

# ユーザー マニュアル



## ADAM-6000シリーズ

イーサネットベースのデータ収集および制御モジュール

**ADVANTECH**

*Enabling an Intelligent Planet*

## 著作権

本製品に含まれるドキュメントおよびソフトウェアの著作権はアドバンテック株式会社に帰属します。すべての権利はアドバンテック株式会社に帰属します。アドバンテック株式会社は、本マニュアルに記載された製品を予告なく改良する権利を。本マニュアルのいかなる部分も、アドバンテック株式会社の事前の書面による許可なく、いかなる形式または手段によっても、複製、複写、翻訳、または転送することを禁じます。本マニュアルに記載されている情報は、正確で信頼できるものであることを保証するものではありません。本マニュアルに記載されている内容は、正確で信頼できるものですが、使用によって生じる第三者の権利侵害について、アドバンテックは一切責任を負いません。

## 謝辞

IntelおよびPentiumはIntel Corporationの商標です。

Microsoft WindowsおよびMS-DOSはMicrosoft Corp. の登録商標です。その他の製品名または商標は、各所有者の財産です。

## 製品保証

アドバンテックは、購入日より2年間、各製品に材料および製造上の欠陥がないことを保証します。

本保証は、アドバンテックが認定した修理担当者以外によって修理または改造された製品、または誤用、乱用、事故、不適切な設置が行われた製品には適用されません。アドバンテックは、このような事象が発生した場合、本保証の条項に基づく責任を負いません。

アドバンテックの高い品質管理基準と厳格なテストにより、ほとんどのお客様は修理利用する必要はありません。アドバンテック製品の不具合につきましては、保証期間内であれば無償で修理・交換させていただきます。保証期間外の修理につきましては、交換品代金、修理時間、運賃をご請求させていただきます。詳しくは販売店にご相談ください。

製品に欠陥があると思われる場合は、以下の手順に従ってください。

1. 発生した問題に関するすべての情報を収集する。(例えば、CPU速度、使用したアドバンテック製品、使用した他のハードウェアやソフトウェアなど) 異常があればメモし、問題が発生したときに表示された画面上のメッセージをリストアップしてください。
2. 販売店に電話し、問題を説明してください。取扱説明書、製品、および役立つ情報をすぐに入手できるようにしてください。
3. 製品に欠陥があると診断された場合は、販売店から返品確認番号 (RMA) を取得してください。これにより、より迅速に返品処理を行うことができます。
4. 不具合のある製品、必要事項を記入した修理・交換注文書、購入日を証明する書類 (レシートのコピーなど) を、発送可能な容器に丁寧に梱包してください。購入日を証明するものがない場合、保証サービスを受けることはできません。
5. パッケージの外側にRMA番号を明記し、梱包材を元払いで販売店に発送してください。

部品番号 2003600006

12版

台湾で印刷

2022年11月

## 適合宣言

### CE

本製品は、外部配線にシールドケーブルを使用した場合の環境仕様に関するCE試験に合格しています。シールドケーブルの使用を推奨します。このタイプのケーブルはアドバンテックから入手可能です。ご注文については、お近くの販売店にお問い合わせください。

合格の試験条件には、機器が工業用エンクロージャ内で動作することも含まれます。静電気放電（ESD）やEMI漏洩による損傷から製品を保護するため、CE準拠の産業用エンクロージャ製品の使用を強く推奨します。

### FCCクラスA

本装置は、FCC 規則パート 15 に従ったクラス A デジタルデバイスの制限に準拠していることが、テストにより確認されています。これらの制限は、本機器が商用環境で使用される場合に、有害な干渉から妥当に保護することを目的としています。本装置は、無線周波数エネルギーを発生、使用、放射する可能性があり、取扱説明書に従って設置および使用されない場合、無線有害な干渉を引き起こす可能性があります。住宅街でこの機器を操作すると、有害な干渉を引き起こす可能性があります。この場合、ユーザーは自己負担で干渉を修正する必要があります。

## 技術サポートと支援

1. アドバンテックのウェブサイト [www.advantech.com/support](http://www.advantech.com/support)、最新の製品情報をご覧ください。
2. 本製品をお買い上げの販売、またはアドバンテックのカスタマサービスセンターにご連絡ください。お問い合わせの前に、以下の情報をご用意ください：
  - 製品名とシリアル番号
  - 周辺アタッチメントの説明
  - ソフトウェアの説明（オペレーティングシステム、バージョン、アプリケーション・ソフトウェアなど）
  - 問題の完全な説明
  - エラーメッセージの正確な文言

## 安全上のご注意

1. この安全に関する指示をよくお読みください。
2. このユーザーマニュアルは、今後のために保管してください。
3. クリーニングの前に、すべての電源コンセントから装置を外してください。クリーニングには湿らせた布のみを使用してください。液体やスプレー式の洗剤は使用しないでください。
4. プラグ式機器の場合、コンセントソケットは機器の近くにあり、容易にアクセスできる必要があります。
5. 機器を湿気から守る。
6. 設置の際は、信頼できる場所に設置してください。装置を落としたり、落下させたりすると、破損の原因となります。
7. 筐体の開口部は空気の対流用です。装置を過熱から保護してください。開口部を塞がないでください。
8. 機器をコンセントに接続する前に、電源の電圧が正しいことを確認してください。
9. 電源コードは人通りの多い置かないでください。電源コードの上に物を置かないでください。
10. 機器に記載されているすべての注意と警告に留意すること。
11. 装置を長期間使用しない場合は、過渡過電圧による損傷を避けるため、電源から切り離してください。
12. 開口部に液体を入れないでください。火災や感電の原因となります。
13. 絶対に装置を開けないでください。安全上の理由から、有資格のサービス担当者以外は装置を開けないでください。
14. 次のような場合は、サービスマンに機器を点検してもらってください：
  - 電源コードまたはプラグが損傷している。
  - 液体が装置に浸透した。
  - 機器が湿気にさらされている。
  - 機器が故障している、または取扱説明書通りに動作しない。
  - 機材が落下して破損した。
  - 機材は明らかに壊れた形跡がある。
15. 保管温度が $-20^{\circ}\text{C}$  ( $-4^{\circ}\text{F}$ ) 未満または  $60^{\circ}\text{C}$  ( $140^{\circ}\text{F}$ ) を超える環境に装置を放置しないでください。装置は管理された保管してください。
16. 注意：バッテリーは、不適切に交換すると破裂する危険性があります。製造元が推奨する同一タイプまたは同等タイプのみと交換してください。使用済みの電池は、製造元の廃棄してください。
17. IEC 704-1:1982仕様に準拠し、オペレーター位置での音圧レベルは70 dB (A) を超えません。

本取扱説明書は IEC 704-1 規格に準拠しています。アドバンテックは、本書に含まれる記述の正確さについて一切の責任を負いません。

# 内容

<b>1</b>	<b>システムを理解する</b>	<b>1</b>
1.1	はじめに	2
	図 1.1 ADAM-6000 モジュール・システム・アーキテクチャ	2
1.2	主な特徴	2
1.3	仕様	4
1.4	寸法図	4
	図 1.2 ADAM-6000 モジュールの寸法	4
1.5	LEDステータス	5
<b>2</b>	<b>ハードウェア選択のガイドライン</b>	<b>7</b>
2.1	I/Oモジュールの選択	8
	表 2.1: I/O選択のガイドライン	8
2.2	リンク端子とケーブルの選択	9
	図 2.1 ADAM-6000 モジュールとイーサネット・ターミナルのケーブル接続	9
	表 2.2: イーサネット RJ-45 ポート・ピン配列表	9
2.3	オペレータインタフェースの選択	10
<b>3</b>	<b>ハードウェアインストールガイド</b>	<b>11</b>
3.1	インターフェース紹介	12
3.2	取付オプション	13
	3.2.1 パネルマウント	13
	図 3.1 パネル取付けブラケットの寸法	13
	図 3.2 取付ブラケットへのモジュールの固定方法	13
	3.2.2 DINレール取付	14
	図 3.3 DIN レールアダプターにモジュールを固定する方法	14
	図 3.4 DIN レールにモジュールを固定する方法	14
3.3	配線と接続	15
	3.3.1 電源配線	15
	図 3.5 モジュール電源線の接続方法	15
	3.3.2 I/Oモジュール配線	16
<b>4</b>	<b>アナログ ADAM-6000 I/O モジュールの紹介</b>	<b>17</b>
4.1	アナログ入力モジュール	18
4.2	ADAM-6015 7チャンネル絶縁RTD入力モジュール	18
	4.2.1 仕様	18
	4.2.2 アプリケーション配線	20
	図 4.1 ADAM-6015 RTD 入力配線	20
	4.2.3 住所割り当て	20
4.3	ADAM-6017 8チャンネルアナログ入力/2チャンネルデジタル出力モジュール	20
	4.3.1 仕様	20
	図 4.2 ADAM-6017 ジャンパー・スイッチ	22
	4.3.2 アプリケーション配線	23
	図 4.3 ADAM-6017 アナログ入力配線	23
	図 4.4 ADAM-6017 アナログ入力タイプ設定	23
	図 4.5 ADAM-6017 デジタル出力配線	23
	4.3.3 住所割り当て	24

4.4	ADAM-6018 Isolated Thermocouple Input/8-ch Digital Output Module .. 24	Figure 4. 6
	ADAM-6018 8-ch Thermocouple Input.....	24
4.4.1	仕様 .....	25
4.4.2	アプリケーション配線 .....	26
	図 4. 7 ADAM-6018 熱電対入力配線.....	26
	図 4. 8 ADAM-6018 デジタル出力配線.....	26
4.4.3	住所割り当て .....	26
4.5	ADAM-6018+ 8チャンネル絶縁熱電対入力モジュール .....	27
4.5.1	仕様 .....	27
4.5.2	アプリケーション配線 .....	28
	図 4. 9 ADAM-6018+ 熱電対の配線.....	28
4.6	ADAM-6024 12チャンネル絶縁ユニバーサルI/Oモジュール .....	28
4.6.1	仕様 .....	28
	図 4. 10ADAM-6024-A1E ジャンパー設定.....	30
	図 4. 11ADAM-6024-D ジャンパー設定.....	30
4.6.2	アプリケーション配線 .....	31
	図 4. 12ADAM-6024 アナログ I/O 配線.....	31
	図 4. 13ADAM-6024 デジタル入力配線.....	31
	図 4. 14ADAM-6024 デジタル出力配線.....	32
4.6.3	住所割り当て .....	32

## 5

### デジタルADAM-6000 I/Oモジュールの紹介33

5.1	デジタルI/Oおよびリレーモジュール .....	34
5.2	ADAM-6050 18チャンネル絶縁デジタルI/Oモジュール .....	34
5.2.1	仕様 .....	34
5.2.2	アプリケーション配線 .....	35
	図 5. 1 ADAM-6050 デジタル入力配線.....	35
	図 5. 2 ADAM-6050 デジタル出力配線.....	35
5.2.3	住所割り当て .....	36
5.3	ADAM-6051 2チャンネルカウンタ付き14チャンネル絶縁デジタルI/Oモジュール.....	36
5.3.1	仕様 .....	36
5.3.2	アプリケーション配線 .....	37
	図 5. 3 ADAM-6051 デジタル入力配線.....	37
	図 5. 4 ADAM-6051 カウンター（周波数）入力.....	38
	図 5. 5 ADAM-6051 デジタル出力配線.....	38
5.3.3	住所割り当て .....	38
5.4	ADAM-6052 16チャンネルソースタイプ絶縁デジタルI/Oモジュール.....	38
5.4.1	仕様 .....	39
	図 5. 6 ADAM-6052 ジャンパ設定.....	40
5.4.2	アプリケーション配線 .....	41
	図 5. 7 ADAM-6052 デジタル入力配線.....	41
	図 5. 8 ADAM-6052 デジタル出力配線.....	42
5.4.3	住所割り当て .....	42
5.5	ADAM-6060 6チャンネルデジタル入力/6チャンネルリレーモジュール.....	42
5.5.1	仕様 .....	42
5.5.2	アプリケーション配線 .....	44
	図 5. 9 ADAM-6060 デジタル入力配線.....	44
	図 5. 10ADAM-6060 リレー出力配線.....	44
5.5.3	住所割り当て .....	44
5.6	ADAM-6066 6チャンネルデジタル入力/6チャンネルパワーリレーモジュール.....	45
5.6.1	仕様 .....	45
5.6.2	アプリケーション配線 .....	46
	図 5. 11ADAM-6066 デジタル入力配線.....	46
	図 5. 12ADAM-6066 リレー出力配線.....	46
5.7	デジタル出力診断機能.....	47
5.7.1	デジタル出力診断ステータスの取得方法.....	48
	図 5. 13異常 DO 診断ステータス.....	48

## 6 システム構成ガイド..... 51

6.1	システム要件.....	52
6.2	Adam/Apax .NET Utilityのインストール.....	52
6.3	Adam/Apax .NET Utilityの概要.....	52
	図 6.1 Adam/Apax .NET Utility操作ウィンドウ.....	53
6.3.1	メニューバー.....	53
6.3.2	ツールバー.....	54
	図 6.2 Adam/Apax .NET Utilityツールバー.....	54
6.3.3	Module Tree表示領域.....	55
	図 6.3 Adam/Apax .NET Utility・モジュールの表示領域.....	55
6.3.4	ステータス表示エリア.....	55
6.3.5	ADAM-6000 モジュールの構成.....	56
	図 6.4 Adam/Apax .NET Utility - デバイスの検索.....	56
6.3.6	グループ構成.....	64
6.3.7	I/O構成.....	66
	図 6.5 全チャンネル、個別チャンネル、および GCL 設定コントロール.....	66
6.4	アナログ入力モジュール (ADAM-6015、ADAM-6017、およびADAM-6018、ADAM-6018+).....	67
6.4.1	全チャンネル構成.....	67
	図 6.6 チャンネル範囲設定エリア.....	67
	図 6.7 アナログ入力トレンドログ.....	70
6.4.2	個別チャンネル構成.....	72
	図 6.8 アナログ入力アラームモードの構成.....	72
6.5	ユニバーサルI/Oモジュール (ADAM-6024).....	74
6.5.1	全チャンネル構成.....	74
	図 6.9 ADAM-6024 チャンネル構成.....	74
	図 6.10 ADAM-6024 出力タブ.....	75
6.6	ユニバーサルデジタルI/Oモジュール (ADAM-6050, ADAM-6051- ADAM-6052, ADAM-6060, ADAM-6066).....	75
6.6.1	全チャンネル構成.....	75
6.6.2	個別チャンネル構成.....	77
	図6.11 デジタル入力モード.....	77
	図6.12 デジタル出力モード.....	80
	図6.13 Low-High遅延出力モードを説明するグラフ.....	82
	図6.14 Low-High遅延出力モードの説明グラフ.....	83
6.7	P2P機能の紹介.....	84
6.7.1	P2P通信モード.....	84
	図6.15 P2Pの基本モード.....	84
	図6.16 P2Pのアドバンスモード.....	85
6.7.2	P2P通信方式.....	85
6.7.3	P2Pイベントトリガー.....	86
6.8	P2P機能の設定方法.....	86
	図6.17 ピアツーピア/イベントタブ.....	86
6.8.1	基本モード設定.....	87
	図 6.18 P2P 基本モードの構成.....	87
6.8.2	アドバンスモード設定.....	88
	図6.19 P2Pアドバンスモードの設定.....	88
	図6.20 設定を他のチャンネルにコピーする.....	89
6.9	ADAM-6000 ウェブサーバ.....	90
6.9.1	HTML 5.....	90

## 7 アプリケーション・プログラムの計画 93

7.1	はじめに.....	94
7.2	ADAM .NET CLASS LIBRARY.....	94

	図 7.1 ADAM-6050 .NET の修正.....	95
	図 7.2 サンプルコードの実行と ADAM モジュールの設定.....	95
7.3	ADAM-6000 モジュール用 Modbus プロトコル.....	96
7.3.1	Modbus プロトコル構造.....	96

	7.3.2	Modbusファンクションコードの紹介	96
7.4		ADAM-6000 モジュール用 ASCII コマンド	101
	7.4.1	ASCII構文	101
	7.4.2	システム・コマンド・セット	101
	7.4.3	アナログ入力コマンドセット	106
	7.4.4	アナログ入力アラームコマンドセット	117
	7.4.5	ユニバーサルI/Oコマンドセット	122
	7.4.6	デジタル I/O コマンドセット	129
7.5		ADAM-6000 モジュール用 SNMP	132
	7.5.1	ADAM MIBファイル	132
	7.5.2	SNMP トラップの設定	132
		図 7.3 Adam/Apax .NET Utilityを使用したトラップ設定	133
		OID 値	134
7.6		ADAM-6000モジュール用MQTT	135
	7.6.1	MQTTの紹介	135
	7.6.2	ADAMモジュールのMQTTフォーマット	135
	7.6.3	MQTTの設定	140
	7.6.4	ADAM-6000モジュールでMQTTを開始する方法	145
	7.6.5	リアルタイムクロック	147
	7.6.6	Adam/Apax .NET UtilityによるSNTP設定	147
	7.6.7	ASCII コマンドを使用した SNTP 設定	148

## 8 グラフィック・コンディション・ロジック (GCL) ... 149

8.1		概要	150
8.2		GCL構成環境	150
		図 8.1 GCL 構成環境	151
		図8.2 1つの論理ルールに対する4つの段階	152
8.3		論理ルールの4段階の設定	153
	8.3.1	入力条件ステージ	153
		図 8.3 入力条件ステージの構成	154
		図 8.4 アナログ入力モードのスケールリング機能	154
		図 8.5 エンジン・ユニットと電流値	155
	8.3.2	ロジックステージ	157
		図 8.6 ロジック・ステージの構成	157
	8.3.3	実行段階	158
		図 8.7 実行ステージの構成	159
		図 8.8 次のルールに送る機能	160
		図8.9 次の論理ルール	160
	8.3.4	出力ステージ	161
		図8.10出力ステージの構成	161
		図8.11リモート・メッセージ出力	164
8.4		ロジック・カスケードとフィードバックのための内部フラグ	166
	8.4.1	ロジック・カスケード	166
		図 8.12ローカル・ロジック・カスケード・アーキテクチャ	166
		図8.13 論理ルール1の構成	166
		図8.14論理ルール2の構成	167
		図8.15論理ルール3の構成	167
		図8.16分散ロジック・カスケード	168
		図8.17 論理ルール1の構成	168
		図8.18 論理ルール2の構成	168
		図8.19 論理ルール3の構成	168
	8.4.2	フィードバック	169
		図8.20ロジック・フィードバックの構築	169

8.5	ロジックダウンロードとオンラインモニタリング	169
	図8.21 オンラインモニタリング機能	170
	図 8.22 GCL 実行シーケンス	171
8.6	GCLの代表的な用途	171
	図8.23 オン/オフ制御のラダー図	172
	図8.24 オン/オフ制御のGCLロジック	172
	図 8.25 シーケンス制御のタイムチャート	173
	図 8.26 シーケンス制御用 GCL ロジック (シーケンスでオンし、オン状態を維持する)	173
	図8.27 12デジタル入力から1デジタル出力までのタイムチャート	174
	図8.28 12デジタル入力から1デジタル出力までのGCLロジック	174
	図 8.29 フリッカ・アプリケーションのタイムチャート	175
	図8.30 フリッカ用GCLロジック	175
	図 8.31 立ち上がりエッジのタイムチャート	175
	図8.32 立ち上がりエッジのラダー図	176
	図 8.33 立ち上がりエッジ用 GCL ロジック	176
	図 8.34 立ち下がりエッジのタイムチャート	177
	図8.35 立ち下がりエッジのラダー図	177
	図 8.36 立ち下がりエッジの GCL ロジック	177
	図 8.37 シーケンス制御のタイムチャート(オンとオフを繰り返す)	178
	図 8.38 シーケンス制御のための GCL ロジック (連続してオンオフ)	178
	図 8.39 イベントトリガの GCL ロジック (1 回のみ発生)	179
	図8.40 イベントトリガ設定 (1回のみ発生)	179

## 付録A デザイン・ワークシート ..... 181

## 付録B データフォーマットとI/O範囲..... 185

B.1	ADAM-6000 コマンド・データ・フォーマット	186
	図 B.1 リクエストコメントの構造	186
	図 B.2 回答コメントの構成	186
B.2	ADAM-6000 I/O Modbus マッピングテーブル	191

## 付録C 接地リファレンス ..... 221

C.1	フィールド接地とシールドの適用	222
	C.1.1 概要	222
C.2	接地	222
	C.2.1 リファレンスとしての地球	222
	図C.1 グラウンドとしての地球を考える	222
	C.2.2 フレームグラウンドとアースバー	223
	図 C.2 アースバー	223
	図 C.3 図 C.3 : ノーマルモードとコモモード	223
	C.2.3 ノーマルモードとコモモード	223
	図C.4 ノーマルモードとコモモード	224
	C.2.4 ワイヤインピーダンス	224
	図C.5 高圧送電	224
	図C.6 ワイヤ・インピーダンス	225
	C.2.5 一点接地	225
	図 C.7 シングルポイント接地	225
	図 C.8 単一点接地	225
C.3	シールド	226
	C.3.1 ケーブル・シールド	226
	図C.9 シングル絶縁ケーブル	226
	図C.10 二重絶縁ケーブル	226

C.32	システム・シールド .....	227
	図C. 11システム・シールド.....	227
	図C. 12ケーブルの特性 .....	227
	図C. 13システム・シールド(1).....	228
	図C. 14システム・シールド(2).....	228
C.4	ノイズ低減テクニック.....	228
	図C. 15ノイズ除去テクニック.....	229
C.5	チェックリスト.....	229

## 付録D

## ADAM-6000のREST ..... 231

D.1	REST入門.....	232
D.2	ADAM用RESTリソース.....	232
	D.2.1 アナログ入力 .....	232
	D.2.2 アナログ出力 .....	233
	D.2.3 デジタル入力 .....	234
	D.2.4 デジタル出力 .....	235
	D.2.5 カウンター .....	236

# 1

システムを理解する

## 1.1 はじめに

ADAM-6000シリーズのイーサネットベースのデータ収集と制御（DA&C）モジュールは、1つのモジュールでI/O、データ収集、およびネットワーク機能を提供しさまざまなアプリケーションのためのコスト効率の高い分散監視および制御ソリューションを構築できます。標準的なイーサネット・ネットワークを通して、ADAM-6000モジュールはセンサからI/O値を取得し、LAN、イントラネット、またはインターネットを介して、ネットワーク・ノードにリアルタイムのI/O値として公開することができます。イーサネット対応テクノロジーにより、ADAM-6000モジュールは、ビル・オートメーション、環境モニタリング、設備管理、および電子製造アプリケーション用のコスト効率の高いDA&Cシステムを構築することができます。図1-1に、ADAM-6000モジュールに採用できるシステム・アーキテクチャの概要を示します。

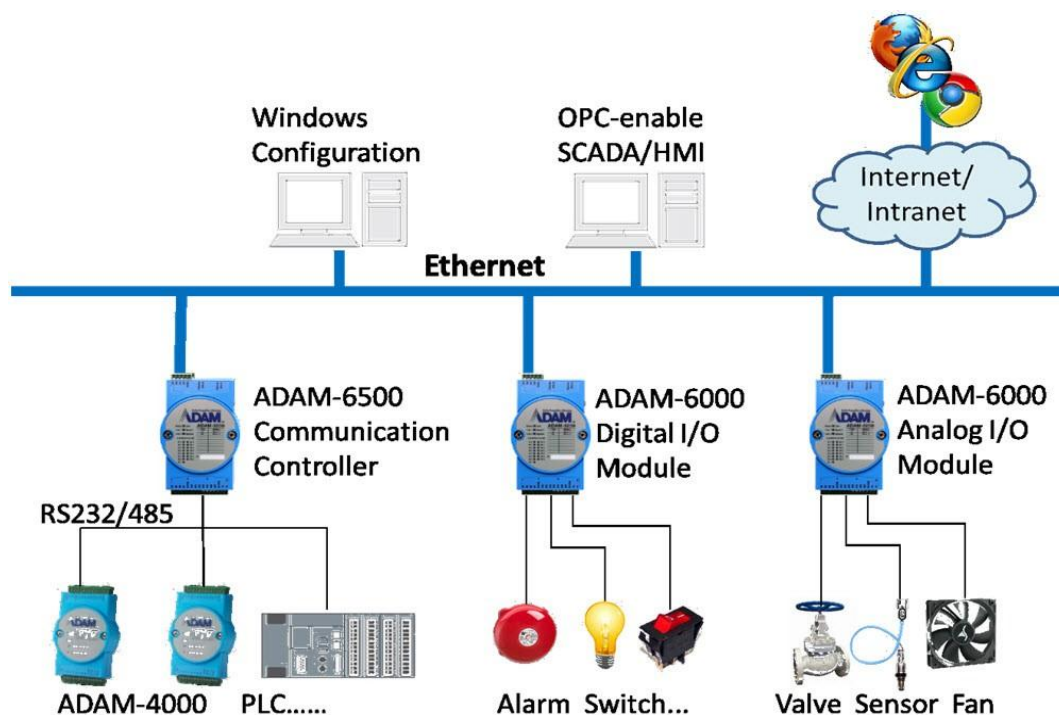


図 1.1 ADAM-6000 モジュール・システム・アーキテクチャ

## 1.2 主な特徴

### イーサネット対応DA&C I/Oモジュール

ADAM-6000モジュールは、ほとんどのビジネス環境で採用されている、広く利用されているイーサネットネットワーク規格に基づいています。ADAM-6000シリーズI/Oモジュールを既存のEthernetネットワークに簡単に追加したり、Ethernet対応の新しいe-manufacturingネットワークで使用することができます。このシリーズのモジュールは、データ接続用に10/100 Mbps EthernetとModbus/TCP over TCP/IP、UDP over Ethernetをサポートしています。UDP/IPにより、ADAM-6000シリーズI/Oモジュールは、最大8つのEthernetノードにデータストリームをアクティブに送信できます。イーサネットネットワーク、HMI/SCADAシステム、コントローラを介して、ADAM-6000 Ethernet対応DA&Cモジュールからリアルタイムデータにアクセスし、取得することができます。そして、そのデータをビジネスシステムと統合して、価値あるビジネス情報を引き出すことができます。

#### 注意



一部のインテリジェント機能はADAM-6000-CEバージョンでのみ提供されません。詳細については付録Fを参照してください。

## インテリジェントI/Oモジュール

従来のI/OモジュールからアップグレードされたADAM-6000シリーズの全モジュールには、システム機能を強化できるインテリジェント機能があらかじめ組み込まれています。例えば、デジタル入力モジュールはカウンタと積算機能を提供し、デジタル出力モジュールはパルス出力と遅延出力機能を提供し、アナログ入力モジュールは記述統計データ計算（例えば、最小、最大、平均）を提供し、アナログ出力モジュールはPIDループ制御機能を提供します。

## あらゆるアプリケーションに対応するミックスドI/O

ADAM-6000シリーズのミックスI/Oデザインは、アプリケーションシステムにコスト効率の良いI/Oオプションを提供します。シングルファンクションユニットのために最も一般的に使用されるI/Oタイプは、単一のモジュールで利用可能です。この設計コンセプトは、I/O使用コストを節約するだけでなく、I/Oオペレーションをスピードアップします。小型DA&Cシステムまたは中大規模システムのスタンドアロン制御ユニットの場合、ADAM-6000シリーズの混合I/O設計は、1つまたは2つのモジュールだけで、アプリケーションのニーズに簡単に適合できます。追加の組み込み制御モジュールを使用すると、これらのモジュールを使用して、局所的で複雑でない、より分散したI/Oアーキテクチャを簡単に作成することができます。

## 遠隔監視と診断

これまでの通信モードやデータフォーマットの違いにより、ITベースのインフラストラクチャにオートメーション制御と監視を実装することは困難でした。特に、ユーザーはデータベースやIT管理転送する前に、SCADAシステムからのI/Oデータストリームを変換するためにデータを変換する必要がありました。

ADAM-6000モジュールは、最新のWeb言語（HTML 5）とWebベースのアーキテクチャスタイル（REST）を統合し、SCADAシステムからルーティングすることなく、ユーザーが任意のスマートデバイスのWebサービスでI/Oデータをリモート取得するための基本的な認証を備えています。一例として、スマートフォンのウェブ・ブラウザを使用してHTTP経由でI/Oモジュールにリモートアクセスすることができます。

各ADAM-6000モジュールは、LANまたはインターネット経由でリアルタイムのI/Oデータ、アラーム、およびモジュールのステータスを表示するために、事前に構築されたI/OモジュールWebページを備えています。任意のポップなインターネット・ブラウザを使用して、ローカル・サイトとリモート・サイトの両方からモニタリングを実行できます。さらに、Web対応のモニタリングは、プログラミングを必要とせずすぐに完了することができます。

## Modbus/TCPプロトコル

ADAM-6000モジュールは、広く使用されている業界標準のModbus/TCPプロトコルをサポートしており、Modbus/TCPをサポートするイーサネットコントローラやHMI/SCADAソフトウェアとの接続を可能にします。また、アドバンテックはModbus/TCP用のOPCサーバーも提供しており、ADAM-6000 I/OモジュールのデータストリームをOPCクライアント対応ソフトウェアと統合することができます。

## カスタマイズされたウェブページ

ADAM-6000モジュールにはデフォルトでWebページが内蔵されているため、Internet Explorerを使用して、どこからでもI/Oステータスを監視および制御できます。さらに、個々のアプリケーション用にカスタム化されたWebページをADAM-6000モジュールにアップロードすることもできます。アドバンテックは、独自のオペレータ・インタフェースを設計し、Adam/Apax .NET Utilityを介して特定のADAM-6000モジュールにアップロードするためのリファレンスとして、JavaScript\*のサンプルコードを提供しています。

## Modbus/TCPソフトウェア・サポート

ADAM-6000モジュールのファームウェアはModbus/TCPサーバを内蔵しています。アドバンテックは、ADAM .NET CLASS LIBRARYとAdam/Apax .NET Utilityを提供します。

設定とカスタマイズこのユーティリティを使用してADAMモジュールを設定することができ、Modbus/TCPをサポートするヒューマン・マシン（HMI）ソフトウェアと統合することができます。また、Modbus/TCP設定を行うためにアドバンテックのOPCサーバーを購入することもできます。

## 1.3 仕様

イーサネット	10/100BASE-T
配線	UTP (Cat5以降)
バス接続	RJ45モジュージャック
通信プロトコル	TCP/IPとUDP上のModbus/TCP
データ転送速度	最大 100 Mbps 非レギュレーテッド 10~30 V <sub>DC</sub>
ステータス・インジケータ	電源、CPU、通信 (リンク、衝突、10/100 Mbps、Tx、Rx)
ケース	キャプティブ取付金具付きPC
ネジ端子台	対応ワイヤーサイズ：#14-28 AWG、ストリップ長：6.5 mm

### 注意



本装置は湿度30%以下で動作するように設計されていますが、湿度が低いと静電気の問題が発生しやすくなります。装置の取り扱いには十分な予防措置を講じてください。低湿度環境で装置を使用する場合は、アースストラップ、帯電防止床材、その他の保護手段を使用することをお勧めします。

## 1.4 寸法

以下の寸法はミリメートルで示されています。これらの寸法はすべての ADAM-6000 モジュールに共通です。

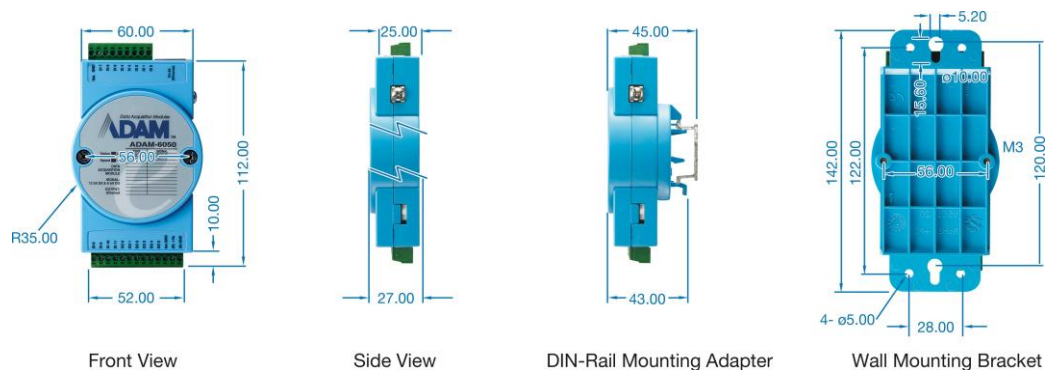
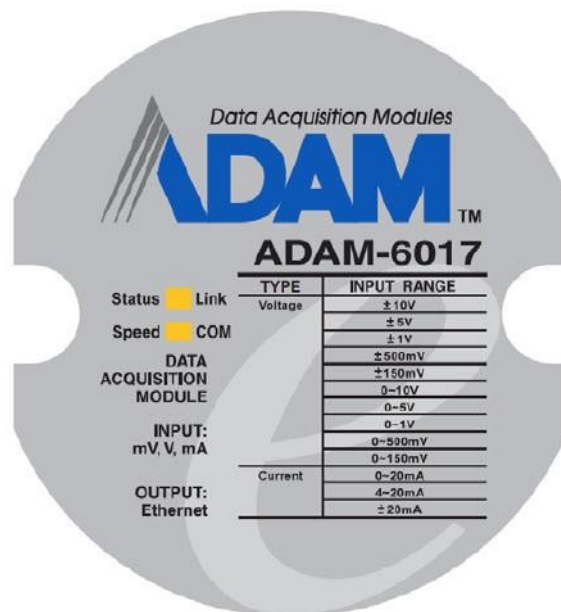








図 1.2 ADAM-6000 モジュールの寸法

## 1.5 LEDステータス

ADAM-6000 モジュールのフロントパネルには、2 つの LED があります。各LEDには、システムのステータスを表す2つのインジケータがあります：



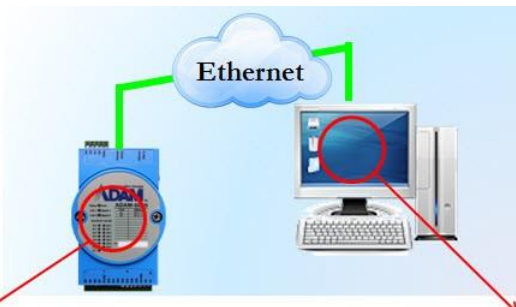
LED	カラー	カラー	表示	行動
Status	オレンジ (ステータスとリンクが同時にオンの場合) 		Red Blink	モジュールは正常に動作している
			Red On for 30s	ユーザーがLOCATE機能を有効にした場合
Link			Green On	イーサネット速度は100Mbps
Speed	オレンジ (スピードとCOMが同時にオンの場合)		Red On	イーサネット速度は100Mbps
COM				

ADAM-6050-D1、ADAM-6060-D1はLED：



LED	カラー	表示	行動
Status		Red Blink	モジュールは正常に動作している
		Red On for 30s	ユーザーがLOCATE機能を有効にした場合
Link		Green On	イーサネット速度は100Mbps
Speed		Red On	イーサネット速度は100Mbps
COM		Green Blink	モジュールがデータを送受信している

## モジュールの見つけ方

ADAM-6000 モジュールには、探している特定のモジュールを物理的に特定するためのロケート機能もあります。この機能を有効にすると、ステータス LED は 30 秒間赤色を維持します。Adam/Apax .NET Utilityでは、**Information** タブの **Enable** をクリックすることで、ロケート機能を有効にすることができます。



The diagram illustrates the physical connection of an ADAM-6000 module to an Ethernet network and a computer. A red circle highlights the module, and another red circle highlights the computer screen. Arrows point from these circles to the 'Enable' button in the software interface.


**Status**  **Link**  
**Speed**  **Com**

**Information** | Network | Stream | Administration | Firmware | Peer to Peer/Event | Access Control | Modbus Address |

Firmware Version: A1.00 E01 Locate

Device Name: ADAM-6251

Device Description:

ADAM Web Page 

Description

Slot	Module	Description
6251		ADAM-6251 16-ch isolated digital input module

# 2

## ハードウェア選択のガイドライン

## 2.1 I/Oモジュールの選択

ADAM-6000 リモート DA&C システムを構成するには、ホスト PC とフィールド・デバイスまたはセンサ間のインターフェイスとして機能する I/O モジュールを選択する必要があります。どの I/O モジュールを選択するかを決定する際には、以下のことを考慮する必要があります。

- お使いのシステムはどのようなI/O信号をいますか？
- システムに必要な入出力数は？
- 分散I/O点配置に必要なモジュールの数は？
- 設置個々のエリアでI/O処理するために、モジュールをどのように配置しますか？
- すべての接続するために必要なハブの数は？
- 各I/Oモジュールに必要な電圧範囲は？
- 各I/Oモジュールに必要な絶縁環境は？
- 各I/Oモジュールのノイズと距離の制限は？

表 2.1 に、I/O モジュールの選択に関する考慮事項の例を示します。

**表 2.1 : I/O選択のガイドライン**

I/Oモジュールのタイプ	操作例	説明
ディスクリット入力	セレクタスイッチ、プッシュボタン、フォトエレ	
モジュールとブロックI/Oモジュール	トリックアイ、リミットスイッチ、サーキットブレーカ、近接スイッチ、レベルスイッチ、モータースターター接点、リレー接点、サムホイールスイッチ	入力モジュールがON/OFFまたはOPENED/CLOSED信号を感知
ディスクリット出力モジュールおよびブロックI/Oモジュール	アラーム、コントロールリレー、ファン、ライト、ホーン、バルブ、モータースターター、ソレノイド	出力モジュールは、ON/OFFまたはOPENED/CLOSEDデバイスと相互接続します。
アナログ入力モジュール	熱電対信号、RTD信号、温度変換器、圧力変換器、ロードセル変換器、湿度変換器、流量変換器、ポテンシヨメータ。	連続アナログ信号をホスト機器の入力値に変換
アナログ出力モジュール	アナログバルブ、アクチュエーター、チャートレコーダー、電動モータードライブ、アナログメーター	ホスト・デバイスの出力を、フィールド・デバイス用のアナログ信号（一般に変換器経由）に設定する。

## 2.2 リンク端子とケーブルの選択

ADAM-6000モジュールのイーサネットポートをハブに接続するには、RJ-45コネクタを使用します。接続に使用するケーブルは、EIA/TIA 586仕様に準拠した Cat 3 (10 Mbps) または Cat 5 (100 Mbps) UTP/STP ケーブルを使用します。ハブとADAM-6000モジュール間の最大長は100m (約330フィート) です。

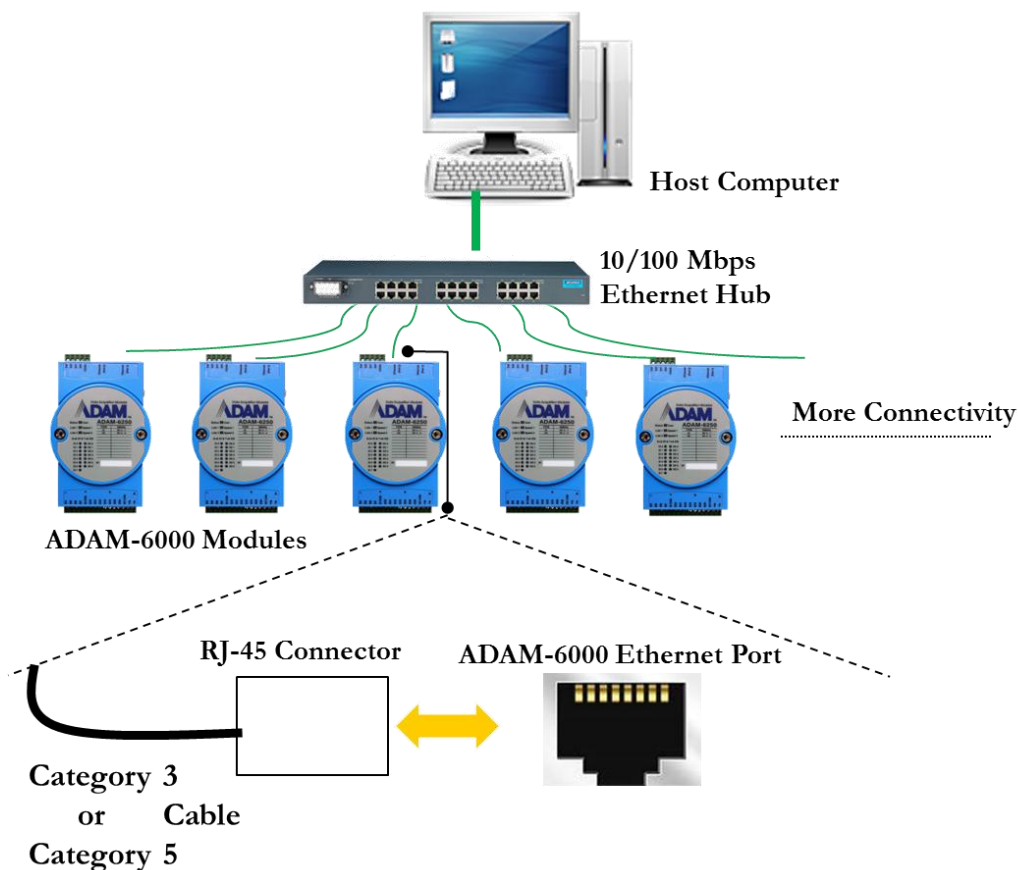


図 2.1 ADAM-6000 モジュールとイーサネットターミナルのケーブル接続

表2.2 : イーサネット RJ-45 ポート・ピン配列表

暗証番号	信号	機能
1	RD+	レシーブ (+)
2	RD	レシーブ (-)
3	TD+	送信 (+)
4	(未使用)	-
5	(未使用)	-
6	TD	送信 (-)
7	(未使用)	-
8	(未使用)	-

## 2.3 オペレーターインターフェースの選択

DA&Cシステムを完成させるには、オペレーターインターフェースを選択する必要があります。Modbus/TCPプロトコルをサポートするADAM-6000モジュールは、様々なために異なるシステムに簡単に統合することができます。

ADAM-6000モジュールのリアルタイムステータスは、以下のブラウザを使用してウェブページから読み取ることができます：

- Microsoft Internet Explorer (バージョン9以降)
- Google Chrome (バージョン30以降)
- Safari (バージョン6以降)
- Mozilla Firefox (バージョン25以降)

ADAM-6000モジュールをSCADAHMIソフトウェアと統合したい場合は、Modbus/TCPをサポートするHMIソフトウェアパッケージを使用できます。例は以下の通りです：

- Advantech PM Designer
- Wonderware InTouch
- Modbus/TCPプロトコルをサポートするすべてのソフトウェア

また、非常にユーザーフレンドリーなデータ交換ツールであるアドバンテックのOPCサーバーを購入することもできます。OPC Clientで設計されたHMIソフトウェアであれば、ADAM-6000モジュールにアクセスできます。

独自のアプリケーションを開発するには、ADAM .NET Class Libraryがユーザーインターフェースの構築に最適です。

これらのすぐに使えるソフトウェア・パッケージを使えば、リモート・データ収集、プロセス制御、履歴トレンドリング、データ分析などのタスクを、わずか数回のキー操作で利用することができる。

# 3

ハードウェア・インストール  
ガイド

## 3.1 インターフェース紹介

### パッケージ内容とシステム要件

ADAM-6000 モジュールを取り付ける前に、以下を確認してください。パッケージには以下の内容物が入っていること：

- ADAM-6000モジュール、ブラケット1個、DINレールアダプター付き
- ADAM-6000モジュール・ユーザー・マニュアル

ホストコンピュータの最小仕様は以下の通り：

- Microsoft Windows XP/7
- 32 MB RAM
- 20MBのハードディスク空き容量
- VGAカラーモニター
- CD-ROMドライブ
- マウスまたはその他のポインティングデバイス
- 10/100 Mbpsイーサネットカード

インストールを完了するには、以下の機器も必要です：

- イーサネット・ハブ（最低2ポート）
- RJ-45コネクタ付きイーサネットケーブル2本
- ADAM-6000モジュール用電源（+10～30 V、非レギュレーテッド）
- アースコネクター



## 3.2 取り付けオプション

ADAM-6000モジュールは、パネル・マウント・ブラケットまたはDINレール・マウント・ブラケットで設置できるコンパクトなユニットです。

### 3.2.1 パネルマウント

ADAM-6000 モジュールを取り付ける前に、図 3.1 に示すブラケット寸法を参照して、パネルまたはキャビネット内の最適な位置を決定する必要があります。まず、ブラケットを固定し、図 3.2 に示すように ADAM-6000 モジュールをブラケットに固定します。

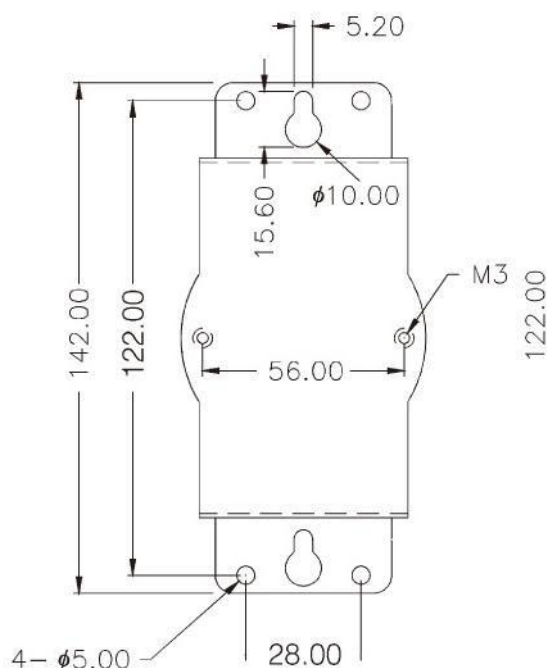


図 3.1 パネル取付けブラケットの寸法

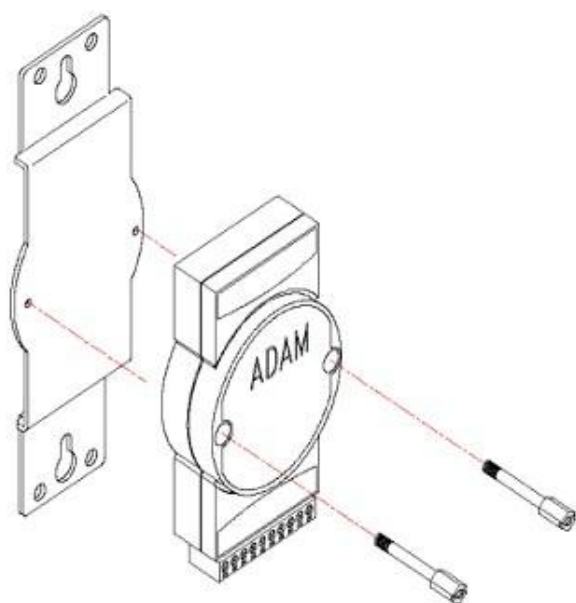


図3.2 取付ブラケットへのモジュールの固定方法

### 3.2.2 DINレールマウント

ADAM-6000 モジュールは、DIN レールを使用してキャビネットに 固定することもできます。まず、ADAM-6000 モジュールを DIN レール・アダプターに固定し（図 3-3）、次に DIN レールに固定します（図 3-4）。モジュールをレールに取り付ける際には、モジュールがスライドするのを防ぐために、レールの両端にエンド・ブラケットを使用することを考慮する必要があります。

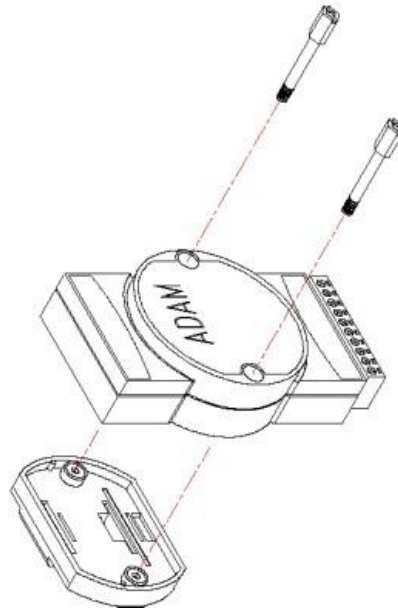


図3.3 DINレールアダプターへのモジュールの固定方法

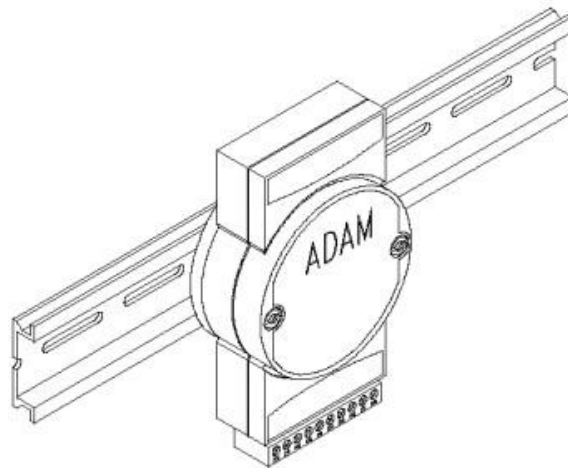
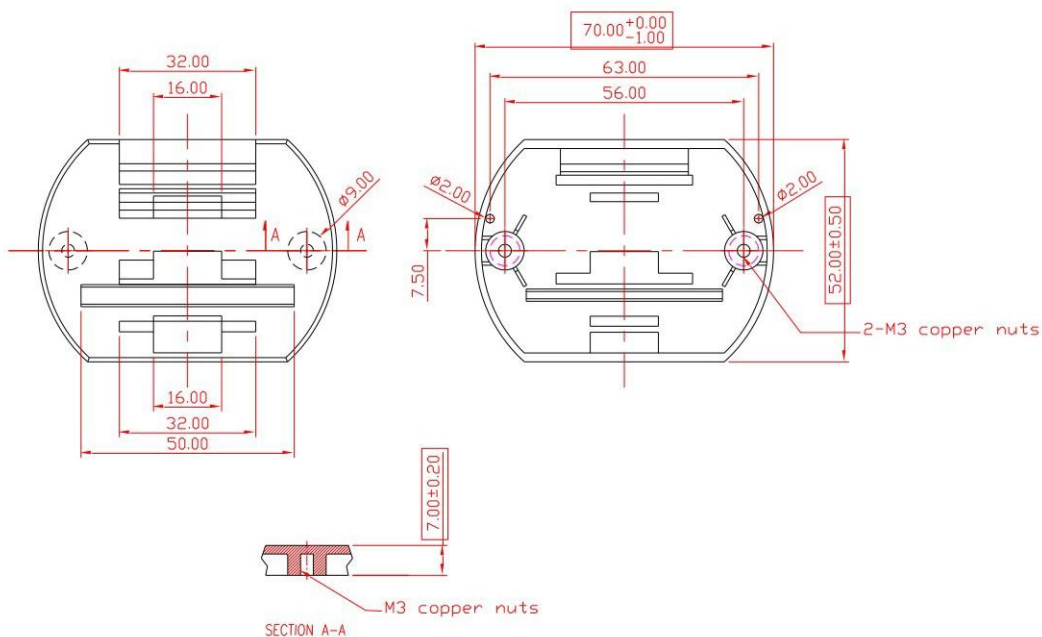


図 3.4 DIN レールにモジュールを固定する方法



### 3.3 配線と接続

このセクションでは、電源、I/Oユニット、およびネットワーク接続の配線に関する基本的な情報を提供します。

#### 3.3.1 電源配線

ADAM-6000/TCPシステムは、標準的な工業用無指定24V<sub>DC</sub>電源用に設計されていますが、+10～30V<sub>DC</sub>の範囲内の入力電力を供給する電源ユニットであれば何でも受け入れます。電源リップルはピーク・ツー・ピークで200mVに制限する必要があり、直近のリップル電圧は+10～30V<sub>(DC)</sub>の間に維持する必要があります。ネジ端子+VsとGNDは電源配線用である。

**注意** ワイヤーは直径2mm以上でなければならない。

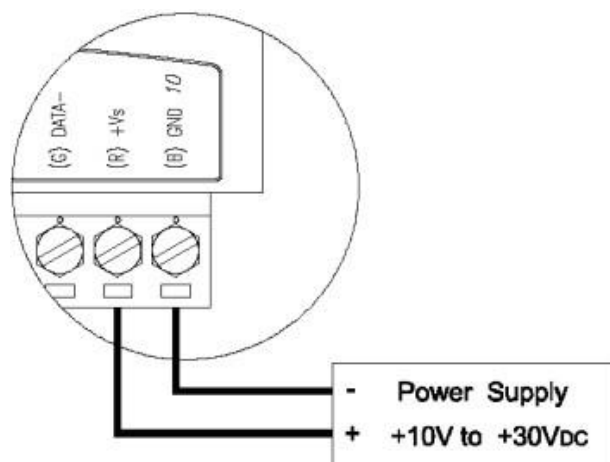


図 3.5 モジュール電源線の接続方法

電源ラインには、以下の標準色（モジュールにも表示されている）を使用することをお勧めします：

+Vs (R) 赤 GND

(B) 黒

### 3.3.2 I/Oモジュール配線

プラグインねじ端子台は、I/O モジュールとフィールドデバイスの間のインターフェイスに使用されます。I/Oモジュールに電気機器を接続する場合は、以下の情報を考慮する必要があります。

- 端子台は電線サイズ#14~28AWG（ストリップ長：6.5mm）に対応しています。
- 常に連続した長さのワイヤーを使用する。
- できるだけ短いワイヤーを使用する
- 配線には可能な限りワイヤートレイを使用する
- 高エネルギー配線の近くへの配線を避ける
- 入力配線と出力配線の近接を避ける
- ワイヤーに鋭い曲げを作らない

# 4

アナログADAM-6000 I/Oの紹介  
モジュール

## 4.1 アナログ入力モジュール

アナログ入力モジュールは、A/Dコンバーターを使用してセンサの電圧、電流、熱電対、RTD信号をデータに変換し、工学単位に変換します。ホスト・コンピュータからプロンプトが表示されると、データは標準10/100BASE-Tイーサネット経由で送信されます。現在のステータスは、Modbus/TCPをサポートするウェブページやHMIソフトウェアを使用して読み取ることができます。アナログ入力モジュールは、A/D入力の光絶縁とトランスベースの絶縁を提供することにより、グラウンドループや電源サージから機器を保護します。

## 4.2 ADAM-6015 7チャンネル絶縁RTD入力モジュール

ADAM-6015は、16ビット、7チャンネルRTD入力モジュールで、全チャンネルにプログラマブル入力レンジを備えています。さまざまなRTD入力（PT100、PT1000、Balco 500、Ni）に対応し、データは工学単位（°C）でホスト・コンピュータに送信されます。各アナログ・チャンネルは独立したレンジに設定できるため、個々のチャンネルを異なるアプリケーションで同時に使用できます。

### 4.2.1 仕様

- 通信10/100BASE-Tイーサネット
- 対応プロトコルModbus/TCP、TCP/IP、UDP、HTTP、ICMP、DHCP、ARP
- P2PとGCLをサポート（セクション6.7と第8章を参照）

#### 高速モード（DEバージョンのみ）

高速モードでは、最大サンプリング・レートは 1 kHz です（つまり、7 チャンネルを使用する場合、サンプリング・レートは  $1000/7$  となり、1 チャンネルあたり約 142 Hz となります）。これは接続されているModbusクライアントの数とイーサネットの品質に影響されません。高速モードでのパフォーマンスを最大化するために、使用されていないチャンネルは無効にしてください。

**注意** 高速モードで抵抗器をシミュレートするために校正器を使用する場合、1つ以上のチャンネルを有効にしてはなりません。



### アナログ入力

- チャンネル7 (ディファレンシャル)
- 入力インピーダンス:  $>10 \text{ M}\Omega$
- 入力接続: 2線式または3線式
- 入力タイプPt 100/1000、Balco 500、Ni 518 RTD
- RTDの種類と温度範囲:
  - Pt 100 :
    - 50~150° C
    - 0~100° C
    - 0~200° C
    - 0~400° C
    - 200~200° C
  - IEC RTD 100  $\Omega$  ( $\alpha = .0385$ )
  - JIS RTD 100  $\Omega$  ( $\alpha = 00392$ )
  - Pt 1000 : -40~160° C
  - バルコ500: -30~120° C
  - Ni 518 : -80~100° C
  - 0~100° C
- 正確さ:
  - $\pm 0.1\%$ 以上
  - $\pm 0.5\%$ 以上 (高速モード) (3線式測温抵抗体にて測定)
- スパンドリフト:  $\pm 25 \text{ ppm}/^\circ \text{C}$
- ゼロ・ドリフト:  $\pm 6 \text{ mV}/^\circ \text{C}$
- 解像度: 16ビット
- サンプル・レート (合計) :
  - 10 Hz
  - 1 kHz (高速モード、DEバージョンのみ)
- CMR @ 50/60 Hz: 90 dB (高速モードではサポートされません)
- NMR @ 50/60 Hz: 60 dB (高速モードではサポートされません)
- ワイヤ焼損検出
- 過電圧保護:  $\pm 35 \text{ V}_{\text{DC}}$
- TVS/ESD保護回路内蔵

### 一般

- ウォッチドッグ・タイマー内蔵
- 絶縁保護:  $2000 \text{ V}_{\text{DC}}$
- 電源入力  $10\sim 30 \text{ V}_{(\text{DC})}$
- 消費電力:  $2.5 \text{ W} @ 24 \text{ V}_{\text{DC}}$
- 電源反転保護
- 作動湿度: 20~95% RH (不凝縮)
- 保管湿度: 0~95% RH (不凝縮)
- 実用温度:  $-10\sim 70^\circ \text{C}$
- 保管温度:  $-20\sim 80^\circ \text{C}$

## 4.2.2 アプリケーション配線

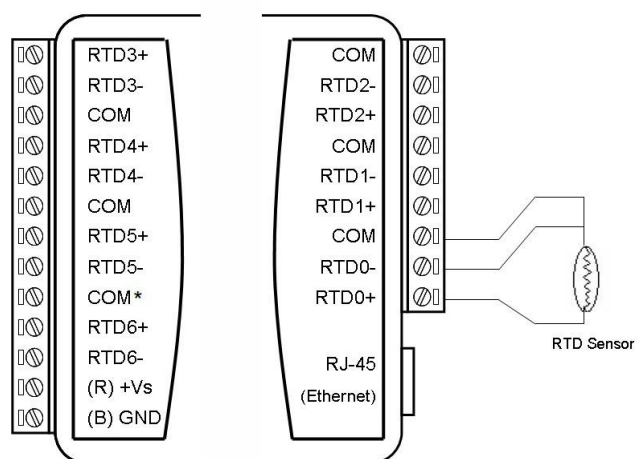


図 4.1 ADAM-6015 RTD 入力配線

注：\* RTD6とRTD5は同じCOMを共有する。

## 4.2.3 アドレス割り当て

Modbus/TCP 標準に基づき、システム内の ADAM-6000 モジュール I/O チャンネルのアドレスは単純なルールで定義されています。I/O アドレスのマッピングについては、付録 B.2.1 を参照してください。

## 4.3 ADAM-6017 8chアナログ入力/2chデジタル出力モジュール

ADAM-6017は16ビット、8チャンネルのアナログ差動入力モジュールで、全チャンネルで入力レンジをプログラマブルに設定できます。このモジュールは8つのアナログ入力と2つのデジタル出力で設計されています。入力タイプは、ミリボルト ( $\pm 150$ ,  $\pm 500$ ,  $0\sim 150$ ,  $0\sim 500$  mV)、電圧 ( $\pm 1$ ,  $\pm 5$ ,  $\pm 10$ ,  $0\sim 1$ ,  $0\sim 5$ ,  $0\sim 10$  V)、電流 ( $0\sim 20$ ,  $4\sim 20$ ,  $\pm 20$  mA) 信号で、データは工学単位 (mV, V, mA) でホスト・コンピュータに送信されます。各アナログ・チャンネルは独立したレンジに設定できるので、個々のチャンネルを異なるアプリケーションで同時に使用することができます。

### 4.3.1 仕様

- 通信10/100BASE-Tイーサネット
- プロトコルMQTT、SNMP、Modbus/TCP、TCP/IP、UDP、HTTP、ICMP、DHCP、ARP
- P2PとGCLをサポート（セクション6.7と第8章を参照）

### アナログ入力

- チャンネル8 (差動)
- 入力インピーダンス: >10 M $\Omega$  (電圧)、120  $\Omega$  (電流)
- 入力タイプ: mV、V、mA
- 入力レンジ:  $\pm 150$  mV、 $\pm 500$  mV、 $\pm 1$  V、 $\pm 5$  V、 $\pm 10$  V、0~150 mV、0~500 mV、0~1 V、0~5 V、0~10 V、0~20 mA、4~20 mA、 $\pm 20$  mA
- 正確さ:
  - $\pm 0.1\%$  of FSR (電圧) @ 25° C
  - FSR (電流) の $\pm 0.2\%$  @ 25° C
- スパンドリフト:  $\pm 25$  ppm/° C
- ゼロ・ドリフト:  $\pm 6$  mV/° C
- 解像度: 16ビット
- サンプル・レート (合計) :
  - 10 Hz
  - 100 Hz
- CMR @ 50/60 Hz: 90 dB
- NMR @ 50/60 Hz: 67 dB
- キャリブレーションオートキャリブレーション
- バーンアウト検出 (4~20mAのみ)
- 同相電圧: 350 V<sub>DC</sub>

### デジタル出力

- チャンネル: 2
- シンクタイプ: オープンコレクタ~30V、100mA (最大負荷)
- 消費電力: 各モジュール300mW
- 出力遅延オン: 100  $\mu$ s
- 出力遅延オフ: 150  $\mu$ s
- 過電圧保護 (最大): 42 V<sub>DC</sub>
- 過電流保護 (最大): 2 A
- リーク電流: Dバージョンで200  $\mu$ A (最大)

### 一般

- 絶縁保護: 2000 V<sub>DC</sub>
- 電源入力10~30 V<sub>DC</sub>
- 消費電力: 2.7 W @ 24 V<sub>DC</sub>
- 電源反転保護
- 作動湿度: 20~95% RH (不凝縮)
- 保管湿度: 0~95% RH (不凝縮)
- 動作温度 (RTC除く): -20~70° C (Dバージョンは-40~70° C)
- 保存温度 (RTC除く): -30~80° C (Dバージョンは-40~85° C)
- ウォッチドッグ・タイマー (システム): 1.6 s
- RTC (Dバージョンのみ): ISO8601フォーマット

**注意** RTC機能の動作/保管温度は-30~70°Cです。



## ジャンパ設定

ADAM-6017-CE		ADAM6017	
チャンネル番号	ジャンパ選択	チャンネル番号	ジャンパ選択
CH0	シーエヌスリー	CH0	JP6
CH1	シーエヌフォー	CH1	JP7
CH2	シーエヌファイブ	CH2	JP8
CH3	シーエヌシックス	CH3	JP1
CH4	シーエヌセブン	CH4	JP2
CH5	シーエヌエイト	CH5	JP3
CH6	シーエヌナイン	CH6	JP4
CH7	シーエヌテン	CH7	JP5

ADAM-6017 (Dバージョン) は、ジャンパ設定を簡単にするため、ケースを開けずにスイッチ操作でアナログ入力タイプを電圧/電流に設定できます。

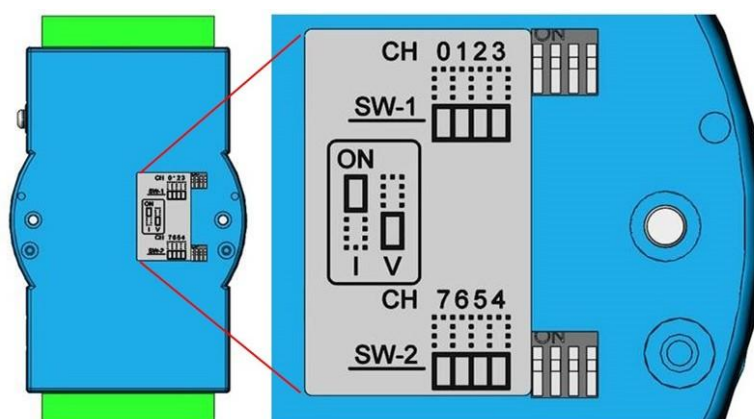
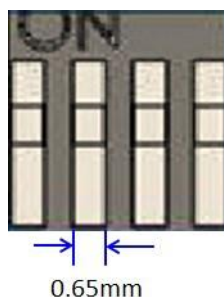


図 4.2 ADAM-6017 ジャンパー・スイッチ

スイッチ	SW1	SW2
アナログ入力チャンネル	CH0, CH1, CH2, CH3	CH4, CH5, CH6, CH7
スイッチ ON	電流入力モード	
スイッチOFF (デフォルト)	電圧入力モード	

**注意** スイッチの調整に0.65mmより太い工具を使用すると、スイッチの破損につながります。



### 4.3.2 アプリケーション配線

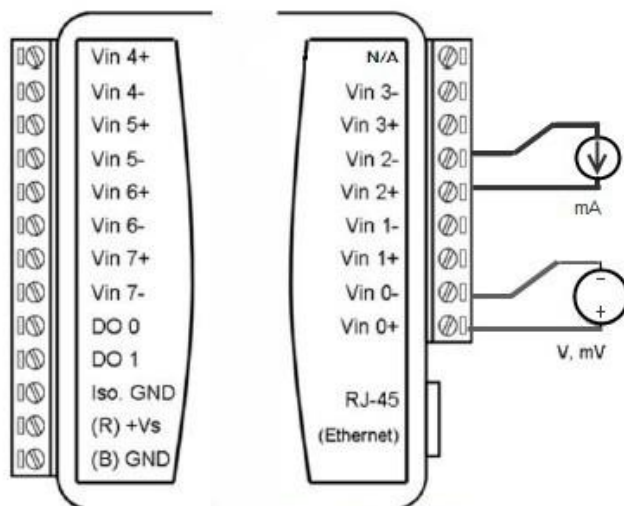


図 4.3 ADAM-6017 アナログ入力配線

ADAM-6017は各チャンネルに120Ωの抵抗を内蔵しているので、電流入力測定用に抵抗を追加する必要はありません。必要な入力タイプに応じてジャンパ設定を調整するだけです。図 4.3 に、入力を電圧モードまたは電流モードに設定するジャンパを示します。

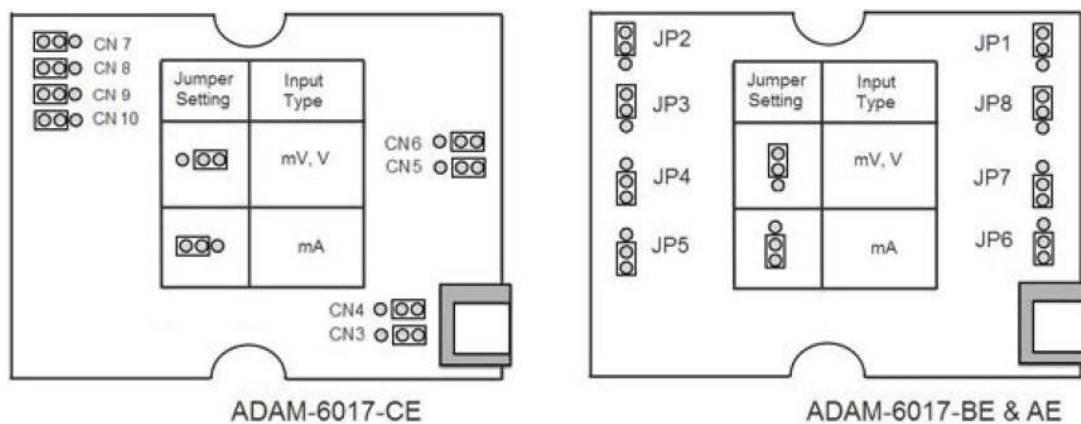


図 4.4 ADAM-6017 アナログ入力タイプ設定

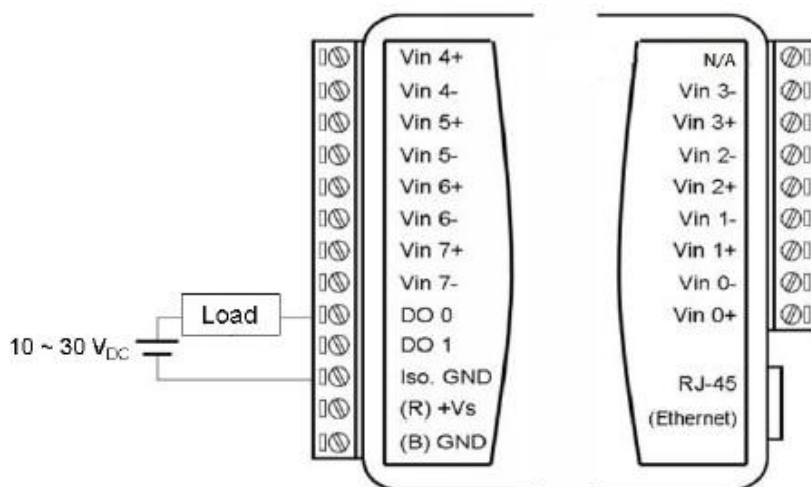


図 4.5 ADAM-6017 デジタル出力配線

### 4.3.3 アドレス割り当て

Modbus/TCP標準に基づき、システムに配置するADAM-6000 I/Oチャンネルのアドレスは、単純なルールで定義されます。I/O アドレスのマッピングについては、付録 B.2.2 を参照してください。

## 4.4 ADAM-6018 絶縁熱電対入力/8チャンネルデジタル出力モジュール

ADAM-6018 は 16 ビット、8 チャンネルの熱電対入力モジュールで、全チャンネルでプログラムブル入力レンジを備えています。このモジュールには、8つの熱電対入力（タイプJ、K、T、E、R、S、B）と8つのデジタル出力があります。各入力は独立したレンジに設定できるため、各チャンネルを異なるアプリケーションで同時に使用できます。



図 4.6 ADAM-6018 8 チャンネル熱電対入力

#### 4.4.1 仕様

- 通信10/100BASE-Tイーサネット
- 対応プロトコルModbus/TCP、TCP/IP、UDP、HTTP、ICMP、DHCP、ARP
- P2PとGCLをサポート（セクション6.7と第8章を参照）

##### アナログ入力

- チャンネル8（差動）
- 入力インピーダンス： $>10\text{ M}\Omega$
- 入力タイプ熱電対
- 熱電対の種類と範囲：
  - タイプJ： $0\sim760^{\circ}\text{C}$
  - タイプK： $0\sim1370^{\circ}\text{C}$
  - タイプT： $-100\sim400^{\circ}\text{C}$
  - タイプE： $0\sim1000^{\circ}\text{C}$
  - タイプR： $500\sim1750^{\circ}\text{C}$
  - タイプS： $500\sim1750^{\circ}\text{C}$
  - タイプB： $500\sim1800^{\circ}\text{C}$
- 精度： $\pm 0.1\%$ 以上
- スパンドリフト： $\pm 25\text{ ppm}/^{\circ}\text{C}$
- ゼロ・ドリフト： $\pm 6\text{ mV}/^{\circ}\text{C}$
- 解像度：16ビット
- サンプルレート10 Hz
- CMR @ 50/60 Hz：90 dB
- NMR @ 50/60 Hz：60 dB
- 過電圧保護  $\pm 35\text{ V}_{\text{DC}}$
- TVS/ESD保護回路内蔵
- ワイヤー焼損検出

##### デジタル出力

- チャンネル8
- シンクタイプ：オープンコレクタ $\sim 30\text{V}$ 、100mA（最大負荷）
- 消費電力：各モジュール300mW

##### 一般

- ウォッチドッグ・タイマー内蔵
- 絶縁保護： $2000\text{ V}_{\text{DC}}$
- 電源入力  $10\sim 30\text{ V}_{(\text{DC})}$
- 消費電力： $2\text{ W @ } 24\text{ V}_{\text{DC}}$
- 電源反転保護
- 作動湿度： $20\sim 95\% \text{ RH}$ （不凝縮）
- 保管湿度： $0\sim 95\% \text{ RH}$ （不凝縮）
- 実用温度： $-10\sim 70^{\circ}\text{C}$
- 保管温度： $-20\sim 80^{\circ}\text{C}$

## 4.4.2 アプリケーション

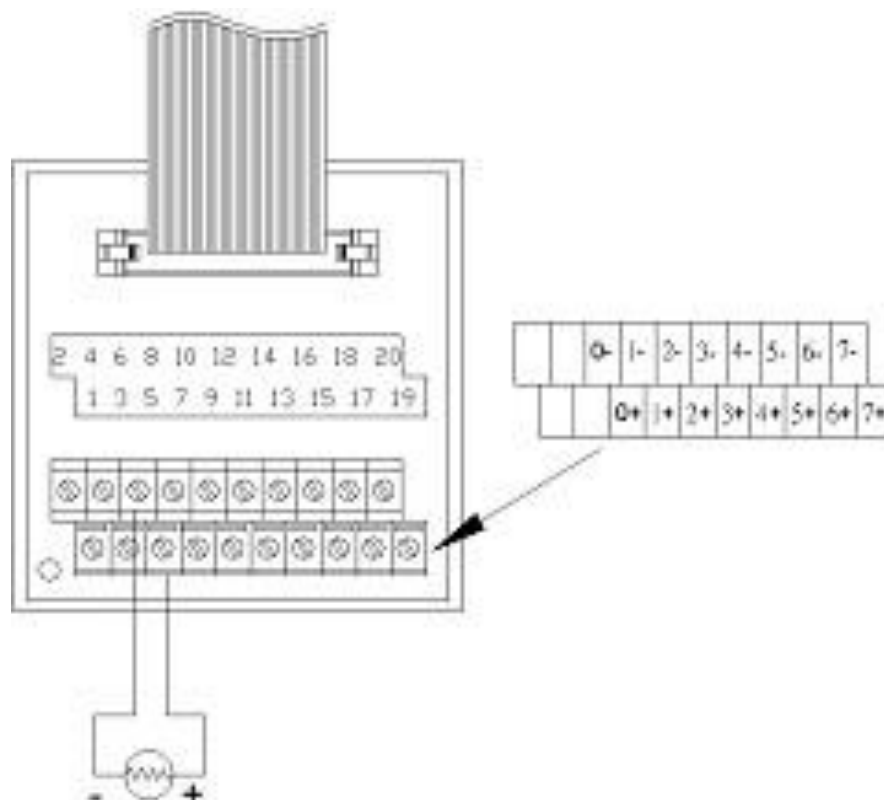


図 4.7 ADAM-6018 熱電対入力配線

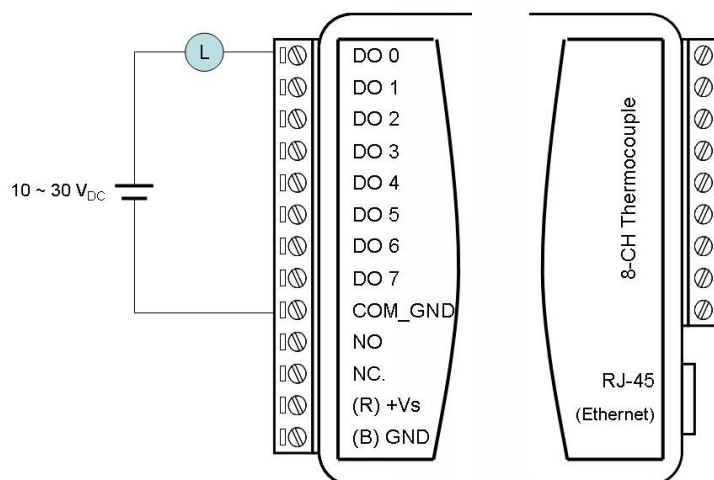


図 4.8 ADAM-6018 デジタル出力配線

## 4.4.3 アドレス割り当て

Modbus/TCP標準に基づき、システムに配置するADAM-6000 I/Oチャンネルのアドレスは簡単なルールで定義されます。I/O アドレスのマッピングについては、付録 B.2.3 を参照してください。

## 4.5 ADAM-6018+ 8チャンネル絶縁熱電対入力モジュール

ADAM-6018+ は、16 ビット、8 チャンネルの熱電対入力モジュールで、すべてのチャンネルで入力レンジをプログラマブル に設定できます。このモジュールは、8つの熱電対入力（タイプJ、K、

T、E、R、S、B）と8つのデジタル出力があります。各入力は独立したレンジに設定できるため、個々のチャンネルを異なるアプリケーションで同時に使用することができます。

### 4.5.1 仕様

- 通信10/100BASE-Tイーサネット
- 対応プロトコルModbus/TCP、TCP/IP、UDP、HTTP、ICMP、DHCP、ARP、MQTT、SNMP
- P2PとGCLをサポート（セクション6.7と第8章を参照）

#### 熱電対入力

- チャンネル8（差動）
- 入力タイプ熱電対
- 熱電対の種類と範囲：
  - タイプ J: 0~760° C
  - タイプ K: 0~1370° C
  - タイプ T: -100~400° C
  - タイプ E: 0~1000° C
  - タイプ R: 500~1750° C
  - タイプ S: 500~1750° C
  - タイプ B: 500~1800° C
- 精度@25°C：（図3. ように垂直方向に取り付ける）
  - タイプ J, K, E, R, S：±0.1%FSR Max
  - タイプ B：±0.15% FSRマックス
  - タイプ T：±0.2%FSRマックス
- スパンドリフト：±25 ppm/° C
- ゼロ・ドリフト：±6 mV/° C
- 解像度：16ビット
- サンプルレート10 Hz
- 過電圧保護 ±35 VDC
- TVS/ESD保護回路内蔵
- ワイヤー焼損検出

#### 一般

- ウォッチドッグ・タイマー内蔵
- 絶縁保護：DC2000V
- 電源入力 10~30 VDC
- 消費電力1 W @ 24 VDC
- 電源反転保護
- 作動湿度：20~95% RH（不凝縮）
- 保管湿度：0~95% RH（不凝縮）
- 作動温度：-40~70° C
- 保管温度：-40~85° C

## 4.5.2 アプリケーション

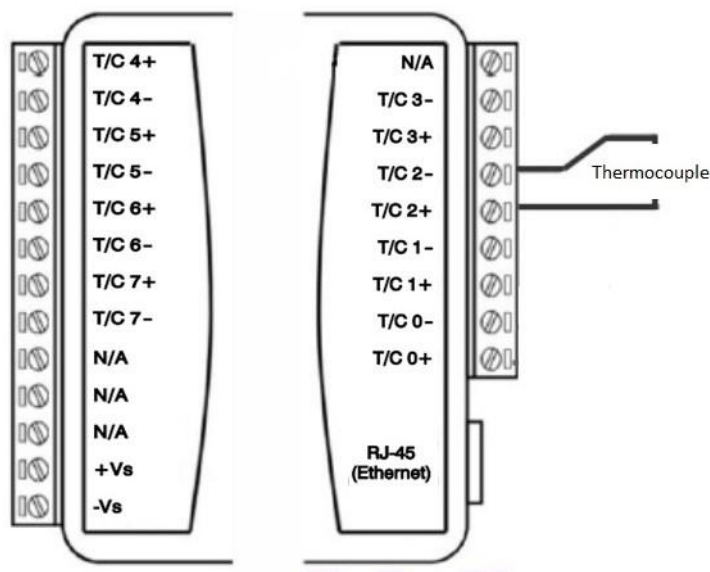


図 4.9 ADAM-6018+ 熱電対の配線

## 4.6 ADAM-6024 12チャンネル絶縁ユニバーサルI/Oモジュール

ADAM-6024は、全チャンネルにプログラマブル入力レンジを備えた12チャンネル・ユニバーサルI/Oモジュールです。このモジュールには、6つのアナログ入力、2つのアナログ出力、2つのデジタル入力、2つのデジタル出力があります。アナログ入力チャンネルは16ビットユニバーサル信号入力で、電圧(±10 V)と電流(0~20、4~20 mA)信号を受け付けます。アナログ出力チャンネルは、電圧(0~10 V)と電流(0~20 mA、4~20 mA)の12ビット出力です。各アナログ・チャンネルは独立したレンジに設定でき、異なるアプリケーションで個々のチャンネルを同時に使用できます。

### 4.6.1 仕様

- 通信10/100BASE-Tイーサネット
- 対応プロトコルModbus/TCP、TCP/IP、UDP、HTTP、ICMP、DHCP、ARP
- P2PおよびGCL機能をサポートする他のモジュールからデータを受信し、アナログ出力信号を生成する(P2PおよびGCLの詳細については、セクション6.7および第8章を参照)。

#### アナログ入力

- チャンネル6 (ディファレンシャル)
- レンジ: ±10V<sub>(DC)</sub>、0~20mA、4~20mA
- 入力インピーダンス: 20 MΩ
- 精度: ±0.1% of FSR at 25° C
- 解像度: 16ビット
- CMR @ 50/60 Hz: 90 dB
- NMR @ 50/60 Hz: 60 dB
- スパンドリフト: ±25 ppm/° C
- ゼロ・ドリフト: ±6 uV/° C
- 絶縁保護: 2000 V<sub>DC</sub>

### アナログ出力

- チャンネル: 2
- レンジ0~10 V<sub>(DC)</sub>、0~20 mA、4~20 mA
- 精度: ±0.1% of FSR at 25° C
- 解像度: 12ビット
- 電流負荷抵抗: 500 Ω 最大)
- 電圧負荷抵抗: 1 kΩ 最小)
- 絶縁保護: 2000 V<sub>DC</sub>
- ドリフト: ±50 ppm/° C

### デジタル入力

- チャンネル: 2
- ドライコンタクト:
  - 論理レベル 0: GND に近い
  - 論理レベル1: オープン
- ウェットコンタクト:
  - 論理レベル 0: 0~3 V<sub>DC</sub>
  - ロジック・レベル 1: 10~30 V<sub>DC</sub>

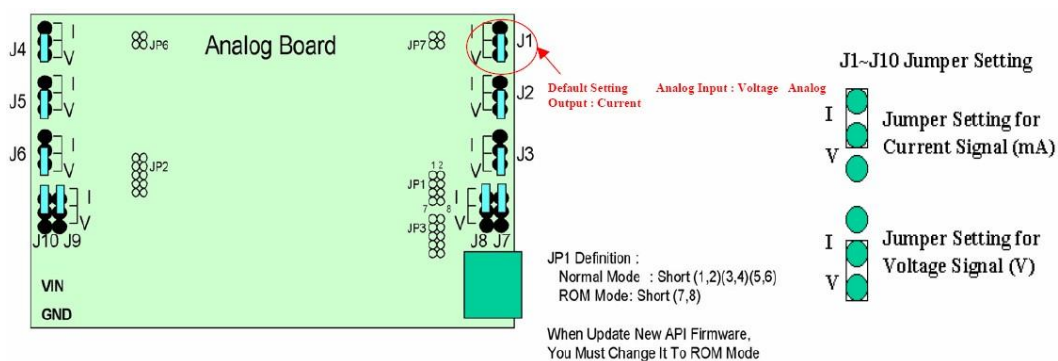
### デジタル出力

- チャンネル: 2
- シンクタイプオープンコレクタ~30V、100mA (最大)
- 消費電力: 各モジュール300mW

### 一般

- ウォッチドッグ・タイマー内蔵
- 絶縁保護: 2000 V<sub>DC</sub>
- 電源入力 10~30 V<sub>(DC)</sub>
- 消費電力4W@24 V<sub>DC</sub>
- 電源反転保護
- 作動湿度: 20~95% RH (不凝縮)
- 保管湿度: 0~95% RH (不凝縮)
- 作動温度: -40~70° C
- 保管温度: -40~85° C

### ジャンパ設定 (ADAM-6024-A1E)



Channel	Jumper	Current	Voltage
AI0	J1	I	V
AI1	J2	I	V
AI2	J3	I	V
AI3	J4	I	V
AI4	J5	I	V
AI5	J6	I	V
AO0	J7	I	V
	J8	I	V
AO1	J9	I	V
	J10	I	V

図 4.10 ADAM-6024-A1E ジャンパ設定

#### ジャンパ設定 (ADAM-6024-D)

ジャンパーは、AIチャンネルのタイプを電流モードと電圧モードの間で変更するためのものです。

Channel	Jumper	Current Mode	Voltage Mode (Default)
AI0	JAI1	1-2 Short	2-3 Short
AI1	JAI2	1-2 Short	2-3 Short
AI2	JAI3	1-2 Short	2-3 Short
AI3	JAI4	1-2 Short	2-3 Short
AI4	JAI5	1-2 Short	2-3 Short
AI5	JAI6	1-2 Short	2-3 Short

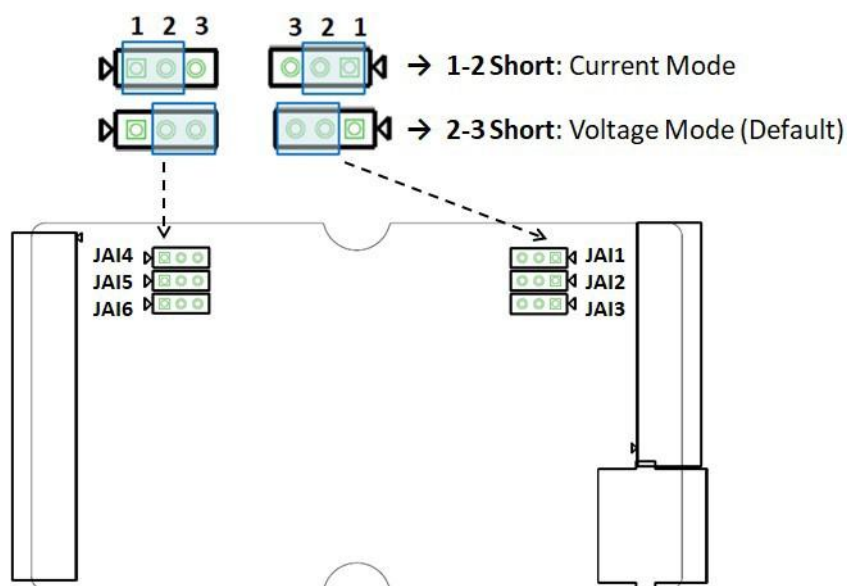


図 4.11 ADAM-6024-D ジャンパー設定

**注意** ADAM-6024-Dのアナログ出力はジャンプ設定が不要です。



## 4.6.2 アプリケーション配線

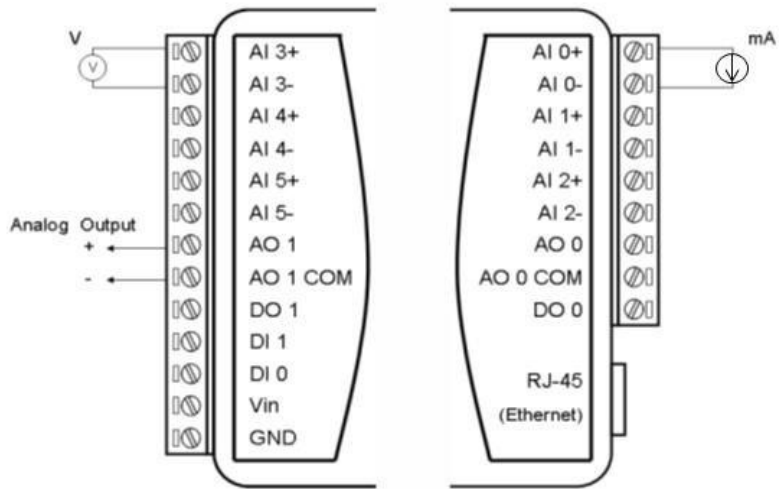


図 4.12 ADAM-6024 アナログ I/O 配線

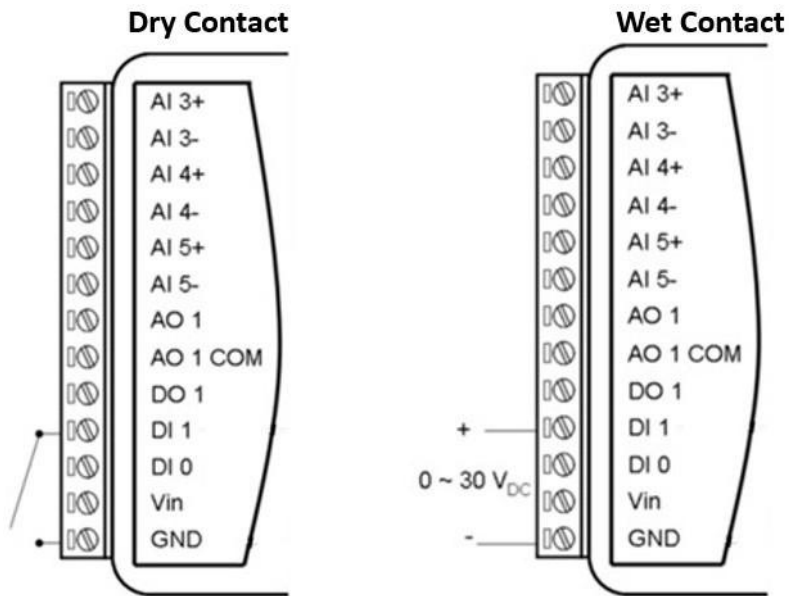


図 4.13 ADAM-6024 デジタル入力配線

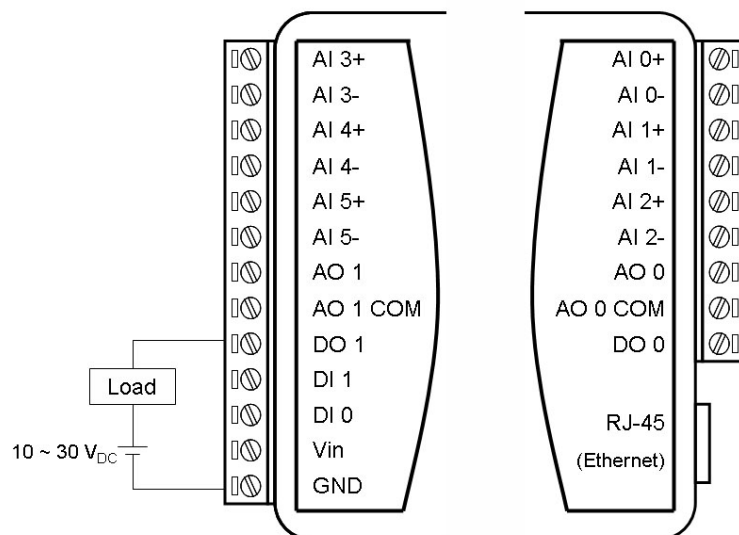


図 4.14 ADAM-6024 デジタル出力配線

### 4.6.3 アドレス割り当て

Modbus/TCP標準に基づき、システムに配置するADAM-6000 I/Oチャンネルのアドレスは簡単なルールで定義されます。I/O アドレスのマッピングについては、付録 B.2.4 を参照してください。

# 5

デジタルADAM-6000 I/Oモジュール  
の紹介

## 5.1 デジタルI/Oおよびリレーモジュール

デジタルI/Oモジュールは、デジタルセンサやアクチュエーターに接続できます。これらのモジュールは、ドライ接点とウェット接点の両方をサポートし、さまざまなアプリケーションに対応します。一方、リレーは電氣的に動作するスイッチです。リレーモジュールは通常、低電力信号を使用して回路を制御するために使用されます。ホストコンピュータからプロンプトが表示されると、標準の10/100BASE-TイーサネットまたはIEEE 802.11b WLANを介してデータが送信されます。Modbus/TCPプロトコルをサポートするWebページまたはHMIソフトウェアを介して、デジタルI/Oステータスの読み取り/設定が可能です。

## 5.2 ADAM-6050 18チャンネル絶縁デジタルI/Oモジュール

ADAM-6050は、シームレスなイーサネット接続のための10/100BASE-Tインターフェースを内蔵した高密度I/Oモジュールです。2000V<sub>DC</sub>絶縁保護付きの12デジタル入力と6デジタル出力を備えています。すべての入力は、重要な信号処理のためのラッチ機能を持ち、3kHzカウンタおよび周波数入力チャンネルとして使用できます。出力はパルス出力に対応しています。

### 5.2.1 仕様

- 通信10/100BASE-Tイーサネット
- 対応プロトコルMQTT、SNMP、Modbus/TCP、TCP/IP、UDP、HTTP、ICMP、DHCP、ARP
- P2PとGCLをサポート（セクション6.7と第8章を参照）

#### デジタル入力

- チャンネル12
- ドライコンタクト：
  - 論理レベル 0：GND に近い
  - 論理レベル 1：オープン
- ウェットコンタクト：
  - 論理レベル 0：0~3 V<sub>DC</sub>
  - ロジック・レベル 1：10~30 V<sub>DC</sub>
- 3kHzカウンタ入力対応（32ビット、オーバーフローフラグ付き）
- 周波数入力範囲0.2~3 kHz
- デジタル入カステータスの反転をサポート

#### デジタル出力

- チャンネル6
- シンクタイプ：オープンコレクタ~30V、100mA（最大負荷）
- 5kHzパルス出力対応
- ハイ・ツー・ローおよびロー・ツー・ハイのディレイ出力に対応
- 漏れ電流：200 μA（最大）（Dバージョン）
- 250 uA（最大）（D1バージョン）

一般

- ウォッチドッグ・タイマー内蔵
- 絶縁保護：2000 V<sub>DC</sub>
- 電源入力 10~30 V<sub>DC</sub>
- 消費電力：2W（最大）@24V<sub>(DC)</sub>
- 電源反転保護
- 作動湿度：20~95% RH（不凝縮）
- 保管湿度：0~95% RH（不凝縮）
- 動作温度：-20~70°C（Dバージョン：-40~70）
- 保管温度：-30~80°C（Dバージョン：-40~85）

5.2.2 アプリケーション配線

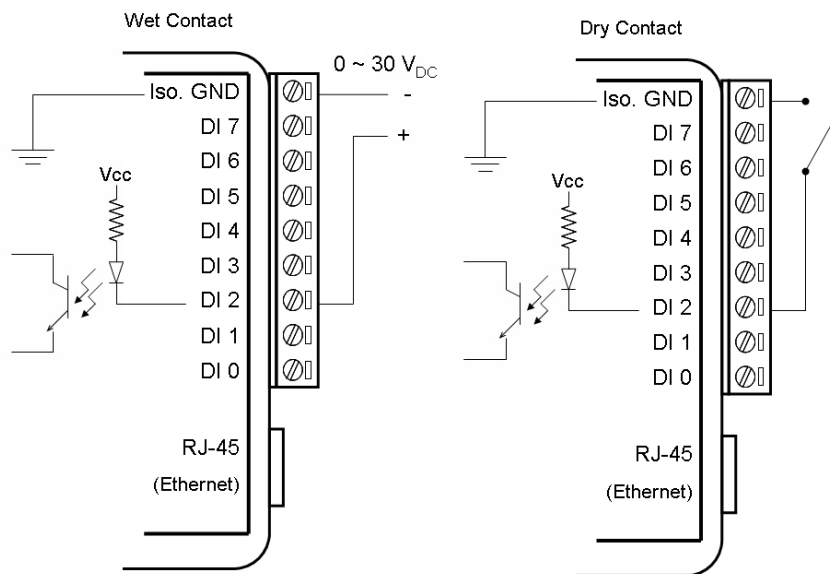


図 5.1 ADAM-6050 デジタル入力配線

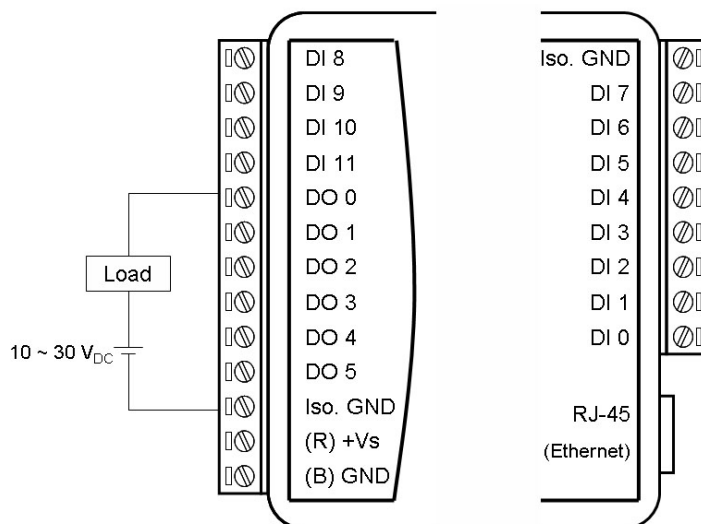


図 5.2 ADAM-6050 デジタル出力配線

### 5.2.3 アドレス割り当て

Modbus/TCP標準に基づき、システムに配置するADAM-6000 I/Oチャンネルのアドレスは簡単なルールで定義されます。I/O アドレスのマッピングについては、付録 B.2.5 を参照してください。ADAM-6050 のすべての入力、Windows Utility (セクション 6.3 を参照) を使用して 32 ビット・カウンタ (各カウンタはロー・ワードとハイ・ワードの 2 つのアドレスを持つ) として使用するよう設定することができます。

## 5.3 ADAM-6051 2チャンネルカウンタ付き14チャンネル絶縁デジタル I/Oモジュール

ADAM-6051は、シームレスなイーサネット接続のための10/100BASE-Tインターフェースを内蔵した高密度デジタルI/Oモジュールです。2000V<sub>DC</sub>絶縁保護付きの12デジタル入力、2カウンター・チャンネル、2デジタル出力を備えています。すべてのデジタル入力は、重要な信号処理のためのラッチ機能を持ち、3kHzカウンタおよび周波数入力チャンネルとして使用できます。デジタル出力はパルス出力に対応しています。

### 5.3.1 仕様

- 通信10/100BASE-Tイーサネット
- 対応プロトコルMQTT、SNMP、Modbus/TCP、TCP/IP、UDP、HTTP、ICMP、DHCP、ARP
- P2PとGCLをサポート (セクション6.7と第8章を参照)

#### デジタル入力

- チャンネル12
- ドライコンタクト：
  - 論理レベル 0: GND に近い
  - 論理レベル 1: オープン
- ウェットコンタクト：
  - 論理レベル 0: 0~3 V<sub>DC</sub>
  - ロジック・レベル 1: 10~30 V<sub>DC</sub>
- 3kHzカウンタ入力対応 (32ビット、オーバーフローフラグ付き)
- 3kHzの周波数入力に対応
- デジタル入カステータスの反転をサポート

#### カウンター入力

- チャンネル: 2 (32ビット、オーバーフローフラグ付き)
- 最大カウント4, 294, 967, 295
- 周波数範囲：
  - 0.2~4500 Hz (周波数モード)
  - 0~4500 Hz (カウンター・モード)
- ウェットコンタクト：
  - 論理レベル 0: 0~3 V<sub>DC</sub>
  - ロジック・レベル 1: 10~30 V<sub>DC</sub>

#### デジタル出力

- チャンネル: 2
- シンクタイプオープンコレクタ~30V、100mA (最大負荷)
- 5kHzパルス出力対応

- ハイ・ツー・ローおよびロー・ツー・ハイのディレイ出力に対応
- 漏れ電流: 200  $\mu$ A (Dバージョン)

#### 一般

- ウォッチドッグ・タイマー内蔵
- 絶縁保護: 2000 V<sub>DC</sub>
- 電源入力 10~30 V<sub>DC</sub>
- 消費電力3 W @ 24 V<sub>DC</sub>
- 電源反転保護
- 作動湿度: 20~95% RH (不凝縮)
- 保管湿度: 0~95% RH (不凝縮)
- 動作温度: -20~70°C (Dバージョン: -40~70)
- 保管温度: -30~80°C (Dバージョン: -40~85)

### 5.3.2 アプリケーション配線

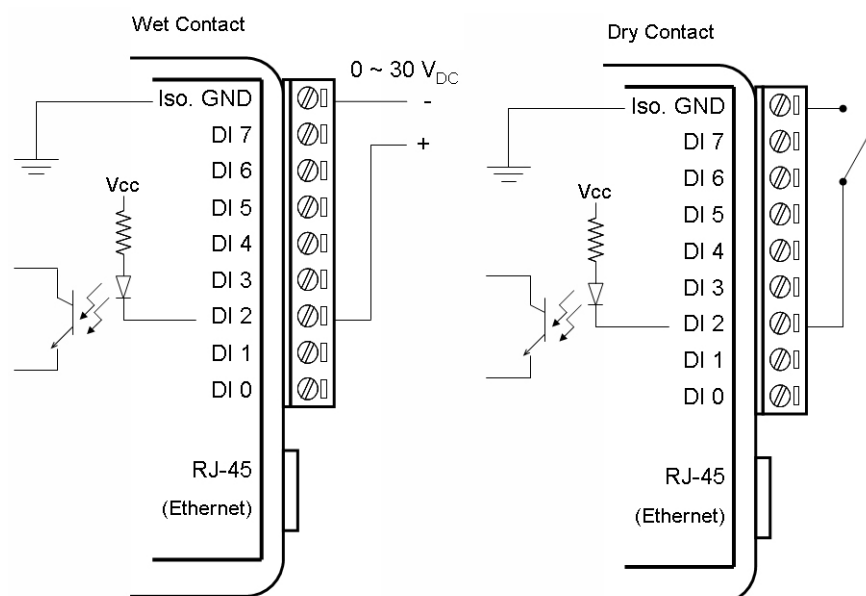


図 5.3 ADAM-6051 デジタル入力配線

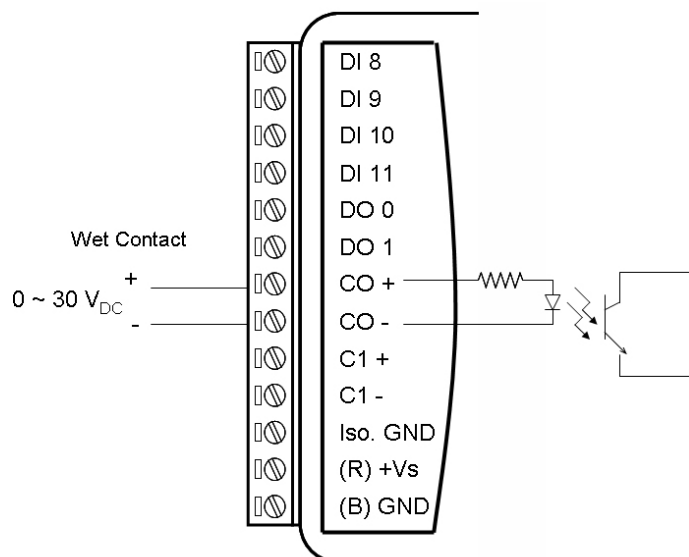


図 5.4 ADAM-6051 カウンター（周波数）入力

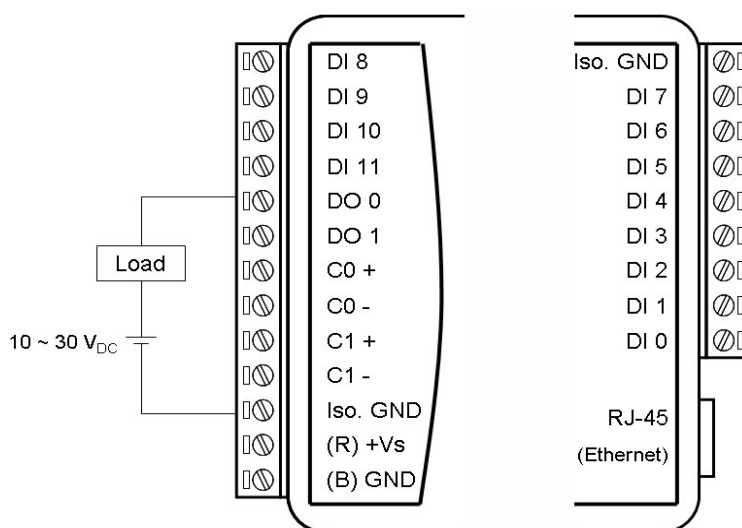


図 5.5 ADAM-6051 デジタル出力配線

### 5.3.3 アドレス割り当て

Modbus/TCP標準に基づき、システムに配置するADAM-6000モジュールI/Oチャンネルのアドレスは簡単なルールで定義されています。I/O アドレスのマッピングについては、付録 B.2.6 を参照してください。ADAM-6051 のすべてのデジタル入力は、Windows Utility（セクション 6.3 を参照）を使用して 32 ビット・カウンタ（各カウンタはロー・ワードとハイ・ワードの 2 つのアドレスを持つ）として使用するように設定することができます。

## 5.4 ADAM-6052 16チャンネル・ソース型絶縁デジタルI/Oモジュール

ADAM-6052は、シームレスなイーサネット接続のための10/100BASE-Tインターフェースを内蔵した高密度デジタルI/Oモジュールです。このモジュールは、8つのデジタル入力と8つのデジタル出力を備えています。すべての入力はラッチ機能を持ち、3kHzカウンタおよび周波数入力チャンネルとして使用できます。出力はソースタイプとパルス出力に対応しています。

## 5.4.1 仕様

- 通信10/100BASE-Tイーサネット
- 対応プロトコルMQTT、SNMP、Modbus/TCP、TCP/IP、UDP、HTTP、ICMP、DHCP、ARP
- P2PとGCLをサポート（セクション6.7と第8章を参照）

### デジタル入力

- チャンネル8
- ドライコンタクト：
  - 論理レベル 0: オープン
  - 論理レベル 1: 接地に近い
- ウェットコンタクト：
  - 論理レベル 0:  $0\sim 3 V_{DC}$
  - ロジック・レベル 1:  $10\sim 30 V_{DC}$
- 3kHzカウンタ入力対応（32ビット、オーバーフローフラグ付き）
- 3kHzの周波数入力に対応
- デジタル入カステータスの反転をサポート

### デジタル出力

- チャンネル8
- ソースタイプ $10\sim 35V_{DC}$ 、1A（チャンネルあたり）
- 注：70°Cで動作する場合、D00～D03およびD04～D07の最大合計電流は3A未満であることが推奨されます。
- 5kHzパルス出力対応
- ハイ・ツー・ローおよびロー・ツー・ハイのディレイ出力に対応

### 一般

- ウォッチドッグ・タイマー内蔵
- 絶縁保護： $2000 V_{DC}$
- 電源入力  $10\sim 30 V_{(DC)}$
- 消費電力：2 W @ 24  $V_{DC}$
- 電源反転保護
- 作動湿度：20～95% RH（不凝縮）
- 保管湿度：0～95% RH（不凝縮）
- 動作温度：-20～70°C（Dバージョン：-40～70）
- 保管温度：-30～80°C（Dバージョン：-40～85）

## ジャンパ設定

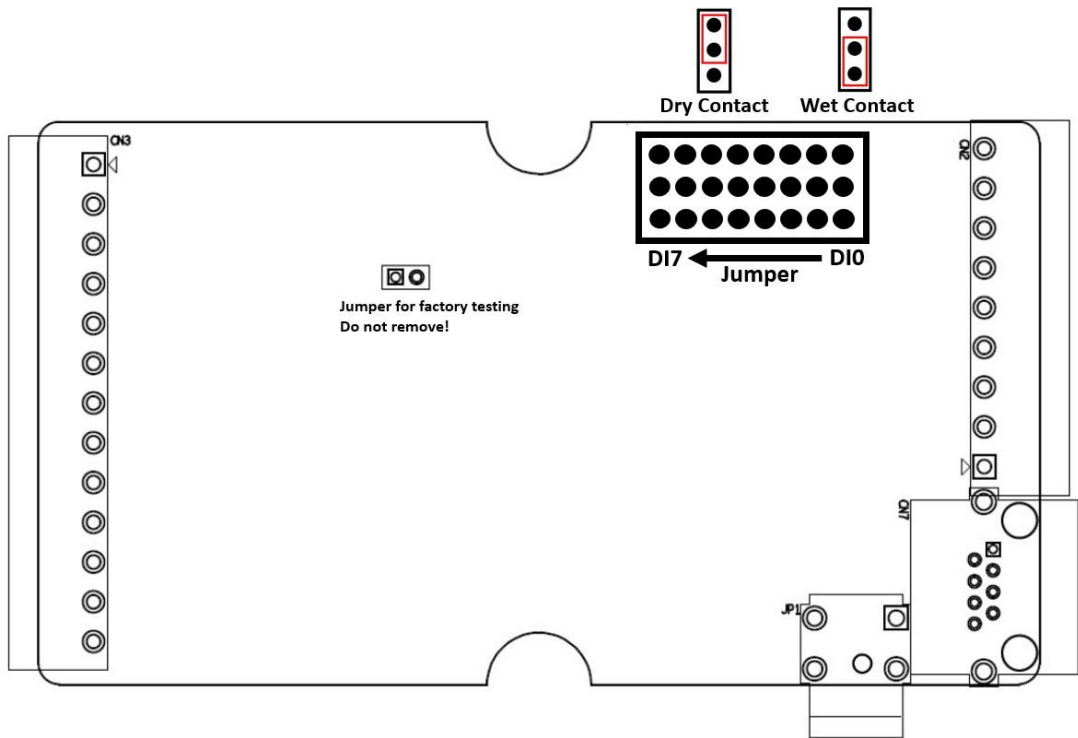


図 5.6 ADAM-6052 ジャンパ設定

## 5.4.2 アプリケーション配線

ADAM-6052は入力にドライ接点とウェット接点の両方をサポートしています。ジャンパを調整することで、ドライ接点モードとウェット接点モードを切り替えることができます。

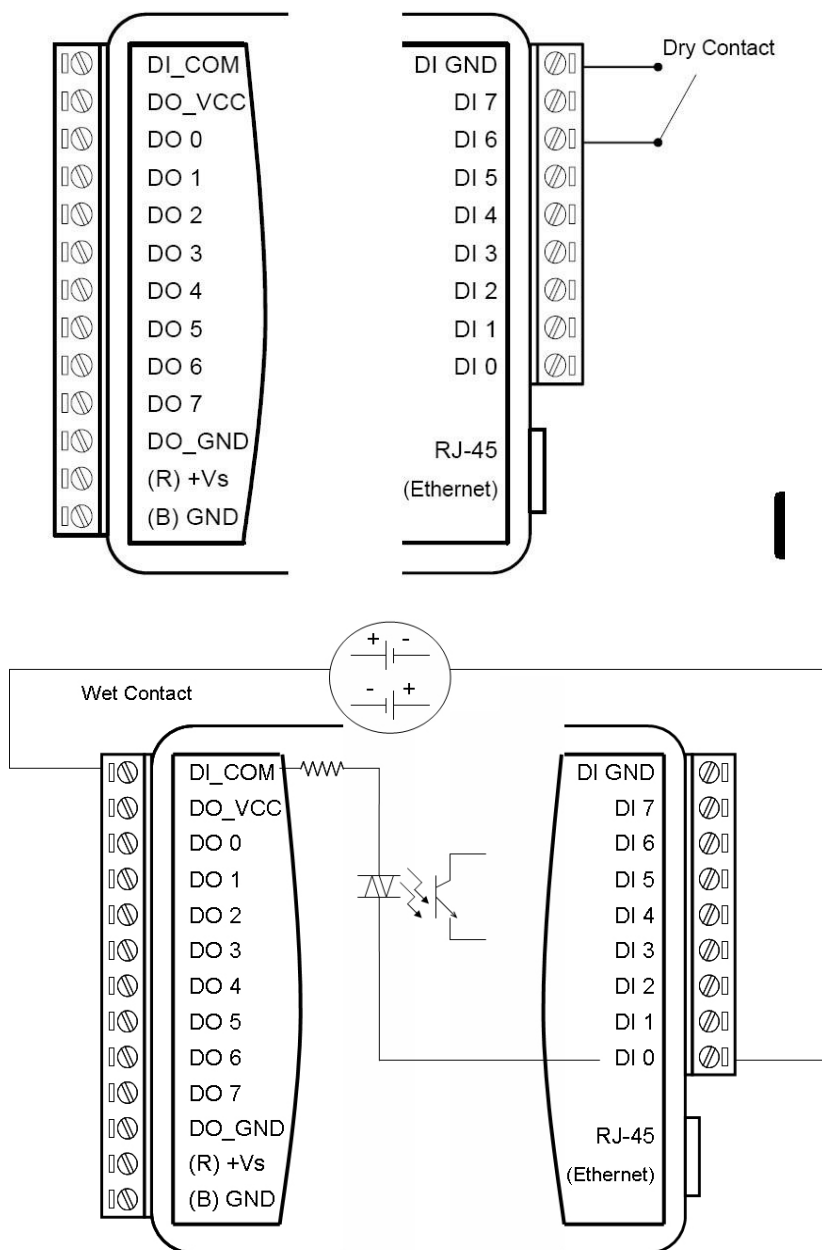


図 5.7 ADAM-6052 デジタル入力配線

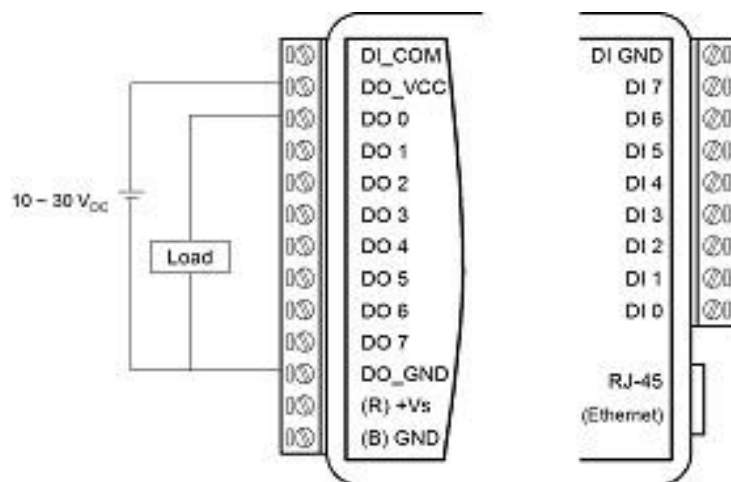


図 5.8 ADAM-6052 デジタル出力配線

### 5.4.3 アドレス割り当て

Modbus/TCP 標準に基づき、ADAM-6000 モジュールの I/O チャンネルのアドレスはシンプルなルールで定義されています。I/O アドレスのマッピングについては、付録 B.2.7 を参照してください。ADAM-6052 入力は、Adam/Apax .NET Utility (セクション 6.3 を参照) を使用して 32 ビット・カウンタ (各カウンタはロー・ワードとハイ・ワードの 2 つのアドレスを持つ) として設定することができます。

## 5.5 ADAM-6060 6チャンネルデジタル入力/6チャンネルリレーモジュール

ADAM-6060は、10/100BASE-Tインターフェースを備えた高密度I/Oモジュールです。イーサネット・ポートとウェブ・ページとの結合により、このモジュールは6つのデジタル入力と6つのリレー出力 (Form A) を提供し、接点定格は120 V<sub>AC</sub> @ 0.5 Aおよび30 V<sub>DC</sub> @ 1です。

A. すべての入力は、重要な信号処理のためのラッチを持っており、3kHzカウンタおよび周波数入力チャンネルとして使用できます。出力はパルス出力に対応しています。

### 5.5.1 仕様

- 通信10/100BASE-Tイーサネット
- 対応プロトコルMQTT、SNMP、Modbus/TCP、TCP/IP、UDP、HTTP、ICMP、DHCP、ARP
- P2PとGCLをサポート (セクション6.7と第8章を参照)

#### デジタル入力

- チャンネル6
- ドライコンタクト :
  - 論理レベル 0 : GND に近い
  - 論理レベル 1 : オープン
- ウェットコンタクト :
  - 論理レベル 0 : 0~3 V<sub>DC</sub>
  - ロジック・レベル 1 : 10~30 V<sub>DC</sub>
- 3kHzカウンタ入力対応 (32ビット、オーバーフローフラグ付き)
- 周波数入力範囲0.2 Hz~3 kHz
- デジタル入力状態の反転をサポート
- 電源オフ時のカウンタ値の保持/破棄

## リレー出力

- チャンネル6 (フォームA)
- 接触定格 (抵抗性) :
  - AC120V @ 0.5A
  - 30 V<sub>DC</sub> @ 1 A
- 絶縁破壊電圧: 500 V<sub>AC</sub> (50/60 Hz)
- リレーオン時間: 7ms
- リレーオフ時間3 ms
- 総スイッチング時間: 10ms
- 絶縁抵抗: 1 GΩ (最小) @ 500 (DC)
- 最大スイッチング速度: 20回/分 (定格負荷時)
- 電氣的耐久性
  - 12V/10mA時:  $5 \times 10^7$ 動作 (代表値)
  - 6V/100mA時:  $1 \times 10^7$ 動作 (代表値)
  - 60V/500mA時:  $5 \times 10^5$ 動作 (代表値)
  - 30V/1000mA時:  $1 \times 10^6$ 動作 (代表値)
  - 30V/2000mA時:  $2 \times 10^5$ 動作 (代表値)
- 機械的耐久性
  - $10^8$ オペレーション (代表値)
- パルス出力対応 (最大3Hz)

## 一般

- ウォッチドッグ・タイマー内蔵
- 絶縁保護: 2000 V<sub>DC</sub>
- 電源入力 10~30 V (DC)
- 消費電力3 W (最大) @ 24 V<sub>DC</sub>
- 電源反転保護
- 作動湿度: 20~95% RH (不凝縮)
- 保管湿度: 0~95% RH (不凝縮)
- 動作温度: -20~70°C (Dバージョン: -40~70)
- 保管温度: -30~80°C (Dバージョン: -40~85)

## 5.5.2 アプリケーション配線

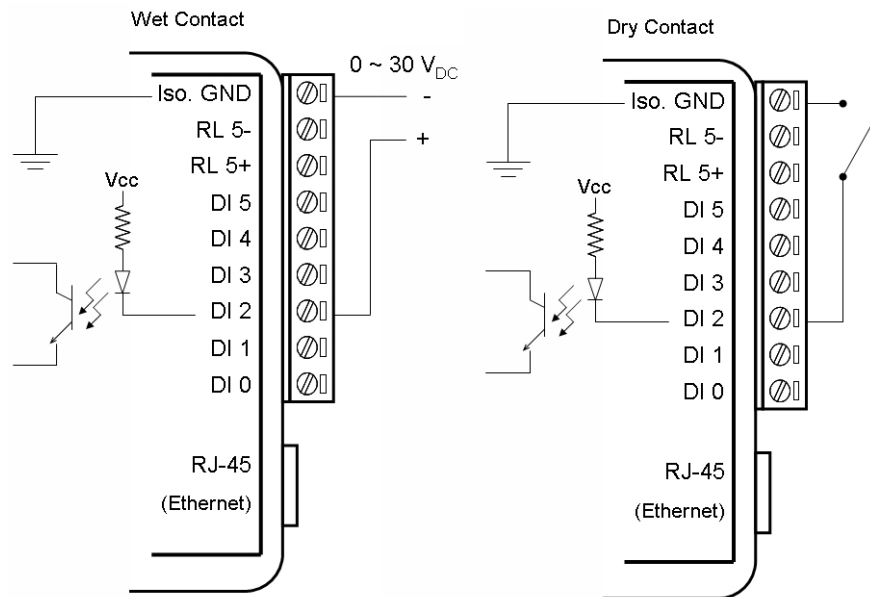


図 5.9 ADAM-6060 デジタル入力配線

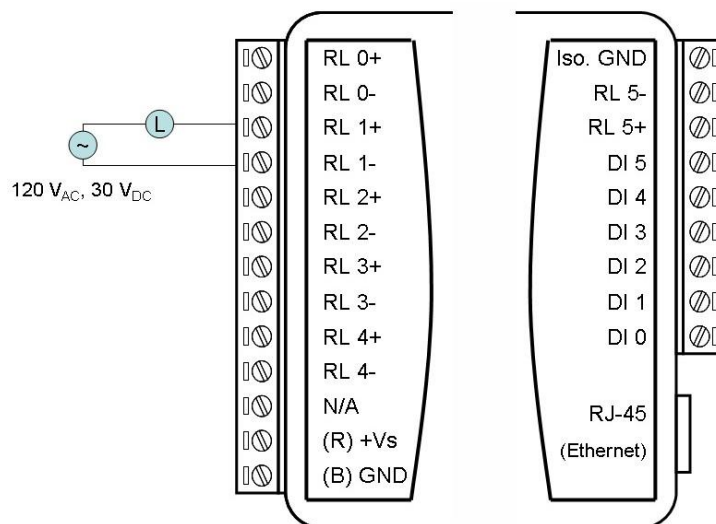


図 5.10 ADAM-6060 リレー出力配線

## 5.5.3 アドレス割り当て

Modbus/TCP標準に基づき、システムに配置するADAM-6000モジュールI/Oチャンネルのアドレスは、簡単なルールで定義されています。付録

B.2.8を参照してください。Windows Utility（セクション 6.3を参照）を使用して、ADAM-6060のすべての入力を32ビット・カウンタ（各カウンタはロー・ワードとハイ・ワードの2つのアドレスで構成される）として使用するように設定することができます。

## 5.6 ADAM-6066 6チャンネルデジタル入力/6チャンネルパワーリレーモジュール

ADAM-6066は、シームレスなイーサネット接続のための10/100BASE-Tインターフェースを備えた高密度I/Oモジュールです。6つのデジタル入力と6つの高電圧リレー出力（Form A）を備えています。モジュールの接点定格は $250\text{ V}_{AC}$  @5A、 $30\text{ V}_{DC}$  @3Aです。すべての入力は、重要な信号処理のためのラッチ機能を持ち、3kHzカウンターや周波数入力チャンネルとして使用できます。出力はパルス出力に対応しています。

### 5.6.1 仕様

- 通信10/100BASE-Tイーサネット
- サポートされているプロトコルMQTT、SNMP、Modbus/TCP、TCP/IP、UDP、HTTP、ICMP、DHCP、ARP Modbus/TCP、SNMP、TCP/IP、UDP、HTTP、ICMP、DHCP、ARP
- P2PとGCLをサポート（セクション6.7と第8章を参照）

#### デジタル入力

- チャンネル6
- ドライコンタクト：
  - 論理レベル 0：GND に近い
  - 論理レベル 1：オープン
- ウェットコンタクト：
  - 論理レベル 0： $0\sim 3\text{ V}_{DC}$
  - ロジック・レベル 1： $10\sim 30\text{ V}_{DC}$
- 3kHzカウンタ入力対応（32ビット、オーバーフローフラグ付き）
- 3kHzの周波数入力に対応
- デジタル入カステータスの反転をサポート

#### リレー出力

- チャンネル6（フォームA）
- 接触定格（抵抗性）：
  - AC250V @ 5A
  - $30\text{ V}_{DC}$  @ 3 A
- 絶縁破壊電圧： $500\text{ V}_{AC}$  (50/60 Hz)
- リレーオン時間：7 ms
- リレーオフ時間3 ms
- 総スイッチング時間：10ms
- 絶縁抵抗： $500\text{ V}_{DC}$ で $1\text{ G}\Omega$ （最小）
- 最大スイッチング速度：20回/分（定格負荷時）
- 電氣的耐久性： $1\times 10^5$ 動作
- 機械的耐久性
  - $2\times 10^7$ オペレーション（代表値）
  - （動作周波数180回/分の無負荷時）
- パルス出力対応（最大3Hz）

一般

- ウォッチドッグ・タイマー内蔵
- 絶縁保護 : 2000 V<sub>DC</sub>
- 電源入力 10~30 V<sub>(DC)</sub>
- 消費電力 : 2.5 W @ 24 V<sub>DC</sub>
- 電源反転保護
- 作動湿度 : 20~95% RH (不凝縮)
- 保管湿度 : 0~95% RH (不凝縮)
- 動作温度 : -20~70°C (Dバージョン : -40~70)
- 保管温度 : -30~80°C (Dバージョン : -40~85)

5.6.2 アプリケーション配線

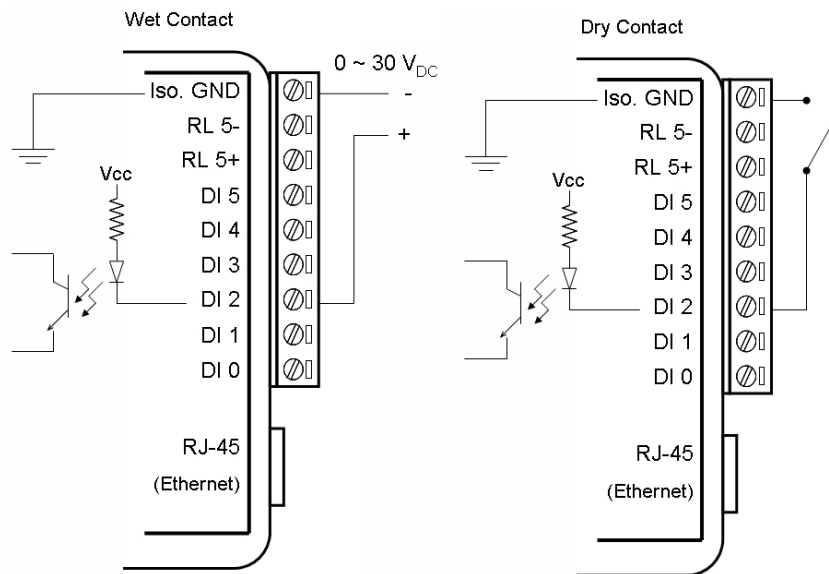


図 5.11 ADAM-6066 デジタル入力配線

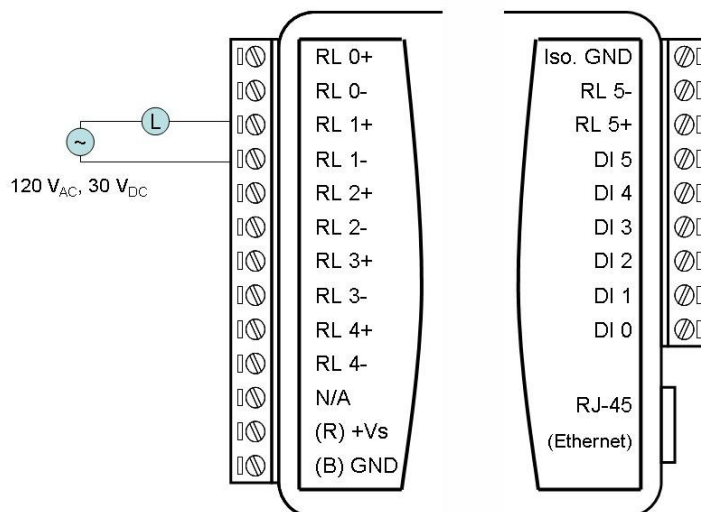


図 5.12 ADAM-6066 リレー出力配線

## 5.7 デジタル出力診断機能

デジタル出力がアクティブのとき、回路の断線やグラウンドへのショートは、出力が故障する原因となります。このような状況を迅速に解明するために、ADAM-6000 モジュール（すべての D バージョン）は、デジタル出力の異常を検出し、通知を発行することができるデジタル出力診断機能を備えています。診断ステータスは以下のグループに従って表示されます：

モジュール	診断グループ	出力チャンネル
ADAM-6017	グループ0	D00、D01
ADAM-6050	グループ0	D00、D01
	グループ1	D02、D03
	グループ2	D04、D05
ADAM-6050-D1	グループ0	D0 0
	グループ1	D0 1
	グループ2	D0 2
	グループ3	D0 3
	第4グループ	D0 4
	第5グループ	D0 5
	第6グループ	D0 6
ADAM-6051	グループ0	D00、D01
ADAM-6052	グループ0	D00
	グループ1	D01
	グループ2	D02
	グループ3	D03
	第4グループ	D04
	第5グループ	D05
	第6グループ	D06
	第7グループ	D07

ADAM-6050とADAM-6051では、各グループはデジタル出力のペアに対応し、ADAM-6052では、各グループは個々のチャンネルに対応することに注意してください。グループ内の片方または両方のチャンネルに異常が発生すると、そのグループの診断ステータスが変わります。考えられる異常の原因は以下の通りです。

### ADAM-6017、ADAM-6050、ADAM-6051用

デジタル出力がアクティブでないとき：

- デジタル出力回路断線発生（負荷オープン）
- デジタル出力接続がグラウンドにショートしている

- 出力が過電流 (>1 A) にさらされた

注：デジタル出力と診断機能を正常に動作させるため、各デジタル出力は各チャンネルの仕様内で設定する必要があります：30V、100mA（最大）。

### ADAM-6052用

デジタル出力がアクティブのとき：

- デジタル出力接続がグラウンドにショートしている
- 出力が過電流 (>1 A、typical) にさらされた

## 5.7.1 デジタル出力診断ステータスの取得方法

デジタル出力の診断ステータスは、Adam/Apax .NET Utility、Modbus アドレス、または ASCII コマンドを使用して取得できます。

### Adam/Apax .NET Utilityによるデジタル出力診断ステータスの取得

バージョン 2.05.10B08 以降、Adam/Apax .NET Utilityにはデジタル出力診断機能があります。図 5.13 の例では、デジタル出力診断ステータスはグループ 1 (D02~D03) とグループ 2 (D04~D05) に対して異常です。これは、これらの出力チャンネルのいずれかに問題が発生したことを示します。

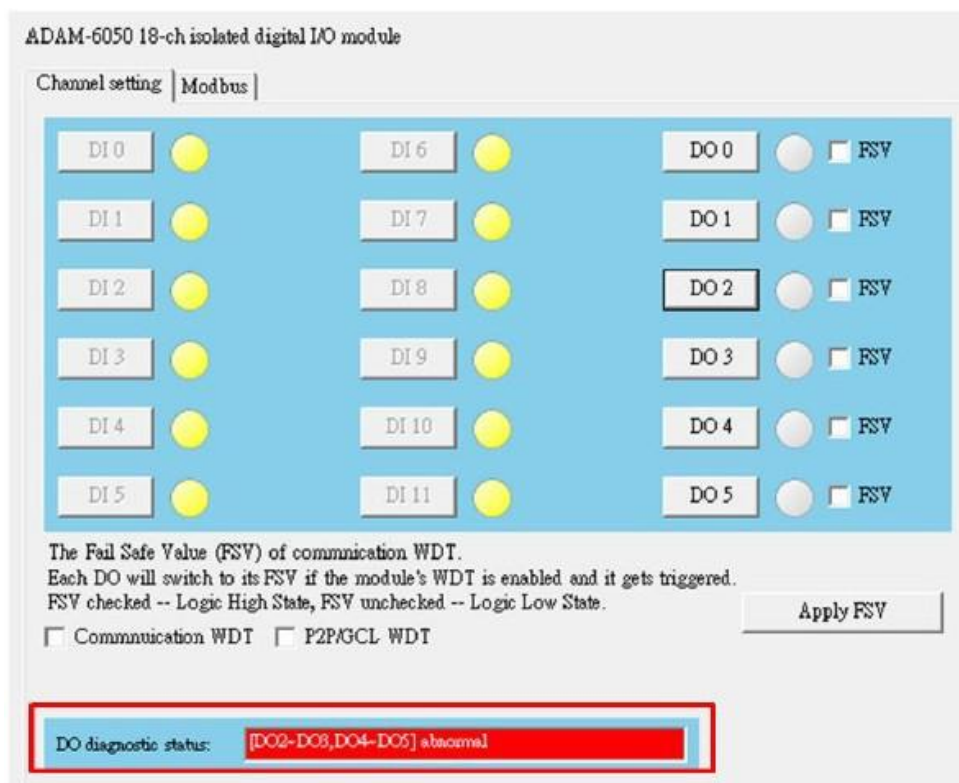


図 5.13 異常 DO 診断ステータス

図5.14では、デジタル出力の診断ステータスが“All normal”となっていますが、これはデジタル出力がアクティブになる前に、グループ内のすべてのチャンネルが正しく接続されている（断線やグラウンドへのショートがない）ことを意味します。

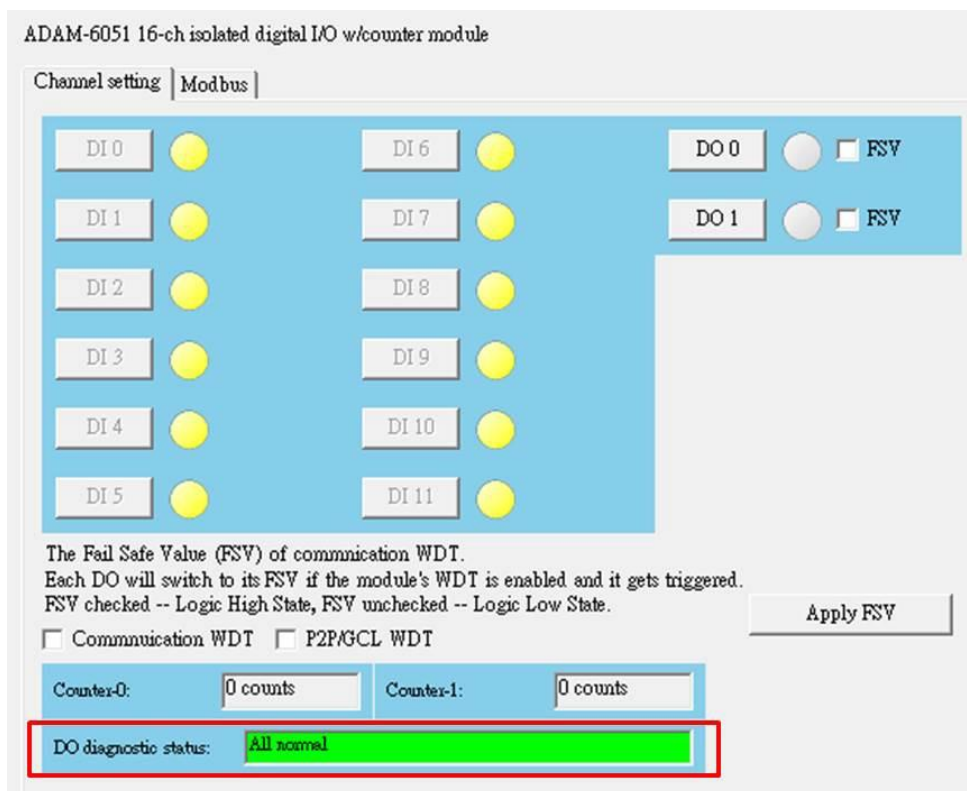


図 5.14 通常の DO 診断ステータス

Modbusアドレス値によるデジタル出力診断ステータスの取得

Address (4X)	Channels	Description	Attribute
40307	All	Digital output diagnostic status (for D version)	Read

次の表は、ADAM-6050、ADAM-6051、ADAM-6052 のグループに対するビット位置を示しています。従って、グループのステータスは各ビット示された値に従って解釈することができます。グループ・ステータスの値はバイナリ値として表示され、ビット 1 が一番右のビット位置、ビット 8 が一番左のビット位置です。

Bit Position for Modbus Address 40307	Relative Group for Interpreting the Digital Output Diagnostic Status Value		
	ADAM-6050	ADAM-6051	ADAM-6050
Bit 1	Group 0	Group 0	Group 0
Bit 2	Group 1	Reserved	Group 1
Bit 3	Group 2		Group 2
Bit 4	Reserved		Group 3
Bit 5			Group 4
Bit 6			Group 5
Bit 7			Group 6
Bit 8			Group 7

例 (ADAM-6050) : 図 5.13 に示した前述の例の場合、グループ・ステータス値は "xxxxx110" となります。ここで、ビット 1、2、3 はそれぞれグループ 0、1、2 のデジタル出力診断ステータスを示します。従って、グループ・ステータスは以下のように解釈できる :

- グループ0=0 (ノーマル)
- グループ1=1 (異常)
- グループ2=1 (異常)

#### ASCIIコマンドによるデジタル出力診断ステータスの取得

この例では、デジタル出力のステータスを要求するための ASCII コマンドと応答を示します。

<b>Syntax</b>	\$017	
<b>Response</b>	!01(Group#n)...(Group #1)(Group#0)(cr)	
<b>Example</b>	Command:	\$017
	Response:	!011110

ADAM-6050は診断ステータス用に3つのデジタル出力グループを持っているので、グループ0~2のステータスを示すビット位置は右から次のようになります :

- グループ0=0 (ノーマル)
- グループ1=1 (異常)
- グループ2=1 (異常)

ADAM-6050-D1では、デジタル出力チャンネルごとに診断ステータスがあります。

<b>Syntax</b>	\$017	
<b>Response</b>	!01(DO#0)...(DO #n)(DO#0)(cr)	
<b>Example</b>	Command:	<b>Example</b>
	Response:	!01101110

- D00 = 1 (異常)                      D01 = 0 (異常)
- D02 = 1 (異常)                      D03 = 1 (異常)
- D04 = 1 (異常)                      D05 = 0 (正常)

# 6

## システム構成ガイド

## 6.1 システム要件

### ホストコンピュータ

- Microsoft Windows XP/7
- 32 MB RAM
- 20MBのハードディスク空き容量
- VGAカラーモニター
- マウスまたはその他のポインティングデバイス
- 10/100-Mbpsイーサネットカード

### 通信インターフェース

- 10/100Mbpsイーサネット・ハブ（最低2ポート）
- イーサネットケーブル2本（RJ-45）
- クロスオーバーイーサネットケーブル（RJ-45）

## 6.2 Adam/Apax .NET Utilityのインストール

Adam/Apax .NET Utility は、ADAM モジュールの設定と操作のためにアドバンテックが提供するアプリケーションです。インストールファイルはADAMモジュールの付属CDに収録されています。また、[http:// www.advantech.com](http://www.advantech.com)（最新バージョンは「サービス&サポート」の「ダウンロードエリア」をクリック）から無料でダウンロードできます。インストールすると、ユーティリティのショートカットがデスクトップに表示されます。

注意：Adam/Apax .NET Utilityをインストールする前に、.NET Framework 2.0 以降をインストールする必要があります。

## 6.3 Adam/Apax .NET Utilityの概要

Adam/Apax .NET Utility は、ADAM モジュールの設定と操作のためのグラフィカルなインターフェイスです。また、リモート DA&C システムのテストや監視にも便利なツールです。以下の説明では、ユーティリティの使用方法を説明します。

Adam/Apax .NET Utility を起動するには、デスクトップ上のショートカットをダブルクリックするか、スタートメニューフォルダ内のアイコンをクリックします。プログラムを最初に開くと、図 6.1 のようなメインウィンドウが表示されます。

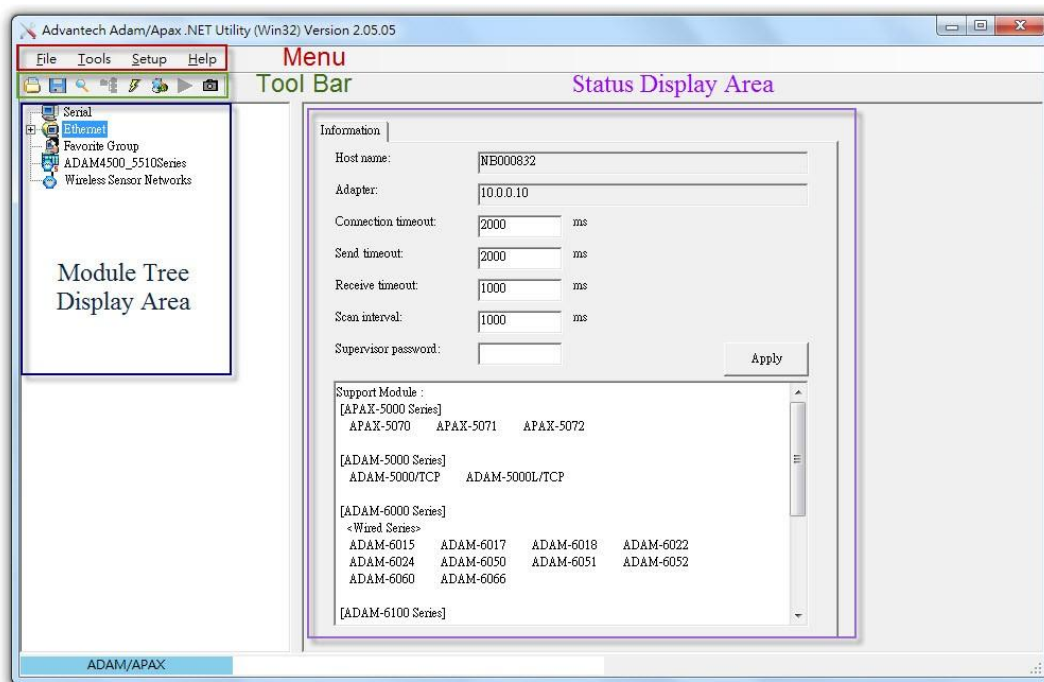


図 6.1 Adam/Apax .NET Utility操作ウィンドウ

図に示すように、このウィンドウには4つの主要エリアがある：1) メニューバー、2) ツールバー、3) Module Tree表示エリア、4) ステータス表示エリアです。

### 6.3.1 メニューバー

メニューバーには4つのメニューがある：File（ファイル）、Tools（ツール）、Setup（セットアップ）、Help（ヘルプ）。各メニューの項目は以下の通りです：

#### ファイルメニュー

Open Favorite Group	保存したお気に入りグループの設定ファイルをロードする。
Save Favorite Group	お気に入りグループを設定ファイルに保存します。
Auto-Initial Group	このオプションをチェックすると、次回 Adam/Apax .NET Utility を起動したときに、同じお気に入りグループの設定が読み込まれます。
Exit	Adam/Apax .NET Utilityを終了する。

#### ツールメニュー

Search Device	ホスト PC に接続されているすべての ADAM モジュールを検索します（セクション 6.3.5 を参照）。
Add Devices to Group	Module Tree表示領域で選択されたデバイスのみがグループに追加されます。この項目は、単一モジュールまたは複数モジュールのファームウェア、コンフィギュレーション、HTML ファイルを更新するためのものです。設定ファイルには、デバイス情報、一般情報、P2P とストリーミング、GCL、Modbus アドレス XML ファイルの設定が含まれます。コンフィギュレーション・ファイルは、ステータス表示エリアの <b>Firmware</b> タブから Cfg ファイルとしてエクスポートできません。
Terminal for Command Testing	ASCII コマンドと Modbus/TCP（詳細はセクション 7.3 と 7.4 を参照）を介して ADAM モジュールと通信するためのターミナルを起動します。

Print Screen	Adam/Apax .NET Utility 画面をイメージファイルとしてエクスポート
Monitor Stream/Event Data	ADAM モジュールはデータストリーム機能をサポートしています。これにより、以下のことが可能になります。 ホスト（PCなど）をIPで定義すると、ADAMモジュールは次のようになる。は定期的にI/Oステータスをホストに送信する。IP アドレスと送信周期は、ステータス表示エリアの [Stream] タブで設定できる。Streamタブについては6.3.5節で説明します。
Monitor Peer to Peer	P2P（イベントトリガー）機能を有効にしたADAMモジュールからのメッセージを受信するには、このオプションを選択します。
Monitor GCL IO Data Message	GCL機能を有効にしたADAMモジュールからI/Oデータ・メッセージを受信するには、このオプションを選択する。

**注意** *GCL機能を有効にすると、GLO機能を無効にするまでデータストリーム機能は自動的に無効になる。*



### 設定メニュー

Favorite Group	デバイスの追加、現在のデバイスの変更または削除、現在のデバイスの並べ替え、デバイスの接続診断を含む、お気に入りグループの設定を行います。
Refresh Serial and Ethernet	これにより、Adam/Apax .NET Utility はシリアルと LAN ネットワーク接続をリフレッシュします。
Add COM Ports	Adam/Apax .NET Utilityにシリアル COM ポートを追加します（ADAM-6000 モジュールには適用されません）。
Show TreeView	クリックすると、Module Tree表示エリアが表示されます。Allow
Calibration	モジュールのキャリブレーションを有効/無効にします。

### ヘルプメニュー

Check Up-to-Date on the We	アドバンテックのダウンロードウェブサイトに接続し、ユーティリティの最新バージョンを確認します。
About	現在コンピュータにインストールされている Adam/Apax .NET Utilityのバージョン情報が表示されます。

## 6.3.2 ツールバー

ツールバー（図6.2）には、よく使うメニュー項目のアイコンがあります。

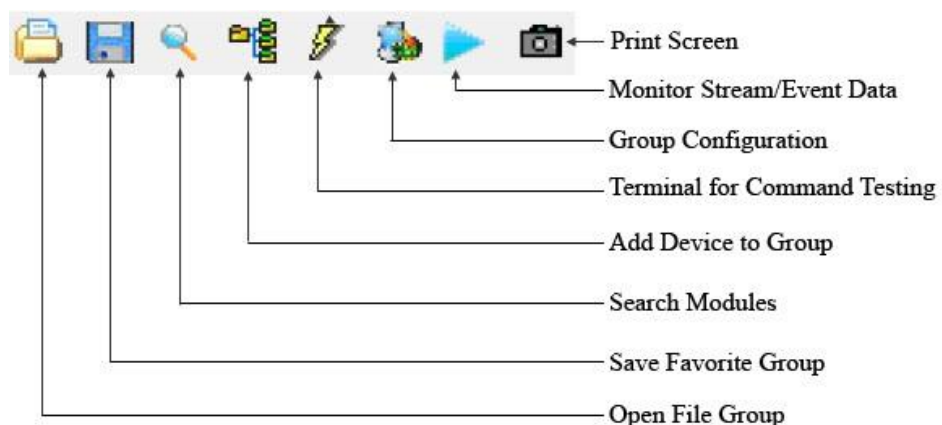


図 6.2 Adam/Apax .NET Utilityツールバー

### 6.3.3 Module Tree表示エリア

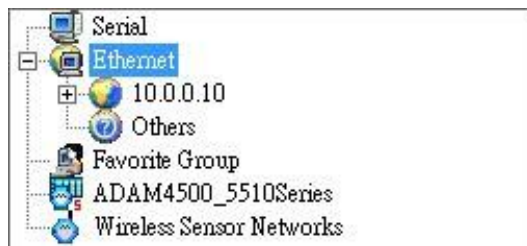


図 6.3 Adam/Apax .NET Utility・モジュールの表示領域

Module Tree表示エリアは、メインウィンドウの左側の部分です。表示エリアには大きく分けて5つのカテゴリーがあり、そのうちのいくつかは特定のモジュールが接続されている場合にのみ表示されます：


シリアル	すべてのシリアルI/Oモジュール（ADAM-4000、ADAM-4100、ADAM-5000 ホストPCIに接続されたRS-485モジュール）は、このカテゴリにリストされます。
イーサネット	ホストPCIに接続されているすべてのイーサネットI/Oモジュール（ADAM-5000、 ADAM-6000、およびADAM- 6100 TCPモジュール）は、このカテゴリにリストされま す。
お気に入りのグループ	個人のお気に入りグループに追加したデバイスは、このカテゴリの下に表示され、 特定のモジュールを見つけやすくなります。お気に入りグループには、複数のグ ループを含めることができます。新しいグループを作成するには、 <b>Favorite Group</b> を右クリックし、 <b>Add New Group</b> を選択します。グループ名を入力するプロ ンプトが表示されます。そのグループにデバイスを追加するには、作成したグ ループを右クリックし、「 <b>新しいデバイスを追加</b> 」を選択します。すると、新しい デバイスの名前を入力するプロンプトが表示されるので、「 <b>Serial Device</b> 」タブ または「 <b>Ethernet Device</b> 」タブからモジュールタイプを選択します。ここでデバ イスのパラメータを入力することもできます。グループの修正[ <b>Modify Group</b> ]を 選択）とグループの削除（[ <b>Delete Group</b> ]を選択）に加えて、グループ名を右ク リックしてグループの接続診断（[ <b>Diagnose Connection</b> ]を選択）を選択すること もできます。
ADAM-4500_5510Series	DOSベースのリモートコントローラ（ADAM-4500やADAM-5510シリーズなど）は、こ のカテゴリにリストされます。
Wireless Sensor Net- work	ホストPCIに接続されているワイヤレスモジュール（WISE-4000シリーズなど） は、このカテゴリにリストされます。

### 6.3.4 ステータス表示エリア

ステータス表示エリアは、あなたが操作するメインウィンドウです。すべてのテストはここ  
で行われます。このウィンドウの内容は、Module Tree表示選択した項目によって異なりま  
す。

### 6.3.5 ADAM-6000モジュールの構成

ADAM-6000 モジュールがホスト PC に接続され、検索されるとModule Tree表示エリアのEther-net カテゴリに表示されます。Module Tree表示領域でEthernetカテゴリを選択し

をクリックし、ツールバーの**モジュール検索アイコン**  します。Adam/Apax .NET Utility は、イーサネット・ネットワーク上のすべての ADAM-6000 モジュールを検索します。モジュールの接続が初めての場合、IP はデフォルトで 10.0.0.1 となり、Module Tree表示エリアのその他に表示されます。

#### 注意



ネットワークファイアウォールが有効になっている場合、ADAM-6000 モジュールに接続できないことがあります。Adam/Apax用の例外を追加する必要があるかもしれません。

Windowsコントロールパネル経由でWindowsファイアウォールの.NETユーティリティ。

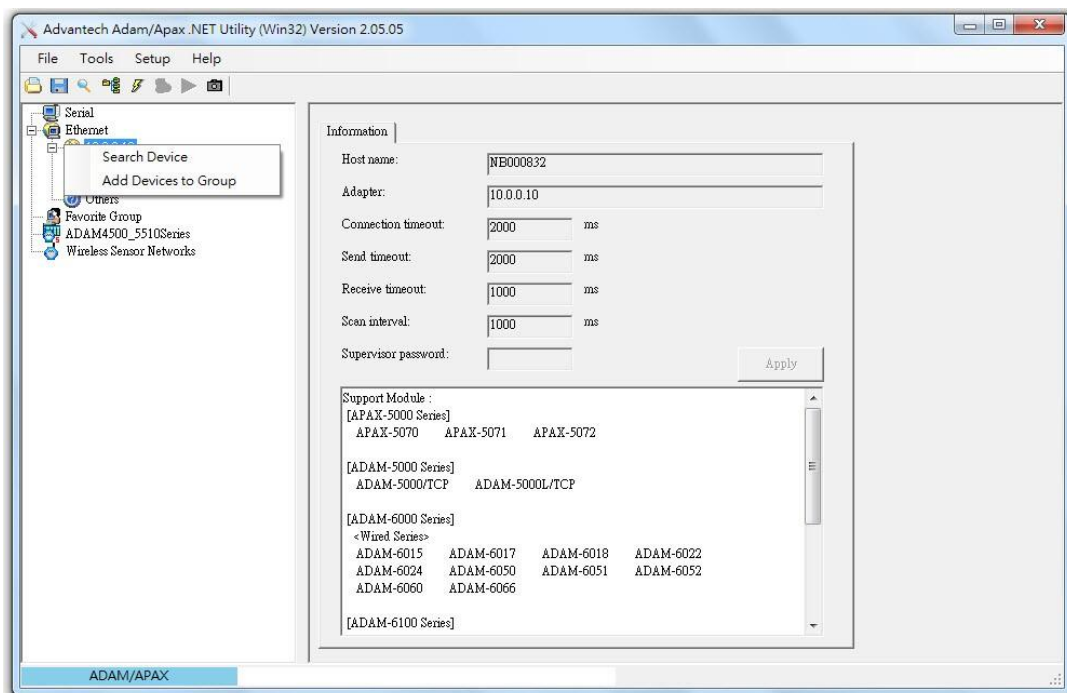
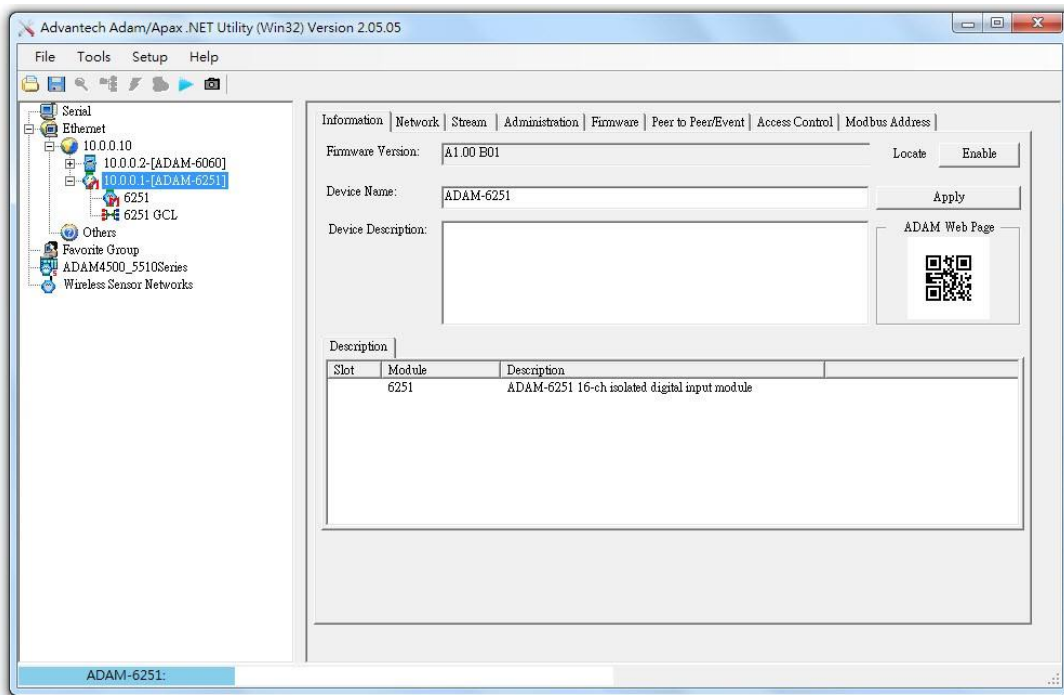


図 6.4 Adam/Apax .NET Utility - デバイスの検索

ADAM-6000モジュールのIPアドレスをホストPCと同じサブネットになるように変更する必要があります。ステータス表示エリアに正しいIPアドレス、サブネットアドレス、デフォルトゲートウェイを入力し、[Apply Change]をクリックします。パスワードの入力を求めるダイアログボックスが表示されます。ADAM-6000モジュールのデフォルトパスワードは“00000000”（引用符なし）です。正しいパスワードを入力すると、ADAM-6000モジュールはホストPCのIP下に入ります。パスワードは後で変更できます。

Module Tree表示エリアで使用する ADAM-6000 モジュールの IP アドレスを選択すると、ステータス表示8つのタブが使用可能になります。これらのタブは、そのモジュールの一般的な設定を行うためのものです。設定を変更したら、忘れずに **Apply** または **Apply Change** をクリックします。これらの8つのタブの詳細は、次のセクションで説明します。

## 情報タブ



このタブには、ファームウェアのバージョン、デバイス名、デバイスの説明が表示されます。モジュールに特定の名前と説明を付けると、複数のADAM-6000モジュールが同じネットワークに接続されている場合に便利です。また、選択したモジュールの物理的な位置を特定するためのロケート機能を有効/無効にすることもできます（基本的に、“Enable”をクリックすると、モジュールのステータス/リンクLEDインジケータが30秒間赤くなります。1.5 LEDステータスの説明）。このタブには、選択したモジュールのウェブサーバの URL 用に生成される QR コードも表示されます。個々のモジュールのコンフィギュレーションは、“Firmware”タブから保存/ロードできることに注意してください（このセクションで後述）。コンフィギュレーション・ファイルには、ネットワーク、ストリーム/イベント・データ、アクセス・コントロール、I/Oコンフィギュレーションの設定が含まれます。

## ネットワーク・タブ

Information Network Stream Administration Firmware Peer to Peer/Event Access Control Modbus Address

Network Setting

MAC Address: 00-D0-C9-FF-FF-A0

IP Address: 10.0.0.1

Subnet Address: 255.255.255.0

Default Gateway: 10.0.0.10

Host Idle (Timeout): 720 second(s)

Apply

IP Mode

Static  DHCP

Note: The 'Host Idle' will affect TCP connection. Please make sure the value is applicable.

Application Network Setting

Datastream Target Port (Default:5168): 5168 Apply

P2P/GCL Target/Local Port (Default:1025): 1025 Apply

Network Diagnostic (Default:On) Apply

このタブには、「ネットワーク設定」パネルと「アプリケーション・ネットワーク設定」パネルの2つのメイン・パネルがあります。これらのパネルの内容については、以下の文章で説明します。

### ネットワーク設定パネル

このパネルは、ADAMモジュールの一般的なネットワーク構成設定を調整するためのものです。ここでは、ネットワーク接続プロトコル（DHCPまたは静的IP）、IPアドレス、サブネットアドレス、デフォルトゲートウェイ、ホストのアイドル時間（タイムアウト）を設定できます。

### アプリケーション・ネットワーク設定パネル

このパネルは、データストリームとP2P/GCLポートを設定するためのものです。Network Diagnosticを選択すると、ADAMモジュールはイーサネットスイッチを定期的に監視および診断します。イーサネットポートを通信に使用しない場合、この機能は無効にしてください。

注1: ADAM-6000モジュールのウェブページを開くためにウェブ・ブラウザを使用する場合、Java仮想マシン（JVM）はJarファイルをダウンロードするために複数のTCPコネクションを使用します。これらのコネクションは、Jarファイルのダウンロードが完了すると解放されません。

注2: GCL/P2Pポート設定が変更された後、モジュールは自動的に再起動します（接続回復時間: 3秒）。

## ストリーム・タ

The screenshot shows the configuration interface for the ADAM-6000 module. The 'Stream' tab is active, and the 'Data Streaming' sub-tab is selected. On the left, under 'Hosts to receive data', there are 8 rows. The first row is checked, and its IP address is set to 10.0.0.2. The other rows are unchecked and have the IP address 255.255.255.255. Each row has an 'Apply' button. On the right, the 'Data Streaming' section is set to 'Adam-5000/TCP Event Trigger'. Below this, the 'Sending Interval' is defined as (50ms ~ 10 hours). The interval is set to 0 hours, 0 minutes, 0 seconds, and 100 milliseconds. An 'Apply change' button is located at the bottom right of the configuration area.

ADAM-6000モジュールは、最大8台のホストにデータを定期的送信するように設定できます。この一連の信号をデータストリームと呼びます。**Stream** タブの **[Hosts to receive data]** パネルで、ADAM-6000モジュールからデータを受信するホストのIPアドレスを定義できます。**Data Streaming** タブ（右側）では、ADAM-6000モジュールがデータをホストに送信する間隔も設定できます。

**注意** 上記の画像では、*ADAM-5000/TCP Event Trigger* タブはADAM-5000専用です。



## 管理タブ

The screenshot shows the 'Administration' tab selected in the top navigation bar. The main content area is divided into three sections:

- Password Setting:** Contains three input fields for 'Old password', 'New password', and 'Verify password', each filled with asterisks. To the right of the 'Old password' field is an 'Apply change' button. To the right of the 'New password' field is a 'Reset password' button.
- Reset to Factory Defaults:** Contains a single 'Apply' button.
- System Restart:** Contains a single 'Apply' button.

**注意** デフォルトのパスワードは "0000000" です。



管理]タブでは、選択したADAM-6000モジュールのパスワードを設定できます。パスワードを変更するには、[Old password]ボックスに現在のパスワードを入力し、[New password]ボックスと[Verify password]ボックスに新しいパスワードを入力する必要があります。パスワードは多くの設定や操作に必要なため、独自のパスワードを設定することで、システムのセキュリティを確保することができます。また、工場出荷時のデフォルト設定を適用し、モジュールを再起動することもできます。

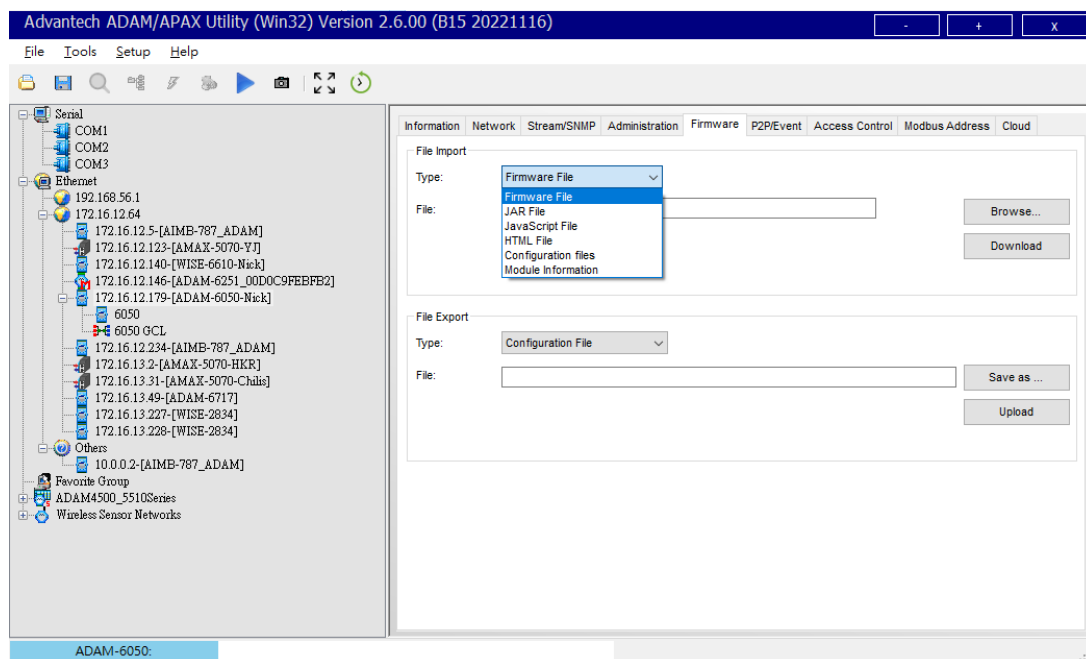
### 工場出荷時の設定にリセット

選択したADAM-6000モジュールのシステム構成をクリアして工場出荷時のデフォルト設定に戻すには、このパネルでApplyをクリックします。

### システム再起動

Apply をクリックしてモジュールを再起動します。現在の設定が保持されることに注意してください。

## ファームウェア・タブ



アドバンテックは、ADAM-6000モジュールの機能を追加または改善するために、新しいファームウェア・バージョンをリリースすることがあります。http://www.advantech.com、最新のファームウェアのダウンロードをご確認ください。ファームウェアのダウンロードには4つのファイルタイプがあります：Bin、Html、Xml、Jarです。Binファイルは実際のファームウェア・ファイルで、HtmlファイルとJarファイルはADAM-6000モジュールのウェブサーバ用です。

### ファイルインポートパネル

ここでADAM-6000モジュールにファームウェアをインポートすることができます。Browseをクリックして、コンピュータ上の3つのファームウェアファイルを選択します。次に、Downloadをクリックして、新しいファームウェアをADAM-6000モジュールにインストールします。

### ファイルエクスポートパネル

このパネルで ADAM モジュール設定ファイルをエクスポートできる。Save As... をクリックし、保存先のファイルパスを選択します。次に、Upload をクリックして設定保存します。

## ピアツーピア/イベントタブ

Information | Network | Stream | Administration | Firmware | **P2P/Event** | Access Control | Modbus Address | Cloud

Mode  
 Basic  Advanced  Disable Apply

Basic (One to One)

Period time:  second(s)

Deviation enable  (C.O.S.)      Deviation Value:  %FSR

Source  → Destination

Modify channel enable

Channel	Enable	Range	Only positive value valid

Refresh | Save | Load | Apply list

このタブでは、P2P（イベント）機能を有効化し、設定することができます。P2P（イベント）機能の詳細については、セクション6.7を参照してください。

## アクセス・コントロール・タブ

Information | Network | Stream | Administration | Firmware | P2P/Event | Access Control | Modbus Address | Cloud

Controlled By

IP address     MAC address    Refresh    Apply

Security IP/MAC Setting

Enable/Disable

<input checked="" type="checkbox"/> 0.	172	18	3	52	Apply	Apply all
<input checked="" type="checkbox"/> 1.	172	18	3	116	Apply	
<input type="checkbox"/> 2.	255	255	255	255	Apply	
<input type="checkbox"/> 3.	255	255	255	255	Apply	
<input type="checkbox"/> 4.	255	255	255	255	Apply	
<input type="checkbox"/> 5.	255	255	255	255	Apply	
<input type="checkbox"/> 6.	255	255	255	255	Apply	
<input type="checkbox"/> 7.	255	255	255	255	Apply	

このタブでは、選択した ADAM-6000 モジュールを制御できるコンピュータ/デバイスを設定します。まず、[Controlled By]パネルでIPアドレスまたはMACアドレスのいずれかを選択し、[Apply]をクリックします。次に、[Security IP/MAC Setting]パネルで[Enable/Disable]チェックボックスを選択し、許可されたコンピュータ/デバイスのIPアドレスまたはMACアドレスを直接入力します。最後に、1つのIP/MACアドレスに変更を適用するには「適用」をクリックするか、すべての変更を適用するには「すべて適用」をクリックします。上の画像では、IPアドレス172.18.3.52と172.18.3.116だけが選択したADAM-6000モジュールの制御を許可されています。チェックボックスが選択されていない場合は、どのコンピュータ/デバイスでも選択したモジュールを制御できます。

## Modbusアドレスタブ

Coils Status (0X) :				Holding Registers (4X) :			
	Item	Length	Base		Item	Length	Base
▶	DO status	02	0017	▶	AI value	08	0001
	Reset historical max AI value	08	0101		AI average	01	0009
	Reset historical max AI average	01	0109		Historical max AI value	08	0011
	Reset historical min AI value	08	0111		Historical max AI average	01	0019
	Reset historical min AI average	01	0119		Historical min AI value	08	0021
	Burnout flag	08	0121		Historical min AI average	01	0029
	High alarm flag	08	0131		AI float value	16	0031
	High alarm flag of average	01	0139		AI float average	02	0047
	Low alarm flag	08	0141		Historical max AI float value	16	0051
	Low alarm flag of average	01	0149		Historical max AI float average	02	0067
	Clear GCL counter	08	0161		Historical min AI float value	16	0071
					Historical min AI float average	02	0087
					AI status	16	0101


Refresh Apply

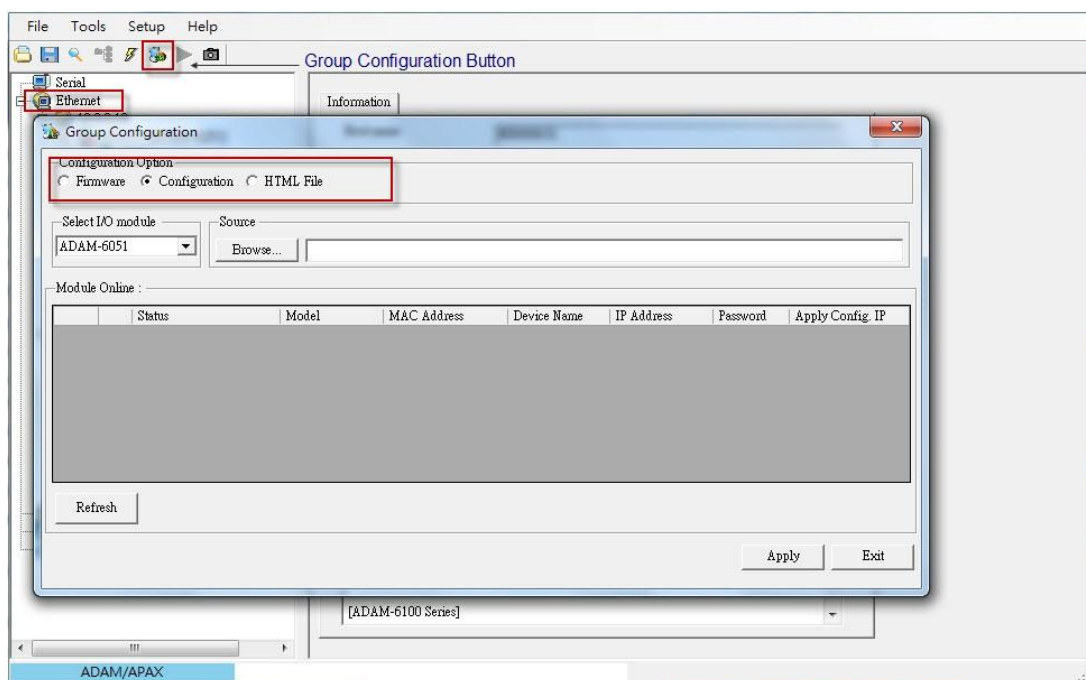
ADAM モジュールのデプロイにおいてより高い柔軟性とスケーラビリティを提供するために、Modbus アドレス設定の制限は、モジュールを可能な限りコンフィギュラブルにするために取り除かれました。基本的に、各機能を設定するために 2 種類の Modbus アドレスセクション (0X と 4X) があります。例えば、上の図は ADAM-6017 の Modbus アドレス設定です。

### 6.3.6 グループ構成

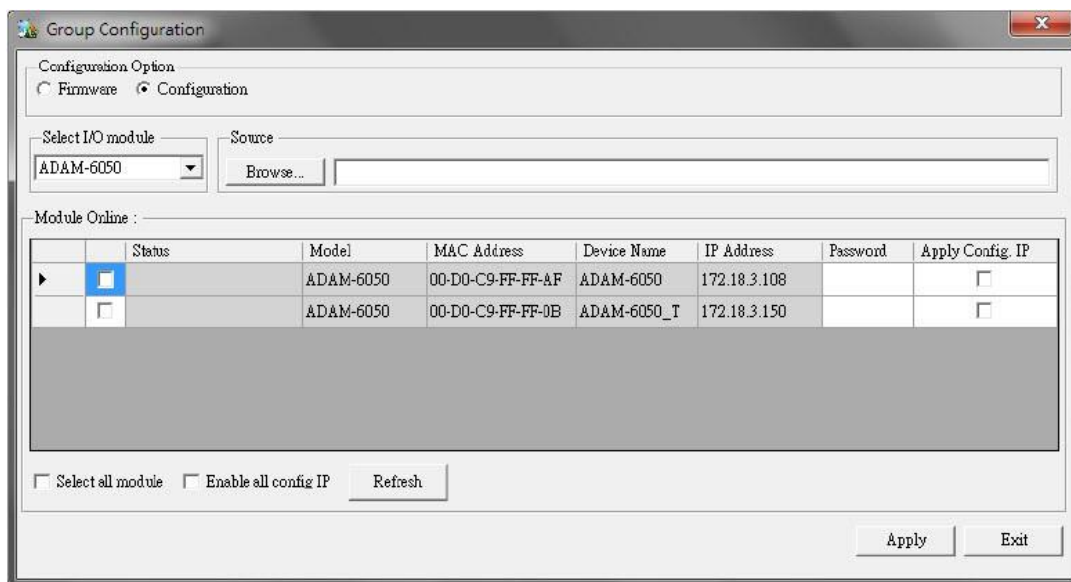
特定のアプリケーションでは、異なるサイトで同じタスクを実行するため、複数のモジュールに同じ設定を適用する必要があります。以前は、ユーザーは、現場配備の前に、各モジュールを個別に設定する必要があった。モジュールがインストールされ、システムが稼働した後も、ファームウェアのアップデートを実行するために、繰り返し労力を必要とする。

この問題を解決するために、ADAM-6000モジュールはグループ・コンフィギュレーション機能を備えており、複数のモジュールのコンフィギュレーションの繰り返しを減らし、コンフィギュレーションを高速化することができます。以下の手順に従って、グループ・コンフィギュレーション・ウィンドウを開いてください：

1. Module Tree表示エリアの **Ethernet** をクリックします。
2. ツールバーの **Group Configuration** アイコン  するか、ツールメニューの **Group Configuration** を選択します。以下のウィンドウが開きます：



3. ファームウェア、コンフィギュレーション、HTMLファイルのいずれかを選択します。
4. アップデートを適用するI/Oモジュールのタイプを選択します（ネットワーク上のこのタイプのすべてのモジュールが選択されます）。
5. **Browse**をクリックすると、使用するファームウェア/設定ファイルを選択するプロンプトが表示されます。
6. どのモジュールの再設定/アップデートを行うかを選択し、パスワードを入力します。デフォルトのパスワードは“00000000”（引用符なし）であることに注意してください。



7. **Apply** をクリックして変更を適用すると、ステータス表示操作の進行状況が表示されます。

**注意** Group Configuration機能の処理中は、モジュールの電源を切らないでください。そうしないと、モジュールがクラッシュします。



### 6.3.7 I/O構成

選択した ADAM-6000 モジュールの一般設定（前のセクションで説明）が完了したら、I/O チャンネルの設定（チャンネル・レンジ、キャリブレーション、アラーム設定など）を行う必要があります。同時に、入力チャンネルの値と出力チャンネルの設定値をステータス・パネルで確認できます。図 6.5 のModule Tree表示エリアを参照してください。設定したいADAM-6000 モジュールのIPアドレスをクリックすると、IPアドレスの下に2つの項目が表示されます：モジュール番号（全チャンネル設定の場合）とモジュール番号の後に “GCL ”が続く（GCL設定の場合）。モジュール番号の横にあるプラスとマイナスのコントロールをクリックすると、選択したモジュールのパスワードを入力するよう求められます。正しいパスワードを入力すると、モジュール番号の下に各チャンネルのリスト（各チャンネル構成の場合）が表示されます。

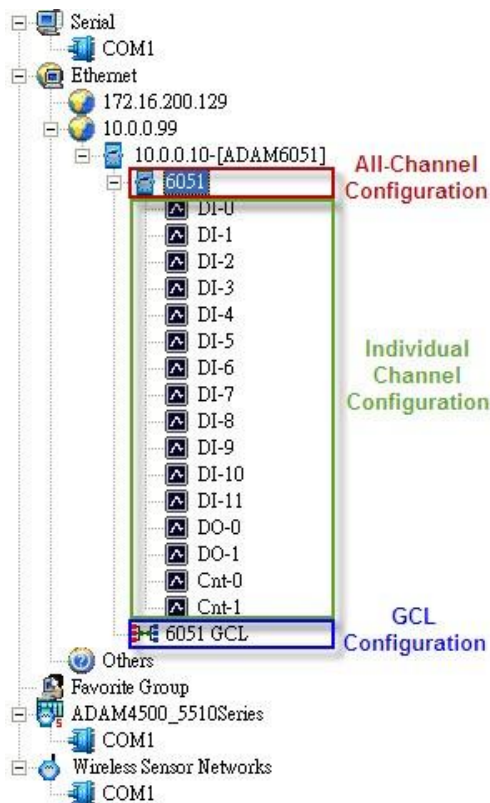


図6.5 全チャンネル、個別チャンネル、およびGCL構成コントロール

モジュール番号をクリックすると、すべてのチャンネルのアナログ入力値と構成設定がステータス表示表示されます。個別チャンネルの項目の 1 つをクリックすると、指定されたチャンネルの値 と構成設定がステータス表示領域に表示されます。以下のセクションでは、全チャンネルのコンフィギュレーションと個別チャンネルのコンフィギュレーションについて詳しく説明します。

## 6.4 アナログ入力モジュール (ADAM-6015、ADAM-6017、およびADAM-6018、ADAM-6018+)

### 6.4.1 全チャンネル構成

これらの ADAM-6000 モジュールでは、Module Tree表示領域でオール・チャンネル・コンフィギュレーション・アイテムをクリックすると、ステータス表示領域の4つの主要な部分は、入力レンジ、積分時間、キャリブレーション、およびチャンネル情報パネルになります。

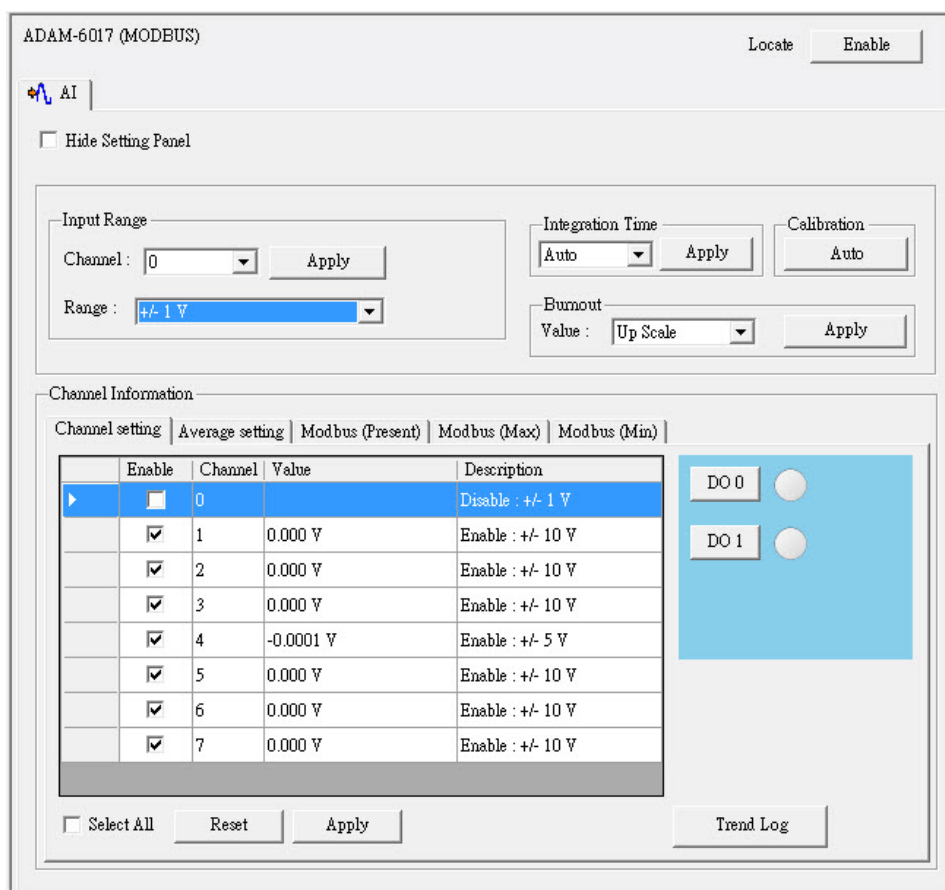


図 6.6 チャンネル範囲設定エリア

#### 入力範囲

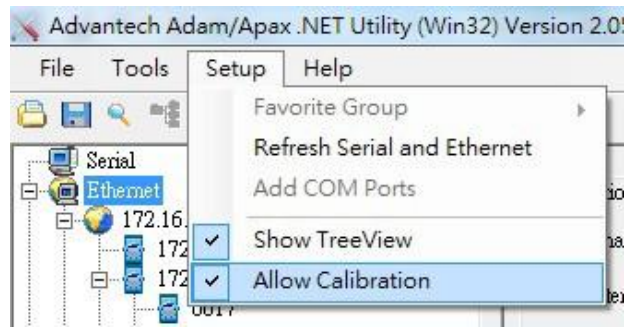
このパネルでは、各チャンネルに異なる範囲を設定することができます。これを行うには、**Channel** ボックスでチャンネル番号を選択し、**Range** ボックスで範囲を選択し、**Apply** をクリックして変更を受け入れます。

#### 統合時間

電源からのノイズを除去するために、ADAM-6000 シリーズのアナログ入力モジュールには内蔵フィルタ (50Hz および 60Hz) が搭載されています。この設定では、**Integration Time** ボックスから適用したいフィルタを選択します。次に、**Apply** をクリックして変更を受け入れます。

#### キャリブレーション

キャリブレーション設定を調整する前に、キャリブレーション機能を有効にする必要があります。これを行うには、**SetupメニューのAllow Calibration**をクリックします。



ADAM-6015、ADAM-6018、ADAM-6024の場合は、以下の手順に従って校正を行ってください：

#### **ゼロ校正**

1. キャリブレーションパネルの**ゼロ**をクリックします。
2. 校正が必要なチャンネルに、フル・スケール・レンジの最小値（例：0 V）の信号を接続する。
3. 配線が完了したら、**Apply**をクリックしてキャリブレーションを開始します。

#### **スパン校正**

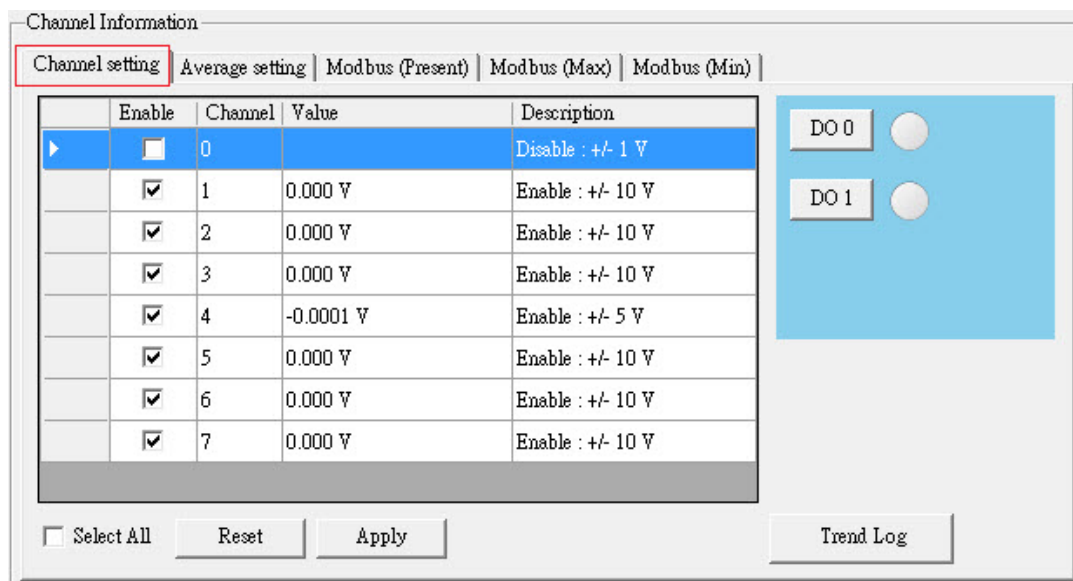
1. キャリブレーションパネルの**スパン**をクリックします。
2. 校正が必要なチャンネルに、フル・スケール・レンジの最大値（例：10 V）の信号を接続する。
3. 配線が完了したら、**Apply**をクリックしてキャリブレーションを開始します。

ADAM-6017では、手動キャリブレーションの代わりに自動キャリブレーションを実行できます。これを行うには、キャリブレーションパネルのAutoをクリックします。

#### **チャンネル情報**

このパネルには、全チャンネルのアナログ入力値を表示・設定するための 5 つのタブがあります：**Channel setting** タブ、**Average setting** タブ、**Modbus (Present)** タブ、**Modbus (Max)** タブ、**Modbus (Min)** タブ。

## チャンネル設定タブ



このタブには、アナログ入力チャンネルの現在値が表示されます。ADAM-6017 と ADAM-6018 の場合、デジタル入力チャンネルの値もこのタブに表示されます。モニターしたいチャンネルを選択して **Apply** をクリックします。

**トレンド・ログ** をクリックすると、選択したチャンネルの過去のトレンドを表示することもできます。図6.7に示すように、画面右側の**チャンネル設定**パネルでログを記録するチャンネルをチェックし、**Apply** をクリックすることで選択できます。**Start (開始)** をクリックすると、データロギングが開始され、リアルタイムで過去のトレンドが表示されます。その後、**停止** をクリックし、**ファイルに保存** をクリックすると、トレンドデータをコンピュータに保存できます。

**履歴を表示** および **履歴をクリア** をクリックすると、過去のトレンドデータが表示/クリアされ、**履歴を保存** をクリックすると、データを Csv ファイルとして保存できます。グラフをクリアするには、**グラフのクリア** をクリックします。画面右側の **BufferSize** ボックスに収集したいデータポイント数を入力し、**PollingInterval** ボックスにデータポーリング間隔を設定できます。

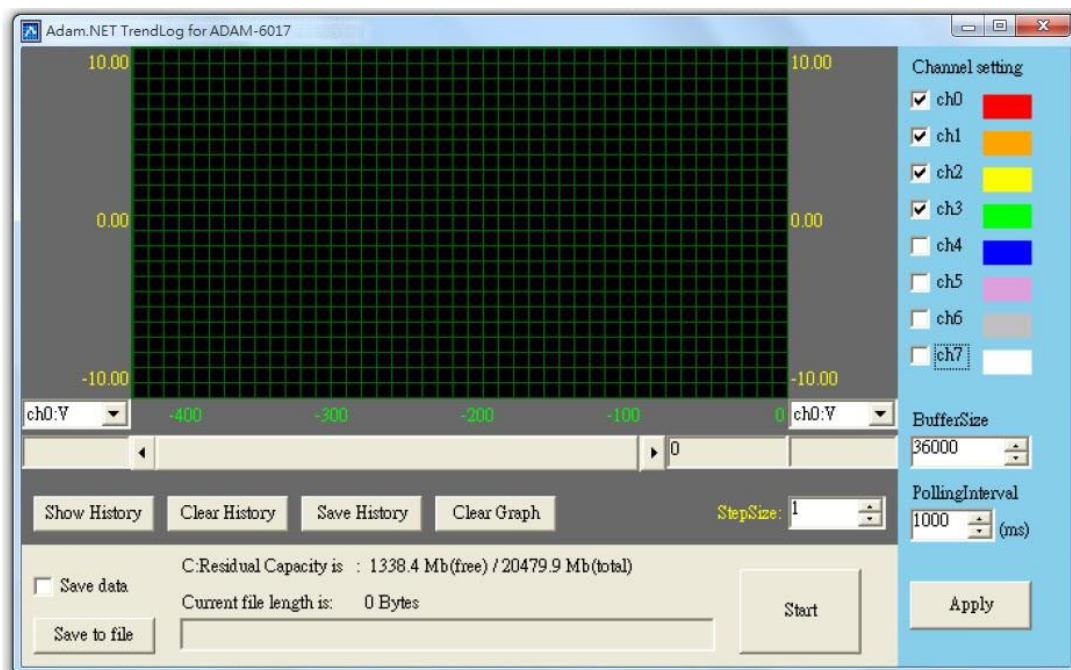
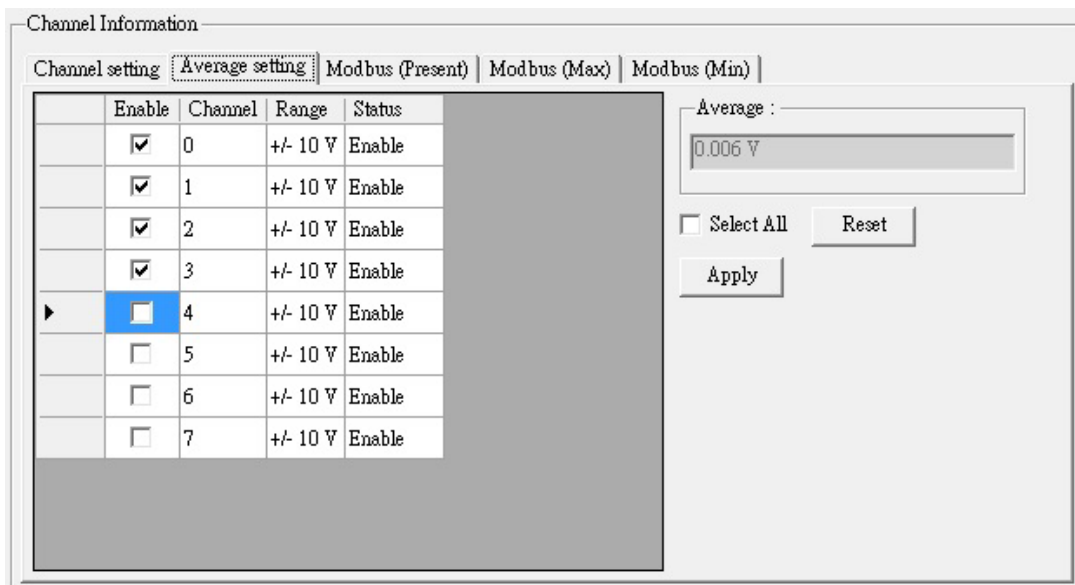


図 6.7 アナログ入力トレンドログ

ADAM-6015、ADAM-6018 の断線検出機能では、入力チャンネルにセンサが接続されていない場合、関連チャンネルの**情報**ボックスに「断線」のメッセージが表示されます。

### 平均設定タブ



ADAM-6015、ADAM-6017、ADAM-6018 は、内蔵プロセッサが実行する平均化機能を備えています。この機能を使用するには、**平均設定**タブで対象チャンネルにチェックを入れるだけです。上の例では、平均化機能は4つのチャンネル（チャンネル0~3）に対して有効になっています。有効化されると、選択された 5 チャンネルの平均値が **Average** ボックスに表示されます。

## Modbus (現在) タブ

Channel Information									
Channel setting		Average setting		Modbus (Present)		Modbus (Max)		Modbus (Min)	
Address	Type	Channel	Value[Dec]	Value[Hex]	Value[Eng]	Description			
▶ 40001	AI	0	*****	*****	*****	Disable : +/- 1 V			
40002	AI	1	32768	8000	0.000 V	Enable : +/- 10 V			
40003	AI	2	32768	8000	0.000 V	Enable : +/- 10 V			
40004	AI	3	32768	8000	0.000 V	Enable : +/- 10 V			
40005	AI	4	32767	7FFF	-0.0001 V	Enable : +/- 5 V			
40006	AI	5	32768	8000	0.000 V	Enable : +/- 10 V			
40007	AI	6	32768	8000	0.000 V	Enable : +/- 10 V			
40008	AI	7	32768	8000	0.000 V	Enable : +/- 10 V			
40009	AI	AVG	*****	*****	*****	Average disabled			

このタブは、現在のアナログ入力値の10進数、16進数、および関連するすべてのModbusアドレスの工学単位を表示します。

## Modbus (Max) タブ

Channel Information								
Channel setting		Average setting		Modbus (Present)		Modbus (Max)		Modbus (Min)
Address	Type	Channel	Value[Dec]	Value[Hex]	Value[Eng]	Description	Reset	
▶ 40011	AI	0	*****	*****	*****	Disable : +/- 1 V	Ch-0	
40012	AI	1	32769	8001	0.000 V	Enable : +/- 10 V	Ch-1	
40013	AI	2	32769	8001	0.000 V	Enable : +/- 10 V	Ch-2	
40014	AI	3	32769	8001	0.000 V	Enable : +/- 10 V	Ch-3	
40015	AI	4	32769	8001	0.0002 V	Enable : +/- 5 V	Ch-4	
40016	AI	5	32769	8001	0.000 V	Enable : +/- 10 V	Ch-5	
40017	AI	6	32768	8000	0.000 V	Enable : +/- 10 V	Ch-6	
40018	AI	7	32770	8002	0.001 V	Enable : +/- 10 V	Ch-7	
40019	AI	AVG	*****	*****	*****	Average disabled	AVG	

ADAM-6015、ADAM-6017、ADAM-6018 は、履歴最大値ログを備えています。関連するすべてのModbus アドレスの過去のアナログ入力最大値を、10進数、16進数、および工学単位で表示できます。ログを再初期化するには、Reset 列の対応するチャンネルボタンをクリックします。

## Modbus (Min) タブ

Channel Information

Channel setting | Average setting | Modbus (Present) | Modbus (Max) | **Modbus (Min)**

	Address	Type	Channel	Value[Dec]	Value[Hex]	Value[Eng]	Description	Reset
▶	40021	AI	0	*****	*****	*****	Disable : +/- 1 V	Ch-0
	40022	AI	1	32767	7FFF	0.000 V	Enable : +/- 10 V	Ch-1
	40023	AI	2	32767	7FFF	0.000 V	Enable : +/- 10 V	Ch-2
	40024	AI	3	32767	7FFF	0.000 V	Enable : +/- 10 V	Ch-3
	40025	AI	4	32766	7FFE	-0.0002 V	Enable : +/- 5 V	Ch-4
	40026	AI	5	32767	7FFF	0.000 V	Enable : +/- 10 V	Ch-5
	40027	AI	6	32767	7FFF	0.000 V	Enable : +/- 10 V	Ch-6
	40028	AI	7	32767	7FFF	0.000 V	Enable : +/- 10 V	Ch-7
	40029	AI	AVG	*****	*****	*****	Average disabled	AVG

ADAM-6015、ADAM-6017、ADAM-6018 は、履歴最小値ログを備えています。すべての関連する Modbus アドレスの過去の最小アナログ入力値を、10 進数、16 進数、および工学単位で表示できます。ログを再初期化するには、Reset 列の対応するチャンネルボタンをクリックします。

### 6.4.2 個別チャンネル構成

各チャンネルの設定項目をクリックすると、アナログ入力値を表示し、各チャンネルの設定を行うことができます（「平均設定」タブで設定した平均値もここに表示されることに注意してください）。ステータス表示エリアの上部には、図 6.8 に示すように、選択したチャンネルの現在のアナログ入力値と定義された範囲が表示されます。

ADAM-6017 Channel[0] alarm setting:

Input value:       Input range:

**High alarm** | Low alarm

Alarm mode:      

Alarm limit:      

Alarm status:      

DO mapping:

Channel:      

図 6.8 アナログ入力アラームモードの構成

ADAM-6015、ADAM-6017、ADAM-6018 では、この画面で内蔵アラーム機能を設定できます。選択したチャンネルの高アラームと低アラームを設定するための 2 つのタブが、ステータス表示エリアの下部にあります。

高アラームと低アラームの両方で、3つのアラームモードから1つを選択できます。

**アラームモードのボックス：**

- **無効**：アラームは無効で、アラーム状態が発生しても何も。
- **ラッチ**：アラーム状態が発生すると、アラームステータスはロジックハイに設定され、**アラームステータス LED** が点灯し続けます。ADAM-6017 および ADAM-6018 の場合、**D0 マッピング**パネルで指定された出力チャンネルは、ロジックハイ値を継続的に生成します。**Clear latch** をクリックして、アラームをクリアします。
- **モメンタリ**：アラームステータスは、アラーム状態が発生したか否かによって動的に変化します。アラーム状態が発生した場合、アラームステータスはロジックハイとなり、アラーム状態が解除されると、アラームステータスはロジックローに変化します。このオプションでは、**アラーム ステータス LED** とデジタル出力チャンネルがアラーム状態に応じて変化します。

アラームモードを選択した後、**Apply mode**をクリックして変更を適用します。

次に、アラームリミットボックスに値を入力し、**[Apply limit]** をクリックすることで、アラーム値の上限または下限を定義できます。アナログ入力値がアラーム上限値以上、またはアラーム下限値未満になると、アラーム条件が満たされ、アラームステータスがロジックハイに設定されます。ADAM-6015、ADAM-6017、ADAM-6018 の場合、アラームステータスは**アラームステータス LED** で表示されます。最後に、アラームを特定のデジタル出力チャンネルにマップするには、**Channel** ボックスから目的のチャンネルを選択し、**Apply** をクリックします。

## 6.5 ユニバーサルI/Oモジュール (ADAM-6024)

### 6.5.1 全チャンネル構成

ADAM-6024 はアナログ I/O とデジタル I/O チャンネルを備えています。Module Tree表示エリアのオール・チャンネル・コンフィギュレーション・アイテムをクリックすると、ステータス表示エリアに **Input** タブと **Output** タブの 2 つのタブが表示されます。**入力** タブには、ADAM-6015、ADAM-6017、および ADAM-6018 ページと同様に、[ステータス表示エリア]に重要な 4 つの主要エリアがあります。**入力範囲と積分時間**パネルのすべての設定は、これら3つのモジュールと同じです。ただし、これらのモジュールと異なり、ADAM-6024 には平均化、最大、最小の機能はありません。従って、ADAM-6024 の**チャンネル情報**パネルには、**チャンネル設定**タブと **Modbus (Present)**タブの2つのタブしかありません。

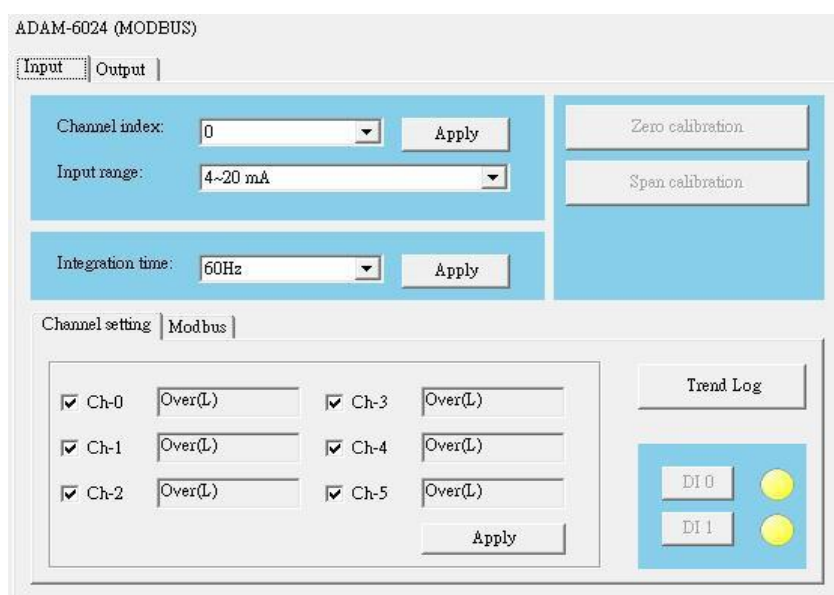


図 6.9 ADAM-6024 チャンネル構成

#### 入力タブ

このタブには、アナログ入力チャンネルの現在値が表示されます。**Enable** 欄のボックスにチェックを入れて監視したいアナログ入力チャンネルを選択し、**Apply** をクリックします。アナログ入力値が入力範囲外の場合、該当するチャンネルのボックスに「Over (L)」と表示されます。右側では、**DI 0** と **DI 1** の LED 表示により、現在のデジタル入力値を確認できます。また、トレンドログボタンをクリックすると、アナログ入力チャンネルの履歴をグラフ表示することができます。トレンドログの操作は、ADAM-6015、ADAM-6017、ADAM-6018 と同じです。

#### 出力タブ

このタブは、すべての関連する Modbus アドレスの現在のアナログ入力値を 10 進と 16 進形式で表示します。**Output** タブでは、アナログまたはデジタル出力チャンネルの値を設定し、関連するすべての行うことができます。

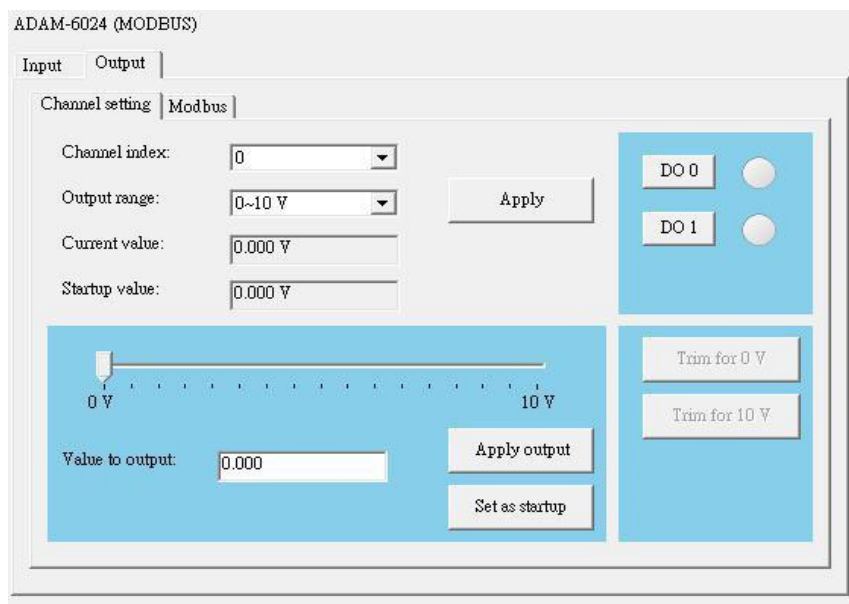


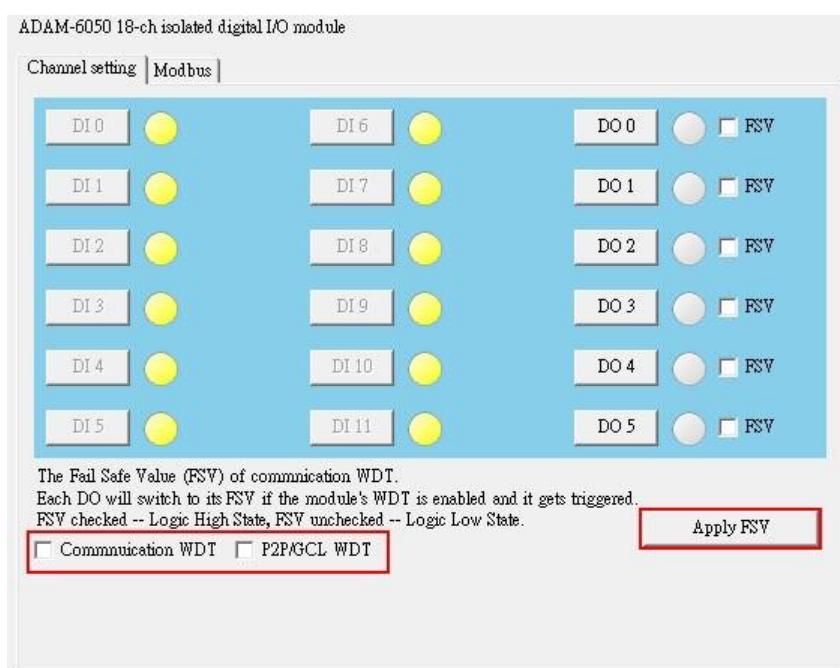
図 6.10 ADAM-6024 出力タブ

## 6.6 ユニバーサルデジタルI/Oモジュール (ADAM-6050, ADAM-6051- ADAM-6052, ADAM-6060, ADAM-6066)

### 6.6.1 全チャンネル構成

Module Tree表示エリアのオール・チャンネル・コンフィギュレーション・アイテムをクリックすると、ステータス表示エリアにチャンネル設定タブと Modbus タブの 2 つのタブが表示されます。以下の文章では、例として ADAM-6050 を使用します。

#### チャンネル設定タブ



このタブでは、各チャンネルのボタンの横にある LED から、すべてのデジタル入力チャンネルのステータスを確認することができます。また、対応するボタンをクリックすることで、すべてのデジタル出力チャンネルのステータスをコントロールすることができます。

### フェイルセーフ値の設定

ホストPCとADAM-6000デジタル・モジュール間の通信が確立されると、デジタル出力チャンネルはフェイルセーフ値（FSV）とれる定義済みの値を生成することができます。

チャンネルの横の FSV ボックス がチェックされているときは、WDT タイムアウトが発生したときにモジュールがその出力チャンネルをロジックHighに設定することを意味します。これには 2 つのアプリケーションがあります。すべての変更が完了したら、**Apply FSV** をクリックして変更を有効にします。

### 通信 WDT

モジュールがクライアントからTCPネットワーク・パケットをしばらく受信しなかった場合、待ち時間がホストのアイドル時間より長ければ、モジュールは自動的にFSVを出力に設定する。

### P2P/GCL WDT

モジュールがしばらくの間 P2P/GCL ネットワークパケットを受信しなかった場合、これは待機時間が入力したアイドル時間より長いことを意味します。

### Modbusタブ

Location	Type	Value	Description	Mode
00001	Bit	1	DI 0	DI
00002	Bit	1	DI 1	DI
00003	Bit	1	DI 2	DI
00004	Bit	1	DI 3	DI
00005	Bit	1	DI 4	DI
00006	Bit	1	DI 5	DI
00007	Bit	1	DI 6	DI
00008	Bit	1	DI 7	DI
00009	Bit	1	DI 8	DI
00010	Bit	1	DI 9	DI
00011	Bit	1	DI 10	DI
00012	Bit	1	DI 11	DI
00017	Bit	0	DO 0	DO
00018	Bit	0	DO 1	DO
00019	Bit	0	DO 2	DO
00020	Bit	0	DO 3	DO
00021	Bit	0	DO 4	DO
00022	Bit	0	DO 5	DO

このタブから、関連するすべての Modbus アドレスの現在のデジタル I/O 出力値を表示できます。

## 6.6.2 個別チャンネル構成

デジタル I/O チャンネルの値を表示し、設定を行うには、個々のチャンネル設定項目のリストで目的のチャンネルをクリックします。

### デジタル入力モード

各チャンネル設定項目のリストからデジタル入力チャンネルを選択すると、図 6.11 に示すようにステータス表示エリアが表示されます。

図6.11 デジタル入力モード

選択したデジタル入力チャンネルに対して、**DI mode** ボックスから異なる入力モードを選択することができます（選択するオプションはハードウェアの仕様によって異なります）。モードを選択した後、**Apply mode** をクリックして保存します。選択できる 5 つのモードの詳細は、以下の文章をご参照ください。

### DIモード：DI

このモードでは、**DIステータスLED**をクリックすることでデジタル入力値を確認できます。一部のデジタル・モジュールは、デジタル入力ステータスの反転をサポートしています。この機能を有効にすると、モジュールは自動的にデジタル入力値を反転します。例えば、実際の外部信号値がロジック・ローの場合、**DI ステータス LED** が点灯します（通常は、信号がロジック・ハイの場合にのみ点灯します）。お使いのモジュールがこの機能をサポートしている場合、**設定パネルに反転信号ボックス**が表示されます。このボックスを選択 / クリアしてこの機能を有効 / 無効にし、**Apply to all**（全チャンネルに適用）または **Apply**（選択したチャンネルに適用）をクリックして設定を完了します。

すべてのADAM-6000デジタル・モジュールは、高周波ノイズと低周波ノイズを除去するためのデジタル・フィルタをサポートしています。フィルタの有効/無効は、“**Enable digital filter**” ボックスを選択/クリアすることで設定できます。フィルタが有効になっている場合、“**Minimum low signal width**” と “**Minimum high signal width**” のボックスで、許容可能な信号幅の最小値と最大値を定義できます（単位は ms）。**Apply to all**（全チャンネルに適用）または “**Apply**”（選択したチャンネルに適用）をクリックして設定を完了します。

### DIモード: カウンター

ADAM-6051 DI[0] setting:

DI mode: Counter [Apply to all] [Apply mode]

Setting:  Invert signal [Apply to all] [Apply this]

Keep last value when power off

Enable digital filter

Minimum low signal width (1 ~ 65535) 1 0.1 ms

Minimum high signal width (1 ~ 65535) 1 0.1 ms

Counter value: 0 times [Stop] [Clear]

カウンターは、選択されたチャンネルからのデジタル信号のパルス数をカウントし、それをレジスタに記録します。**DI モード**ボックスで **Counter** を選択すると、ステータス表示エリアは **DI** を選択したときと同様に表示されます。このモードでは、選択したチャンネルの現在のカウント値が**カウンタ値**ボックスに表示されます。**カウンタ値**ボックスの横にある “**Start/Stop** ” をクリックすることで、カウンタを開始または停止することができます。“**Clear** ” をクリックすることで、カウンタをリセットすることもできます（レジスタの値もゼロに初期化されます）。

**DIモード**・ボックスでDIを選択した場合と同様に、**設定パネル**でデジタル入力ステータス反転機能とデジタル・フィルタを有効/無効にすることができます。ただし、追加設定として、モジュールの電源がオフになったときにカウンタが最後の値を保持するかどうかを定義できます。そうでない場合、モジュールの電源がオンになると、カウンターはゼロにリセットされます。**電源オフ時に最後の値を保持する** ボックスを選択 / クリアし、**[すべてに適用]**（すべてのチャンネルに適用）または **[この値を適用]**（選択したチャンネルに適用）をクリックして設定を完了すると、この機能を有効 / 無効にできます。

## DIモードロー・ハイ・ラッチ

ADAM-6051 DI[0] setting:

DI mode:

Setting:  Invert signal

Latch status:

Low-to-Highラッチ・モードとは、デジタル入力チャンネルがロジック・レベルのLowからHighへの変化を検出すると、手動でラッチをクリアするまでロジック・ステータスがロジックHighのままとなり、ロジック・ステータスがロジックLowに戻ることを意味します。ロジック・ステータスはラッチ・ステータスLEDで確認できます。ラッチをクリアするには、“Clear latch”をクリックします。このモードはデジタル入カステータスの反転機能もサポートしており、**Invert signal**ボックスをチェック／クリアしてから**Apply to all**（全チャンネルに適用）または**Apply this**（選択したチャンネルに適用）をクリックして設定を完了することで、有効／無効を切り替えることができます。

## DIモード：ハイ・ツー・ロー・ラッチ

ADAM-6051 DI[0] setting:

DI mode:

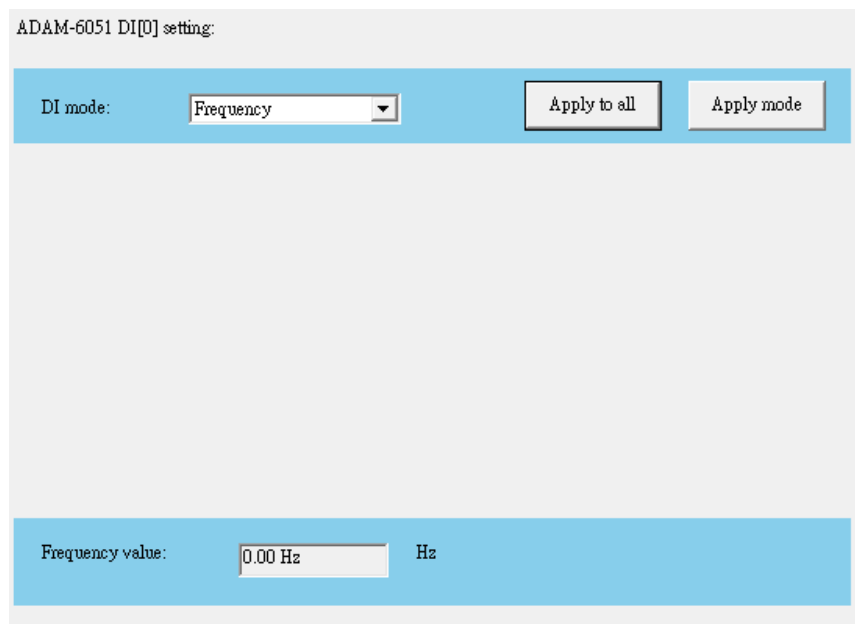
Setting:  Invert signal

Latch status:

High-to-Lowラッチ・モードとは、デジタル入力チャンネルがロジック・レベルのHighからLowへの変化を検出すると、手動でラッチをクリアするまでロジック・ステータスがロジックHighのままとなり、ロジック・ステータスがロジックLowに戻ることを意味します。ロジック・ステータスはラッチ・ステータスLEDで確認できます。ラッチをクリアするには、“Clear latch”をクリックします。このモードは、デジタル入カステータスの反転機能もサポートしています。

Invert シグナルボックスを選択 / クリアし、Apply to all（全チャンネルに適用）または Apply this（選択したチャンネルに適用）をクリックして設定を完了します。

### DIモード：周波数



ADAM-6051 DI[0] setting:

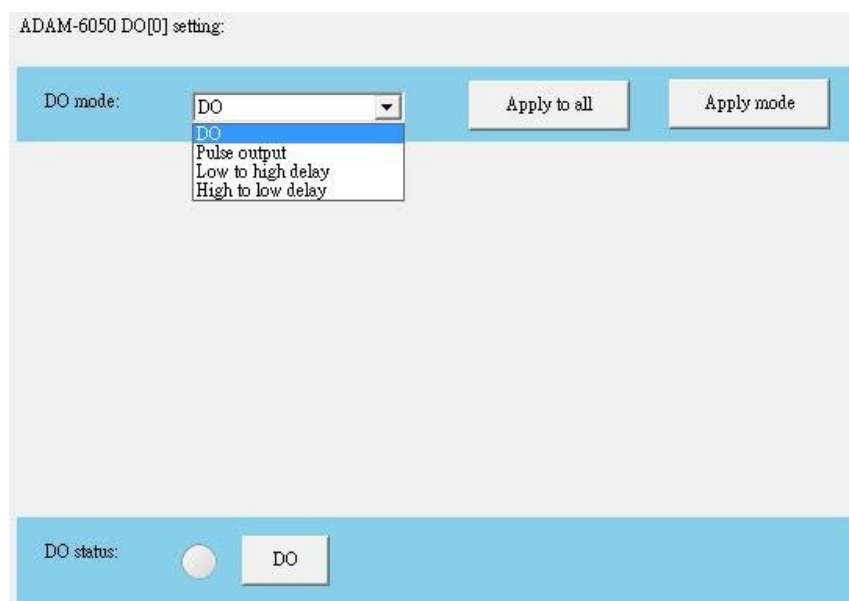
DI mode: Frequency [Apply to all] [Apply mode]

Frequency value: 0.00 Hz Hz

DIモードボックスでFrequencyを選択すると、モジュールは選択されたチャンネルのデジタル入力信号の周波数を計算します。この値は“Frequency”ボックスに表示されます。

### デジタル出力モード

各チャンネル設定項目のリストからデジタル出力チャンネルを選択すると、図 6.12 に示すようにステータス表示エリアが表示されます。



ADAM-6050 DO[0] setting:

DO mode: [DO] [Apply to all] [Apply mode]

DO status: [DO]

図6.12 デジタル出力モード

選択したデジタル出力チャンネルに対して、**DO mode** ボックスから異なる出力モードを選択できます（選択するオプションはハードウェアの仕様によって異なります）。モードを選択したら、**Apply mode** をクリックして保存します。選択できるモードは以下の 4 つです。

#### DOモード : DO

The screenshot shows the 'ADAM-6050 DO[0] setting' window. At the top, there is a 'DO mode:' dropdown menu set to 'DO'. To its right are two buttons: 'Apply to all' and 'Apply mode'. Below this is a large empty space. At the bottom, there is a 'DO status:' section with a yellow circular LED indicator and a 'DO' button.

このモードでは、選択したチャンネルのデジタル出力値をコントロールすることができます。現在のデジタル出力値は**DOステータスLED**で表示されます。

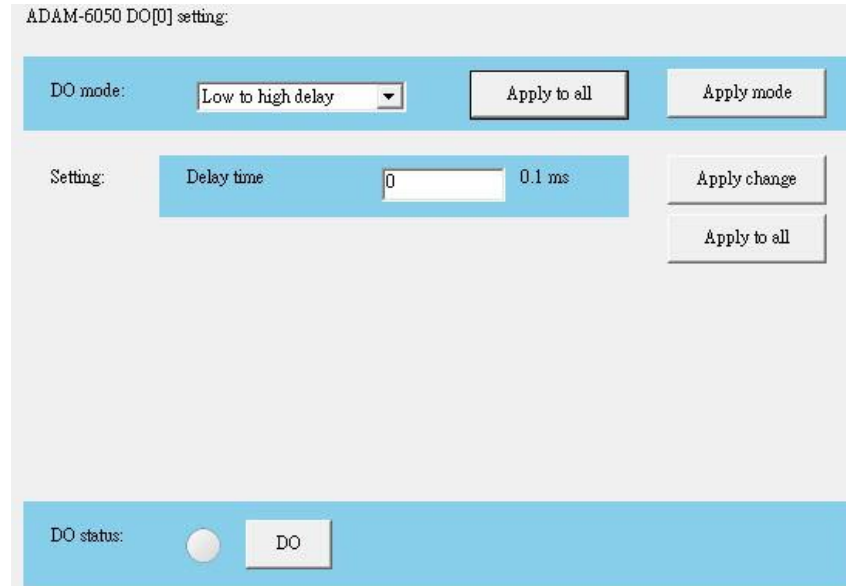
#### DOモード : パルス出力

The screenshot shows the 'ADAM-6050 DO[0] setting' window in 'Pulse output' mode. The 'DO mode:' dropdown is set to 'Pulse output'. To its right are 'Apply to all' and 'Apply mode' buttons. Below is a 'Setting:' section with an 'Apply change' button and an 'Apply to all' button. The settings are: 'Low signal width' (0 ms), 'High signal width' (0 ms), 'Output frequency' (Hz), and 'Duty cycle' (%). At the bottom, there is a 'Pulse output:' section with radio buttons for 'Continue' and 'Fixed total' (selected), a '100' input field, and 'Start' and 'Stop' buttons.

DOモードボックスで**パルス出力**を選択すると、選択されたデジタル出力チャンネルは連続パルス列または有限数のパルスを生成します。パルス幅は設定パネルのLow信号幅とHigh信号幅のボックスで定義できます（単位：0.1ms）。

パルス出力信号の周波数とデューティサイクルが自動的に計算され、**Output frequency** と **Duty cycle** のボックスに表示されます。設定が完了したら、**Apply mode**個別チャンネル用) または**Apply to all** (全チャンネル用) をクリックします。続いて、**Continue** (パルス列の場合) または **Fixed total** (固定パルス数の場合) を選択して、連続パルス列または固定パルス数を生成することができます。**Fixed total**を選択した場合は、生成したいパルス数を入力する必要があります。パルス出力モードが選択されたら、**Start/Stop**をクリックしてパルス出力を生成/停止します。

### DOモード: ロー・ツー・ハイ遅延



DOモードボックスから**Low to high delay**を選択すると、図6.13に示されているように、出力値がロジックLowからロジックHighに変化するまでに遅延があることを除けば、DOを選択したのと同じである。

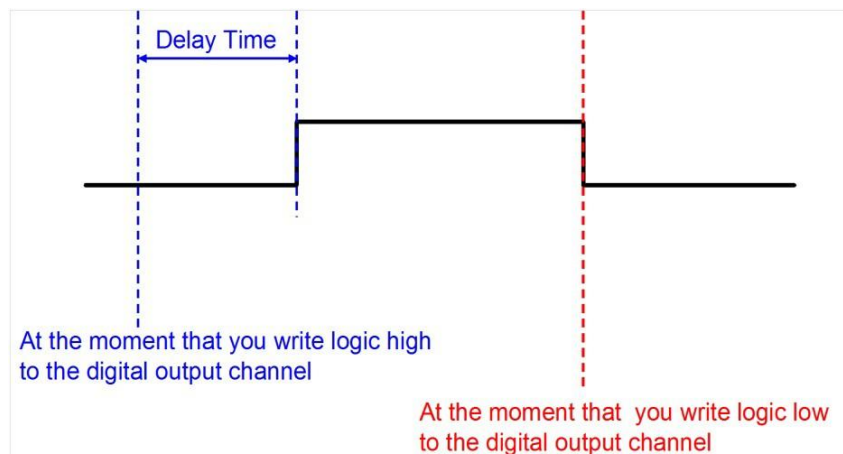
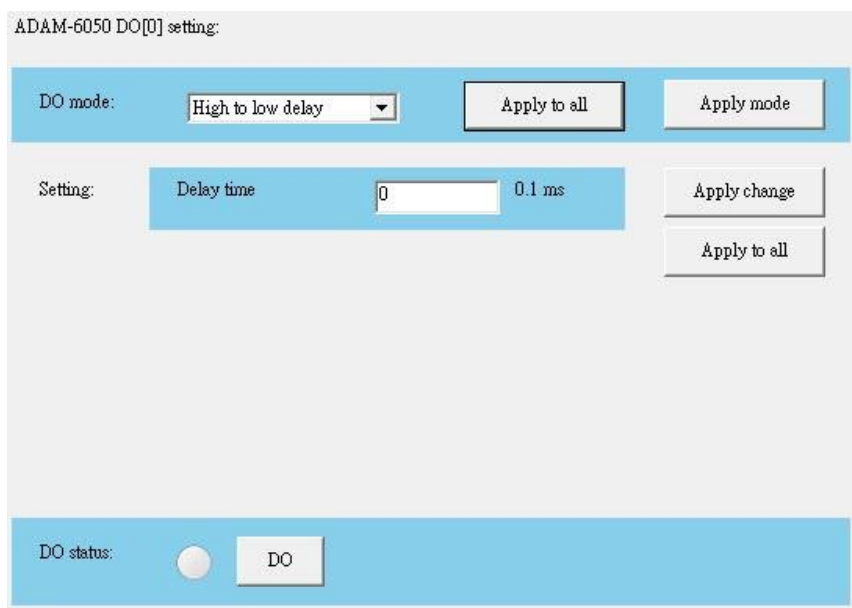


図6.13 Low-High遅延出力モードを説明するグラフ

遅延時間を設定するには、**Delay time** ボックスに値を入力し、**Apply** をクリックして設定を完了します。その後、**DO**をクリックすることでデジタル出力値を制御ことができ、**DO** ステータスLEDから現在値を確認することができます。

## DOモード：ハイ・ツー・ロー遅延



DOmodeボックスでHigh-to-low delayを選択すると、図6. 14に示されているように、出力値がロジックHighからロジックLowに変化するまでに遅延があることを除けば、DOを選択するのと同じである。

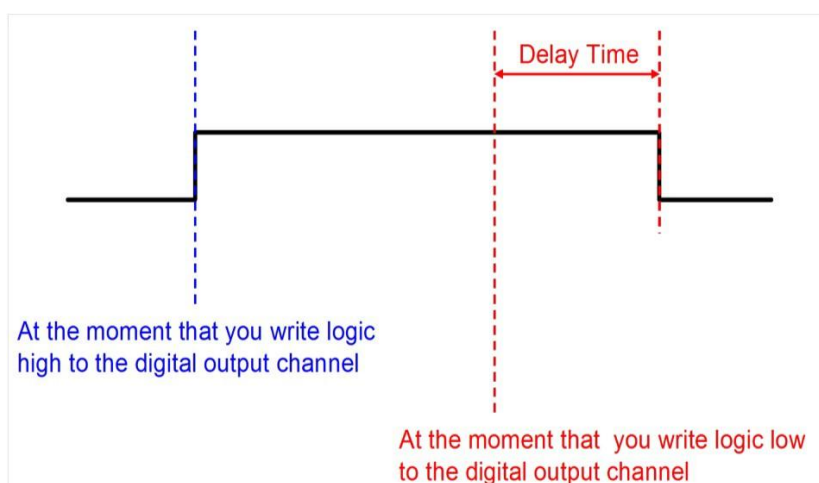


図6. 14 Low-High遅延出力モードを説明するグラフ

遅延時間を設定するには、Delay time ボックスに値を入力し、Apply をクリックして設定を完了します。その後、DOをクリックすることでデジタル出力値を制御することができ、DOステータスLEDから現在値を確認することができます。

## 6.7 P2P機能の紹介

あるモジュールから別のモジュールに信号を送りたい場合、P2Pは理想的なソリューションです。P2P機能を有効にすると、ADAM-6000モジュールは入力値をPCや他のADAM-6000モジュールなどの他のデバイスにアクティブに更新することができます。典型的なアプリケーションは、ADAM-6000モジュールのペアを使用することで、一方のモジュールの入力チャンネルの値が、もう一方のモジュールの出力チャンネルに自動的に更新されます。2つのADAM-6000モジュール間の接続がすでに確立されている限り、データは自動的に転送され、通信を処理するコントローラは必要ありません。

注



1. データパケットの衝突を防ぐため、1組のP2Pモジュール間にはイーサネットスイッチを使用してください（イーサネットハブは使用しないでください）。
2. ADAM-6000 モジュールは 2 つの機能をサポートしています：P2P（イベント）と GCL（第 8 章を参照）です。これら 2 つの機能を同時に有効にすることはできません。したがって、GCL が有効で P2P を使用したい場合は、最初に GCL を無効にする必要があります（GCL を無効にする方法については、セクション 8.2 を参照）。
3. P2P機能を使用するには、ADAM-6000モジュールのファームウェアバージョンを3.x以降にアップグレードする必要があります。

### 6.7.1 P2P通信モード

すべてのADAM-6000モジュールは、2種類のP2P機能を備えている：1) ベーシック・モードと2) アドバンス・モード。

#### 基本モード

ベーシック・モードでは、ソース・モジュール（モジュールA）からデータを受信するターゲット・デバイス（モジュールB）は1つだけです。通常、モジュールBは別のADAM-6000モジュールです。モジュールAの入力チャンネルはモジュールBの出力チャンネルにマッピングされ、モジュールAの全入力チャンネルの値がモジュールBの出力に自動的に更新されます。マスクを定義して、一部の入力と出力の関係を切断することもできます。

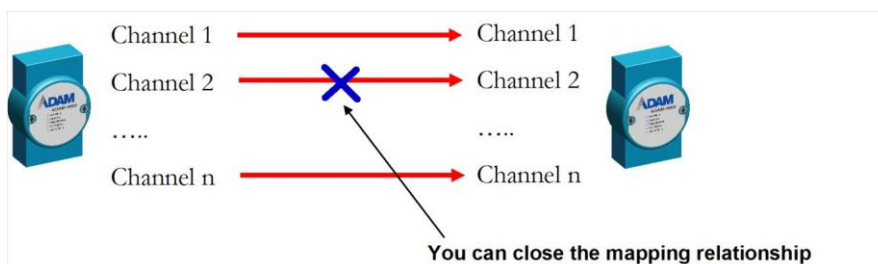


図6.15 P2Pの基本モード

### アドバンス・モード

アドバンス・モードでは、ソース・モジュール（モジュールA）からデータを受信する複数のターゲット・デバイス（モジュールB、モジュールCなど）が存在します。例えば、モジュール A の入力チャンネル 1 をモジュール B の出力チャンネル 3 にマッピングし、モジュール A の入力チャンネル 2 をモジュール C の出力チャンネル 4 にマッピングすることができます。

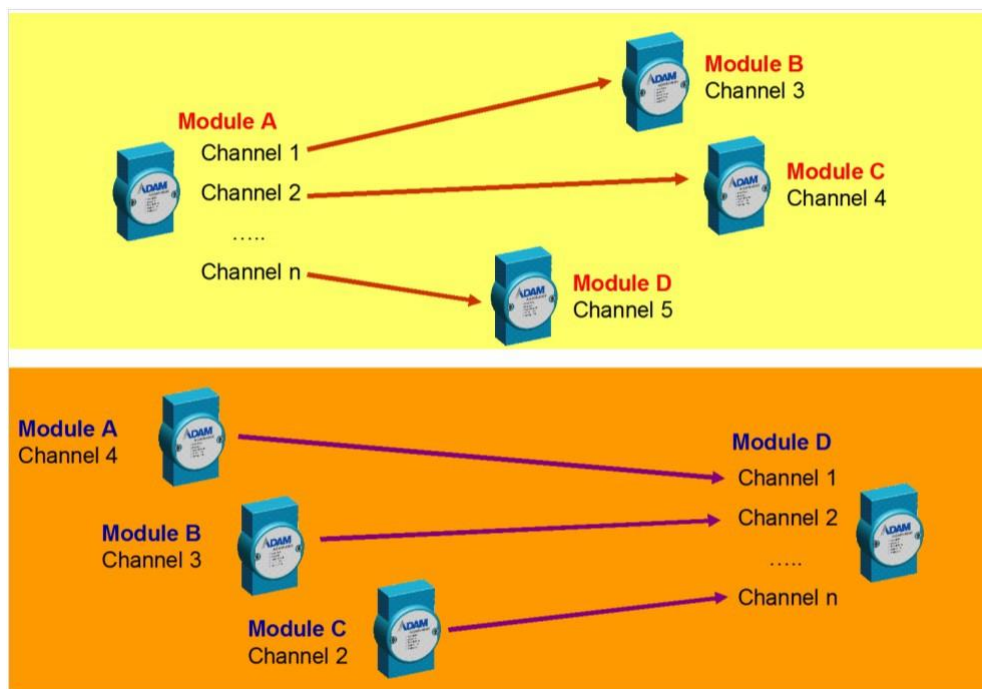


図6.16 P2Pのアドバンスモード

### 6.7.2 P2P通信方式

データがソース・モジュールからターゲット・デバイスに更新されるタイミングについては、1) 期間タイム、2) 期間タイム+COS (Change-of-Status) の2つのオプションから選択できる。

#### 期間

この機能により、入力チャンネルの値は、定義された周期でターゲット機器に更新される。

#### 期間+COS

このオプションでも、入力チャンネルの値は定義された周期でターゲット機器に更新されますが、COSが発生した場合（すなわち、アナログ入力値が指定された偏差よりも大きく変化した場合、またはデジタル入力のステータスが変化した場合）、入力チャンネルの値は即座にターゲット機器に更新されます。

### 6.7.3 P2Pイベントトリガー

多くのアプリケーションでは、デジタル信号やアナログ信号が変化した時など、特定のイベントが発生した時のみホスト・コンピュータにデータを送信する。このようなアプリケーションでは、P2P機能が理想的です。ターゲットとなるP2Pデバイスはコンピュータで、IPアドレスを入力し、通信モードとしてベーシックモード、通信方法としてピリオドタイム+COSを選択するだけです。

ホスト・コンピュータ上でデータを受信するためのプログラムが1つ動作している必要があります。コンパニオンCDにサンプルCプログラム（VC++ 6.0）を用意しています。ADAM-6000モジュールは定期的にホストコンピュータにデータを送信しますが（通信セキュリティのため）、メッセージが送信されたかどうかは、周期時間またはCOS機能によって区別できます。メッセージには、どのチャンネルが変更されたかの情報が含まれています。したがって、メッセージがすべてのチャンネルで変化なしを示す場合、イベントはいません。

**注意** ネットワーク通信には、必ずある程度の不確実性が存在する。イベント発生時にパケットロスが発生する場合があります。そのため、期間+COS機能（COS機能のみはなし）を用意しています。イベント発生時にパケットロスが発生しても、次の周期で再度データを送信します。これにより、システムの信頼性が向上します。

## 6.8 P2P機能の設定方法

Module Tree表示エリアから ADAM-6000 モジュールの IP アドレスを選択し、Peer to Peer/Event タブをクリックします。図 6.17 に示す画面が表示されます。

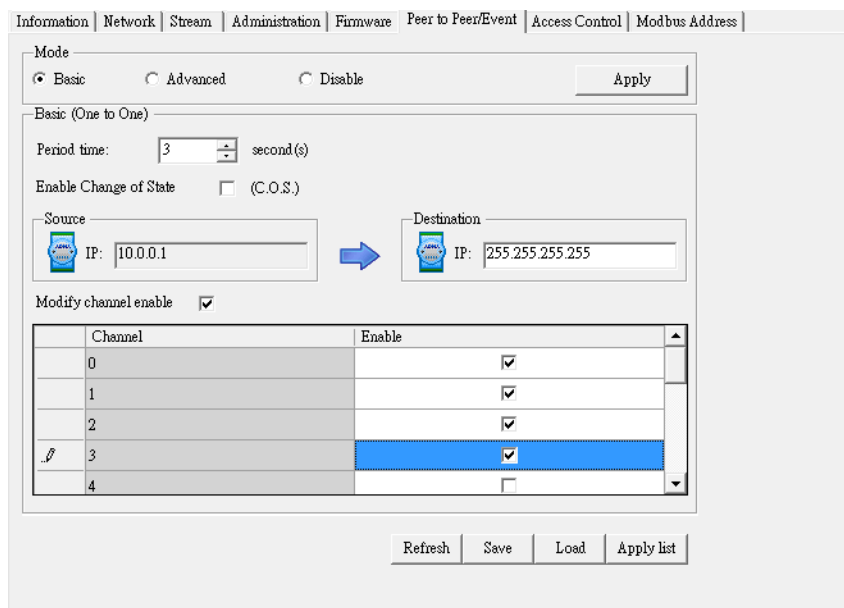


図6.17 ピアツーピア/イベントタブ

デフォルトでは、P2P機能は無効になっています。ModeパネルでBasicまたはAdvancedを選択し、Applyをクリックすることで有効にできます。ADAM-6000モジュールは、P2PとGCL機能の両方をサポートしていますが（GCLについては第8章を参照）、一度に有効にできるのはどちらか一方のみです。GCLが既に有効で、P2Pを有効にすることを選択した場合、最初にGCLを無効にするよう求めるアラートが表示されます（GCLを無効にする方法については、第8.2節を参照）。GCLを無効にした後、BasicまたはAdvancedを選択してP2Pを有効にすることができます。

## 6.8.1 基本モード設定

基本モードでは、図6.18に示すようにステータス表示エリアが表示されます。基本(One to One)パネルのDestinationボックスにIPアドレスを入力することで、ターゲットデバイスを定義できます。

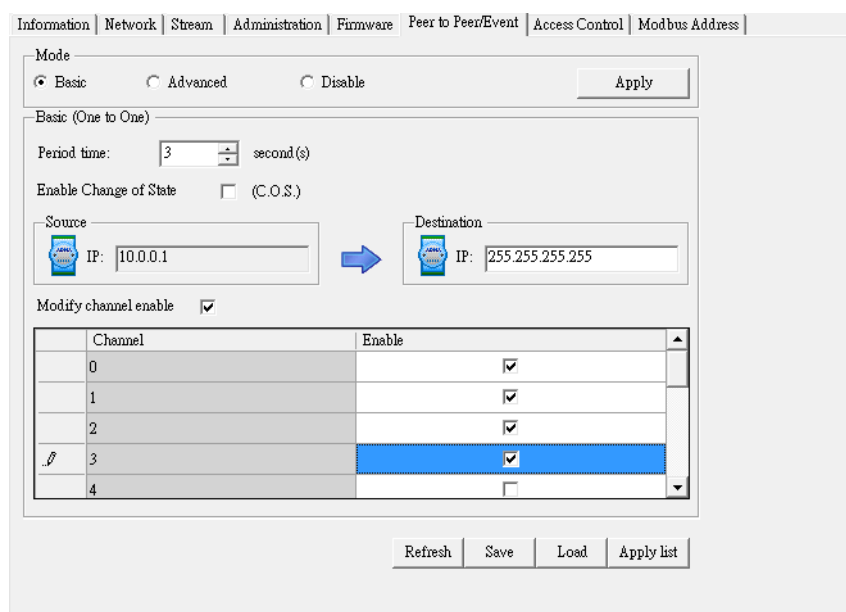


図 6.18 P2P 基本モードの構成

周期時間 + COS を選択するには、Deviation Enable (C.O.S) ボックス（アナログモジュール用、この例では図示せず）または Enable Change of State ボックス（デジタルモジュール用）を選択する必要があります。このボックスを選択しないと、通信方法はピリオドタイムになります。

ソース・モジュールからデスティネーション・モジュールへデータを転送する期間は、基本(One to One)パネルの Period time ボックスで設定できます。アナログ入力の偏差は、偏差率の数値コントロールで定義できます（値はパーセンテージで、変化値を全レンジで割ったものです）。

デフォルトでは、ソースモジュールのすべてのインプットチャンネルはすべて、デスティネーションモジュールのすべてのアウトプットチャンネルにマップされます。しかし、Modify channel enable ボックスをクリックすることで、どのチャンネルをマップするかを手動で設定することができます。Enable 列でチャンネルを選択し、Apply list をクリックすることで、対応する出力チャンネルにマッピングする入力チャンネルを選択できます。図6.18では、ソースモジュールの入力チャンネル0~3の値が、デスティネーションモジュールの出力チャンネル0~3に更新されます。Save をクリックすると、現在のマッピング関係を設定ファイルに保存できます。Load をクリックすると、マッピング設定ファイルをロードできます。Refresh をクリックすると、ソースモジュールの現在のマッピング設定が表示されます。

## 6.8.2 アドバンスモード設定

詳細モードでは、図6. 19に示すようにステータス表示エリアが表示される。マッピング関係は、ソースパネルとデスティネーションパネルのコントロールを使用して設定します。

Ch	C.O.S.	Period time	Map to IP	Map to ch	Map to Module	Deviation...
0	No	5	255.255.255.255	1	ADAM-6060/W	****
1	No	5	255.255.255.255	1	ADAM-6060/W	****
2	No	5	255.255.255.255	1	ADAM-6060/W	****
3	No	5	255.255.255.255	1	ADAM-6060/W	****
4	No	5	255.255.255.255	1	ADAM-6060/W	****
5	No	5	255.255.255.255	1	ADAM-6060/W	****

図6. 19 P2Pアドバンスト・モード設定

以下の手順に従って、マッピング関係を定義してください：

1. ソースパネルのチャンネルボックスから入力チャンネルを選択します。
2. 周期時間、偏差有効 (C. O. S) ボックス (アナログモジュールの場合) または状態変化 (C. O. S) ボックス (デジタルモジュールの場合)、およびソースパネルの偏差レートを使用して、そのチャンネルのデータを転送するタイミングを定義します。
3. DestinationパネルのIPボックスにターゲットモジュールのIPアドレスを入力します。
4. Nameボックスからターゲット・モジュールの名前を選択する。
5. チャンネルボックスから、データを受信ターゲットモジュールの出力チャンネルを選択します。
6. コンフィグをクリックしてリストを表示する

これらの手順を完了すると、Advanced (OnetoMulti) パネルの下部にあるマッピング・テーブルにそのチャンネルのコンフィギュレーションが表示されます。

マッピングしたい入力チャンネルごとに、ステップ1~4を繰り返す必要があります。すべての入力チャンネルを設定したら、Apply list をクリックしてマッピング設定をターゲットモジュールにダウンロードします。Save をクリックすると、マッピングテーブルのすべての設定をファイルに保存できます。Load をクリックすると、以前の構成ファイルをロードすることもできます。Refresh をクリックすると、ソース・モジュールの現在の構成がマッピング・テーブルに表示されます。

### 注意



1つのチャンネルの設定を何度もダウンロードするのではなく、全チャンネルのマッピング設定を一度にダウンロードすることをお勧めします。これは、ターゲット・モジュールのフラッシュ・メモリーの使用回数を減らすことができ、フラッシュ・メモリーの寿命を延ばすことができるからです。

時間を節約するために、チャンネル設定を他のチャンネルにコピーしてから、各チャンネルの最終調整を行うことができます。これを行うには、**Copy To** をクリックして図 6.20 のウィンドウを開きます。

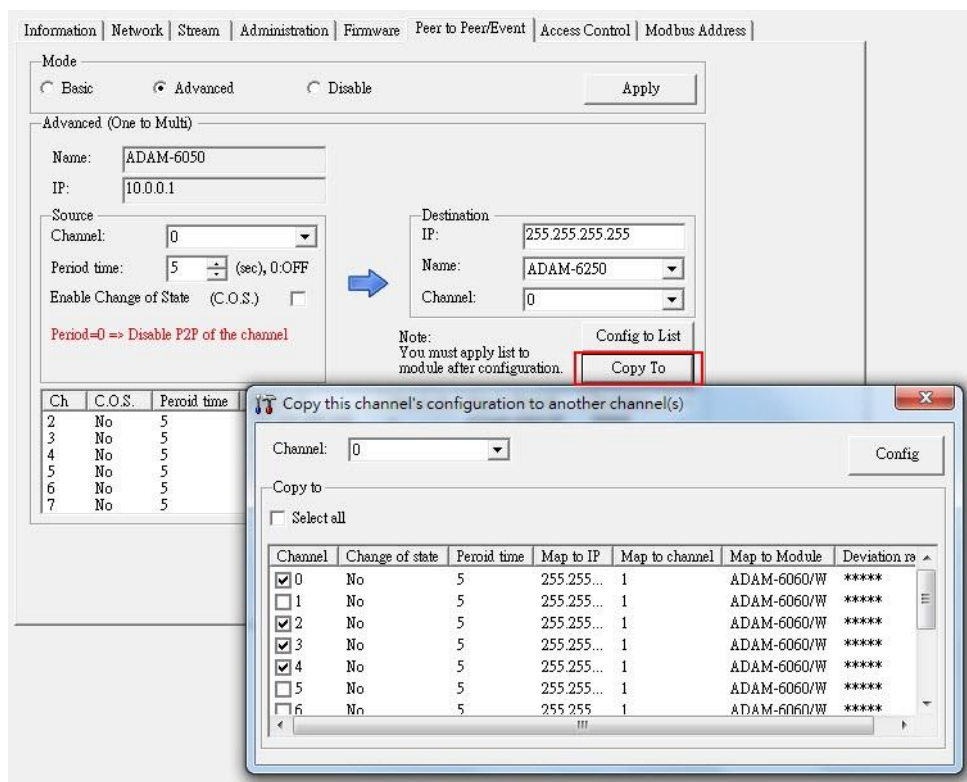


図6.20 一つの設定を他のチャンネルにコピーする

ここで、**Channel** ボックスからコピーしたいチャンネルを選択し、**Copy to** パネルの **Channel** 列から設定をコピーしたいチャンネルを選択し、**Config** をクリックします（すべてのチャンネルにコピーする場合は **Select all** にチェックを入れます）。この例では、チャンネル 0 の設定がチャンネル 0、2、3、4 にコピーされます。

ピアツーピア/イベントタブに戻ると、選択したチャンネルの設定がマッピングテーブルに表示されていることがわかります。必要なチャンネルを選択し、パラメータを変更します。

## P2Pデータ転送性能

### 有線LANモジュール

条件：1つのイーサネット・スイッチを介して、ADAM-6050モジュールの1つのチャンネルから別のADAM-6050モジュールの1つのチャンネルにデータを転送する。

データ転送時間： <1.2 ms

## 6.9 ADAM-6000 ウェブサーバ

ADAM-6000モジュールはすべて、標準的なウェブ・ブラウザを使ってアクセスできるウェブサーバを内蔵しています。このウェブサービスにより、プログラマはウェブプログラミング言語を使用して、強力な顧客向けウェブページを作成することができます。したがって、リモート・コンピュータやデバイスを使用して、ウェブ・ブラウザ経由でADAM-6000モジュールのI/Oステータスをリモートで監視および制御することができます。ADAM-6000モジュールには、HTML5を使用して変更できるデフォルトの内蔵ウェブページが付属しています。

ADAM-6000モジュールのデフォルトのHTML設定は、HTML5をサポートしていません。HTML5をサポートするには、アドバンテックのウェブサイトから新しいファイルをダウンロードする必要があります。

ウェブサーバにアクセスするには、ブラウザにIPアドレスを入力してADAM-6000モジュールに接続します。ユーザー名（デフォルト：root）とパスワード（デフォルト：00000000）の入力を求められます。正しいパスワードを入力したら、ADAM-6000I/Oチャンネルの監視/制御を開始できます。

### 6.9.1 HTML 5

#### HTML 5入門

HTMLは、ウェブコンテンツのデザインで最も広く使われている言語です。最新バージョンであるHTML 5では、構文構造が強化され、複数のウェブ技術（CSSやJavaScriptなど）が組み込まれているため、モバイル通信におけるウェブサービスやAPI、インタラクティブ・アプリケーションの追加実装が可能になっている。

#### ADAM-6000 Webサーバーによる遠隔監視と制御

この新機能は、イーサネット・ネットワークを介してどこからでもフィールド・メンテナンスが行えるという点で、明らかなメリットをもたらします。このセクションでは、PC、ラップトップ、スマートフォンやパッドなどのポータブル・デバイスからリモート・モニタリングやリモート・コントロールを行うための、システムとデバイスの接続と設定方法について説明します。

プリセット条件：

1. ADAM-6000モジュールのインストールとネットワーク設定を完了します。
2. ADAM-6000モジュールがローカル・イーサネット・ネットワークに接続されていることを確認してください。

注意ADAM-6000モジュールは、パブリックHTML5ベースで開発されていますが、詳細表示やデータ転送モードは、オペレーティングシステムのウェブページで異なる場合があります。最低限必要なブラウザは以下の通りです：

#### スマートフォンのブラウザ要件

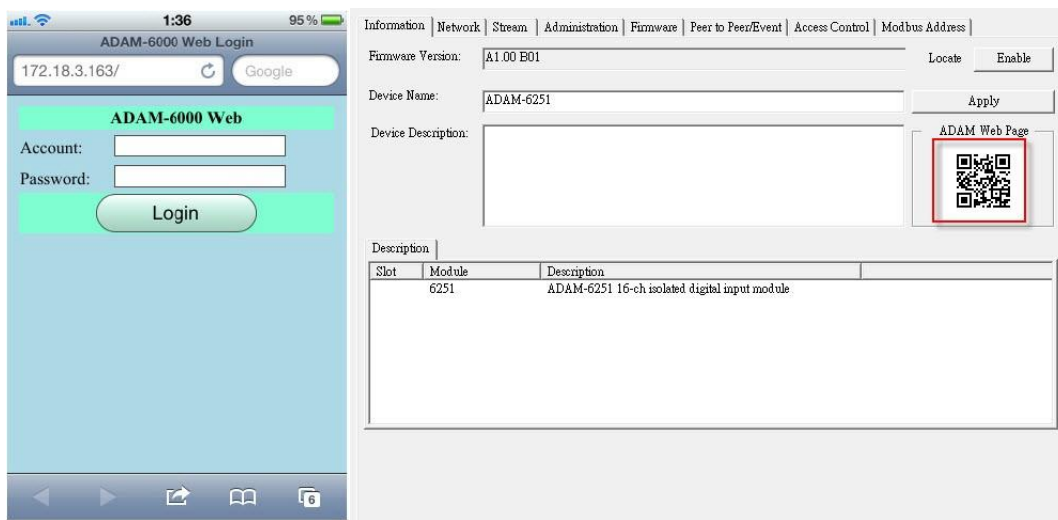
- アプリiOS版Safari5
- Google Android 4.0 (Ice Cream Sandwich) 用ウェブ・ブラウザ
- Chrome for Google Android 4.0 (Ice Cream Sandwich)

#### PCブラウザの要件

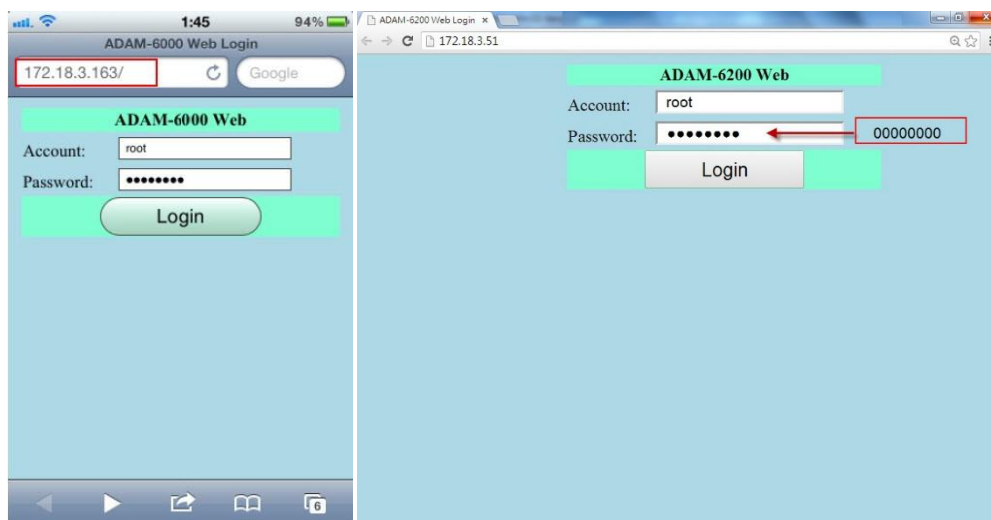
- インターネットエクスプローラー（バージョン9）
- Google Chrome（バージョン8）
- Mozilla Firefox（バージョン10）

### スマートフォンの操作手順

1. スマートフォンをローカル・イーサネット・ネットワークに接続し、ブラウザを開きます。
2. スマートフォンを使用して、ユーティリティに表示されるQRコードをスキャンするか、スマートフォンのブラウザのアドレスバーにモジュールのIPアドレスを入力し、ログインページにアクセスします。



3. アカウントとパスワードを入力し、ログインをクリックします。



#### 注



1. デフォルトのアカウントは "root" で、デフォルトのパスワードは "00000000" (引用符なし) です。
2. コンピュータからログインするには、IPアドレスをブラウザに入力するだけです。

4. ログイン後、操作ページが表示されます。このページでは、I/O ステータス（トレンドログ）を監視し、個々のチャンネルの出力ロギングを有効にすることができます。例として、下の画像は、DO 0、DO 2、DO 3、DO 5 にチェックを入れ、Apply Output をクリックすると、選択したチャンネルの電球が点灯し、トレンドログも変化します。



# 7

アプリケーション・プログラムの計画

## 7.1 はじめに

システム・コンフィギュレーションが完了したら、アプリケーション・プログラムの計画を始めることができます。この章では、DA&C システムを実装するための 2 つのプログラミングツールを紹介します。DLLドライバとコマンドセットは、アプリケーションとADAM-6000 I/Oモジュールを相互接続するためのユーザーフレンドリーなツールです。

## 7.2 ADAM .NET CLASS LIBRARY

アドバンテックのADAM .NET Class Libraryを使用すると、.NET Framework 2.0をサポートするアプリケーションプログラムを素早く簡単に開発することができます。Adam .NET Class Libraryには、ADAM-6000モジュールを利用するために必要なすべての機能が含まれています。

ADAM .NET Class Libraryをインストールする前に、お使いの PC が .NET Framework 2.0。 <http://support.advantech.com/support/> からダウンロードできます。ADAM .NET Class Library をインストールするには、セットアップファイル ADAM .NET Class Library.msi を実行してください。

ADAM .NET Class Libraryのインストールが完了すると、Win32 クラス・ライブラリが以下のパスにインストールされます：

- また、WinCE Class Libraryは、以下のインストールされます：
- Program Files\Advantech\AdamApax.NET Class Library\Class Library\WinCE  
短時間でアプリケーションプログラムの開発に慣れるために、以下の多くのサンプルプログラムがあります：
- Program Files\Advantech\AdamApax.NET Class Library\Sample Code これらの例は、以下のプログラム・コードの書き方の基本的なガイドを提供する。ADAM-6000 モジュールの制御多くの場合、IPアドレスとモジュール名を変更した後、単純にプログラムをコンパイルして実行できます。以下の例は ADAM-6052 用です。コードは Microsoft Visual Studio の Visual Basic で記述されています。この例のコード (ADAM 60xxDIO.sln) は以下のパスにあります：
- Program Files\Advantech\AdamApax.NET Class Library\Sample Code\ADAM\Win32\VB\ADAM-6000 Series\Adam60XXDIO.  
ソースコードを開き、IPアドレスとモジュール名を必要に応じて修正するだけだ（コメントとして無視されないようにアポストロフィを削除することを忘れずに）。

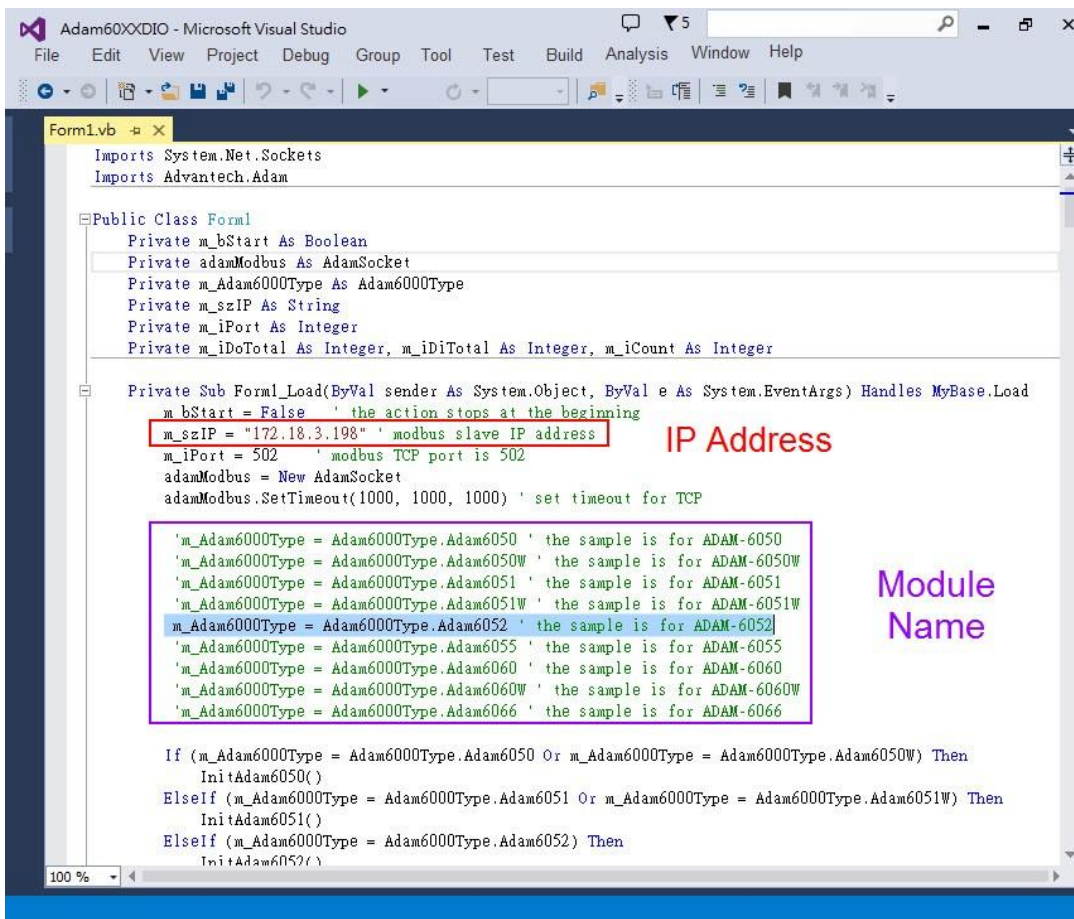


図7.1 ADAM-6050 .NETの修正

コードを修正した後、プログラムをコンパイルし、それを実行してアプリケーションを起動し、モジュールを設定することができます。

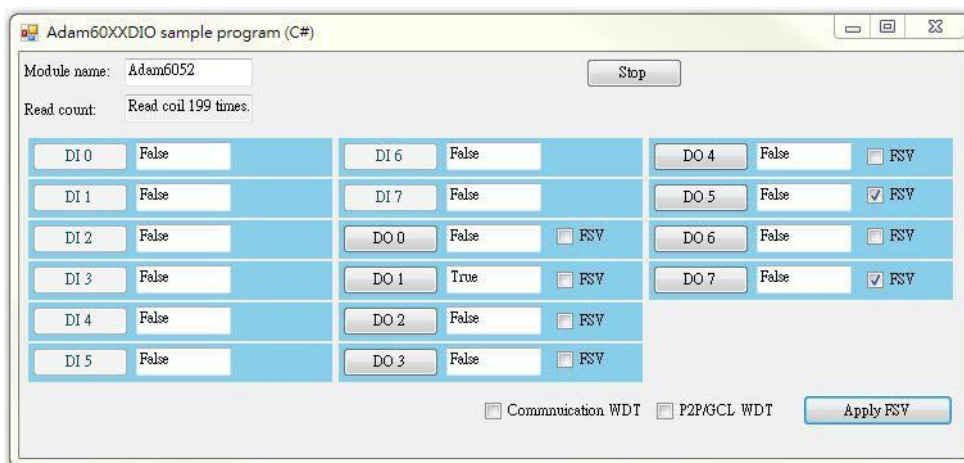


図 7.2 サンプルコードの実行と ADAM モジュールの設定

ADAM .NET CLASS LIBRARYには多くの関数が含まれています。これらの関数を理解するためのドキュメントが提供されています。これはスタートメニューのフォルダからアクセスできます。

## 7.3 ADAM-6000 モジュール用 Modbus プロトコル

ADAM-6000 モジュールは、コマンド・レスポンス形式でホスト PC と通信できます。データが送信されていないとき、モジュールはリスンモードになります。各モジュールには固有のアドレスが割り当てられます。システムにコマンドを発行する場合、ホストPCはこれらのアドレスを使用して特定のモジュールと通信し、応答を待ちます。何も検出されない場合、タイムアウトが発生し、シーケンスは中断され、制御はホストに戻されます。

この章の残りの部分では、関連する Modbus/TCP コマンドの構造について説明します。ADAM-6000 モジュールの Modbus/TCP アドレスの読み書きを支援するために、ADAM .NET CLASS LIBRARYにサンプル・コードが用意されています。WinCE と Win32 の例は以下のとおりです：

- Program Files\Advantech\AdamApax.NET Class Library\Sample Code\ADAM ADAM-6000モジュールの Modbus/TCPアドレスについては、付録 B.2 を参照してください。

### 7.3.1 Modbus プロトコルの構造

Modbus/TCPネットワーク上のModbusリクエストまたはレスポンスのカプセル化を理解することは重要です。完全なコマンドは“コマンド・ヘッド” (Modbus アプリケーション・プロトコル・ヘッド) と “コマンド・ボディ” (プロトコル・データ・ユニット) で構成されます。コマンド・ヘッドは6バイトで始まり、Modbusデータ・パケット・フォーマットに従います。コマンド・ボディはターゲット・デバイスと要求されたアクションを定義します。コマンド・ボディはターゲット・デバイスと要求されたアクションを定義します。次のセクションの例はこの理解するのに役立ちます。

### 7.3.2 Modbusファンクションコードの紹介

以下のファンクションコードは、プログラミングの必要条件を理解するための参考として提供されています：

Code (Hex)	Name	Usage
01	Read coil status	Read discrete output bit
02	Read input status	Read discrete input bit
03	Read holding registers	Read 16-bit register; used to read integer or
04	Read input registers	floating point process data
05	Force single coil	Write data to force coil On/Off
06	Preset single register	Write data in 16-bit format
08	Loopback diagnosis	Diagnostic testing of the communication port
15	Force multiple coils	Write multiple data to force coil On/Off
16	Preset multiple registers	Write data in 16-bit format

**機能コード 01**

ADAM-6000 モジュールのディスクリート出力の ON/OFF ステータスをバイナリ・フォーマットで読み取ります。

リクエストメッセージのフォーマット：

Command Body					
Station Address	Function Code	Start Address High Byte	Start Address Low Byte	Requested Number of Coil High Byte	Requested Number of Coil Low Byte

例ADAM-6000モジュールからコイル1~8（アドレス00017~00024）を読み出す。

01 01 00 17 00 08

応答メッセージの形式：

Command Body				
Station Address	Function Code	Byte Count	Data	Data

例コイル2~7はオン、その他はオフ。

01 01 01 42

レスポンスでは、コイル1~8のステータスはバイト値42（16進数）で示され、これはバイナリ・フォーマットでは0100 0010に相当する。

**ファンクションコード 02**

ADAM-6000 モジュールのディスクリート入力 ON/OFF ステータスをバイナリ・フォーマットで読み取る。

リクエストメッセージのフォーマット：

Command Body					
Station Address	Function Code	Start Address High Byte	Start Address Low Byte	Requested Number of Input High Byte	Requested Number of Input Low Byte

例ADAM-6000モジュールからコイル1~8（アドレス00001~00008）を読み出す。

01 02 00 01 00 08

応答メッセージの形式：

Command Body				
Station Address	Function Code	Byte Count	Data	Data

例入力2と3はオン、その他はオフ。

01 02 01 60

レスポンスでは、入力1~8のステータスがバイト値60（16進数）で示され、これはバイナリ・フォーマットでは0110 0000に相当する。

### ファンクションコード03と04

入力レジスタのバイナリ内容を読み出す。リクエスト・メッセージ・フォーマット：

Command Body					
Station Address	Function Code	Start Address High Byte	Start Address Low Byte	Requested Number of Register High Byte	Requested Number of Register Low Byte

例ADAM-6017 モジュールから浮動小数点値としてアドレス 40001 ~ 40002 のアナログ入力 1 と 2 を読み出す。

01 04 00 01 00 02

応答メッセージの形式：

Command Body				
Station Address	Function Code	Byte Count	Data	Data

例アナログ入力1=17096、アナログ入力2=0の生データ 入力レンジが0~10Vに設定されている場合、これらの入力の電圧は以下ようになる：アナログ入力1 =

$(17097/65535) * 10 V = 2.608 V$

アナログ入力 2 =  $(0/65535)*10 V = 0 V$

01 04 04 42 C8 00 00

### ファンクションコード 05

単一のコイルを強制的に ON または OFF にする。要求される ON/OFF 状態は、クエリ・データ・フィールドの定数で指定される。FF 00 (16進数) の値はONを要求し、00 00 (16進数) の値はOFFを要求し、FF FF (16進数) の値は強制値の要求する。

リクエストメッセージのフォーマット：

Command Body					
Station Address	Function Code	Coil Address High Byte	Coil Address Low Byte	Force Data High Byte	Force Data Low Byte

例ADAM-6000 モジュールのコイル 3 (アドレス 00003) を強制的にオンにします。

01 05 00 03 ff 00

応答メッセージの形式：

Command Body					
Station Address	Function Code	Coil Address High Byte	Coil Address Low Byte	Force Data High Byte	Force Data Low Byte

通常の応答は、コイル状態が強制された後に返される問い合わせのエコーである。

**ファンクションコード 06**

整数値を1つのレジスタにプリセットする。

リクエストメッセージのフォーマット：

Command Body					
Station Address	Function Code	Register Address High Byte	Register Address Low Byte	Preset Data High Byte	Preset Data Low Byte

例ADAM-6000 モジュールのプリセット・レジスタ 40002 を 00 04 (16 進) に設定する。

01 06 00 02 00 04

応答メッセージの形式：

Command Body					
Station Address	Function Code	Register Address High Byte	Register Address Low Byte	Preset Data High Byte	Preset Data Low Byte

通常の応答は、コイル状態がプリセットされた後に返される問い合わせのエコーである。

**機能コード 08**

受信した問い合わせメッセージをエコーする。メッセージは、データバッファの長さの半分から8バイトを引いた長さまでであれば、どのような長さでもよい。

リクエストメッセージのフォーマット：

Command Body		
Station Address	Function Code	Any data, length limited to approximately half the length of the data buffer

例：01 08 00 02 00 04

応答メッセージの形式：

Command Body		
Station Address	Function Code	Data bytes received

例：01 08 00 02 00 04

### ファンクションコード15 (16進数で「0F」)

一連のコイルの各コイルを強制的に ON または OFF 状態にする。リクエスト・メッセージのフォーマット :

#### Command Body

Station Address	Function Code	Start Address High Byte	Start Address Low Byte	Requested Number of Coil High Byte	Requested Number of Coil Low Byte	Byte Count	Force Data High Byte	Force Data Low Byte
-----------------	---------------	-------------------------	------------------------	------------------------------------	-----------------------------------	------------	----------------------	---------------------

例ADAM-6000 モジュールのアドレス 00017 (16 進数で "11") から始まる一連の 10 個のコイルを強制する要求。

01 0f 00 11 00 0a 02 cd 01

クエリデータの内容は2バイトである : CD01 (16進数)、バイナリフォーマットでは1100 1101 0000 0001に相当する。バイナリ・ビットは以下のようにアドレスにマッピングされる。

Bit:	1	1	0	0	1	1	0	1	0	0	0	0	0	0	0	1
Address (000XX):	24	23	22	21	20	19	18	17	-	-	-	-	-	-	26	25

応答メッセージの形式 :

#### Command Body

Station Address	Function Code	Start Address High Byte	Start Address Low Byte	Requested Number of Coil High Byte	Requested Number of Coil Low Byte
-----------------	---------------	-------------------------	------------------------	------------------------------------	-----------------------------------

正常な応答は、強制コイルのステーションアドレス、ファンクションコード、スタートアドレス、要求番号を返す。

例 : 01 0F 00 11 00 0A

### ファンクションコード16 (16進数で「10」)

保持レジスタのシーケンスにプリセット値を適用する。リクエスト・メッセージ・フォーマット :

#### Command Body

Station Address	Function Code	Start Address High Byte	Start Address Low Byte	Requested Number of Register High Byte	Requested Number of Register Low Byte	Byte Count	Data
-----------------	---------------	-------------------------	------------------------	--	---------------------------------------	------------	------

例ADAM-6000 モジュールの定数 1 (アドレス 40009) を 100.0 にプリセットします。01

10 00 09 00 02 04 42 c8 00 00

応答メッセージの形式 :

#### Command Body

Station Address	Function Code	Start Address High Byte	Start Address Low Byte	Requested Number of Register High Byte	Requested Number of Register Low Byte
-----------------	---------------	-------------------------	------------------------	--	---------------------------------------

正常な応答は、ステーション・アドレス、ファンクション・コード、スタート・アドレス、要求されたプリセット・レジスタ数を返す。

例 : 01 10 00 09 00 02

## 7.4 ADAM-6000モジュール用ASCIIコマンド

Modbus プロトコルに不慣れな方のために、アドバンテックはASCIIコマンドをModbus/TCP構造に統合するプロトコル・トランスレータとしてファンクション・ライブラリを提供しています。したがって、ASCIIコマンドに精通していれば、ADAM-6000モジュールにアクセスするために使用することができます。

### 7.4.1 ASCII構文

注：コマンドはすべて大文字で書かなければならない！

コマンドセットは以下の5つのカテゴリに分かれている：

- システム・コマンド・セット
- アナログ入力コマンドセット
- アナログ入力アラームコマンドセット
- ユニバーサルI/Oコマンドセット
- デジタルI/Oコマンドセット

これらのカテゴリは以下の文章にまとめられており、その後に個々のコマンドに関する続く。いくつかのコマンドは同じ形式を共有しているが、その効果はモジュールによって異なる。そのため、各モジュールの完全なコマンドセットを、そのコマンドが指定されたモジュールに与える影響の説明とともに列挙する。

### 7.4.2 システム・コマンド・セット

Command Syntax	Command Name	Description
\$aaM	Read module name	Returns the name of a specific module
\$aaF	Read firmware version	Returns the firmware version of a specific module
#aaVdbbbbd dddddd	Write GCL internal flag values	Writes one or more GCL internal flag values to a specific module
\$aaVd	Read GCL internal flag values	Returns all GCL internal flag values of a specific module
\$aa7	Get DO diagnostic status	Digital output diagnostic status 0=normal, 1=abnormal

注：\$aaMと\$aaFは、すべてのADAM-6000 I/Oモジュールでサポートされています。

#aaVdbbbbdは、ADAM-6050、ADAM-6051、ADAM-6052、ADAM-6060、ADAM-6066でサポートされている。

#aaVdはADAM-6050、ADAM-6051、ADAM-6052、ADAM-6060、ADAM-6066でサポートされている。

ADAM-6000 CEは、ここに記載されているコマンド以外にもサポートしています。詳細は付録Fを参照のこと。

## \$aaM

**Name** Read module name

**Description** Returns the name of a specific module

**Syntax** \$aaM(cr)  
\$ is a delimiter  
aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01)  
M refers to the read model number command  
(cr) is the terminating character, carriage return (0Dh)

**Response** !aa60bb(cr) if the command is valid  
?aa(cr) if an invalid command was entered  
There is no response if the module detects a syntax error, communication error, or if the address does not exist  
! is a delimiter indicating a valid command was received  
? is a delimiter indicating the command was invalid  
aa (range 00~FF) represents the 2-character hex slave address of the specified module  
bb (range 00~FF) represents the 2-character model number (hex) of the specified module  
(cr) is the terminating character, carriage return (0Dh)

**Example**           \$01M(cr)  
          com       !016050(cr)  
          mand:  
          response:

This command requests the module at Address 01h to return its model number. The module responds with "6050," indicating that the module is an ADAM-6050.

## \$aaF

**Name** Read firmware version

**Description** Returns the firmware version of a specific module

**Syntax** \$aaF(cr)  
\$ is a delimiter  
aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01)  
F refers to the read firmware version command  
(cr) is the terminating character, carriage return (0Dh)

**Response** !aa(version)(cr) if the command is valid  
?aa(cr) if an invalid command was entered  
There is no response if the module detects a syntax error, communication error, or if the address does not exist  
! is a delimiter indicating a valid command was received  
? is a delimiter indicating the command was invalid  
aa (range 00~FF) represents the 2-character hex slave address of the specified module  
(version) represents the firmware version of the module  
(cr) is the terminating character, carriage return (0Dh)

**Example**       command:       \$01F(cr)  
                  response:       !01 1.01(cr)

This command requests the module at Address 01h to return its firmware version. The response indicates that it is version 1.01.

**#aaVdbbbbddddd**

<b>Name</b>	Write GCL internal flag values (auxiliary flags)
<b>Description</b>	Writes one or more GCL internal flag values to a specific module (see Sections 8.3.1 and 8.3.4 for a definition of GCL internal flags)
<b>Syntax</b>	<pre>#aaVdbbbbddddd(cr) # is a delimiter aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01) Vd refers to the write GCL internal flag values command bbbb indicates which GCL internal flag(s) to set To write to all GCL internal flags: 0000 To write to a single GCL internal flag: First character is 1, and Characters 2~4 indicate the GCL internal flag number which can range from 0h to Fh ddddddd is the hex representation of the GCL internal flag value(s) Each character represents 4 GCL internal flag values (cr) is the terminating character, carriage return (0Dh)</pre>
<b>Response</b>	<pre>&gt;aa(cr) if the command is valid ?aa(cr) if an invalid command was entered There is no response if the module detects a syntax error, communication error, or if the address does not exist &gt; is a delimiter indicating a valid command was received ? is a delimiter indicating the command was invalid aa (range 00~FF) represents the 2-character hex slave address of the responding module (cr) is the terminating character, carriage return (0Dh)</pre>
<b>Example</b>	<pre>command: #01VDO0000000000000(cr) response: &gt;01(cr) This command sets all GCL internal flags (Flag 0~15) to logic low.</pre>
<b>Example</b>	<pre>command: #01VDO0000000FFFF(cr) response: &gt;01(cr) This command sets all GCL internal flags (Flag 0~15) to logic high.</pre>
<b>Example</b>	<pre>command: #01Vd1003000000001(cr) response: &gt;01(cr) This command sets one specific GCL internal flag (Flag 3) to logic high.</pre>
<b>Example</b>	<pre>command: #01Vd100E00000000(cr) Response: &gt;01(cr) This command sets one specific GCL internal flag (Flag 14) to logic low.</pre>

## \$aa7

Name	Get DO diagnostic status
Description Syntax	Get digital output diagnostic status 0=normal, 1=abnormal \$aa7(cr) \$ is a delimiter aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01) 7 refers to the Get DO diagnostic status command (cr) is the terminating character, carriage return (0Dh)
Response	!aa(DO0) (DO1)...(DO5) (cr) if the command is valid ?aa(cr) if an invalid command was entered There is no response if the module detects a syntax error, communication error, or if the address does not exist ! is a delimiter indicating a valid command was received ? is a delimiter indicating the command was invalid Aa (range 00~FF) represents the 2-character hex slave address of the specified module (DO0)(DO1)...(DO5) represents the firmware version of the module (cr) is the terminating character, carriage return (0Dh)
Example	command:                 \$017 Response:                !01101110

- DO0 = 1 (Abnormal)   DO1 = 0 (Abnormal)
- DO2 = 1 (Abnormal)   DO3 = 1 (Abnormal)
- DO4 = 1 (Abnormal)   DO5 = 0 (Normal)

**\$aaVd**

<b>Name</b>	Read GCL internal flag values (auxiliary flags)
<b>Description</b>	Reads all GCL internal flag values of a specific module (see Sections 8.3.1 and 8.3.4 for a definition of GCL internal flags)
<b>Syntax</b>	<p>\$aaVd(cr)          \$ is a delimiter          aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01)          Vd refers to the read GCL internal flag values command          (cr) is the terminating character, carriage return (0Dh)</p>
<b>Response</b>	<p>&gt;aaddddddd(cr) if the command is valid          ?aa(cr) if an invalid command was entered          There is no response if the module detects a syntax error, communication error, or if the address does not exist          &gt; is a delimiter indicating a valid command was received          ? is a delimiter indicating the command was invalid          aa (range 00~FF) represents the 2-character hex slave address of the specified module          dddddddd is the hex representation of the GCL internal flag value(s)          Each character represents 4 GCL internal flag values          (cr) is the terminating character, carriage return (0Dh)</p>
<b>Example</b>	<p>command:     \$01Vd (cr)          response:    &gt;010000002B(cr)</p> <p>This command reads all GCL internal flags values.          The character "B" (hex) = "1011" (binary) and the character "2" (hex) = "0010" (binary).          Therefore, GCL Internal Flags 0, 1, 3, 5 are logic high while the other GCL internal flags are logic low.</p>

### 7.4.3 アナログ入力コマンドセット

(ADAM-6015、6017、6018、6018+)

Command Syntax	Command Name	Description
#aan	Read single analog input	Returns the input value of a specific analog input channel
#aa	Read all analog inputs	Returns the input values of all analog input channels
\$aa0	Span calibration	Calibrates a module to correct for gain errors
\$aa1	Zero calibration	Calibrates a module to correct for offset errors
\$aa6	Read channel enable/disable statuses	Returns the enable/disable status of all analog input channels
\$aa5mm	Set all channel enable/disable statuses	Sets the enable/disable status of all analog input channels
#aaMH	Read all max. data values	Returns the max. data values of all analog input channels
#aaMHn	Read single max. data value	Returns the max. data value of a specific analog input channel
#aaML	Read all min. data values	Returns the min. data values of all analog input channels
#aaMLn	Read single min. data value	Returns the min. data of a specific analog input channel
#aaDnd	Set single digital output status	Sets the output status of a specific digital output channel
\$aaBnn	Read single analog input range code	Returns the input range code of a specific analog input channel
\$aaBRCnn	Read single analog input range code	Returns the input range code of a specific analog input channel (ADAM-6017 only)
\$aaAccrr	Write single analog input code	Writes the input range code to a specific analog input channel
\$aaAccrrrr	Write single analog input code	Writes the input range code to a specific analog input channel (ADAM-6017 only)

**#aan**

**Name** Read single analog input

**Description** Returns the input value of a specific analog input channel

**Syntax** #aan(cr)  
 # is a delimiter  
 aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01)  
 n (range 0~8) refers to the specific channel you want to read the input data from  
 (cr) is the terminating character, carriage return (0Dh)

**Response** >(data)(cr) if the command is valid  
 ?aa(cr) if an invalid command was entered  
 There is no response if the module detects a syntax error, communication error, or if the address does not exist  
 > is a delimiter indicating a valid command was received  
 ? is a delimiter indicating the command was invalid  
 (cr) is the terminating character, carriage return (0Dh)

**Example** command: #012(cr)  
 response: >+10.000  
 Channel 2 of the ADAM-6000 analog module at Address 01h responds with an input value of +10.000.

**#aa**

**Name** Read all analog inputs

**Description** Returns the input data of all analog input channels

**Syntax** #aa(cr)  
 # is a delimiter  
 aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01)  
 (cr) is the terminating character, carriage return (0Dh)

**Response** >(data)(data)(data)(data)(data)(data)(data)(data)(data)(cr) if the command is valid  
 ?aa(cr) if an invalid command was entered  
 There is no response if the module detects a syntax error, communication error, or if the address does not exist  
 > is a delimiter indicating a valid command was received  
 ? is a delimiter indicating the command was invalid  
 (cr) is the terminating character, carriage return (0Dh)

**Note:** The last data returned is the average value of the preset channels in this module

**Example** command: #01(cr)  
 response: >+10.000+10.000+10.000+10.000+10.000+10.000+10.000+10.000+10.000+10.000

## \$aa0

<b>Name</b>	Span calibration
<b>Description</b>	Calibrates a module to correct for gain errors
<b>Syntax</b>	\$aa0(cr) \$ is a delimiter aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01) 0 refers to the span or auto calibration command (cr) is the terminating character, carriage return (0Dh)
<b>Response</b>	!aa(cr) if the command is valid ?aa(cr) if an invalid command was entered There is no response if the module detects a syntax error, communication error, or if the address does not exist ! is a delimiter indicating a valid command was received ? is a delimiter indicating the command was invalid aa (range 00~FF) is the 2-character hex slave address of the specified module (cr) is the terminating character, carriage return (0Dh)
<b>Note:</b>	To successfully calibrate an analog input module's input range, a reliable calibration input signal should be connected to the analog input module before and during the calibration.

## \$aa1

<b>Name</b>	Zero calibration
<b>Description</b>	Calibrates a module to correct for offset errors
<b>Syntax</b>	\$aa1(cr) \$ is a delimiter aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01) 1 refers to the offset or auto calibration command (cr) is the terminating character, carriage return (0Dh)
<b>Response</b>	!aa(cr) if the command is valid ?aa(cr) if an invalid command was entered There is no response if the module detects a syntax error, communication error, or if the address does not exist ! is a delimiter indicating a valid command was received ? is a delimiter indicating the command was invalid aa (range 00~FF) represents the 2-character hex slave address of the specified module (cr) is the terminating character, carriage return (0Dh)
<b>Note:</b>	To successfully calibrate an analog input module's input range, a reliable calibration input signal should be connected to the analog input module before and during the calibration.

**\$aa6**

<b>Name</b>	Read all channel enable/disable statuses
<b>Description</b>	Returns the enable/disable status of all analog input channels
<b>Syntax</b>	\$aa6(cr) \$ is a delimiter aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01) 6 refers to the read channel enable/disable statuses command (cr) is the terminating character, carriage return (0Dh)
<b>Response</b>	!aamm(cr) if the command is valid ?aa(cr) if an invalid command was entered There is no response if the module detects a syntax error, communication error or if the address does not exist ! is a delimiter indicating a valid command was received ? is a delimiter indicating the command was invalid aa (range 00~FF) is the 2-character hex slave address of the specified module mm are two hexadecimal values (each value is interpreted as 4 bits). The first 4-bit value represents the status of channels 7-4 and the second value represents the status of Channels 3-0. A value of 0 means the channel is disabled, while a value of 1 means the channel is enabled. (cr) is the terminating character, carriage return (0Dh)
<b>Example</b>	command: \$016(cr) response: !01FF(cr)  This command requests the module at Address 01h to return the enable/disable status of all analog input channels. The module responds with "FF" (hex), which is equivalent to "1111" (binary) and "1111" (binary), meaning that all channels are enabled.

**\$aa5mm**

<b>Name</b>	Set all channel enable/disable statuses
<b>Description</b>	Sets the enable/disable status of all analog input channels
<b>Syntax</b>	\$aa5mm(cr) \$ is a delimiter aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01) 5 refers to the set all enable/disable statuses command mm are two hexadecimal values (each value is interpreted as 4 bits). The first 4-bit value represents the status of channels 7-4 and the second value represents the status of Channels 3-0. A value of 0 means the channel is disabled, while a value of 1 means the channel is enabled. (cr) is the terminating character, carriage return (0Dh)
<b>Response</b>	!aa(cr) if the command is valid ?aa(cr) if an invalid command was entered There is no response if the module detects a syntax error, communication error, or if the address does not exist ! is a delimiter indicating a valid command was received ? is a delimiter indicating the command was invalid aa (range 00~FF) is the 2-character hex slave address of the specified module (cr) is the terminating character, carriage return (0Dh)



**#aaMHn**

<b>Name</b>	Read single max. data value
<b>Description</b>	Returns the max. data value of a specific analog input channel
<b>Syntax</b>	#aaMHn(cr) # is a delimiter aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01) MH refers to the read single max. data value command n (range 0-8) represents the specified channel you want to read the input data from (cr) is the terminating character, carriage return (0Dh)
<b>Response</b>	>(data)(cr) if the command is valid ?aa(cr) if an invalid command was entered There is no response if the module detects a syntax error, communication error, or if the address does not exist > is a delimiter indicating a valid command was received ? is a delimiter indicating the command was invalid aa (range 00~FF) represents the 2-character hex slave address of the specified module (cr) is the terminating character, carriage return (0Dh)
<b>Example</b>	command: #01MH2(cr) response: >+10.000  This command requests the module at Address 01h to return the historic max. value from Analog Input Channel 2.

**#aaML**

<b>Name</b>	Read all min. data values
<b>Description</b>	Returns the min. data values of all analog input channels
<b>Syntax</b>	#aaML(cr) # is a delimiter aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01) ML refers to the read all min. data values command (cr) is the terminating character, carriage return (0Dh)
<b>Response</b>	>(data)(data)(data)(data)(data)(data)(data)(data)(data)(data)(cr) if the command is valid ?aa(cr) if an invalid command was entered There is no response if the module detects a syntax error, communication error, or if the address does not exist > is a delimiter indicating a valid command was received ? is a delimiter indicating the command was invalid aa (range 00~FF) represents the 2-character hex slave address of the specified module (cr) is the terminating character, carriage return (0Dh)
<b>Example</b>	command: #01ML(cr) response: >+10.000+10.000+10.000+10.000+10.000+10.000+10.000+10.000+10.000  This command requests the module at Address 01h to return the historic min. value from each analog input channel.
<b>Note:</b>	The last data returned is the average value of all channels.

## #aaMLn

**Name** Read single min. data value

**Description** Reads the min. data value of a specific analog input channel

**Syntax** #aaMLn(cr)  
# is a delimiter  
aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01)  
ML refers to the read single min. data value command  
n (range 0-8) represents the specified channel you want to read the input data from  
(cr) is the terminating character, carriage return (0Dh)

**Response** >(data)(cr) if the command is valid  
?aa(cr) if an invalid command was entered  
There is no response if the module detects a syntax error, communication error, or if the address does not exist  
> is a delimiter indicating a valid command was received  
? is a delimiter indicating the command was invalid  
aa (range 00~FF) is the 2-character hex slave address of the specified module  
(cr) is the terminating character, carriage return (0Dh)

**Example** command: #01ML3(cr)  
response: >+10.000

This command requests the module at Address 01h to return the historic min. value from Analog Input Channel 3.

## #aaDnd

**Name** Set single digital output status

**Description** Sets the output status of a specific digital output channel

**Syntax** #aaDnd(cr)  
# is a delimiter  
aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01)  
D refers to the set digital output statuses command  
n (range 0-1) represents the specific channel you want to set the output status of  
d (range 0-1) represents the status you want to set to the specified channel  
(cr) is the terminating character, carriage return (0Dh)

**Response** !aa(cr) if the command is valid  
?aa(cr) if an invalid command was entered  
There is no response if the module detects a syntax error, communication error, or if the address does not exist  
! is a delimiter indicating a valid command was received  
? is a delimiter indicating the command was invalid  
aa (range 00~FF) is the 2-character hex slave address of the specified module  
(cr) is the terminating character, carriage return (0Dh)

**Example** command: #01DO1(cr)  
response: !01

This command set Digital Output Channel 0 to "ON" for the module at Address 01h.

**\$aaBnn**

<b>Name</b>	Read single analog input range code
<b>Description</b>	Returns the range code of a specific analog input channel
<b>Syntax</b>	<p>\$aaBnn(cr)          \$ is a delimiter          aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01)          B refers to the read single analog input range code command          nn (range 00~07) is the channel you want to read the range code from          (cr) is the terminating character, carriage return (0Dh)</p>
<b>Response</b>	<p>!aa(data)(code) if the command is valid          ?aa(cr) if an invalid command was entered          There is no response if the module detects a syntax error, communication error, or if the address does not exist          ! is a delimiter indicating a valid command was received          ? is a delimiter indicating the command was invalid          (cr) is the terminating character, carriage return (0Dh)          (code) is the range code (refer to the following tables)</p>

**ADAM-6017-BE アナログ入力チャンネルレンジコード**

Range Code (Hex)	Range Code (Decimal)	Range Description
08	8	+/-10 V
09	9	+/-5V
0A	10	+/-1V
0B	11	+/-500 mV
0C	12	+/-150 mV
0D	13	0~20 mA
07	7	4~20 mA
0x0148*	-	0~10 V
0x0147*	-	0~5 V
0x0145*	-	0~1 V
0x0106*	-	0~500 mV
0x0105*	-	0~150 mV
0x0181*	-	+/-20 mA

\* これらのレンジコードは、ADAM-6017-CE によってのみサポートされます。

**注意** ADAM-6017-CE の単一アナログ入力レンジ・コードを読み取るには、*\$aaBRCnn* を使用します。



### ADAM-6015 アナログ入力チャンネル範囲コード

Range Code (Hex)	Range Code (Decimal)	Range Description
20	32	Pt 100 (a=0.00385) -50~150°C
21	33	Pt 100 (a=0.00385) 0~100°C
22	34	Pt 100 (a=0.00385) 0~200°C
23	35	Pt 100 (a=0.00385) 0~400°C
24	36	Pt 100 (a=0.00385) -200~200°C
25	37	Pt 100 (a=0.00392) -50~150°C
26	38	Pt 100 (a=0.00392) 0~100°C
27	39	Pt 100 (a=0.00392) 0~200°C
28	40	Pt 100 (a=0.00392) 0~400°C
29	41	Pt 100 (a=0.00392) -200~200°C
2A	42	Pt 1000 -40~160°C
2B	43	Balco 500 -30~120°C
2C	44	Ni 518 -80~100°C
2D	45	Ni 518 0~100°C

### ADAM-6018/6018+ アナログ入力チャンネル範囲コード

Range Code (Hex)	Range Code (Decimal)	Range Description
0E	14	Thermocouple J (0~760°C)
0F	15	Thermocouple K (0~1370°C)
10	16	Thermocouple T (-100~400°C)
11	17	Thermocouple E (0~1000°C)
12	18	Thermocouple R (500~1750°C)
13	19	Thermocouple S (500~1750°C)
14	20	Thermocouple B (500~1800°C)

### \$aaBRCnn

<b>Name</b>	Read single analog input range code
<b>Description</b>	Returns the range code of a specific analog input channel (ADAM-6017 only)
<b>Syntax</b>	<p>\$aaBRCnn</p> <p>\$ is a delimiter</p> <p>aa (range 00~FF) represents the 2-character hex slave address of the specified module (always 01)</p> <p>B refers to the read single analog input range code command</p> <p>RC is the range code</p> <p>nn (range 00-07) is the channel you want to read the range code from</p>
<b>Response</b>	<p>!aa(data)(code) if the command is valid</p> <p>?aa(cr) indicates that an invalid command was entered</p> <p>There is no response if the module detects a syntax error, communication error, or if the address does not exist</p> <p>! is a delimiter indicating that a valid command was received</p> <p>? is a delimiter indicating that the command was invalid</p> <p>(cr) is a terminating character, carriage return (0Dh)</p> <p>(code) is the range code (refer to the following tables)</p>

## ADAM-6017-CE &amp; D アナログ入力チャンネル範囲コード

Range Code (Hex)	Range Description
0x0143	+/-10 V
0x0142	+/-5 V
0x0140	+/-1 V
0x0104	+/-500 mV
0x0103	+/-150 mV
0x0148	0~10 V
0x0147	0~5 V
0x0145	0~1 V
0x0106	0~500 mV
0x0105	0~150 mV
0x0182	0~20 mA
0x0180	4~20 mA
0x0181	+/-20 mA

**\$aaAccrr/\$aaAccrrrr**

<b>Name</b>	Write single analog input range code
<b>Description</b>	Writes the analog input range code to a specific analog input channel and responds whether the setting is successful
<b>Syntax</b>	<p>\$aaAccrr/\$aaAccrrrr</p> <p>\$ is a delimiter</p> <p>aa (range 00~FF) represents the 2-character hex slave address of the specified module (always 01)</p> <p>A refers to the write single analog input range code command</p> <p>Cc is the channel you want to set</p> <p>rr/rrrr is the specified channel you want to write the range code to</p>
<b>Response</b>	<p>?aa(cr) indicates that an invalid command was entered</p> <p>There is no response if the module detects a syntax error, communication error, or if the address does not exist</p> <p>! is a delimiter indicating that a valid command was received</p> <p>? is a delimiter indicating that the command was invalid</p> <p>(cr) is the terminating character, carriage return (0Dh)</p> <p>(code) is the range code (refer to the following tables)</p> <p>Note: \$aaAccrrrr is only for the ADAM-6017-CE</p>

## ADAM-6017 アナログ入力レンジ

Range Description	CE & D Version Range Code	AE & BE Version Range Code
+/-10 V	0x0143	08
+/-5 V	0x0142	09
+/-1 V	0x0140	0A
+/-500 mV	0x0104	0B
+/-150 mV	0x0103	0C
0~10 V	0x0148	
0~5 V	0x0147	
0~1 V	0x0145	
0~500 mV	0x0106	
0~150 mV	0x0105	
0~20 mA	0x0182	0D
4~20 mA	0x0180	07
+/-20 mA	0x0181	

### 注



1. ADAM-6017-AEおよびADAM-6017-BEバージョンのレンジコードは、ADAM-6017-CEにも適用できます。ただし、弊社モジュールの新しいレンジコードを使用することをお勧めします。

2. ADAM-6015 については、上記の ADAM-6018 の表を参照のこと。

<b>Example</b>	command:	\$01A010147(cr) Channel 1 is set to 0x0147, which means "0~5 V"
		!010B(cr) - setting successful
	response:	!010147(cr) - setting successful
		?01 - setting fail
<b>Example</b>	command:	\$01A0109(cr) channel 1 set to 9, which means "+/-5 V"
		!01(cr) - setting successful
	response:	?01-setting fail
		?01AVG(cr) - setting fail, because the channel is set to "ON" of the average channel

## 7.4.4 アナログ入力アラームコマンドセット

(ADAM-6015、6017、6018、6018+)

Command Syntax	Command Name	Description
\$aaCjAhs	Set alarm mode	Sets the high/low alarm of a specific channel to either momentary or latching mode
\$aaCjAh	Read alarm mode	Returns the alarm mode of a specific channel
\$aaCjAhEs	Enable/disable high/low alarm	Enables/disables the high/low alarm of a specific channel
\$aaCjCh	Clear latch alarm	Resets a latched alarm for a specific channel
\$aaCjAhCCn	Set alarm connection (for ADAM-6018/6017)	Connects the high/low alarm of a specific analog input channel to interlock with a specific digital output channel
\$aaCjRhC	Read alarm connection (for ADAM-6017/6018)	Returns the alarm configuration of a specific channel
\$aaCjAhU	Set alarm limit	Sets the high/low alarm limit value of a specific channel
\$aaCjRhU	Read alarm limit	Returns the high/low alarm limit value of a specific channel
\$aaCjS	Read alarm status	Returns the alarm status of a specific channel

### \$aaCjAhs

**Name** Set alarm mode

**Description** Sets the high/low alarm of a specific channel to either latching or momentary mode

**Syntax** \$aaCjAhs(cr)  
 \$ is a delimiter  
 aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01)  
 Cj specifies the channel j (j: 0~7)  
 A refers to the set alarm mode command  
 h indicates the alarm type (H = high alarm, L = low alarm)  
 s indicates the alarm mode (M = momentary mode, L = latching mode)  
 (cr) is the terminating character, carriage return (0Dh)

**Response** !aa(cr) if the command is valid  
 ?aa(cr) if an invalid command was entered  
 There is no response if the system detects a syntax error, communication error, or if the address does not exist  
 ! is a delimiter indicating a valid command was received  
 aa is the 2-character hexadecimal slave address of the specified module  
 (cr) is the terminating character, carriage return (0Dh)

**Example** command: \$01C1AHL(cr)  
 response: !01(cr)

Channel 1 of the module at Address 01h is instructed to set its high alarm to latching mode. The module confirms that the command has been received.

## \$aaCjAh

**Name** Read alarm mode

**Description** Returns the alarm mode of a specific channel

**Syntax** \$aaCjAh(cr)  
\$ is a delimiter  
aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01)  
Cj specifies the channel j (j: 0~7)  
A refers to the read alarm mode command  
h indicates the alarm type (H = high alarm, L = low alarm)  
(cr) is the terminating character, carriage return (0Dh)

**Response** !aa(cr) if the command is valid  
?aa(cr) if an invalid command was entered  
There is no response if the system detects a syntax error, communication error, or if the address does not exist  
! is a delimiter indicating a valid command was received  
aa is the 2-character hexadecimal slave address of the specified module  
s indicates the alarm mode (M = momentary mode, L = latching mode)  
(cr) is the terminating character, carriage return (0Dh)

**Example** command: \$01C1AL(cr)  
response: !01M(cr)  
Channel 1 of the module at Address 01h is instructed to return its low alarm mode. The module responds that it is in momentary mode.

## \$aaCjAhEs

**Name** Enable/disable high/low alarm

**Description** Enables/disables the high/low alarm of a specific channel

**Syntax** \$aaCjAhEs(cr)  
\$ is a delimiter  
aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01)  
Cj specifies the channel j (j: 0~7)  
AhEs refers to the enable/disable high/low alarm command  
h indicates the alarm type (H = high alarm, L = low alarm)  
s indicates the alarm enable/disable status (E = enable, D = disable)  
(cr) is the terminating character, carriage return (0Dh)

**Response** !aa(cr) if the command was valid  
?aa(cr) if an invalid command is entered  
There is no response if the system detects a syntax error, communication error, or if the address does not exist  
! is a delimiter indicating a valid command was received  
aa is the 2-character hexadecimal slave address of the specified module  
(cr) is the terminating character, carriage return (0Dh)

**Example** command: \$01C1ALEE(cr)  
response: !01(cr)  
Channel 1 of the module at Address 01h is instructed to enable its low alarm function. The module confirms that its low alarm function has been enabled.

**Note:** An analog input module requires up to 2 s after it receives an enable/disable alarm command for the setting take effect.

**\$aaCjCh**

<b>Name</b>	Clear latch alarm
<b>Description</b>	Resets a latched alarm for a specific channel
<b>Syntax</b>	<p>\$aaCjCh(cr)          \$ is a delimiter          aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01)          Cj specifies the channel j (j: 0~7)          Ch refers to the clear latch alarm command          h indicates the alarm type (H = high alarm, L = low alarm)          (cr) is the terminating character, carriage return (0Dh)</p>
<b>Response</b>	<p>!aa(cr) if the command is valid          ?aa(cr) if an invalid command was entered          There is no response if the system detects a syntax error, communication error, or if the address does not exist          ! is a delimiter indicating a valid command was received          aa is the 2-character hexadecimal slave address of the specified module          (cr) is the terminating character, carriage return (0Dh)</p>
<b>Example</b>	<p>command:     \$01C1CL(cr)          response:    !01(cr)</p> <p>Channel 1 of the module at Address 01h is instructed to set its low alarm state to OFF. The module confirms that it has done so.</p>

**\$aaCjAhCCn**

<b>Name</b>	Set alarm connection
<b>Description</b>	Connects the high/low alarm of a specific analog input channel to interlock with a specific digital output channel
<b>Syntax</b>	<p>\$aaCjAhCCn(cr)          \$ is a delimiter          aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01)          Cj specifies the analog input channel j (j: 0~7)          AhC refers to the set alarm connection command          h indicates the alarm type (H = high alarm, L = low alarm)          Cn specifies the digital output channel n (n: 0~1)          To disconnect the digital output, n should be set as ??          (cr) is the terminating character, carriage return (0Dh)</p>
<b>Response</b>	<p>!aa(cr) if the command is valid          ?aa(cr) if an invalid command was entered          There is no response if the system detects a syntax error, communication error, or if the address does not exist          ! is a delimiter indicating a valid command was received          aa is the 2-character hexadecimal slave address of the specified module          (cr) is the terminating character, carriage return (0Dh)</p>
<b>Example</b>	<p>command:     \$01C1ALCC0(cr)          response:    !01(cr)</p> <p>Channel 1 of the module at Address 01h is instructed to connect its low alarm to the digital output of Channel 0. The system confirms that it has done so.</p>

## \$aaCjRhC

<b>Name</b>	Read alarm connection
<b>Description</b>	Returns the high/low alarm limit value of a specific channel
<b>Syntax</b>	<code>\$aaCjRhC(cr)</code> \$ is a delimiter aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01) Cj specifies the analog input channel j (j: 0~7) RhC refers to the read alarm connection command h indicates the alarm type (H = high alarm, L = low alarm) (cr) is the terminating character, carriage return (0Dh)
<b>Response</b>	!aaCn(cr) if the command is valid ?aa(cr) if an invalid command was entered There is no response if the system detects a syntax error, communication error, or if the address does not exist ! is a delimiter indicating a valid command was received aa is the 2-character hexadecimal slave address of the specified module Cn identifies whether the desired digital output channel n (n: 0~1) interlocks with the alarm of the specific analog input channel. If the values of n are "*", then the analog input has no connection with the digital output point. (cr) is the terminating character, carriage return (0Dh)
<b>Example</b>	command: <code>\$01C1RLC(cr)</code> response: <code>!01C0(cr)</code> Channel 1 of the module at Address 01h is instructed to return its low alarm output connection. The system responds that the low alarm output connects to the digital output at Channel 0.

## \$aaCjAhU

<b>Name</b>	Set alarm limit
<b>Description</b>	Sets the high/low alarm limit value of a specific channel
<b>Syntax</b>	<code>\$aaCjAhU(data)(cr)</code> \$ is a delimiter aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01) Cj specifies the analog input channel j (j: 0~7) AhU refers to the set alarm limit command h indicates the alarm type (H = high alarm, L = low alarm) (data) specifies the alarm limit setting The format is always in engineering units (cr) is the terminating character, carriage return (0Dh)
<b>Response</b>	!aa(cr) if the command is valid ?aa(cr) if an invalid command was entered There is no response if the system detects a syntax error, communication error, or if the address does not exist ! is a delimiter indicating a valid command was received aa is the 2-character hexadecimal slave address of the specified module (cr) represents terminating character, carriage return (0Dh)
<b>Example</b>	command: <code>\$01C1AHU+080.00(cr)</code> response: <code>!01(cr)</code> The high alarm limit of Channel 1 in the module at Address 01h has been set +80. The system confirms that the command has been received.
<b>Note:</b>	An analog input module requires up to 2 s after receiving a set alarm limit command for the settings to take effect.

**\$aaCjRhU****Name** Read alarm limit**Description** Returns the high/low alarm limit value of a specific channel

**Syntax** \$aaCjRhU(cr)  
 \$ is a delimiter  
 aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01)  
 Cj specifies the analog input channel j (j: 0~7)  
 RhU refers to the read alarm limit command  
 h indicates the alarm type (H = high alarm, L = low alarm)  
 (cr) is the terminating character, carriage return (0Dh)

**Response** !aa(data)(cr) if the command is valid  
 ?aa(cr) if an invalid command was entered  
 There is no response if the system detects a syntax error, communication error, or if the address does not exist  
 ! is a delimiter indicating a valid command was received  
 aa is the 2-character hexadecimal slave address of the specified module  
 (data) represents the alarm limit setting (in engineering units)  
 (cr) is the terminating character, carriage return (0Dh)

**Example** command: \$01C1RHU(cr)  
 response: !01+2.0500(cr)  
 Channel 1 of the module at Address 01h has been configured to accept a 5 V input. The command instructs the system to return the high alarm limit value for that channel. The system responds that the high alarm limit value for this channel is 2.0500 V.

**\$aaCjS****Name** Read alarm status**Description** Returns the alarm status of a specific channel

**Syntax** \$aaCjS(cr)  
 \$ is a delimiter  
 aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01)  
 Cj specifies the analog input channel j (j: 0~7)  
 S refers to the read alarm status command  
 (cr) is the terminating character, carriage return (0Dh)

**Response** !aahl(cr) if the command is valid  
 ?aa(cr) if an invalid command was entered  
 There is no response if the system detects a syntax error, communication error, or if the address does not exist  
 ! is a delimiter indicating that a valid command was received  
 aa is the 2-character hexadecimal slave address of the specified module  
 h represents the high alarm status, where "1" = the high alarm has been triggered, "0" = the high alarm has not been triggered  
 l represents the low alarm status, where "1" = the low alarm has been triggered, "0" = the low alarm has not been triggered  
 (cr) is the terminating character, carriage return (0Dh)

**Example** command: \$01C1S(cr)  
 response: !0101(cr)  
 The command requests the module at Address 01h to return its alarm status for Channel 1. The system responds that a high alarm has not been triggered, but the low alarm has been triggered.

## 7.4.5 ユニバーサルI/Oコマンドセット

(ADAM-6024)

Command Syntax	Command Name	Description
\$aa5mm	Set all enable/disable statuses	Sets the enable/disable status of all analog input channels
\$aa6	Read all enable/disable statuses	Returns the enable/disable status of all analog input channels
#aa	Read all analog input values	Returns the input values of all analog input channels
#aacc	Read single analog input value	Returns the input value of a specific analog input channel
\$aaDcc	Read single analog output startup value	Returns the startup output value of a specific analog output channel
\$aaDcchhh	Set single analog output startup value	Sets the startup output value of a specific analog output channel
#aacdd.ddd	Write single analog output value	Writes a value to a specific analog output channel
#aa7	Read all digital input statuses	Returns the statuses of all digital input channels
#aacdd	Write digital output values	Writes a value to one or all digital output channels
\$aa2	Read digital output values	Reads the value of all digital output channels
\$aaBcc	Read single analog input range code	Returns the channel range code of a specific analog input channel
\$aaCcc	Read single analog output range code	Returns the channel range code from the specified analog output channel

### \$aa5mm

**Name** Set all enable/disable statuses

**Description** Sets the enable/disable statuses of all analog input channels

**Syntax** \$aa5mm(cr)  
 \$ is a delimiter  
 aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01)  
 5 refers to the set all enable/disable statuses command  
 mm (range 00~FF) are two hexadecimal characters (each character is interpreted as 4 bits). The first 4-bit value represents the status of Channels 5~4; the second 4-bit value represents the status of Channels 3~0. A value of 0 means that the channel is disabled, whereas a value of 1 means that the channel is enabled.  
 (cr) is the terminating character, carriage return (0Dh)

**Response** !aa(cr) if the command is valid  
 ?aa(cr) if an invalid command was entered  
 There is no response if the module detects a syntax error, communication error, or if the address does not exist  
 ! is a delimiter indicating a valid command was received  
 ? is a delimiter indicating the command was invalid  
 aa (range 00~FF) is the 2-character hex slave address of the specified module  
 (cr) is the terminating character, carriage return (0Dh)

**Example**      command:     \$01521(cr)  
                  response:    !01(cr)  
 The command enables/disables all analog input channels of the module at Address 01h. A value of "2" (hex) = "0010" (binary), which enables Channel 5 and disables Channel 4; a value of 1 (hex) = "0001" (binary), which disables Channels 1~3 and enables Channel 0.

## \$aa6

**Name**            Read all enable/disable statuses

**Description** Returns the enable/disable statuses of all analog input channels

**Syntax**            \$aa6(cr)  
 \$ is a delimiter  
 aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01)  
 6 refers to the read all enable/disable statuses command  
 (cr) is the terminating character, carriage return (0Dh)

**Response**        !aamm(cr) if the command is valid  
 ?aa(cr) if an invalid command was entered  
 There is no response if the module detects a syntax error, communication error, or if the address does not exist  
 ! is a delimiter indicating a valid command was received  
 ? is a delimiter indicating the command was invalid  
 aa (range 00~FF) is the 2-character hex slave address of the specified module  
 mm are two hex values (each value is interpreted as 4 bits). The first 4-bit value represents the status of Channels 7~4 and the second value represents the status of Channels 3~0. A value of 0 means that the channel is disabled, whereas a value of 1 means that the channel is enabled.  
 (cr) is the terminating character, carriage return (0Dh)

**Example**        command:     \$016(cr)  
                  response:    !013F(cr)  
 The command requests the module at Address 01h to return the enable/disable statuses of all analog input channels. The value "3F" (hex) = "0011" and "1111" (binary), meaning that Channels 0~5 are all enabled.

## #aa

**Name**            Read all analog input values

**Description** Returns the input values of all analog input channels

**Syntax**            #aa(cr)  
 # is a delimiter  
 aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01)  
 (cr) is the terminating character, carriage return (0Dh)

**Response**        >(data)(data)(data)(data)(data)(data)(cr) if the command is valid  
 ?aa(cr) if an invalid command was entered  
 There is no response if the module detects a syntax error, communication error, or if the address does not exist  
 > is a delimiter indicating a valid command was received  
 ? is a delimiter indicating the command was invalid  
 (cr) is the terminating character, carriage return (0Dh)

**Example**        command:     #01(cr)  
                  response:    >+10.000+10.000+10.000+10.000+10.000+10.000

## #aacc

**Name** Read single analog input value

**Description** Returns the input value of a specific analog input channel

**Syntax** #aacc(cr)  
# is a delimiter  
aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01)  
cc (range 00-05) specifies the channel you want to read the input data from  
(cr) is the terminating character, carriage return (0Dh)

**Response** >(data)(cr) if the command is valid.  
?aa(cr) if an invalid command is entered  
There is no response if the module detects a syntax error, communication error, or if the address does not exist  
> is a delimiter indicating a valid command was received  
? is a delimiter indicating the command was invalid  
(cr) is the terminating character, carriage return (0Dh)

**Example** command: #0103(cr)  
response: >+10.000  
Analog Input Channel 3 of the module at Address 01h responds with an input value of +10.000.

## \$aaDcc

**Name** Read single analog output startup value

**Description** Returns the startup value of a specific analog output channel

**Syntax** \$aaDcc(cr)  
\$ is a delimiter  
aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01)  
D refers to the read single analog output startup value command  
cc (range 00-01) specifies the channel you want to read the startup value from  
(cr) is the terminating character, carriage return (0Dh)

**Response** !aahhh(cr) if the command is valid  
?aa(cr) if an invalid command was entered  
There is no response if the module detects a syntax error, communication error, or if the address does not exist  
! is a delimiter indicating a valid command was received  
? is a delimiter indicating the command was invalid  
aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01)  
hhh (range 000~FFF) is the 3-character hex startup value of the specified analog output channel  
(cr) is the terminating character, carriage return (0Dh)

**Example** command: \$01D01(cr)  
response: !01FFF(cr)  
Analog Output Channel 1 of the module at Address 01h responds with a startup value of +10.000 (i.e., the analog output range of Channel 1 is 0~10 V).

**\$aaDcchhh**

**Name** Set single analog output startup value

**Description** Sets the startup value of a specific analog output channel

**Syntax** \$aaDcchhh(cr)  
 \$ is a delimiter  
 aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01)  
 D refers to the set single analog output startup value command  
 cc (range 00-01) specifies the channel you want to set the startup value of  
 hhh (range 000~FFF) is the 3-character hex startup value of the specified analog output channel  
 (cr) is the terminating character, carriage return (0Dh)

**Response** !aa(cr) if the command is valid  
 ?aa(cr) if an invalid command was entered  
 There is no response if the module detects a syntax error, communication error, or if the address does not exist  
 ! is a delimiter indicating a valid command was received  
 ? is a delimiter indicating the command was invalid  
 aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01)  
 (cr) is the terminating character, carriage return (0Dh)

**Example** command: \$01D01FFF(cr)  
 response: !01(cr)  
 The startup value of Analog Output Channel 1 of the module at Address 01h is set to +10.000 (i.e., the analog output range of Channel 1 is 0~10 V).

**#aacdd.ddd**

**Name** Write single analog output value

**Description** Writes a value to a specific analog output channel

**Syntax** #aacdd.ddd(cr)  
 # is a delimiter  
 aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01)  
 cc (range 00-01) specifies the channel you want to write the output value to  
 dd.ddd (in engineering unit) is the analog output value of the specified analog output channel  
 (cr) is the terminating character, carriage return (0Dh)

**Response** >(cr) if the command is valid  
 ?aa(cr) if an invalid command was entered  
 There is no response if the module detects a syntax error, communication error, or if the address does not exist  
 > is a delimiter indicating a valid command was received  
 ? is a delimiter indicating the command was invalid  
 aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01)  
 (cr) is the terminating character, carriage return (0Dh)

**Example** command: #010105.555(cr)  
 response: >(cr)  
 The value of Analog Output Channel 1 of the module at Address 01h is set to +05.555.

## \$aa7

**Name** Read all digital input statuses

**Description** Returns the values of all digital input channels

**Syntax** \$aa7(cr)  
\$ is a delimiter  
aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01)  
7 refers to the read all digital input statuses command  
(cr) is the terminating character, carriage return (0Dh)

**Response** !aa(data)(cr) if the command is valid  
?aa(cr) if an invalid command was entered  
There is no response if the module detects a syntax error, communication error, or if the address does not exist  
! is a delimiter indicating a valid command was received  
? is a delimiter indicating the command was invalid  
aa (range 00~FF) is the 2-character hex slave address of the specified module  
(data) is a 2-character hexadecimal value representing the statuses of the digital input channels  
(cr) is the terminating character, carriage return (0Dh)

**Example** command: \$017(cr)  
response: !0101 (cr)

This command requests the module at Address 01h to return the values of all digital input channels. Digital Input Channel 0 is ON and Channel 1 is OFF since the return value is 1 (01b).

## #aacdd

**Name** Write digital output values

**Description** Writes a value to one or all digital output channels

**Syntax** #aacdd(cr)  
# is a delimiter  
aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01)  
cc specifies which channel(s) you want to set  
Write to all channels (byte): Both characters should be 0  
Write to a single channel (bit): First character is 1  
The second character indicates the channel number (0~1)  
dd is the hexadecimal representation of the digital output value(s)  
Write to all channels (byte): The decimal equivalent of these hex values represents the channel values  
Write to a single channel (bit): First character is always 0  
The second character is either 0 or 1 (i.e., the digital output value)

**Response** >(cr) if the command is valid  
?aa(cr) if an invalid command was entered  
There is no response if the module detects a syntax error, communication error, or if the address does not exist  
> is a delimiter indicating a valid command was received  
? is a delimiter indicating the command was invalid  
aa (range 00~FF) is the 2-character hex slave address of the responding module  
(cr) is the terminating character, carriage return (0Dh)

**Example** command: #011101(cr)  
 response: >(cr)  
 An output byte with value 1 is sent to Digital Output Channel 1 of a module at Address 01h. Digital Output Channel 1 of the module is thus set to ON.  
 command: #010002(cr)  
 response: >(cr)  
 An output byte with value 02 (10b) is sent to the module at address 01h. Digital Output Channel 1 is set to ON; Channel 0 is set to OFF.

## \$aa2

**Name** Read digital output values

**Description** Read the value of all digital output channels

**Syntax** \$aa2(cr)  
 \$ is a delimiter  
 aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01)  
 2 is a symbol of reading digital output

**Response** !(cr) if the command is valid  
 ?aa(cr) if an invalid command was entered  
 There is no response if the module detects a syntax error, communication error, or if the address does not exist  
 ! is a delimiter indicating a valid command was received  
 ? is a delimiter indicating the command was invalid  
 aa (range 00~FF) is the 2-character hex slave address of the responding module  
 (cr) is the terminating character, carriage return (0Dh)

**Example** command: \$aa2(cr)  
 response:!0101(cr)  
 An output byte with value 1(01b) is read from Digital Output Channel 0 of a module at Address 01h.  
 Digital Output Channel 0 is set to ON; Channel 1 is set to OFF. →  
 command: \$aa2(cr)  
 response:!0102(cr)  
 An output byte with value 2 (10b) is read from the module at address 01h. Digital Output Channel 1 is set to ON; Channel 0 is set to OFF.

## \$aaBcc

<b>Name</b>	Read single analog input range code
<b>Description</b>	Returns the channel range code of a specific analog input channel
<b>Syntax</b>	\$aaBnn(cr) \$ is a delimiter aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01) B refers to the single analog input range code command cc (range 00-07) specifies the channel you want to read the range code from (cr) is the terminating character, carriage return (0Dh)
<b>Response</b>	!aa(data)(code) if the command is valid ?aa(cr) if an invalid command was entered There is no response if the module detects a syntax error, communication error, or if the address does not exist ! is a delimiter indicating a valid command was received ? is a delimiter indicating the command was invalid (cr) is the terminating character, carriage return (0Dh) (code) is the range code (refer to the following tables)

### ADAM-6024 アナログ入力チャンネル・レンジ・コード

Range Code (Hex)	Range Code (Decimal)	Range Description
07	7	4~20 mA
08	8	±10 V
0D	13	0~20 mA

<b>Example</b>	command:	\$01B01(cr) !010D
	response:	The range code of Channel 1 is 0D (hex), meaning "0~24 mA"



## \$aa6

**Name** Read all channel statuses

**Description** Returns the statuses of all digital input channels

**Syntax** \$aa6(cr)  
\$ is a delimiter  
aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01)  
6 refers to the read all channel statuses command  
(cr) is the terminating character, carriage return (0Dh)

**Response** !aa00(data)(data)(data)(data)(cr) if the command is valid  
?aa(cr) if an invalid command was entered  
There is no response if the module detects a syntax error, communication error, or if the address does not exist  
! is a delimiter indicating a valid command was received  
? is a delimiter indicating the command was invalid  
aa (range 00~FF) represents the 2-character hex slave address of the specified module  
(data) is a 2-character hex value representing the values of the digital input module  
(cr) is the terminating character, carriage return (0Dh)

**Example** command: \$016(cr)

response: !01000FFD(cr)

This command requests the module at Address 01h to return the values of all channels. The first 2-character portion of the response (excluding the "!" delimiter) indicates the module address; the second 2-character portion of the response is reserved (always 00). Characters 5~8 in the response give the values for Channels 15~12, 11~8, 7~4, and 3~0, respectively. In this example, the character "0" (hex) = "0000" (binary), meaning that Channels 15~12 are all OFF; the character "F" (hex) = "1111" (binary), meaning that Channels 11~8 and 7~4 are all ON; and the character "8" (hex) = "1101" (binary), meaning that Channel 0, 2, and 3 are ON whereas Channel 1 is OFF.

## #aabb(data)

**Name** Set digital output status

**Description** This command sets a single or all digital output channels to the specific ADAM-6000 module.

**Syntax** #aabb(data)(cr)  
# is a delimiter.  
aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01)  
bb specifies which channel(s) you want to set  
Write to all channels (byte): Both characters should be equal to 0 (BB=00)  
Write to a single channel (bit): The first character is 1, the second character indicates the channel number (range 0h~Fh)  
(data) is the hex representation of the digital output value(s)  
Write to a single channel (bit): The first character is always 0, the second character is either 0 or 1  
Write to all channels (byte): the binary equivalent of these hex values represents the channel values

<b>Response</b>	<p>&gt;(cr) if the command is valid          ?aa(cr) if an invalid command was entered          There is no response if the module detects a syntax error, communication error, or if the address does not exist          &gt; is a delimiter indicating a valid command was received          ? is a delimiter indicating the command was invalid          aa (range 00~FF) is the 2-character hex slave address of the responding module          (cr) is the terminating character, carriage return (0Dh)</p>
<b>Example</b>	<p>command: #011201(cr)          response: &gt;(cr)</p> <p>An output bit with value 1 is sent to Channel 2 of the digital output module at Address 01h.          Channel 2 of the digital output module is thus set to ON.</p> <p>command: #010012(cr)          response: &gt;(cr)</p> <p>An output byte with value 12h (00010010) is sent to the digital output module at Address 01h. Thus, Channels 1 and 4 will be set to ON, and all other channels will be set to OFF.</p>

### \$aaJCFFFFssmm

<b>Name</b>	Read digital input counter values
<b>Description</b>	Returns the counter value of one or more digital input channels
<b>Syntax</b>	<p>\$aaJCFFFFssmm(cr)          \$ is a delimiter          aa (range 00~FF) is the 2-character hex slave address of the specified module (always 01)          JCFFFF refers to the read digital input counter values command          ss (range 00-07) specifies the start channel you want to read the counter value from          mm (range 00-07) specifies the number of channels you want to read the counter values from          (cr) is the terminating character, carriage return (0Dh)</p>
<b>Response</b>	<p>&gt;aa(data) if the command is valid          ?aa(cr) if an invalid command was entered          There is no response if the module detects a syntax error, communication error, or if the address does not exist          &gt; is a delimiter indicating a valid command was received          ? is a delimiter indicating the command was invalid          (data) is the counter value          (cr) is the terminating character, carriage return (0Dh)</p>
<b>Example</b>	<p>Command: \$01JCFFFF0001(cr)          Response: &gt;01000000A(cr)</p> <p>This command requests the module at Address 01h to return the counter value from Channel 0 (the value "00" means that the first read channel is 0; the value "01" means that only one channel is read). The module returns the count value 0000000A(h) from Channel 0.</p>
<b>Example</b>	<p>Command: \$01JCFFFF0C02(cr)          Response: &gt;01000000A00000001(cr)</p> <p>This command requests the module at Address 01h to return the counter value from Channels 12 and 13 (the value "0C" means that the first read channel is 12; the value "02" means that two channels are read). That module return the count value 0000000A(h) from Channel 12 and 00000001(h) from Channel 13.</p>

## 7.5 ADAM-6000モジュール用SNMP

ADAM-6000 (D バージョン) は SNMP v2c をサポートしており、ネットワーク上の ADAM-6000 モジュールの管理に便利です。SNMP トラップ機能を有効にすると、ADAM-6000 モジュールは、I/O ステータスの変更を SNMP メッセージで管理ステーションにアクティブに通知します。

### 7.5.1 ADAM MIBファイル

アドバンテックは、ADAMモジュール用の標準SNMPフォーマットに基づいて設計されたADAM MIBファイルを提供し、NMS（ネットワーク管理ソフトウェア）との統合を簡素化します。ADAM MIBファイルをNMSにインポートすることで、ADAM-6000をネットワーク上で簡単に監視することができます。

注：カウンタ用オブジェクト Identity(node)CounterObjは、ADAM-6051のカウンタ・モードでのみサポートされる。

### 7.5.2 SNMPトラップの設定

SNMP トラップ機能は、Adam/Apax .NET Utilityまたは ASCII コマンドを使用して設定できます。

Host IP:	SNMP トラップ・メッセージを送信する管理ステーションの IP。注：ホストIPを設定する前に、トラップ機能を無効にする必要があります。
Deadband:	トラップがノイズによって過剰にトリガーされるのを防ぐために、2つのトラップのトリガー間の最小間隔としてデッドバンドを設定することができます。
Specific type:	トラップ情報には、トラップの特定のタイプに関する詳細が含まれています。イベントの場所を特定するために使用される。具体的なタイプは下表の通り。

#### 特定のトラップ・タイプ

Specific Type	Channel	Description
1	0	
2	1	
3	2	
4	3	
5	4	デジタル入力変更
6	5	
7	6	
8	7	
17	0	
18	1	
19	2	
20	3	
21	4	デジタル出力変更
22	5	
23	6	
24	7	

Specific Type	Channel	Description
131	0	
132	1	
133	2	
134	3	アナログ入力ハイアラーム
135	4	
136	5	
137	6	
138	7	
141	0	
142	1	
143	2	
144	3	アナログ入力低下アラーム
145	4	
146	5	
147	6	
148	7	

#### Adam/Apax .NET Utilityによる設定

SNMP トラップメッセージの送信先ホスト IP は、Adam/ Apax .NET Utility V2.05.10 (B04)以降を使用して設定できます。以下の手順に従ってください：

1. Module Tree表示エリアから、設定したいモジュールを選択します。
2. ステータス表示エリアのStream/Trapタブをクリックし、SNMP Trapタブをクリックします。
3. トラップ機能が無効になっていることを確認します（ホストIPの横のチェックボックスをオフにします）。
4. ホストIPを入力し、適用をクリックします。
5. トラップ機能を有効にする（ホストIPの横にあるチェックボックスを選択する）
6. デッドバンドの値を設定し、Apply changeをクリックします。

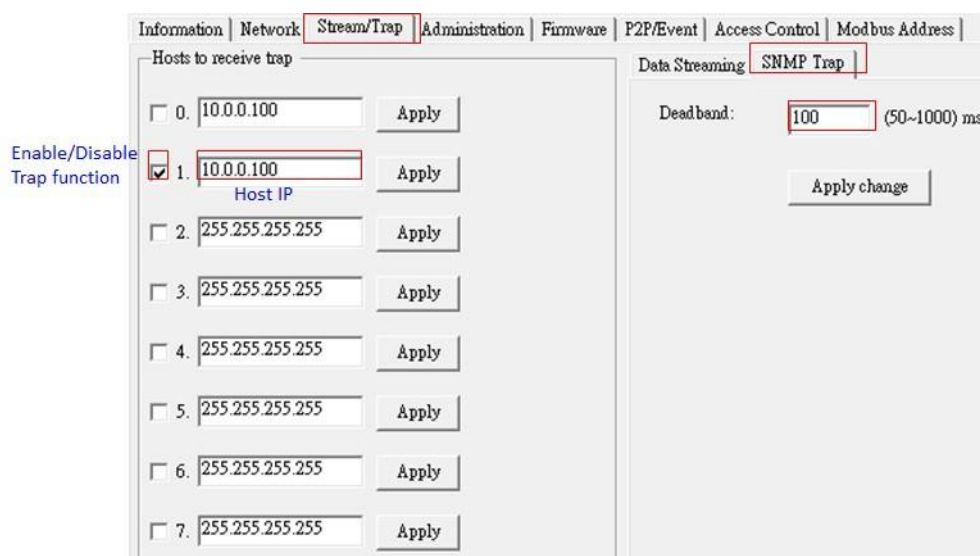


図 7.3 Adam/Apax .NET Utilityによるトラップ設定

## ASCIIコマンドによる設定

Command Syntax	Description	Example
%01SETTRAPcR c: Host ID (0~7)	Enable trap	Situation: Enable Host 1 SNMP trap notifications Command: %01SETTRAP1R Response: >01
%01SETTRAPcP c: Host ID (0~7)	Disable trap	Situation: Disable host 1 SNMP trap notice Command: %01SETTRAP1P Response: >01
%01SETTRAPcaabbdee aabbdee: Host IP in hex c: Host ID (0~7)	Set the host IP	Situation: Set Host 1 IP to 10.0.0.100 Command: %01SETTRAP10A000064 Response: >01
%01SETTRAPST00000xxx  xxx: ms (hex)	Set Trap deadband	Situation: Set Trap deadband to 100 ms Command: %01SETTRAPST00000064 Response: >01
%01GETTRAPc  c: Host ID (0~7)	Get the IP of the host	Situation: Get IP address of host 1 Command: %01GETTRAP1 Response: !0A000064
%01GETTRAPST	Get trap deadband	Situation: Get trap deadband (100 ms) Command: %01GETTRAPST Response: !00000064
%01GETTRAPcSTU c: Host ID (0~7)	Get trap enable/disable flag	Situation: Get trap enable/disable status of Host 1 Command: %01GETTRAP1STU Response: !00 / !01 (00=disable,01=enable)

### 7.5.3 SNMP OID値

SNMP OIDの値を設定することで、SNMP経由でモジュールの設定、I/O値、ネットワーク情報などの情報を設定/取得することができます。設定情報はMIB記述から取得できます。

#### aiRangeCode : アナログ入力レンジ設定用OID

値	入力範囲
143	±10 V
142	±5 V
140	±1 V
104	±500 mV
103	±150 mV
148	0~10 V
147	0~5 V
145	0~1 V
106	0~500 mV
105	0~150 mV
182	0~20 mA
180	4~20 mA
181	±20 mA

**注意** すべてのOIDの設定については、MIBファイルの説明を参照してください。



## 7.6 ADAM-6000モジュール用MQTT

### 7.6.1 MQTTの紹介

MQTTプロトコルは、IoTアプリケーション用の軽量メッセージング・トランスポートである。クライアントはブローカーに接続し、ブローカーはMQTTメッセージを転送する。ADAM-6000モジュールには、IoTアプリケーションの柔軟性を高めるいくつかの機能があります。

#### 機能1：MQTTメッセージを積極的に公開する

ADAM-6000モジュールは、ユーザー定義の間隔でMQTTメッセージの形でI/Oデータをアクティブに公開するように設定できます。この機能は、データを送信する効率的な手段を提供し、システム負荷を低減します。

#### 機能2：アクティブアラームイベント通知によるダウンタイムの短縮

アラームイベントは通常、デジタル入カステータスの変化、またはアナログ入力定義された範囲内または範囲を超えている場合に発生します。ADAM-6000モジュールは、即時通知を発行できるアラーム・トリガ・メカニズムを備えています。アラーム条件が満たされると、MQTTメッセージが直ちにブローカーに発行されます。

#### 特徴3：MQTTプロトコルを扱うゲートウェイが不要

一般的なアプリケーションでは、MQTTプロトコル変換に対応するためにゲートウェイが必要となり、システム構成が不必要に複雑になる可能性があります。ADAM-6000モジュールはMQTTプロトコル処理機能を内蔵しており、ゲートウェイを介さずにMQTT経由でブローカーに直接データを送信できるため、MQTTアプリケーションの確立にかかるコストと労力を削減できます。

### 7.6.2 ADAMモジュールのMQTTフォーマット

デジタルI/OモジュールADAM-6050-D, ADAM-6051-D, ADAM-6052-D, ADAM-6060-D, ADAM-6066-D

Description	MQTT Topic	JSON data	Firmware
Gets the I/O data of ADAM digital input/output module	Advantech/MAC ID /data Example: Advantech/0013430C981C/data	{"s":1,"t":0,"q":192,"c":1,"dix":"DI status","dox":"DO status"}	V6.01B11 or later
Set the value of a digital output of ADAM module	Advantech/MAC ID/ctl/dox  Example: Advantech/0013430C981F/ctl/do1	{"v":"DO status"},	V6.01B11 or later
Will Topic	Advantech/MACID/Device_Status Example: Advantech/0013430C981F/Device_Status	{"status":"Device Status", "name":"Device Name", "macid":"MACID", "ipaddr":"IP Address"}	V6.01B13 or later

**注** 青で示された項目については、次の「索引設定」のセクションを参照してください。



### アナログ入力モジュールADAM-6017-D

Description	MQTT Topic	JSON data	Firmware
Get the I/O data of an ADAM analog input module	Advantech/MAC ID /data Example: Advantech/0013430C981C/data	{"s":1,"t":time,"q":192,"c":1,"aix":AI value,"ai_stx":condition, "dox":DO status, "do_stx":condition}	V6.02B00 or later
Get an analog input range configuration	Advantech/MAC ID/cfg/sensor/aix Example: Advantech/0013430C981F/cfg/sensor/ai1	{"typ":"AI Range"}	V6.02B00 or later
Set the value of a digital output of an ADAM module	Advantech/MAC ID/ctl/dox Example: Advantech/0013430C981F/ctl/do1	{"v":DO status},	V6.02B00 or later
Set an analog input configuration	Advantech/MAC ID/set/sensor/aix Example: Advantech/00D0C9F94344/set/sensor/ai1	{"typ":"AI Range"}	V6.02B00 or later
Will topic	Advantech/MACID/Device_Status Example: Advantech/0013430C981F/Device_Status	{"status":"Device Status", "name":"Device Name", "macid":"MACID", "ipaddr":"IP Address"}	V6.02B00 or later

### 熱電対入力モジュール : ADAM-6018+-D

Description	MQTT Topic	JSON data	Firmware
Gets the I/O data of ADAM Analog input module*1	Advantech/MAC ID /data <b>Example:</b> Advantech/0013430C981C/data	{"s":1,"t":time,"q":192,"c":1,"aix":AI value,"ai_stx":condition, }	V6.01 B10 and later version
Set to trigger the I/O data update to broker (data update immediately)	Advantech/MACID/read/data <b>Example:</b> Advantech/0013430C981C/read/data	{"v":true}	V6.01 B10 and later version
Gets an analog input range configuration.	Advantech/MAC ID/cfg/sensor/aix <b>Example:</b> Advantech/0013430C981F/cfg/sensor/ai1	{"typ":"Thermocouple range"}	V6.01 B10 and later version

Set an analog input configuration	Advantech/MAC ID/set/sensor/aix <b>Example:</b> Advantech/00D0C9F94344/set/sensor/ai1	{"typ":"Thermocouple range"}	V6.01 B10 and later version
Will Topic	Advantech/MACID/Device_Status <b>Example:</b> Advantech/0013430C981F/Device_Status	{"status":"Device Status", "name":"Device Name", "macid":"MACID", "ipaddr":"IP Address"}	V6.01 B10 and later version

**注意**

\*1 データは設定に従って定期的にブローカーに更新されるため、このトピックでブローカーから取得できるのは、前回ブローカーに更新されたデータです。ブローカーですぐに更新されたデータを取得したい場合、Advantech/MACID/read/dataトピックを使用して、モジュールがブローカーに現在のI/Oデータを更新するようにトリガーします。

**ユニバーサルI/Oモジュール : ADAM-6024-D**

Description	MQTT Topic	JSON data	Firmware
Gets the I/O data of ADAM digital input/output module	Advantech/MAC ID /data Example: Advantech/0013430C981C/data	{"s":1,"t":0,"q":192,"c":1,"dix":"DI status","dox":"DO status"}	V6.01 B06 or later
Set the value of a digital output of ADAM module	Advantech/MAC ID/ctl/dox Example: Advantech/0013430C981F/ctl/do1	{"v":"DO status"},	V6.01 B06 or later
Get the I/O data of an ADAM analog input module	Advantech/MAC ID /data Example: Advantech/0013430C981C/data	{"s":1,"t":"time","q":192,"c":1,"aix":"AI value","ai_stx":"condition","dox":"DO status","do_stx":"condition"}	V6.01 B06 or later
Get an analog input range configuration	Advantech/MAC ID/cfg/sensor/aix Example: Advantech/0013430C981F/cfg/sensor/ai1	{"typ":"AI Range"}	V6.01 B06 or later
Set an analog input configuration	Advantech/MAC ID/set/sensor/aix Example: Advantech/00D0C9F94344/set/sensor/ai1	{"typ":"AI Range"}	V6.01 B06 or later
Get the I/O data of ADAM analog output module	Advantech/MAC ID /data Example: Advantech/0013430C981A/data	{"s":1,"t":0,"q":192,"c":1,"aox":"AO value","dix":"DI status"}	V6.01 B06 or later
Set the I/O data of ADAM analog output module	Advantech/MAC ID/ctl/aox Example: Advantech/0013430C981F/ctl/ao2	{"v":"AO Value"}	V6.01 B06 or later

Set an analog output configuration	Advantech/MAC ID/set/sensor/aox Example: Advantech/0013430C981F/set/sensor/ao2	{"typ": "AO Range"}	V6.01 B06 or later
Will Topic	Advantech/MACID/Device_Status Example: Advantech/0013430C981F/Device_Status	{ "status": "Device Status", "name": "Device Name", "macid": "MACID", "ipaddr": "IP Address" }	V6.01 B06 or later

## インデックス設定

### 一般

s:	Reserved for further use, default value = 1
t:	Trigger time Format: YYYY-MM-DDThh:mm:ss YYYY = year, MM = month, DD = date, hh = hour, mm = minute, ss = second  Note: the function is not applied on ADAM-6050/6051/6052/6060/6066, t = 0
q:	Reserved for further use, default value 192
c:	Reserved for further use, default value 1
Device Status:	connect = module is connected disconnect = module is disconnected

### デジタルI/Oモジュール

dix:	Digital input status of channel (x-1) example: {"di2": true} means the status of Digital Input Channel 1 = true
dox:	Digital output status of channel (x-1) example: {"do2": true} means the status of Digital Output Channel 1 = true
DO status	true: on; false: off
DI status	true: on; false: off

### アナログI/Oモジュール

aix:	Analog input value of channel (x-1)  Note: If the analog input channel is disabled, the analog input value will be "9999.9999"
ai_stx:	Condition of analog input channel (x-1)

ai_stx value	Condition
0	Channel disable
1	Streaming, normal
2	High latch
3	High momentary
4	Low latch
5	Low momentary

**do\_stx**

デジタル出力チャンネル(x-1)の状態 (ADAM-6017のみ)

do_stx Value	Condition
1	Streaming, normal
2	DO change

**アナログ入力範囲**

対応するレンジの値は、入力レンジを設定するためのものです：

AI Range Command	Input Range
0-20mA	0~20 mA
4-20mA	4~20 mA
+20mA	±20 mA
0-5V	0~5 V
1-5V	1~5 V
0-10V	0~10 V
0-1V	0~1 V
0-500mV	0~500 mV
0-150mV	0~150 mV
+0V	±10 V
+5V	±5 V
+2.5V	±2.5 V
+1V	±1 V
+500mV	±500 mV

**熱電対レンジ**

対応するレンジの値は、入力レンジを設定するためのものです：

Thermocouple range command	Thermocouple range
K Type:0-1370C	Set type to K Type:0-1370°C
J Type:0-760C	Set type to K Type: 0-760°C
E Type: 0-1000C	Set type to K Type: 0-1000°C
T Type:-100-400C	Set type to K Type: -100-400°C
R Type:500-1750C	Set type to K Type: 500-1750°C
S Type:500-1750C	Set type to K Type: 500-1750°C
B Type:500-1800C	Set type to K Type: 500-1800°C

### 7.6.3 MQTTの設定

ADAM-6000 モジュールの MQTT は、Adam/Apax.Net Utility (V2.05.11B17 以降) または ASCII コマンドを使用して設定できます。

注：設定前にMQTT機能を無効にし、設定後に有効にする必要があります。

#### ホスト（ブローカーIP）

ブローカーの URL または IP アドレスを設定できます。ADAM-6000モジュールは、標準の MQTTプロトコルを介してブローカーに接続します。

#### ハートビート（キープ・アライヴ）

ブローカーは、ハートビート（Keep Alive）設定によって定義された間隔で、ADAM-6000モジュールとの接続を定期的にチェック。最小間隔は 5 秒です。

#### デッドバンド

デッドバンドは、2 つの MQTT メッセージを公開する間の最小間隔を決定するために設定します。ノイズによって MQTT メッセージが過剰に発行されるのを防ぐために設定します。

#### メッセージの保持

retain 機能が有効な場合、ブローカーはトピックの最後のメッセージを保存する。トピックに対する新しいサブスクリプションが作成されると、そのメッセージがクライアントに送信されます。クライアントは最後のメッセージをすぐに取得でき、次のメッセージが更新されるまで待つ必要はありません。

#### ウィル・トピック

クライアントが切断された ADAM-6000 モジュールのトピックをサブスクライブしている場合、ブローカーは、will トピックをサブスクライブしているクライアントに will メッセージを送信してクライアントに通知します。

Will Topic of ADAM: Advantech/MACID/Device\_Status

Will message: { "status":"Device Status", "name":"Device Name","macid":"MACID", "ipaddr":"IP Address"}

Example: { "status":"disconnect", "name":"ADAM6051", "macid": "00D0C9FEFFF5", "ipaddr": "10.0.0.1"}

#### QoS（サービス品質）

ユーザーはパブリッシュ/サブスクライブのQoSレベルを選択できる。MQTTでは3つのQoS（Quality of Service）レベルが定義されている。

Level 0: broker/client deliver the message at most once

Level 1: broker/client deliver the message at least once

Level 2: broker/client deliver the message exactly once

#### 発行/購読トピック

MQTT メッセージは、MQTT トピックに基づいてブローカーによって転送される。各メッセージにはデータ含まれます。クライアントがブローカーに MQTT メッセージをパブリッシュすると、トピックをサブスクライブしたクライアントはそれに応じて MQTT 受信します。

## ユーザー名/パスワード

認証制御を必要とする一部のアプリケーションでは、ブローカーがデータに対するサブスクライバの権限を制限する。ADAM-6000 モジュールでは、Adam/Apax .NET Utilityを使用してユーザー名/パスワードを設定できます。ADAM-6000 モジュールからの MQTT メッセージには、ブローカーにアクセスするためのユーザー名とパスワードが含まれます。

モデル	ファームウェアバージョン
ADAM6050-D、ADAM6051-D ADAM6052-D、ADAM6060-D ADAM6066-D	V6.02B01以降
ADAM-6017-D	V6.02B00以降
ADAM-6018+-D	V6.01B10以降

## MQTT TLS暗号化

ADAM は送信中の TLS 暗号化をサポートする。TLS機能を有効にし、適用をクリックして設定を実行します。

Model	Firmware version	Adam/Apax .NET utility version
ADAM-6017-D	V6.10 B07 or later version	2.05.11 B23 or later version
ADAM-6018+-D	V6.01 B10 or later version	2.05.11 B23 or later version
ADAM-6050-D, ADAM-6051-D, ADAM-6052-D, ADAM-6060-D, ADAM-6066-D	V6.10 B02 or later version	2.05.11 B23 or later version

## ユーザー定義のMQTTトピック

ADAMはユーザー定義トピックをサポートしており、ユーザーはAdam/Apax .NET Utilityでデフォルトのトピック名を変更できます。

Model	Firmware version	Adam/Apax .NET utility version
ADAM-6017-D	V6.10 B07 or later version	2.05.11 B23 or later version
ADAM-6050-D, ADAM-6051-D, ADAM-6052-D, ADAM-6060-D, ADAM-6066-D	V6.10 B02 or later version	2.05.11 B23 or later version

Basic **Advanced**

User Defined MQTT Name:  Enable Apply Advanced Settings

Overwrite Will Topic Name:

Overwrite Will Message:

Overwrite Connect Message:

Overwrite Publish Topic Name:

Overwrite AI Type Code Message:

Overwrite AI Channel Publish:

AI	Message	Status	Type Code Topic
CH0	<input type="text" value="AI value"/>	<input type="text" value="AI status"/>	<input type="text" value="AIIO_Type"/>

### Enable the user defined MQTT function

**Overwrite Will Topic Name:** change the will topic name

**Overwrite Will Message:** change the will message

**Overwrite Connect Message:** change the connect message when connecting

**Overwrite Publish Topic Name:** change the I/O value Topic

**Overwrite AI Type code Message:** change the AI type code title of analog input channel in MQTT message payload

**Overwrite the AI channel publish:** Double click the field and change the I/O value Message, Status, Type Code Topic of each channel

#### 注



1. 設定を有効にするには、「詳細設定を適用」をクリックします。
2. ユーザーが定義したMQTT機能が無効にして、すべてのトピックとメッセージをデフォルトに設定し、「詳細設定を適用」をクリックします。

## Adam/Apax .NET Utilityによる設定

[Cloud] タブをクリックして、MQTT 設定を構成します。

ブローカーの URL または IP アドレスを **Host** ボックスに設定できます。3つのパブリックブローカーソースリンクがユーティリティにリストされています：

- [iot.eclipse.org](http://iot.eclipse.org)
- [test.mosquitto.org](http://test.mosquitto.org)
- [broker.mqttdashboard.com](http://broker.mqttdashboard.com)

## ASCIIを使った設定

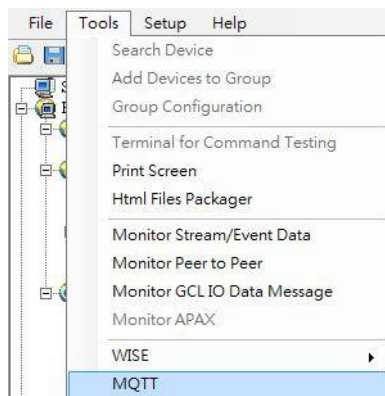
Command	Description	Remarks
%aaSETMQTTENxx	Set MQTT enable/disable aa: always 01 xx: 01 (enable) 00 (disable)	Return: >01 Error: ?01
%aaSETMQTTADxx...x	Set IP address of the broker aa: always 01 xx...x: IP address/domain (0~49 character)	Return: >01 Error: ?01
%aaSETMQTTHBxxxx	Set heartbeat interval aa: always 01 xxxx: heartbeat interval in second (0005~FFFF)	Return: >01 Error: ?01
%aaSETMQTTPDxxxx	Set publishing deadband aa: always 01 xxxx: publishing deadband in millisecond (0032~03E8)	Return: >01 Error: ?01
%aaSETMQTTPRxx	Set publishing retain enable/disable aa: always 01 xx: 01 (enable), 00 (disable)	Return: >01 Error: ?01
%aaSETMQTTPQxx	Set publishing Qos aa: always 01 (xx): publishing Qos (00~02)	Return: >01 Error: ?01

%aaSETMQTTSQxx	Set subscribing Qos aa: always 01 (xx): publishing Qos (00~02)	Return: >01 Error: ?01
ujhhj%aaGETMQTTEN	Get MQTT enable/disable aa: always 01	Return: !01 (enable) !00 (disable) Error: ?01
%aaGETMQTTAD	Get IP address of the broker aa: always 01	Return: !IP Address/ Domain (IP Address/ DomainName) Error: ?01
%aaGETMQTTHB	Get heartbeat interval aa: always 01	Return: !xxxx (heartbeat interval in hex format) Error: ?01
%aaGETMQTTPD	Get publishing deadband aa: always 01	Return: !xxxx (deadband in hex format) Error: ?01
%aaGETMQTTPR	Get publishing retain enable/disable aa: always 01	Return: !00 (enable) !01 (disable) Error: ?01
%aaGETMQTTPQ	Get publishing Qos aa: always 01	Return: !xx (publishing Qos in hex format) Error: ?01
%aaGETMQTTSQ	Get subscribing Qos aa: always 01	Return: !xx (subscribing Qos in hex format) Error: ?01
%aaSETMQTTUNxx...x	Set MQTT user name aa: always 01 xx...x: user name, if set null module will disable the username and pass- word function (0~49 character)	Return: >01 Error: ?01
%aaSETMQTTPWxx...x	Set MQTT password aa: always 01 xx...x: password, if set null module will disable the username and password function (0~99 character)	Return: >01 Error: ?01
%aaGETMQTTUN	Get MQTT user name aa: always 01	Return: !UserName Error: ?01
%aaGETMQTTPW	Get MQTT password aa: always 01	Return: !Password Error: ?01

## 7.6.4 ADAM-6000モジュールでMQTTを開始する方法

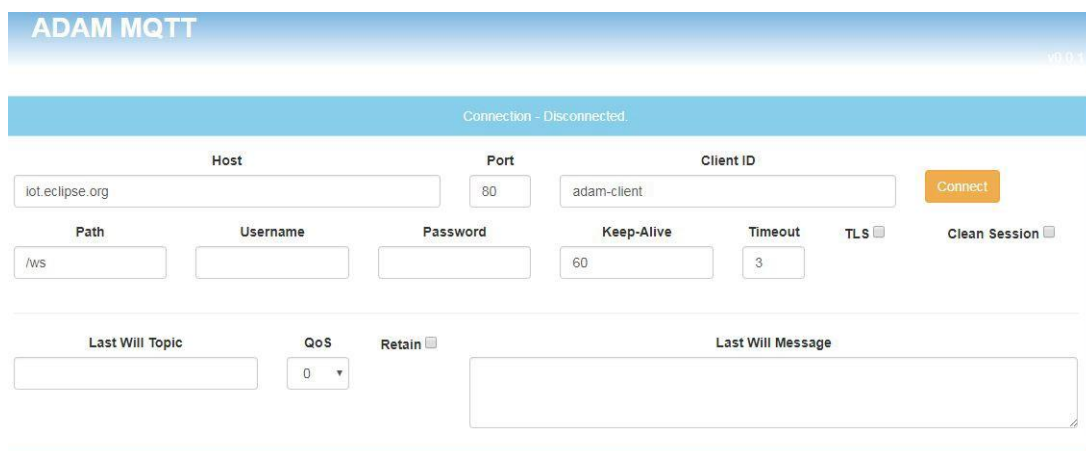
Adam/Apax .NET Utility (V2.05.11 以降) は、ADAM モジュールの MQTT 機能をテストするために、MQTT クライアントをシミュレートするページを提供します。このように、4つのステップでMQTTを使用したADAMモジュールの利点を体験することができます。

1. **Tools**メニューから**MQTT**を選択する。  
ページ



2. 接続の設定

接続設定ページでは、クライアント情報を設定できます。デフォルトのホストは、ポート80のパブリックブローカーソース "iot.eclipse.org" です。ホストのURLやIP設定することもできます。完了したら、[Connect] をクリックします。

A screenshot of the 'ADAM MQTT' web interface. The page title is 'ADAM MQTT' and the version is 'v2.0.1'. The status bar shows 'Connection - Disconnected'. The form contains several input fields and controls: 'Host' (iot.eclipse.org), 'Port' (80), 'Client ID' (adam-client), 'Connect' button, 'Path' (/ws), 'Username', 'Password', 'Keep-Alive' (60), 'Timeout' (3), 'TLS' checkbox, 'Clean Session' checkbox, 'Last Will Topic', 'QoS' (0), 'Retain' checkbox, and 'Last Will Message' text area.

### 注



1. Path, Username, Password, TLS, Clean Session機能はリリースされていません。
2. ウェブページはウェブソケット経由でのブローカーへの接続のみをサポートしています。

### 3. 購読／公開機能の設定

#### サブスクライブ

トピックを設定し、QoSレベルを選択し、[Subscribe]をクリックします。トピックのメッセージが履歴フィールドに表示されます。

#### 出版

発行トピック設定、QoS、メッセージを構成し、[Publish] をクリックします。MQTT メッセージがブローカーにパブリッシュされます。retain機能が有効になっている場合、ADAM-6000モジュールはトピックにサブスクライブするときに最後のメッセージを受信します。

The left screenshot shows the 'Subscribe' form with the following fields: Topic: Advantech/#, QoS: 0. Below the fields are 'Subscribe' and 'Unsubscribe' buttons.

The right screenshot shows the 'Publish Message' form with the following fields: Topic: Advantech/00D0C9FEFF66/ct/do3, QoS: 0, Retain: , Message: {\"v\":true}. Below the fields is a 'Publish' button.

### 4. MQTTメッセージを確認する

直近のMQTTメッセージと過去のメッセージは、last messageカラムとhistoryカラムで読むことができる。

Topic	Payload	Time	QoS
Advantech/00D0C9FEFF66/data	{\"s\":1,\"t\":0,\"q\":192,\"c\":1,\"d1\":true,\"d2\":true,\"d3\":true,\"d4\":true,\"d5\":true,\"d6\":true,\"do1\":true,\"do2\":true,\"do3\":true,\"do4\":false,\"do5\":false,\"do6\":f		
Advantech/00D0C9FEFF66/ct/do3	{\"v\":true}	2017-07-28T08:45:21.416Z	0
Advantech/00D0C9FEFF66/ct/do2	{\"v\":true}	2017-07-28T08:44:49.252Z	0

上の画像は、ADAMモジュールによって発行された最後のメッセージを示している。

Topic	Payload	Time	QoS
Advantech/00D0C9FEFF66/data	{\"s\":1,\"t\":0,\"q\":192,\"c\":1,\"d1\":true,\"d2\":true,\"d3\":true,\"d4\":true,\"d5\":true,\"d6\":true,\"do1\":true,\"do2\":true,\"do3\":true,\"do4\":false,\"do5\":false,\"do6\":f		
Advantech/00D0C9FEFF66/ct/do3	{\"v\":true}	2017-07-28T08:45:21.416Z	0
Advantech/00D0C9FEFF66/ct/do2	{\"v\":true}	2017-07-28T08:44:49.252Z	0

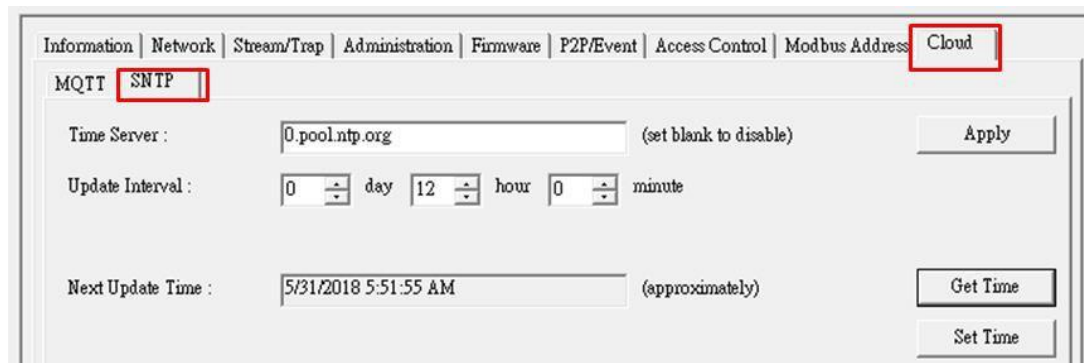
上の画像はADAMモジュールの履歴メッセージである。

## 7.6.5 リアルタイムクロック

ADAM-6017 (Dバージョン以降) は、UTCフォーマットのリアルタイム・クロック (RTC) をサポートしています。RTCを利用して、データやタイムスタンプを決定できます。RTCはSNTP経由で校正できます。SNTP 設定は、Adam/Apax を使用して構成できます。  
.NETユーティリティまたはASCIIコマンド。

## 7.6.6 Adam/Apax .NET UtilityによるSNTP設定

Adam/Apax .NET Utilityのバージョン 2.05.11 B16 以降では、SNTP 設定はクラウドタブから行うことができます。



### タイムサーバー

ターゲットモジュールのRTCと同期するSNTPサーバーを設定します。Time Serverの項目が空白のままだと、この機能は無効になります。設定がしたら **Apply** をクリックします。

### 更新間隔

ターゲットモジュールのRTCは、更新されたインターバル設定に基づくインターバル時間でタイムサーバーと同期します。設定が完了したら **Apply** をクリックします。

### 次の更新時間

このフィールドは、RTCがSNTPサーバーと同期する時刻を示します。**Get Time**をクリックすると、ユーティリティはADAMモジュールのUTCを取得し、コンピュータのタイムゾーンに基づいてローカルタイムに同期します。**Set Time**をクリックすると、モジュールのRTCがコンピュータのUTCと同期します。

## 7.6.7 ASCIIコマンドによるSNTP設定

Description	Command	Remark
Set SNTP server	%01SETSNTPADxxxxxxx XXXXXXXX: the SNTP server domain or IP	Return: >01 Error: ?01
Get SNTP server	%01GETSNTPAD	Return: the SNTP server domain or IP exam- ple:!0.pool.ntp.org\r the SNTP server at 0.pool.ntp.org
Set SNTP update interval	%01SETSNTPPTxxxxxxx xxxxxxx the update interval	Return: >01 Error: ?01
Get SNTP update interval	%01GETSNTPPT	Return: SNTP update interval in second example:!0000A8C0 the update interval is 12 hr (43200 s)
Set RTC time in UTC format	#01TMYYYY-MM-DDThh:mm:ssZ YYYY:year  MM:month DD:day hh:hour mm:minute ss:second	Return: >01 Error: ?01
Get RTC time in UTC for- mat	\$01TM	Return: the RTC time in UTC format  Example: >01TM2018-05- 02T05:54:03Z

# 8

グラフィック・コンディ  
ション・ロジック (GCL)

## 8.1 概要

従来のDA&Cシステムでは、システムは1台のコントローラで管理されます。ADAM-6000モジュールのようなリモートI/Oモジュールは、センサからデータを取得するか、他のデバイス/機器を制御するための信号を生成するためにのみ使用されます。このようなセットアップでは、PLCのようなコンピュータまたはコントローラが、入力モジュールからデータを取得し、データを操作し、入力データに従って論理演算と処理を実行し、その後、論理決定に基づいて出力データが生成され、出力モジュールに送信されます。

コンピュータ/コントローラとリモートI/Oモジュールは、同じネットワーク内で完全な制御システムを形成する。ロジック操作と処理の複雑さはアプリケーションに依存し、操作はコンピュータ/コントローラ上に書かれたプログラムによって実装される。プログラム作成に使用できるソフトウェアアプリケーションは多数あります。例えば、コンピュータを使用する場合はMicrosoft Visual Studio (C言語)、PLCコントローラを使用する場合はRSLogix (ラダー言語) などがあります。

多くのアプリケーションでは、ロジック操作とプロセスは比較的単純であり、そのような単純なアプリケーションのために強力すぎるコンピュータやコントローラを使用する必要はないように思われます。ファームウェアバージョン4.x (またはそれ以降) のADAM-6000モジュールは、GCLによるロジック操作とプロセス機能を備えています。これにより、ADAM-6000モジュールは、スタンドアロン制御システムとして機能するスマートなI/Oモジュールとなりました。

Adam/Apax .NET Utilityでロジック操作とプロセスルールを定義し、そのルールをADAM-6000モジュールにアップロードします。その後、ADAM-6000モジュールがロジックルールを実行し、入力条件に応じて異なるアクションを処理します。GCLを有効にすると、ADAM-6000モジュールがその役割を処理できるため、従来のシステムで使用されていたコンピュータ/コントローラを取り除くことができます。

Adam/Apax .NET Utility の GCL 設定環境は、完全にグラフィカルで、非常に簡単かつ直感的にロジックルールを設定を完了することができます。ロジックルールを設定とダウンロードが完了すると、Adam/Apax .NET Utilityでリアルタイムの実行状況と入力値を見ることができます。

## 8.2 GCL構成環境

ADAM/エイパックス.NET ユーティリティのModule Tree表示エリアにある **GCL コンフィギュレーション**項目リストをクリックすると、GCL 関連のすべての設定を行うことができます。

したがって、P2Pを有効にしている場合、GCL設定項目をクリックしてGCL設定環境を起動すると、P2Pが無効になっていることがわかります。同様に、GCL機能を有効にすると、P2P機能は無効になります。図8.1は、GCL設定項目が選択された後に、ステータス表示領域がどのように表示されるかを示しています。GCLメニュー、ロジックルール・セット・エリア、および個別ロジックルール設定エリアです。

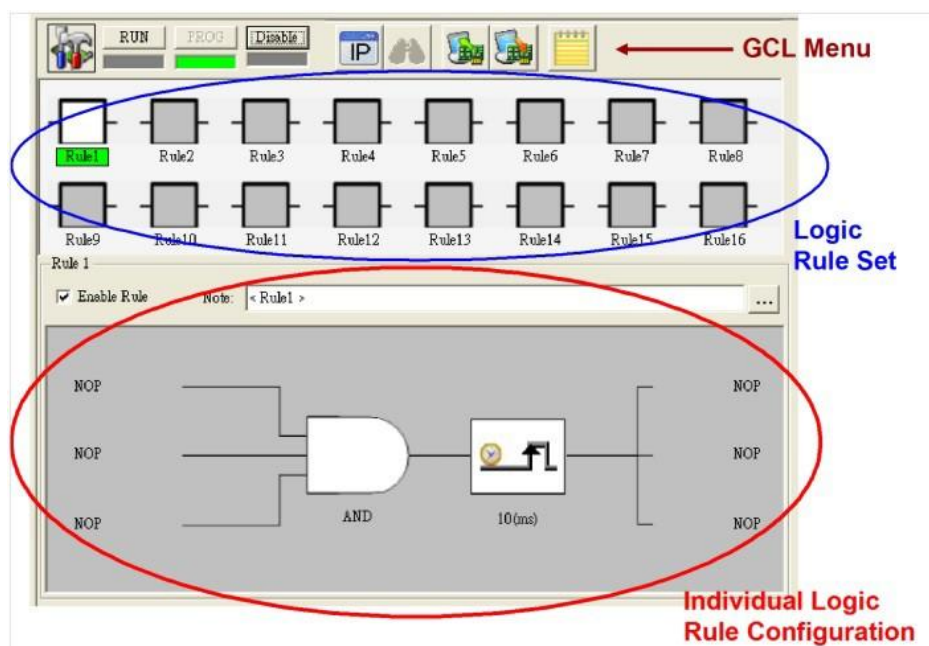


図8.1 GCL構成環境

### GCLメニューエリア

GCLメニューエリアのアイコンは、以下の表で説明されています：

Icon	Function	Description
	Current Status	This icon shows the current GCL status. The status is either disable, programming, or running mode (from top to button). Note: You cannot enable peer-to-peer/datastream function and GCL function at the same time. So if you want to enable GCL, the peer-to-peer and datastream function will be disabled automatically.
	Run GCL	When running mode is selected, this button will be lit.
	Program GCL	When programming mode is selected, this button will be lit.
	Disable GCL	When disable mode is selected, this button will be lit.
	IP Table Configuration	Click this button to configure IP table, which can used to define the output destination. This is available only in programming mode.
	Monitoring	Click this to enable online monitoring. This is available only in running mode.
	Upload Project	Click this to upload a GCL configuration from the module to the computer. This is available only in programming mode.
	Download Project	Download the current GCL configuration to the module. This is available only in programming mode.
	Project Content	Click this show the current GCL configuration. You can also save the current configuration to file or load a previous configuration from a saved file.

## ロジックルール・セット・エリア

各 ADAM-6000 モジュールは 16 個のロジックルールを保持することができ、それらはロジックルール・セット・エリアに 16 個のロジックルール・アイコンとして表示されます。ルールを設定するには、対応するロジックルールアイコンをクリックします。これにより、テキストの背景が緑色にハイライトされ、ルールがロジックルール設定エリアに表示されます。

## ロジックルール設定エリア

ロジックルール・セット・エリアからルールを選択したら、[Enable Rule] チェックボックスを選択してそのルールを有効にします。選択されたロジックのアイコンが白くなります。  
 Note テキストボックスの横のボタンをクリックすると、各ロジックルールの説明を書くことができます。各ルールは4つのステージで構成されます：図8.2の「個々のロジックルール設定エリア」に示すように、各ルールは 4つのステージから構成されます：入力条件、ロジック、実行、出力。図に示すステージの1つをクリックすると、対応する設定ウィンドウが表示されます。

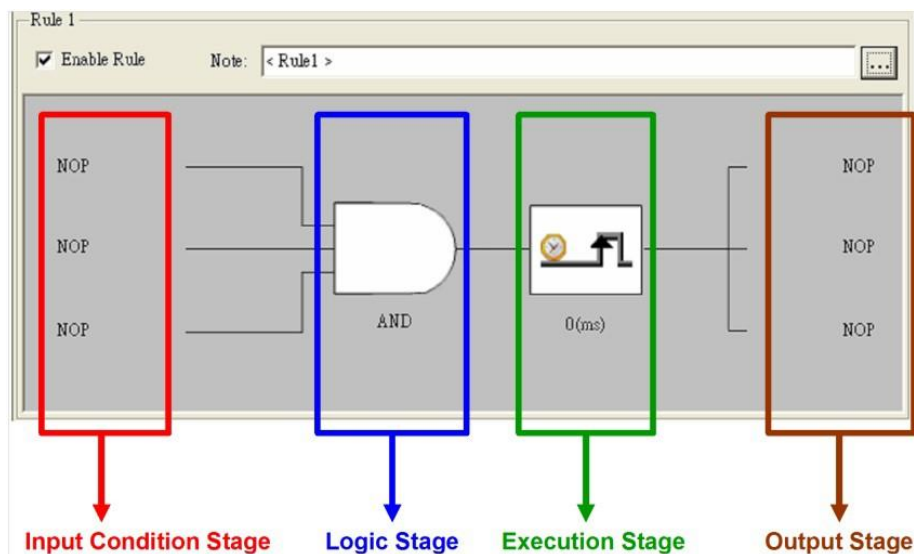


図8.2 1つの論理ルールに対する4つのステージ

	Option	Description	Condition	Section
Input コンディション Stage	NoOperation	No operation	N/A	
	AI	Local analog input channel value	>=, <=, =	
	DI	Local digital input channel value	True, False	
	DI_Counter	Local counter input channel value	>=, <=, =	
	DI_Frequency	Local frequency input channel value	>=, <=, =	8.3.1
	Timer	Local internal Timer value	>=	
	AuxFlag	Local internal Flag value	True, False	
	DO	Local digital output channel value	True, False	
	Counter	Local internal counter value	>=, <=, =	
	Logic Stage	Option	Description	
AND		AND operation		8.3.2
OR		OR operation		
NAND		NAND operation		
NOR		NOR operation		

Execution	Option	Description	Section
Stage	Execution_Period	Define the execution time for this logic rule	8.3.3
	SendTo NextRule	Combine output of this logic rule to next logic rule to form a logic cascade	

	Option	Description	Section
Output Stage	NoOperation	No operation	8.3.4
	AO	Local or remote analog output channel value	
	DO	Local or remote digital output channel value	
	DI_Counter	Local or remote counter input channel setting	
	DO_Pulse	Local or remote pulse output channel setting	
	Timer	Local internal Timer setting	
	AuxFlag	Local or remote internal Flag value	
	RemoteMessage	Remote message	
	Counter	Local internal counter setting	

## 8.3 論理ルールの4段階を設定する

### 8.3.1 入力条件ステージ

入力条件ステージは、入力データに基づいて論理条件判定を行う。結果の判定は論理真または論理偽となり、これが論理ステージに送られて論理演算が行われます。アナログ入力モードを例にとると、アナログ入力値が特定の値（限界値）より大きい場合、入力ステージは「True」をロジックステージに転送するように条件を定義できます（そうでない場合は「False」）。

入力条件ステージのアイコンをクリックすると、図8.3のようなウィンドウが表示される。**Mode（モード）** ボックスから入力モードを選択できる。入力モードのオプションは以下の通りである：

- NoOperation (default)
- AI (local analog input channel)
- DI (local digital input channel)
- DI\_Counter (local counter input channel)
- DI\_Frequency (local frequency input channel)
- Timer (internal timer)
- AuxFlag (internal flag)
- DO (local digital output channel)
- Counter (internal counter)

入力モードを選択し、関連するすべての設定を完了したら、「OK」をクリックする。入力コンディションのステージアイコンが、選択されたコンディションを表すアイコンに変わる。各モードの詳細は以下のセクションで説明する。

入力モード：操作なし

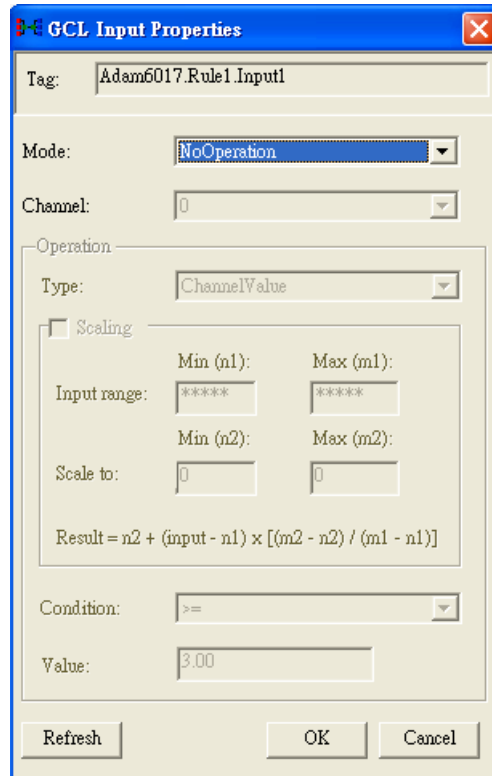


図8.3 入力条件ステージの構成

デフォルトのモードはNoOperationである。このモードでは、入力条件はない。

入力モード：AI（ローカル・アナログ入力チャンネル）

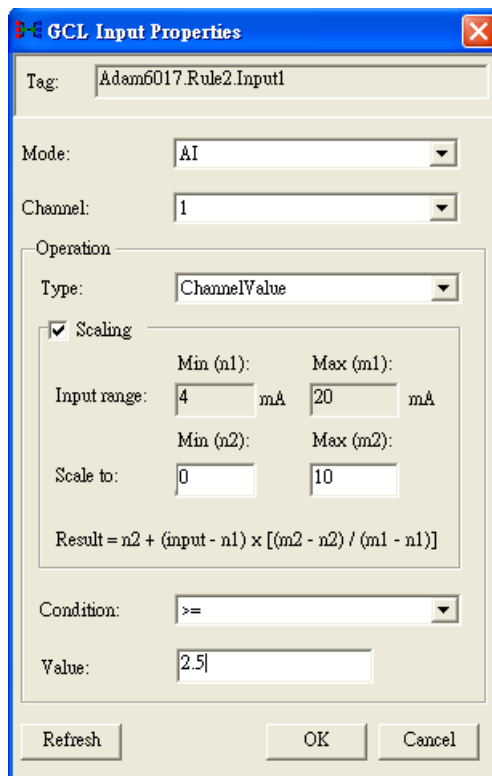


図 8.4 アナログ入力モードのスケール機能

以下の手順に従って、アナログ入力条件を設定してください：

1. **Mode**ボックスから**AI**を選択し、**Channel**ボックスから設定したいチャンネルを選択します。
2. **Operation** パネルで入力条件操作を定義する。これは、**ChannelValue**（選択されたアナログ入力チャンネルの現在値を条件の入力として使用する）または**Deviation**（偏差を条件の入力として使用する。）
3. 入力チャンネルの条件を **Condition** ボックスから選択し、しきい値を **Value** ボックスに入力します。次の表の例は、アナログ入力条件を定義する方法を示しています。

Channel	Type	Condition	Value	Description
0	ChannelValue	>=	5	If the value of Analog Input Channel 0 is more than or equal to 5, the condition is "logic true".
2	ChannelValue	=	3.2	If the value of Analog Input Channel 2 is equal to 3.2, the condition result is "logic true" (otherwise, "logic false").
3	ChannelValue	<=	1.7	If the value of Analog Input Channel 3 is less than or equal to 1.7, the condition is "logic true" (otherwise, "logic false").
5	Deviation	N/A	20	If the deviation of Analog Input Channel 5 is greater than 20%, the condition result is "logic true" (otherwise, "logic false").

アナログ入力は、指定されたチャンネルから電圧（または電流）を読み取ります。通常、電圧（または電流）値は現実の物理的な単位値（すなわち技術者単位値）を表し、電圧（または電流）値と技術者単位値との間には通常、直線的な関係があります。この線形関係を下图に示します：

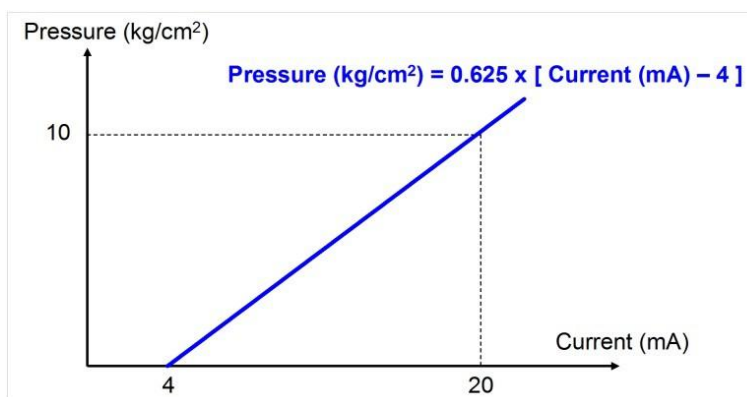


図 8.5 エンジニア・ユニットと電流値

ADAM-6000 アナログ入力モジュールは、電圧（または電流）値をエンジニアの単位値に変換するスケール機能を持っています。例えば、圧力値  $\geq 2.5 \text{ kg/cm}^2$  という条件があるとします。スケール機能がなければ、圧力値 ( $2.5 \text{ kg/cm}^2$ ) を電流値 (8 mA) に変換する必要があります。そうすると

条件を定義するには、**操作パネルの値ボックス**に現在の値を入力する必要があります。

その代わりに、**Operation** パネルで **Scaling** チェックボックスを選択すると有効になるスケール機能を使用することができます。これにより、**Scale to** 項目の **Min (n2)** と **Max (m2)** ボックスにエンジニア・ユニットの最小値と最大値を入力することができます。図 8.4 の例では、値 0 と 10 が最小圧力値と最大圧力値として入力されています。モジュールは圧力値を電流値に自動変換することができるため、条件を定義するには **Value** ボックスに圧力値 (2.5 kg/cm<sup>2</sup>) を入力するだけです (値のみを入力し、測定単位を含める必要はないことに注意してください)。この機能により、対応する電流計算することなく、より直感的に条件を設定することができます。

#### 入力モード : DI (ローカル・デジタル入力チャンネル)

入力モードとして **DI** を選択し、**Channel** ボックスから設定したいチャンネルを選択します。選択したチャンネルの入力値がコンディション入力として使用される。値がロジックHighの場合、コンディションの結果はロジックTrueとなり、ロジックLowの場合、コンディションの結果はロジックFalseとなる。

#### 入力モード : カウンタ (ローカルカウンタ入力チャンネル)

入力モードとして**DI\_Counter**を選択し、**Channel**ボックスからコンディションを設定したいチャンネルを選択します。選択したチャンネルのカウント値がコンディション入力として使用されます。AIモードと同様に、**Condition**ボックスから選択した入力チャンネルに適切なコンディションを選択し、**Value**ボックスに対応する値を入力する必要があります。コンディションは、入力チャンネルからのカウンタ値と**値**ボックスの値を比較します。コンディションが満たされた場合、コンディションの結果はロジック TRUE になります (そうでない場合はロジック FALSE)。

#### 入力モード : DI\_Frequency (ローカル周波数入力チャンネル)

入力モードとして **DI\_Frequency** を選択し、**Channel** ボックスから設定したいチャンネルを選択します。周波数入力周波数値がコンディション入力として使用されます。カウンタ・モードと同様に、**Condition** ボックスから入力チャンネルのコンディションを選択し、**Value** ボックスで応答する値を確保する必要があります。コンディションは、入力チャンネルからの周波数値と、**Value** ボックスに入力した値を比較します。コンディションが満たされた場合、コンディションの結果はロジック TRUE となります (そうでない場合はロジック FALSE)。

#### 入力モードタイマー (内部タイマー)

各 ADAM-6000 モジュールには 16 個のローカル・タイマーがあります。タイマーがスタートした後、その値は明らかに時間の表します。この入力モードでは、タイマーの値を条件入力として使用できます。これを行うには、入力モードとしてタイマーを選択し、**インデックス・ボックス**から適切なタイマーを選択します (範囲は0~15)。次に、**Condition** ボックスから条件を定義し、**Value** ボックスに対応する値を入力します (増加 : 0.01 秒)。


コンディションが満たされると、コンディション結果はロジックハイになる。例えば、**Condition** ボックスから **>=** を選択し、**Value** ボックスに "500" を入力した場合、これはタイマー値が 5 秒を超えるまでコンディションの結果がロジック ローのままであることを意味する。

### 入力モード：AuxFlag（内部フラグ）

各 ADAM-6000 モジュールには 16 個の内部フラグがあります。内部フラグのデータ・タイプはデジタルで、その値はロジック “true” またはロジック “false” のいずれかです。内部フラグの値をコンディション入力として使用するには、入力モードとして **AuxFlag** を選択し、**Index** ボックスから適切な内部フラグを選択します（範囲は 0 ～ 15）。次に、**Condition** ボックスから条件を定義します。

**コンディション** ボックスで **[トウル]** を選択すると、内部フラグ値がロジック トウルーのとき、コンディションの結果もロジック トウルーになります。**コンディション** ボックスで **[False]** を選択すると、内部フラグ値がロジック False のとき、コンディションの結果はロジック True になります。

内部フラグを使用して、ロジック・カスケードまたはロジック・フィードバックを 実装できます。この方法の詳細は、第 8.4 節を参照してください。

**注意**  他のプログラム・アプリケーションを使用して、ASCII コマンドや Modbus/TCP アドレスを介して内部フラグを読み書きすることができます。詳細はセクション 7.4.2 と付録 B.2 を参照してください。

### 入力モードDO（ローカル・デジタル出力チャンネル）

入力モードとして **DO** を選択し、**Channel** ボックスから設定したいチャンネルを選択します。選択したデジタル出力チャンネルの値がコンディション入力として使用されます。**Condition** ボックスで **True** を選択すると、選択したデジタル出力チャンネルの値がロジック TRUE のとき、コンディションの結果もロジック TRUE になります。**Condition** ボックスで **[False]** を選択すると、選択したデジタル出力チャンネルの値がロジック False のとき、コンディションの結果はロジック True になります。

### 入力モードカウンタ（内部カウンタ）

各 ADAM-6000 モジュールには 8 つの内部カウンタがあり、その値をコンディション入力として使用できます。これを行うには、入力モードとして “Counter” を選択し、“Channel” ボックスからカウンタのインデックスを選択します（範囲は0～7）。選択した内部カウンタのカウント値が条件入力として使用されます。周波数入力モードと同様に、**Condition** ボックスからカウンタの条件を選択し **Value** ボックスに対応する値を入力する必要があります。コンディションは内部カウンタ値と **Value** ボックスの値を比較します。コンディションが満たされた場合、コンディションの結果は論理真となります（そうでない場合は論理偽）。

## 8.3.2 ロジックステージ

ロジックステージのアイコンをクリックすると、図8.6のようなウィンドウが表示されます。

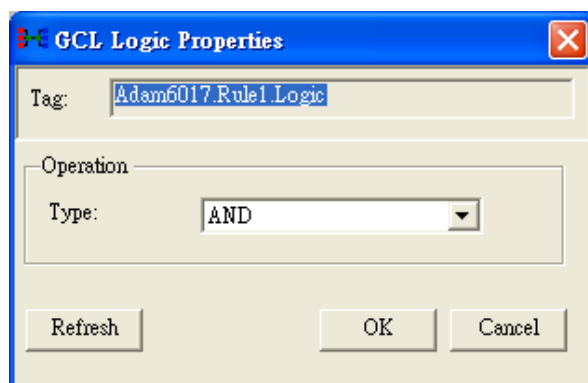


図 8.6 ロジック・ステージ構成

各ロジックルールには、ロジックのTrueまたはFalseの値をロジックステージに渡す最大3つの入力条件があります。タイプボックスから4つの論理演算を選択できます：and、or、nand、nor。ロジック演算は入力ロジック値を処理し、実行ステージに渡されるロジック結果を生成します。適切な論理演算を選択したら、[OK] をクリックします。論理ステージアイコンが現在の論理演算を表すアイコンに変わります。

以下の文章では、4つの論理演算がどのように機能するかを示すために、真理値表が用いられている。T “は論理真、”F “は論理偽を意味する。

#### 論理ゲート：AND

入力条件1	入力条件2	実行ステージに渡されるロジック値
F	F	F
F	T	F
T	F	F
T	T	T

#### 論理ゲート：OR

入力条件1	入力条件2	実行ステージに渡されるロジック値
F	F	F
F	T	T
T	F	T
T	T	T

#### 論理ゲート：NAND (ANDではない)

入力条件1	入力条件2	実行ステージに渡されるロジック値
F	F	T
F	T	T
T	F	T
T	T	F

#### 論理ゲート：NOR (ORではない)

入力条件1	入力条件2	実行ステージに渡されるロジック値
F	F	T
F	T	F
T	F	F
T	T	F

### 8.3.3 実行段階

実行ステージアイコンをクリックすると、図 8.7 のようなダイアログウィンドウが表示されます。Operation] パネルの [Type] ボックスで選択できる実行設定は、[Execution\_Period] (論理結果が出力ステージに渡されるまでに遅延がある) または [SendToNextRule] (入力値がすぐに次のルールに渡される) の2つです。実行設定を選択したら

OK実行ステージのアイコンが現在の設定を表すアイコンに変わります。各タイプの詳細は以下の文章で説明します。

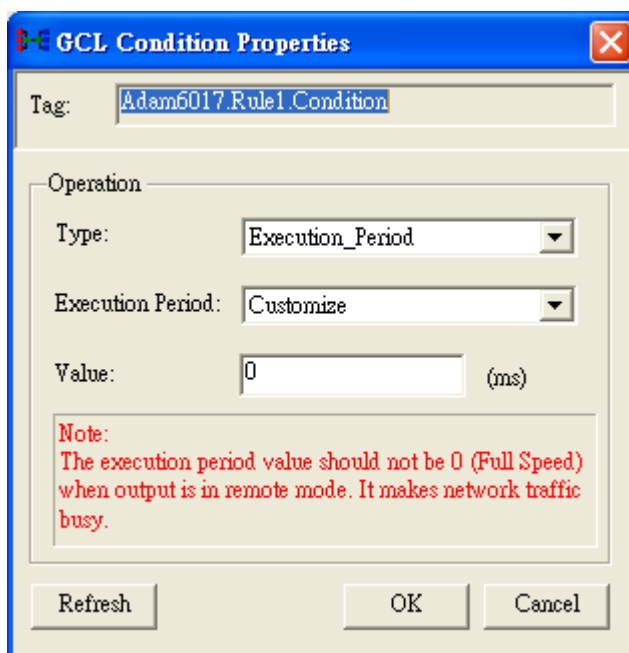


図8.7 実行ステージの構成 実行タイプ実行期間

前述したように、論理ステージは論理結果（すなわち論理真または論理偽）を実行ステージに転送する。実行ステージは特定の期間が経過すると、この値を出力ステージに渡します。この期間の長さを設定するには、以下の手順に従ってください：

1. **Type**ボックスから**Execution\_Period**を選択する。
2. **実行期間**ボックスから適切な期間を選択します。1～60000msの間であらかじめ定義された期間を選択することができます。また、**Customize** を選択して、渡されたテキストに値を入力して期間を定義することもできます（単位は ms）。
3. **OK**をクリックして設定を完了する。

**注**



1. **実行期間**のボックスで**フルスピード**を選択した場合、実行速度は可能な限り速くなります。データ出力先が他のモジュールの場合、ネットワーク通信トラフィックの問題が発生する可能性があります（受信モジュールが処理できる以上のパケットが転送される可能性があります）。

2. *Adam/Apax .NET Utility* を使用して、すでに *GCL* ルールを実行している *ADAM-6000* モジュールを設定する場合は、最初に *GCL* ロジックルールを停止することを忘れないでください。

### 実行タイプSendToNextRule

この設定により、異なるロジックルールを1つのルールにまとめることができ、より複雑なロジック・アーキテクチャの構築に役立ちます。異なるロジックルールを組み合わせるには、次のルールに送る関数を使用する方法と、内部フラグを使用する方法があります。

send to next rule 機能を使用すると、1つの論理ルールの出力が後続の論理ルールの入りに設定されます。これは、同じモジュール上で連続する2つの論理ルールしか組み合わせることができないことを意味します。異なる

ロジックルールが連続しない、または異なるモジュールにある場合、ロジックルール・カスケードの内部フラグを使用する必要があります（これはセクション8.4で紹介します）。

**Type]** ボックスで **[SendToNextRule]** を選択すると、出力アイコンの1つが次のルールになる。図8.8に例を示す。

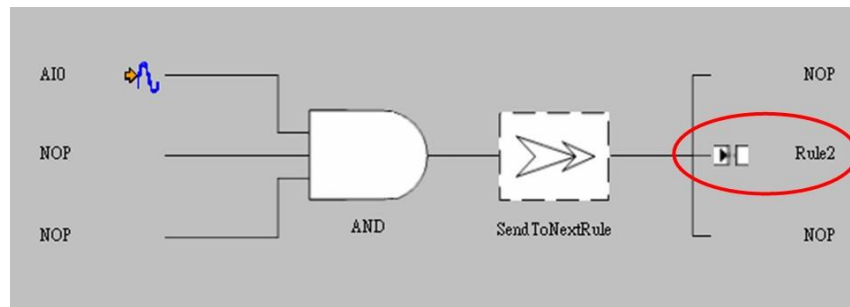


図8.8 次のルールに送る機能

次の論理ルールアイコンをクリックすると、入力条件の1つが前の論理ルールであることがわかります（図8.9では、入力ステージに「Rule1」が表示されています）。したがって、前の論理ルールの論理結果は、現在の論理ルール（この例では「Rule2」）の論理入力値の1つになります。これは、隣接する2つの論理ルールを結合し、論理カスケードと呼ばれます。

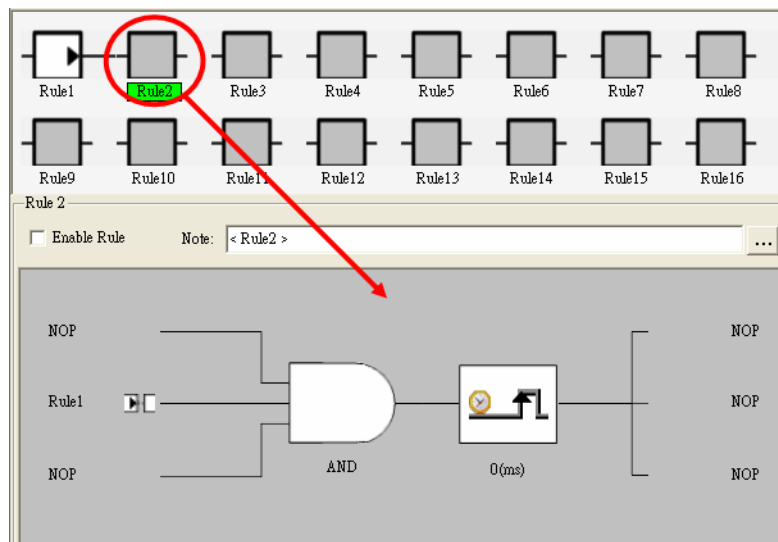


図8.9 次の論理ルール

### 8.3.4 出力ステージ

出力ステージのアイコンをクリックすると、図8.10のようなウィンドウが表示されます。論理ルールごとに3つの出力があります。実行ステージからのロジック結果はこれら3つの出力に渡されます。3つの出力が取るアクションは、ロジック結果によって異なります。

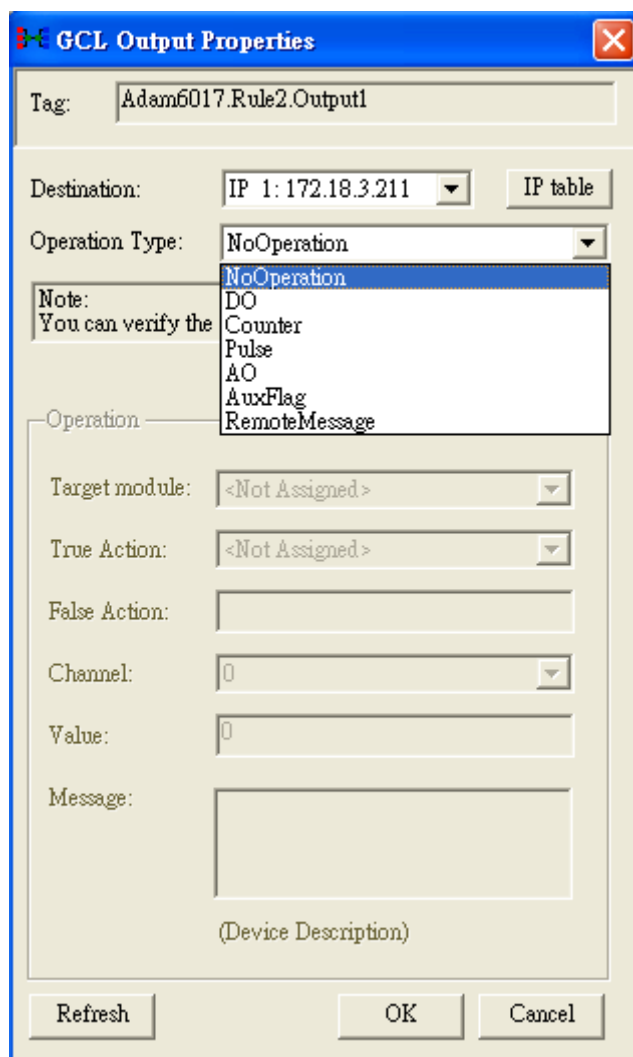



図8.10 出力段の構成

出力ステージを設定するには、まず、**Destination** ボックスから、出力のターゲット・デバイスのアドレスを選択する必要があります。これは出力信号の先を定義します。デスティネーション・ボックスに表示されるIPアドレスによって、**ローカル**（出力が同じモジュール上にあることを意味する）または別のリモート・モジュールを選択できます（IPアドレスはIPテーブルで定義されることに注意してください）。

このウィンドウの**IPテーブル**をクリックするか、GCLメニューエリアのIPテーブルアイコン（）をクリックします。出力モジュールの名前は、**ターゲット・モジュール**・ボックスから選択できます。

#### 注意



出力先がローカルでない場合は、データ・パケットの衝突を防ぐために、イーサネット・スイッチを使用してADAM-6000モジュールをターゲット・デバイスに接続することを忘れないでください（イーサネット・ハブは使用しないでください）。

ターゲットデバイスを決めたら、出力アクションを  
[Operation Type] ボックス。オプションの一覧は以下の通り：

- NoOperation (default)
- AO (analog output)
- DO (digital output)
- DI\_counter (counter channel setting)
- DO\_Pulse (pulse output)
- Timer (local timer)
- AuxFlag (local or remote internal flag)
- RemoteMessage (remote message output)
- Counter (local internal counter setting)

出力アクションを選択した後、**Verify**をクリックしてターゲット・デバイスが存在し、それがGCLをサポートしているかどうかを確認する必要があります（これはNoOperation設定には適用されません）。

[True Action] ボックスでは、実行ステージから渡されたロジック結果がロジック True の場合に実行されるアクションを設定します。**False Action**] ボックスには、実行ステージから渡されたロジック結果がロジック False の場合に実行されるアクションが定義され、定義された True Action に従って自動的に設定されます。

完了すると、出力ステージのアイコンが現在の状態を表すように変わります。各出力アクションを設定する手順は、以下の文章に記載されています。

#### オペレーション・タイプ操作なし

これはデフォルトの設定である。これが選択されている場合、出力アクションはない。

#### 動作タイプAO（アナログ出力）

以下の手順に従って、アナログ出力を設定してください：

1. 操作タイプボックスからAOを選択
2. ターゲットモジュールのボックスからターゲットモジュールを選択します（**Destination** が **Local** に設定されている場合は、この手順をスキップします）。
3. **TargetRange**ボックスから適切な出力範囲を選択する。
4. **Channel**ボックスから、どのチャンネルがターゲットデバイスの出力信号を生成するかを設定。
5. 生成される値を **Value** ボックスに入力して定義します（値の単位は **TargetRange** ボックスの範囲によって異なります）。
6. **OK**をクリックして設定を完了する

#### 注意



アクションの説明は、**True Action/False Action** ボックスで確認できません。論理 TRUE の場合、選択したアナログ出力チャンネルは定義した新しい値を生成します。論理偽の、選択したアナログ出力チャンネルの出力値は変更されません。

### 動作タイプD0（デジタル出力）

以下の手順に従って、デジタル出力を設定してください：

1. 操作タイプボックスからD0を選択
2. ターゲットモジュールのボックスからターゲットモジュールを選択します（Destination が Local に設定されている場合は、この手順をスキップします）。
3. True Actionボックスで、Trueアクションに対してTrueまたはFalseのデジタル出力信号を生成するかどうかを定義します（Falseアクションは自動的にTrueアクションと逆になります）。
4. Channelボックスから、どのチャンネルがターゲット・デバイス上で出力信号を生成するかを定義します。
5. OKをクリックして設定を完了する

### 動作タイプDI\_Counter（カウンター・チャンネル設定）

以下の手順に従って、カウンター・チャンネルの設定を行います：

1. 操作タイプボックスからDI\_Counterを選択
2. ターゲットモジュールのボックスからターゲットモジュールを選択します（Destination が Local に設定されている場合は、この手順をスキップします）。
3. True Action] ボックスから、[Start]（カウンタの開始）、[Stop]（カウンタの停止）、または [Reset]（カウンタのリセット）を選択して、True Action（実行ステージから渡されたロジック結果がロジックTrueの場合）のアクションを定義します。
4. チャンネル・ボックスから、どのカウンター・チャンネルが定義されたアクションを実行するかを定義します。
5. OKをクリックして設定を完了する

### 動作タイプD0\_パルス（パルス出力）

以下の手順に従って、パルス出力を設定してください：

1. 操作タイプボックスからD0\_Pulseを選択する。
2. ターゲット・モジュール・ボックスからターゲット・モジュールを選択
3. True Action] ボックスから、[Continue]（継続的にパルス列を生成する）、[Stop]（パルスの生成を停止する）、または [Num of pulse]（一定数のパルスを生成する）を選択して、どのようなTrueのアクションを実行するか（実行ステージから渡されたロジック結果がロジック TRUE の場合）を定義します。フォールス・アクションは常に現在のステータスを保持し、選択されたデジタル出力チャンネルのアクションは変更されないことに注意してください。
4. Channelボックスから、どのデジタル出力チャンネルが定義されたアクション（パルス生成の開始または停止）を実行するかを定義します。
5. Num of pulse（パルス数）」が選択されている場合は、「Value（値）」ボックスにパルス数を入力します。
6. OKをクリックして設定を完了する

### 動作タイプタイマー（ローカルタイマー）

以下の手順でタイマーを設定する：

1. 操作の種類] ボックスで [タイマー] を選択する。
2. どのタイマーを設定するかは、OperationのIndexボックスから選択します。  
パネル（各 ADAM-6000 モジュールには 16 個のローカルタイマーがあります。）
3. 操作パネルの[タイプ]ボックスから、ON-Delay（ロジック結果がロジックtrueのときにタイマーを開始し、ロジックfalseのときにカウントを停止して値をゼロにリセットする）またはOFF-Delay（ON-Delayの逆）を選択し、タイマーの動作を定義します。
4. OKをクリックして設定を完了する

### 操作タイプ : AuxFlag (ローカルまたはリモート内部フラグ)

以下の手順に従って、実行ステージのロジック結果をローカルまたはリモートの内部フラグに割り当ててください :

1. **操作タイプ**ボックスから**Auxflag**を選択する。
2. **インデックス**ボックスから、設定したい内部フラグを選択します。
3. **True Action**ボックスから、Trueアクションの内部フラグに割り当てたい値を定義する (FalseアクションはTrueアクションの反対となる) 。
4. **OK**をクリックして設定を完了する

### 操作タイプRemoteMessage (リモートメッセージ出力)

デバイスの説明をメッセージとしてターゲット・デバイスに送ることができる。

1. **操作の種類**ボックスから**[リモートメッセージ]**を選択する。
2. **Value** ボックスに値を入力して、メッセージにインデックスを付けます (複数のロジックルールがメッセージを送信する場合、どのロジックルールが ターゲットメッセージを送信するかを指定することが重要 です) 。
3. **メッセージ**ボックスに送信したいメッセージを入力してください。
4. **OK**をクリックして設定を完了する

この操作タイプでは **[True Action]** ボックスを設定する必要はありません。ロジックの結果がロジックtrueの場合、メッセージはターゲット機器に送信されます。論理結果が論理偽の場合、メッセージは送信されません。

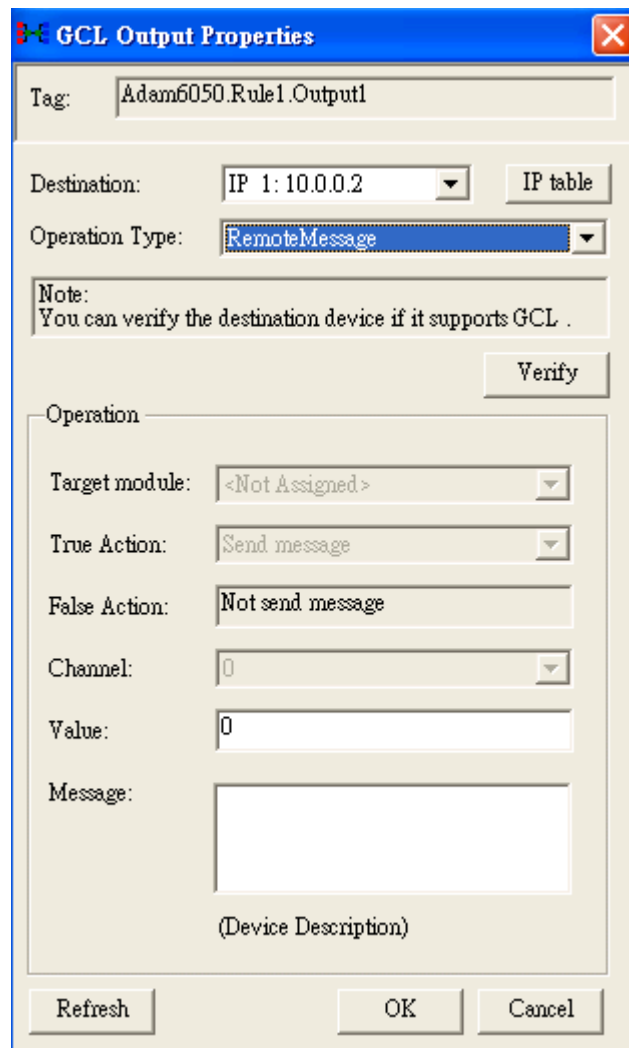


図8.11 リモート・メッセージ出力

**注意**

送信されるメッセージは、メッセージ・ボックスの内容（デバイスの説明）、メッセージを送信するロジックルールの番号、メッセージ・インデックス、モジュールの IP アドレス、モジュール名、すべての I/O ステータスで構成されます。

**動作タイプカウンタ（ローカル内部カウンタ設定）**

以下の手順に従って、内部カウンタの設定を行います：

1. 操作タイプボックスからカウンタを選択
2. True Action ボックスから、Positive edge trigger (F→T)（内部カウンタを 1 インクリメント）または Reset（内部カウンタをリセット）を選択して、True Action に対してどのようなアクションを取るかを定義します。
3. False Action ボックスに表示される False Action は、自動的に True Action の反対側になる。下の表を参照して、True のアクションと False のアクションの関係を確認してください。
4. チャンネル・ボックスから、どのカウンタ・チャンネルが定義されたアクションを実行するかを定義します。
5. OK をクリックして設定を完了する

**注**

注：正エッジトリガ (F→T) を動作として選択した場合、カウンタは実行ステージからのロジック結果がロジックハイである最初の1回のみ1カウント加算されます。初回以降は、実行ステージからのロジック結果がロジックハイのままであっても、カウンタ値は変化しません。これがポジティブ・エッジ・トリガーと呼ばれる理由です。

次の表は、さまざまな出力アクションの true アクションと false アクションを示している：

Output Action	True action	False action
No Operation	Do nothing	Do nothing
AO	Change the analog output value	Keep current status
DO	Output true value	Output false value
	Output false value	Output true value
DI_Counter	Start counter	Stop counter
	Stop counter	Start counter
	Reset counter	Do nothing
DO_Pulse	Generate a continuous pulse train	Keep current status
	Generate a finite number of pulses	
	Stop generation pulses	
Timer	Start counting time	Stop timer and reset value to zero
	Stop timer and reset value to zero	Start timer
Internal Flag	Assign true value to a flag	Assign false value to a flag
	Assign false value to a flag	Assign true value to a flag
Remote Message	Send a message to the target device	Do nothing
Counter	Increment the counter by 1	Do nothing
	Reset counter	

## 8.4 ロジック・カスケードおよびフィードバック用内部フラグ

### 8.4.1 ロジック・カスケード

内部フラグをインターフェイスとして使用することで、異なるロジックルールを組み合わせて1つのロジックルールとし、より複雑なロジック・アーキテクチャを構築することができます。ロジックルールは同じモジュール上でも、異なるモジュール上でも組み合わせることができます。内部フラグの動作を理解するには、このセクションの例を参照してください。

#### ローカル・ロジック・カスケード

ここでは簡単な例でロジック・カスケードを説明します。ADAM-6017の2つのアナログ入力チャンネル（チャンネル0と1）を使ってセンサからの信号を測定します。2つの入力チャンネルのどちらかが3~5Vの電圧信号を読み取る限り、デジタル・チャンネル0はロジック・ハイ生成します。そうでない場合、チャンネルはロジック・ロー値を生成します。図8.12に論理アーキテクチャを示す。

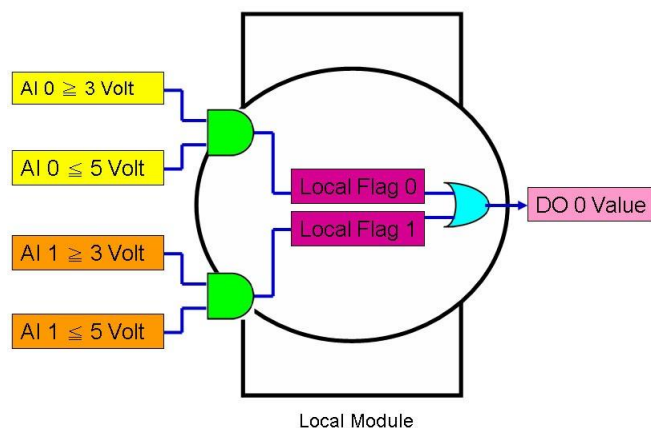


図8.12 ローカル・ロジック・カスケード・アーキテクチャ

この論理アーキテクチャを実現するには、3つの論理ルールと2つの内部フラグを使用する必要があります。3つの論理ルールの設定方法については、図8.13~図8.15を参照されたい。

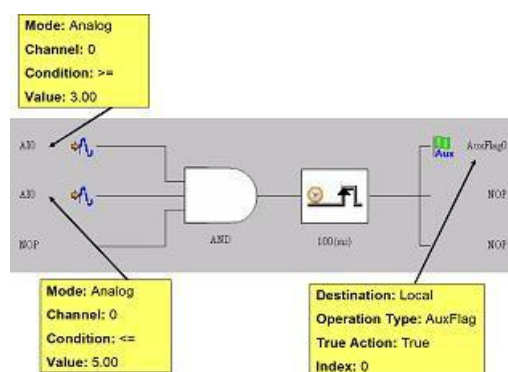


図8.13 論理ルール1の構成

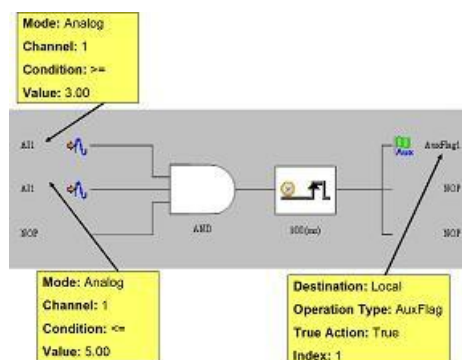


図8.14 論理ルール2の構成

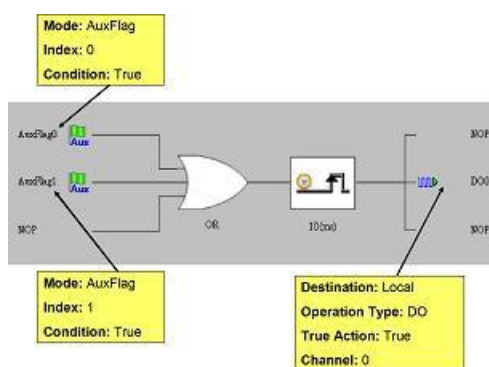


図 8.15 論理ルール3の構成

論理ルール 1 を使用して、ADAM-6017 のアナログ入力チャンネル 0 の値が 3~5 V 以内であるかどうかをチェックします。論理ルール 1 と 2 の比較結果は内部フラグ 0 と 1 に割り当てられ、論理ルール 3 はこれら 2 つの内部フラグの値を読み出し、OR 論理演算を使用してデジタル出力チャンネル 0 の出力を定義します。図8.12に示すように、この論理アーキテクチャは内部フラグを使用して構築されている。

### 分散ロジック・カスケード

ロジック・カスケード関数は、1 つのモジュールに限定されません。内部フラグを別のモジュールに定義できるため、ロジック・カスケード構造は異なるモジュールにまたがることができます。前のアプリケーションを例にすると、図8.16はロジックルール1~3がモジュールA~Cで動作している様子を示しています。ロジック構造は3つのADAM-6000モジュールにまたがっており、これを分散ロジック・カスケードと呼びます。3つのロジックルールの構成を図8.17~8.19に示します。

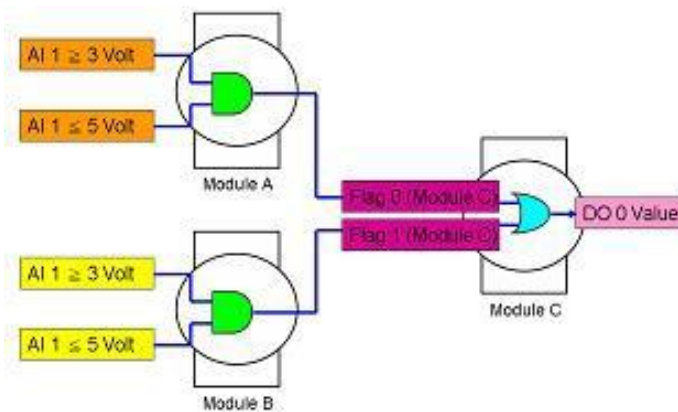


図8.16 分散ロジック・カスケード

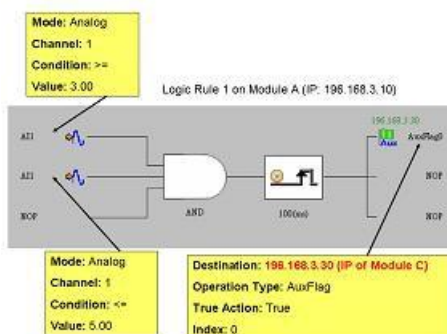


図 8.17 論理ルール1の構成

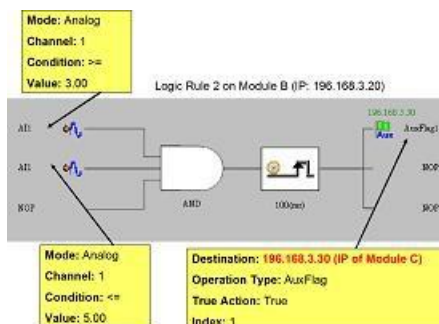


図 8.18 論理ルール2の構成

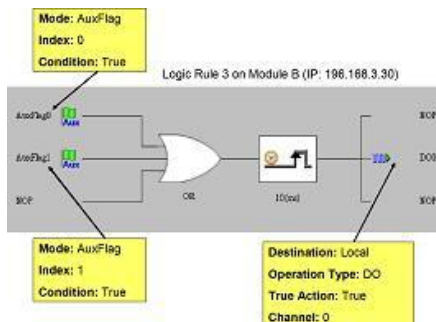


図 8.19 論理ルール3の構成

ローカルまたは分散ロジック・カスケード・アーキテクチャを採用する場合、ロジックルールの入力数に制限はありません。このため、アプリケーションの要件を満たす任意のロジックアーキテクチャーを構築できます。

#### 8.4.2 フィードバック

1つの論理ルールの入力条件と出力に同じ内部フラグを選択すると、その論理ルールには論理フィードバック機能があります。図8.20の例では、1つの入力条件と1つの出力が同じ内部フラグ（AuxFlag 0）に専用化されています。従って、現在の実行の出力値は次の実行の入力になります。これにより、この論理ルールにはフィードバック能力が与えられます。

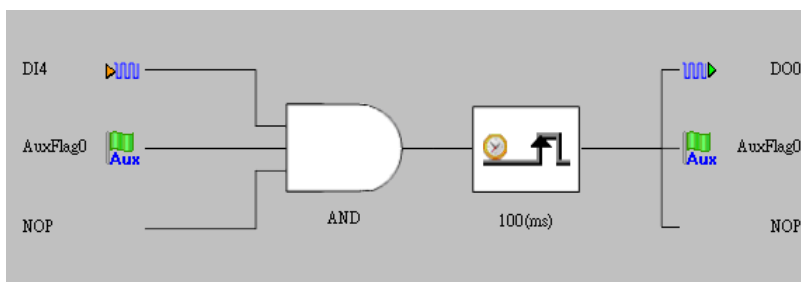





図 8.20 ロジック・フィードバックの構築


## 8.5 ロジック・ダウンロードとオンライン・モニタリング

GCLロジックルールのコンフィギュレーションが全て完了したら、コンフィギュレーション全体をダウンロードするために、GCLメニューエリアのDownload Projectアイコンをクリックします。

レーションをターゲット・デバイスに送信します。次に、ターゲットモジュール上でプロジェクトを実行するために、Run GCL アイコンをクリックします。現在のステータスが“Running”（実行中）に切り替わります。

モードアイコン .

ADAM-6000モジュールは、特別なオンライン・モニタリング機能を備えています。実行モードでは

GCL Menu AreaのMonitoringアイコンをクリックして、この機能を有効にします。そうすると、“Individual Logic Rule Configuration Area”に実行ステータスが表示されます。黄色の点は、プロセスフローがどの段階にあるかを示します。現在の入力値も、“Input Condition Stage Area”の横に表示されます。図

8.21 に、オンライン・モニタリング機能の動作例を示す。この例では、DI 1とDI 3の入力条件が満たされているため、2つの入力条件アイコンの横に黄色い点が表示され、3つの入力段階アイコンの上に現在の入力値が表示されています。

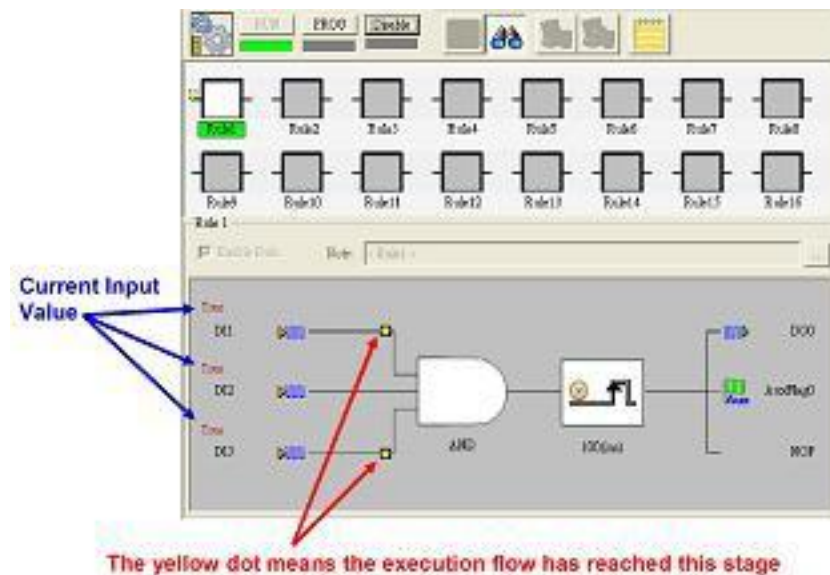


図 8.21 オンライン・モニタリング機能

**注意** 内部フラグ (AuxFlag) を GCL 論理ルール入力として使用する場合、Adam/Apax .NET Utility のオンラインモニタウィンドウでフラグ値を動的に変更することができます。内部フラグを表す入力アイコンをダブルクリックすると、フラグの値が true から false に変更されます。

### GCLルール実行シーケンス

各 ADAM-6000 モジュールには 16 個のロジックルールがあります。図 8.22 に 1 サイクルの実行フローを示します。この図では、1 サイクルには、1) 入力条件 + ロジック、2) 実行、3) 出力の 3 つのグループがあります。入力条件+論理の段階で有効化されたすべてのルールが最初に実行され、次に実行段階で有効化されたすべてのルールが実行され、最後に出力段階で有効化されたすべてのルールが実行されます（すべての段階のすべてのルールが順番に実行することに注意してください）。

一部の高度なアプリケーションでは、ロジック・カスケード・アーキテクチャ（セクション 8.4.1 参照）を採用して異なるルールを組み合わせることができます。たとえば、両方のルールに同じ内部フラグを割り当てることで、ルール 1 の出力をルール 2 の入力に接続できます。前述の実行フローに基づき、ルール 1 の入力条件+論理、実行、出力の各ステージが順次実行される。従って、最初のサイクルの出力段階でルール1の出力が更新され、次のサイクルでルール2の入力がルール1の出力の変化を検出することになる。

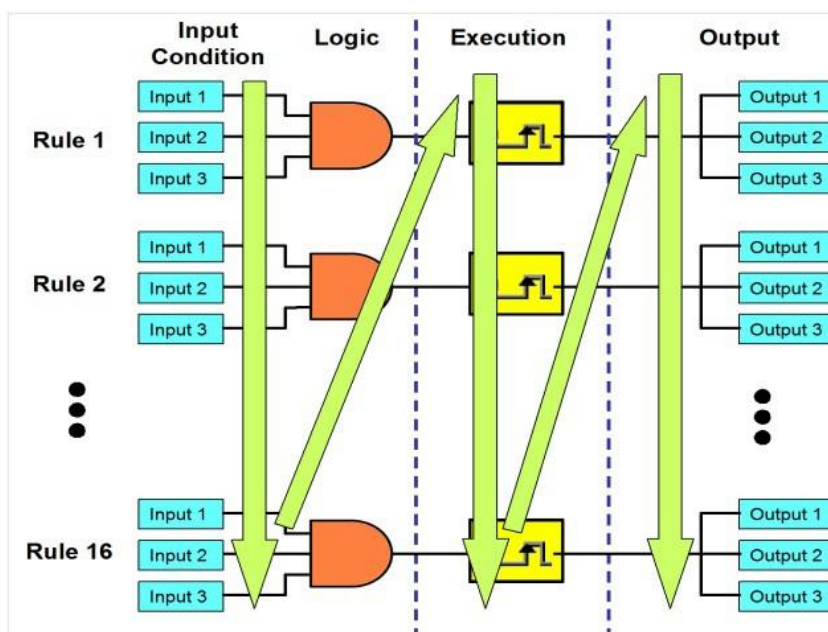


図8.22 GCL実行シーケンス

### GCL実行とデータ転送のパフォーマンス

#### ローカル出力（ローカル・カスケード）

条件1つのADAM-6050モジュールで1つのロジックルールを実行する

処理時間：<1ms（ハードウェア入力遅延時間、1つの論理ルールの実行時間、ハードウェア出力遅延時間を含む）

複数の論理ルールを使用する場合、処理時間は以下の式で見積もることができる：

$$\text{論理ルール数 } n \leq 16$$

$$\text{おおよその処理時間（1サイクル）} = 600 + n \times 370 \text{（秒）}$$

#### リモート出力（分散カスケード）


条件1つのADAM-6050で1つのロジックルールを実行し、出力はイーサネット・スイッチを介して別のADAM-6050モジュールに送る。

処理+通信時間：<3 ms

## 8.6 GCLの代表的な用途

GCLの設定時間を短縮するために、アドバンテックはいくつかの典型的なアプリケーションのサンプル・プロジェクト・ファイルを提供しています。これらのアプリケーション・サンプル・ファイルは、NETクラス・ライブラリに含まれており、インストール後にHDD上で利用可能です。デフォルトの場所は以下の通りです：

GCL Example Project.

GCLメニューエリアのプロジェクトコンテンツアイコン  クリックして、これらのファイルをロードするだけです。これらのサンプルプロジェクトは、アプリケーションの要求に応じて簡単に変更することができます。修正したプロジェクトをモジュールにダウンロードし、実行します。

かわいい。各プロジェクト例は以下の文章で詳しく説明されている：

## 空のプロジェクト

GCLのすべてのコンフィギュレーションをクリアしたい場合、シンプルな方法はこのサンプルプロジェクトをロードすることです。そうすれば、すべてのコンフィギュレーションを手動でクリアする必要はありません。

## オン/オフコントロール（2つのボタンでオンとオフを別々にコントロール）

いくつかのオートメーション・アプリケーションでは、2つのデジタル入力（例えば、DI 0とDI 1）が1つのデジタル出力ステータス（例えば、DO 0）を制御するために使用されます。DI0がロジックハイになるとDOステータスはロジックハイになり、DI1がロジックハイになるとDOステータスはロジックローに戻ります。たとえば、モータの動作を2つのボタンで制御する場合、最初のボタンが押されるとモータが始動し、2番目のボタンが押されるとモータが停止する。この種の産業用オートメーションアプリケーションには通常 PLC が使用され、ラダーダイアグラムは図 8.23 のようになります。

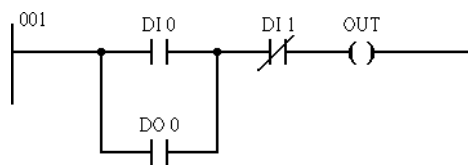


図8.23 オン/オフ制御のラダー図

今、GCL論理を利用して同じ制御動作を実現することができる。このために、2つの論理ルールが使用される。完全な論理アーキテクチャを図8.24に示す。

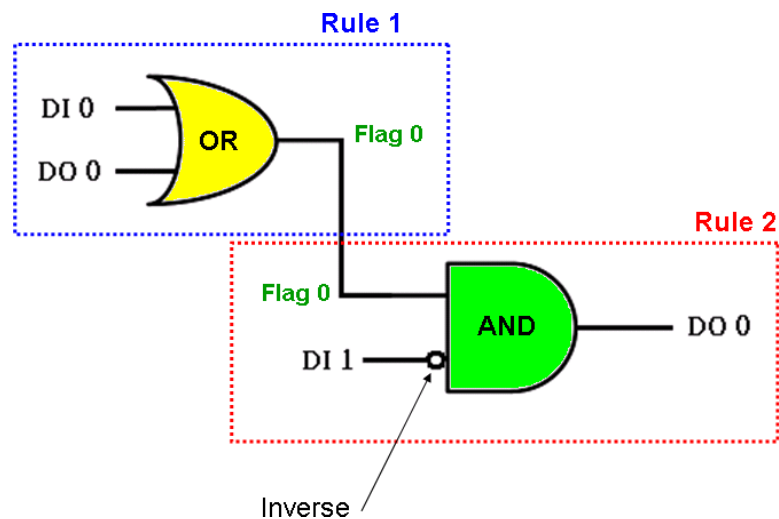


図8.24 オン/オフ制御のGCLロジック

サンプル・プロジェクト・ファイルをロードすると、ルール1とルール2が使用されていることがわかる。ルール 1 の 1 つの出力とルール 2 の 1 つの入力は、同じインターナル・フラグに割り当てられています：これにより、2 つ以上のロジックルールを組み合わせてロジック・カスケードを構成できます（セクション 8.4.1 を参照）。この例では、DI 1の条件がFalseであるため、ルール2のAND演算子に入る前にディジタル入力値が反転されます。GCLロジックアーキテクチャは、図8.23のPLCラダー図に似ています。

### シーケンシャル制御（順番に点灯し、点灯し続ける）

このタイプのアプリケーションでは、複数のデジタル出力が順番にアクティブになり、その値をラッチします。このサンプル・プロジェクトでは、DO 0～DO 5 を順次制御してステータスを変更します。このアプリケーションのタイムチャートを図8. 25に示します。

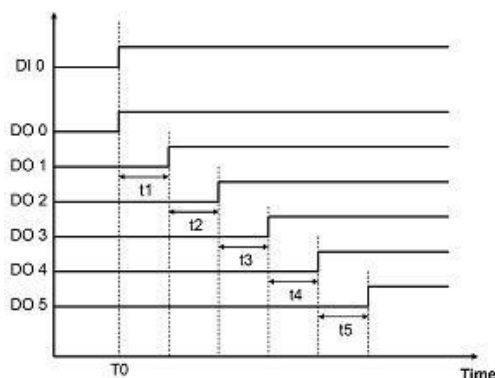


図8. 25 シーケンス制御のタイムチャート（デジタル出

### 力はシーケンスでオンになり、オン状態を維持する）

例のプロジェクトでは、DI 0がシーケンシャルな制御動作を開始するトリガとして使用される。したがって、DI 0がロジックハイになると（T0で）、そのDO 0も即座にロジックハイになる。その後、DO1～DO5が特定の時間間隔をおいて順次ロジックハイになります。t1～t5の時間間隔はあなたが定めることができます（一意な値でかまいません）。このサンプル・プロジェクトでは、t1～t5はすべて5秒に設定されています。

このGCLアプリケーションには、6つの論理ルールと1つの内部タイマーを採用できる。最初の論理ルールでは、DI 0がタイマー0とDO 0のトリガーに使用される。タイマーがトリガされたので、カウンタが開始され、一定時間経過後にDO 1～DO 5が起動される。GCLアーキテクチャを図8. 26に示す。

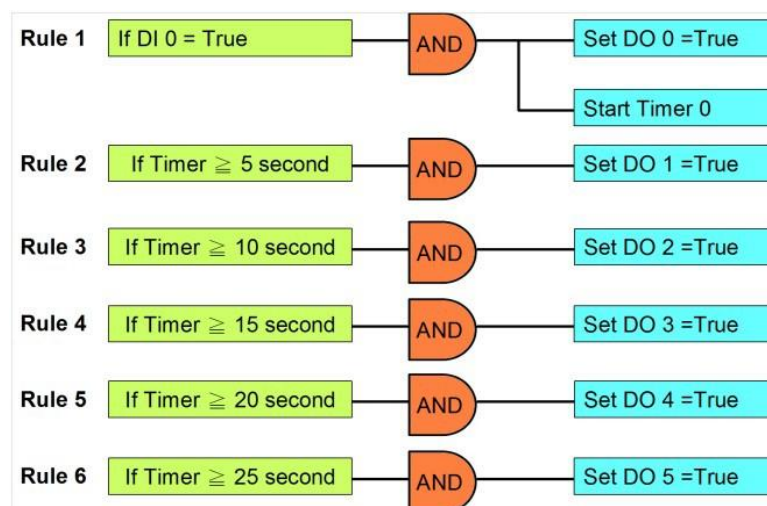


図8. 26 シーケンス制御のGCLロジック（シーケンスでオンし、オン状態を維持する）

### 複数のデジタル入力から1つのデジタル出力を制御（12デジタル入力から1デジタル出力）

多くのアプリケーションでは、複数のデジタル入力がロジックハイになったとき（つまり、関連するすべての条件が満たされたとき）だけ、デジタル出力のステータスがロジックハイになります。この例では、DI 0～DI 11がすべて同時にロジックハイになったときだけ、DO 0がロジックハイになります。このアプリケーションのタイムチャートを図8. 27に示します。緑色の帯域は、12個のデジタル入力がすべてロジックハイになった瞬間を示します。

この時点でD0 0もロジックハイになる。他のすべての時点では、少なくとも1つのデジタル入力チャンネルがロジック・ハイでないため、D0 0はロジック・ローとなる。

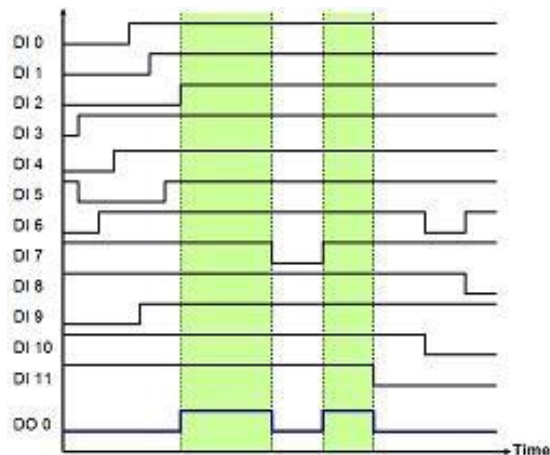


図8.27 12デジタル入力対1デジタル出力のタイムチャート

この制御システムを実現するには、単純にAND論理演算子を1つ実装すればよい。しかし、1つの論理ルールには3つの入力しかないため、12個の入力を持つためには論理カスケードを実装する必要があります。論理カスケードを実現するには2つの方法があります：

1. ある論理ルールの実行段階で **SendtoNextRule** を選択する。これにより、このロジックルールは後続のロジックルールに接続されます（セクション8.3.3参照）。
2. ある論理ルールの出力と別の論理ルールの入力を同じ内部フラグに割り当て、2つの論理ルールを組み合わせる（セクション8.4.1参照）

最初の方法では、2つの論理ルールは連続していなければならない。例えば、ルール1とルール2は一緒に組み合わせることができますが、ルール1とルール3はできません。2番目の方法を使用する場合、この制限は適用されません。2番目の方法を使用すると、異なるモジュール上の2つのルールを組み合わせることもできます。このサンプル・プロジェクトでは、最初の方法をロジック・カスケードに適用します。GCL ロジック・アーキテクチャを図 8.28 に示します。

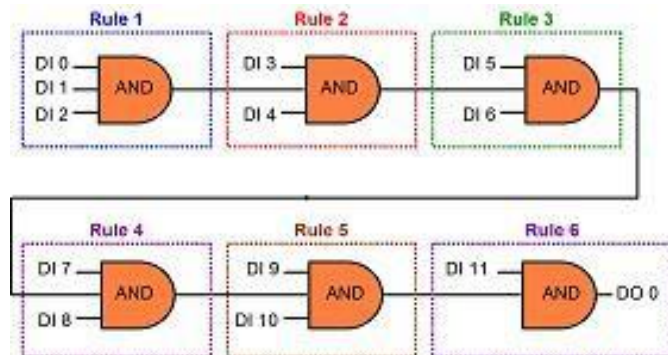


図8.28 12デジタル入力対1デジタル出力のGCLロジック

## フリッカー

フリッカーは、オートメーション制御アプリケーションでよく使用されます。典型的な例は、デジタル出力の制御下でアラームを点滅させることです。このアプリケーションでは、デジタル出力チャンネルで連続パルス列を生成する必要があり、決定する必要があるのはパルス列の周期だけです。点滅アプリケーションのタイムチャートを図8.29に示します。

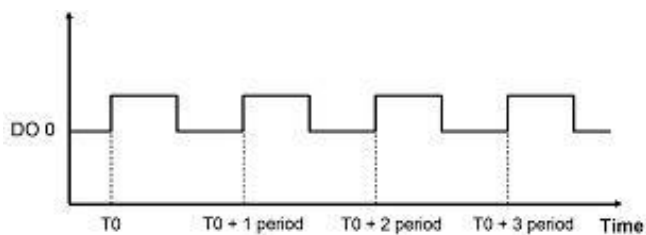


図 8.29 フリッカー・アプリケーションのタイムチャート

このフリッカー・アプリケーションでは、1つの内部フラグ (Flag 0) と2つの論理ルールを使用する必要があります。論理ルール 1 では、フラグ 0 の値を周期的に反転させます (論理ステージで NAND を選択)。この例では、周期は0.5秒で、これは実行段階で Execution\_Periodを選択することで定義されます (セクション8.3.3参照)。DO 0の状態は論理ルール2のフラグ0によって制御されるため、DO 0は0.5秒ごとに変化する。

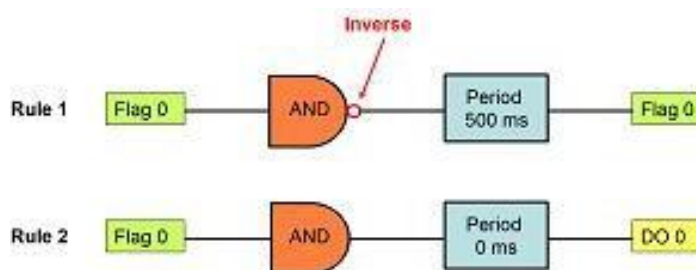


図8.30 フリッカーのGCLロジック

## 立ち上がりエッジ

立ち上がりエッジ・アプリケーションの場合、デジタル入力値がロジック・ローからロジック・ハイに変化すると (すなわち、立ち上がりエッジが発生すると)、デジタル出力ステータスはロジック・ハイになります。しかし、デジタル出力値は継続的にロジック・ハイのままではなく、特定の時間間隔 (この例では1秒) が経過すると、デジタル出力値はロジック・ローに戻ります。この例のタイムチャートを図8.31に示す。

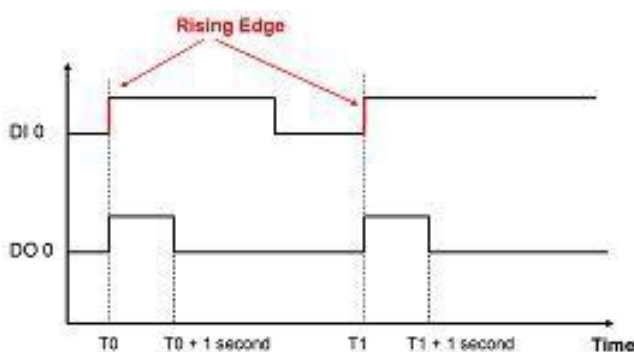


図 8.31 立ち上がりエッジのタイムチャート

この図はDI 0が立ち上がりエッジを示したときにのみDO 0がトリガーされることを示している。この例のプロジェクトでは、デジタル出力ステータスは 1 秒間ロジックハイのままでその後ロジックローに戻ります。このタイプのアプリケーションに PLC を使用する場合、ラダー図は図 8.32 に示すようになります。

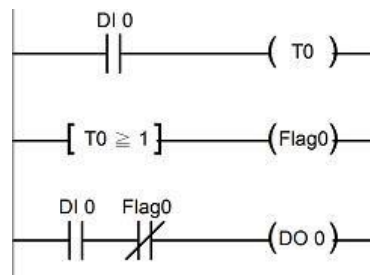


図8.32 立ち上がりエッジのラダー図

立ち上がりエッジ・アプリケーションにGCLを使用するには、3つの論理ルール、1つの内部タイマー (Timer 0)、1つの内部フラグ (Flag 0) を使用する必要があります。GCL論理アーキテクチャは図8.33を参照してください。論理ルール3では、DO 0の値はDI 0とフラグ0によって制御されます。

立ち上がりエッジが発生すると（すなわち、デジタル入力値がロジック・ローからロジック・ハイに変化すると）、デジタル出力がアクティブになり（すなわち、論理ルール3が満たされる）、タイマー0がカウントを開始する（論理ルール1が満たされる）。タイマー0が特定の時間間隔（この例では1秒）に達すると、論理ルール2の結果、フラグ0が論理真になり、DO0の値が論理ローになる（すなわち、論理ルール3は満たされない）。GCLアーキテクチャは図8.32のラダー図に似ている。

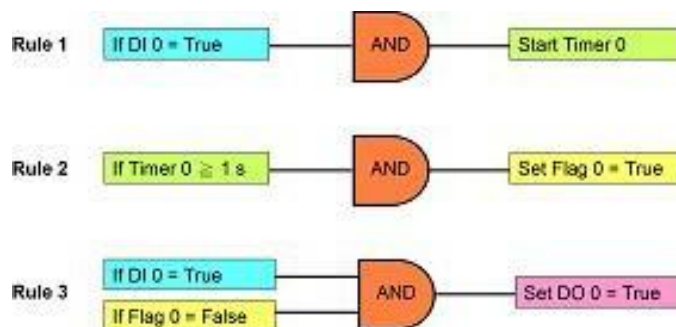


図 8.33 立ち上がりエッジ用 GCL ロジック

### フォーリングエッジ

立ち下がりエッジ・アプリケーションの場合、デジタル入力値がロジック・ハイからロジック・ローに変化するとき（すなわち、立ち下がりエッジが発生するとき）、デジタル出力値はロジック・ハイに設定されます。しかし、デジタル出力値は継続的にロジック・ハイのままではありません。その代わりに、特定の期間（例では1秒）が経過すると、デジタル出力値はロジック・ローに戻る。タイムチャートについては図8.34を参照してください。

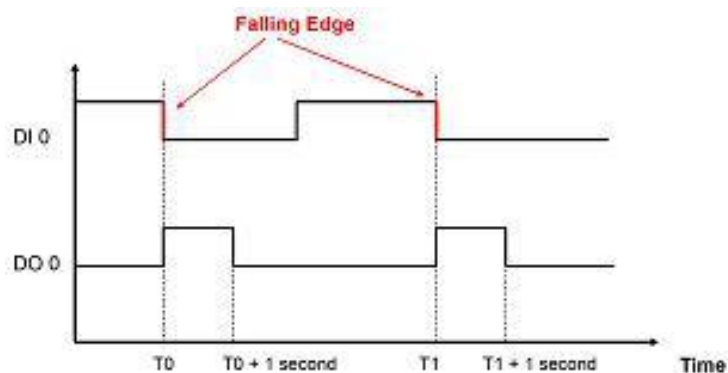


図 8.34 立ち下がりエッジのタイムチャート

図に示すように、DO 0はDI 0が立ち下がりエッジを示したときのみトリガされる。この例のプロジェクトでは、デジタル出力ステータスは 1 秒間ロジックハイのままロジックローに戻ります。この種のアプリケーションに PLC を使用する場合、ラダーダイアグラムは図 8.35 に示すようになります。

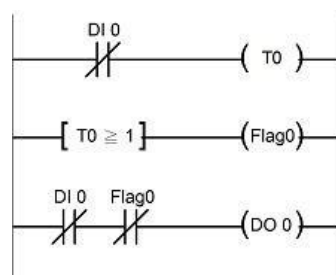


図8.35 立ち下がりエッジのラダー図

立ち下がりエッジ・アプリケーションにGCLを使用するには、3つの論理ルール、1つの内部タイマー（タイマー0）、1つの内部フラグ（フラグ0）が必要です。GCL論理アーキテクチャは図8.36を参照してください。論理ルール3では、DO 0の値はDI 0とフラグ0によって制御されます。

立ち下がりエッジが発生すると（すなわち、デジタル入力値がロジック・ハイからロジック・ローに変化すると）、デジタル出力がアクティブになり（すなわち、論理ルール3が満たされる）、タイマー0がカウントを開始します（論理ルール1が満たされる）。タイマー0が特定の時間（この例では1秒）に達すると、論理ルール2の結果、フラグ0が論理真となり、DO0の値は論理ローとなる（論理ルール3は満たされない）。GCLアーキテクチャは図8.35のラダー図に似ています。

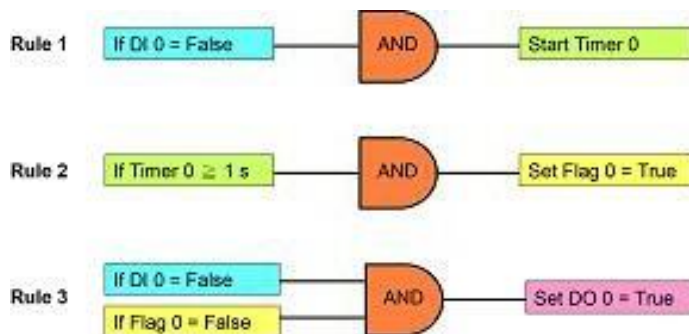


図 8.36 立ち下がりエッジの GCL ロジック

### シーケンシャル制御 (連続的にオン・オフを繰り返す)

このタイプのオートメーション・アプリケーションは、このアプリケーションが連続ループを形成することを除けば、前のシーケンシャル制御アプリケーション (すなわち、このセクションで紹介する3番目のアプリケーション) と同様である。このアプリケーションのタイムチャートを図に示します。

8.37. デジタル出力シーケンスの制御には、1つのタイムベースが必要である。この例では、1つのデジタル出力をオフにし、後続のデジタル出力をオンにするタイムベースの周期は1秒である。

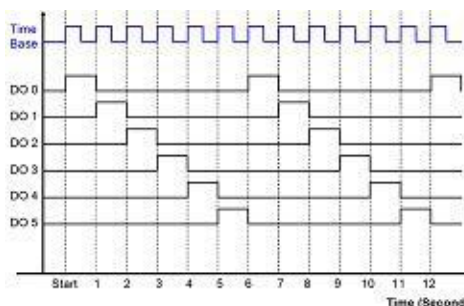


図8.37 シーケンス制御のタイムチャート (オンとオフを繰り返す)

このタイプのアプリケーションには、9つの論理ルール、1つの内部カウンタ (カウンタ0)、および1つの内部フラグ (フラグ0) が必要です。このサンプル・プロジェクトでは、タイム・ベースを作成するために論理ルール1と8が採用されています。ロジックルール8によって、フラグ0の値は0.5秒ごとに変化します。ロジックルール1では、フラグ0の値がロジック・ハイになると、カウンタ0が1ずつ増加します。したがって、1秒ごとにカウンタ0が1ずつ増加し、カウンタ0がタイム・ベースとなります。

どの論理ルールが実行されるかは、カウンタ0の値によって決まる。カウンタ0の値は1秒ごとに1ずつ増加し続けるので、論理ルール9~14は1秒ごとに順番に実行される。論理ルール15が終了すると、カウンタ0はリセットされ、値はゼロに戻る。これにより、論理ルールの実行は連続ループとなる。GCLアーキテクチャは図8.38を参照されたい。

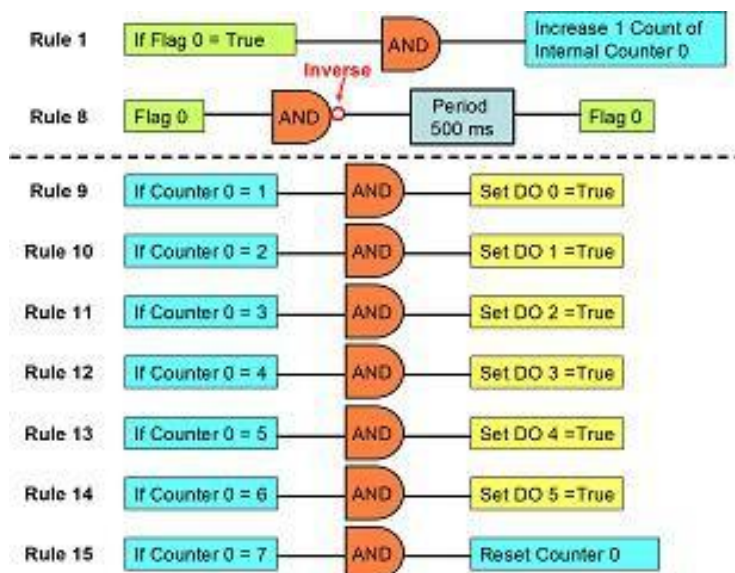


図8.38 シーケンス制御のGCLロジック (連続してオンオフ)

### デジタル入力イベントトリガー（1回のみ発生）

GCLは、イベントトリガーを実行するために採用することができる。このタイプのアプリケーションでは、デジタル入力チャンネルがアクションをトリガーするために使用されます。GCL論理ルールの入力条件は、デジタル入力値が論理真である場合であり、ルールの出力は希望するアクション（この場合、リモートメッセージの送信）である。デジタル入力値が論理真になると、入力条件は満たされます。GCL論理ルールはデジタル入力値が論理偽に戻るまでメッセージを送信し続けます。しかし、この特定のアプリケーションでは、メッセージは連続的に送信されるのではなく、条件が満たされたときに一度だけ送信されます。

この種のアプリケーションは、1つのカウンタ入力チャンネルを使用することで実現できる。GCLロジックを図8.39に示します。1つの論理ルールに対して、入力条件ステージでローカル・カウンタ入力チャンネル（DI\_Counter）を選択するだけです。1つはカウンタ入力チャンネルをリセットする出力で、もう1つ希望する動作です。そして、カウンタ入力チャンネルがデジタル入力信号を検出すると、条件が満たされ、所望のアクションが実行される。同時に、GCLルールはカウンタ入力チャンネルをリセットするため、所望の動作は一度だけ実行される。

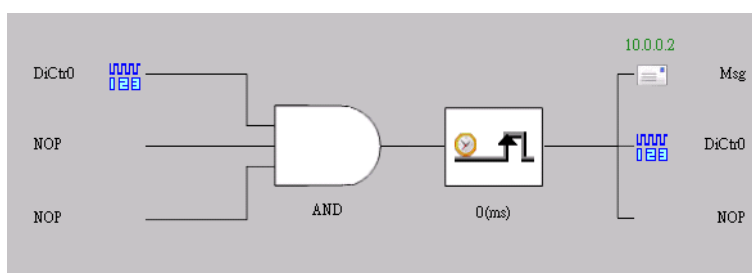


図 8.39 イベントトリガーの GCL ロジック（1 回のみ発生）

Adam/Apax .NET UtilityのGCLロジックルールのコンフィギュレーションのイメージを図8.40に示します。この例では、必要なアクションはリモートメッセージを送信することです。

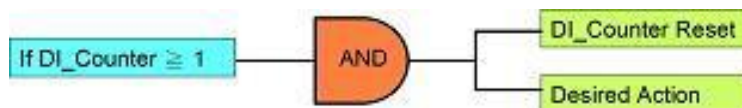


図8.40 イベントトリガー設定（1回のみ発生）



付録

A

デザイン・ワークシート







# 付録

# B

データ形式と入出力範囲

## B.1 ADAM-6000 コマンドデータ形式

ADAM-6000モジュールは、コマンド・レスポンス形式でホスト・コンピュータと通信できます。データが送信されていないとき、モジュールはリッスンモードになります。各モジュールには固有のアドレスが割り当てられます。システムにコマンドを発行する場合、ホスト・コンピュータはこれらのアドレスを使用して特定のモジュールと通信し、応答を待ちます。何も検出されない場合、タイムアウトが発生し、シーケンスは中断され、制御はホストに戻されます。

### 司令部体制

Modbus/TCPネットワーク上のModbusリクエストまたはレスポンスのカプセル化を理解することは重要です。完全なコマンドは“コマンド・ヘッド”（Modbus アプリケーション・プロトコル・ヘッダ）と“コマンド・ボディ”（プロトコル・データ・ユニット）で構成されます。コマンド・ヘッドは6バイトで始まり、Modbusデータ・パケット・フォーマットに従います。コマンド・ボディはターゲット・デバイスと要求されたアクションを定義します。次の例はこの構造を理解するのに役立ちます。

例

ADAM-6017 の最初の 2 つの値（アドレス 40001~40002）を読み出すためのリクエストは以下のように構成される：

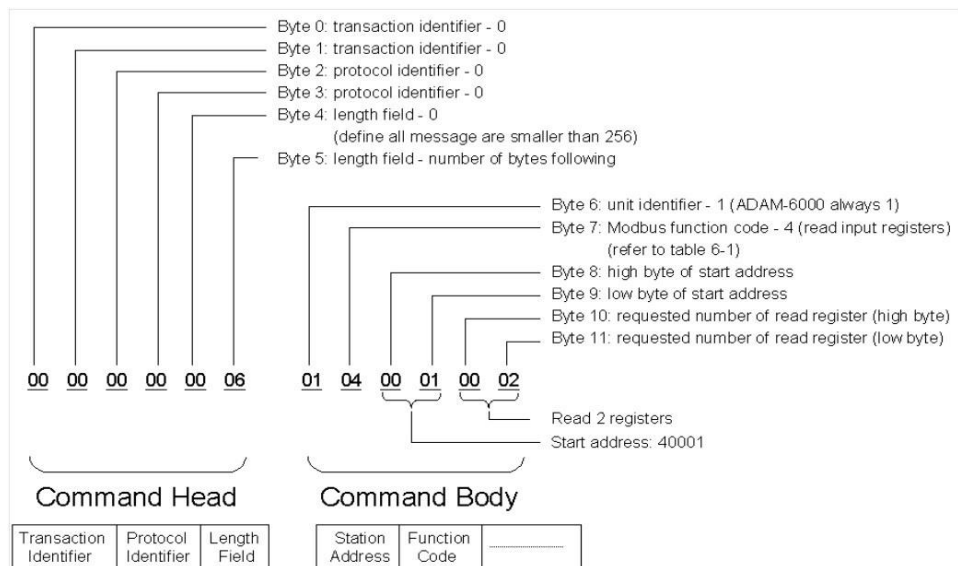


図 B.1 リクエストコメントの構造

このリクエストに対するレスポンスは以下ようになる：

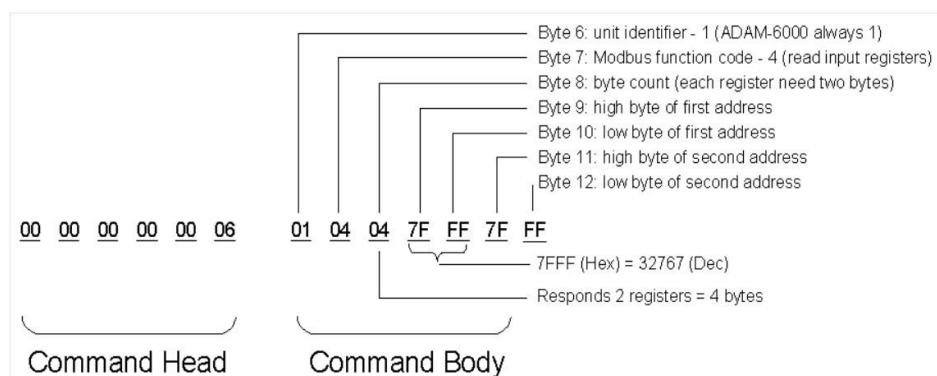


図 B.2 回答コメントの構成

## Modbus ファンクション・コード

## レスポンス・コメント

Code (Hex)	Name	Usage
01	Read coil status	Read discrete output bit
02	Read input status	Read discrete input bit
03	Read holding registers	Read 16-bit register; used to read integer or
04	Read input registers	floating point process data
05	Force single coil	Write data to force coil ON/OFF
06	Preset single register	Write data in 16-bit integer format
08	Loopback diagnosis	Diagnostic testing of the communication port
0F	Force multiple coils	Write multiple data to force coil ON/OFF
10	Preset multiple registers	Write multiple data in 16-bit integer format

## 機能コード 01

ADAM-6000 モジュールのディスクリット出力の ON/OFF ステータスをバイナリ・フォーマットで読み取ります。

リクエストメッセージのフォーマット：

Command Body					
Station Address	Function Code	Start Address High Byte	Start Address Low Byte	Requested Number of Coil High Byte	Requested Number of Coil Low Byte

例：ADAM-6000モジュールからコイル1～8（アドレス00017～00024）を読み出す。

01 01 00 17 00 08

応答メッセージの形式：

Command Body					
Station Address	Function Code	Byte Count	Data	Data	...

例：コイル2～7はオン、その他はオフ。

01 01 01 42

レスポンスでは、コイル1～8のステータスはバイト値42（16進数）で示され、これはバイナリ・フォーマットでは0100 0010に相当する。

## ファンクションコード 02

ADAM-6000 モジュールのディスクリット入力 ON/OFF ステータスをバイナリ形式で読み取ります。

リクエストメッセージのフォーマット：

Command Body					
Station Address	Function Code	Start Address High Byte	Start Address Low Byte	Requested Number of Input High Byte	Requested Number of Input Low Byte

例：ADAM-6000モジュールからコイル1～8（アドレス00001～00008）を読み出す。

01 02 00 01 00 08

応答メッセージの形式：

Command Body					
Station Address	Function Code	Byte Count	Data	Data	...

例入力2と3はオン、その他はオフ。

01 02 01 60

レスポンスでは、入力1～8のステータスがバイト値60（16進数）で示され、これはバイナリ・フォーマットでは0110 0000に相当する。

#### ファンクションコード03と04

入力レジスタのバイナリ内容を読み取る。

リクエストメッセージのフォーマット：

Command Body					
Station Address	Function Code	Start Address High Byte	Start Address Low Byte	Requested Number of Register High Byte	Requested Number of Register Low Byte

例：アドレス 40001 ～ 40002 のアナログ入力 1 と 2 を ADAM-6017 モジュールから浮動小数点値として読み出す。

01 04 00 01 00 02

応答メッセージの形式：

Command Body					
Station Address	Function Code	Byte Count	Data	Data	...

例：アナログ入力1と2の生データは、それぞれ17096と0である。入力レンジが0～10Vに設定されている場合、電圧は以下のように計算できる：

$$\text{アナログ入力1} = (17097/65535) \times 10 \text{ V} = 2.608 \text{ V}$$

$$\text{アナログ入力2} = (0/65535) \times 10 \text{ V} = 0$$

V 01 04 04 42 C8 00 00

#### ファンクションコード 05

単一のコイルを強制的に ON または OFF にする。要求される ON/OFF 状態は、クエリ・データ・フィールドの定数で指定される。FF 00（16進数）の値はONを要求し、00 00（16進数）の値はOFFを要求し、FF FF（16進数）の値は強制値の要求する。

リクエストメッセージのフォーマット：

Command Body					
Station Address	Function Code	Coil Address High Byte	Coil Address Low Byte	Force Data High Byte	Force Data Low Byte

例ADAM-6000 モジュールのコイル 3（アドレス 00003）を強制的にオンにします。

01 05 00 03 ff 00

応答メッセージの形式：

通常のレスポンスは、コイル状態を強制した後に返されるクエリーのエコーである。

Command Body					
Station Address	Function Code	Coil Address High Byte	Coil Address Low Byte	Force Data High Byte	Force Data Low Byte

#### ファンクションコード 06

整数値を 1 つのレジスタにプリセットする。

リクエスト・メッセージのフォーマット：

Command Body					
Station Address	Function Code	Coil Address High Byte	Coil Address Low Byte	Force Data High Byte	Force Data Low Byte

例ADAM-6000 モジュールのプリセット・レジスタ 40002 を 00 04 (16 進) に設定する。

01 06 00 02 00 04

応答メッセージの形式：

通常の応答は、コイル状態がプリセットされた後に返される問い合わせのエコーである。

Command Body					
Station Address	Function Code	Coil Address High Byte	Coil Address Low Byte	Force Data High Byte	Force Data Low Byte

#### 機能コード 08

受信した問い合わせメッセージをエコーする。メッセージは、データバッファの長さの半分から8バイトを引いた長さまでであれば、どのような長さでもよい。

リクエストメッセージのフォーマット：

Command Body		
Station Address	Function Code	Any data, length limited to approximately half the length of the data buffer

例：01 08 00 02 00 04

応答メッセージの形式：

Command Body		
Station Address	Function Code	Data bytes received

例：01 08 00 02 00 04

### ファンクションコード15 (16進数で「0F」)

一連のコイルの各コイルを強制的に ON または OFF 状態にする。

リクエスト・メッセージのフォーマット：

Command Body

Station Address	Function Code	Start Address High Byte	Start Address Low Byte	Requested Number of Coil High Byte	Requested Number of Coil Low Byte	Byte Count	Force Data High Byte	Force Data Low Byte
-----------------	---------------	-------------------------	------------------------	------------------------------------	-----------------------------------	------------	----------------------	---------------------

例ADAM-6000 モジュールのアドレス 00017 (16 進数で “11”) から始まる一連の 10 個のコイルを強制する要求。

01 0f 00 11 00 0a 02 cd 01

クエリデータの内容は2バイトである：CD01 (16進数)、バイナリフォーマットでは1100 1101 0000 0001に相当する。バイナリ・ビットは以下のようにアドレスにマッピングされる。

Bit:	1	1	0	0	1	1	0	1	0	0	0	0	0	0	1
Address (000XX):	24	23	22	21	20	19	18	17	-	-	-	-	-	26	25

応答メッセージの形式：

正常な応答は、強制コイルのステーションアドレス、ファンクションコード、スタートアドレス、要求番号を返す。

Command Body

Station Address	Function Code	Start Address High Byte	Start Address Low Byte	Requested Number of Coil High Byte	Requested Number of Coil Low Byte
-----------------	---------------	-------------------------	------------------------	------------------------------------	-----------------------------------

例：01 0F 00 11 00 0A

### ファンクションコード16 (16進数で「10」)

保持レジスタのシーケンスにプリセット値を適用する。

リクエストメッセージのフォーマット：

Command Body

Station Address	Function Code	Start Address High Byte	Start Address Low Byte	Requested Number of Register High Byte	Requested Number of Register Low Byte	Byte Count	Data
-----------------	---------------	-------------------------	------------------------	--	---------------------------------------	------------	------

例ADAM-6000 モジュールの定数 1 (アドレス 40009) を 100.0 にプリセットします。01

10 00 09 00 02 04 42 C8 00 00

応答メッセージの形式：

正常な応答は、ステーション・アドレス、ファンクション・コード、スタート・アドレス、要求されたプリセット・レジスタ数を返す。

Command Body

Station Address	Function Code	Start Address High Byte	Start Address Low Byte	Requested Number of Register High Byte	Requested Number of Register Low Byte
-----------------	---------------	-------------------------	------------------------	--	---------------------------------------

例：01 10 00 09 00 02

## B.2 ADAM-6000 I/O Modbusマッピングテーブル

ADAM-6015

アドレス (0X) :

Address (0X)	Channel	Description	Attribute
00101	0		R/W
00102	1		R/W
00103	2		R/W
00104	3	Reset historical max. value	R/W
00105	4		R/W
00106	5		R/W
00107	6		R/W
00109	Average 0~6		R/W

Address (0X)	Channel	Description	Attribute
00111	0		R/W
00112	1		R/W
00113	2		R/W
00114	3	Reset historical min. value	R/W
00115	4		R/W
00116	5		R/W
00117	6		R/W
00119	Average 0~6		R/W

Address (0X)	Channel	Description	Attribute
00121	0		Read
00122	1		Read
00123	2		Read
00124	3	Burnout flag <sup>1</sup>	Read
00125	4		Read
00126	5		Read
00127	6		Read

Address (0X)	Channel	Description	Attribute
00131	0		Read
00132	1		Read
00133	2		Read
00134	3	High alarm flag <sup>2</sup>	Read
00135	4		Read
00136	5		Read
00137	6		Read
00139	Average 0~6		Read

Address (0X)	Channel	Description	Attribute
00141	0		Read
00142	1		Read
00143	2		Read
00144	3	Low alarm flag <sup>3</sup>	Read
00145	4		Read
00146	5		Read
00147	6		Read

Address (0X)	Channel	Description	Attribute
00149	Average 0~6	Low alarm flag <sup>3</sup>	Read

Address (0X)	Channel	Description	Attribute
00301	0		Write
00302	1		Write
00303	2		Write
00304	3	Clear GCL internal counter value	Write
00305	4		Write
00306	5		Write
00307	6		Write
00308	7		Write

**Address (4X):**

Address (4X)	Channel	Description	Attribute
40001	0		Read
40002	1		Read
40003	2		Read
40004	3	AI value	Read
40005	4		Read
40006	5		Read
40007	6		Read
40009	Average 0~6		Read

Address (4X)	Channel	Description	Attribute
40011	0		Read
40012	1		Read
40013	2		Read
40014	3	Historical max. AI value	Read
40015	4		Read
40016	5		Read
40017	6		Read
40019	Average 0~6		Read

Address (4X)	Channel	Description	Attribute
40021	0		Read
40022	1		Read
40023	2		Read
40024	3	Historical min. AI value	Read
40025	4		Read
40026	5		Read
40027	6		Read
40029	Average 0~6		Read

Address (4X)	Channel	Description	Attribute
40031~40032	0		Read
40033~40034	1		Read
40035~40036	2		Read
40037~40038	3	AI floating point value (IEEE754)	Read
40039~40040	4		Read
40041~40042	5		Read
40043~40044	6		Read
40047~40048	Average 0~6		Read

Address (4X)	Channel	Description	Attribute
40051~40052	0		Read
40053~40054	1		Read
40055~40056	2		Read
40057~40058	3	Historical max. AI floating point value (IEEE754)	Read
40059~40060	4		Read
40061~40062	5		Read
40063~40064	6		Read
40067~40068	Average 0~6		Read

Address (4X)	Channel	Description	Attribute
40071~40072	0		Read
40073~40074	1		Read
40075~40076	2		Read
40077~40078	3	Historical min. AI floating point value (IEEE754)	Read
40079~40080	4		Read
40081~40082	5		Read
40083~40084	6		Read
40087~40088	Average 0~6		Read

Address (4X)	Channel	Description	Attribute
40101~40102	0		Read
40103~40104	1		Read
40105~40106	2		Read
40107~40108	3	AI status <sup>4</sup>	Read
40109~40110	4		Read
40111~40112	5		Read
40113~40114	6		Read

Address (4X)	Channel	Description	Attribute
40201	0		R/W
40202	1		R/W
40203	2		R/W
40204	3	Type code <sup>5</sup>	R/W
40205	4		R/W
40206	5		R/W
40207	6		R/W
40209	Average		R/W

Address (4X)	Channel	Description	Attribute
40211		Module Name 1	Read
40212		Module Name 2	Read

Address (4X)	Channel	Description	Attribute
40305	0~15	GCL internal flag value	R/W

Address (4X)	Channel	Description	Attribute
40311~40312	0		Read
40313~40314	1		Read
40315~40316	2		Read
40317~40318	3	GCL internal counter	Read
40319~40320	4	value	Read
40321~40322	5		Read
40323~40324	6		Read
40325~40326	7		Read

#### 備考

1. チャンネルがRTD信号を検出できない場合、ビット値は1になります。
2. 高アラーム値は、Adam/Apax .NET Utilityを使用して設定できます。アナログ入力値が高アラーム値より高い場合、ビット値は 1 になります。
3. 低アラーム値は、Adam/Apax .NET Utilityを使用して設定できます。アナログ入力値が低アラーム値より低い場合、ビット値は 1 になります。

## 4. アナログ入力ステータス

Bit	Description
0	Normal
3	Open circuit/burnout

## 5. タイプコード

Type Code (Hex)	Input Range	
0x03A4	PT100(385)	-50~150°C
0x03A5	PT100(385)	0~100°C
0x03A6	PT100(385)	0~200°C
0x03A7	PT100(385)	0~400°C
0x03A2	PT100(385)	-200~200°C
0x03C4	PT100(392)	-50~150°C
0x03C5	PT100(392)	0~100°C
0x03C6	PT100(392)	0~200°C
0x03C7	PT100(392)	0~400°C
0x03C2	PT100(392)	-200~200°C
0x03E2	PT1000	-40~160°C
0x0300	Balco500	-30~120°C
0x0320	NI604(518)	-80~100°C
0x0321	NI604(518)	0~100°C

6. ADAM モジュールのモデル名は、Modbus アドレス 40211 に保存されている 16 進値を読み取ることで再試行できます。

## ADAM-6017

アドレス (0X) :

Address (0X)	Channel	Description	Attribute
00017	0	DO value	R/W
00018	1		R/W

Address (0X)	Channel	Description	Attribute
00101	0		R/W
00102	1		R/W
00103	2		R/W
00104	3		R/W
00105	4	Reset historical max. AI value	
00106	5		R/W
00107	6		R/W
00108	7		R/W
00109	Average 0~7		R/W

Address (0X)	Channel	Description	Attribute
00111	0		R/W
00112	1		R/W
00113	2		R/W
00114	3		R/W
00115	4	Reset historical min. AI value	R/W
00116	5		R/W
00117	6		R/W
00118	7		R/W
00119	Average 0~7		R/W

Address (0X)	Channel	Description	Attribute
00121	0		Read
00122	1		Read
00123	2		Read
00124	3	Open-circuit flag <sup>1</sup> (burn out)	Read
00125	4		Read
00126	5		Read
00127	6		Read
00128	7		Read

Address (0X)	Channel	Description	Attribute
00131	0		Read
00132	1		Read
00133	2		Read
00134	3		Read
00135	4	High alarm flag <sup>2</sup>	Read
00136	5		Read
00137	6		Read
00138	7		Read
00139	Average 0~7		Read

Address (0X)	Channel	Description	Attribute
00141	0		Read
00142	1		Read
00143	2		Read
00144	3		Read
00145	4	Low alarm flag <sup>3</sup>	Read
00146	5		Read
00147	6		Read
00148	7		Read
00149	Average 0~7		Read

Address (0X)	Channel	Description	Attribute
00301	0		Write
00302	1		Write
00303	2		Write
00304	3	Clear GCL internal counter value	Write
00305	4		Write
00306	5		Write
00307	6		Write
00308	7		Write

**Address (4X)**

Address (4X)	Channel	Description	Attribute
40001	0	AI value	Read
40002	1		Read
40003	2		Read
40004	3		Read
40005	4		Read
40006	5		Read
40007	6		Read
40008	7		Read
40009	Average 0~7		Read

Address (4X)	Channel	Description	Attribute
40011	0		Read
40012	1		Read
40013	2		Read
40014	3		Read
40015	4	Historical max. AI value	Read
40016	5		Read
40017	6		Read
40018	7		Read
40019	Average 0~7		Read

Address (4X)	Channel	Description	Attribute
40021	0		Read
40022	1		Read
40023	2		Read
40024	3		Read
40025	4	Historical min. AI value	Read
40026	5		Read
40027	6		Read
40028	7		Read
40029	Average 0~7		Read

Address (4X)	Channel	Description	Attribute
40031~40032	0		Read
40033~40034	1		Read
40035~40036	2		Read
40037~40038	3		Read
40039~40040	4	AI floating point value (IEEE754)	Read
40041~40042	5		Read
40043~40044	6		Read
40045~40046	7		Read
40047~40048	Average 0~7		Read

Address (4X)	Channel	Description	Attribute
40051~40052	0		Read
40053~40054	1		Read
40055~40056	2		Read
40057~40058	3		Read
40059~40060	4	Historical max. AI floating point value (IEEE754)	Read
40061~40062	5		Read
40063~40064	6		Read
40065~40066	7		Read
40067~40068	Average 0~7		Read

Address (4X)	Channel	Description	Attribute
40071~40072	0		Read
40073~40074	1		Read
40075~40076	2		Read
40077~40078	3		Read
40079~40080	4	Historical min. AI floating point value (IEEE754)	Read
40081~40082	5		Read
40083~40084	6		Read
40085~40086	7		Read
40087~40088	Average 0~7		Read

Address (4X)	Channel	Description	Attribute
40101	0		Read
40103	1		Read
40105	2		Read
40107	3		Read
40109	4	AI status <sup>4</sup>	Read
40111	5		Read
40113	6		Read
40115	7		Read

アドレス (4X)	チャンネル	説明	属性
40201	0		R/W
40202	1		R/W
40203	2		R/W
40204	3		R/W
40205	4	タイプコード <sup>5</sup>	R/W
40206	5		R/W
40207	6		R/W
40208	7		R/W
40209	平均 0~7		読む

アドレス (4X)	チャンネル	説明	属性
40211		モジュール名 1	読む
40212		モジュール名 2	読む
40221	すべて	AIチャンネル有効	R/W
40305	0~15	GCL内部フラグ値	R/W

アドレス (4X)	チャンネル	説明	属性
40311~40312	0		読む
40313~40314	1		読む
40315~40316	2		読む
40317~40318	3	GCL内部カウンター	読む
40319~40320	4	値	読む
40321~40322	5		読む
40323~40324	6		読む
40325~40326	7		読む

**注意** 青色のModbusアドレスはADAM-6000 GEまたはDバージョンでのみサポートされます。



#### 備考

1. チャンネルが 4~20mA 入力レンジの電流入力信号を検出できない場合、ビット値は 1 になります。
2. Adam/Apax .NET Utilityを使用して、ハイアラーム値を設定することができます。アナログ入力値がハイアラームより高い場合、ビット値は 1 になります。
3. 低アラーム値は、Adam/Apax .NET Utilityを使用して設定できます。アナログ入力値が低アラーム値より低い場合、ビット値は 1 になります。
4. アナログ入カステータス

Bit	Description
0	Fail to provide the AI value
1	Over range
2	Under range
3	Open circuit/burnout

4	Reserved
5	Reserved
6	Reserved
7	ADC initialization error
8	Reserved
9	Zero/span calibration error
10	Reserved
11	Reserved
12	Reserved
13	Reserved
14	Reserved
15	Reserved

## 5. タイプコード

Type Code (Hex)	Input Range
0x0103	±150 mV
0x0104	±500 mV
0x0105	0~150 mV
0x0106	0~500 mV
0x0140	±1 V
0x0142	±5 V
0x0143	±10 V
0x0145	0~1 V
0x0147	0~5 V
0x0148	0~10 V
0x0181	±20 mA
0x0180	4~20 mA
0x0182	0~20 mA

- ADAM モジュールのモデル名は、Modbus アドレス 40211 に格納されている 16 進値を読み取ることで再試行できます。

### ADAM-6018、ADAM-6018+（アナログ入力のみ対応）

#### アドレス (0X)

Address (0X)	Channel	Description	Attribute
00017	0		R/W
00018	1		R/W
00019	2		R/W
00020	3	DO value (only for ADAM-6018)	R/W
00021	4		R/W
00022	5		R/W
00023	6		R/W
00024	7		R/W

Address (0X)	Channel	Description	Attribute
00101	0		R/W
00102	1		R/W
00103	2		R/W
00104	3		R/W
00105	4	Reset historical max. value	R/W
00106	5		R/W
00107	6		R/W
00108	7		R/W
00109	Average 0~7		R/W

Address (0X)	Channel	Description	Attribute
00111	0		Read
00112	1		Read
00113	2		Read
00114	3		Read
00115	4	Reset historical min. value	Read
00116	5		Read
00117	6		Read
00118	7		Read
00119	Average 0~7		Read

Address (0X)	Channel	Description	Attribute
00121	0		Read
00122	1		Read
00123	2		Read
00124	3	Burnout flag <sup>1</sup>	読む
00125	4		Read
00126	5		Read
00127	6		Read
00128	7		Read

Address (0X)	Channel	Description	Attribute
00131	0		Read
00132	1		Read
00133	2		Read
00134	3		Read
00135	4	High alarm flag <sup>2</sup>	Read
00136	5		Read
00137	6		Read
00138	7		Read
00139	Average 0~7		Read

Address (0X)	Channel	Description	Attribute
00141	0		Read
00142	1		Read
00143	2		Read
00144	3		Read
00145	4	Low alarm flag <sup>3</sup>	Read
00146	5		Read
00147	6		Read
00148	7		Read
00149	Average 0~7		Read

#### Address (4X)

Address (4X)	Channel	Description	Attribute
40001	0		Read
40002	1		Read
40003	2		Read
40004	3		Read
40005	4	AI value	Read
40006	5		Read
40007	6		Read
40008	7		Read
40009	Average 0~7		Read

Address (4X)	Channel	Description	Attribute
40011	0		Read
40012	1		Read
40013	2		Read
40014	3		Read
40015	4	Historical max. AI value	Read
40016	5		Read
40017	6		Read
40018	7		Read
40019	Average 0~7		Read

Address (4X)	Channel	Description	Attribute
40021	0		Read
40022	1		Read
40023	2		Read
40024	3		Read
40025	4	Historical min. AI value	Read
40026	5		Read
40027	6		Read
40028	7		Read
40029	Average 0~7		Read

Address (4X)	Channel	Description	Attribute
40305	0~15	GCL internal flag value	R/W

#### 備考

1. 特定のチャンネルが熱電対信号を検出できない場合、ビット値は 1 になります。
2. ユーザーは、Adam/Apax .NET Utilityを使用して、高アラーム値を設定できます。アナログ入力値が高アラーム値より高い場合、ビット値は 1 になります。
3. 低アラーム値は、Adam/Apax .NET Utilityを使用して設定できます。アナログ入力値が低アラーム値より低い場合、ビット値は 1 になります。

#### ADAM-6024

##### アドレス (0X)

Address(0X)	Channel	Description	Attribute
00001	0	DI value	Read
00002	1		Read
00017	0	DO value	R/W
00018	1		R/W
00101~00106	0~5	Reset historical max AI value	R/W
00111~00116	0~5	Reset historical min AI value	R/W

##### Address(4X)

Address(4X)	Channel	Description	Attribute
40001	0		Read
40002	1		Read
40003	2		Read
40004	3	AI value	Read
40005	4		読む
40006	5		読む
40011	0		R/W
40012	1	AO値	R/W

40021	0		Read
40022	1		Read
40023	2	AI status <sup>1</sup>	Read
40024	3		Read
40025	4		Read
40026	5		Read
40111~40116	0~5	Historical max AI value	Read
40121~40126	0~5	Historical min AI value	Read
40201~40206	0~5	AI type code <sup>2</sup>	R/W
40209~40210	0~1	AO type code <sup>3</sup>	R/W
40211	-	Module name	Read
40221	0~5	AI channel enable	R/W
40401~40402	0~1	AO startup value	R/W

#### Remark

##### 1. AI status

- Bit value = 0: normal
- Bit value = 1: over high
- Bit value = 2: over low
- Bit value = 0: invalid calibration

##### 2. AI type

- 7 in hex → 4~20mA
- 8 in hex → +/- 10V
- D in hex → 0~20mA

##### 1. AO type

- 0 in hex → 0~20mA
- 1 in hex → 4~20mA
- 2 in hex → 0~10V

## ADAM-6050

アドレス  
(0X)

Address (0X)	Channel	Description	Attribute
00001	0		Read
00002	1		Read
00003	2		Read
00004	3		Read
00005	4		Read
00006	5		Read
00007	6	DI value	Read
00008	7		Read
00009	8		Read
00010	9		Read
00011	10		Read
00012	11		Read

Address (0X)	Channel	Description	Attribute
00017	0	DO value	R/W
00018	1		R/W
00019	2		R/W
00020	3		R/W
00021	4		R/W
00022	5		R/W

Address (0X)	Channel	Description	Attribute
00033	0	Counter start (1)/stop (0)	R/W
00034		Clear counter (1)	Write
00035		Clear overflow <sup>3</sup>	R/W
00036		DI latch status <sup>4</sup>	R/W
00037	1	Counter start (1)/stop (0)	R/W
00038		Clear counter (1)	Write
00039		Clear overflow <sup>3</sup>	R/W
00040		DI latch status <sup>4</sup>	R/W
00041	2	Counter start (1)/stop (0)	R/W
00042		Clear counter (1)	Write
00043		Clear overflow <sup>3</sup>	R/W
00044		DI latch status <sup>4</sup>	R/W
00045	3	Counter start (1)/stop (0)	R/W
00046		Clear counter (1)	Write
00047		Clear overflow <sup>3</sup>	R/W
00048		DI latch status <sup>4</sup>	R/W
00049	4	Counter start (1)/stop (0)	R/W
00050		Clear counter (1)	Write
00051		Clear overflow <sup>3</sup>	R/W
00052		DI latch status <sup>4</sup>	R/W
00053	5	Counter start (1)/stop (0)	R/W
00054		Clear counter (1)	Write
00055		Clear overflow <sup>3</sup>	R/W
00056		DI latch status <sup>4</sup>	R/W
00057	6	Counter start (1)/stop (0)	R/W
00058		Clear counter (1)	Write
00059		Clear overflow <sup>3</sup>	R/W
00060		DI latch status <sup>4</sup>	R/W
00061	7	Counter start (1)/stop (0)	R/W
00062		Clear counter (1)	Write
00063		Clear overflow <sup>3</sup>	R/W
00064		DI latch status <sup>4</sup>	R/W
00065	8	Counter start (1)/stop (0)	R/W
00066		Clear counter (1)	Write

00067		Clear overflow <sup>3</sup>	R/W
00068		DI Latch status <sup>4</sup>	R/W
00069	9	Counter start (1)/stop (0)	R/W
00070		Clear counter (1)	Write
00071		Clear overflow <sup>3</sup>	R/W
00072		DI latch status <sup>4</sup>	R/W
00073	10	Counter start (1)/stop (0)	R/W
00074		Clear counter (1)	Write
00075		Clear overflow <sup>3</sup>	R/W
00076		DI latch status <sup>4</sup>	R/W
00077	11	Counter start (1)/stop (0)	R/W
00078		Clear counter (1)	Write
00079		Clear overflow <sup>3</sup>	R/W
00080		DI latch status <sup>4</sup>	R/W

Address (0X)	Channel	Description	Attribute
00301	0		Write
00302	1		Write
00303	2		Write
00302	3	GCL内部カウンタのクリア	Write
00305	4		Write
00306	5		Write
00307	6		Write
00308	7		Write

#### Address (4X)

Address (4X)	Channel	Description	Attribute
40301	All	DI value	Read
40303	All	DO value	R/W
40305	0~15	GCL internal flag value	R/W
40307	All	DO diagnostic status (for D version or later)	Read
40308	All	DO diagnostic mask Read (for D version or later)	Read

Address (4X)	Channel	Description	Attribute
40001~40002	0		Read
40003~40004	1		Read
40005~40006	2		Read
40007~40008	3		Read
40009~40010	4		Read
40011~40012	5	Counter/Frequency value <sup>1</sup>	Read
40013~40014	6		Read
40015~40016	7		Read
40017~40018	8		Read
40019~40020	9		Read
40021~40022	10		Read
40023~40024	11		Read

Address (4X)	Channel	Description	Attribute
40025~40026	0		Read
40027~40028	1		Read
40029~40030	2	Pulse output low-level width <sup>2</sup>	Read
40031~40032	3		Read
40033~40034	4		Read
40035~40036	5		Read

Address (4X)	Channel	Description	Attribute
40037~40038	0		Read
40039~40040	1		Read
40041~40042	2	Pulse output high-level width <sup>2</sup>	Read
40043~40044	3		Read
40045~40046	4		Read
40047~40048	5		Read

Address (4X)	Channel	Description	Attribute
40049~40050	0		R/W
40051~40052	1		R/W
40053~40054	2	Set absolute pulse <sup>5</sup>	R/W
40055~40056	3		R/W
40057~40058	4		R/W
40059~40060	5		R/W

Address (4X)	Channel	Description	Attribute
40061~40062	0		R/W
40063~40064	1		R/W
40065~40066	2	Set incremental pulse <sup>6</sup>	R/W
40067~40068	3		R/W
40069~40070	4		R/W
40071~40072	5		R/W

Address (4X)	Channel	Description	Attribute
40211	All	Module name 1	Read
40212	All	Module name 2	Read

Address (4X)	Channel	Description	Attribute
40311~40312	0		Read
40313~40314	1		Read
40315~40316	2		Read
40317~40318	3	GCL internal counter value	Read
40319~40320	4		Read
40321~40322	5		Read
40323~40324	6		Read
40325~40326	7		Read

**注意** 青いModbusアドレスはCEバージョン（またはそれ以降）でのみサポートされています。



#### 備考

- カウンタ/周波数値の取得方法：  

$$\text{カウンタ (10進数)} = (40002\text{の値}) \times 65536 + (40001\text{の値}) \quad \text{周波数}$$

$$(10進数) = (40001\text{の値}) \div 10\text{Hz}$$
- 時間の増分：0.1 ms
- カウント数がオーバーフローの場合、ビット値は1になる。このビットが読み出されると、値は0に戻る。
- デジタル入力チャンネルが“High-to-Lowラッチ”または“Low-to-Highラッチ”に設定されている場合、ラッチ条件が満たされるとビット値は1になります。その後、このビットに0を書き込むまで（つまり、ラッチ状態をクリアするまで）、ビット値は1のままです。
- 何個のパルスが発生させるかを決める。このビットに0を書き込むと、連続パルスがする。
- パルス発生中に、このビットを使って追加パルスが発生させることができる。例えば、“Absolute pulse”が100に設定されているとします。パルス発生中に“Incremental pulse”を10に設定すると、さらに10パルスが発生します。100パルスは生成され続けます。
- ADAM モジュールのモデル名は、Modbus アドレス 40211 に保存されている 16 進値を読み取ることで再試行できます。

## ADAM-6051

## Address (0X)

Address (0X)	Channel	Description	Attribute
00001	0		Read
00002	1		Read
00003	2		Read
00004	3		Read
00005	4		Read
00006	5		Read
00007	6	DI値	Read
00008	7		Read
00009	8		Read
00010	9		Read
00011	10		Read
00012	11		Read
00013	12		Read
00014	13		Read

Address (0X)	Channel	Description	Attribute
00033	0	Counter start (1)/stop (0)	R/W
00034		Clear counter (1)	Write
00035	1	Clear overflow <sup>3</sup>	R/W
00036		DI latch status <sup>4</sup>	R/W
00037		Counter start (1)/stop (0)	R/W
00038		Clear counter (1)	Write
00039	2	Clear overflow <sup>3</sup>	R/W
00040		DI latch status <sup>4</sup>	R/W
00041		Counter start (1)/stop (0)	R/W
00042		Clear counter (1)	Write
00043	3	Clear overflow <sup>3</sup>	R/W
00044		DI latch status <sup>4</sup>	R/W
00045		Counter start (1)/stop (0)	R/W
00046		Clear counter (1)	Write
00047	4	Clear overflow <sup>3</sup>	R/W
00048		DI latch status <sup>4</sup>	R/W
00049		Counter start (1)/stop (0)	R/W
00050		Clear counter (1)	Write
00051	5	Clear overflow <sup>3</sup>	R/W
00052		DI latch status <sup>4</sup>	R/W
00053		Counter start (1)/stop (0)	R/W
00054		Clear counter (1)	Write
00055	6	Clear overflow <sup>3</sup>	R/W
00056		DI latch status <sup>4</sup>	R/W
00057		Counter start (1)/stop (0)	R/W

00058		Clear counter (1)	Write
00059		Clear overflow <sup>3</sup>	R/W
00060		DI latch status <sup>4</sup>	R/W
00061	7	Counter start (1)/stop (0)	R/W
00062		Clear counter (1)	Write
00063		Clear overflow <sup>3</sup>	R/W
00064		DI latch status <sup>4</sup>	R/W
00065	8	Counter start (1)/stop (0)	R/W
00066		Clear counter (1)	Write
00067		Clear overflow <sup>3</sup>	R/W
00068		DI latch status <sup>4</sup>	R/W
00069	9	Counter start (1)/stop (0)	R/W
00070		Clear counter (1)	Write
00071		Clear overflow <sup>3</sup>	R/W
00072		DI latch status <sup>4</sup>	R/W
00073	10	Counter start (1)/stop (0)	R/W
00074		Clear counter (1)	Write
00075		Clear overflow <sup>3</sup>	R/W
00076		DI latch status <sup>4</sup>	R/W
00077	11	Counter start (1)/stop (0)	R/W
00078		Clear counter (1)	Write
00079		Clear overflow <sup>3</sup>	R/W
00080		DI latch status <sup>4</sup>	R/W
00081	12 7	Counter start (1)/stop (0)	R/W
00082		Clear counter (1)	Write
00083		Clear overflow <sup>3</sup>	R/W
00084		DI latch status <sup>4</sup>	R/W
00085	137	Counter start (1)/stop (0)	R/W
00086		Clear counter (1)	Write
00087		Clear overflow <sup>3</sup>	R/W
00088		DI latch status <sup>4</sup>	R/W

Address (0X)	Channel	Description	Attribute
00301	0		Write
00302	1		Write
00303	2		Write
00304	3	Clear GCL internal counter value	Write
00305	4		Write
00306	5		Write
00307	6		Write
00308	7		Write

**Address (4X)**

Address (4X)	Channel	Description	Attribute
40001~40002	0		Read
40003~40004	1		Read
40005~40006	2		Read
40007~40008	3		Read
40009~40010	4		Read
40011~40012	5		Read
40013~40014	6	Counter/frequency value <sup>1</sup>	Read
40015~40016	7		Read
40017~40018	8		Read
40019~40020	9		Read
40021~40022	10		Read
40023~40024	11		Read
40025~40026	12		Read
40027~40028	13		Read

Address (4X)	Channel	Description	Attribute
40029~40030	0		
Low Level Width 2	Read	Pulse output	
40031~40032	1		Read

Address (4X)	Channel	Description	Attribute
40033~40034	0	Pulse output high-level	Read
40035~40036	1	width <sup>2</sup>	Read

Address (4X)	Channel	Description	Attribute
440037~40038	0	Set absolute pulse <sup>5</sup>	Read
40039~40040	1		Read

Address (4X)	Channel	Description	Attribute
440041~40042	0	Set incremental pulse <sup>6</sup>	Read
40043~40044	1		Read

Address (4X)	Channel	Description	Attribute
40033~40034	0	Pulse output high-level width	R/W
40035~40036	1		R/W

Address (4X)	Channel	Description	Attribute
40301	All	DI value	Read
40303	All	DO value	R/W
40305	0~15	GCL internal flag value	R/W
40307	All	DO diagnostic status (for D version)	Read

Address (4X)	Channel	Description	Attribute
40211		Module name 1	Read
40212		Module name 2	Read

Address (4X)	Channel	Description	Attribute
40311~40312	0	GCL internal counter value	Read
40313~40314	1		Read
40315~40316	2		Read
40317~40318	3		Read
40319~40320	4		Read
40321~40322	5		Read
40323~40324	6		Read
40325~40326	7		Read

**注意** 青いModbusアドレスはGEバージョン以降でのみサポートされています。



**備考**

1. カウンタ/周波数値の取得方法：  
 カウンタ（10進数）＝（40002の値）×65536＋（40001の値） 周波数（10進数）＝（40001の値）／10Hz
2. 時間の増分：0.1 ms
3. カウント数がオーバーフローの場合、ビット値は1になる。このビットが読み出されると、値は0に戻る。
4. デジタル入力チャンネルが“High-to-Lowラッチ”または“Low-to-Highラッチ”に設定されている場合、ラッチ条件が満たされるとビット値は1になります。その後、このビットに0を書き込むまで（つまりラッチ状態をクリアするまで）、ビット値は1のままです。
5. 何個のパルスを発生させるかを定める。このビットに0を書き込むと、連続パルスがする。
6. パルス発生中に、このビットを使って追加パルスを発生させることができる。例えば、“Absolute pulse”が100に設定されているとします。パルス発生中に“Incremental pulse”を10に設定すると、さらに10パルスが発生します。100パルスは生成され続けます。
7. ADAM モジュールのモデル名は、Modbus アドレス 40211 に保存されている 16 進値を読み取ることで再試行できます。

**ADAM-6052****Address (0X)**

Address (0X)	Channel	Description	Read
00001	0		Read
00002	1		Read
00003	2		Read
00004	3		Read
00005	4	DI value	Read
00006	5		Read
00007	6		Read
00008	7		Read

Address (0X)	Channel	Description	Attribute
00017	0		R/W
00018	1		R/W
00019	2		R/W
00020	3		R/W
00021	4	DO value	R/W
00022	5		R/W
00023	6		R/W
00024	7		R/W

Address (0X)	Channel	Description	Attribute
00033	0	Counter start (1)/stop (0)	R/W
00034		Clear counter (1)	Write
00035		Clear overflow <sup>3</sup>	R/W
00036		DI latch status <sup>4</sup>	R/W
00037	1	Counter start (1)/stop (0)	R/W
00038		Clear counter (1)	Write
00039		Clear overflow <sup>3</sup>	R/W
00040		DI latch status <sup>4</sup>	R/W
00041	2	Counter start (1)/stop (0)	R/W
00042		Clear counter (1)	Write
00043		Clear overflow <sup>3</sup>	R/W
00044		DI latch status <sup>4</sup>	R/W
00045	3	Counter start (1)/stop (0)	R/W
00046		Clear counter (1)	Write
00047		Clear overflow <sup>3</sup>	R/W
00048		DI latch status <sup>4</sup>	R/W
00049	4	Counter start (1)/stop (0)	R/W
00050		Clear counter (1)	Write
00051		Clear overflow <sup>3</sup>	R/W
00052		DI latch status <sup>4</sup>	R/W
00053	5	Counter start (1)/stop (0)	R/W
00054		Clear counter (1)	Write
00055		Clear overflow <sup>3</sup>	R/W
00056		DI latch status <sup>4</sup>	R/W
00057	6	Counter start (1)/stop (0)	R/W
00058		Clear counter (1)	Write
00059		Clear overflow <sup>3</sup>	R/W
00060		DI latch status <sup>4</sup>	R/W
00061	7	Counter start (1)/stop (0)	R/W
00062		Clear counter (1)	Write
00063		Clear overflow <sup>3</sup>	R/W
00064		DI latch status <sup>4</sup>	R/W

Address (0X)	Channel	Description	Attribute
00301	0		Write
00302	1		Write
00303	2		Write
00304	3	Clear GCL internal counter value	Write
00305	4		Write
00306	5		Write
00307	6		Write
00308	7		Write

**Address (4X)**

Address (4X)	Channel	Description	Attribute
40001~40002	0		Read
40003~40004	1		Read
40005~40006	2		Read
40007~40008	3	Counter/frequency value <sup>1</sup>	Read
40009~40010	4		Read
40011~40012	5		Read
40013~40014	6		Read
40015~40016	7		Read
40017~40018	0		R/W
40019~40020	1		R/W
40021~40022	2	R/W	
40023~40024	3	Pulse output low-level width <sup>2</sup>	R/W
40025~40026	4		R/W
40027~40028	5		R/W
40029~40030	6		R/W
40031~40032	7		R/W
40033~40034	0		R/W
40035~40036	1		R/W
40037~40038	2	R/W	
40039~40040	3	Pulse output high-level width <sup>2</sup>	R/W
40041~40042	4		R/W
40043~40044	5		R/W
40045~40046	6		R/W
40047~40048	7		R/W

Address (4X)	Channel	Description	Attribute
40049~40050	0		R/W
40051~40052	1		R/W
40053~40054	2		R/W
40055~40056	3	Set absolute pulse <sup>5</sup>	R/W
40057~40058	4		R/W
40059~40060	5		R/W
40061~40062	6		R/W
40063~40064	7		R/W
40065~40066	0		R/W
40067~40068	1		R/W
40069~40070	2	R/W	
40071~40072	3	Set incremental pulse <sup>6</sup>	R/W
40073~40074	4		R/W
40075~40076	5		R/W
40077~40078	6		R/W
40079~40080	7		R/W

Address (4X)	Channel	Description	Attribute
40301	All	DI value	Read
40303	All	DO value	R/W
40305	0~15	GCL internal flag value	R/W
40307	All	DO diagnostic status (for D version)	Read

Address (4X)	Channel	Description	Attribute
40211		Module name 1	Read
40212		Module name 2	Read

Address (4X)	Channel	Description	Attribute
40311~40312	0		Read
40313~40314	1		Read
40315~40316	2	GCL internal counter	Read
40317~40318	3	value	Read
40319~40320	4		Read
40321~40322	5		Read
40323~40324	6		Read
40325~40326	7		Read

**注意** 青いModbusアドレスはCEバージョン以降でのみサポートされています。



#### 備考

- カウンタ/周波数値の取得方法：  

$$\text{カウンタ (10進数)} = (40002\text{の値}) \times 65536 + (40001\text{の値}) \quad \text{周波数}$$

$$(10進数) = (40001\text{の値}) / 10\text{Hz}$$
- 時間の増分：0.1 ms
- カウント数がオーバーフローの場合、ビット値は1になる。このビットが読み出されると、値は0に戻る。
- デジタル入力チャンネルが“High-to-Lowラッチ”または“Low-to-Highラッチ”に設定されている場合、ラッチ条件が満たされるとビット値は1になります。その後、このビットに0を書き込むまで（つまりラッチ状態をクリアするまで）、ビット値は1のままです。
- 何個のパルスを発生させるかを決める。このビットに0を書き込むと、連続パルスがする。
- パルス発生中に、このビットを使って追加パルスを発生させることができる。例えば、“Absolute pulse”が100に設定されているとします。パルス発生中に“Incremental pulse”を10に設定すると、さらに10パルスが発生します。100パルスは生成され続けます。
- ADAM モジュールのモデル名は、Modbus アドレス 40211 に保存されている 16 進値を読み取ることで再試行できます。

## ADAM-6060/6066

## Address (0X)

Address (0X)	Channel	Description	Attribute
00001	0	DI値	Read
00002	1		Read
00003	2		Read
00004	3		Read
00005	4		Read
00006	5		Read

Address (0X)	Channel	Description	Attribute
00017	0	DO値	R/W
00018	1		R/W
00019	2		R/W
00020	3		R/W
00021	4		R/W
00022	5		R/W

Address (0X)	Channel	Description	Attribute
00033	0	Counter start (1)/stop (0)	R/W
00034		Clear counter (1)	Write
00035		Clear overflow <sup>3</sup>	R/W
00036		DI latch status <sup>4</sup>	R/W
00037	1	Counter start (1)/stop (0)	R/W
00038		Clear counter (1)	Write
00039		Clear overflow <sup>3</sup>	R/W
00040		DI latch status <sup>4</sup>	R/W
00041	2	Counter start (1)/stop (0)	R/W
00042		Clear counter (1)	Write
00043		Clear overflow <sup>3</sup>	R/W
00044		DI latch status <sup>4</sup>	R/W
00045	3	Counter start (1)/stop (0)	R/W
00046		Clear counter (1)	Write
00047		Clear overflow <sup>3</sup>	R/W
00048		DI latch status <sup>4</sup>	R/W
00049	4	Counter start (1)/stop (0)	R/W
00050		Clear counter (1)	Write
00051		Clear overflow <sup>3</sup>	R/W
00052		DI latch status <sup>4</sup>	R/W
00053	5	Counter start (1)/stop (0)	R/W
00054		Clear counter (1)	Write
00055		Clear overflow <sup>3</sup>	R/W
00056		DI latch status <sup>4</sup>	R/W

Address (0X)	Channel	Description	Attribute
00301	0		Write
00302	1		Write
00303	2		Write
00304	3	Clear GCL internal counter value	Write
00305	4		Write
00306	5		Write
00307	6		Write
00308	7		Write

#### Address (4X)

Address (4X)	Channel	Description	Attribute
40001~40002	0		Read
40003~40004	1		Read
40005~40006	2	Counter/frequency value <sup>1</sup>	Read
40007~40008	3		Read
40009~40010	4		Read
40011~40012	5		Read

Address (4X)	Channel	Description	Attribute
40013~40014	0		R/W
40015~40016	1		R/W
40017~40018	2	Pulse output low-level width <sup>2</sup>	R/W
40019~40020	3		R/W
40021~40022	4		R/W
40023~40024	5		R/W

Address (4X)	Channel	Description	Attribute
40025~40026	0		R/W
40027~40028	1		R/W
40029~40030	2	Pulse output high-level width <sup>2</sup>	R/W
40031~40032	3		R/W
40033~40034	4		R/W
40035~40036	5		R/W

Address (4X)	Channel	Description	Attribute
40037~40038	0		R/W
40039~40040	1		R/W
40041~40042	2	Set absolute pulse <sup>5</sup>	R/W
40043~40044	3		R/W
40045~40046	4		R/W
40047~40048	5		R/W

Address (4X)	Channel	Description	Attribute
40049~40050	0		R/W
40051~40052	1		R/W
40053~40054	2	Set incremental pulse <sup>6</sup>	R/W
40055~40056	3		R/W
40057~40058	4		R/W
40059~40060	5		R/W

Address (4X)	Channel	Description	Attribute
40301	All	DI value	Read
40303	All	DO value	R/W
40305	0~15	GCL internal flag value	R/W

Address (4X)	Channel	Description	Attribute
40211		Module name 1	Read
40212		Module name 2	Read

Address (4X)	Channel	Description	Attribute
40311~40312	0		Read
40313~40314	1		Read
40315~40316	2		Read
40317~40318	3	GCL internal counter value	Read
40319~40320	4		Read
40321~40322	5		Read
40323~40324	6		Read
40325~40326	7		Read

**注意** 青いModbusアドレスはGEバージョン以降でのみサポートされています。



#### 備考

- カウンタ/周波数値の取得方法：  
 カウンタ（10進数）=（40002の値）×65536+（40001の値） 周波数（10進数）=（40001の値）／10Hz
- 時間の増分：0.1 ms
- カウント数がオーバーフローの場合、ビット値は1になる。このビットが読み出されると、値は0に戻る。
- デジタル入力チャンネルが“High-to-Lowラッチ”または“Low-to-Highラッチ”に設定されている場合、ラッチ条件が満たされるとビット値は1になります。その後、このビットに0を書き込むまで（つまりラッチ状態をクリアするまで）、ビット値は1のままです。

- 
5. 何個のパルスが発生させるかを定める。このビットに0を書き込むと、連続パルスがする。
  6. パルス発生中に、このビットを使って追加パルスが発生させることができる。例えば、“Absolute pulse ”が100に設定されているとします。パルス発生中に “Incremental pulse ”を10に設定すると、さらに10パルスが発生します。100パルスは生成され続けます。
  7. ADAM モジュールのモデル名は、Modbus アドレス 40211 に保存されている 16 進値を読み取ることで再試行できます。

付録

C

接地基準

## C.1 フィールド接地とシールドの適用

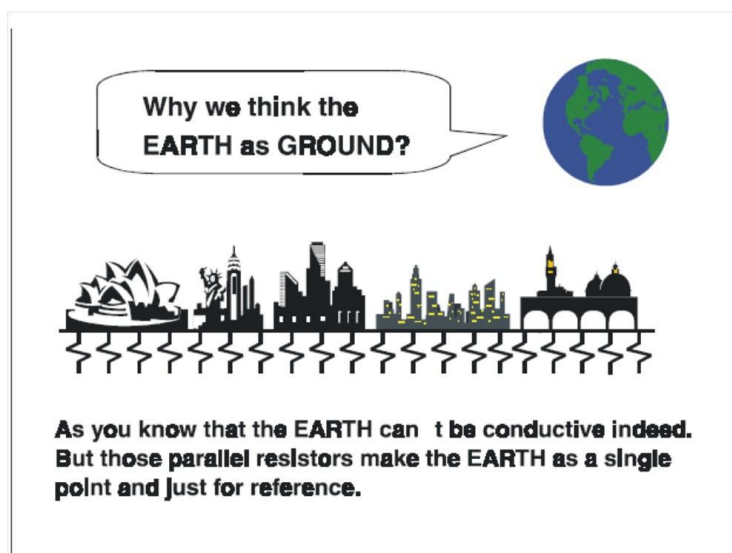
### C.1.1 概要

残念ながら、現場で予期せぬ問題が発生するため、システム統合作業を度で完了できることはめったにない。通信ネットやシステムが不安定だったり、機器の損傷や近くの雷雨によるノイズがあったりする。しかし、最も一般的な問題は不適切な配線であり、そのような問題は一般的にアースとシールドに関連している。配線やケーブリングにかかるコストはハードウェアに比べればかなり少ないが、ネットワークの信頼性は実質的にアースとシールドにほとんど依存している。従って、システムの信頼性を確保するためには、時間と品質の両面で配線・ケーブリングに投資する価値がある。この付録では、フィールド接地とシールドに関する基本的な概念について説明する。

接地	C. 2
参考：地球	C. 2. 1
フレーム・グラウンドとアース・バー	C. 2. 2
ノーマルモードとコモンモード	C. 2. 3
ワイヤーインピーダンス	C. 2. 4
一点接地	C. 2. 5
シールド	C. 3
ケーブル・シールド	C. 3. 1
システム・シールド	C. 3. 2
ノイズ低減テクニック	C. 4
チェックリスト	C. 5

## C.2 接地

### C.2.1 リファレンスとしての地球



図C.1 地球を地面として考える

明らかに、大地は導電性ではない。すべての建物は大地の上（または中）にある。

電力システムはアースに接続されている。これらを抵抗器と考えるとよい。数え切れないほどの並列抵抗器が、アースをひとつの基準点にしている。

## C.2.2 フレーム・グラウンドとアース・バー

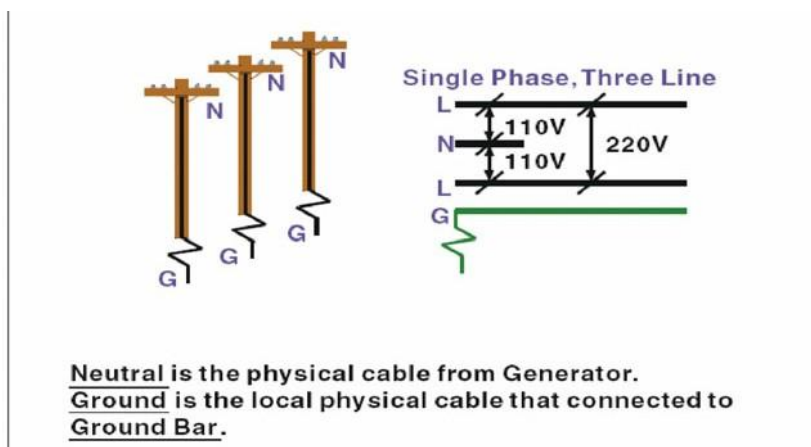
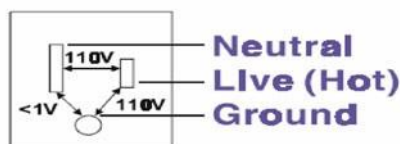


図 C.2 アース・バー

接地はシステムにとって最も重要な問題の一つである。コンピュータのフレーム・グラウンドと同じように、チャンネルによって取得されたデータ信号は、コンピュータ内部の電子回路の基準点を提供する。このコンピュータと通信する場合、互いの電子回路の基準点を作るために、シグナル・グラウンドとフレーム・グラウンドの両方を互いに接続する必要がある。コンピュータ・ネットワーク、電力システム、通信ネットワークなどでは、一般的にシステムごとにアース・バーを設置する必要がある。これらのアース・バーは、個々の基準点を提供するだけでなく、大地をアースとする。

## Normal Mode & Common Mode



**Normal Mode:** refers to defects occurring between the live and neutral conductors. Normal mode is sometimes abbreviated as NM, or L-N for live - to - neutral.

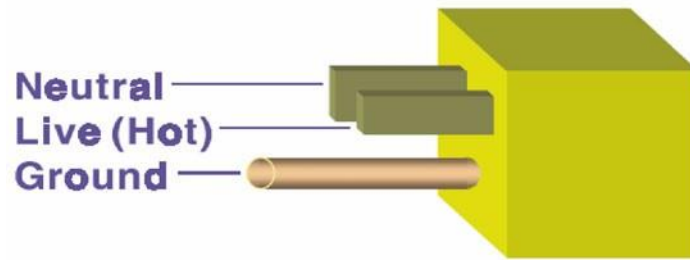
**Common Mode:** refers to defects occurring between either conductor and ground. It is sometimes abbreviated as CM, or N-G for neutral - to - ground.

図 C.3 図 C.3: ノーマルモードとコモンモード

## C.2.3 ノーマルモードとコモンモード

ライブ回路とコンクリート床、またはニュートラルとコンクリート床の間の電圧を測定しようとする、無意味な値が得られます。「ホット」と「ニュートラル」は単なる関係信号であり、これらの信号を測定すれば110Vまたは220V<sub>(AC)</sub>が得られる。ノーマルモードとコモンモードは、フレーム・グラウンドがすべてのシステムや機器にとって最も重要な基準信号であることを示しています。

## Normal Mode & Common Mode



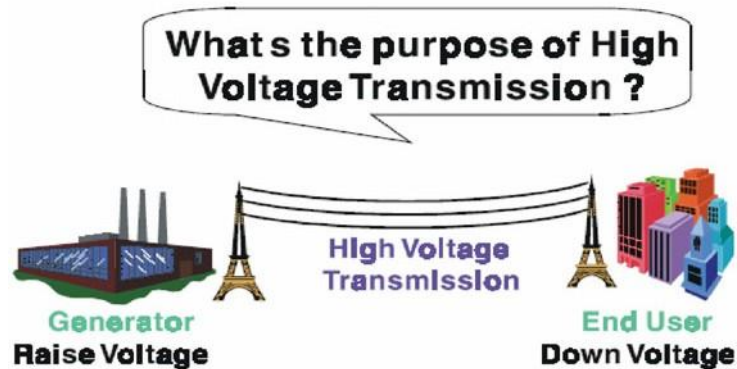
Ground-pin is longer than others, for first contact to power system and noise bypass.

Neutral-pin is broader than Live-pin, for reduce contacted Impedance.

図C.4 ノーマルモードとコモンモード

- グランドピンは他のピンより長く、電源システムと最初に接触し、ノイズバイパスとして機能する。
- 接触インピーダンスを低減するため、ニュートラル・ピンはライブ・ピンより広い。

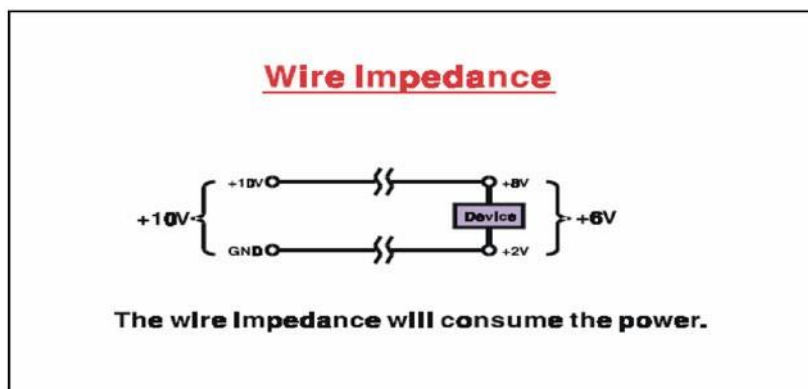
### C.2.4 ワイヤインピーダンス



Referring to OHM rule, above diagram shows that how to reduce the power loss on cable.

図C.5 高圧送電

高圧送電線が必要なのは、発電所が高圧電流を発生させ、それを地域の発電所が電圧を下げて配電するからである。電力の公式、 $P = I * V$ によると、電圧を上げると電流は減少する。ケーブルのインピーダンスはその材質によって決まることから、オームの法則 ( $V = I * R$ ) は、インピーダンスの高い材質を使用すれば、電流の減少によって送電中の電力損失が減少することを意味している。従って、高圧送電線を使用すれば、電力をある場所から別の場所へ移動させるコストを削減できる。



図C.6 ワイヤー・インピーダンス

### C.2.5 一点接地

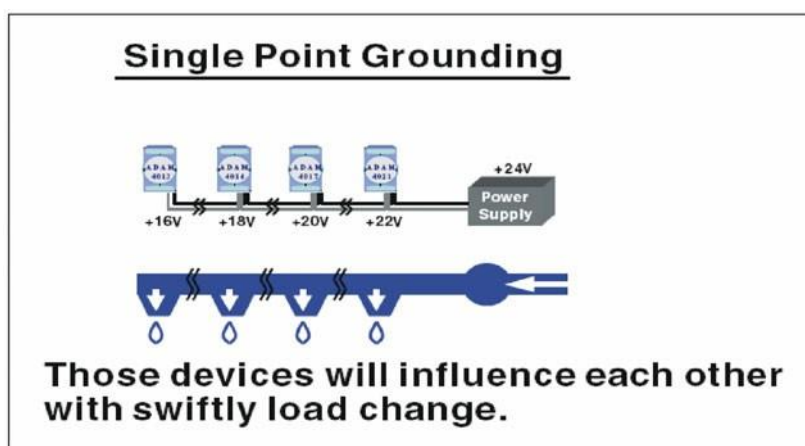


図 C.7 一点接地

ある人が熱いシャワーを浴びているときに、他の人が熱い水の蛇口をひねると、シャワーの水は冷たくなる。図

C.7は、この概念を示し、負荷が急激に変化したときに機器がどのような影響を受けるかを示している。この例では、4つの排出口がすべて開いていると仮定する。排出口3と4が閉じられると、他の2つの排出口への流量が増加する。つまり、流量は一定ではありません。

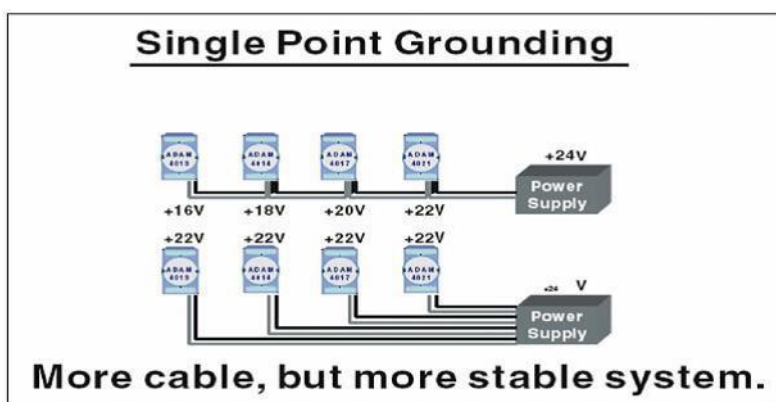


図 C.8 単一点接地

図C. 8は、1本のケーブルだけを使用する場合よりも、1点接地システムの方がより安定したシステムになることを示しています。これらのデバイスへの給電に細いケーブルを使用した場合、エンド・デバイスが受け取る電力は他のデバイスよりも少なくなります。細いケーブルはエネルギーを消費します。

## C.3 シールド

### C.3.1 ケーブル・シールド

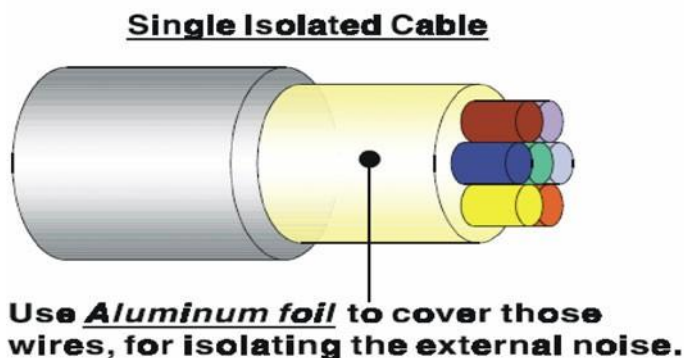


図 C.9 単一絶縁ケーブル

シングル絶縁ケーブル

図C. 9にアイソレーション・ケーブルの構造を示す。この図は、ワイヤーを覆うアルミ箔を螺旋状に巻いたアイソレーション・レイヤーを示している。このスパイラル構造は、外部ノイズからケーブルを保護するシールド層となる。

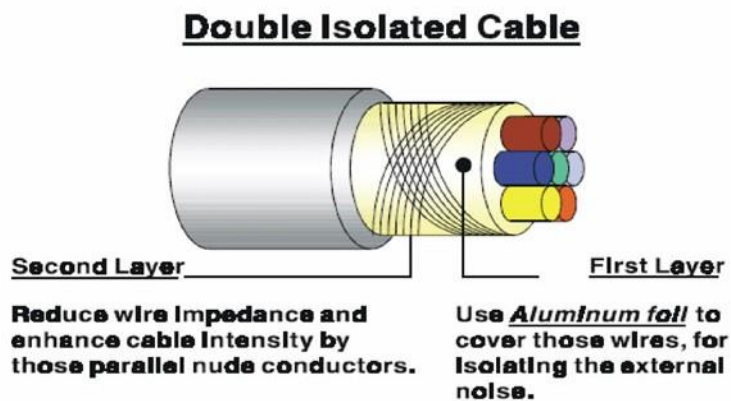


図 C.10 ダブル・アイソレーション・ケーブル

ケーブル ダブル・アイソレーション・ケーブル

図 C. 10 はダブル・アイソレーション・ケーブルの構造を示しています。ここでは、螺旋状に巻かれたアルミ箔の第一の絶縁層が導体を覆い、複数の裸導体からなる第二の絶縁層が螺旋状に巻かれて第一のシールド層の上を横切っています。この螺旋状の構造が絶縁層を形成し、シングル・アイソレーション・ケーブルに比べて外部ノイズをより効率的に低減します。

ノイズを最大限に低減するために、以下のヒントを考慮する必要があります：

- ケーブルのシールドをシグナルグラウンドとして使用しないでください。シールドはノイズを伝搬するように設計されているため、環境ノイズが混入するとシステムに干渉します。
- 特に通信ネットワークでは、シールド密度が高い方が良い。

- 通信アナログI/Oアプリケーションには、二重絶縁ケーブルを使用するのが最適です。
- シールドの両側は、装置内部ではフレームに接続する（EMIを考慮するため）。
- はんだ付けの際、プラスチックカバーを過度に剥がさないでください。

### C.3.2 システム・シールド

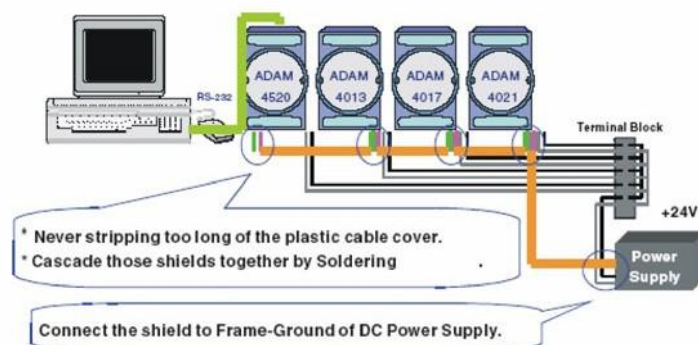
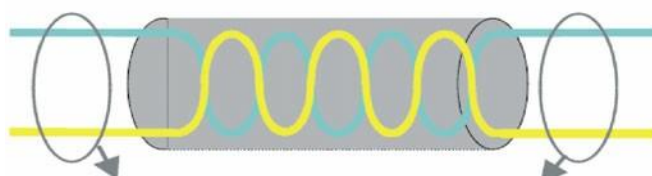


図 C.11 システム・シールド

- プラスチック・ケーブル・カバーを剥き過ぎないでください。STPケーブルの特性が損なわれる可能性がありますし、裸線のシールドはノイズを伝導しやすくなります。
- ハンダ付けによってシールドをカスケード接続する（下図を参照）。
- シールドをDC電源のフレームグラウンドに接続し、ダクトノイズをDC電源のフレームグラウンドに強制的に流す（DC電源のフレームグラウンドはシステムグラウンドに接続する）。

### Characteristic of Cable



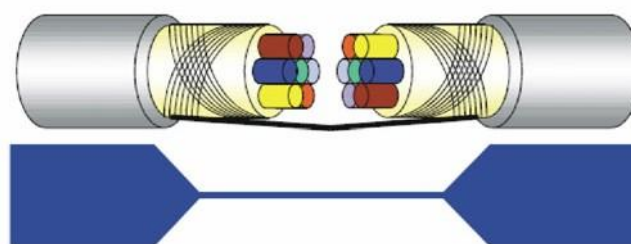
**This will destroy the twist rule.**

**Don't strip off too long of plastic cover for soldering, or will influence the characteristic of twisted pair cable.**

図C.12 ケーブルの特性

はんだ付けのために絶縁体を剥きすぎたはけないことを覚えておいてください。これは、STPケーブルの効果を低下させ、不要なノイズを導入する経路を開く可能性があります。

## System Shielding



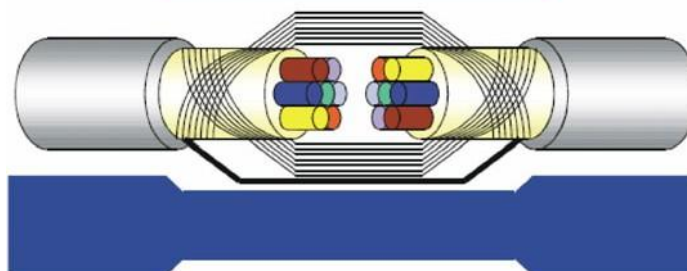
**A difficult way for signal.**

図C.13 システムシールド(1)

### シールド接続 (1)

ケーブルの断線につながりやすいので、システムをセットするときは急がないように注意してください。すべての電子回路がそうであるように、信号は常に最も抵抗の少ない経路をとります。したがって、2本のケーブルの接続が悪いと、信号の経路が悪くなります。その結果、ノイズが抵抗の少ない別の経路を見つけ、干渉を引き起こす可能性があります。

## System Shielding



**A more easy way for signal.**

図C.14 システムシールド(2)

### シールド接続 (2)

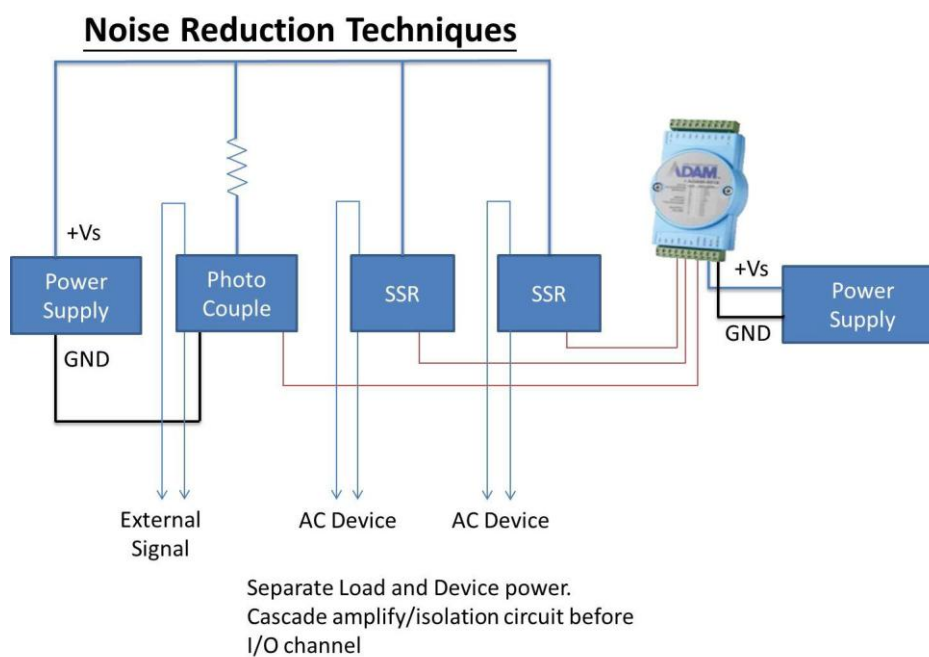
図C.14は、フィル・ハンダ付けが単に信号の通り道を作るだけであることを示している。

## C.4 ノイズ低減テクニック

ノイズリダクションを実施しようとする場合、以下のテクニックを考慮してください：

- シールドされたエンクロージャでノイズ源を分離
- 機密性の高い機器は、シールドされた筐体に入れ、コンピュータ機器から離す
- ノイズ源と信号の間に別々のグラウンドを使用する
- グラウンド／信号のリード線をできるだけ短くする
- ツイストおよびシールドされた信号リード線を使用する
- 基準グラウンドが同じでない場合のみ、片側にシールドを接地する。
- 通信回線の安定性をチェック
- 必要に応じてアースを追加する
- 電源ケーブルの直径は、2.0 mm<sup>2</sup> 以上でなければならない。

- ジャンパーボックス使用時は、アナログI/Oと通信ネットワークに独立した接地が必要
- 必要に応じてノイズ低減フィルタを使用する（TVSなど）
- 負荷に起因するコモンモードノイズを除去するため、コンピュータシステムを電源に近づけることを推奨しているFIPS94規格を参照。



図C.15 ノイズ低減テクニック

## C.5 チェックリスト

- 一点接地ルールに従いましたか？
- ノーマルモード電圧とコモンモード電圧
- DCとACのグラウンドは分離されていますか？
- ノイズ要因を排除する？
- シールドは正しく接続されているか？
- ワイヤーのサイズは正しいですか？
- はんだ付けされた接続は良好ですか？
- 端子のネジは締まっていますか？



付録

D

ADAM-6000のREST

## D.1 RESTの紹介

REST (Representational State Transfer) は、画像表示、リソースリクエスト/レスポンス、メッセージ配信などのウェブアプリケーションやサービスのためのソフトウェアアーキテクチャである。HTTP、URI、XML、HTMLといった一般的なプロトコルや標準と互換性を持たせて開発することができる。スケーラビリティ、シンプルさ、パフォーマンスなどの利点から、RESTはAmazonやYahooのウェブサービスに採用されている。

ADAM-6000 シリーズモジュールのウェブサービスは HTML5 に基づいています。これを他のウェブサービスに統合する必要がある場合は、以下の情報/コマンドリストが実装の参考となります。

## D.2 ADAMのRESTリソース

### D.2.1 アナログ入力

#### GET/analoginput/(all|{id})/value

Request	ContentType: application/x-www-form-urlencoded {id}: the analog input channel ID (starting from 0)
Examples	Use the following URI to get the value of AI-0: http://10.0.0.1/analoginput/0/value Use the following URI to get the all analog input values: http://10.0.0.1/analoginput/all/value
Response	ContentType: text/xml If the result is OK, the content will appear as follows: <?xml version="1.0" ?> <ADAM-6017 status="OK"> <AI> <ID>0</ID> <VALUE>7FFF</VALUE> </AI> </ADAM-6017> If the request fails, the content will appear as follows: <?xml version="1.0" ?> <ADAM-6017 status="{error}"> </ADAM-6017> {error}: The error message
Remarks	If the {id} is out of range, the response will return HTTP status code 501 (not implemented). The content of <VALUE> is in hex format (range 0000~FFFF), which maps to the min./max. of the range (analog input value is 16 bits).

#### GET /analoginput/(all|{id})/range

Request	ContentType: application/x-www-form-urlencoded {id}: the analog input channel ID (starting from 0)
Examples	Use the following URI to get the input range of AI-0: http://10.0.0.1/analoginput/0/range Use the following URI to get all analog input ranges: http://10.0.0.1/analoginput/all/range

	<p>ContentType: text/xml</p> <p>If the result is OK, the content will appear as follows:</p> <pre>&lt;?xml version="1.0" ?&gt; &lt;ADAM-6017 status="OK"&gt;   &lt;AI&gt;     &lt;ID&gt;0&lt;/ID&gt;     &lt;RANGE&gt;7&lt;/RANGE&gt;     &lt;NAME&gt;4~20 mA&lt;/NAME&gt;     &lt;MAX&gt;20&lt;/MAX&gt;   &lt;/AI&gt;   &lt;MIN&gt;4&lt;/MIN&gt;   &lt;UNIT&gt;mA&lt;/UNIT&gt; &lt;/ADAM-6017&gt;</pre> <p>If the request fails, the content will appear as follows:</p> <pre>&lt;?xml version="1.0" ?&gt; &lt;ADAM-6017 status="{error}"&gt; &lt;/ADAM-6017&gt; {error}: The error message</pre>
Response	
Remarks	If the {id} is out of range, the response will return HTTP status code 501 (not implemented).

## D.2.2 アナログ出力

### GET /analogoutput/{all|{id}}/value

Request	<p>ContentType: application/x-www-form-urlencoded</p> <p>{id}: the analog output channel ID (starting from 0)</p>
Examples	<p>Use the following URI to get the value of AI-0: http://10.0.0.1/analogoutput/0/value</p> <p>Use the following URI to get the all analog output values: http://10.0.0.1/analogoutput/all/value</p>
Response	<p>ContentType: text/xml</p> <p>If the result is OK, the content will appear as follows:</p> <pre>&lt;?xml version="1.0" ?&gt; &lt;ADAM-6224 status="OK"&gt;   &lt;AO&gt;     &lt;ID&gt;0&lt;/ID&gt;     &lt;VALUE&gt;0FFF&lt;/VALUE&gt;   &lt;/AO&gt; &lt;/ADAM-6224&gt;</pre> <p>If the request fails, the content will appear as follows:</p> <pre>&lt;?xml version="1.0" ?&gt; &lt;ADAM-6224 status="{error}"&gt; &lt;/ADAM-6224&gt; {error}: The error message</pre>
Remarks	<p>If the {id} is out of range, the response will return HTTP status code 501 (Not implemented)</p> <p>The content of &lt;VALUE&gt; is in hex format (range 0000~0FFF), which maps to the min./max. value of the range (analog output value is 12 bits).</p>

### POST /analogoutput/all/value

Request	ContentType: application/x-www-form-urlencoded
---------	--

例	<p>Use the following URI to set the analog output value(s)  <a href="http://10.0.0.1/analogoutput/all/value">http://10.0.0.1/analogoutput/all/value</a>          The data accompanying the request will be given as {name}={value} pair(s)          {name}: The channel name (e.g., AO-0)          {value}: The value to be set to the indicated channel          For example, if the request is going to set Channels 0, 1, and 2 to a value of 3, then the name-value pairs will appear as follows:          AO0=00FF&amp;AO1=0000&amp;AO2=0FFF          ContentType: text/xml          The content will appear as follows:</p>
応答	<pre>&lt;?xml version="1.0" ?&gt; &lt;ADAM-6224 status="{status}"&gt; &lt;/ADAM-6224&gt;</pre> <p>{status}: The result (if successful, the result will be 'OK'; otherwise, the result will be the error message).</p>
Remarks	<p>The {value} of the post data is in HEX format and from 0000 to 0FFF, which maps to the min./max. value of the range (analog output value is 12 bits).</p>

### GET /analogoutput/(all|{id})/range

Request	<p>ContentType: application/x-www-form-urlencoded          {id}: the AO channel ID (starting from 0)</p>
Examples	<p>Use the following URI to get the AO-0 range:  <a href="http://10.0.0.1/analogoutput/0/range">http://10.0.0.1/analogoutput/0/range</a>          Use the following URI to get all analog output ranges:  <a href="http://10.0.0.1/analogoutput/all/range">http://10.0.0.1/analogoutput/all/range</a></p>
Response	<p>ContentType: text/xml          If the result is OK, the content will appear as follows:  <pre>&lt;?xml version="1.0" ?&gt; &lt;ADAM-6224 status="OK"&gt;   &lt;AO&gt;     &lt;ID&gt;0&lt;/ID&gt;     &lt;RANGE&gt;7&lt;/RANGE&gt;     &lt;NAME&gt;4~20 mA&lt;/NAME&gt;     &lt;MAX&gt;20&lt;/MAX&gt;   &lt;/AO&gt;   &lt;MIN&gt;4&lt;/MIN&gt;   &lt;UNIT&gt;mA&lt;/UNIT&gt; &lt;/ADAM-6224&gt;</pre>         If the request fails, the content will appear as follows:  <pre>&lt;?xml version="1.0" ?&gt; &lt;ADAM-6224 status="{error}"&gt; &lt;/ADAM-6224&gt;</pre>         {error}: The error message</p>
Remarks	<p>If the {id} is out of range, the response will return HTTP status code 501 (not implemented)</p>

## D.2.3 デジタル入力

### GET /digitalinput/(all|{id})/value

Request	<p>ContentType: application/x-www-form-urlencoded          {id}: the DI channel ID (starting from 0)</p>
Examples	<p>Use the following URI to get the DI-0 value:  <a href="http://10.0.0.1/digitalinput/0/value">http://10.0.0.1/digitalinput/0/value</a>          Use the following URI to get all digital input values:  <a href="http://10.0.0.1/digitalinput/all/value">http://10.0.0.1/digitalinput/all/value</a></p>

Response	<p>ContentType: text/xml</p> <p>If the result is OK, the content will appear as follows:</p> <pre>&lt;?xml version="1.0" ?&gt; &lt;ADAM-6050 status="OK"&gt;   &lt;DI&gt;     &lt;ID&gt;0&lt;/ID&gt;     &lt;VALUE&gt;0&lt;/VALUE&gt;   &lt;/DI&gt; &lt;/ADAM-6050&gt;</pre> <p>If the request fails, the content will appear as follows:</p> <pre>&lt;?xml version="1.0" ?&gt; &lt;ADAM-6050 status="{error}"&gt; &lt;/ADAM-6050&gt; {error}: The error message</pre>
Remarks	If the {id} is out of range, the response will return HTTP status code 501 (not implemented)

## D.2.4 デジタル出力

### GET /digitaloutput/(all|{id})/value

Request	<p>ContentType: application/x-www-form-urlencoded</p> <p>{id}: the digital output channel ID (starting from 0)</p>
Examples	<p>Use the following URI to get the DO-0 value:</p> <pre>http://10.0.0.1/digitaloutput/0/value</pre> <p>Use the following URI to get all digital output values:</p> <pre>http://10.0.0.1/digitaloutput/all/value</pre>
Response	<p>ContentType: text/xml</p> <p>If the result is OK, the content will appear as follows:</p> <pre>&lt;?xml version="1.0" ?&gt; &lt;ADAM-6050 status="OK"&gt;   &lt;DO&gt;     &lt;ID&gt;0&lt;/ID&gt;     &lt;VALUE&gt;1&lt;/VALUE&gt;   &lt;/DO&gt; &lt;/ADAM-6050&gt;</pre> <p>If the request fails, the content will appear as follows:</p> <pre>&lt;?xml version="1.0" ?&gt; &lt;ADAM-6050 status="{error}"&gt; &lt;/ADAM-6050&gt; {error}: The error message</pre>
Remarks	If the {id} is out of range, the response will return HTTP status code 501 (not implemented)

### POST /digitaloutput/all/value

Request	<p>ContentType: application/x-www-form-urlencoded</p> <p>Use the following URI to set the digital output value(s):</p> <pre>http://10.0.0.1/digitaloutput/all/value</pre> <p>The data accompanying the request will be given as {name}={value} pair(s)</p>
Examples	<p>{name}: The channel name (e.g., DO-0)</p> <p>{value}: The value to be set to the indicated channel</p> <p>For example, if the request is going to set Channel 0, 1, and 2 to a value of 1, then the name-value pairs would be as follows:</p> <pre>DO0=1&amp;DO1=1&amp;DO2=1</pre>

Response	ContentType: text/xml The content will appear as follows: <?xml version="1.0" ?> <ADAM-6050 status="{status}"> </ADAM-6050> {status}: The result. If successful, the result will be "OK"; otherwise, the result will be the error message.
Remarks	

## D.2.5 カウンター

### GET /counter/{all|{id}}/value

Request	ContentType: application/x-www-form-urlencoded {id}: the counter channel ID (starting from 0)
Examples	Use the following URI to get the Counter-0 value: http://10.0.0.1/counter/0/value Use the following URI to get the all Counter values: http://10.0.0.1/counter/all/value
Response	ContentType: text/xml If the result is OK, the content will appear as follows: <?xml version="1.0" ?> <ADAM-6051 status="OK"> <CNT> <ID>0</ID> <VALUE>102938</VALUE> </CNT> </ADAM-6051> If the request fails, the content will appear as follows: <?xml version="1.0" ?> <ADAM-6051 status="{error}"> </ADAM-6051> {error}: The error message
Remarks	If the {id} is out of range, the response will return HTTP status code 501 (not implemented)



[www.advantech.com](http://www.advantech.com)

お見積り前に仕様をご確認ください。このガイドは参考用です。

製品の仕様は予告なく変更されることがあります。

本書のいかなる部分も、出版社からの書面による事前の許可なく、  
電子的、複写、記録、その他のいかなる方法によっても複製することを  
禁じます。

すべてのブランド名および製品名は、各社の商標または登録商標です。

© Advantech Co, Ltd. 2022