



MUON Cathode
FE Electronics



BUCKEYE ASIC TEST

Code: /home/fast/fastdaq/daqmb-new2.8/utils/rad_test2.c

- Running:**
1. select BUCK run test
 2. write down time for run start
 3. in separate window type */cern/pro/bin/pawX11*, then *exec buckloop.kumac..*
 4. monitor current for latchup

Test1 Shifting: digital : (1:1)

48 bit patterns are continuously shifted through Buckeye 3.

```
000000000000
111111111111
222222222222
```

...

```
ffffffffffffff
```

for each shift 2x48 bits are shifted, and thus shifted and stored bits are checked.

```
e.g.  shift in  333333333333
      output  3333333333332222222222
                new          old
```

Test2 Calibration: analog : (1:5000)

Using the external pulser, pulse shapes for each buckeye 3 channel are stored for 6 voltage pulser settings.

Output:

```
/home/fast/data/tmp_files/buckpulMMDD_HH:MM.dat
storage of pulser data in format:
secs chan volt# volts 16 time samples
```

```
/home/fast/data/tmp_files/buckshftMMDD_HH:MM.dat
on error shift information is output
***  shift error  cnt hrs mins sec errs
input 333333333333
output 3333333333332222222222
```

Display:

Terminal: every 500 pulses summary is displayed

```
*** shift error cnt hrs mins sec errs
```

Paw: 16 buckeye channels highest voltage pulse peak
is monitored as a function of time

Format:

/home/fast/data/tmp_files/buckpulMMDD_HH:MM.dat

time event channel pulse# voltage {16 adc values}

example:

```
0.00 01 00 00 0.000 606 614 598 599 607 606 606 607 605 608 603 605 602 605 606 607
0.00 01 00 01 0.941 605 831 1247 987 693 573 554 554 556 566 568 573 579 584 586 587
0.00 01 00 02 1.882 604 1063 1906 1365 772 544 503 506 515 528 538 548 554 564 570 577
0.00 01 00 03 2.824 603 1269 2563 1757 850 513 448 454 468 492 508 519 531 540 552 560
0.00 01 00 04 3.765 602 1485 3214 2147 932 483 400 405 429 452 476 493 509 525 538 546
0.00 01 00 05 4.706 604 1690 3846 2532 1012 453 352 360 390 421 448 493 510 524 537 539
```

/home/fast/data/tmp_files/buckshftMMDD_HH:MM.dat

*event hr min sec errcount
input shift bits
output shift bits*

example:

```
**** shft error 2 0 0 0.00 1
input 222222222222
return 222222222222111111111111
**** shft error 3 0 0 0.00 2
input 333333333333
return 333333333333222222222222
```