Notes on Changing the Settings of the PDA Clock

Some restrictions involving changing the settings of the PDA clock have been discovered. Please be aware of these restrictions as they are important in the development of PDA applications.

[Restriction]
Do not change the clock settings in ARM mode.

[Reason]
Changing the clock settings in ARM mode can result in unstable operation. This is due to variations between individual chips used in the PDAs.

[Description]
Changing clock settings using the PMFrequency register should be done in Thumb mode. The change clock system call (SWI 4) uses Thumb mode internally so no additional operation is required.
The SWI 4 system call should be used whenever possible to change clock settings. However, SWI 4 is not able to change the clock frequency to 32 KHz, so in this case, the appropriate value should be set directly to the PMFrequency register.

[Additional notes on changing the clock settings]
If values are written directly to the PMFrequency register, and this is done frequently, the clock frequency can become unstable and, worst case, may hangup the operation of the PDA. This can also occur even if the same value is written frequently to the register.

The following are additional issues to be aware of when changing the frequency of the PDA clock.

1) Changing the frequency to other than 32 KHz
   Using the SWI 4 system call is recommended, but when writing directly to the PMFrequency register, wait for bit 4 (LOCKBIT) of PMFrequency to become 1 after writing to the register before performing additional operations.

2) Changing to 32KHz
   When the frequency is changed to 32KHz, LOCKBIT will not become 1. Thus, please insert an empty loop to delay at least 1 msec after a write.

[Sample program]

• Changing the clock frequency to other than 32KHz

[main.s]

```
        CODE32
        AREA    MAIN, CODE, READONLY, INTERWORK
                :
        MOV     r0, #7                  ; 4MHz
        SWI     4
                :
                :
```

• Changing the clock frequency to 32 KHz

[main.s]

```
        IMPORT  Freq_32K
```

```
        CODE32
        AREA    MAIN, CODE, READONLY, INTERWORK
                :
        BL      Freq_32K
                :
                :
```

[freq.s]

```
        EXPORT  Freq_32K
        CODE16
        AREA    FREQ, CODE, READONLY, INTERWORK
Freq_32K
        PUSH    { r4-r5 }
        PUSH    { lr }

        MOV     r4, #0                          ; 32KHz
        LDR     r5, =0x0b000000
        STR     r4, [r5]

        MOV     r4, #100            ; loop for at least 1 msec
waitloop
        SUBS    r4, #1
        BPL     waitloop

        POP     { r4 }
        MOV     lr, r4
        POP     { r4-r5 }
        BX      lr
```