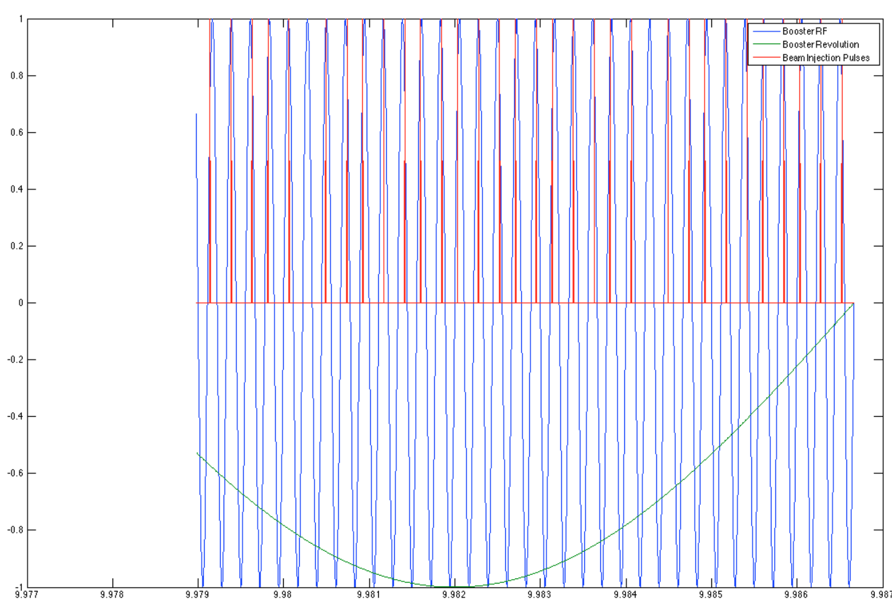
Introduction:
The Linac beam structure is based at the RFQ frequency of 162.5 MHz and its sub-harmonics as generated by the beam chopper in the MEBT. These frequencies are not harmonically related to the Booster or Recycler Ring RF frequencies, requiring a unique, non-repetitive chopping pattern derived in real-time. A non-repetitive pattern does not give time to properly pre-distorting the chopper waveform to correct for the distortions present in both the chopper and chopper amplifier.

Proposed Solution:
This approach turns the problem around by having the Linac Chopper Program Module (CPM) generate the target machine's RF signal and revolution marker and have the target machine frequency and marker lock (clock lock) to these CPM signals just before injection. These RF and marker signals will be generated in two new 2.6 GSPS ARB channels with a memory depth equal to the injection period. (10 msec for RR, 1msec for Booster) These two RF signals are sent to the target machine from the Linac CPM over fiber or coax.

The RF waveform tables are calculated from the target machine parameters of injection frequency, harmonic number, and RF bucket area, along with the Linac beam current and max beam frequency. The exact length or period of the ring buffer in memory is determined to be is exactly some "n" times the target revolution period so that these signals may play out continuously before beam transfer. Frequency resolution for the RF frequency is 17 Hz divided by the number of milliseconds of buffer length. A Matlab script is included in this document to show the details of the waveform calculations as shown in the plot below.



Operationally the CPM will first inhibit Linac beam, then play RF and revolution markers to the target machine followed by the target clock locking. After clock lock, the CPM will enable beam in the LEBT and play the pre-calculated and pre-distorted waveforms to the MEBT choppers. There are many advantages to this scheme including; producing a predetermined, predistorted waveform, removing the issue of metastable states from a real-time beam selector circuit as had been previously proposed, and providing the ability to exactly control the beam allotted to each target bucket though the injection process.

```
%% Project-X kicker pattern generator t
% The pattern is repeatable when  numRev * revPeriod = m * Tinc
% numRev/m = tInc/revPeriod  where numRev and m are integers
% for 1 ms long injection, numRev =535
% m = floor(numRev * revPeriod/tInc)
% revPeriodAdjusted= m*tInc/535
% once a close value is found the injection frequency may be changed by
% adjusting m with a frequency resolution in Frf of deltaF=1/m*Frf



clear all;

%Select target machine
RR=588;
Booster=84;

%TargetMachine=RR;
TargetMachine=Booster;


if TargetMachine==RR  %RR parameters
    harmonicNumber=588; % harmonic number for RR
    Frf=52.8114e6;      %injection frequency
    injPeriod=1e-3;     %beam injection period
end

if TargetMachine==Booster  % Booster parameters
    harmonicNumber=84;  %harmonic number for Booster
    Frf=45.0e6;         % injection frequency
    injPeriod=1e-3;     % beam injection period
end

sampleRate=2.6e9;
tInc=1/sampleRate; %2.6GHz sample rate and 2x linac frequency
sumInj=0;

% compute
Frev=Frf/harmonicNumber;
revPeriod=1/Frev;
numRev=floor(injPeriod/revPeriod);
m=floor(numRev*revPeriod/tInc);
```

```
revPeriodAdjusted= m*tInc/535;
FrevAdjusted=1/revPeriodAdjusted;
FrfAdjusted=FrevAdjusted*harmonicNumber;
harmonicPeriod=numRev*revPeriodAdjusted;
w=FrfAdjusted*2*pi;
wRev=FrevAdjusted*2*pi;
deltaF=1/m*Frf;
beamRate=sampleRate/162.5e6; %RFQ frequency

% generate waveform arrays
a=zeros(4,m);  % 1- time, 2- Frf, 3- Frev, 4- beam pattern
for x=1:m
    a(1,x)=x*tInc;
    a(2,x)=sin(x*tInc*w);
    a(3,x)=sin(x*tInc*wRev);
    if (mod(x,beamRate)==0)
        a(4,x)=1;
    end
    if (a(2,x)<0.707)   %allow for injection over +-45 deg of RF bucket
        a(4,x)=0;
    end
    if a(4,x)==1
         sumInj=1+sumInj;
    end
end

averageinj=sumInj/(injPeriod*162.5e6)

%%
n=m-2000;
figure(1);
plot(a(1,n:m),a(2,n:m),a(1,n:m),a(3,n:m),a(1,n:m),a(4,n:m)); % Frev
with time
```