

# Recycler BPM Calibration Application Users Guide

**Author: Marc Mengel, Fermilab**

This document will attempt to describe the normal usage of the Recycler BPM Calibration Application.

It assumes you have a working knowledge of the ACNET Console system (i.e. how to log on, start a console session, and find applications in the index page, etc.)

NOTE: to make this document print well, the graphics screen images here are presented as negated images, so the following colors are swapped:

- black vs white
- blue vs yellow
- red vs cyan

## Start-up

The application is currently assigned to page R26.

## Main screen

The main screen looks like:

### **BPM Calibration Main**

```
◆New Calibration Run◆  
◆Retrieve Old Run◆  
◆View BPM History◆
```

and lets you choose between three major modes of operation:

- Starting a new calibration test run
- Retrieving results of a previous calibration test stored in the database
- Reviewing results of a given single BPM over multiple past tests.

All three of these will present you with a screen to fill in parameters appropriate to that choice, and then take you to a results screen, where you can examine graphs of results and/or save them in various formats. Each of these will now be described in more detail.

## A New Calibration Run

Choosing "New Calibration Run" on the main screen takes you to a screen

### Run Calibration

```
House Range: -<RBPM10 >+--<RBPM31 >+
Waveform: -<Fake >+
Levels: -<All >+

Progress: < >

◆Start◆           ◆Dismiss◆
```

to let you choose options about the test you run. For most normal cases, you will want to simply choose "Bunched" for the "Waveform" type and hit "Start", but there are several options you may want to choose:

#### House Range

This lets you pick which "houses" (individual ACNET nodes with connections to BPMs) you wish to test. Often a particular house has had maintenance done to it, etc. and you would like to either test it in particular, or not test a particular house. So you can set the high and low ends of a range of houses to display.

Set both selections to the same house to test just that house.

#### Waveform

This lets you choose which waveform will be:

1. generated by the signal generator in the houses
2. listened for when collecting data

There is no option to set these differently from each other.

NOTE: The "Fake" waveform, if chosen, does **not** actually run a calibration test; instead data is filled in to the tables for graphing *as if a test had been run*.

#### Levels

This lets you pick the voltage level to set in the waveform generator. The default "All" (which does all 3 levels in sequence) is recommended for normal testing, as some of the displays are much less informative with out this setting.

Having filled out the request information as you desire, you can either ♦Start♦ the test, or hit the ♦Dismiss♦ button to go back To the main screen.

Once you hit ♦Start♦, an **Abort** button will come up to let you abort the test if you need to for any reason, and the Progress: slider will move as the test progresses.

When the test is completed, the "View Results" screen will come up, and a graphics window with 2 graphs will be displayed.

## View Results

The View results screen

### View Results

```
Date: 05-DEC-2003 12:05:00
Note: THURSDAY RUN
Graph: -+
Level: -+
Data: -<H Raw Ratio>+
HRange: -<R:HP100 >+   -<R:HP518 >+
VRange: -1                1

Save in          List Bad
♦Spreadsheet♦   ♦Screen♦
♦Database♦      ♦Email♦
♦Email♦                ♦Dismiss♦
```

and accompanying graphics screen

lets you:

- View a variety of graphical views of the calibration data
- Modify the graph displays (i.e. bounds, etc.) on the graphics screen
- set the Note string that will be saved with the run results if you save them in the database
- Save the results in either a spreadsheet file, or as a database entry, or both -- if we are displaying the results of a run.

Data column names are 3 characters (just like the column headings in the database):

```
Level  Type      Ratio
-----  ----      -
```

```

l(ow)   r(aw)       1(:1)
m(ed)   s(caled)   2(:1)
h(igh)  i(ntensity)
e(error)

```

- Request list of out-of-range BPMs either to the screen or to email.

## Modifying graph displays

### Graph

controls whether we are modifying the upper or lower graph

### Level

which voltage level's results we display: normally the High voltage gives us the smallest error bars on our ratio and position displays.

### Data

which type of data we're displaying -- for each Axis (H == Horizontal or V == Vertical):

- Raw Ratio: shows unscaled  $(A-B)/A+B$  ratio values.

This is the normal display to spot bad BPMS (i.e. VP303 in the preceding graphs). There *should* be a row of blue squares between the lower yellow lines, and a row of white diamonds between the upper yellow lines. Both should have error-bar lines not much wider than the yellow acceptable-range lines

- Position: scaled  $(A-B)/A+B$  ratio values

This shows the actual position of our test data, if you want a physical number to show how far away from where it should be the BPM is reading.

- Intensity: scaled  $A+B$  values

this shows the intensity value.

- Gain Bias: shows multiple-voltage-level data to show differing gain

between A and B channels.

If the BPM has equal gain between the A and B channels, it will have a red diamond at 0, and white square at 1, showing the  $(A-B)/(A+B)$  and  $A/B$  values, respectively. For example R:HP108 in the graph

If the BPM has fixed, but uneven gain ratio between the A and B channels, there will be a spread of colored diamonds near 0, and a white box

somewhat above or below 1 indicating the (constant) A/B ratio. For example R:HP112 in the graph.

If the BPM has *varying* gain ratio between A and B, you would get a spread of colored boxes near 1, (but so far we've never seen this).

- Histograms of first 3 These are so those who are statistically inclined can get a feel for the distribution of the various BPM readings. This is mostly useful when you've gotten the history for a given BPM.

### Ranges

Let you set the ranges for the two axes of the graph; the BPM range (or run number range if this is history data) for the horizontal axis, the vertical range for the graph in whichever units we are currently operating.

Note: changing the ranges for Histograms currently doesn't work.

### Saving data

You can currently save run results in the database, (which allows you to retrieve them later using the Retrieve old Run screen) and also to a spreadsheet file, and you will soon be able to have the spreadsheet file email-ed to you by filling in the email address and clicking the email button.

### 'Bad' BPM lists

A list of BPMs which are out of the tolerance range can be generated to the screen, or e-mailed to an email address. If you want to send it out of the beams cluster, y Screenou need to specify `smtp%user@fnal.gov`

## Retrieve Old Runs

This screen

### Retrieve Old Run

Run	Date	Note
4	05-NOV-2003 12:00:00	TEST RUN
8	05-NOV-2003 14:05:03	ANOTHER TEST

◆Load◆ ◆Delete◆

◆Dismiss◆

lets you pick results previously saved to the database, and review the graphs, save the data as a spreadsheet, etc. First you get a screen with the list of saved items, which you can scroll through using the scrollbar, and select one by clicking on it.

Pressing the ♦Load♦ button will load the selected data, and take you to the Results screen as if you had just run that test.

Pressing the ♦Dismiss♦ button will take you back to the main screen as if nothing had happened.

Pressing ♦Delete♦ lets you delete an old run from the database.

## Review BPM history

This screen

### Retrieve BPM History

```
BPM: -<Horiz>+ -<R:HP100 >+
```

```
♦Get History♦ ♦Dismiss♦
```

lets you pick a BPM to review the previous values read in calibrating that BPM, and to look for trends in changes in the values associated with that BPM.

You pick either horizontal or vertical, and pick the BPM from that list.

Clicking the ♦Get History♦ button gets you a trimmed down Results screen, and lets you select various graphs for that item.

NOTE: A given BPM's history will only valid data for Horizontal or Vertical graphs, but not both.

## Accessing the Database

If you've never used the database directly, you may need to see the [intro database](#) page, and get a database account.

Currently the BPM Calibration application uses two tables in the "appdb" database:

```
appdb.mengel.BPM_Calibration_Run_Info (
    run_number  int          not null,
    date        smalldatetime not null,
```

```

        note          char(30)          not null,
    )
    appdb.mengel.BPM_Calibration_Values (
        bpm_name      char(8)            not null,
        run_number    int                not null,
        lr1           real               not null,
        mr1           real               not null,
        hr1           real               not null,
        lr2           real               not null,
        mr2           real               not null,
        hr2           real               not null,
        ls1           real               not null,
        ms1           real               not null,
        hs1           real               not null,
        ls2           real               not null,
        ms2           real               not null,
        hs2           real               not null,
        lil           real               not null,
        mil           real               not null,
        hil           real               not null,
        li2           real               not null,
        mi2           real               not null,
        hi2           real               not null,
        le1           real               not null,
        me1           real               not null,
        he1           real               not null,
        le2           real               not null,
        me2           real               not null,
        he2           real               not null,
    )

```

Data feild names are 3 characters (just like the column headings in the spreadsheet save format):

Level	Type	Ratio
-----	-----	-----
l(ow)	r(aw)	1(:1)
m(ed)	s(caled)	2(:1)
h(igh)	i(ntensity)	
	e(rror)	

## Sample Database session

Lets look at a simple database session on CNS55:

```
CNS55> isql/server=ADBS/user="your_username"
```

```
Password:
```

```
1> use appdb
```

```
2> go
```

```
1> select * from appdb.mengel.BPM_Calibration_Run_Info
```

```
2> go
```

run_number	date	note
-----	-----	-----
4	Nov 5 2003 12:00PM	TEST BEFORE CHOIR
8	Nov 5 2003 3:23PM	FIRST AFTERNOON RUN MI50
11	Nov 6 2003 4:11PM	YA AFTERNOON RUN
17	Nov 20 2003 4:18PM	TEST LISTEN MODE
18	Nov 20 2003 4:22PM	TEST FULL RR RUN

```

19      Nov 20 2003  4:34PM EVERYBODYS STUCK AGAIN...
20      Feb 27 2004 10:14AM MI60 W/ PROTONS
21      Feb 27 2004 10:59AM
22      Feb 27 2004 10:59AM FULL RING NO BEAM

```

(9 rows affected)

1>quit

CNS55>

Here we've connected to the database, and just dumped out the run information table. But there are lots of possibilities. Here are some sample queries:

View runs and descriptions:

```
SELECT * FROM appdb.mengel.BPM_Calibration_Run_Info
```

View all values for a given run:

```
SELECT * FROM appdb.mengel.BPM_Calibration_Values
where run_number=4
```

Show the 2 low voltage raw readings for the R:HP5xx BPMS in run 4:

```
SELECT bpm_name,lr1,lr2 FROM appdb.mengel.BPM_Calibration_Values
WHERE run_number=4 AND bpm_name like 'R:HP5%'
```

Show the history of low voltage readings for R:HP502 with run comments and dates:

```
SELECT v.run_number, v.bpm_name, v.lr1, i.date, i.note
FROM
  BPM_Calibration_Values as v
  JOIN BPM_Calibration_Run_Info as i
  ON i.run_number = v.run_number
WHERE bpm_name LIKE 'R:HP100%'
```