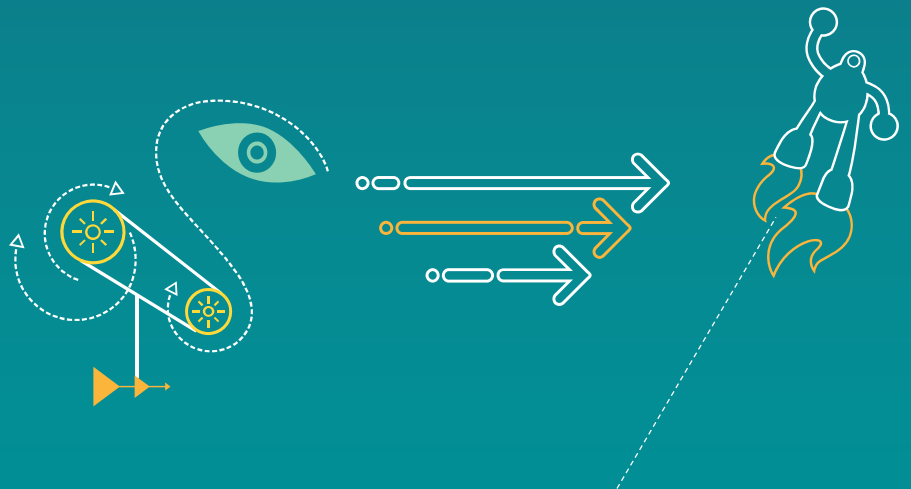
Application Note

Adaptive Voltage Scaling (AVS)

Qualcomm Technologies, Inc.

80-N8715-14 Rev. G



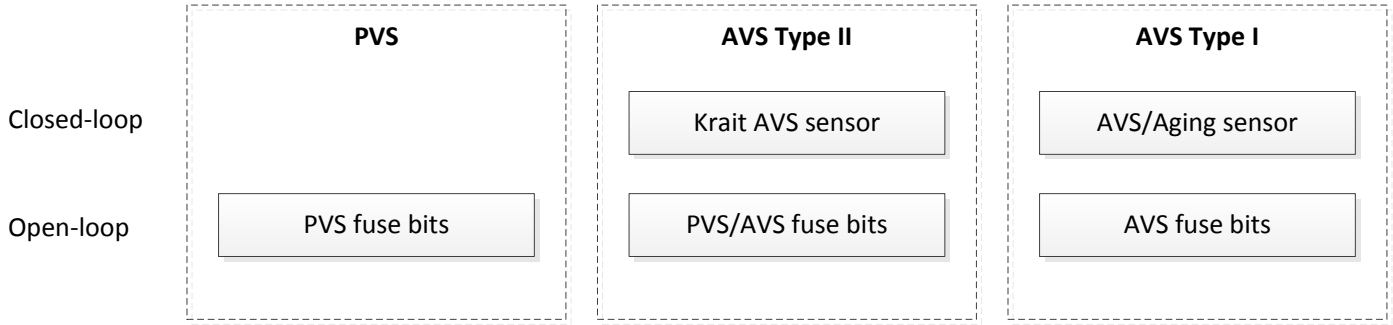
Qualcomm Technologies, Inc.
5775 Morehouse Drive
San Diego, CA 92121
U.S.A.

Revision History

Revision	Date	Description
A	March 2013	Initial release
B	August 2013	Slide 6: Added last bullet clarifying AVS support Slide 7: • Added MSM8974AB, MSM8x26, and MSM8926 rows and table note 1 • Added note about pre-CS devices • Updated information about device support for PVS and AVS
C	April 2014	Slide 7: Added MSM8916, MSM8936/MSM8939, MSM8x10/MSM8x12, APQ8084, MDM9x30/MDM9x35M and note 2
D	September 2014	Slide 4: Added processor information Slide 7: • Added MSM8994, APQ8074AB, and APQ8026 information • Replaced notes Slide 8: Replaced RPM FW with processor Slide 11: • Removed the AVS Type II example • Added information about disabling/enabling AVS in production devices
E	December 2014	Slide 7: Updated MSM8994, add MSM8992, MSM8952 and MSM8909 Slide 4 and 11: Made minor revisions Slide 12: Added appendix to explain PVS
F	September 2015	Slide 7: Updated MSM8996 and MSM8956/8976 Slide 9: Added block diagram of Hardware-based AVS Type I
G	March 2016	Slides 4 through 8, and 11: Added aging compensation related information Slide 10: Moved some chipsets that support aging compensation to slide 11

Introduction – What is AVS (1 of 2)

- Adaptive voltage scaling (AVS) is a closed-loop mechanism that extends the benefits of the open-loop process voltage scaling (PVS) technology (see the Appendix). There are two types of AVS: Type I and Type II.
- With **PVS**, device operating voltages are defined based on the device characteristics determined during production testing, see the Appendix for more information about PVS.
- **AVS Type II** includes both closed-loop and open-loop techniques. It works in conjunction with PVS where the initial device operating voltages are defined by the PVS/AVS fuse bits. However, the operating voltages are then dynamically adjusted by a Qualcomm® Krait™ AVS controller to the PMIC, based on an internal AVS sensor (see slide 9).
- **AVS Type I (CPR)** is similar to AVS Type II; it sets the initial operating voltages by AVS fuse bits and then dynamically adjusts the operating voltages by AVS controller via RPM/processor or hardware based controller, based on internal sensors. Later chipsets (MSM8996, MDM9x45, etc.) start to support the aging compensation feature (see slides 7 and 8).



Note: AVS Type I may also be referred to as Core Power Reduction (CPR) in the Qualcomm Technologies, Inc. (QTI) Advanced Mobile Station Software (AMSS).

Introduction – What is AVS (2 of 2)

Why enable AVS?

- AVS is designed to dynamically optimize the power consumption for each individual QTI baseband device by setting the minimum but adequate operating voltage level. AVS compensates the operating voltage for any PMIC DC error, temperature changes, and any DC voltage drop between the PMIC sense point and the QTI baseband device.

- AVS can be supported on different power rails, depending on the capability of the MSM™ device (e.g., VDD_APCx, VDD_GFX, or VDD_CORE).

Important: AVS does not compensate for voltage drops caused by the instantaneous current transients (-L * di/dt), and, therefore, it is important for customers to meet the power distribution network (PDN) impedance specifications to ensure there are no V_{min} violations.

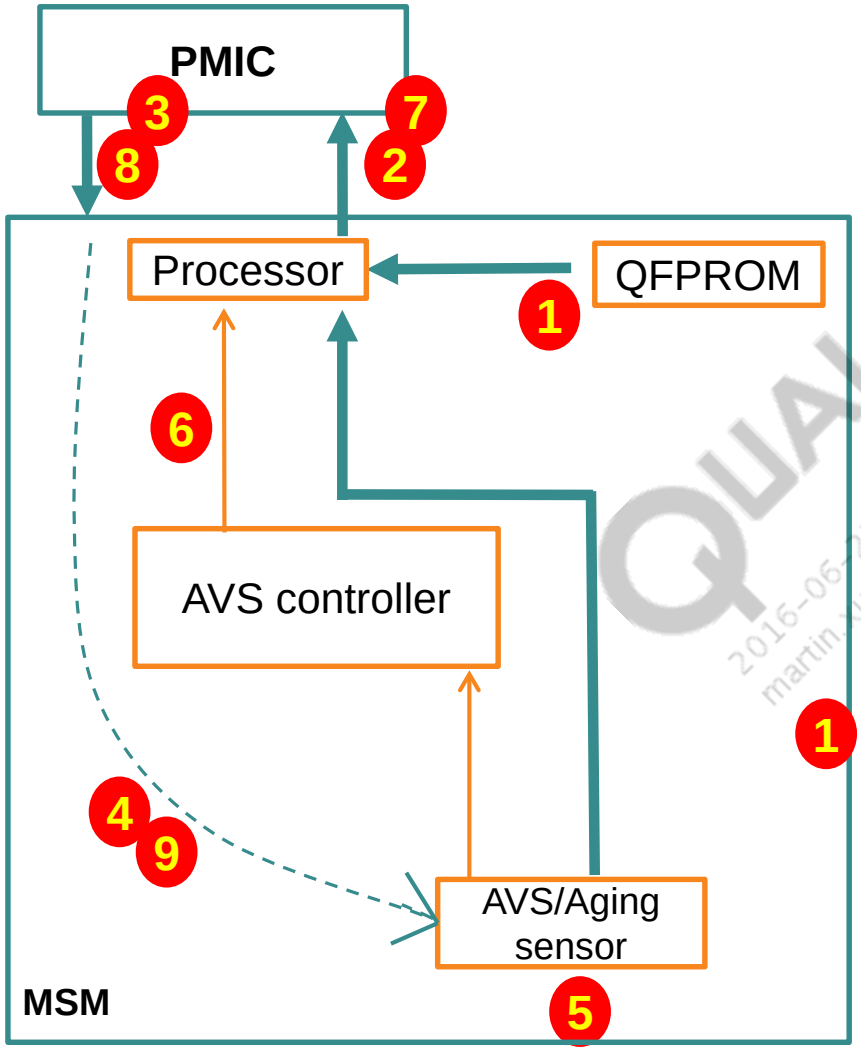
How is AVS Implemented

AVS has three main components:

- Qfuse bit settings determine the initial PMIC voltage settings.
- Voltage lookup tables are created for each Qfuse setting during device characterization. These tables are typically stored in the phone software image.
- New functionalities in the form of AVS sensors are embedded into the device subsystems that provide feedback to the RPM processor or hardware based controller for AVS Type I or directly to the PMIC for AVS Type II.

Check the MSM device software release notes and the device-specific revision guide (80-xxxxx-4) to confirm that AVS is enabled. It is the customer's responsibility to confirm that all PCB PDN specifications have been met.

AVS Type I Block Diagram (Software Based)



Open-loop operation (bootup):

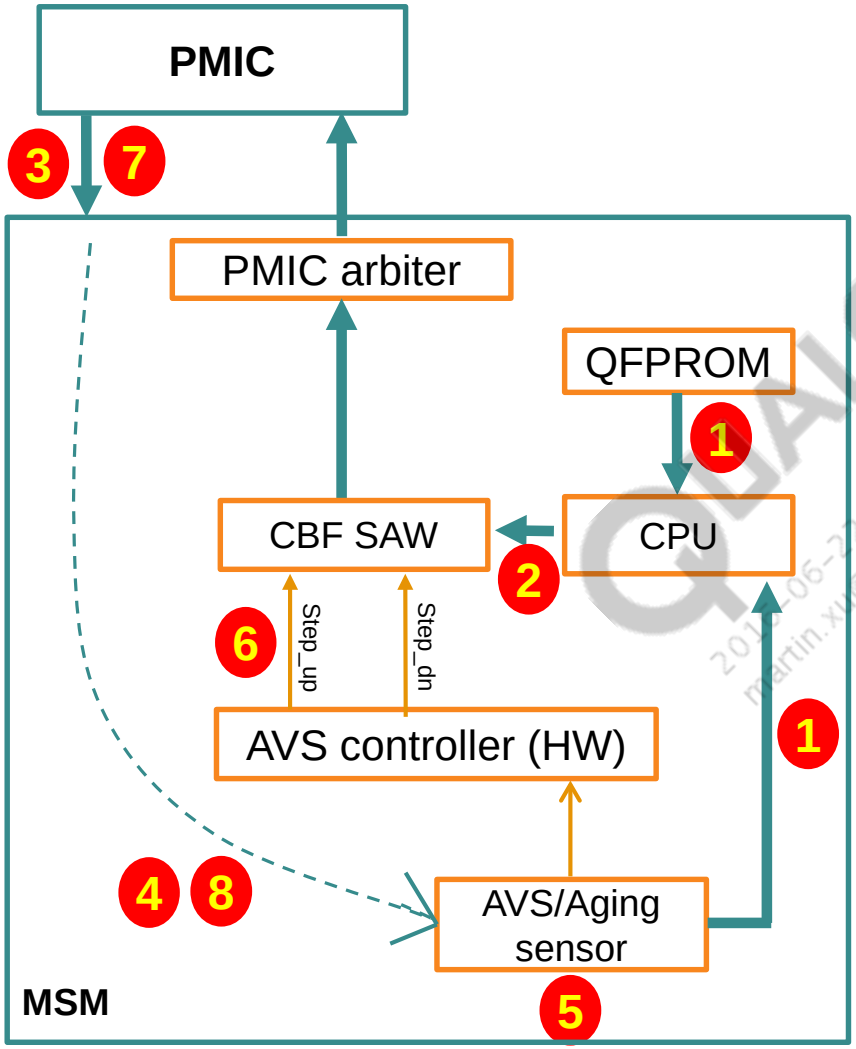
1. Processor reads eFuses and Aging sensor.
2. Processor sets initial voltages in the PMIC.
3. PMIC outputs updated voltage to the MSM.
4. Updated voltage affects AVS sensors.

Closed-loop operation:

5. AVS sensors generate AVS data.
6. AVS controller computes voltage re-adjustment and interrupts the processor.
7. Processor writes voltage re-adjustment to PMIC.
8. PMIC outputs updated voltage to the MSM.
9. Updated voltage affects the AVS sensors.

Processor: Resource and power management (RPM), modem, application processor
QFPROM: Qualcomm Fuse Programmable Read-Only Memory

AVS Type I Block Diagram (Hardware Based)



Open-loop operation (bootup):

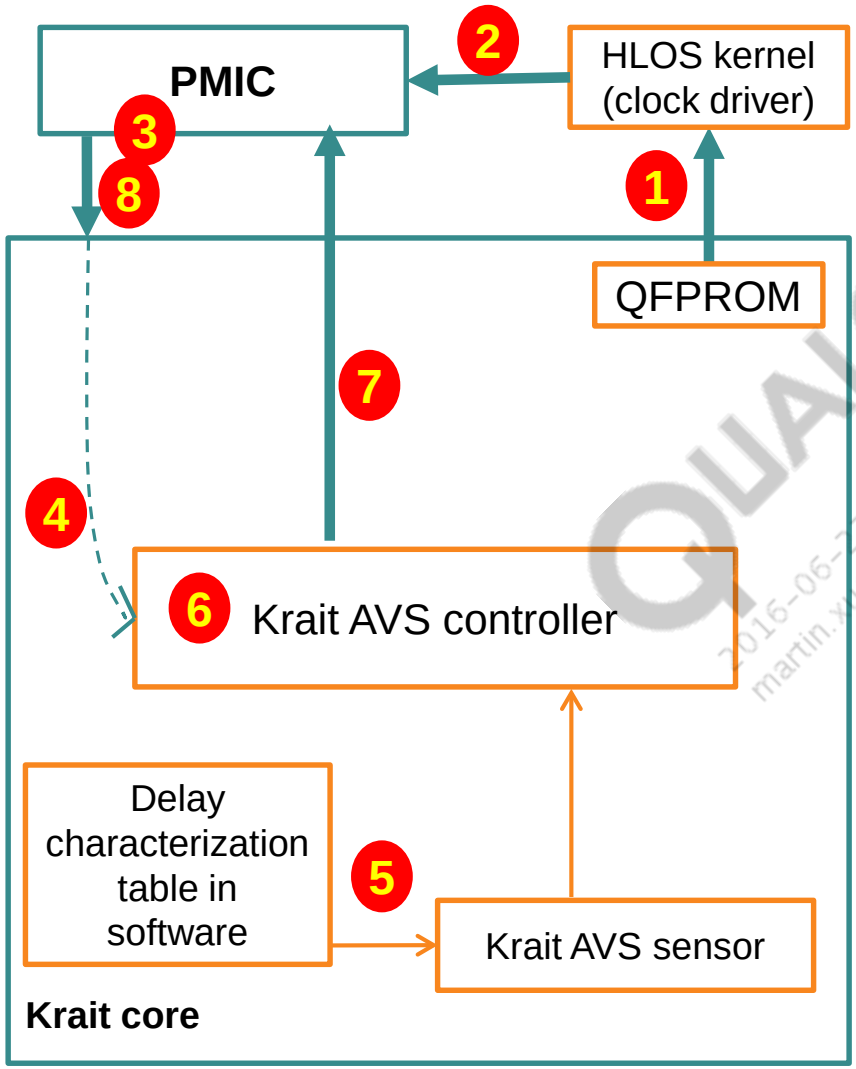
1. CPU reads eFuses and Aging sensor.
2. CPU sets initial voltages to PMIC through SAW and arbiter.
3. PMIC outputs updated voltage to the MSM.
4. Updated voltage affects AVS sensors.

Closed-loop operation:

5. AVS sensors generate AVS data.
6. AVS controller computes voltage re-adjustment and send to PMIC through SAW and Arbiter.
7. PMIC outputs updated voltage to the MSM.
8. Updated voltage affects the AVS sensors.

Note: Hardware based AVS type I is only available on MSM8998 and after chipsets.

AVS Type II Block Diagram



Open-loop operation (bootup):

1. The HLOS kernel (clock driver) reads the PVS eFuses.
2. The HLOS kernel (clock driver) sets the initial PVS recommended Krait voltages in the PMIC.
3. The PMIC outputs updated voltage to the Krait.
4. The updated voltage is logged by the Krait AVS controller.

Closed-loop operation:

5. Krait AVS sensors generate the AVS data based on the delay characterization table in the application software.
6. The Krait AVS controller computes the voltage readjustment based on this and the current operating voltage from step 4.
7. The Krait AVS controller writes the voltage readjustment to the PMIC.
8. The PMIC outputs the updated voltage to the Krait, and this is also logged by the Krait AVS controller.

Chipsets Supporting PVS and AVS

Chipset	PVS-only (open-loop only)	AVS Type II	AVS Type I (CPR)	
			Closed-loop	Open-loop
MSM8994/MSM8992	None	None	VDD_APC0, VDD_APC1, VDD_Core, VDD_GFX, VDD_MEM[1] and VDD_Modem	VDD_EBI[2]
MSM8974	Krait	None	None	Core[3], GFX[3]
MSM8974AB	Krait	None	Modem and GFX	Core
MSM8956/MSM8976			Core, APC (VDD_A72, VDD_A53), modem, and GFX	VDD_Mx
MSM8x26	None	None	Core, APC	
MSM8926	None	None	Core, APC, and modem	
MSM8x10/MSM8x12	None	None	Core, APC	
MSM8916	None	None	Core, APC	
MSM8939	None	None	Core, APC, and modem	
MSM8952	None	None	Core, APC, and modem	VDD_Mx
MSM8909	None	None	Core (with PM8909); Core, APC (with PM8916-1)	VDD_MEM (with both PMICs)
APQ8026	None	None	Core, APC	
APQ8064 V1	QDSP, Krait	None	None	
APQ8064 V2 **APQ8064-MB**	QDSP, Krait (< 1.89 GHz parts)	Krait (1.89 GHz parts only)	None	
APQ8074AB	Krait	None	GFX	Core
APQ8084	Krait	None	Core, GFX	
MDM9x15(M)	QDSP, core (VDD_CORE)	None	None	
MDM9x25(M)	None	None	Core, modem	
MDM9x30/MDM9x35M	None	None	Core, modem, and MEM	

Notes:

1. For turbo mode (when modem is not voted for turbo) and nominal mode.
2. Based on DDR frequency, the voltage will be different, offset from open-loop voltage.
3. For turbo and super turbo modes only.
4. Devices that have not been through both hardware and software commercial samples may still be in the process of characterization, and implementation is pending system validation.

Chipsets Supporting AVS Type I and Aging Compensation

Chipset	AVS Type I (CPR)		Aging compensation[2]
	Closed-loop	Open-loop	
MSM8996	VDD_Cx, VDD_APC, VDD_GFX, VDD_MSS, and VDD_Mx	VDDA	Yes
MSM8998	VDD_Cx, VDD_APC[1], VDD_GFX, VDD_MSS, and VDD_Mx	VDDA, VDD_SSC_Cx, VDD_SSC_Mx and VDD_WCSS	Yes
MDM9x40/MDM9x45	VDD_Cx, VDD_MSS, and VDD_Mx		Yes
MDM9x50/MDM9x55	VDD_Cx, VDD_MSS, and VDD_Mx		Yes

Notes:

1. Hardware based closed-loop AVS Type I.
2. VDD_Mx and VDDA do not support aging compensation.
3. Devices that have not been through both hardware and software commercial samples may still be in the process of characterization, and implementation is pending system validation.

Important

OEMs should not disable any AVS (Type I) in the production devices.

Side effects of AVS (Type I) disabling:

- Power consumption increases
- Potential thermal issues
- Constraints on 20 nm parts
  - On 20 nm parts (MDM9x35, MSM8994, and future chipsets), no single voltage can be used safely on all parts to run at turbo frequencies. At a minimum, AVS fuses must be used to calculate a suitable turbo voltage for the rail/part.
  - Running a part in low AVS bins with a voltage for high AVS bins (i.e., lower voltage) causes hold-time violations due to insufficient voltage. Raising VDD_MX at turbo voltage on 20 nm parts by more than 50 mV shortens the lifespan of the memory cells.

Affected chipsets: any chipsets that have AVS (Type I) enabled (see slides 10 and 11).

Note: AVS (Type I) ensures proper operation across the full range of devices (process variation). AVS (Type I) settings are based on the testing of a substantial number of devices.

Appendix – PVS Introduction

What is PVS?

- PVS is an extension to the dynamic clock and voltage scaling (DCVS) power management scheme.
  - In DCVS, the device voltage is adjusted based on the operating clock frequency.
  - In PVS, the device voltage is adjusted based on characteristics of the device itself.

Why is PVS needed?

- PVS was enabled to improve the system power performance of the products.
- Part-to-part differences in power consumption naturally increase as transistor sizes shrink. This effect is most evident at process nodes of 45 nm and below. The purpose of PVS is to minimize these differences.

How does PVS work?

- PVS works by using a lookup table to select the operating voltage for each device. This lookup table is created by QTI during production testing by blowing eFuses.
- Devices supporting PVS are listed on slide 10.

Questions

For additional information or to submit technical questions, go to: https://createpoint.qti.qualcomm.com