

ADC Sampling Information **ADM1275/ADM1276/ADM1075**

by Paul O'Sullivan

INTRODUCTION

The [ADM1275](#), [ADM1276](#), and [ADM1075](#) all share the same basic analog-to-digital converter (ADC) core and PMBus interface. There are some subtle differences between the devices with regard to averaging calculations and ADC register updates. There are also factors and limitations to consider when reading data quickly from any of the [ADM1275](#), [ADM1276](#), or [ADM1075](#) devices. This document describes the ADC operation for each of these devices and describes how to maximize the data rate from each device, if required.

ADC CONVERSIONS

The [ADM1275](#) features current and voltage readback via the PMBus interface. Figure 1 shows the sampling order. The current is sampled first, followed by a current register update. Then, the voltage is sampled followed by a voltage register update. The register update completes in one clock cycle. On the [ADM1275](#), the user can select whether to monitor the VIN voltage or the VOUT voltage but cannot monitor both at the same time.

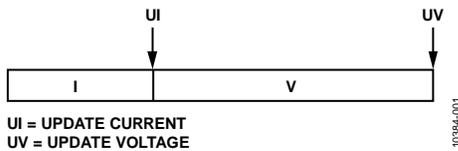


Figure 1. [ADM1275](#) ADC Updates, Single Sample

Figure 2 shows the current and voltage conversions and updates in the case where averaging is enabled on the device. The current register updates after the final current sample; the voltage register updates after the final voltage sample. Again, the voltage conversion corresponds to VIN or VOUT.



Figure 2. [ADM1275](#) ADC Updates, Device Averaging

The [ADM1276](#) and [ADM1075](#) both have power metering as well as voltage and current monitoring. The power is calculated as $VIN \times I$; therefore, the VIN voltage is always sampled. The user can opt to sample another ADC voltage input, as well. In the case of the [ADM1276](#), the VOUT voltage can be measured. After the power calculation is complete, all of the voltage, current, power, and energy registers update. See Figure 3 for details.

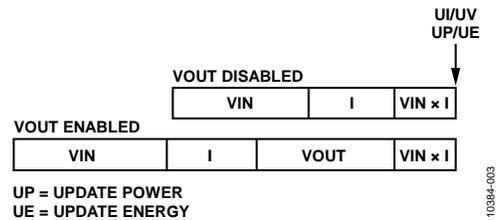


Figure 3. [ADM1276](#) ADC Updates, Single Sample

As shown in Figure 4, when averaging is enabled, a single power and energy calculation is completed following the last voltage and current samples. After these calculations complete, all of the registers update.

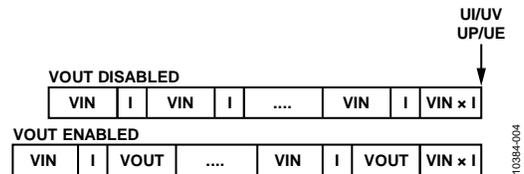


Figure 4. [ADM1276](#) ADC Updates, Device Averaging

The [ADM1075](#) is very similar to the [ADM1276](#). The order of current and voltage conversions is the same. The main difference between them is that the VOUT measurement on the [ADM1276](#) is a VAUX measurement on the [ADM1075](#), as shown in Figure 5 and Figure 6.

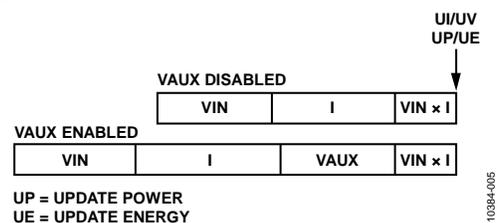


Figure 5. [ADM1075](#) ADC Updates, Single Sample

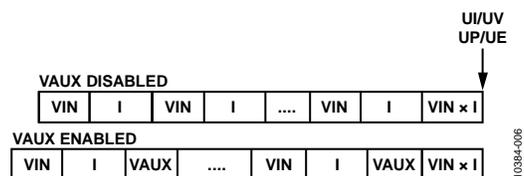


Figure 6. [ADM1075](#) ADC Updates, Device Averaging

Figure 7 shows how the power and energy calculations are performed.

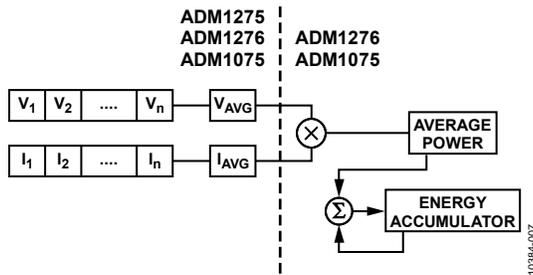


Figure 7. Power and Energy Calculations

The average power or accumulated energy can be read directly from the device. To perform an energy metering function, an external microprocessor is required for a real-time clock. The average power used over a period of time can be calculated from the delta in energy (ΔE) between the register readbacks divided by the delta in samples (between the register readbacks). The total energy can then be calculated by multiplying this result with the delta in time (Δt) from the time stamps of each readback.

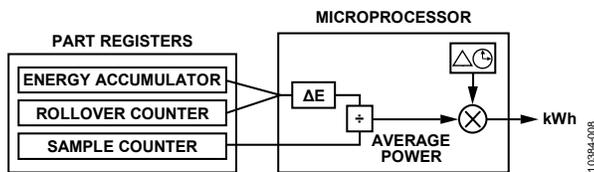


Figure 8. Calculating Average Power in Time

CONVERSIONS AND TIMING

In the majority of cases where device averaging is enabled, the ADC timing is not a major concern to the user. However, in the case where the user wants to read back all of the ADC samples and postprocess the data externally, care needs to be taken to ensure all ADC samples are captured.

The first example is the case of the [ADM1275](#) wherein the user reads back all of the current samples. The total time for a voltage and current conversion is 305 μs maximum (see Table 1).

Table 1. Maximum Conversion Times (μs)

	ADM1275	ADM1276	ADM1075
IOUT	115	VIN	146
VIN or VOUT	190	IOUT	118
		VOUT	146
		PIN	16
Total	305	Total	426
(1× averaging)		(1× averaging)	
		VIN	82
		IOUT	120
		VAUX	82
		PIN	17
		Total	301
		(1× averaging)	

Therefore, each I²C/PMBus readback command must be spaced at least 305 μs apart. If the current is being read back, a minimum

of 115 μs bus free time is needed after the readback command is completed to allow for the current conversion and register update to complete. The time required to update the registers is very small (a single clock period).

The ADC is locked out from voltage and current updates while the I²C bus is active (or more accurately, if the device address is recognized, then lockout occurs between the acknowledge at the end of the command code and the stop bit). Therefore, the bus free time includes the time from the end of the last PMBus command until the command code is recognized in the next PMBus command.

In the case of a voltage readback, a minimum bus free time of 115 μs guarantees no duplicate samples. The voltage readback is one sample offset unless there is a bus free time of at least 190 μs to allow the voltage conversion to occur before the next readback.

These values determine the maximum duration of the PMBus command and, therefore, determine the minimum speed at which the I²C needs to operate for optimum sampling. For example, in the case of the [ADM1275](#), when the I²C is optimized for current readback, set the time between readbacks to 305 μs (see Figure 9). That means that the portion of the READ_IOUT command after the command code has been recognized must be less than 190 μs .

The read current command is three or four bytes of data following the command code, depending on whether the packet error checking (PEC) byte is used (see Figure 10). Assuming the PEC byte is used, and including the acknowledge and stop bits, there are a total of 40 bits of data after the command code. Consequently, the I²C needs to operate at >210 kHz (40 bits in 190 μs) to ensure that the remainder of the current readback command is completed within 190 μs .

In the case where averaging is enabled, set the bus free time to greater than the voltage conversion time. This setting ensures that both the current and the voltage registers are updated before the next readback command (see Figure 2). Set the time between readbacks to greater than

$$\text{Number of Averages} \times 305 \mu\text{s}$$

Use the CONV function of the GPIO1/ALERT1/CONV pin to synchronize ADC samples to ensure that the sampling always occurs at the desired time. This procedure allows for other I²C activity after the readback command and before the next ADC sampling sequence initiates. CONV functionality is not available on the [ADM1276](#).

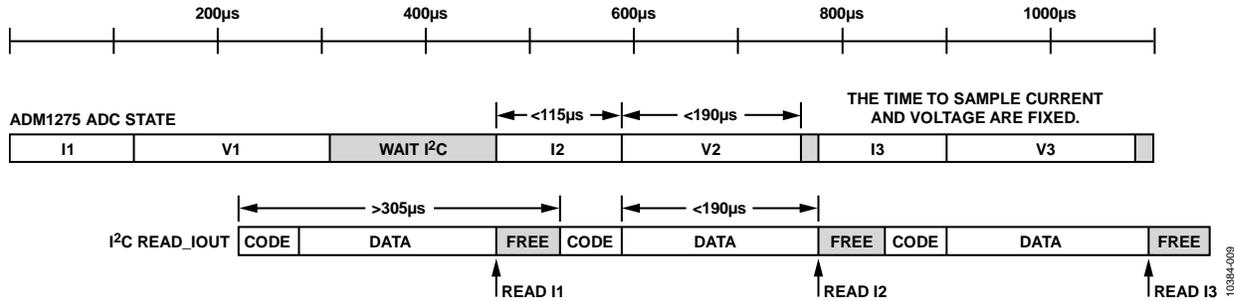


Figure 9. Optimal Current Readback on ADM1275

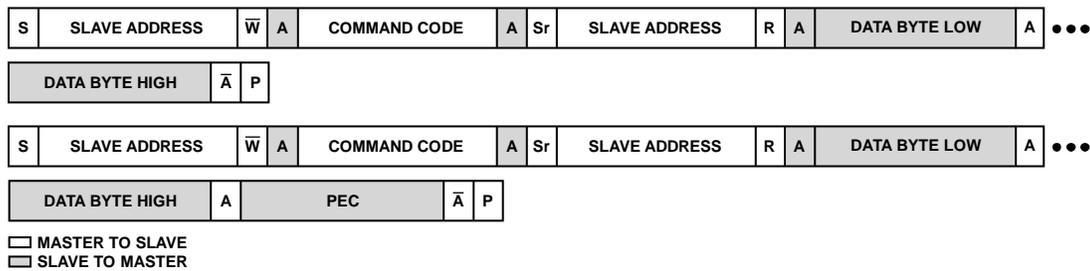


Figure 10. Read Word and Read Word with PEC

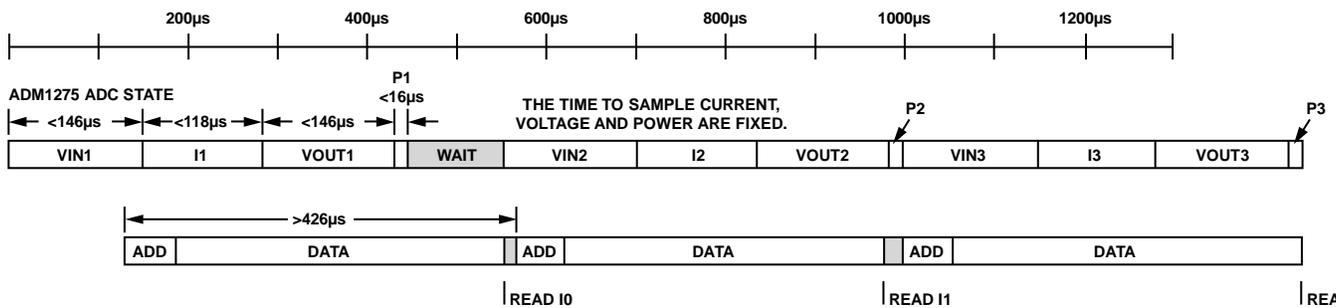


Figure 11. Register Readback on ADM1276

In the case of the [ADM1276](#) and [ADM1075](#), the update of all registers occurs at the end of the sampling sequence and after the power calculation has been completed. Hence, the time taken by the read command is not important; rather, the critical time is the time between commands. The time between commands must be longer than the entire sampling to avoid duplicate samples (see Figure 11).

Similarly, in the case where averaging is enabled, all register updates are done at the end of the sampling sequence after the power calculation. Therefore, set the time between readbacks to greater than the sampling time. This time can be calculated from Table 1 using the following formula:

$$\text{Min Readback Time} = [\text{Number Averages} \times (\text{Voltage Sample} + \text{Current Sample} + \text{Optional Voltage Sample})] + \text{Power Calculation Time}$$

CONCLUSION

The [ADM1275](#), [ADM1276](#), and [ADM1075](#) provide different combinations of voltage, current, and power monitoring. There are subtle differences between the order of sampling, where ADC register updates occur, and the ADC timing on each device. Consider these differences to optimize the PMBus transactions for maximum data rate and to avoid duplicate samples.

The minimum I²C data rate and minimum bus free time (for the [ADM1275](#) only) are the critical specifications when optimizing data readback, for example, where postprocessing of data is con-

ducted externally to the device (that is, in a microprocessor). In a case such as this, the user needs to set single average sampling and must readback every sample from the device. In most cases where device averaging is used, this level of readback optimization is not required.

When timing is critical, the CONV function of the GPIO1/ALERT1/CONV pin helps to synchronize the sampling. On the [ADM1275](#), where some bus free time is required for optimal readback, use the CONV pin function to delay ADC sampling to allow for other I²C activity following the initial readback command.