Wed 12/7/2005 6:45 AM

Hi Alvin,
Could you please help to load the item Mr. Albright sent below on the ITB box? Please check with Dan or Ed if they already did so or not.

Thank you,
Showin

503-808-4205 desk
503-808-4204 computer rm
503-808-4277 fax

From: DudleyDevices@aol.com [mailto:DudleyDevices@aol.com]
Sent: Tue 12/6/2005 9:39 AM
To: Fu, Showin M NWP
Cc: Wittinger, Rodney J NWP; Lee.H.Sheldon@nwp01.usace.army; Ebner, David A NWP
Subject: How to setup the Limits function on the ITB.

Showin,
I'm glad that you are working with this now. I'd like to step you through a few topics on how it works this week to get you up to speed and make sure the ITB is working correctly.

To resume where we left off in September, please verify that the Limits Reset function is working properly.

When they went to McNary in early September, this function was not working because the folder on the hard drive that this file was in was misspelled (or spelled correctly to be technically accurate).

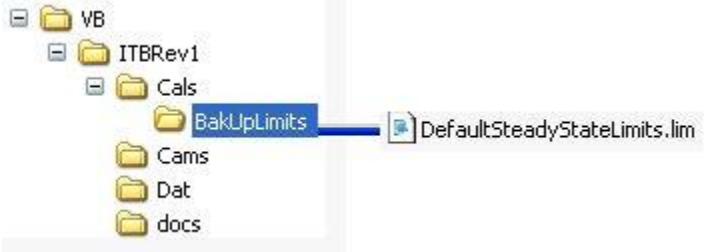
Because the folder name wasn't what the program was expecting, it didn't know where to look for the Default Limits file that has the reset values in it. This prevented these values from being reset to the minimal values shown on the monitor display pictured below.

To setup the Limits function on the ITB, the attached file must be in the properly named folder.

The Limits values are stored in data files in the "**C:\VB\ITBRev1\Cals**" folder on the ITB hard drive.

The "**DefaultSteadyStateLimits.lim**" file (attached) is stuck away into the "**C:\VB\ITBRev1\Cals\BakUpLimits**" folder to get it out of the way so It doesn't get overwritten by accident by the Limits file utility while running the program. I've fumbled it a few times and overwritten it unintentionally. Hiding it in this obscure folder prevents that.

Please make sure the folders are named as shown below and the attached file is in the BakUpLimits folder.



How It Works

When you click on the "Store Limits" button on the Index Test Control panel, all of the Limits values are stored in a file on disk. When the program is restarted, this file is automatically reopened and these values restored to the Limits form.

The name of this file is set with a utility accessible by clicking on the "\/" button that is immediately to the right of the "Limits" button on the right side of the second row of buttons.

To test this function,
1. Click on the "Limits" button in the upper right corner of the ITB screen. The Limits display shown below will appear:


Test Unit
Security WKCal Index Test
ITB PostProces...
XYGraph
Return to ITB
Limits
ForeDay
Forebay in Fee:
.0001
ForeStd
HdStd Ft
.00001
ft/pnt
slope Ft/point
.000001
Fee:
Forebay in Fee:
.0001
ForeStd
Forebay St. Deviation
.00001
ft/sec
slope Ft/sec
.000001
Tail
Tailwater in Feet
0001
TlStd
TailStd Ft
.00001
ft/pnt
slope Ft/point
.000001
Fee:
TailWaterFeet
0001
TlStd
Tailwater St. Deviation
.0001
ft/sec
slope Ft/sec
.000001
Gate
Gate in Percent
0001
GtStd
Gate Std %
.0001
%/pnt
slope %/point
.000001
Percent
Gate Average
0001
GtStd
Gate Std. Deviation
.00001
pct/sec
slope %/sec
.000001
Blade
Blade in Degrees
0001
BlStd
Blade Std Deg
.00001
ft/pnt
slope %/point
.000001
Angle
Blade Average
0001
BlStd
Blade Std. Deviation
.00001
deg/sec
slope deg/sec
.000001
Flow
Flow dP Volts
0001
FlStd
Flow Std dP
.00001
V/pnt
slope dPVolts/point
.000001
volts
Flow Average
0001
FlStd
Flow Std. Deviation
.00001
cfs/sec
slope CFS/sec
.000001
Power
Power in MW
0001
PwStd
Power Std MW
.00001
MW/s
slope MW/point
.000001
volts
Powe Average
0001
PwStd
Power Std. Deviation
.00001
MW/sec
slope MW/sec
.000001
TailControl
0
32
0
0
ITB SteadyState
Avg Data
Samps
MidYs
Slopes
STimes
AvgMidYs
AvgSlopes
AvgTime
Limit Name
Limit
Std Dev
Limit
Limit Value Err Val
O'Lier
ForeSampOutLierLimit .0001
ForeSampStdLim .00001
ForeSampSlopeLim .000001
ForeAvgOutLierLimit .0001
ForeAvgStdLim .00001
ForeAvgSlopeLim .000001
TailSampOutLierLimit .0001
TailSampStdLim .00001
TailSampSlopeLim .000001
TailAvgOutLierLimit .0001
TailAvgStdLim .00001
TailAvgSlopeLim .000001
GateSampOutLierLimit .0001
GateSampStdLim .00001
GateSampSlopeLim .000001
GateAvgOutLierLimit .0001
GateAvgStdLim .00001
GateAvgSlopeLim .000001
BladeSampOutLierLimit .0001
BladeSampStdLim .00001
BladeSampSlopeLim .000001
BladeAvgOutLierLimit .0001
BladeAvgStdLim .00001
BladeAvgSlopeLim .000001
FlowSampOutLierLimit .0001
FLowSampStdLim .00001
FlowSampSlopeLim .000001
FlowAvgOutLierLimit .0001
FlowAvgStdLim .00001
FlowAvgSlopeLim .000001
PowerSampOutLierLimit .0001
PowerSampStdLim .00001
PowerSampSlopeLim .000001
PowerAvgOutLierLimit .0001
PowerAvgStdLim .00001
start
PaperPort
America Online 9.0...
7 Visual Basic
C:\
AutoLimitsFile

2 .Click on the "**Rst**" button next to the "AutoLimits" button to reset all of the Limits values to the minimal values shown above.

Clicking the AutoLimits button allows the Limits values to migrate up to just-barely allow the existing noise signatures on all 6 channels to be accepted.

It takes several seconds for the Limits function to initialize and settle out as the data buffers fill, and several clicks on the "AutoLimits" button to get the Limits opened enough to accept any data.

There are still some unknowns on how this is going to behave with real data from a unit, but we need to get it working properly on the bench before we can evaluate how best to apply the Limits data filtering in the field.

To demonstrate how the Limits part of the ITB works, I put a Noise button on the Control panel to add random noise to the data input signals.

The Noise simulator writes to the same registers the OPC communication is using, so it probably won't work with the OPC connection to GDACS engaged. I'm not sure about this; I don't have a SoftPLC to work with here.

To see how this works, you can click on the Noise button to get a more interesting signal to look at.

Then click the AutoLimits button several times to open the Limits enough to let some data through while observing the 6 stripcharts across the bottom to gauge if the unit appears to be operating steadystate.

While monitoring these stripcharts for continued steadystate operation, click on the AutoLimits button and watch the Limits values increase.

As they gradually increase, note that the indicator buttons on the left side are flashing red, indicating out of limits conditions on these individual variables. As the Limits adjust up, these red flashing diminishes over time until it almost stops, and if the "Record SS" button is selected, every time they are all green a datapoint will be stored.

This Record SS button has 3 settings that alternate with repeated clicks on the button:
1. **Record Off** - nothing is recorded.
2. **Record SS** - record only data points that have slope, Standard Deviation and outlier count values that are within preset limits.
3. **Record All** - records every data point.

There are two addition recording control buttons, the Stats button and the Record All button, both in the upper right corner of the ITB screen. The data files these two buttons generate are intended for use in analyzing the dynamic behavior of the Limits routine, and will be removed from the ITB when the research and development phase is done.

When you have time, please call me for a walk-through of the Noise and simulation part of the program.
Best,
Doug Albright

Fri 12/9/2005 3:18 PM

503-808-4205 desk
503-808-4204 computer rm

503-808-4277 fax

From: DudleyDevices@aol.com [mailto:DudleyDevices@aol.com]
Sent: Mon 12/5/2005 2:10 PM
To: Fu, Showin M NWP
Subject: A bug was found... Hope it's the only one. Thanks.

<<d:\gdacs\[mcnary]\ITB project\ITBRev1v31.executable >>

Showin,
You're right, a bug was found...
I changed the logic construct of the perturbation limit from an absolute value evaluation to a pair of greaterthan / lessthan statements, as shown below.
Oops.
Sorry,
Doug.

```
Sub LessThanZero()
If (TestUnitOPC.PertRangeLimit.Caption) < 0 Then TestUnitOPC.PertRangeLimit.Caption = 0
If Val(TestUnitOPC.OffSetSize.Caption) > Val(TestUnitOPC.PertRangeLimit) Then
TestUnitOPC.OffSetSize.Caption = TestUnitOPC.PertRangeLimit
If Val(TestUnitOPC.OffSetSize.Caption) < -Val(TestUnitOPC.PertRangeLimit) Then
TestUnitOPC.OffSetSize.Caption = -TestUnitOPC.PertRangeLimit

End Sub
```