

取扱説明書



TRIdent トライデントシリーズ

Basic V1.0

■ 概要

本製品は高効率・高応答性を実現した高性能モータードライバーボードです。

BLDC（ブラシレス DC）モーターおよび PMSM（Permanent Magnet Synchronous Motor, 永久磁石同期電動機）に対応し、FOC（フィールド指向制御）をサポートします。

■ 特徴

MCU

- ◆ STM32F405RGT6（ST マイクロエレクトロニクス社）
最大動作周波数 168MHz の ARM® Cortex®-M4 をベースとする高性能 MCU。

BLDC ドライバ

- ◆ DRV8316（Texas Instruments 社）
電流センシング機能と複数の FET 内蔵、最大 40V、ピーク 8A の 3 相モータードライバ。

角度センサー

- ◆ TLE5012B-E1000（Infineon 社）
GMR ベース、15 ビット絶対角度センサー。

インタフェース

- ◆ XT30PW(2+2)コネクタ（XT30 電源コネクタ + 2mm ピッチ CAN コネクタ）
- ◆ 1.25mm ピッチ 3 ピン UART コネクタ
- ◆ 1mm ピッチ 4 ピン ST-LINK コネクタ
- ◆ 1.27mm ピッチ 2 ピンソケット
（CAN120Ω ターミネータ用、ジャンパーキヤップ付属）
- ◆ NTC 温度センサー接続用ランド

指示灯

- ◆ 6×LED 指示灯
電源ランプ 1 つ（ブルー）、カスタマイズ可能な指示灯が 5 つ（ブルー×1、グリーン×1、レッド×2、イエロー×1）

目次

1.	製品紹介	3
2.	テクニカル仕様	3
3.	レイアウト図	4
3. 1.	表面	4
3. 2.	裏面	5
3. 3.	寸法	6
4.	セットアップガイドライン	7
5.	簡易回路図	8
6.	実装ガイドライン	9
6. 1.	簡易実装フロー	9
6. 2.	注意事項	14
6. 3.	FAQ	15

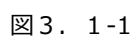
1. 製品紹介

本製品は高効率・高応答性を実現した高性能 BLDC モータードライバーボードです。FOC（フィールド指向制御）プログラムを実装することで、高トルク且つ高精度のモーター制御ができます。

2. テクニカル仕様

入力電圧：	24V
最大電圧：	40V
定格電流：	2A
最大電流：	8A
最大 PWM 周波数：	200kHz
最小動作温度：	-40 °C
最大動作温度：	+125 °C
通信：	CAN, UART
デバッグ IF：	ST-LINK
角度センサー：	GMR ベース, 15 ビット絶対角度
制御方式：	FOC / PWM
対応モーター：	BLDC / PMSM
サイズ：	43mm x 41mm

3. 1. 表面



3. 2. 裏面

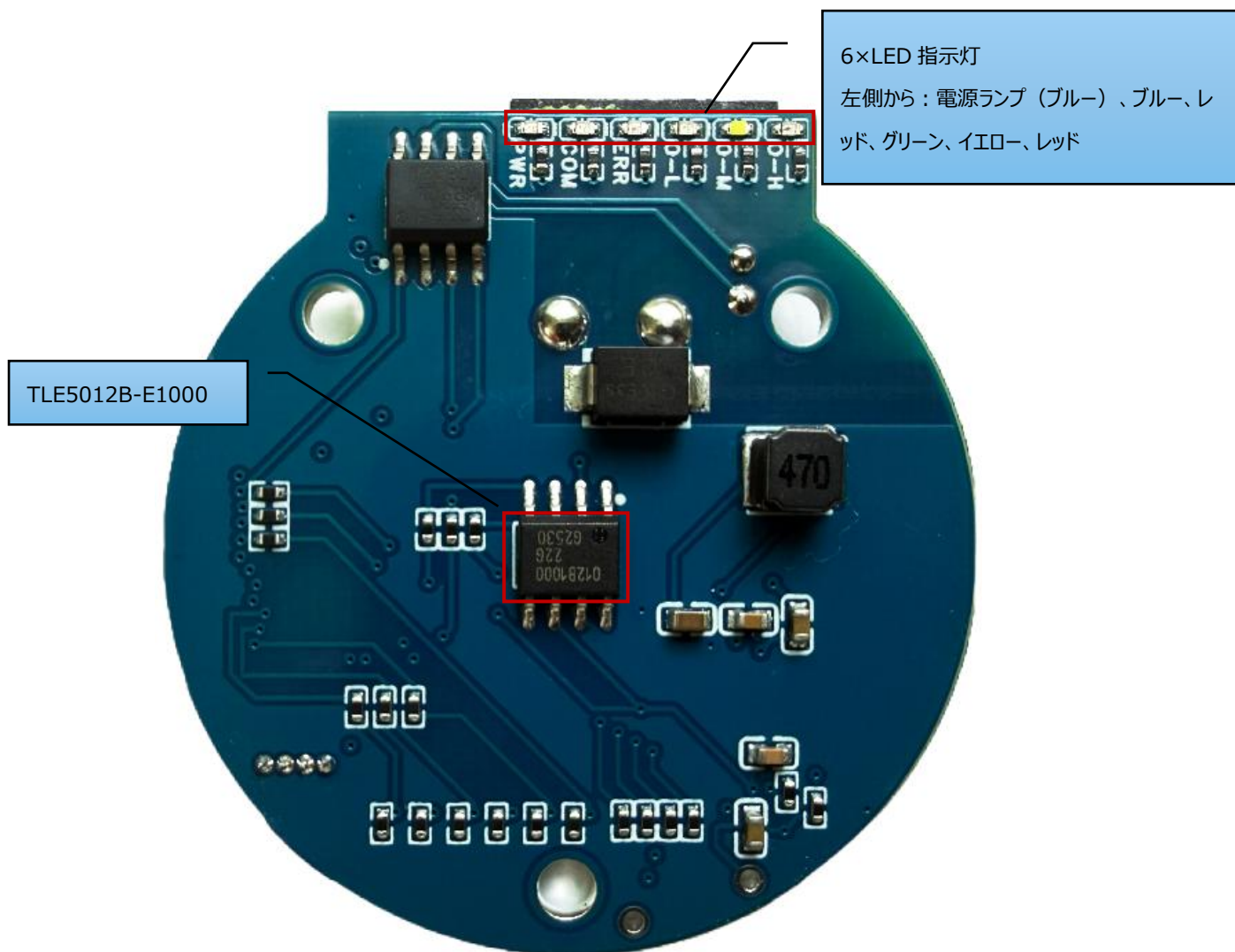


図3. 2-1

3. 3. 寸法

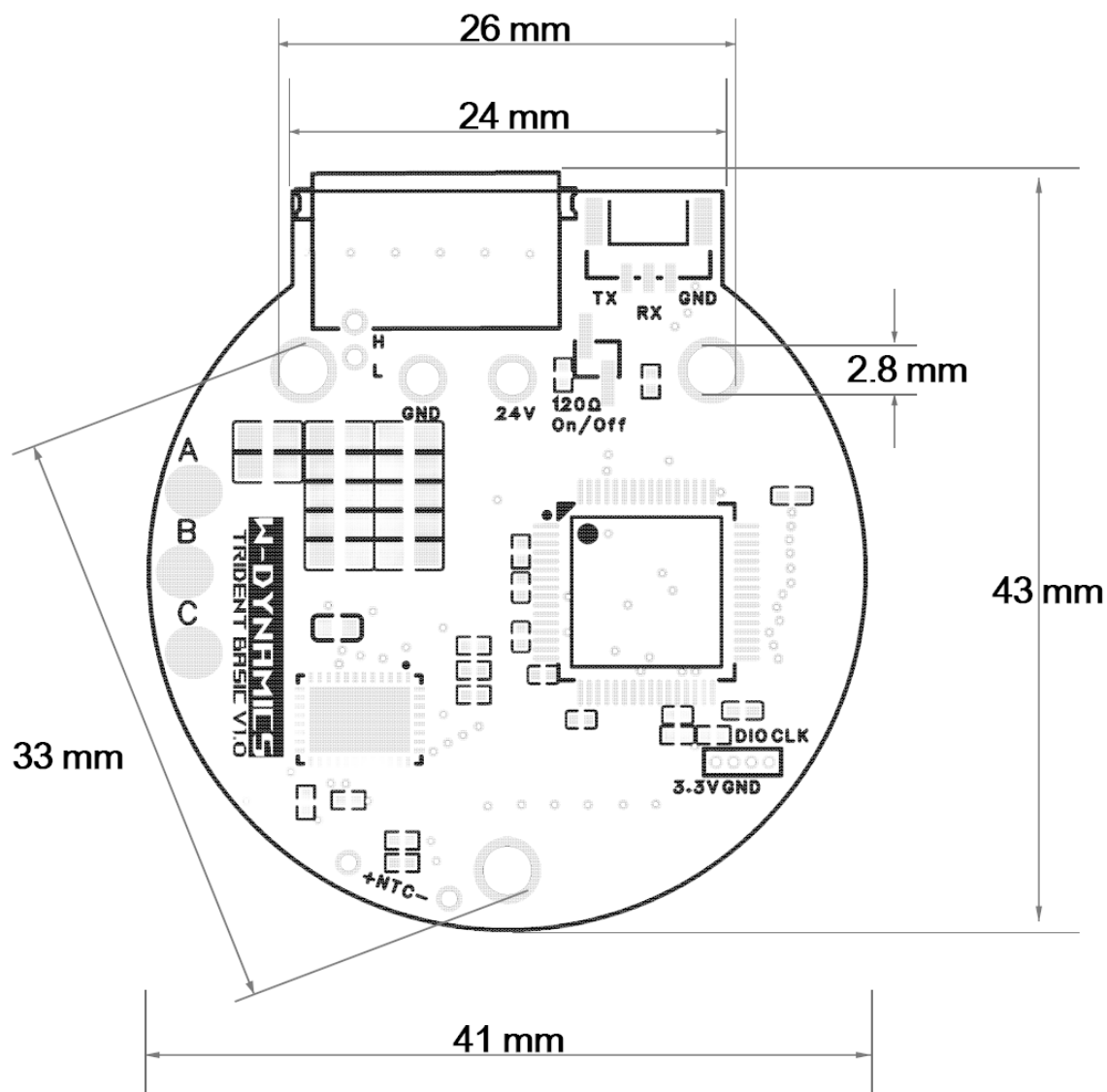


図3. 3-1

4. セットアップガイドライン

角度センサーTLE5012B-E1000 の中心点をモーターシャフトにあるマグネットの中心点に揃えるようにしてください。マグネットとの間隔は 1mm~2mm にしてください。

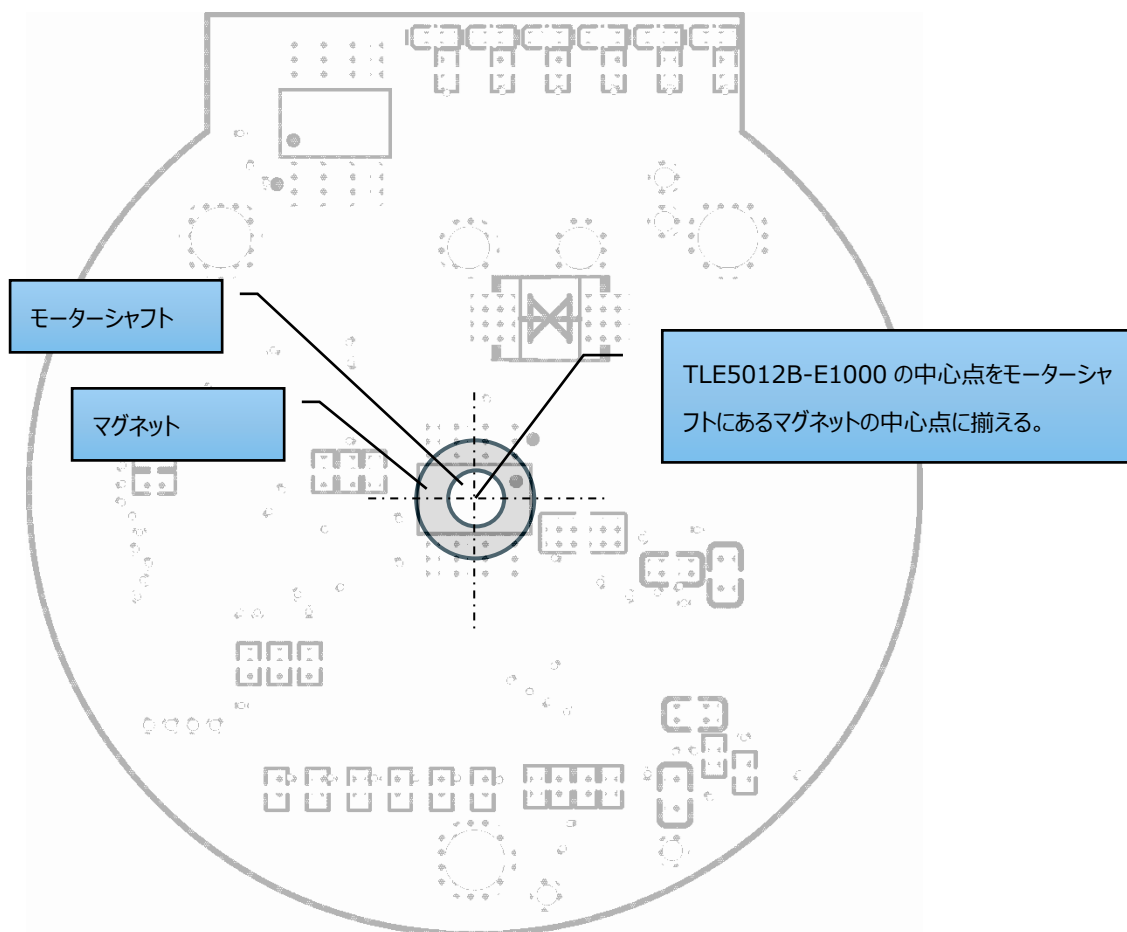


図 4-1

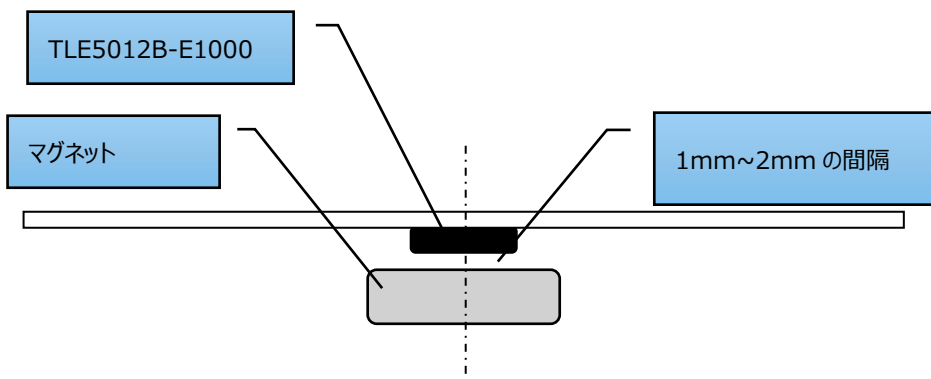


図 4-2

5. 簡易回路図

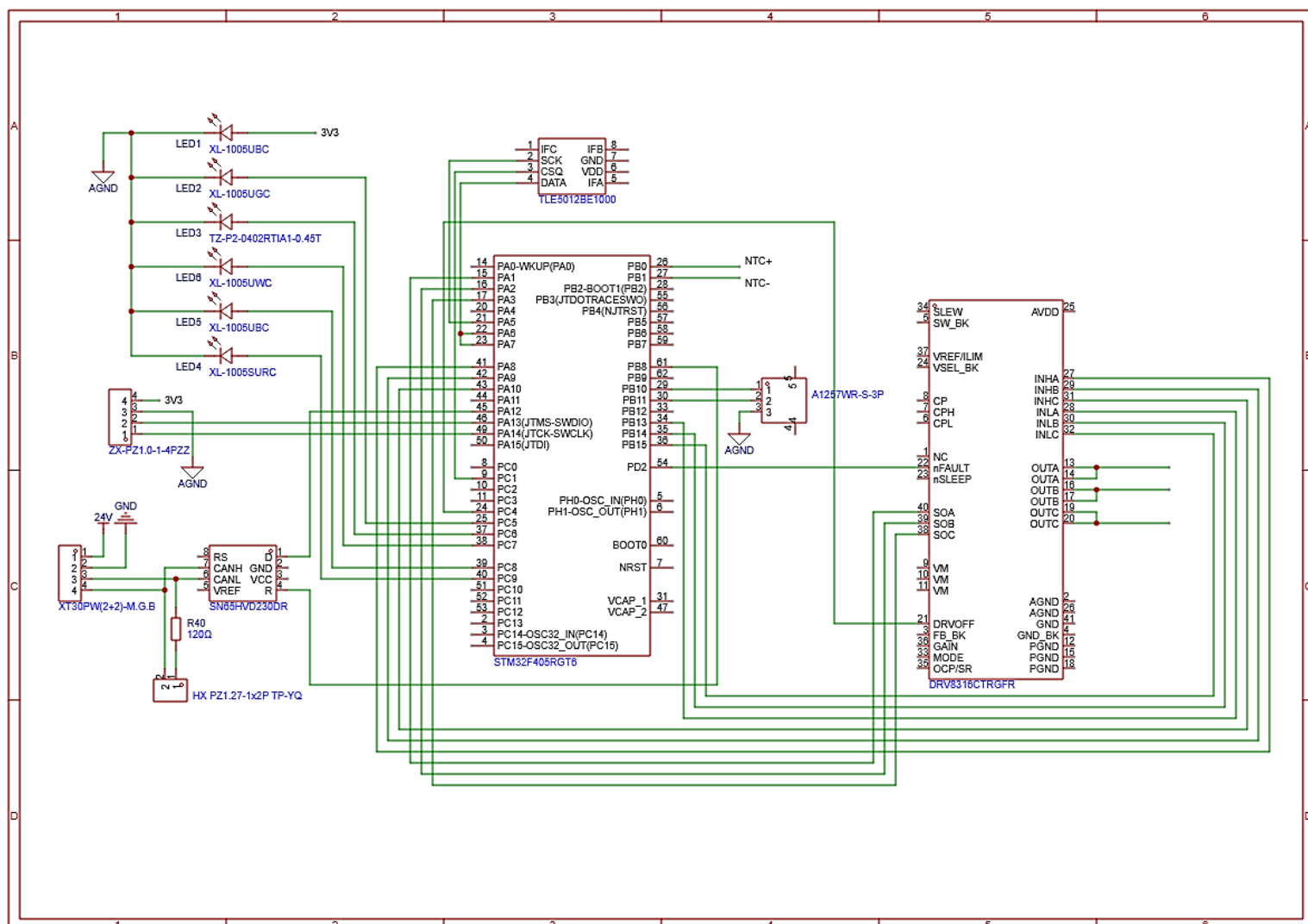


图 5-1

6. 実装ガイドライン

6. 1. 簡易実装フロー

1、ST-LINK/V2 デバッガーでドライバボードにある ST-LINK コネクタを PC へ接続する。

◆ST マイクロエレクトロニクス社の ST-LINK/V2 デバッガー：

<https://www.st.com/ja/development-tools/st-link-v2.html>

2、STM32CubeProgrammer で正常に STM32F405RGT6 へ接続することを確認する。

◆STM32CubeProgrammer のダウンロード URL：

<https://www.st.com/ja/development-tools/stm32cubeprog.html>



図 6. 1-1

接続できたら、以下の「ターゲット情報」及び「デバイスメモリ」が表示される。

ターゲット情報	
ボード	--
デバイス	STM32F405xx/F407xx/F415xx/F...
タイプ	MCU
デバイスID	0x413
リビジョンID	--
NVMサイズ	1 MB
CPU	Cortex-M4
ブートローダーバージョン	0x91

図 6. 1-2

デバイスメモリ					
ファイルを開く +					
アドレス	0x08000000	サイズ	0x400	データ幅	32ビット
				データの検索	0x
					読み込み
アドレス	0	4	8	C	ASCII
0x08000000	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	YYYYYYYYYYYYYY
0x08000010	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	YYYYYYYYYYYYYY
0x08000020	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	YYYYYYYYYYYYYY
0x08000030	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	YYYYYYYYYYYYYY
0x08000040	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	YYYYYYYYYYYYYY
0x08000050	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	YYYYYYYYYYYYYY
0x08000060	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	YYYYYYYYYYYYYY
0x08000070	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	YYYYYYYYYYYYYY
0x08000080	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	YYYYYYYYYYYYYY
0x08000090	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	YYYYYYYYYYYYYY

図 6. 1-3

3、STM32CubeMX で STM32F405RGT6 の初期化設定及びコードを生成する。

◆STM32CubeMX のダウンロード URL :

<https://www.st.com/ja/development-tools/stm32cubemx.html>

■推奨設定値 :

①PWM 周波数 : 20 kHz

SYSCLK : 72 MHz APB2 : 72 MHz

<-----サンプル初期化コード Start----->

```
htim1.Instance = TIM1;
htim1.Init.Prescaler = 0;
htim1.Init.CounterMode = TIM_COUNTERMODE_CENTERALIGNED1;
htim1.Init.Period = 1799;
htim1.Init.ClockDivision = TIM_CLOCKDIVISION_DIV1;
htim1.Init.RepetitionCounter = 0;
htim1.Init.AutoReloadPreload = TIM_AUTORELOAD_PRELOAD_DISABLE;
```

<-----サンプル初期化コード End----->

②CAN ボーレート : 1 Mbps

SYSCLK : 72 MHz APB1 : 36 MHz

<-----サンプル初期化コード Start----->

```
hcan1.Instance = CAN1;
hcan1.Init.Prescaler = 2;
hcan1.Init.Mode = CAN_MODE_NORMAL;
hcan1.Init.SyncJumpWidth = CAN_SJW_1TQ;
hcan1.Init.TimeSeg1 = CAN_BS1_15TQ;
hcan1.Init.TimeSeg2 = CAN_BS2_2TQ;
hcan1.Init.TimeTriggeredMode = DISABLE;
hcan1.Init.AutoBusOff = DISABLE;
hcan1.Init.AutoWakeUp = DISABLE;
hcan1.Init.AutoRetransmission = DISABLE;
hcan1.Init.ReceiveFifoLocked = DISABLE;
hcan1.Init.TransmitFifoPriority = DISABLE;
```

<-----サンプル初期化コード End----->

③UART ボーレート : 115200 bps

<-----サンプル初期化コード Start----->

```
huart3.Instance = USART3;  
huart3.Init.BaudRate = 115200;  
huart3.Init.WordLength = UART_WORDLENGTH_8B;  
huart3.Init.StopBits = UART_STOPBITS_1;  
huart3.Init.Parity = UART_PARITY_NONE;  
huart3.Init.Mode = UART_MODE_TX_RX;  
huart3.Init.HwFlowCtl = UART_HWCONTROL_NONE;  
huart3.Init.OverSampling = UART_OVERSAMPLING_16;
```

<-----サンプル初期化コード End----->

4、STM32CubeIDE でモーター制御ソースコードを実装する。

◆STM32CubeIDE のダウンロード URL :

<https://www.st.com/ja/development-tools/stm32cubeide.html>

■推奨実装順番：

Step1：オープンループ正弦波駆動（角度センサーを使用せず）

★以下は正弦波生成のサンプルコード：

<-----サンプルコード Start----->

// theta：電気角、M_PI：円周率

void SetThreePhasePWM(float theta)

```
{  
    // 3 相正弦波生成（120 度間隔）  
    float sin_a = sinf(theta);  
    float sin_b = sinf(theta - 2.0f*M_PI/3.0f);  
    float sin_c = sinf(theta + 2.0f*M_PI/3.0f);  
  
    // Duty 生成  
    float Ua = 0.5f + amplitude *sin_a;  
    float Ub = 0.5f + amplitude *sin_b;  
    float Uc = 0.5f + amplitude *sin_c;  
  
    // Duty 制限  
    if (Ua > 1) Ua = 1; if (Ua < 0) Ua = 0;  
    if (Ub > 1) Ub = 1; if (Ub < 0) Ub = 0;  
    if (Uc > 1) Uc = 1; if (Uc < 0) Uc = 0;  
  
    // Duty 出力。PWM_PERIOD：PWM パルス周期  
    __HAL_TIM_SET_COMPARE(&htim1, TIM_CHANNEL_1, (uint32_t)(Ua * PWM_PERIOD));  
    __HAL_TIM_SET_COMPARE(&htim1, TIM_CHANNEL_2, (uint32_t)(Ub * PWM_PERIOD));  
    __HAL_TIM_SET_COMPARE(&htim1, TIM_CHANNEL_3, (uint32_t)(Uc * PWM_PERIOD));  
}
```

<-----サンプルコード End----->

Step2 : 電流制御（角度センサー及び ADC を使用）★最重要★

ADC から取得した電流で Clark 変換を行う

↓

角度センサーから取得した電気角を加えて Park 変換を行う

↓

PI 制御

↓

逆 Park 変換

↓

3 相波形生成

Step3 : 速度制御

角度センサーから取得した速度を元に、電流制御をし、速度制御を行う。

Step4 : 位置制御

角度センサーから取得した位置を元に、速度、電流制御をし、位置制御を行う。

Step5 : ベクトル制御（FOC）

dq 軸の出力パラメータを最適化し、高トルク、高効率、スムーズな回転ができるモーター制御を実装。

5、オシロスコープでドライバボードのモーター接続用パッド A、B、C の出力波形を確認する。

6、モーターにセットアップし、直流安定化電源を用いて入力電流を制限した上テストを行う。直接バッテリーを使うと定格電流超過の可能性がある。

6. 2. 注意事項

- ★ 1 定格電流が 2A であるため、同様な定格電流のモーターを選定してください。
- ★ 2 モーターを本格稼働させる前に必ずテストを行ってください。
- ★ 3 モーター稼働時は周りの安全を注意してください。
- ★ 4 電源ケーブルを接続する際に、電源の極性を間違いないようにしてください。
- ★ 5 通電しながらの抜き差しをしないでください。
- ★ 6 ドライバボードの散熱を注意してください。

6. 3. FAQ

Q1. 入力電圧は 24V 固定ですか？

A1. 26V 以下でしたら問題ございません。例えば、リチウムイオン電池 6S を使う場合は満充電時の電圧が 25.2V になりますが、使用可能です。

Q2. 使用するモーターの瞬時最大電流が 6.5A であるが、使用不可ですか？

A2. 本製品の瞬時最大電流が 8A のため、使用可能です。ただし、定格電流 2A を満たす必要があります。

Q3. ドライバボードのモーターセットアップ用のホールは使用するモーターと合わないです。どうすれば良いですか？

A3. セットアップ用のフレームを 3D プリンターで作製してください。弊社はドライバボードの OEM も提供するので、お気軽にお問い合わせください。

◆問い合わせ用メールアドレス：sales@w-dynamics.co.jp

Q4. オシロスコープで出力波形を見ると想定通りの波形ですが、モーターは振動して回転しません。

A4. モーターの UVW 相の相順が違った可能性があります。任意の 2 相を入れ替えてみてください。

Q5. CAN120Ω ターミネータ用のジャンパーキャップは常にセットする必要がありますか？

A5. CAN ライン上 1 つのドライバボードのみが存在する場合は常にセットしてください。2 つ以上のドライバボードが存在する場合は、任意 1 つのドライバボードのジャンパーキャップをセットしてください。それ以外のドライバボードのジャンパーキャップを外してください。