								
	初期値	[	0					* *							
•	カウント	ボタン		有効											
			Ok	<								د	キャンセ	ur 🛛	

③ カウンタ値を、どの固定テキストに設定するかを選択します。 ここでは「テキスト 01」の「1」を選択しています。

・カウント値の出力 工王											
<ul><li>マーテキン</li></ul>	ג⊦01 <u>_</u> 1	-									
	テキスト01	•	1								
・初期値	テキスト02	×	2								
	テキスト03	۲	3								
・カウントボ!	テキスト04	►									
	-*- 7 1.05										

④ 「OK」をクリックします。

・カウント値の出力 TE	
- *刀與1值 0	
・カウントボタン 🗹 有効	
ОК	**>セル

⑤ 固定テキストアイコンをクリックし、固定テキスト編集画面を開きます。 カウンタ値の出力先として設定されていることを確認します。 ここでは先ほど設定した「テキスト01」の「1」が固定テキストとして指定されています。

ex.				
	<u> </u>			
F年2101 📝 [おちンタ]おちンタ1 おちント派 ]		1	2	3
#2/102				 
#2.104				
F#2.h05 🖉		ן אאַרענע דעענענענעני		
F#2,h06 🖉				
P#2,107 🖉				
F421F08 🖉				
F#3.100 📝				
F#3,h10 🖌				
F#3,h11 📝				
F#21-12 /				
F#2,h13 🖉				
F#2/14 🖉				
F#21-15 🖉				
P42h16 🖉				

テキストのタイトルを、「テキスト01」から「カウント値」などに変更します。

固定テキスト編集		
_	1	2
テキストơ 📝	[カウンタ]カウンタ1 カウント数 ),	,
テキストの2 🗾	固定テキスト タイトル編集	
テキスト፡፡ଃ 🗾		
テキスト04 🖍	固定ナキスト カワント10	
テキスト05 🖍	ОК	閉じる
テキスト06 🖍		

⑥ ログ保存ファンクションを追加します。「保存項目」をクリックし、「固定テキスト」から 先ほど設定した固定テキスト(ここでは「カウント値」)をチェックします。



⑦ 条件やファイル設定を行い、通信を開始します。
 生成された csv ファイルにカウンタ値が記録されています。



- 3. 作業中の担当者の名前を、csv ファイルに出力する
- ① 固定テキストアイコンをクリックし、固定テキスト編集画面を開きます。

No IoT Programmer for SiOt v122β			
ファイル(E) オプション			
	0 ms		
	固定テキスト編集	PERSONAL PROPERTY AND	 10
	= hal a	2	
	7727		
	742102		
	742108		
	7年2104		
	7年2106		
	742106 🎽		
	742107 🖉		
	772108		
	77210		
	77XP10		
	77XP11		
	TAAPis 🖉		
	Texpia		
	74210		
	74,116		
		開ける	

② 「テキスト01」右側の鉛筆アイコンをクリックし、「担当者」と入力。



③ コントローラを追加し、ストップウォッチのファンクションを追加します。

Si001
機器タイナ SiO 機器名 SiO01 IPアドレス 192.168.0.100 ポートNo. 40001
<b>+</b>

④ 「保存項目」をクリックし、「固定テキスト」から先ほど編集した「担当者」を チェックします。



⑤ 条件、ファイル設定を行います。

-								_					
Ø	IN01 -	ON	▼ の時間を	◎ 分	◎ 秒	単位で計測	表示	1	保存項目	指定したファイル	-	参照	C:¥20211224101204.csv
_													

⑥通信を開始します。

担当者は、固定テキストに自分の名前を記入しておきます。

(または、あらかじめ名前をバーコードリーダーで作成しておき、読込みます)

📎 IoT Programmer v132β	
ファイル オプション	
コントローラ登録	
🛜 Si001	固定テキスト編集
機器タイプ SiO 機器名 SiO01     IPアドレス 192	
+	1 2
IN01  ▼ ON  ▼ の時間を ○ 分 ◎ 秒	担当者 🖌 山田花子 🛛
	¯≠λh02 🖉,
	テキスト፡፡ 🎤
	न्द्रिम् 🧨 🕺

⑦ 担当が替わったら、通信したまま名前を変更します。

2

#### ⑧ ログ保存で、入力した担当者の名前が保存されています。

日付(計測開始)	時分秒(計測開始)	日付(計測終了)	時分秒(計測終了)	信号(IO)名	ラップ	min/sec	担当者
2023/1/18	44:42.8	2023/1/18	44:46.9	IN01	4.044	sec	山田花子
2023/1/18	44:47.4	2023/1/18	44:51.0	IN01	3.599	sec	山田花子
2023/1/18	44:51.4	2023/1/18	44:56.3	IN01	4.919	sec	山田花子
2023/1/18	44:56.7	2023/1/18	45:00.1	IN01	3.49	sec	山田花子
2023/1/18	45:00.4	2023/1/18	45:04.4	IN01	3.999	sec	田中一郎
2023/1/18	45:04.8	2023/1/18	45:09.0	IN01	4.249	sec	田中一郎
2023/1/18	45:09.3	2023/1/18	45:13.8	IN01	4.459	sec	田中一郎

4.読み取ったバーコードの値を、csvファイルに出力する

バーコードファンクションでは、スキャンしたバーコードが設定と一致した際に、 そのバーコードを固定テキストに保存することができます。 ログ出力ファンクションにて保存項目にこの固定テキストを指定することで、 スキャンしたバーコードの値を csv に出力できます。

以下の例では、2つのバーコードを順番にスキャンし、そのバーコード値を csv 保存しています。



① バーコードのファンクションを追加し、設定を行う。

愛品コード	バーコードが	SUC-	と「部分一致する。*	EtherFlg/OUT	ŧ	1秒間 🛛 🔻	をONする	出力	テキスト01_1 ▼	読取通知なし	-
作業者コード	バーコードが	¥d{3,]\$	と 正規表現で一↓ 致すると	EtherFlg/OUT	æ [	1秒間 🗸	をONする	出力	テキスト01_2	読取通知なし	

バーコードファンクションの「出力」設定にて、スキャンしたバーコードの保存先を設定します。 ここでは製品コードに「テキスト01」の「1」、作業者コードに「テキスト01」の「2」を 選択します。



② 固定テキストアイコンをクリックし、固定テキスト編集画面を開きます。
 バーコード値の出力先として設定されていることを確認します。
 製品コードが「テキスト01」の「1」、

作業者コードが「テキスト01」の「2」に指定されています。

固定テキスト編集				
1	2	3		
テキストの 🖍 [パーコード]一致(製品	コード) [パーコード]一致(作業者コード)			
7#21-02		4	0	0
7#21-08			Z	J
7#2h04 🖍				
テキスト05 💉	ታችለቦጣ 🥒	[パーコード]一致(製品コード)	) [バーコード]一致(作業者コード	
テキスト06 💉				
テキスト07 🖍	,			
テキスト08 🖍	,			
テキスト09 🖍	,			
テキスト10 🖍	,			
テキスト11 🖍	,			
	閉じる			

テキストのタイトルを、「テキスト01」から「バーコード」などに変更します。

	1 2
ד#גים 💽 🕅	-コード]一致(製品コード) / [パーコード]一致(作業者コード /
テキストα 📝	固定テキスト タイトル編集
テキストα 🔀 🗌	固定テキスト バーコード
テキスト04 📝 🗌	
テキスト05 📝 🗌	OK 閉じる
テキスト06 📝 🗌	

③ ログ保存ファンクションを追加します。「保存項目」をクリックし、「固定テキスト」から 先ほど設定した固定テキスト(ここでは「バーコード」)をチェックします。



④ 条件やファイル設定を行い、通信を開始します。

	製品コード バーコ	]ードが SUC-	٤	部分一致する。	EtherFlg/OUT	æ	1秒間 🔹	をONする 出	氻	テキスト01_1 ▼	読取通知なし	•
	作業者コード バーコ	]ードが [¥d{3.]\$	٤ 🛛	正規表現で一、 致すると	EtherFlg/OUT	を	1秒間 🔹	をONする 出	力	テキスト01_2	読取通知なし	•
LOG	IN01 - 0	N 🕝 すると	保存項目をログ出力	指定したファ	ะ1.1.	参照	C:¥202305121	80538csv				ר

以下のようにスキャンし、csv 保存した際の値は以下のとおりです。



生成された csv ファイルにバーコードが記録されています。

日付	時分秒	信号(IO)名	ONOFF	バーコード	_
2023/5/15	15:54:36	IN1	ON	SUC-001	001

# 6 オプション

オプションは、メイン画面のメニュー「オプション」から設定できます。

📎 IoT Pro	arammer v161
ファイル	オプション
321-5	目動起動
Q	タイムアウト設定
SiO01	言語設定
	機器タイプ SiO コントローラ名 SiOO1 I
+	

# 1.自動起動

IoT-Monitor を起動した際に指定したファイルの設定を読込み、コントローラとの接続・通信を 自動で開始する機能です。

「次回起動時に自動で通信を開始する」にチェックを入れ、読込むファイルを指定してください。

自動起動					
☑ 次回起動時に自動で通信を開始する					
参照 C.¥Sample.iot					
※Iot-Programmerのショートカットをスタートアップに設定しておく必要があります。					
OK キャンセル					

※<u>IoT-Monitor のショートカット</u>から起動させた場合のみ、自動起動を行います。

「.iot」ファイルをクリックして起動させた場合は自動起動を行わず、そのファイルを開きます。

※PCの起動と同時に IoT-Monitor の自動起動を行う場合は、上記の設定を行った上で、 お使いの PC のスタートアップに IoT-Monitor のショートカットを指定してください。

1.スタートアップフォルダにアクセスします。パスは以下の通りです。

 $C: {\tt ¥Users {\tt ¥} & USERNAME {\tt `` {\tt AppData {\tt ¥Roaming {\tt ¥Microsoft {\tt Windows {\tt ¥Start Menu {\tt ¥Programs {\tt ¥Startup and {\tt YS} } } } }}} } } } } } } } }$ 

- 「USERNAME」はユーザ名です。
- ・「AppData」が表示されない場合、エクスプローラのメニューで「隠しファイル」にチェックを入れます。
- 2. スタートアップフォルダに IoT-Monitor のショートカットをコピーしてください。

<b>GO</b> • <b>•</b> • =	2- <del>1</del> 7- •	AppData   Roamin	g ▶ Microsoft ▶ V	Windows 🖡 スタート メニュー	・・ プログラム ・ ジ	スタートアップ	
整理 ▼ ライフ	ブラリに追加 ▼ 共有	- 書き込む 業	fしいフォルダー				
34 34 34	UProof Vault VisualStudio	◆ 名前	nmer Ver1.61	更新日時 22/11/07 11:04	種類 ショートカット	サイズ 2 KB	
	VsGraphics						

# 2.タイムアウト設定

コントローラと通信を行う際、タイムアウト(=通信エラー)になる時間を設定します。 デフォルトは 3000ms に指定されています。通常は変更する必要はありません。

タイムアウトの設定					
応答待ち時間の設定 3000 🖢 mg	3				
デフォルト :3000ms					
通信の待ち時間を変更することができます。 通信でこの値を超過すると、通信エラーが発生します。					
ОК <b>*</b> *>±21+					

「通信に 3000ms 以上かかるため、頻繁に通信エラーになってしまう」などの場合で、 タイムアウト時間を伸ばすことができます。

タイムアウト時間が長くなるほど通信エラーは発生しにくくなりますが、IOの取得漏れが 発生しやすくなります。

例:タイムアウトを 8000ms にすると、2500ms 間 ON していた IN1 の情報を取得できない

※以下のように PC とコントローラを直接接続した場合は、通常 10ms 以内に応答があります。



無線接続の場合や混雑したネットワークに接続した場合などに、時間がかかる可能性があります。

#### 3.言語設定

表示言語を設定します。日本語と英語を選択できます。

言語設定		
言語	JAPANESE	
ОК	キャンセル	

# 4.通信切断時のメール送信設定

コントローラと通信中、通信が切断された場合にメールを送信することができます。メール送信の仕組 みについてはメールファンクション(→p.18)を参照ください。

𝑀 通信失敗メール設定	_	$\times$
<ul> <li>✓ 通信切断時にメールを送信する</li> <li>✓ 送信サーバ設定</li> <li>グ 然の1</li> <li>✓ SiO01</li> <li>✓ SiO02</li> <li>▼ SiO02</li> <li>♥ Sio</li></ul>	切断されまし	
OK キャンセル		.:

送信サーバ設定	メールを送信するアドレスのサーバを設定します (→p.19)。
対象コントローラ	メールを送信する条件です。 チェックしたコントローラが切断されたときに メールを送信します。
送信先コントローラ	メールの送信先です。 10アドレス設定できます。すべて宛先(To)で送信されます。
送信内容	送信する内容です。件名は固定です。

# 改版履歴

版	年月日	内 容	変更ページ	
1.00	<u>'21/08/19</u>	第1版制定		
1.10	'21/10/29	ファンクション(パワーポイント)を追加	p. 8 / p. 23	
		OUT出力に関する記述を追加	p.31 - p.38	
1.20	<sup>•</sup> 21/12/27	ランプファンクションの画像差し替え	p.12	
		ファンクション(マクロ)を追加	p.24	
		ボタンファンクションの色を追加	p.32	
		固定テキストの説明を修正	p.39	
1.30	<i>22/2/1</i>	ファンクション(共有メモリ)を追加	р.26	
			p.36	
1.40	<i>22/4/26</i>	動画ファンクションに録音を追加	p.20	
1.50	<i>22/6/30</i>	ファンクション(ファイル実行)(I0一括保存)を追加	p. 28 – p. 30	
		マクロファンクションに設定項目を追加	p.24	
		メモ読込/削除ボタンを追加	p.7	
1.51	·22/11/25	Windows11に対応	p. 5	
		警告ファンクションに画面種類・閉じる機能を追加	р.18	
		ストップウォッチファンクションに「リセット条件」を	p.22	
		追加		
		オプションを追加	p. 47 – p. 48	
1.60	<sup>•</sup> 23/2/3	一括モニタを追加	p. 7	
		保存項目「ミリ秒」を削除	p. 13/ p. 16/ p. 22/ p. 29	
		共有メモリ生成を追加	p.26 / p.36	
1.70	<u>'23/5/31</u>	ファンクションの順番を変更	p. 8 – p. 9	
		カウンタファンクションに「一定カウントで出力」追加	p. 11	
		ストップウォッチの設定項目を変更	p. 13	
		コントローラ→エクセル出力ファンクション追加	p.29 – p.30	
		バーコードファンクションに「出力先」追加	p. 36	
		エクセル→コントローラ出力ファンクションを追加	p. 39	
		バーコード値のcsv出力事例を追加	p. 50 – p. 51	
1.71	<sup>•</sup> 23/6/7	MiOコントローラの対応バージョン表記を追加	p. 5	
1.80	<i>24/4/30</i>	クラウドカメラファンクションを追加	p. 8 / p. 23 - p. 24	
		歯車マークを追加(カウンタ・インジケータ・ストップウォッチ)	p. 10 - p. 13	
		Csvモニタ追加(ログ保存、一括ログ保存)	p.14 / p.16	
		エクセル連携ファンクションの注意事項を追加	p. 32	
		警告画面ファンクションのフォントサイズ調整を追加	p. 21	
1.90	<i>25/6/6</i>	クラウドカメラファンクションの時間指定を追加	P. 23	
		PDFファンクションの追加	P. 28	
		通信切断時のメール送信設定を追加	P. 57	
1.91	<i>25/6/10</i>	メモ読込みに関する記述を追加	P.7 / P.14	