**------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------**

**Mail from Greg to Team**

**Register changes made to ObiWan from ElsaOlaf\_ES5**

The following are the changes made to the register on Wan. Changes are compared to the ElsaOlaf\_ES5 version.

|  |  |  |  |
| --- | --- | --- | --- |
| **Register address, Bit #** | **Function on ElsaOlaf** | **Function on ObiWan** | **Notes** |
| 0x0c, <7> | Unused |  |  |
| 0x0c, <4> | Unused | *PreAmpMode<2>*- Internal use by MMC to change ADC amplifier gain and achieve a 150mV range on ADC Isense input. | This is not to be provided for external customer use. To be used only by MMC. |
| 0x0d, <7:0> | Unused | Internal use by MMC to store inductor DCR. | Register was a test typ, now is customer type. |
| 0x16, <2> | Was *PreAmpMode<2>* | Moved to customer area, see above. |  |
| 0x32, <3> | Unused | *Vx\_uv\_sel*- Provides and option to program VX undervoltage threshold to either 100mv (when low) or 200mV (when high). | Was fixed at 100mV on ElsaOlaf. |
| 0x32, <2> | Unused | *global\_olaf\_fault*- When ‘1’, a fault on one Obi causes both Obis to shutdown.  When ‘0’, only the Obi which has a fault shuts down, the second Obi continues operating. | Valid only when dual Olaf app with independent Vouts is configured. |
| 0x52, <7:6> | Unused | *Vout\_uv\_ov<1:0>*- These bits indicate when vout is in an UV fault. |  |
| 0x52, <5:4> | Unused | *Buck\_short<1>*- These bits indicate when vout is in a short fault. |  |
| 0x52, <3:2> | Unused | *Buck\_ocp<1:0>*- These bits indicate when the secondary ilim fault has been triggered. |  |
| 0x52, <1:0> | Unused | *Olaf\_ot<1:0>*- These bits indicate when an over temperature fault has been triggered on an olaf. |  |
| 0x53, <4> | *buckduty\_flt*- removed | Unused |  |
| 0x53, <3> | *Cmpclamp\_flt*- removed | Unused |  |
| 0x53, <2> | *Vxlo\_flt*- removed | Unused |  |
| 0x6d, <7> | Unused | *tr\_listen*- When this bit is ‘0’, the internal power good comparators are enabled immediately after soft start finishes. If ‘1’, there is a fixed 30µs delay after soft start timeout before the power good comparators are enabled. |  |

**------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------**

**Mail from Greg to Team**

**PE24106-PE24107 GUI modifications:**

1. Present GUI is insufficient for all configurations – add configuration selector for 8A, Dual 6A and combined 12A before opening GUI (as in PE24103).
2. Add Vout1 and Vout2 short fault indicators (depends on Global Mode selection). Can Global Mode Reg0x32<2> be preset upon GUI configuration selection?
3. Add extended current range selector switch on GUI for reg0x0C<4> (Item 2 in attached document was requested by Murata when using inductors with high DCR). If bit4=0, the calculation for the current sense registers (0x48, 49, 4A, 4B, 4C and 4D) remains the same “100mV/256\*bin2dec(IOUT0<7:0>)/DCR”. However, if bit4=1, the calculation changes to “150mV/256\*bin2dec(IOUT0<7:0>)/DCR”.
4. Change **Faults** Label to PE24107 Status, change **Buck1 Status** label to PE24106(1) Status and change Buck2 Status label to **PE24106(2) Status.**
5. Determine why does CCM selector does not control Buck Reg0x20? This worked for PE24101-PE24102. Is it GUI related?

**------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------**

**Mail from Mike To Team**

**Hello**

I am not able to attend calls during business hours 9am – 5pm EST

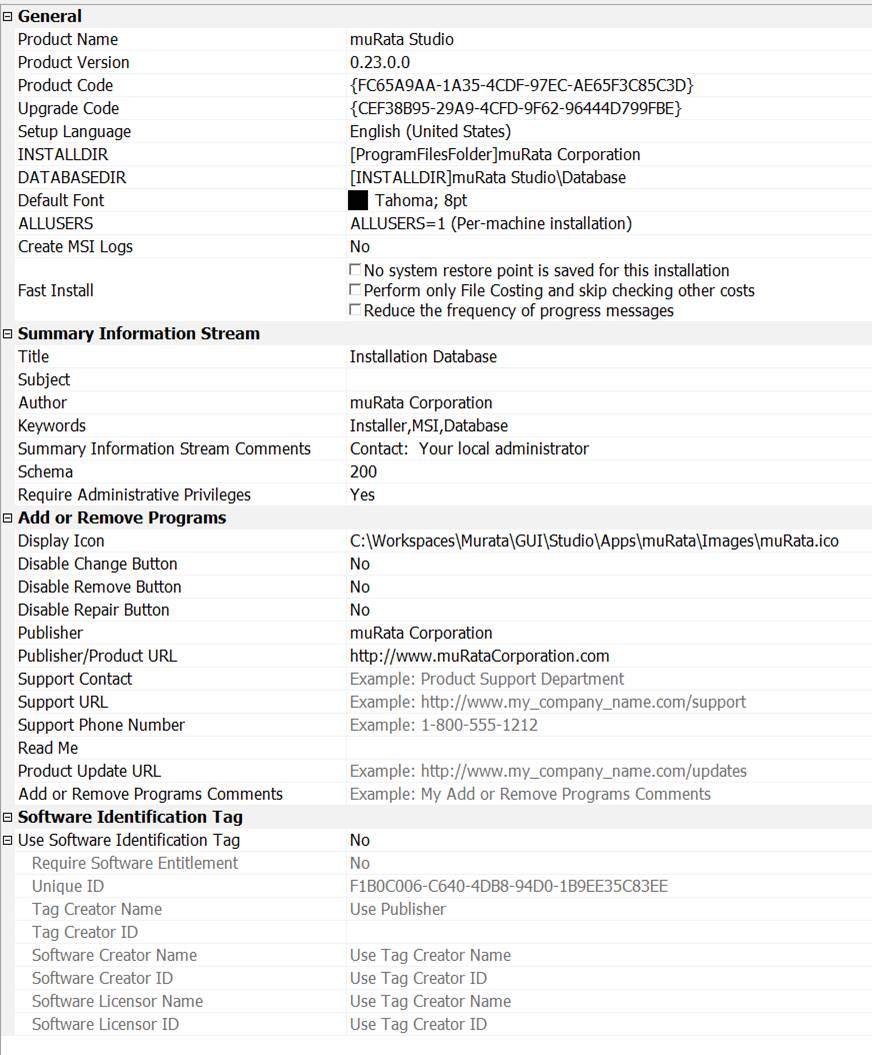
Anytime after 5PM EST is ok Mon, Tue, Wed most preferred.

I misspoke about the Product Code. The product code will need to be changed every time you build a new release.

It is the Update Code that generally stays consistent. Otherwise the installer wants you to uninstall the previous version.

Here is the current Update Code: {CEF38B95-29A9-4CFD-9F62-96444D799FBE}

Here are some screenshots of the setup I had in install shield



**------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------**

**Mail from Mike to Team**

**Sigining**

**Below are the screenshots for the PE24103**

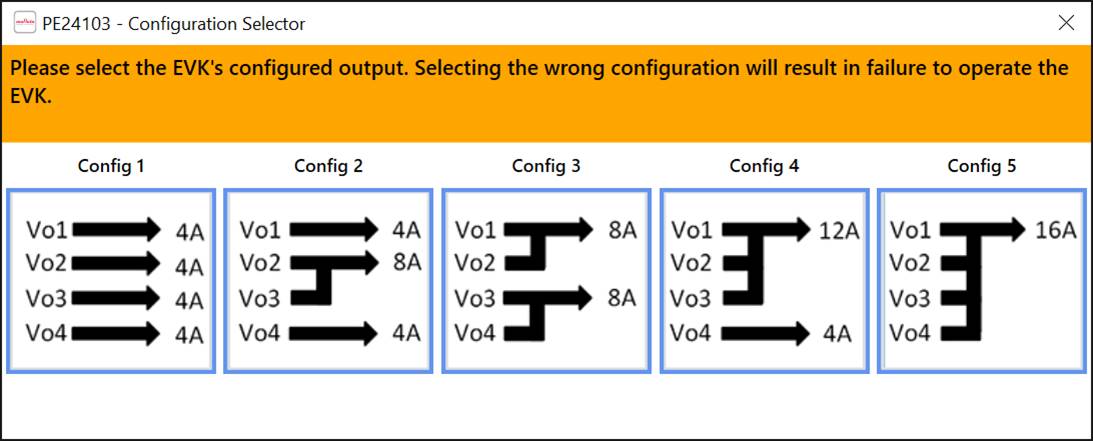
Also, the Visual Studio Solution automatically signs all assemblies via a post build script when in release mode.

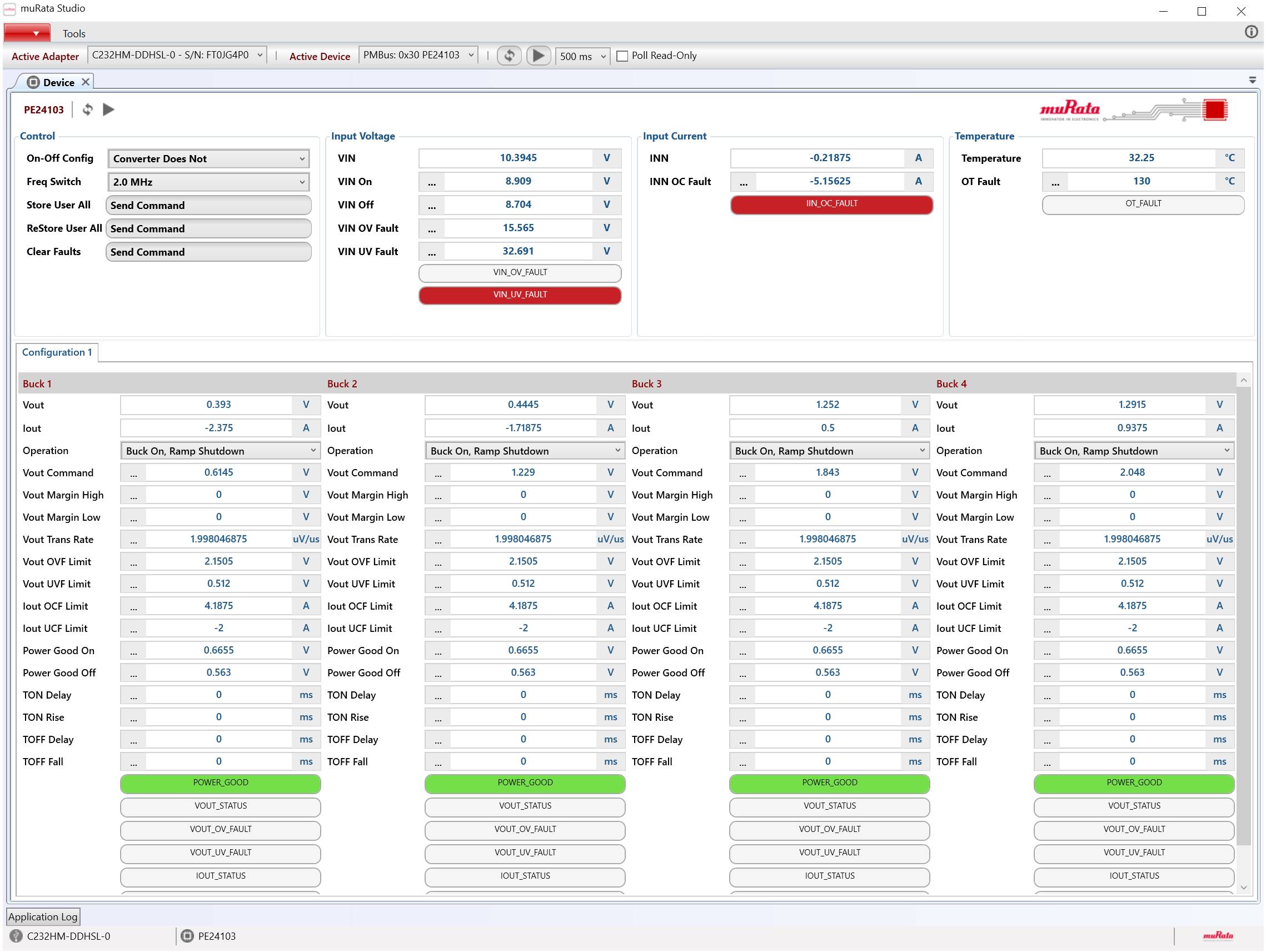
However the installer itself is NOT part of the post script signing. This step is done after the installer has built the package.

Install Shield has built in support for signing the assembly. However I have never used Visual Studio’s package and deployment to sign the output.

Please do some research to implement this requirement. All of the signing information is included in the project:

GUI -> Studio -> Tools -> Signing



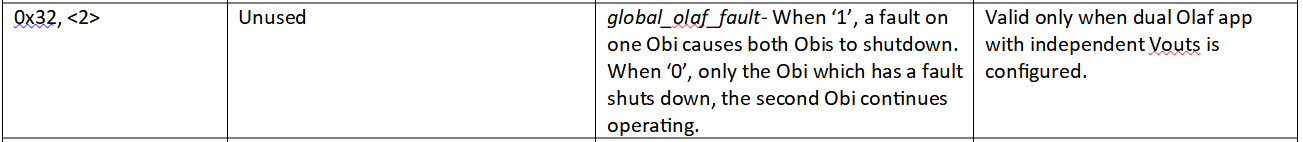
**------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------**

**Mail from Irfan to Mike**

**Please help us to clarify the below concerns regarding Action Items 2 and 3.**

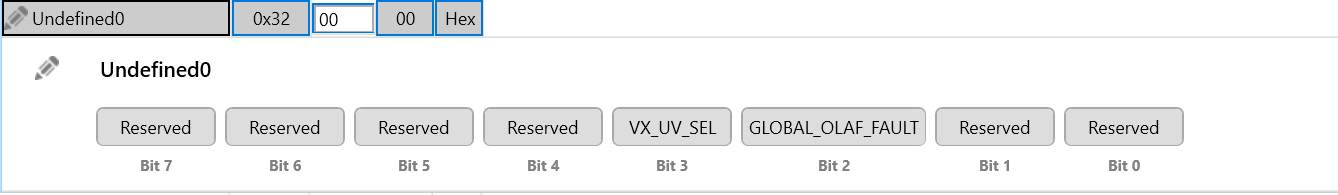
Action item 2:

2) Add Vout1 and Vout2 short fault indicators (depends on Global Mode selection). Can Global Mode Reg0x32<2> be preset upon GUI configuration selection?



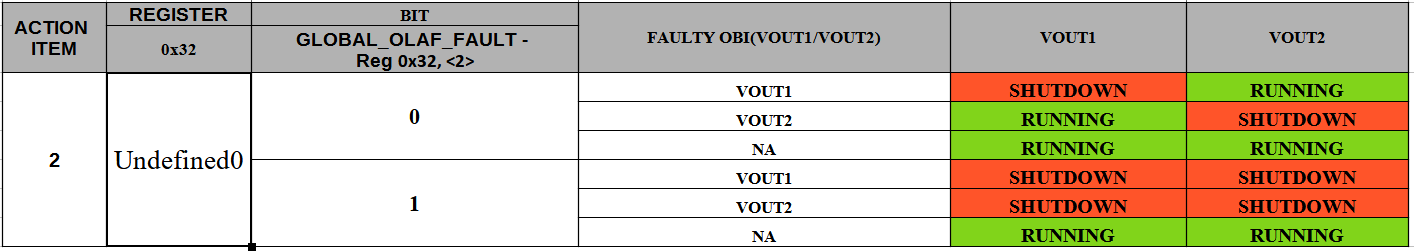
Understanding & Questions :

1) Set 0X32<2> as GLOBAL\_OLAF\_FAULT only for DUAL 6 Configuration .GLOBAL\_OLAF\_FAULT bit should be shown as "Reserved" for 8A and Combined 12A configurations in MuRata UI. Correct ?



2) Add VOUT1 and VOUT2 INDICATORS in FAULTS only for DUAL6 Configuration ?

The indicators should behave as given below :

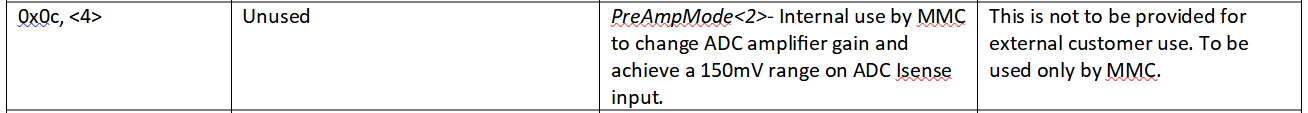


3) What is the Register and bit mappings of VOUT1 and VOUT2 FAULT INDICATORS?

Action item :3

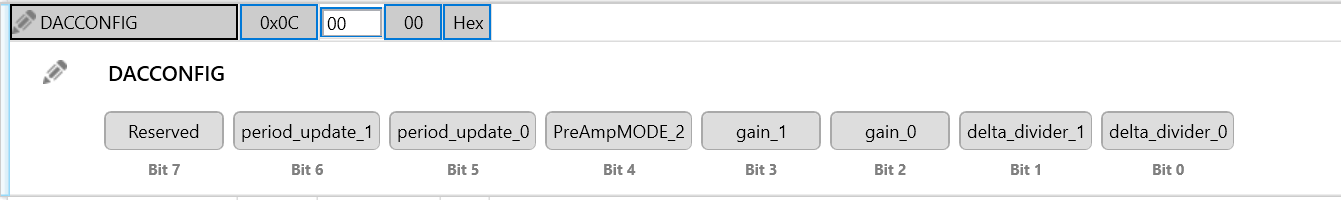
3) Add extended current range selector switch on GUI for reg0x0C<4> (Item 2 in attached document was requested by Murata when using inductors with high DCR).

If bit4=0, the calculation for the current sense registers (0x48, 49, 4A, 4B, 4C and 4D) remains the same “100mV/256\*bin2dec(IOUT0<7:0>)/DCR”. However, if bit4=1, the calculation changes to “150mV/256\*bin2dec(IOUT0<7:0>)/DCR”.

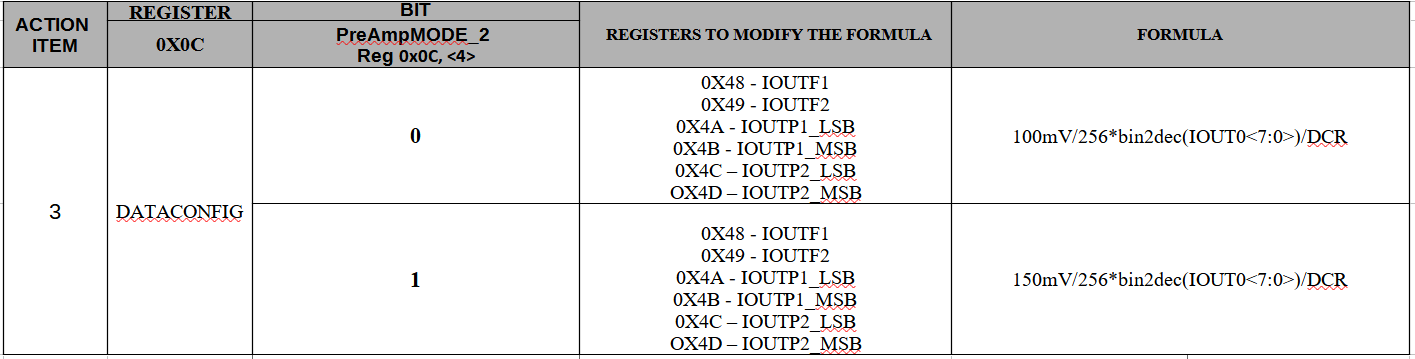


Understanding & Questions :

1) Add PreAmpMODE\_2 in 0x0c<4>.



2) When PreAmpMODE\_2 is turned ON/OFF, the formula modification has to happen for all the below registers .



3) Do we need to add a Label with dropdown in Device GUI to Enable/Disable the current range selector mapped to 0x0c?

4) Do we need to change the transform attribute of all current sense registers?

5) Can you give us some example formulas for preampmode\_2=0 and preampmode\_2=1

Thanks

Irfan

**------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------**

**Mail from Mike to Greg**

**Greg**

We need another short meeting to discuss these 2 action items again. It can be just us if that is easier for time sake and then I can explain a solution to the team.

But at the moment I am a bit confused on the notes in the register\_changes\_obiwan.docx.

All of the below questions are valid and make sense but something seems missing in the instruction details.

It suggests showing faults for VOUT1/2 but we can't just add fault bits unless they are defined in the register map.

Also, for some unknown reason the GUI controls for IOUTF1/F2 are commented out on both ElsaOlaf and Obi and not in use on the GUI.

They have the formula that is called out to be changed from 100mv to 150mv however the IOUTP1/P2 controls use a completely different formula.

Again, something is not being translated clearly.

Let me know when you can be available for a call tomorrow.

Regards

Mike

**------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------**

**Mail from Greg to Mike**

**Hi Mike,**

Based on our conversation, here is the response from the design team:

For 0x32 <3>- This setting will be burnt into MTP by test and fixed. MMC wanted to have an option for the 200mV setting, this was not intended to be programmable by the customer.

For 0x32 <2>- Similar to above, this setting will also be burnt into MTP. When the changes were made to fault handling for fully independent operation, MMC wanted to have a fallback option in which fault handling is no different from Elsa Olaf, so this bit was introduced. This bit is also not customer programmable.

For both the bits above, the app must be evaluated for both bit settings and then decide which setting was to be used.

Let me know if you still have questions.

Best Regards,

Greg

**------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------**

Mail from Mike to Greg

Greg

This explanation is vague and offers no help as to what is required to do in the GUI aside from making the bit available to the user on /devmode.

If that is all that is needed then very little is required on our side. However if there is some other expected behavior then we will need a detailed explanation.

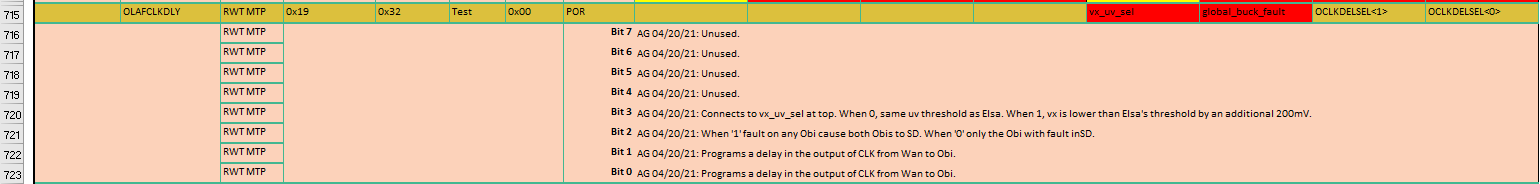
Mike

**------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------**

**Mail from Greg to Mike**

**Hi Mike,**

Not sure if you have a copy of the register map. Hopefully, this will clear things up.



When Reg0x32 <3> is set to 0, the vx\_uv threshold is set to (VIN/3)-100mV by default. When set to 1, the vx\_uv threshold changes to (VIN/3)-200mV. This is already coded up in the digital block, so the selection just needs to be made available in /devmode.

Regarding Reg0x32 <2>, when a fault occurs, the fault indicator turns red. Does that happen automatically or does the GUI need to be refreshed to see it? If the latter is the case, then the user will have to refresh the GUI to see what caused the buck(s) to shut down. In any case, there is no other expected behavior.

Best Regards,

Greg

**------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------**

**Reply of Mike to Greg**

Ok

Then for these 2 action items, all that is needed is to add the bits to the register map xml and ensure the private flag is true.

The correct status flags will update automatically when the GUI is polling.

I will comment on the other action item for changing the formula in another email.

Thanks Greg

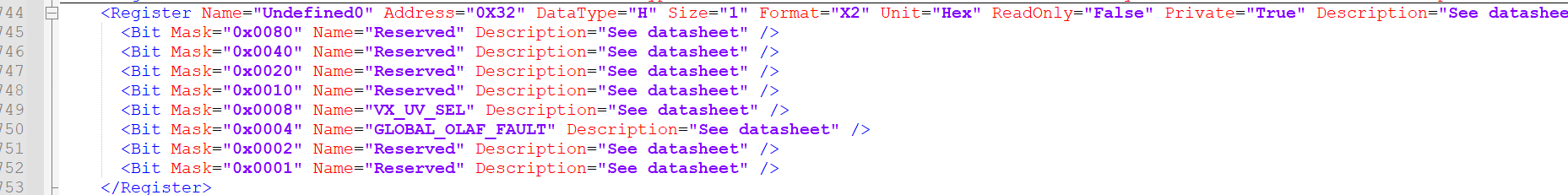
**------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------**

**Mail from Jyothi to Mike**

Hi Mike,

Thank you for the update.

i) We have already modified the device configuration file with the bits mentioned as shown below and is shared in the dropbox location([https://www.dropbox.com/home/Mirafra's%20shared%20workspace/Mirafra?preview=PE24106-R01.adz](https://www.dropbox.com/home/Mirafra's shared workspace/Mirafra?preview=PE24106-R01.adz)). The changes are already existing in the PR(<https://github.com/Mirafra-Technologies/murataStudio/pull/8>).



Also , we would like to discuss the below points over the call? or should we wait for your email?

"When Reg0x32 <3> is set to 0, the vx\_uv threshold is set to (VIN/3)-100mV by default. When set to 1, the vx\_uv threshold changes to (VIN/3)-200mV. This is already coded up in the digital block, so the selection just needs to be made available in /devmode."

"Regarding Reg0x32 <2>, when a fault occurs, the fault indicator turns red. Does that happen automatically or does the GUI need to be refreshed to see it? If the latter is the case, then the user will have to refresh the GUI to see what caused the buck(s) to shut down. In any case, there is no other expected behavior.

Thanks,

Jyothi

**------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------**

**Mail from Chiradeep to Greg**

**Hi Greg**

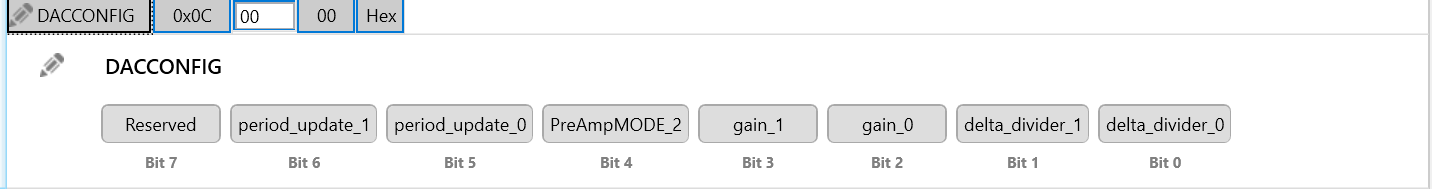
We already have a ***sample version*** of PE24106/107 available with the newly created current selector - please see attached [video](https://www.dropbox.com/home/Mirafra's shared workspace/Mirafra?preview=CurrentSelector.mp4) and [installer](https://www.dropbox.com/home/Mirafra's shared workspace/Mirafra?preview=muRataStudioSetup.msi). Please note, this is a sample GUI with different configurations showing up as label changes (CONFIG1/2/3) for now - we do need the actual changes required in the GUI elements based on config selection that is needed. We would be happy to demo this over a call if required.

However, the team are still waiting for the following clarifications (highlighted in yellow) which we was discussed in a separate email thread :

"When Reg0x32 <3> is set to 0, the vx\_uv threshold is set to (VIN/3)-100mV by default. When set to 1, the vx\_uv threshold changes to (VIN/3)-200mV. This is already coded up in the digital block, so the selection just needs to be made available in /devmode."

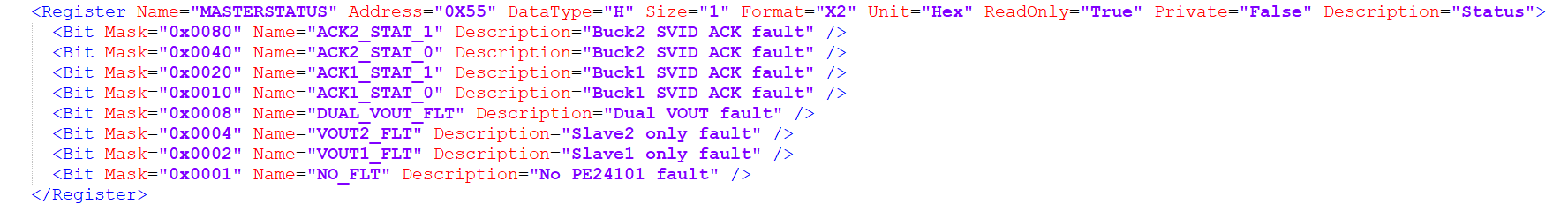
           - Earlier, it was communicated that the current selector formula change(100 to 150 mV) will happen in  **PreAmpMODE\_2 in 0x0c<4>**. But this email refers about the formula change(100 to 200 mV) in **VX\_UV\_SeL of Reg0x32 <2>**.

           - As of now **both the bits are now available in GUI** and are **available only in dev mode** .



**"Regarding Reg0x32 <2>, when a fault occurs, the fault indicator turns red. Does that happen automatically or does the GUI need to be refreshed to see it? If the latter is the case, then the user will have to refresh the GUI to see what caused the buck(s) to shut down. In any case, there is no other expected behavior.**

             - Observed that VOUT1\_FLT and VOUT2\_FLT  fault indicators are present in MASTERSTATUS - Reg0x55<1> and Reg0x55<2> available in GUI . Please reconfirm if the mappings are correct?



Thanks

Chiradeep

**------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------**

**Hi Chradeep,**

I don’t see the attached video and installer. Please see responses below.

"When Reg0x32 <3> is set to 0, the vx\_uv threshold is set to (VIN/3)-100mV by default. When set to 1, the vx\_uv threshold changes to (VIN/3)-200mV. This is already coded up in the digital block, so the selection just needs to be made available in /devmode."

           - Earlier, it was communicated that the current selector formula change(100 to 150 mV) will happen in  **PreAmpMODE\_2 in 0x0c<4>**. But this email refers about the formula change(100 to 200 mV) in **VX\_UV\_SeL of Reg0x32 <2>**.  Reg0x0C<4> and Reg0x32,<3> are two separate functions and both need to be incorporated.

           - As of now **both the bits are now available in GUI** and are **available only in dev mode** . Correct.

Mike – can you comment on the second item (mappings)?

"Regarding Reg0x32 <2>, when a fault occurs, the fault indicator turns red. Does that happen automatically or does the GUI need to be refreshed to see it? If the latter is the case, then the user will have to refresh the GUI to see what caused the buck(s) to shut down. In any case, there is no other expected behavior.

             - Observed that VOUT1\_FLT and VOUT2\_FLT  fault indicators are present in MASTERSTATUS - Reg0x55<1> and Reg0x55<2> available in GUI . Please reconfirm if the mappings are correct?

Thanks,

Greg

**------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------**

**Mail from Chiradeep to Greg**

**Hello Greg**

Sorry for that - some missed out view permissions on the dropbox. Can you let me know whether the following links work now ?

1.[Video with PE24106-107 current selector and sample UI changes](https://www.dropbox.com/home/Mirafra's shared workspace/Mirafra?preview=CurrentSelector.mp4)

2.[Installer packaging](https://www.dropbox.com/home/Mirafra's shared workspace/Mirafra?preview=muRataStudioSetup.msi) with the following changes

a. VS 2019 support (signed and with updated EULA)

b. Sample current configuration selection with sample GUI changes

c. Device file modifications of adding the following bits (coloured in green in the below table from the register\_changes\_obiwan.docx) Other changes mentioned in the document we have observed are already present. (Refer to table 1 below)

d. Label changes :Change **Faults** Label to PE24107 Status, change **Buck1 Status** label to PE24106(1) Status and change Buck2 Status label to **PE24106(2) Status.**

For the existing discussion thread below for changes with respect to **Reg0x0C<4> and Reg0x32<3>** (including formula changes) I feel we should have a call - emails are not specific enough to understand what exactly should we change in the GUI. We have experimented with a sample extended current range selector as seen in image 1 below however, wanted to confirm we are all on the same page.

I am setting up a call for 16th Jun 7AM IST to discuss the same including the larger group. Request everyone to let me know if this works or we should reschedule.

Thanks

Chiradeep

**------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------**