

Peripheral Driver Generator ご使用上のお願い

Peripheral Driver Generator の使用上の注意事項を連絡します。

- RX210グループMCUのクロック周波数精度測定回路 (CAC) を使用する際の注意事項

1. 該当製品

Peripheral Driver Generator V.2.03~V.2.05

2. 内容

該当製品で生成した以下の関数をユーザプログラムで呼び出すと、0番地に誤った書き込みをします。

R_PG_CAC_GetStatusFlags

R_PG_CAC_GetCounterBufferRegister

これらの関数は、RX210グループのクロック周波数精度測定回路 (CAC) の設定画面で必要項目を設定すると生成されます。

3. 発生例

(1) R_PG_CAC_GetStatusFlags 関数の場合

```
-----  
// "R_PG_<プロジェクトファイル名>.h" のインクルード
```

```
// 例: プロジェクト名 "default" の場合
```

```
#include "R_PG_default.h"
```

```
bool g_err;
```

```
bool g_end;
```

```
bool g_ov;
```

```
void function(void)
```

```
{
```

```
    R_PG_CAC_GetStatusFlags(&g_err, &g_end, &g_ov);
```

```
}
```

(2) R_PG_CAC_GetCounterBufferRegister 関数の場合

```
-----  
// "R_PG_<プロジェクトファイル名>.h" のインクルード  
// 例: プロジェクト名 "default" の場合  
#include "R_PG_default.h"  
  
uint16_t cacntbr_val;  
  
void function(void)  
{  
    R_PG_CAC_GetCounterBufferRegister(&cacntbr_val);  
}
```

4. 回避策

(1) R_PG_CAC_GetStatusFlags 関数の場合

R_PG_CAC_GetStatusFlags関数を使用せず、CACステータスレジスタを読み出して、ステータスフラグを取得してください。

例 :

```
-----  
// "R_PG_<プロジェクトファイル名>.h" のインクルード  
// 例: プロジェクト名 "default" の場合  
#include "R_PG_default.h"  
  
// I/Oレジスタ定義ファイルのインクルード  
#include "iodefine_RPDL.h"  
  
bool g_err;  
bool g_end;  
bool g_ov;  
  
void function(void)  
{  
    g_err = CAC.CASTR.BIT.FERRF;  
    g_end = CAC.CASTR.BIT.MENDF;  
    g_ov = CAC.CASTR.BIT.OVFF;  
}
```

(2) R_PG_CAC_GetCounterBufferRegister 関数の場合

R_PG_CAC_GetCounterBufferRegister関数を使用せず、CACカウンタバッファレジスタを読み出して、CACカウンタバッファ

レジスタの値を取得してください。

例：

```
-----  
// "R_PG_<プロジェクトファイル名>.h" のインクルード  
// 例: プロジェクト名 "default" の場合  
#include "R_PG_default.h"
```

```
// I/Oレジスタ定義ファイルのインクルード  
#include "iodefine_RPDL.h"
```

```
uint16_t cacntbr_val;
```

```
void function(void)  
{  
    cacntbr_val = CAC.CACNTBR;  
}
```

5. 恒久対策

Peripheral Driver Generator V.2.06で改修する予定です。

[免責事項]

過去のニュース内容は発行当時の情報をもとにしており、現時点では変更された情報や無効な情報が含まれている場合があります。ニュース本文中のURLを予告なしに変更または中止することがありますので、あらかじめご承知ください。