Dynastat Embedded Systems Manual

Table of Contents

Control and Power Cape

The BeagleBone Control Cape is a board that fits on top of the Beagle Bone Black or Beagle Bone Green embedded boards.

The cape provides the following features:

- Regulated 5V power supply for BBB, Remote boards and motors.
- Provides connectors for 10x RMCS-220X motors.
  I2C and 12V are present on the connectors (no UART).
- 5 connectors to 10x Control switches for a homing sequence.
- 2 connectors for a remote chain of pressure sensor boards.
- An I2C Real Time Clock (DS3231).

Connectors

Power Supply

Power is provided via a 2-way screw terminal block.
Expected supply is 12V DC, 10A; however the regulator is capable of a wide input range (6V to 25V).

Polarity is marked on the PCB.

Motors

The motor connectors are a 6-pin JST, labelled P7,P8,P10,P11,P13,P14,P16,P17,P20,P21 on the schematic.
All of these are connected to the 12V screw terminal power input via a trace capable of carrying approximately 10A. As such, only one motor may be operated at a time.

Pin	Description
1	Ground
2	I2C2 Clock
3	I2C2 Data
4	N/C
5	N/C
6	+12V

Control Switches

The control switch connectors have provision for 2 switches on each connector. They also provide +5V should a proximity switch or other active device be required in the future.

The connector is a wire insertion type to allow solder less assembly. The outer diameter of each wire (including insulation) should be less than 1.5mm.

Pin	Description
1	+5V
2	Input 1 (10K Pull up to +5V)
3	Ground
4	+5V
5	Input 2 (10K Pull up to +5V)
6	Ground

Sensor Chain

There are two sensor connectors on the board to allow connection to one or more remote sensor boards.

The suggested configuration is to have two chains, each containing 1 pressure array board and 1 satellite sensor board.

Pin	Description
1	+5V
2	I2C1 Clock
3	I2C2 Data
4	Ground

Real Time Clock

The RTC connector is designed to accept an off the shelf RTC module based on the DS3231 chip.

This will be connected to I2C2 and can be accessed via I2C2 (i2c-1 under Linux) at address 0x68.

It may be used by the kernel with the following command:

echo ds3231 0x68 >/sys/bus/i2c/devices/i2c-1/new_device

Linux Notes

The BBB by default has the I2C1 bus disabled. It can be enabled by issuing the following command:
echo BB-I2C1 > /sys/devices/bone_capemgr.9/slots

Note that the bus numbering under Linux is dependent on the order in which the buses are initialised so doesn't always match the schematics.

The following command can be used to check the actual bus ordering:
*ls -l /sys/bus/i2c/devices/i2c-**

The lower the HEX address towards the end of the location, the lower the I2C bus:

44e0b000 = I2C0
4802a000 = I2C1
4819c000 = I2C2

In all cases so far, I2C1 and I2C2 have been swapped around on my system.

An example /etc/rc.local is as follows:

```
#!/bin/sh -e
#
# rc.local
#
# This script is executed at the end of each multiuser runlevel.
# Make sure that the script will "exit 0" on success or any other
# value on error.
#
# In order to enable or disable this script just change the execution
# bits.
#
# By default this script does nothing.

echo ds3231 0x68 >/sys/bus/i2c/devices/i2c-1/new_device
echo BB-I2C1 > /sys/devices/bone_capemgr.9/slots
sleep 1
hwclock -f /dev/rtc1 -s

exit 0
```

Programming Registers

The control board contains an MCU that monitors the state of the control switches.

The micro controller is connected to I2C1 and the IRQ line is connected to GPIO1_16 of the BBB.

The default I2C slave address of the MCU is 0x20.
All registers are 16-bits wide. The address word is 16-bits.
The following I2C registers are available:

Address	Description	Read	Write
0x0000	ID Register	0xFE00	N/A
0x0001	Interrupt Status Each bit corresponding to inputs 1-10 will be set if an input has gone low since the last clear. Writing a bit will clear the status (write 0x3FF to clear all).	0x0000	0x03FF
0x0002	Interrupt Enable For each input that you wish to cause an interrupt (set the IRQ line low), set the corresponding bit.	0x0000 (default)	0x03FF
0x0003	Control Switch Values 1-10 Bits 0-9 will represent the raw logic state of each input 1-10. The default is high (1) due to the 10K pull up.	0x03FF	N/A
0x0004	Set Device Address This will set the device's I2C address after the next chip reset.	Current Address	0x00-0x7F

Sensor Interface (Array)

The sensor interface board provides a localised system capable of measuring two sensor grids up to 16x16 and 8x16.

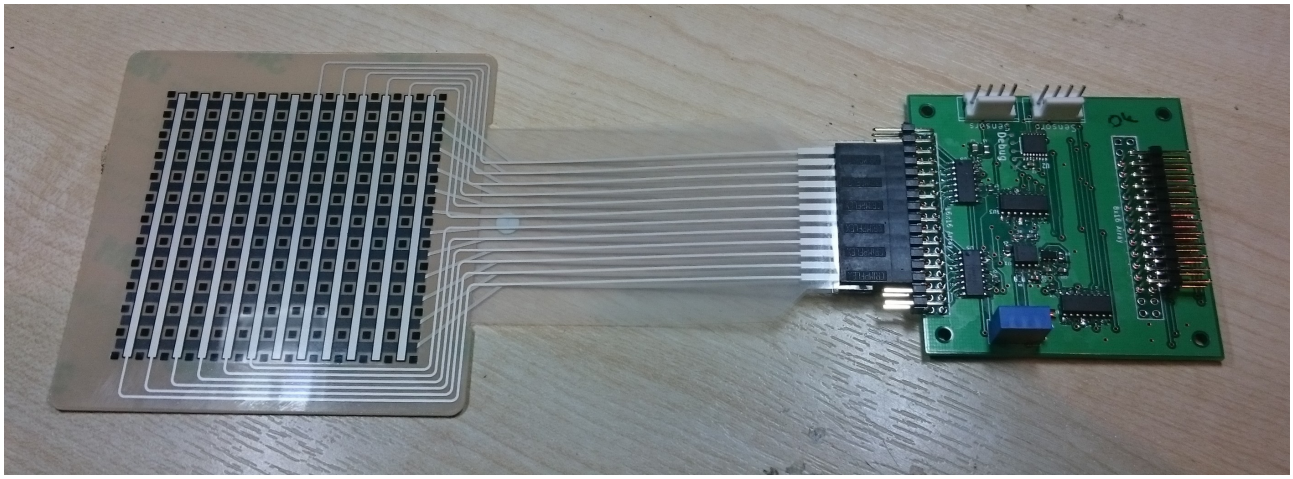
Linearity is achieved by using an inverting amplifier to perform a 1/x in the analogue domain. This avoids losing resolution in the ADC and doing conversion in software.

The results matrix has values from 0x000 to 0xFFF.

Sensor Connection

When configured in forefoot mode, the 16x10 sensor connects to the “16x16 Array” connector and the 6x12 sensor connects to the “8x16 Array” connector. The connectors should be central with an equal number of unconnected pins on each side.

When configured in heel mode, the 12x12 sensor connects to the “16x16 Array” connector.



Please pay close attention to the orientation of the board and sensors.

Programming Registers

The sensor board contains an MCU that continuously scans the sensors.

The default I2C slave address of the MCU is 0x21.
All registers are 16-bits wide unless otherwise stated. The address word is 16-bits.
The following I2C registers are available:

Address	Description	Read	Write
0x0000	ID Register	0xFE01	N/A
0x0001	Set matrix map to predefined set: 0: 1 to 1 mapping. No reordering. 1: Map for 16x10 and 6x12 matrices 2: Map for 12x12 matrix	0x0000 (default)	0x0000-0x0002
0x0002	Reserved	0x0000 (default)	N/A
0x0003	Reserved	0x0000 (default)	N/A
0x0004	Set Device Address This will set the device's I2C address after the next chip reset.	Current Address	0x00-0x7F
0x0100-0x01FF	256 Registers in a Column (16) x Row (16) Array (Matrix 1)	0x0000 (default)	N/A
0x0200-0x027F	128 Registers in a Column (8) x Row (16) Array (Matrix 2)	0x0000 (default)	N/A