LFR-FSW - Task #559

Garantir que la réception est désactive sur l'APBUART.

03/11/2015 02:34 PM - Alexis Jeandet

Status:	Closed	**Start date:**	03/11/2015
Priority:	Normal	**Due date:**	
Assignee:	Alexis Jeandet	**% Done:**	0%
Category:		**Estimated time:**	0.00 hour
Target version:		**Spent time:**	0.00 hour
revision:	r0		

Description

Suite au keypoint FPGA nous avons décidé de fixer la PIN RXD de l'APBUART à '0' et ne pouvant garantir par design que cela ne pose pas de problème quand la réception est activée. Le soft de vol ne doit pas activer la réception sur l'APBUART. Il faut peut-être l'ajouter dans la SRS?

1. Par design:
   - dans **fsw_init.c**:

```
    // UART settings
    send_console_outputs_on_apbuart_port();
    set_apbuart_scaler_reload_register(REGS_ADDR_APBUART, APBUART_SCALER_RELOAD_VALUE);
    enable_apbuart_transmitter();

    DEBUG_PRINTF("\n\n\n\n\nIn INIT *** Now the console is on port COM1\n")

    PRINTF("\n\n\n\n\n")
```

   - dans **DEV_PLE/src/fsw_misc.c**:

```
int send_console_outputs_on_apbuart_port( void )
// Send the console outputs on the apbuart port
{
    struct apbuart_regs_str *apbuart_regs = (struct apbuart_regs_str *) REGS_ADDR_APBUART;

    apbuart_regs->ctrl = APBUART_CTRL_REG_MASK_TE;

    return 0;
}

int enable_apbuart_transmitter( void )
// set the bit 1, TE Transmitter Enable to 1 in the APBUART control register
{
    struct apbuart_regs_str *apbuart_regs = (struct apbuart_regs_str *) REGS_ADDR_APBUART;

    apbuart_regs->ctrl = apbuart_regs->ctrl | APBUART_CTRL_REG_MASK_TE;

    return 0;
}

void set_apbuart_scaler_reload_register(unsigned int regs, unsigned int value)
{
    /** This function sets the scaler reload register of the apbuart module
     *
     * @param regs is the address of the apbuart registers in memory
     * @param value is the value that will be stored in the scaler register
     *
     * The value shall be set by the software to get data on the serial interface.
     *
     */
```

```
    struct apbuart_regs_str *apbuart_regs = (struct apbuart_regs_str *) regs;

    apbuart_regs->scaler = value;
    BOOT_PRINTF1("OK  *** apbuart port scaler reload register set to 0x%x\n", value)
}
```

Au passage je ne vois pas trop l'intérêt d'appeler **enable_apbuart_transmitter** et **send_console_outputs_on_apbuart_port**.

1. Pour la vérification sur cible:

Il faut lire le registre de contrôle de l'APBUART et vérifier le bit **RE**(Bit 0), celui-ci doit être à **'0'**.

APBUART_CTRL_REG.png
L'APBUART est à l'adresse **0x80000100** et l'offset du registre de contrôle est **0x8** donc l'adresse du registre de contrôle est **0x80000108**.

History

#1 - 25/01/2016 04:02 PM - paul leroy

- Assignee changed from paul leroy to Veronique bouzid

J'ai enlevé la fonction qui ne servait pas.

De mon côté, j'initialise avec le soft le registre de contrôle de l'APBUART en mettant le bit 0 à '0'. Ceci dit, ça doit être sa valeur au reset matériel.
Une vérification après POWER-ON et avant lancement du logiciel permettra de s'en assurer sur la carte sous test.

Pour ma MINI-LFR avec VHDL 0.1.89, le bit 0 est à '0' à la mise sous tension.

#2 - 21/06/2018 01:37 PM - Veronique bouzid

- Assignee changed from Veronique bouzid to Alexis Jeandet

#3 - 22/03/2021 12:20 PM - Alexis Jeandet

- Status changed from New to Closed

Il y a un strap sur le connecteur de l'UART pour maintenir un état stable sur le RX.

Files

APBUART_CTRL_REG.png	422 KB	03/11/2015	Alexis Jeandet