# FreeRTOS Cortex M33 NTZ, without MPU

forums.freertos.org/t/freertos-cortex-m33-ntz-without-mpu/9260

Hi!
I just found a strange case where FreeRTOS Kernel V10.3.1 causes a systematic hardfault in vRestoreContextOfFirstTask(); when vTaskStartScheduler() is called for the first time.

The basic setup is:

- Nrf9160 Cortex M33
- No TrustZone, with FPU, without MPU
- Eveything is running in non_secure.
- Using the existing GCC/ARM_CM33_NTZ/non_secure ports.

> #define configENABLE_MPU 0
> #define configENABLE_FPU 1
> #define configENABLE_TRUSTZONE 0

The hardfault occurs when vRestoreContextOfFirstTask() branches (BX R2) to 0xFFFF FFBC, trying to exit the SVCall_Exception.

It is my first time playing around with M33 cortexes, so I might have missconfigured something else, but I guess this specific configuration results in an incorrect EXC_RETURN value.

Is this a known problem?

Solved by Gaurav Aggarwal in post #2

> How do you determine that the MCU is running non-secure. Is the TrustZone disabled in hardware? I do not have the hardware you mentioned but I will check the configuration on a different hardware and update. Meanwhile, would you please try with configRUN_FREERTOS_SECURE_ONLY set to 1 in your FreeR…

aggarg Gaurav AggarwalAWS
Mar 2020

> Eveything is running in non_secure.

How do you determine that the MCU is running non-secure. Is the TrustZone disabled in hardware?

I do not have the hardware you mentioned but I will check the configuration on a different hardware and update. Meanwhile, would you please try with configRUN_FREERTOS_SECURE_ONLY set to 1 in your FreeRTOSConfig.h:

```
#define configRUN_FREERTOS_SECURE_ONLY 1
```

Thanks.
Solution

fusedFET
Mar 2020
Hey, thank you for your answer.
It seems like adding

> #define configRUN_FREERTOS_SECURE_ONLY 1

did fix the problem.
I did set all flash regions as non-secure, as such I expected everything to run as non-secure.
I think I'll need to look up some more documentation about TZ.

aggarg Gaurav AggarwalAWS
Mar 2020
Glad that it worked for you. What it means is that the TrustZone is not disabled and the MCU is running as secure. As you rightly said, you need to check the document about how to disable TrustZone in hardware.

Thanks.