

LibUc Porting Guide

Developpers guide

Alexis JEANDET

October 2011

Table des matières

1	Stream devices	5
1.1	The stream device representation	5
1.2	UART	6
1.2.1	Generals :	6
1.2.2	Details	6
2	Addressed devices	7

Chapitre 1

Stream devices

Introduction This family is composed of all the devices that treat data as an unaddressed stream. For example SPI and UART are both stream devices, a software or hardware FIFO is also a stream device.

1.1 The stream device representation

The stream device is represented by a standard structure.

```
struct streamDev
{
    int  writen(void* data, int n);
    int  write(void* data);
    int  readn(void* data, int n);
    int  read(void* data);
    int  streamPt;
    void* dev;
}
```

1.2 UART

1.2.1 Generals :

On many microcontrolers you can use **U**niversal **A**synchronous **R**eceiver **T**ransmitter, in order to provide a standart access to this device you have to implement th folowing functions :

- `uart_t uartopen(int)` ;
- `uartclose(uart_t)` ;
- `uartsetup(uart_t,int,int)` ;
- `uartputc(uart_t, char)` ;
- `char uartgetc(uart_t)` ;
- `uartputstring(uart_t,char*)` ;
- `uartgetstring(uart_t,char*)` ;

1.2.2 Details

Opening uart

`uart_t uartopen(int)`

This function has to open N^e **uart** device, so it has to perform basical configuration such as powering up the device ...This fnction has also to return the pointer to the devicce handle.

Closing uart

`uartclose(uart_t)`

Configure uart

`uartsetup(uart_t,int baudrate,int cpuclk)`

Sending a char through uart

`uartputc(uart_t, char)`

Receiving a char from uart

`char uartgetc(uart_t)`

Sending a string through uart

`uartputstring(uart_t,char*)`

Receiving a string from uart

`uartgetstring(uart_t,char*)`

Chapitre 2

Addressed devices

Introduction This family is composed of all the devices that treat data as an addressed space. For example IIC and memories are both addressed devices.