Digital PFC CCM boost converter

300 W design example using XMC1400 microcontroller

Robert Valascho
Sam Abdel-Rahman

About this document

Scope and purpose

This document introduces a digital control implementation for a Power Factor Correction (PFC) Continuous Conduction Mode (CCM) boost converter.

Intended audience

This document is intended for power and digital design engineers who wish to develop digital controllers for power converters using the XMC™ microcontroller family.

Table of contents

1 Introduction

Power Factor Correction (PFC) shapes the input current of the power supply to be in synchronization with the AC input voltage, in order to maximize the real power drawn from the AC supply. In a perfect PFC circuit, the input current is in phase with the input voltage (as it would be with a pure resistor), without any input current harmonics. This document presents the digital implementation of a CCM PFC boost converter.

1.1 Boost topology

Although active PFC can be achieved in several ways, the boost converter (Figure 1) is the most popular topology used in PFC applications.

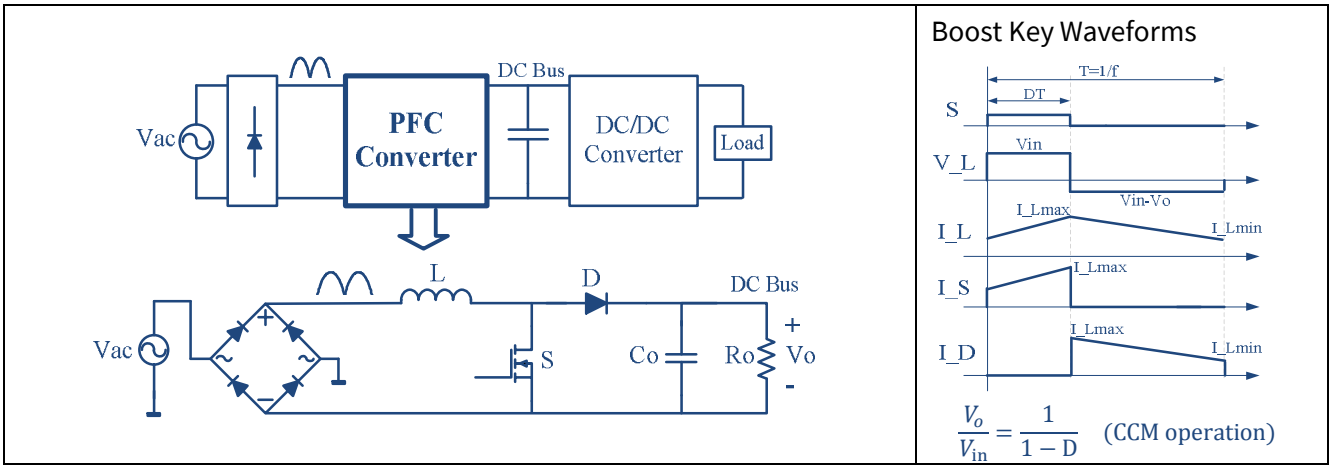

Figure 1 Structure and key waveforms of a boost converter

1.2 PFC modes of operation

The boost converter can operate in three modes: continuous conduction mode (CCM), discontinuous conduction mode (DCM), and critical conduction mode (CrCM). Figure 2 shows simulated waveforms that illustrate the inductor and input currents in the three operating modes, for exactly the same voltage and power conditions.

It is possible to state that, for low power applications, the CrCM boost topology has advantages in power saving and improving power density. However, at some medium/high power level the poor filtering ability and high peak current start to have significant disadvantages. At this point, the CCM boost topology starts to be a better choice for high power applications.

As this document supports high power PFC applications, a digital controller implementation for a CCM PFC boost converter will be discussed throughout, as it has unique advantages in handling the various conditions required in PFC applications.

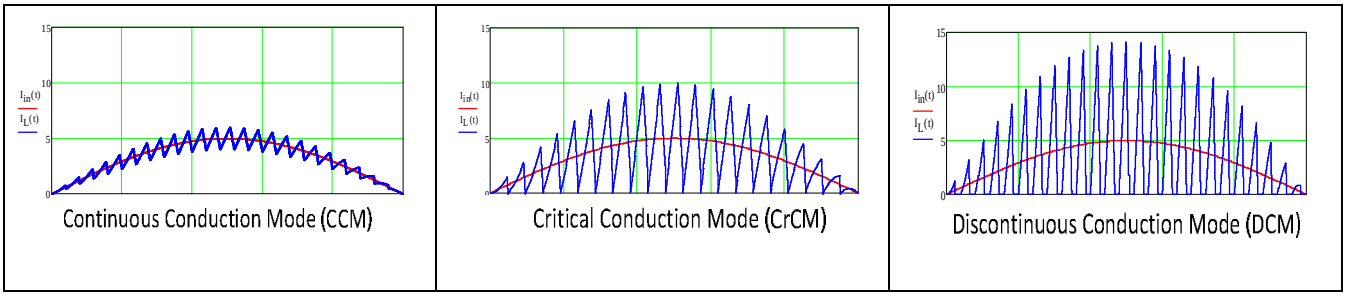

Figure 2 PFC Inductor and input line current waveforms in the three different operating modes

1.3 Analog vs. digital Power Factor Correction (PFC)

Figure 3 and Figure 4 show the average current control implementaion using analog and digital controllers. The outer loop is a voltage controller that regulates the boost output to the desired DC bus voltage, which is typically supplying a secondary dc-dc stage. The output of the voltage compensator is fed into a multiplier along with two other inputs; the first is an inverse of the RMS input voltage squared, which is used to normalize the loop gain and make it independent of the line voltage throughout the universal input range. The second input to the multiplier is the instantaneous input voltage, this will shape the current reference to be proportional to the line voltage and achieve PFC. The result of the multiplier is a rectified sinusoidal signal that is passed to the cuurent loop reference input.

The current loop is the inner and faster loop. Its main function is to regulate the sensed inductor current to the sinusoidal reference; the current controller will modulate the PWM duty ratio accordingly. This is discussed in more detail in the control loop design section 3.

The analog controller has some design considerations, where the flexibility of the digital controller can benefit the optimization of PFC. Benefits of the digital PFC include:

1. Due to the low frequency ripple at the output, the voltage compensator ripple will have a direct effect on the line current total harmonic distortion (THDi). Digital control can filter out the low frequency ripple and drive the inner current control loop with an almost constant reference through the line cycle, for a better THDi.
2. For the reason stated above and for stability purposes, the voltage loop compensator has to be designed to pass Nyquist criterion with the limitation being the AC line cycle itself. This creates design targets below 23.5 Hz crossover frequency (47 Hz considered worst case). The slow dynamic voltage compensator has a negative effect on the output voltage transient during load steps. To overcome this, a non-linear voltage compensator can be implemented with digital control to satisfy both of the steady state slow dynamics (for better THDi) and also the faster dynamic compensator during load jumps (for better output regulation).
3. Current measurement filtering adds an inherent delay to the real-time current, and also shows an error in DCM. As there will always be a transition from DCM to CCM, and vice versa, in every line cycle; the same filter cannot be optimized for both modes. Digital control can sense the operational mode (CCM or DCM) and apply a DCM current sense correction accordingly.
4. Digital control can adapt the converter operation and control based upon the different line and load conditions. Several fuctions can be implemented to improve THDi including frequency increase at high line, different optimized compensators at low and high line etc.

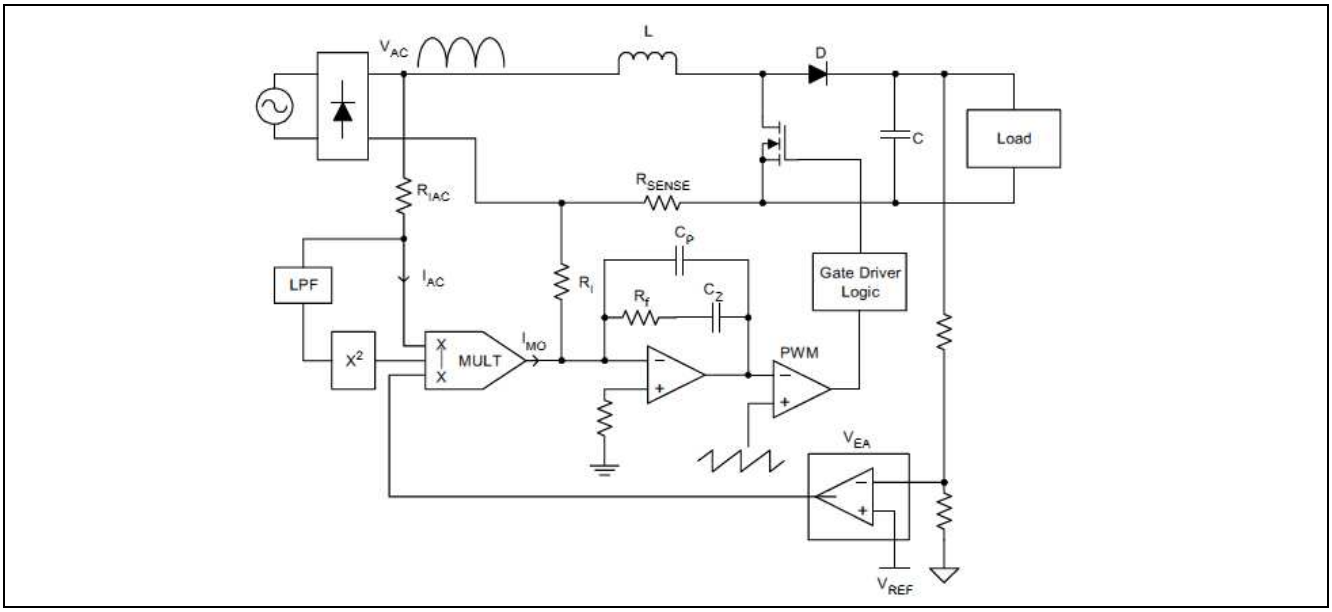

Figure 3 Analog PFC control diagram

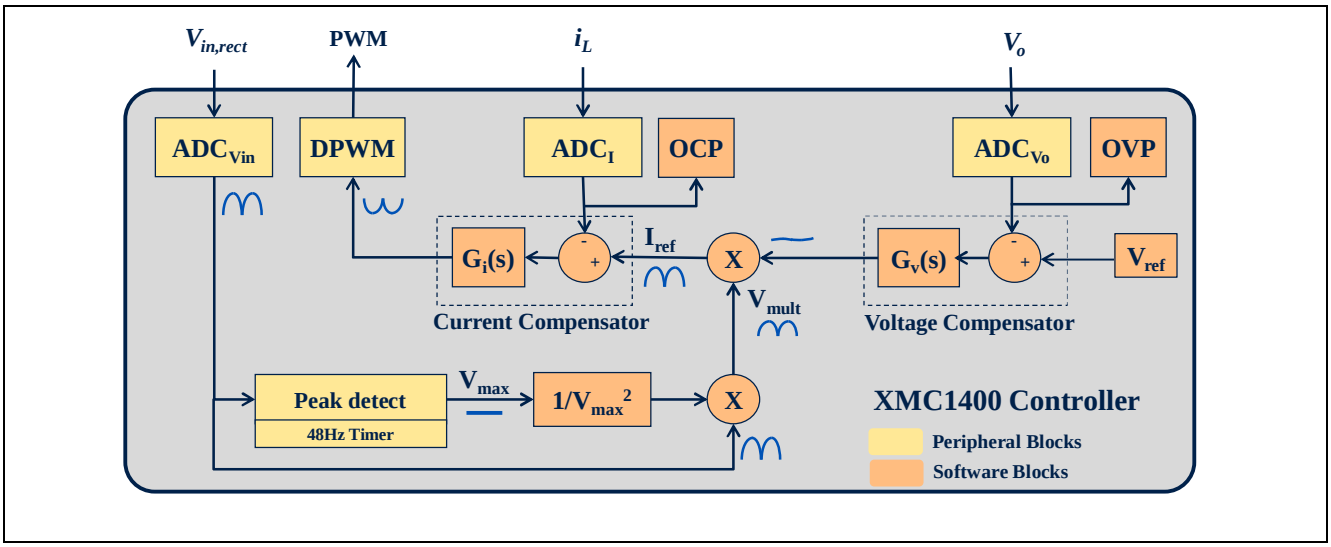

Figure 4 Digital PFC control diagram

2 300 W PFC implementation using XMC1400

2.1 Design example hardware specifications and schematics

Table 1 Specifications

Input voltage	85-265 VAC 60 Hz
Output voltage	380 V
Maximum power steady state	300 W
Switching frequency	65 kHz
Hold-up time	16.6 ms @ $V_{O,min}$=340 V
Controller	Digital XMC1400 Analog ICE3PCS01G
Inductor	~570-800 uH
Output capacitor	220 µF/450 V

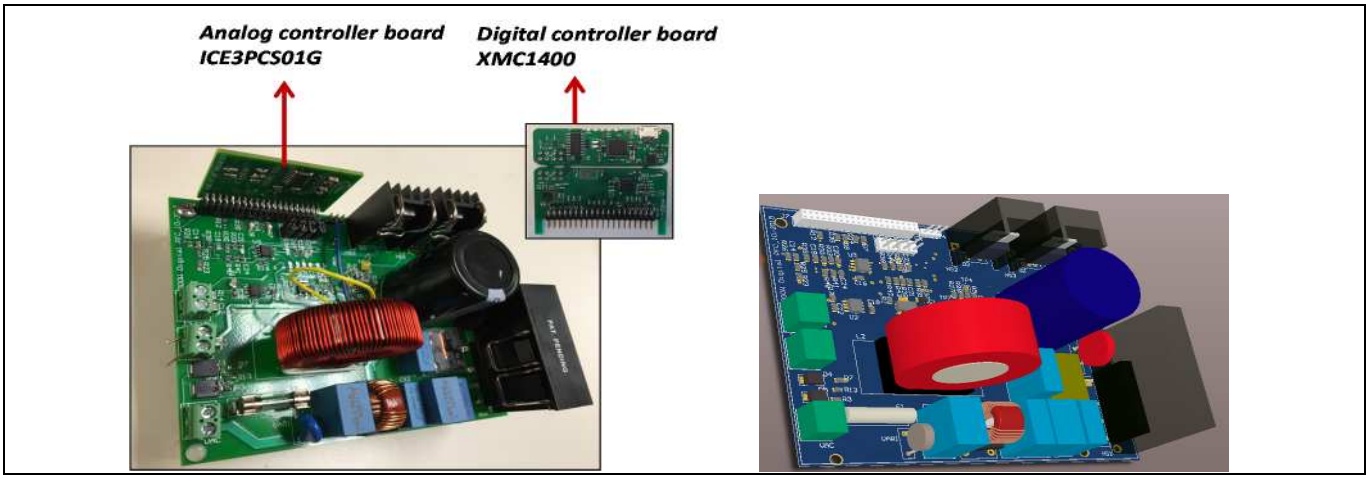

Figure 5 Board picture

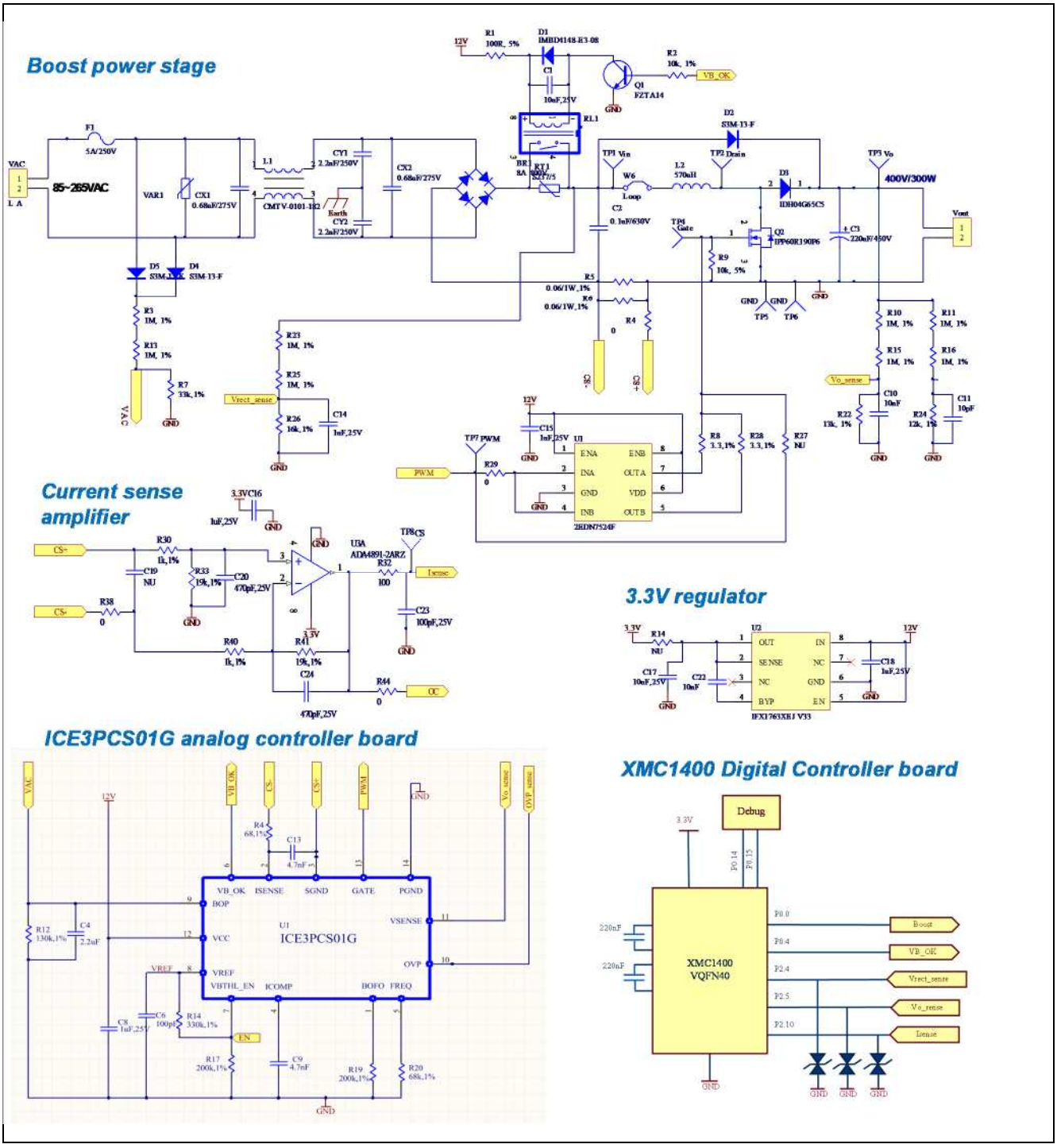

Figure 6 Schematics

2.2 Flow chart

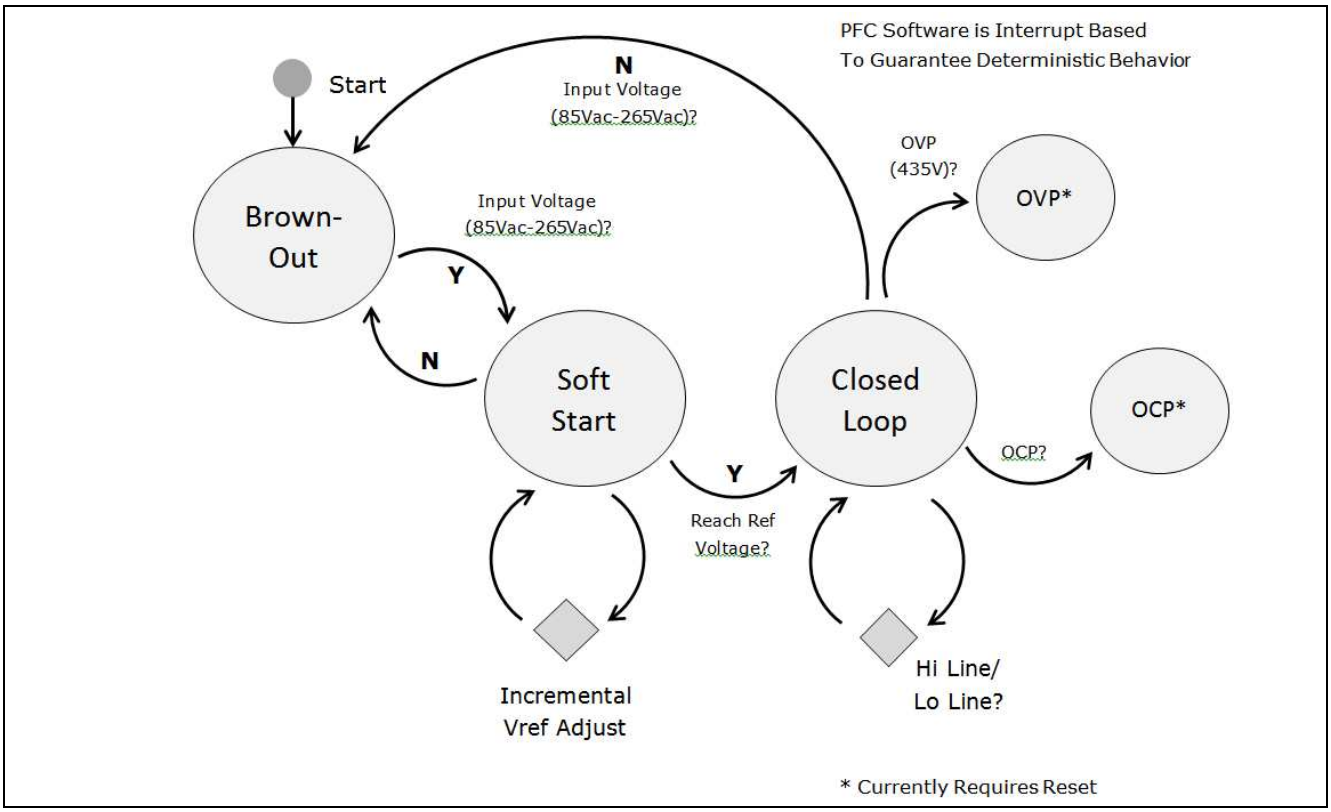

Figure 7 Flow chart

Figure 7 shows that, upon a reset, the controller is considered to be in a brown-out condition. In order to exit the brown-out state, the line voltage is evaluated to be between 85 VAC and 265 VAC via analog measurement of the input line voltage. At this point, the controller enters soft start mode.

Soft start will initiate closed loop control for the inner current control loop and the outer voltage control loop. The design of these loops will be discussed in detail later in this document. As opposed to steady state control, or closed loop, we will slowly increment the voltage output to reach the reference voltage target. Both controllers are running, but with a modified, incremental adjustment to the targeted voltage, until the desired reference voltage is reached. Once the targeted reference voltage is reached, the controller is considered to be in the closed loop state.

The targeted reference voltage and control loops are adjusted based on two different input criteria; a HI line criteria of >=185 VAC, and a LO line criteria of <=170 VAC. The closed loop HI/LO criteria allow for efficiency improvements and also control loop adjustments, thus optimizing the power supply input/output conditions. When in closed loop, as sudden load changes occur, transients can be observed in the output. If an out of range voltage or current condition occurs, the controller can activate OCP (over current protection) and OVP (over voltage protection). Entry to the closed loop condition is signaled by soft-start criteria, and exit of the closed loop condition can be either an OCP or OVP fault or the line voltage returning to a brown-out condition.

2.3 Timing diagram

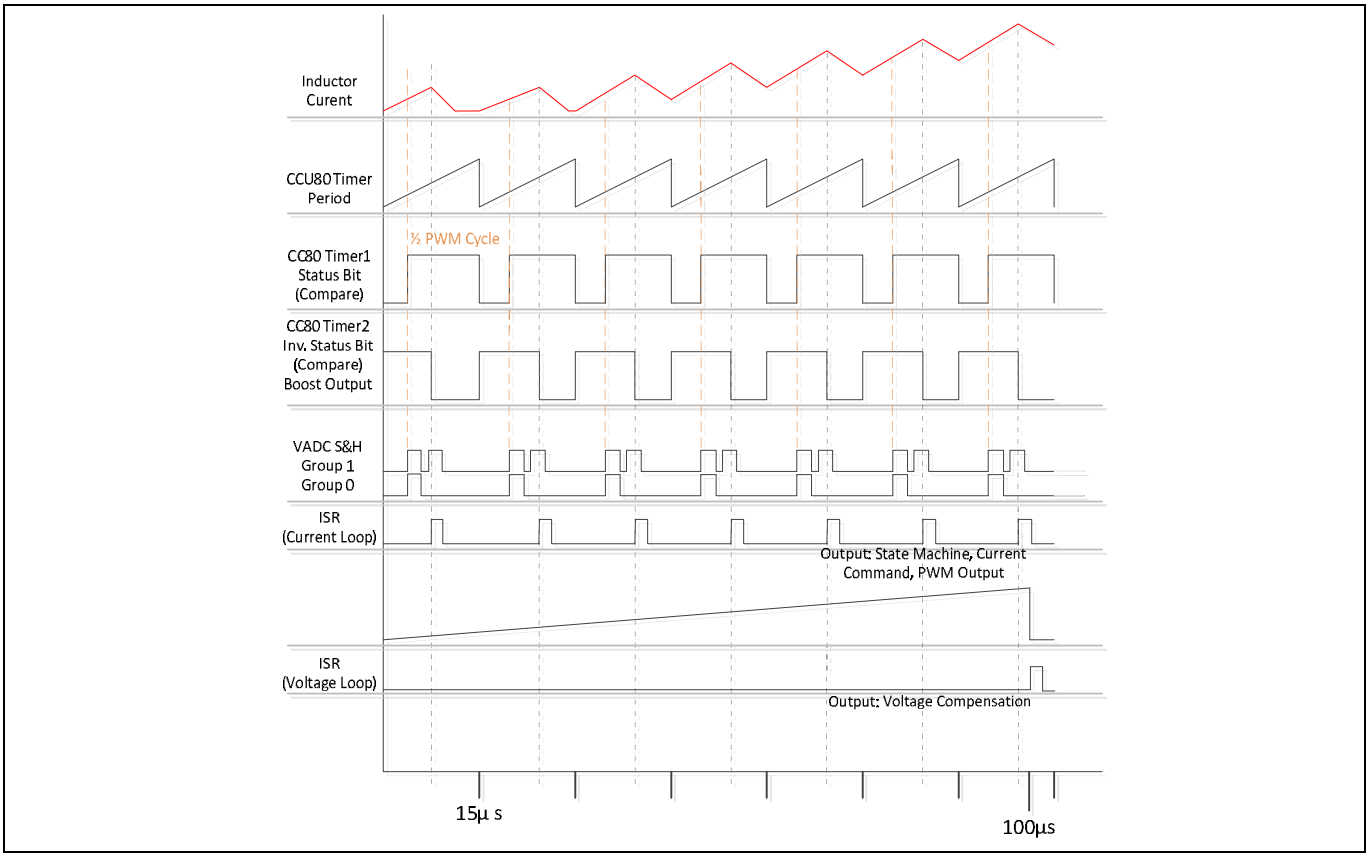

Figure 8 Timing diagram

Figure 8 shows the timing diagram. The outer voltage controller runs at 10 kHz; its main purpose is to monitor any error related to the regulated voltage output. Within the voltage loop, there are additional adjustable parameters related to features, such as soft start and boost follower mode (discussed more in the controller features section).

The output of the voltage loop ISR is the voltage compensator output. This is fed into the multiplier for the current command, which is contained in the inner current loop. The voltage ISR is calculation only and does not affect actuation; it simply provides output to the inner loop.

The inner current loop is executed every switching cycle. Using the CCU8 slice 0 peripheral, the first compare register, CCU80_CC80.CR1S, can be used to control the PWM output. The CCU is configured so that the output duty cycle increases when the compare register value increases. The ST bit has been changed to output high PWM first, which in turn re-orders the error command so that a positive error causes a positive duty increase. CCU80_CC80.CR2S tracks the same period, but has an independent compare value that can be used as either an output or a trigger. For this design, we are using the compare value as half of the PWM period to create a trigger for analog sampling. The intention is that the voltage and current at the mid-point of the switching cycle can be discretely sampled, providing a representation of the average current.

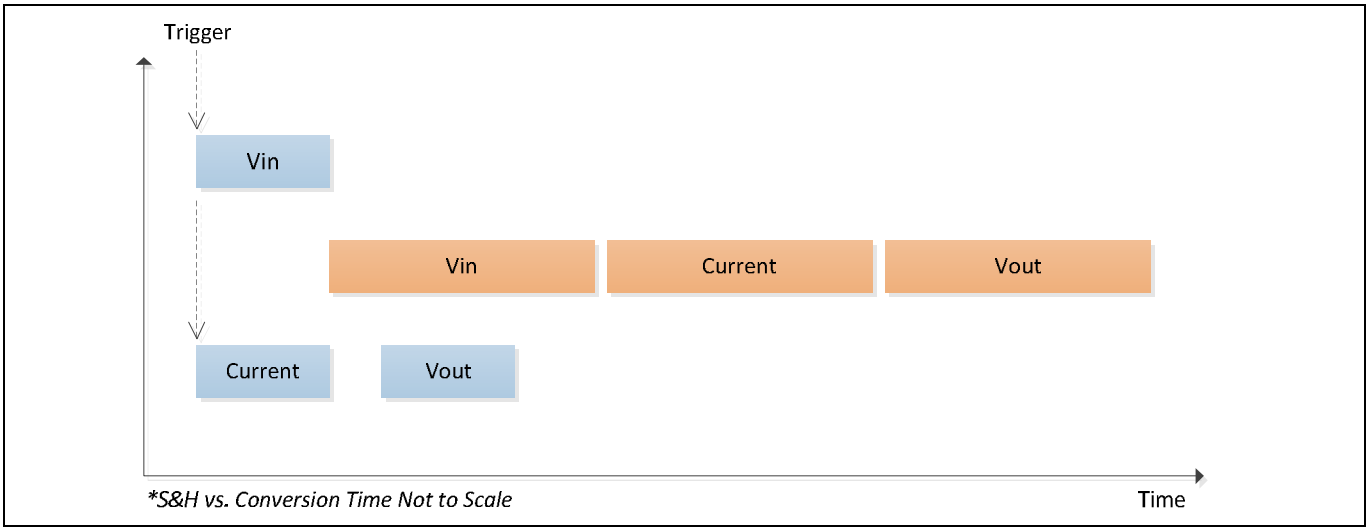

Figure 9 Sampling time (blue) vs. conversion time (orange)

Figure 9 shows that, at trigger, voltage and current are sampled simultaneously, using the two sample and hold units available in the VADC. This eliminates any additional error due to series measurements or jitter. The output voltage is then sampled after completion of the sample of the input voltage. Upon every switching cycle, and after the voltage and current measurements are complete, an interrupt service routine is executed. The routine first runs through the state machine check to determine PFC status, and then executes the current control algorithm as previously described. Based on the input voltage and current measurements, from the same cycle, the current command is calculated. The current command can then be compared against the inductor current, and the error is calculated. This error is fed through the inner loop compensator and the output PWM is adjusted.

2.4 XMC™ peripheral setup

The CCU (Capture Compare Unit) timer structure allows for inclusion of many of the necessary features for CCM PFC control. The CCU8 can be used to manage both actuation and control sensing times. In Figure 10, a single slice of CCU8 (4 slices per module) is used to manage current loop sensing and boost actuation. This is handled by the two compare registers per slice. One compare register, using CCU80_CC80 (Slice #0 of CCU8 Module #0), can be used to control the boost output while the other is halved to represent average current calculation. The updated values can be handled synchronously through a single shadow transfer.

Upon a period match the current loop ISR is serviced and CCU80_CC81 (Slice #1 of CCU8 Module #0) is controlling the voltage loop compensator. This loop is running at 10 kHz as opposed to the 65/100 kHz of the current loop. Global Enable starts this synchronously to the current loop, but it has its own time base. Finally, a slow loop, equivalent to a worst case line cycle of 47 Hz, is implemented to time an RMS measurement value. The RMS value will be used as a scalar for the current command explained in the theory portion of this document. The RMS timer expires and, using the ADC peak detection feature, a peak sampling over a full line cycle is taken to give a representation of RMS voltage.

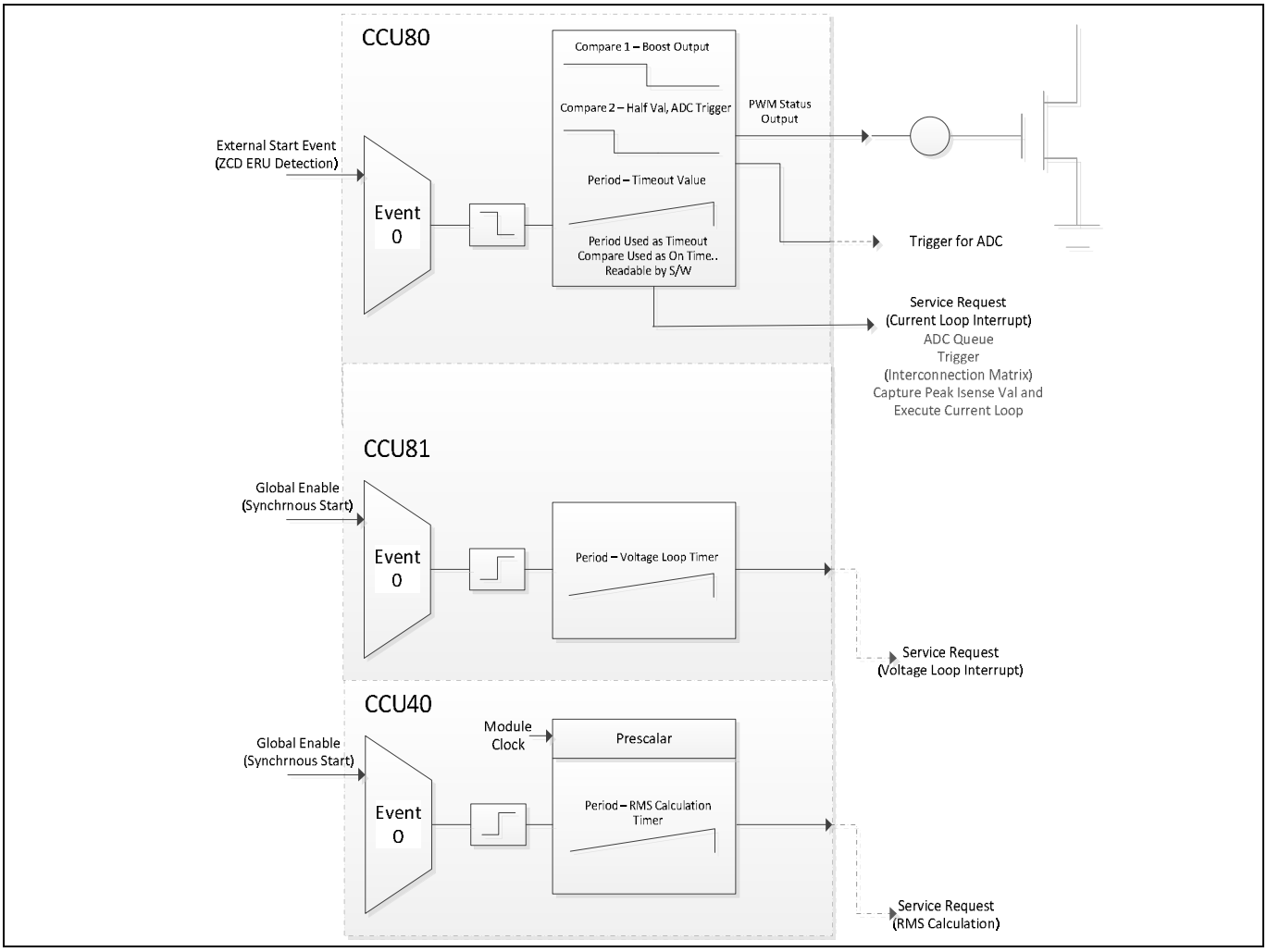

Figure 10 CCU timer setup

In Figure 11, the VADC setup is partitioned into two groups (two sample and hold circuits and one analog-to-digital converter). The synchronized measurements for the current loop, which is the current and input voltage, are separated to two different groups. This allows simultaneous measurement, avoiding any offset due to a timing difference. The output voltage is taken post sampling of the input voltage.

In addition, the most recent input voltage ADC result is stored in result register 15. By using the adjacent result register 14 as maximum value storage (VADC_G0.RCR, FEN bits), a peak value can be stored for use in the current command. This peak value can be used as the equivelant of an RMS scalar. The result register for each sensed input, input voltage, output voltage, and sense current, are updated at the boost frequency.

In addition, the VADC has built in automatic limit checking. Both the sensed current channel and voltage output channel have boundary conditions configured. The boundary conditions are set as OVP (over voltage) and OCP (over current). These boundary settings should only happen in fault condition. If the boundaries are crossed, then a cycle independent service request is set, and the fault condition is managed immediately upon reaching the interrupt.

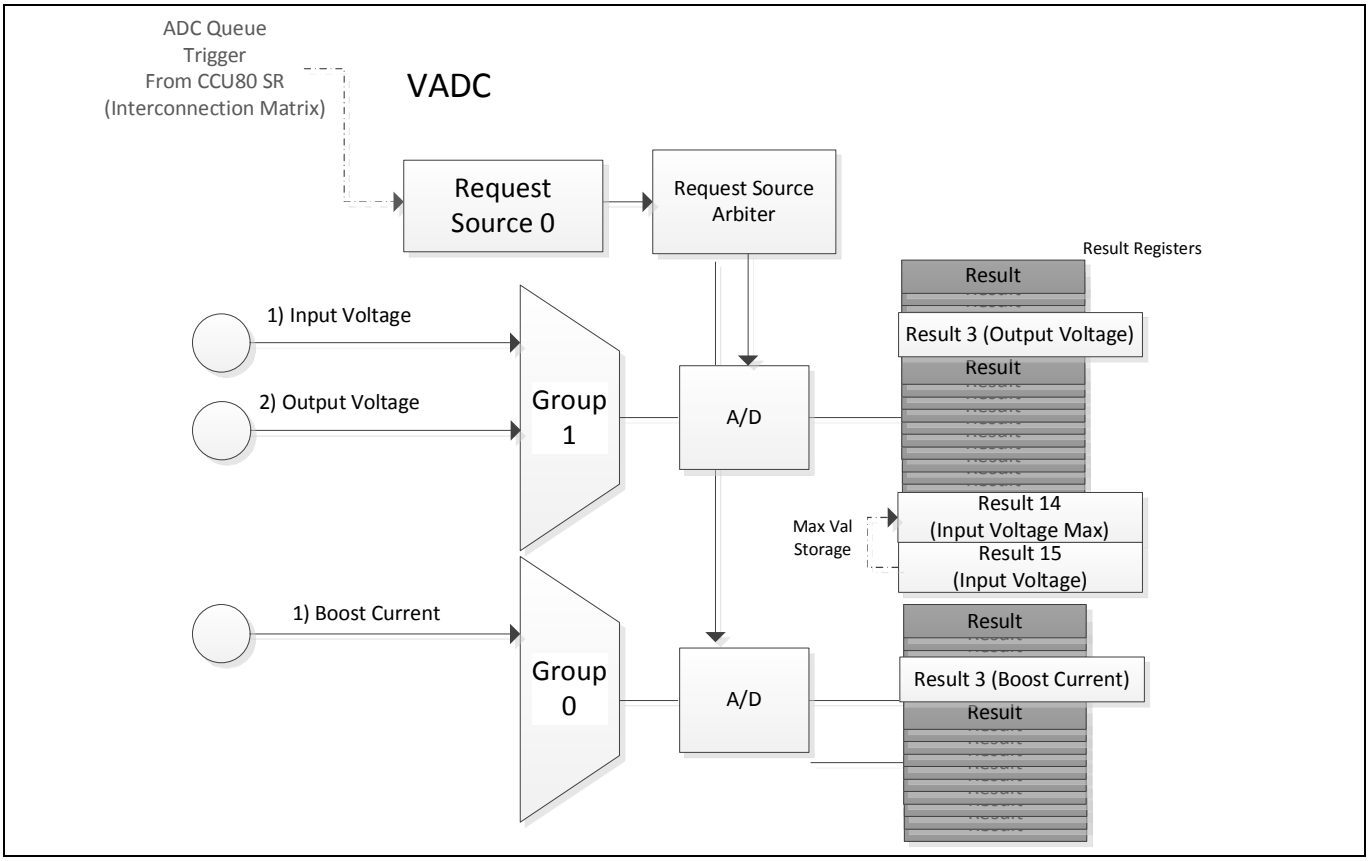

Figure 11 VADC setup

3 Control loop design

Figure 12 shows the complete boost converter plant with the current and voltage digital control loops. The main objective is to implement a stable and dynamic current and voltage digital compensators ($G_{i\text{-}PID}$, G_{v_PID}). Hence, we need an ac model for both loops to plot their bode plots in the frequency domain, and test for stability with adequate bandwidth and gain margin. As this is a hybrid system with an analog (continuous) plant and a discrete controller, the approach used is to model both loops in the analog domain and design s-domain controllers using an analog zeros and poles placement technique, then use the s-to-z bilinear transformation method to digitize the designed s-domain controllers. Finally, the PID coefficients are derived from the z-transfer function.

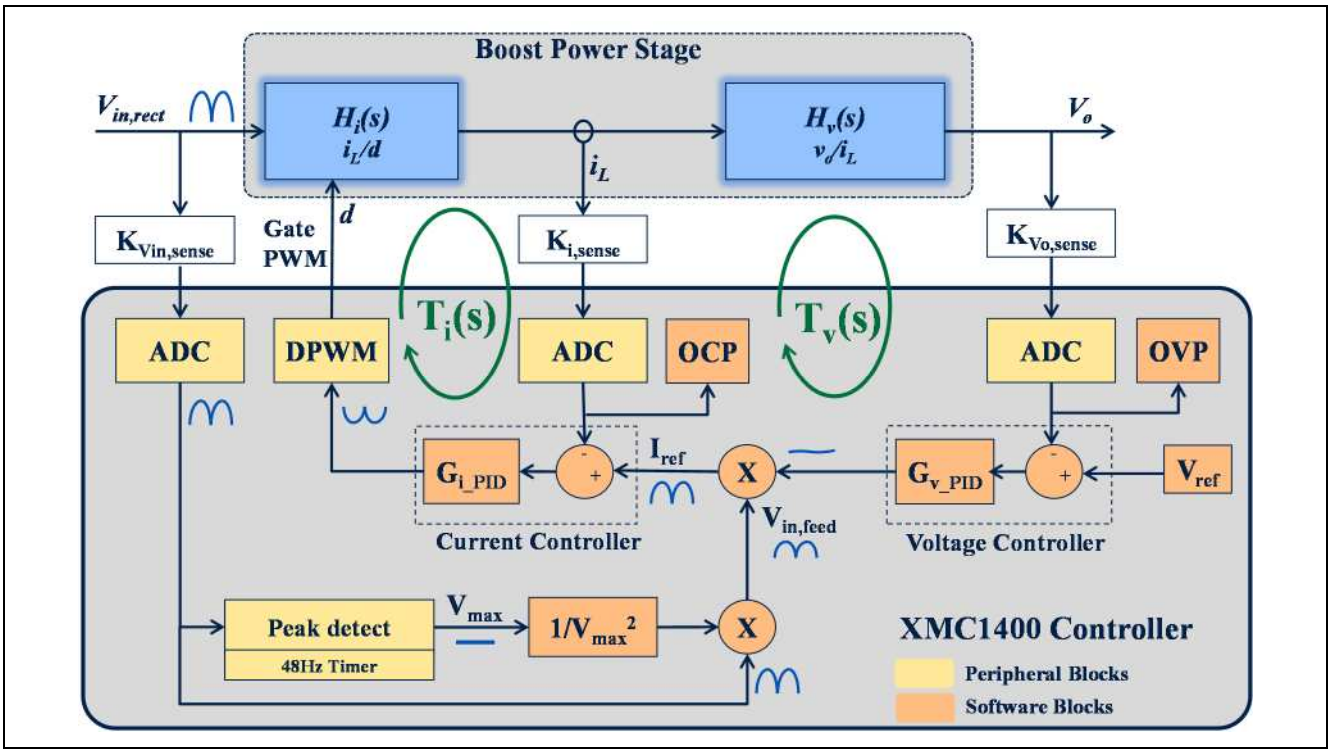

Figure 12 Boost converter current and voltage digital control loops

3.1 Current loop

Figure 13 shows the current loop, which gets its reference (or inductor current command) from the outer voltage loop. The current reference is a rectified sine waveform with twice the line frequency; the current controller will work to keep the error between the reference and the sense inductor current at zero. So for the current loop to be dynamic enough to track the $2xf_{line}$ reference signal, its bandwidth must be at least one order of magnitude higher. Typically the current loop is designed with a crossover of 10% of the switching frequency (65 kHz), or 6.5 kHz. The output of the current controller is fed to the digital PWM to generate a driving pulse for the boost switch. The dynamic pulse duty ratio through the line cycle will shape the inductor current in synchronization with the reference signal, which is synchronized to the line voltage. Therefore the input current is proportional to line voltage, and power factor is near unity.

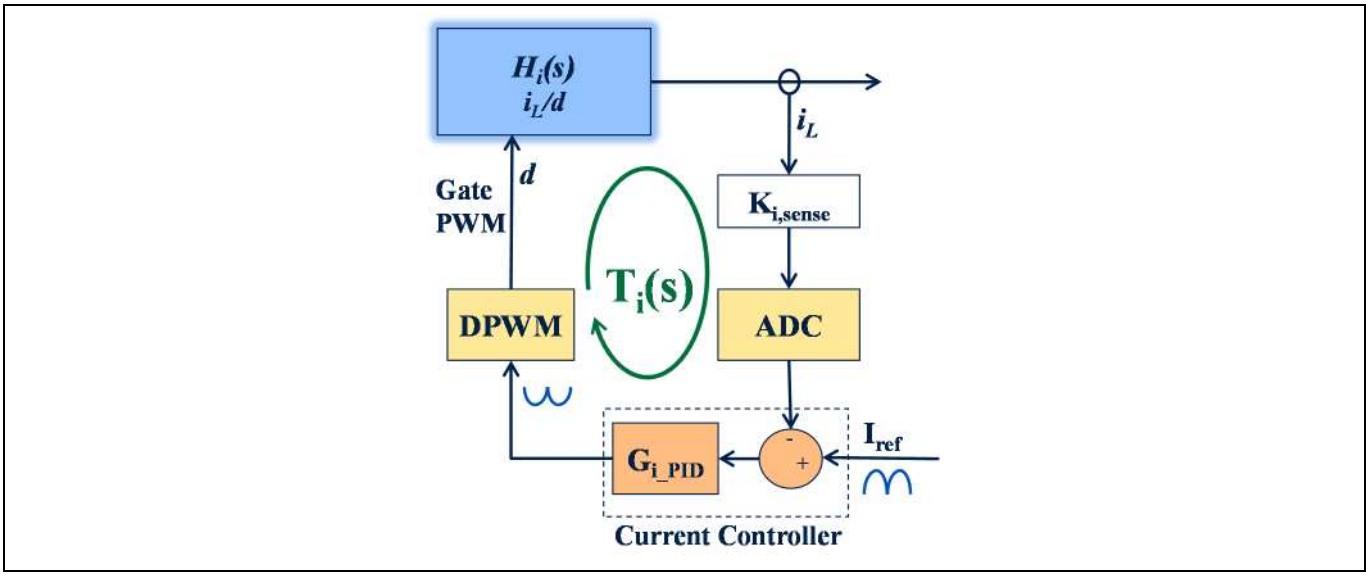

Figure 13 Current loop diagram

In order to design the digital current loop, the following steps are required:

Step 1 - Bode plots of the uncompensated loop

From Figure 13 the loop gain Ti(s) is:

$$Ti(s) = Hi(s) * Ki,sense * ADC * DPWM * Gi(s) \quad \text{Eq. 1}$$

The uncompensated loop gain is:

$$\frac{Ti(s)}{Gi(s)} = Hi(s) * Ki,sense * ADC * DPWM \quad \text{Eq. 2}$$

Boost transfer function, duty-to-inductor current (i_L/d)

$$Hi(s) = \frac{Vo}{s * L} \quad \text{Eq. 3}$$

Inductor current sense gain

$$Ki,sense = Rsense * Kop\_amp = 0.03 * 19 \quad \text{Eq. 4}$$

Analog-to-Digital converter gain

$$ADC = \frac{2^n}{Vref} = \frac{2^{12}}{3.3V} \quad \text{Eq. 5}$$

Digital PWM gain (based on 96 MHz timer resolution, 10.42 ns tick)

$$DPWM = \frac{1}{\text{DPWM Counts}} = \frac{1}{1457} \quad \text{Eq. 6}$$

Figure 14 shows the MathCAD magnitude and phase bode plots for the uncompensated current loop of Eq. 2

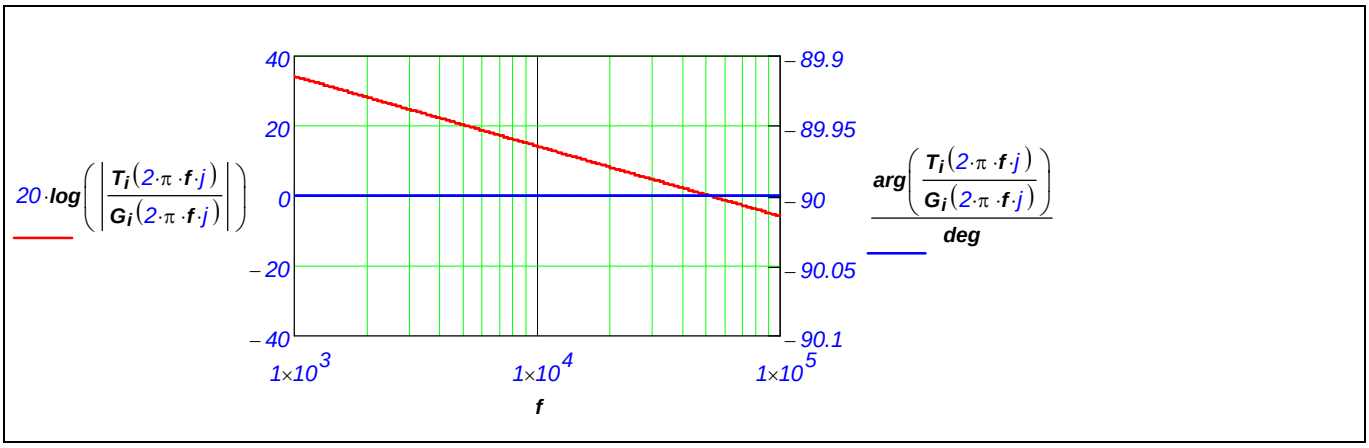

Figure 14 Uncompensated current loop bode plot

Step 2 - Design the current controller, place zeros and poles to reach a desired loop response with adequate crossover frequency and phase margin.

The current controller is one zero, one pole and an integrator, as in the transfer function below

$$Gi(s) = \mathrm{k} * \frac{\left(\frac{s}{2 * \pi * f_z} + 1\right)}{s * \left(\frac{s}{2 * \pi * f_p} + 1\right)} \qquad \text{Eq. 7}$$

Based on the uncompensated bode plots in Figure 14, the compensator was designed with the following zero, pole and gain values. This results a controller bode plot as shown in Figure 15.

$f_z = \frac{f_{sw}}{17} = 3.824\ kHz$, $f_p = \frac{f_{sw}}{2} = 32.5\ kHz$, $k = 1800$

Figure 16 shows the compensated current loop with a cross over frequency= 4.875 kHz and gain margin= 44°.

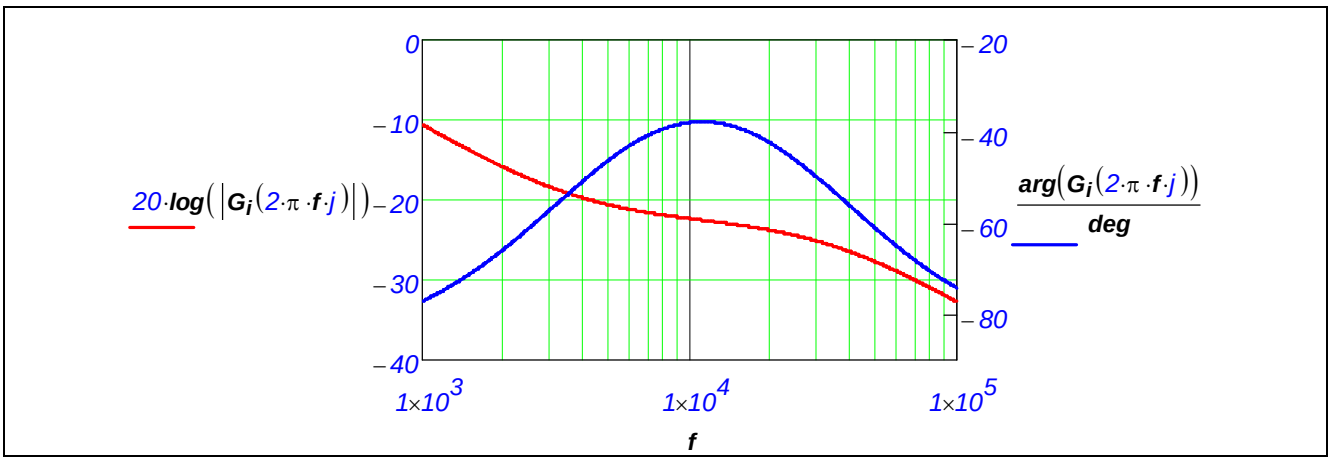

Figure 15 Current controller bode plot

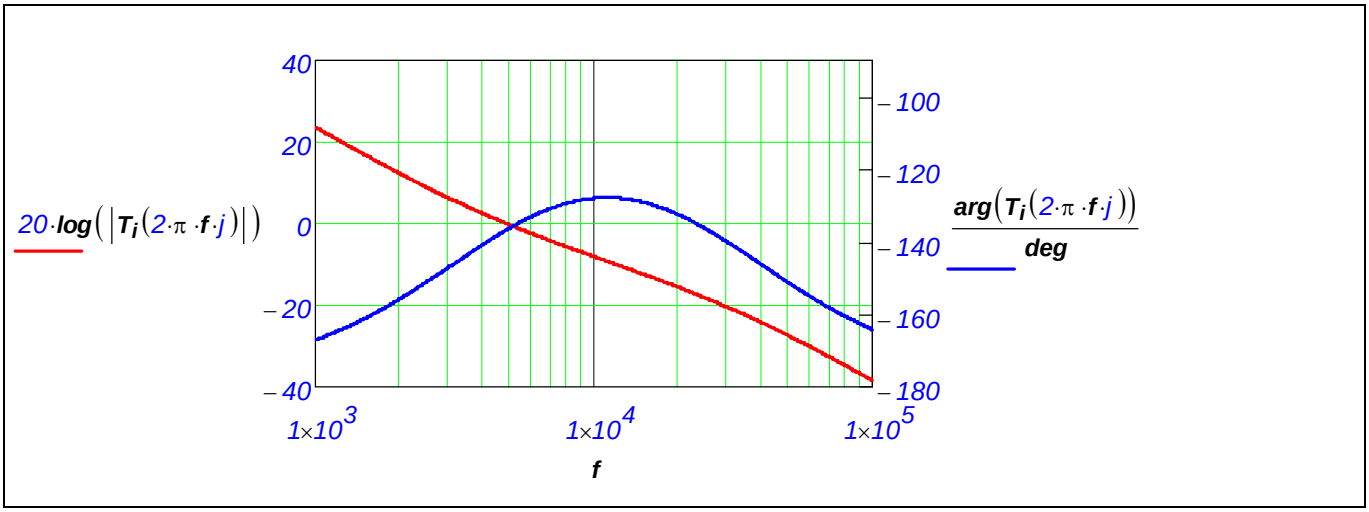

Figure 16 Compensated current loop bode-plot

Step 3 - Use the s-to-z bilinear transformation method to digitize the designed s-domain controllers.

In bilinear transformation, the formuls below is used.

$$s = \frac{2}{T} * \frac{z-1}{z+1} \qquad \text{Eq. 8}$$

where T is the current controller running period, in this case the current controller runs at same frequency as the switching frequency, so T=1/65 kHz.

By substituting the bilinear formula in the $Gi(s)$ transfer function, we obtain the following z transfer function

$$Gi(z) = \frac{1.34 * 10^4 * z^2 + 4.18 * 10^3 * z - 9.22 * 10^3}{2.47 * 10^5 * z^2 - 1.922 * 10^5 * z - 5.485 * 10^4} \qquad \text{Eq. 9}$$

Step 4 - Derive the PID coefficients from the z-transfer function.

The two-pole, two-zero digital filter expressed in Eq. 9 can be realized as a parallel filter structure, such as the the classic proportional, integral, derivative (PID) controller in Figure 17, which is described by Eq. 10.

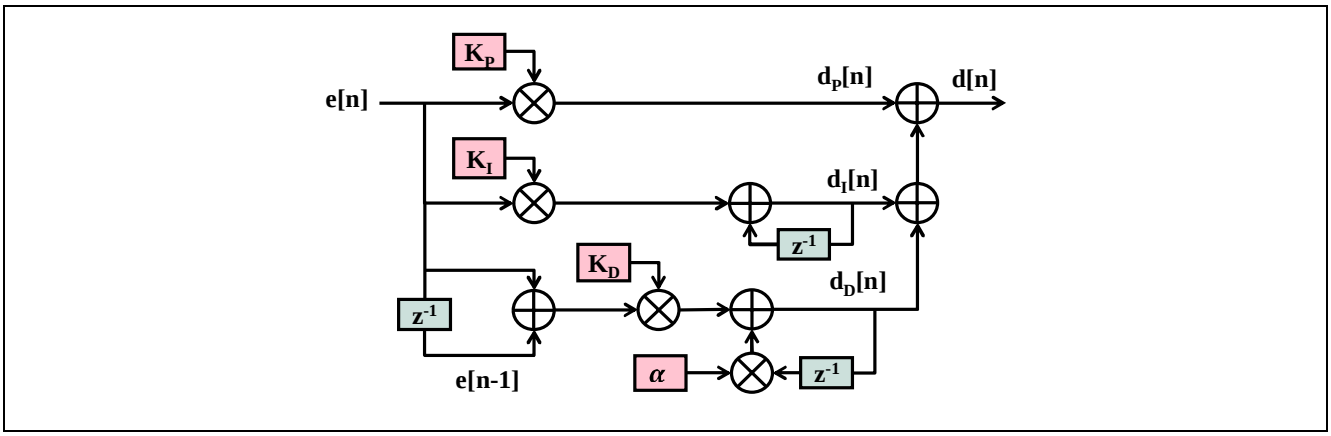

Figure 17

$$Gi(z) = \frac{(KP + KI + KD) * z^2 - (KP * (1 + \alpha) + KI * \alpha + 2 * KD) * z + (KP * \alpha + KD)}{z^2 - (1 + \alpha) * z + \alpha} \quad \text{Eq. 10}$$

By reformatting the transfer function in Eq. 9 to the format of Eq. 10, we obtain the transfer function shown in Eq. 11.

$$Gi(z) = \frac{0.054 * z^2 + 0.017 * z^1 - 0.037}{z^2 - 0.778 * z^1 - 0.222} \quad \text{Eq. 11}$$

By mapping Eq. 11 to the formula in Eq. 10, the PID coefficienct can be derived from Eq.12-15 as shown below.

$$\alpha = -0.222 \quad \text{Eq. 12}$$

$$Kp + KI + KD = 0.054 \quad \text{Eq. 13}$$

$$-(Kp * (1 + \alpha) + KI * \alpha + 2 * KD) = 0.017 \quad \text{Eq. 14}$$

$$Kp * \alpha + KD = -0.037 \quad \text{Eq. 15}$$

⇨ KP= 0.052

KI= 0.028

KD= - 0.026

Step 5 - Software implementation

In software, we have implemented the current controller for Low Line Voltage (<170 VAC) as:

```
void PI_Iq_LoLine_Init(void)
{
	// PI controller initialization values for 170VAC or less
	PI_Current.Kp = 475;	//FP Equivalent .057983398437
	PI_Current.Ki = 230;	//FP Equivalent .028076171875
	PI_Current.Scale_KpKi = 13;

	PI_Current.Ik_limit_min = 0;
	PI_Current.Ik_limit_max = 0x5C3 << 13;

	PI_Current.Uk_limit_min = 0;
	PI_Current.Uk_limit_max = 0x5C3;
```

```
        PI_Current.Uk = 0;              // PI output
        PI_Current.Ik = 0;              // Integral output
}
```

As the XMC™ 1400 implements uses fixed point calculation, it is necessary to use a scaled integer value; ideally using the majority of the 32 bit width for highest precision calculation. In this case the error is multiplied by Kp and Ki, as represented by integer values, and then a shift operation is used to return to the intended output value.

We can translate to the equation below:

DPWM = ((error *Kp) + (error*Ki) + Ik_{t-1}) / $2^{\wedge 13}$

Where Ik_{t-1} is the previous integration value.

For representation in decimal values, the integer values can be divided down to interpret the bilinear transform calculation above:

Kp is 475 / $2^{\wedge 13}$ = 0.0579834

Ki is 230 / $2^{\wedge 13}$ = 0.0280762

It can be seen that these are closely tracking to the designed compensator values. The implementation of the derivative term is not used. The application uses proportional-integral control only, in an effort to decrease execution time.

It will also be noticed in the code above that there are saturation limits. The Ik (integrator) value is calculated and maintained prior to the right shift of 13 bits. Therefore, its limit is left-shifted 13 bit positions outward from the DPWM limit of 1475.

The Uk (compensator output) is managed with the PWM output scale of 1:1, and therefore has a limit of 1475. This represented in hex as 0x5C3.

The code has Kp and Ki values for both low line and high line conditions and the peak input voltage is used to decide which compensator to use. The calculation for the high line follows the same format, and will obtain a similar result to the code implementation.

3.2 Voltage loop

Figure 18 shows the voltage loop / compensator and compares the output voltage feedback to a constant reference. Its output is fed into a multiplier along with 2 other components; the first is an inverse squared root of the RMS input voltage, which is used to normalize the loop gain and make it independent of the line voltage throughout the universal input range. The second component in the multiplier is the instantaneous input voltage; this will shape the current reference to be proportional to the line voltage and achieve a high power factor. The result of the multiplier is a rectified sinusoidal signal that is passed to the current loop reference input.

The main design consideration for the voltage compensator is to have a slow dynamic signal across one half line cycle, so that it does not distort the sine wave fed from the line voltage. If the voltage compensator is dynamic enough during the line cycle, then the current command will be distorted and not match the line feedforward signal, causing higher THDi. Therefore, the voltage controller is designed with a much slower bandwidth compared to the current loop. The typical bandwitch of ~5 Hz can ensure that it will not be dynamic over the 50-60 Hz cycle.

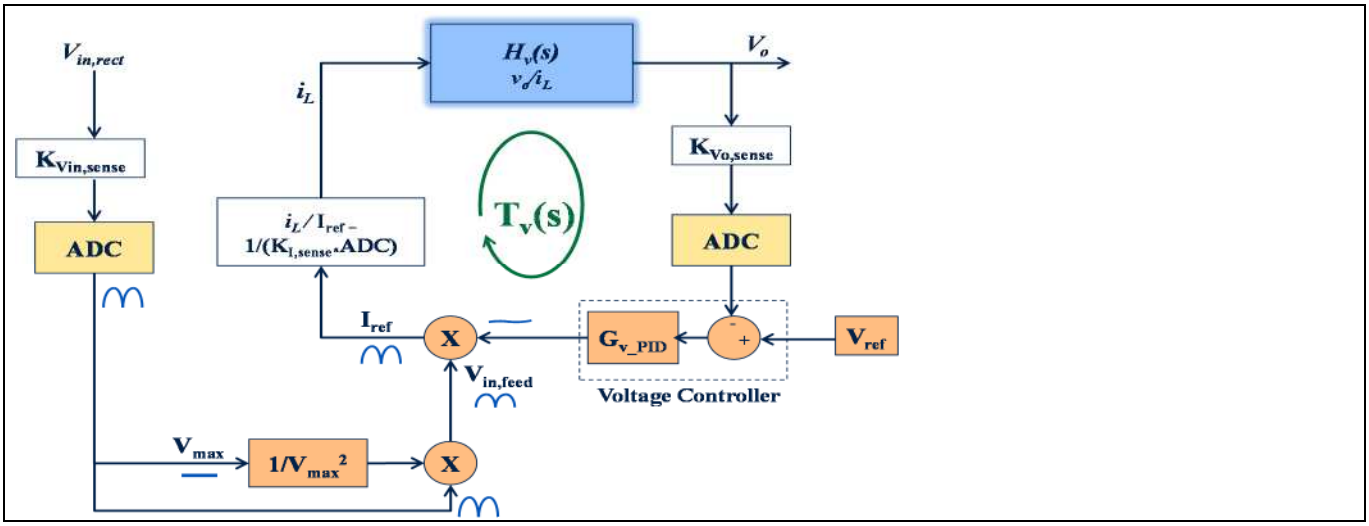

Figure 18 Voltage loop diagram

In order to design the digital current loop, the following steps are required:

Step 1 - Bode plots of the uncompensated loop

From Figure 13 the loop gain $T_v(s)$ is:

$$T_v(s) = H_v(s) * K_{Vo,sense} * ADC * G_v(s) * V_{in,feed} * \frac{i_L}{I_{ref}} \quad \text{Eq. 16}$$

The uncompensated loop gain is:

$$\frac{T_v(s)}{G_v(s)} = H_v(s) * K_{Vo,sense} * ADC * V_{in,feed} * \frac{i_L}{I_{ref}} \quad \text{Eq. 17}$$

Transfer function, inductor current-to- output voltage (Vo/i_L)

$$H_v(s) = \frac{R_o}{s * C_o * R_o + 1} * \frac{V_{in}}{V_o} \quad \text{Eq. 18}$$

Output voltage sense gain

$K_{Vo,sense} = 0.006458$ (resistor divider)

Analog-to-Digital converter gain

$$ADC = \frac{2^n}{Vref} = \frac{2^{12}}{3.3V} \quad \text{Eq. 19}$$

Input voltage feed forward multiplier

$$V_{in,feed} = \frac{V_{in} * K_{Vin,sense} * ADC}{V_{max}^2} = \frac{V_{in} * K_{Vin,sense} * ADC}{(\sqrt{2} * V_{in} * K_{Vin,sense} * ADC)^2} \quad \text{Eq. 20}$$

Inductor current sense gain

$$Ki,sense = Rsense * Kop\_amp = 0.03 * 19$$

Reference current-to- inductor current ($i_L/Iref$) transfer function

$$\frac{i_L}{I_{ref}} = \frac{1}{K_{i,sense} * ADC_i} \quad \text{Eq. 21}$$

Figure 19 shows the MathCAD magnitude and phase bode-plots for the uncompensated voltage loop of Eq. 17

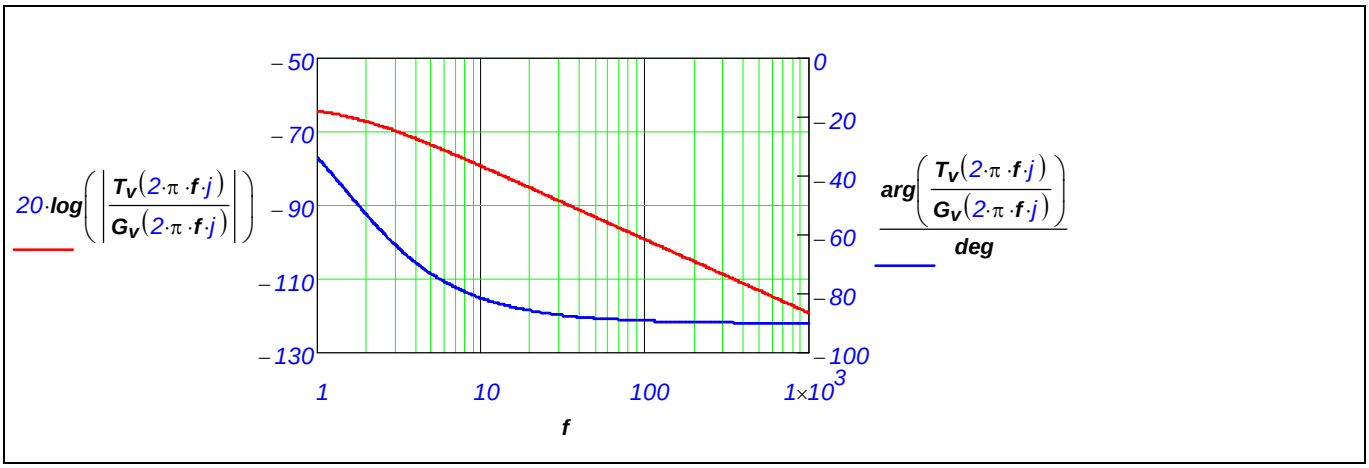

Figure 19 Uncompensated voltage loop bode-plot

Step 2 - Design the current controller, place zeros and poles to reach a desired loop response with adequate crossover frequency and phase margin.

The current controller is a one zero, one pole and an integrator, as in the transfer function below

$$Gi(s) = \mathrm{k} * \frac{\left(\frac{s}{2 * \pi * f_z} + 1\right)}{s * \left(\frac{s}{2 * \pi * f_p} + 1\right)} \quad \text{Eq. 22}$$

Based on the uncompensated bode plots in Figure 19, the compensator was designed with the following zero, pole and gain values. This results a controller bode plot as shown in Figure 20.

$f_z = 5\ Hz$, $f_p = 25\ Hz$, $k = 110000$

Figure 21 shows the compensated current loop with a cross over frequency= 4.875 kHz and gain margin= 44°.

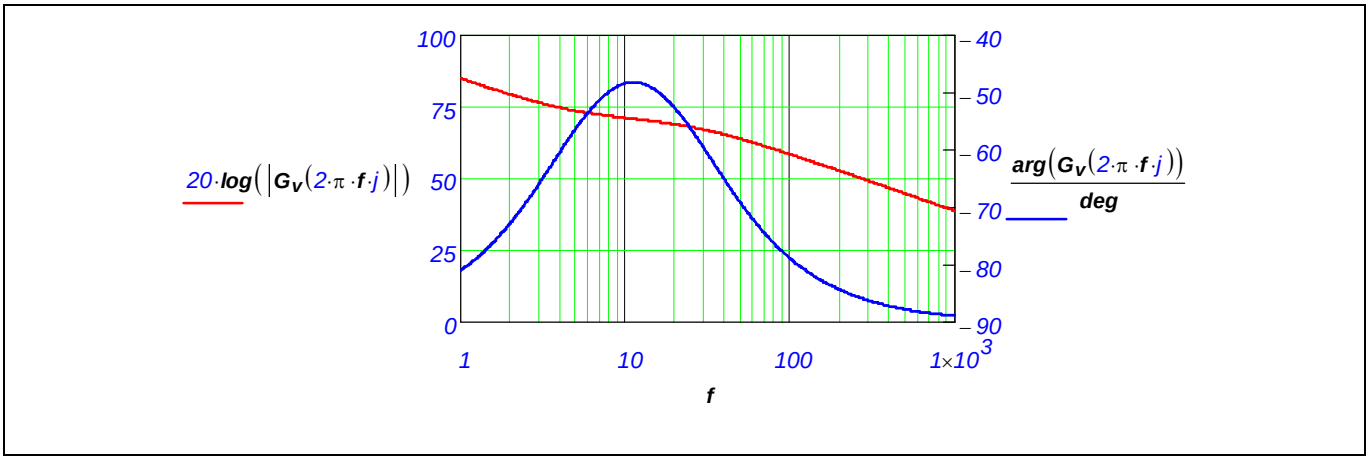

Figure 20 Voltage controller bode plot

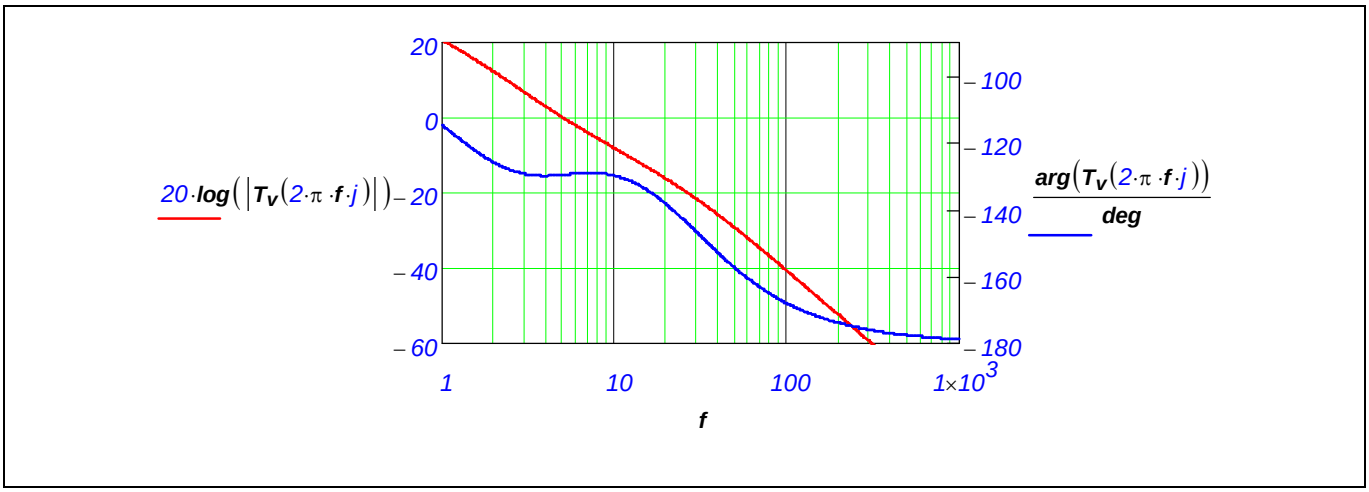

Figure 21 Compensated voltage loop bode-plot

Step 3 - Use the s-to-z bilinear transformation method to digitize the designed s-domain controllers.

Following the same approach described in the current loop section, we obtain the following z transfer function

$$Gi(z) = \frac{1.729 * 10^3 * z^2 + 2.714 * z - 1.727 * 10^3}{126.157 * z^2 - 251.327 * z + 125.17} \quad \text{Eq. 23}$$

Step 4 - Derive the PID coefficients from the z-transfer function.

Following the same approach described in the current loop section, transfer function can be described by Eq. 24.

$$Gi(z) = \frac{0.054 * z^2 + 0.017 * z^1 - 0.037}{z^2 - 0.778 * z^1 - 0.222} \quad \text{Eq. 24}$$

And PID coefficienct can be solved as

⇨ KP= 2798

KI= 5.5

KD= -2790

Step 5 - Software implementation

In software, we have implemented the voltage controller as follows:

```
void PI_Voltage_Init(void)
{
      // PI controller initialization values for Voltage Loop
      PI_Voltage.Kp = 716288;               //FP Equivalent 2798
      PI_Voltage.Ki = 1408;                 //FP Equivalent 5.5
      PI_Voltage.Scale_KpKi = 8;

      PI_Voltage.Ik_limit_min = 0;
      PI_Voltage.Ik_limit_max = 179200000;

      PI_Voltage.Uk_limit_min = 0;
      PI_Voltage.Uk_limit_max = 700000;

      PI_Voltage.Uk = 0;            // PI output
      PI_Voltage.Ik = 0;            // Integral output

}
```

It can be seen that the implementation matches the calculation. We are scaling to the power of 8 this time, due to the values being significantly larger:

i.e., Kp = 716288 / 2^8 = 2798

The voltage controller saturation limits were tested. A higher input voltage causes the voltage controller output to float higher due to the RMS scalar that it is divided by. So to confirm the high limit, it is imperative to test the control at maximum input voltage.

4 Controller features

The controller has several useful features implemented. As this is a microncontroller based design, the user has the ability to modify any given feature to meet requirements. An explanation of the feature and its default state are listed below. This may provide guidance on how to edit the implementation.

4.1 Bulk voltage OK Signal_ Relay control

During soft start timing (4.2) with reduced bulk voltage, the controller opens the relay inline with the boost, so that inrush current is limited by an NTC thermistor. The relay is not turned on until closed loop state is reached.

4.2 Soft start, V_{ref} ramp

If the controller has not reached its steady state voltage output level, and the input voltages have satisfied brown-out requirements, then the controller is in soft start mode. This mode runs the current controller as if it is in steady state, but the voltage controller has an adjusted regulated level. This level is lower than the intended regulated output, and is ramped at a slow rate until steady state output is reached.

4.3 Boost follower mode with adjustable bulk voltage at low line/light load

The bus voltage is is reduced at low line input to reduce MOSFET switching losses. At low line, VAC<170 VAC, the bulk output voltage is lowered to Vo=330 VDC. With a hysteresis of 10 V, at 180 VAC the boost follower mode is exited and the bulk output voltage returns to 390 V.

4.4 Frequency increase at high line for better THDi,

At high line voltage, above 180 VAC, the switching frequency is changed from 65 kHz to 100 kHz. This is beneficial in that there is significant current ripple reduction that extends the time of operation in Continuous Conduction Mode (CCM).

4.5 Optimized compensation values for low and high line conditions, for better THDi

With the low to high AC input range, the current compensator parameters are adjusted for each condition. Two compensators can split the input voltage range to improve power factor and THDi across the entire input range.

5 Protection

5.1 Brown Out Protection (BOP)

If the input voltage is too low, VAC<85 VAC, the controller will turn off the PWM signal. This is important to limit high current in the boost circuit at very low voltage conditions.

5.2 Peak Current Limit (PCL)

The ADC uses a boundary condition managed by vadc_group0_handle.boundary0/1 in ADC_Config.c. If any measured current value exceeds the boundary value, an exclusive boundary ISR is set at high priority. The ISR forces a shut down of the output until reset. The boundary condition is intended to protect against an overcurrent condition on the MOSFET.

5.3 Peak Power Limit (PPL)

The peak current value is limited by the voltage compensation saturation limits. Managing the saturation values will cause the controller to reduce output voltage, thereby limiting the power output.

5.4 Over Voltage Protection (OVP)

The ADC uses a boundary condition managed by vadc_group1_handle.boundary0/1 in ADC_Config.c. If any measured voltage value exceeds the boundary value, an exclusive boundary ISR is set at high priority. The ISR forces a shut down of the output until reset. The boundary condition is intended to protect against an overvoltage condition on the MOSFET. The boundary is intended to represent a 10% overage of the intended max Vbus output. Currently, the condition is undone via a reset.

6 Experimental results

Table 2 compares results of the PFC circuit using the digital control board versus the analog control board.

Table 2

Digital controller XMC1400

	V_{out}	I_o	P_o	V_{in}	I_{in}	P_{in}	E_{ff}	PF	THDi
115 VAC	335.0	0.2	68.7	114.7	0.6	72.4	94.8	97.8	8.1
	335.0	0.4	136.3	114.5	1.4	142.2	95.8	98.6	5.2
	335.0	0.6	200.3	114.2	1.9	208.4	96.1	99.3	3.6
	335.0	0.8	272.6	114.0	2.5	283.7	96.1	99.6	2.7
230 VAC	385.0	0.2	77.5	230.2	0.4	84.6	91.6	90.7	12.1
	385.0	0.4	152.2	229.9	0.7	158.4	96.1	96.6	9.0
	385.0	0.6	232.2	229.7	1.0	238.4	97.3	98.1	5.5
	385.0	0.8	304.1	229.5	1.4	312.1	97.4	99.0	4.7

Analog controller ICE3PCS01G

	V_{out}	I_o	P_o	V_{in}	I_{in}	P_{in}	E_{ff}	PF	THDi
115 VAC	341.3	0.2	68.3	114.7	0.6	71.9	94.9	98.7	4.1
	341.3	0.4	136.6	114.3	1.2	142.4	95.9	97.5	6.7
	341.3	0.6	204.9	114.0	1.9	213.2	96.1	98.8	6.0
	341.3	0.8	273.1	113.6	2.5	284.5	96.0	99.2	5.5
230 VAC	383.5	0.2	76.7	230.2	0.4	85.2	90.1	89.0	18.3
	383.5	0.4	153.4	229.9	0.7	160.2	95.7	97.5	9.9
	383.5	0.6	230.1	229.7	1.0	237.2	97.0	98.8	5.2
	383.5	0.8	306.8	229.5	1.4	314.6	97.5	99.2	4.7

PFC waveforms at 115 VAC line voltage, at light and heavy loads

- Green – Inductor current, ADC
- Blue – Output voltage
- Yellow – Input Voltage
- Pink – Inductor Current

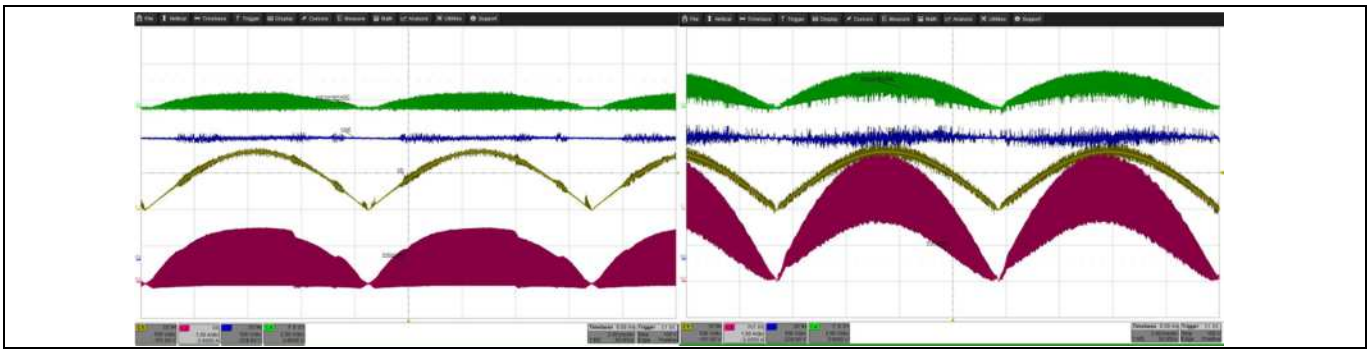

Figure 22 PFC waveforms at 115 VAC input voltage: light load (left), heavy load (right)

PFC waveforms at 230 VAC line voltage

- Green – Inductor current, ADC
- Blue – Output voltage
- Yellow – Input Voltage
- Pink – Inductor Current

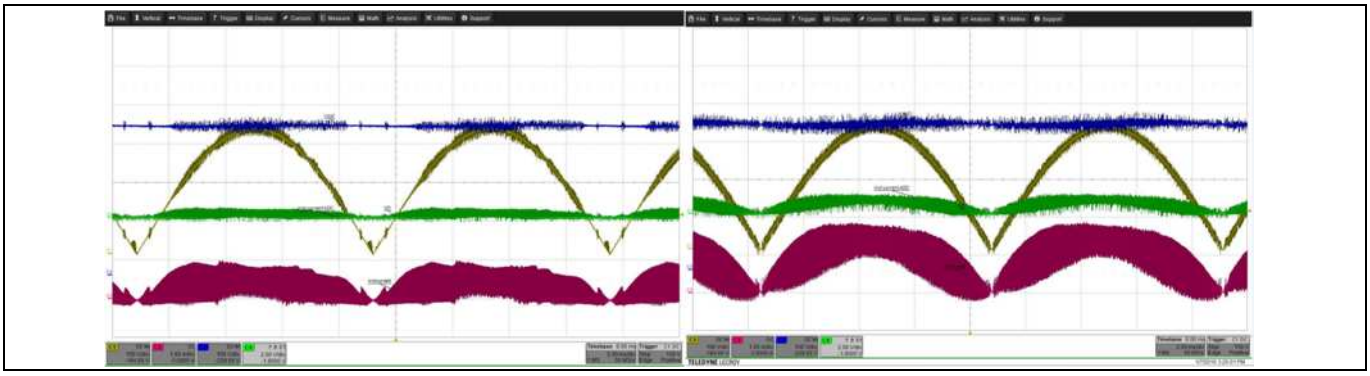

Figure 23 PFC waveforms at 230 VAC Input voltage: light load (left), heavy load (right)

Startup waveforms at 115 V_{AC} and 230 V_{AC}

- Blue – Output voltage
- Yellow – Input voltage
- Pink – Inductor current

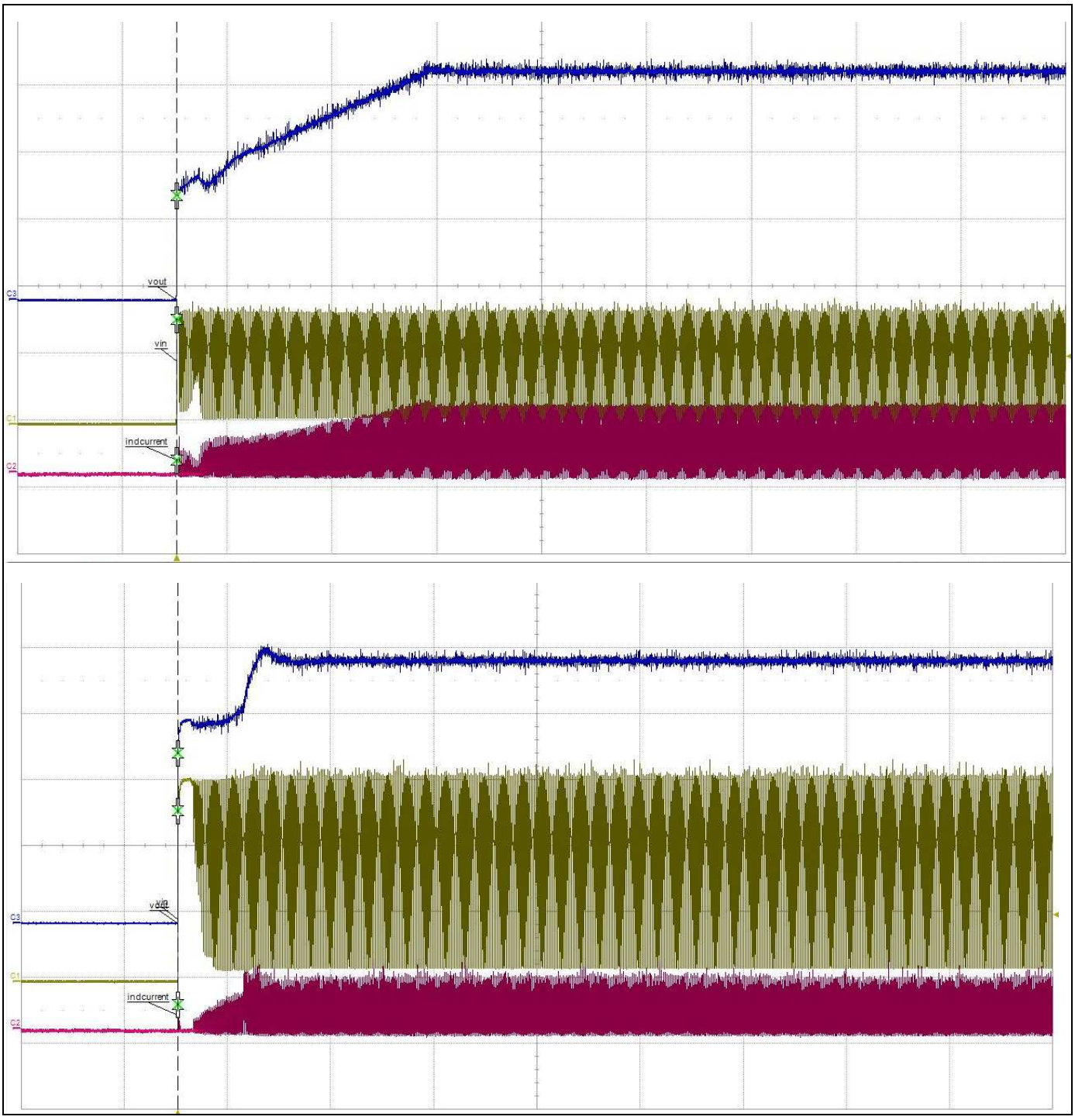

Figure 24 Startup waveforms: 115 V_{AC} (top), 230 V_{AC} (bottom)

7 Summary

In this document, PFC control structures of analog versus digital were discussed, with highligting the benefits of XMC™ microcontroller, followed by a design example of a 300 W PFC implementation with XMC1400, with detailed explanantion of timers, ACDs and peripherals. Also it includes a systematics design procedure for the digital current/voltage loops, showing the steps for s-to-z bilinear digital transformation and PID parameters calculation, and then showing the software implementation of the loops.

Analog PFC controller has some performance limitation, while the felexibility of the XMC™ digital control can enhance power factor correction. Digital XMC™ power can adapt the converter operation and control parameters to the different line and load condition to improve THDi, as was illustrated in the experimental THDi data for the XMC™ controller compared to the analaog controller. In addition to the smaller low frequency output ripple and faster output dynamic during load jumps for better regulation.

8 References

[1] Power Factor Correction (PFC) with XMC™.

https://www.infineon.com/dgdl/Infineon-Application_Power_Conversion_Power_Factor_Correction_(PFC)_with_XMC-TR-v01_00-EN.pdf?fileId=5546d4625696ed7601569c7cf64607ab

[2] Infineon Technologies Application Note: " 800W 130kHz PFC with 600V CoolMOS™ C7"September 2016.

http://www.infineon.com/dgdl/Infineon-ApplicationNote_Demoboard_PFC_System_Solution-AN-v01_00-EN.pdf?fileId=5546d4624cb7f111014d6b42c932713c

[3] Alois Steiner; Franz Stuckler; Florian Zechner , "EVALPFC-300W-IPP60R190P6 300W PFC Evaluation Board with CCM PFC Controller," Application note, **www.infineon.com**

[4] L. Rossetto; G. Spiazzi; P. Tenti, "Control Techniques for Power Factor Correction Converters," Department of Electrical Engineering, University of Padova, Italy.

[5] S. Abdel-Rahman; F. Stueckler; K. Siu, "PFC Boost Converter Design Guide" Infineon Technologies Application Note, November 2014.

Edition 2016-11-23
Published by
Infineon Technologies AG
81726 Munich, Germany

Do you have a question about this document?
Email: erratum@infineon.com

Document reference
AppNote Number