

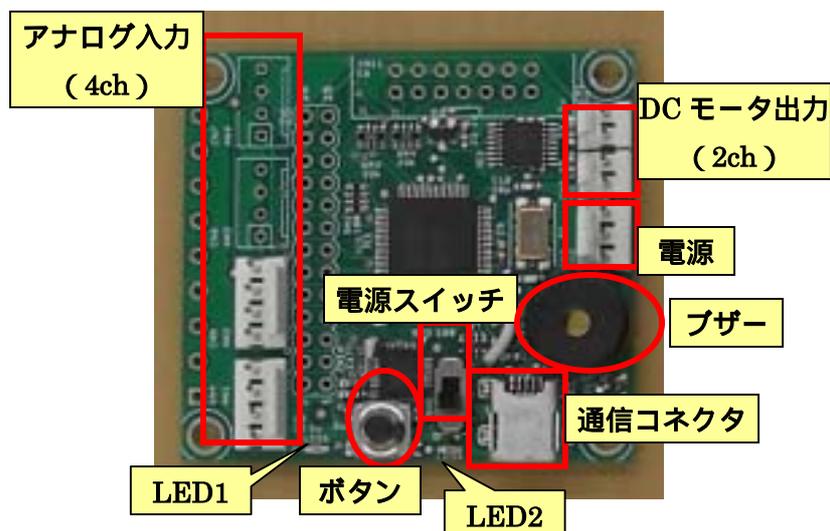
VS-WRC003 テクニカルマニュアル

ガイストン株式会社

本説明書は、ロボットプロセッサ「VS-WRC003」に関する通信仕様やメモリマップ及び拡張機能について説明した技術資料です。VS-WRC003 は、「Beauto Chaser」にも搭載されているロボット制御用のマイコンボードです。

・ 本体説明

VS-WRC003 の本体の各部名称及び概要について説明します。VS-WRC003 に備わっている入出力ポート・及びその他各種インタフェースは下記の通りです。



・ VS-WRC003 の基本操作の説明

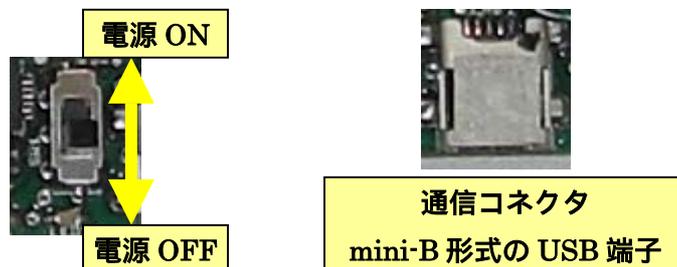
VS-WRC003 本体の基本操作や、各部位の概要について説明します。

プログラム (シーケンス) の実行

VS-WRC003 は専用ソフトウェア「BeautoBuilderNEO」より動作シーケンスを作成して記録させることができます。書き込んだシーケンスは、本体を起動してボタンを押すことで開始します。また、シーケンスの実行中にボタンを押すと、もう一度最初からシーケンスを実行します。BeautoBuilderNEO の取り扱いについては、「BeautoBuilderNEO 取扱説明書.pdf」をご参照ください。

通信コネクタ・電源スイッチの仕様

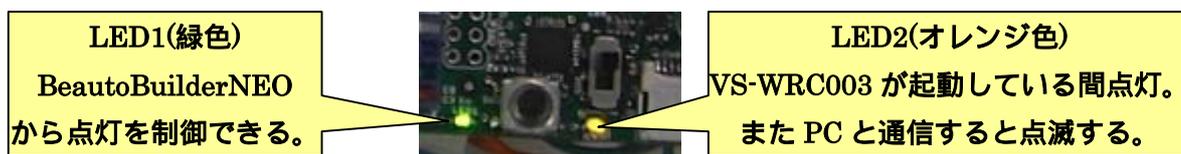
VS-WRC003 の電源スイッチは、レバーを下にスライドさせると OFF、上にスライドさせると ON になります。



通信コネクタは、mini-B の USB 端子となっており、付属の通信ケーブル以外に市販の USB ケーブルを接続することができます。また、VS-WRC003 は USB からの供給電力で動作するため、PC と通信ケーブルで接続すると VS-WRC003 が起動します。ただし、DC モータを動かす場合は VS-WRC003 本体の電源スイッチを ON にする必要があります。

LED1,LED2 の仕様

VS-WRC003 には LED1,LED2 という二つの LED が備わっています。LED1 は BeautoBuilderNEO のプログラムやメモリマップの書き換えによって点灯・点滅させることができ、緑色に光ります。LED2 は VS-WRC003 が起動している場合にオレンジ色に光り、また PC と通信している間は LED が点滅します。



・ コネクタ仕様説明

VS-WRC003 に備わっているコネクタについて、各ピンの仕様や取り扱い注意事項を説明します。VS-WRC003 に DC モータやセンサなど拡張部品を接続する場合は、下記の説明に従って正しく部品を準備・接続してください。なお、各項目の後のカッコ内は、基板上に印刷されているコネクタの名称を表しています。

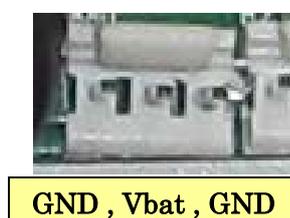
DC モータ (CN1,2 / M1,2)

VS-WRC003 より制御する DC モータを接続するためのコネクタです。1つのコネクタには2つのピンがあり、それぞれのピンに DC モータのケーブルを接続して制御します。



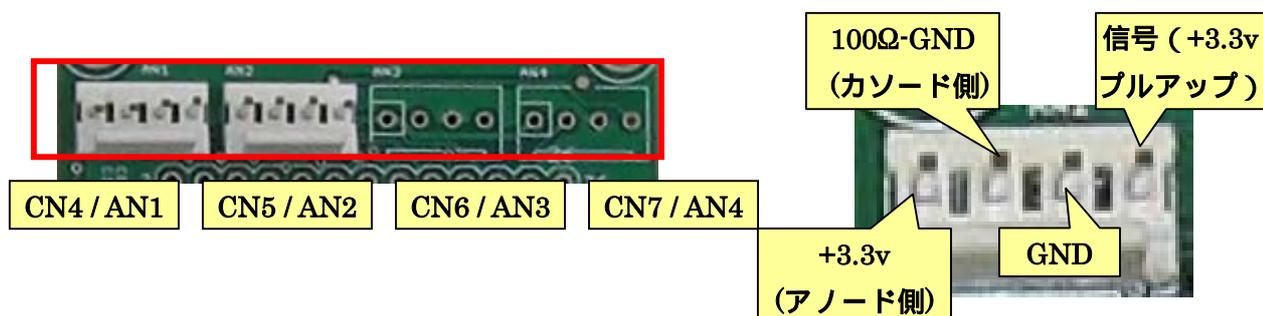
電源 (CN3 / POW)

VS-WRC003 に駆動用の電源を供給するためのコネクタです。VS-WRC003 の対応電圧は 4V ~ 9V です。規定の範囲外の電圧を供給すると故障などの不具合が発生する恐れがあるのでおやめください。ピンの概要は、中央が Vbat、両端が GND となっております。



アナログ入力 (CN4,5,6,7 / AN1,2,3,4)

VS-WRC003 にセンサなどのアナログ入力デバイスを接続するためのコネクタです。BeutoBuilderNEO 上のセンサ 1~4 は、下記の CN4~7 の順で対応しています。また、お掃除機能付き教材ロボット「Beuto」や VS-WRC001 に対応したセンサデバイスもそのまま接続できます。1つのコネクタには4つのピンがあり、それぞれのピンの機能は下写真の通りです。CN3,CN4 には別途コネクタをはんだ付けする必要があります。自作のアナログ入力デバイスを接続する場合は、各ピンの仕様に従って作成してください。



集合 IO (CN10 / EXT)

VS-WRC003 に搭載されたマイコン「H8/36064G」の各 I/O ポートに相当するコネクタです。対応独自にファームウェアを開発したり、改造を施したりする場合にお使いいただけます。また、デフォルトのファームウェアの場合、PB4～PB7 のピンを、それぞれアナログ入力の 5～8 (Ain_DATA5～8) として利用できます。なお、コネクタを使用する際は、別途ピンヘッダを用意してはんだ付けを行なってください。コネクタの各ピンの仕様は下記の通りです。

PB6 , PB4 , P71 , P74 , P66 , P62 , P56 , P54 , P52 , P50 , NMI , Vbat , GND

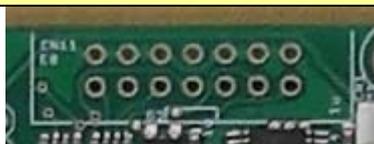


PB7 , PB5 , P70 , P72 , P67 , P63 , P57 , P55 , P53 , P51 , RES , -- , +3.3v

E8 デバッグ端子 (E8)

Renesas Technology 社の E8 デバッグを接続するためのコネクタです。別売りの E8 デバッグ接続用のコネクタをお買い求めいただき、はんだ付けすることで、E8 デバッグを接続できるようになります。コネクタの各ピンの仕様は下記の通りです。

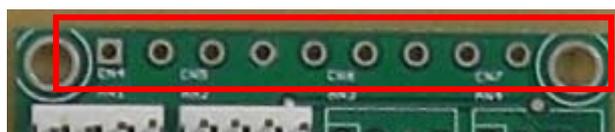
GND , GND , GND , +3.3v , GND , GND , GND



P87 , -- , P86 , NMI , -- , P85 , RES

ゲームパッド接続コネクタ

市販ゲーム機用のパッドを接続するためのコネクタです。別売りのゲームパッド接続コネクタをはんだ付けしてゲーム機用のパッドを接続すると、メモリマップよりボタンやアナログスティックの情報を取得できるようになります。



ゲームパッド接続コネクタ

・ 通信仕様・プロトコルの説明

VS-WRC003 は PC と USB ポートで接続しますが、基板上に実際の通信はシリアル通信で行なわれるため、RS232C のようにシリアルポートを用いた通信制御のプログラムで操作することが可能です。具体的な通信仕様や通信に用いるプロトコルについて、下記に説明します。

・ 通信ポート設定

VS-WRC003 と通信を行なう際のシリアルポートの設定は下記の通りです。

ビットレート：115200bps

データビット：8

パリティ：偶数

ストップビット：1

フロー制御：なし

・ 通信プロトコル

VS-WRC003 はモータやセンサなどの制御情報を記録したメモリマップを持っており、PC から通信制御でメモリマップを読み書きすることで、センサ情報を取得したりモータを任意の方向・速度で回してロボットを動かしたりすることができます。なお、メモリマップの詳細につきましては、後述の「メモリマップについて」をご参照ください。

メモリマップを書き換えるための通信メッセージのプロトコルは下記の通りになります。メッセージは基本的にキャラクター（テキスト）を使用するため、実際のプログラムにおいても通信メッセージを文字列で作成するなどしてください。

プロンプトとエコーバックについて

VS-WRC003 が受信したメッセージを処理した後に、必ず返信のメッセージとして「>」という記号を返します。これは VS-WRC003 との通信におけるプロンプトになります。VS-WRC003 との通信処理を作成する際は、この記号の受信によって同期を取るようにしてください。また、PC から書き込みメッセージの送信中は、確認のため VS-WRC003 より文字をエコーバックで送信します。

エラーメッセージについて

VS-WRC003 に送信したメッセージ内容に誤りがあったり、メッセージの送受信自体に失敗したりして、VS-WRC003 が正しくメッセージを処理できなかった場合、VS-WRC003 はエラーを表す文字列「ERROR」を送信します。この文字列を受信した場合は、改めて送信内容や通信状態を確認し、正しいメッセージを送信してください。

メモリマップへの書き込み

VS-WRC003 のメモリマップへ数値を書き込む場合は、「w (書き込み開始アドレス:16 進数) (書き込みデータ:16 進数)¥n」という表記になります。メッセージを送信すると、「書き込み開始アドレス」に指定した場所から、「書き込みデータ」に記述された数値を連続的に書き込みます。

「w」、「書き込み開始アドレス」、「書き込みデータ」の間には半角スペースを一つ入れてください。「書き込み開始アドレス」、「書き込みデータ」は 16 進数で表記し、「書き込みデータ」の末尾に連続して改行コード「¥n」を追記してください。また、「書き込みデータ」は必ず書き込み先のメモリのサイズに従い、偶数の桁数を記述してください。書き込みデータのサイズ上限は 40byte なので、書き込みデータは「w」から含めて 40 文字以内にしてください。

例：ブザーから音を鳴らす場合（ブザータイマーのアドレス'0x2c'に'0x40'を書き込み）

'w 2c 40¥n'

書き込み開始アドレスと書き込みデータは 16 進数で表記し、必ず偶数の桁数（1byte 区切り）にします。

例：すべての DC モータにブレーキをかける場合（モータスピードのアドレス'0x09'に 2ch 分=2byte '0x80'を書き込む）

'w 9 8080 ¥n'

書き込みデータは、メッセージに記述した分量だけ指定した書き込み開始アドレスより連続してメモリマップに書き込まれます。

メモリマップからの読み込み

VS-WRC003 のメモリマップから数値を読み込む場合は、「r (読み込み開始アドレス) (読み込みデータサイズ)¥n」という表記になります。メッセージを送信すると、「読み込み開始アドレス」に指定した場所から、「読み込みデータサイズ」に記述された数値を連続的に読み込み返信します。

基本的な書式はメモリマップの書き込みと同じです。「r」、「読み込み開始アドレス」、「読み込みデータサイズ」の間に一つ半角スペースを入れる、「読み込み開始アドレス」、「読み込みデータサイズ」は 16 進数で表記する、「読み込みデータサイズ」の末尾に連続して改行コード「¥n」を追記するなどの規定があります。

PC よりメッセージを送信すると、VS-WRC003 より指定したアドレスから指定したサイズだけデータが連続して送信されます。データの送信が完了すると、プロンプトを表す「>」を送信します。

例：すべてのアナログ入力の情報を取得

‘r 17 8¥n’

アナログセンサのメモリマップはアドレスが 0x17 で、2byte 分のデータが 4 個続くので、2 × 4 = 8 を読み込みデータサイズに記述します。

メモリマップのダンプ表示

VS-WRC003 には、メモリマップの内容を一度に全て受信するダンプ機能があります。多くのメモリ情報を読み込むときは、前述の読み込みコマンドで目的のアドレスから個別に読み出すより、ダンプ機能で一度に全ての情報を受信したほうが簡潔な処理になります。

メモリマップのダンプを行う場合は、「d m¥n」を PC より送信します。「d」と「m」の間には一つ空白（半角スペース）を入れてください。

PC よりメッセージを送信すると、VS-WRC003 よりメモリマップのすべてのデータが連続して送信されます。データの送信が完了すると、プロンプトを表す「>」を送信します。

メモリマップの ROM への書き込み

PC よりメモリマップへ書き込んだ数値は RAM へ記録されるため、VS-WRC003 の電源を切ると値がリセットされます。メモリマップに書き込んだ値を保持する場合は、VS-WRC003 の ROM へデータを書き写します。これを「フラッシュ」といいます。例えばモータゲインなど、ロボットの起動時に自分で設定した値に合わせておきたい場合は、メ

モリマップのモータゲインに値を書き込んでフラッシュを実行することで、次にロボットを起動したときは、はじめからモータゲインを任意の値に合わせておくことができます。

VS-WRC003 のフラッシュ機能を使用する場合は、PC より「f m n」を送信します。「f」と「m」の間には一つ空白（半角スペース）を入れてください。なお、ROM への書き写しには若干時間がかかるため、書き込み処理が完了し VS-WRC003 よりプロンプトが送信されるまで PC からメッセージを送信せず待機するようにしてください。また、アナログセンサの現在値など一部の値は ROM に記録されないためご注意ください。

・ メモリマップについて

VS-WRC003 のメモリマップは下記の通りです。なお、メモリマップの内容を書き換える場合は、各機能についてよく確認し本体の暴走やそれに伴う破損などが発生しないように十分ご注意ください。

アドレス	名前	バイト数	R/W	備考	初期値
0x00	Product_ID	2	R	ファームウェアの作成日	-
0x02	Version	2	R	ファームウェアのバージョン情報	-
0x04	MODE	1	R/W	VS-WRC003 の動作モード 0~2(0x00~0x02) 0:モータ電源 OFF 1:モータ、サーボ電源 ON 2: プログラム(シーケンス)再生モード 2を書き込むとプログラム開始、 1を書き込むと停止	0
0x06	MTR_ON_TIME	1	R/W	モータ起動時間(1カウント:1/60sec) 0~255(0x00~0xFF)	28(0x1C)
0x07	MTR_PWM_CYC	2	R/W	モータ制御のPWMの周期(usec単位) 0~65535(0x0000~0xFFFF)	16160(0x3F20)
0x09	MTR_1	1	R/W	DCモータの各chの速度 0(0x00):脱力 -128(0x80):ブレーキ 1~127(0x01~0x7F):正転 -1~-127(0xFF~0x81):逆転	0
0x0A	MTR_2	1	R/W	速度は可変 この数値を書き換えるとMTR_ON_TIMEの 時間(step)をかけて車輪の速度が変化する	0
0x0D	MTR_CW_GAIN_1	1	R/W	DCモータの各chのゲイン(正転時)	192(0xC0)
0x0E	MTR_CW_GAIN_2	1	R/W	0~255(0x00~0xFF)	192(0xC0)
0x11	MTR_CCW_GAIN_1	1	R/W	DCモータの各chのゲイン(逆転時)	192(0xC0)
0x12	MTR_CCW_GAIN_2	1	R/W	0~255(0x00~0xFF)	192(0xC0)
0x17	Ain_DATA1	2	R	アナログ入力の数値(数値は12bit)	-
0x19	Ain_DATA2	2	R	0~4096(0x0000~0xFFFF)	-
0x1B	Ain_DATA3	2	R	Ain_DATA3~4はCN3,CN4に、	-
0x1D	Ain_DATA4	2	R	Ain_DATA5~8はEXT(CN10)のPB4~PB7に	-

0x1F	Ain_DATA5	2	R	ピンヘッダのはんだ付けが必要	-
0x21	Ain_DATA6	2	R		-
0x23	Ain_DATA7	2	R		-
0x25	Ain_DATA8	2	R		-
0x27	SWin	1	R	本体のボタン入力の状態 0=OFF 1=ON	-
0x28	LED_CYCLE	1	R/W	LED1(緑色)を点滅させる周期 時間は 0.1 秒単位で、LED_TIME が 1 以上の 場合、書き込まれた時間ごとに点灯・消灯を 繰り返す	30(0x1E)
0x29	LED_TIME	1	R/W	LED1(緑色)を点灯させる時間。 数値を書き込むと 0.1 秒単位で自動的に 減算し、数値が 0 になるまでの間まで、 LED_CYCLE の設定に従って LED を点滅させる	-
0x2A	BZR_VOL	1	R/W	ブザーの音量(パルス幅) 0 ~ 255(0x00 ~ 0xFF)	128(0x80)
0x2B	BZR_PITCH	1	R/W	ブザーの音程(周期) 0 ~ 255(0x00 ~ 0xFF)	60(0x3C)
0x2C	BZR_TIME	1	R/W	ブザーの発音時間 数値を書き込むと 0.1 秒単位で自動的に減算 し、数値が 0 になるまでの間ブザーを発音する	0
0x2D	PS_DATA0	1	R	ゲームパッドのボタン入力 1 ボタン一つの入力が各ビットに相当 0=OFF 1=ON 下位より 0~3 ビット目が十字ボタンの左・下・ 右・上、4~7 ビットが START・R3・L3・SELECT	
0x2E	PS_DATA1	1	R	ゲームパッドのボタン入力 1 下位より 0~3 ビット目が ・x・ ・、 4~7 ビットが R1・L1・R2・L2	-
0x2F	PS_AN0	1	R	ゲームパッドのアナログスティック入力 右アナログスティックの X 方向の傾き 0 ~ 255(0x00 ~ 0xFF) 128 が中心、0 が左端、255 が右端	-
0x30	PS_AN1	1	R	ゲームパッドのアナログスティック入力 右アナログスティックの Y 方向の傾き	-

				0 ~ 255 (0x00 ~ 0xFF) 128 が中心、0 が上端、255 が下端	
0x31	PS_AN2	1	R	ゲームパッドのアナログスティック入力 左アナログスティックの X 方向の傾き 0 ~ 255 (0x00 ~ 0xFF)	-
0x32	PS_AN3	1	R	ゲームパッドのアナログスティック入力 左アナログスティックの Y 方向の傾き 0 ~ 255 (0x00 ~ 0xFF)	-
0x33	TIMER	1	R/W	汎用のタイマー	0x80
0x34	UNI_VAR	52	R/W	ユーザ変数 (ユーザ用フリーメモリエリア) 汎用のメモリエリアなので、自由に値を 読み書きして問題ない	すべて 0

メモリマップの書き換えにより VS-WRC003 が正しく動作しなくなった場合は、「VS-WRC003 の基本操作の説明」の「メモリマップの初期化」の説明に従ってメモリマップを初期化してください。

UNI_VAR より前のアドレスで説明未記入のものは、全てリザーブアドレスとなっております。こちらにはデータの読み書きを行なわないようにしてください。

・ お問い合わせ

ヴイストーン株式会社

〒555-0012 大阪市西淀川区御幣島 2-15-28

Tel:06-4808-8701 Fax:06-4808-8702

URL: <http://www.vstone.co.jp/>

e-mail: infodesk@vstone.co.jp

(2010.7.14)