

Prepar3D Sensor Gauge

By Robert Graham and Peter Story

Beta Release.

legal and Warning etc.

This is the normal legal and intro etc segment, by using this gauge you understand that you can not hold Robert Graham or Peter Story responsible for any damages, issues or loss that may occur. If you do not agree with the fact that you are waving your right to sue etc then just don't use the gauge no one is forcing you to.

Secondly this gauge is supplied as is, it is supplied for NON-COMMERICAL use only, if you want to use it for something your charging money for contact Robert Graham and he'll get back to you.

Introduction

With all that out of the way, what is this gauge?

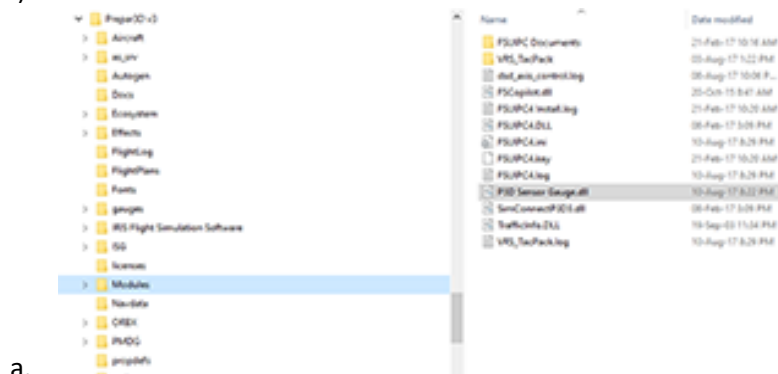
This gauge is actually 2 things, 1. It's an implementation of the Lockheed Martin Radar System as given in the SDK, it is also a Module .DLL that enables complex camera use for sensors that is also linked into the radar as well. It's intent is to simulate the MX-20 and similar systems on P-8, P-3 and S-8 aircraft though it can be used to simulate just about any camera system.

The gauge is supplied with a standard xml gauge (built on the Lockheed Martin Radar System example) but if you just want to use the camera system (what our module adds) then you can do so if you understand some basic xml etc coding as we use Cvar's to handle everything and some render to texture options.

Installing the gauge and the supplied Radar/Camera system.

Once you have downloaded the .zip file you need to do as follows:

1. Copy the .dll file into either **Prepar3D\Gauges** or **Prepar3D\Modules** (example is for the later)



2. Copy the folder Sensor into your **Prepar3D\Gauges** folder
3. Copy the sniperpod.swf file into your **Prepar3D\scaleform** folder.
4. In your dll.xml you need to add:


```
<Launch.Addon>
  <Name>P3D Sensor Gauge</Name>
  <DLLType>PDK</DLLType>
  <Disabled>False</Disabled>
  <ManualLoad>>false</ManualLoad>
  <Path>modules\P3D Sensor Gauge.dll</Path>
</Launch.Addon>
```
5. If you have put the dll into gauges you need to update the path to read
 - a.

```
<Path>guages\P3D Sensor Gauge.dll</Path>
```
6. If you want to have the module ask you to load it in each time change manual load from false to true
7. To actually 'use' the gauge you need to add the following items into the aircraft that you plan on using it in.
 - a. In the Aircraft.CFG file you need to add 4 new cameras, these are the 4 sensor cameras. You may modify these if you want but the TITLES must REMAIN the same. Change the [CameraDefinition.###] to suit your aircraft.

```
[CameraDefinition.012]
Title = "SENSORTURRET1"
Guid = {91d9eb11-dd3c-454d-9ada-b9235ce1ee00}
Description = MX-20HD Infrared sensor
Origin = Center
SnapPbhAdjust = Swivel
SnapPbhReturn = FALSE
PanPbhAdjust = Swivel
PanPbhReturn = FALSE
Track = None
ShowAxis = FALSE
AllowZoom = TRUE
MinZoom = 0.3
MaxZoom = 256
InitialZoom = 1.50
SmoothZoomTime = 6.0
ShowWeather = Yes
InitialXYZ = 0, -1.722, 8.375
InitialPbh = -5, 0, 0
XYZAdjust = FALSE
Category=Aircraft
MomentumEffect=False
ClipMode=Minimum
AllowBelowGround=false
RenderDesignators = True
SensorMode = IRWhiteHot
RenderToTexture=TRUE
ScaleformOverlay = =..\..\Scaleform\sniperpod.swf
```

[CameraDefinition.013]
Title = "SENSORTURRET2"
Guid = {438d7e8f-0b60-4b11-ba9d-351623e8e2fb}
Description = MX-20HD Grey Scale E/O
Origin = Center
SnapPbhAdjust = Swivel
SnapPbhReturn = FALSE
PanPbhAdjust = Swivel
PanPbhReturn = FALSE
Track = None
ShowAxis = FALSE
AllowZoom = TRUE
MinZoom = 0.3
MaxZoom = 256
InitialZoom = 1.50
SmoothZoomTime = 6.0
ShowWeather = Yes
InitialXyz = 0, -1.722, 8.375
InitialPbh = 0, 0, 0
XyzAdjust = FALSE
Category=Aircraft
MomentumEffect=False
ClipMode=Minimum
AllowBelowGround=false
RenderDesignators = True
PostProcess00 = GrayScaleColorizer
RenderToTexture=TRUE
ScaleformOverlay = =..\..\Scaleform\sniperpod.swf

[CameraDefinition.014]
Title = "SENSORTURRET3"
Guid = {36a87c25-6152-4e86-8d8c-e45b9b751dd1}
Description = MX-20HD Normal View.
Origin = Center
SnapPbhAdjust = Swivel
SnapPbhReturn = FALSE
PanPbhAdjust = Swivel
PanPbhReturn = FALSE
Track = None
ShowAxis = FALSE
AllowZoom = TRUE
MinZoom = 0.3
MaxZoom = 256
InitialZoom = 1.50
SmoothZoomTime = 6.0
ShowWeather = Yes
InitialXyz = 0, -1.722, 8.375

InitialPbh = 0, 0, 0
XyzAdjust = FALSE
Category=Aircraft
MomentumEffect=False
ClipMode=Minimum
AllowBelowGround=false
RenderToTexture=TRUE
ScaleformOverlay = =..\..\Scaleform\sniperpod.swf

[CameraDefinition.015]
Title = "SENSORTURRET4"
Guid = {91d9eb11-dd3c-454d-9ada-b9235ce1ee00}
Origin = Center
SnapPbhAdjust = Swivel
SnapPbhReturn = FALSE
PanPbhAdjust = Swivel
PanPbhReturn = FALSE
Track = None
ShowAxis = FALSE
AllowZoom = TRUE
MinZoom = 0.3
MaxZoom = 256
InitialZoom = 1.50
SmoothZoomTime = 6.0
ShowWeather = Yes
InitialXyz = 0, -1.722, 8.375
InitialPbh = -5, 0, 0
XyzAdjust = FALSE
Category=Aircraft
MomentumEffect=False
ClipMode=Minimum
AllowBelowGround=false
RenderDesignators = True
SensorMode = IRBlackHot
RenderToTexture=TRUE
ScaleformOverlay = =..\..\Scaleform\sniperpod.swf

8. In the panel.cfg file add the following

Under [Window Titles]

Window##=Radar and Camera

Remember to change ## to the correct number.

Then add under the window section

[Window##]

size_mm=530,620

Background_color=1,1,1

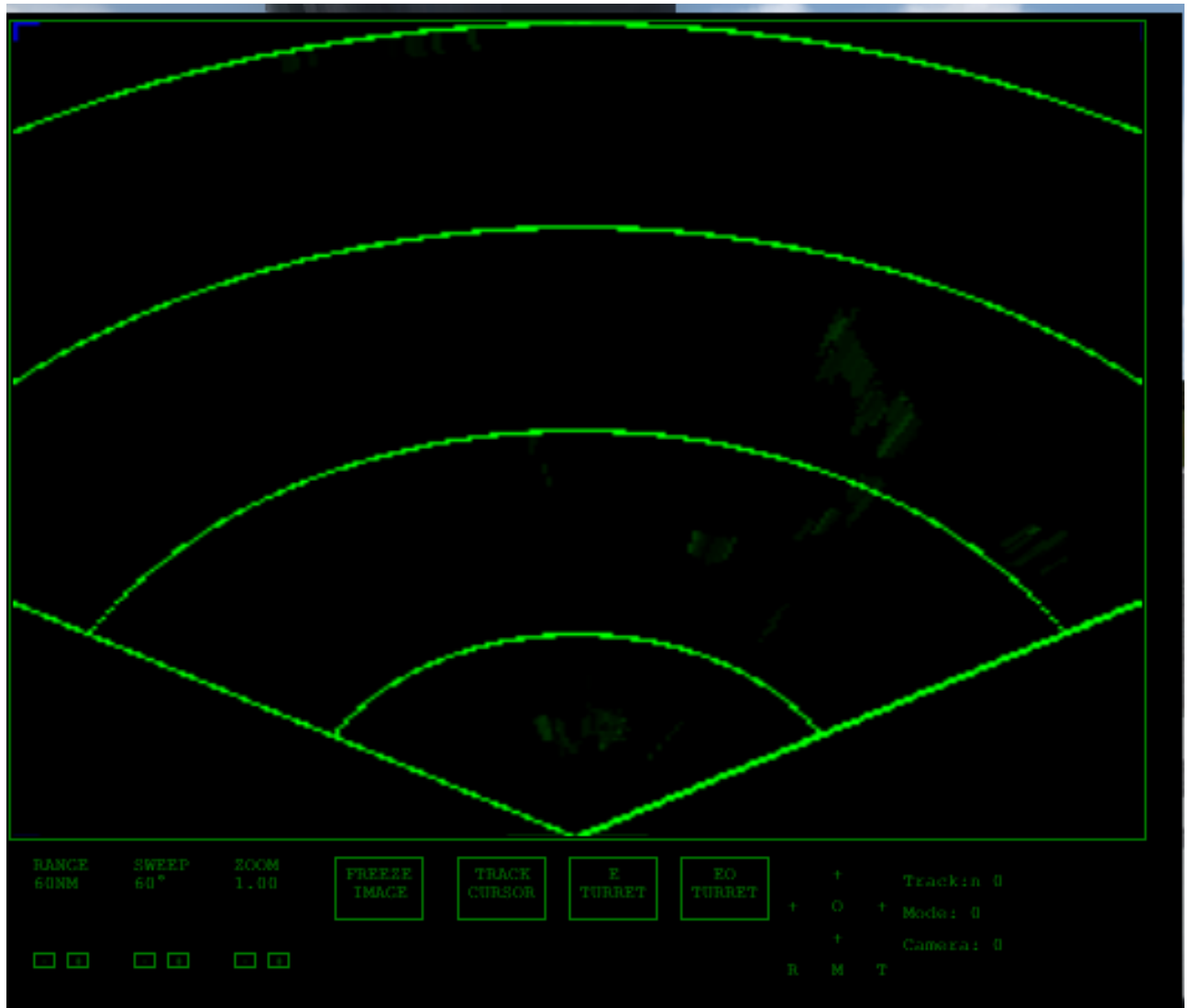
visible=0

position=8

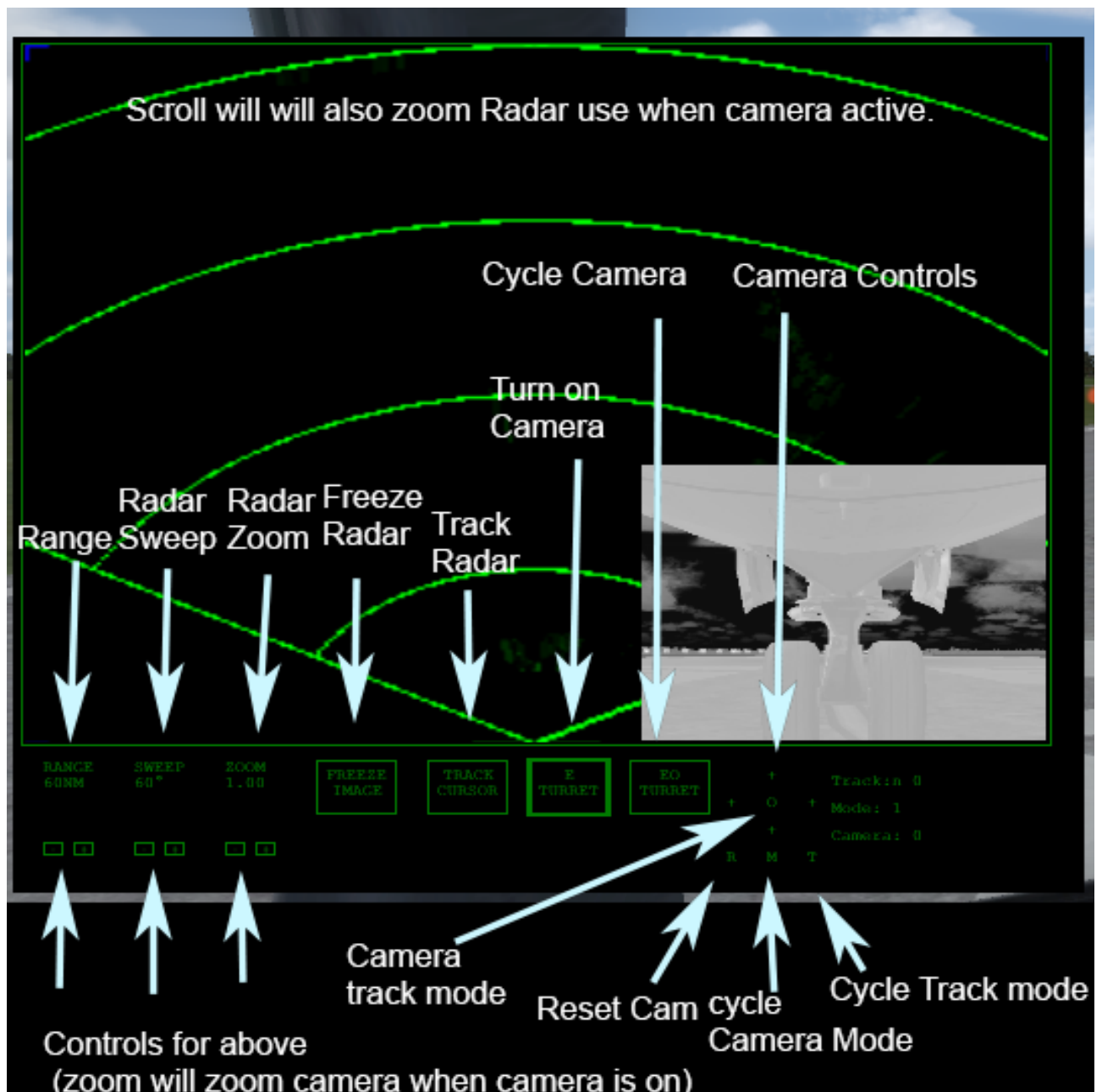
window_size_ratio=1.0

gauge00=Sensor!RGSensor, 5, 5, 510, 610

If the above has worked correctly you should have a new Instrument Panel option called Radar and Camera that looks as below.



Clicking on the E turret Button will result in it looking like this note you can You can interact with the Radar By clicking on the radar itself which will put a cursor down which can then be used to lock the camera to that point with Track Cursor.



For those wanting to build their own Front end with the gauge Gauge, the gauge itself has the following Cvars which you can call using C:SENSOR:

- c_active: read only current active camera 0,1,2,3 (SENSOR 1, 2,3,4)
- c_mactive: write only 0 – 3 sets camera active. Note when changing active camera it will attempt to slew the new camera to the same settings as the previous camera.
- ReadCam : True/False Reads camera data (suggest using this on 1 second update timer but up to you) returns to false.
- GPitch, GBank, Gheading : Read only floats (numbers) the camera relative to origin Pitch, Bank and Heading in degrees.
- GX, GY, GZ: read only floats the current camera X,Y,Z offsets.

- GSensorMode: read only returns 0 – 3 where 0 – None, 1 – IR White Hot, 2 – IR Black Hot, 3 – Gdata
- GHorzFOV GVertFOV : read only returns current Camera FOV for Horizontal and Vertical FOV
- GtrackMode: the current tracking mode the camera is in 0 is none, 1 is Entity/Target Track, 2 is Ground Stabilized, 3 is Position Tracking
- GetLat, GetLon, GetAlt: Read only gets the current Lat/Lon and ground alt that the camera is looking at in degree's
- GSpeed:
- C_mtrackmode: Sets the tracking mode on the camera on a move command.
- MPitch, MBank, MHeading : read (same as G values)/Write number sets camera relative to origin commands for Pitch, Bank and Heading that will be read in when a move is commanded.
- MSpeed: Read – get the current movement time in seconds for each move, Write - time in seconds for a movement command to happen (sets globally)
- SetLat, SetLon, SetAlt – Set the Lat/Lon and Altitude for the camera to look at in track mode 3, However these must be entered in as Radians (as the Radar gauge supplies them in). Camera will move when a move is sent.
- Move: 0 or 1 commands a move to happen, returns to 0 on move.
- SetMode: Write 0 – 3 where 0 – None, 1 – IR White, 2 – IR Black 3 – Gdata. We do not recommend using G data it seems busted.
- PitchUp, PitchDown, HeadingLeft, HeadingRight : 0 / 1 Moves the Camera 0.05F in the direction given sets to 0 after.
- RollLeft, RollRight : 0/1 currently resets roll to 0.
- cyclemode: 0/1 cycles the camera mode by 1. Does not go to 3.
- cyclecamera: 0/1 cycles the camera from 0 – 3.
- cycletrack : 0/1 cycles the track between 0,1,2.
- reset: 0/1 resets the camera to Default settings.
- debug: 0/1 sets how much information is written to the log, 0 is basic while 1 will write all data.

To Invoke the actual gauge to WORK you require a Texture Element in your gauge with the name Private View eg:

```
<Texture Name="Private View" Width="200" Height="200" />
```

A log is written into the ROOT p3D folder called Sensor Gauge.log which can be used to check that things are functioning as they should.