

WAT-910BD

H/W マニュアル

Ver.1.10

2016.07.01

ワテック株式会社

改訂履歴

Ver. No.	Date	Changes	Remark
1.00	2013.12.25		初版
1.02	2014.10.23	P.34 MANUAL GAIN ビット範囲訂正 P.34 HSBLC WINDOW H/V SIZE アドレス誤記訂正 P.41 BPC LEVEL ビット範囲訂正	
1.10	2016.07.01	P.16-19 ラインデータフォーマット(EIA/CCIR)に H/Vブランキングパルスタイミングを加筆 P.23 誤記訂正 (220k[Ω]→4.7k[Ω]) それに伴い、図6-3を訂正	

目次

1.	はじめに	1
2.	免責事項	1
3.	概要	1
4.	各部の詳細	2
4.1	電源入力	5
4.2	ビデオ信号出力	6
4.3	アイリスレンズ制御（DCアイリス）	7
4.4	アイリスレンズ制御（ビデオアイリス）	8
4.5	GPIO（動体検出出力）	9
4.6	OSD（オンスクリーンディスプレイ）	10
5.	デジタルビデオ出力	11
5.1	電氣的仕様	12
5.2	デジタル出力フォーマット	14
6.	SPI 通信	22
6.1	電氣的仕様	23
6.2	[カメラ制御コマンド] プロトコル	28
6.3	カメラ制御コマンド	31
6.4	カメラパラメータ（アドレスマップ）	35
6.5	SPI 通信例	50
7.	仕様	61
8.	詳細寸法	62

1. はじめに

本書は、WAT-910BDのハードウェア機能、特徴について解説するマニュアルです。

内容にはWAT-910BDを適切に使用するために必要な、電氣的知識を要する記述が含まれています。

本書を良くお読みになり十分理解した上で、WAT-910BDをご使用・お使いのシステムへの導入を行ってください。

不適切な使用をした場合、WAT-910BDまたは接続される機器に深刻な損傷を発生させる可能性があります。

本書の内容に不明な点がございましたら、お買い求めの代理店・弊社営業・サポート部門にご連絡ください。

OSD（オンスクリーンディスプレイ）メニューを使用して、各機能を設定する方法、各機能の詳細については、併せて「WAT-910BD OSDマニュアル」をご一読ください。

また、WAT-910BDの導入方法、操作方法については、「WAT-910BD 取扱説明書」を併せてご参照ください。

2. 免責事項

本書の内容は、改良を目的とした仕様変更、デザイン変更に伴い予告無く変更される可能性があります。

本書に掲載した技術資料は、使用上の参考として示したものです。ご使用の際、ワテック株式会社および第三者の知的財産権・その他の権利の実施、使用を許諾したものではありません。

以上より、その使用・応用に起因する権利の侵害につきましては当社は一切の責任を負いません。

3. 概要

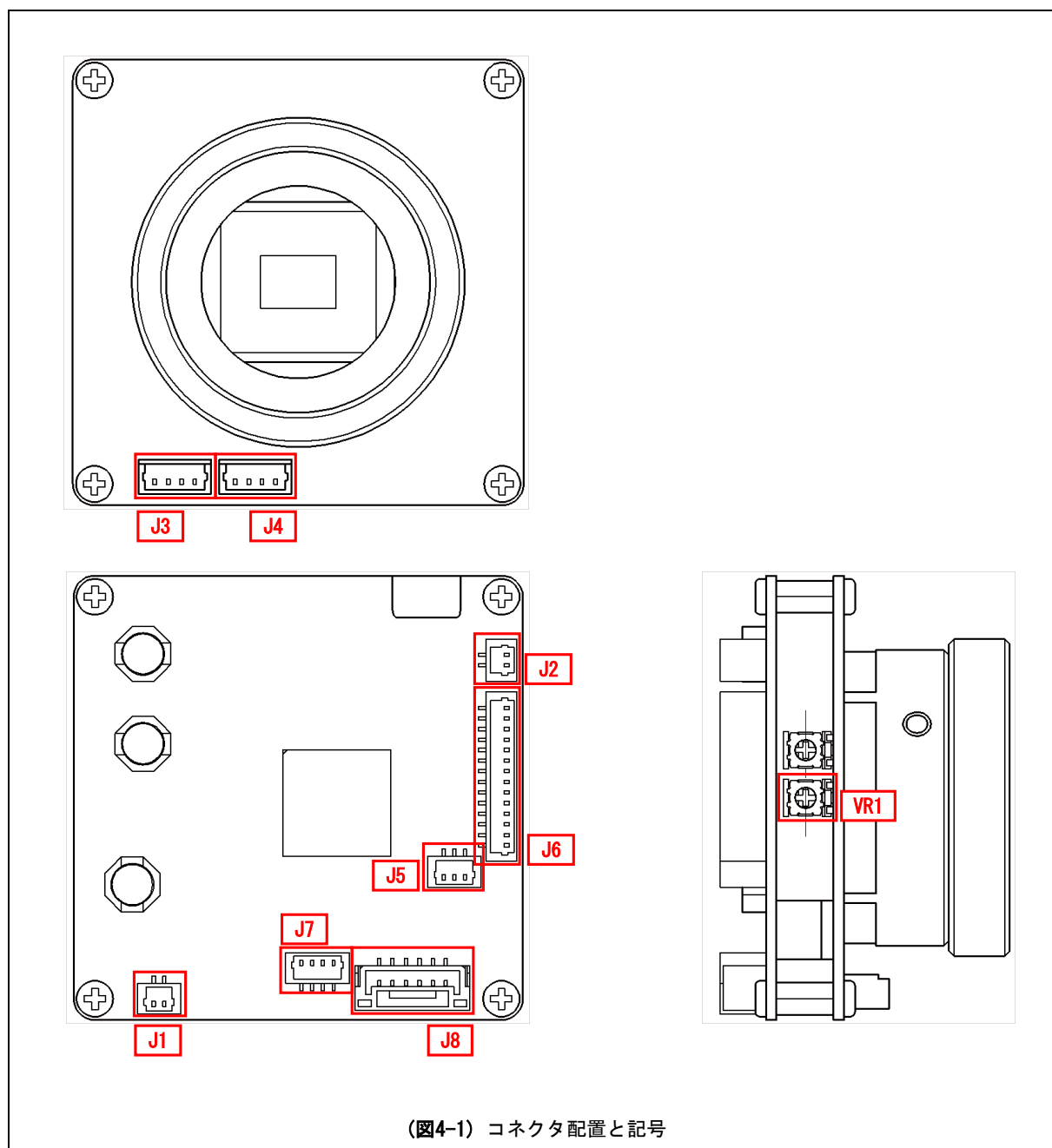
WAT-910BDは、1/2"CCDを搭載した超高感度白黒ボードカメラです。撮像部分に最新の高感度CCD、信号処理部に多彩な機能を有するDSPを採用したことにより、様々な機能・性能を実現しました。

- ・ 高速シャッター、電子アイリスによる自動露光制御機能に加え、暗所の撮像に便利な低速シャッター（フィールド蓄積）機能を装備しています。これらの併用により被写体の照度を選ばない、高品位な撮像が可能になりました。
- ・ 映像処理ブロックにDSP（デジタルシグナルプロセッサ）を採用することにより、ノイズリダクション、ダイナミックレンジ補正、白点補正、動体検出などの多彩なデジタル信号処理機能を使用することが出来ます。
- ・ 各機能の設定はユーザフレンドリなOSDメニュー操作と、機能を直接制御できる通信機能を装備しております。
- ・ 外部出力端子には従来のアナログ出力に加えてデジタル出力（パラレル出力）端子を備えております。
- ・ 42（mm）正形状、約40（g）の小型軽量設計ですので、様々な組み込み用途に最適です。

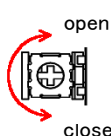
4. 各部の詳細

本章では、各部の名称と機能の詳細を説明します。

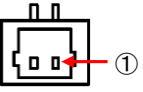
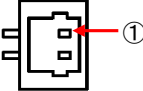
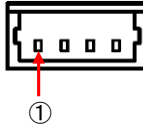
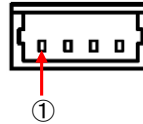
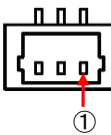
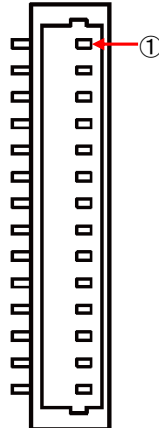
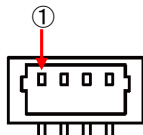
コネクタ類の配線の際は、ピン配列に注意し確実な配線をお心がけください。



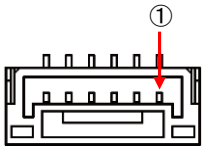
(図4-2) DCアイリスレンズ調整VR

	形状	名称	詳細
VR1		DCアイリスレンズ 調整VR	DCアイリスレンズ装着時、絞りの開き具合を調整するためのVRです。 このVRはJ3コネクタにのみ効果が現れます。

(図4-3) コネクタピン配列と機能

	形状	名称	部品名称	製造メーカ	ピン番号	機能	I/O
J1		電源入力端子	BM02B-SRSS	JST	①	電源 (DC+12V)	I
					②	電源 GND	GND
J2		ビデオ出力端子	BM02B-SRSS	JST	①	ビデオ出力	O
					②	ビデオ GND	GND
J3		DCアイリスレンズ 接続端子	53047-0410	molex	①	Sense-	O
					②	Sense+	-
					③	Drive+	O
					④	Drive-	GND
J4		ビデオアイリス レンズ接続端子	53047-0410	molex	①	Vcc (DC+12V) 出力	O
					②	未接続 (NC)	-
					③	アイリス信号出力	O
					④	GND	GND
J5		GPIOコネクタ	BM03B-SRSS	JST	①	動体検出出力	O
					②	GND	GND
					③	(予約)	-
J6		デジタル映像出力 端子	BM14B-SRSS	JST	①	垂直ランキング信号	O
					②	水平部ランキング信号	O
					③	GND	GND
					④	クロック出力	O
					⑤	GND	GND
					⑥	データBIT 7 (MSB)	O
					⑦	データBIT 6	O
					⑧	データBIT 5	O
					⑨	データBIT 4	O
					⑩	データBIT 3	O
					⑪	データBIT 2	O
					⑫	データBIT 1	O
					⑬	データBIT 0 (LSB)	O
					⑭	GND	GND
J7		SPI 通信端子	BM04B-SRSS	JST	①	SPI SLD	I
					②	SPI SCL	I
					③	SPI SDA	I/O
					④	GND	GND

(図4-3) コネクタピン配列と機能 (続き)

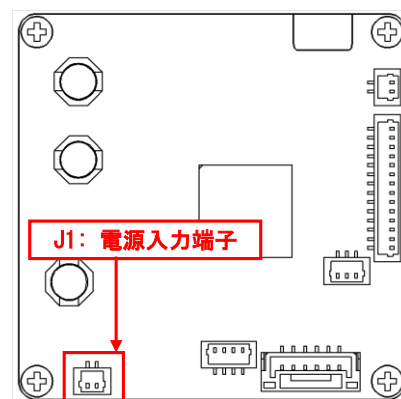
	形状	名称	部品名称	製造メーカー	ピン番号	機能	I/O
J8		OSD制御端子 (5キーリモコン接 続端子)	BM06B-GHS-TBT	JST	①	GND	GND
					②	SET	I
					③	LEFT	I
					④	RIGHT	I
					⑤	DOWN	I
					⑥	UP	I

4.1 電源入力

(1) 電源電圧範囲

WAT-910BDの推奨電源電圧、絶対最大定格は以下の通りです。

(図4-4) 電源電圧仕様	
仕様項目	範囲
(推奨) 電源入力範囲	12[V] ±10[%]
(最大) 電源入力範囲	DC+8.0 ~ 14.0[V]
最大定格 (限界値)	DC+16.0[V] 以下



通常は (推奨) 電源入力範囲でのご使用をお願いいたします。

WAT-910BDの諸性能はこの電源範囲にて測定、規定されております。

(最大) 電源入力範囲でのご使用では、WAT-910BDが破損することはありませんが、

- ・ 著しく明るい被写体を撮像した際、ブルーミング、スミアの発生が増加する可能性がある (低電圧時の使用)
- ・ 電源入力電圧の高いビデオアリスレンズ併用時、絞り開閉動作が遅い／出来ない (低電圧時の使用)

等の問題が発生する場合があります。

この範囲でご使用の際は、十分にカメラ評価／システム評価を行ってください。

最大定格を越えた電源電圧でのご使用については、WAT-910BDの破損、またカメラ破損による他の機器への悪影響が及ぶ可能性があります。絶対に行わないようご注意ください。

(2) 消費電力

WAT-910BDの消費電力は、DC+12V入力時、約1.32W (消費電流: 110mA) です。

この値は以下の条件で測定された代表値です。

- 測定条件:
- ①カメラ機能設定デフォルト (OSDメニューのRESET実行後の状態)
 - ②マニュアルアイリスレンズ使用、絞り全閉 (遮光状態)

WAT-910BDの消費電力 (および消費電力効率) は電源電圧に依存します。以下に、電源電圧値・消費電流・消費電力を上記条件で測定した結果を示します。(代表値)

(図4-5) 消費電力									
電源電圧 (V)	8.0	8.5	9.0	9.5	10.0	10.5	11.0	11.5	12.0
消費電流 (mA)	148.8	141.6	134.9	128.6	123.8	119.0	114.8	111.0	107.0
消費電力 (W)	1.19	1.20	1.21	1.22	1.24	1.25	1.26	1.28	1.28
電源電圧 (V)	12.5	13.0	13.5	14.0	※代表値: 保証値ではありません。				
消費電流 (mA)	104.2	101.1	98.4	95.7					
消費電力 (W)	1.30	1.31	1.33	1.34					

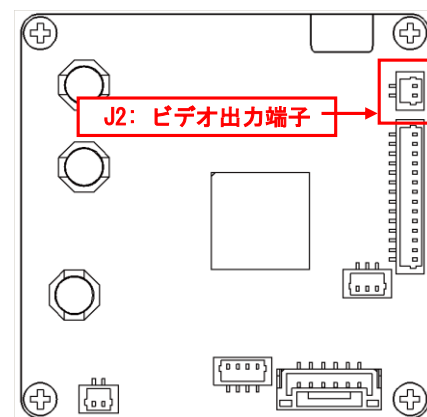
使用電源の選定・設計時には以下の点をご確認ください。

- ・ アイリスレンズ (ビデオアイリス・DCアイリス) 使用時には消費電力は増加します。
消費電流／電力の増加はご使用のレンズにより異なりますので、レンズの仕様をご確認ください。
- ・ 使用するカメラ機能のON／OFFで電流／電力値は上下します。
実際のカメラ使用状況を考慮して十分な性能の電源をご用意／設計してください。


4.2 ビデオ信号出力

75 オームインピーダンスのアナログビデオ出力端子です。

映像信号を受け取る機器側で、必ず75 オーム終端を行ってご使用ください。



(図4-6) ビデオ出力端子

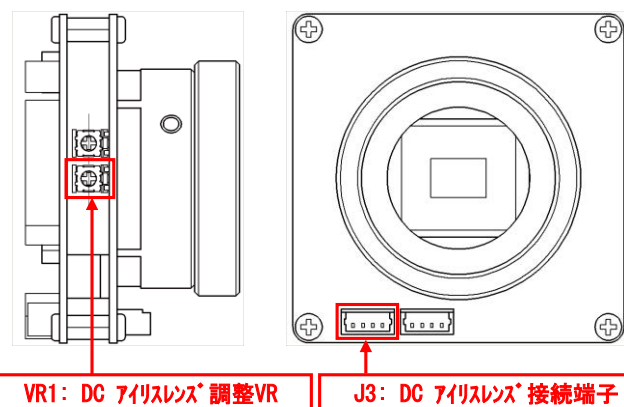
	部品形状とピン配列		ピン名称	詳細
J2		①	ビデオ信号出力	75 オームインピーダンス
		②	GND	ビデオ信号用GND

4.3 アイリスレンズ制御（DCアイリス）

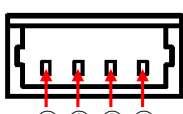
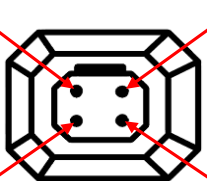
DCアイリスレンズを接続するためのコネクタです。
レンズを接続する際は、ピン配列をご確認の上
誤配線の無いようご注意ください。

(1) 配線

J 3 接続端子（コネクタ）のピン配列と
E I A J 規格アイリスコネクタのピン配列は
以下のように対応しています。



(図4-5) DCアイリスレンズ接続端子

カメラコネクタのピン配列		信号		E I A J コネクタのピン配列	
J3		①	Sense-	*1	
		②	Sense+	*2	
		③	Drive+	*3	
		④	Drive-	*4	

(2) アイリスレンズの調整

アイリスレンズの調整は次の手順で行います。

①シャッタースピードを固定にします。

SETUP → EXPOSURE メニューで、SHUTTER を変更します。

推奨：OFF（EIA:1/60, CCIR:1/50）またはFL（EIA:1/100, CCIR:1/120）

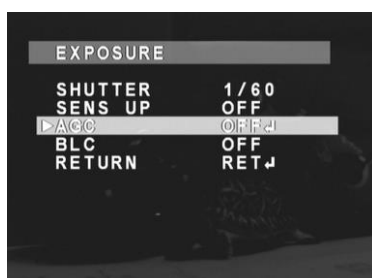
②SENSE-UP機能をOFFにします。

③AGC を OFF にします。

AGC OFF → MANUAL GAIN メニューで、GAIN を最小ゲインに設定します。

④EXPOSURE メニューに戻ります。VR1 を調整して、適切な絞りを設定してください。

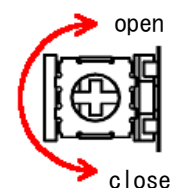
⑤AGC を OFF → LOW に切り換えたとき、少しだけ明るくなる絞りが適切なレンズ絞り設定です。



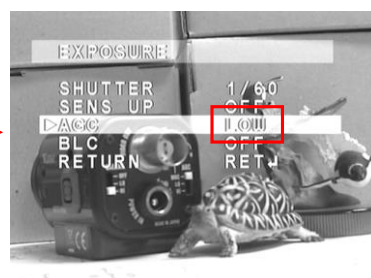
①, ②：レンズ絞り調整時の設定



③：AGCゲイン設定



④：レンズ絞りの調整



⑤AGC を OFF → LOW に切り換えたとき、若干明るくなる絞りが適正值です。

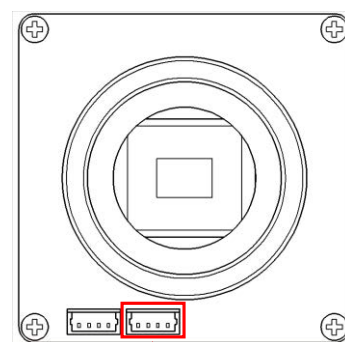
4.4 アイリスレンズ制御（ビデオアイリス）

ＤＣアイリスレンズを接続するためのコネクタです。

レンズを接続する際はピン配列をご確認の上、誤配線の無いようご注意ください。

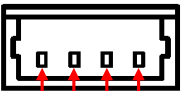
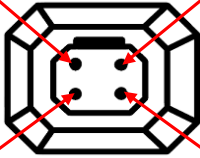
(1) 配線

J 3 接続端子（コネクタ）のピン配列と E I A J 推奨アイリスコネクタのピン配列は以下のように対応しています。



J4: VIDEO アイリスレンズ* 接続端子

(図4-6) ビデオアイリスレンズ接続端子

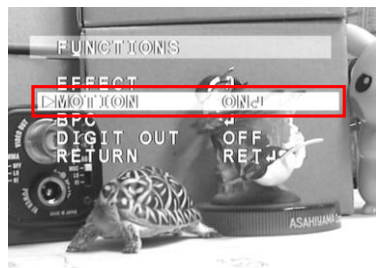
カメラコネクタのピン配列		信号		E I A J コネクタのピン配列	
J4		①	Vcc (DC+12V) 出力	*1	
		②	未接続 (N C)	*2	
		③	アイリス信号出力	*3	
		④	G N D	*4	

(2) アイリスレンズの調整

アイリスレンズの調整方法は、ＤＣアイリスレンズと同じですが、レンズの絞り具合はレンズの絞り調整ＶＲを使用します。調整ＶＲについては、レンズの取扱説明書をご確認ください。

4.5 G P I O（動体検出出力）

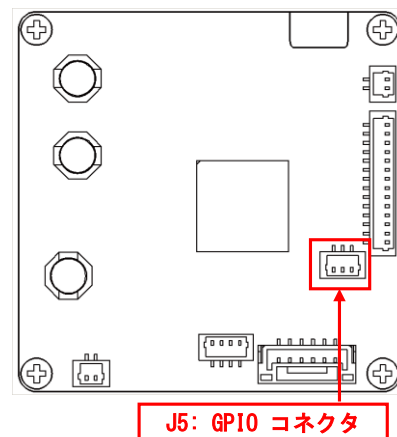
本機の動体検出機能をご使用の場合、カメラが動体検出を行ったときに G P I Oコネクタから検出結果を出力することが出来ます。
動体検出機能の設定方法、使用方法については O S Dマニュアルも合わせてご参照ください。



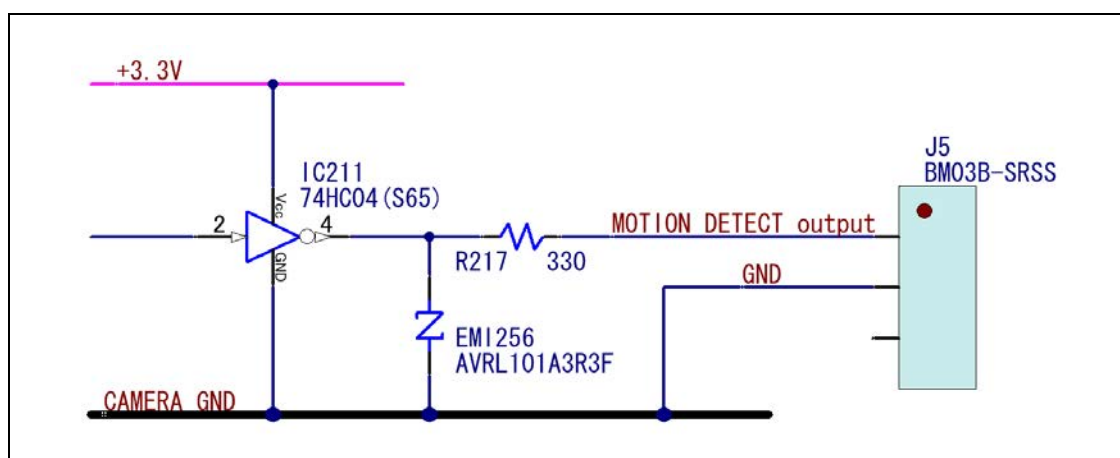
FUNCTIONS → MOTION ON



動体検出時の画面



(図4-7) G P I O（動体検出出力）端子				
	ピン配列		端子名	詳細
J5		①	動体検出出力	動体検出時、Hレベル
		②	GND	GND
		③	(予約)	機能拡張時の予約ピン



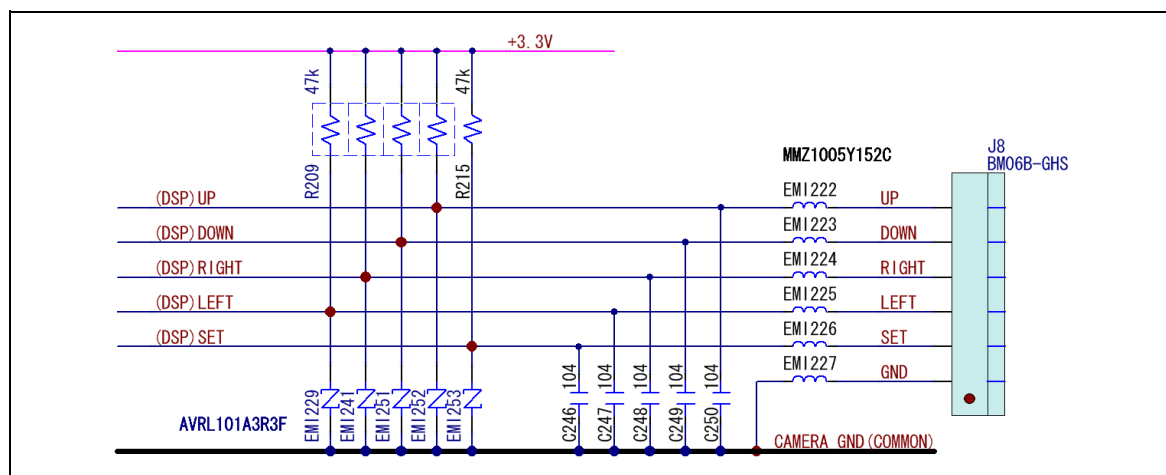
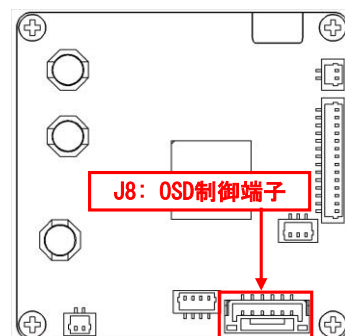
(図4-8) 電氣的仕様

仕様項目	記号	Min	Typ	Max	単位	測定条件
出力電圧（Hレベル）	Voh	2.4			V	Ioh = 1mA
出力電圧（Lレベル）	Vol			0.4	V	Iol = -1mA
出力時間	T		2		S	

動体検出出力ピンは、動体検出時、Hレベルとなります。

4.6 OSD（オンスクリーンディスプレイ）

OSDのコントロールは、5キーに割り当てられたJ8コネクタの端子をリモートコントローラ（RC-01）または電氣的な接点を使用し操作します。各端子は、カメラ内部でプルアップされていますので、GND端子に接触しLパルスを入力することにより各方向と決定の指示を行うことができます。OSDメニュー構成と操作については、別紙OSDマニュアルをご参照ください。



(図4-9) OSD制御端子

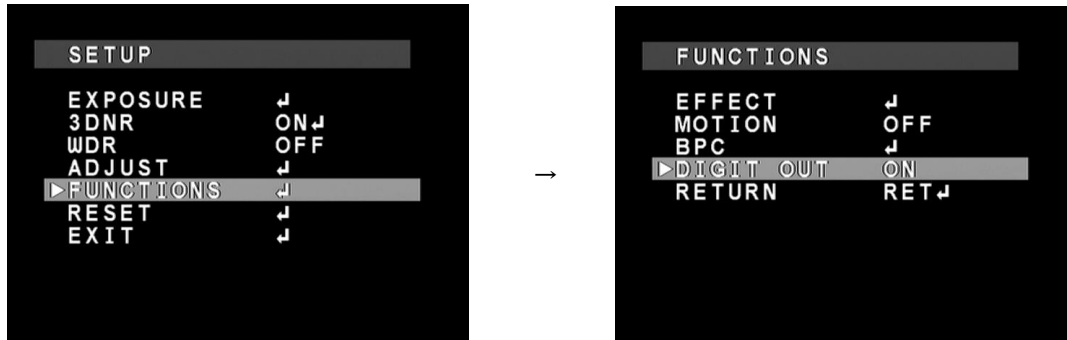
	Parts Shape (Pin No.)	Pin Name	Description
J8		① GND (COMMON)	<p>Lアクティブ</p> <p>ラッチ 長押し時、リピータ入力</p> <p>キー入力は、Lアクティブです。</p> <p>長押し時、リピータ入力となります。</p>
		② SET	
		③ LEFT	
		④ RIGHT	
		⑤ DOWN	
		⑥ UP	

5. デジタルビデオ出力

WAT-910BDには、デジタル映像出力端子（J 6）を使用しデジタル映像信号を出力する機能があります。
デジタル信号出力のON/OFFは、OSDメニューまたはSPI通信により切り替え可能です。

(1) デジタル映像出力ON/OFF（OSD）

[SETUP]→[FUNCTIONS]→[DIGITAL OUT] を選択し、デジタル出力のON/OFFを切り替えます。



操作の詳細については、OSDマニュアルも併せてご覧ください。

(2) デジタル映像出力ON/OFF（SPI通信）

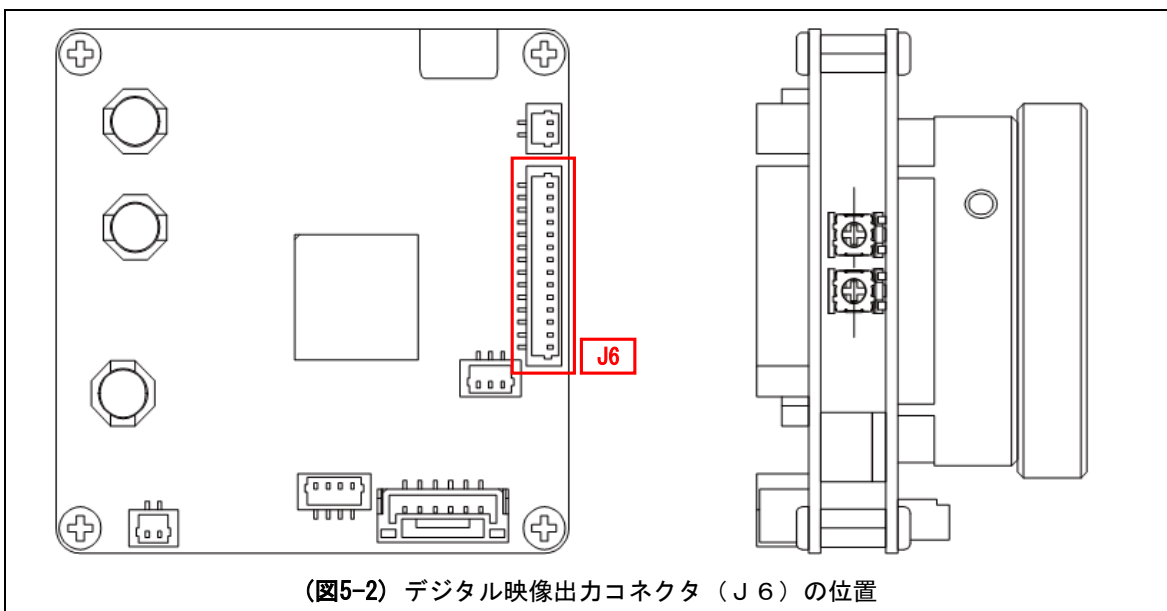
SPI通信によりデジタル映像出力のON/OFFを切り替えるには、下記のアドレスに割り当てられたパラメータを操作します。

(図5-1) SPI通信パラメータアドレス（デジタル出力）		
パラメータ名	アドレス	詳細
DIGIT OUT	0x5E2, bit 0	(Digital Output On/Off) 0:OFF, 1:ON

SPI通信によるパラメータ操作方法については、本書6章SPI通信に詳細を説明します。

(3) デジタル映像出力端子

デジタル映像出力端子はカメラ背面の図の位置にあります。



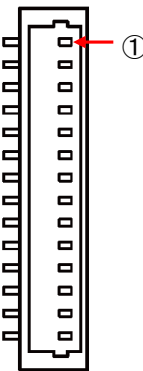
(図5-2) デジタル映像出力コネクタ（J 6）の位置

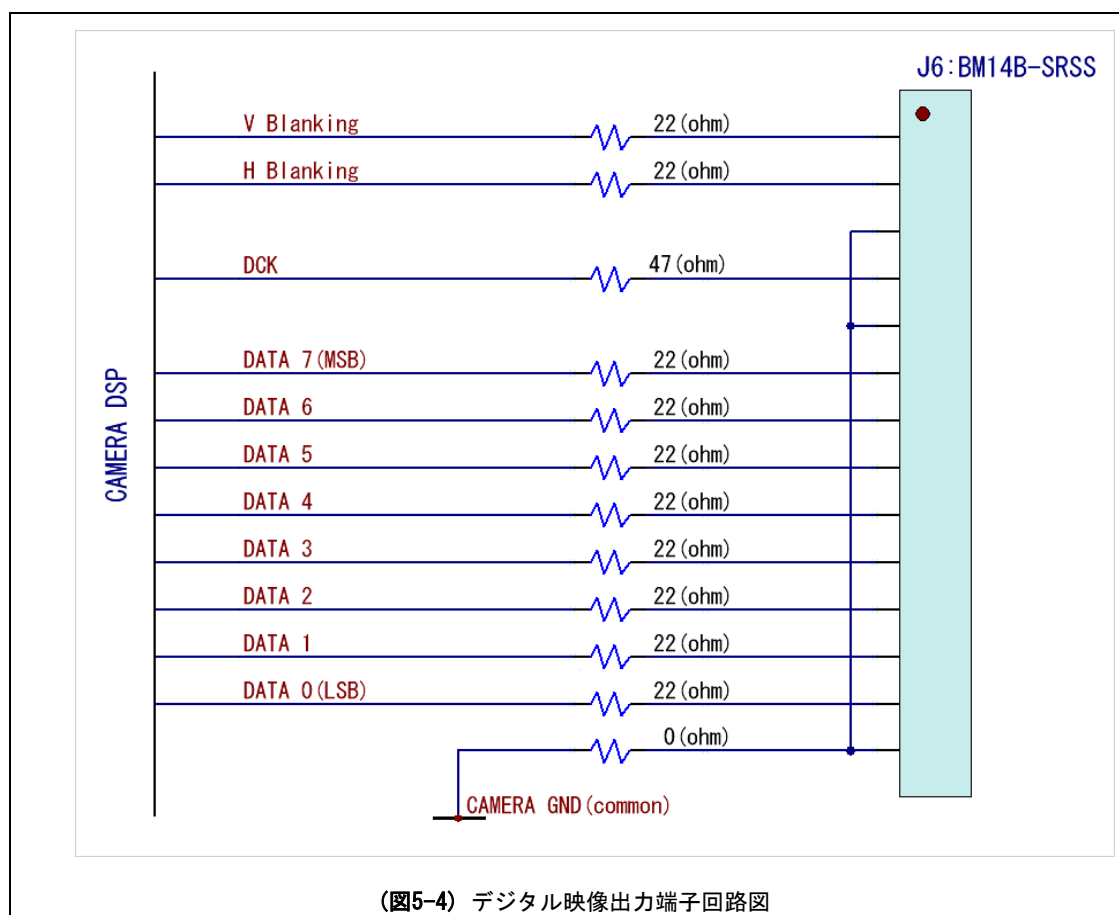
5.1 電氣的仕様

デジタル映像出力端子の各ピンは、保護抵抗を介しカメラDSPに直結されています。

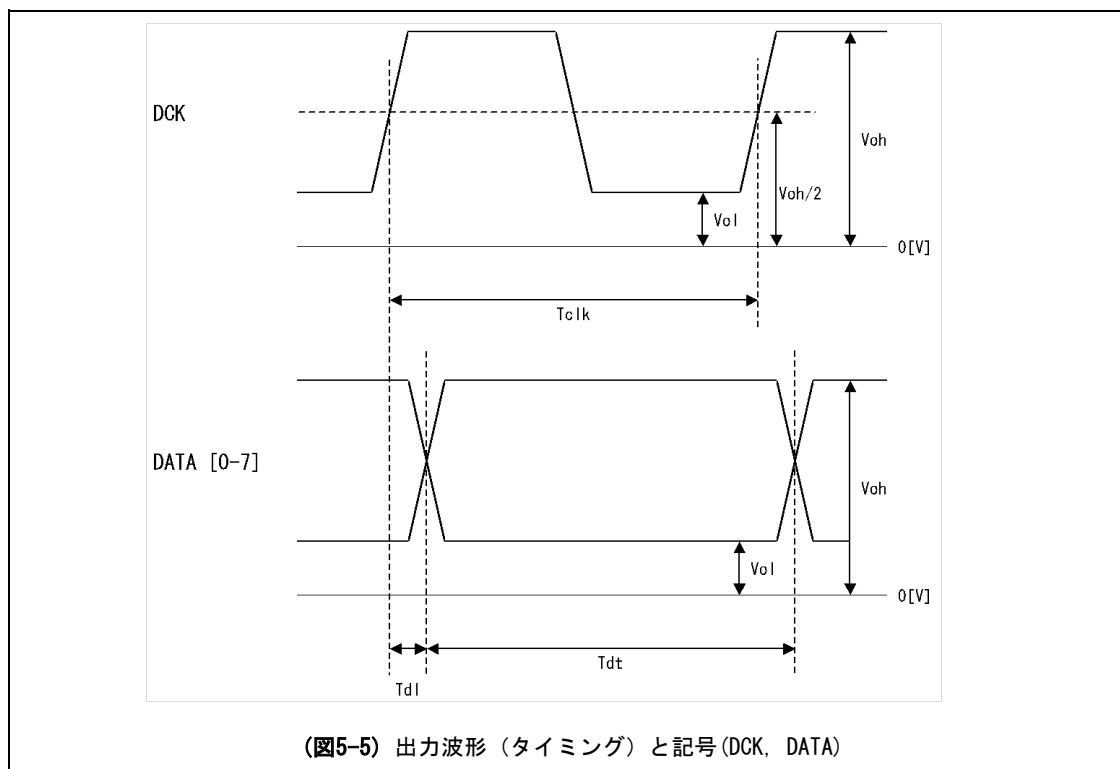
(3. 3V CMOSレベル出力。バッファ／ドライバ等は用意されておりません。)

端子の詳細と、出力部分の回路図を以下に示します。

(図5-3) デジタル映像出力端子詳細						
	部品形状	部品名	製造メーカ	ピン番号	詳細	I/O
J6		BM14B-SRSS	JST	①	垂直ブランキング信号	0
				②	水平部ランキング信号	0
				③	GND	GND
				④	クロック出力	0
				⑤	GND	GND
				⑥	データBIT 7 (MSB)	0
				⑦	データBIT 6	0
				⑧	データBIT 5	0
				⑨	データBIT 4	0
				⑩	データBIT 3	0
				⑪	データBIT 2	0
				⑫	データBIT 1	0
				⑬	データBIT 0 (LSB)	0
				⑭	GND	GND



各端子の電氣的仕様は以下の通りです。



(図5-6) 電氣的仕様 (DCK, DATA)

仕様項目	記号	Min	Typ	Max	単位	測定条件
出力電圧 (H レベル)	Voh	2.4			V	Ioh = 6mA
出力電圧 (L レベル)	Vol			0.4	V	Iol = -6mA
DCK 出力周期 (EIA)	Tclk		34.9		nS	1/(28.636363MHz)
DCK 出力周期 (CCIR)	Tclk		35.2		nS	1/(28.375MHz)
Data 出力時間 (EIA)	Tdt		34.9		nS	
Data 出力時間 (CCIR)	Tdt		35.2		nS	
Data デイレイ	Tdl			1.5	nS	

5.1 デジタル出力フォーマット

(1) ビデオデータフォーマット

WAT-910BDは、EIA/CCIR規格に準拠したビデオカメラです。

アナログビデオ端子（J 2）より出力されるビデオ信号については、EIA/CCIR規格をご参照ください。

デジタルビデオ出力（J 6）より出力されるデジタルビデオデータはBT. 656規格に準拠しますが、ピクセルクロックレートが異なる関係上、27MHzクロックのBT. 656規格と異なる部分があります。

デジタルデータ出力のビデオフォーマットについては下表をご参照ください。

(図5-7) デジタルビデオフォーマット					
		EIA		CCIR	
		ODDフィールド	EVENフィールド	ODDフィールド	EVENフィールド
フレームレート [Frame/sec]		30		25	
全ライン数 [Line/Frame]		525		625	
ブランキング [Line/Field]		22	21	27	26
有効ライン数	[Line/Field]	241	241	286	286
	[Line/Frame]	482		572	
有効画素数 [Pixel/line]		756		739	

(2) データフォーマット

BT. 656規格に対しWAT-910BDのデータクロックレートは異なります。

(図5-8) デジタル出力クロック		
	EIA	CCIR
DCK [MHz]	28. 636363	28. 375000

※注意：BT. 656 規格では DCK=27[MHz] となります。WAT-910BDはこのクロックレートには対応していません。

デジタル出力コネクタ（J 6）からは、クロマ信号、輝度信号、ブランキング信号、TRCコードに相当するデジタルデータがシーケンシャルに出力されます。

クロマ信号、輝度信号、TRCコードのデータフォーマットは BT. 656規格に準拠したフォーマットで出力されますが、DCKの違いにより各々のデータバイト数は BT. 656 (DCK=27MHz) 規格とは異なります。

各データの種別とデータバイト数の違いは下表の通りになります。

(図5-9) 1ライン分のデジタルデータ構成			参考値 (BT. 656 EIA/CCIR)
	EIA	CCIR	
EAV コード [Byte]	4		4/4
水平ブランキング [Byte]	300	330	268/280
SAV コード [Byte]	4		4/4
有効画像データ [Byte]	1512	1478	1440/1440
総データ数 [Byte]	1820	1808	1716/1728
有効データビット数 [bit]	8		
有効ビデオデータ範囲 [Decimal]	1 - 254		

(3) デジタルデータ出力タイミング

有効画像データ部分は、Y/Cb/Cr=4:2:2 フォーマットで出力されます。(Cb, Y, Cr, Y, … の繰り返しとなります。)
 しかしながら、WAT-910BDは白黒カメラですので、クロマ信号 (Cb/Crデータ) はすべて80[h]となります。
 ブランキングデータ部分 (TRCコードを除く) は、Cb, Cr=80[h], Y=10[h]の繰り返しとなります。

TRCコードは、すべてのラインデータに付加されます。TRCコードはBT. 656に準拠した4バイトコードです。

SAV : Start of Active Video [hex]

FF	00	00	XX
----	----	----	----

EAV : End of Active Video [hex]

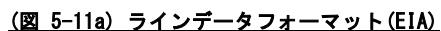
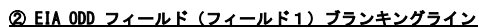
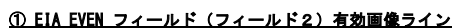
FF	00	00	XX
----	----	----	----

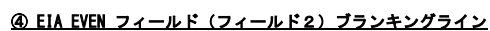
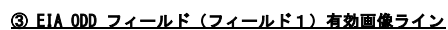
TRCコードの4バイト目 (上図、XX部分) は、その後に続くデータの種別に従い、下表のように変化します。

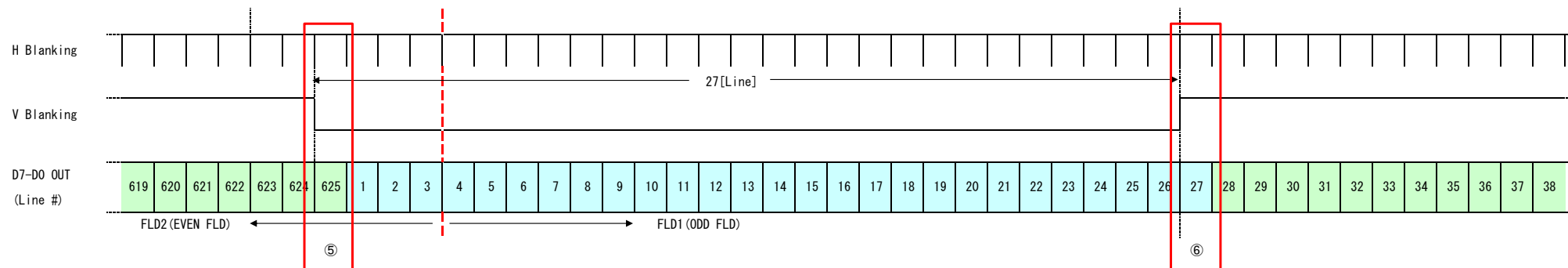
(図5-10) TRCコード									
	(Fix)	F	V	H	P3	P2	P1	P0	CODE
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	[hex]
ODD Field H Blanking (SAV)	1	0	0	0	0	0	0	0	80
ODD Field H Blanking (EAV)	1	0	0	1	1	1	0	1	9D
ODD Field V Blanking (SAV)	1	0	1	0	1	0	1	1	AB
ODD Field V Blanking (EAV)	1	0	1	1	0	1	1	0	B6
EVEN Field H Blanking (SAV)	1	1	0	0	0	1	1	1	C7
EVEN Field H Blanking (EAV)	1	1	0	1	1	0	1	0	DA
EVEN Field V Blanking (SAV)	1	1	1	0	1	1	0	0	EC
EVEN Field V Blanking (EAV)	1	1	1	1	0	0	0	1	F1
*NOTE: F=0 : during field 1 F=1 : during field 2 V=0 : elsewhere V=1 : during field blanking H=0 : in SAV H=1 : in EAV P0-P3 : プロテクションビット									

図5-11 (EIA) / 図5-12 (CCIR) に、EIA/CCIR各バージョンの1ライン分のデジタルデータ構成を示します。

図5-13 (EIA) / 図5-14 (CCIR) に、EIA/CCIR各バージョンの1フレーム分のデジタルデータ構成を示します。







The diagram illustrates the timing relationship between the D7-D0 OUT data stream and the DCK, H Blanking, and V Blanking signals. The DCK signal is a periodic clock. H Blanking and V Blanking signals indicate horizontal and vertical blanking periods. The D7-D0 OUT signal shows the data stream, including EAV code (FF 00 00 B6), H BLK DATA (80 10 80 10 80 10 80 10), SAV code (FF 00 00 C7), and Active Video Data (80 Y1 80 Y2 80 Y3 80 Y735 80 Y736 80 Y737 80 Y738 80 Y739 FF 00).

Key timing intervals and data segments are labeled:

- 7 [DCK]**: Initial clock period.
- 4 [DCK]**: Clock period after initial setup.
- 330 [DCK]**: Duration of H BLK DATA.
- 4 [DCK]**: Clock period after H BLK DATA.
- 1808 [DCK/Line]**: Total duration of the active video data segment.
- 1478 [DCK]**: Duration of Active Video Data.
- 6 [DCK]**: Duration of V Blanking.
- 7 [DCK]**: Duration of H Blanking.

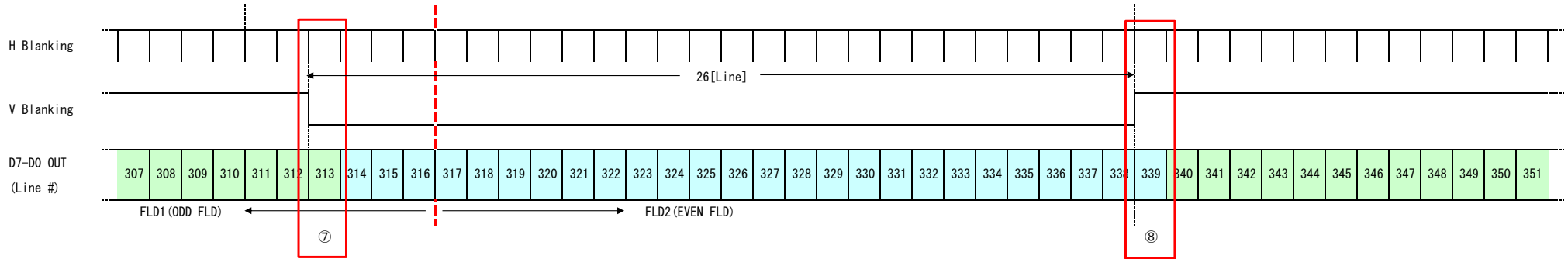
The data stream (D7-D0 OUT) is segmented as follows:

- EAV code**: FF 00 00 B6
- H BLK DATA (Cb, Cr: 80[h], Y: 10[h]) 330 [byte]**: 80 10 80 10 80 10 80 10
- SAV code**: FF 00 00 C7
- Active Video Data (Cb, Cr: 80[h], Y1-Y739: luminance DATA (01-FE) [h]) 1478 [byte]**: 80 Y1 80 Y2 80 Y3 80 Y735 80 Y736 80 Y737 80 Y738 80 Y739 FF 00

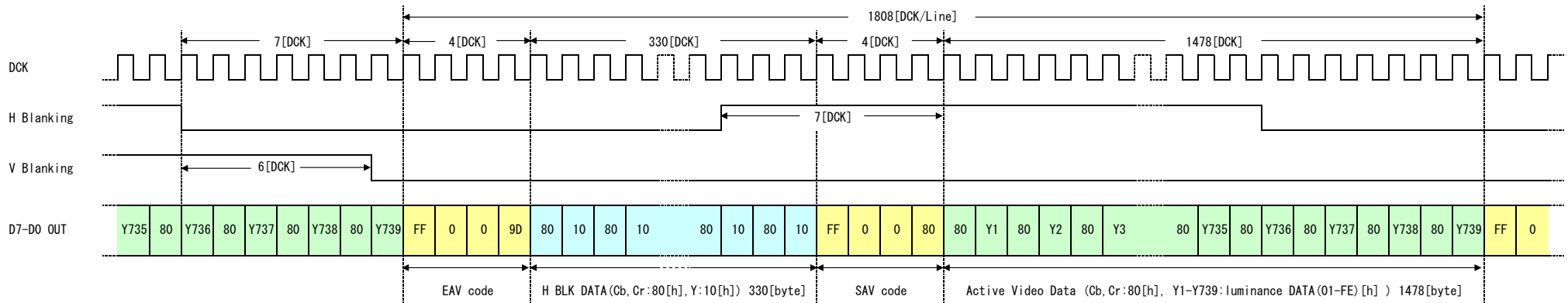
The diagram illustrates the timing of the D7-D0 OUT signal relative to the DCK, H Blanking, and V Blanking signals. The D7-D0 OUT signal is divided into several sections: a sequence of 10 bytes (10, 80, 10, 80, 10, 80, 10, 80, 10, 80), followed by the EAV code (FF, 0, 0, B6), then the H BLK DATA (Cb, Cr:80[h], Y:10[h]) (330 bytes), followed by the SAV code (FF, 0, 0, AB), and finally the V Blanking DATA (Cb, Cr:80[h], Y:10[h]) (1478 bytes). The H Blanking signal is active during the H BLK DATA and SAV code sections. The V Blanking signal is active during the V Blanking DATA section. The DCK signal is a periodic clock signal.

- 18 -

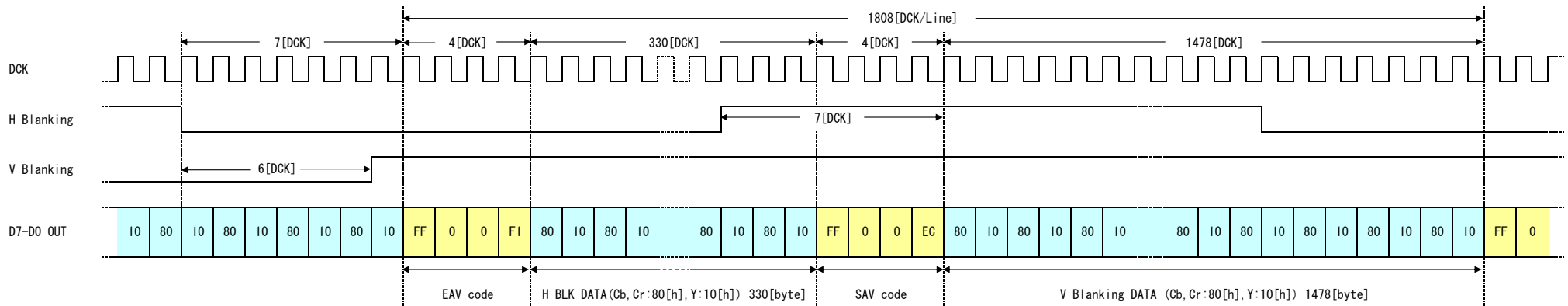
④ CCIR EVEN フィールド (フィールド2) Vブランキングタイミング



⑦ CCIR ODD フィールド (フィールド1) 有効画像ライン



⑧ CCIR EVEN フィールド (フィールド2) ブランキングライン



(図 5-12b) ラインデータフォーマット (CCIR)

(図 5-13) デジタルデータフォーマット (EIA)

		Line#	EAV (4byte)				H Blanking Data (300byte)								SAV (4byte)				V Blanking / Active Video Data (1512byte)																	
FLD 1	V BLK	4	FF	0	0	B6	80	10	80	10	→	80	10	80	10	FF	0	0	AB	80	10	80	10	80	10	80	10	80	10	80	10	80	10	80	10	
		5	FF	0	0	B6	80	10	80	10	→	80	10	80	10	FF	0	0	AB	80	10	80	10	80	10	80	10	80	10	80	10	80	10	80	10	
		↓	↓				↓ (H Blanking Data)								↓				↓ (V Blanking Data)																	
		21	FF	0	0	B6	80	10	80	10	→	80	10	80	10	FF	0	0	AB	80	10	80	10	80	10	80	10	80	10	80	10	80	10	80	10	
		22	FF	0	0	B6	80	10	80	10	→	80	10	80	10	FF	0	0	AB	80	10	80	10	80	10	80	10	80	10	80	10	80	10	80	10	
	VIDEO	23	FF	0	0	9D	80	10	80	10	→	80	10	80	10	FF	0	0	80	80	Y1	80	Y2	80	Y3	80	Y4	80	Y753	80	Y754	80	Y755	80	Y756	
		24	FF	0	0	9D	80	10	80	10	→	80	10	80	10	FF	0	0	80	80	Y1	80	Y2	80	Y3	80	Y4	80	Y753	80	Y754	80	Y755	80	Y756	
		↓	↓				↓ (H Blanking Data)								↓				↓ (Active Video Data)																	
		262	FF	0	0	9D	80	10	80	10	→	80	10	80	10	FF	0	0	80	80	Y1	80	Y2	80	Y3	80	Y4	80	Y753	80	Y754	80	Y755	80	Y756	
		263	FF	0	0	9D	80	10	80	10	→	80	10	80	10	FF	0	0	80	80	Y1	80	Y2	80	Y3	80	Y4	80	Y753	80	Y754	80	Y755	80	Y756	
FLD2	V BLK	264	FF	0	0	B6	80	10	80	10	→	80	10	80	10	FF	0	0	AB	80	10	80	10	80	10	80	10	80	10	80	10	80	10	80	10	
		265	FF	0	0	B6	80	10	80	10	→	80	10	80	10	FF	0	0	AB	80	10	80	10	80	10	80	10	80	10	80	10	80	10	80	10	
		266	FF	0	0	B6	80	10	80	10	→	80	10	80	10	FF	0	0	AB	80	10	80	10	80	10	80	10	80	10	80	10	80	10	80	10	
		267	FF	0	0	F1	80	10	80	10	→	80	10	80	10	FF	0	0	EC	80	10	80	10	80	10	80	10	80	10	80	10	80	10	80	10	
		268	FF	0	0	F1	80	10	80	10	→	80	10	80	10	FF	0	0	EC	80	10	80	10	80	10	80	10	80	10	80	10	80	10	80	10	
	↓	↓				↓ (H Blanking Data)								↓				↓ (V Blanking Data)																		
	283	FF	0	0	F1	80	10	80	10	→	80	10	80	10	FF	0	0	EC	80	10	80	10	80	10	80	10	80	10	80	10	80	10	80	10		
	284	FF	0	0	F1	80	10	80	10	→	80	10	80	10	FF	0	0	EC	80	10	80	10	80	10	80	10	80	10	80	10	80	10	80	10		
	VIDEO	285	FF	0	0	DA	80	10	80	10	→	80	10	80	10	FF	0	0	C7	80	Y1	80	Y2	80	Y3	80	Y4	80	Y753	80	Y754	80	Y755	80	Y756	
		286	FF	0	0	DA	80	10	80	10	→	80	10	80	10	FF	0	0	C7	80	Y1	80	Y2	80	Y3	80	Y4	80	Y753	80	Y754	80	Y755	80	Y756	
↓		↓				↓ (H Blanking Data)								↓				↓ (Active Video Data)																		
524		FF	0	0	DA	80	10	80	10	→	80	10	80	10	FF	0	0	C7	80	Y1	80	Y2	80	Y3	80	Y4	80	Y753	80	Y754	80	Y755	80	Y756		
525		FF	0	0	DA	80	10	80	10	→	80	10	80	10	FF	0	0	C7	80	Y1	80	Y2	80	Y3	80	Y4	80	Y753	80	Y754	80	Y755	80	Y756		
V BLK	1	FF	0	0	F1	80	10	80	10	→	80	10	80	10	FF	0	0	EC	80	10	80	10	80	10	80	10	80	10	80	10	80	10	80	10		
	2	FF	0	0	F1	80	10	80	10	→	80	10	80	10	FF	0	0	EC	80	10	80	10	80	10	80	10	80	10	80	10	80	10	80	10		
	3	FF	0	0	F1	80	10	80	10	→	80	10	80	10	FF	0	0	EC	80	10	80	10	80	10	80	10	80	10	80	10	80	10	80	10		
DCK#			1513	1514	1515	1516	1517	1518	1519	1520	→	1813	1814	1815	1816	1817	1818	1819	1820	1	2	3	4	5	6	7	8	→	1505	1506	1507	1508	1509	1510	1511	1512
Pixel#			757		758		759		760		→	907		908		909		910		Y1		Y2		Y3		Y4		→	Y753		Y754		Y755		Y756	

(図 5-13) デジタルデータフォーマット (EIA)

(図 5-14) デジタルデータフォーマット (CCIR)

		Line#	EAV (4byte)				H Blanking Data (330byte)								SAV (4byte)				V Blanking / Active Video Data (1478byte)																			
FLD 1	V BLK	4	FF	0	0	B6	80	10	80	10	→	80	10	80	10	FF	0	0	AB	80	10	80	10	80	10	80	10	80	10	80	10	80	10					
		5	FF	0	0	B6	80	10	80	10	→	80	10	80	10	FF	0	0	AB	80	10	80	10	80	10	80	10	80	10	80	10	80	10					
		↓	↓				↓ (H Blanking Data)								↓				↓ (V Blanking Data)																			
		26	FF	0	0	B6	80	10	80	10	→	80	10	80	10	FF	0	0	AB	80	10	80	10	80	10	80	10	80	10	80	10	80	10					
		27	FF	0	0	B6	80	10	80	10	→	80	10	80	10	FF	0	0	AB	80	10	80	10	80	10	80	10	80	10	80	10	80	10					
	VIDEO	28	FF	0	0	9D	80	10	80	10	→	80	10	80	10	FF	0	0	80	80	Y1	80	Y2	80	Y3	80	Y4	80	Y736	80	Y737	80	Y738	80	Y739			
		29	FF	0	0	9D	80	10	80	10	→	80	10	80	10	FF	0	0	80	80	Y1	80	Y2	80	Y3	80	Y4	80	Y736	80	Y737	80	Y738	80	Y739			
		↓	↓				↓ (H Blanking Data)								↓				↓ (Active Video Data)																			
		312	FF	0	0	9D	80	10	80	10	→	80	10	80	10	FF	0	0	80	80	Y1	80	Y2	80	Y3	80	Y4	80	Y736	80	Y737	80	Y738	80	Y739			
		313	FF	0	0	9D	80	10	80	10	→	80	10	80	10	FF	0	0	80	80	Y1	80	Y2	80	Y3	80	Y4	80	Y736	80	Y737	80	Y738	80	Y739			
FLD2	V BLK	314	FF	0	0	B6	80	10	80	10	→	80	10	80	10	FF	0	0	AB	80	10	80	10	80	10	80	10	80	10	80	10	80	10					
		315	FF	0	0	B6	80	10	80	10	→	80	10	80	10	FF	0	0	AB	80	10	80	10	80	10	80	10	80	10	80	10	80	10					
		316	FF	0	0	B6	80	10	80	10	→	80	10	80	10	FF	0	0	AB	80	10	80	10	80	10	80	10	80	10	80	10	80	10					
		317	FF	0	0	F1	80	10	80	10	→	80	10	80	10	FF	0	0	EC	80	10	80	10	80	10	80	10	80	10	80	10	80	10					
		318	FF	0	0	F1	80	10	80	10	→	80	10	80	10	FF	0	0	EC	80	10	80	10	80	10	80	10	80	10	80	10	80	10					
	↓	↓				↓ (H Blanking Data)								↓				↓ (V Blanking Data)																				
	338	FF	0	0	F1	80	10	80	10	→	80	10	80	10	FF	0	0	EC	80	10	80	10	80	10	80	10	80	10	80	10	80	10						
	339	FF	0	0	F1	80	10	80	10	→	80	10	80	10	FF	0	0	EC	80	10	80	10	80	10	80	10	80	10	80	10	80	10						
	VIDEO	340	FF	0	0	DA	80	10	80	10	→	80	10	80	10	FF	0	0	C7	80	Y1	80	Y2	80	Y3	80	Y4	80	Y736	80	Y737	80	Y738	80	Y739			
		341	FF	0	0	DA	80	10	80	10	→	80	10	80	10	FF	0	0	C7	80	Y1	80	Y2	80	Y3	80	Y4	80	Y736	80	Y737	80	Y738	80	Y739			
↓		↓				↓ (H Blanking Data)								↓				↓ (Active Video Data)																				
624		FF	0	0	DA	80	10	80	10	→	80	10	80	10	FF	0	0	C7	80	Y1	80	Y2	80	Y3	80	Y4	80	Y736	80	Y737	80	Y738	80	Y739				
625		FF	0	0	DA	80	10	80	10	→	80	10	80	10	FF	0	0	C7	80	Y1	80	Y2	80	Y3	80	Y4	80	Y736	80	Y737	80	Y738	80	Y739				
V BLK	1	FF	0	0	F1	80	10	80	10	→	80	10	80	10	FF	0	0	EC	80	10	80	10	80	10	80	10	80	10	80	10	80	10	80	10				
	2	FF	0	0	F1	80	10	80	10	→	80	10	80	10	FF	0	0	EC	80	10	80	10	80	10	80	10	80	10	80	10	80	10	80	10				
	3	FF	0	0	F1	80	10	80	10	→	80	10	80	10	FF	0	0	EC	80	10	80	10	80	10	80	10	80	10	80	10	80	10	80	10				
DCK#			1479	1480	1481	1482	1483	1484	1485	1486	→	1809	1810	1811	1812	1813	1814	1815	1816	1	2	3	4	5	6	7	8	→	1471	1472	1473	1474	1475	1476	1477	1478		
Pixel#			740		741		742		743		→		905		906		907		908		Y1		Y2		Y3		Y4		→		Y736		Y737		Y738		Y739	

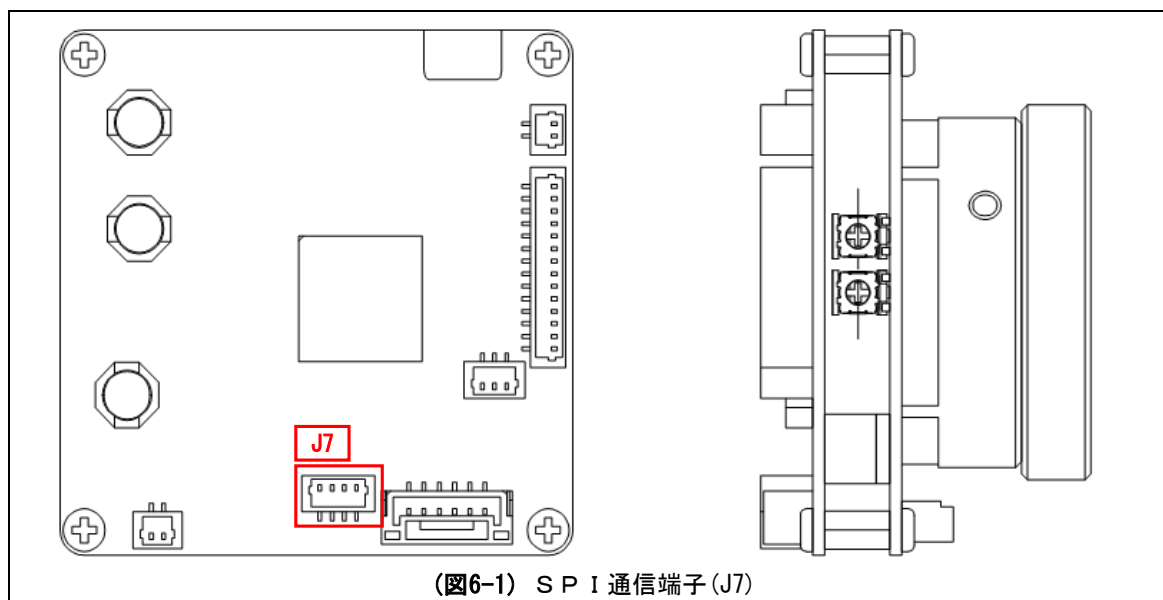
(図 5-14) デジタルデータフォーマット (CCIR)

6. S P I 通信

S P I 通信端子（J 7）を使用することにより、WAT-910BDの様々な機能をO S Dメニューを使用することなく直接制御する事が出来ます。

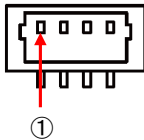
(1) S P I 通信端子

S P I 通信端子（J 7）は下図の位置にあります。



6.1 電氣的仕様

S P I 通信端子の配列と、入出力部分の回路図は下記のようになっています。

(図 6-2) ピン配列と機能詳細							
	部品形状	名称	部品名	製造メーカー	ピン番号	詳細	I/O
J7		S P I 通信端子	BM04B-SRSS	JST	①	SPI SLD (Lアクティブ)	I
					②	SPI SCL	I
					③	SPI SDA	I/O
					④	GND	GND

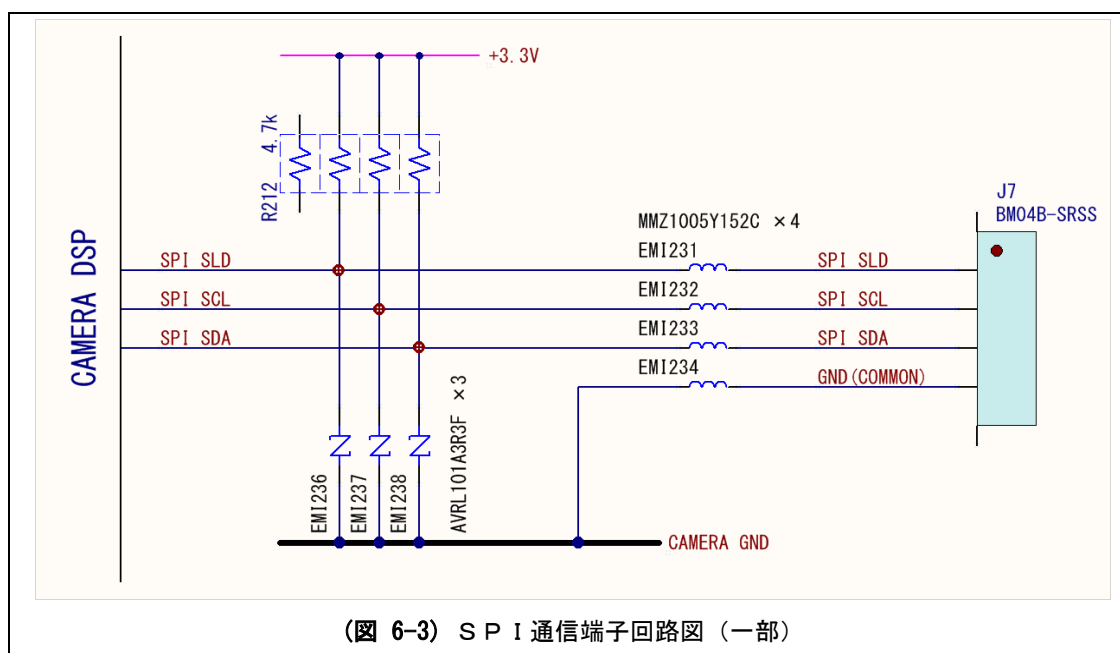
J 7 端子の全てのピンはビーズを通してカメラ DSP に直結されています。(EM1231-234 : 高周波 E M I 対策部品)

入出力レベルは、3.3[V] CMOS レベルです。ドライバー／バッファ相当部品は搭載されていません。

GND ピンを除く全てのピンには、静電破壊対策として GND 間にチップバリスタが搭載されています。

(EM1236-238 : 静電破壊対策部品)

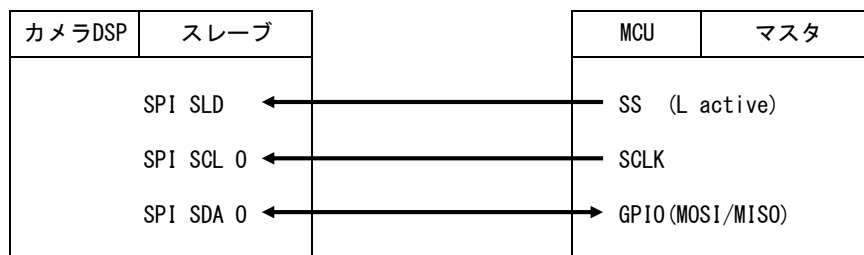
また GND を除く全てのピンは、カメラ内部の 3.3[V] 電源に 4.7k[Ω] を介してプルアップされています。



(1) 通信の概要

WAT-910BD の S P I 通信仕様は基本的に 3 線式 S P I 通信仕様に準拠しています。

外部マイクロコントローラ（MCU：マスタ）とカメラDSP（スレーブ）の結線は下図のようになります。



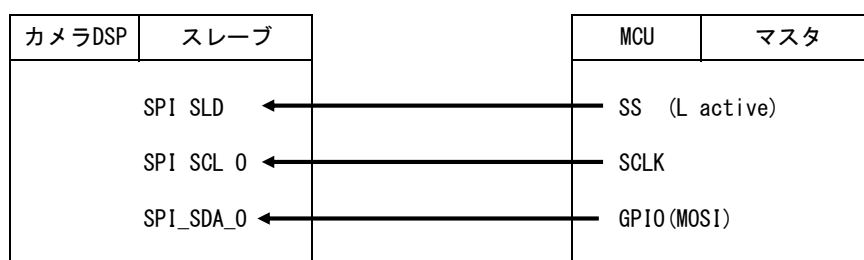
3 線式の S P I 通信を行うためには、外部MCUのG P I Oピンは双方向に対応したものがが必要です。

また、このG P I OピンはS P I 通信におけるM I S O / M O S I の機能を入出力を切り換えることにより実装する必要があります。

(2) コマンドとタイミングチャート（S P I 書込コマンド）

書込コマンドは全て、外部MCUからカメラDSPに対して行われます。（マスタからスレーブの方向）

このとき外部MCUのG P I OピンはM O S I として動作し、コマンドをカメラDSPのS D A に対して出力する必要があります。（コマンド書込の間、G P I Oピンは出力状態です。）

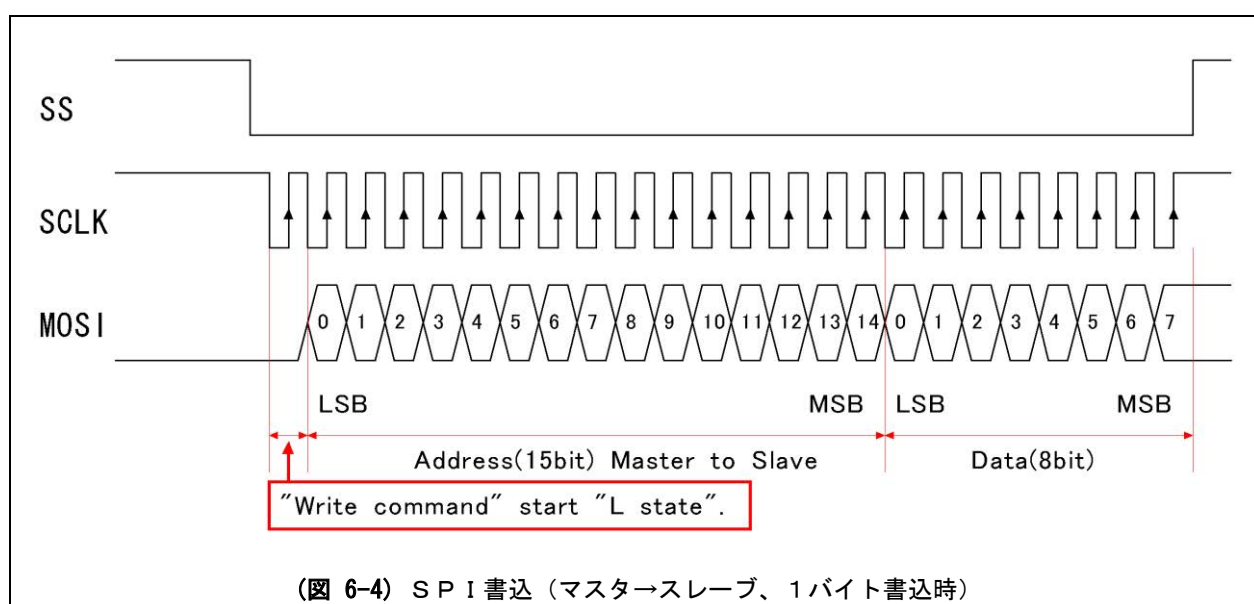


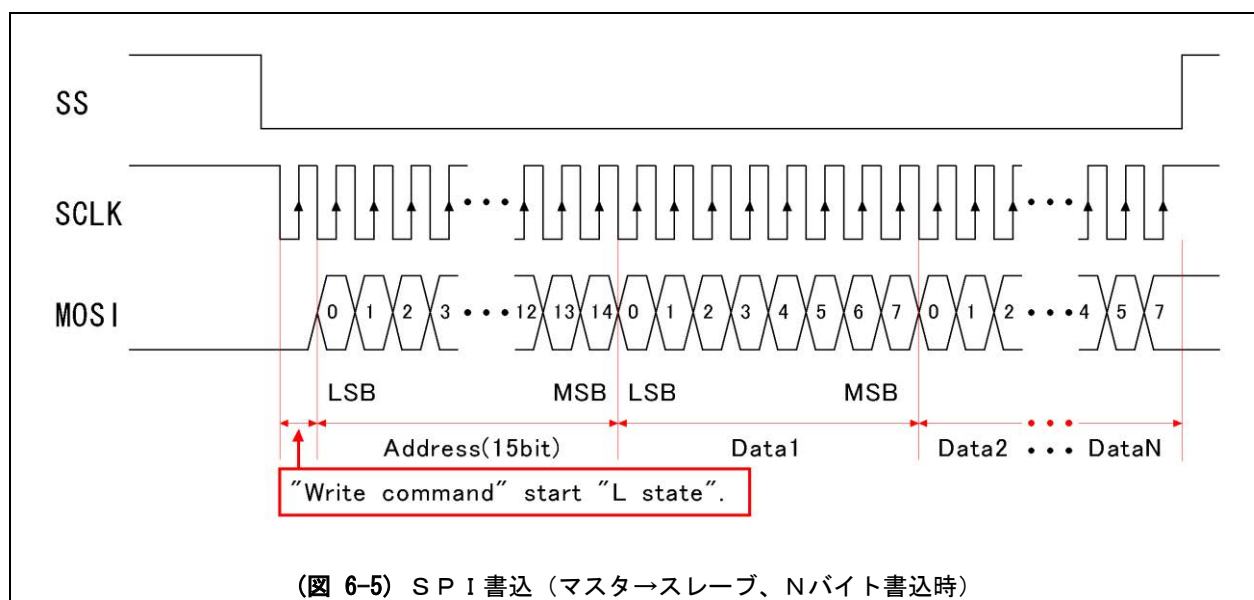
M I S O / M O S I のHレベル/Lレベルの切替は、S C K の立ち下がりエッジで変化します。

またデータラッチはS C L K の立ち上がりエッジで行います。

この関係は、（マスタ）→（スレーブ）、（スレーブ）→（マスタ）の双方向で変わりません。

S P I 書込コマンド送信時のタイミングチャートを以下に示します。



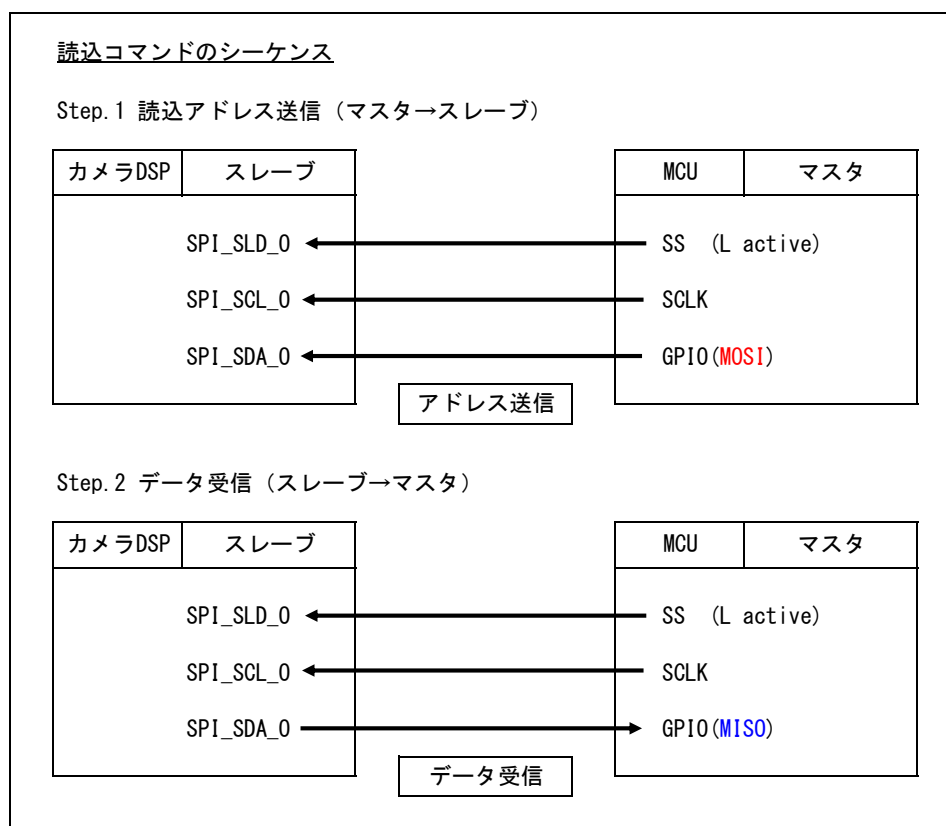


(3) コマンドとタイミングチャート (S P I 読込)

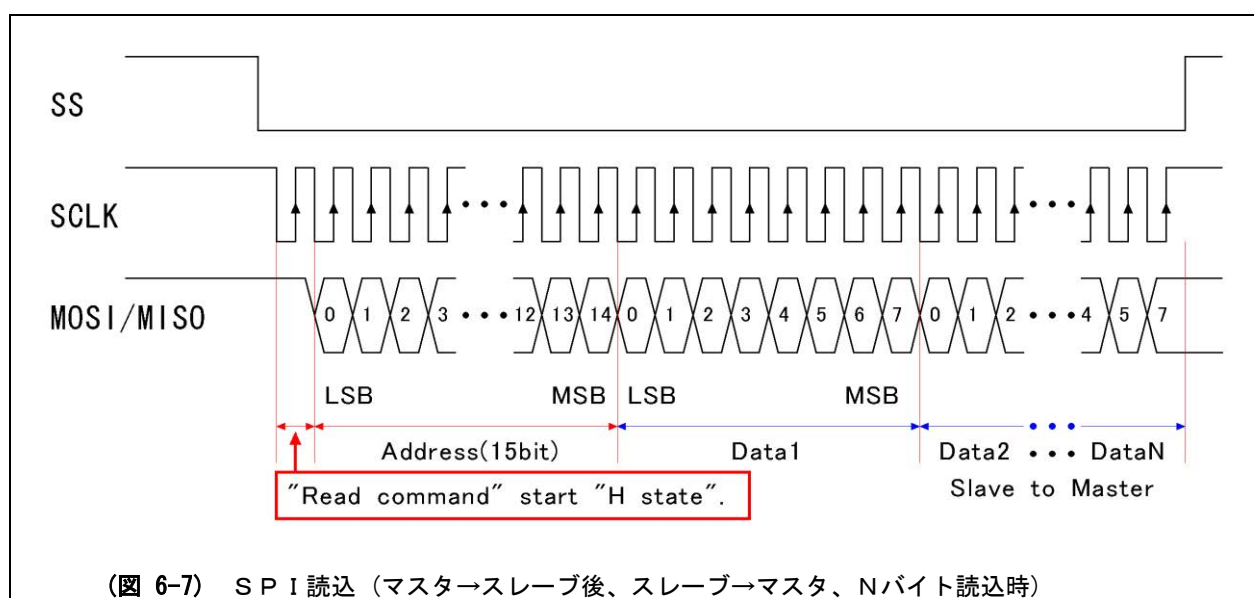
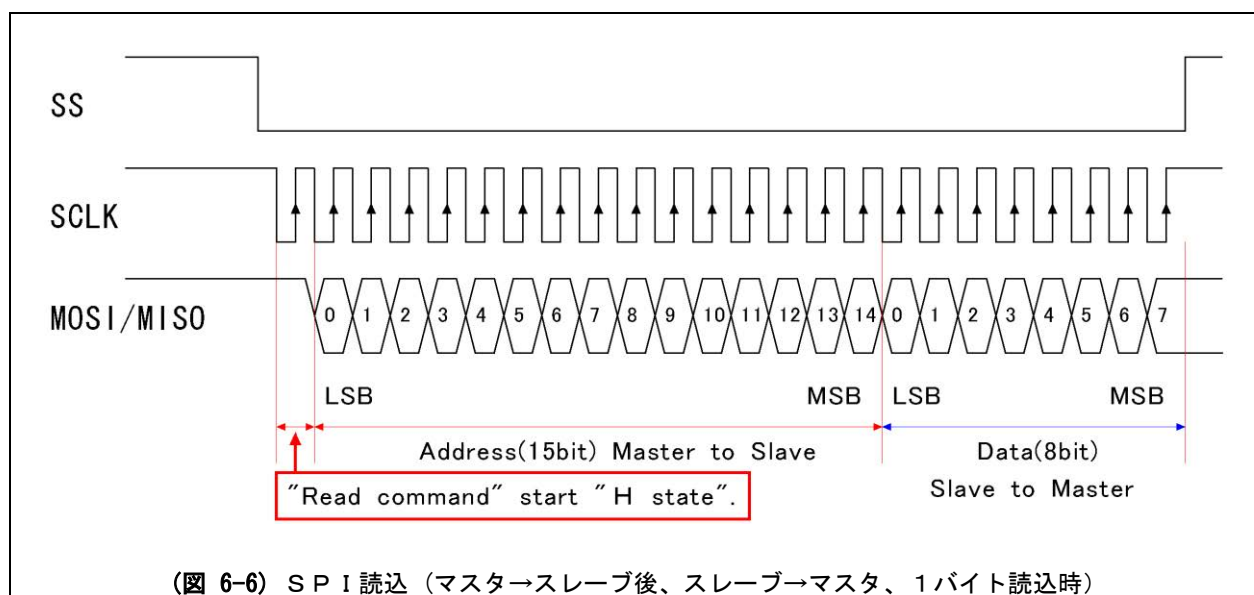
読み込みコマンド発行時、外部MCUからカメラDSPにアドレスを送信 (マスタースレーブ) 後、カメラDSPからカメラMCUに対して (スレーブ→マスタ) データが送られます。

その為、外部MCUのG P I Oピンは、アドレス送信時はM O S Iとして、データ受信時はM I S Oとして動作する必要があります。

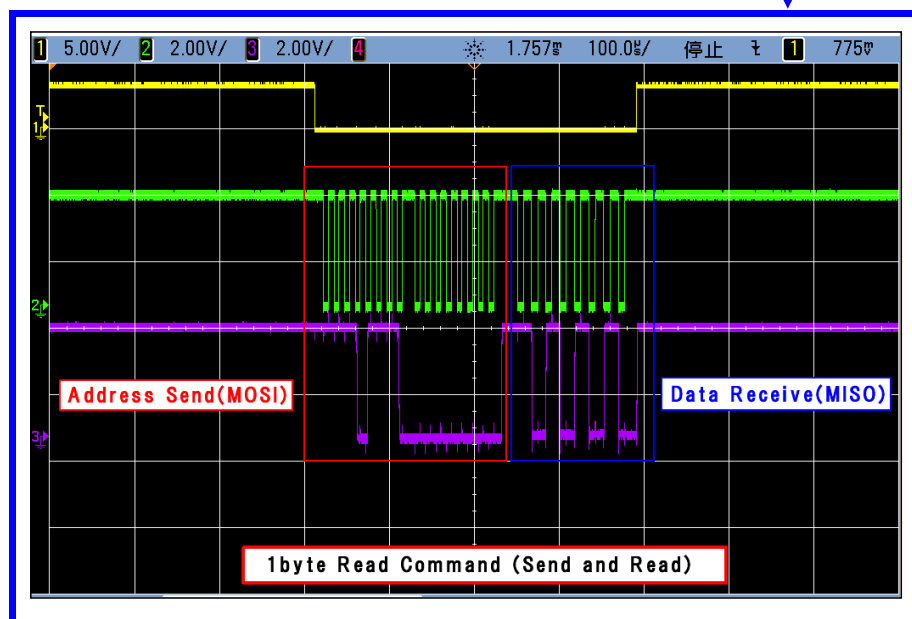
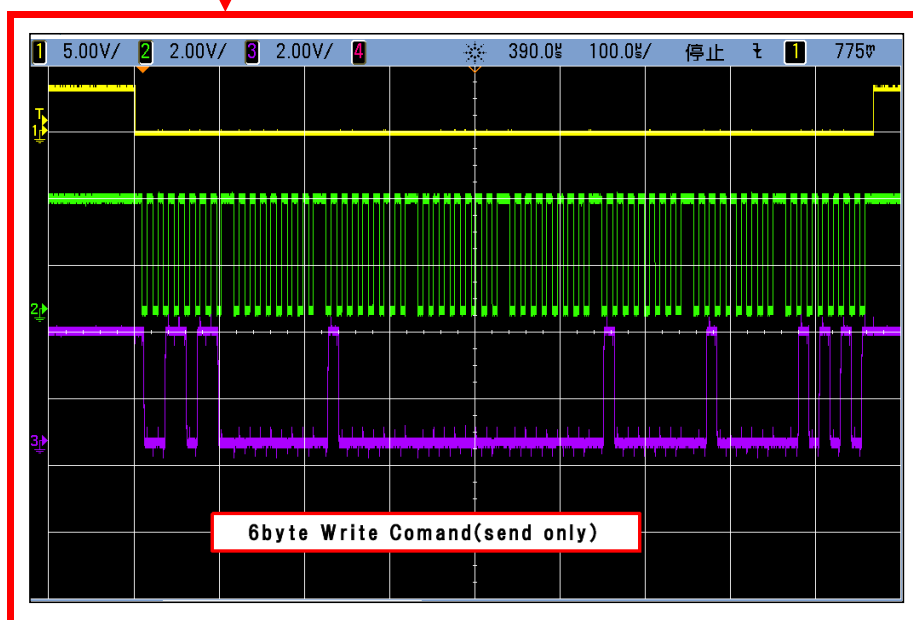
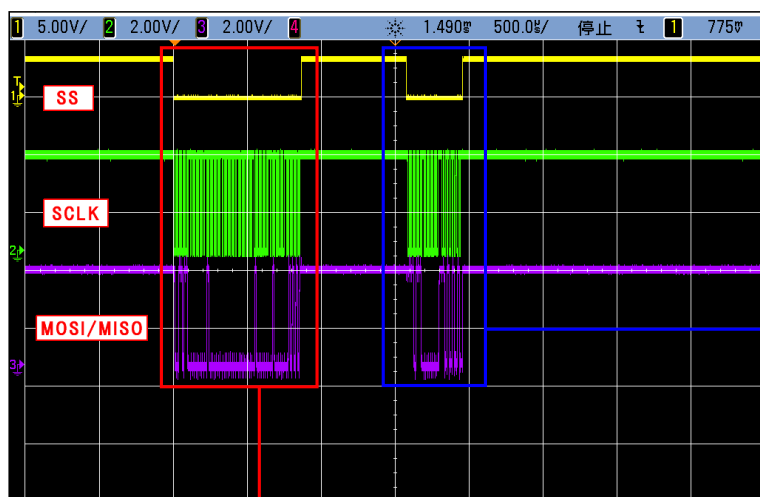
(アドレス送信時はG P I Oピンは出力、データ受信時は入力ピンとして動作する必要があります。)



S P I 読込コマンド発行時のタイミングチャート (アドレス送信後、データ受信) を以下に示します。



(4) SPIコマンド発行時の実波形 (例)



6.2 [カメラ制御コマンド] プロトコル

[カメラ制御コマンド] はカメラDSP内蔵MCUにより実行されます。

[カメラ制御コマンド] はDSP内部のインダイレクト（間接）レジスタに対し書込を行うことで実行されます。カメラ内蔵MCUは常時インダイレクトレジスタをチェックし、[カメラ制御コマンド]が見つかった場合、即座にコマンドを実行します。

インダイレクトレジスタの構成と、アドレスは下表のように定義されています。

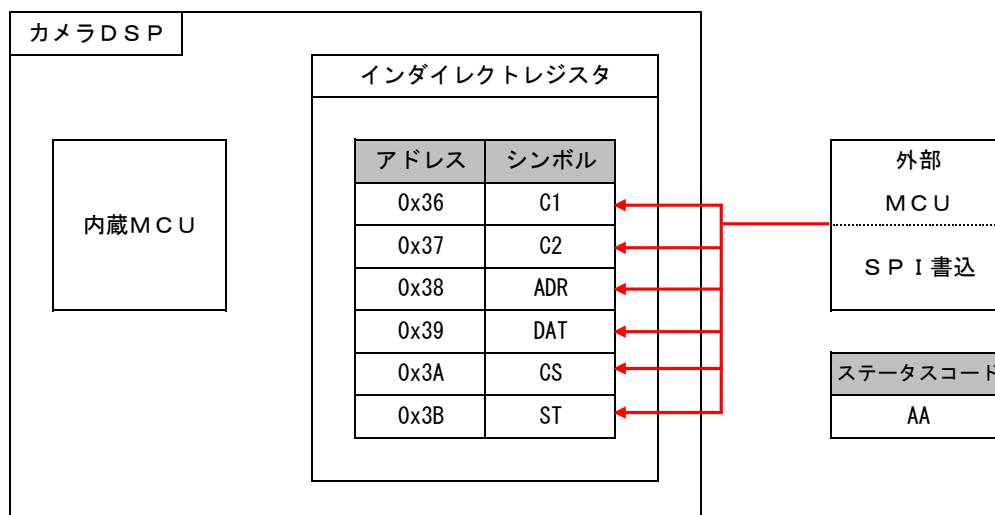
[カメラ制御コマンド] はこれらのレジスタにSPI書込で計6バイトのデータを送信することにより発行されます。

(図 6-8) インダイレクトレジスタ構成			
項番	シンボル	アドレス	レジスタの詳細
1	C1	0x0036	コマンド 1
2	C2	0x0037	コマンド 2
3	ADR	0x0038	パラメータアドレス
4	DAT	0x0039	データ（パラメータ）
5	CS	0x003A	チェックサム（C1+C2+ADR+DAT）
6	ST	0x003B	ステータスコード

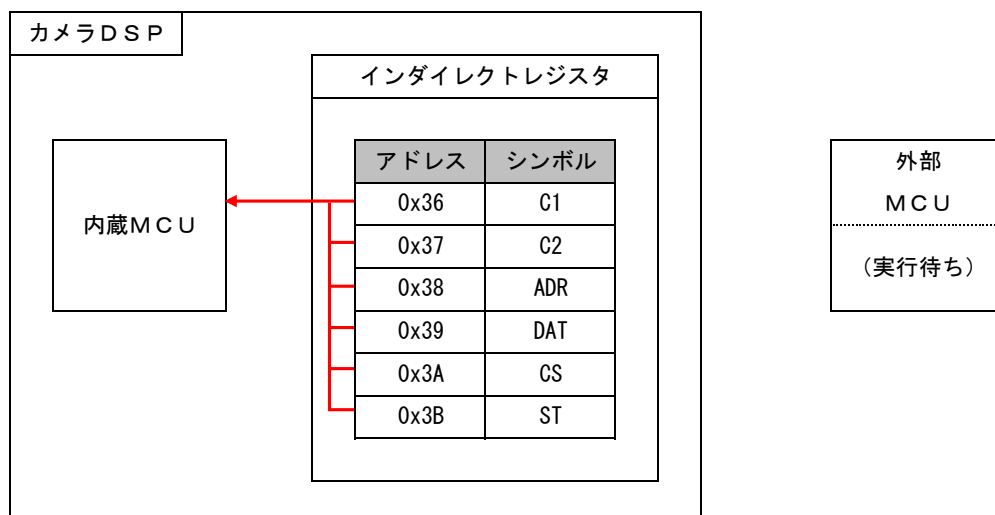
ステータスコード（ST:0x003B）の内容はコマンド発行後、内蔵MCUにより正常終了（またはエラー）状態を表すコードで上書きされます。ステータスコードの意味は下表の通りです。

(図 6-9) ステータスコード一覧		
項番	コード	コードの意味
1	AA	コマンド実行（外部MCUが書き込む、実行を指示するコード）
2	55	正常終了（カメラ内蔵MCUが上書きするコード）
3	A5	実行エラー（カメラ内蔵MCUが上書きするコード）

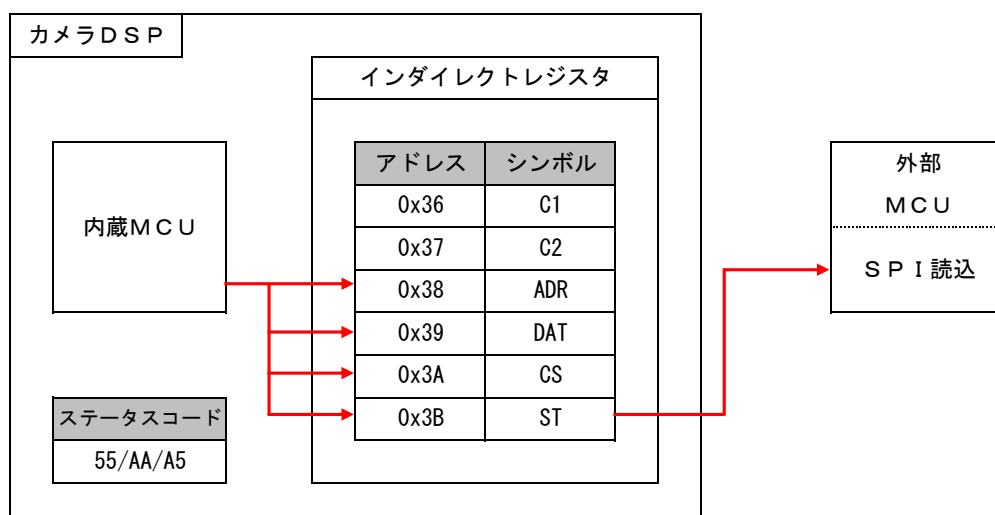
[カメラ制御コマンド] 実行は以下に示すステップで行われます。

Step.1 [カメラ制御コマンド] 送信 (外部MCU→カメラ内蔵MCU)

[カメラ制御コマンド] をSPI 書込を使用してインダイレクトレジスタ（アドレス0x36から6バイト）に書き込みます。

Step.2 [カメラ制御コマンド] のチェックと実行 (カメラ内蔵MCU)

カメラ内蔵MCUはインダイレクトレジスタをチェックし、コマンドが見つかると即実行します。

Step.3 ステータスコードの確認 (外部MCU)

カメラ内蔵MCUはコマンド実行後、インダイレクトレジスタに実行結果を上書きします。

コマンドの実行結果（または状態）は、インダイレクトレジスタ（ステータスコード：アドレス0x003B）をSPI読込を使用してチェックします。

ステータスコード“0x55”は、コマンドは正常に実行された事を意味します。

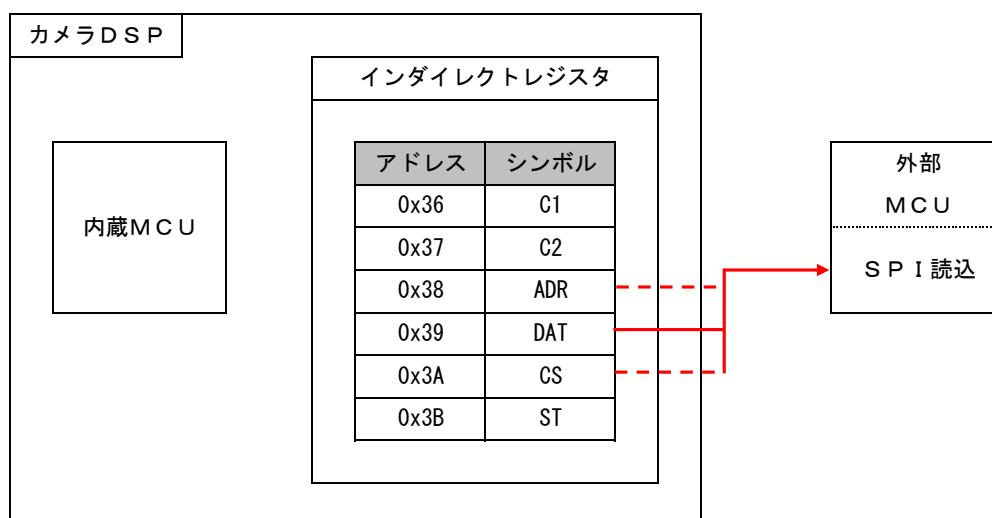
このとき、カメラ内蔵MCUは「カメラ制御コマンド」に従い、ADR/DAT/CS(アドレス 0x0038/0x0039/0x003A)に実行結果を上書きしています。

ステータスコード“0xA5”は、コマンドの誤りまたは実行エラーを意味します。

ADR/DAT/CS/ST に誤りがないかお確かめください。問題がない場合、通信エラーの可能性がありますのでコマンドを再送してください。

ステータスコードが“0xAA”のままであるときは、コマンドがまだ実行中であることを意味します。

Step.4 コマンド実行結果受信



ステータスコードが“0x55”であることを確認後、SPI読込を使用してカメラのコマンド実行結果を読み出します。

カメラパラメータを読み出すコマンドの際は、インダイレクトレジスタ（アドレス 0x0039:DAT）に指定したカメラアドレスのパラメータがセットされています。

得られたパラメータ値（DAT）が間違いないことを確認するためには、ADR/DAT/CSの3バイトを読み出し、CSの値とADR+DATの値が一致しているか比較します。

※コマンド発行後、Step.3 の手順を省き Step.4 でステータスコードを同時に読みとり、「カメラ制御コマンド」の実行状態判断とデータ取得を一度に行うことも可能です。

6.3 カメラ制御コマンド

カメラ制御コマンドは以下の7種類が定義されています。

(1) CAMERA PARAMETER READ1(パラメータアドレス: 0x400 - 0x4FF)

SPI書込データ(外部MCUからの送信データ)

シンボル	C1	C2	ADR	DAT	CS	ST
値	00	00	*1	00	*2	AA

*1: パラメータアドレスの下位1バイトです。0x00 から 0xFFの値を指定します。

*2: チェックサムは (C1 + C2 + ADR + DAT) で計算された値の下位1バイトを指定します。

※PARAMETER READ コマンドでは、DAT は 0x00 を送信してください。

SPI読込データ(インダイレクトレジスタにセットされるデータ)

シンボル	ADR	DAT	CS	ST
値	*1	*2	*3	*4

*1: パラメータアドレス (外部MCUが送信した ADR と同じです)

*2: パラメータ値

*3: チェックサム (ADR + DAT) で計算された値の下位1バイト。

*4: ステータスコード (0xAA: 実行中、0x55: 正常終了、0xA5: 実行エラー)

(2) CAMERA PARAMETER READ2(パラメータアドレス: 0x500 - 0x5FF)

SPI書込データ(外部MCUからの送信データ)

シンボル	C1	C2	ADR	DAT	CS	ST
値	00	01	*1	00	*2	AA

*1: パラメータアドレスの下位1バイトです。0x00 から 0xFFの値を指定します。

*2: チェックサムは (C1 + C2 + ADR + DAT) で計算された値の下位1バイトを指定します。

※PARAMETER READ コマンドでは、DAT は 0x00 を送信してください。

SPI読込データ(インダイレクトレジスタにセットされるデータ)

シンボル	ADR	DAT	CS	ST
値	*1	*2	*3	*4

*1: パラメータアドレス (外部MCUが送信した ADR と同じです)

*2: パラメータ値

*3: チェックサム (ADR + DAT) で計算された値の下位1バイト。

*4: ステータスコード (0xAA: 実行中、0x55: 正常終了、0xA5: 実行エラー)

(3) CAMERA PARAMETER WRITE1(パラメータアドレス : 0x400 - 0x4FF)**SPI 書込データ (外部MCUからの送信データ)**

シンボル	C1	C2	ADR	DAT	CS	ST
値	00	80	*1	*2	*3	AA

*1 : パラメータアドレスの下位 1 バイトです。0x00 から 0xFFの値を指定します。

*2 : 書き込むパラメータ値

*3 : チェックサムは (C1 + C2 + ADR + DAT) で計算された値の下位 1 バイトを指定します。

SPI 読込データ (インダイレクトレジスタにセットされるデータ)

シンボル	ADR	DAT	CS	ST
値	*1	*2	*3	*4

*1 : パラメータアドレス (外部MCUが送信した ADR と同じです)

*2 : パラメータ値 (外部MCUが送信した DAT と同じです)

*3 : チェックサム (ADR + DAT) で計算された値の下位 1 バイト。

*4 : ステータスコード (0xAA : 実行中、0x55 : 正常終了、0xA5 : 実行エラー)

(4) CAMERA PARAMETER WRITE2(パラメータアドレス : 0x500 - 0x5FF)**SPI 書込データ (外部MCUからの送信データ)**

シンボル	C1	C2	ADR	DAT	CS	ST
値	00	81	*1	*2	*3	AA

*1 : パラメータアドレスの下位 1 バイトです。0x00 から 0xFFの値を指定します。

*2 : 書き込むパラメータ値

*3 : チェックサムは (C1 + C2 + ADR + DAT) で計算された値の下位 1 バイトを指定します。

SPI 読込データ (インダイレクトレジスタにセットされるデータ)

シンボル	ADR	DAT	CS	ST
値	*1	*2	*3	*4

*1 : パラメータアドレス (外部MCUが送信した ADR と同じです)

*2 : パラメータ値 (外部MCUが送信した DAT と同じです)

*3 : チェックサム (ADR + DAT) で計算された値の下位 1 バイト。

*4 : ステータスコード (0xAA : 実行中、0x55 : 正常終了、0xA5 : 実行エラー)

(5) AREA DISPLAY ON/OFF**SPI 書込データ（外部MCUからの送信データ）**

シンボル	C1	C2	ADR	DAT	CS	ST
値	01	*1	00	*2	*3	AA

*1：ウィンドウの選択（BLC = 0x00, HSBLC = 0x01）

*2：ウィンドウ領域表示（ON = 0x01, OFF = 0x00）

*3：チェックサムは（C1 + C2 + ADR + DAT）で計算された値の下位 1 バイトを指定します。

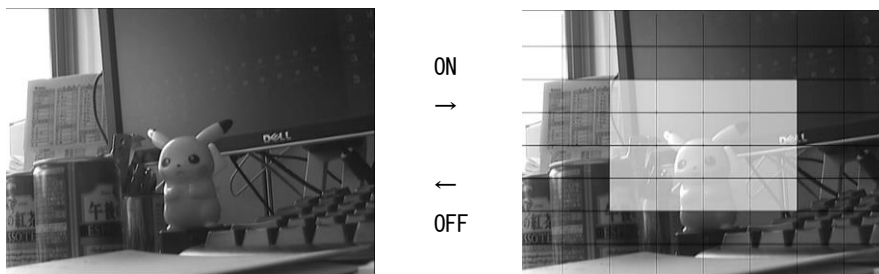
SPI 読込データ（インダイレクトレジスタにセットされるデータ）

シンボル	ADR	DAT	CS	ST
値	00	*1	*2	*3

*1：パラメータ値（外部MCUが送信した DAT と同じです）

*2：チェックサム（ADR + DAT）で計算された値の下位 1 バイト。

*3：ステータスコード（0xAA：実行中、0x55：正常終了、0xA5：実行エラー）

**(6) CAMERA OSD CONTROL (OSDメニュー操作)****SPI 書込データ（外部MCUからの送信データ）**

シンボル	C1	C2	ADR	DAT	CS	ST
値	02	00	00	*1	*2	AA

*1：5キーの方向（下表参照）

*2：チェックサムは（C1 + C2 + ADR + DAT）で計算された値の下位 1 バイトを指定します。

キー方向	DAT	CS
LEFT	01	03
RIGHT	02	04
UP	03	05
DOWN	04	06
SET	05	07

SPI 読込データ（インダイレクトレジスタにセットされるデータ）

シンボル	ADR	DAT	CS	ST
値	00	*1	*2	*3

*1：パラメータ値（外部MCUが送信した DAT と同じです）

*2：チェックサム（ADR + DAT）で計算された値の下位 1 バイト。

*3：ステータスコード（0xAA：実行中、0x55：正常終了、0xA5：実行エラー）

(7) CAMERA PARAMETER SAVES P I 書込データ（外部MCUからの送信データ）

シンボル	C1	C2	ADR	DAT	CS	ST
値	01	FF	00	00	00	AA

S P I 読込データ（インダイレクトレジスタにセットされるデータ）

シンボル	ADR	DAT	CS	ST
値	00	00	00	*1

*1：ステータスコード（0xAA：実行中、0x55：正常終了、0xA5：実行エラー）

※本コマンドを発行することにより、電源再投入後も変更したパラメータ値が有効となります。

6.4 カメラパラメータ（アドレスマップ）

表中、“—”の表記のあるビットは変更しないようご注意ください。

（変更後、パラメータ保存を行うとカメラ動作が保証されません。）

初期設定値は、表中に赤字下線付き（または、defaultの表記付き）で記載されています。

Address	Bit							
	7	6	5	4	3	2	1	0
0x400	—	—	—	—	—	—	—	—
0x401	—	—	—	—	AGC MODE		EI (*1)	—
					0x00:OFF	0x01:LOW	0:OFF	
					0x02:MID	0x03:HI	1:ON	
0x402	—	—	—	SHUTTER MODE				
				0x00:x256, 0x01:x128, 0x02:x64, 0x03:x32, 0x04:x16, 0x05:x8, 0x06:x4, 0x07:x2, 0x08:EI, 0x09:OFF (E:1/60, C:1/50), 0x0A:FL, 0x0B:1/250, 0x0C:1/500, 0x0D:1/1000, 0x0E:1/2000, 0x0F:1/5000, 0x10:1/10000, 0x11:1/100000				
0x403	—	—	—	—	—	—	—	—
0x404	—	—	—	—	—	—	—	—
0x405	—	—	—	—	—	—	—	—
0x406	—	—	—	—	SENS UP 0:OFF 1:ON	SENS UP MAX		
						0x00:x2, 0x01:x4, 0x02:x8,		
						0x03:x16, 0x04:x32, 0x05:x64, 0x06:x128, 0x07:x256 (FLD)		
0x407	—	—	—	—	—	—	—	—
0x408	—	—	—	—	—	—	—	—
0x409	—	—	—	—	—	—	—	—
0x40A	—	—	—	—	—	—	—	—
0x40B	—	—	—	—	—	—	—	—
0x40C	—	—	—	—	—	—	—	—
0x40D	—	—	—	—	—	—	—	—
0x40E	—	—	—	—	—	—	—	—
0x40F	—	—	—	—	—	—	—	—
0x410	—	—	—	—	—	—	—	—
0x411	AGC LOW MAX							
	AGC LOW MAX (dB) = ([0x05 … 0xFF] * 4 * 0.035) + 5.3 (default:0xB0, 約30 (dB))							
	※ 1 : 0x05 以上の値を設定してください。 ※ 2 : AGC MIN よりも大きな値を設定してください。							
0x412	AGC MID MAX							
	AGC MID MAX (dB) = ([0x05 … 0xFF] * 4 * 0.035) + 5.3 (default:0xD0, 約34.5 (dB))							
	※ 1 : 0x05 以上の値を設定してください。 ※ 2 : AGC MIN よりも大きな値を設定してください。							
0x413	AGC HI MAX							
	AGC HI MAX (dB) = ([0x05 … 0xFF] * 4 * 0.035) + 5.3 (default:0xFF, 約41 (dB))							
	※ 1 : 0x05 以上の値を設定してください。 ※ 2 : AGC MIN よりも大きな値を設定してください。							

（* 1）電子アイリスへの切替は、“SHUTTER MODE”パラメータを使用することを推奨します。

Address	Bit							
	7	6	5	4	3	2	1	0
0x414	AGC MIN							
	AGC MIN(dB) = ([0x05 ... 0xFF] * 4 * 0.035) + 5.3 (default:0x05, 約5.3(dB))							
	※ 1 : 0x05 以上の値を設定してください。							
	※ 2 : AGC LOW/MID/HI MAX よりも小さな値を設定してください。							
0x415	—	—	—	—	—	—	—	—
0x416	—	—	—	—	—	—	—	—
0x417	—	—	—	MANUAL GAIN(AGC OFF GAIN)				
				0x01:6(dB) ... 0x24:41(dB)				
0x418	—	—	—	—	HSBLC LEVEL			
					0x01:1(min) ... 0x04 ... 0x08:8(max)			
0x419	HSBLC WINDOW V START POSITION				HSBLC WINDOW H START POSITION			
	0x00:0(TOP) ... 0x02 ... 0x07:7(BOTTOM)				0x00:0(LEFT) ... 0x02 ... 0x07:7(RIGHT)			
0x41A	HSBLC WINDOW V SIZE				HSBLC WINDOW H SIZE			
	0x01:1(min) ... 0x04 ... 0x08:8(max)				0x01:1(min) ... 0x04 ... 0x08:8(max)			
0x41B	—	—	—	—	BLC GAIN		BLC MODE	
					0x00:LOW, 0x01:MID, 0x02:HI		0x00:OFF, 0x01:BLC, 0x02:HSBLC	
0x41C	BLC WINDOW V START POSITION				BLC WINDOW H START POSITION			
	0x00:0(TOP) ... 0x02 ... 0x07:7(BOTTOM)				0x00:0(LEFT) ... 0x02 ... 0x07:7(RIGHT)			
0x41D	BLC WINDOW V SIZE				BLC WINDOW H SIZE			
	0x01:1(min) ... 0x04 ... 0x08:8(max)				0x01:1(min) ... 0x04 ... 0x08:8(max)			
0x41E	—	—	—	—	—	—	—	—
0x41F	—	—	—	—	—	—	—	—
0x420	—	—	—	—	—	—	—	—
0x421	—	—	—	—	—	—	—	—
0x422	—	—	—	—	—	—	—	—
0x423	—	—	—	—	—	—	—	—
0x424	—	—	—	—	—	—	—	—
0x425	—	—	—	—	—	—	—	—
0x426	—	—	—	—	—	—	—	—
0x427	—	—	—	—	—	—	—	—
0x428	—	—	—	—	—	—	—	—
0x429	—	—	—	—	—	—	—	—
0x42A	—	—	—	—	—	—	—	—
0x42B	—	—	—	—	—	—	—	—
0x42C	—	—	—	—	—	—	—	—
0x42D	—	—	—	—	—	—	—	—
0x42E	—	—	—	—	—	—	—	—
0x42F	—	—	—	—	—	—	—	—
0x430	—	—	—	—	—	—	—	—

Address	Bit							
	7	6	5	4	3	2	1	0
0x431	—	—	—	—	—	—	—	—
0x432	—	—	—	—	—	—	—	—
0x433	—	—	—	—	—	—	—	—
0x434	—	—	—	—	—	—	—	—
0x435	—	—	—	—	—	—	—	—
0x436	—	—	—	—	—	—	—	—
0x437	—	—	—	—	—	—	—	—
0x438	—	—	—	—	—	—	—	—
0x439	—	—	—	—	—	—	—	—
0x43A	—	—	—	—	—	—	—	—
0x43B	—	—	—	—	—	—	—	—
0x43C	—	—	—	—	—	—	—	—
0x43D	—	—	—	—	—	—	—	—
0x43E	—	—	—	—	—	—	—	—
0x43F	—	—	—	—	—	—	—	—
0x440	—	—	—	—	—	—	—	—
0x441	—	—	—	—	—	—	—	—
0x442	—	—	—	—	—	—	—	—
0x443	—	—	—	—	—	—	—	—
0x444	—	—	—	—	—	—	—	—
0x445	—	—	—	—	—	—	—	—
0x446	—	—	—	—	—	—	—	—
0x447	—	—	—	—	—	—	—	—
0x448	—	—	—	—	—	—	WDR MODE	
							0x00:OFF	
							0x01:USER1	
							0x02:USER2	
0x449	WDR USER1 H-LEVEL				WDR USER1 L-LEVEL			
	0x00:0 (LEVEL) ... 0x03 ... 0x0F:15 (LEVEL)				0x00:0 (LEVEL) ... 0x0D ... 0x0F:15 (LEVEL)			
0x44A	WDR USER2 H-LEVEL				WDR USER2 L-LEVEL			
	0x00:0 (LEVEL) ... 0x09 ... 0x0F:15 (LEVEL)				0x00:0 (LEVEL) ... 0x0B ... 0x0F:15 (LEVEL)			
0x44B	—	—	—	—	—	—	—	—
0x44C	—	—	—	—	—	—	—	—
0x44D	—	—	—	—	—	—	—	—
0x44E	—	—	—	—	—	—	—	—
0x44F	—	—	—	—	—	—	—	—
0x450	—	—	—	—	—	—	—	3DNR
								0:OFF
								1:ON
0x451	—	3DNR LEVEL						
		0x00:0 (LEVEL) ... 0x32 ... 0x64:100 (LEVEL)						
0x452	—	—	—	—	—	—	—	—
0x453	—	—	—	—	—	—	—	—

Address	Bit							
	7	6	5	4	3	2	1	0
0x454	—	—	—	—	—	—	—	—
0x455	—	—	—	—	—	—	—	—
0x456	—	—	—	—	—	—	—	—
0x457	—	—	—	—	—	—	—	—
0x458	—	—	—	SHARPNESS				
				0x00:0 (LEVEL) … <u>0x04</u> … 0x1F:31 (LEVEL)				
0x459	—	—	—	—	—	—	—	—
0x45A	—	—	—	—	—	—	—	—
0x45B	—	—	—	—	—	—	—	—
0x45C	—	—	—	—	—	—	—	—
0x45D	—	—	—	—	—	—	—	—
0x45E	—	—	—	—	—	—	—	—
0x45F	—	—	—	—	—	—	—	—
0x460	—	—	—	—	—	—	—	MOTION
								<u>0:OFF</u> 1:ON
0x461	—	MOTION VIEW	—	—	—	—	—	—
		0:OFF <u>1:ON</u>						
0x462	MOTION AREA1 SENSITIVITY							
	0x32:0, 0x31:1 … 0x28:10 … 0x1E:20 … 0x14:30 … 0x0B:39, <u>0x0A:40</u> ※設定可能範囲は、0x0A～0x32です。値が大きいくほど検出レベルは小さくなります。							
0x463	MOTION AREA2 SENSITIVITY							
	0x32:0, 0x31:1 … 0x28:10 … 0x1E:20 … 0x14:30 … 0x0B:39, <u>0x0A:40</u> ※設定可能範囲は、0x0A～0x32です。値が大きいくほど検出レベルは小さくなります。							
0x464	MOTION AREA3 SENSITIVITY							
	0x32:0, 0x31:1 … 0x28:10 … 0x1E:20 … 0x14:30 … 0x0B:39, <u>0x0A:40</u> ※設定可能範囲は、0x0A～0x32です。値が大きいくほど検出レベルは小さくなります。							
0x465	MOTION AREA4 SENSITIVITY							
	0x32:0, 0x31:1 … 0x28:10 … 0x1E:20 … 0x14:30 … 0x0B:39, <u>0x0A:40</u> ※設定可能範囲は、0x0A～0x32です。値が大きいくほど検出レベルは小さくなります。							
0x466	MOTION AREA1 START H POSITION							
	0x04: (LEFT) … <u>0x46</u> … 0xBB: (RIGHT) ※“H POSITION”と“H size”の合計値が“0xBF”を越えないように設定してください。							
0x467	MOTION AREA1 START V POSITION							
	0x01: (TOP) … <u>0x28</u> … 0x8F: (BOTTOM) ※“V POSITION”と“V size”の合計値が“0x90”を越えないように設定してください。							
0x468	MOTION AREA1 H SIZE							
	0x04: (min) … <u>0x3E</u> … 0xBB: (max) ※“H POSITION”と“H size”の合計値が“0xBF”を越えないように設定してください。							
0x469	MOTION AREA1 V SIZE							
	0x01: (min) … <u>0x28</u> … 0x90: (max) ※“V POSITION”と“V size”の合計値が“0x90”を越えないように設定してください。							

Address	Bit							
	7	6	5	4	3	2	1	0
0x46A	MOTION AREA2 START H POSITION							
	0x04: (LEFT) ... <u>0x08</u> ... 0xBB: (RIGHT) ※“H POSITION”と“H size”の合計値が“0xBF”を越えないように設定してください。							
0x46B	MOTION AREA2 START V POSITION							
	0x01: (TOP) ... <u>0x28</u> ... 0x8F: (BOTTOM) ※“V POSITION”と“V size”の合計値が“0x90”を越えないように設定してください。							
0x46C	MOTION AREA2 H SIZE							
	0x04: (min) ... <u>0x3F</u> ... 0xBB: (max) ※“H POSITION”と“H size”の合計値が“0xBF”を越えないように設定してください。							
0x46D	MOTION AREA2 V SIZE							
	0x01: (min) ... <u>0x28</u> ... 0x90: (max) ※“V POSITION”と“V size”の合計値が“0x90”を越えないように設定してください。							
0x46E	MOTION AREA3 START H POSITION							
	0x04: (LEFT) ... <u>0x83</u> ... 0xBB: (RIGHT) ※“H POSITION”と“H size”の合計値が“0xBF”を越えないように設定してください。							
0x46F	MOTION AREA3 START V POSITION							
	0x01: (TOP) ... <u>0x28</u> ... 0x8F: (BOTTOM) ※“V POSITION”と“V size”の合計値が“0x90”を越えないように設定してください。							
0x470	MOTION AREA3 H SIZE							
	0x04: (min) ... <u>0x3E</u> ... 0xBB: (max) ※“H POSITION”と“H size”の合計値が“0xBF”を越えないように設定してください。							
0x471	MOTION AREA3 V SIZE							
	0x01: (min) ... <u>0x28</u> ... 0x90: (max) ※“V POSITION”と“V size”の合計値が“0x90”を越えないように設定してください。							
0x472	MOTION AREA4 START H POSITION							
	0x04: (LEFT) ... <u>0x08</u> ... 0xBB: (RIGHT) ※“H POSITION”と“H size”の合計値が“0xBF”を越えないように設定してください。							
0x473	MOTION AREA4 START V POSITION							
	0x01: (TOP) ... <u>0x51</u> ... 0x8F: (BOTTOM) ※“V POSITION”と“V size”の合計値が“0x90”を越えないように設定してください。							
0x474	MOTION AREA4 H SIZE							
	0x04: (min) ... <u>0xB9</u> ... 0xBB: (max) ※“H POSITION”と“H size”の合計値が“0xBF”を越えないように設定してください。							
0x475	MOTION AREA4 V SIZE							
	0x01: (min) ... <u>0x28</u> ... 0x90: (max) ※“V POSITION”と“V size”の合計値が“0x90”を越えないように設定してください。							
0x476	—	—	—	—	MOTION AREA4 DISPLAY	MOTION AREA3 DISPLAY	MOTION AREA2 DISPLAY	MOTION AREA1 DISPLAY
					<u>0:OFF</u> 1:ON	<u>0:OFF</u> 1:ON	<u>0:OFF</u> 1:ON	<u>0:OFF</u> 1:ON
0x477	—	—	—	—	—	—	—	—
0x478	—	—	—	—	—	—	—	—
0x479	—	—	—	—	—	—	—	—
0x47A	—	—	—	—	—	—	—	—
0x47B	—	—	—	—	—	—	—	—

Address	Bit							
	7	6	5	4	3	2	1	0
0x47C	—	—	—	—	—	—	—	—
0x47D	—	—	—	—	—	—	—	—
0x47E	—	—	—	—	—	—	—	—
0x47F	—	—	—	—	—	—	—	—
0x480	—	—	—	—	—	—	—	—
0x481	—	—	—	—	—	—	—	—
0x482	—	—	—	—	—	—	—	—
0x483	—	—	—	—	—	—	—	—
0x484	—	—	—	—	—	—	—	—
0x485	—	—	—	—	—	—	—	—
0x486	—	—	—	—	—	—	—	—
0x487	—	—	—	—	—	—	—	—
0x488	—	—	—	—	—	—	—	—
0x489	—	—	—	—	—	—	—	—
0x48A	—	—	—	—	—	—	—	—
0x48B	—	—	—	—	—	—	—	—
0x48C	—	—	—	—	—	—	—	—
0x48D	—	—	—	—	—	—	—	—
0x48E	—	—	—	—	—	—	—	—
0x48F	—	—	—	—	—	—	—	—
0x490	—	—	—	—	—	—	—	—
0x491	—	—	—	—	—	—	—	—
0x492	—	—	—	—	—	—	—	—
0x493	—	—	—	—	—	—	—	—
0x494	—	—	—	—	—	—	—	—
0x495	—	—	—	—	—	—	—	—
0x496	—	—	—	—	—	—	—	—
0x497	—	—	—	—	—	—	—	—
0x498	—	—	—	—	—	—	—	—
0x499	—	—	—	—	—	—	—	—
0x49A	—	—	—	—	—	—	—	—
0x49B	—	—	—	—	—	—	—	—
0x49C	—	—	—	—	—	—	—	—
0x49D	—	—	—	—	—	—	—	—
0x49E	—	—	—	—	—	—	—	—
0x49F	—	—	—	—	—	—	—	—
0x4A0	—	—	—	—	—	—	—	—
0x4A1	—	—	—	—	—	—	—	—
0x4A2	—	—	—	—	—	—	—	—
0x4A3	—	—	—	—	—	—	—	—
0x4A4	—	—	—	—	—	—	—	—
0x4A5	—	—	—	—	—	—	—	—
0x4A6	—	—	—	—	—	—	—	—
0x4A7	—	—	—	—	—	—	—	—

Address	Bit							
	7	6	5	4	3	2	1	0
0x4A8	—	—	—	—	—	—	—	ZOOM 0:OFF 1:ON
0x4A9	ZOOM (倍率) <u>0x00:x1.0</u> , 0x01:x1.1 ... 0x09:x1.9, 0x0A:x2.0, 0x0B:x2.1 ... 0x13:x2.9, 0x14:x3.0, 0x15:x3.1 ... 0x1D:x3.9, 0x1E:x4.0, 0x1F:x4.5, 0x20:x5.0, 0x21:x5.5, 0x22:x6.0, 0x23:x6.5, 0x24:x7.0, 0x25:x7.5, 0x26:x8.0, 0x27:x9, 0x28:x10, 0x29:x11, 0x2A:x12, 0x2B:x13, 0x2C:x14, 0x2D:x15, 0x2E:x16, 0x2F:x18, 0x30:x20, 0x31:x22, 0x32:x24, 0x33:x26, 0x34:x28, 0x35:x30, 0x36:x32							
0x4AA	ZOOM (水平位置) 0x00:-100 (LEFT), 0x01:-99 ... 0x63:-1, <u>0x64:0 (CENTER)</u> , 0x65:+1 ... 0xC7:+99, 0xC8:+100 (RIGHT)							
0x4AB	ZOOM (垂直位置) 0x00:-100 (UP), 0x01:-99 ... 0x63:-1, <u>0x64:0 (CENTER)</u> , 0x65:+1 ... 0xC7:+99, 0xC8:+100 (DOWN)							
0x4AC	—	—	—	—	—	—	—	—
0x4AD	—	—	—	—	—	—	—	—
0x4AE	—	—	—	—	—	—	—	—
0x4AF	—	—	—	—	—	—	—	—
0x4B0	—	—	—	—	NEG. IMAGE	FLIP		FREEZE
					0:OFF 1:ON	0x00:OFF 0x02:V	0x01:H 0x03:HV	0:OFF 1:ON
0x4B1	—	—	—	GAMMA 0x00:USER, 0x01:0.05 ... <u>0x09:0.45</u> ... 0x14:1.0				
0x4B2	—	—	—	—	—	—	—	—
0x4B3	—	—	—	—	—	—	—	—
0x4B4	—	—	—	—	—	—	—	—
0x4B5	—	—	—	—	—	—	—	—
0x4B6	—	—	—	—	—	—	—	—
0x4B7	—	—	—	—	—	—	—	—
0x4B8	—	—	—	—	—	—	—	—
0x4B9	—	—	—	—	—	—	—	—
0x4BA	—	—	—	—	—	—	—	—
0x4BB	—	—	—	—	—	—	—	—
0x4BC	—	—	—	—	—	—	—	—
0x4BD	—	—	—	—	—	—	—	—
0x4BE	—	—	—	—	—	—	—	—
0x4BF	—	—	—	—	—	—	—	—
0x4C0	—	—	—	—	—	—	—	—
0x4C1	—	—	—	—	—	—	—	—
0x4C2	—	—	—	—	—	—	—	—
0x4C3	—	—	—	—	—	—	—	—
0x4C4	—	—	—	—	—	—	—	—
0x4C5	—	—	—	—	—	—	—	—
0x4C6	—	—	—	—	—	—	—	—

Address	Bit							
	7	6	5	4	3	2	1	0
0x4C7	—	—	—	—	—	—	—	—
0x4C8	—	—	—	—	—	—	—	—
0x4C9	—	—	—	—	—	—	—	—
0x4CA	—	—	—	—	—	—	—	—
0x4CB	—	—	—	—	—	—	—	—
0x4CC	—	—	—	—	—	—	—	—
0x4CD	—	—	—	—	—	—	—	—
0x4CE	—	—	—	—	—	—	—	—
0x4CF	—	—	—	—	—	—	—	—
0x4D0	—	—	—	—	—	—	—	—
0x4D1	—	—	—	—	—	—	—	—
0x4D2	—	—	—	—	—	—	—	—
0x4D3	—	—	—	—	—	—	—	—
0x4D4	—	—	—	—	—	—	—	—
0x4D5	—	—	—	—	—	—	—	—
0x4D6	—	—	—	—	—	—	—	—
0x4D7	—	—	—	—	—	—	—	—
0x4D8	—	—	—	—	—	—	—	—
0x4D9	—	—	—	—	—	—	—	—
0x4DA	—	—	—	—	—	—	—	—
0x4DB	—	—	—	—	—	—	—	—
0x4DC	—	—	—	—	—	—	—	—
0x4DD	—	—	—	—	—	—	—	—
0x4DE	—	—	—	—	—	—	—	—
0x4DF	—	—	—	—	—	—	—	—
0x4E0	—	—	—	—	—	—	—	—
0x4E1	—	—	—	—	—	—	—	—
0x4E2	—	—	—	—	—	—	—	—
0x4E3	—	—	—	—	—	—	—	—
0x4E4	—	—	—	—	—	—	—	—
0x4E5	—	—	—	—	—	—	—	—
0x4E6	—	—	—	—	—	—	—	—
0x4E7	—	—	—	—	—	—	—	—
0x4E8	—	—	—	—	—	—	—	—
0x4E9	—	—	—	—	—	—	—	—
0x4EA	—	—	—	—	—	—	—	—
0x4EB	—	—	—	—	—	—	—	—
0x4EC	—	—	—	—	—	—	—	—
0x4ED	—	—	—	—	—	—	—	—
0x4EE	—	—	—	—	—	—	—	—
0x4EF	—	—	—	—	—	—	—	—
0x4F0	—	—	—	—	—	—	—	—
0x4F1	—	—	—	—	—	—	—	—
0x4F2	—	—	—	—	—	—	—	—

Address	Bit							
	7	6	5	4	3	2	1	0
0x4F3	—	SETUP (*1)	SETUP LEVEL (*1) 0x00 ... <u>0x1C</u> ... 0x0x3F					
		ON/OFF						
		0:0 (IRE) <u>1:7.5 (IRE)</u>						
0x4F4	—	—	—	—	—	—	—	—
0x4F5	—	—	—	—	—	—	—	—
0x4F6	—	—	—	—	—	—	—	—
0x4F7	—	—	—	—	—	—	—	—
0x4F8	—	—	—	—	—	—	—	—
0x4F9	—	—	—	—	—	—	—	—
0x4FA	—	—	—	—	—	—	—	—
0x4FB	—	—	—	—	—	—	—	—
0x4FC	—	—	—	—	—	—	—	—
0x4FD	—	—	—	—	—	—	—	—
0x4FE	BPC LEVEL					BPC FLD		
	0x01:1 ... <u>0x03:3</u> ... 0x04:4					0x00:x4, 0x01:x8, 0x02:x16, 0x03:x32, <u>0x04:x64</u>		
0x4FF	—	—	—	—	—	—	—	—

(* 1) E I Aバージョンのみ有効

Address	Bit							
	7	6	5	4	3	2	1	0
0x500	—	—	—	—	—	—	—	—
0x501	—	—	—	—	—	—	—	—
0x502	—	—	—	—	—	—	—	—
0x503	—	—	—	—	—	—	—	—
0x504	—	—	—	—	—	—	—	—
0x505	—	—	—	—	—	—	—	—
0x506	—	—	—	—	—	—	—	—
0x507	—	—	—	—	—	—	—	—
0x508	—	—	—	—	—	—	—	—
0x509	—	—	—	—	—	—	—	—
0x50A	—	—	—	—	—	—	—	—
0x50B	—	—	—	—	—	—	—	—
0x50C	—	—	—	—	—	—	—	—
0x50D	—	—	—	—	—	—	—	—
0x50E	—	—	—	—	—	—	—	—
0x50F	—	—	—	—	—	—	—	—
0x510	—	—	—	—	—	—	—	—
0x511	—	—	—	—	—	—	—	—
0x512	—	—	—	—	—	—	—	—
0x513	—	—	—	—	—	—	—	—
0x514	—	—	—	—	—	—	—	—
0x515	—	—	—	—	—	—	—	—
0x516	—	—	—	—	—	—	—	—
0x517	—	—	—	—	—	—	—	—
0x518	—	—	—	—	—	—	—	—
0x519	—	—	—	—	—	—	—	—
0x51A	—	—	—	—	—	—	—	—
0x51B	—	—	—	—	—	—	—	—
0x51C	—	—	—	—	—	—	—	—
0x51D	—	—	—	—	—	—	—	—
0x51E	—	—	—	—	—	—	—	—
0x51F	—	—	—	—	—	—	—	—
0x520	—	—	—	—	—	—	—	—
0x521	—	—	—	—	—	—	—	—
0x522	—	—	—	—	—	—	—	—
0x523	—	—	—	—	—	—	—	—
0x524	—	—	—	—	—	—	—	—
0x525	—	—	—	—	—	—	—	—
0x526	—	—	—	—	—	—	—	—
0x527	—	—	—	—	—	—	—	—
0x528	—	—	—	—	—	—	—	—
0x529	—	—	—	—	—	—	—	—
0x52A	—	—	—	—	—	—	—	—
0x52B	—	—	—	—	—	—	—	—
0x52C	—	—	—	—	—	—	—	—
0x52D	—	—	—	—	—	—	—	—

Address	Bit							
	7	6	5	4	3	2	1	0
0x52E	—	—	—	—	—	—	—	—
0x52F	—	—	—	—	—	—	—	—
0x530	—	—	—	—	—	—	—	—
0x531	—	—	—	—	—	—	—	—
0x532	—	—	—	—	—	—	—	—
0x533	—	—	—	—	—	—	—	—
0x534	—	—	—	—	—	—	—	—
0x535	—	—	—	—	—	—	—	—
0x536	—	—	—	—	—	—	—	—
0x537	—	—	—	—	—	—	—	—
0x538	—	—	—	—	—	—	—	—
0x539	—	—	—	—	—	—	—	—
0x53A	—	—	—	—	—	—	—	—
0x53B	—	—	—	—	—	—	—	—
0x53C	—	—	—	—	—	—	—	—
0x53D	—	—	—	—	—	—	—	—
0x53E	—	—	—	—	—	—	—	—
0x53F	—	—	—	—	—	—	—	—
0x540	—	—	—	—	—	—	—	—
0x541	—	—	—	—	—	—	—	—
0x542	—	—	—	—	—	—	—	—
0x543	—	—	—	—	—	—	—	—
0x544	—	—	—	—	—	—	—	—
0x545	—	—	—	—	—	—	—	—
0x546	—	—	—	—	—	—	—	—
0x547	—	—	—	—	—	—	—	—
0x548	—	—	—	—	—	—	—	—
0x549	—	—	—	—	—	—	—	—
0x54A	—	—	—	—	—	—	—	—
0x54B	—	—	—	—	—	—	—	—
0x54C	—	—	—	—	—	—	—	—
0x54D	—	—	—	—	—	—	—	—
0x54E	—	—	—	—	—	—	—	—
0x54F	—	—	—	—	—	—	—	—
0x550	—	—	—	—	—	—	—	—
0x551	—	—	—	—	—	—	—	—
0x552	—	—	—	—	—	—	—	—
0x553	—	—	—	—	—	—	—	—
0x554	—	—	—	—	—	—	—	—
0x555	—	—	—	—	—	—	—	—
0x556	—	—	—	—	—	—	—	—
0x557	—	—	—	—	—	—	—	—
0x558	—	—	—	—	—	—	—	—
0x559	—	—	—	—	—	—	—	—
0x55A	—	—	—	—	—	—	—	—
0x55B	—	—	—	—	—	—	—	—

Address	Bit							
	7	6	5	4	3	2	1	0
0x55C	—	—	—	—	—	—	—	—
0x55D	—	—	—	—	—	—	—	—
0x55E	—	—	—	—	—	—	—	—
0x55F	—	—	—	—	—	—	—	—
0x560	—	—	—	—	—	—	—	—
0x561	—	—	—	—	—	—	—	—
0x562	—	—	—	—	—	—	—	—
0x563	—	—	—	—	—	—	—	—
0x564	—	—	—	—	—	—	—	—
0x565	—	—	—	—	—	—	—	—
0x566	—	—	—	—	—	—	—	—
0x567	—	—	—	—	—	—	—	—
0x568	—	—	—	—	—	—	—	—
0x569	—	—	—	—	—	—	—	—
0x56A	—	—	—	—	—	—	—	—
0x56B	—	—	—	—	—	—	—	—
0x56C	—	—	—	—	—	—	—	—
0x56D	—	—	—	—	—	—	—	—
0x56E	—	—	—	—	—	—	—	—
0x56F	—	—	—	—	—	—	—	—
0x570	—	—	—	—	—	—	—	—
0x571	—	—	—	—	—	—	—	—
0x572	—	—	—	—	—	—	—	—
0x573	—	—	—	—	—	—	—	—
0x574	—	—	—	—	—	—	—	—
0x575	—	—	—	—	—	—	—	—
0x576	—	—	—	—	—	—	—	—
0x577	—	—	—	—	—	—	—	—
0x578	—	—	—	—	—	—	—	—
0x579	—	—	—	—	—	—	—	—
0x57A	—	—	—	—	—	—	—	—
0x57B	—	—	—	—	—	—	—	—
0x57C	—	—	—	—	—	—	—	—
0x57D	—	—	—	—	—	—	—	—
0x57E	—	—	—	—	—	—	—	—
0x57F	—	—	—	—	—	—	—	—
0x580	—	—	—	—	—	—	—	—
0x581	—	—	—	—	—	—	—	—
0x582	—	—	—	—	—	—	—	—
0x583	—	—	—	—	—	—	—	—
0x584	—	—	—	—	—	—	—	—
0x585	—	—	—	—	—	—	—	—
0x586	—	—	—	—	—	—	—	—
0x587	—	—	—	—	—	—	—	—
0x588	—	—	—	—	—	—	—	—
0x589	—	—	—	—	—	—	—	—

Address	Bit							
	7	6	5	4	3	2	1	0
0x58A	—	—	—	—	—	—	—	—
0x58B	—	—	—	—	—	—	—	—
0x58C	—	—	—	—	—	—	—	—
0x58D	—	—	—	—	—	—	—	—
0x58E	—	—	—	—	—	—	—	—
0x58F	—	—	—	—	—	—	—	—
0x590	—	—	—	—	—	—	—	—
0x591	—	—	—	—	—	—	—	—
0x592	—	—	—	—	—	—	—	—
0x593	—	—	—	—	—	—	—	—
0x594	—	—	—	—	—	—	—	—
0x595	—	—	—	—	—	—	—	—
0x596	—	—	—	—	—	—	—	—
0x597	—	—	—	—	—	—	—	—
0x598	—	—	—	—	—	—	—	—
0x599	—	—	—	—	—	—	—	—
0x59A	—	—	—	—	—	—	—	—
0x59B	—	—	—	—	—	—	—	—
0x59C	—	—	—	—	—	—	—	—
0x59D	—	—	—	—	—	—	—	—
0x59E	—	—	—	—	—	—	—	—
0x59F	—	—	—	—	—	—	—	—
0x5A0	—	—	—	—	—	—	—	—
0x5A1	—	—	—	—	—	—	—	—
0x5A2	—	—	—	—	—	—	—	—
0x5A3	—	—	—	—	—	—	—	—
0x5A4	—	—	—	—	—	—	—	—
0x5A5	—	—	—	—	—	—	—	—
0x5A6	—	—	—	—	—	—	—	—
0x5A7	—	—	—	—	—	—	—	—
0x5A8	—	—	—	—	—	—	—	—
0x5A9	—	—	—	—	—	—	—	—
0x5AA	—	—	—	—	—	—	—	—
0x5AB	—	—	—	—	—	—	—	—
0x5AC	—	—	—	—	—	—	—	—
0x5AD	—	—	—	—	—	—	—	—
0x5AE	—	—	—	—	—	—	—	—
0x5AF	—	—	—	—	—	—	—	—
0x5B0	—	—	—	—	—	—	—	—
0x5B1	—	—	—	—	—	—	—	—
0x5B2	—	—	—	—	—	—	—	—
0x5B3	—	—	—	—	—	—	—	—
0x5B4	—	—	—	—	—	—	—	—
0x5B5	—	—	—	—	—	—	—	—
0x5B6	—	—	—	—	—	—	—	—
0x5B7	—	—	—	—	—	—	—	—

Address	Bit							
	7	6	5	4	3	2	1	0
0x5B8	—	—	—	—	—	—	—	—
0x5B9	—	—	—	—	—	—	—	—
0x5BA	—	—	—	—	—	—	—	—
0x5BB	—	—	—	—	—	—	—	—
0x5BC	—	—	—	—	—	—	—	—
0x5BD	—	—	—	—	—	—	—	—
0x5BE	—	—	—	—	—	—	—	—
0x5BF	—	—	—	—	—	—	—	—
0x5C0	—	—	—	—	—	—	—	—
0x5C1	—	—	—	—	—	—	—	—
0x5C2	—	—	—	—	—	—	—	—
0x5C3	—	—	—	—	—	—	—	—
0x5C4	—	—	—	—	—	—	—	—
0x5C5	—	—	—	—	—	—	—	—
0x5C6	—	—	—	—	—	—	—	—
0x5C7	—	—	—	—	—	—	—	—
0x5C8	—	—	—	—	—	—	—	—
0x5C9	—	—	—	—	—	—	—	—
0x5CA	—	—	—	—	—	—	—	—
0x5CB	—	—	—	—	—	—	—	—
0x5CC	—	—	—	—	—	—	—	—
0x5CD	—	—	—	—	—	—	—	—
0x5CE	—	—	—	—	—	—	—	—
0x5CF	—	—	—	—	—	—	—	—
0x5D0	—	—	—	—	—	—	—	—
0x5D1	—	—	—	—	—	—	—	—
0x5D2	—	—	—	—	—	—	—	—
0x5D3	—	—	—	—	—	—	—	—
0x5D4	—	—	—	—	—	—	—	—
0x5D5	—	—	—	—	—	—	—	—
0x5D6	—	—	—	—	—	—	—	—
0x5D7	—	—	—	—	—	—	—	—
0x5D8	—	—	—	—	—	—	—	—
0x5D9	—	—	—	—	—	—	—	—
0x5DA	—	—	—	—	—	—	—	—
0x5DB	—	—	—	—	—	—	—	—
0x5DC	—	—	—	—	—	—	—	—
0x5DD	—	—	—	—	—	—	—	—
0x5DE	—	—	—	—	—	—	—	—
0x5DF	—	—	—	—	—	—	—	—

Address	Bit							
	7	6	5	4	3	2	1	0
0x5E0	—	—	—	—	—	—	—	BPC 0: (OFF) 1: START
0x5E1	—	—	—	—	—	—	—	FACTORY RESET 0: OFF 1: ON
0x5E2	—	—	—	—	—	—	—	DIGIT OUT 0: OFF 1: ON
0x5E3	—	—	—	—	—	—	—	—
0x5E4	—	—	—	—	—	—	—	—
0x5E5	—	—	—	—	—	—	—	—
0x5E6	—	—	—	—	—	—	—	—
0x5E7	—	—	—	—	—	—	—	—
0x5E8	—	—	—	—	—	—	—	—
0x5E9	—	—	—	—	—	—	—	—
0x5EA	—	—	—	—	—	—	—	—
0x5EB	—	—	—	—	—	—	—	—
0x5EC	—	—	—	—	—	—	—	—
0x5ED	—	—	—	—	—	—	—	—
0x5EE	—	—	—	—	—	—	—	—
0x5EF	—	—	—	—	—	—	—	—
0x5F0	—	—	—	—	—	—	—	—
0x5F1	—	—	—	—	—	—	—	—
0x5F2	—	—	—	—	—	—	—	—
0x5F3	—	—	—	—	—	—	—	—
0x5F4	—	—	—	—	—	—	—	—
0x5F5	—	—	—	—	—	—	—	—
0x5F6	—	—	—	—	—	—	—	—
0x5F7	—	—	—	—	—	—	—	—
0x5F8	—	—	—	—	—	—	—	—
0x5F9	—	—	—	—	—	—	—	—
0x5FA	—	—	—	—	—	—	—	—
0x5FB	—	—	—	—	—	—	—	—
0x5FC	—	—	—	—	—	—	—	—
0x5FD	—	—	—	—	—	—	—	—
0x5FE	—	—	—	—	—	—	—	—
0x5FF	—	—	—	—	—	—	—	—

6.5 S P I 通信例

以下のコードは S P I 通信を使用してカメラパラメータを変更する具体的なプログラム例です。
この例では、A G C モードを 3 線式 S P I 通信を使用して切り換えます。

(1) サンプルコード

このサンプルコードは、Renesas 製ワンチップマイコン、R8C/1B を使用して実行確認されています。
お客様の応用では、実際にお使いになる M C U に合わせてコードを修正する必要があります。
このコードを参考にした際のお客様の不具合については、弊社は保証しかねますのでご了承ください。

```
//-----
//
// FILE      : SPI_TEST_3W.c
//            SPI communication test program
// DATE      : 2012.02.27
// DESCRIPTION : Test program to change camera parameters
//            by using MCU GPIO.
//            (SPI 3wire communication test)
// Wattec S.Igarashi
// CPU GROUP :1B
//
//-----

#include    sfr_r81b.h

#define     TRUE      1
#define     FALSE     0
#define     CLKMS     500           // for wait counter

// global variables-----
unsigned char SEND_BUFF[0x10] ;    // Send Buffer(16 byte)
unsigned char READ_BUFF[0x10] ;    // Receive Buffer(16 byte)
unsigned char SW_STATUS ;          // SW input value
unsigned char SW_S_C[4] ;          // temporary variables for avoid SW chattering
unsigned int  Adr_C = 0x400 ;      // Address counter

// prototyping -----
// checking to which SW is ON
unsigned char  check_key_status(void) ;

// SPI Write (3wire)
unsigned char  SPI_WRITE_3W(unsigned char, unsigned char) ;

// SPI Read (3wire)
unsigned char  SPI_READ_3W(unsigned char, unsigned char) ;

// CAMERA CONTROL SUB ROUTINES
unsigned char  AGC_MODE(void) ;
unsigned char  BLC_AREA_ON(void) ;
unsigned char  BLC_AREA_OFF(void) ;
unsigned char  INDIRECT_READ(unsigned int) ;
```

```

// etc.
void          wait_ms(unsigned int) ;
void          wait_nop(void) ;

// ----- main loop -----
void main(void)
{
    int      i ;                // loop counter

    // initialize MCU
    // inhibit interrupt(INT0)
    int0en = 0 ;
    prc0   = 1 ;
    hra00  = 1 ;
    cm06   = 0 ;
    wait_nop() ;
    hra01  = 1 ;
    ocd2 = 1 ;

    // initialize GPIO port -----
    // port direction (input = 0 / output = 1)
    // port in group1(P1) are all SW input
    pd1_0 = 0 ;                // UP
    pd1_1 = 0 ;                // DOWN
    pd1_2 = 0 ;                // LEFT
    pd1_3 = 0 ;                // RIGHT
    pd1_4 = 0 ;                // SET
    pd1_5 = 0 ;                // DIP1
    pd1_6 = 0 ;                // DIP2
    pd1_7 = 0 ;                // DIP3

    // port I/O setting
    // P3_3,4,5,7 are using SPI communication
    pd3_3 = 0 ;                // MISO/SIMO(3wire)
    pd3_4 = 1 ;                // SCLK
    pd3_5 = 1 ;                // SS
    pd3_7 = 1 ;
    pd4_5 = 1 ;                // LED for status display

    // initialize output port
    p3_4 = 1 ;                // SCLK = H
    p3_5 = 1 ;                // SS = H
    p3_7 = 1 ;
    p4_5 = 1 ;                // LED OFF

    // initialize SW input port ALL OFF (ON = L)
    p1 = 0xFF ;

    //----- MAIN LOOP -----
    while(1) {
        i = 0 ;                // reset counter
        while(i<3) {           // read SW status in 3 times
            SW_S_C[i] = p1 ;

```

```

        wait_ms(10) ;                // 5ms waite
        SW_S_C[i+1] = p1 ;           // read one more
        if (SW_S_C[i] == SW_S_C[i+1]) {
            i++ ;
        }else{
            i = 0 ;
        }
    }

    // if SW status equal 3 times, check previous SW status.
    // if it is not equal, key input was changed and stable.
    if(SW_STATUS != SW_S_C[3]) {
        SW_STATUS = SW_S_C[3] ;
        // judge which SW is ON and processing SPI communication.
        if(check_key_status() != TRUE) {
            p4_5 = 0 ;                // LED lit ON
        }else{
            p4_5 = 1 ;
        }
    }
}

// end of main-----
//-----
// FUNCTION: check_key_status
//
// Decode SW status and process SPI communication.
// if illegal code was read, return FALSE.
//-----
unsichack_key_status(void)
{
    unsigned    char    i ;
    unsigned    char    RetV ;        // TRUE/FALSE

    // decode 5 key
    switch( SW_STATUS & 0x1F ) {
        case 0x1E:                // 5key UP
            // INC address and Read Parameter
            if (Adr_C == 0x5FF) {
                Adr_C = 0x400 ;
            }else{
                Adr_C++ ;
            }
            RetV = INDIRECT_READ(Adr_C) ;
            break ;
        case 0x1D:                // 5key DOWN
            // DEC adress and Read Parameter
            if (Adr_C == 0x400) {
                Adr_C = 0x5FF ;
            }else{
                Adr_C-- ;
            }
            RetV = INDIRECT_READ(Adr_C) ;
    }
}

```

```

        break ;
    case 0x1B:                // 5key LEFT
        RetV = BLC_AREA_OFF() ;
        break ;
    case 0x17:                // 5key RIGHT
        RetV = BLC_AREA_ON() ;
        break ;
    case 0x0F:                // 5key SET
        // AGC MODE change
        RetV = AGC_MODE() ;
        break ;
    default:

        RetV = TRUE ;        // nothing to do
        break ;
}

return RetV ;
}

unsigned char AGC_MODE(void)
{
    unsigned char    AGC_V ;        // AGC MAX = 0:OFF/1:LOW/2:MID/3:HI
    unsigned char    RetV ;        // temp

    // AGC MAX
    // ADDRESS 0x401 READ
    SEND_BUFF[2] = 0x00 ;
    SEND_BUFF[3] = 0x00 ;
    SEND_BUFF[4] = 0x01 ;
    SEND_BUFF[5] = 0x00 ;
    SEND_BUFF[6] = 0x01 ;
    SEND_BUFF[7] = 0xAA ;
    if (SPI_WRITE_3W(0x36, 8) == FALSE) {
        return FALSE ;
    }else{
        RetV = SPI_READ_3W(0x38, 4) ;
    }

    // AGC MODE value (bit3-2)
    AGC_V = (READ_BUFF[1] & 0x0C) >> 2 ;    // mask 00001100 and shift
    if (AGC_V == 3) {                        // 0->1->2->3->0
        AGC_V = 0;
    }else{
        AGC_V++;
    }
    wait_ms(1) ;

    // ADDRESS 0x401 WRITE
    SEND_BUFF[2] = 0x00 ;                    // BANK-0
    SEND_BUFF[3] = 0x80 ;
    SEND_BUFF[4] = 0x01 ;
    SEND_BUFF[5] = (READ_BUFF[1] & 0xF3) + (AGC_V << 2) ;
    SEND_BUFF[6] = SEND_BUFF[2] + SEND_BUFF[3] + SEND_BUFF[4] + SEND_BUFF[5] ;

```



```

    SEND_BUFF[7] = 0xAA ;
    return SPI_WRITE_3W(0x36, 8) ;
}

unsigned char BLC_AREA_ON(void)
{

    SEND_BUFF[2]    = 0x01 ;
    SEND_BUFF[3]    = 0x00 ;
    SEND_BUFF[4]    = 0x00 ;
    SEND_BUFF[5]    = 0x01 ;           // BLC AREA ON
    SEND_BUFF[6]    = 0x02 ;
    SEND_BUFF[7]    = 0xAA ;

    return SPI_WRITE_3W(0x36, 8) ;
}

unsigned char BLC_AREA_OFF(void)
{

    SEND_BUFF[2]    = 0x01 ;
    SEND_BUFF[3]    = 0x00 ;
    SEND_BUFF[4]    = 0x00 ;
    SEND_BUFF[5]    = 0x00 ;           //BLC AREA OFF
    SEND_BUFF[6]    = 0x01 ;
    SEND_BUFF[7]    = 0xAA ;

    return SPI_WRITE_3W(0x36, 8) ;
}

unsigned char INDIRECT_READ(unsigned int Pm_Adr)
{
    unsigned char Bank ;
    unsigned char Adrs ;

    Bank = (unsigned char) (Pm_Adr >> 8) - 0x04 ;
    Adrs = (unsigned char) (Pm_Adr) ;

    SEND_BUFF[2]    = 0x00 ;
    SEND_BUFF[3]    = Bank ;
    SEND_BUFF[4]    = Adrs ;
    SEND_BUFF[5]    = 0x00 ;
    SEND_BUFF[6]    = (unsigned char) (Bank + Adrs) ;
    SEND_BUFF[7]    = 0xAA ;

    if (SPI_WRITE_3W(0x36, 8) == TRUE ) {
        return SPI_READ_3W(0x38, 4) ;
    }else{
        return FALSE ;
    }
}

```

```

//-----
// FUNCTION: SPI_WRITE_3W
//
// SPI WRITE SEND_BUFF[0..(W_Byte-1)]
// In case of Indirect access, it must W_Adr = 0x36 and W_Byte = 8.
// In case of 3Wire, it need change GPIO(MISO/MOSI) direction
// input/output.
//-----
unsigned char   SPI_WRITE_3W(unsigned char W_Adr, unsigned char W_Byte)
{
    int i ;
    int j ;
    int WriteCMD ;
    unsigned char   SendBit ;

    // make SPI WRITE COMMAND
    WriteCMD = (int)W_Adr << 1 ;
    SEND_BUFF[0] = (unsigned char)WriteCMD ;           // lower 8bit
    SEND_BUFF[1] = (unsigned char)(WriteCMD >> 8) ;    // higher 8bit

    i = 0 ;           // Byte Counter

    // initialize SPI bus
    p3_5 = 1 ;        // SS   = H
    p3_4 = 1 ;        // SCLK = H

    wait_nop() ;

    // SPI WRITE COMMAND send start
    pd3_3 = 1;           // set port direction MOSI(output)
    p3_3 = 1 ;           // MOSI = H
    p3_5 = 0 ;           // SS   = L
    while( i < W_Byte ){ // W_Byte must 8 in INDIRECT ACCESS
        j = 1 ;          // Bit Mask
        while(j < 0xFF) { // 8bit send
            SendBit = SEND_BUFF[i] & j ? 1 : 0 ; // bit test
            j = j << 1 ; // j *= 2 ;
            p3_4 = 0 ;    // SCLK = L
            p3_3 = SendBit ; // output LSB first bit data on MOSI
            wait_nop() ;
            p3_4 = 1 ;    // SCLK = H(latch)
        }
        i ++ ;           // Next byte
        wait_nop() ;
    }

    p3_5 = 1 ;           // SS = H(END)
    pd3_3 = 0;           // set port direction MISO(input)

    wait_ms(1) ;

    // CTL-COMMAND execution is done ?
    READ_BUFF[0] = 0xAA ; // initialize
    while( READ_BUFF[0] == 0xAA ){ // 0xAA means busy

```

```

        SPI_READ_3W(0x3B, 1) ;           // ST CODE read
        wait_ms(1) ;                     // 1mS wait
    }
    if( READ_BUFF[0] == 0x55) {           // 0x55 means EXEC. NORMALLY
        return TRUE ;
    }else{
        return FALSE ;
    }
}

//-----
// FUNCTION: SPI_READ_3W
//
// SPI WRITE READ_BUFF[0.. (R_Byte-1)]
// In case of 3Wire, it need change GPIO(MISO/MOSI) direction
// input/output.
//-----
unsigned char SPI_READ_3W(unsigned char R_Adr, unsigned char R_Byte)
{
    int      ReadCMD ;                   // SPI READ COMMAND
    int i ;                               // counter
    int j ;
    unsigned char  SendBit ;

    // make SPI READ COMMAND
    ReadCMD = ((int)R_Adr << 1) + 1 ;
    SEND_BUFF[0] = (unsigned char)ReadCMD ; // lower 8bit
    SEND_BUFF[1] = (unsigned char)(ReadCMD >> 8) ; // higher 8bit

    i = 0 ;                               // Byte Counter

    // initialize SPI bus
    p3_5 = 1 ;                             // SS  = H
    p3_4 = 1 ;                             // SCLK = H

    wait_nop() ;

    // SPI READ COMMAND send start
    pd3_3 = 1 ;                             // set port direction MOSI(output)
    p3_3 = 1 ;                             // MOSI = H
    p3_5 = 0 ;                             // SS  = L
    wait_nop() ;
    while( i < 2 ){                         // 2 byte loop
        j = 1 ;                             // bit mask
        while(j < 0xFF) {                   // 8bit send
            SendBit = SEND_BUFF[i] & j ? 1 : 0 ; // Bit Test
            j = j << 1 ;                     // j *= 2 ;
            p3_4 = 0 ;                       // SCLK = L
            p3_3 = SendBit ;                 // output LSB first bit data on MOSI
            wait_nop() ;
            p3_4 = 1 ;                       // SCLK = H(latch)
        }
    }
}

```

```

        i ++ ;                                // next byte
        wait_nop() ;
    }

    // SPI READ start
    pd3_3 = 0 ;                                // set port direction MISO(input)

    wait_nop() ;
    wait_nop() ;
    wait_nop() ;
    wait_nop() ;
    i = 0 ;
    while( i < R_Byte ){                        // Loop R_Byte times
        j = 0 ;                                // reset bit counter
        READ_BUFF[i] = 0x00 ;                  // clear buffer
        while(j < 8){                          // 8bit receive
            p3_4 = 0 ;                          // SCLK = L(Data Change)
            wait_nop() ;
            wait_nop() ;
            p3_4 = 1 ;                          // SCLK = H(latch)
            // LSB first receive from MISO
            READ_BUFF[i] += (unsigned char)p3_3 << j ;
            j ++ ;                              // Next Bit
        }
        i ++ ;                                // Next Byte
    }
    p3_5 = 1 ;                                // SS = H(END)
    return TRUE ;
}

//-----
// msec wait
//-----
void    wait_ms(unsigned int wms)
{
    unsigned int w_cnt ;

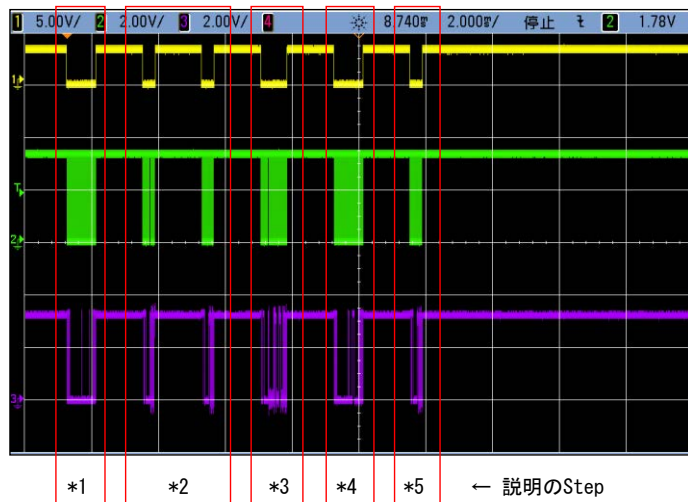
    while ( wms--){
        w_cnt = CLKMS ;
        while(w_cnt--){}
    }
}

void    wait_nop(void)
{
    asm( "nop" ) ;
    asm( "nop" ) ;
    asm( "nop" ) ;
    asm( "nop" ) ;
    asm( "nop" ) ;
    asm( "nop" ) ;
}

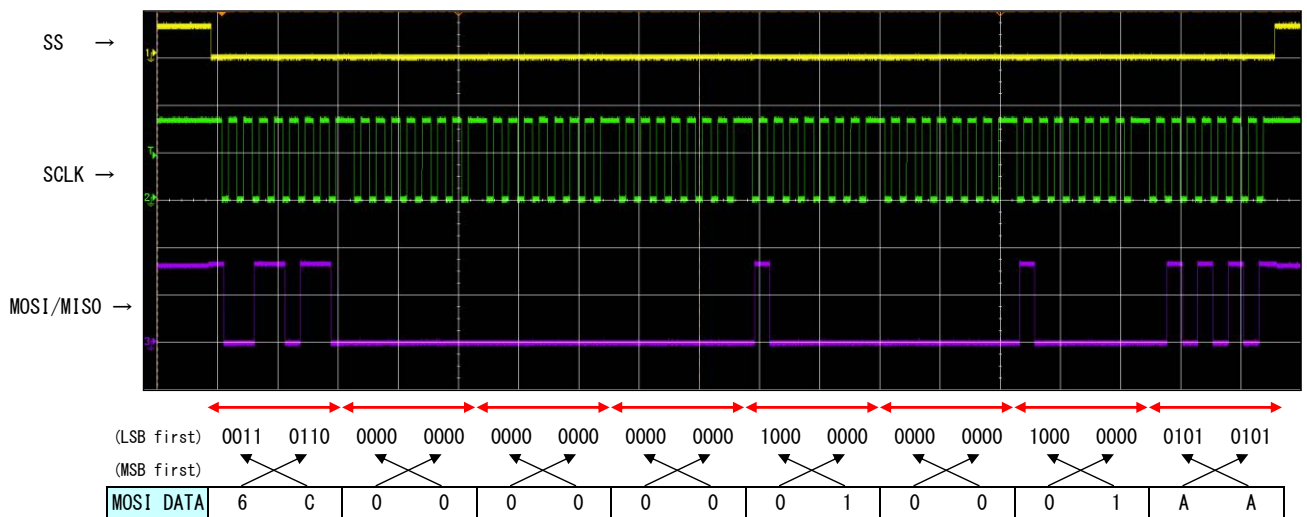
```

(2) サンプルの実行例 (実波形)

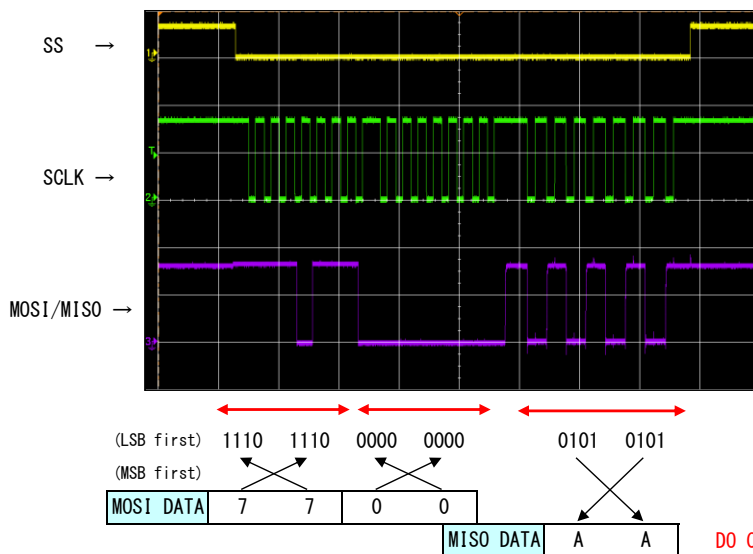
例) AGCモードを、“MID”から“HI”に切換



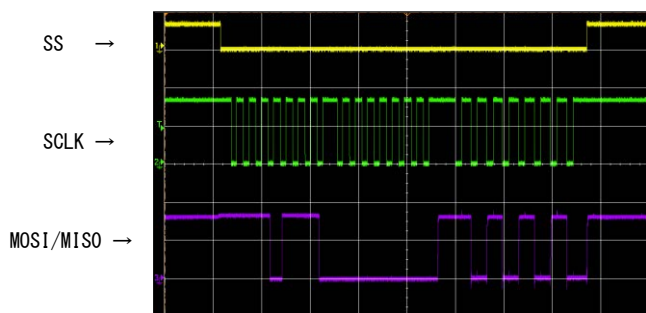
Step.1 “CAMERA PARAMETER READ1”コマンドを使用してインダイレクトレジスタ（アドレス：0x0036-0x003B）に
“AGC MODE”（パラメータアドレス：0x401）のパラメータを読み出します。
S P I 書込発行時の実波形は下図のようになります。



Step.2 S P I 読込で、ステータスコード（アドレス：0x003B）をチェックします。

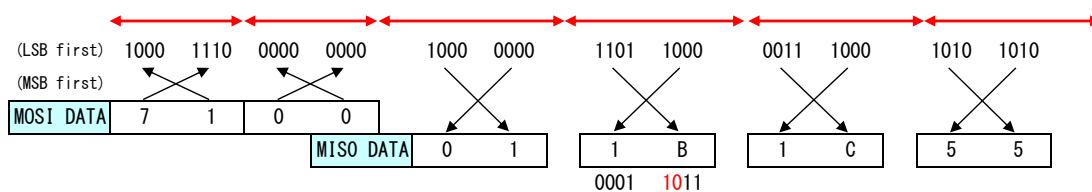
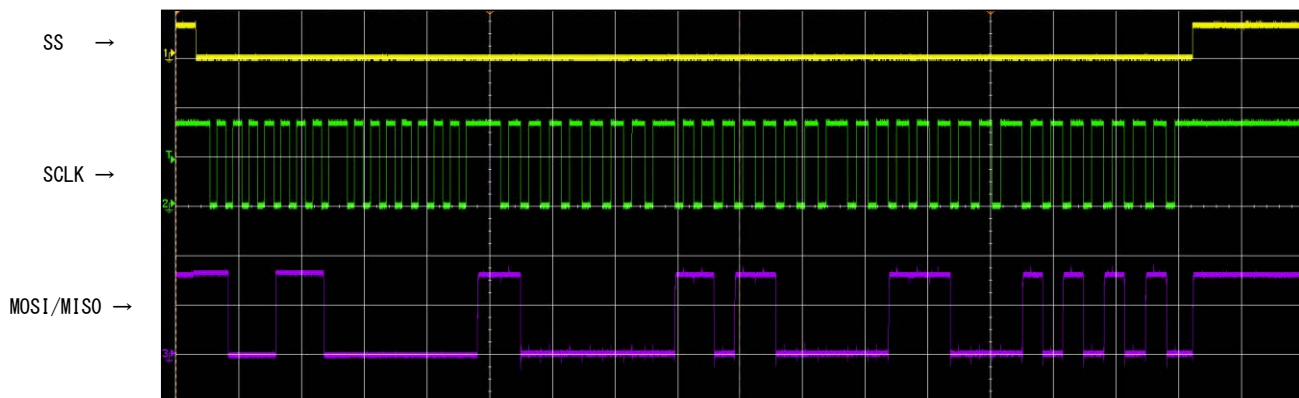


この例では、ステータスコードが“0xAA”となっています。
これはカメラDSPがビジー状態、またはコマンド
未実行状態であることを表しています。
この例では約1 (ms) ウェイト後、
再度、ステータスコード読みだしを行います。



← リトライ後、正常終了コード"0x55"が読み出された。

Step.3 S P I 読込を行いインダイレクトレジスタから、“AGC MODE”（パラメータアドレス：0x401）のパラメータを読み出します。



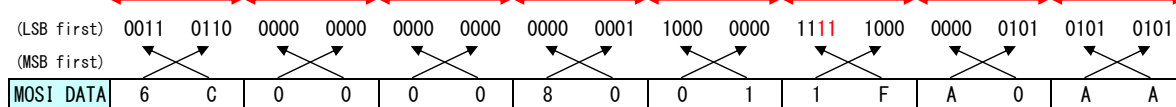
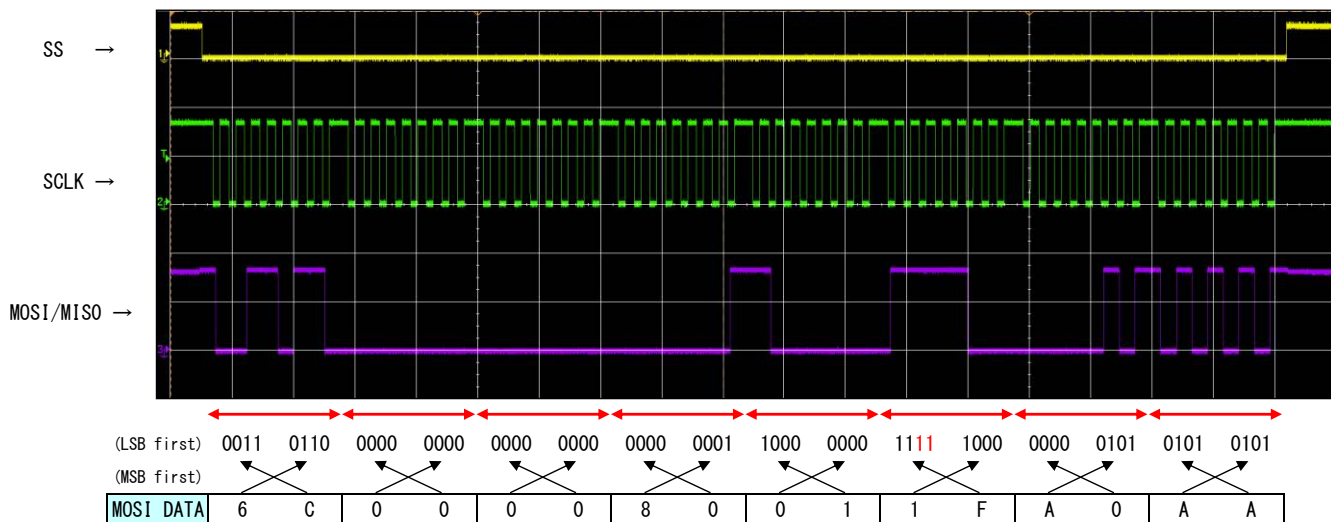
“AGC MODE”パラメータは、“Bit 4-3”に割り当てられています。2ビット分のデータは“0x02”であることから現在のカメラ設定は、“AGC MODE = MID”であることが分かります。

“AGC MODE = HI”に設定するために、Bit 4 に1、Bit 3 に1をセットします。

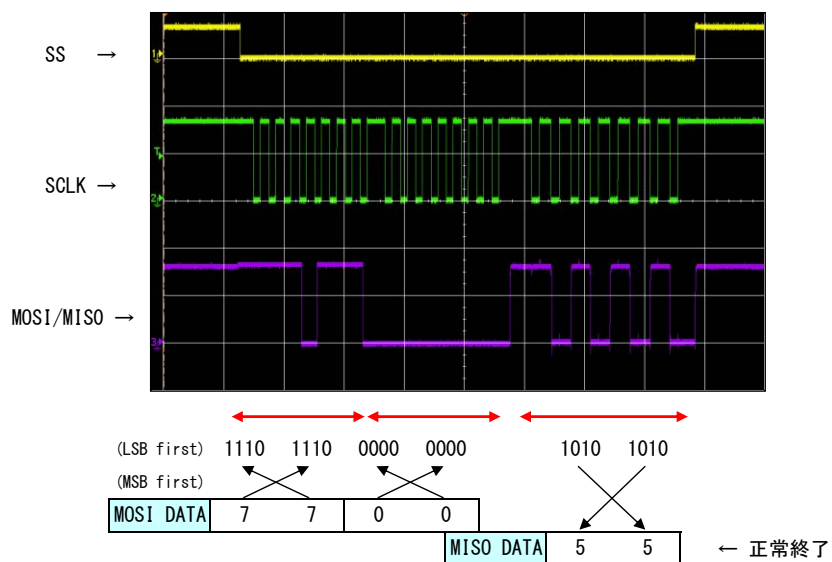
この結果、カメラに設定すべきデータは、(0x1F=00011111)となります。

Step.4 “CAMERA PARAMETER WRITE1”コマンドを使用してインダイレクトレジスタ（アドレス：0x0036-0x003B）に“AGC MODE”（パラメータアドレス：0x401）のパラメータを書き込みます。

S P I 書込発行時の実波形は下図のようになります。



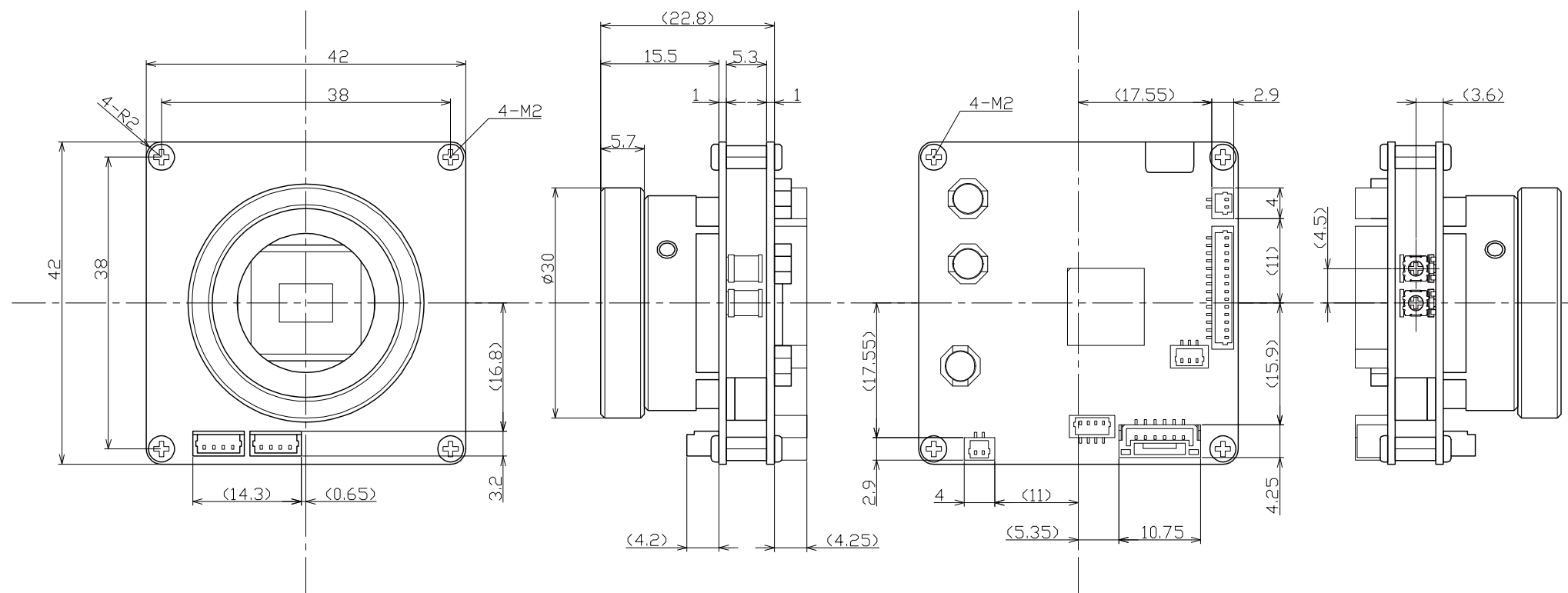
Step.5 S P I 読込を行い、ステータスコード（アドレス：0x003B）をチェックします。



7. 仕様

製品名		WAT-910BD (EIA)	WAT-910BD (CCIR)
撮像素子		1 / 2 インチ型インターライン転送CCD固体撮像素子	
総画素数		811 (H) × 508 (V)	795 (H) × 596 (V)
有効画素数		768 (H) × 494 (V)	752 (H) × 582 (V)
セルサイズ		8.4μm (H) × 9.8μm (V)	8.6μm (H) × 8.3μm (V)
同期方式		内部同期	
操作方式		2 : 1 インタレース	
映像出力		コンポジットビデオ, 1.0 V(p-p), 75Ω (不平衡)	
解像度 (水平)		570 TV本以上 (画面中央)	
最低被写体照度		0.0000025lx, F1.4 (Shutter: x256, AGC: HIGH, γ : 0.35)	
S/N		52dB以上 (AGC OFF=6dB, γ=1.0)	
機能設定		5キーリモコンによるOSD (オンスクリーンディスプレイ) 操作	
シャッター	固定	低速シャッター	x256, x128, x64, x32, x16, x8, x4, x2(field)
		高速シャッター	OFF: 1/60 (EIA), 1/50 (CCIR), FL: 1/100 (EIA), 1/120 (CCIR), 1/250, 1/500, 1/1000, 1/2000, 1/5000, 1/10000, 1/100000
	電子アイリス	1/60 (EIA), 1/50 (CCIR) - 1/100000 (sec), x256 (フィールド, 初期設定 x8) - 1/100000 (sec)	
AGC	ON	L0: 6-30dB / MID: 6-34.5dB / HI: 6-41dB	
	OFF	6-41dB (1dB step) selectable	
ガンマ補正		0.05 - 0.45 - 1.0 (0.05 step) / USR (0.45 S字カーブ)	
ノイズリダクション		3 DNR ON / OFF (レベル選択可能, 初期設定 ON:50)	
ダイナミックレンジ補正		D-WDR (Digital Wide Dynamic Range) OFF / USER1 / USER2 (USER設定時、特性変更可能)	
動体検出		ON / OFF 最大4領域指定可能 (検出結果出力端子有り)	
通信制御		3線式SPI通信	
デジタルビデオ出力		ON / OFF BT.656 準拠, 8bit デジタルビデオ出力端子 (3.3V CMOS レベル) ピクセルクロック: 28.6363MHz (EIA), 28.375MHz (CCIR)	
逆光補正		OFF / BLC / HSBLG	
白点補正		最大64点補正可能	
シャープネス		32段階から選択可能 (初期設定 4)	
セットアップレベル		OIRE / 7.5IRE	OIRE
静止画		ON / OFF	
画角反転機能		OFF / H-FLIP (水平) / V-FLIP (垂直) / HV-FLIP (水平・垂直)	
ズーム		ON / OFF ズーム倍率 (x1.0~x32.0), ズーム位置選択可能	
ネガポジ反転		ON / OFF	
レンズマウント		CS	
レンズアイリス		Video / DC	
電源電圧		DC+12V±10%	
消費電力		1.32W (110mA)	
動作温度		- 10°C ~ + 50°C	
保管温度		- 30°C ~ + 70°C	
動作/保管湿度		95%RH 以下 (ただし、結露無きこと)	
重量		約 40g	

8. 詳細寸法



単位: (mm)