## FUNCTION

System registers and interfaces for motor Field Oriented Control (FOC).

## VHDL File

mosysfoc.vhd

## Applicable Devices

Spartan3ADSP, Spartan6, 7-Family, UltraScale+

## Xilinx primitive used

DSP48A/A1/E1
RAMB16_S18_S18

## Sub modules used

motorfoc.vhd

## Execution time

-

## Introduction

This module is an interface toward the system registers of the motorfoc (motor Field Oriented Control) module. It implements the addressing of the registers and the read and write process.

## PARAMETERS

| Parameter | Type | Values | Default | Description |
|---|---|---|---|---|
| C_FAMILY | string | spartan3adsp<br>spartan6<br>artix7<br>kintex7<br>virtex7<br>zynq | zynq | Xilinx FPGA Family name |
| Inverter analog inputs | | | | |
| C_INV_IN_MAP[11:0] | Std_logic_vector | 0x000..0xFFF | 0x083 | Bit map enabled input channels:<br>0=IPHS_A<br>1=IPHS_B<br>2=IPHS_C<br>3=IBUS_X<br>4=VPHS_A<br>5=VPHS_B<br>6=VPHS_C<br>7=VBUS_X<br>8=VPHS_N |
| C_INV_IN_NOT[11:0] | Std_logic_vector | 0..511 | 0x000 | Bit map inverted channels. Bit values as per C_INV_IN_MAP |
| C_INV_OFSV_MODE | Integer | 0..3 | 1 | Offset set mode functions bit definition:<br>0=Self-Zero<br>1=S/W registers |
| C_INV_OVER_IPHS | integer | 0..1 | 1 | Overcurrent detection motor phases |
| C_INV_OVER_IBUS | integer | 0..1 | 1 | Overcurrent detection dc_link |
| C_INV_FILTER | Integer | 0..1 | 1 | 2nd order LPF inputs |
| C_INV_EVAL_VPHS | Integer | 0..1 | 1 | Evaluation of Vs |
| C_INV_R2P_VPHS | Integer | 0..1 | 1 | Rectangular to polar instance for Vs |
| C_CLARKE_NPHS | Integer | 0,2,3 | 3 | Clarke transform input phases.<br>0=transparent (2-phases only for bipolar stepper motor)<br>2=2-phases used in 3-phases motor<br>3=3-phases used in 3-phases motor |

| | | | | |
|---|---|---|---|---|
| C_DCPL_KJFF | Integer | 0..1 | 1 | Enable Decoupling function after PI control |
| C_VSVEC_AJFF | Integer | 0..1 | 0 | Enable user Vs feed forward |
| C_BEMF_KLIN | Integer | 0..1 | 1 | Enable BEMF Linearization LUT |
| C_PWM_MODULATOR | integer | 0..2 | 1 | Include PWM modulator IP |
| C_RPFM_MODULATOR | integer | 0..2 | 1 | Include RPFM modulator IP |
| C_RPFM_3_LEVEL | integer | 0..1 | 1 | RPFM 3-level extension |
| C_RPFM_TPNC | integer | 0..1 | 1 | RPFM 3-level T-PNC variant |
| C_SMO_EVAL | integer | 0..1 | 1 | Include SMO Position Estimator IP |
| C_SPD_EVAL | integer | 0..1 | 1 | Include Speed Measurement IP |
| C_SPEED_CTRL | Integer | 0..1 | 1 | Include speed loop |
| | | | | |
| C_EXTANGLE | integer | 0..2 | 1 | Include external angle sensor IP |
| C_RESOLVER | integer | 0..2 | 1 | Include resolver sensor IP |
| C_RSV_EXCMIN | integer | 1..65536 | 50000 | Resolver exciter minimum value |
| C_RSV_SOCMIN | integer | 1..65536 | 50000 | Resolver SIN/COS minimum value |
| C_RSV_EXCTMO | integer | 1..65536 | 100 | Resolver timeout for exciter |
| C_RSV_EXCPOL | integer | 0..2 | 0 | Exciter polarity: 0=both, 1=positive, 2=negative |
| C_RSV_ANGLUT | integer | 0..3 | 0 | Resolver LUT enable 0=disable, 1=1K, 2=2K, 3=4K |
| C_RSV_ANGFLT | integer | 0..1 | 0 | Resolver angle filter enable |
| C_RSV_ALRELK | integer | 0..18,255 | 5 | Resolver angle LUT mode: 0..18=relative, 255=absolute |
| | | | | |
| C_HALLSENSOR | integer | 0..1 | 1 | Include hall sensor IP |

| | | | | |
|---|---|---|---|---|
| C_ENCODER | Integer | 0..1 | 2 | Include enhanced 2..3 wire encoder IP |
| C_MANROT | Integer | 0..2 | 2 | Scalar mode rotor angle IP: 0=no, 1=angle update, 2=enhanced with ramp and 2nd order filter |
| C_MRT_ACCMAX_DLN2 | Integer | 8..16 | 16 | Base 2 logarithm of MRT acceleration limiter |
| C_DCK_SP_DLN2 | Integer | 0..n | 9 | Base 2 logarithm of speed divider in decoupling |
| C_DCK_KE_DLN2 | Integer | 0..n | 24 | Base 2 logarithm of magnetic flux constant divider in decoupling |
| C_DCK_KL_DLN2 | Integer | 0..n | 41 | Base 2 logarithm of electromagnetic flux constant divider in decoupling |
| C_SMO_ZS_DLN2 | integer | 22..26 | 24 | Base 2 logarithm of proportional error divisor |
| C_SMO_F2_DLN2 | integer | 26..30 | 28 | Base 2 logarithm divider use to eval K of 2nd LPF |
| | | | | |
| | | | | |
| C_SLP_PRO_DLN2 | Integer | 0..n | 1 | Base 2 logarithm of speed loop proportion error regulator |
| C_SLP_INT_DLN2 | Integer | 0.n | 5 | Base 2 logarithm of speed loop integrative error regulator |
| C_SLP_INDWP_DLN2 | Integer | 0.n | 1 | Base 2 logarithm of speed loop integrative error regulator for anti windup |
| C_SLP_INDWP_KDIV | Integer | 0.n | 1 | Base 2 logarithm of speed loop integrative error regulator for anti windup |
| | | | | |
| C_ISOVERFLOW_CMAX | integer | 1..15 | 1 | Current modulo overflow counter limit |

| | | | | |
|---|---|---|---|---|
| C_PI_ERPRO_DLN2 | integer | 0..n | 12 | Base 2 logarithm of current loop PI proportional error divisor |
| C_PI_ERINT_DLN2 | integer | 0..n | 18 | Base 2 logarithm of current loop PI integrative error divisor |

## SIGNALS

| Signal | I/O | Description |
|---|---|---|
| S_AXI_ACLK | IN | Clock (rising edge). |
| S_AXI_ARESETN | IN | Reset. Active low. |
| | | |
| S_AXI: slave AXI write access signals | | |
| S_AXI: slave AXI read access signal | | |
| Core to Core multiple register access | | |
| mckc_rdata[4095:0] | OUT | Register Data Read. 128 registers, 32-bits each |
| mckc_wdata[4095:0] | IN | Register Data Write. 128 registers, 32-bits each |
| mckc_wstrb[127:0] | IN | Register Strobe Write. 128 registers |
| | | |
| Motor control Interface | | |
| | | |
| paipol[7:0] | OUT | Motor pole pairs |
| | | |
| ext_0_angofs[15:0] | OUT | External_0 electric angle offset |
| ext_0_angle[31:0] | IN | External_0 electric angle value |
| | | |
| ext_1_angofs[15:0] | OUT | External_1 electric angle offset |
| ext_1_angle[31:0] | IN | External_1 electric angle value |
| | | |
| rsv_m2rppk [7:0] | OUT | Motor pole pairs vs Resolver pole pairs |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| Analog resolver sensor inputs for position evaluation (see QD_TDS_124 for description) | | |
| rsv_0_angofs[15:0] | OUT | Resolver_0 angle offset |
| rsv_0_angle[31:0] | OUT | Resolver_0 angle value |
| rsv_0_mangle[31:0] | OUT | Resolver_0 mechanical angle |
| rsv_0_sync | OUT | Resolver_0 angle sync |
| rsv_0_msync | OUT | Resolver_0 mechanical angle sync |
| rsv_0_priexc[17:0] | IN | Resolver_0 primary winding exciter |
| rsv_0_secsin[17:0] | IN | Resolver_0 secondary winding sine |
| rsv_0_seccos[17:0] | IN | Resolver_0 secondary winding cosine |
| | | |
| rsv_1_angofs[15:0] | OUT | Resolver_1 angle offset |
| rsv_1_angle[31:0] | OUT | Resolver_1 angle value |

| | | |
|---|---|---|
| rsv_1_mangle[31:0] | OUT | Resolver_1 mechanical angle |
| rsv_1_sync | OUT | Resolver_1 angle sync |
| rsv_1_msync | OUT | Resolver_1 mechanical angle sync |
| rsv_1_priexc[17:0] | IN | Resolver_1 primary winding exciter |
| rsv_1_secsin[17:0] | IN | Resolver_1 secondary winding sine |
| rsv_1_seccos[17:0] | IN | Resolver_1 secondary winding cosine |
| | | |
| | | |
| | | |
| Hall sensor digital signals position evaluation (see QD_TDS_122 for description) | | |
| hls_angle[31:0] | OUT | Hall sensors angle |
| hls_hallsig[5:0] | IN | Hall sensors digital inputs |
| | | |
| Digital encoder position evaluation (see QD_TDS_101 for more details) | | |
| enc_angle[31:0] | OUT | Encode sensors angle |
| enc_cha | IN | Encoder channel A. |
| enc_chb | IN | Encoder channel B. |
| enc_chi | IN | Encoder channel index. |
| | | |
| Inverter analog, values are SIGNED18. | | |
| acq_iphs_a[17:0] | IN | Motor current Phase-A |
| acq_iphs_b[17:0] | IN | Motor current Phase-B |
| acq_iphs_c[17:0] | IN | Motor current Phase-C |
| acq_ibus_x[17:0] | IN | Dc_link current |
| acq_vphs_a[17:0] | IN | Motor voltage Phase-A |
| acq_vphs_b[17:0] | IN | Motor voltage Phase-B |
| acq_vphs_c[17:0] | IN | Motor voltage Phase-C |
| acq_vbus_x[17:0] | IN | Dc_link voltage |
| acq_sync | IN | Acquisition synchronization. |
| Inverter analog offset, values are SIGNED18. | | |
| ofs_iphs_a[17:0] | OUT | Motor current Phase-A |
| ofs_iphs_b[17:0] | OUT | Motor current Phase-B |
| ofs_iphs_c[17:0] | OUT | Motor current Phase-C |
| ofs_ibus_x[17:0] | OUT | Dc_link current |
| ofs_vphs_a[17:0] | OUT | Motor voltage Phase-A |
| ofs_vphs_b[17:0] | OUT | Motor voltage Phase-B |
| ofs_vphs_c[17:0] | OUT | Motor voltage Phase-C |
| ofs_vbus_x[17:0] | OUT | Dc_link voltage |
| Inverter analog normalized, values are SIGNED18. | | |
| nrm_iphs_a[17:0] | OUT | Motor current Phase-A |
| nrm_iphs_b[17:0] | OUT | Motor current Phase-B |
| nrm_iphs_c[17:0] | OUT | Motor current Phase-C |

| | | |
|---|---|---|
| nrm_ibus_x[17:0] | OUT | Dc_link current |
| nrm_vphs_a[17:0] | OUT | Motor voltage Phase-A |
| nrm_vphs_b[17:0] | OUT | Motor voltage Phase-B |
| nrm_vphs_c[17:0] | OUT | Motor voltage Phase-C |
| nrm_vbus_x[17:0] | OUT | Dc_link voltage |
| nrm_sync | OUT | Data synchronization. |
| Inverter analog filtered, values are SIGNED18. | | |
| flt_iphs_a[17:0] | OUT | Motor current Phase-A |
| flt_iphs_b[17:0] | OUT | Motor current Phase-B |
| flt_iphs_c[17:0] | OUT | Motor current Phase-C |
| flt_ibus_x[17:0] | OUT | Dc_link current |
| flt_vphs_a[17:0] | OUT | Motor voltage Phase-A |
| flt_vphs_b[17:0] | OUT | Motor voltage Phase-B |
| flt_vphs_c[17:0] | OUT | Motor voltage Phase-C |
| flt_vbus_x[17:0] | OUT | Dc_link voltage |
| flt_sync | OUT | Data synchronization. |
| | | |
| ovi_iphs_a | Out | Motor Phase-A overcurrent |
| ovi_iphs_b | Out | Motor Phase-B overcurrent |
| ovi_iphs_b | Out | Motor Phase-C overcurrent |
| ovi_ibus_x | Out | DC_link overcurrent |
| | | |
| Ismodover | Out | Is modulo overcurrent |
| | | |
| modtype[0:0] | IN | Selected modulator 0=PWM, 1=RPFM |
| modlevels[0:0] | IN | Modulator levels 0=2-levels, 1=3-levels |
| is3phases | IN | Three-phase motor selection flag. Active high (“1”). |
| PWM 2-phases, 2-levels output | | |
| pwm2p_l2c1w1 [1:0] | OUT | coil-1, winding-1 |
| pwm2p_l2c2w1 [1:0] | OUT | coil-2, winding-1 |
| pwm2p_l2c1w2 [1:0] | OUT | coil-1, winding-2 |
| pwm2p_l2c2w2 [1:0] | OUT | coil-2, winding-2 |
| pwm2p_l2sync | OUT | Sync signal |
| PWM 3-phases, 2-levels output | | |
| pwm3p_l2cxw1 [2:0] | OUT | Coil 321, winding-1 |
| pwm3p_l2cxw2 [2:0] | OUT | Coil 321, winding-2 |
| pwm3p_l2sync | OUT | Sync signal |
| RPFM 3-phases, 2-levels output | | |
| rpfm3p_l2cxw1 [2:0] | OUT | Coil 321, winding-1 |
| rpfm3p_l2cxw2 [2:0] | OUT | Coil 321, winding-2 |
| rpfm3p_l2sync | OUT | Sync signal |

| RPFM 3-phases, 3-levels output | | |
|---|---|---|
| rpfm3p_l3c1w1[1:0] | OUT | Coil1, winding-1 |
| rpfm3p_l3c2w1[1:0] | OUT | Coil2, winding-1 |
| rpfm3p_l3c3w1[1:0] | OUT | Coil3, winding-1 |
| rpfm3p_l3c1w2[1:0] | OUT | Coil1, winding-2 |
| rpfm3p_l3c2w2[1:0] | OUT | Coil2, winding-2 |
| rpfm3p_l3c3w2[1:0] | OUT | Coil3, winding-2 |
| rpfm3p_l3sync | OUT | Sync signal |
| | | |
| deadtval[7:0] | Out | UNSIGNED8 gate unit dead time valid code |
| | | |
| gwswcnt[31:0] | IN | UNSIGNED32, gate unit switches diagnostic counter |
| | | |
| hwfail | IN | Gate unit h/w failure |
| hwkill | IN | Gate unit KILL input (fast cut-off) |
| | | |
| coilgear | OUT | Coil signal gear |
| coilenab | OUT | Coil signals enable |
| | | |
| syserr | OUT | Motor control IP system error |
| | | |
| reseterr | OUT | Reset latched errors |
| zerocurr | OUT | Zero current offset |
| zerovbusx | Out | Zero DC_link offset command |
| zerovphsx | Out | Zero motor phases command |
| Current control PI probe/signals | | |
| pix_setval[17:0] | OUT | SIGNED18. Target X-frame |
| piy_setval[17:0] | OUT | SIGNED18. Target Y-frame |
| pix_fbkval[17:0] | OUT | SIGNED18. Feedback X-frame |
| piy_fbkval[17:0] | OUT | SIGNED18. Feedback Y-frame |
| pix_outpro[17:0] | OUT | SIGNED18. Proportional Output X-frame |
| piy_outpro[17:0] | OUT | SIGNED18. Proportional Output y-frame |
| pix_outint[17:0] | OUT | SIGNED18. Integrative Output X-frame |
| piy_outint[17:0] | OUT | SIGNED18. Integrative Output X-frame |
| pix_outval[17:0] | OUT | SIGNED18. Total Output X-frame |
| piy_outval[17:0] | OUT | SIGNED18. Total Output Y-frame |
| Decoupling (feed forward contribute) | | |
| dck_er_out_x[17:0] | OUT | SIGNED18. Bemf Output X-frame |
| dck_er_out_y[17:0] | OUT | SIGNED18. Bemf Output Y-frame |
| dck_vs_out_x[17:0] | OUT | SIGNED18. Total Output X-frame |
| dck_vs_out_y[17:0] | OUT | SIGNED18. Total Output Y-frame |

| | | |
|---|---|---|
| nparkangofs[15:0] | OUT | Spare/no more implemented |
| | | |
| vsmodval[16:0] | OUT | UNSIGNED17, Vs module |
| vsangval[31:0] | OUT | Vs angle |
| | | |
| ismodval[16:0] | OUT | UNSIGNED17, Is module |
| isangval[31:0] | OUT | Is angle |
| | | |
| vamodval[16:0] | OUT | UNSIGNED17, Va module (measured Vs) |
| vaangval[31:0] | OUT | Va angle (measured Vs) |
| | | |
| smo_bemf_p[31:0] | OUT | SMO Zs angle |
| smo_bemf_m[16:0] | OUT | UNSIGNED17, SMO Zs module |
| smo_angle[31:0] | OUT | SMO angle |
| | | |
| mrt_angle[31:0] | OUT | Scalar mode IP angle |
| | | |
| rot_angle[31:0] | OUT | Rotor angle |
| rot_select[3:0] | OUT | Rotor selector |
| | | |
| spd_speed [31:0] | OUT | Motor speed |
| pos_position[31:0] | OUT | Motor position |
| | | |
| tsclocks[11:0] | OUT | Ts cycle time in clock units |

## Detailed Description

This module implements the system register for the set up and run time execution of the ***motorfoc*** module.

In the following section is reported the map of the registers used by the module.
All registers are 32 bits size. The register offset is "register index" * 4.

### Registers description table

| REGISTERS DESCRIPTION | | | |
|---|---|---|---|
| **Register index** | **Register name** | **Access** | **Description** |
| 0 | ip_ident | *READ* | Motor control IP identification |
| | | | |
| 1 | tsclocks | *READ* | Measured cycle time (clocks) |
| | | | |
| 2 | motor_control | *R/W* | Motor control |
| 3 | motor_status | *READ* | Motor status / event |
| 4 | val_iphs_a | *READ* | Current phase-A |
| 5 | val_iphs_b | *READ* | Current phase-B |
| 6 | val_iphs_c | *READ* | Current phase-C |
| 7 | val_ibus_x | *READ* | Current DC_link |
| 8 | val_vphs_a | *READ* | Voltage phase-A |
| 9 | val_vphs_b | *READ* | Voltage phase-B |
| 10 | val_vphs_c | *READ* | Voltage phase-C |
| 11 | val_vbus_x | *READ* | Voltage DC_link |
| 12 | val_vphs_n | *READ* | Voltage neutral |
| 13 | ofs_iphs_a | *R/W* | Current phase-A |
| 14 | ofs_iphs_b | *R/W* | Current phase-B |
| 15 | ofs_iphs_c | *R/W* | Current phase-C |
| 16 | ofs_ibus_x | *R/W* | Current DC_link |
| 17 | ofs_vphs_a | *R/W* | Voltage phase-A |
| 18 | ofs_vphs_b | *R/W* | Voltage phase-B |
| 19 | ofs_vphs_c | *R/W* | Voltage phase-C |
| 20 | ofs_vbus_x | *R/W* | Voltage DC_link |
| 21 | ofs_vphs_n | *R/W* | Voltage neutral |
| 22 | mul_iphs_a | *R/W* | Current phase-A |
| 23 | mul_iphs_b | *R/W* | Current phase-B |
| 24 | mul_iphs_c | *R/W* | Current phase-C |
| 25 | mul_ibus_x | *R/W* | Current DC_link |
| 26 | mul_vphs_a | *R/W* | Voltage phase-A |
| 27 | mul_vphs_b | *R/W* | Voltage phase-B |
| 28 | mul_vphs_c | *R/W* | Voltage phase-C |
| 29 | mul_vbus_x | *R/W* | Voltage DC_link |

| REGISTERS DESCRIPTION | | | |
|---|---|---|---|
| **Register index** | **Register name** | **Access** | **Description** |
| 30 | mul_vphs_n | *R/W* | Voltage neutral |
| | | | |
| | | | |
| 31 | xbus_fk1 | *R/W* | dc_link LPFT1 #1 const |
| 32 | xbus_fk2 | *R/W* | dc_link LPFT1 #2 const |
| | | | |
| 33 | xphs_fk1 | *R/W* | Phases LPFT1 #1 const |
| 34 | xphs_fk2 | *R/W* | Phases LPFT1 #2 const |
| | | | |
| 35 | ibus_limit | *R/W* | Dc_link current limit |
| 36 | iphs_limit | *R/W* | Phases current limit |
| | | | |
| 37 | ismodmax | *R/W* | Is current limit |
| | | | |
| 38 | ismodval | *READ* | Is real time current value |
| 39 | isangval | *READ* | Is real time angle value |
| | | | |
| 40 | vamodval | *READ* | Va real time modulo |
| 41 | vaangval | *READ* | Va real time angle |
| | | | |
| 42 | vsmodval | *READ* | Vs real time modulo |
| 43 | vsangval | *READ* | Vs real time angle |
| | | | |
| 44 | vs_rotor_x | *READ* | Vs rotor reference X-frame |
| 45 | vs_rotor_y | *READ* | Vs rotor reference Y-frame |
| | | | |
| 46 | rot_angdef | *R/W* | Rotor position reset angle |
| 47 | rot_angle | *READ* | Rotor position angle |
| | | | |
| Speed evaluation registers | | | |
| 48 | spd_fktau1 | *R/W* | Speed evaluation LPFT1 #1 const |
| 49 | spd_fktau2 | *R/W* | Speed evaluation LPFT1 #2 const |
| 50 | spd_speed | *READ* | Speed evaluation current speed |
| | | | |
| SMO Position evaluation registers | | | |
| 51 | smo_smontmr | *R/W* | On timer |
| 52 | smo_vs_mult | *R/W* | Vs multiplier |
| 53 | smo_is_mult | *R/W* | Is multiplier |
| 54 | smo_zs_max | *R/W* | BEMF max error |
| 55 | smo_es1_kflt | *R/W* | BEMF LPF1 #1 |
| 56 | smo_es2_kflo | *R/W* | LPF1 #2 base |

| REGISTERS DESCRIPTION | | | |
|---|---|---|---|
| **Register index** | **Register name** | **Access** | **Description** |
| 57 | smo_es2_kfmx | *R/W* | LPF1 #2 mult |
| 58 | smo_es2_kflt | *READ* | BEMF LPF1 #2 |
| 59 | smo_angofs | *R/W* | angle offset |
| 60 | smo_bemf_p | *READ* | BEMF Angle |
| 61 | smo_bemf_m | *READ* | BEMF Modulo |
| 62 | smo_angle | *READ* | Output Angle |
| | | | |
| 63 | pos_position | *READ* | Low resolution current position |
| | | | |
| Speed Loop Control | | | |
| 64 | slp_spdset | *R/W* | Speed set point |
| 65 | slp_kmpro | *R/W* | Proportional gain |
| 66 | slp_kmint | *R/W* | Integrative gain |
| 67 | slp_outlim | *R/W* | Current limit |
| 68 | slp_kmultx | *R/W* | Current X multiplier |
| 69 | slp_kmulty | *R/W* | Current Y multiplier |
| | | | |
| 70 | paipol | *R/W* | Motor pair poles (1..n) |
| External rotor angle sensor | | | |
| 71 | ext_0_angofs | *R/W* | External_0 angle offset |
| 72 | ext_1_angofs | *R/W* | External_1 angle offset |
| 73 | ext_0_angle | *READ* | External_0 angle |
| 74 | ext_1_angle | *READ* | External_1 angle |
| Resolver angle sensor | | | |
| 75 | rsv_m2rppk | *R/W* | Motor to Resolver Pair Poles Ratio |
| 76 | rsv_spdfkt | *R/W* | Resolver speed LPF1 filter |
| 77 | rsv_angfkt | *R/W* | Resolver angle LPF1 filter |
| 78 | rsv_0_angofs | *R/W* | Resolver_0 angle offset |
| 79 | rsv_1_angofs | *R/W* | Resolver_1 angle offset |
| 80 | rsv_0_angle | *READ* | Resolver_0 angle |
| 81 | rsv_1_angle | *READ* | Resolver_1 angle |
| Hall sensor | | | |
| 82 | hls_ctolim | *R/W* | Counter timeout limit |
| 83 | hls_angle | *READ* | Hall sensor angle |
| | | | |
| Incremental encoder | | | |
| 84 | enc_index | *READ* | Encoder index |
| 85 | enc_phase | *READ* | Encoder phase |
| 86 | enc_phcpt | *READ* | Encoder phase hold |
| 87 | enc_cyprnd | *R/W* | Cycles per round |
| 88 | enc_angphs | *R/W* | Encoder angle increments per phase |

| REGISTERS DESCRIPTION | | | |
|---|---|---|---|
| **Register index** | **Register name** | **Access** | **Description** |
| 89 | enc_angle | *READ* | Encoder Electric rotor angle |
| Manual rotor angle | | | |
| 90 | mrt_speed | *R/W* | Encoder simulator angle increments per netmot s-link packet (see detailed description) |
| 91 | mrt_accmax | *R/W* | Acceleration limit |
| 92 | mrt_fktau1 | *R/W* | First LPF1 filter speed |
| 93 | mrt_fktau2 | *R/W* | Second LPF1 filter speed |
| 94 | mrt_spdout | *READ* | Speed set |
| 95 | mrt_angle | *READ* | Encoder simulator Electric rotor |
| | | | |
| Pi_control regulator for X-frame or D-frame and Y-frame or Q-frame | | | |
| 96 | pi_setvalx | *R/W* | Target X-frame |
| 97 | pi_setvaly | *R/W* | Target Y-frame |
| 98 | pi_kmprox | *R/W* | Proportional gain X-frame |
| 99 | pi_kmproy | *R/W* | Proportional gain Y-frame |
| 100 | pi_kmintx | *R/W* | Integrative error gain X-frame |
| 101 | pi_kminty | *R/W* | Integrative error gain Y-frame |
| Decupling, PI extension for feed forward | | | |
| 102 | dck_ldkmul | R/W | Motor Ld multiplier (ele. Flux) |
| 103 | dck_lqkmul | R/W | Motor Lq multiplier (ele. Flux) |
| 104 | dck_kekmul | R/W | Motor Ke multiplier (mag. Flux) |
| User feed forward voltage vector | | | |
| 105 | vs_ajff_x | *R/W* | Vs vector X-frame |
| 106 | vs_ajff_y | *R/W* | Vs vector Y-frame |
| | | | |
| Bemf feed forward linearization LUT | | | |
| 107 | bemfangofs | R/W | Angle offset |
| | | | |
| Common modulators | | | |
| 108 | mod2angskw | R/W | Second modoulator angle skew |
| | | | |
| | | | |
| PWM modulator | | | |
| 109 | pwm_angofs | *R/W* | Angle offset |
| 110 | pwm_kmod | *R/W* | gain multiplier |
| 111 | pwm_presc | *R/W* | Frequency prescaler |
| 112 | pwm_mdmax | *R/W* | Modulation limit |
| 113 | pwm_mdval | *READ* | real time modulation value |
| 114 | pwm_minpw | *R/W* | Minimum allowed pulse width |
| 115 | pwm_ctrl | *R/W* | Control register |
| | | | |

| REGISTERS DESCRIPTION | | | |
|---|---|---|---|
| **Register index** | **Register name** | **Access** | **Description** |
| RPFM modulator | | | |
| 116 | rpfm_angofs | *R/W* | Angle offset |
| 117 | rpfm_ctrl | *R/W* | Control register |
| | | | |
| 118 | deadtval | *R/W* | Gate unit dead time value |
| 119 | gwswcnt | *R* | Gate unit switches counter |
| | | | |
| LUT access | | | |
| 120 | lut_addr | *R/W* | LUT address register |
| 121 | lut_data | *R/W* | LUT data register |
| | | | |

### *Motor_ip_ident - Motor IP identification register*

This register reports the IP identification register.
The current value is for test purpose only.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 24..31 | Ip_id_code | READ | 1 | Code |
| 16..23 | Ip_id_majv | READ | 1 | Major version |
| 8..15 | Ip_id_minv | READ | 2 | Minor version |
| 0..7 | Ip_id_hscrt | READ | 97 | H/W S/W compatibility |
| | | | | |

### *TSclocks – FOC clock cycles*

This register reports the number of system clocks between two consecutive FOC activation

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 12..31 | | READ | 0 | Unused/reserved |
| 11..0 | tsclocks | READ | - | System clocks on FOC cycle |
| | | | | |

### *Motor_control – Motor control register*

This register controls the FOC IP.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 29..31 | | READ | 0 | Unused/reserved |
| 28 | pi_himdovf | R/W | 0 | Enable lock PI integration when modulator overflow |
| 27 | smo_usevsact | R/W | 0 | SMO Vs selector: 0=target, 1=measured |
| 26 | zerovphsx | R/W | 0 | Zero motor voltage phases (0->1 only) |
| 25 | zerovbusx | R/W | 0 | Zero dc_link voltage (0->1 only) |
| 24 | reseterr | R/W | 0 | Reset latched error |
| 23 | zerocurr | R/W | 0 | Zero current offset command (0->1 only) |
| 22 | slp_mrt2spd | R/W | 0 | Pos&Speed: MRT to Speed loop link |
| 21 | slp_spd2trq | R/W | 0 | Speed loop feeds the current loop |
| 20 | enc_xmcha | R/W | 0 | Encoder CH-A match for index event |
| 19 | enc_xmchb | R/W | 0 | Encoder CH-B match for index event |
| 18 | enc_xmchi | R/W | 0 | Encoder CH-I match for index event |
| 17 | enc_inten | R/W | 0 | Encoder interpolator enable |
| 16 | enc_rstang | R/W | 0 | Encoder electric rotor angle evaluation reset control 1=reset-lock, 0=running |
| 15 | dck_useisset | READ | 0 | Used Is target in decoupling |
| 14 | mrt_rstang | R/W | 0 | Encoder emulator electric rotor angle evaluation reset control |

| | | | | |
|---|---|---|---|---|
| | | | | 1=reset-lock, 0=running |
| 13 | smo_rstang | R/W | 0 | Position Eval electric rotor angle reset control<br>1=reset-lock, 0=running |
| 12 | fbk_angdef | R/W | 0 | Default angle selector<br>0=default register<br>1=rot_angle feedback |
| 11 | pi_deafmd | R/W | 0 | PI current feedback control<br>1=ignore feedback, 0=use feedback |
| 10 | is3phases | R/W | 0 | 1=3-phase motor, 0=stepper motor |
| 9 | spare_9 | READ | 0 | Unused/reserved |
| 8 | modlevels | R/W | 0 | 0=2-levels, 1=3-levels |
| 7..4 | rot_select | R/W | 0 | Rotor selector angle:<br>0=manual : “rot_angdef” register,<br>1=MRT : scalar mode,<br>2=ENC : incremental encoder,<br>3=SMO : sliding mode observer,<br>4=HLS : hall sensors,<br>5=RSV_0 : resolver_0,<br>6=EXT_0 : external_0 sensor<br>7=RSV_1 : resolver_1,<br>8=EXT_1 : external_1 sensor |
| 3 | spare_3 | READ | 0 | Unused/reserved |
| 2 | modtype | R/W | 0 | Modulator selector request<br>0=PWM<br>1=RPFM |
| 1 | spare_1 | READ | 0 | Unused/reserved |
| 0 | coilenab | R/W | 0 | Coil enable<br>1=IP control, 0=drive LOW |

### *Motor_status - Motor status register*

This register report FOC status and let reset of latched events.
The read access freeze some status registers for atomic read.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 17..31 | | READ | 0 | Unused/reserved |
| 16 | hwkill | READ | 0 | Kill input from Gate unit |
| 15 | hwfail | READ | 0 | H/W failure from Gate unit |
| 14 | ovi_iphs_a | READ | 0 | Overcurrent motor Phase-A |
| 13 | ovi_iphs_b | READ | 0 | Overcurrent motor Phase-B |
| 12 | ovi_iphs_c | READ | 0 | Overcurrent motor Phase-C |
| 11 | ovi_ibus_x | READ | 0 | Overcurrent dc_link |
| 10 | syserr | READ | 0 | Global system error |
| 9 | spare_9 | READ | 0 | - |

| | | | | |
|---|---|---|---|---|
| 8 | spare_8 | READ | 0 | - |
| 7 | spare_7 | READ | 0 | - |
| 6..5 | rpfmzmd | READ | 0 | RPFM modulation zone<br>0=IDLE (not operational)<br>1=Sinusoidal/Extended sinusoidal(linear zone)<br>2=Hexagon zone (partial saturation)<br>3=Saturation zone (square wave or six step) |
| 4 | modtype0 | READ | 0 | Modulator type in use<br>0=PWM, 1=RPFM |
| 3 | spd_fwdir | READ | 0 | speed evaluation moving direction<br>1=FWD, 0=REV |
| 2 | spd_moving | READ | 0 | speed evaluation moving status<br>1=moving, 0=still |
| 1 | pwm_mdovf | READ | 0 | PWM modulation overflow |
| 0 | ismodover | READ | 0 | Is current overflow. The 1=overflow will stop the motor to a safe condition. The event reset requires switch off the motor writing “0” in coil enable control register bit |

### *Inverter values from A/D acquisition.*

This register returns the voltage and current of motor phases and dc-link. The values are normalized as SIGNED18

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 18..31 | | READ | 0 | Unused/reserved |
| 0..17 | val_iphs_a<br>val_iphs_b<br>val_iphs_c<br>val_ibus_x<br>val_vphs_a<br>val_vphs_b<br>val_vphs_c<br>val_vbus_x<br>val_vphs_n | READ | 0 | Motor Phase-A current<br>Motor Phase-B current<br>Motor Phase-C current<br>Dc_link current<br>Motor Phase-A voltage<br>Motor Phase-B voltage<br>Motor Phase-C voltage<br>Dc_link voltage<br>Motor neutral voltage |

The value is SIGNED18. For external representation in “Amperes” and “Volts”, multiply by proper gain floating point constant.

| BIT NUMBER/INDEX | INVERTER ANALOG CHANNEL |
|---|---|
| 0 | IPHS_A |
| 1 | IPHS_B |
| 2 | IPHS_C |
| 3 | IBUS_X |

| | |
|---|---|
| 4 | VPHS_A |
| 5 | VPHS_B |
| 6 | VPHS_C |
| 7 | VBUS_X |
| 8 | VPHS_N |

The configuration constant C_INV_IN_MAP is used to define the number of active channels. The configuration constant C_INV_IN_NOT is used to force two complement values (for negative inputs). Both constants are integer bit mask where each bit refer a specific analog channel.
The default value 131 (decimal) in C_INV_IN_MAP correspond to binary value '01000011b'. This means the IPHS_A, IPHS_B and VBUS_X are implemented.

### *Inverter offset value.*

This register let set/get the offset values for zero calibration for each analog channel. The values are normalized as SIGNED18

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 18..31 | | READ | 0 | Unused/reserved |
| 0..17 | ofs_iphs_a<br>ofs_iphs_b<br>ofs_iphs_c<br>ofs_ibus_x<br>ofs_vphs_a<br>ofs_vphs_b<br>ofs_vphs_c<br>ofs_vbus_x<br>ofs_vphs_n | R/W | 0 | Motor Phase-A current<br>Motor Phase-B current<br>Motor Phase-C current<br>Dc_link current<br>Motor Phase-A voltage<br>Motor Phase-B voltage<br>Motor Phase-C voltage<br>Dc_link voltage<br>Motor neutral voltage |

These registers let zero calibration of analog acquisition system.
Two methods of zero calibration are allowed if enabled in C_INV_OFSV_MODE.

| BIT | NAME | DEFAULT | DESCRIPTION |
|---|---|---|---|
| 1 | ZERO_SW | 0 | Individual offset register can be set |
| 0 | ZERO_AUTO | 1 | Automatic zero function on command |

### *Inverter multiplier value.*

This register let set/get the multiplier values for gain calibration for each analog channel. The values are normalized as SIGNED18

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|

| | | | | |
|---|---|---|---|---|
| 18..31 | | READ | 0 | Unused/reserved |
| 0..17 | mul_iphs_a<br>mul_iphs_b<br>mul_iphs_c<br>mul_ibus_x<br>mul_vphs_a<br>mul_vphs_b<br>mul_vphs_c<br>mul_vbus_x<br>mul_vphs_n | R/W | 0 | Motor Phase-A current<br>Motor Phase-B current<br>Motor Phase-C current<br>Dc_link current<br>Motor Phase-A voltage<br>Motor Phase-B voltage<br>Motor Phase-C voltage<br>Dc_link voltage<br>Motor neutral voltage |

These registers let gain calibration of analog acquisition system.
The value is multiplied by $2^{16}$ so the effective gain is: gain = register / $2^{16}$.
He gain range is -2..2. The default gain is 1 with register value of 65536.

### *xbus_fk1, xbus_fk2 – DC_LINK LPF1 parameter register*

These registers are used to setup the cutting frequency of LPF1 used to filter the dc_link both voltage and current.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 17..31 | | READ | 0 | Unused/reserved |
| 0..16 | xbus_fk1/xbus_fk2 | R/W | 0 | LPF1 multiplier constant in range 0..$2^{17}$-1 |

A second order filter is implemented using two LPF1 first order filters in cascade configuration.
The xbus_fk1 is used to configure the first filter, and the xbus_fk2 is used to configure the second filter.
The filter IP is enabled if C_INV_FILTER = 1.

### *xphs_fk1, xphs_fk2 – Motor phases LPF1 parameter register*

These registers are used to setup the cutting frequency of LPF1 used to filter the motor phases both voltage and current.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 17..31 | | READ | 0 | Unused/reserved |
| 0..16 | xphs_fk1/xphs_fk2 | R/W | 0 | LPF1 multiplier constant in range 0..$2^{17}$-1 |

A second order filter is implemented using two LPF1 first order filters in cascade configuration.
The xphs_fk1 is used to configure the first filter, and the xphs_fk2 is used to configure the second filter.
The filter IP is enabled if C_INV_FILTER = 1.

### *Ibus_limit – current limit for dc_link*

The register is used to set the dc_link current limit to protect the inverter and dc_link source from overcurrent.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 17..31 | | READ | 0 | Unassigned/reserved |
| 0..16 | ibus_limit | R/W | 0 | dc_link current limit |

### *Iphs_limit – current limit for motor phases*

The register is used to set the motor phases current limit to protect the inverter, and the motor from overcurrent.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 17..31 | | READ | 0 | Unassigned/reserved |
| 0..16 | iphs_limit | R/W | 0 | Motor phases current limit |

### *Ismodmax – Is current limit*

The register is used to set the Is current limit to protect the motor against overcurrent.
The FOC IP monitor the Is current and compare the value with the defined limit. In case of overflow the ismodover bit is set into motor status register: the motor will stop immediately and coil driver will be set to neutral position.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 17..31 | | READ | 0 | Unassigned/reserved |
| 0..16 | ismodmax | R/W | 0 | Is current limit |

### *Ismodval – Is current value*

The read only register report the real time Is current modulo.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 17..31 | | READ | 0 | Unassigned/reserved |
| 0..16 | ismodval | READ | 0 | Is current modulo |

### *Isangval – Is current angle*

The read only register report the real time Is current angle. The value resolution is $2^{32}$=360 degrees.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 0..31 | isangval | READ | 0 | Is current angle |

### *Vamodval – Va voltage value*

This read only UNSIGNED17 register reports the real time Va voltage modulo from acquisition

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 17..31 | | READ | 0 | Unassigned/reserved |
| 0..16 | vamodval | READ | 0 | Va voltage modulo |

### *Vaangval – Va voltage angle*

This read only register reports the real time Va voltage angle. The value resolution is $2^{32}$=360 degrees.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 0..31 | vaangval | READ | 0 | Va voltage angle |

### *Vsmodval – Vs voltage value*

This read only UNSIGNED17 register reports the Vs voltage modulo to modulator.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 17..31 | | READ | 0 | Unassigned/reserved |
| 0..16 | vsmodval | READ | 0 | Vs voltage modulo |

### *Vsangval – Vs voltage angle*

This read only register reports the Vs voltage angle to modulator. The value resolution is $2^{32}$=360 degrees.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|

| | | | | |
|---|---|---|---|---|
| 0..31 | vsangval | READ | 0 | Vs voltage angle |

### *Vs_rotor_x/y– Vs voltage D/Q rotor*

This read only SIGNED18 registers reports the Vs voltage vector relative to rotor or before inverse park transforming.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 18..31 | | READ | 0 | Unassigned/reserved |
| 0..17 | vs_rotor_x<br>vs_rotor_y | READ | 0 | Vs voltage vector X/Y frame |

### *rot_angdef – rotor initial angle*

This register is used to set the initial angle value for all position IP cores. The value is defined as electric rotor angle. The value is loaded at h/w and manual reset of IP. The register is UNSIGNED32 value with $2^{32}$=360 degrees. With number of pair poles = 1 the angle correspond to mechanical rotor angle.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 0..31 | rot_angdef | R/W | 0 | Initial angle for position IP cores |

### *rot_angle – rotor angle*

This read only register contains the real time angle generated by IP. The angle resolution is $2^{32}$=360 degrees. The value is loaded from selected IP cores according to rot_selector defined in motor control register. In case of IP core reset, rot_angle assumes the value according to rot_selector and fbk_angdef values.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 0..31 | rot_angle | READ | 0 | Angle value |

### *Spd_fktau1, spd_fktau2 – Speed evaluation LPF1 parameter register*

These registers are used to setup the cutting frequency of LPF1 used in speed evaluation IP core.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 17..31 | | READ | 0 | Unused/reserved |

| | | | | |
|---|---|---|---|---|
| 0..16 | spd_fktau1<br>spd_fktau2 | R/W | 0 | LPF1 multiplier constant in range $0...2^{17-1}$ |

A second order filter is implemented using two LPF1 first order filters in cascade configuration.
The spd_fktau1 is used to configure the first filter, and the spd_fktau2 is used to configure the second filter.

The LPF1 formula is the following: Y = Y + (X – Y) * K
Where:
Y = internal accumulator and result;
X = new current sample;
K = error weight in range 0...1.

The spd_fktau1/2 register is an UNSIGNED17 value in range 0…131071 for a corresponding:
K = [$0/2^{17}$… ($2^{17}$-1)/$2^{17}$].
The filter function is executed for every IP activation. Refer to LPF1 IP for details.

### *Spd_speed – Speed evaluation current speed register*

The speed is evaluated by integration of “rot_angle” signal that correspond of electric speed of rotor.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 0..31 | spd_speed | READ | 0 | SIGNED32 speed value |

The speed is defined as angle/time where:
Angle is the register content in range -$2^{31}$...$2^{31}$-1 that correspond to: A = value / $2^{32}$ * 2 π
The time is defined by IP core activation time.
Example: the value 1374390 with 3.2 µSec of IP core activation time correspond to about 100 Hz
NOTE: The speed is defined as electric speed.

### *smo_smontmr – position evaluation ON timer*

The value is an UNSIGNED15 FOC cycle timer to enable SMO BEMF usage for smooth start.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 16..31 | | READ | 0 | Unused/reserved |
| 0..15 | smo_smontmr | R/W | 0 | Vs multiplier |

### *smo_vs_mult – position evaluation Vs multiplier*

The value is an UNSIGNED32 multiplier used by the position evaluation IP core. Refer to the specific IP document for a detailed description.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 0..31 | smo_vs_mult | R/W | 0 | Vs multiplier |

### *smo_is_mult – position evaluation Is multiplier*

The value is an UNSIGNED32 multiplier used by the position evaluation IP core. Refer to the specific IP document for a detailed description.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 0..31 | smo_is_mult | R/W | 0 | Is multiplier |

### *smo_zs_max – position evaluation maximum error*

The value is an UNSIGNED17 that represents the absolute maximum error in the BEMF evaluation. Refer to the specific IP document for a detailed description.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 17..31 | | READ | 0 | Unassigned/reserved |
| 0..16 | smo_zs_max | R/W | 0 | Absolute max BEMF error |

### *smo_es1_kflt – position evaluation BEMF LPF1 #1*

The value is an UNSIGNED17 that sets up the first LPF1 used to evaluate the BEMF. Refer to the specific IP document for a detailed description.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 17..31 | | READ | 0 | Unassigned/reserved |
| 0..16 | smo_es1_kflt | R/W | 0 | LFP1 #1 filter coefficient for BEMF evaluation |

### *smo_es2_kflo – position evaluation BEMF LPF1 #2*

The value is an UNSIGNED17 that sets up the second LPF1 used to evaluate the BEMF.
The value is used to define the minimum Fcut for speed=0.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 17..31 | | READ | 0 | Unassigned/reserved |
| 0..16 | smo_es2_kflo | R/W | 0 | LFP1 #2 filter coefficient for BEMF evaluation |

### *smo_es2_kfmx – position evaluation BEMF LPF1 #2*

The value is an UNSIGNED17 that sets up the second LPF1 used to evaluate the BEMF.
The value is used to extend the Fcut according speed.
Refer to the specific IP document for a detailed description.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|

| | | | | |
|---|---|---|---|---|
| 17..31 | | READ | 0 | Unassigned/reserved |
| 0..16 | smo_es2_kfmx | R/W | 0 | LFP1 #2 filter coefficient for BEMF evaluation |

### *smo_es2_kflt – position evaluation BEMF LPF1 #2*

The value is an UNSIGNED17 that sets up the second LPF1 used to evaluate the BEMF. Refer to the specific IP document for a detailed description.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 17..31 | | READ | 0 | Unassigned/reserved |
| 0..16 | smo_es2_kflt | READ | 0 | LFP1 #2 filter coefficient for BEMF evaluation |

### *smo_angofs – position evaluation angle offset*

The value is an SIGNED16 angle value in range [–π..π] to compensate the bemf output angle on SMO.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 16..31 | | READ | 0 | Unassigned/reserved |
| 0..15 | smo_angofs | R/W | 0 | Angle offset |

### *smo_bemf_p – position evaluation BEMF angle*

Read only UNSIGNED32 register. It represents the BEMF vector angle. The value $2^{32}$ corresponds to 360 degrees.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 0..31 | smo_bemf_p | READ | 0 | BEMF vector phase |

### *smo_bemf_m – position evaluation BEMF modulo*

Read only UNSIGNED17 register. It represents the BEMF vector modulo. Refer to the specific IP document for a detailed description.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 17..31 | | READ | 0 | Unassigned/reserved |
| 0..16 | smo_bemf_m | READ | 0 | BEMF vector modulo |

### *smo_angle– position evaluation rotor electric angle*

Read only UNSIGNED32 register. It represents the rotor electric angle (flux vector). The value $2^{32}$ corresponds to 360 degrees.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 0..31 | smo_angle | READ | 0 | Rotor electric angle ($2^{32}$) |

### *Pos_position– low resolution position rotor electric angle*

Read only SIGNED32 register. The range is $-2^{31}...2^{31}-1$. The value $2^{16}$ corresponds to 360 degrees. The range rounds is $-2^{15}...2^{15}-1$. The value represents the current rotor electric angle according the selected rotor control come.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 0..31 | pos_position | READ | 0 | Rotor electric angle ($2^{16}$) |

### *slp_spdset– speed setpoint for speed loop control*

Read/Write SIGNED32 register. Refer to ***spd_speed*** register for unit definition.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 0..31 | slp_spedset | R/W | 0 | Speed setpoint |

The speed setpoint is used in speed loop when ***spd2trq***=1 to feed the current loop. The speed loop is also used to feed the position loop when ***spd4pos***=1.

### *slp_kmpro- speed proportional error gain for speed loop control*

Read/Write IEEE-754 32-bits FLOAT register. The proportional gain is ***slp_kmpro / 2*** $^{P\_SLP\_PRO\_DLN2.}$

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 0..31 | slp_kmpro | R/W | 0 | Proportional error gain |

### *slp_kmint - speed integrative error gain for speed loop control*

Read/Write IEEE-754 32-bits FLOAT register. The integrative gain is ***slp_kmint / 2*** $^{C\_SLP\_INT\_DLN2.}$

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 0..31 | slp_kmint | R/W | 0 | integrative error gain |

### *slp_outlim - speed output current limit*

Read/Write UNSIGNED17 register. The values is the current limit for speed control loop.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 17..31 | | READ | 0 | Unassigned/reserved |
| 0..16 | slp_outlim | R/W | 0 | Current limit |

### *slp_kmultx, slp_kmulty- speed output current multiplier*

Read/Write SIGNED18 registers. The values is a K / $2^{16}$ multiplier for X/Y current output setpoint.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 18..31 | | READ | 0 | Unassigned/reserved |
| 0..17 | slp_kmultx<br>slp_kmulty | R/W | 0 | Current X/Y control. |

The effective gain are usually evaluated by:

slp_kmultx = cos(angle) * $2^{16}$

slp_kmulty = sin(angle) * $2^{16}$

The angle value for maximum torque is π/2.

### *Paipol – motor pair poles*

This register setup the number of motor pair poles.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 8..31 | | READ | 0 | Unused/reserved |
| 0..7 | paipol | R/W | 0 | Motor pair poles in range 1...255 |

The number of pair poles (“pair-poles” = “poles / 2”) is used on several sub functional IP. Set the proper value in range “1..n” according motor characteristics.

IMPORTANT: do not set value “0” (zero).

### *ext_0_angofs/ext_1_angofs – External sensors angle offset*

This SIGNED16 read/write register defines the angle offset for tuning of external sensors IP. The angle resolution is $2^{16}$=360 degrees.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 16..31 | | READ | 0 | Unused/reserved |
| 0..15 | ext_0_angofs<br>ext_1_angofs | R/W | 0 | External angle offset |

### *ext_0_angle/ext_1_angle – External sensors angle*

This SIGNED32 read only register contains the real time angle generated by external sensors IP. The angle resolution is $2^{32}$=360 degrees.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 0..31 | ext_0_angle<br>ext_1_angle | READ | 0 | External angle |

### *rsv_m2rppk – Motor to Resolver Pair Poles Ratio*

This register defines the ratio between motor pair poles and resolver pair poles.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 8..21 | | READ | 0 | Unassigned/reserved |
| 0..7 | rsv_m2rppk | R/W | 1 | Pair Poles Ratio = Motor PP / Resolver PP |

Example: if motor pair poles is 6, and resolver pair poles is 2, the register shall be set to 3.

### *rsv_0_angofs/rsv_1_angofs – Resolver sensors angle offset*

This SIGNED16 register set the angle offset to correct alignment between resolver sensor and motor flux. The angle resolution is $2^{16}$=360 degrees.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 16..31 | | READ | 0 | Unassigned/reserved |
| 0..15 | rsv_0_angofs<br>rsv_1_angofs | R/W | 0 | Angle value |

### *rsv_0_angle/rsv_1_angle – Resolver sensors angle*

This SIGNED32 read only register contains the real time angle generated by resolver sensors IP. The angle resolution is $2^{32}$=360 degrees.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 0..31 | rsv_0_angle<br>rsv_1_angle | READ | 0 | Angle value |

### *hls_ctolim – Hall sensors coasting mode counter timeout limit*

This r/w register contains the clock timeout for coasting mode used to control the angle interpolator. The value is system clock units.
Example: with system clock @ 100 MHz, the maximum value $2^{20}$-1 is about 10.5 msec.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 20..31 | | READ | 0 | Unassigned/reserved |
| 0..19 | hls_ctolim | R/W | 0 | Timeout value |

### *hls_angle – Hall sensors angle*

This read only register contains the real time angle generated by hall sensors IP. The angle resolution is $2^{32}$=360 degrees.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 0..31 | hls_angle | READ | 0 | Angle value |

### *enc_index – Encoder index register*

It counts the index rising edges pulses or motor revolutions. This register is unipolar (unsigned) and is incremented regardless of rotation direction (forward or reverse).

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 0..31 | enc_index | READ | 0 | Rounds counter |

### *enc_phase – Encoder phase register*

It counts the sensor phases. This register is bipolar (signed). The counter is incremented in forward direction and decremented in reverse direction. The counter counts four phases for each encoder pulse. If the register size is less than 32 bits then the most significant bits are at fixed value of “0” (no signed extend function is applied).

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 0..31 | enc_phase | READ | 0 | Phases counter |

### *Enc_phcpt – Encoder phase hold*

The register hold Enc_phase on rising edge of encode index. The first sampled value corresponds to index position at power on, all others values are: initial_offset + revolutions * phases_per_revolutions.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 0..31 | enc_phcpt | READ | 0 | Phases counter. |

### *enc_cyprnd - Encoder cycles per round*

This UNSIGNED12 register is used to set the pulses (or cycles) per round for the given encoder. An encoder pulse correspond to 4 phases. The value limit is 4095 pulses or 16380 phases.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 12..31 | | READ | 0 | Unassigned/reserved |
| 0..11 | enc_cpyrnd | R/W | 0 | Angle increment per phase. |

### *enc_angphs - Encoder angle increment*

This register is used to set the angle increment for each phase transition. For a given encoder of 256 pulses per round we have 256 x 4 = 1024 phases per mechanical round. If the motor have 2 pair poles (2=electric rounds for each mechanical round), the electric rotor phases per rounds are 1024 / 2 = 512. The angle resolution is $2^{32}$=360 degrees, so the register shall be set to $2^{32}$ / 512 = 8388608.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 0..31 | enc_angphs | R/W | 0 | Angle increment per phase. |

### *Enc_angle - Encoder angle*

This read only register contains the real time angle generated by encoder IP. The angle resolution is $2^{32}$=360 degrees. The H/W reset or IP user reset can be used to force load of enc_angdef value.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 0..31 | enc_angle | READ | Enc_angdef | Angle value |

### *Mrt_speed – scalar mode rotor speed setpoint*

This register is used to set the angle increment for each internal time base event.

Example:
IP cycle time = 3.2 µS,
Motor Pair Poles = 2,
Desired speed = 1000 RPM = 16.667 RPS (round per seconds)Rotor Electric speed = 16.667 RPS * 2 pair poles = 33.333 Hz (flux speed is electric rotor speed)
Enc_mrtspeed = 33.333 Hz / $3.2^{-6}$ = 10,416,667

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 0..31 | mrt_speed | R/W | 0 | Speed set point |

The ***mrt_speed*** is used directly when C_MANROT = 1,
If C_MANROT = 2 the maximum acceleration is applicable and a double LPF1 filter is used to smooth the effective speed.

### *Mrt_accmax – scalar mode rotor acceleration limit*

This register is used to control the speed ramp from current speed to target speed.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 0..31 | mrt_accmax | R/W | 0 | Acceleration limit |

The value is application with C_MANROT = 2.

### *Mrt_fktau1/mrt_fktau2 – scalar mode rotor double LPF1filters*

The value is an UNSIGNED17 that sets up the first LPF1 used to smooth the speed setpoint in scalar mode rotor control. Refer to the specific IP document for a detailed description.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 17..31 | | READ | 0 | Unassigned/reserved |
| 0..16 | mrt_fktau1<br>mrt_fktau2 | R/W | 0 | LFP1 filter coefficient for speed value |

### *Mrt_spdout – scalar mode rotor speed value*

This read only register report the speed value after double LPF filter.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 0..31 | mrt_spdout | READ | 0 | Speed value |

### *Mrt_angle – scalar mode rotor angle*

This read only register contains the real time angle generated by encoder emulator IP. The angle resolution is $2^{32}$=360 degrees. The H/W reset or IP user reset can be used to force load of ***enc_angle*** value.

The encoder emulator IP input default value is internally connected to encoder IP angle output for a cascade operation. The encoder emulator is in reset=inhibit state, the encoder angle output pass through the encoder emulator.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 0..31 | mrt_angle | READ | Enc_angle | Angle value |

### *Pi_setvalx, pi_setvaly – PI setpoint value*

These SIGNED18 registers are used to setup the desired X/Y currents. The value is compared with current feedback to evaluate the Vs vector. The bit resolution depends on h/w implementation.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 18..31 | | READ | 0 | Unassigned/reserved |
| 0..17 | pi_setvalx<br>pi_setvaly | R/W | 0 | Current setpoint |

### *Pi_kmprox/pi_kmproy – PI proportional error multiplier*

These UNSIGNED17 registers are proportional error multiplier for error gain.
The effective value is ***pi_kmpro / $2^{C\_PI\_ERPRO\_DLN2}$***.
The value is used for both **kmpro_x** and **kmpro_y** internal signals.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 17..31 | | READ | 0 | Unassigned/reserved |
| 0..16 | pi_kmprox<br>pi_kmproy | R/W | 0 | Proportional error multiplier |

### *Pi_kmintx/pi_kmint_y – PI integrativ error multiplier*

These UNSIGNED17 register are integrative error multiplier for error gain.
The effective value is ***pi_kmint / $2^{C\_PI\_ERINT\_DLN2}$***.
The value is used for both **kmint_x** and **kmint_y** internal signals.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 17..31 | | READ | 0 | Unassigned/reserved |
| 0..16 | pi_kmintx<br>pi_kminty | R/W | 0 | Integrator error multiplier |

### *dck_ldkmul,dck_lqkmul – PI decoupling Id and Iq multiplier*

These UNSIGNED17 registers are the mantissa of Ld and Lq multipliers used to evaluate both D and Q frames feedforward for decoupling.
The effective value are:
Kd = ***dck_ldkmul / $2^{C\_DCK\_KL\_DLN2}$***.
Kq = ***dck_lqkmul / $2^{C\_DCK\_KL\_DLN2}$***.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 17..31 | | READ | 0 | Unassigned/reserved |

| | | | | |
|---|---|---|---|---|
| 0..16 | dck_ldkmul<br>dck_lqkmul | R/W | 0 | Electromagnetic flux Id/Iq multiplier |

The Ld * Id is used to evaluate the Q-frame contribution.
The Lq * Iq is used to evaluate the D-frame contribution.

### *dck_kekmul – PI decoupling speed multiplier*

The UNSIGNED17 register is the mantissa of Ke multiplier used to evaluate the Q-frame feedforward for decoupling.
The effective value is:
Ke = $dck\_kekmul / 2^{C\_DCK\_KE\_DLN2}$.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 17..31 | | READ | 0 | Unassigned/reserved |
| 0..16 | dck_kekmul | R/W | 0 | Magnetic flux speed multiplier |

The Ke * speed is used to evaluate the Q-frame contribution.

### *bemfangofs – BEMF module compensation LUT angle offset*

This SIGNED16 angle is used as displacement in LUT index evaluation.
The angle resolution is $2^{16}$=360 degrees.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 16..31 | | READ | 0 | Unassigned/reserved |
| 0..15 | bemfangofs | R/W | 0 | Angle value |

### *Mod2angskw – second modulator angle skew*

In double independent 3-phases windings of BLDC/PMSM motor, this register SIGNED16 shall be programmed with angle offset of 2nd winding.
The angle resolution is $2^{16}$=360 degrees.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 16..31 | | READ | 0 | Unassigned/reserved |
| 0..15 | mod2angskew | R/W | 0 | Angle value |

### *Pwm_angofs – PWM modulator angle offset*

This SIGNED16 angle is added to PWM modulator vector angle
The angle resolution is $2^{16}$=360 degrees.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 16..31 | | READ | 0 | Unassigned/reserved |
| 0..15 | pwm_angofs | R/W | 0 | Angle value |

### *Pwm_kmod – PWM modulation multiplier*

The register is a multiplier applied to Vs voltage modulator for scale correction (PWM frequency and DC_link compensation).

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 0..31 | pwm_kmod | R/W | 0 | Pwm argument multiplier |

### *Pwm_presc – PWM prescaler*

Two PWM prescaler UNSIGNED17 bit register is used to set the PWM frequency. The PWM wave generator runs at main clock frequency.
The value width is 17 bits for a range in 0…$2^{17}$-1.
The PWM frequency is obtained by this formula: pwm_presc = sysclock / pwm_freq / 2 – 1
Example for system clock of 50 MHz and desired pwm_freq of 20 KHz;
Pwm_presc = 50 MHz / 20 KHz / 2 – 1 = 1249.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 17..31 | | READ | 0 | Unassigned/reserved |
| 0..16 | pwm_presc | R/W | 0 | PWM prescaler value: range is 0…$2^{17}$-1 |

### *Pwm_mdmax – PWM modulation limit*

This register is used to limit the modulation duty cycle to desired value. The bit resolution is the same of pwm_presc.
The PWM modulator compares the PWM modulation value with the pwm_mdmax limit. In case of overflow the pwm_mdovf bit of motor status register is set.
It is highly recommended to set this register at a proper value in order to protect the gate power stage against too long high side driver time ON.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|

| | | | | |
|---|---|---|---|---|
| 17..31 | | READ | 0 | Unassigned/reserved |
| 0..16 | pwm_mdmax | R/W | 0 | PWM modulation limit value: range is 0.. $2^{17}$-1 |

### *Pwm_mdval – PWM modulation value*

This read only register report the real time modulation value. The value is in range 0...pwm_mdmax with pwm_presc = 100%.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 17..31 | | READ | 0 | Unassigned/reserved |
| 0..16 | pwm_mdval | READ | 0 | PWM modulation limit value: range is 0.. .. $2^{17}$-1 |

### *Pwm_minpw – PWM minimum pulse width*

This UNSIGNED17 register is used discard narrow and potentially dangerous pulse width.
The time unit is FPGA system clock.
The value of register is evaluated by: *minpw_time * FPGA_clock – 2.*
The number "0" disable the feature.
Example: @ 100 MHz of FPGA clock, the value 98 will discard all pulses narrow than 1 microsecond.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 17..31 | | READ | 0 | Unassigned/reserved |
| 0..16 | pwm_minpw | R/W | 0 | PWM minimum pulse width<br>0=disable<br>1..n=(2+value) * FPGA_tclock |

### *Pwm_ctrl – PWM control register*

This register selects various option of PWM modulaor.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 3..31 | | READ | 0 | Unassigned/reserved |
| 2 | cmmofs | R/W | 0 | 3-Phase common mode offset |
| 0..1 | table | R/W | 0 | PWM modulation table selector |

### *Rpfm_angofs – RPFM modulator angle offset*

This SIGNED16 angle is added to RPFM modulator vector angle
The angle resolution is $2^{16}$=360 degrees.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 16..31 | | READ | 0 | Unassigned/reserved |
| 0..15 | rpfm_angofs | R/W | 0 | Angle value |

### *Rpfm_ctrl – PFM control register*

This register controls the PFM parameters setup.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 14..31 | | READ | 0 | Unassigned/reserved |
| 13 | rpfm_v7h | R/W | 0 | Null vector 7 hold mode:<br>0=vector 7 fall to vector 0<br>1=vector 7 hold for reduced switches |
| 12 | rpfm_v07n | R/W | 0 | Null vector 0/7 transition mode:<br>0=vector 0<br>1=vector 0 or 7 is selected for reduced switches |
| 8..11 | rpfm_mdpr | R/W | 0 | RPFM modulator prescaler 0...15 |
| 0..7 | rpfm_mdkd | R/W | 0 | RPMF modulator clock divider 0...255 |

| Vector 0/7 nice (v07n) | Vector 7 hold (v7h) | Vector 0/7 modes |
|---|---|---|
| 0 | x | Vector 7 disable |
| 1 | 0 | Vector 7 only one shot |
| 1 | 1 | Symmetric mode |

The modulator pulse width is defined by system clock divider.
clock_divider = (rpfm3p_mdkd + 1) * (rpfm3p_mdpr + 1)

Example:
IP core activation time = 3.2 µS,
System clock = 62.5 MHz,
Desired modulation pulse with = 6.4 µS,
Rpfm_mdkd = 24
rprm_mdpr = 15

Refer to RPFM specific IP datasheet for details and limitations.

### *deadtval – Gate unit dead time value*

This register can be used to set value for gate unit dead time.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 8..31 | | READ | 0 | Unassigned/reserved |
| 0..7 | deadtval | R/W | 255 | Value for gate unit dead time |

The value is not used in motor control IP. The signal is delivered to gate unit IP as-is.

### *gwswcnt – Gate unit switches counter*

This register can be used to report the gate unit switches

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 0..31 | gwswcnt | READ | - | Value from gate unit |

The value is not used in motor control IP. The signal is reported from gate unit.

### *LUT ACCESS from HOST side*

#### *Lut_addr – LUT address register*

This UNSIGNED32 register is used to address the LUT dual ported memory.
The lower 16 bits (0..15) select the LUT element.
The upper 3 bits (16..18) selects up to 8 LUTs.
The remain 13 bits are unused.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 19..31 | | R/W | 0 | Unassigned/reserved |
| 16..18 | lut_addr_sel | R/W | 0 | LUT selector: 0...7 |
| 0..15 | lut_addr | R/W | 0 | LUT address register: 0...65535 |

| Lut_addr_sel | LUT | description |
|---|---|---|
| 0 | NONE | Not implemented, void address |
| 1 | Hall Sensor IP | 16 entries to configure angles |
| 2 | PWM waveforms | PWM modulator 4 waveforms of 256 entries each |
| 3 | BEMF linearization | 1024 entries |
| 4 | RSV_0 linearization | 1K,2K, 4K entries for resolver_0 linearization |
| 5 | RSV_1 linearization | 1K,2K, 4K entries for resolver_1 linearization |
| 6…7 | n/a | Unassigned/reserved |

NOTE: the value “0” shall be stored to “lut_addr” register when not used to reduce energy consumption.

#### *hls_lut[0…15] – hall sensor LUT angles*

This set of SIGNED18, 16 registers is used by the hall sensor IP to evaluate the motor electric angle.
Look to specific IP description.
The angle resolution is $2^{18}$=360 degrees.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 18..31 | | READ | 0 | Unassigned/reserved |
| 0..17 | hls_dprangle[0…15] | R/W | 0 | SIGNED18 angle |

#### *pwm_waveform[0…1023] – PWM waveform LUT*

A set of four PWM waveform tables are defined using a single Xilinx BRAM 1024x18 primitive.
The DPR can be accessed in R/W modes from host side and READ only from PWM IP side.
The four tables are addressed by vector index range:

The lut_addr[9:8] address four PWM waveforms
The lut_addr[7:0] address the specific waveform entry.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 18..31 | | READ | 0 | Unassigned/reserved |
| 0..17 | Pwm_waveform[0…255] | R/W | 0 | SIGNED18 value |

### *Bemf_correction_lut[0…1023] – BEMF LUT module*

The LUT is used to correct the motor BEMF non linearity. Each LUT entry correspond to about 2.84 degrees.
The neutral value is 65536 that correspond do gain = 1.

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 18..31 | | READ | 0 | Unassigned/reserved |
| 0..17 | Bemf_correction_lut[0…1024] | R/W | 0 | SIGNED18 value |

### *rsv_0_lut/rsv_1_lut – correct resolve angle*

The size of entry depends on resolver IP configuration.
Implemented lengths are: 0,1K, 2K, 4K

| BIT | NAME | ACCESS | RESET VALUE | DESCRIPTION |
|---|---|---|---|---|
| 18..31 | | READ | 0 | Unassigned/reserved |
| 0..17 | Rsv_angle_lut[0…n] | R/W | 0 | SIGNED18 value |

# Execution time

| Start event | End of execution | clock cycles[1] | Time @ 50 MHz | Time @ 100 MHz |
|---|---|---|---|---|
| Curr_sync | Pwm(2-phases) | 188 | 3.76 µS | 1.88 µS |
| Curr_sync | Pwm(3-phases) | 201 | 4.04 µS | 2.01 µS |
| Curr_sync | RPFM(3-phases) | 172 | 3.44 µS | 1.72 µS |

[1] Unless otherwise noted.

## Reference Documents

n/a

## Support

QDESYS provides technical support for this LogiCORE product when used as described in the product documentation.
QDESYS cannot guarantee timing, functionality, or support of product if implemented in devices that are not defined in the documentation, if customized beyond that allowed in the product documentation, or if changes are made to any section of the design labeled DO NOT MODIFY.

## Ordering Information

For information on pricing and availability of QDESYS modules and software, please contact info@qdesys.com

## Revision History

| Date | Version | Description |
|---|---|---|
| 16/09/2011 | 1.0 | QDeSys first release. |
| 19/11/2011 | 1.1 | Added some registers. Modified Devices Utilization table. Removed pwm_dmod register. |
| 23/12/2011 | 1.2 | Correct execution time table. Update for new PI_control IP |
| 23/03/2012 | 1.3 | Updated registers |
| 12/05/2012 | 1.4 | Added Kintex 7 and Zynq support |
| 11/07/2013 | 1.5 | Position and speed loop control. BEMF feed forward compensation. Register map review. |
| 13/05/2014 | 1.6 | Hall sensor IP, double 3-phase modulator, enhanced feature for MRT IP. |
| 14/02/2015 | 1.7 | RPFM 3-levels modulator, Resolver sensor IP, extra trigger in current acquisition, direct access to dc_link and currents. |
| 19/05/2016 | 1.8 | Inclusion of analog preprocessing IP for offset, filtering and zero offset. Extend interface to motor board specific IP for diagnostic and probes. |
| 6-Jan-17 | 1.9 | Update speed loop, update current loop, remove bemf compensation |
| March 21, 2017 | 1.10 | Remove acquisition feature. Optimized PI-control |
| August 6, 2017 | 1.11 | Added PI current control probes, resolver ip update |
| 20-Dec-17 | 1.12 | Motor status register modification and PWM control register added. |
| June 1, 2018 | 1.13 | Update interface for MRT acceleration scaling |
| June 27, 2019 | 1.14 | Update control register |

| | | |
|---|---|---|
| September 6, 2019 | 1.15 | Resolver pair poles ratio added.<br>Correct some register definitions |
| April 23, 2021 | 1.16 | Update IP interface and register description |
| April 13, 2022 | 1.17 | Added several features and changed register map. |

## Disclaimer

In disclosing the information contained in this document QDeSys assumes no obligation to correct any errors herein contained, or to advise you of any corrections or updates. QDeSys expressly disclaims any liability in connection with technical support or assistance that may be provided to you in connection with the information.

THE DOCUMENTATION IS DISCLOSED TO YOU "AS-IS" WITH NO WARRANTY OF ANY KIND. QDESYS MAKES NO OTHER WARRANTIES, WHETHER EXPRESS, IMPLIED, OR STATUTORY, REGARDING THE DOCUMENTATION, INCLUDING ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT OF THIRD-PARTY RIGHTS. IN NO EVENT WILL QDESYS BE LIABLE FOR ANY CONSEQUENTIAL, INDIRECT, EXEMPLARY, SPECIAL, OR INCIDENTAL DAMAGES, INCLUDING ANY LOSS OF DATA OR LOST PROFITS, ARISING FROM YOUR USE OF THE DOCUMENTATION