

Technical Note

Renesas RH850/U2A: Hardware Property Area

May 2023



A **TASKING** Company

This Technical Note describes how to program Hardware Property Area on RH850/U2A devices via Python scripts.

Tool requirements:

- ✓ winIDEA 9.21.35 or newer
- ✓ BlueBox iC5700, iC5000
- ✓ RH850 Debug Adapter

Hardware Property Area on U2A devices consists of seven areas, but the focus here is on the following four:

- Configuration Setting Area (CSA)
- Security Setting Area (SSA)
- Block protection area 0 (BPA0)
- Block protection area 1 (BPA1 is not available on smaller devices)

Each of these areas are further divided to 2 areas (Area 0 and Area 1) that can be mapped to front or back side address range:

- Front side - Contains valid area which cannot be updated (programmed).
- Back side - Contains invalid area which can be programmed with new settings.

For details please refer to the device User Manual, Chapter 51.3.3 Mapping of Hardware Property Area in Data Flash Memory.

Programming CSA



The following is an example for programming CSA. For programming BPA follow the same procedure. For SSA please refer to a separate technical note, available on request.

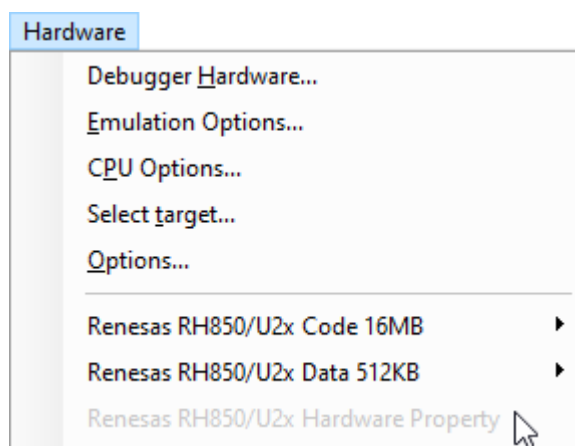
To program new settings in Hardware Property Area follow the steps below:

1. [Enable Hardware Property Area programming](#) in Hardware menu / Options / Programming.
2. [Prepare the Program File](#).
3. [Area Switching](#).

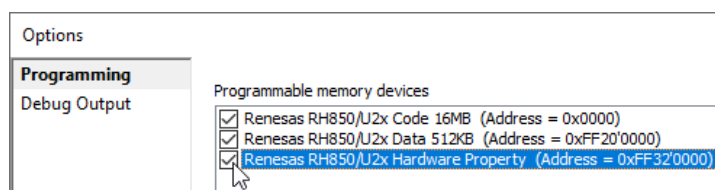
Enabling Hardware Property Area programming

To limit the chance of accidental area changes, the Hardware Property Area programming is not enabled in the *Hardware* menu by default. Open the Hardware menu / Options and check the *Renesas RH850/U2A Hardware Property* option to enable programming.


Hardware menu



Hardware menu / Options




Preparing the Program File



The sample workspace is based on the Renesas Getting Started package. You should familiarize yourself first with this package. Carefully check and double-check with your team that the settings match your requirements.

- 1. Save the attached sample winIDEA workspace to a local drive, then open it and edit the OPBT settings if required.
- 2. Make sure that **One-Time Programmable** settings - OTPs are **not activated**. That they are all 0xFFFFFFFF.



If OTPs need to be activated, then they should be prepared in a completely **separate HEX file** and should be programmed in a **separate step** after programming the Option Bytes. **OTP settings are applied immediately. Therefore, make sure that you program them last!** Otherwise, you won't be able to program the OPBTs.

Do not define both OPBT and OTP HEX files in the same Program Files dialog of the workspace. We recommend having a separate workspace for the OTPs for added safety. It does not suffice to have OTPs at the end of the area Program File, or as a separate download file in a block of files, because winIDEA forms a complete download image from all selected files upfront, before programming.

A note from the "UM Chapter 47.5.2 OTP (One Time Programmable) Function": OTP can be individually set for each block of the user area and the user boot area of code flash memory. When the OTP setting is made for an area, programming by serial programming and by self programming is prohibited. **Once set, the OTP setting cannot be released.**

- 3. Hit Compile and then check the HEX Program File.
- 4. If it's OK, perform a regular Download.
- 5. The programmed OPBTs should be visible in the Back area together with validation flags. All flags must be set for the next step - Area Switching - to succeed.

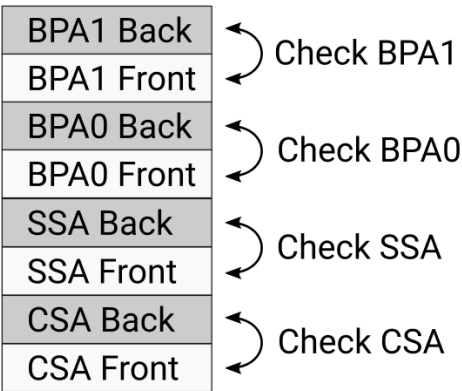
Area Switching


Front side and Back side will be switched, and new settings will be applied. Changes in settings become effective after TAG Update and release from the reset state. OTP Setting of Configuration Setting Area becomes effective without reset. Valid / Invalid area switching for read access also becomes effective without reset.

- 1. Save the script *HWPPropertyAreaSwitch.py*.
- 2. Open the script. Edit the parameters to only enable CSA switch:

```
inParams['CheckCSA'] = 'true'  
inParams['CheckSSA'] = 'false'  
inParams['CheckBPA0'] = 'false'  
inParams['CheckBPA1'] = 'false'
```

- 3. Save it in the same folder as winIDEA workspace.
- 4. Select your Python script in *Tools menu / External Scripts*.The script will execute the required operation.
- 5. Refresh the Watches Window (shortcut F8), check FSWASTAT_0 for the changes, and if all is as expected reset the device to apply the changes.
- 6. To return back to the previous OPBTs, simply call the script again. This way it is easy to toggle between the two sets of OPBTs. By performing the switch again, previous settings will be restored.





For more information on how to run Python scripts in winIDEA refer to [winIDEA Help](#). You can use [isystem.connect API](#) for running Python scripts.

Python script and winIDEA workspace examples

Download and unzip the scripts and workspace examples via the link below.

RH850_U2A_programming.7z



In the zipped directory you can find:

- HWPropertyAreaSwitch.py
- winIDEA workspace Configuration Setting Area (Option Bytes) *U2A_OPTBT_REE_GHS*
- Optional Python script examples listed below

Optional Python script examples

Scripts can be used to minimize the chance that something wrong gets programmed to the area.



Use the scripts at your own discretion.

Scripts copy the Front side to the Back side:

- CSA_copy_front_to_back.py
- SSA_copy_front_to_back.py
- BPA0_copy_front_to_back.py
- BPA1_copy_front_to_back.py

Scripts compare the Front side with the Back side and show differences.

- CSA_compare_front_back.py
- SSA_compare_front_back.py
- BPA0_compare_front_back.py
- BPA1_compare_front_back.py