Date: Aug. 6, 2019

RENESAS TECHNICAL UPDATE

TOYOSU FORESIA, 3-2-24, Toyosu, Koto-ku, Tokyo 135-0061, Japan
Renesas Electronics Corporation

Product Category	System LSI		Document No.	TN-RZ*-A0059A/E	Rev.	1.00
Title	Advanced 5port Switch (A5PSW) register write access issue		Information Category	Technical Notification		
Applicable Product	See below	Lot No. See below	Reference Document	RZ/N1 Series User's Manual (See below for details)		

We would like to inform about potential issues regarding Advanced 5port switch (A5PSW) register write access as described below.

1. Applicable Product

Product Group	Part Number	Package Type	Configuration
RZ/N1D	R9A06G032NGBG	400BGA	Dual Cortex-A7, PRP/HSR
	R9A06G032VGBG	400BGA	Dual Cortex-A7
	R9A06G032VGBA	324BGA	Dual Cortex-A7
RZ/N1S	R9A06G033NGBG	324BGA	Single Cortex-A7, PRP
	R9A06G033VGBA	196BGA	Single Cortex-A7
RZ/N1L	R9A06G034VGBA	196BGA	Cortex-M3

2. Reference Document and software

Reference document name	Document Number	Current Revision	Revised Revision
RZ/N1D Group, RZ/N1S Group, RZ/N1L Group User's Manual: System Introduction, Multiplexing, Electrical and Mechanical Information	R01UH0750EJ****	V1.00	V1.10

Reference software	Note
YCONNECT-IT-RZN_V1.4.2	This version will be downloadable from Renesas WEB SITE late August ,2019. It includes sample software for the workaround.

3. Issue and condition

In case the write access is performed to the specific registers in A5PSW, there is a possibility that an incorrect data would be written into the registers. A5PSW registers to be affected are below.

- Register addresses: 0x4500 0800 to 0x4405 3F3C (the specific registers in A5PSW)
- Affected functions: MACs, DLR, PRP, HUB, Pattern Matchers, TDMA Scheduler

However, no issues happen on the read access, and normal Ethernet communication not using A5PSW is no problem.

4. Revision Plan

- RZ/N1 devices will be updated to fix the issue. That is scheduled in mid of November, 2019.

- In the fixed device, product version register (0x4000 C19C) will be updated together.

  -Fixed device VERSION[8:0] = 0x013(RZ/N1D), 0x111(RZ/N1S, RZ/N1L)

  -Not fixed device VERSION[8:0] = 0x011(RZ/N1D), 0x110(RZ/N1S, RZ/N1L)

Above register value change will be reflected to the User's manual.

<table>
<tr><th>Page</th><th>Revision</th></tr>
<tr><td>157</td><td>[Current]
<table>
<tr><th>Bit Position</th><th>Bit Name</th><th>Function</th><th>R/W</th></tr>
<tr><td>b7～b0</td><td>VERSION</td><td>Latest HW version
0x11：RZ/N1D
0x10：RZ/N1S、RZ/N1L</td><td>R</td></tr>
</table>
[Revised]
<table>
<tr><th>Bit Position</th><th>Bit Name</th><th>Function</th><th>R/W</th></tr>
<tr><td>b7～b0</td><td>VERSION</td><td>HW version *1
RZ/N1D: 0x13
RZ/N1S, RZ/N1L: 0x11</td><td>R</td></tr>
</table>
Note 1. If the version shows RZ/N1D:0x11 or RZ/N1S, RZ/N1L:0x10, A5PSW register write access issue might happen. Please refer to Technical Update with regard to the workaround.</td></tr>
</table>

- "R1" marking will be added left side of MADE IN marking on the PKG.

RZ/N1D LFBGA400

RZ/N1D LFBGA324

RZ/N1S LFBGA324

RZ/N1S LFBGA196

RZ/N1L LFBGA196

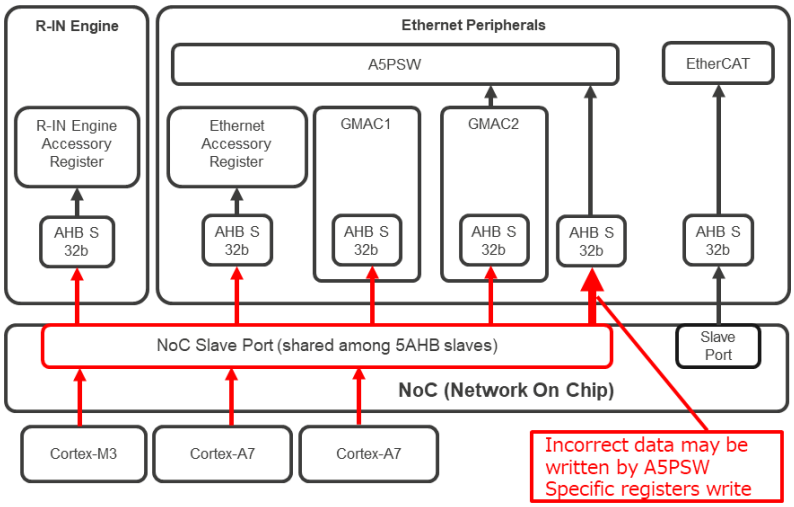
5. Workaround

If you use the not fixed device, implement the following software workarounds when writing to the specific registers in A5PSW. When reading data, this software workaround is not necessary.

By writing the target data to the dummy address (4405 5000h) of A5PSW, then writing the same data to the target register atomically, it can avoid the issue of writing incorrect data. However, since A5PSW shares the register access bus (AHB slave) with other peripherals, other CPU core write to the AHB slaves can be cut into the atomic access. In addition, there is a possibility that another write cut into between the atomic writes when the same "CPU core“ interrupt processing writes to “the AHB slaves”. The following implementation steps takes that into account.

The AHB slaves— R-IN Engine Accessory Register, Ethernet Accessory Register, GMAC1, GMAC2, A5PSW

The CPU cores — Cortex-A7 processor0, Cortex-A7 processor1, Cortex-M3,



When writing to the AHB slaves from only one of “the CPU cores”, refer to [Implementation 1].

When writing to the AHB slaves from multiple “the CPU cores”, refer to [Implementation 2].

[Implementation 1]

- Limit “the CPU cores” to one to access “the AHB slaves”
- Stop an interrupt routine which executes a write access to “the AHB slaves” during the atomic writes

— Write to A5PSW registers, as following steps

(1) Disable interrupts
(2) Write the target data to the dummy address (4405 5000h)
(3) Write the same data to the target register
(4) Restore any interrupts to the previous setting

[Implementation 2]

- While writing to the A5PSW register, make sure that no other “the CPU cores” writes to “the AHB slaves”.
- Stop an interrupt routine which executes a write access to “the AHB slaves” during the atomic writes
- Use RZ/N1 hardware semaphore for exclusive accesses between “the CPU cores”.
- All “the CPU cores” are required to implement workarounds.

— Write to A5PSW registers, as following steps

(1) Disable interrupts
(2) Get the semaphore (hardware)
(3) Write the target data to the dummy address (4405 5000h)
(4) Write the same data to the target register
(5) Release the semaphore
(6) Restore any interrupts to the previous setting

— To write to R-IN Engine Accessory Register, Ethernet Accessory Register, GMAC1, or GMAC2, follow these steps

(1) Get the semaphore (hardware)
(2) Write the data to the register
(3) Release the semaphore