

The original code that I used to count triggers is below.

```
l=importdata('0102_080216.txt'); %import veto data
L=l(:,1:8); % cut off timing values
C=de2bi(L); % binary representation of triggers
TrigCount=0; %initialize trig counts for 'actual' events
SingTrig=0; %initialize trig counts for single events

A=sum(C,2); %Sum rows of matrices
for k=1:40000;
    if A(k,:)==1; %logic statement- if the rows of A==1, then increment counter
        SingTrig=SingTrig+1;
    end
end

for j=1:29 %logic loop that counts 1001 triggers
    for k=1:40000
        if C(k,j) && C(k,j+3)==1
            TrigCount=TrigCount+1;
        end
    end
end

for p=1:31 %logic loop that counts 0110 1100 and 0011 triggers
    for k=1:40000
        if C(k,p) && C(k,p+1)==1
            TrigCount=TrigCount+1;
        end
    end
end

for k=1:40000; %setting summed row matrix values to 1 or 0 depending on
    if A(k,:)>1; % whether or not they were triggered.
        A(k,:)=1;
    end
end
B=sum(A); %total number of events
NonTrig=B-SingTrig-TrigCount; % nontriggerable events will be the difference between the total events
                             %the single events and the triggerable
                             %events.
```

This code had a problem with it that I'm still not quite sure about. While counting the nontriggerable events for the northeast corner at stretcher 1 mux 2, the counts came back negative. I knew it was a problem with the total trigger counts as this value was less than the value of the sum of the single triggers and the regular triggers. As a Result I decided to brute force the counting of the nontriggerable events as this was the easiest way I could think of.

```
l=importdata('0102_080222.txt'); %import veto data
L=l(:,1:8); % cut off timing values
C=de2bi(L); % binary representation of triggers
TrigCount=0; %initialize trig counts for 'actual' events
SingTrig=0; %initialize trig counts for single events
Nontrig=0;
```

```
A=sum(C,2); %Sum rows of matrices
for k=1:40000;
    if A(k,:)==1; %logic statement- if the rows of A==1, then increment counter
        SingTrig=SingTrig+1;
    end
end
```

```
for j=1:29 %logic loop that counts 1001 triggers
    for k=1:40000
        if C(k,j) && C(k,j+3)==1
            TrigCount=TrigCount+1;
        end
    end
end
```

```
for p=1:31; %logic loop that counts 0110 1100 and 0011 triggers
    for k=1:40000;
        if C(k,p) && C(k,p+1)==1
            TrigCount=TrigCount+1;
        end
    end
end
```

%loops that count various configurations of 1's and 0's other than 0110 or 1001 (ex 100000010000
%etc) These configurations are nontriggerable events.

```
for p=1:30;
    for k=1:4000;
        if C(k,p) && C(k,p+2)==1;
```

```

        Nontrig=Nontrig+1;
    end
end
end

for p=1:28;
    for k=1:4000;
        if C(k,p) && C(k,p+4)==1;
            Nontrig=Nontrig+1;
        end
    end
end

for p=1:27
    for k=1:4000
        if C(k,p) && C(k,p+5)==1;
            Nontrig=Nontrig+1;
        end
    end
end

for p=1:26;
    for k=1:4000;
        if C(k,p) && C(k,p+6)==1;
            Nontrig=Nontrig+1;
        end
    end
end

for p=1:25
    for k=1:4000
        if C(k,p) && C(k,p+7)==1;
            Nontrig=Nontrig+1;
        end
    end
end

for p=1:24;
    for k=1:4000;
        if C(k,p) && C(k,p+8)==1;
            Nontrig=Nontrig+1;
        end
    end
end

for p=1:23;
    for k=1:4000;
        if C(k,p) && C(k,p+9)==1;
            Nontrig=Nontrig+1;
        end
    end
end

```

```
for p=1:22;
    for k=1:4000;
        if C(k,p) && C(k,p+10)==1;
            Nontrig=Nontrig+1;
        end
    end
end
```

```
for p=1:21;
    for k=1:4000;
        if C(k,p) && C(k,p+11)==1;
            Nontrig=Nontrig+1;
        end
    end
end
```

```
for p=1:20;
    for k=1:4000;
        if C(k,p) && C(k,p+12)==1;
            Nontrig=Nontrig+1;
        end
    end
end
```

```
for p=1:19
    for k=1:4000
        if C(k,p) && C(k,p+13)==1;
            Nontrig=Nontrig+1;
        end
    end
end
```

```
for p=1:18;
    for k=1:4000;
        if C(k,p) && C(k,p+14)==1;
            Nontrig=Nontrig+1;
        end
    end
end
```

```
for p=1:17;
    for k=1:4000;
        if C(k,p) && C(k,p+15)==1;
            Nontrig=Nontrig+1;
        end
    end
end
```

```
for p=1:16;
    for k=1:4000;
        if C(k,p) && C(k,p+16)==1;
            Nontrig=Nontrig+1;
        end
    end
end
```

end

```
for p=1:15
    for k=1:4000
        if C(k,p) && C(k,p+17)==1;
            Nontrig=Nontrig+1;
        end
    end
end
```

```
for p=1:14;
    for k=1:4000;
        if C(k,p) && C(k,p+18)==1;
            Nontrig=Nontrig+1;
        end
    end
end
```

```
for p=1:13
    for k=1:4000
        if C(k,p) && C(k,p+19)==1;
            Nontrig=Nontrig+1;
        end
    end
end
```

```
for p=1:12;
    for k=1:4000;
        if C(k,p) && C(k,p+20)==1;
            Nontrig=Nontrig+1;
        end
    end
end
```

```
for p=1:11;
    for k=1:4000;
        if C(k,p) && C(k,p+21)==1;
            Nontrig=Nontrig+1;
        end
    end
end
```

```
for p=1:10;
    for k=1:4000;
        if C(k,p) && C(k,p+22)==1;
            Nontrig=Nontrig+1;
        end
    end
end
```

```
for p=1:9;
    for k=1:4000;
        if C(k,p) && C(k,p+23)==1;
```

```

        Nontrig=Nontrig+1;
    end
end
end

for p=1:8
    for k=1:4000
        if C(k,p) && C(k,p+24)==1;
            Nontrig=Nontrig+1;
        end
    end
end

for p=1:7
    for k=1:4000
        if C(k,p) && C(k,p+25)==1;
            Nontrig=Nontrig+1;
        end
    end
end

for p=1:6;
    for k=1:4000;
        if C(k,p) && C(k,p+26)==1;
            Nontrig=Nontrig+1;
        end
    end
end

for p=1:5;
    for k=1:4000;
        if C(k,p) && C(k,p+27)==1;
            Nontrig=Nontrig+1;
        end
    end
end

for p=1:4;
    for k=1:4000;
        if C(k,p) && C(k,p+28)==1;
            Nontrig=Nontrig+1;
        end
    end
end

for p=1:3;
    for k=1:4000;
        if C(k,p) && C(k,p+29)==1;
            Nontrig=Nontrig+1;
        end
    end
end

```

```

for p=1:2;
    for k=1:4000;
        if C(k,p) && C(k,p+30)==1;
            Nontrig=Nontrig+1;
        end
    end
end

for p=1:1;
    for k=1:4000;
        if C(k,p) && C(k,p+31)==1;
            Nontrig=Nontrig+1;
        end
    end
end
TotTrig=Nontrig+SingTrig+TrigCount; %total number of events
Nontrigperc=Nontrig/TotTrig;      %percentage of nontrig events
SingTrigperc=SingTrig/TotTrig;    %percentage of SingTrig events
TrigCountperc=TrigCount/TotTrig;  %percentage of triggerable events

```

As a check to see if this code was correct I tested it on a smaller batch of triggers and counted the triggers manually and with the code to see they would match. The counts produced by the count did match the manual counts.

The below data is very preliminary but I decided to take a look at the NE corner, specifically I decided to compare data between mux 1 and mux 2 on 08/02/13 between 15:00 and 22:00 as this date was given as a good data collection period.



