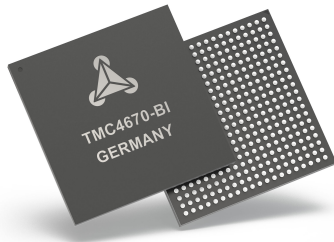


# TMC4670 Datasheet

IC Version V1.01 | Document Revision V1.00 • 2018-Mar-08

The TMC4670 is servo controller, providing Field Oriented Control for BLDC/PMSM and 2-phase Stepper Motors. Main control functions as torque, velocity and position control are implemented in hardware. Integrated ADCs, position sensor interfaces, position interpolators, enables a fully functional servo controller.



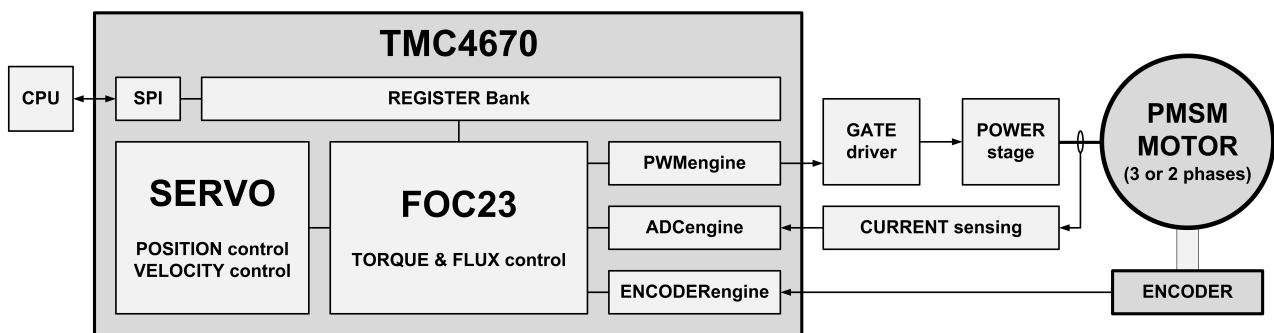
## Features

- Field Oriented Control (FOC) w/ Servo Controller
- Torque Control (FOC), Velocity Control, Position Control
- Feed Forward Offsets
- Integrated ADCs
- Encoder Engine: Hall analog/digital, Encoder analog/digital, 2nd digital Encoder
- Supports 3-Phase PMSM (FOC3) and 2-Phase Stepper Motors (FOC2)
- PWM Engine including SVPWM
- SPI Communication Interface

## Applications

- Robotics
- Pick and Place Machines
- Semiconductor Handling
- Factory Automation
- E-Mobility
- Laboratory Automation
- Blowers
- Pumps

## Simplified Block Diagram



# Contents

<b>1</b>	<b>Order Codes</b>	<b>4</b>
<b>2</b>	<b>Functional Summary</b>	<b>5</b>
<b>3</b>	<b>Functional Description</b>	<b>6</b>
3.1	Functional Blocks	6
3.2	Communication Interface	7
3.2.1	SPI Slave User Interface	7
3.3	Register Bank	8
3.3.1	Register Bank - Read and Write	8
3.3.2	Register Access Datagram Examples	9
3.3.3	Identification of Silicon via Type, Version, Date, and Time	9
3.3.4	Read of RAW Inputs & RAW Outputs	9
3.4	Numerical Representation, Electrical Angle, Mechanical Angle, and Pole Pairs	9
3.4.1	Numerical Representation	9
3.4.2	N_POLE_PAIRS, PHI_E, PHI_M	10
3.4.3	Numerical Representation of Angles PHI	11
3.5	ADC Engine	12
3.5.1	Internal ADC	13
3.5.2	External ADC (LTC2351)	13
3.5.3	ADC Selector & ADC Scaler w/ Offset Correction	13
3.6	Encoder Engine	15
3.6.1	Open Loop Encoder	15
3.6.2	Incremental ABN Encoder	15
3.6.3	Secondary Incremental ABN Encoder	17
3.6.4	Digital Hall Sensor Interface with optional Interim Position Interpolation	17
3.6.5	Digital Hall Sensor - Interim Position Interpolation	18
3.6.6	Digital Hall together with Incremental Encoder	18
3.6.7	Analog Hall and Analog Encoder Interface (SinCos of 0°90° or 0°120°240°)	19
3.6.8	Analog Position Decoder (SinCos of 0°90° or 0°120°240°)	20
3.7	FOC23 Engine	20
3.7.1	PI(D) Controllers	20
3.7.2	PI(D) Controller Calculations	21
3.7.3	PI(D) Controller - Clipping	21
3.7.4	PI Flux & PI Torque	22
3.7.5	PI Velocity	22
3.7.6	P(I) Position	23
3.7.7	Inner FOC Control Loop - Flux & Torque	23
3.7.8	FOC Transformations and PI(D) for control of Flux & Torque	23
3.7.9	Motion Modes	24
3.8	PWM Engine	24
3.8.1	PWM Polarities	25
3.8.2	PWM frequency	25
3.8.3	PWM Resolution	25
3.8.4	PWM Modes	25
3.8.5	Brake-Before-Make (BBM)	25
3.8.6	Space Vector PWM (SVPWM)	26
<b>4</b>	<b>Safety Functions</b>	<b>26</b>
<b>5</b>	<b>Register Map</b>	<b>27</b>
<b>6</b>	<b>Pinning</b>	<b>85</b>



<b>7</b>	<b>Package Dimensions</b>	<b>95</b>
<b>8</b>	<b>Characteristics</b>	<b>96</b>
8.1	Absolute Maximum Ratings	96
8.2	Recommended Operation Conditions	96
8.3	DC Characteristics	97
8.4	Timing Characteristics	97
8.5	Power Dissipation	97
<b>9</b>	<b>Supplemental Directives</b>	<b>98</b>
9.1	Producer Information	98
9.2	Copyright	98
9.3	Trademark Designations and Symbols	98
9.4	Target User	98
9.5	Disclaimer: Life Support Systems	98
9.6	Disclaimer: Intended Use	98
9.7	Collateral Documents & Tools	99
<b>10</b>	<b>Figures Index</b>	<b>100</b>
<b>11</b>	<b>Tables Index</b>	<b>101</b>
<b>12</b>	<b>Revision History</b>	<b>102</b>
12.1	IC Revision	102
12.2	Document Revision	102



## 1 Order Codes

Order Code	Description	Size [mm]
TMC4670-BI	TMC4670 FOC Servo Controller IC	17 x 17 x 2.5
TMC4670-EVAL	TMC4670 Evaluation Board	85 x 79

*Table 1: Order codes*



## 2 Functional Summary

- **Field Oriented Control (FOC) Servo Controller**
  - torque (and flux) control mode
  - velocity control mode
  - position control mode
  - update rate 200 kHz (w/ 4 kHz velocity meter sampling frequency)
- **PI Controllers**
  - programmable clipping of inputs and outputs of interim results
  - error sum (error integral over time) clipping
  - programmable target torque change in time (dTargetTorque/dt) limiter
  - programmable circular ( $\sqrt{U_D^2 + U_Q^2}$ ) limiter
  - PI controller clipping status bit vector for real time Monitoring
  - Feed Forward Offsets for Target Values
- **Supported Motor Types**
  - FOC3 : three-phase permanent magnet synchronous motors (PMSM)
  - FOC2 : two-phase stepper motors
- **ADC Engine with Offset Correction and Scaling**
  - integrated 12 bit ADCs as analog interface for currents and analog encoders
  - interface to external AD (LTC2351, 14 bit or 12 bit)
  - ADC register to write externally sampled data via SPI communication interface
- **Encoder Engine**
  - open loop position generator (programmable [rpm], [rpm/s]) for initial setup
  - digital incremental encoder (ABN resp. ABZ, up to 5MHz)
  - secondary digital incremental encoder
  - digital hall sensor interface (H<sub>1</sub>, H<sub>2</sub>, H<sub>3</sub> resp. H<sub>U</sub>, H<sub>V</sub>, H<sub>W</sub>)
  - digital hall sensor interface with interpolation of interim positions
  - analog encoder/analog hall sensor interface (SinCos (0°, 90°) or 0°, 120°, 240°)
  - position multi-turn counter (32 Bit)
- **PWM Engine including SVPWM**
  - programmable PWM frequency within range 25kHz ... 200kHz
  - programmable Brake-Before-Make (BBM) times (high side, low side) 0ns ... 2.5μs in 10ns steps
  - PWM auto scaling for transparent change of PWM frequency during motion
- **SPI Communication Interface**
  - 40 bit datagram length (1 ReadWrite bit + 7 address bits + 32 data bits)
  - immediate SPI read response (register read access by single datagram)
- **Supply Voltages** 3.3V, 2.5V, 1.2V
- **IO Voltage** 3.3V
- **Clock Frequency** 25MHz
- **Package** 17mm x 17mm BGA w/ 1mm ball pitch



### 3 Functional Description

TMC4670 is a fully integrated controller for field-oriented control (FOC) of either one 2-phase stepper motor (FOC2) or one 3-phase brushless motor (FOC3). It contains the complete control loop core architecture (position, velocity, torque) as well as required peripheral interfaces for communication with an application controller, for feedback (digital encoder, analog interpolator encoder, digital Hall, decoder Hall position interpolator, analog inputs for current and motor supply voltage measurement), and helpful additional IO. It supports highest control loop speed and PWM frequencies.

The TMC4670 is the building block for the user application that takes care of all real time critical tasks of field oriented motor control. It decouples the real time field oriented motor control and its real sub-tasks as current measurement, real time sensor signal processing, real time PWM signal generation from the user application layer as outlined by figure 17.

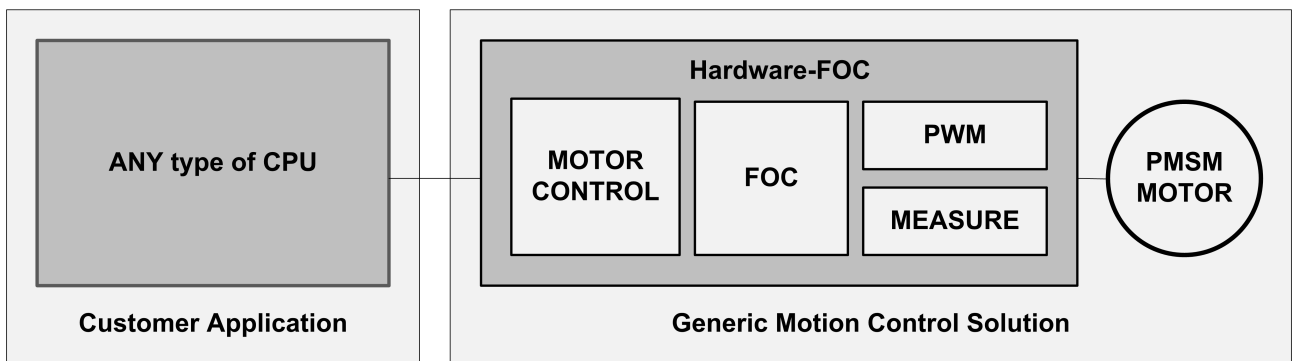


Figure 1: Hardware FOC Application Diagram

#### 3.1 Functional Blocks

Application interface, register bank, ADC engine, encoder engine, FOC torque PI controller, velocity PI controller, position P controller, and PWM engine form the TMC4670. The TMC4670 supports 3-phase PMSM motors (FOC3) and 2-phase PMSM stepper motors (FOC2).

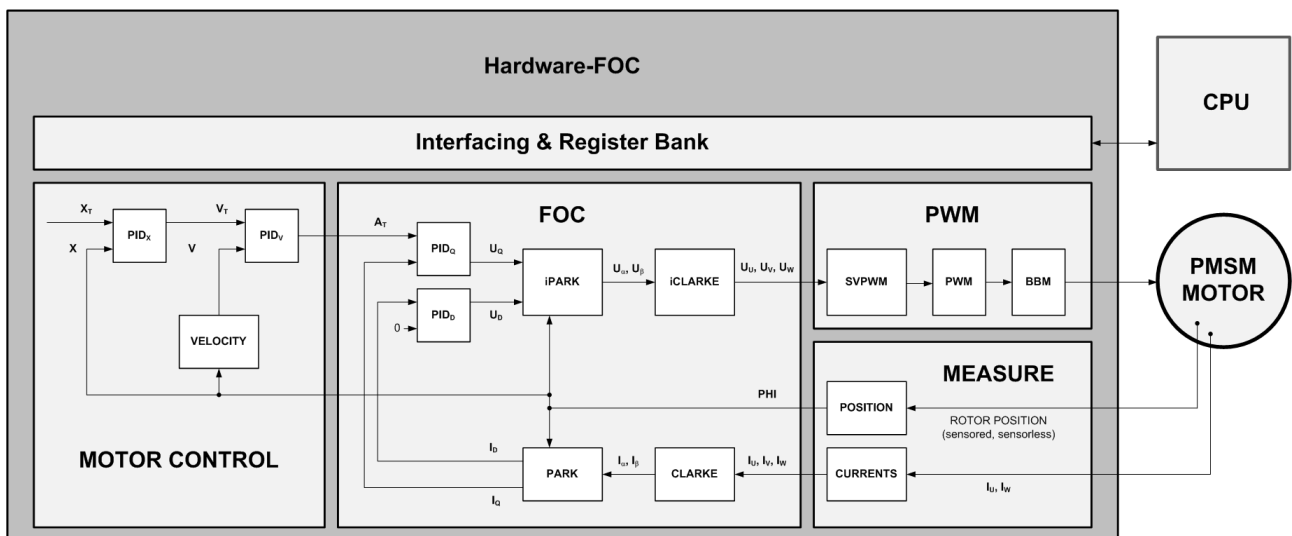


Figure 2: Hardware FOC Block Diagram



The ADC engine interfaces integrated ADC channels and maps raw ADC values to signed 16 bit (s16) values for the inner FOC current control loop based on programmable offset and scaling factors. The FOC torque PI controller forms the inner base component including required transformations (Clark, Park, inverse Park, inverse Clark). All functional blocks are pure hardware.

### 3.2 Communication Interface

The TMC4670 is equipped with an SPI interface for access to all registers of the TMC4670.

#### 3.2.1 SPI Slave User Interface

The SPI of the TMC4670 for the user application has an easy command and control structure. The TMC4670 user SPI acts as a slave. The SPI datagram length is 40 bit with up to 2Mbit/s. The MSB (bit#39) is sent first. The LSB (bit#0) is sent last. The MSB (bit#39) is the WRITE\_notREAD (WRnRD) bit. The bits (bit#39 to bit#32) are the address bits (ADDR). Bits (bit#31 to bit#0) are (up to) 32 data bits. The SPI of the TMC4670 immediately responds within the actual SPI datagram on read and write for ease-of-use communication.

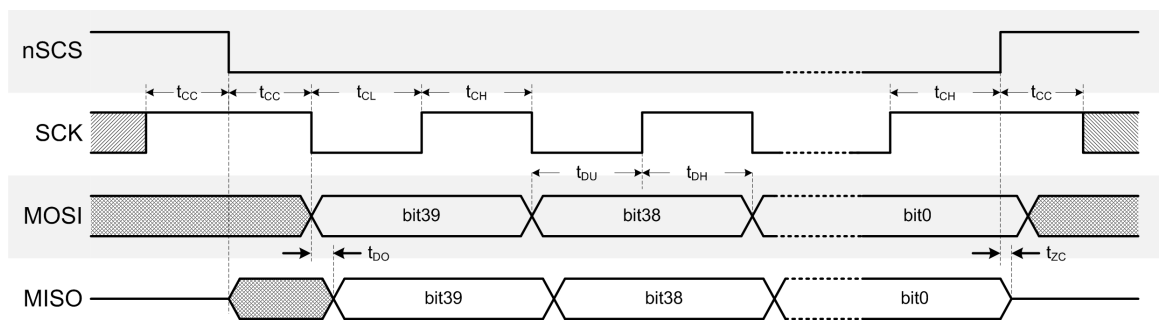


Figure 3: SPI Timing

SPI Interface Timing		Characteristics, fCLK = 25MHz				
Parameter	Symbol	Condition	Min	Typ	Max	Unit
SCK valid before or after change of nSCS	$t_{CC}$		250			ns
nSCS high time	$t_{CSH}$		250			ns
nSCS low time	$t_{CSL}$		250			ns
SCK high time	$t_{CH}$		250			ns
SCK low time	$t_{CL}$		250			ns
SCK low time	$t_{CL}$		250			ns
SCK frequency	$f_{SCK}$				2	MHz
MOSI setup time before rising edge of SCK	$t_{DU}$		250			ns
MOSI hold time after falling edge of SCK	$t_{DH}$		250			ns
MISO data valid time after falling edge of SCK	$t_{DO}$				10	ns

Table 2: SPI Timing Parameter



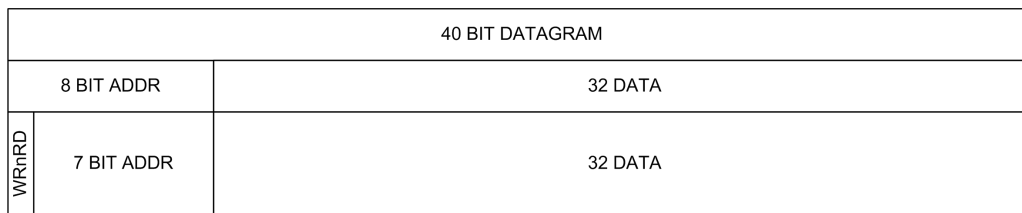


Figure 4: SPI Datagram Structure

### 3.3 Register Bank

#### **Info**

This section gives a functional description as an overview. The section 5 starting page 27 gives the detailed description of each register.

The register bank is the interface to the user application. Each register of the TMC4670 has 8 bit address followed by up to 32 data bits. Some addresses hold more than one data registers for simultaneous access or composed control bit vectors. Section 5 page 27 describes all registers in detail.

During initialization, the user writes parameters into associated registers. These parameters are scaling factors and offsets, sensor configuration parameters, limits for clipping, selections, P and I parameters for the FOC torque controller, P and I parameters for velocity controller, and P parameter for the position controller.

During application, the user writes application parameters into associated registers. These are - depending on the motion mode - target torque, target velocity, or target position.

The TMC4670 has direct access registers and indirect access registers. Most registers are direct access registers with read or write access by a single datagram. Some less often used registers (e.g. silicon version registers, internal values for read out) are accessed via two registers: address register and data register. The address register selects the address, the second register holds the data.

#### 3.3.1 Register Bank - Read and Write

From the access point of view there are two kind of registers: read-only and read-write. The most significant bit (MSB) of each register access datagram defines read (=0) or write (=1). So, there are 128 read addresses (0x00<sub>h</sub> ... 0x7F<sub>h</sub>) and 128 write addresses (0x80<sub>h</sub> ... 0xFF<sub>h</sub>) available. The TMC4670 ignores write accesses to read-only registers.

- Fixed Read Only Register (e.g. SILICON\_TYPE, SILICON\_VERSION, SILICON\_DATE, SILICON\_TIME)
- Read Only Register for internal values (e.g. scaled ADC values)
- Read Only Register for external Signals (e.g. ADC raw values, ABN encoder inputs, Hall signal inputs)
- Read Write Register for configurations (e.g. P and I parameter of PI controller, clipping parameters)
- Read Write Register for target values (e.g. target torque, target velocity, target position)
- Dual Ported Read Write Register (e.g. encoder count, actual position)





### 3.3.2 Register Access Datagram Examples

0x0100000000<sub>h</sub> : reads data from address 0x01<sub>h</sub>

0x8123456789<sub>h</sub> : writes data 0x23456789<sub>h</sub> to address 0x01<sub>h</sub>

### 3.3.3 Identification of Silicon via Type, Version, Date, and Time

The read-only registers of silicon type, version with date and time identify the type of the silicon, the version of the silicon together with unique date stamp and time stamp. This enables the automatic identification of IC and version and enable the automatic handling of different IC and different versions.

### 3.3.4 Read of RAW Inputs & RAW Outputs

For ease-of-use while setting up the configuration, raw input signals and raw output signals are mapped into the register bank for user read out. With this, the user can initially check without a scope that the desired signals come into the TMC4670 as expected. Examples of readable raw input signals are digital Hall signals and incremental encoder signals.

## 3.4 Numerical Representation, Electrical Angle, Mechanical Angle, and Pole Pairs

The TMC4670 uses different numerical representations for different parameters, measured values, and interim results. The terms electrical angle PHI\_E, mechanical angle PHI\_M, and number of pole pairs (N\_POLE\_PAIRS) of the motor are important for setup of FOC. This section describes the different numerical representations of parameters and terms.

### 3.4.1 Numerical Representation

The TMC4670 uses signed and unsigned values of different length and fixed point representation for parameters that require a non-integer granularity.

Symbol	Description	Min	Max
u16	unsigned 16 bit value	0	65535
s16	signed 16 bit values, 2 <sup>th</sup> complement	-32767	32767
u32	unsigned 32 bit value	0	2 <sup>32</sup> = 4294967296
s32	signed 32 bit values, 2 <sup>th</sup> complement	-2147483647	2 <sup>31</sup> - 1 = 2147483647
q8.8	signed fix point value with 8 bit interger part and 8 bit fractional part	-32767/256	32767/256
q4.12	signed fix point value with 4 bit interger part and 12 bit fractional part	-32767/4096	-32767/4096

Table 3: Numerical Representations

#### **i** Info

Two's complement of n bit is  $-2^{n-1} \dots -2^{n-1} - 1$ . To avoid un-wanted overflow, the range is clipped to  $-2^{n-1} + 1 \dots -2^{n-1} - 1$ .

Because the zero is interpreted as a positive number for 2<sup>th</sup> complement representation of interger n bit number, the smallest negative number is  $-2^{(n-1)}$  where the largest positive number is  $2^{(n-1)} - 1$ . Using



the smallest negative number  $-2^{(n-1)}$  might cause critical under-flow or over-flow. Internal clipping takes this into account by mapping  $-2^{(n-1)}$  to  $-2^{(n-1)} + 1$ .

Hexadecimal Value	u16	s16	q8.8	q4.12
0x0000 <sub>h</sub>	0	0	0.0	0.0
0x0001 <sub>h</sub>	1	1	1 / 256	1 / 4096
0x0001 <sub>h</sub>	2	2	2 / 256	2 / 4096
0x0080 <sub>h</sub>	128	128	0.5	0.03125
0x0100 <sub>h</sub>	256	256	1.0	0.0625
0x0200 <sub>h</sub>	512	512	2.0	0.125
0x3FFF <sub>h</sub>	16383	16383	16383 / 256	16383 / 4096
0x5A81 <sub>h</sub>	23169	23169	23169 / 256	23169 / 4096
0x7FFF <sub>h</sub>	32767	32767	32767 / 256	32767 / 4096
0x8000 <sub>h</sub>	32768	-32768	-32768 / 256	-32768 / 4096
0x8001 <sub>h</sub>	32769	-32767	-32767 / 256	-32767 / 4096
0x8002 <sub>h</sub>	32770	-32766	-32766 / 256	-32766 / 4096
0xC001 <sub>h</sub>	49153	-16383	-16383 / 256	-16383 / 4096
0xFFFF <sub>h</sub>	65534	-2	-2 / 256	-2 / 4096
0xFFFF <sub>h</sub>	65535	-1	-1 / 256	-1 / 4096

Table 4: Examples of u16, s16, q8.8, q4.12

The q8.8 and q4.12 are used for P and I parameters which are positive numbers but q8.8 and q4.12 are used as signed numbers. This is because these values are multiplied with signed error values resp. error integral values.

### 3.4.2 N\_POLE\_PAIRS, PHI\_E, PHI\_M

The parameter N\_POLE\_PAIRS defines the factor between electrical angle PHI\_E and mechanical angle PHI\_M of a motor (pls. refer figure 5).

A motor with one (1) pole pair turns once for each electrical period. A motor with two (2) pole pairs turns once for each two electrical periods. A motor with three (3) pole pairs turns once for each three electrical periods. A motor with four (4) pole pairs turns once for each four electrical periods.

The electrical angle PHI\_E is relevant for the commutation of the motor. It is relevant for the torque control of the inner FOC loop.

$$\text{PHI}_E = \text{PHI}_M \cdot \text{N\_POLE\_PAIRS} \quad (1)$$

The mechanical angle PHI\_M is primarily relevant for velocity control and for positioning. This is because one wants to control the motor speed in terms of mechanical turns and not in terms of electrical turns.



$$\text{PHI\_M} = \text{PHI\_E} / \text{N\_POLE\_PAIRS} \tag{2}$$

Different encoders give different kind of position angles. Analog hall sensors normally give the electrical position PHI\_E that can be used for commutation. Analog encoders give - depending on their resolution - angles that have to be scaled first to mechanical angles PHI\_M and to electrical angles PHI\_E for commutation.

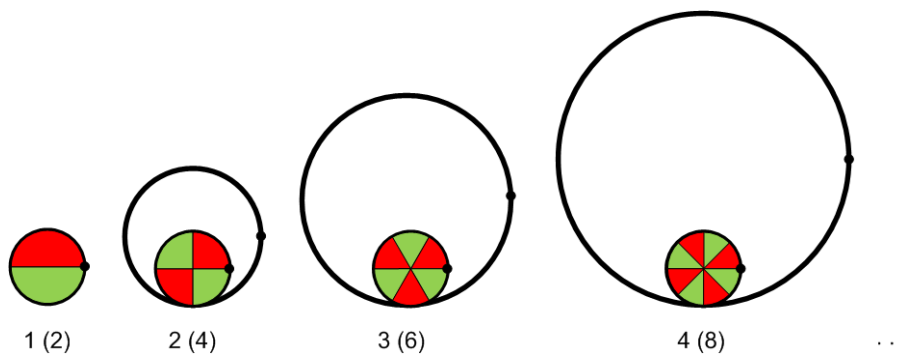


Figure 5: N\_POLE\_PAIRS - Number of Pole Pairs

### 3.4.3 Numerical Representation of Angles PHI

Electrical angles and mechanical angles are represented as 16 bit integer values. One full revolution of 360 deg is equivalent to  $2^{16} = 65536$  steps. Any position coming from a sensor is mapped to this integer range. Adding an offset of PHI\_OFFSET causes a rotation of an angle  $\text{PHI\_OFFSET} / 2^{16}$ . Subtraction of an offset causes a rotation of an angle PHI\_OFFSET in opposite direction.

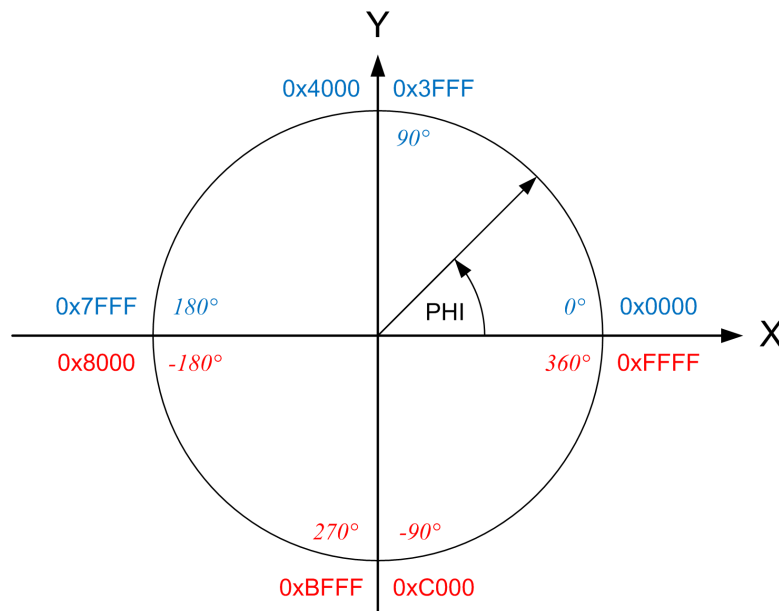


Figure 6: Integer Representation of Angles with 16 Bit as s16 resp. u16



Hexadecimal Value	u16	s16	PHI[°]	±PHI[°]
0x0000 <sub>h</sub>	0	0	0.0	0.0
0x1555 <sub>h</sub>	5461	5461	30.0	-330.0
0x2AAA <sub>h</sub>	10922	10922	60.0	-300.0
0x4000 <sub>h</sub>	16384	16384	90.0	-270.0
0x5555 <sub>h</sub>	21845	21845	120.0	-240.0
0x6AAA <sub>h</sub>	27306	27768	150.0	-210.0
0x8000 <sub>h</sub>	32768	-32768	180.0	-180.0
0x9555 <sub>h</sub>	38229	-27307	210.0	-150.0
0xAAAA <sub>h</sub>	43690	-21846	240.0	-120.0
0xC000 <sub>h</sub>	49152	-16384	270.0	-90.0
0xD555 <sub>h</sub>	54613	-10923	300.0	-60.0
0xEAAA <sub>h</sub>	60074	-5462	330.0	-30.0

Table 5: Examples of u16, s16, q8.8

The option of adding an offset is for adjustment of angle shift between motor and stator and rotor and encoder. Finally, the relative orientations between motor and stator and rotor and encoder can be adjusted by just one offset. Alternatively, one can set the counter position of an incremental encoder to zero on initial position. For absolute encoders one needs to use the offset to set an initial position.

### 3.5 ADC Engine

The ADC engine controls the sampling of different available ADC channels.

The FOC engine expects offset corrected ADC values, scaled into the FOC engine 16 bit (s16) fixed point representation. The integrated scaler and offsetter maps raw ADC samples of current measurement channels to 16 bit two's complement values (s16). Both, offset and scale calculations are signed. With this, the user can change the signs of current according to the application by the scaling factors.

The s16 scaled ADC values are available for read out from the register by the user. ADC samples for motor supply voltage (VM), MOSFET temperature, motor temperature, general purpose analog input (AIN) are only raw values without scaling.

- ADC samples of integrated ADC
- ADC samples from external ADC (LTC2351)
- ADC samples from external sources can be written into dedicated registers (ADC EXT)
- ADC values are for:
  - phase current measurement (most important task)
  - Supply voltage measurement (for monitoring or brake chopper)
  - Analog Hall signal measurement
  - analog Sine/Cosine encoder signal measurement
  - analog voltage input for MOS-FET temperature signal monitoring



- analog voltage input for motor temperature signal monitoring

---

**Info**

Wrong scaling factors or wrong offsets might cause damages when the closed current regulation is active. Integrated hardware limiters allow protection especially in the setup phase when using careful limits.

---

### 3.5.1 Internal ADC

The TMC4670 is equipped with internal ADCs with input voltage range of 0V ... 2.5V for current measurement, supply voltage measurement, analog hall signal measurement, analog encoder.

### 3.5.2 External ADC (LTC2351)

Alternatively to the integrated ADCs, the TMC4670 supports external SPI ADCs LTC2351 from Linear Technology for current measurements. This is intended for current sensing on separate power stages.

#### 3.5.2.1 ADC RAW

The sampled raw ADC values are available for read out by the user. This is important during the system setup phase to determine offset and scaling factors.

#### 3.5.2.2 ADC EXT

The user can write ADC values into the ADC EXT registers of the register bank from external ADC sources. For example it there are high precision ADC values available from an external ADC.

### 3.5.3 ADC Selector & ADC Scaler w/ Offset Correction

The ADC selector selects ADC channels for FOC. The 3-phase FOC used two of three ADC channels for measurement and calculates the third channel via Kirchhoff's Law from the scaled and offset corrected ADC values. The 2-phase FOC just used two ADC channels because for the 2-phase stepper motor the two phases are independent from each other.

---

**Note**

The Open Loop Encoder is useful for setup of ADC channel selection, scaling, and offset by turning a motor open loop.

---

The FOC23 Engine processes currents as 16 bit signed (s16) values. Raw ADC values are expanded to 16 bit width independent of their resolution. With this, each ADC is available for read out as a 16 bit number. The ADC scaler w/ offset correction is for pre-processing of measured raw current values. It might be used to map to own units (e.g. A or mA). For scaling, gains of current amplifiers, reference voltages, and offsets have to be taken into account.

---

**Info**

Raw ADC values generally are of 16 bit width independent of their real resolution.

---





The job of the ADC scaler is to map raw ADC values to the 16 bit signed (s16) range and to center the values to zero by removing of offsets.

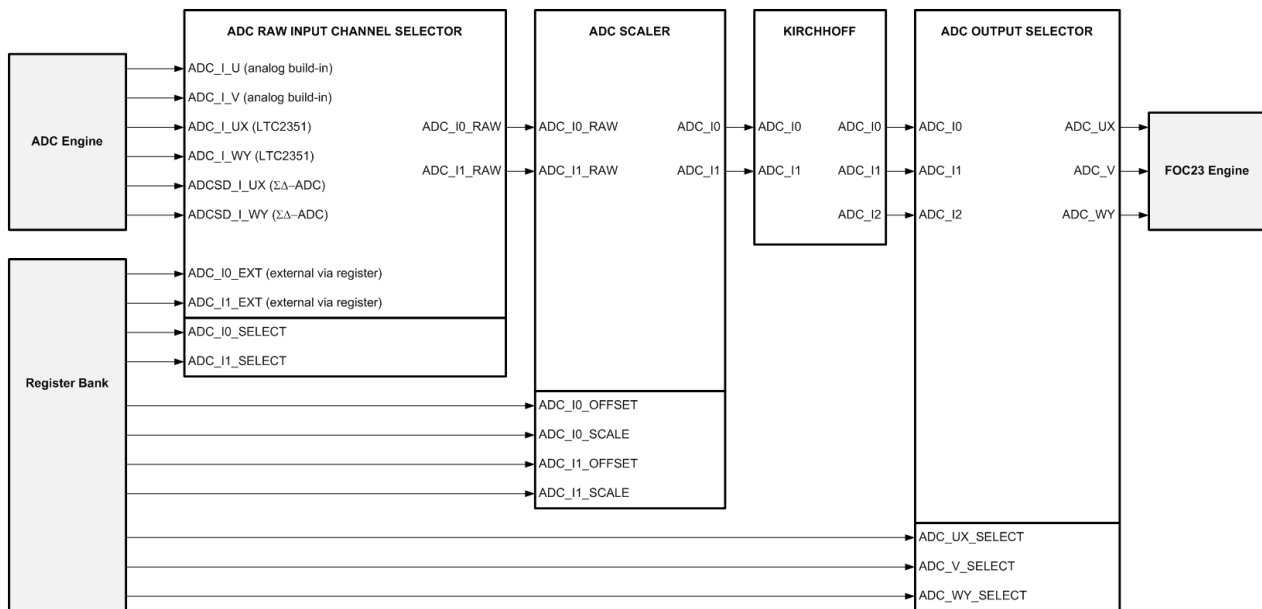


Figure 7: ADC Selector & Scaler w/ Offset Correction

ADC offsets and ADC scalers for the analog current measurement input channels need to be programmed into the associated registers. Each ADC\_I\_U, ADC\_I\_V, ADC\_I\_UX, ADC\_I\_WY, ADCSD\_I\_UX, ADCSD\_I\_WY, ADC\_I0\_EXT, ADC\_I1\_EXT is mapped either to ADC\_I0\_RAW or to ADC\_I1\_RAW by ADC\_I0\_SELECT and ADC\_I1\_SELECT.

In addition, the ADC\_OFFSET is for conversion of unsigned ADC values into signed ADC values as required for the FOC.

$$\text{ADC\_I0} = (\text{ADC\_I0\_RAW} + \text{ADC\_I0\_OFFSET}) \cdot \text{ADC\_I0\_SCALE} \quad (3)$$

$$\text{ADC\_I1} = (\text{ADC\_I1\_RAW} + \text{ADC\_I1\_OFFSET}) \cdot \text{ADC\_I1\_SCALE} \quad (4)$$

For FOC3 the third current ADC\_I2 is calculated via Kirchhoff's Law. This requires the correct scaling and offset correction before. For FOC2 there is no calculation of a third current.

The ADC\_UX\_SELECT selects one of the three ADC channels ADC\_I0, ADC\_I1, ADC\_I2 for ADC\_UX.

The ADC\_V\_SELECT selects one of the three ADC channels ADC\_I0, ADC\_I1, ADC\_I2 for ADC\_V.

The ADC\_WY\_SELECT selects one of the three ADC channels ADC\_I0, ADC\_I1, ADC\_I2 for ADC\_WY.

For FOC3 the third current ADC\_I2 is calculated via Kirchhoff's Law. This requires the correct scaling and offset correction before. For FOC2 there is no calculation of a third current.



The ADC\_UX, ADC\_V, ADC\_WY are for the FOC3 (U, V, W). The ADC\_UX. and ADC\_WY (X, Y) are for the FOC2.

---

**Note** The Open Loop Encoder is useful for setup of ADC channel selection, scaling, and offset by turning a motor open loop.

---

## 3.6 Encoder Engine

The encoder engine is an unified position sensor interface. It maps the selected encoder position information to electrical position (PHI\_E) and to mechanical position (PHI\_M). Both are 16 bit values. The encoder engine maps single turn positions from position sensors to multi-turn position. The user can overwrite the multi-turn position for initialization.

The different position sensors are the position sources for torque and flux control via FOC, for velocity control, and for position control. The PHI\_E\_SELECTION selects the source of the electrical angle PHI\_E for the inner FOC control loop. VELOCITY\_SELECTION selects the source for velocity measurement. With PHI\_E selected as source for velocity measurement, one gets the electrical velocity. With the mechanical angle PHI\_M selected as source for velocity measurement one gets the mechanical velocity taking the set number of pole pairs (N\_POLE\_PAIRS) of the motor into account. Nevertheless, for high precision position it might be useful to do positioning based on the electrical angle PHI\_E.

### 3.6.1 Open Loop Encoder

For initial system setup the encoder engine is equipped with an open loop position generator. With one can turn the motor open-loop by specifying speed in rpm and acceleration in rpm/s together with a voltage UD\_EXT in D direction. So, the open-loop encoder it is not a real encoder, it just gives positions as an encoder does. The open-loop decoder has a direction bit to define once the direction of motion for the application.

---

**Note** The open loop encoder is useful for initial ADC setup, encoder setup, hall signal validation, and for validation of the number of pole pairs of a motor. The open loop encoder turns a motor open with programmable velocity in unit [RPM] with programmable acceleration in unit [RPM/s].

---

So, with the open loop encoder one can turn a motor without any position sensor and without any current measurement as the first step of doing the system setup. With the turning motor one can adjust the ADC scales and offsets and set up position sensors (hall, incremental encoder, ...) according to resolution, orientation, direction of rotation.

### 3.6.2 Incremental ABN Encoder

Incremental encoders give two phase shifted incremental pulse signals A and B. Some incremental encoders have an additional null position signal N or zero pulse signal Z. An incremental encoder (called ABN encoder or ABZ encoder) has an individual number of incremental pulses per revolution. The number of incremental pulses defines the number of positions per revolution (PPR). The PPR might mean pulses per revolution or periods per revolution. Instead of positions per revolution some incremental encoder vendors call these CPR counts per revolution.

The PPR parameter is the most important parameter of the incremental encoder interface. With that, it forms a modulo (PPR) counter, counting from 0 to (PPR-1). Depending on the direction, it counts up or



down. The modulo PPR counter is mapped into the register bank as a dual ported register. the user can overwrite it with an initial position. The ABN encoder interface provides both, the electrical position and the multi-turn position are dual-ported read-write registers.

---

**Note** The PPR parameter must be set exactly according to the used encoder.

---

The N pulse from an encoder triggers either sampling of the actual encoder count to fetch the position at the N pulse or it re-writes the fetched N position on an N pulse. The N pulse can either be used as stand alone pulse or and-ed with NAB = N and A and B. It depends on the decoder what kind of N pulse has to be used, either N or NAB. For those encoders with precise N pulse within on AB quadrature, the N pulse must be used. For those encoders with N pulse over four AB quadratures the user can enhance the precision of the N pulse position detection by using NAB instead of N, which is recommended.

---

**Note** Incremental encoders are available with N pulse and without N pulse.

---

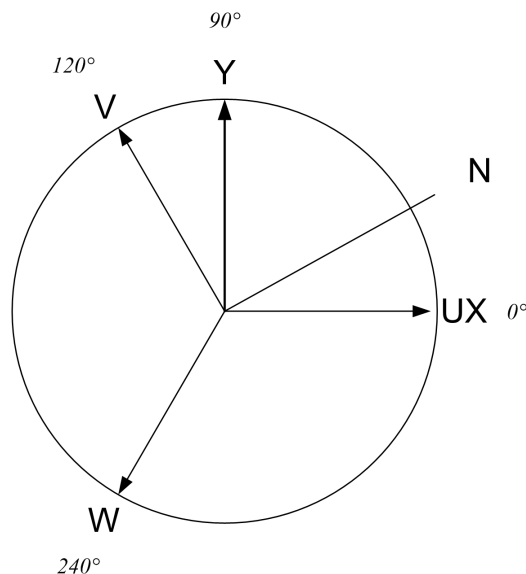


Figure 8: ABN Incremental Encoder N Pulse

The polarity of N pulse, A pulse and B pulse are programmable. The N pulse is for re-initialization with each turn of the motor. Once fetched, the ABN decoder can be configured to write back the fetched N pulse position with each N pulse.

---

**Note** The ABN encoder interface has a direction bit to set once the direction of motion for the application.

---

Logical ABN = A and B and N might be useful for incremental encoders with low resolution N pulse to enhance the resolution. On the other hand, for incremental encoders with high resolution n pulse a logical abn = a and b and n might totally suppress the resulting n pulse.





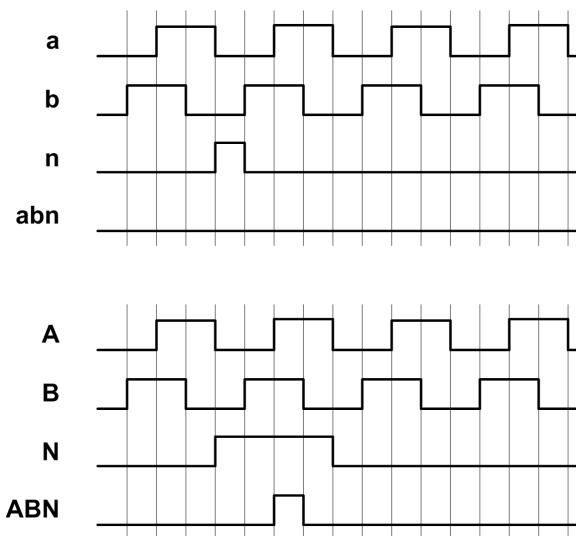


Figure 9: Encoder ABN Timing - high precise *n* pulse and less precise *N* pulse

### 3.6.3 Secondary Incremental ABN Encoder

For commutating a motor with FOC one selects a position sensor source (digital incremental encoder, digital hall, analog hall, analog incremental encoder, ...) that is mounted close to the motor. The inner FOC loop controls torque and flux of the motor based on the measured phase currents and the electrical angle of the rotor.

The TMC4670 is equipped with a secondary incremental encoders interface. This secondary encoder interface is available as source for velocity control or position control. This is for applications where a motor turns an object with a gear to position the object. An example is a robot arm where a motor moves an angle with a the mechanical angle of the arm as the target.

#### **i** Info

The secondary incremental encoder is not available for commutation (PHI\_E) for the inner FOC. In others words, there is no electrical angle PHI\_E selectable from the secondary encoder.

### 3.6.4 Digital Hall Sensor Interface with optional Interim Position Interpolation

The digital hall interface is the position sensor interface for digital hall signals. The digital hall signal interface first maps the digital hall signals to an electrical position PHI\_E\_RAW. An offset PHI\_E\_OFFSET can be used to rotate the orientation of the hall signal angle. The electrical angle PHI\_E is for commutation. Optionally, the default electrical positions of the Hall sensors can be adjusted by writes into the associated registers.



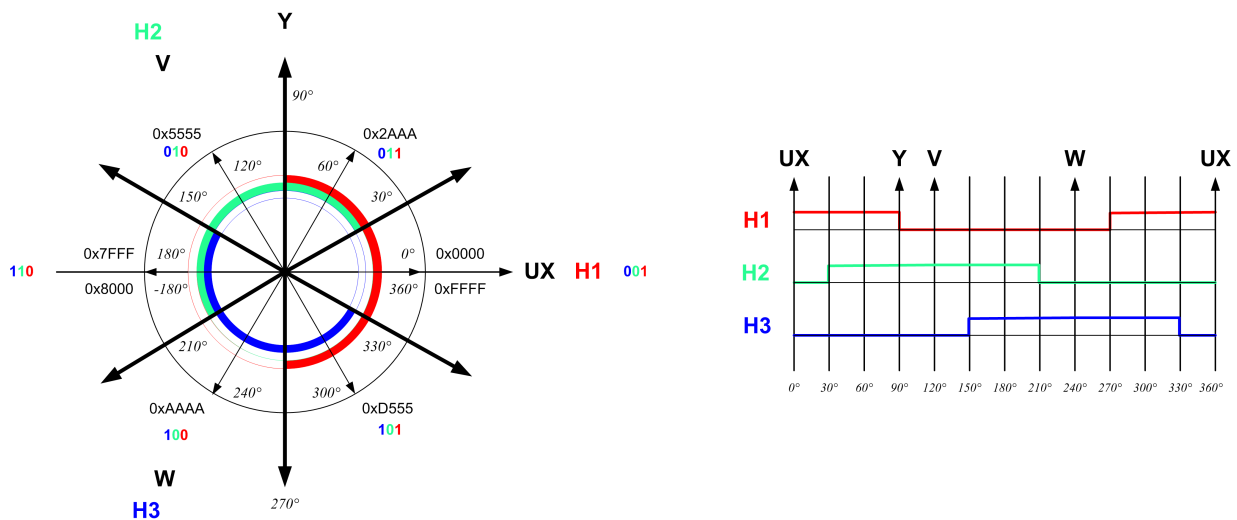


Figure 10: Hall Sensor Angles

Hall Sensors give absolute positions within an electrical period with a resolution of 60° as 16 bit positions (s16 resp. u16) PHI. With activated interim hall position interpolation the TMC4670 additionally generates high resolution interim positions, when the motor is running at speed beyond 60 rpm.

### 3.6.5 Digital Hall Sensor - Interim Position Interpolation

For lower torque ripple the user can enable the position interpolation of interim hall positions. This function is useful for motors, which are compatible with sine wave commutation, but are equipped with digital hall sensors.

When the position interpolation is switched on, it becomes active on speed beyond 60 rpm. For lower speed it is automatically disabled. This is important especially when the motor has to be at rest.

Motors that are intended for block commutation might smarter turn with hall signal interpolation but the user should not expect too much for those motors.

### 3.6.6 Digital Hall together with Incremental Encoder

If a motor is equipped with both Hall sensors and incremental encoder, the hall sensors can be used for the initialization as a low resolution absolute position sensor and later the incremental encoder can be used as a high resolution sensor for commutation.



### 3.6.7 Analog Hall and Analog Encoder Interface (SinCos of 0°/90° or 0°/120°/240°)

An analog encoder interface is part of the decoder engine. It is able to handle analog position signals of 0° and 90° and 0°/120°/240°. The analog decoder engine adds offset, scales the raw analog encoder signals and calculates the electrical angle PHI\_E from these analog position signals.

ADC offsets and ADC scalers need to be programmed into the associated registers to use analog Hall sensors or analog encoders. Each AENC\_0\_SELECT, AENC\_1\_SELECT, AENC\_2\_SELECT, and AENC\_3\_SELECT, selects one raw analog ADC input channel AENC out of AENC\_UX\_RAW, AENC\_VN\_RAW, AENC\_WY\_RAW, AENC\_N\_RAW, or one AENC register channel AENC\_UX\_EXT, AENC\_VN\_EXT, AENC\_WY\_EXT, AENC\_N\_EXT.

An individual signed offset is added to each associated raw ADC channel and scaled by its associated scaling factor according to

$$AENC\_VALUE = (AENC\_RAW + AENC\_OFFSET) \cdot AENC\_SCALE \tag{5}$$

In addition, the AENC\_OFFSET is for conversion of unsigned ADC values into signed ADC values as required for the FOC.

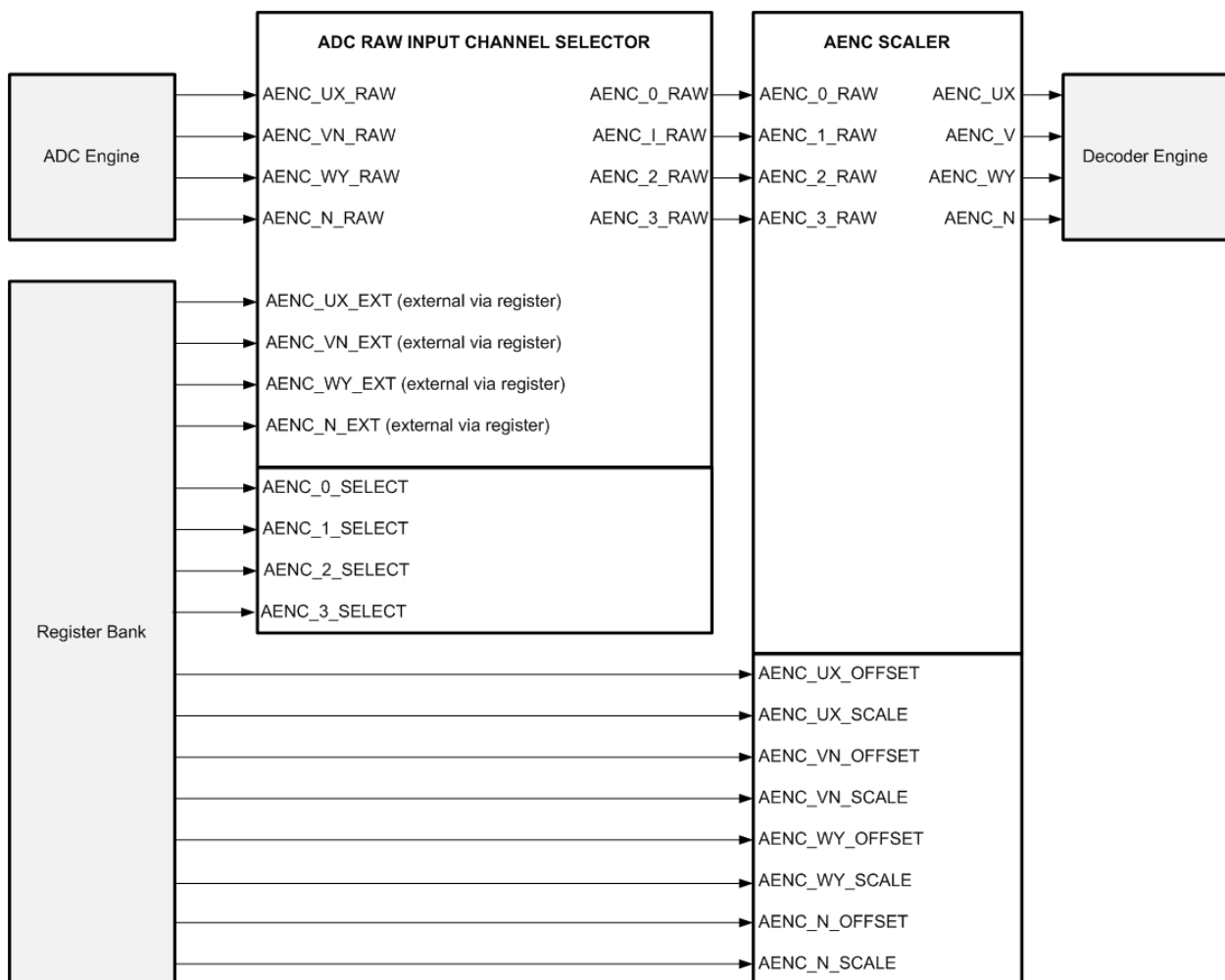


Figure 11: Analog Encoder (AENC) Selector & Scaler w/ Offset Correction



**Info**

The analog N pulse is just a raw ADC value. Scaling, offset correction, manual handling of analog N pulse similar to N pulse handling of digital encoder N pulse is not implemented for analog encoder.

### 3.6.8 Analog Position Decoder (SinCos of 0°90° or 0°120°240°)

The extracted positions from the analog decoder are available for read out from registers.

#### 3.6.8.1 Multi-Turn Counter

Electrical angles are mapped to a multi-turn position counter. The user can overwrite this multi-turn position for initialization purposes.

#### 3.6.8.2 Encoder Engine Phi Selector

The angle selector selects the source for the commutation angel PHI\_E. That electrical angle is available for commutation.

#### 3.6.8.3 External Position Register

A register value written via the application interface into the register bank is available for commutation also. With this, the user can interface to any encoder by just writing positions extracted from external encoder into this regulator. From the decoder engine point of view this is just one more selectable encoder source. As the application interface is not fast enough for high commutation frequencies, this mode of operation is only recommended for initialization.

## 3.7 FOC23 Engine

**Info**

Support for the TMC4670 is integrated into the TMCL-IDE including wizards for system setup, which allow easy and fast commissioning and even turn the motor with a few steps. With the TMCL-IDE the user has direct access to all registers of the TMC4670.

The FOC23 engine performs the inner current control loop for the torque current  $I_Q$  and the flux current  $I_D$  including the required transformations. Programmable limiters take care of clipping of interim results. Per default, the programmable circular limiter clips  $U_D$  and  $U_Q$  to  $U_{D,R} = \sqrt{2} \cdot U_Q$  and  $U_{R,R} = \sqrt{2} \cdot U_D$ . PI controllers perform the control tasks.

### 3.7.1 PI(D) Controllers

PI controllers are used for current control and velocity control. A P controller is used for position control. The D part is not yet supported, it is just a register place holder for future variants.



### 3.7.2 PI(D) Controller Calculations

The PI controllers performs the calculation

$$dXdT = P \cdot e + I \cdot \int_0^t e(t) dt \quad (6)$$

with

$$e = X\_TARGET - X \quad (7)$$

where X\_TARGET stands for target flux, target torque, target velocity, or target position with error e that is the difference between target value and actual values. The time constant dt is 1μs with the integral part is divided by 256.

### 3.7.3 PI(D) Controller - Clipping

The limiting of target values for PI controllers and output values of PI controllers is programmable. Per power on default theses limiter are set to maximum values. Before one starts a motor one should set the limiters for clipping.

The target input is clipped to X\_TARGET\_LIMIT. The output of a PI(D) is named dXdT because it gives the desired derivative d/dt as a target value to the following stage: The position (x) controller gives velocity (dx/dt). The output of the PI(D) is clipped to dXdT\_LIMIT. The error integral of (6) is clipped to dXdT\_LIMIT / I.



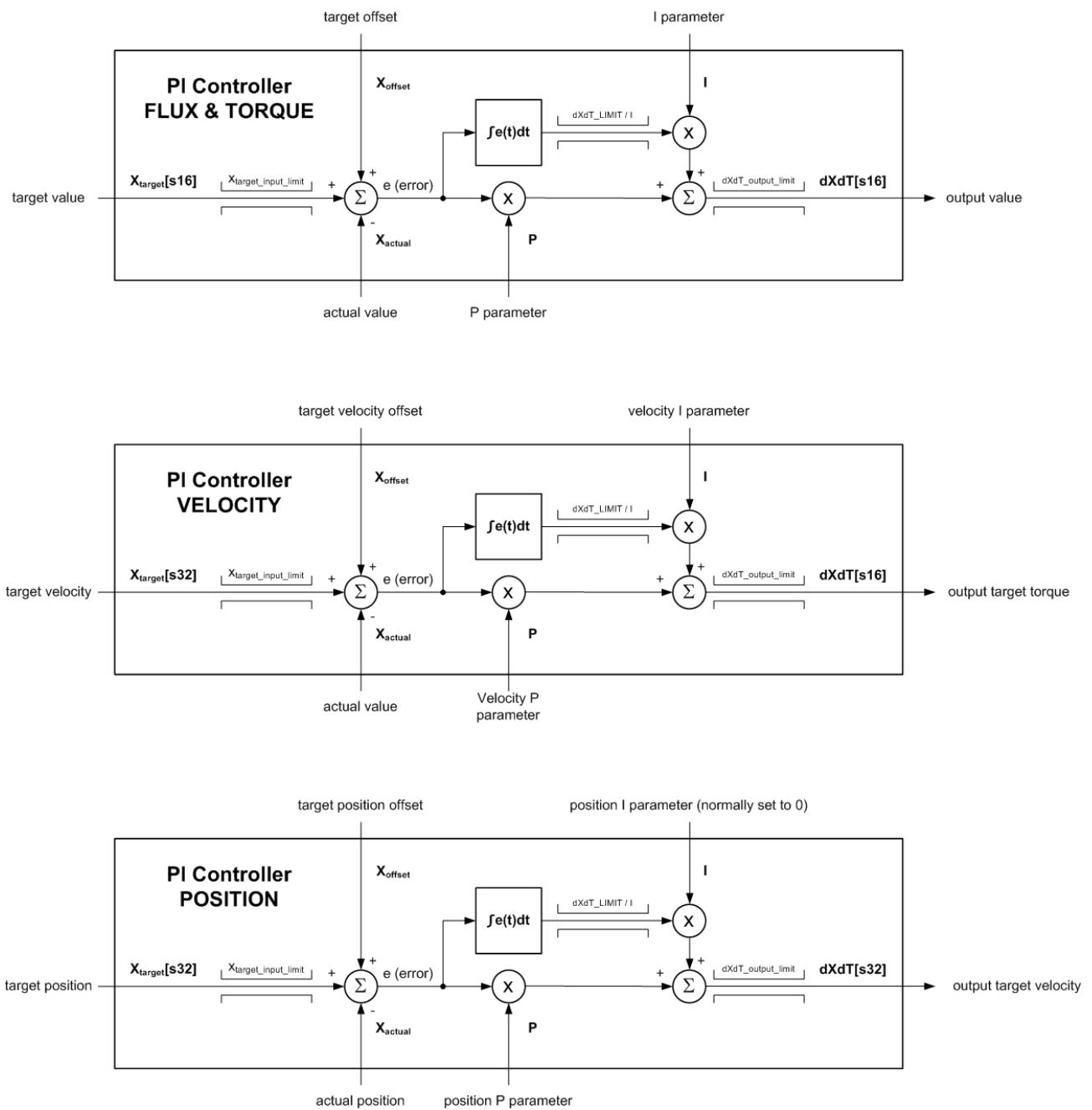


Figure 12: PID Architectures

### 3.7.4 PI Flux & PI Torque

The P part is represented as q8.8 and I is the I part represented as q0.15.

### 3.7.5 PI Velocity

The P part is represented as q8.8 and I is the I part represented as q0.15.



### 3.7.6 P(I) Position

For the position regulator, the P part is represented as q4.12 to be compatible with the high resolution positions - one single rotation is handled as an s16.

This is because  $e = x - x_{\text{target}}$  might result in larger  $e[s32]$  for  $x[s32]$  and  $x_{\text{target}}[s32]$  represented as s32 for  $e = x - x_{\text{target}}$  for  $x[s16]$  and  $x_{\text{target}}[s16]$  represented as s16.

### 3.7.7 Inner FOC Control Loop - Flux & Torque

The inner FOC loop (figure 13) controls the flux current to a flux target and the torque current to the desired torque target. The inner FOC loop performs the desired transformations according to figure 14 for 3-phase motors (FOC3). For 2-phase motors (FOC2) both Clarke (CLARKE) transformation and inverse Clarke (iCLARKE) a by-passed.

The inner FOC control loop gets a target torque value ( $I_Q\_TARGET$ ) that represents acceleration, the rotor position, and the measured currents as input data. Together with the programmed P and I parameters, the inner FOC loop calculates three target voltage values as input for the PWM engine.

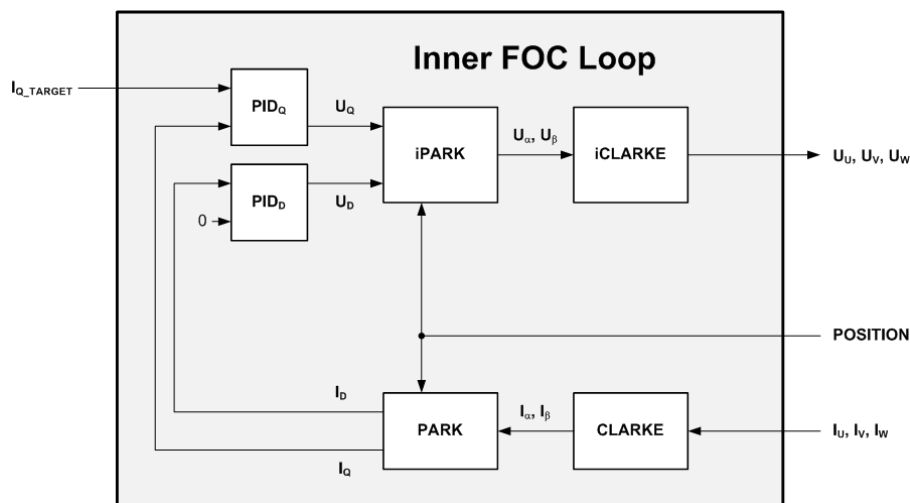


Figure 13: Inner FOC Control Loop

### 3.7.8 FOC Transformations and PI(D) for control of Flux & Torque

The Clarke transformation (CLARKE) maps three motor phase currents ( $I_U, I_V, I_W$ ) to a two dimensional coordinate system with two currents ( $I_{\alpha}, I_{\beta}$ ). Based on the actual rotor angle determined by an encoder or via sensorless techniques, the Park transformation (PARK) maps these two currents to a quasi-static coordinate system with two currents ( $I_D, I_Q$ ). The current  $I_D$  represents flux and the current  $I_Q$  corresponds to the torque. The flux just pulls on the rotor and effects the torque constant. The torque is effected by  $I_Q$ . Two PI controllers determine two voltages ( $U_D, U_Q$ ) to drive desired currents for a target torque and a target flux of zero. The determined voltages ( $U_D, U_Q$ ) are re-transformed into the stator system by the inverse Parke transformation (iPARK). The inverse Clarke Transformation (iCLARKE) transforms these two currents into three voltages ( $U_U, U_V, U_W$ ). These three voltage are the input of the PWM engine to drive the power stage.

In case of the FOC2, Clarke transformation CLARKE and inverse Clarke Transformation iCLARKE are skipped.



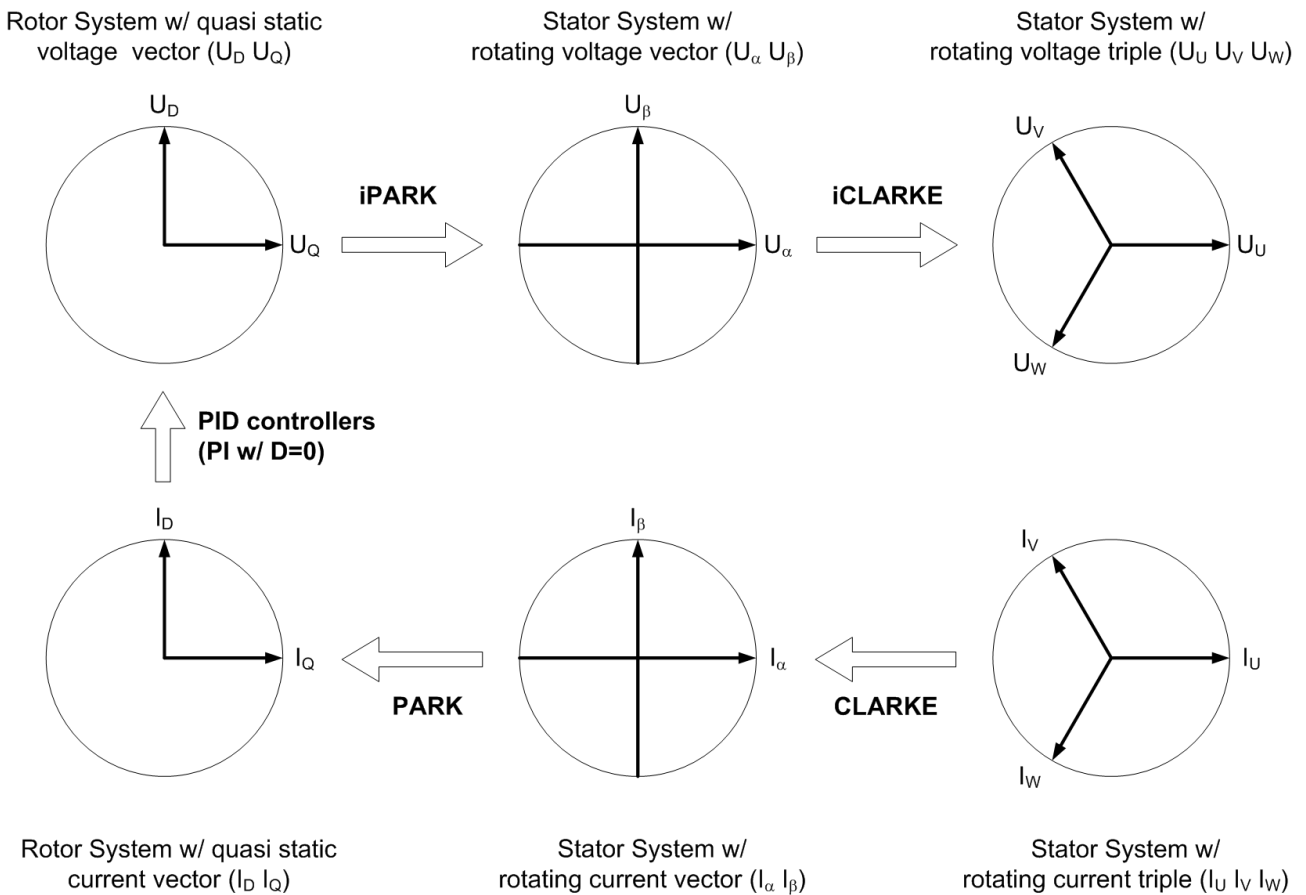


Figure 14: FOC3 Transformations (FOC2 just skips CLARKE and iCLARKE)

### 3.7.9 Motion Modes

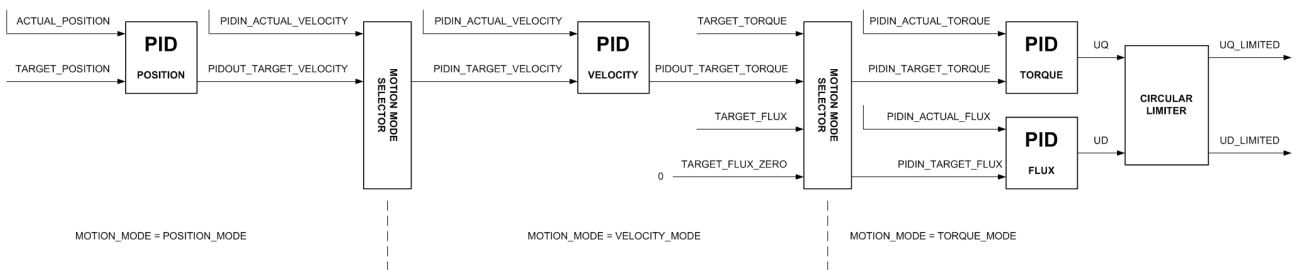


Figure 15: Motion Modes

## 3.8 PWM Engine

The PWM engine takes care of converting voltage vectors to pulse width modulated (PWM) control signals. These digital PWM signals control the gate drivers of the power stage. For detailed description of the PWM control registers and PWM register control bits pls. refer section 5 page 27.

The ease-of-use PWM Engine requires just a couple of parameter settings. Primarily, the polarities for the gate control signal of high side and low side must be set. The power on default PWM mode is 0 that





means PWM = OFF. For operation, the centered PWM mode must be set on by setting the PWM mode to 7. A single bit controls the Space vector PWM (SVPWM). For 3-phase PMSM the SVPWM = ON gives more effective voltage. Nevertheless, for some applications it makes sense to switch the SVPWM = OFF to keep the star point voltage of the motor almost at rest.

### 3.8.1 PWM Polarities

The PWM polarities register PWM\_POLARITIES controls the polarities of the control signals. Positive polarity for gate control means 1 represents ON and 0 represents OFF. The gate control signal polarities are individually programmable for high side gate control and for low side gate control. The PWM polarities register controls the polarity of other control signals as well.

### 3.8.2 PWM frequency

The PWM counter maximum length register PWM\_MAXCNT controls the PWM frequency. For a clock frequency  $f_{CLK} = 25 \text{ MHz}$ , the PWM frequency  $f_{PWM}[\text{Hz}]$  is  $(4.0 * f_{CLK}[\text{Hz}]) / (\text{PWM\_MAXCNT} + 1)$ . With  $f_{CLK} = 25 \text{ MHz}$  and power-on reset (POR) default of PWM\_MAXCNT = 3999 the PWM frequency is  $f_{PWM} = 25 \text{ kHz}$ . The PWM frequency  $f_{PWM}$  is recommended to be in the range of 25 kHz to 200 kHz by setting PWM\_MAXCNT between 3999 to 499.

---

**Note** The PWM frequency can be changed any time also during motion.

---

### 3.8.3 PWM Resolution

The base resolution of the PWM is 12 bit internally mapped to 16 bit range. MAX\_PWMCNT=4095 gives the full resolution of 12 bit with  $\approx 25 \text{ kHz}$  w/  $f_{CLK} = 25 \text{ MHz}$ . MAX\_PWMCNT = 2047 results in 11 bit resolution but with  $\approx 50 \text{ kHz}$  w/  $f_{CLK} = 25 \text{ MHz}$ . So the PWM\_MAXCNT defines the PWM frequency but effects the resolution of the PWM.

### 3.8.4 PWM Modes

The power-on reset (POR) default of the PWM is OFF. The standard PWM scheme is the centered PWM. Passive Breaking and Free Wheeling Modes are available on demand. Please refer [?] concerning the settings.

### 3.8.5 Brake-Before-Make (BBM)

One register controls BBM time for the high side. One register controls BBM time for the low side. The BBM times are programmable in 10 ns steps. The BBM time can be set to zero for gate drivers that have their own integrated BBM timers.



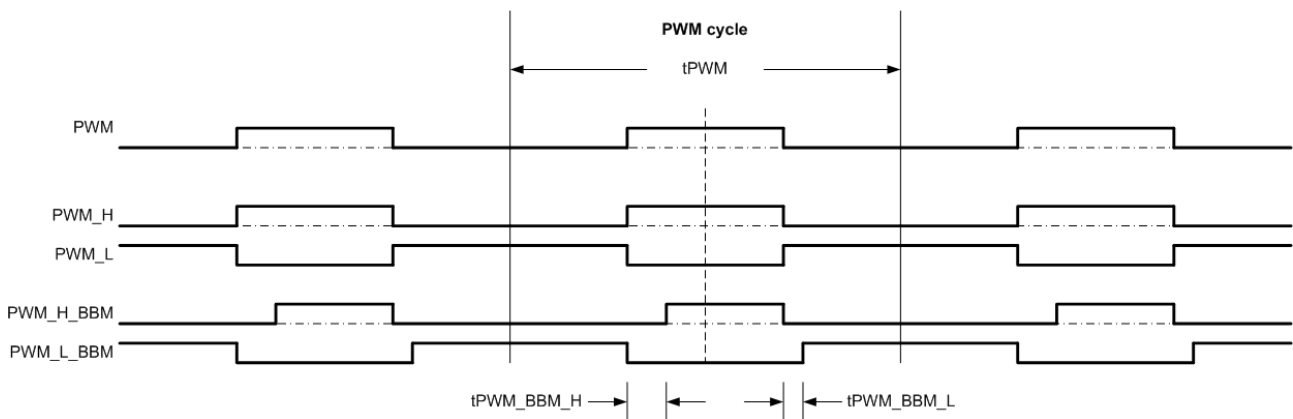


Figure 16: BBM Timing

### Info

Measured BBM times at MOS-FET gates differs from programmed BBM times due to driver delays and possible additional gate driver BBM times. The programmed BBM times are for the digital control signals.

### Note

Too small BBM times cause electrical short of the MOS-FET bridges - so called shoot through - that shorts the power supply and might damage the power stage and the power supply.

### 3.8.6 Space Vector PWM (SVPWM)

A single bit controls the internal Space vector PWM (SVPWM) enable. No further settings are required for the space vector PWM - just ON or OFF. The power on default for the SVPM is OFF.

### Note

The SVPWM is for 3-phase motors only. For 2-phase motors there is no SVPWM.

## 4 Safety Functions

Different safety functions are integrated and mapped to status bits. Two programmable mask register select those bits for WARNING or ERROR. Warning just indicated the warning status at the WARNING output. An Error will cause programmable actions on error conditions and indicate the error status directly at ERROR output, as PWM = OFF on over current condition or ADC raw values with permanent zero or ADC raw values at maximum.

### Info

Programmable autonomous error handling or warning handling is not available.

Internal hardware limiters for real time clipping and monitoring of interim values are available. LIMIT or LIMITS is part of register names of registers associated to internal limiters. Please refer 5.



## 5 Register Map

Register Map for TMC4670						
Address	Registername				Access	
0x00 <sub>h</sub>	CHIPINFO_DATA				R	
	Variant 0					
	Mask	Name		Type		
	0xFFFFFFFF <sub>h</sub>	SI_TYPE				ASCII
		Min	Max	Default		Unit
		0	4294967295	0		
	Hardware type (ASCII).					
	Variant 1					
	Mask	Name		Type		
	0xFFFFFFFF <sub>h</sub>	SI_VERSION				Version
		Min	Max	Default		Unit
		0	4294967295	0		
	Hardware version (u16.u16).					
	Variant 2					
	Mask	Name		Type		
	0xFFFFFFFF <sub>h</sub>	SI_DATE				Date
		Min	Max	Default		Unit
0		4294967295	0			
Hardware date (nibble wise date stamp yyymmdd).						
Variant 3						
Mask	Name		Type			
0xFFFFFFFF <sub>h</sub>	SI_TIME			Time		
	Min	Max	Default	Unit		
	0	16777215	0			
Hardware time (nibble wise time stamp -hhmmss)						
Variant 4						
Mask	Name		Type			
0xFFFFFFFF <sub>h</sub>	SI_VARIANT			Unsigned		
	Min	Max	Default	Unit		
	0	4294967295	0			
0x01 <sub>h</sub>	CHIPINFO_ADDR				RW	
	Mask	Name		Type		



Address	Registername				Access	
0x000000FF <sub>h</sub>	CHIP_INFO_ADDRESS			Choice		
	Min	Max	Default	Unit		
	0	4	0			
	0: SI_TYPE					
	1: SI_VERSION					
2: SI_DATE						
3: SI_TIME						
4: SI_VARIANT						
0x02 <sub>h</sub>	ADC_RAW_DATA				R	
	Variant 0					
	Mask	Name		Type		
	0x0000FFFF <sub>h</sub>	ADC_I_UX_RAW				Unsigned
		Min	Max	Default		Unit
		0	65535	0		
	Raw phase current U resp. X (LTC2351).					
	Mask	Name		Type		
	0xFFFF0000 <sub>h</sub>	ADC_I_WY_RAW				Unsigned
		Min	Max	Default		Unit
		0	65535	0		
	Raw phase current W resp. Y (LTC2351).					
	Variant 1					
	Mask	Name		Type		
	0x0000FFFF <sub>h</sub>	ADC_I_U_RAW				Unsigned
		Min	Max	Default		Unit
		0	65535	0		
	Raw phase current U (ADC_I_UX analog input for FOC2).					
	Mask	Name		Type		
	0xFFFF0000 <sub>h</sub>	ADC_I_V_RAW				Unsigned
		Min	Max	Default		Unit
		0	65535	0		
	Raw phase current V (ADC_I_WY analog input for FOC2).					
Variant 2						
Mask	Name		Type			
0x0000FFFF <sub>h</sub>	ADC_I_B_RAW			Unsigned		
	Min	Max	Default	Unit		



Address	Registername				Access
	0	65535	0		
	Raw phase current Bottom (analog input).				
	Variant 3				
	Mask	Name		Type	
	0x0000FFFF <sub>h</sub>	ADC_VM_RAW		Unsigned	
		Min	Max	Default	Unit
		0	65535	0	
	Raw supply voltage.				
	Variant 4				
	Mask	Name		Type	
	0x0000FFFF <sub>h</sub>	ADC_T_MOSFETS_RAW		Unsigned	
		Min	Max	Default	Unit
		0	65535	0	
	Raw mosfet temperature.				
	Mask	Name		Type	
	0xFFFF0000 <sub>h</sub>	ADC_T_MOTOR_RAW		Unsigned	
		Min	Max	Default	Unit
		0	65535	0	
	Raw motor temperature.				
	Variant 5				
	Mask	Name		Type	
	0x0000FFFF <sub>h</sub>	ADC_U_UX_RAW		Unsigned	
		Min	Max	Default	Unit
		0	65535	0	
	Raw voltage terminal U resp. X.				
	Mask	Name		Type	
	0xFFFF0000 <sub>h</sub>	ADC_U_WY_RAW		Unsigned	
		Min	Max	Default	Unit
		0	65535	0	
	Raw voltage terminal W resp. Y.				
	Variant 6				
	Mask	Name		Type	
	0x0000FFFF <sub>h</sub>	ADC_U_V_RAW		Unsigned	
		Min	Max	Default	Unit
		0	65535	0	



Address	Registername				Access
	Raw voltage terminal V.				
	Variant 7				
	Mask	Name		Type	
	0x0000FFFF <sub>h</sub>	AENC_UX_RAW		Unsigned	
		Min	Max	Default	Unit
		0	65535	0	
	Raw analog encoder voltage U resp. X.				
	Mask	Name		Type	
	0xFFFF0000 <sub>h</sub>	AENC_WY_RAW		Unsigned	
		Min	Max	Default	Unit
		0	65535	0	
	Raw analog encoder voltage W resp. Y.				
	Variant 8				
	Mask	Name		Type	
	0x0000FFFF <sub>h</sub>	AENC_V_RAW		Unsigned	
		Min	Max	Default	Unit
		0	65535	0	
	Raw analog encoder voltage V.				
	Mask	Name		Type	
	0xFFFF0000 <sub>h</sub>	AENC_N_RAW		Unsigned	
		Min	Max	Default	Unit
		0	65535	0	
	Raw analog encoder zero position voltage N.				
	Variant 9				
	Mask	Name		Type	
	0x0000FFFF <sub>h</sub>	ANALOG_GPI_RAW		Unsigned	
		Min	Max	Default	Unit
		0	65535	0	
	Raw analog input voltage.				
	Variant 10				
	Mask	Name		Type	
	0x0000FFFF <sub>h</sub>	ADCSD_I_UX_RAW		Signed	
		Min	Max	Default	Unit
		-32768	32767	0	
	SigmaDeltaADC current UX.				



Address	Registername				Access	
	Mask	Name			Type	
	0xFFFF0000 <sub>h</sub>	ADCSD_I_WY_RAW			Signed	
		Min	Max	Default	Unit	
		-32768	32767	0		
	SigmaDeltaADC current WY.					
	Variant 11					
	Mask	Name			Type	
0x0000FFFF <sub>h</sub>	ADCSD_I_B_RAW			Signed		
	Min	Max	Default	Unit		
	-32768	32767	0			
SigmaDeltaADC current at bottom (PCB S (sense) instead of B (bottom)).						
0x03 <sub>h</sub>	ADC_RAW_ADDR				RW	
	Mask	Name				Type
	0x000000FF <sub>h</sub>	ADC_RAW_ADDR				Choice
		Min	Max	Default		Unit
		0	11	0		
0: ADC_I_WY_RAW & ADC_I_UX_RAW 1: ADC_I_V_RAW & ADC_I_U_RAW 2: ADC_I_B_RAW 3: ADC_VM_RAW 4: ADC_T_MOSFETS_RAW & ADC_T_MOTOR_RAW 5: ADC_U_WY_RAW & ADC_U_UX_RAW 6: ADC_U_V_RAW 7: AENC_WY_RAW & AENC_UX_RAW 8: AENC_N_RAW & AENC_V_RAW 9: ANALOG_GPI_RAW 10: ADCSD_I_WY_RAW & ADCSD_I_UX_RAW 11: ADCSD_I_B_RAW						
0x04 <sub>h</sub>	ADCSD_CLKCFG				RW	
	Mask	Name				Type
	0x0000FFFF <sub>h</sub>	ADCSD_CLKCFG				Unsigned
		Min	Max	Default		Unit
	0	65535	0			
0x06 <sub>h</sub>	ADC_I1_IO_EXT				RW	



Address	Registername				Access	
	Mask	Name		Type		
	0x0000FFFF <sub>h</sub>	ADC_I0_EXT		Unsigned		
		Min	Max	Default		Unit
		0	65535	0		
	Register for write of ADC_I0 value from external source (eg. CPU).					
	Mask	Name		Type		
	0xFFFF0000 <sub>h</sub>	ADC_I1_EXT		Unsigned		
		Min	Max	Default		
0		65535	0			
Register for write of ADC_I1 value from external source (eg. CPU).						
0x08 <sub>h</sub>	ADC_I1_SCALE_OFFSET				RW	
	Mask	Name		Type		
	0x0000FFFF <sub>h</sub>	ADC_I1_OFFSET		Signed		
		Min	Max	Default		Unit
		-32768	32767	0		
	Offset for current ADC channel 1.					
	Mask	Name		Type		
	0xFFFF0000 <sub>h</sub>	ADC_I1_SCALE		Signed		
Min		Max	Default	Unit		
-32768		32767	0			
Scaling factor for current ADC channel 1.						
0x09 <sub>h</sub>	ADC_I0_SCALE_OFFSET				RW	
	Mask	Name		Type		
	0x0000FFFF <sub>h</sub>	ADC_I0_OFFSET		Signed		
		Min	Max	Default		Unit
		-32768	32767	0		
	Offset for current ADC channel 0.					
	Mask	Name		Type		
	0xFFFF0000 <sub>h</sub>	ADC_I0_SCALE		Signed		
Min		Max	Default	Unit		
-32768		32767	0			
Scaling factor for current ADC channel 0.						
0x0A <sub>h</sub>	ADC_I_SELECT				RW	





Address	Registername				Access
	Mask	Name			Type
	0x000000FF <sub>h</sub>	ADC_I0_SELECT			Choice
		Min	Max	Default	Unit
		0	7	0	
		Select input for raw current ADC_I0_RAW.			
		0: ADC_I0_RAW (analog input ADC_I_U)			
		1: ADC_I1_RAW (analog input ADC_I_V)			
		2: ADC_I0_EXT (from register)			
		3: ADC_I1_EXT (from register)			
		4: ADCSD_I0_RAW (sigma delta ADC)			
		5: ADCSD_I1_RAW (sigma delta ADC)			
		6: ADCSPI_I0_RAW (SPI ADC_I_UX)			
		7: ADCSPI_I1_RAW (SPI ADC_I_WY)			
	Mask	Name			Type
	0x0000FF00 <sub>h</sub>	ADC_I1_SELECT			Choice
		Min	Max	Default	Unit
		0	7	0	
		Select input for raw current ADC_I1_RAW.			
		0: ADC_I0_RAW (analog input ADC_I_U)			
		1: ADC_I1_RAW (analog input ADC_I_V)			
		2: ADC_I0_EXT (from register)			
		3: ADC_I1_EXT (from register)			
		4: ADCSD_I0_RAW (sigma delta ADC)			
		5: ADCSD_I1_RAW (sigma delta ADC)			
		6: ADCSPI_I0_RAW (SPI ADC_I_UX)			
		7: ADCSPI_I1_RAW (SPI ADC_I_WY)			
	Mask	Name			Type
	0x03000000 <sub>h</sub>	ADC_I_UX_SELECT			Choice
		Min	Max	Default	Unit
		0	2	0	
		0: UX = ADC_I0 (default)			
		1: UX = ADC_I1			
		2: UX = ADC_I2			
	Mask	Name			Type
	0x0C000000 <sub>h</sub>	ADC_I_V_SELECT			Choice



Address	Registername					Access
		Min	Max	Default	Unit	
		0	2	0		
		0: V = ADC_I0 1: V = ADC_I1 (default) 2: V = ADC_I2				
	Mask	Name			Type	
	0x30000000 <sub>h</sub>	ADC_I_WY_SELECT			Choice	
		Min	Max	Default	Unit	
		0	2	0		
		0: WY = ADC_I0 1: WY = ADC_I1 2: WY = ADC_I2 (default)				
0x0D <sub>h</sub>	AENC_0_SCALE_OFFSET					RW
	Mask	Name			Type	
	0x0000FFFF <sub>h</sub>	AENC_0_OFFSET			Signed	
		Min	Max	Default	Unit	
		-32768	32767	0		
		Offset for Analog Encoder ADC channel 0.				
	Mask	Name			Type	
	0xFFFF0000 <sub>h</sub>	AENC_0_SCALE			Signed	
		Min	Max	Default	Unit	
		-32768	32767	0		
		Scaling factor for Analog Encoder ADC channel 0.				
0x0E <sub>h</sub>	AENC_1_SCALE_OFFSET					RW
	Mask	Name			Type	
	0x0000FFFF <sub>h</sub>	AENC_1_OFFSET			Signed	
		Min	Max	Default	Unit	
		-32768	32767	0		
		Offset for Analog Encoder ADC channel 1.				
	Mask	Name			Type	
	0xFFFF0000 <sub>h</sub>	AENC_1_SCALE			Signed	
		Min	Max	Default	Unit	
		-32768	32767	0		
		Scaling factor for Analog Encoder ADC channel 1.				
0x0F <sub>h</sub>	AENC_2_SCALE_OFFSET					RW



Address	Registername				Access	
	Mask	Name			Type	
	0x0000FFFF <sub>h</sub>	AENC_2_OFFSET			Signed	
		Min	Max	Default	Unit	
		-32768	32767	0		
	Offset for Analog Encoder ADC channel 2.					
	Mask	Name			Type	
	0xFFFF0000 <sub>h</sub>	AENC_2_SCALE			Signed	
		Min	Max	Default	Unit	
-32768		32767	0			
Scaling factor for Analog Encoder ADC channel 2.						
0x10 <sub>h</sub>	AENC_3_SCALE_OFFSET				RW	
	Mask	Name				Type
	0x0000FFFF <sub>h</sub>	AENC_3_OFFSET				Signed
		Min	Max	Default		Unit
		-32768	32767	0		
	Offset for Analog Encoder ADC channel 3.					
	Mask	Name				Type
	0xFFFF0000 <sub>h</sub>	AENC_3_SCALE				Signed
Min		Max	Default	Unit		
-32768		32767	0			
Scaling factor for Analog Encoder ADC channel 3.						
0x11 <sub>h</sub>	AENC_SELECT				RW	
	Mask	Name				Type
	0x000000FF <sub>h</sub>	AENC_0_SELECT				Choice
		Min	Max	Default		Unit
		0	3	0		
	Select analog encoder ADC channel for raw analog encoder signal AENC_0_RAW. 0: AENC_UX_RAW (default) 1: AENC_VN_RAW 2: AENC_WY_RAW 3: AENC_N_RAW					
	Mask	Name				Type
	0x0000FF00 <sub>h</sub>	AENC_1_SELECT				Choice
Min		Max	Default	Unit		



Address	Registername				Access
	0	3	0		
	Select analog encoder ADC channel for raw analog encoder signal AENC_1_RAW. 0: AENC_UX_RAW 1: AENC_VN_RAW (default) 2: AENC_WY_RAW 3: AENC_N_RAW				
	Mask	Name		Type	
	0x00FF0000 <sub>h</sub>	AENC_2_SELECT		Choice	
		Min	Max	Default	Unit
		0	3	0	
	Select analog encoder ADC channel for raw analog encoder signal AENC_2_RAW. 0: AENC_UX_RAW 1: AENC_VN_RAW 2: AENC_WY_RAW (default) 3: AENC_N_RAW				
	Mask	Name		Type	
	0xFF000000 <sub>h</sub>	AENC_3_SELECT		Choice	
		Min	Max	Default	Unit
		0	3	0	
	Select analog encoder ADC channel for raw analog encoder signal AENC_3_RAW. 0: AENC_UX_RAW 1: AENC_VN_RAW 2: AENC_WY_RAW 3: AENC_N_RAW (default)				
0x12 <sub>h</sub>	ADC_IWY_IUX				R
	Mask	Name		Type	
	0x0000FFFF <sub>h</sub>	ADC_IUX		Signed	
		Min	Max	Default	Unit
		-32768	32767	0	
	Register of scaled current ADC value including signed added offset as input for the FOC.				
	Mask	Name		Type	
	0xFFFF0000 <sub>h</sub>	ADC_IWY		Signed	



Address	Registername					Access
	Min	Max	Default	Unit		
	-32768	32767	0			
	Register of scaled current ADC value including signed added offset as input for the FOC.					
0x13 <sub>h</sub>	ADC_IV					R
	Mask	Name			Type	
	0x0000FFFF <sub>h</sub>	ADC_IV			Signed	
	Min	Max	Default	Unit		
	-32768	32767	0			
	Register of scaled current ADC value including signed added offset as input for the FOC.					
0x15 <sub>h</sub>	AENC_WY_UX					R
	Mask	Name			Type	
	0x0000FFFF <sub>h</sub>	AENC_UX			Signed	
	Min	Max	Default	Unit		
	-32768	32767	0			
	Register of scaled analog encoder value including signed added offset as input for the interpolator.					
	Mask	Name			Type	
	0xFFFF0000 <sub>h</sub>	AENC_WY			Signed	
	Min	Max	Default	Unit		
-32768	32767	0				
	Register of scaled analog encoder value including signed added offset as input for the interpolator.					
0x16 <sub>h</sub>	AENC_N_VN					R
	Mask	Name			Type	
	0x0000FFFF <sub>h</sub>	AENC_VN			Signed	
	Min	Max	Default	Unit		
	-32768	32767	0			
	Register of scaled analog encoder value including signed added offset as input for the interpolator.					
	Mask	Name			Type	
	0xFFFF0000 <sub>h</sub>	AENC_N			Signed	
	Min	Max	Default	Unit		
-32768	32767	0				
	Register of scaled analog encoder value including signed added offset of analog N pulse.					



Address	Registername				Access	
0x17 <sub>h</sub>	PWM_POLARITIES				RW	
	Mask	Name		Type		
	0x00000001 <sub>h</sub>	PWM_POLARITIES[0]				Bool
		Min	Max	Default		Unit
		0	1	0		
	polarity of Low Side (LS) gate control signal 0: off 1: on					
	Mask	Name		Type		
	0x00000002 <sub>h</sub>	PWM_POLARITIES[1]				Bool
		Min	Max	Default		Unit
		0	1	0		
	polarity of High Side (HS) gate control signal 0: off 1: on					
	Mask	Name		Type		
	0x00000004 <sub>h</sub>	PWM_POLARITIES[2]				Bool
		Min	Max	Default		Unit
		0	1	0		
pulse AB polarity 0: off 1: on						
Mask	Name		Type			
0x00000008 <sub>h</sub>	PWM_POLARITIES[3]			Bool		
	Min	Max	Default	Unit		
	0	1	0			
pulse B polarity 0: off 1: on						
Mask	Name		Type			
0x00000010 <sub>h</sub>	PWM_POLARITIES[4]			Bool		
	Min	Max	Default	Unit		
	0	1	0			
pulse C center polarity 0: off						



Address	Registername				Access
		1: on			
	Mask	Name		Type	
	0x00000020 <sub>h</sub>	PWM_POLARITIES[5]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		pulse A polarity			
		0: off			
		1: on			
	Mask	Name		Type	
	0x00000040 <sub>h</sub>	PWM_POLARITIES[6]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		pulse zero Z pulse polarity			
		0: off			
		1: on			
	Mask	Name		Type	
	0x00000080 <sub>h</sub>	PWM_POLARITIES[7]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		over current signal polarity			
		0: off			
		1: on			
0x18 <sub>h</sub>	PWM_MAXCNT				RW
	Mask	Name		Type	
	0x0000FFFF <sub>h</sub>	PWM_MAXCNT		Unsigned	
		Min	Max	Default	Unit
		0	65535	0	
		PWM maximum (count-1), PWM frequency is $f_{PWM}[Hz] = 100MHz/(PWM\_MAXCNT+1)$			
0x19 <sub>h</sub>	PWM_BBM_H_BBM_L				RW
	Mask	Name		Type	
	0x000000FF <sub>h</sub>	PWM_BBM_L		Unsigned	
		Min	Max	Default	Unit
		0	255	0	



Address	Registername				Access	
		Break Before Make time tBBM_L[10ns] for low side MOS-FET gate control				
	Mask	Name		Type		
	0x0000FF00 <sub>h</sub>	PWM_BBM_H		Unsigned		
		Min	Max	Default		Unit
		0	255	0		
	Break Before Make time tBBM_H[10ns] for high side MOS-FET gate control					
0x1A <sub>h</sub>	PWM_SV_CHOP				RW	
	Mask	Name		Type		
	0x000000FF <sub>h</sub>	PWM_CHOP		Choice		
		Min	Max	Default		Unit
		0	7	0		
	PWM chopper mode, defining how to chopper					
	0: PWM = OFF, free running					
	1: PWM = OFF, Low Side (LS) permanent = ON					
	2: PWM = OFF, High Side (HS) permanent = ON					
	3: PWM off, free running					
4: PWM off, free running						
5: PWM low side (LS) chopper only, high side (HS) off; not suitable for FOC						
6: PWM high side (HS) chopper only, low side (LS) off; not suitable for FOC						
7: centered PWM for FOC						
Mask	Name		Type			
0x00000100 <sub>h</sub>	PWM_SV		Bool			
	Min	Max	Default	Unit		
	0	1	0			
use Space Vector PWM						
0: Space Vector PWM disabled						
1: Space Vector PWM enabled						
0x1B <sub>h</sub>	MOTOR_TYPE_N_POLE_PAIRS				RW	
	Mask	Name		Type		
	0x0000FFFF <sub>h</sub>	N_POLE_PAIRS		Unsigned		
		Min	Max	Default		Unit
		1	65535	1		





Address	Registername				Access	
		Number n of pole pairs of the motor for calculation $\phi_e = \phi_m / N\_POLE\_PAIRS$ .				
	Mask	Name		Type		
	0x00FF0000 <sub>h</sub>	MOTOR_TYPE				Choice
		Min	Max	Default		Unit
	0	3	0			
	0: FOC2 1: FOC3 2: reserved 3: reserved					
0x1C <sub>h</sub>		PHI_E_EXT			RW	
	Mask	Name		Type		
	0x0000FFFF <sub>h</sub>	PHI_E_EXT				Signed
		Min	Max	Default		Unit
	-32768	32767	0			
	Electrical angle $\phi_{e\_ext}$ for external writing into this register.					
0x1D <sub>h</sub>		PHI_M_EXT			RW	
	Mask	Name		Type		
	0x0000FFFF <sub>h</sub>	PHI_M_EXT				Signed
		Min	Max	Default		Unit
	-32768	32767	0			
	Mechanical angle $\phi_{m\_ext}$ for external writing into this register.					
0x1E <sub>h</sub>		POSITION_EXT			RW	
	Mask	Name		Type		
	0xFFFFFFFF <sub>h</sub>	POSITION_EXT				Signed
		Min	Max	Default		Unit
	-2147483648	2147483647	0			
	Mechanical (multi turn) position for external writing into this register.					
0x1F <sub>h</sub>		OPENLOOP_MODE			RW	
	Mask	Name		Type		
	0x00001000 <sub>h</sub>	OPENLOOP_PHI_DIRECTION				Bool
		Min	Max	Default		Unit
	0	1	0			
	Open loop phi direction.					



Address	Registername				Access	
		0: positive 1: negative				
0x20 <sub>h</sub>	OPENLOOP_ACCELERATION				RW	
	Mask	Name		Type		
	0xFFFFFFFF <sub>h</sub>	OPENLOOP_ACCELERATION				Signed
		Min	Max	Default		Unit
0	4294967295	0		Acceleration of open loop phi.		
0x21 <sub>h</sub>	OPENLOOP_VELOCITY_TARGET				RW	
	Mask	Name		Type		
	0xFFFFFFFF <sub>h</sub>	OPENLOOP_VELOCITY_TARGET				Signed
		Min	Max	Default		Unit
-2147483648	2147483647	0		Target velocity of open loop phi.		
0x22 <sub>h</sub>	OPENLOOP_VELOCITY_ACTUAL				RW	
	Mask	Name		Type		
	0xFFFFFFFF <sub>h</sub>	OPENLOOP_VELOCITY_ACTUAL				Signed
		Min	Max	Default		Unit
-2147483648	2147483647	0		Actual velocity of open loop generator.		
0x23 <sub>h</sub>	OPENLOOP_PHI				RW	
	Mask	Name		Type		
	0x0000FFFF <sub>h</sub>	OPENLOOP_PHI				Signed
		Min	Max	Default		Unit
-32768	32767	0		Angle phi open loop (either mapped to electrical angel phi <sub>e</sub> or mechanical angle phi <sub>m</sub> ).		
0x24 <sub>h</sub>	UQ_UD_EXT				RW	
	Mask	Name		Type		
	0x0000FFFF <sub>h</sub>	UD_EXT				Signed
		Min	Max	Default		Unit
	-32768	32767	0			External writable parameter for open loop voltage control mode, usefull during system setup, U_D component.
Mask	Name		Type			



Address	Registername				Access	
0xFFFF0000 <sub>h</sub>	UQ_EXT			Signed		
	Min	Max	Default	Unit		
	-32768	32767	0			
External writable parameter for open loop voltage control mode, usefull during system setup, U_Q component.						
0x25 <sub>h</sub>	ABN_DECODER_MODE				RW	
	Mask	Name		Type		
	0x00000001 <sub>h</sub>	apol				Bool
		Min	Max	Default		Unit
		0	1	0		
	Polarity of A pulse. 0: off 1: on					
	Mask	Name		Type		
	0x00000002 <sub>h</sub>	bpol				Bool
		Min	Max	Default		Unit
		0	1	0		
	Polarity of B pulse. 0: off 1: on					
	Mask	Name		Type		
	0x00000004 <sub>h</sub>	npol				Bool
		Min	Max	Default		Unit
		0	1	0		
	Polarity of N pulse. 0: off 1: on					
	Mask	Name		Type		
0x00000008 <sub>h</sub>	use_abn_as_n			Bool		
	Min	Max	Default	Unit		
	0	1	0			
0: Ignore A and B polarity with Npulse = N, 1 : Npulse = N and A and B 0: Ignore A and B polarity with Npulse = N 1: Npulse = N and A and B						
Mask	Name		Type			



Address	Registername				Access	
0x00000100 <sub>h</sub>	cIn			Bool		
	Min	Max	Default	Unit		
	0	1	0			
	Clear writes ABN_DECODER_COUNT_N into decoder count at Npulse. 0: off 1: on					
	Mask	Name		Type		
0x00001000 <sub>h</sub>	direction			Bool		
	Min	Max	Default	Unit		
	0	1	0			
	Decoder count direction. 0: positive 1: negative					
0x26 <sub>h</sub>	ABN_DECODER_PPR				RW	
	Mask	Name		Type		
	0x00FFFFFF <sub>h</sub>	ABN_DECODER_PPR				Unsigned
		Min	Max	Default		Unit
0	16777215	0				
Decoder pules per mechanical revolution.						
0x27 <sub>h</sub>	ABN_DECODER_COUNT				RW	
	Mask	Name		Type		
	0x00FFFFFF <sub>h</sub>	ABN_DECODER_COUNT				Unsigned
		Min	Max	Default		Unit
0	16777215	0				
Raw decoder count; the digital decoder engine counts modulo (decoder_ppr).						
0x28 <sub>h</sub>	ABN_DECODER_COUNT_N				RW	
	Mask	Name		Type		
	0x00FFFFFF <sub>h</sub>	ABN_DECODER_COUNT_N				Unsigned
		Min	Max	Default		Unit
0	16777215	0				
Decoder count latched on N pulse, when N pulse clears decoder_count also decoder_count_n is 0.						
0x29 <sub>h</sub>	ABN_DECODER_PHI_E_PHI_M_OFFSET				RW	
	Mask	Name		Type		



Address	Registername				Access	
	0x0000FFFF <sub>h</sub>	ABN_DECODER_PHI_M_OFFSET			Signed	
		Min	Max	Default	Unit	
		-32768	32767	0		
	ABN_DECODER_PHI_M_OFFSET to shift (rotate) angle DECODER_PHI_M.					
	Mask	Name			Type	
	0xFFFF0000 <sub>h</sub>	ABN_DECODER_PHI_E_OFFSET			Signed	
Min		Max	Default	Unit		
-32768		32767	0			
ABN_DECODER_PHI_E_OFFSET to shift (rotate) angle DECODER_PHI_E.						
0x2A <sub>h</sub>	ABN_DECODER_PHI_E_PHI_M				R	
	Mask	Name				Type
	0x0000FFFF <sub>h</sub>	ABN_DECODER_PHI_M				Signed
		Min	Max	Default		Unit
		-32768	32767	0		
	ABN_DECODER_PHI_M = ABN_DECODER_COUNT * 2 <sup>16</sup> / ABN_DECODER_PPR + ABN_DECODER_PHI_M_OFFSET;					
	Mask	Name				Type
	0xFFFF0000 <sub>h</sub>	ABN_DECODER_PHI_E				Signed
Min		Max	Default	Unit		
-32768		32767	0			
ABN_DECODER_PHI_E = (ABN_DECODER_PHI_M * N_POLE_PAIRS_) + ABN_DECODER_PHI_E_OFFSET						
0x2C <sub>h</sub>	ABN_2_DECODER_MODE				RW	
	Mask	Name				Type
	0x00000001 <sub>h</sub>	apol				Bool
		Min	Max	Default		Unit
		0	1	0		
	Polarity of A pulse. 0: off 1: on					
	Mask	Name				Type
0x00000002 <sub>h</sub>	bpol			Bool		
	Min	Max	Default	Unit		
	0	1	0			



Address	Registername				Access
		Polarity of B pulse. 0: off 1: on			
	Mask	Name		Type	
	0x00000004 <sub>h</sub>	npol		Bool	
		Min	Max	Default	Unit
		0	1	0	
		Polarity of N pulse. 0: off 1: on			
	Mask	Name		Type	
	0x00000008 <sub>h</sub>	use_abn_as_n		Bool	
		Min	Max	Default	Unit
		0	1	0	
		0: Ignore A and B polarity with Npulse = N, 1 : Npulse = N and A and B 0: Ignore A and B polarity with Npulse = N 1: Npulse = N and A and B			
	Mask	Name		Type	
	0x00000100 <sub>h</sub>	cln		Bool	
		Min	Max	Default	Unit
		0	1	0	
		Clear writes ABN_2_DECODER_COUNT_N into decoder count at Npulse. 0: off 1: on			
	Mask	Name		Type	
	0x00001000 <sub>h</sub>	direction		Bool	
		Min	Max	Default	Unit
		0	1	0	
		Decoder count direction. 0: positive 1: negative			
0x2D <sub>h</sub>	ABN_2_DECODER_PPR				RW
	Mask	Name		Type	
	0x00FFFFFF <sub>h</sub>	ABN_2_DECODER_PPR		Unsigned	



Address	Registername					Access
	Min	Max	Default	Unit		
	1	16777215	0			
	Decoder_2 pules per mechanical revolution. This 2nd ABN encoder interface is for positioning or velocity control but NOT for motor commutation.					
0x2E <sub>h</sub>	ABN_2_DECODER_COUNT					RW
	Mask	Name			Type	
	0x00FFFFFF <sub>h</sub>	ABN_2_DECODER_COUNT			Unsigned	
	Min	Max	Default	Unit		
	0	16777215	0			
	Raw decoder_2 count; the digital decoder engine counts modulo (decoder_2_ppr).					
0x2F <sub>h</sub>	ABN_2_DECODER_COUNT_N					RW
	Mask	Name			Type	
	0x00FFFFFF <sub>h</sub>	ABN_2_DECODER_COUNT_N			Unsigned	
	Min	Max	Default	Unit		
	0	16777215	0			
	Decoder_2 count latched on N pulse, when N pulse clears decoder_2_count also decoder_2_count_n is 0.					
0x30 <sub>h</sub>	ABN_2_DECODER_PHI_M_OFFSET					RW
	Mask	Name			Type	
	0x0000FFFF <sub>h</sub>	ABN_2_DECODER_PHI_M_OFFSET			Signed	
	Min	Max	Default	Unit		
	-32768	32767	0			
	ABN_2_DECODER_PHI_M_OFFSET to shift (rotate) angle DECODER_2_PHI_M.					
0x31 <sub>h</sub>	ABN_2_DECODER_PHI_M					R
	Mask	Name			Type	
	0x0000FFFF <sub>h</sub>	ABN_2_DECODER_PHI_M			Signed	
	Min	Max	Default	Unit		
	-32768	32767	0			
	$ABN\_2\_DECODER\_PHI\_M = ABN\_2\_DECODER\_COUNT * 2^{16} / ABN\_2\_DECODER\_PPR + ABN\_2\_DECODER\_PHI\_M\_OFFSET;$					
0x33 <sub>h</sub>	HALL_MODE					RW
	Mask	Name			Type	
	0x00000001 <sub>h</sub>	polarity			Bool	
	Min	Max	Default	Unit		



Address	Registername				Access
		0	1	0	
		polarity 0: off 1: on			
	Mask	Name		Type	
	0x00000100 <sub>h</sub>	interpolation		Bool	
		Min	Max	Default	Unit
		0	1	0	
		interpolation 0: off 1: on			
	Mask	Name		Type	
	0x00001000 <sub>h</sub>	direction		Bool	
		Min	Max	Default	Unit
		0	1	0	
		direction 0: off 1: on			
0x34 <sub>h</sub>	HALL_POSITION_060_000				RW
	Mask	Name		Type	
	0x0000FFFF <sub>h</sub>	HALL_POSITION_000		Signed	
		Min	Max	Default	Unit
		-32768	32767	0	
		s16 hall sensor position at 0°			
	Mask	Name		Type	
	0xFFFF0000 <sub>h</sub>	HALL_POSITION_060		Signed	
		Min	Max	Default	Unit
		-32768	32767	0	
		s16 hall sensor position at 60°.			
0x35 <sub>h</sub>	HALL_POSITION_180_120				RW
	Mask	Name		Type	
	0x0000FFFF <sub>h</sub>	HALL_POSITION_120		Signed	
		Min	Max	Default	Unit
		-32768	32767	0	
		s16 hall sensor position at 120°.			





Address	Registername				Access	
	Mask	Name		Type		
	0xFFFF0000 <sub>h</sub>	HALL_POSITION_180		Signed		
		Min	Max	Default		Unit
		-32768	32767	0		
s16 hall sensor position at 180°.						
0x36 <sub>h</sub>	HALL_POSITION_300_240				RW	
	Mask	Name		Type		
	0x0000FFFF <sub>h</sub>	HALL_POSITION_240		Signed		
		Min	Max	Default		Unit
		-32768	32767	0		
	s16 hall sensor position at 240°.					
	Mask	Name		Type		
	0xFFFF0000 <sub>h</sub>	HALL_POSITION_300		Signed		
Min		Max	Default	Unit		
-32768		32767	0			
s16 hall sensor position at 300°.						
0x37 <sub>h</sub>	HALL_PHI_E_PHI_M_OFFSET				RW	
	Mask	Name		Type		
	0x0000FFFF <sub>h</sub>	HALL_PHI_M_OFFSET		Signed		
		Min	Max	Default		Unit
		-32768	32767	0		
	Offset of mechanical angle hall_phi_m of hall decoder.					
	Mask	Name		Type		
	0xFFFF0000 <sub>h</sub>	HALL_PHI_E_OFFSET		Signed		
Min		Max	Default	Unit		
-32768		32767	0			
Offset for electrical angle hall_phi_e of hall decoder.						
0x38 <sub>h</sub>	HALL_DPHI_MAX				RW	
	Mask	Name		Type		
	0x0000FFFF <sub>h</sub>	HALL_DPHI_MAX		Unsigned		
		Min	Max	Default		Unit
0	65535	0				
Maximum dx for interpolation (default for digital hall: u16/6).						
0x39 <sub>h</sub>	HALL_PHI_E_INTERPOLATED_PHI_E				R	
	Mask	Name		Type		



Address	Registername				Access	
	0x0000FFFF <sub>h</sub>	HALL_PHI_E			Signed	
		Min	Max	Default	Unit	
		-32768	32767	0		
		Raw electrical angle hall_phi_e of hall decoder, selection programmed via HALL_MODE control bit.				
	Mask	Name			Type	
	0xFFFF0000 <sub>h</sub>	HALL_PHI_E_INTERPOLATED			Signed	
		Min	Max	Default	Unit	
		-32768	32767	0		
		Interpolated electrical angle hall_phi_e_interpolated, selection programmed via HALL_MODE control bit.				
	Mask	Name			Type	
	0x0000FFFF <sub>h</sub>	HALL_PHI_M			Signed	
	0x3A <sub>h</sub>		Min	Max	Default	Unit
		-32768	32767	0		
		Mechanical angle hall_phi_m of hall decoder.				
Mask		Name			Type	
0x0000FFFF <sub>h</sub>		HALL_PHI_M			Signed	
0x3B <sub>h</sub>			Min	Max	Default	Unit
		0	1	0		
		nXY_UVW : 0: SinCos Mode // 1: 0° 120° 240° Mode 0: off 1: on				
	Mask	Name			Type	
	0x00000001 <sub>h</sub>	AENC_DECODER_MODE[0]			Bool	
	0x3C <sub>h</sub>		Min	Max	Default	Unit
		0	1	0		
		decoder count direction 0: positive 1: negative				
Mask		Name			Type	
0x00001000 <sub>h</sub>		AENC_DECODER_MODE[12]			Bool	
0x3C <sub>h</sub>			AENC_DECODER_N_THRESHOLD			RW
	Mask	Name				
	0x0000FFFF <sub>h</sub>	AENC_DECODER_N_THRESHOLD				



Address	Registername					Access
	Min	Max	Default	Unit		
	-32768	32767	0			
	Threshold for generating of N pulse from analog AENC_N signal (only needed for analog SinCos encoders with analog N signal).					
0x3D <sub>h</sub>	AENC_DECODER_PHI_A_RAW					R
	Mask	Name			Type	
	0x0000FFFF <sub>h</sub>	AENC_DECODER_PHI_A_RAW			Signed	
		Min	Max	Default	Unit	
		-32768	32767	0		
	Raw analog angle phi calculated from analog AENC inputs (analog hall, analog SinCos, ...).					
0x3E <sub>h</sub>	AENC_DECODER_PHI_A_OFFSET					RW
	Mask	Name			Type	
	0x0000FFFF <sub>h</sub>	AENC_DECODER_PHI_A_OFFSET			Signed	
		Min	Max	Default	Unit	
		-32768	32767	0		
	Offset for angle phi from analog decoder (analog hall, analog SinCos, ...).					
0x3F <sub>h</sub>	AENC_DECODER_PHI_A					R
	Mask	Name			Type	
	0x0000FFFF <sub>h</sub>	AENC_DECODER_PHI_A			Signed	
		Min	Max	Default	Unit	
		-2147483648	2147483647	0		
	Resulting phi available for the FOC (phi_e might need to be calculated from this angle via aenc_decoder_ppr, for analog hall sensors phi_a might be used directly as phi_e depends on analog hall signal type).					
0x40 <sub>h</sub>	AENC_DECODER_PPR					RW
	Mask	Name			Type	
	0x0000FFFF <sub>h</sub>	AENC_DECODER_PPR			Signed	
		Min	Max	Default	Unit	
		-32768	32767	1		
	Number of periods per revolution also called lines per revolution (different nomenclatur compared to digital ABN encoders).					
0x41 <sub>h</sub>	AENC_DECODER_COUNT					RW
	Mask	Name			Type	
	0xFFFFFFFF <sub>h</sub>	AENC_DECODER_COUNT			Signed	



Address	Registername					Access	
	Min	Max	Default	Unit			
	-2147483648	2147483647	0				
	Decoder position, raw unscaled.						
0x42 <sub>h</sub>	AENC_DECODER_COUNT_N					RW	
	Mask	Name			Type		
	0xFFFFFFFF <sub>h</sub>	AENC_DECODER_COUNT_N			Signed		
	Min	Max	Default	Unit			
	-2147483648	2147483647	0				
	Latched decoder position on analog N pulse event.						
0x43 <sub>h</sub>	AENC_DECODER_COUNT_N_MASK					RW	
	Mask	Name			Type		
	0x0000FFFF <sub>h</sub>	AENC_DECODER_COUNT_N_MASK			Signed		
	Min	Max	Default	Unit			
	-32768	32767	0				
	Optional position mask (position) for the analog N pulse within phi_a period to be and-ed with the digital N pulse generated via aenc_decoder_n_threshold.						
0x45 <sub>h</sub>	AENC_DECODER_PHI_E_PHI_M_OFFSET					RW	
	Mask	Name			Type		
	0x0000FFFF <sub>h</sub>	AENC_DECODER_PHI_M_OFFSET			Signed		
	Min	Max	Default	Unit			
	-32768	32767	0				
		Offset for mechanical angle phi_m.					
	Mask	Name			Type		
0xFFFF0000 <sub>h</sub>	AENC_DECODER_PHI_E_OFFSET			Signed			
Min	Max	Default	Unit				
-32768	32767	0					
	Offset for electrical angle phi_e.						
0x46 <sub>h</sub>	AENC_DECODER_PHI_E_PHI_M					R	
	Mask	Name			Type		
	0x0000FFFF <sub>h</sub>	AENC_DECODER_PHI_M			Signed		
	Min	Max	Default	Unit			
	-32768	32767	0				
	Resulting angle phi_m.						
	Mask	Name			Type		



Address	Registername				Access	
	0xFFFF0000 <sub>h</sub>	AENC_DECODER_PHI_E			Signed	
		Min	Max	Default	Unit	
		-32768	32767	0		
		Resulting angle phi_e.				
0x47 <sub>h</sub>	AENC_DECODER_POSITION				RW	
	Mask	Name		Type		
	0xFFFFFFFF <sub>h</sub>	AENC_DECODER_POSITION				Signed
		Min	Max	Default		Unit
		-2147483648	2147483647	0		
Multi-turn position.						
0x50 <sub>h</sub>	VELOCITY_SELECTION				RW	
	Mask	Name		Type		
	0x000000FF <sub>h</sub>	VELOCITY_SELECTION				Choice
		Min	Max	Default		Unit
		0	12	0		
		0: phi_e selected via PHI_E_SELECTION				
1: phi_e_ext						
2: phi_e_openloop						
3: phi_e_abn						
4: reserved						
5: phi_e_hal						
6: phi_e_aenc						
7: phi_a_aenc						
8: reserved						
9: phi_m_abn						
10: phi_m_abn_2						
11: phi_m_aenc						
12: phi_m_hal						
0x51 <sub>h</sub>	POSITION_SELECTION				RW	
	Mask	Name		Type		
	0x000000FF <sub>h</sub>	POSITION_SELECTION				Choice
		Min	Max	Default		Unit
		0	12	0		
		0: phi_e selected via PHI_E_SELECTION				
1: phi_e_ext						



Address	Registername				Access																									
	2: phi_e_openloop 3: phi_e_abn 4: reserved 5: phi_e_hal 6: phi_e_aenc 7: phi_a_aenc 8: reserved 9: phi_m_abn 10: phi_m_abn_2 11: phi_m_aenc 12: phi_m_hal																													
0x52 <sub>h</sub>	PHI_E_SELECTION <table border="1"> <thead> <tr> <th>Mask</th> <th colspan="3">Name</th> <th>Type</th> </tr> </thead> <tbody> <tr> <td>0x000000FF<sub>h</sub></td> <td colspan="3">PHI_E_SELECTION</td> <td>Choice</td> </tr> <tr> <td></td> <th>Min</th> <th>Max</th> <th>Default</th> <th>Unit</th> </tr> <tr> <td></td> <td>0</td> <td>7</td> <td>0</td> <td></td> </tr> <tr> <td></td> <td colspan="4">                             0: reserved                              1: phi_e_ext                              2: phi_e_openloop                              3: phi_e_abn                              4: reserved                              5: phi_e_hal                              6: phi_e_aenc                              7: phi_a_aenc                         </td> </tr> </tbody> </table>				Mask	Name			Type	0x000000FF <sub>h</sub>	PHI_E_SELECTION			Choice		Min	Max	Default	Unit		0	7	0			0: reserved 1: phi_e_ext 2: phi_e_openloop 3: phi_e_abn 4: reserved 5: phi_e_hal 6: phi_e_aenc 7: phi_a_aenc				RW
Mask	Name			Type																										
0x000000FF <sub>h</sub>	PHI_E_SELECTION			Choice																										
	Min	Max	Default	Unit																										
	0	7	0																											
	0: reserved 1: phi_e_ext 2: phi_e_openloop 3: phi_e_abn 4: reserved 5: phi_e_hal 6: phi_e_aenc 7: phi_a_aenc																													
0x53 <sub>h</sub>	PHI_E <table border="1"> <thead> <tr> <th>Mask</th> <th colspan="3">Name</th> <th>Type</th> </tr> </thead> <tbody> <tr> <td>0x0000FFFF<sub>h</sub></td> <td colspan="3">PHI_E</td> <td>Signed</td> </tr> <tr> <td></td> <th>Min</th> <th>Max</th> <th>Default</th> <th>Unit</th> </tr> <tr> <td></td> <td>-32768</td> <td>32767</td> <td>0</td> <td></td> </tr> <tr> <td></td> <td colspan="4">Angle used for the inner FOC loop.</td> </tr> </tbody> </table>				Mask	Name			Type	0x0000FFFF <sub>h</sub>	PHI_E			Signed		Min	Max	Default	Unit		-32768	32767	0			Angle used for the inner FOC loop.				R
Mask	Name			Type																										
0x0000FFFF <sub>h</sub>	PHI_E			Signed																										
	Min	Max	Default	Unit																										
	-32768	32767	0																											
	Angle used for the inner FOC loop.																													
0x54 <sub>h</sub>	PID_FLUX_P_FLUX_I <table border="1"> <thead> <tr> <th>Mask</th> <th colspan="3">Name</th> <th>Type</th> </tr> </thead> <tbody> <tr> <td>0x0000FFFF<sub>h</sub></td> <td colspan="3">PID_FLUX_I</td> <td>Signed</td> </tr> <tr> <td></td> <th>Min</th> <th>Max</th> <th>Default</th> <th>Unit</th> </tr> <tr> <td></td> <td>0</td> <td>32767</td> <td>0</td> <td></td> </tr> </tbody> </table>				Mask	Name			Type	0x0000FFFF <sub>h</sub>	PID_FLUX_I			Signed		Min	Max	Default	Unit		0	32767	0		RW					
Mask	Name			Type																										
0x0000FFFF <sub>h</sub>	PID_FLUX_I			Signed																										
	Min	Max	Default	Unit																										
	0	32767	0																											



Address	Registername					Access	
	Mask	Name			Type		
	0xFFFF0000 <sub>h</sub>	PID_FLUX_P			Signed		
		Min	Max	Default	Unit		
		0	32767	0			
0x55 <sub>h</sub>	PID_FLUX_D					RW	
	Mask	Name			Type		
	0x0000FFFF <sub>h</sub>	PID_FLUX_D			Signed		
		Min	Max	Default	Unit		
0		32767	0				
0x56 <sub>h</sub>	PID_TORQUE_P_TORQUE_I					RW	
	Mask	Name			Type		
	0x0000FFFF <sub>h</sub>	PID_TORQUE_I			Signed		
		Min	Max	Default	Unit		
		0	32767	0			
	Mask	Name			Type		
		0xFFFF0000 <sub>h</sub>	PID_TORQUE_P				Signed
			Min	Max	Default		Unit
0	32767	0					
0x57 <sub>h</sub>	PID_TORQUE_D					RW	
	Mask	Name			Type		
	0x0000FFFF <sub>h</sub>	PID_TORQUE_D			Signed		
		Min	Max	Default	Unit		
0		32767	0				
0x58 <sub>h</sub>	PID_VELOCITY_P_VELOCITY_I					RW	
	Mask	Name			Type		
	0x0000FFFF <sub>h</sub>	PID_VELOCITY_I			Signed		
		Min	Max	Default	Unit		
		0	32767	0			
	Mask	Name			Type		
		0xFFFF0000 <sub>h</sub>	PID_VELOCITY_P				Signed
			Min	Max	Default		Unit
0	32767	0					
0x59 <sub>h</sub>	PID_VELOCITY_D					RW	
	Mask	Name			Type		
	0x0000FFFF <sub>h</sub>	PID_VELOCITY_D			Signed		



Address	Registername					Access	
	Mask	Min	Max	Default	Unit		
		0	32767	0			
0x5A <sub>h</sub>	PID_POSITION_P_POSITION_I					RW	
	Mask	Name			Type		
	0x0000FFFF <sub>h</sub>	PID_POSITION_I					Signed
		Min	Max	Default	Unit		
		0	32767	0			
	Mask	Name			Type		
	0xFFFF0000 <sub>h</sub>	PID_POSITION_P					Signed
Min		Max	Default	Unit			
	0	32767	0				
0x5B <sub>h</sub>	PID_POSITION_D					RW	
	Mask	Name			Type		
	0x0000FFFF <sub>h</sub>	PID_POSITION_D					Signed
		Min	Max	Default	Unit		
	0	32767	0				
0x5C <sub>h</sub>	PID_TORQUE_FLUX_TARGET_DDT_LIMITS					RW	
	Mask	Name			Type		
	0xFFFFFFFF <sub>h</sub>	PID_TORQUE_FLUX_TARGET_DDT_LIMITS					Unsigned
		Min	Max	Default	Unit		
		0	32767	32767	[1/?s]		
Limits of change in time [d/dt] of the target torque and target flux.							
0x5D <sub>h</sub>	PIDOUT_UQ_UD_LIMITS					RW	
	Mask	Name			Type		
	0x0000FFFF <sub>h</sub>	PIDOUT_UQ_UD_LIMITS					Unsigned
		Min	Max	Default	Unit		
		0	32767	0			
Two dimensional circular limiter for inputs of iPark.							
0x5E <sub>h</sub>	PID_TORQUE_FLUX_LIMITS					RW	
	Mask	Name			Type		
	0x0000FFFF <sub>h</sub>	PID_TORQUE_FLUX_LIMITS					Unsigned
		Min	Max	Default	Unit		
	0	32767	0				





Address	Registername				Access
	PID torque limit and PID flux limit, limits the target values coming from the target registers.				
0x5F <sub>h</sub>	PID_ACCELERATION_LIMIT				RW
	Mask	Name		Type	
	0xFFFFFFFF <sub>h</sub>	PID_ACCELERATION_LIMIT		Unsigned	
		Min	Max	Default	
0	4294967295	0			
Acceleration limit.					
0x60 <sub>h</sub>	PID_VELOCITY_LIMIT				RW
	Mask	Name		Type	
	0xFFFFFFFF <sub>h</sub>	PID_VELOCITY_LIMIT		Unsigned	
		Min	Max	Default	
0	4294967295	0			
Velocity limit.					
0x61 <sub>h</sub>	PID_POSITION_LIMIT_LOW				RW
	Mask	Name		Type	
	0xFFFFFFFF <sub>h</sub>	PID_POSITION_LIMIT_LOW		Signed	
		Min	Max	Default	
-2147483648	2147483647	0			
Position limit low, programmable position barrier.					
0x62 <sub>h</sub>	PID_POSITION_LIMIT_HIGH				RW
	Mask	Name		Type	
	0xFFFFFFFF <sub>h</sub>	PID_POSITION_LIMIT_HIGH		Signed	
		Min	Max	Default	
-2147483648	2147483647	0			
Position limit high, programmable position barrier.					
0x63 <sub>h</sub>	MODE_RAMP_MODE_MOTION				RW
	Mask	Name		Type	
	0x000000FF <sub>h</sub>	MODE_MOTION		Choice	
		Min	Max	Default	
0	8	0			
0: stopped_mode 1: torque_mode 2: velocity_mode 3: position_mode					



Address	Registername				Access
		4: reserved 5: reserved 6: reserved 7: reserved 8: uq_ud_ext			
	Mask	Name		Type	
	0x0000FF00 <sub>h</sub>	MODE_RAMP		Choice	
		Min	Max	Default	Unit
		0	7	0	
		0: no velocity ramping			
		1: reserved			
		2: reserved			
		3: reserved			
		4: reserved			
		5: reserved			
		6: reserved			
		7: reserved			
0x64 <sub>h</sub>	PID_TORQUE_FLUX_TARGET				RW
	Mask	Name		Type	
	0x0000FFFF <sub>h</sub>	PID_FLUX_TARGET		Signed	
		Min	Max	Default	Unit
		-32768	32767	0	
	Mask	Name		Type	
	0xFFFF0000 <sub>h</sub>	PID_TORQUE_TARGET		Signed	
		Min	Max	Default	Unit
		-32768	32767	0	
0x65 <sub>h</sub>	PID_TORQUE_FLUX_OFFSET				RW
	Mask	Name		Type	
	0x0000FFFF <sub>h</sub>	PID_FLUX_OFFSET		Signed	
		Min	Max	Default	Unit
		-32768	32767	0	
		Flux offset for feed forward control.			
	Mask	Name		Type	
	0xFFFF0000 <sub>h</sub>	PID_TORQUE_OFFSET		Signed	
		Min	Max	Default	Unit



Address	Registername					Access
		-32768	32767	0		
	Torque offset for feed forward control.					
0x66 <sub>h</sub>	PID_VELOCITY_TARGET					RW
	Mask	Name			Type	
	0xFFFFFFFF <sub>h</sub>	PID_VELOCITY_TARGET			Signed	
		Min	Max	Default	Unit	
		-2147483648	2147483647	0		
	Target velocity register (for velocity mode).					
0x67 <sub>h</sub>	PID_VELOCITY_OFFSET					RW
	Mask	Name			Type	
	0xFFFFFFFF <sub>h</sub>	PID_VELOCITY_OFFSET			Signed	
		Min	Max	Default	Unit	
		-2147483648	2147483647	0		
	Velocity offset for feed forward control.					
0x68 <sub>h</sub>	PID_POSITION_TARGET					RW
	Mask	Name			Type	
	0xFFFFFFFF <sub>h</sub>	PID_POSITION_TARGET			Signed	
		Min	Max	Default	Unit	
		-2147483648	2147483647	0		
	Target position register (for position mode).					
0x69 <sub>h</sub>	PID_TORQUE_FLUX_ACTUAL					R
	Mask	Name			Type	
	0x0000FFFF <sub>h</sub>	PID_FLUX_ACTUAL			Signed	
		Min	Max	Default	Unit	
		-32768	32767	0		
	Mask	Name			Type	
	0xFFFF0000 <sub>h</sub>	PID_TORQUE_ACTUAL			Signed	
		Min	Max	Default	Unit	
		-32768	32767	0		
0x6A <sub>h</sub>	PID_VELOCITY_ACTUAL					R
	Mask	Name			Type	
	0xFFFFFFFF <sub>h</sub>	PID_VELOCITY_ACTUAL			Signed	
		Min	Max	Default	Unit	
		-2147483648	2147483647	0		



Address	Registername				Access	
	Actual velocity.					
0x6B <sub>h</sub>	PID_POSITION_ACTUAL				RW	
	Mask	Name		Type		
	0xFFFFFFFF <sub>h</sub>	PID_POSITION_ACTUAL		Signed		
		Min	Max	Default		Unit
		-2147483648	2147483647	0		
Actual multi turn position for positioning. WRITE on PID_POSITION_ACTUAL writes same value into PID_POSITION_TARGET to avoid unwanted move.						
0x6C <sub>h</sub>	PID_ERROR_DATA				R	
	Variant 0					
	Mask	Name		Type		
	0xFFFFFFFF <sub>h</sub>	PID_TORQUE_ERROR		Signed		
		Min	Max	Default		Unit
		-2147483648	2147483647	0		
	PID torque error.					
	Variant 1					
	Mask	Name		Type		
	0xFFFFFFFF <sub>h</sub>	PID_FLUX_ERROR		Signed		
		Min	Max	Default		Unit
		-2147483648	2147483647	0		
	PID flux error.					
	Variant 2					
	Mask	Name		Type		
	0xFFFFFFFF <sub>h</sub>	PID_VELOCITY_ERROR		Signed		
		Min	Max	Default		Unit
		-2147483648	2147483647	0		
	PID velocity error.					
	Variant 3					
Mask	Name		Type			
0xFFFFFFFF <sub>h</sub>	PID_POSITION_ERROR		Signed			
	Min	Max	Default	Unit		
	-2147483648	2147483647	0			
PID position error.						
Variant 4						



Address	Registername				Access	
	Mask	Name			Type	
	0xFFFFFFFF <sub>h</sub>	PID_TORQUE_ERROR_SUM			Signed	
		Min	Max	Default	Unit	
		-2147483648	2147483647	0		
	PID torque error.					
	Variant 5					
	Mask	Name			Type	
	0xFFFFFFFF <sub>h</sub>	PID_FLUX_ERROR_SUM			Signed	
		Min	Max	Default	Unit	
		-2147483648	2147483647	0		
	PID flux error sum.					
	Variant 6					
	Mask	Name			Type	
	0xFFFFFFFF <sub>h</sub>	PID_VELOCITY_ERROR_SUM			Signed	
		Min	Max	Default	Unit	
-2147483648		2147483647	0			
PID velocity error sum.						
Variant 7						
Mask	Name			Type		
0xFFFFFFFF <sub>h</sub>	PID_POSITION_ERROR_SUM			Signed		
	Min	Max	Default	Unit		
	-2147483648	2147483647	0			
PID position error sum.						
0x6D <sub>h</sub>	PID_ERROR_ADDR				RW	
	Mask	Name				Type
	0x000000FF <sub>h</sub>	PID_ERROR_ADDR				Choice
		Min	Max	Default		Unit
		0	7	0		
0: PID_TORQUE_ERROR 1: PID_FLUX_ERROR 2: PID_VELOCITY_ERROR 3: PID_POSITION_ERROR 4: PID_TORQUE_ERROR_SUM 5: PID_FLUX_ERROR_SUM 6: PID_VELOCITY_ERROR_SUM						



Address	Registername				Access	
	7: PID_POSITION_ERROR_SUM					
0x6E <sub>h</sub>	INTERIM_DATA				RW	
	Variant 0					
	Mask	Name		Type		
	0xFFFFFFFF <sub>h</sub>	PIDIN_TARGET_TORQUE				Signed
		Min	Max	Default		Unit
		-2147483648	2147483647	0		
	PIDIN target torque.					
	Variant 1					
	Mask	Name		Type		
	0xFFFFFFFF <sub>h</sub>	PIDIN_TARGET_FLUX				Signed
		Min	Max	Default		Unit
		-2147483648	2147483647	0		
	PIDIN target flux.					
	Variant 2					
	Mask	Name		Type		
	0xFFFFFFFF <sub>h</sub>	PIDIN_TARGET_VELOCITY				Signed
		Min	Max	Default		Unit
		-2147483648	2147483647	0		
	PIDIN target velocity.					
	Variant 3					
Mask	Name		Type			
0xFFFFFFFF <sub>h</sub>	PIDIN_TARGET_POSITION			Signed		
	Min	Max	Default	Unit		
	-2147483648	2147483647	0			
PIDIN target position.						
Variant 4						
Mask	Name		Type			
0xFFFFFFFF <sub>h</sub>	PIDOUT_TARGET_TORQUE			Signed		
	Min	Max	Default	Unit		
	-2147483648	2147483647	0			
PIDOUT target torque.						
Variant 5						
Mask	Name		Type			
0xFFFFFFFF <sub>h</sub>	PIDOUT_TARGET_FLUX			Signed		



Address	Registername				Access
	Min	Max	Default	Unit	
	-2147483648	2147483647	0		
	PIDOUT target flux.				
	Variant 6				
Mask	Name		Type		
0xFFFFFFFF <sub>h</sub>	PIDOUT_TARGET_VELOCITY		Signed		
	Min	Max	Default	Unit	
	-2147483648	2147483647	0		
	PIDOUT target velocity.				
	Variant 7				
Mask	Name		Type		
0xFFFFFFFF <sub>h</sub>	PIDOUT_TARGET_POSITION		Signed		
	Min	Max	Default	Unit	
	-2147483648	2147483647	0		
	PIDOUT target position.				
	Variant 8				
Mask	Name		Type		
0x0000FFFF <sub>h</sub>	FOC_IUX		Signed		
	Min	Max	Default	Unit	
	-32768	32767	0		
Mask	Name		Type		
0xFFFF0000 <sub>h</sub>	FOC_IWY		Signed		
	Min	Max	Default	Unit	
	-32768	32767	0		
	Variant 9				
Mask	Name		Type		
0x0000FFFF <sub>h</sub>	FOC_IV		Signed		
	Min	Max	Default	Unit	
	-32768	32767	0		
	Variant 10				
Mask	Name		Type		
0x0000FFFF <sub>h</sub>	FOC_IA		Signed		
	Min	Max	Default	Unit	
	-32768	32767	0		
Mask	Name		Type		



Address	Registername				Access
0xFFFF0000 <sub>h</sub>	FOC_IB			Signed	
	Min	Max	Default	Unit	
	-32768	32767	0		
Variant 11					
Mask	Name			Type	
0x0000FFFF <sub>h</sub>	FOC_ID			Signed	
	Min	Max	Default	Unit	
	-32768	32767	0		
Mask	Name			Type	
0xFFFF0000 <sub>h</sub>	FOC_IQ			Signed	
	Min	Max	Default	Unit	
	-32768	32767	0		
Variant 12					
Mask	Name			Type	
0x0000FFFF <sub>h</sub>	FOC_UD			Signed	
	Min	Max	Default	Unit	
	-32768	32767	0		
Mask	Name			Type	
0xFFFF0000 <sub>h</sub>	FOC_UQ			Signed	
	Min	Max	Default	Unit	
	-32768	32767	0		
Variant 13					
Mask	Name			Type	
0x0000FFFF <sub>h</sub>	FOC_UD_LIMITED			Signed	
	Min	Max	Default	Unit	
	-32768	32767	0		
Mask	Name			Type	
0xFFFF0000 <sub>h</sub>	FOC_UQ_LIMITED			Signed	
	Min	Max	Default	Unit	
	-32768	32767	0		
Variant 14					
Mask	Name			Type	
0x0000FFFF <sub>h</sub>	FOC_UA			Signed	
	Min	Max	Default	Unit	
	-32768	32767	0		





Address	Registername				Access
	Mask	Name			Type
	0xFFFF0000 <sub>h</sub>	FOC_UB			Signed
		Min	Max	Default	Unit
		-32768	32767	0	
Variant 15					
	Mask	Name			Type
	0x0000FFFF <sub>h</sub>	FOC_UUX			Signed
		Min	Max	Default	Unit
		-32768	32767	0	
	Mask	Name			Type
	0xFFFF0000 <sub>h</sub>	FOC_UWY			Signed
		Min	Max	Default	Unit
		-32768	32767	0	
Variant 16					
	Mask	Name			Type
	0x0000FFFF <sub>h</sub>	FOC_UV			Signed
		Min	Max	Default	Unit
		-32768	32767	0	
Variant 17					
	Mask	Name			Type
	0x0000FFFF <sub>h</sub>	PWM_UX			Signed
		Min	Max	Default	Unit
		-32768	32767	0	
	Mask	Name			Type
	0xFFFF0000 <sub>h</sub>	PWM_WY			Signed
		Min	Max	Default	Unit
		-32768	32767	0	
Variant 18					
	Mask	Name			Type
	0x0000FFFF <sub>h</sub>	PWM_V			Signed
		Min	Max	Default	Unit
		-32768	32767	0	
Variant 19					
	Mask	Name			Type
	0x0000FFFF <sub>h</sub>	ADC_I_0			Signed



Address	Registername				Access
	Min	Max	Default	Unit	
	-32768	32767	0		
Mask	Name			Type	
0xFFFF0000 <sub>h</sub>	ADC_I_1			Signed	
	Min	Max	Default	Unit	
	-32768	32767	0		
Variant 192					
Mask	Name			Type	
0x0000FFFF <sub>h</sub>	DEBUG_VALUE_0			Signed	
	Min	Max	Default	Unit	
	-32768	32767	0		
Mask	Name			Type	
0xFFFF0000 <sub>h</sub>	DEBUG_VALUE_1			Signed	
	Min	Max	Default	Unit	
	-32768	32767	0		
Variant 193					
Mask	Name			Type	
0x0000FFFF <sub>h</sub>	DEBUG_VALUE_2			Signed	
	Min	Max	Default	Unit	
	-32768	32767	0		
Mask	Name			Type	
0xFFFF0000 <sub>h</sub>	DEBUG_VALUE_3			Signed	
	Min	Max	Default	Unit	
	-32768	32767	0		
Variant 194					
Mask	Name			Type	
0x0000FFFF <sub>h</sub>	DEBUG_VALUE_4			Signed	
	Min	Max	Default	Unit	
	-32768	32767	0		
Mask	Name			Type	
0xFFFF0000 <sub>h</sub>	DEBUG_VALUE_5			Signed	
	Min	Max	Default	Unit	
	-32768	32767	0		
Variant 195					
Mask	Name			Type	



Address	Registername				Access
0x0000FFFF <sub>h</sub>	DEBUG_VALUE_6			Signed	
	Min	Max	Default	Unit	
	-32768	32767	0		
Mask	Name			Type	
0xFFFF0000 <sub>h</sub>	DEBUG_VALUE_7			Signed	
	Min	Max	Default	Unit	
	-32768	32767	0		
Variant 196					
Mask	Name			Type	
0x0000FFFF <sub>h</sub>	DEBUG_VALUE_8			Unsigned	
	Min	Max	Default	Unit	
	0	65535	0		
Mask	Name			Type	
0xFFFF0000 <sub>h</sub>	DEBUG_VALUE_9			Unsigned	
	Min	Max	Default	Unit	
	0	65535	0		
Variant 197					
Mask	Name			Type	
0x0000FFFF <sub>h</sub>	DEBUG_VALUE_10			Unsigned	
	Min	Max	Default	Unit	
	0	65535	0		
Mask	Name			Type	
0xFFFF0000 <sub>h</sub>	DEBUG_VALUE_11			Unsigned	
	Min	Max	Default	Unit	
	0	65535	0		
Variant 198					
Mask	Name			Type	
0x0000FFFF <sub>h</sub>	DEBUG_VALUE_12			Unsigned	
	Min	Max	Default	Unit	
	0	65535	0		
Mask	Name			Type	
0xFFFF0000 <sub>h</sub>	DEBUG_VALUE_13			Unsigned	
	Min	Max	Default	Unit	
	0	65535	0		
Variant 199					



Address	Registername				Access
	Mask	Name			Type
	0x0000FFFF <sub>h</sub>	DEBUG_VALUE_14			Unsigned
		Min	Max	Default	Unit
		0	65535	0	
	Mask	Name			Type
	0xFFFF0000 <sub>h</sub>	DEBUG_VALUE_15			Unsigned
		Min	Max	Default	Unit
		0	65535	0	
	Variant 200				
	Mask	Name			Type
	0xFFFFFFFF <sub>h</sub>	DEBUG_VALUE_16			Signed
		Min	Max	Default	Unit
		-2147483648	2147483647	0	
	Variant 201				
	Mask	Name			Type
	0xFFFFFFFF <sub>h</sub>	DEBUG_VALUE_17			Signed
		Min	Max	Default	Unit
		-2147483648	2147483647	0	
	Variant 202				
	Mask	Name			Type
	0xFFFFFFFF <sub>h</sub>	DEBUG_VALUE_18			Signed
		Min	Max	Default	Unit
		-2147483648	2147483647	0	
	Variant 203				
Mask	Name			Type	
0xFFFFFFFF <sub>h</sub>	DEBUG_VALUE_19			Signed	
	Min	Max	Default	Unit	
	-2147483648	2147483647	0		
0x6F <sub>h</sub>	INTERIM_ADDR				RW
Mask	Name			Type	
0x000000FF <sub>h</sub>	INTERIM_ADDR			Choice	
	Min	Max	Default	Unit	
	0	203	0		
	0: PIDIN_TARGET_TORQUE 1: PIDIN_TARGET_FLUX				



Address	Registername				Access
	2: PIDIN_TARGET_VELOCITY 3: PIDIN_TARGET_POSITION 4: PIDOUT_TARGET_TORQUE 5: PIDOUT_TARGET_FLUX 6: PIDOUT_TARGET_VELOCITY 7: PIDOUT_TARGET_POSITION 8: FOC_IWY_IUX 9: FOC_IV 10: FOC_IB_IA 11: FOC_IQ_ID 12: FOC_UQ_UD 13: FOC_UQ_UD_LIMITED 14: FOC_UB_UA 15: FOC_UWY_UUX 16: FOC_UV 17: PWM_WY_UX 18: PWM_UV 19: ADC_I1_I0 192: DEBUG_VALUE_1_0 193: DEBUG_VALUE_3_2 194: DEBUG_VALUE_5_4 195: DEBUG_VALUE_7_6 196: DEBUG_VALUE_9_8 197: DEBUG_VALUE_11_10 198: DEBUG_VALUE_13_12 199: DEBUG_VALUE_15_14 200: DEBUG_VALUE_16 201: DEBUG_VALUE_17 202: DEBUG_VALUE_18 203: DEBUG_VALUE_19				
0x75 <sub>h</sub>	ADC_VM_LIMITS				RW
	Mask	Name		Type	
	0x0000FFFF <sub>h</sub>	ADC_VM_LIMIT_LOW		Unsigned	
		Min	Max	Default	Unit
		0	65535	0	



Address	Registername				Access
		Low limit for brake chopper output BRAKE_OUT.			
	Mask	Name		Type	
	0xFFFF0000 <sub>h</sub>	ADC_VM_LIMIT_HIGH		Unsigned	
		Min	Max	Default	Unit
		0	65535	0	
		High limit for brake chopper output BRAKE_OUT.			
0x76 <sub>h</sub>	TMC4670_INPUTS_RAW				R
	Mask	Name		Type	
	0x00000001 <sub>h</sub>	TMC4670_INPUTS_RAW[0]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		A of ABN_RAW			
		0: off			
		1: on			
	Mask	Name		Type	
	0x00000002 <sub>h</sub>	TMC4670_INPUTS_RAW[1]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		B of ABN_RAW			
		0: off			
		1: on			
	Mask	Name		Type	
	0x00000004 <sub>h</sub>	TMC4670_INPUTS_RAW[2]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		N of ABN_RAW			
		0: off			
		1: on			
	Mask	Name		Type	
	0x00000008 <sub>h</sub>	TMC4670_INPUTS_RAW[3]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		—			
		0: off			
		1: on			



Address	Registername				Access
	Mask	Name			Type
	0x00000010 <sub>h</sub>	TMC4670_INPUTS_RAW[4]			Bool
		Min	Max	Default	Unit
		0	1	0	
		A of ABN_2_RAW			
		0: off			
		1: on			
	Mask	Name			Type
	0x00000020 <sub>h</sub>	TMC4670_INPUTS_RAW[5]			Bool
		Min	Max	Default	Unit
		0	1	0	
		B of ABN_2_RAW			
		0: off			
		1: on			
	Mask	Name			Type
	0x00000040 <sub>h</sub>	TMC4670_INPUTS_RAW[6]			Bool
		Min	Max	Default	Unit
		0	1	0	
		N of ABN_2_RAW			
		0: off			
		1: on			
	Mask	Name			Type
	0x00000080 <sub>h</sub>	TMC4670_INPUTS_RAW[7]			Bool
		Min	Max	Default	Unit
		0	1	0	
		—			
		0: off			
		1: on			
	Mask	Name			Type
	0x00000100 <sub>h</sub>	TMC4670_INPUTS_RAW[8]			Bool
		Min	Max	Default	Unit
		0	1	0	
		HALL_UX of HALL_RAW			
		0: off			
		1: on			



Address	Registername				Access
	Mask	Name		Type	
	0x00000200 <sub>h</sub>	TMC4670_INPUTS_RAW[9]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		HALL_V of HALL_RAW			
		0: off			
		1: on			
	Mask	Name		Type	
	0x00000400 <sub>h</sub>	TMC4670_INPUTS_RAW[10]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		HALL_WY of HALL_RAW			
		0: off			
		1: on			
	Mask	Name		Type	
	0x00000800 <sub>h</sub>	TMC4670_INPUTS_RAW[11]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		—			
		0: off			
		1: on			
	Mask	Name		Type	
	0x00001000 <sub>h</sub>	TMC4670_INPUTS_RAW[12]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		REF_SW_R_RAW			
		0: off			
		1: on			
	Mask	Name		Type	
	0x00002000 <sub>h</sub>	TMC4670_INPUTS_RAW[13]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		REF_SW_H_RAW			
		0: off			
		1: on			





Address	Registername				Access
	Mask	Name		Type	
	0x00004000 <sub>h</sub>	TMC4670_INPUTS_RAW[14]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		REF_SW_L_RAW			
		0: off			
		1: on			
	Mask	Name		Type	
	0x00008000 <sub>h</sub>	TMC4670_INPUTS_RAW[15]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		ENABLE_IN_RAW			
		0: off			
		1: on			
	Mask	Name		Type	
	0x00010000 <sub>h</sub>	TMC4670_INPUTS_RAW[16]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		STP of DIRSTP_RAW			
		0: off			
		1: on			
	Mask	Name		Type	
	0x00020000 <sub>h</sub>	TMC4670_INPUTS_RAW[17]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		DIR of DIRSTP_RAW			
		0: off			
		1: on			
	Mask	Name		Type	
	0x00040000 <sub>h</sub>	TMC4670_INPUTS_RAW[18]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		PWM_IN_RAW			
		0: off			
		1: on			



Address	Registername				Access
	Mask	Name			Type
	0x00080000 <sub>h</sub>	TMC4670_INPUTS_RAW[19]			Bool
		Min	Max	Default	Unit
		0	1	0	
		—			
		0: off			
		1: on			
	Mask	Name			Type
	0x00100000 <sub>h</sub>	TMC4670_INPUTS_RAW[20]			Bool
		Min	Max	Default	Unit
		0	1	0	
		ESI_0 of ESI_RAW			
		0: off			
		1: on			
	Mask	Name			Type
	0x00200000 <sub>h</sub>	TMC4670_INPUTS_RAW[21]			Bool
		Min	Max	Default	Unit
		0	1	0	
		ESI_1 of ESI_RAW			
		0: off			
		1: on			
	Mask	Name			Type
	0x00400000 <sub>h</sub>	TMC4670_INPUTS_RAW[22]			Bool
		Min	Max	Default	Unit
		0	1	0	
		ESI_2 of ESI_RAW			
		0: off			
		1: on			
	Mask	Name			Type
	0x00800000 <sub>h</sub>	TMC4670_INPUTS_RAW[23]			Bool
		Min	Max	Default	Unit
		0	1	0	
		—			
		0: off			
		1: on			



Address	Registername				Access
	Mask	Name		Type	
	0x01000000 <sub>h</sub>	TMC4670_INPUTS_RAW[24]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		CFG_0 of CFG			
		0: off			
		1: on			
	Mask	Name		Type	
	0x02000000 <sub>h</sub>	TMC4670_INPUTS_RAW[25]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		CFG_1 of CFG			
		0: off			
		1: on			
	Mask	Name		Type	
	0x04000000 <sub>h</sub>	TMC4670_INPUTS_RAW[26]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		CFG_2 of CFG			
		0: off			
		1: on			
	Mask	Name		Type	
	0x08000000 <sub>h</sub>	TMC4670_INPUTS_RAW[27]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		CFG_3 of CFG			
		0: off			
		1: on			
	Mask	Name		Type	
	0x10000000 <sub>h</sub>	TMC4670_INPUTS_RAW[28]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		PWM_IDLE_L_RAW			
		0: off			
		1: on			



Address	Registername				Access	
	Mask	Name			Type	
	0x20000000 <sub>h</sub>	TMC4670_INPUTS_RAW[29]			Bool	
		Min	Max	Default	Unit	
		0	1	0		
		PWM_IDLE_H_RAW				
		0: off				
		1: on				
	Mask	Name			Type	
	0x40000000 <sub>h</sub>	TMC4670_INPUTS_RAW[30]			Bool	
		Min	Max	Default	Unit	
		0	1	0		
		DRV_ERR_IN_RAW				
	0: off					
	1: on					
Mask	Name			Type		
0x80000000 <sub>h</sub>	TMC4670_INPUTS_RAW[31]			Bool		
	Min	Max	Default	Unit		
	0	1	0			
	—					
	0: off					
	1: on					
0x77 <sub>h</sub>	TMC4670_OUTPUTS_RAW				R	
	Mask	Name				Type
	0x00000001 <sub>h</sub>	TMC4670_OUTPUTS_RAW[0]				Bool
		Min	Max	Default		Unit
		0	1	0		
		PWM_UX1_L				
		0: off				
	1: on					
Mask	Name			Type		
0x00000002 <sub>h</sub>	TMC4670_OUTPUTS_RAW[1]			Bool		
	Min	Max	Default	Unit		
	0	1	0			
	PWM_UX1_H					
	0: off					



Address	Registername				Access
		1: on			
	Mask	Name		Type	
	0x00000004 <sub>h</sub>	TMC4670_OUTPUTS_RAW[2]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		PWM_VX2_L			
		0: off			
		1: on			
	Mask	Name		Type	
	0x00000008 <sub>h</sub>	TMC4670_OUTPUTS_RAW[3]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		PWM_VX2_H			
		0: off			
		1: on			
	Mask	Name		Type	
	0x00000010 <sub>h</sub>	TMC4670_OUTPUTS_RAW[4]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		PWM_WY1_L			
		0: off			
		1: on			
	Mask	Name		Type	
	0x00000020 <sub>h</sub>	TMC4670_OUTPUTS_RAW[5]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		PWM_WY1_H			
		0: off			
		1: on			
	Mask	Name		Type	
	0x00000040 <sub>h</sub>	TMC4670_OUTPUTS_RAW[6]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		PWM_Y2_L			
		0: off			



Address	Registername				Access
		1: on			
	Mask	Name		Type	
	0x00000080 <sub>h</sub>	TMC4670_OUTPUTS_RAW[7]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		PWM_Y2_H			
		0: off			
		1: on			
0x7C <sub>h</sub>	STATUS_FLAGS				RW
	Mask	Name		Type	
	0x00000001 <sub>h</sub>	STATUS_FLAGS[0]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		pid_x_target_limit			
		0: off			
		1: on			
	Mask	Name		Type	
	0x00000002 <sub>h</sub>	STATUS_FLAGS[1]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		pid_x_target_ddt_limit			
		0: off			
		1: on			
	Mask	Name		Type	
	0x00000004 <sub>h</sub>	STATUS_FLAGS[2]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		pid_x_errsum_limit			
		0: off			
		1: on			
	Mask	Name		Type	
	0x00000008 <sub>h</sub>	STATUS_FLAGS[3]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		pid_x_output_limit			



Address	Registername				Access
		0: off 1: on			
	Mask	Name		Type	
	0x00000010 <sub>h</sub>	STATUS_FLAGS[4]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		pid_v_target_limit			
		0: off 1: on			
	Mask	Name		Type	
	0x00000020 <sub>h</sub>	STATUS_FLAGS[5]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		pid_v_target_ddt_limit			
		0: off 1: on			
	Mask	Name		Type	
	0x00000040 <sub>h</sub>	STATUS_FLAGS[6]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		pid_v_errsum_limit			
		0: off 1: on			
	Mask	Name		Type	
	0x00000080 <sub>h</sub>	STATUS_FLAGS[7]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		pid_v_output_limit			
		0: off 1: on			
	Mask	Name		Type	
	0x00000100 <sub>h</sub>	STATUS_FLAGS[8]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		pid_id_target_limit			



Address	Registername				Access
		0: off 1: on			
	Mask	Name		Type	
	0x00000200 <sub>h</sub>	STATUS_FLAGS[9]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		pid_id_target_ddt_limit			
		0: off 1: on			
	Mask	Name		Type	
	0x00000400 <sub>h</sub>	STATUS_FLAGS[10]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		pid_id_errsum_limit			
		0: off 1: on			
	Mask	Name		Type	
	0x00000800 <sub>h</sub>	STATUS_FLAGS[11]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		pid_id_output_limit			
		0: off 1: on			
	Mask	Name		Type	
	0x00001000 <sub>h</sub>	STATUS_FLAGS[12]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		pid_iq_target_limit			
		0: off 1: on			
	Mask	Name		Type	
	0x00002000 <sub>h</sub>	STATUS_FLAGS[13]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		pid_iq_target_ddt_limit			





Address	Registername				Access
		0: off 1: on			
	Mask	Name		Type	
	0x00004000 <sub>h</sub>	STATUS_FLAGS[14]			Bool
		Min	Max	Default	Unit
		0	1	0	
		pid_iq_errsum_limit			
		0: off 1: on			
	Mask	Name		Type	
	0x00008000 <sub>h</sub>	STATUS_FLAGS[15]			Bool
		Min	Max	Default	Unit
		0	1	0	
		pid_iq_output_limit			
		0: off 1: on			
	Mask	Name		Type	
	0x00010000 <sub>h</sub>	STATUS_FLAGS[16]			Bool
		Min	Max	Default	Unit
		0	1	0	
		ipark_cirlim_limit_u_d			
		0: off 1: on			
	Mask	Name		Type	
	0x00020000 <sub>h</sub>	STATUS_FLAGS[17]			Bool
		Min	Max	Default	Unit
		0	1	0	
		ipark_cirlim_limit_u_q			
		0: off 1: on			
	Mask	Name		Type	
	0x00040000 <sub>h</sub>	STATUS_FLAGS[18]			Bool
		Min	Max	Default	Unit
		0	1	0	
		ipark_cirlim_limit_u_r			



Address	Registername				Access
		0: off 1: on			
	Mask	Name		Type	
	0x00080000 <sub>h</sub>	STATUS_FLAGS[19]			Bool
		Min	Max	Default	Unit
		0	1	0	
		— 0: off 1: on			
	Mask	Name		Type	
	0x00100000 <sub>h</sub>	STATUS_FLAGS[20]			Bool
		Min	Max	Default	Unit
		0	1	0	
		ref_sw_r 0: off 1: on			
	Mask	Name		Type	
	0x00200000 <sub>h</sub>	STATUS_FLAGS[21]			Bool
		Min	Max	Default	Unit
		0	1	0	
		ref_sw_h 0: off 1: on			
	Mask	Name		Type	
	0x00400000 <sub>h</sub>	STATUS_FLAGS[22]			Bool
		Min	Max	Default	Unit
		0	1	0	
		ref_sw_l 0: off 1: on			
	Mask	Name		Type	
	0x00800000 <sub>h</sub>	STATUS_FLAGS[23]			Bool
		Min	Max	Default	Unit
		0	1	0	
		—			



Address	Registername				Access
		0: off 1: on			
	Mask	Name		Type	
	0x01000000 <sub>h</sub>	STATUS_FLAGS[24]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		pwm_min 0: off 1: on			
	Mask	Name		Type	
	0x02000000 <sub>h</sub>	STATUS_FLAGS[25]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		pwm_max 0: off 1: on			
	Mask	Name		Type	
	0x04000000 <sub>h</sub>	STATUS_FLAGS[26]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		adc_i_clipped 0: off 1: on			
	Mask	Name		Type	
	0x08000000 <sub>h</sub>	STATUS_FLAGS[27]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		aenc_clipped 0: off 1: on			
	Mask	Name		Type	
	0x10000000 <sub>h</sub>	STATUS_FLAGS[28]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		enc_n			



Address	Registername				Access
		0: off 1: on			
	Mask	Name		Type	
	0x20000000 <sub>h</sub>	STATUS_FLAGS[29]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		enc_2_n			
		0: off			
		1: on			
	Mask	Name		Type	
	0x40000000 <sub>h</sub>	STATUS_FLAGS[30]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		aenc_n			
		0: off			
		1: on			
	Mask	Name		Type	
	0x80000000 <sub>h</sub>	STATUS_FLAGS[31]		Bool	
		Min	Max	Default	Unit
		0	1	0	
		—			
		0: off			
		1: on			
0x7D <sub>h</sub>	WARNING_MASK				RW
	Mask	Name		Type	
	0xFFFFFFFF <sub>h</sub>	WARNING_MASK		Unsigned	
		Min	Max	Default	Unit
		0	4294967295	0	
0x7E <sub>h</sub>	ERROR_MASK				RW
	Mask	Name		Type	
	0xFFFFFFFF <sub>h</sub>	ERROR_MASK		Unsigned	
		Min	Max	Default	Unit
		0	4294967295	0	

Table 6: Register Map for TMC4670



## 6 Pinning

### **i** Info

All power supply pins (1V2, 2V5, 3V3) must be connected.

All ground pins (GND, GND\_REF) must be connected.

All pins specified as n.c. (=not connected) must be left open.

Analog inputs (AI) are 2.5V single ended inputs. Voltage dividers are required to scale down higher input voltages.

Digital inputs (I) resp. (IO) are 3.3V single ended inputs. Voltage dividers are required to scale done higher input voltages.

IO	Description
AI	analog input, 2.5V
I	digital input, 3.3V
IO	digital input or digital output, direction programmable, 3.3V
O	digital output, 3.3V

*Table 7: Pin Type Definition*



Name	Ball	IO	Description
nRST	B10	I	active low reset input
CLK	M3	I	clock input, 25 MHz
ENABLE_IN	T13	I	active high enable input
ENABLE_OUT	A5	O	enable output
SPI_nSCS	R7	I	SPI active low chip select input
SPI_SCK	T7	I	SPI clock input
SPI_MOSI	T6	I	SPI master out slave input
SPI_MISO	R6	O	SPI master in slave output
WARNING_OUT	P10	O	Maskable Warning Output
ERROR_OUT	T11	O	Maskable Error Output
ESI_0	H16	I	Emergency Switch Input #0
ESI_1	J15	I	Emergency Switch Input #1
ESI_2	J16	I	Emergency Switch Input #2
ENC_A	L1	I	A input of incremental encoder
ENC_B	L2	I	B input of incremental encoder
ENC_N	M1	I	N input of incremental encoder
ENC2_A	R1	I	A input of incremental encoder
ENC2_B	P2	I	B input of incremental encoder
ENC2_N	P4	I	N input of incremental encoder
HALL_UX	N2	I	digital hall input H1 for 3-phase (U) or 2-phase (X)
HALL_V	N1	I	digital hall input H2 for 3-phase (V)
HALL_WY	M2	I	digital hall input H3 for 3-phase (W) or 2-phase (Y)
REF_SW_L	A9	I	Left (L) reference switch
REF_SW_H	A8	I	Home (H) reference switch
REF_SW_R	A10	I	Right (R) reference switch
AENC_UX	C1	AI	analog hall or analog encoder, 3-phase (U) or 2-phase (X (cos))
AENC_V	D1	AI	analog hall or analog encoder, 3-phase (V)
AENC_WY	E1	AI	analog hall or analog encoder, 3-phase (W) or 2-phase (Y (sin))
AENC_N	B1	AI	analog encoder N pulse input
ADC_I_0	F2	AI	phase current measurement voltage input I_0 (I_U, I_X)
ADC_I_1	F4	AI	phase current measurement voltage input I_1 (I_V, I_W, I_Y)
ADC_VM	E3	AI	analog input for motor supply voltage (VM) measurement
AGPI	G2	AI	Analog general purpose input (analog GPI)
ADC_T_MOSFETS	C3	AI	analog input for MOSFET temperature signal



Name	Ball	IO	Description
ADC_T_MOTOR	C4	AI	analog input for motor temperature signal
ADC_U_UX	F1	AI	analog voltage input for terminal UX
ADC_U_V	F5	AI	analog voltage input for terminal V
ADC_U_WY	B2	AI	analog voltage input for terminal WY
ADC_I_S	C2	AI	analog bottom current measurement voltage input, reserved
ADC_START	B5	O	ADC start pulse for LTC2351
ADC_BIP	B4	O	output driving ADC LTC2351 BIP input
ADC_nSCS	B7	O	ADC LTC2351 nSCS signal
ADC_SCK	B6	O	ADC LTC2351 SCK signal
ADC_MISO	A6	I	from ADC LTC2351 SDO output
ADC_MOSI	A7	O	reserved for ADC w/ SDI input
PWM_IDLE_H	A11	I	Idle Level of high side gate control signals
PWM_IDLE_L	B11	I	Idle Level of low side gate control signals
PWM_UX1_H	D16	O	high side gate control output U (3-phase) resp. X1 (2-phase)
PWM_UX1_L	D15	O	low side gate control output U (3-phase) resp. X1 (2-phase)
PWM_VX2_H	C16	O	high side gate control output V (3-phase) resp. X2 (2-phase)
PWM_VX2_L	D14	O	low side gate control output V (3-phase) resp. X2 (2-phase)
PWM_WY1_H	B16	O	high side gate control output W (3-phase) resp. Y1 (2-phase)
PWM_WY1_L	C15	O	low side gate control output W (3-phase) resp. Y1 (2-phase)
PWM_Y2_H	B15	O	high side gate control output Y2 (2-phase only)
PWM_Y2_L	C14	O	low side gate control output Y2 (2-phase only)
PWM_Z_OUT	L15	O	pulse, indicating start of PWM period (zero count)
PWM_C_OUT	L16	O	pulse, indicating center of PWM period
BRAKE_OUT	A13	O	brake chopper control signal
DRV_ERR_IN	A12	I	driver error input (from gate driver)

*Table 8: Functional Pin Description*



Name	Ball	IO	Supply Voltage Pins and Ground Pins
VCCIO3V3	K4	3.3V	
VCCIO3V3	L4	3.3V	
VCCIO3V3	N6	3.3V	
VCCIO3V3	N7	3.3V	
VCCIO3V3	N8	3.3V	
VCCIO3V3	N10	3.3V	
VCCIO3V3	N11	3.3V	
VCCIO3V3	K13	3.3V	
VCCIO3V3	L13	3.3V	
VCCIO3V3	M13	3.3V	
VCCIO3V3	F13	3.3V	
VCCIO3V3	G13	3.3V	
VCCIO3V3	H13	3.3V	
VCCIO3V3	J13	3.3V	
VCCIO3V3	D10	3.3V	
VCCIO3V3	D11	3.3V	
VCCIO3V3	C7	3.3V	
VCCIO3V3	D7	3.3V	
VCCIO3V3	D8	3.3V	
VCCrefR1	E8	3.3V via 10K	reference resistor to be connected to 3.3V supply voltage
VCCrefR2	F7	3.3V via 10K	reference resistor to be connected to 3.3V supply voltage
VCCrefR3	E7	3.3V via 10K	reference resistor to be connected to 3.3V supply voltage
ADCVREF2V5	E4	2.5V	
VCCA2V5	D5	2.5V	
VCCA2V5	E12	2.5V	
VCCA2V5	L5	2.5V	
VCCA2V5	M12	2.5V	
VCCA2V5	E5	2.5V	
VCCA2V5	G4	2.5V	
VCCA2V5	H4	2.5V	
VCCA2V5	J4	2.5V	
VCC1V2	G7	1.2V	
VCC1V2	G9	1.2V	
VCC1V2	H8	1.2V	





Name	Ball	IO	Supply Voltage Pins and Ground Pins
VCC1V2	H9	1.2V	
VCC1V2	H10	1.2V	
VCC1V2	J7	1.2V	
VCC1V2	J8	1.2V	
VCC1V2	J9	1.2V	
VCC1V2	K8	1.2V	
VCC1V2	K10	1.2V	
VCCINT1V2	F6	1.2V	
VCCPLL1V2	D4	1.2V	
VCCPLL1V2	D13	1.2V	
VCCPLL1V2	M5	1.2V	
VCCPLL1V2	N13	1.2V	
GND	A1	0V	
GND	A16	0V	
GND	B14	0V	
GND	C8	0V	
GND	C11	0V	
GND	D2	0V	
GND	D3	0V	
GND	D6	0V	
GND	E6	0V	
GND	E13	0V	
GND	E14	0V	
GND	E15	0V	
GND	E16	0V	
GND	F8	0V	
GND	F15	0V	
GND	F3	0V	
GND	G3	0V	
GND	G8	0V	
GND	G10	0V	
GND	H7	0V	
GND	H14	0V	
GND	J10	0V	



<b>Name</b>	<b>Ball</b>	<b>IO</b>	<b>Supply Voltage Pins and Ground Pins</b>
GND	K1	0V	
GND	K3	0V	
GND	K7	0V	
GND	K9	0V	
GND	K16	0V	
GND	L14	0V	
GND	M4	0V	
GND	N9	0V	
GND	N12	0V	
GND	N15	0V	
GND	P3	0V	
GND	P7	0V	
GND	R8	0V	
GND	R10	0V	
GND	R11	0V	
GND	R13	0V	
GND	T1	0V	
GND	T2	0V	
GND	T3	0V	
GND	T4	0V	
GND	T5	0V	
GND	T8	0V	
GND	T10	0V	
GND	T14	0V	
GND	T15	0V	
GND	T16	0V	
REFGND	E2	0V	

*Table 9: Supply Voltage Pins and Ground Pins*



<b>Name</b>	<b>Ball</b>	<b>IO</b>	<b><i>Not Connect (n.c.) Pins</i></b>
n.c.	A2	-	not connect
n.c.	A3	-	not connect
n.c.	A4	-	not connect
n.c.	A14	-	not connect
n.c.	A15	-	not connect
n.c.	B3	-	not connect
n.c.	B8	-	not connect
n.c.	B9	-	not connect
n.c.	B12	-	not connect
n.c.	B13	-	not connect
n.c.	C5	-	not connect
n.c.	C6	-	not connect
n.c.	C9	-	not connect
n.c.	C10	-	not connect
n.c.	C12	-	not connect
n.c.	C13	-	not connect
n.c.	D9	-	not connect
n.c.	D12	-	not connect
n.c.	E9	-	not connect
n.c.	E10	-	not connect
n.c.	E11	-	not connect
n.c.	E14	-	not connect
n.c.	E15	-	not connect
n.c.	E16	-	not connect
n.c.	F9	-	not connect
n.c.	F10	-	not connect
n.c.	F11	-	not connect
n.c.	F12	-	not connect
n.c.	F14	-	not connect
n.c.	F16	-	not connect
n.c.	G5	-	not connect
n.c.	G1	-	not connect
n.c.	G11	-	not connect
n.c.	G12	-	not connect



<b>Name</b>	<b>Ball</b>	<b>IO</b>	<b><i>Not Connect (n.c.) Pins</i></b>
n.c.	G14	-	not connect
n.c.	G6	-	not connect
n.c.	G15	-	not connect
n.c.	G16	-	not connect
n.c.	H5	-	not connect
n.c.	H6	-	not connect
n.c.	H1	-	not connect
n.c.	H11	-	not connect
n.c.	H12	-	not connect
n.c.	H15	-	not connect
n.c.	H2	-	not connect
n.c.	H3	-	not connect
n.c.	J1	-	not connect
n.c.	J2	-	not connect
n.c.	J3	-	not connect
n.c.	J5	-	not connect
n.c.	J6	-	not connect
n.c.	J11	-	not connect
n.c.	J12	-	not connect
n.c.	J14	-	not connect
n.c.	K2	-	not connect
n.c.	K5	-	not connect
n.c.	K6	-	not connect
n.c.	K11	-	not connect
n.c.	K12	-	not connect
n.c.	K14	-	not connect
n.c.	K15	-	not connect
n.c.	L3	-	not connect
n.c.	L6	-	not connect
n.c.	L7	-	not connect
n.c.	L8	-	not connect
n.c.	L9	-	not connect
n.c.	L10	-	not connect
n.c.	L11	-	not connect



<b>Name</b>	<b>Ball</b>	<b>IO</b>	<b><i>Not Connect (n.c.) Pins</i></b>
n.c.	L12	-	not connect
n.c.	M6	-	not connect
n.c.	M7	-	not connect
n.c.	M8	-	not connect
n.c.	M9	-	not connect
n.c.	M10	-	not connect
n.c.	M11	-	not connect
n.c.	M14	-	not connect
n.c.	M15	-	not connect
n.c.	M16	-	not connect
n.c.	N3	-	not connect
n.c.	N4	-	not connect
n.c.	N5	-	not connect
n.c.	N14	-	not connect
n.c.	N16	-	not connect
n.c.	P1	-	not connect
n.c.	P5	-	not connect
n.c.	P6	-	not connect
n.c.	P13	-	not connect
n.c.	P14	-	not connect
n.c.	P16	-	not connect
n.c.	R2	-	not connect
n.c.	R3	-	not connect
n.c.	R4	-	not connect
n.c.	R5	-	not connect
n.c.	R8	-	not connect
n.c.	R9	-	not connect
n.c.	R10	-	not connect
n.c.	R11	-	not connect
n.c.	R12	-	not connect
n.c.	R14	-	not connect
n.c.	R15	-	not connect
n.c.	R16	-	not connect
n.c.	T15	-	not connect



<b>Name</b>	<b>Ball</b>	<b>IO</b>	<b>Not Connect (n.c.) Pins</b>
n.c.	T14	-	not connect
n.c.	P8	-	not connect
n.c.	P9	-	not connect
n.c.	P10	-	not connect
n.c.	P12	-	not connect
n.c.	P11	-	not connect
n.c.	P15	-	not connect
n.c.	T2	-	not connect
n.c.	T3	-	not connect
n.c.	T4	-	not connect
n.c.	T5	-	not connect
n.c.	T8	-	not connect
n.c.	T9	-	not connect
n.c.	T11	-	not connect
n.c.	T12	-	not connect

*Table 10: List of pins that must not be connected (must be left open)*



## 7 Package Dimensions

Package: FBGA 256 balls, 1.0mm pitch, size 17mm x 17mm, industrial temperature range -40°C . . . 100°C, RoHS compliant.

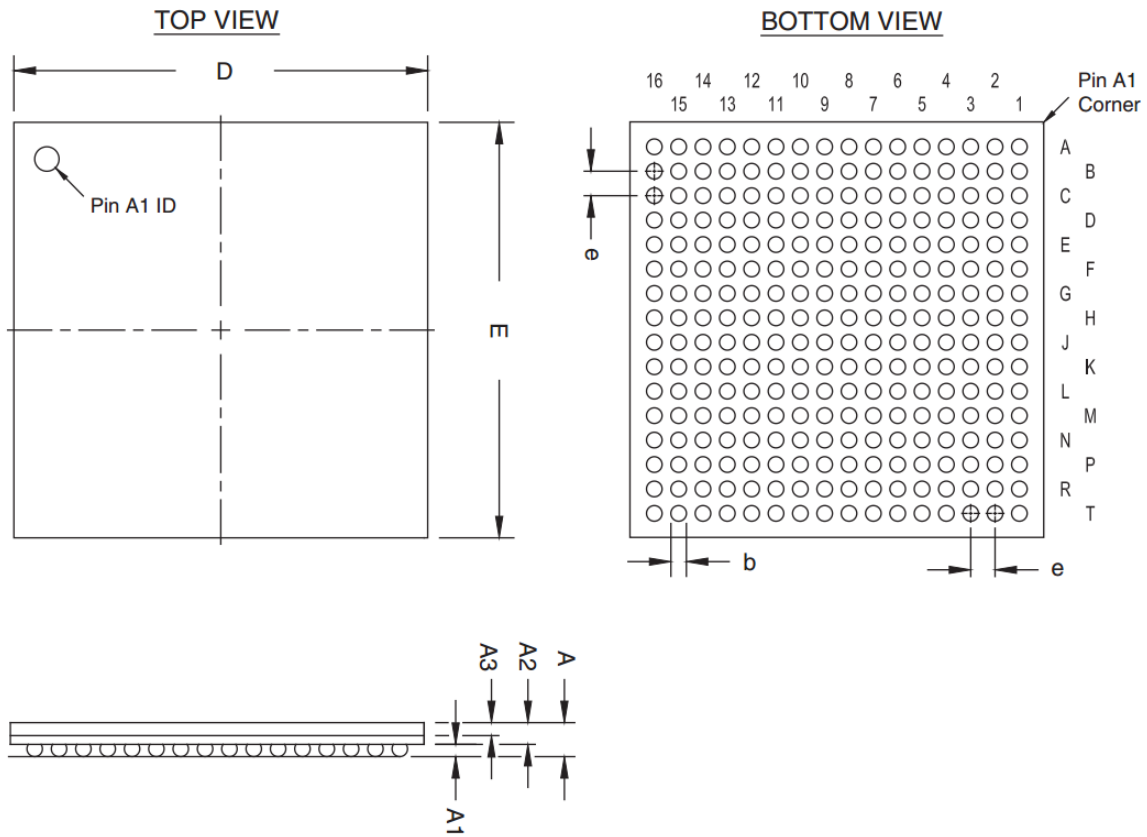


Figure 17: FBGA256 Package Outline

FBGA256 Package Dimensions in mm			
	min.	typ.	max.
A	-	-	2.20
A1	0.30	-	-
A2	-	-	1.80
A3		0.70 REF	
D		17.00 BSC	
E		17.00 BSC	
b	0.50	0.60	0.70
e		1.00 BSC	

JEDEC MS-034, Variation: AAF-1

Table 11: Package Outline Dimensions



## 8 Characteristics

### 8.1 Absolute Maximum Ratings

This section defines the maximum operating conditions for the device. Normal operation must occur within recommended operation conditions at any time. The functional operation of the device is not implied for these absolute maximum rating conditions.

Symbol	Parameter	Condition	min.	typ.	max.	Unit
VCC3V3	DC supply voltage IO		-0.5		3.9	V
VCC2V5	DC analog supply voltage		-0.5		-3.4	V
VCC1V2	DC core supply voltage		-0.5		1.6	V
$V_I$	DC input voltage		-0.5		3.6	V
$V_O$	DC output voltage		0		VCC3V3	V
$T_{JOP}$	Junction temperature range (industrial)		-40	25	100	°C
$T_{STG}$	Storage temperature range		-65		150	°C

Table 12: Absolute Maximum Ratings

#### **i** Info

Conditions out of absolute maximum rating range may cause permanent damage to the device. Operation at absolute maximum ratings for extended time periods may cause damage of the device.

### 8.2 Recommended Operation Conditions

Symbol	Parameter	Condition	min.	typ.	max.	Unit
VCC3V3	DC supply voltage IO		3.14	3.3	3.465	V
VCC2V5	DC analog supply voltage		2.375	2.5	2.625	V
VCC1V2	DC core supply voltage		1.15	1.2	1.25	V

Table 13: Recommended Operation Conditions





### 8.3 DC Characteristics

Symbol	Parameter	Condition	min.	typ.	max.	Unit
$V_{IH}$	High level input voltage threshold		1.7		3.6	V
$V_{IL}$	Low level input voltage threshold		0.3		0.8	V
$V_{IHY}$	Schmitt-Trigger hysteresis				180	mV
$V_{OH}$	High level output voltage		VCC3V3 - 0.2			V
$V_{OL}$	Low level output voltage				0.2	V
$I_{LEAK}$	Input leakage current		-10		10	$\mu$ A
$I_{OPINDC}$	DC output current per pin				300	$\mu$ A
$I_{OPINAC}$	DC output current per pin				8	mA

Table 14: Recommended Operation Conditions

### 8.4 Timing Characteristics

Symbol	Parameter	Condition	min.	typ.	max.	Unit
fCLK	Clock frequency	quarz stabilized clock		25		MHz
tCLK	Clock period			40		ns
tCLK <sub>H</sub>	Clock high time			20		ns
tCLK <sub>L</sub>	Clock low time			20		ns
tSU	setup time	relative to rising CLK edge		-0.75		ns
tHD	hold time	relative to rising CLK edge		1.25		ns
tPD	propagation delay time	relative to rising CLK edge		5		ns

Table 15: Characteristics

### 8.5 Power Dissipation

Symbol	Parameter	Condition	min.	typ.	max.	Unit
PD3V3	IO supply power dissipation	25°C		75		mW
PD2V5	Analog supply power dissipation	25°C		5		mW
PD1V2	Core power dissipation	25°C		200		mW

Table 16: Recommended Operation Conditions



## 9 Supplemental Directives

### 9.1 Producer Information

### 9.2 Copyright

TRINAMIC owns the content of this user manual in its entirety, including but not limited to pictures, logos, trademarks, and resources. © Copyright 2018 TRINAMIC. All rights reserved. Electronically published by TRINAMIC, Germany.

Redistributions of source or derived format (for example, Portable Document Format or Hypertext Markup Language) must retain the above copyright notice, and the complete Datasheet User Manual documentation of this product including associated Application Notes; and a reference to other available product-related documentation.

### 9.3 Trademark Designations and Symbols

Trademark designations and symbols used in this documentation indicate that a product or feature is owned and registered as trademark and/or patent either by TRINAMIC or by other manufacturers, whose products are used or referred to in combination with TRINAMIC's products and TRINAMIC's product documentation.

This Datasheet is a non-commercial publication that seeks to provide concise scientific and technical user information to the target user. Thus, trademark designations and symbols are only entered in the Short Spec of this document that introduces the product at a quick glance. The trademark designation /symbol is also entered when the product or feature name occurs for the first time in the document. All trademarks and brand names used are property of their respective owners.

### 9.4 Target User

The documentation provided here, is for programmers and engineers only, who are equipped with the necessary skills and have been trained to work with this type of product.

The Target User knows how to responsibly make use of this product without causing harm to himself or others, and without causing damage to systems or devices, in which the user incorporates the product.

### 9.5 Disclaimer: Life Support Systems

TRINAMIC Motion Control GmbH & Co. KG does not authorize or warrant any of its products for use in life support systems, without the specific written consent of TRINAMIC Motion Control GmbH & Co. KG.

Life support systems are equipment intended to support or sustain life, and whose failure to perform, when properly used in accordance with instructions provided, can be reasonably expected to result in personal injury or death.

Information given in this document is believed to be accurate and reliable. However, no responsibility is assumed for the consequences of its use nor for any infringement of patents or other rights of third parties which may result from its use. Specifications are subject to change without notice.

### 9.6 Disclaimer: Intended Use

The data specified in this user manual is intended solely for the purpose of product description. No representations or warranties, either express or implied, of merchantability, fitness for a particular purpose



or of any other nature are made hereunder with respect to information/specification or the products to which information refers and no guarantee with respect to compliance to the intended use is given.

In particular, this also applies to the stated possible applications or areas of applications of the product. TRINAMIC products are not designed for and must not be used in connection with any applications where the failure of such products would reasonably be expected to result in significant personal injury or death (safety-Critical Applications) without TRINAMIC's specific written consent.

TRINAMIC products are not designed nor intended for use in military or aerospace applications or environments or in automotive applications unless specifically designated for such use by TRINAMIC. TRINAMIC conveys no patent, copyright, mask work right or other trade mark right to this product. TRINAMIC assumes no liability for any patent and/or other trade mark rights of a third party resulting from processing or handling of the product and/or any other use of the product.

## 9.7 Collateral Documents & Tools

This product documentation is related and/or associated with additional tool kits, firmware and other items, as provided on the product page at: [www.trinamic.com](http://www.trinamic.com).



## 10 Figures Index

1	Hardware FOC Application Diagram . . . . .	6	9	Encoder ABN Timing . . . . .	17
2	Hardware FOC Block Diagram . . . . .	6	10	Hall Sensor Angles . . . . .	18
3	SPI Timing . . . . .	7	11	nPolePairsNumberOfPolePairs . . . . .	19
4	SPI datagram Structure . . . . .	8	12	PID Architectures and Motion Modes . . . . .	22
5	nPolePairsNumberOfPolePairs . . . . .	11	13	Inner FOC Control Loop . . . . .	23
6	Integer Representation of Angles with 16 Bit as s16 resp. u16 . . . . .	11	14	FOC Transformations . . . . .	24
7	ADC Selector and Scaler with Offset Correction . . . . .	14	15	Motion Modes . . . . .	24
8	ABN Incremental Encoder N Pulse any- where between 0° and 360° . . . . .	16	16	BBM Timing . . . . .	26
			17	FBGA256 Package Outline . . . . .	95



## 11 Tables Index

1	Order codes . . . . .	4	10	Functional Pin Description . . . . .	94
2	TABspiTimingParameter . . . . .	7	11	Package Outline Dimensions . . . . .	95
3	Numerical Representations . . . . .	9	12	Absolute Maximum Ratings . . . . .	96
4	Examples of u16, s16, q8.8, q4.12 . . . . .	10	13	Recommended Operation Conditions . . . . .	96
5	Examples of u16, s16, q8.8 . . . . .	12	14	Recommended Operation Conditions . . . . .	97
6	Register Map for TMC4670 . . . . .	84	15	Timing Characteristics . . . . .	97
7	Pin Type Definition . . . . .	85	16	Recommended Operation Conditions . . . . .	97
8	Functional Pin Description . . . . .	87	17	IC Revision . . . . .	102
9	Supply Voltage Pins and Ground Pins . . . . .	90	18	Document Revision . . . . .	102



## 12 Revision History

### 12.1 IC Revision

Version	Date	Author	Description
V0.90	2016-JUN-30	LL	Preliminary release.
V0.99	2016-SEP-30	LL	Engineering Sample Release.
V1.01	2016-DEC-21	LL	First Revision for Production.

Table 17: IC Revision

### 12.2 Document Revision

Version	Date	Author	Description
V0.9	2016-MAY-23	LL	Initial version.
V0.9	2016-JUL-29	LL	First draft committed.
V0.91	2016-OCT-13	LL	Functional summary added, updated register set of TMC4670 0v99, ADC selector with scaling and offset correction added, Analog Encoder (AENC) selector with scaling and offset correction added,
V0.91	2016-OCT-19	LL	Pinning updated.
V0.91	2016-NOV-02	LL	Short Spec Block Diagram added.
V0.91	2016-NOV-04	LL	PID architecture and PID motion mode drawings added
V0.91	2016-NOV-07	LL	Functional description hierachies updated
V0.91	2016-NOV-08	LL	First version for www.trinamic.com
V0.99	2017-MAR-01	LL, OM	Typos and minor corrections
V1.00	2017-MAR-08	LL	Characteristics added, table form updated
V1.00	2017-MAR-10	LL	Characteristics added, table form updated
V1.00	2017-APR-27	LL	n.c. (not connected) corrected to (not connect); pin table updated; size of package corrected in order codes table;
V1.00	2018-MAR-08	LL	industrial temperature range updated;

Table 18: Document Revision

