VA108x0 Power Management Application Note

March 20, 2017 Version 1.0

VA10800/VA10820

Abstract

The VA108x0 family is designed in a low power CMOS process. To fully make use of the low power capability of the device, a designer should make use of clock gating, smart pin configuration techniques and efficient software strategies. This application note provides tactics for reducing power consumption and shows bench measurements to confirm the techniques work.

Table of contents

1 MCU power consumption sources

The VA108x0 family of MCUs have two supply rails. The 1.5V supply (VDD15) powers the digital logic of the chip and the SRAM memory. The digital logic uses minimum geometry transistors for fast switching and lower power. The 3.3V supply (VDD33) powers the input and output circuits for the pins. The GPIO circuits use larger transistors that have a higher breakdown voltage and can supply higher currents. Characteristics of the two power domains are described separately in the next two sections.

1.1 Small geometry logic in 1.5V domain

CMOS logic gates consume very little current when in a static state at room temperature. Below approximately 50°C, gate-induced-drain-leakage (GIDL) is the predominant current contributor for a static transistor and is not temperature sensitive. For the VA10800, the static current draw at 25°C with 1.5 V applied is approximately 100 µA. The VA10820, with extra circuitry and memory to prevent single event upsets (SEUs), has a slightly larger static IDD of approximately 140 µA.

As the temperature increases above 50°C, subthreshold transistor leakage from drain to source becomes the dominant static contributor. For the 130 nm CMOS process used, the leakage will approximately double for every 25°C increase.

When the device is operating, most of the current draw comes when a logic gate switches state which requires transistor parasitic capacitors to be charged or discharged. Using i = C dv/dt and applying it to a periodic square wave, the equation for DC current is I = C * V * f. The smaller the number of transitions per unit time, the smaller the current draw will be.

Transitions can be controlled by:

- Clock speed
- Peripheral and timer activity
- Specific operations of code being executed

1.2 Larger geometry input / output logic in 3.3V domain

Larger transistors with high voltage thresholds (Vt) are used to handle the higher voltage and current of the I/O pins. These transistors are less prone to damage from voltage transients than the smaller devices and do not leak as much at higher temperatures. For the 130 nm CMOS process used, the VDD33 leakage increase approximately 25% from 25°C to 200°C with all GPIO pins set as input and driven low externally.

General purpose input/output (I/O) pins have input stages that can detect whether an incoming signal should be interpreted as a logic one or zero. When an input is close to VDD or VSS, the input stage pulls very little current. However, if an input is held around the mid-point between VDD and VSS, a relatively large amount of current can be realized, on the order of hundreds of μA per pin. Hence it is best to have digital inputs to the MCU transition swiftly through the switching point which is near VDD33/2.

2 Power saving techniques

The following three sections contain information on how to reduce current consumption.

2.1 Clock gating

To prevent unused peripherals from drawing valuable current, the VA108x0 implements a series of switches to turn-off the system clock to circuits not being used. Each peripheral and each timer has a separate clock gate enable bit. Out of RESET, all the peripheral and timers clock gates are off with the CPU is running. Before any registers can be configured for a peripheral, the clock gating for that block must be enabled.

Clock gating can be enabled or disabled while a program is running. For instance, it is possible to only clock an SPI peripheral or timer when it is active. This requires the module to be completely reconfigured each time it is used but can save a substantial amount of power.

Tip #1 – Only enable clocks to peripherals and timers you will be using.

2.2 Port pin configuration

If the gate of a CMOS transistor is left floating, the device has the potential to oscillate on and off or stay in an active region that can pull lots of current. At room temperature, as much as 35 mA of I/O current has been measured when all 52 GPIO pins were left floating. Oscillations can cause electrical noise which can corrupt analog signals and exacerbate switching problems with slow changing input signals.

Input pins and unused pins which may encounter a floating condition may benefit from termination. This can be done several different ways.

Method	Advantage	Disadvantage
Use an external pull-up or pull-down resistor	Always safe and immediately effective. Produces lowest boot current scenario.	Extra cost for resistor components. Can cause excess current when opposing input signal.
Enable the internal pull-up or pull-down resistor (Nominally 33 kΩ) once the device starts running code	No hardware considerations required and no cost.	Does not help while part is in RESET and going through boot sequence. Resistors are not tightly controlled over temperature
Set pin as output and drive low	No hardware considerations required and no cost.	Does not help while part is in RESET and going through boot sequence
Tie unused pins to VDD or GND	No hardware required	Can damage pin if errant code sets pin to output with opposite polarity.

The use of either internal or external pull-up/pull-down resistor will cause additional current draw when the signal is driven to the opposing value. Consider the case of a switch input that is normally closed and tying the pin to ground. If a 10 kΩ resistor was used as a pull-up, the pin would be pulling 0.33 mA most of the time. For that case, it might be better to use an internal pull-up resistor that is enabled only when polling the input after it was known to be low.

Variations of these four termination methods can be combined depending upon how long the device will be held in reset, boot time current limitations, routing restrictions and board space.

Tip #2 – Terminate all 56 GPIO, JTAG pins, and I2C pins.

2.2.1 Calculating I/O current

The DC current (IDD) of a switched output in a purely capacitive load can be estimated as:

IDD (switched digital output) ~= frequency * voltage * capacitance.

For instance, if a SPI clock was running continually at 10 MHz to 5 peripherals each with a 10 pF input impedance, the resultant average current would be: 10e6 Hz *50e-12 F * 3.3V = 1.5 mA. Section 3.4 shows some data for various output pin activity and loading.
DC loads such as LEDs can be calculated using

IDD = (V / R) * % time on.

Using a spreadsheet and the above two formulas, it is possible to accurately calculate the IDD for the VDD33 supply rail.

2.3 Sleep mode - Cortex-M WFI instruction

The Cortex-M family of cores provides an instruction, Wait-For-Interrupt (WFI), that stops the CPU from executing instructions until one of three events occur: 1) an interrupt is received, 2) the device is reset or 3) a debug operation occurs. This can save a lot of current draw and should be used whenever power consumption is a concern. Interrupt sources can be communication peripherals, timers or port pins.

Implementing this instruction is quite simple by inserting, "__WFI" (two underscore characters followed by WFI). All registers will remain intact while the CPU is in sleep mode so there is no need to run any recovery routines.

Note: Measuring IDD when a debugger is attached will give bogus results not matching end application numbers. The debug activity prevents the CPU from entering sleep mode and the activity on the I/O pins can cause extra current draw. If a debugger is attached but not powered via its USB cable, the MCU will try to power the debugger which can be an extra 5 -10 mA on VDD33 depending upon series resistance.

Tip #3 – Make use of the WFI instruction to idle the CPU when it is not required.

3 Bench level data collection

The following sections present data collected on several devices at various operating conditions, temperatures and voltages. These values are indicative of what you should see but should not be taken as minimum or maximum values.

3.1 1.5 V domain across frequency

Device activity – 3 cases

1. CPU running, all peripherals and all timers off
2. CPU running, all peripherals and all timers on
3. CPU in sleep mode, all peripherals and all timers off

Figure 1 - Plot of IDD1.5 vs Frequency for VA10800

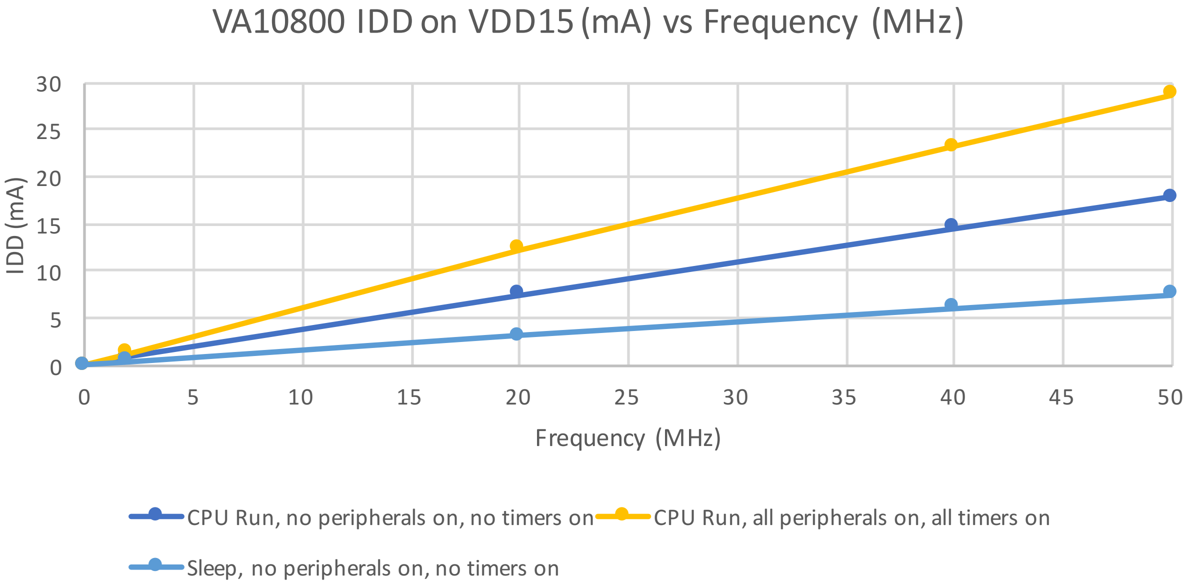


Table 1 - Table of data for IDD versus frequency plot shown above

	VDD = 1.5V					VDD = 1.65 V				
Conditions	Frequency in MHz					Frequency in MHz				
	0	2	20	40	50	0	2	20	40	50
CPU Run, no peripherals on, no timers on	0.076	0.84	7.54	14.5	17.8	0.102	0.95	8.3	15.97	19.59
CPU Run, all peripherals on, all timers on	0.076	1.36	12.26	23.15	28.7	0.102	1.528	13.57	25.5	30.98
Sleep, no peripherals on, no timers on	0.076	0.408	3.1	6.03	7.46	0.102	0.443	3.46	6.71	8.29

Figure 2 - Plot of IDD1.5 vs Frequency for VA10820

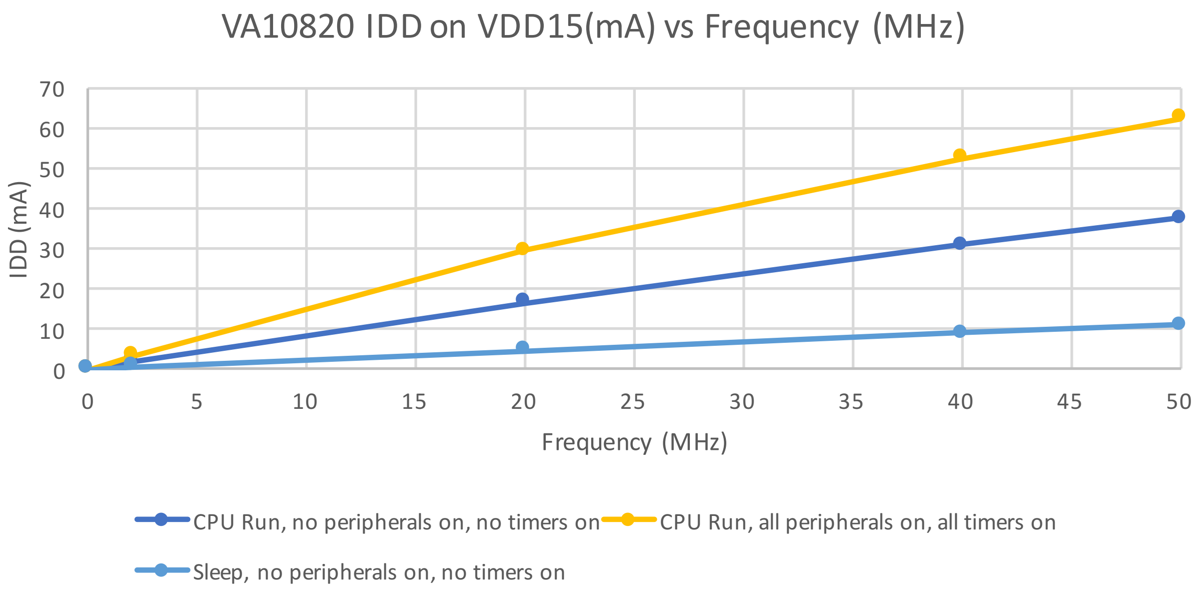


Table 2 - Data for VA10820 IDD vs Frequency plot above

	VDD = 1.5V				
Conditions	Frequency in MHz				
	0	2	20	40	50
CPU Run, no peripherals on, no timers on	0.143	1.886	16.55	30.94	37.49
CPU Run, all peripherals on, all timers on	0.143	3.4	29.39	52.5	62.35
Sleep, no peripherals on, no timers on	0.143	0.599	4.62	8.95	11.04

3.2 1.5 V domain with and without peripheral and timers

Device activity –Matrix of conditions

All timers on or off

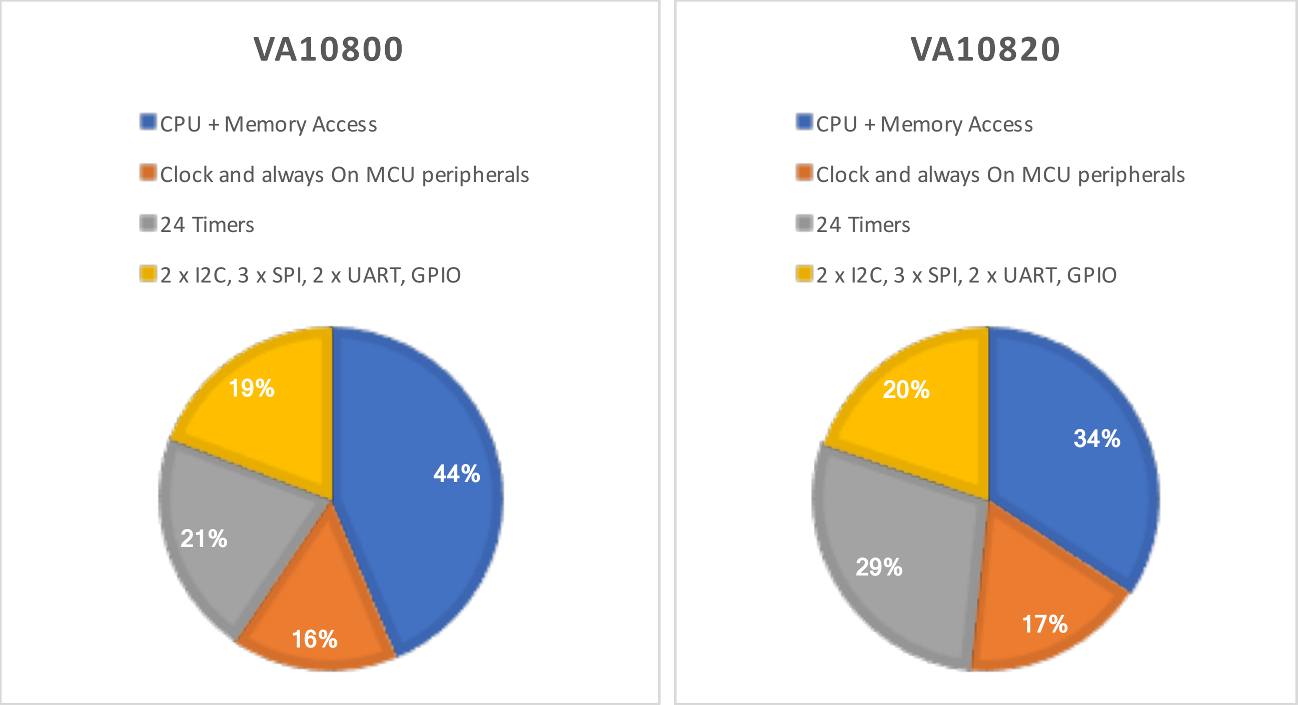
All Peripherals on or off

CPU either in wait mode or running

Table 3 - Tables of IDD on VDD15 for both devices with frequency = 50 MHz.

IDD on VDD15 for Various operating conditions				
Frequency = 50 MHz, VDD15 = 1.5V				
CPU state	Timers (24)	Peripherals (2xI2C, 2xUART, 3xSPI)	VA10800	VA10820
			(IDD15 in mA)	
Run	off	off	17.0	33.8
Run	on	off	22.6	49.7
Run	off	on	22.5	45.8
Run	on	on	27.8	59.9
Sleep	off	off	3.9	11.2
Sleep	on	off	10.0	30.1
Sleep	off	on	9.4	24.4
Sleep	on	on	15.3	41.4

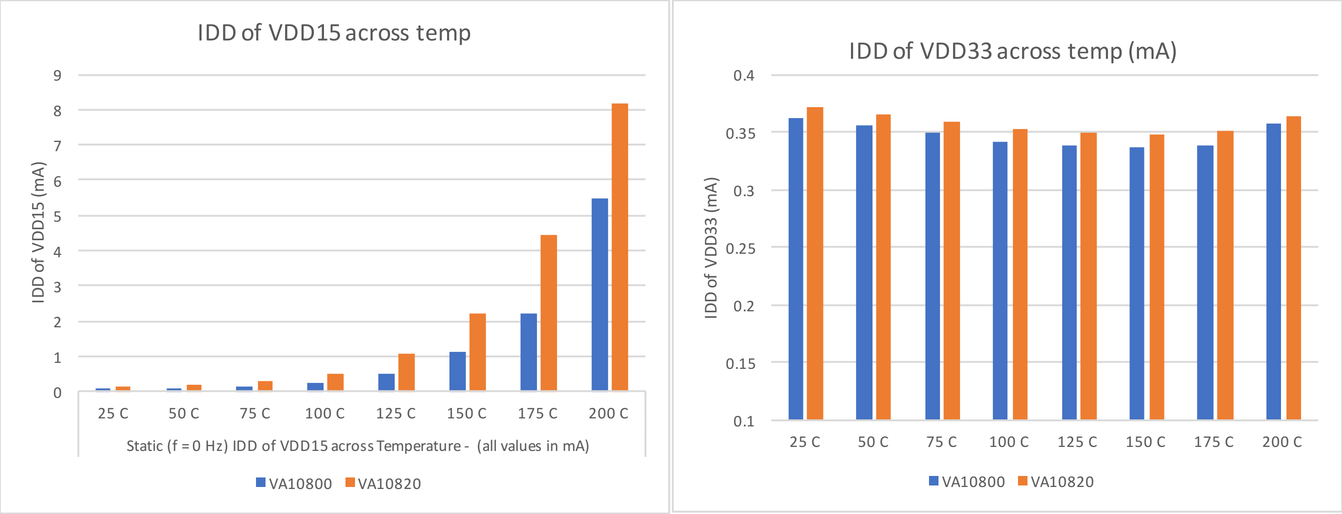
Figure 3 - Pie charts of current consumption as percentage of total



3.3 1.5V & 3.3V Domain, frequency = 0 MHz across temperature

Graphs showing IDD on VDD15 and VDD33 with Clock = 0 MHz across temperature for both devices.

Figure 4 - Chart of IDD on VDD15 and VDD33 over temperature



	Static (f = 0 Hz) IDD of VDD15 across Temperature - (all values in mA)							
	25 C	50 C	75 C	100 C	125 C	150 C	175 C	200 C
VA10800	0.0765	0.0875	0.123	0.251	0.492	1.121	2.221	5.498
VA10820	0.142	0.175	0.275	0.512	1.05	2.222	4.43	8.17
Static (f = 0 Hz) IDD of VDD33 across Temperature - (all values in mA)								
All GPIO pulled to ground with internal resistor, JTAG inputs and I2C have external pull-up resistors								
	25 C	50 C	75 C	100 C	125 C	150 C	175 C	200 C
VA10800	0.362	0.355	0.349	0.342	0.338	0.336	0.338	0.358
VA10820	0.372	0.365	0.359	0.353	0.35	0.348	0.351	0.364

3.4 3.3V Domain high frequency pin activity

Device activity – Matrix of conditions

- PWM loads: 0 / 1 / 2 / 4 PWMs (load = .01 μF)
- PWM frequency: 10 / 100 kHz
- Supply voltage: 3.0, 3.3, 3.6
- All other port pins terminated with internal pull-down enabled.

Table for VA10800 with VDD = 3.0, 3.3 and 3.6 V. (Note: Very similar data taken for VA10820)

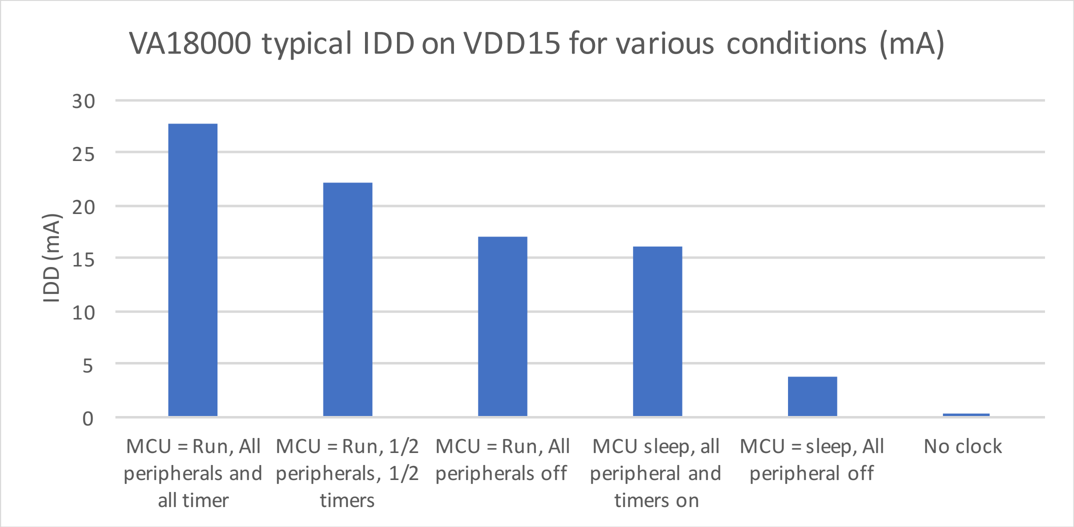
Table 4 - IDD on VDD33 for various output activity

IDD of VDD33 (mA)				
		VDD33 =		
PWM f (kHz)	# of loads	3	3.3	3.6
10	0	1.25	1.65	2.17
10	1	1.7	2.14	2.72
10	2	2.13	2.66	3.24
10	4	3	3.62	4.29
100	0	1.25	1.65	2.17
100	1	4.37	5.11	5.93
100	2	7.29	8.26	9.46
100	4	13	14.7	16.4

4 Conclusions

This application note has described the various source of power consumption in the VA108x0 devices. Data taken from bench evaluation has been presented to support the explanation.

Figure 5 - IDD on VDD15 for VA10800 with different activity



A system designer can use this information to effectively manage the current draw on both the 1.5V and 3.3V voltage rails.

5 Other Resources

VORAGO VA108x0 programmers guide:
http://www.voragotech.com/sites/default/files/VA10800_VA10820_PG_July2016revision1.16%5B4%5D.pdf

VORAGO MCU products: http://www.voragotech.com/VORAGO-products

VORAGO Application notes: http://www.voragotech.com/resources

VORAGO VA108xx REB1board user guide: Part of Board support package
http://www.voragotech.com/products/reb1