

Summary

Functions

SEQ_GP_Button
SEQ_GP_Button_CS
SEQ_GP_Button_NoCS
SEQ_GP_LED_Update
SEQ_GP_LED_Update_Hook
SEQ_GP_LED_Update_NoF1L
SEQ_GP_LED_Update_NoF1R
SEQ_GP_LED_Update_NoHook
SEQ_GP_Mode0_Button
SEQ_GP_Mode0_Button_Loop
SEQ_GP_Mode0_Button_Loop_NoDrums
SEQ_GP_Mode0_Button_Next
SEQ_GP_Mode0_Button_Depr
SEQ_GP_Mode0_LED
SEQ_GP_Mode0_LED_Cont
SEQ_GP_Mode3_Button
SEQ_GP_Mode3_LED
SEQ_GP_Mode4_Button
SEQ_GP_Mode4_LED

Variables

Function Descriptions

[SEQ_GP_Button](#) first checks to see if the hook `CS_MENU_CheckMenuGPCallbackHook` is installed in `cs_menu_buttons.inc`. If not it branches to `SEQ_GP_Button_NOCS`.

[SEQ_GP_Button_CS](#) simply calls `CS_MENU_ExecMenuGPCallbackHook`

in `SEQ_GP_Button_NoCS` `BIFSET` is used to determine which mode is active, branching to the proper `SEQ_GP_MODE*_Button` function

In `SEQ_GP_Mode0_Button`, `MIOS_PARAMETER2` is first checked to see if the event is a button 'depress' (release?), if so then branching to `SEQ_CP_Mode0_Depr`. `MIOS_PARAMETER1` is then movff'd into `SEQ_CURRENT_STEP` ... unknown code

`SEQ_GP_Mode0_Button_Loop` first checks if the event comes from the selected track (?) by movf-ing `SEQ_EVNTT` into `W` and calling `MIOS_HLP_GetBitOrMask`

If the button press did not come from one of the GP's than the program branches to
CS_MENU_BUTTON_Handler_NoGP[Internal Link](#)

From:

<http://wiki.midibox.org/> - **MIDIbox**

Permanent link:

http://wiki.midibox.org/doku.php?id=seq_gp

Last update: **2006/10/15 10:35**

