Implementing on the MC56F8257 A Single Phase Two-Channel Interleaved Critical Conduction Mode

by: Petr FrGaL
System Application Engineer
IMM Freescale Roznov

1 Introduction
This application note describes in detail the digital implementation of an Interleaved Critical Conduction Mode (CRM) on Freescale’s MC56F8257 Digital Signal Controller (DSC). The application demonstrates the advantage of DSC peripherals dedicated for power conversion applications.

Interleaved CRM is used in power factor correction (PFC) circuits and also as a boost DC/DC converter for low and middle power step-up conversion. Interleaved topology delivers benefits of increasing power density, reducing ripple current at input which reduces the input filter, boost inductors size, reducing current ripple in the output capacitors, contemporary thermal management, and efficiency compared to the single boost converter.

This application note aims to describe in detail Interleaved CRM implementation, understand the problematics of such application and understand the use of DSC peripherals in this application.
This application note will help application engineers make easier implementing this application in power factor correction circuitries where input power is AC or in applications where DC voltage needs to be boosted to a higher level, such as in a solar inverter or battery powered applications.

This application note includes interleaved control scheme, system design concept, peripheral setup and usage, and software design.

The MC56F8257 is member of the DSC family that uses the 56800E core. Main advantages and features of this controller are described in following section.

2 MC56F8257 DSC Advantages and Features

The MC56F82xx digital signal controller family combines on a single chip, the processing power of a DSP and the functionality of a microcontroller with a flexible set of peripherals to create a cost-effective solution. Because of its low cost, configuration flexibility, and compact program code, it is well-suited for many applications. The MC56F8257 includes many peripherals that are especially useful for cost-sensitive applications, including:

- Switched-mode power supplies and power management
- Solar Inverters
- Battery chargers and management
- Power metering
- Lighting ballast
- Motor control
- Industrial control
- Home appliances
- Smart sensors

The MC56F8257 provides the following peripherals:

- One Enhanced Flex Pulse Width Modulator (eFlexPWM) module
- Two independent 12-bit analog-to-digital converters (ADCs)
- Inter-Crossbar Switch (XBAR)
- Three analog comparators (CMPs)
- One 12-bit digital-to-analog converter (12-bit DAC)
- Two four-channel 16-bit multi-purpose timer (TMR) modules
- Two queued serial communication interface (QSPI) modules with LIN slave function
- One queued serial peripheral interface (QSPI) module
- Two inter-integrated circuit (I2C) ports
- One Freescale Scalable Controller Area Network (MSCAN) module
- Computer operationg properly (COP) watchdog timer capable of selecting different clock sources
- Power supervisor (PS)
- Phase lock loop (PLL) providing a high-speed clock to the core and peripherals
• Clock source
• Cyclic redundancy check (CRC) Generator
• Up to 54 general-purpose I/O (GPIO) pins
• JTAG.EOnCE debug programming interface for real-time debugging

The interleaved boost converter system benefits from the eFlexPWM module, High speed comparator (HSCMP) module, Timer (TMR) module, Analog to Digital Converters (ADC) and Voltage reference (VREF_DAC). All these modules offer flexibility in its configuration enabling efficient two-channel interleaved Critical Conduction mode control. The configuration of each peripheral used in this application are described in detail in Section 4.4, “Peripheral usage.”

3 Interleaved Control Scheme

The system configuration and key waveforms are shown in Figure 1. The switch Q1 (Q2) is turned on when the corresponding inductor current reaches zero. The turn-on time is constant. Because of the high current and voltage ripples, interleaving operation is used to reduce the current and voltage stress. In converters operating with constant switching frequency interleaving can be easily achieved. For converters operating in critical conduction mode interleaving is more difficult. The interleaving signal can be generated from a turn-on or turn-off signal by shifting the signal with half of a switching period. The interleaving control scheme used here shifts the slave turn-on signal to achieve interleaving. The master phase is turned on by zero current detectors. Turn-on time is given by Voltage controller. Assume that the difference between the two adjacent operation periods can be omitted. As a result, the previous operation period to determine the present shift period can be used. For high switching frequency operation (tens to hundreds kilo Hertz), this assumption is feasible.
The slave leg switch is turned on by shifting the driving signal for the master leg by 180°. Turn-on times for both legs are equal. If each leg has different parameters the slave leg can then work in continuous conduction mode (CCM) or in discontinuous conduction mode (DCM). The different parameters can be boost inductances, MOSFET parameters, sensing resistors, and also driving circuits. Algorithm for master/slave leg identification is used for satisfying master leg operation in CRM and slave leg operation in CRM or slightly in DCM.

Because the inductor currents are 180 degree out of phase, they cancel each other reducing the input ripple current. The input ripple current and the inductor current varies with the duty cycle change. The best input ripple current cancellation happens at a 50% duty cycle. The duty cycle of the step-up converter operating from dc input varies with the input voltage. The inductor ripple current cancellation is constant during constant input voltage. The duty cycle of the CRM PFC converter is variable with input voltage and
phase angle changing. Based on this, the inductor ripple current cancellation will not be 100% in the whole grid period on PFC circuit, but it is dramatically decreased.

The inductor ripple current cancellation allows reduction of the boost inductor size. This is due to the energy storage requirement of the two interleaved inductors being half that of single stage converter designed for the same power level, switching frequency and inductance.

In general, the DM noise is related to the input ripple current. The DM noise of the CRM is much larger than CCM in the single phase converter. However, the DM noise of a two phase interleaving boost converter is much smaller than the single phase CCM and CRM converter. Therefore, the DM-EMI filter size is smaller.

The output capacitor current is the sum of the two diode currents. Interleaving reduces the output capacitor ripple current as a function of duty cycle. As the duty cycle approaches 0 percent, 50 percent and 100 percent duty cycle, the sum of the two diode currents approaches dc. At these points, the output capacitor only has to filter the inductor ripple current. This reduction in RMS current will reduce electrical stress in the output capacitor and improve converter reliability.

Based on all the benefits mentioned, the power density of the system can be significantly increased. Drawbacks of the interleaved approach are that there are more components, and complicated control algorithm. This is not such real disadvantage because the algorithm can be easily implemented and tuned, and also reused for another application.

4 System Design Concept

4.1 System architecture

The application system incorporates the high voltage power stage, signal conditioning stage, auxiliary power supply stage, and the controller stage based on the MC56F8257. The system architecture is depicted in Figure 3.
The power stage consists of two parallel connected boost converters operating in critical conduction mode. The interleaved approach has more components than the single boost converter. The control algorithm is complicated, but the components are smaller with lower operating limits and can reach a better overall performance and power density. As hardware is designed for the micro-solar inverter the EMI filter and bridge rectifier is skipped in case of PFC circuit.

The sensing circuitries are used for sensing input voltage, output voltage, and leg currents, accommodating them to the controller acceptable voltage level.

The drivers are used for amplification of driving PWM signal for boost converters MOSFET transistors.

The core of the controller stage side is the MC56F8257 controller on the controller board connected via two 40-pin connectors. The MCU controls all the circuits mentioned above.

4.2 System specification

The application meets the following performance specification:

- Hardware used
— Power Stage board
— MC56F8257 controller board

• Control technique incorporates:
  — Constant on-time control (voltage control)
  — DC-bus voltage loop (2 kHz control frequency and a variable switching frequency)

• Fault protection:
  — DC-bus over-voltage
  — Input under-voltage and over-voltage
  — Both legs over-current and cycle by cycle current limitation

4.3 System control process

This converter is operating in MASTER/SLAVE mode and therefore the control scheme in this application can be designed according to a traditional single-channel converter. The function of this converter operating in voltage mode control is to control the output voltage and keep it stable during load variation. One DC bus voltage control loop is implemented in the system to meet steady state and dynamic requirements for interleaved boost converter operating in CRM.

The DC bus voltage loop is meant to:

• Control the DC bus voltage to quickly follow the change of reference voltage and implement non-error control in steady state
• Determine the permitted maximum current output through limiting the voltage regulator output
• Provide a timely anti-jamming function against the change of load

To achieve a good performance of the DC bus voltage loop a PI controller is used.

Figure 4 shows the system control scheme of a single-phase two-channel interleaved boost converter.

When the start-up phase is reached, the reference DC bus voltage is increased with a ramp according to the required DC bus voltage. The comparison between the reference DC bus voltage and the actual measured DC bus voltage generates a DC bus voltage error. Based on the error, the DC bus voltage controller generates the output voltage for both channels of the PWM modulator. After the DC bus required value is reached, the DC bus controller is keeping an output voltage on the required level and operating in Run mode. If the switching period (output power) is below 20% slave leg stops to generate the PWM signal and thereafter only the master leg is operating. More details relating operating one or both legs are in Section 4.3.4, “Phase management.”
To protect the converter against undesirable performance, software safety features are implemented. These safety features are enumerated in Section 4.2, “System specification”.

As this interleaved converter operates in MASTER/SLAVE mode the identification routine of which leg is master and which slave is important. A routine is executed every start-up of the application. The details of this algorithm are summarized in section Section 4.3.2, “Master/slave identification”.

When slave leg operation is required based on an increased output power, it needs synchronization with MASTER leg, it is important for a correct interleaved operation. See section Section 4.3.3, “Master/slave synchronization” for more details.

4.3.1 Over current and cycle by cycle limitation

Two over-current fault signals are available in this design. Each over-current signal serves for over-current protection with the support cycle by a cycle current limitation for each converter leg. The over-current threshold value can be set in software only changing VREF_DAC_x (A, B or C) values. Both fault signals are internally connected via XBAR to PWM Fault 0 and Fault 1 signal and both trigger Fault_ISR. Detail Fault_ISR processing is depicted in section x.x Fault_ISR.

4.3.2 Master/slave identification

The parameters of each leg of the boost converter are not identical. The inductances, mosfet parameters, sensing resistors, and control circuits which generate PWM signals can vary. Therefore, generating the same PWM signals for each leg does not generate the same output voltage. It is also difficult to synchronize both legs to operate in interleaved and critical conduction mode. The master/slave algorithm is used for identifying what converter leg switching period takes longer for the same duty cycle. That converter leg is the master leg.

This algorithm is executed one time in the AppStartUp state. During algorithm execution it captures timer values, several times. These timer values represent the switching period from Timer B0 for leg 1, which is summarized and the average value calculated from this sum. The same capturing values from Timer B1 and sum for the leg 2. The average switching period value is bigger and that leg is the MASTER leg. The MASTER leg then operates in critical conduction mode and slave leg too or slightly in discontinuous mode. Operation in continuous conduction mode of slave leg is therefore eliminated.
4.3.3 Master/slave synchronization

When both leg operations are required the SLAVE leg is synchronized with the MASTER leg. The driving signal for the SLAVE leg lags the MASTER signal about half of the switching period. In critical conduction mode the switching period can vary every switching cycle. Hardware and software support is used for correct implementation of the interleaved critical conduction mode algorithm. Hardware synchronization can be called the control where the slave delay time is based on capturing the switching period by capturing timer values using Timer B0 and Timer B1. Software synchronization can be called the control where slave delay time is calculated based on formula.

Software support is described detailly in section Section 4.3.5, “Slave delay calculation.” Software support is needed because the synchronization signal based on the captured switching period using timers is inaccurate during transient.

4.3.4 Phase management

Phase management is the algorithm that judges if one leg or both legs will be operating. The decision algorithm can be easily understand from Figure 4. Judge parameter is the MASTER switching period. The maximum switching period is normalized and relates to the maximum output power. This feature is useful for operating on a low load that one leg can be switched off and a higher overall efficiency reached.

![Figure 4. Phase management operation](image)

4.3.5 Slave delay calculation

As a slave is lagged half of the switching period and the switching period can be calculated using the equation in Figure 6. It can be used for slave leg synchronization with the master leg.

The switching cycle generally consist of two phases—when the switch is on (t-on time) and when the switch off (t-off time)—this is conduction time. During t-on time, the input voltage is applied on the coil and current is increases linearly see Figure 5.
During switch off time, the difference between output and input voltage is applied on the coil and current flowing through coil is decreasing linearly. Next switching cycle begins when the coil current reaches zero. Switching period is the sum of switch on and off time. Half of the switching period is the slave delay synchronization signal and is expressed in the equation below.

$$t_{delay} = \frac{1}{2} \cdot t_{on} \frac{V_{OUT}}{V_{OUT} - V_{IN}}$$

There:

- t-on—On-time
- Vout—Output voltage
- Vin—Input voltage

### 4.4 Peripheral usage

The application uses only essential peripherals for the control technique implemented in the application code—eFlexPWM, ADC, QTIMER, XBAR, VREF_DAC, and HSCMP (see Figure 3 for details). Other peripherals are disabled. A peripheral initialization is executed using Freescale QuickStart tool which offers a simple-to-use interface for all device peripheral settings. Detail configuration of used peripherals are in the following sections.

#### 4.4.1 QTIMER

The MC56F8257 contains two timer modules each with four timers. Each timer can operate as a timer or as a counter. The counter provides the ability to count internal or external events. QTIMER_B0 and...
QTIMER_B1 are used and operate in count mode. QTIMER_B0 is used for counting the switching period for leg 1 and QTIMER_B1 for counting the switching period for leg 2. A configured MC56F8257 Quad Timer module B is as follows:

- **QTIMER_B0**
  - Running at frequency 60 MHz (IPB clock/1)
  - Capture rising edges of input T0 (bus clock signal)
  - Input capture on secondary source
  - Secondary source is comparator B output (CMPB_OUT), CMPB_OUT signal is connected via XBAR

- **QTIMER_B1**
  - Running at frequency 60 MHz (IPB clock/1)
  - Capture rising edges of input T0 (bus clock signal)
  - Input capture on secondary source
  - Secondary source is comparator C output (CMPC_OUT), CMPC_OUT signal is connected via XBAR

### 4.4.2 XBAR

The crossbar switch module implements an array of 30 outputs and 22 inputs of combinational digital multiplexes. All 30 multiplexes share the same 22 inputs in the same order, but each multiplex has its own independent select field.

This module is designed to provide a flexible crossbar switching matrix that allows any input (typically from external GPIO or internal module outputs) to be connected to any output (typically to external GPIO or internal module inputs) under user control. This is used to allow user configuration of data paths between internal modules and between internal modules and GPIO.

The application configuration is done as follows:

- **Channel 6. ADCA Trigger**
  - PWM0_TRIG_COMB signal is used for ADC synchronization

- **Channel 16. PWM submodule 0 External synchronization signal**
  - CMPB_OUT signal is used for PWM synchronization

- **Channel 17. PWM submodule 1 External synchronization signal**
  - CMPC_OUT signal is used for PWM synchronization

- **Channel 21. PWM module Fault0**
  - CMPA_OUT/ CMPB_OUT/CMPC_OUT signal is used as the PWM fault signal

- **Channel 22. PWM module Fault1**
  - CMPA_OUT/ CMPB_OUT/CMPC_OUT signal is used as the PWM fault signal

- **Channel 26. Quad Timer B0 Input**
  - CMPB_OUT signal is used as secondary source input signal

- **Channel 27. Quad Timer B1 Input**
— CMPC_OUT signal is used as secondary source input signal

### 4.4.3 eFlexPWM

Enhanced Flex Pulse Width Modulator is a dedicated peripheral that enables generation for driving PWM signals for MOSFET transistors. PWM_A driving signal from the PWM submodule 0 is used for driving MOSFET for leg 1. PWM_B driving signal for the PWM submodule 1 is used for driving MOSFET from leg 2.

Two PWM submodules are used configure as follows:

- **PWM submodule 0**
  - Clock source IPBus clock
  - Running at variable frequency
  - INIT value = 0
  - VAL1 modulo 2999 - resolution 11bits
  - Independent mode
  - PWM_A output in positive polarity is driving signal for MOSFET transistor
  - Trigger 4 enabled provides synchronization signal for slave leg (PWM0_TRIG_COMB) in case leg 1 it is the MASTER leg
  - PWM middle cycle reload is generated every period
  - External synchronization signal is selected—CMPB_OUT in case leg 1 is the MASTER leg or PWM1_TRIG_COMB in case leg 1 is the SLAVE leg.

- **PWM submodule 1**
  - Clock source IPBus clock
  - running at variable frequency
  - INIT value = 0
  - VAL1 modulo 2999 - resolution 11bits
  - Independent mode
  - PWM_A output in positive polarity is driving signal for MOSFET transistor
  - Trigger 4 enabled provides synchronization signal for slave leg (PWM1_TRIG_COMB) in case leg 2 it is the MASTER leg
  - PWM middle cycle reload is generated every period
  - External synchronization signal is selected—CMPC_OUT in case leg 2 is the MASTER leg or PWM0_TRIG_COMB in case leg 2 is the SLAVE leg.

- **PWM Fault0 and Fault1**
  - Fault signals with HIGH level detection
  - Fault clearing automatic
  - Fault_ISR execution (for details see chapter x.x Fault_ISR)
  - Fault input filter disabled
4.4.4 ADC

The analog-to-digital (ADC) converter function consists of two separate 12-bit analog-to-digital converters, each with eight analog inputs and its own sample and hold circuit.

The fast ADC converter module is configured as follows:
- Input clock IPBus/6 = 10MHz
- Sample x set to DC-Bus voltage channel - ANAx
- Sample x set to DC-Bus voltage channel - ANAx
- Scan mode—Triggered parallel
- Trigger source—SYNC0 input (PWM trigger connected via XBAR)

4.4.5 HSCMP

The High Speed Comparator module (HSCMP) provides a circuit for comparing two analog input voltages. The comparator circuit is designed to operate across the full range of the supply voltage (rail-to-rail operation). All three comparators are used for boost inductors current zero-crossing signal detection and for over-current protection.

- HSCMP_A
  - Over-current leg 1 protection or over-current leg 2 protection
- HSCMP_B
  - Zero-crossing signal detection for leg 1 or over-current leg 1 protection
- HSCMP_C
  - Zero-crossing signal detection for leg 2 or over-current leg 2 protection

4.4.6 VREF_DAC

The 5-bit VREF_DAC is a selectable voltage reference for applications that need a voltage reference. A 5-bit digital signal input selects the output voltage from 32 levels, which varies from VDDA to VDDA/32. In this application all three voltage references are used as a threshold values for boost inductors current zero-crossing signal and as an over-current threshold.

- VREF_A
  - Over-current leg 1 threshold value or over-current leg 2 threshold value
- VREF_B
  - Threshold value for boost inductor zero-crossing signal for leg 1 or over-current leg 1 threshold value
- VREF_C
  - Threshold value for boost inductor zero-crossing signal for leg 2 or over-current leg 2 threshold value
5 Software Design

This section describes the software design of the one-phase two-channel interleaved boost converter application operating in critical conduction mode. The system processing is interrupt driven with the application state machine running in the background. The software is described in terms of the following:

- Main Software Flow Chart
- Application Interrupts —PWM_ISR and FAULT_ISR

Code can be easily incorporated into the target application as a part of the whole system because the application is really independent from other tasks. Attention must be be taken onto correct timing of this algorithm.

5.1 Main software flow chart

After a reset the application performs the following routines and enters the endless (main()) loop.

- PeripheralCoreInit
  — Initialization of peripherals and controller core
- AppInit
  — Initialization of application variables after reset
- archEnableInt
  — Global interrupt flag enabled
Main software flow chart, see in Figure 7.

![Main SW Flow Chart](image)

**Figure 7. Main SW Flow Chart**

The application background main() loop contains the application state machine and clear watchdog timer. The application state machine incorporates three states (see Figure 11):

- **AppStartUp**
  - Master/slave indentification
  - Voltage controller parameters setting for start-up and voltage controller enabling
  - Soft start—Increasing required output voltage of the increment value until it reaches the required DC-bus voltage value.

- **AppRun**
  - Normal operation of control algorithm—DC-bus voltage is kept at the required value.
  - Phase Management (algorithm described in section Section 4.3.4, “Phase management”).

- **AppError**
  - Disables PWM outputs
Transition between main states are performed using four transient states:

- **AppStartUpToRun**
  - State is executed after DC-bus required value is reached, continue in Run mode
  - Set voltage controller parameters for Run mode
- **AppStartUpToError**
  - Set Error flag
- **AppRunToError**
  - Set Error flag
- **AppErrorToStartUp**
  - Clear Error flag

### 5.2 PWM ISR description

The PWM interrupt service routine performs the entire control algorithm. The algorithm consists of subtasks which are:

- Read ADC quantities
- Phase management
- Slave leg synchronization with master
- Voltage Proportional-Integral (PI) controller execution (every 500 μs)
- Limit checking of all measured quantities
• Update PWM registers

The PWM_ISR interrupt service routine is executed regularly every 62.5μs. Initialization of the ADC conversion is executed every 62.5μs, and is initialized by the VAL0 value register. The conversion is running in the background. The voltage PI controller design is well known and is described in detail in many papers. The PWM ISR flow chart can be seen in figure 12.

**Figure 9. PWM_ISR Flow Chart**

5.3 **Fault_ISR description**

The Fault_ISR performs the over-current protection of each leg of the boost converter. This interrupt is triggered by Fault0 and Fault1 in case of an over-current on any of the converter legs. Faults are configured as the automatic clear mode to support the cycle-by-cycle current limit function. When the interrupt is continuously triggered for a specified time the over-current fault is latched and the converter will shut down. Therefore, this interrupt has a higher priority than the PWM_ISR interrupt. The flow chart of the fault interrupt is shown in Figure 10.
Interrupt void Fault_ISR(void)

Initialize Fault timer (1st time in ISR)

If Fault timer > 100us
  yes
  Disable PWM
  Set Error flag
  Clear Fault timer

End of interrupt

Figure 10. FAULT_ISR Flow Chart