NSLS-II TECHNICAL NOTE BROOKHAVEN NATIONAL LABORATORY	NUMBER **265**
AUTHOR: W. Lee	DATE **01/18/2018**
FXI Mis-steered Pink Beam	

Revision notes:

December 14, 2017

Changes in this version are:

1. Removed hard stop figure because the detailed design is not relevant to this technical note.
2. Based on new simulations, revised the hard stop limits on the CM Y motions. In particular, the simulations show that for the upstream Y stage, it is the lower limit that matters, while for the downstream Y stages, it is the upper limit that matters. Accordingly, we remove the unnecessary limits.

Introduction

The white beam mirror at the 18-ID Full-field Imaging (FXI) beamline serves to collimate the beam for the downstream monochromator and also to filter out undesired high energy x-rays. Depending on the selected mirror coating stripe, the reflected beam (pink beam) power can be about 1 kW. It is thus important to ensure that this pink beam is safely captured under all possible mirror misalignment/mis-steer conditions.

In most cases, the mis-steered pink beam issue is dealt with by 'mis-steered beam ray traces'. However, for FXI, this approach is only feasible by making overly conservative unrealistic assumptions on the origin of the pink beam on the mirror surface and the reflected angle. This is because ray traces generally do not capture the necessary relationship between the source point, a point on the mirror surface, and the reflected angle of the ray as required for a reflection to occur. For FXI, we cannot use the mis-steered beam ray trace approach because it would overly constrain our ability to operate. Instead, we tackle this issue by computer simulation. In a computer simulation, all the necessary spatial and angular relationships between the source points, the mirror reflecting surface parameter space and the reflected rays can be fully captured.

In particular, the specific recommendations per (PS-C-XFD-PRC-008, *Synchrotron and Bremsstrahlung Ray Trace Procedure*) are:

I. 1 mm minimum clearance between the synchrotron (includes pink) beam and internal surfaces of shielded beam transport tubes

II. 2 mm minimum clearance between the synchrotron (includes pink) beam and the edge of the aperture taper at the entrance

III. 3 mm minimum clearance between the synchrotron (includes pink) beam and the edge of a beam stop

This technical note will document the simulations to support the proper setting of hard stops on the white beam collimating mirror (CM) motions in order to ensure that the pink beam is safely captured

Simulation details

The key components in 18-ID-A, First Optical Enclosure (FOE) are given in Figure 1. In general, the coordinate system used at the beamline is with the beam travelling in the +Z direction and +Y is up and +X is outboard. But for the 2-dimensional computer simulations, the beam travels in the +X direction and +Y is up.

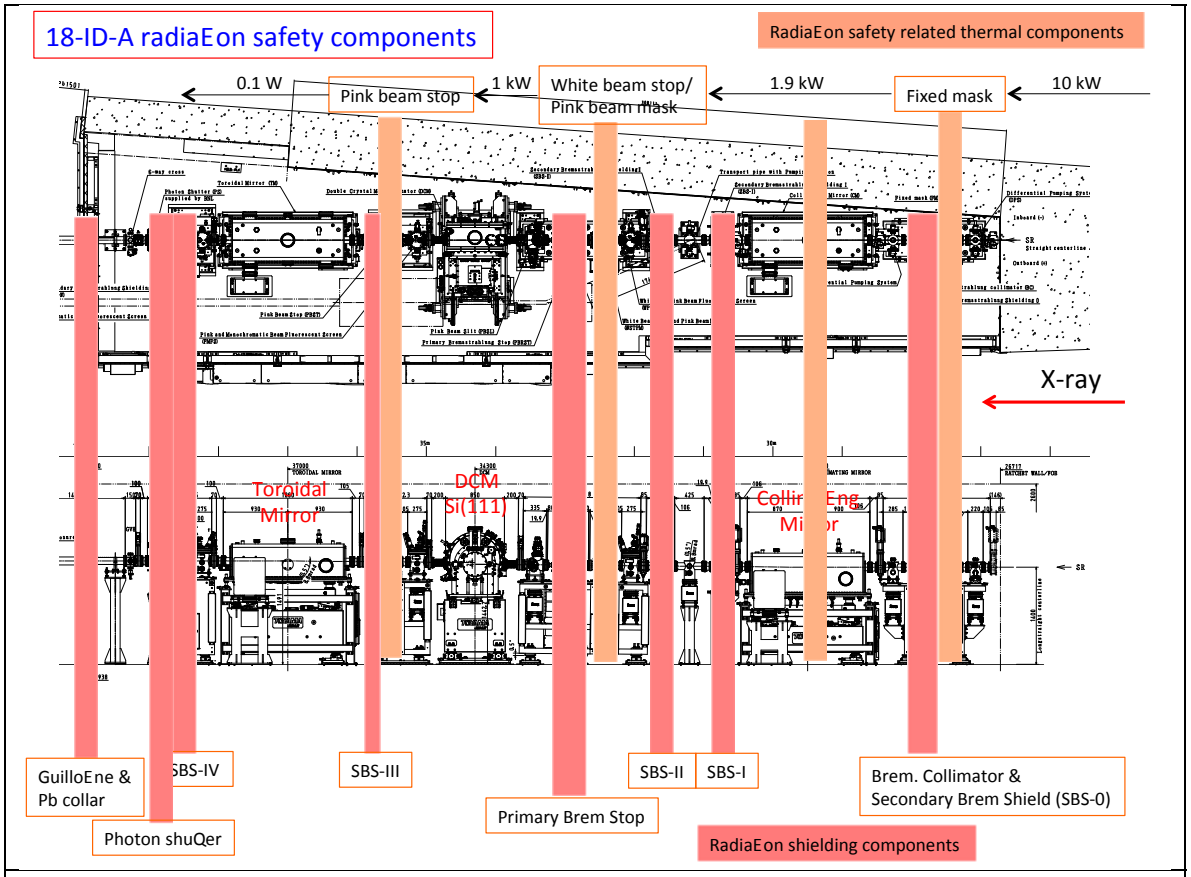


Figure 1: Key Radiation Safety Components in 18-ID-A

Figure 2 shows the details of the CM component. The mirror itself is a water-cooled 1450 mm long flat Silicon that will be bent to a 13.4 km radius. There are 3 user stripes on the mirror: Si, Cr and Rh. The mirror reflects up. The mirror is in vacuum and rests on two in-vacuum X motion stages to enable mirror yaw and X translation for access to the different mirror reflection stripes. Beneath these X stages is a metal in-vacuum base plate. 4 rods attached to the top of Granite B support this base plate. Thus, in the Y direction, the mirror is rigidly fixed to Granite B. There are 3 vertical stages between Granite A and Granite B, 2 downstream and 1 upstream. The two downstream vertical stages are symmetrically located about the beamline in the X direction with 350 mm separation between them. These 3 vertical stages enable the CM roll, pitch and vertical translation motions. They are of a 'wedge' design – vertical translation is achieved via a horizontal motion. The X:Y ratio of these stages are 4:1. These wedges are fixed on Granite A, which rests on 4 manual XY stages attached to the floor that are used for Granite A alignment. At issue here are the motion ranges for the 3 vertical stages. Note that only Y translation and the pitch of the mirror need be considered. The yaw plays no effect and the roll effect on mis-steered beam is negligible due to the small angles involved.

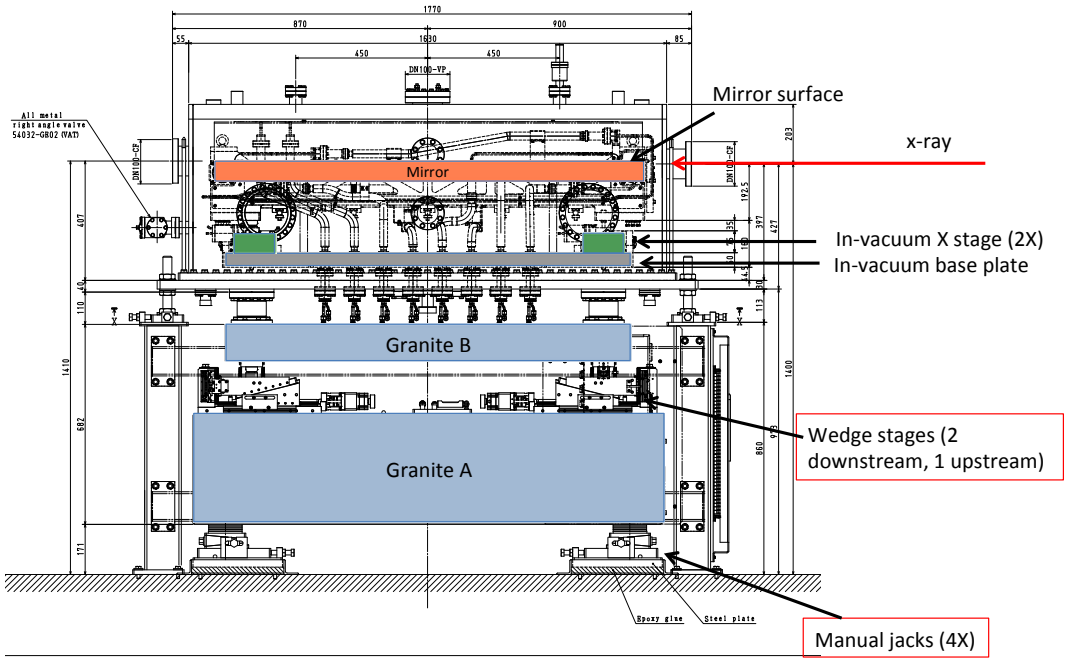


Figure 2: Key Parts of the CM

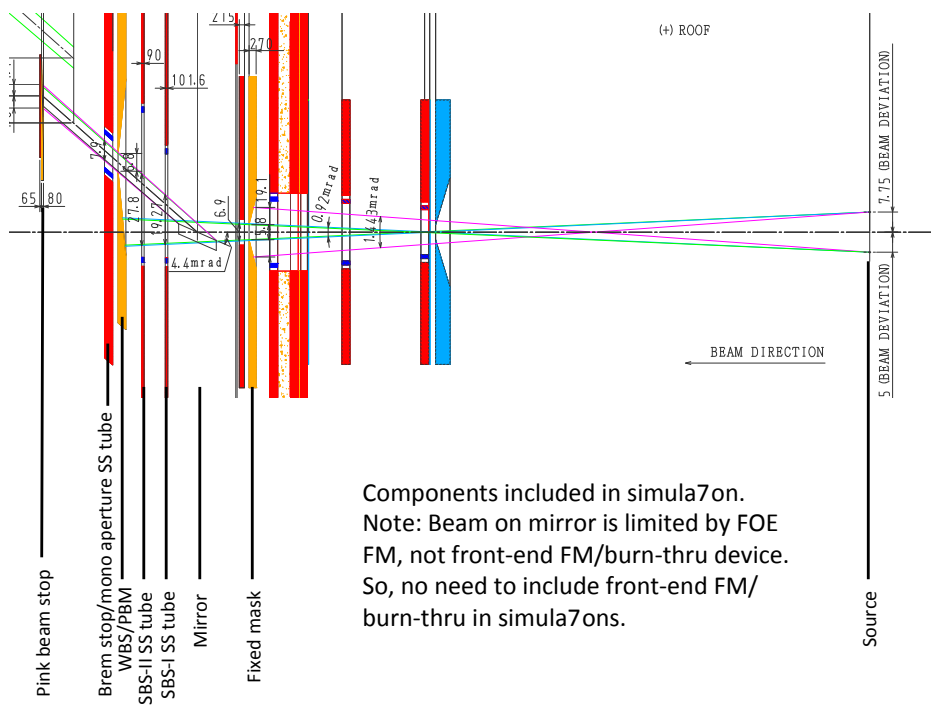


Figure 3: Magnified Region of the Ray Trace, Identifying Components Included in Simulation

Appendix A shows the vertical synchrotron ray trace. The fixed mask defines the white beam to the CM. There are two secondary bremsstrahlung shields (SBS) immediately downstream of the CM (SBS-I and SBS-II) and of relevance here are the stainless steel (SS) tubes through these shields. The WBS/PBM stops the white beam and allows the nominal pink beam to pass. The pink beam is stopped by the pink beam stop (PBS). A magnified region of the ray trace is shown in Figure 3 that identifies the components modeled in the simulation: (a) source, (b) fixed mask, (c) mirror, (d) SS tube for SBS-I, (e) SS tube for SBS-II, (f) white beam stop/pink beam mask, (g) SS tube for Bremsstrahlung stop and (h) pink beam stop.

Except for the mirror, all components are fixed and their simulation parameters are given in Appendix B. For the simulation, fully toleranced FM and WBS/PBM apertures are assumed. The mirror motion is modeled as two vertical motions, separated by 1200 mm along the beamline, and centered at 29500 mm from the center of the straight section. As we stated above, the roll of the mirror has negligible effect and thus, is omitted. Thus, the simulation is performed in 2D. The question is what should be the lower and upper limits of these two vertical mirror motions. The simulation program is given in Appendix C.

As a way to validate the computer program, we perform a simple 3 ray simulation (see Figure 4) and compare the center nominal ray position relative to the components with those from the ray traces. The results are shown in Table I.

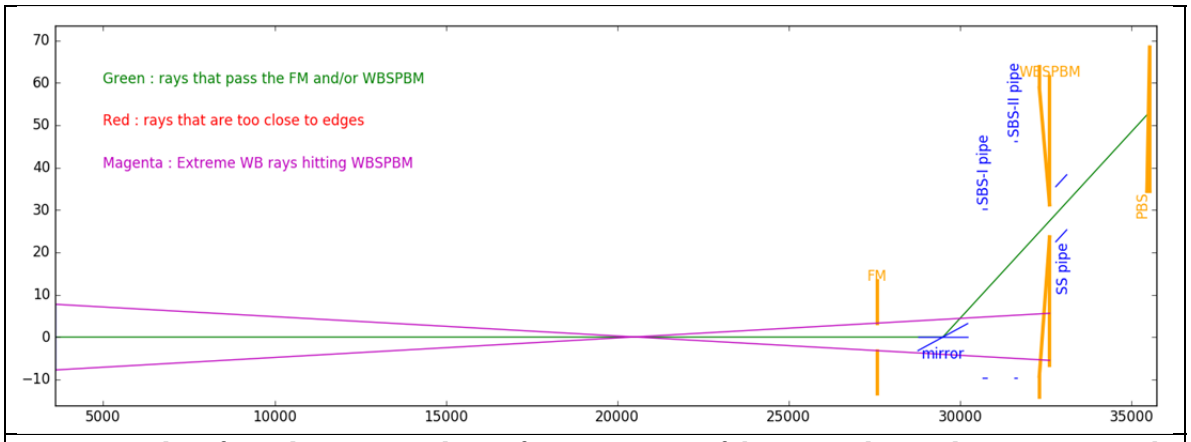


Figure 4: Plot of simple 3 ray simulation for comparison of the nominal central ray position with those from the ray traces

Table I: Validation of program by comparison of a few of the relevant positions between the simulations and ray traces. The small differences are due to rounding in the ray traces and simulation granularity.

	Simulation	Ray Trace
Ray to SBS-I pipe exit high side	19.09 mm	19.06 mm
Ray to SBS-II pipe exit high side	27.22 mm	27.21 mm
Ray to WBSPBM edge of incline at entrance	34.0 mm	34.0 mm
Ray to upper edge of SS tube entrance in Primary Brem Stop	6.52 mm	6.5 mm
Ray to top of Pink Beam Stop at entrance	15.98 mm	16.0 mm
Ray to top of Pink Beam Stop at exit	15.27 mm	15.3 mm

Simulation Results

Simulation results are summarized in Tables II. The simulation program is included in Appendix C and the program output for the two cases is also included.

Table II: Results for CM Y Vertical Motion Simulations

Case I		
Upstream CM Y vertical motion from zero	Restricted to -4.5 mm to +10 mm	
Downstream CM Y vertical motion from zero	Restricted to -10 mm to +4.5 mm	
Parameter	Value	Where
Minimum distance from rays to inside tube surface	1.25 mm	SS tube in Bremsstrahlung stop
Minimum distance from rays to edge of upstream face of aperture taper	3.73 mm	White beam at WBS/PBM
Minimum distance from rays to edge of beam stop	3.11 mm	Pink beam stop
Case II		
Upstream CM Y vertical motion from zero	Restricted to -5.2 mm to +10 mm	
Downstream CM Y vertical motion from zero	Restricted to -10 mm to +5.2 mm	
Parameter	Value	Where
Minimum distance from rays to inside tube surface	1.13 mm	SS tube in Bremsstrahlung stop
Minimum distance from rays to edge of upstream face of aperture taper	3.73 mm	White beam at WBS/PBM
Minimum distance from rays to edge of beam stop	2.42 mm	Pink beam stop

Based on the Table II (Case I), the upstream CM Y vertical motion shall be limited on the low side to -4.5 mm from zero, while the downstream CM Y vertical motion shall be limited on the high side to +4.5 mm from zero. As discussed in the next section, the calculation for +/- 5.2 mm values (Case II) are included to account for the worst case manufacturing, survey, hard-stop positional tolerances.

Plan for Setting Hard Limits

As described above, an important aspect of this is to ensure that the 'zero' of the CM is known. The attached survey report (Appendix D) explains how this is achieved. Prior to this, Granite A has already been surveyed in place and has not been moved. With the mirror in air, the mirror reflection surface is brought to its zero position by moving the vertical stages. Then, the fiducials on Granite B and Granite A are measured. The reference is the beam center line. These fiducial positions are thus directly linked the mirror position. With this, we can set the hard stops correctly. The specified survey tolerances are given in Table IV. With this procedure, the tolerances on Granite A fiducials do not affect the final mirror location tolerances.

Table IV: Survey tolerances for the CM

	Mirror surface	Granite B fiducials	Granite A fiducials
X	+/- 0.5 mm	+/- 0.5 mm	+/- 0.5 mm
Y	+/- 0.050 mm	+/- 0.15 mm	+/- 0.5 mm
Z	+/- 5 mm	+/- 5 mm	+/- 5 mm

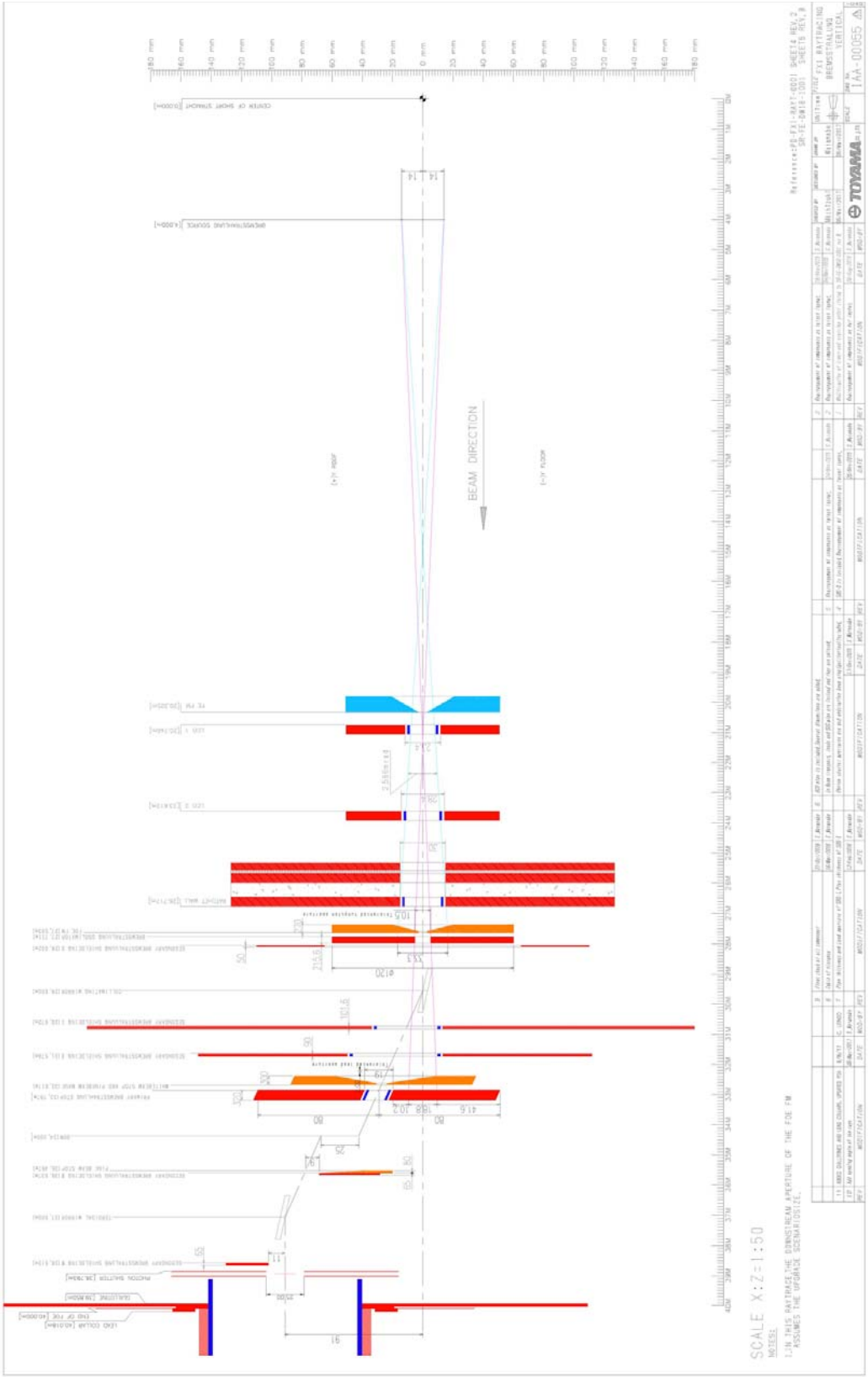
We assume that we can set the hard-stops to within a tolerance of 0.5 mm Y vertical motion for the stage. Thus, although we intend to set the hard stops to 4.5 mm Y vertical motion, the total error can be as large as 0.7 mm (0.5+0.15+0.05=0.7). That is the reason we also simulated the +/- 5.2 mm vertical motion case in the previous section.

In practice, to limit the vertical Y motion to +/- 4.5 mm +/- 0.5 mm, due to the X:Y=4:1 wedge ratio, this means that the hard stops themselves have to be set to +/-18 mm +/- 2 mm since the hard-stops function in the X direction. Setting the hard-stops to within 2 mm tolerance is easily achievable.

Summary

With the hard-stops set at -4.5 mm (from zero) for the upstream CM Y wedge stage, and +4.5 mm (from zero) for the downstream CM Y wedge stages, the pink beam is safely captured against all possible mirror mis-steers. No limits are needed for the high side (>0) of the upstream or the low side (<0) of the downstream CM Y stages.

Appendix A: Vertical Synchrotron Ray Trace



Appendix B: Component Parameters Used in Simulations
Coordinate system: Beam travels in +Z direction. +Y is up.
All Z locations relative to center of straight section
All Y locations relative to white beam line

Component	Value	Comment
Source Z	3590.5 mm	
Source size	15.5 mm	
Fixed mask exit Z	27583 mm	
Fixed mask aperture size	6.6 mm	Fully toleranced. Nominal is 5.8 mm
Collimating mirror Z	29500 mm	
Mirror length	1450 mm	
Distance between upstream and downstream Y stages	1200 mm	
WBS/PBM entrance Z	32317	WBS/PBM is tilted at 8.8 mrad
WBS/PBM exit Z	32617 mm	
WBS/PBM entrance Y	24.79 mm	
WBS/PBM exit Y	27.43 mm	
WBS/PBM tapered Y entrance size	68 mm	
WBS/PBM pink beam aperture size	7.6 mm	Fully toleranced. Nominal is 6.8 mm
SBS-I pipe center Z	30722.8	
SBS-I pipe center Y	10.3	
SBS-I pipe ID	40	
SBS-I pipe length	101.6	Length covered by Pb
SBS-II pipe center Z	31623	
SBS-II pipe center Y	18.3	
SBS-II pipe ID	56	
SBS-II pipe length	90	Length covered by Pb
Brem. Stop pipe center Z	32957	Brem stop pipe is tilted at 8.8 mrad
Brem. Stop pipe center Y	30.4	
Brem. Stop pipe ID	13	
Brem. Stop pipe length	320	Length covered by Pb
Pink beam stop DS Z	35537	
Upper Y edge of slope	68.4	
Lower Y edge of sloe	34.4	

Appendix C: Simulation Program

```
# -*- coding: utf-8 -*-
"""
Created on

This program is to simulate mirror mis-steered beam through the pi
beam pbm, the primary bremsstrahlung ss pipe and onto the pink bea
all dimensions in mm

@author: wklee
"""

from IPython import get_ipython
get_ipython().magic('reset -sf')

import numpy as np
import matplotlib.pyplot as plt
import time
from multiprocessing import Pool
from functools import partial

def raise_window(figname=None):
    if figname: plt.figure(figname)
    cfm = plt.get_current_fig_manager()
    cfm.window.activateWindow()
    cfm.window.raise_()


plt.close("all")


sourcex=3590.5  # source location measured from center of straight
sourcesize=15.5  # full size of source (pipecenterypipecenteryS)
sourcenpts=81  # npts used in for loop

fixedmaskx=27583 # fm exit location relative to center of straight
fixedmaskcentery=0
fixedmaskaperture=6.6  # 5.8 nominal
fixedmasklo=fixedmaskcentery-fixedmaskaperture/2
fixedmaskhi=fixedmaskcentery+fixedmaskaperture/2
print('Fixed mask aperture = {0} mm, located at {1} mm'.\
format(fixedmaskaperture,fixedmaskx))

mirrorx=29500  # mirror center from center of straight section
mirrorlength=1450  # length of mirror
d=1200.0  # distance between the mirror Y jacks
# following are mirror hardstops for the vertical jacks
# note that jacks are d apart and d is not necessarily the mirror
```

1

```
usmirrorhilimit=10
usmirrorlolimit=-5.2
dsmirrorhilimit=5.2
dsmirrorlolimit=-10

minmirroroffset=np.maximum(usmirrorlolimit,dsmirrorlolimit)
maxmirroroffset=np.minimum(usmirrorhilimit,dsmirrorhilimit)
mirrorthetamax=(dsmirrorhilimit-usmirrorlolimit)/d

mirrorthetanpts=81  # npts for the range of mirror theta in for lo
mirrorsurfacenpts=81  #npts over the mirror surface in for loop
mirroroffsetnpts=81 # npts over the mirror offset in for loop

print('Minmirroroffset = {0}, Maxmirroroffset = {1}, Mirrorthetama
format(minmirroroffset,maxmirroroffset,mirrorthetamax))


#  white beam stop & pink beam mask abbreviated as pbm
pbmx=32617  # pbm exit from center of straight section
pbmxentrance=32317  # pbm entrance position; pbm = 270 mm long
pbmy=27.43  # center of pbm from center line at exit
pbmyentrance=24.79  # center of pbm at entrance
pbmslopedlength=68 # total height of the sloped entrance surface
pbmsize=7.6 # 6.8 nominal
print('Pink beam mask aperture = {0} mm, Exit X = {1} mm, Exit Y =
format(pbmsize,pbmx,pbmy))


pbmlo=pbmy-pbmsize/2 # height of lower edge of pbm aperture at exi
pbmhi=pbmy+pbmsize/2


# edges of sloped surfaces at exit
pbmloedge=pbmy-pbmslopedlength/2  # bottom ofsloped edge
pbmhiedge=pbmy+pbmslopedlength/2

# edges of sloped surfaces at entrnace
pbmloedgeentrance=pbmyentrance-pbmslopedlength/2
pbmhiedgeentrance=pbmyentrance+pbmslopedlength/2

sbs1pipecentery=10.3  # upstream edge of lead, like in ray trace
sbs1pipeid=40  # assume 2 mm pipe thickness
sbs1pipecenterx=30722.8
sbs1pipelength=101.6  # 101.6 Pb only; 286 flange-flange
sbs1top=sbs1pipecentery+sbs1pipeid/2
print('SBS-I pipe length = {0} mm, Center X = {1} mm, Center Y = {
format(sbs1pipelength,sbs1pipecenterx,sbs1pipecentery))
```

2

```
sbs2pipecentery=18.3
sbs2pipeid=56
sbs2pipecenterx=31623
sbs2pipelength=90  # 90 Pb only; 200 flange to flange
sbs2top=sbs2pipecentery+sbs2pipeid/2
print('SBS-II pipe length = {0} mm, Center X = {1} mm, Center Y = 
format(sbs2pipelength,sbs2pipecenterx,sbs2pipecentery))

pipecenterx=32957  # center of the pipe from center of straight se
pipecentery=30.4  # center of pipe vertically to center line
pipelength=320  # 320 Pb only, 430 flange to flange
pipeid=13
piper=pipeid/2
pipeangle=8.8e-3  # angle of pipe in radians
pipeentrancex=pipecenterx-pipelength/2
pipeexitx=pipecenterx+pipelength/2
pipeentrancey=pipecentery-(pipelength/2)*pipeangle
pipeexity=pipecentery+(pipelength/2)*pipeangle
print('Primary Brem Stop pipe length = {0} mm, Center X = {1} mm, 
Center Y = {2} mm'.format(pipelength,pipecenterx,pipecentery))

pbstophi=68.4  # maximum height of pink beam stop
pbstoplo=34.4  # minimum height of the angled part of the pink bea
pbstopx=35537.0  # location of the downstream edge of the pink bea
pbstopxentrance=pbstopx-80  # pbstop glidcop is 80 mm long
print('Pink beam stop exit X = {0} mm, Y Low = {1} mm, Y Hi = {2} 
format(pbstopx,pbstoplo,pbstophi))
# desired minimum distance between beam and pipe id
pipebeammin=1.0
# desired minimum distance between ray and pink beam stop edges
pbstopbeammin=3.0
# desired minimum distance between ray and edge of sloped section 
pbmedgelimit=2.0

# height of the lower lead edge at pbm entrance
pipeloentrance=pipeentrancey-piper
# height of the lower lead edge at pbm exit
pipeloexit=pipeexity-piper
pipehientrance=pipeloentrance+pipeid
pipehiexit=pipeloexit+pipeid

"""
#xmin=mirrorx-mirrorlength/2-10
xmin=sourcex
xmax=pbstopx+10
ymin=mirroroffset-mirrorlength*mirrorthetamax/2-5
```

3

```
ymin=np.minimum(ymin,-sourcesize/2)
ymax=pbstophi+5
"""
"""
temp for mirrorthetamax
"""
#mirrorthetamax=5.25e-3

mintheta=mirrorthetamax
sourcerange=np.linspace(-sourcesize/2, sourcesize/2,sourcenpts)
xrange=np.linspace(-mirrorlength/2,mirrorlength/2,mirrorsurfacenpt
mirrorthetarange=np.linspace(0,mirrorthetamax,mirrorthetanpts)

# These are dummy initialization
hsbs1topipemin=100
hsbs2topipemin=100
hpbmhiedgemin=100
hpbmloedgemin=100
hpbmhiedgeminentrance=100
hpbmloedgeminentrance=100
hpipeloentrancemin=100
hpipehientrancemin=100
hpipeloexitmin=100
hpipehiexitmin=100
hpbstopexithimin=100
hpbstopexitlomin=100
hpbstopentrancelomin=100
hpbstopentrancehimin=100
minincident=4.4e-3


starttime=time.time()

for mirroroffset in np.linspace(minmirroroffset,maxmirroroffset,\
    mirroroffsetnpts):
    for sourcey in sourcerange:
        for x in xrange:  # distance from mirror center
            for mirrortheta in mirrorthetarange:  # mirror angle
                # check to see if mirror motion are within limits
                usmirrorh=mirroroffset-mirrortheta*(d/2)
                dsmirrorh=mirroroffset+mirrortheta*(d/2)

                if(usmirrorh>usmirrorhilimit)or(usmirrorh<usmirror
                    continue
                if(dsmirrorh>dsmirrorhilimit)or(dsmirrorh<dsmirror
                    continue
```

4

```
# incident beam hits mirror at point (mirrorx+x,tempoffset,
        tempoffset=x*mirrortheta+mirroroffset


# incident angle deviation from CL due to non-parallel inc.
        phi=(sourcey-tempoffset)/(mirrorx-sourcex+x)

        # equation of this ray is y=slope.x+c
        slope=-phi
        c=sourcey-slope*sourcex

        # check to see if this ray gets through the fixed r
        hfixedmask=slope*fixedmaskx+c
        if (hfixedmask>fixedmaskhi)or(hfixedmask<fixedmask

            continue



        # from here on, its reflected beams
        raym=2*mirrortheta+phi  #slope of mirror reflected
        rayc=tempoffset-raym*(mirrorx+x) #y intercept of ra



        # check for distances between beam and SBS I&II pip
        # for these, only need to look at exit location
        hsbs1=raym*(sbs1pipecenterx+sbs1pipelength/2)+rayc
        hsbs2=raym*(sbs2pipecenterx+sbs2pipelength/2)+rayc

        hsbs1topipe=sbs1top-hsbs1
        hsbs2topipe=sbs2top-hsbs2

        hsbs1topipemin=np.minimum(hsbs1topipe,hsbs1topipem
        hsbs2topipemin=np.minimum(hsbs2topipe,hsbs2topipem

        # beam height at pbm entrance
        h=raym*pbmxentrance+rayc
        hpbmhiedgeentrance=pbmhiedgeentrance-h
        hpbmloedgeentrance=h-pbmloedgeentrance
        hpbmhiedgeminentrance=np.minimum(hpbmhiedgeminentra
        hpbmhiedgeentrance)
        hpbmloedgeminentrance=np.minimum(hpbmloedgeminentra
        hpbmloedgeentrance)
```

```
# height of reflected rays at pbm exit
h=raym*pbmx+rayc

# rays that pass through pbm
if (h>pbmlo)and(h<pbmhi):

    # heights of beam at entrance and exit of pipe
    # inside the primary brem stop
    hentrance=raym*pipeentrancex+rayc
    hexit=raym*pipeexitx+rayc
    # beam height at pink beam stop exit
    hstopexit=raym*pbstopx+rayc
    # beam height at pink beam stop entrance
    hstopentrance=raym*pbstopxentrance+rayc
    # mintheta = lowest mirror incident angle for l

    mintheta=np.minimum(mintheta,mirrortheta+phi)


    # distances between rays and edge of pipe or pl
    hpipeloentrance=hentrance-pipeloentrance
    hpipeloexit=hexit-pipeloexit
    hpipehientrance=pipehientrance-hentrance
    hpipehiexit=pipehiexit-hexit
    hpbstophiexit=pbstophi-hstopexit
    hpbstoploexit=hstopexit-pbstoplo
    hpbstophientrance=pbstophi-hstopentrance
    hpbstoploentrance=hstopentrance-pbstoplo

    hpipeloentrancemin=np.minimum(hpipeloentrancem
    hpipeloentrance)
    hpipehientrancemin=np.minimum(hpipehientrancem
    hpipehientrance)
    hpipeloexitmin=np.minimum(hpipeloexitmin,hpipe
    hpipehiexitmin=np.minimum(hpipehiexitmin,hpipel

    hpbstopexithimin=np.minimum(hpbstopexithimin,\
    hpbstophiexit)
    hpbstopexitlomin=np.minimum(hpbstopexitlomin,\
    hpbstoploexit)
    hpbstopentrancehimin=np.minimum(hpbstopentranc
    hpbstophientrance)
    hpbstopentrancelomin=np.minimum(hpbstopentranc
    hpbstoploentrance)
    # rays that are too close to lower pipe edge
    if (hpipeloentrance<pipebeammin)or\
    (hpipeloexit<pipebeammin) :
```

6

```
                        print('h: low problem in pipe moffset= {0}
    theta = {2}, sourcey = {3},hpipeloentrance = {4}, hpipeloexit =
    .format(mirroroffset,x,mirrortheta,sourcey,hpipeloentrance,hpi

#
                        # rays that are too close to upper pipe ed
                    elif (hpipehientrance<pipebeammin)or\
                    (hpipehiexit<pipebeammin):
                        print('h: hi problem in pipe moffset= {0},
    theta = {2}, sourcey = {3},hpipehientrance = {4}, hpipehiexit =
    .format(mirroroffset,x,mirrortheta,sourcey,hpipehientrance,hpi


                        # rays that are too close to pink beam sto
                    elif (hpbstophiexit<pbstopbeammin)or\
                    (hpbstoploexit<pbstopbeammin):
                        print('problem at pink beam stop exit,\
 moffset= {0},x = {1},theta = {2}, sourcey = {3},hpbstophiexit = {
    .format(mirroroffset,x,mirrortheta,sourcey,hpbstophiexit,hpbst

                        # rays that are too close to pink beam sto
                    elif (hpbstophientrance<pbstopbeammin)or\
                    (hpbstoploentrance<pbstopbeammin):
                        print('problem at pink beam stop entrance
                        moffset= {0},x = {1},theta = {2}, sourcey =
                        hpbstophientrance = {4}, hpbstoploentrance
    .format(mirroroffset,x,mirrortheta,sourcey,hpbstophientrance,\
    hpbstoploentrance))

            # rays that are stopped by the pbm and are too close t
            # the sloped edges
                else:
                    if(hpbmhiedgeentrance<pbmedgelimit)or\
                    (hpbmloedgeentrance<pbmedgelimit):
                        print('h: h problem in pbm moffset= {0},\
                        x = {1},theta = {2}, sourcey = {3}, \
                        hpbmloedge = {4}, hpbmhiedge = {5}'\
                        .format(mirroroffset,x,mirrortheta,sourcey
                        hpbmloedgeentrance,hpbmhiedgeentrance))



"""
add extreme white beam lines through the fixed mask
"""

m=(sourcesize/2-fixedmasklo)/(sourcex-fixedmaskx)
```

7

```
c=(sourcesize/2)-(m*sourcex)
# at sbs1pipe entrance
h=m*(sbs1pipecenterx-sbs1pipelength/2)+c  # pipe entrance, like in
#print('white beam at sbs1 pipe low side = {} mm'.format(h))
print('white beam to sbs1 pipe low side entrance = {} mm'.\
format(h-(sbs1top-sbs1pipeid)))
# at sbs2pipe entrance
h=m*(sbs2pipecenterx-sbs2pipelength/2)+c # pipe entrance, like n ra
#print('white beam at sbs2 pipe low side = {} mm'.format(h))
print('white beam to sbs2 pipe low side entrance = {} mm'.\
format(h-(sbs2top-sbs2pipeid)))

# at sbs1pipe exit
h=m*(sbs1pipecenterx+sbs1pipelength/2)+c  # pipe entrance, like in
#print('white beam at sbs1 pipe low side = {} mm'.format(h))
print('white beam to sbs1 pipe low side exit = {} mm'.\
format(h-(sbs1top-sbs1pipeid)))
# at sbs2pipe exit
h=m*(sbs2pipecenterx+sbs2pipelength/2)+c # pipe entrance, like n ra
#print('white beam at sbs2 pipe low side = {} mm'.format(h))
print('white beam to sbs2 pipe low side exit = {} mm'.\
format(h-(sbs2top-sbs2pipeid)))


# at wbspbm
hlo=m*pbmxentrance+c
print('lower white beam to low pbm slope edge at entrance = {} mm'
format(hlo-(pbmyentrance-pbmslopedlength/2)))
# high lines
m=(-sourcesize/2-fixedmaskhi)/(sourcex-fixedmaskx)
c=(-sourcesize/2)-(m*sourcex)
hhi=m*pbmxentrance+c
print('upper white beam to low pbm slope edge at entrance = {} mm'
format(hhi-(pbmyentrance-pbmslopedlength/2)))


print('hsbs1topipemin={}'.format(hsbs1topipemin))
print('hsbs2topipemin={}'.format(hsbs2topipemin))
#print('hpbmhiedgemin={}'.format(hpbmhiedgemin))
#print('hpbmloedgemin={}'.format(hpbmloedgemin))
print('hpbmhiedgeminentrance={}'.format(hpbmhiedgeminentrance))
print('hpbmloedgeminentrance={}'.format(hpbmloedgeminentrance))
print('hpipeloentrancemin={}'.format(hpipeloentrancemin))
print('hpipehientrancemin={}'.format(hpipehientrancemin))
print('hpipeloexitmin={}'.format(hpipeloexitmin))
print('hpipehiexitmin={}'.format(hpipehiexitmin))
print('hpbstopexithimin={}'.format(hpbstopexithimin))
```

```
print('hpbstopexitlomin={}'.format(hpbstopexitlomin))
print('hpbstopentrancehimin={}'.format(hpbstopentrancehimin))
print('hpbstopentrancelomin={}'.format(hpbstopentrancelomin))

endtime=time.time()
worktime=endtime-starttime
print('Elapsed time={}'.format(worktime))
```

9

Program output

Table II, Case I

```
In [2]: runfile('/Users/OldMac/Google Drive/CMMisteerSimulation_NoPLot_v2.py', wdir='/Users/OldMac/Google Drive')
Fixed mask aperture = 6.6 mm, located at 27583 mm
Minmirroroffset = -4.5, Maxmirroroffset = 4.5, Mirrorthetamax = 0.0075
Pink beam mask aperture = 7.6 mm, Exit X = 32617 mm, Exit Y = 27.43 mm
SBS-I pipe length = 101.6 mm, Center X = 30722.8 mm, Center Y = 10.3 mm
SBS-II pipe length = 90 mm, Center X = 31623 mm, Center Y = 18.3 mm
Primary Brem Stop pipe length = 320 mm, Center X = 32957 mm, Center Y = 30.4 mm
Pink beam stop exit X = 35537.0 mm, Y Low = 34.4 mm, Y Hi = 68.4 mm
white beam to sbs1 pipe low side entrance = 4.977328331770343 mm
white beam to sbs2 pipe low side entrance = 4.560060435552778 mm
white beam to sbs1 pipe low side exit = 4.930535375638218 mm
white beam to sbs2 pipe low side exit = 4.518609982286133 mm
lower white beam to low pbm slope edge at entrance = 3.7297061581744284 mm
upper white beam to low pbm slope edge at entrance = 14.690293841825573 mm
hsbs1topipemin=6.5794658934336105
hsbs2topipemin=8.756572341744729
hpbmhiedgeminentrance=11.216319753365873
hpbmloedgeminentrance=5.509724284778336
hpipeloentrancemin=2.2195837761073207
hpipehientrancemin=2.167068151152627
hpipeloexitmin=1.3225188353699693
hpipehiexitmin=1.254068687010097
hpbstopexithimin=3.1075279684205555
hpbstopexitlomin=6.742465221043695
hpbstopentrancehimin=4.041033446721002
hpbstopentrancelomin=6.262731456228046
Elapsed time=213.2869668006897
```

Table II, Case II

```
In [5]: runfile('/Users/OldMac/Google Drive/CMMisteerSimulation_NoPLot_v2.py', wdir='/Users/OldMac/Google Drive')
Fixed mask aperture = 6.6 mm, located at 27583 mm
Minmirroroffset = -5.2, Maxmirroroffset = 5.2, Mirrorthetamax = 0.008666666666666666
Pink beam mask aperture = 7.6 mm, Exit X = 32617 mm, Exit Y = 27.43 mm
SBS-I pipe length = 101.6 mm, Center X = 30722.8 mm, Center Y = 10.3 mm
SBS-II pipe length = 90 mm, Center X = 31623 mm, Center Y = 18.3 mm
Primary Brem Stop pipe length = 320 mm, Center X = 32957 mm, Center Y = 30.4 mm
Pink beam stop exit X = 35537.0 mm, Y Low = 34.4 mm, Y Hi = 68.4 mm
white beam to sbs1 pipe low side entrance = 4.977328331770343 mm
white beam to sbs2 pipe low side entrance = 4.560060435552778 mm
white beam to sbs1 pipe low side exit = 4.930535375638218 mm
white beam to sbs2 pipe low side exit = 4.518609982286133 mm
lower white beam to low pbm slope edge at entrance = 3.7297061581744284 mm
upper white beam to low pbm slope edge at entrance = 14.690293841825573 mm
hsbs1topipemin=3.505118016934194
hsbs2topipemin=3.591914826508784
hpbmhiedgeminentrance=4.53487896941332
hpbmloedgeminentrance=5.550543786853026
hpipeloentrancemin=2.1856666243843605
hpipehientrancemin=2.124527968893723
hpipeloexitmin=1.2278983366091651
hpipehiexitmin=1.1324184367705996
hpbstopexithimin=2.4255901000894653
hpbstopexitlomin=6.179913914974158
hpbstopentrancehimin=3.377617483120247
hpbstopentrancelomin=5.715648937176958
Elapsed time=181.29479908943176
```

Explanation of output:
The program prints out the input parameters and all the desired information pertaining to the requirements as stated in the Introduction. In particular, if one of these requirements is violated, it prints out the affected component (eg 'problem at pink beam stop exit') and some of the relevant information relating to the offending ray. In Case II, there is an issue because the extremal ray on the pink beam stop is < 3 mm from the edge (see 5th line from bottom of output). In the above output, we temporarily commented out the print statement for 'problem at pink beam stop exit' simply because there were too many of these statements to show here. For the simulation parameters here, the closest the pink beam gets to the edge of the pink beam stop (see above: hpbstophiexit) is 2.42 mm.

Appendix D: CM (and TM) survey report

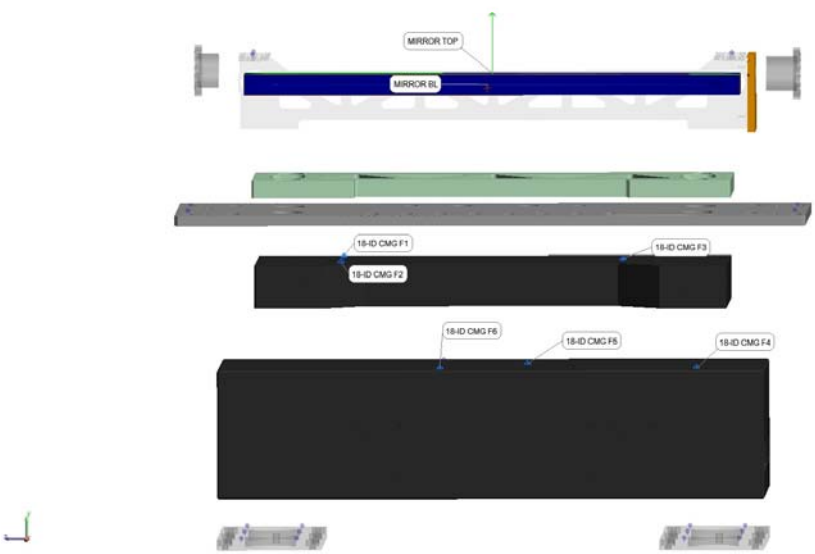
Collimating Mirror And Toroidal Focusing Mirror Survey Result At 18-ID FXI

Measurement Date: 07/20/2017
Measured By : D. Davis, R. Hubbard, F. Karl
Reported By: M. Ke

The reference control is updated in May 2017.

Leica AT401 was used to measure the control and the components.

PRACTICAL FRAME is created in the following manner:
Origin is the center of the cell 18 straight;
+Z is from Origin to the actual downstream aperture center of the front end fixed mask with -2.032 mm offset along X direction ;
+Y is up;
+X is outboard.



Plane A HUTCH MIR 1::MIR TOP				
	A	B	C	D
(mm)	0.000027	1.000000	0.000009	-0.263749
Proj. Ang. (deg)	Rx from Y 0.0005	Ry from Z 72.0402	Rz from X 89.9985	

CM top surface

Y = 0.006 mm

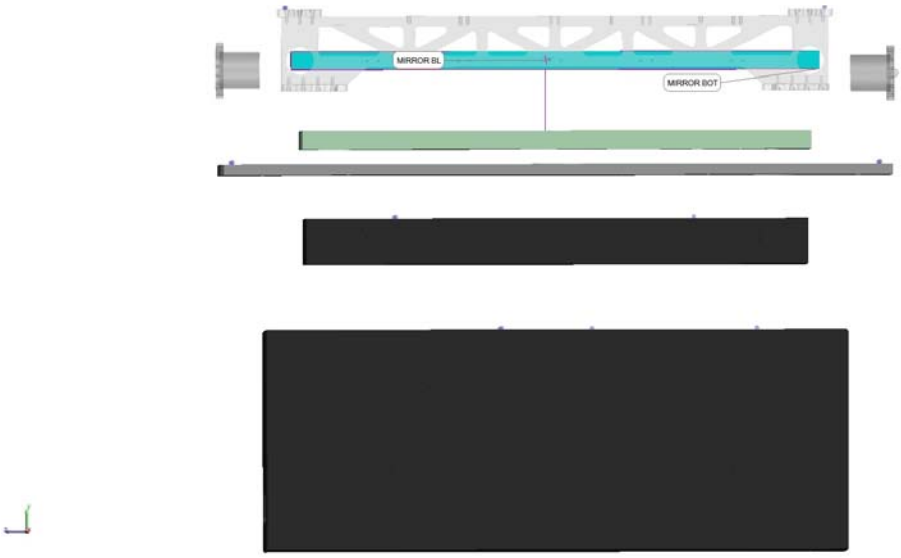
Plane A HUTCH MIR 1::MIR BL				
	A	B	C	D
(mm)	0.999999	-0.001447	-0.000006	-50.116840
Proj. Ang. (deg)	Rx from Y -179.7436	Ry from Z 90.0004	Rz from X -0.0829	

CM outboard face

X = 50.237 mm

Point Group A::COL MIRROR GRANITE FIDS			
Point Name	X (mm)	Y (mm)	Z (mm)
18-ID CMG F1	-150.037	-542.284	29939.524
18-ID CMG F2	149.908	-542.261	29939.594
18-ID CMG F3	-0.131	-542.271	29119.627
18-ID CMG F4	209.882	-848.732	28899.596
18-ID CMG F5	-0.114	-848.677	29399.734
18-ID CMG F6	210.135	-848.750	29649.904

CM fiducials on granite



Plane SET MIR::BOT MIR				
	A	B	C	D
(mm)	0.000531	-1.000000	-0.000122	93.667511
Proj. Ang. (deg)	Rx from Y -179.9930	Ry from Z 102.9515	Rz from X -89.9696	

TFM bottom flat face

Y = 89.149 mm

Plane SET MIR::BL MIR				
	A	B	C	D
(mm)	0.999998	0.002170	-0.000044	-36.042819
Proj. Ang. (deg)	Rx from Y -1.1740	Ry from Z 90.0025	Rz from X 0.1244	

TFM outboard face

X = 37.428 mm