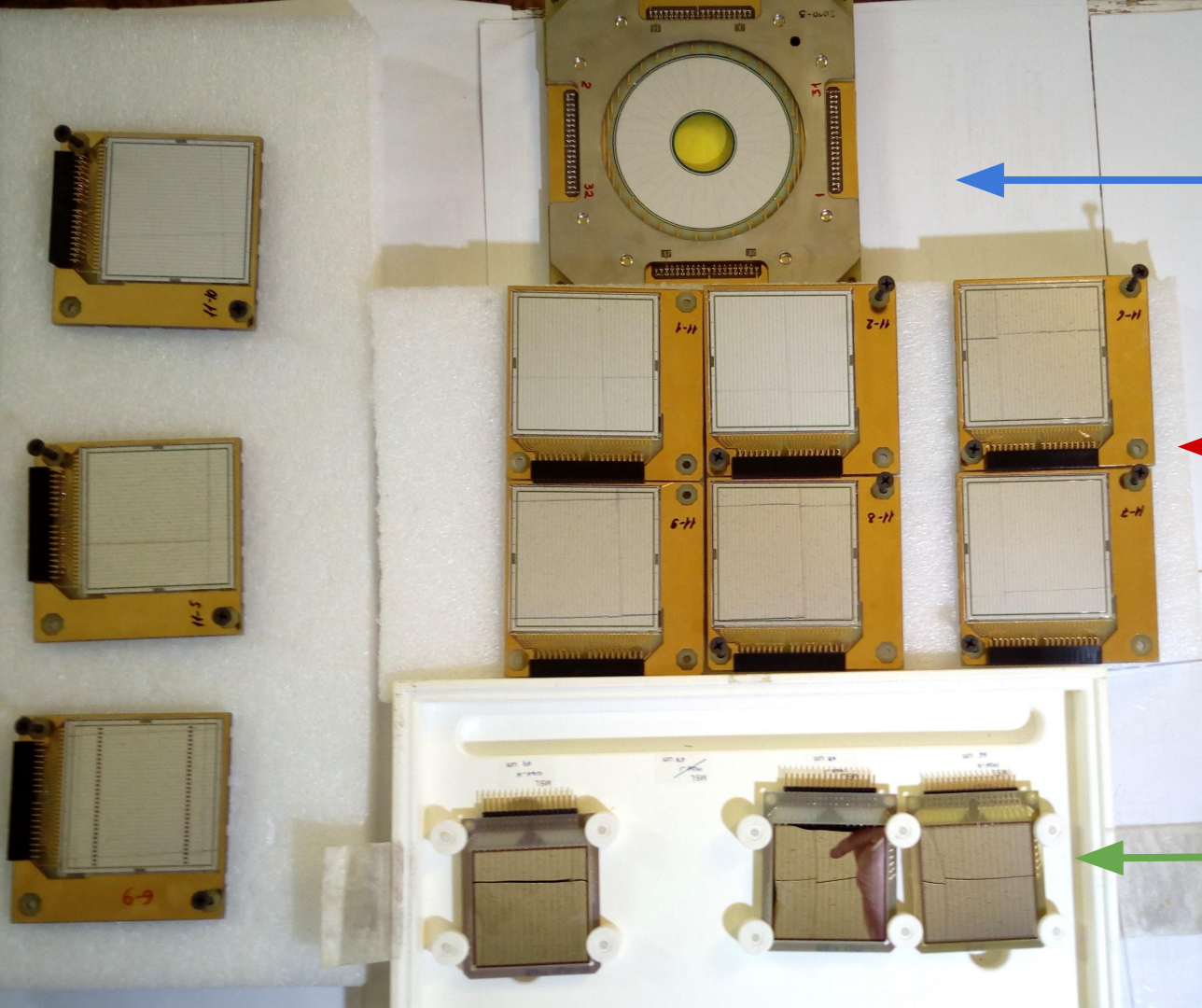




# Light ion detection at ACCULINNA ion separator

Jakub Malczewski - AGH  
Arkadiusz Siwiec - UW, PW

supervisor:  
Bogumił Zalewski

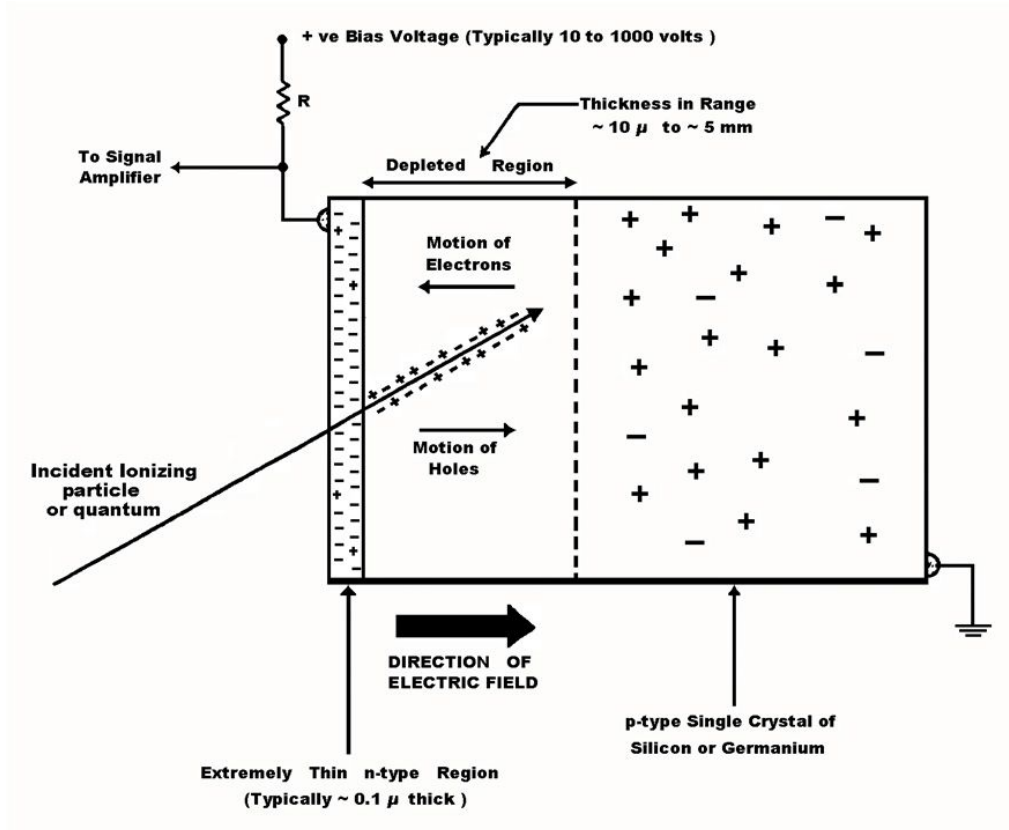


2 sides  
 $d = 1\text{mm}$

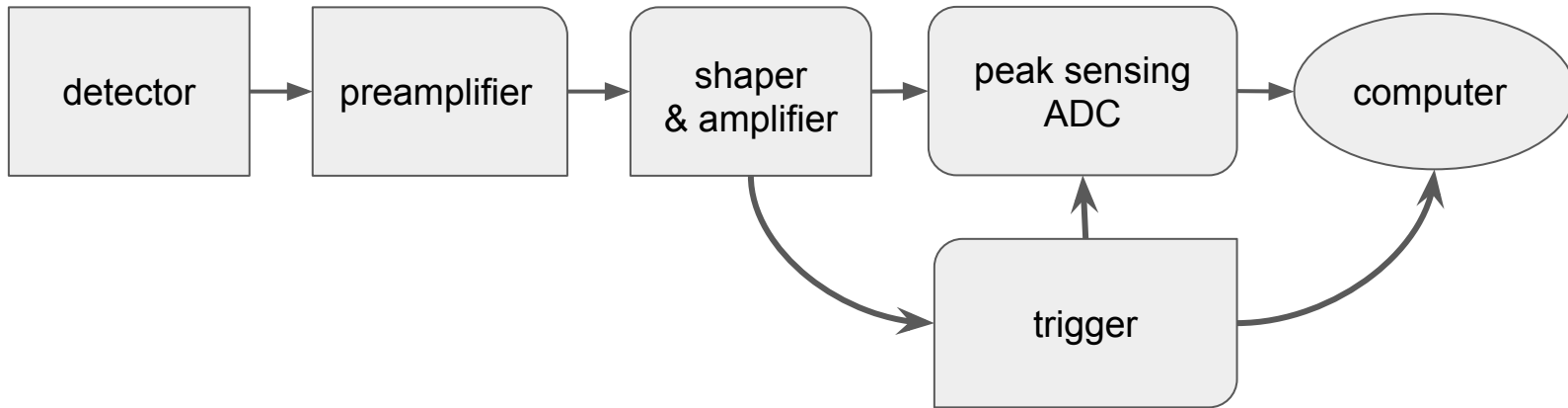
1 side  
 $d = 1\text{mm}$

1 side  
 $d \approx 67\mu\text{m}$

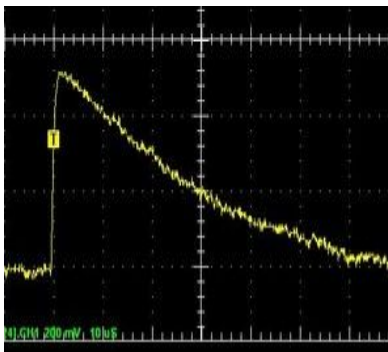
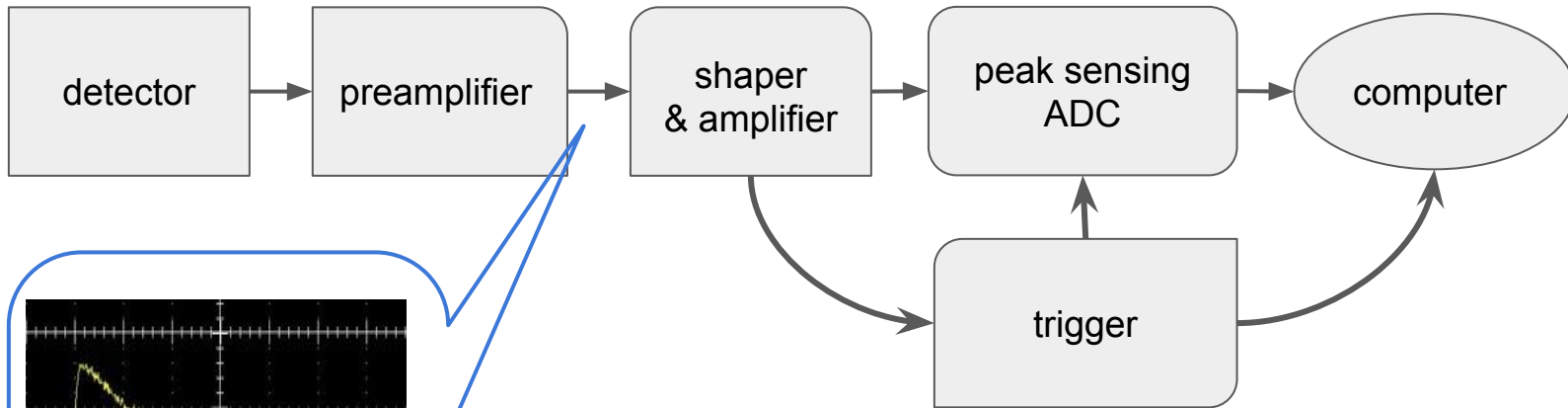
# Silicon detectors - principle of work



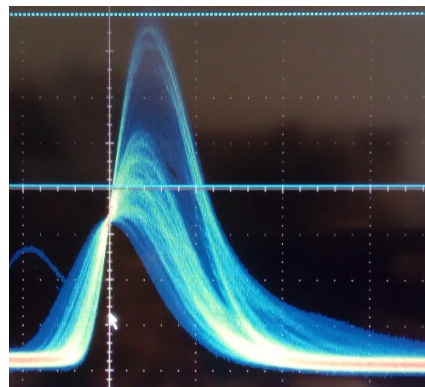
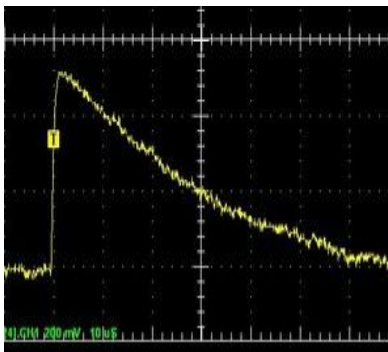
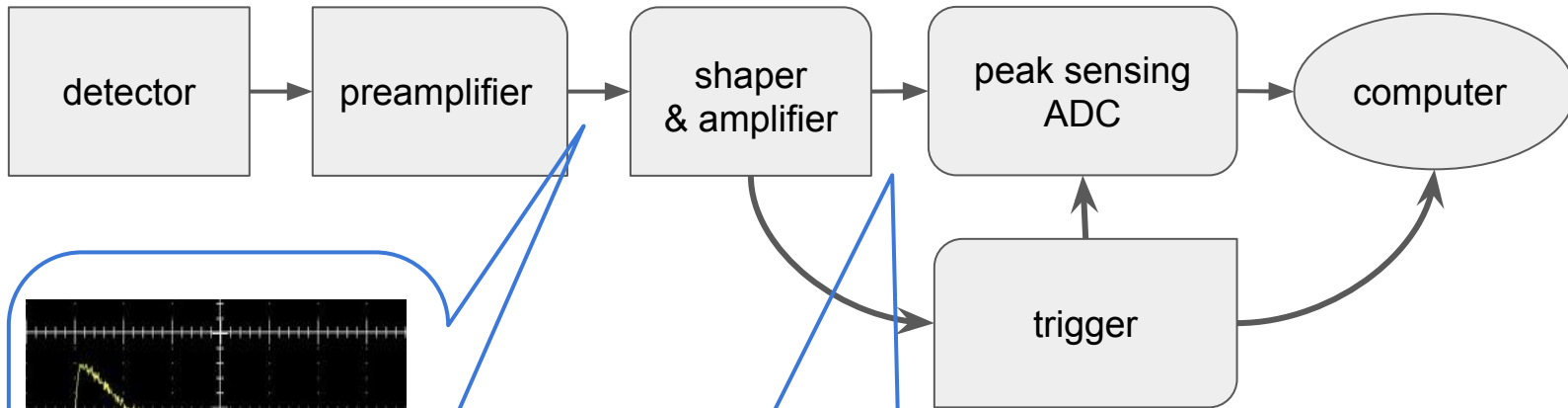
## Silicon detectors - electronic reading



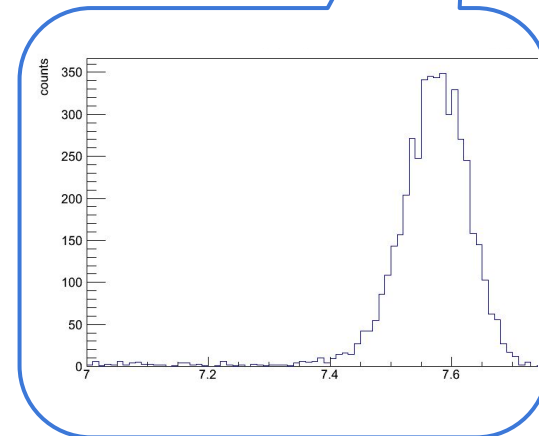
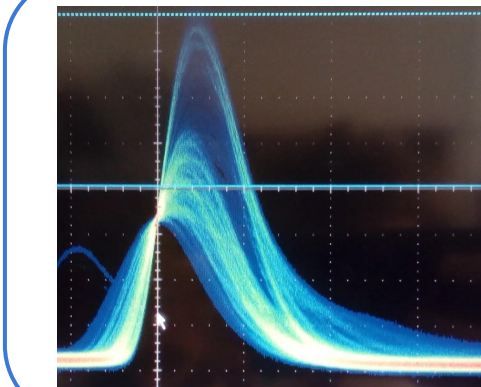
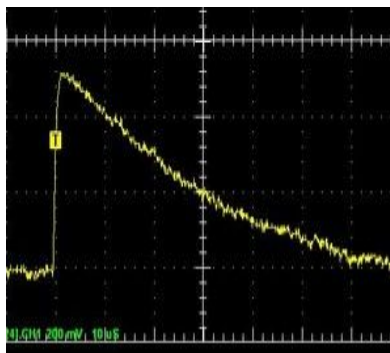
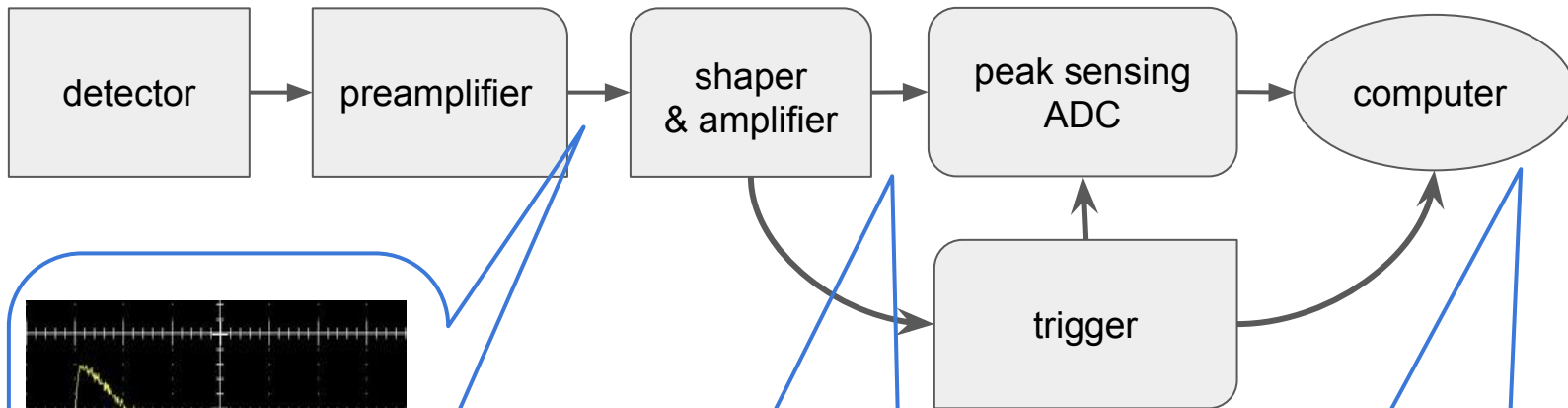
# Silicon detectors - electronic reading



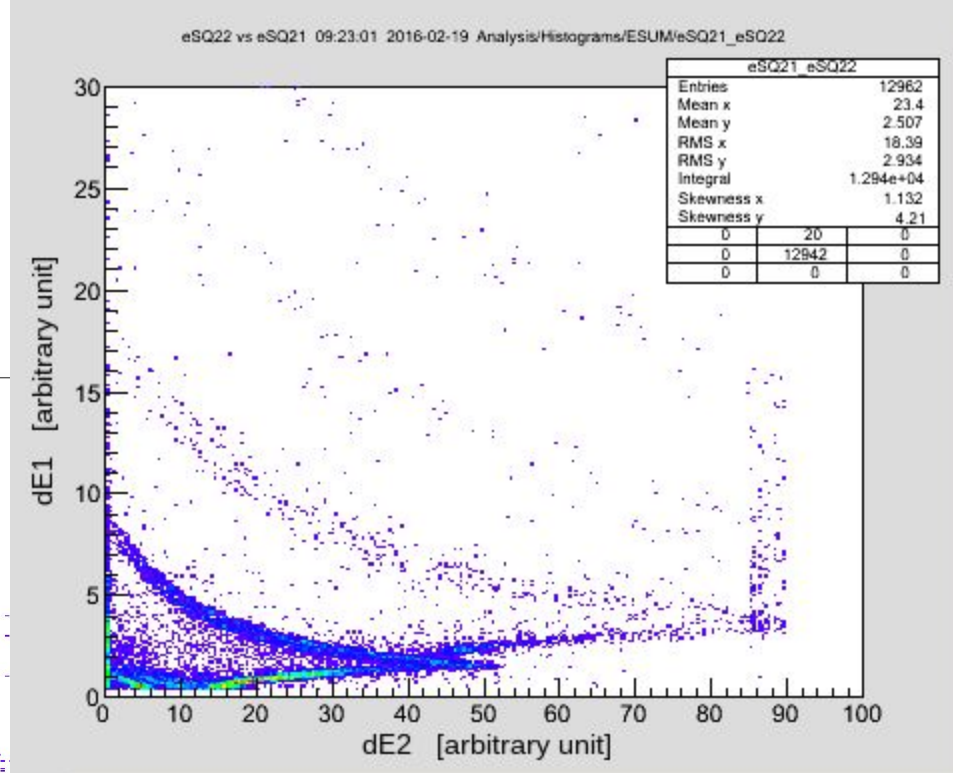
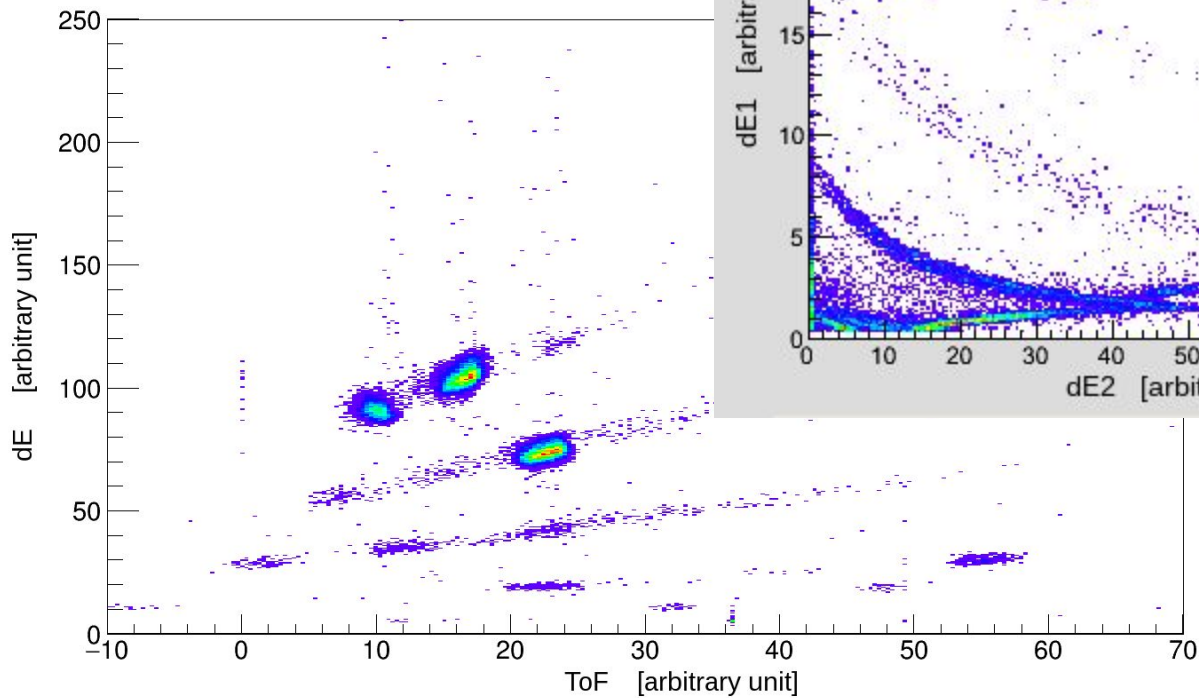
# Silicon detectors - electronic reading



# Silicon detectors - electronic reading

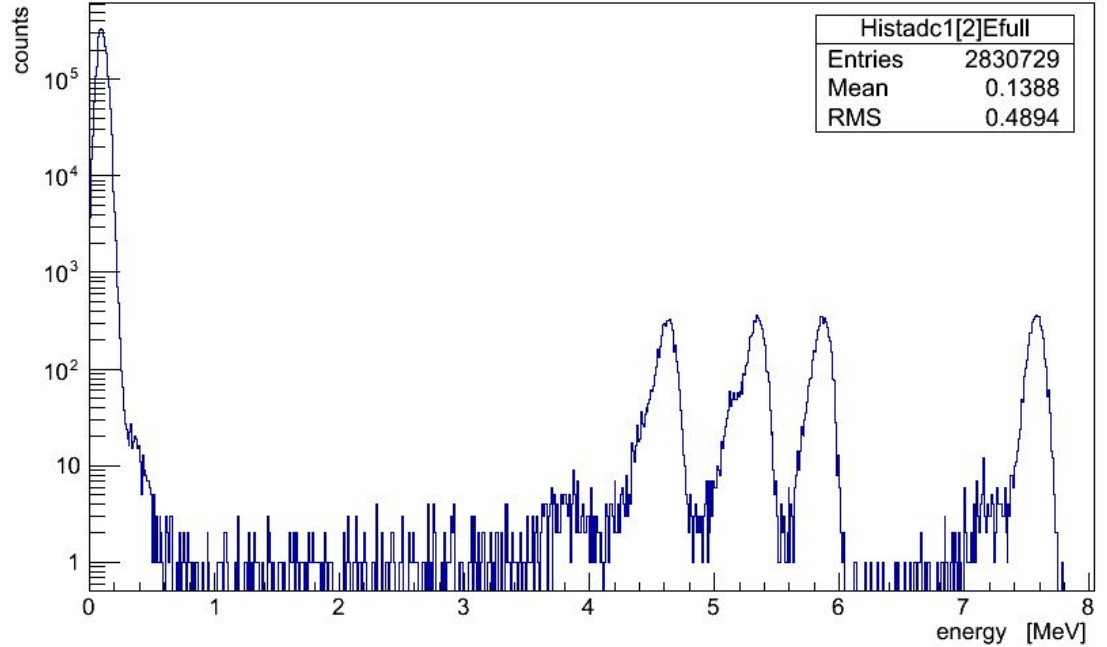
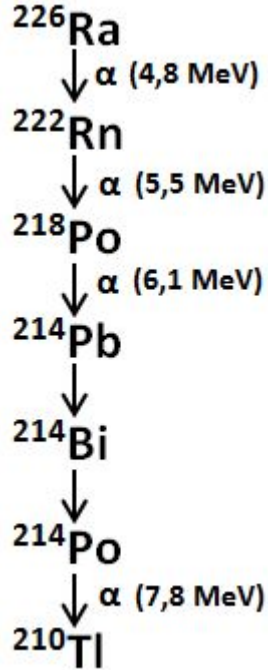


# Silicon detectors applications



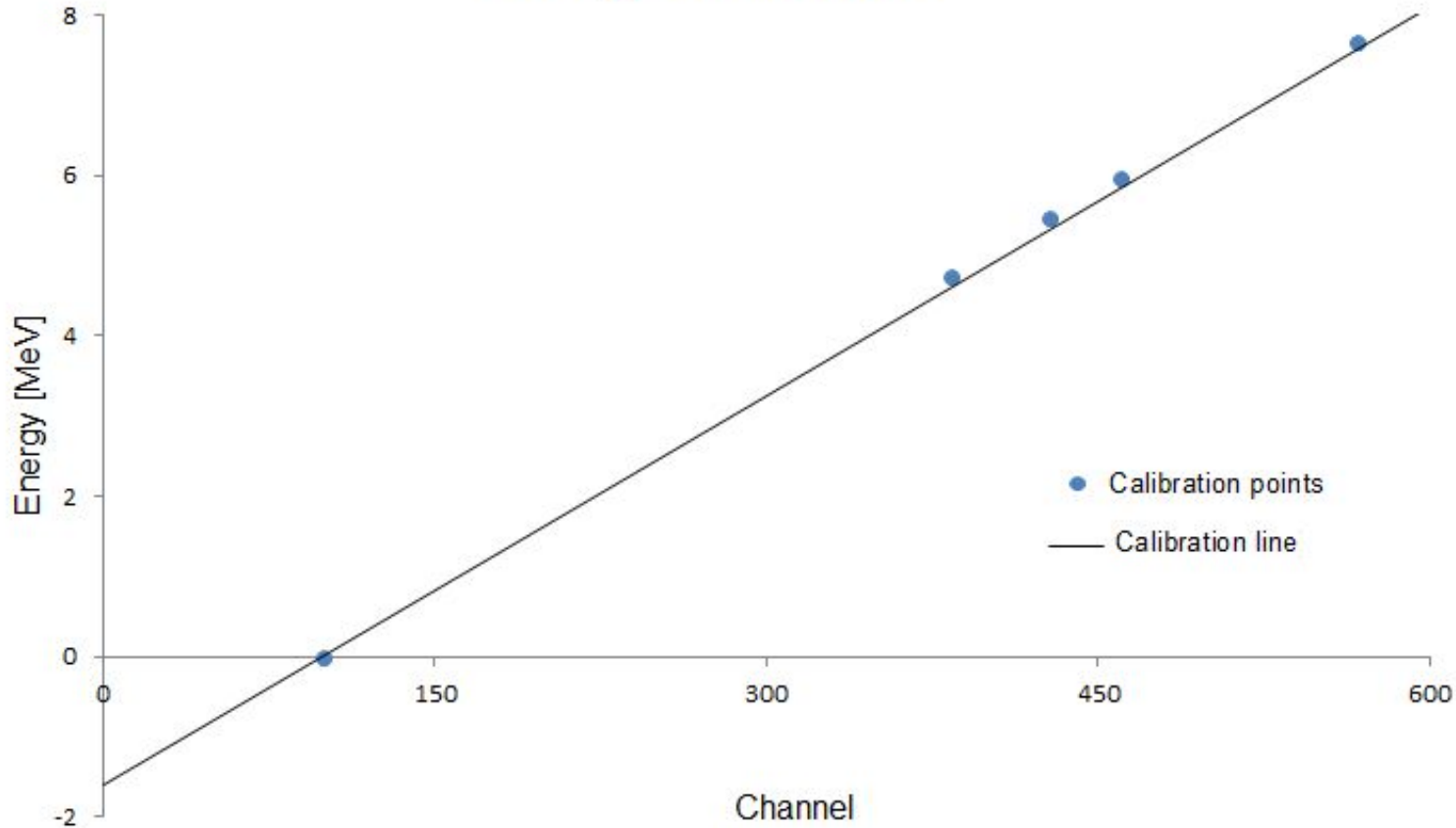


# Detection of alpha particles and pedestal

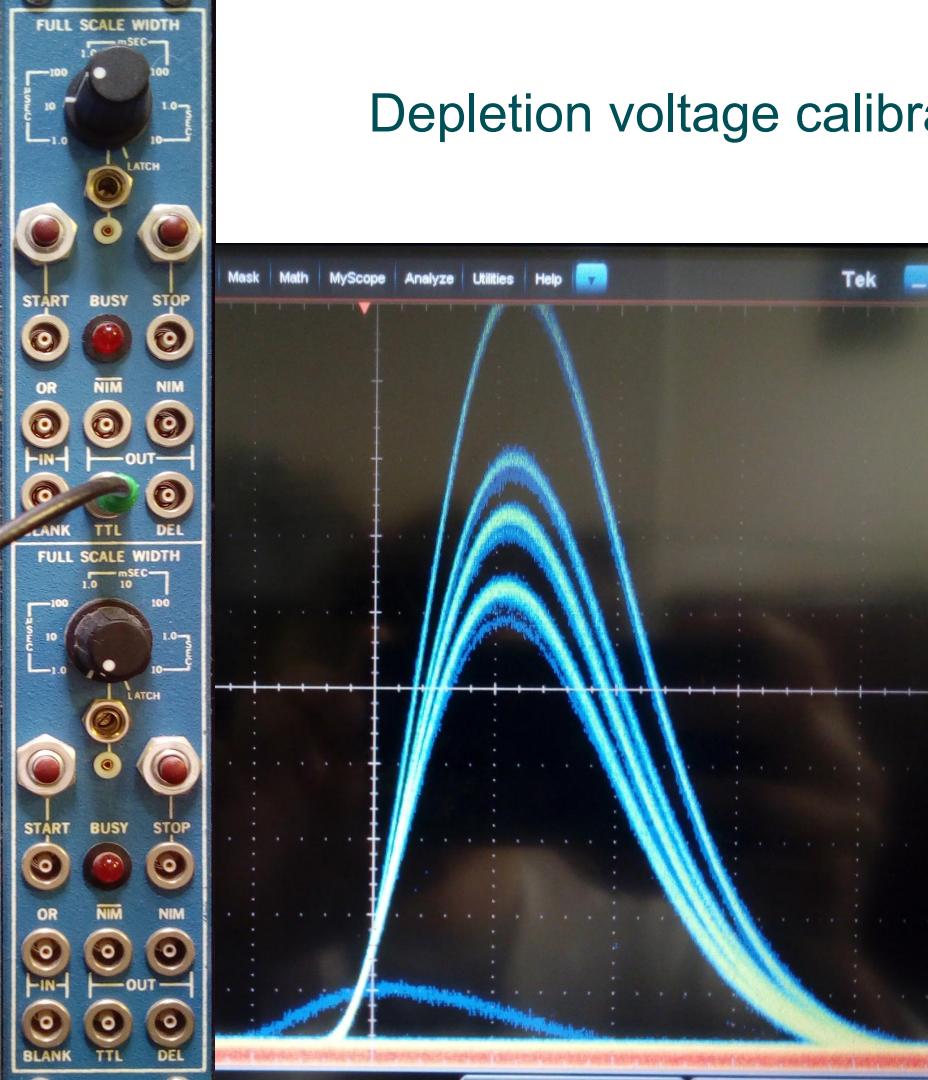




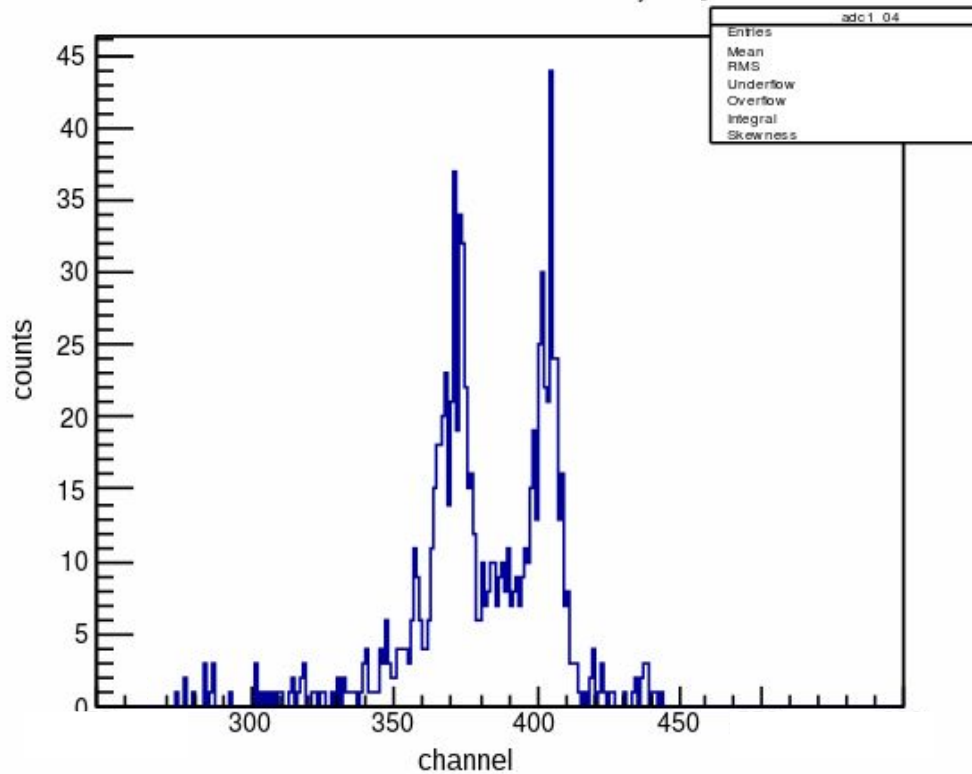
## Energy calibration



# Depletion voltage calibration



1 STILBENE DET CH 05 12:11:55 2017-07-17 Analysis/H\_04



# Automation of data processing



```
void generate_pedestals() {
    string input_file_name = "68_strip10001.root";
    string output_file_name = "pedestals.par";
    Int_t number_of_strips = 16; // 16 or 32
    Int_t max_channel = 200;
```

```
TTree *tree;
UShort_t LiEvent_adc1[32];
TBranch *b_LiEvent_adc1;
```

```
f_in = new TFile(input_file_name.c_str());
f_in->GetObject("AnalysisxTree", tree);
tree->SetMakeClass(1);
tree->SetBranchAddress("LiEvent_adc1[32]",
LiEvent_adc1, &b_LiEvent_adc1);
```

```
Long64_t nentries = tree->GetEntries();
```

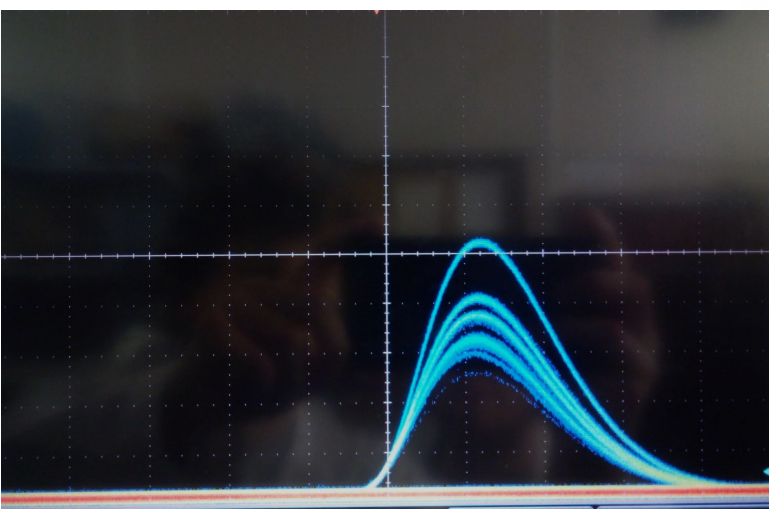
```
TH1I *hist[32];
for (int i(0); i < number_of_strips; i++) {
    string name = string("hist") + IntToString(i);
    hist[i] =
        new TH1I(name.c_str(), name.c_str(), max_channel,
0., max_channel);
}
```

```
for (Long64_t jentry = 0; jentry < nentries; jentry++) {
    if (jentry % (nentries / 20) == 0)
        cout << " progress: " << double(jentry * 100) / nentries
<< " %" << endl;
```

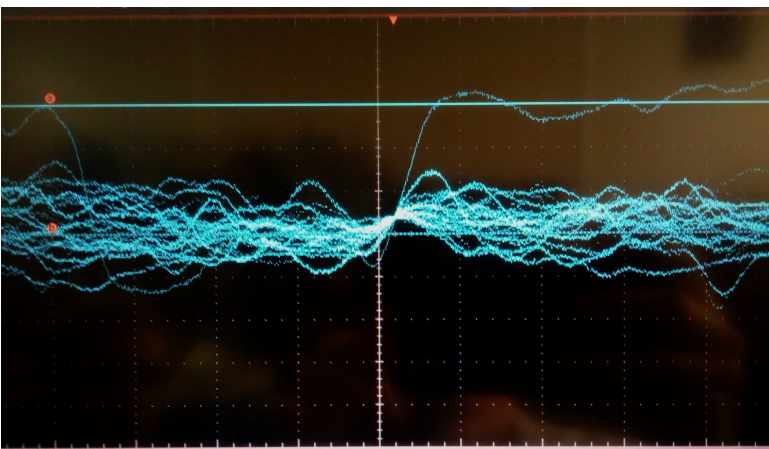
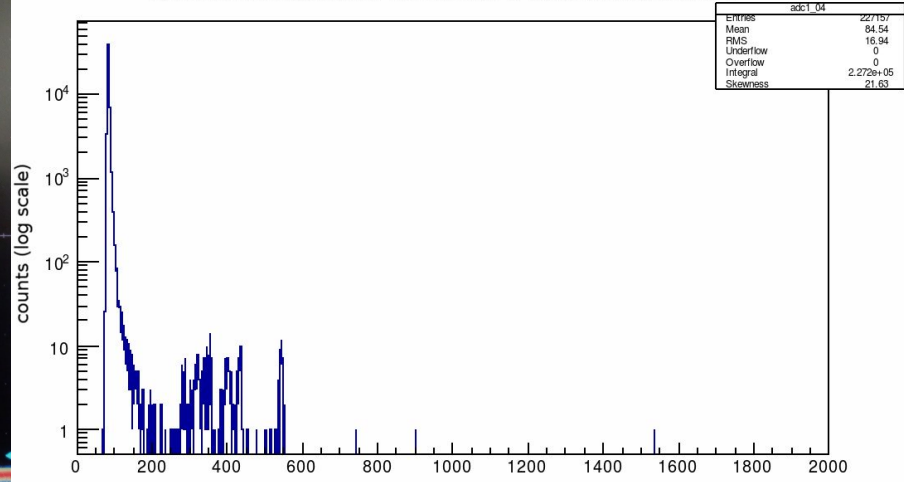
```
tree->LoadTree(jentry);
tree->GetEntry(jentry);
```

```
for (int i(0); i < number_of_strips; i++) {
    if (LiEvent_adc1[i] >= max_channel)
        continue;
    hist[i]->Fill(LiEvent_adc1[i]);
}
}
```

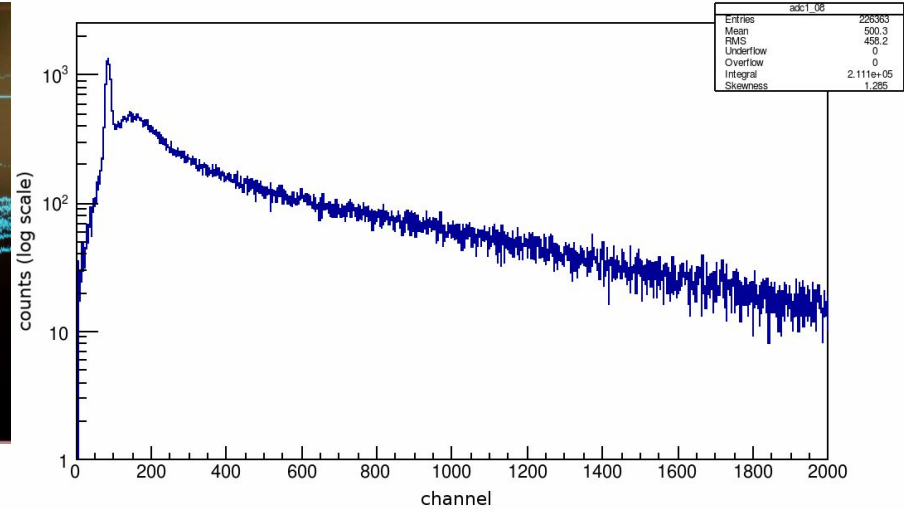
```
TF1 *fun[32];
Double_t *para[32];
for (int i(0); i < number_of_strips; i++) {
    string name = string("fun") + IntToString(i);
    fun[i] = new TF1(name.c_str(), "gaus", 0, max_channel);
    hist[i]->Fit(fun[i], "R");
    para[i] = new Double_t(3);
    fun[i]->GetParameters(para[i]);
}
}
```



AMPLITUDE OF ADC1 STILBENE DET CH 05 17:01:22 2017-07-12 Analysis/Histograms/Amp/adc1\_04

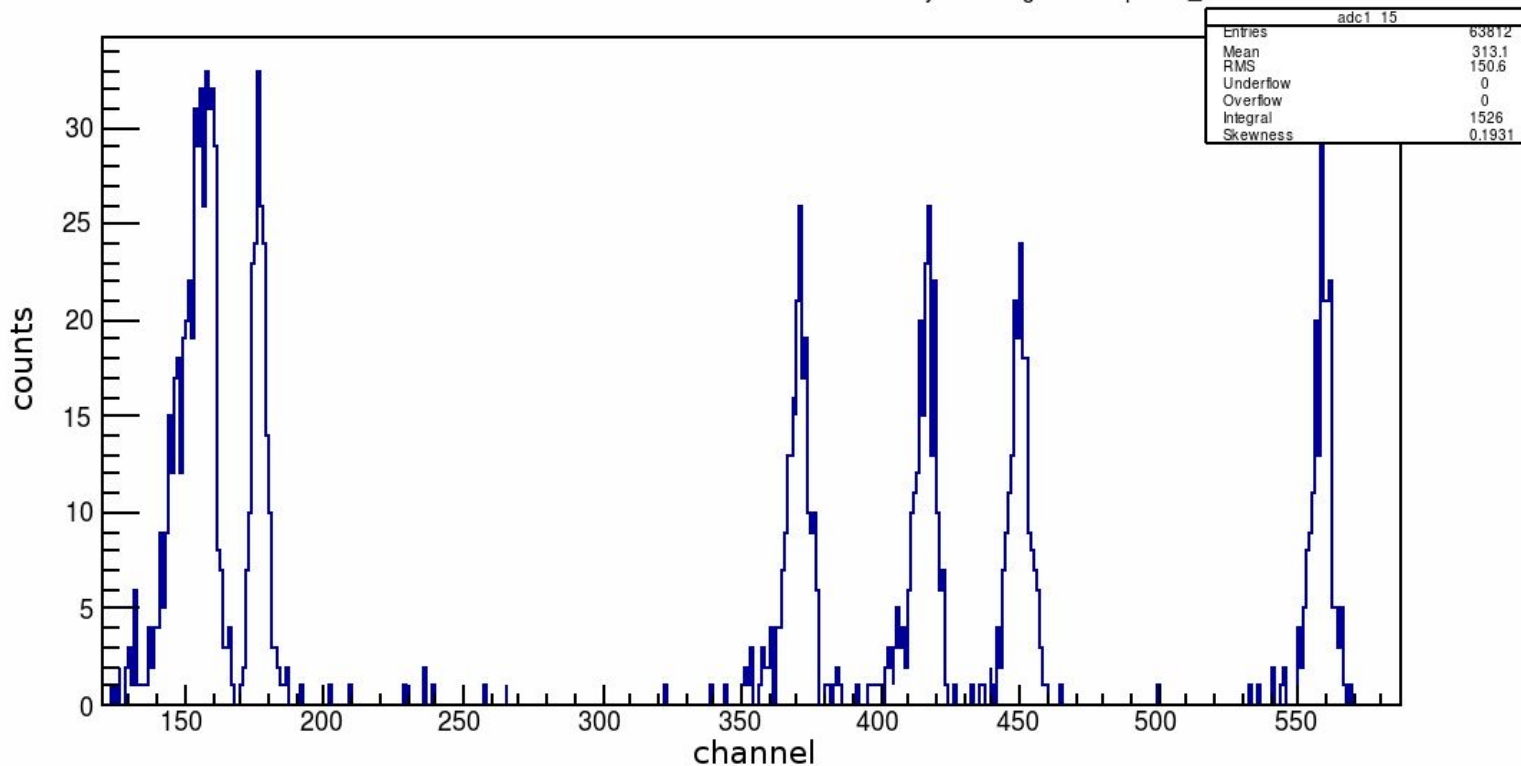


AMPLITUDE OF ADC1 STILBENE DET CH 09 17:00:38 2017-07-12 Analysis/Histograms/Amp/adc1\_08

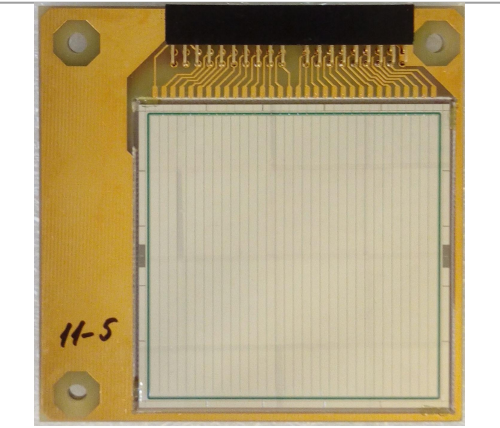


# Example of bad strips connection

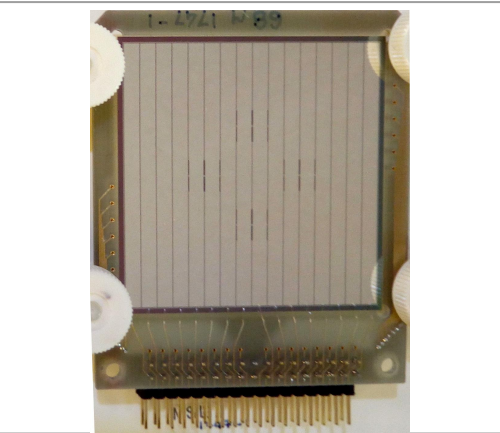
AMPLITUDE OF ADC1 STILBENE DET CH 16 11:28:09 2017-07-13 Analysis/Histograms/Amp/adc1\_15



# Example Results

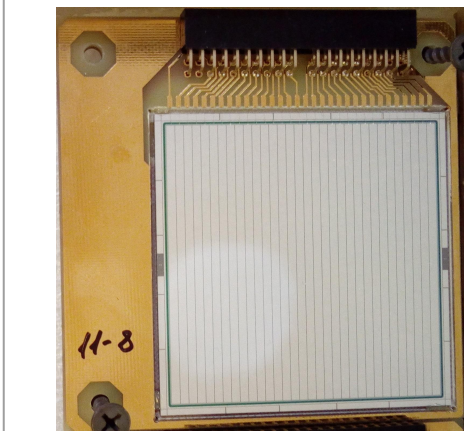
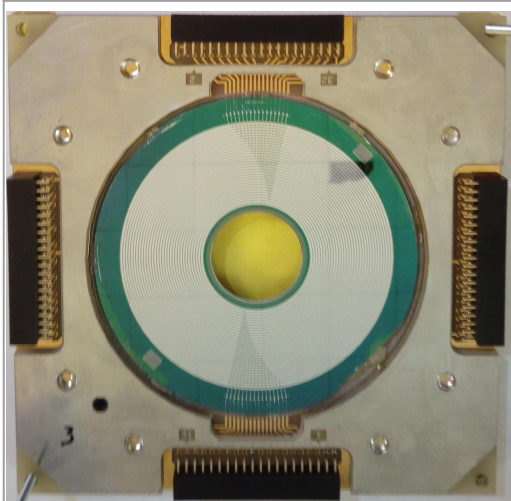


name	11_5
sizes of silicon wafer	64 x 64 x 1 mm
full depletion voltage	210 V
current	0.85 $\mu$ A
condition	strip 31 or 32 is broken



name	68
sizes of silicon wafer	50 x 50 x 0.068 mm
full depletion voltage	20 V
current	0.15 $\mu$ A
condition	good





name	4
thickness	1 mm
structural properties	2 sides: rings and sectors
full depletion voltage	150 V
current	4.8 $\mu$ A
condition	good
name	11_8
sizes of silicon wafer	64 x 64 x 1 mm
full depletion voltage	-
current	-
condition	strips 9,11,12,13,14,16 are broken





Thank you for your attention