# Vesuvius Quick Start Application Note

## Rev A03: 02/16/2021


This document contains information on a product under development. The parametric information contains target parameters that are subject to change.

## Document Revision History

| Revision | Date | Description |
|---|---|---|
| A01 | 06/06/17 | Initial Draft |
| A02 | 12/07/17 | Added Force Current Setup |
| A03 | 02/16/21 | Fixed some errors in Table 5 |

## Table of Contents



 

# 1 Introduction

This document describes the steps to perform an initial system check-out of a Vesuvius device.

These instructions assume the customer system can set the Vesuvius supplies, has a mechanism to read/write registers, has the ability to measure voltages (either using an external DMM or system resource), control the digital input pins, and so on. The customer should also be able to adequately cool the Vesuvius.

Important Note: The steps described below illustrate a cause and effect to demonstrate how to interface with the device. In a normal application; sequencing, calibration and other factors may require the registers to be written in a different order.

## 1.1 Set Vesuvius Supplies

The first step is to apply the appropriate voltage. After power is applied, it is recommended to toggle the hardware reset (active high) on the Vesuvius device. Ideally all power supplies would become active simultaneously. However, since it is difficult to guarantee simultaneous levels, the following sequence is recommended:

1. VEE
2. VCC
3. VCCO_#
4. VDD
5. VOH_SDIO
6. V_REF

Schottky diodes are also recommended on a once per board basis to protect against any power supply restrictions. This can be reviewed in more detail in the power supply section of the Vesuvius datasheet.

Verify the voltage at the Vesuvius pins are within close proximity to ensure there are no IR drops. In addition, the customer should verify the approximate current flow for each supply. Table 1 below shows some typical current values after a hardware reset has been applied to the part.

Note: In some systems, it may not be possible to measure the supply currents on a device by device basis.

**Table 1: Power-ON Current Values after Reset**

| Supply | Voltage (V) | Approximate Current (mA) |
|---|---|---|
| VCC_OUT | 5.0 | 2.0 |
| VCC | 13.0 | 105.0 |
| VEE | -3.0 | 109.0 |
| VDD/VOH_SDIO | 3.3 | 80.0 |
| VREF | 3.0 | 0.0 |

### 1.1.1 Measure REXT

The next step is to measure the voltage of REXT. REXT voltage should track the VREF voltage within 25mV.

## 1.2 Register/RAM Access (CPU Port Transactions)

The following steps perform simple register/RAM access (write/read) to verify that the CPU port is functioning.

### 1.2.1 Read Die ID Register

Read the Die ID register (address = 0xC0FF); Table 2 lists the expected return values for different silicon revisions.

**Table 2: Die ID Register Values**

| Rev | Read-back Data (HEX) |
|---|---|
| 1 | 0xF101 |
| 2 | 0xF102 |
| # | 0xF10# |

### 1.2.2 Register Access

Follow Table 3 to verify the basic ability to write/read registers. The following bullet items highlight some key aspects of a register access:

- Address bit A15 (Register Bit) must be '1' to access a register
- Address bit A14 (Central Bit) is used to distinguish between Per-Chan and Central Registers
- WE bit must be '1' for the corresponding Data bits to be written
- WE bits should be read back as a don't care (X), however they typically read back '0'.
- Writing to Undefined Data bits has no effect
- Undefined Data bits (typically) read back '0'
- A RESET sets all registers to a default value of '0000' (except Read-Only registers)
- Refer to the CPU Protocol Timing diagram in the Vesuvius datasheet for details.
- Address bits A6-A13 are used to select which channel will be written. The CPU port can write to any or all selected channels with one write transaction by setting the selected channel bit high.

**Table 3: Write/Read Registers**

| Description | Address (HEX) | Write Data (HEX) | Read Data (HEX) |
|---|---|---|---|
| Reg Addr#4, Chan#0: Set all bits high | 0x8044 | 0xFFFF | 0x7BEF |
| Reg Addr#4: Chan#0: Clear Kelvin# bits | 0x8044 | 0x8000 | 0x03EF |
| Reg Addr#4: Chan#0: Clear rest of bits | 0x8044 | 0x0410 | 0x0000 |
| Central Addr#2: Set all bits high | 0xC002 | 0x0FFF | 0x07B7 |
| Central Addr#2: Clear all bits | 0xC002 | 0x0848 | 0x0000 |
| Central Reg Addr#0: Software RESET | 0xC000 | <don't care> | 0x0000 (All registers. Does not include DC levels storage) |

### 1.2.3 RAM (DC Level Memory) Access

Follow Table 4 to verify the basic ability to write/read RAM. The following bullet items highlight some key aspects of a RAM access:

- The Read-back Data matches the Written Data
- Address bit A15 (Register Bit) must be '0' to access RAM
- Address bit A14 (Central Bit) must be '0' to access RAM
- A RESET does NOT set the RAM contents to a default value. They are left unchanged
- A RAM read-back requires 2 addition clock cycles more than a register read-back
- Refer to the CPU Protocol Timing diagram for details
- Address bits A6-A13 are used to select which channel will be written. The CPU port can write to any or all selected channels with one write transaction by setting the selected channel bit high.

**Table 4: Write/Read RAM (DC Level Memory)**

| Description | Address (HEX) | Write Data (HEX) | Write Data (HEX) |
|---|---|---|---|
| ForceB# Level<15:0>, Chan#1 | 0x0081 | 0xFFFF | 0XFFFF |
| ForceB# Offset<15:0>, Chan#1 | 0x0091 | 0xAAAA | 0XAAAA |
| ForceB# Gain<15:0>, Chan#1 | 0x00A1 | 0x5555 | 0X5555 |

## 1.3 Reset State

Whenever a Hardware or Software reset is issued, the device is configured into the following state.

Note: The rest of the examples rely on the default (reset) configuration to demonstrate the minimum register transactions to achieve the desired configuration.

- RAM (DC Level Memory) – Unknown at power ON. Left unchanged if reset issued from a valid configuration while power is still ON.
- Registers – all registers are initialized to '0'; which implies the following (all settings are not listed below):
  - All switches open
  - All DPS channels are in High-Z
  - Feedback = Tight Loopback
  - Sel-FV# = ForceA#<15:0>
  - All current ranges are open
  - Voltage Range = 8V Range
  - FV-Mode = High Current
  - I-Clamps are disabled
  - Monitor and MI_Monitor are in High-Z
  - CPU-EN = 0: Channel is disabled
  - Kelvin and Current Alarms disabled

 

## 1.4 Set a Force Voltage Output on FORCE_0

Use the following sequencing to output a voltage on the FORCE_# pin using the ForceA#<15:0> level on channel 0. The FORCE_0 pin and SENSE_0 pin must be connected external to the Vesuvius if the SENSE feedback option is used. The example only shows register values that must be changed from default values.

Note: The FORCE_0 "Output Voltage" is an approximate voltage since these are un-calibrated settings.

**Table 5: Force Voltage Example**

| Description | Address (HEX) | Data (HEX) | Feed back | FORCE_0 (V) | MONITOR (V) |
|---|---|---|---|---|---|
| Software Reset | 0xC000 | 0x0000 | Tight | HiZ | HiZ |
| IR = IR5 (512mA Range) | 0x8045 | 0x1800 | Tight | HiZ | HiZ |
| ForceA0<15:0> Value = 0.5V | 0x0040 | 0x4FFF | Tight | HiZ | HiZ |
| ForceA0<15:0> Offset = 0V | 0x0050 | 0x7FFF | Tight | HiZ | HiZ |
| ForceA0<15:0> Gain = 1.0 | 0x0060 | 0x7FFF | Tight | HiZ | HiZ |
| CPU-EN0 = 1 | 0x8041 | 0x0009 | Tight | 0.5V | HiZ |
| Sel-MU-Pos0<1:0>=1, (FORCE) | 0x8045 | 0x0005 | Tight | 0.5V | HiZ |
| Set MUR<1:0>=1 (4V Range) (Gain=0.5V/V Offset=0.5V) | 0xC001 | 0x0280 | Tight | 0.5V | HiZ |
| Set CPU-Mon-OE=1 | 0xC001 | 0x0050 | Tight | 0.5V | 0.75V |
| Set FVR0=1 (16V Range) (DAC code doesn't change, but output voltage changes based on new voltage range. For more detail see Note1 below and Voltage Range Selection section of Vesuvius Data Sheet.) | 0x8040 | 0x0600 | Tight | 3.0V | 2.0V |
| ForceA0<15:0> Value = 2.5V (16V Range) | 0x0040 | 0x47FF | Tight | 2.5V | 1.75V |
| Sel-MU-POS0<1:0>=0, MI-S (Measure Current Mode) | 0x8045 | 0x0004 | Tight | 2.5V | 1.25V |
| Sel-MU-NEG0<1:0>=2, FI-Zero (Measure Current Mode) | 0x8045 | 0x0030 | Tight | 2.5V | 0.5V |
| Set MUR<1:0>=0, MI Range (Gain=1.33V/V Offset=1.5V) | 0xC001 | 0x0200 | Tight | 2.5V | 1.5V |
| Add 10 ohm resistor between the FORCE_A0 and GND (250mA) | N/A | N/A | Tight | 2.44V | 1.7V (244mA) |
| Tight-Loop*0=1 Take out of tight loopback | 0x8040 | 0x0003 | FORCE_# | 2.46V | 1.71V (246mA) |
| Sel-V-FB0<1:0>=1 Set Sense Feedback | 0x8040 | 0x0050 | SENSE_# | 2.5V | 1.72V (250mA) |

Note 1: Please see Tables 6, 7, and 8 for more details on the Voltage Range DAC mapping.

**Table 6: Voltage Range Selection**

| FVR# Bit | Voltage Range (V) |
|---|---|
| 0 | 8 |

| | |
|---|---|
| 1 | 16 |

**Table 7: 8V Range DAC Mapping**

| 8V Range | Programmed DC Level (V) |
|---|---|
| 0x0000 | -2 |
| 0x7FFF | +2 |
| 0xFFFF | +6 |

**Table 8: 16V Range DAC Mapping**

| 16V Range | Programmed DC Level (V) |
|---|---|
| 0x0000 | -2 |
| 0x7FFF | +6 |
| 0xFFFF | +14 |

## 1.5 Set a Force Current Output on FORCE_0

Use the following sequencing to output a current on FORCE_# pin using the ForceB#<15:0> level on channel 0. The example only shows register values that must be set in order to connect the Vesuvius in Force Current Mode. A load needs to be connected to the FORCE_0 pin so the output will not be open loop.

The sequence below sets the output to Force Voltage mode when first connecting to the load, then switches the output to Force Current Mode. Some of the Register/Level locations are a setup (FORCE_B, Cl-Vch-Isrc, Cl-Vcl-Isnk) while in Force Voltage Mode in anticipation to switching to Force Current Mode.

This example uses a 2 Ω resistor connected to the FORCE_0 pin for verification. The final step will force 256mA into the 2 Ω resistor which should produce near 0.5V.

Note: The FORCE_0 “Output Current” is an approximate current since these are un-calibrated settings

| Description | Address (HEX) | Data (HEX) | Feed back | FORCE_0 (V) | MONITOR (V) |
|---|---|---|---|---|---|
| Software Reset | 0xC000 | 0x0000 | Tight | HiZ | HiZ |
| IR = IR5 (512mA Range) | 0x8045 | 0x1800 | Tight | HiZ | HiZ |
| Con-Cap#<4:0> = 0x1F | 0x8041 | 0x87C0 | Tight | HiZ | HiZ |
| ForceA0<15:0> Value = 0V | 0x0040 | 0x3FFF | Tight | HiZ | HiZ |
| ForceA0<15:0> Offset = 0V | 0x0050 | 0x7FFF | Tight | HiZ | HiZ |
| ForceA0<15:0> Gain = 1.0 | 0x0060 | 0x7FFF | Tight | HiZ | HiZ |
| ForceB0<15:0> Value | 0x0041 | 0x7FFF | Tight | HiZ | HiZ |
| ForceB0<15:0> Offset = 0V | 0x0051 | 0x7FFF | Tight | HiZ | HiZ |
| ForceB0<15:0> Gain = 1.0 | 0x0061 | 0x7FFF | Tight | HiZ | HiZ |
| Cl-Vch-Isrc0<15:0> Value = 3.0V | 0x0042 | 0x56DB | Tight | HiZ | HiZ |
| Cl-Vch-Isrc0<15:0> Offset = 0V | 0x0052 | 0x7FFF | Tight | HiZ | HiZ |
| Cl-Vch-Isrc0<15:0> Gain = 1.0 | 0x0062 | 0x7FFF | Tight | HiZ | HiZ |
| Cl-Vch-Isnk0<15:0> Value = -1.0V | 0x0043 | 0xA108 | Tight | HiZ | HiZ |
| Cl-Vch-Isnk0<15:0> Offset = 0V | 0x0053 | 0x7FFF | Tight | HiZ | HiZ |
| Cl-Vch-Isnk0<15:0> Gain = 1.0 | 0x0063 | 0x7FFF | Tight | HiZ | HiZ |
| Sel-MU-Pos0<1:0>=1, (FORCE) | 0x8045 | 0x0005 | Tight | HiZ | HiZ |
| Sel-MU-Neg0<2:0>=0, (GND) | 0x8045 | 0x0020 | Tight | HiZ | HiZ |
| Set MUR<1:0>=3 (16V Range) (Gain=0.125 V/V Offset=0.75V) | 0x8051 | 0x01C0 | Tight | HiZ | HiZ |
| Set CPU-Mon-OE#=1 | 0x8051 | 0x0028 | Tight | HiZ | ?? |
| CPU-EN0 = 1 | 0x8041 | 0x0009 | Tight | 0.0V | 0.75V |
| ForceA0<15:0> Value = 0.5V (This step is just to test force voltage output. It is not necessary) | 0x0040 | 0x4FFF | Tight | 0.5V | 0.8125V |
| ForceA0<15:0> Value = 0.0V (Set Force Voltage level back to 0V) | 0x0040 | 0x3FFF | Tight | 0.0V | 0.75V |
| Tight-Loop*0=1 Take out of tight loopback | 0x8040 | 0x0003 | FORCE_# | 0.0V | 0.75V |
| FI/FV*# = 1 and CPU-D#<1:0> = 1 | 0x8040 | 0x4980 | FORCE_# | 0.0V | 0.75V |
| ForceB0<15:0> Value = 256mA | 0x0041 | 0xBFFF | FORCE_# | 0.5V | 0.8125V |
| | | | | | |

## 1.6 Measure Junction Temperature

The following steps show how to measure and calculate the junction temperature using the internal global temperature sensors. There are also per-channel sensors, but those are not discussed here.

Tj = (V2 – V1) * 1987 – 267

The measured temperature will be dependent on if there is a heat sink and/or air-flow present.

**Table 9: Measure Junction Temperature**

| Description | Address (HEX) | Data (HEX) | MONITOR Output (V) |
|---|---|---|---|
| Software Reset | 0xC000 | <don't care> | Hi-Z |
| Ch#2: Sel-MU-Pos2<1:0>=3 | 0x8105 | 0x0007 | Hi-Z |
| Ch#2: Sel-MU<2:0>=2 | 0xC001 | 0x000A | Hi-Z |
| Set MUR <1:0>=1 (4V Range) (Gain=0.5V/V Offset=0.5V) | 0xC001 | 0x0280 | Hi-Z |
| Ch#2: Sel-Diag2<3:0>=0xE (Central-Diag-A, Va) | 0x8204 | 0x001F | Hi-Z |
| Set CPU-Mon-OE=1 Enable Monitor | 0xC001 | 0x0050 | Measure v1 at monitor pin. |
| Ch#2: Sel-Diag2<3:0>=0xF (Central-Diag-B, Vb) | 0x8204 | 0x001E | Measure v2 at monitor pin |

Example: Using the previous setup: V1 = 1.559V Vb = 1.637V measured at monitor for the test setup. Since the 4V range was used in the MUR register, the voltages will have to be adjusted according to the gain and offset of the measurement unit range.

1) 4V MUR (measurement unit range): gain = 0.5V/V, offset = 0.5V
2) Solve for Vx voltage using the following equation: Vx = (gain * vx) + offset
   vx=measured value at monitor (scaled by measurement unit)
   Vx=actual input to measurement unit.
3) Rearrange above equation to the following: Vx = (vx – offset)/gain
   This will find actual V1 and V2 voltages.
4) V2 = 2.274 = ( V2 – offset ) / gain = (1.637V – 0.5V)/0.5
5) V1 = 2.118 = ( V1 – offset ) / gain = (1.559V – 0.5V)/0.5
6) Tj = 43C = (V2-V1) * 1987 – 267

 