BLDCPWM_DRV	*Three-phase BLDC PWM Driver*

Description

This module generates the 6 switching states of a 3-ph power inverter used to drive a 3-ph BLDC motor. These switching states are determined by the input variable *CmtnPointer*. The module also controls the PWM duty cycle by calculating appropriate values for the compare registers. The duty cycle values for the PWM outputs are determined by the input *DutyFunc*.

Availability

This 16-bit module is available in one interface format:

1) The C interface version

Module Properties

Type: Target Dependent, Application Independent

Target Devices: 28x Fixed Point or Piccolo

C Version File Names: f2803xbldcpwm.h (for x2803x)

IQmath library files for C: N/A

C Interface

Object Definition

The structure of PWMGEN object is defined by following structure definition

```
typedef struct {  Uint16 CmtnPointer;  // Input: Commutation (or switching) state pointer input (Q0)
                  int16 MfuncPeriod;    // Input: Duty ratio of the PWM outputs (Q15)
                  Uint16 PeriodMax;    // Parameter: Maximum period  (Q0)
                  int16 DutyFunc;       // Input: PWM period modulation input (Q15)
                  Uint16 PwmActive;   // Parameter: 0 = active low, 1 = active high (0 or 1)
                } PWMGEN;

typedef PWMGEN *PWMGEN_handle;
```

Item	Name	Description	Format	Range(Hex)
Inputs	CmtnPointer	Commutation (or switching) state pointer input	Q0	0 - 5
	MfuncPeriod	Duty ratio of the PWM outputs	Q15	8000-7FFF
	DutyFunc	PWM period modulation input	Q15	8000-7FFF
Outputs	PWMx (x=1,2,3,4,5,6)	Output signals from the 6 PWM pins	N/A	0-3.3 V
PWMGEN parameter	PeriodMax	PWM Period in CPU clock cycles	Q0	8000-7FFF
	PwmActive	0 = PWM active low 1 = PWM active high	Q0	0 or 1

Special Constants and Data types

PWMGEN
The module definition is created as a data type. This makes it convenient to instance an interface to the PWMGEN driver. To create multiple instances of the module simply declare variables of type PWMGEN.

PWMGEN_handle
User defined Data type of pointer to PWMGEN module

PWMGEN_DEFAULTS
Structure symbolic constant to initialize PWMGEN module. This provides the initial values to the terminal variables as well as method pointers.

Methods

BLDCPWM_INIT_MACRO(PWMGEN *);
BLDCPWM_MACRO(PWMGEN *);

This default definition of the object implements two methods – the initialization and the runtime compute macro for PWMGEN generation. This is implemented by means of a macro pointer, and the initializer sets this to BLDCPWM_INIT_MACRO and BLDCPWM_MACRO macros for x280x. The argument to this macro is the address of the PWMGEN object.

Module Usage

Instantiation
The following example instances one PWMGEN object
PWMGEN pwm1;

Initialization
To Instance pre-initialized objects
PWMGEN pwm1 = PWMGEN_DEFAULTS;

Invoking the computation macro
BLDCPWM_INIT_MACRO (pwm1);
BLDCPWM_MACRO (pwm1);

Example

The following pseudo code provides the information about the module usage.

```
main()
{

        pwm1.PeriodMax = 7500;          // PWM frequency = 20 kHz, clock = 150 MHz
        BLDCPWM_INIT_MACRO(pwm1);                   // Call init macro for pwm1

}

void interrupt periodic_interrupt_isr()
{

        pwm1.CmtnPointer = (int)(CmtnPointer1);          // CmtnPointer1 is in Q0
        pwm1.DutyFunc = (int)_IQtoIQ15(DutyFunc1);       // DutyFunc1 is in GLOBAL_Q
        BLDCPWM_MACRO(pwm1);                             // Call update macro for pwm1


}
```

Technical Background

Figure 1 shows the 3-phase power inverter topology used to drive a 3-phase BLDC motor. In this arrangement, the motor and inverter operation is characterized by a *two phases ON* operation. This means that two of the three phases are always energized, while the third phase is turned off. This is achieved by controlling the inverter switches in a periodic 6 switching or commutation states. The bold arrows on the wires in Figure 1 indicate the current flowing through two motor stator phases during one of these commutation states. The direction of current flowing into the motor terminal is considered as positive, while the current flowing out of the motor terminal is considered as negative. Therefore, in Figure 1, Ia is positive, Ib is negative and Ic is 0.

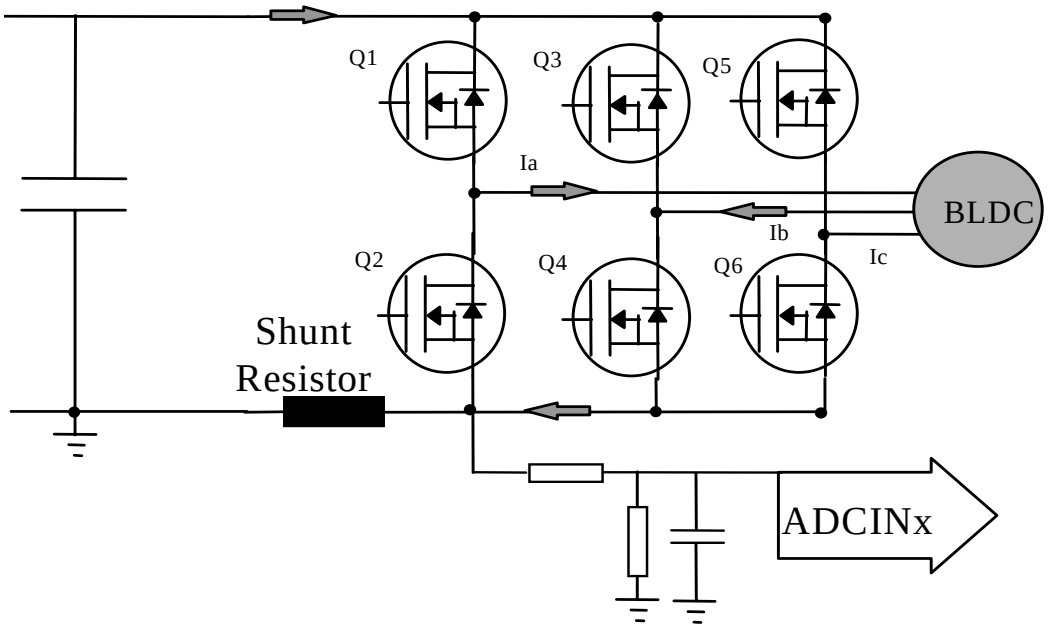

Figure 1: Three Phase Power Inverter for a BLDC Motor Drive

In this control scheme, torque production follows the principle that current should flow in only two of the three phases at a time and that there should be no torque production in the region of Back EMF zero crossings. Figure 2 depicts the phase current and Back EMF waveforms for a BLDC motor during the *two phases ON* operation. All the 6 switching states of the inverter in Figure 1 are indicated in Figure 2 by S1 through S6. As evident from Figure 2, during each state only 2 of the 6 switches are active, while the remaining 4 switches are turned OFF. Again, between the 2 active switches in each state, the odd numbered switch (Q1or Q3 or Q5) are controlled with PWM signal while the even numbered switch (Q2 or Q4 or Q6) is turned fully ON. This results in motor current flowing through only two of the three phases at a time. For example in state S1, Ia is positive, Ib is negative and Ic is 0. This is achieved by driving Q1 with PWM signals and turning Q4 fully ON. This state occurs when the value in the commutation state pointer variable, *CmtnPointer*, is 0. Table 1 summarizes the state of the inverter switches

and the corresponding values of the related peripheral register, the commutation pointer and the motor phase currents.

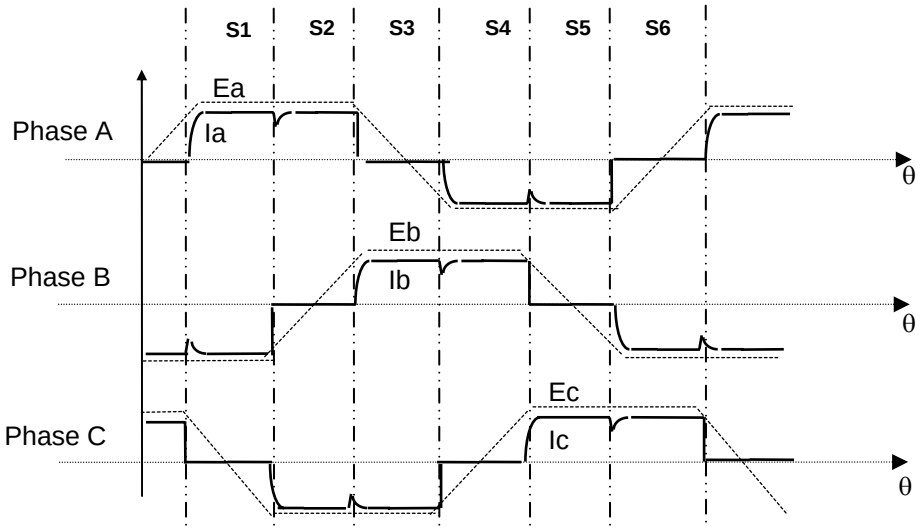

Figure 2: Phase Current and Back EMF Waveforms in 3-ph BLDC Motor control

State	CmtnPointer	ACTR	Q1	Q2	Q3	Q4	Q5	Q6	Ia	Ib	Ic
S1	0	00C2	**PWM**	OFF	OFF	**ON**	OFF	OFF	+ve	-ve	0
S2	1	0C02	**PWM**	OFF	OFF	OFF	OFF	**ON**	+ve	0	-ve
S3	2	0C20	OFF	OFF	**PWM**	OFF	OFF	**ON**	0	+ve	-ve
S4	3	002C	OFF	**ON**	**PWM**	OFF	OFF	OFF	-ve	+ve	0
S5	4	020C	OFF	**ON**	OFF	OFF	**PWM**	OFF	-ve	0	+ve
S6	5	02C0	OFF	OFF	OFF	**ON**	**PWM**	OFF	0	-ve	+ve

Table 1: Commutation States in 3-ph BLDC Motor control