

```

@Done
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@
@
@
@
@
@
@
@
@
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@

@ Description: Defines register values used by the audio system.
@
@ Revision History:
@ Name          Comment          Date
@ Will Werst    Initial version  Some lonely night around 6/10/17
@ Will Werst    Comment          October 2017

@Macro for setting Hardware register at destination with value val
@Registers modified: r0, r1
.macro mSET_HREG dest, value

LDR r0, =\dest
LDR r1, =\value
STR r1, [r0]

.endm

.macro mLOADTOREG reg, mem_loc

    LDR \reg,    =\mem_loc
    LDR \reg,    [\reg]

.endm

.macro mSTOREFROMREG reg, temp_reg, mem_loc

    LDR \temp_reg, =\mem_loc
    STR \reg, [\temp_reg]

.endm

.macro mSTARTFNC
    PUSH {lr}
.endm

.macro mRETURNFNC
    POP {pc}
.endm

.macro mSTARTINT
    PUSH {r0-r3, lr}
.endm

.macro mRETURNINT
    mSET_HREG AIC_EOICR, 1
    POP {r0-r3, lr}
    SUBS pc, lr, #WORD_SIZE
.endm

@ Macros to handle critical code. Note that r7 is used, the value is restored
@ after critical code section, but since the stack is used to store value
@ in critical code start, its value cannot be preserved during the critical
@ code section.

```

```
.macro mSTARTCRITCODE
    PUSH {r7}
    MRS r7, cpsr
    PUSH {r7}
    ORR r7, r7, #(I_BIT | F_BIT)
    MSR cpsr_c, r7
.endm

.macro mENDCRITCODE
    POP {r7}
    MSR cpsr, r7
    POP {r7}
.endm
```