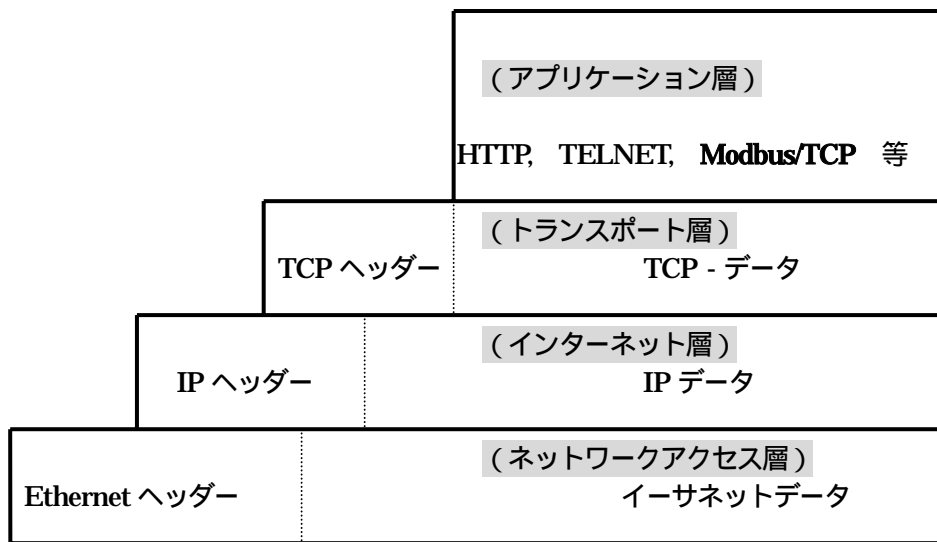


## 5 - 2 Ethernet Modbus / TCP プロトコル

### 5 - 2 - 1 Modbus/TCP プロトコル

イーサネットバスカップラでは TCP/IP のアプリケーション層においてデータの交換を行いますので、TCP/IP のトランスポート層以下の機能を考慮する必要はありません。BootP テーブルソフトウェアで書込まれた IP アドレスを指定して通信を確立した後は、HTTP や TELNET 等と同様にアプリケーションレベルでのデータ通信を行います。通信プロトコルはオープンプロトコルである「**Modbus/TCP**」を使用しており、このアプリケーションプロトコルはポート番号 502 で通信を行います。



### 5 - 2 - 2 Modbus/TCP のデータ構造

アプリケーション層での「Modbus/TCP」のデータは以下のように byte 単位で区切られます。実際の TCP/IP 通信上では各々のバイトデータを 16 進数に変換して通信を行います。

Byte No	0	1	2	3	4	5	6	7	8 ~
データの 内容	転送 ID 「0」で 固定		プロトコル ID 「0」で 固定		Byte6 以降 の総バイト数		エント ID 使用せず、 「1」で固定	Modbus/TCP の機能コード	機能コードに続 く転送データ

<データ例>

Byte No	0	1	2	3	4	5	6	7	8 ~
Byte データ	0	0	0	0	0	6	1	1	0   8
16進数 (転送データ)	00 00		00 00		00 06		01	01	00 08

### 5 - 2 - 3 データ読み出しと書き込みの機能分類

スレーブであるワゴ I/O からデータを読み出、あるいはデータを書込む動作は、ホスト（パソコン）側から送信される Modbus/TCP 中のコマンド（Byte 7）にある機能コードによって区別します。同一のノード内に入力モジュールと出力モジュールが混在していても、読み出し動作（入力モジュール）と書き込み動作（出力モジュール）はそれぞれ独立した機能として扱います（シリアル Modbus プロトコルと同じ）すなわち読み出し動作時は入力モジュールのみが対象となり、書き込み動作時は出力モジュールのみが対象となります。尚、750-404 カウンターモジュールについては入力、出力ともに 6 バイト（48 ビット）のビット幅を占有しますのでご注意ください。

読み出し、書き込み動作は、以下の表のように 8 種類の機能に分類されます。5 - 2 - 4 以降に各々の動作の詳細を説明してありますが、表中の「機能コード」は Modbus/TCP の機能コード（16 進数）に対応しています。

#### ワゴ I/O からデータを読み出す（入力）

分類	機能	対象モジュール	「 」内は 16 進数 機能コード：16 進数 (Byte 7 のデータ)
	ビット単位で範囲を指定してデータを読みだす。	デジタル入力モジュール	「01」または「02」
	ワード単位で範囲を指定してデータを読み出す	アナログ入力モジュールおよびデジタル入力モジュール	「03」または「04」
	電源投入以降に実行されたコマンドの回数を読み出す	バスカップラ	「0B」

#### ワゴ I/O へデータ書き込む（出力）

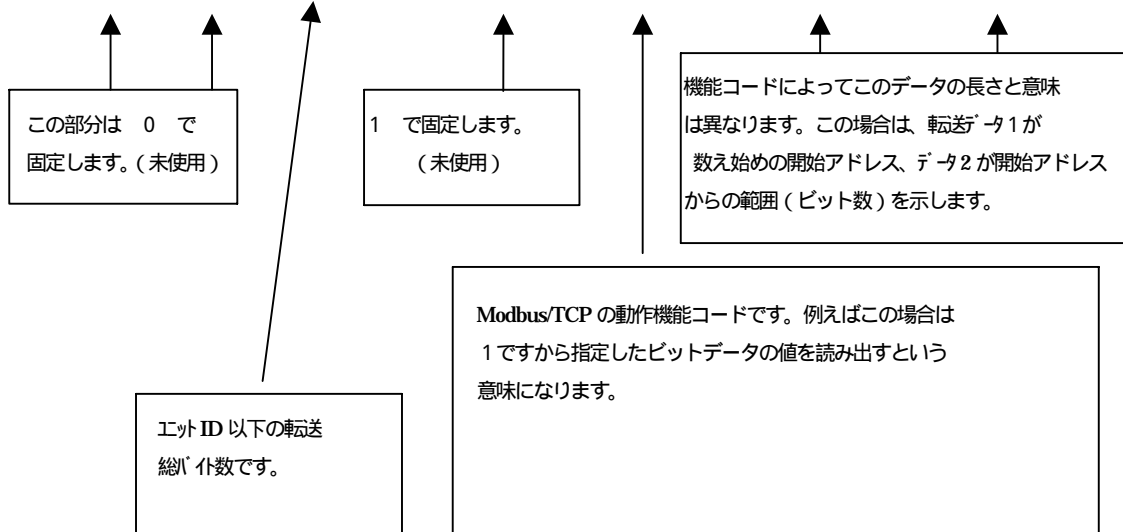
分類	機能	対象モジュール	機能コード：16 進数
	単一のビットを指定してデータを書き込む	デジタル出力モジュール	「05」
	範囲を指定して複数ビットにデータを書き込む	デジタル出力モジュール	「0F」
	単一のワードを指定してデータを書き込む	アナログ出力モジュールおよびデジタル出力モジュール	「06」
	範囲を指定して複数ワード単位でデータを書き込む	アナログ出力モジュールおよびデジタル出力モジュール	「10」
	最初の 8 ビット分の出力データを読む	デジタル出力モジュール	「07」

### 5 - 2 - 4 読み出し、書き込みコマンドの送出方法

マスター（ホストコンピュータ）とスレーブ（ワゴI/O）のIPアドレスにてTCP接続が確立された後はPort502経由で通信を行います。ホストコンピュータからのリクエスト信号を受け取ったワゴI/Oは機能コードで指定された動作を行い、確認のアンサーバックをホストコンピュータへ返信します。

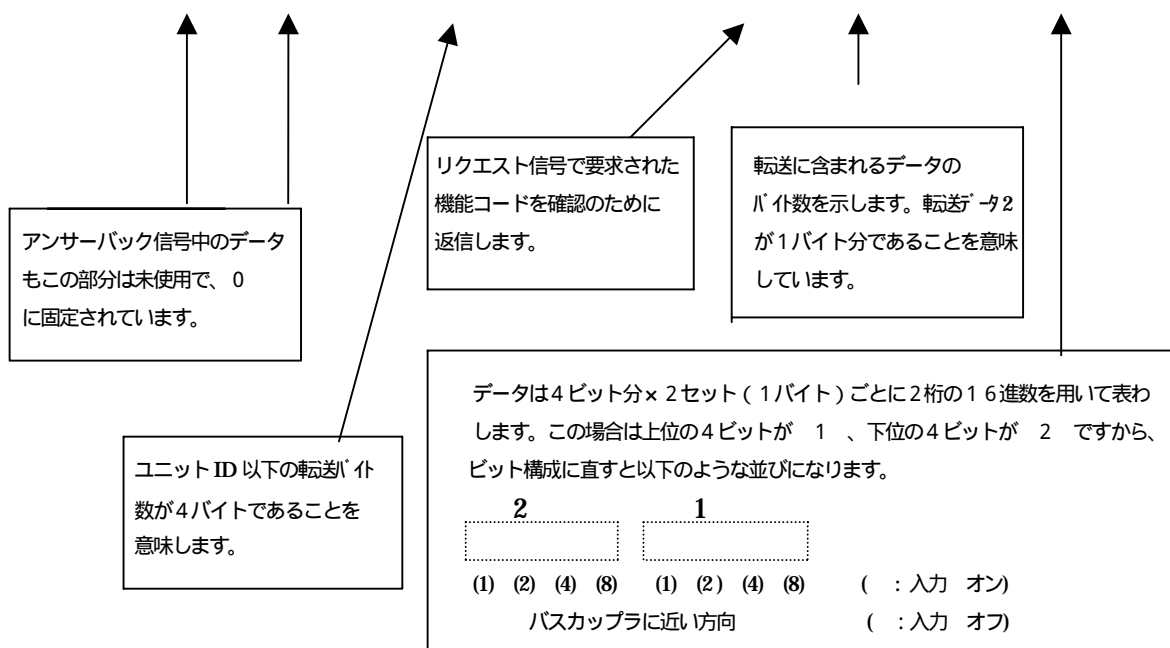
リクエスト信号（マスター → ワゴI/O）の例

Byte No	転送ID		プロトコルID		転送バイト数		ユニットID	機能コード	転送データ1		転送データ2	
Byte データ	0	0	0	0	0	6	1	1	0	0	0	8
16進数 (転送コード)	00	00	00	00	00	06	01	01	00 00 開始アドレス		00 08 範囲(ビット)	



アンサーバック信号（ワゴI/O → マスター）の例

Byte No	転送ID		プロトコルID		転送バイト数		ユニットID	機能コード	転送データ1		転送データ2	
Byte データ	0	0	0	0	0	4	1	1	1	18		
16進数 (転送コード)	00	00	00	00	00	04	01	01	01	データのバイト数		12 データ



## 5 - 2 - 5 ワゴI/O からデータを読み出す

### ビット単位で範囲を指定してデータを読み出す (機能コード 「01」または「02」)

このコマンドはビット単位で開始位置を設定、ビット単位で範囲を指定して、その入力値(「1」か「0」または「High」か「Low」)を読み出します。リクエスト信号とアンサーバックの信号の例は5-2-4で説明したとおりです。このコマンドは**デジタル入力モジュールのみが対象**となります。また、ビット単位での最高点数は256点(32バイト)分です。すなわちリクエスト信号の中の開始アドレス(転送データ1)における最高値は2桁16進数で「FF」です。

アンサーバック信号中のデータ(転送データ2)は8ビット単位(バイト単位)で区切られ、バスカップラに近いデジタル入力モジュールの4ビット分の16進数値がデータの低位桁によって表され、バスカップラから遠いモジュールの4ビット分の16進数値が上位桁に表されます。

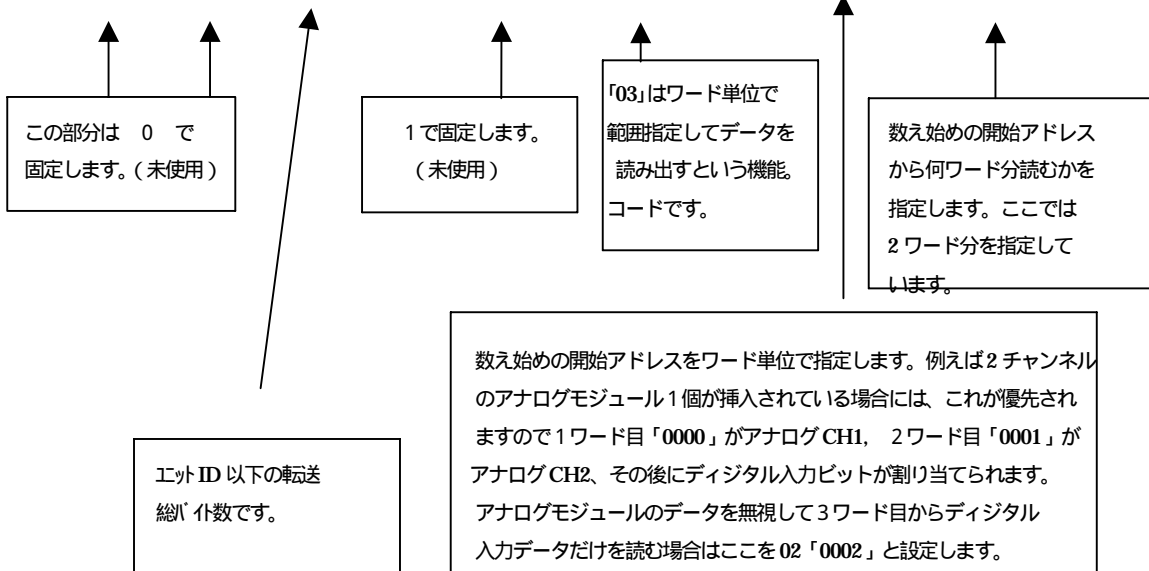
### 範囲を指定してワード単位でデータを読み出す (機能コード 「03」または「04」)

このコマンドは16ビット単位(ワード単位)で範囲を指定してワゴI/Oからデータを読み出します。デジタル入力モジュールが接続されている場合は、アンサーバック信号中の16進数値の構成は4ビット×2セット(1バイト)を単位として上位と下位2桁ずつの組み合わせで表します。またアナログモジュールが挿入されていた場合はアナログ入力モジュール(1チャンネルあたり16ビット)が優先されます。すなわちデジタル入力モジュールのデータはアナログデータの後に続くことになります。

図5 - 2 - 5

リクエスト信号 (マスター ワゴI/O)

Byte No	転送ID		プロトコルID		転送バイト数		エントID	機能コード	転送データ1		転送データ2	
Byte データ	0	0	0	0	0	6	1	3	0	0	0	2
16進数 (転送コード)	00	00	00	00	00	06	01	03	00	00	00	02
									開始アドレス		何ワード分	



アンサーバック信号 (ワゴI/O マスター)

Byte No	転送ID		プロトコルID		転送バイト数		エットID	機能コード	転送データ1	転送データ2	転送データ3	転送データ4	転送データ5
Byte データ	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>7</b>	<b>1</b>	<b>3</b>	<b>4</b>	<b>63</b>	<b>251</b>	<b>0</b>	<b>0</b>
16進数 (転送コード)	00 00		00 00		00 07		01	03	04 データ量	3F FB 読込データ		00 00 読込データ	

エットID以降のデータ量をByte単位で表します。

転送データ量 (転送データ2以降) が4バイトであることを示しています。

データは4ビット分×2セット(1バイト)ごとに4桁の16進数(1ワード)を用いて表わします。この場合は下位のワードが「3FFB」で、この16ビット分が下位ワード(レジスタ0)になります。これを分解すると以下のようなビット構成になります。仮にすべてがデジタル入力である場合はビットの並びはバスカップラに近い方から以下の並びと同じになります。

下位4ビット	上位4ビット	下位4ビット	上位4ビット				
<b>B</b>	<b>F</b>	<b>F</b>	<b>3</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>
1 2 4 8	1 2 4 8	1 2 4 8	1 2 4 8				

バスカップラに近い方向 ( : 入力 オン) ( : 入力 オフ)

2CHアナログ入力モジュールが挿入されているときは、1チャンネルあたり16ビットが割り当てられます。下位ワード(レジスタ0)の12ビット分を使います。「3FFB」の場合は、以下のように分解され、12ビットのうちの11ビット分がオンの状態を示します。ただし、アナログの場合は下位3ビットは使用せず、通常はビットがHiのままになりますので数値は「3FFF」になります。

3	F	F	B
8 4 2 1	8 4 2 1	8 4 2 1	8 4 2 1

A/D変換時の有効な12ビット

仮に0~10Vのアナログ入力モジュールに適用すると以下ようになります。

$$\frac{2^{11} \times 0 + 2^{10} \times 1 + 2^9 \times 1 + 2^8 \times 1 + 2^7 \times 1 + 2^6 \times 1 + 2^5 \times 1 + 2^4 \times 1 + 2^3 \times 1 + 2^2 \times 1 + 2^1 \times 1 + 2^0 \times 1}{2^{12}} \times 10V$$

$$= \frac{2047}{4095} \times 10V = 5V$$

あるいは、電圧アナログモジュールは 0008(hex) ~ 7FFF(hex) が 0~10V に相当しますので、3FFB(hex)=16379, よって

$$10V \div (7FFF[hex] - 0008[hex]) \times (3FFB[hex] - 0008[hex]) =$$

$$= 10V \div (32768 - 8) \times (16379 - 8) = 5V$$

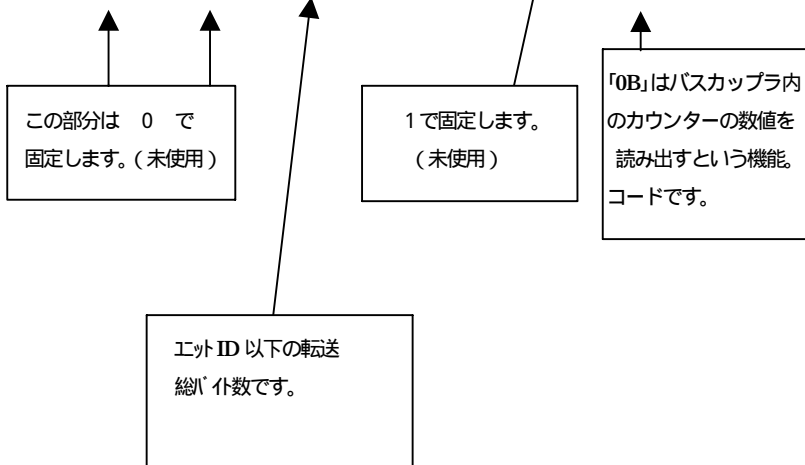
尚、各モジュールのビットフォーマットについては各モジュールの説明の項をご覧ください。

実行されたコマンド回数を読み出す (機能コード「0B」)

このコマンドはバスカップラ内にあるカウンタのカウンタ数値を読み出します。バスカップラ内のコマンドカウンタは電源が投入された後、正常に動作が行われたコマンドの回数をカウントしていますので、マスター側から送り出したコマンドの数とこの数をチェックすることによってそのノードとの通信が正常に行われているかどうかを確認することができます。

リクエスト信号 (マスター ワゴI/O)

Byte No	転送ID		プロトコルID		転送バイト数		エイトID	機能コード
Byte データ	0	0	0	0	0	2	1	11
16進数 (転送コード)	00 00		00 00		00 02		01	0B



アンサーバック信号 (ワゴI/O マスター)

Byte No	転送ID		プロトコルID		転送バイト数		エイトID	機能コード	転送データ1		転送データ2	
Byte データ	0	0	0	0	0	6	1	11	0	0	2	212
16進数 (転送コード)	0000		0000		0006		01	0B	0000		02 D4 カウンタ数値	

エイトID以降のデータ量をByte単位で表します。

ステータスをあらわします。常に「0000」です。

電源投入以降のコマンド実行回数が724回目であることを示しています。16ビット、最大65535回までのカウントができます。

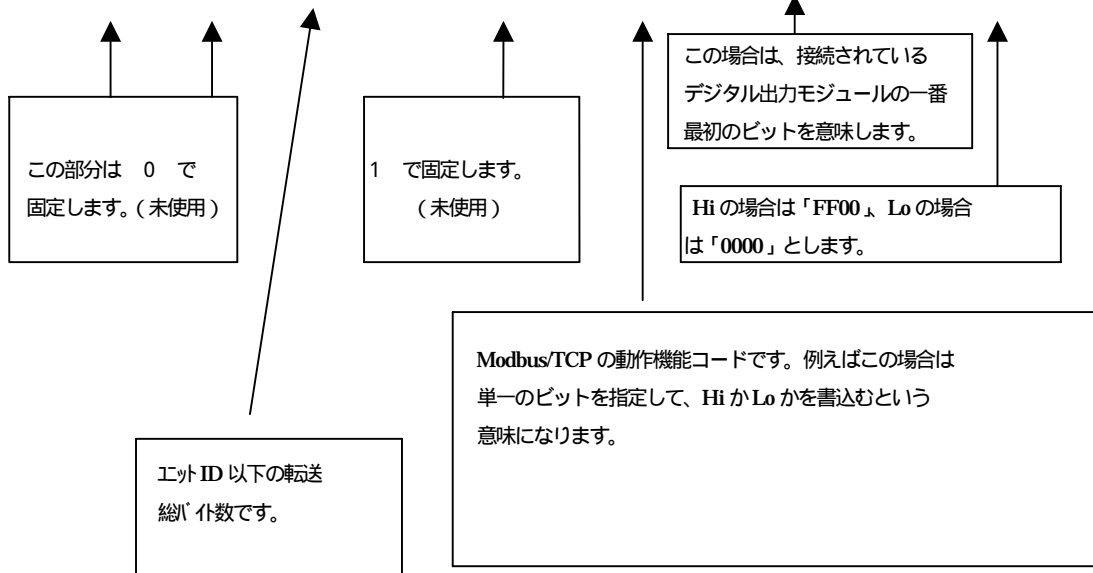
## 5 - 2 - 6 ワゴ I/O ヘデータを書き込む

単一のビットを指定してデータを書き込む (機能コード「05」)

このコマンドは単一のビットを指定して、ON/OFFの出力を実行します。出力の実行は単一のビットですが、設定できる範囲は0000~00FFまでの256ポイントの範囲です。この機能の対象モジュールはデジタル出力モジュールのみです。アナログ出力モジュールは対象外です。

リクエスト信号 (マスター ワゴ I/O) の例

Byte No	転送 ID		プロトコル ID		転送バイト数		ユニット ID	機能コード	転送データ1		転送データ2	
Byte データ	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>6</b>	<b>1</b>	<b>5</b>	<b>0</b>	<b>0</b>	<b>255</b>	<b>0</b>
16進数 (転送コード)	00	00	00	00	00	06	01	05	00	00	FF	00
									開始アドレス		Hi/Lo の指示値	



アンサーバック信号 : ( ワゴ I/O マスター )

Byte No	転送 ID		プロトコル ID		転送バイト数		ユニット ID	機能コード	転送データ1		転送データ2	
Byte データ	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>6</b>	<b>1</b>	<b>5</b>	<b>0</b>	<b>0</b>	<b>255</b>	<b>0</b>
16進数 (転送コード)	00	00	00	00	00	06	01	05	00	00	FF	00
									開始アドレス		Hi/Lo の指示値	

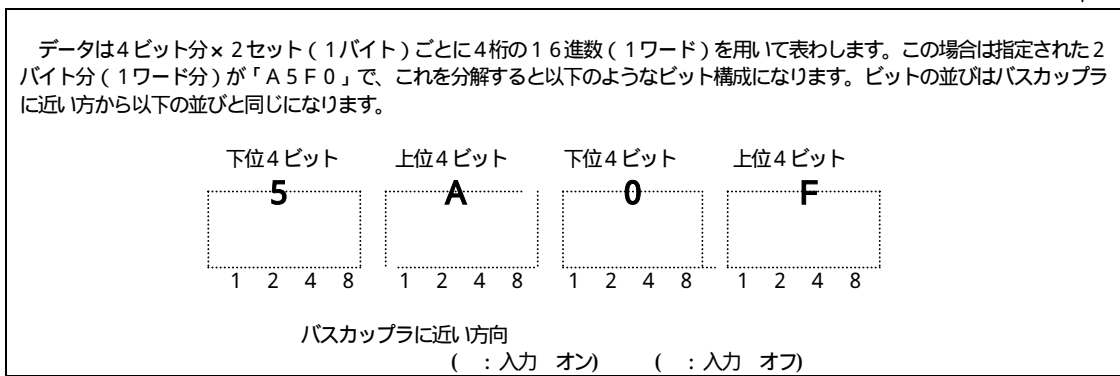
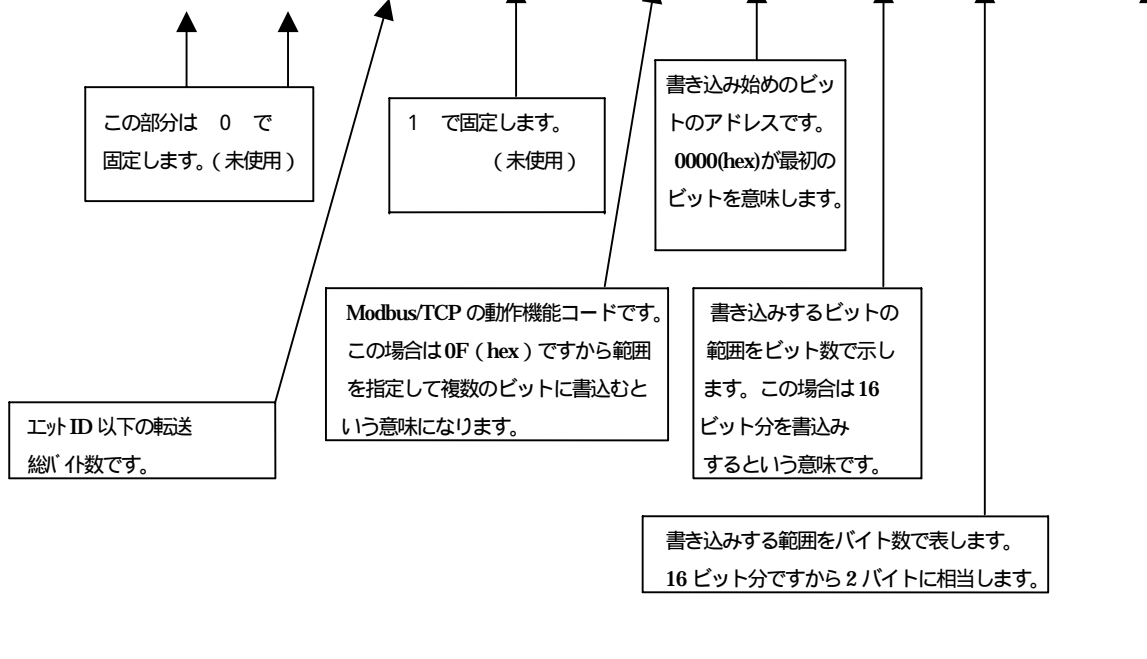
この機能コードの場合は、動作した内容をそのまま確認のため返信しますので、送信した内容と同じ値がアンサーバック信号として返信されます。

**範囲を指定して複数ビットにデータを書き込む (機能コード「0F」)**

このコマンドは複数のビットを指定して、データを書き込みます。アナログモジュールは対象外です。

リクエスト信号 (マスター ワゴI/O) の例

Byte No	転送ID		プロトコル ID		転送バイト数		ユニットID	機能コード	転送データ1		転送データ2		転送データ3	転送データ4	
Byte データ	0	0	0	0	0	9	1	15	0	0	0	16	2	165	240
16進数 (転送コード)	00	00	00	00	00	09	01	0F	00	00	00	10	02	A5	F0
									開始アドレス	範囲		範囲	書き込み値		



アンサーバック信号 (ワゴI/O マスター)

Byte No	転送ID		プロトコル ID		転送バイト数		ユニットID	機能コード	転送データ1		転送データ2		以下省略
Byte データ	0	0	0	0	0	6	1	15	0	0	0	16	
16進数 (転送コード)	00	00	00	00	00	06	01	0F	00	00	00	10	
									開始アドレス	範囲			

アンサーバック信号は上のように転送データ2までの内容を確認のため返信します。



**単一のワードを指定してデータを書き込む**

このコマンドは単一のレジスタ（16ビット分）に対してデータを書き込むためのコマンドです。例えばアナログ出力モジュールが挿入されている場合にはこれが優先されます。アドレス「0000」（レジスタ0）がアナログ1CH目、アドレス「0001」（レジスタ1）がアナログ2CH目になります。

リクエスト信号（マスター ワゴI/O）

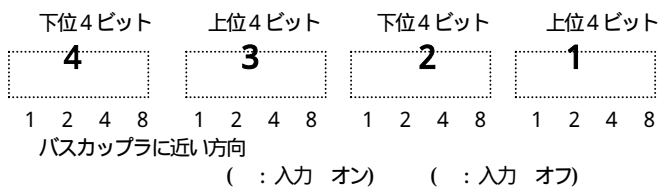
Byte No	転送ID		プロトコルID		転送バイト数		ユニットID	機能コード	転送データ1		転送データ2	
Byte データ	0	0	0	0	0	6	1	6	0	1	18	52
16進数 (転送コード)	00	00	00	00	00	09	01	06	00	01	12	34
									開始アドレス		書き込み値	

転送ID、プロトコルIDは「0000」で固定、ユニットIDも1で固定、転送バイト数はユニットID以下の総バイト数をあらわします。

単一のレジスタ（16ビット分）のみを書き込むコマンドは「06」です。

数え始めのスタートアドレスをワード単位で指定します。16進数で設定します。例えば、2チャンネルのアナログ出力モジュール1個が挿入されているときには、これが優先されますので、この場合は1ワード目「0000」がアナログCH1、2ワード目「0001」がアナログCH2、その後にデジタル出力ビットが、バスカップラに近い方から割り当てられます。  
アナログ出力モジュールのデータを無視して3ワード目からデジタル出力モジュールにのみデータを書き込みたい場合はここを「0002」と設定します。

データは4ビット分×2セット（1バイト）ごとに4桁の16進数（1ワード）を用いて表わしますこのコマンドはレジスタ1個分、すなわち16ビット分の書き込みしかできませんので、4ビット×2セット×2セット分のデータを以下のように4桁の16進数で表現します。仮にすべてがデジタル出力モジュールである場合はビットの並びはバスカップラに近い方から以下の並びと同じになります。



2CHアナログ出力モジュールが挿入されているときは、アドレスを「0000」に設定してデータを送ると、まずアナログ出力のCH1にデータが適用されます。アナログ出力1CHあたり16ビット（1ワード）が割り当てられ、以下のように12ビット分を使います。例えば10Vの出力モジュールで「1234」を出力すると1.42Vになります。



$$\frac{2^{11} \times 0 + 2^{10} \times 0 + 2^9 \times 1 + 2^8 \times 0 + 2^7 \times 0 + 2^6 \times 1 + 2^5 \times 0 + 2^4 \times 0 + 2^3 \times 0 + 2^2 \times 1 + 2^1 \times 1 + 2^0 \times 0}{2^{12}} \times 10V$$

$$= \frac{582}{4095} \times 10V = 1.42V$$

### アンサーバック信号 (ワゴI/O マスター)

Byte No	転送ID		プロトコルID		転送バイト数		ユニットID	機能コード	転送データ1		転送データ2	
Byte データ	0	0	0	0	0	6	1	6	0	1	18	52
16進数 (転送コード)	00	00	00	00	00	06	01	06	00	01	12	34

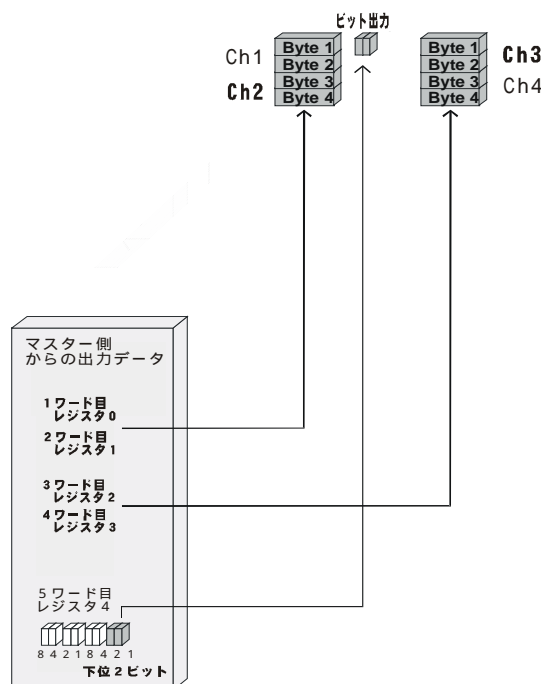
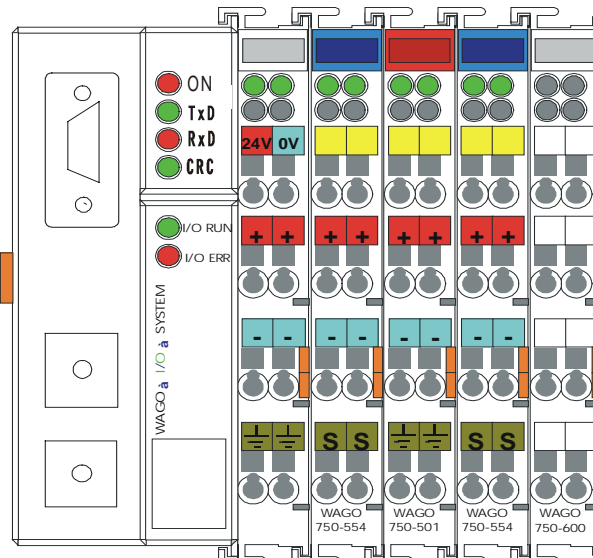
アンサーバック信号は確認の意味でリクエスト信号と同じ内容を返信します。

#### 範囲を指定して複数ワード単位でデータを書き込む

このコマンドは複数のレジスタ (16ビット分×N) に対してデータを書き込むためのコマンドです。アナログ出力モジュールが挿入されている場合にはこれが優先されて割り当てられます。出力データはレジスタの低い方から順番に4桁の16進数で送し出します。

図5-4-1

### ワゴI/Oへデータを書き込む (出力)



リクエスト信号 (マスター ワゴ I/O)

Byte No	転送 ID		プロトコル ID		転送バイト数		ユニット ID	機能コード	転送データ 1		転送データ 2		転送データ 3		転送データ 4		転送データ 5	
Byte データ	0	0	0	0	0	11	1	16	0	0	0	2	4	18	52	86	12	0
16進数 (転送コード)	00	00	00	00	00	0B	01	10	00	00	00	02	04	12	34	56	78	
									開始 アドレス	範囲 (ワード)		範囲 (バイト)		書き込み値		書き込み値		

数え始めのスタートアドレスをワード単位で指定します。16進数で設定します。例えば、2チャンネルのアナログ出力モジュール1個が挿入されているときには、これが優先されますので、この場合は1ワード目「0000」アナログCH1、2ワード目「0001」がアナログCH2、その後デジタル出力ビットが、バスカップラに近い方から割り当てられます。  
アナログ出力モジュールのデータを無視して3ワード目からデジタル出力モジュールにのみデータを書き込みたい場合はここを「0002」と設定します。

出力する範囲をワード単位で設定します。この場合はスタートアドレス「0000」から数えて「0002」(2ワード分=32ビット分)を範囲とするという意味になります。

スタートアドレスから何バイト分を範囲にするかを指定します。この場合は32ビット分を範囲設定しているため4バイトを設定しています。出力範囲ビットカバーする数値に設定してください。

データは4ビット分×2セット(1バイト)ごとに2桁の16進数を用いて表わしますこのコマンドはレジスタを複数個分、すなわち16ビット分×n個の書き込みが可能です。16ビットごとに4桁の16進数で置き換えながらデータを作ります。仮にすべてがデジタル出力モジュールである場合はビットの並びはバスカップラに近い方から以下の並びと同じになります。

4	3	2	1	8	7	6	5
1 2 4 8	1 2 4 8	1 2 4 8	1 2 4 8	1 2 4 8	1 2 4 8	1 2 4 8	1 2 4 8

バスカップラに近い方向

( : 入力 オン) ( : 入力 オフ)

アナログ出力モジュールが挿入されているときは、アドレスを「0000」に設定してデータを送ると、まずアナログ出力のCH1にデータが適用されます。アナログ出力1CHあたり16ビット(1ワード)が割り当てられ、以下のように12ビット分を使います。例えば「1234」のデータをCH1に送ると、以下のようにVが出力されます。

1	2	3	4
8 4 2 1	8 4 2 1	8 4 2 1	8 4 2 1

A/D変換時の有効な12ビット

$$\frac{2^{11} \times 0 + 2^{10} \times 0 + 2^9 \times 1 + 2^8 \times 0 + 2^7 \times 0 + 2^6 \times 1 + 2^5 \times 0 + 2^4 \times 0 + 2^3 \times 0 + 2^2 \times 1 + 2^1 \times 1 + 2^0 \times 0}{2^{12}} \times 10V$$

$$= \frac{582}{4095} \times 10V = 1.42V$$

### アンサーバック信号 (ワゴ I/O マスター)

Byte No	転送 ID		プロトコル ID		転送バイト数		ユニット ID	機能コード	転送データ 1		転送データ 2	
Byte データ	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>6</b>	<b>1</b>	<b>16</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>2</b>
16進数 (転送コード)	00	00	00	00	00	06	01	10	00	00	00	02

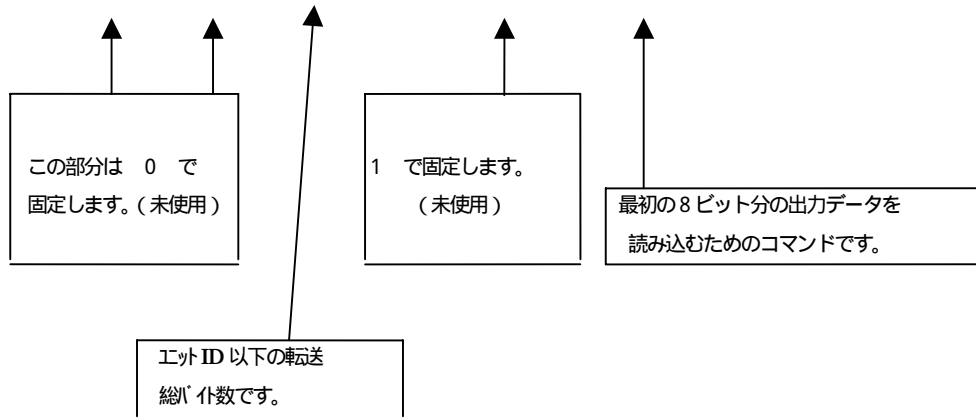
アンサーバック信号は上のように転送データ 2 までの内容を確認のため返信します。

最初の 8 ビット分の出力データを読む

このコマンドは既に出力されている 8 ビット分の出力データの状態を確認するためのコマンドです。フィールドバスカップラに一番近い 8 ビット分のデジタル出力モジュールのみが対象になります。

### リクエスト信号 (マスター ワゴ I/O)

Byte No	転送 ID		プロトコル ID		転送バイト数		ユニット ID	機能コード
Byte データ	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>2</b>	<b>1</b>	<b>7</b>
16進数 (転送コード)	00	00	00	00	00	02	01	07



### アンサーバック信号 : (ワゴ I/O マスター)

Byte No	転送 ID		プロトコル ID		転送バイト数		ユニット ID	機能コード	転送データ 1
Byte データ	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>3</b>	<b>1</b>	<b>7</b>	<b>91</b>
16進数 (転送コード)	00	00	00	00	00	06	01	07	5B データ

