# Jupiter Quick Start Application Note

## Rev B03: 05/20/2013

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 | 7/13/04 | Initial Draft – Rev 3 only |
| B01 | 6/2/2006 | Rev 8 support |
| B02 | 7/2/2006 | For Sel-FV example, section 1.4<br>◆ Fixed Local-Sense* = 1 register value<br>◆ Added Sel-FB = SENSE; remote feedback option. |
| B03 | 05/20/2013 | Change to Elevate Semiconductor |

## Table of Contents

# 1 Introduction

This document describes the steps to perform an initial system check-out of a Jupiter Rev 8+ device. These procedures are not applicable to earlier revisions as the register map and functionality are different.

These instructions assume the customer system can set the Jupiter 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 be able to adequately cool Jupiter.

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. Please refer to the Planet ATE Software Driver documentation for details.

## 1.1 Set Jupiter Supplies

The first step is to apply the appropriate voltage. For initial checkout, it is recommended to set VCC_OUT and VCC to the same voltage. After power is applied, it is recommended to toggle the hardware reset (active high) on Jupiter device.

Verify the voltage at the Jupiter pins or within close proximity to ensure there are no IR drops. In addition, the customer should verify the approximate current flow for each supply.

Note: In most systems, it may not be possible to measure the supply currents.

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

| Supply | Voltage (V) | Approximate Current (mA) |
|---|---|---|
| VCC_OUT | 16.0 | 5 |
| VCC | 16.0 | 20 |
| VEE | -5.0 | -25 |
| VDD | 3.3 | 15 |
| VREF | 3.0 | 0.0 |

### 1.1.1 Measure REXT

The next step is to measure voltage of the 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 to CPU port is functioning.

### 1.2.1 Read Die ID Register

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

**Table 2: Die ID Register Values**

| Rev | Read-back Data (HEX) |
|---|---|
| 8 | 0x1F48 |
| # | 0x1F4# |

### 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 D15 (Register Bit) must be '1' to access a register
- Address bit D14 (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 (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 for details

**Table 3: Write/Read Registers**

| Description | Address (HEX) | Write Data (HEX) | Read Data (HEX) |
|---|---|---|---|
| Static Addr#7: set all bits high | 0x8007 | 0xFFFF | 0x03DB |
| Static Addr#7: clear SR-Adj bits | 0x8007 | 0x0400 | 0x001B |
| Static Addr#7: clear rest of bits | 0x8007 | 0x0024 | 0x0000 |
| Static Addr#64: set all bits high | 0x8040 | 0xFFFF | 0x0FBF |
| Static Addr#64: clear all bits | 0x8040 | 0x1040 | 0x0000 |
| Rt-DAC Addr#6: set all bits high (note: no WE bits) | 0x8086 | 0xABCD | 0xABCD |
| Central Reg: Software RESET | 0xC001 | <don't care> | 0x0000 (all registers) |

### 1.2.3 RAM Access

Follow Table 3 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 D15 (Register Bit) must be '0' to access a register
- Address bit D14 (Central Bit) is used to distinguish between Per-Chan and Central Registers
- 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
- Refer to the CPU Protocol Timing diagram for details

**Table 4: Write/Read Registers**

| Description | Address (HEX) | Data (HEX) |
|---|---|---|
| VCH Value DAC | 0x0002 | 0xFFFF |
| VCH Offset DAC | 0x0022 | 0xAAAA |
| VCH Gain DAC | 0x0042 | 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 – Unknown at power ON. Left unchanged if reset issued from a valid configuration
- Registers – all registers are initialized to '0'; which implies:
  - All switches open
  - Feedback = FORCE_A (tight loop)
  - Sel-Force = GND (FORCE_A output is always active and therefore will be 0V approximately)
  - IR = 15uA
  - Voltage Range 0
  - V-Clamps and I-Clamps are disabled
  - Monitor is in high-Z
  - OT-Alarm enabled (OT-Alarm-Dis=0)
  - Kelvin and Current Alarms disabled
  - CAP_DIS*, DPS_EN*, and CBIT* high-z (no current flowing)

## 1.4 Set FV = 3V on FORCE-A and FORCE-B

Use the following sequencing to output 3V on the FORCE_A, FORCE_B and MONITOR pins.

The outputs are an uncalibrated voltage. Notice that the FORCE_A voltage is always present.

**Table 5: Force Voltage Example**

| Description | Address (HEX) | Data (HEX) | Feed back | FORCE_A (V) | FORCE_B (V) | MONITOR (V) |
|---|---|---|---|---|---|---|
| Software Reset | 0xC001 | 0x0000 | FORCE_A | 0V | HiZ | HiZ |
| Con-CompA/B caps | 0x8001 | 0x0038 | FORCE_A | 0V | HiZ | HiZ |
| IR = IR5 (512mA) | 0x8002 | 0x000D | FORCE_A | 0V | HiZ | HiZ |
| Vmid = 1.5V | 0xC002 | 0x001B | FORCE_A | 0V | HiZ | HiZ |
| VF0 Value = 3V | 0x8086 | 0xDFFF | FORCE_A | 0V | HiZ | HiZ |
| VF0 Offset = 0V | 0x8088 | 0x7FFF | FORCE_A | 0V | HiZ | HiZ |
| VF0 Gain = 1.0 | 0x808A | 0x7FFF | FORCE_A | 0V | HiZ | HiZ |
| Sel-Force = RT-DAC | 0x8003 | 0x000B | FORCE_A | 3V | HiZ | HiZ |
| Con-FA-FB | 0x8006 | 0x000C | FORCE_A | 3V | 3V | HiZ |
| Enable Monitor in MV mode (Sel-MV = FORCEA) | 0x8004 | 0x0018 | FORCE_A | 3V | 3V | 3V (MV) |
| Set Voltage Range = VR1. (DAC code doesn't change so output voltage is doubled) | 0x8080 | 0x0009 | FORCE_A | 6V | 6V | 6V (MV) |
| Set FV = 5V | 0x8086 | 0xBFFF | FORCE_A | 5V | 5V | 5V (MV) |
| Put into MI mode | 0x8004 | 0x0005 | FORCE_A | 5V | 5V | 0V (0uA) |
| Add 50 ohm resistor between FORCE_B to GND | N/A | N/A | FORCE_A | 5V | 4.9V | 200mV (100mA) |
| Sel-FB = FORCE_B | 0x8004 | 0x0280 | FORCE_A | 5.0V | 4.9V | 200mV (100mA) |
| Local-Sense* = 1 and CPU-EN = 1 | 0x8005 | 0x00CC | FORCE_B | 5.1V | 5V | 200mV (100mA) |
| Connect FORCE_B to SENES pin | NA | NA | FORCE_B | 5.1V | 5V | 200mV (100mA) |
| Sel-FB = SENSE | 0x8004 | 0x0300 | SENSE | 5.1V | 5V | 200mV (100mA) |

The following table shows different RT-DAC (FORCE_A) output voltage DAC codes when Vmid=1.5V. These voltages don't reflect any Vcc/Vee headroom limitations.

| RT-DAC Code | VR0 | VR1 | VR2 | VR3 |
|---|---|---|---|---|
| 0x0000 | -0.5 | -1.0 | -2.0 | -4.0 |
| 0x1FFF | 0.0 | 0.0 | 0.0 | 0.0 |
| 0x3FFF | 0.5 | 1.0 | 2.0 | 4.0 |
| 0x5FFF | 1.0 | 2.0 | 4.0 | 8.0 |
| 0x7FFF | 1.5 | 3.0 | 6.0 | 16.0 |
| 0x9FFF | 2.0 | 4.0 | 8.0 | 20.0 |
| 0xBFFF | 2.5 | 5.0 | 10.0 | 24.0 |
| 0xDFFF | 3.0 | 6.0 | 12.0 | 28.0 |
| 0xFFFF | 3.5 | 7.0 | 14.0 | 32.0 |