Embedded Systems

Laboratory 1

Introduction

The aim of the laboratory classes is the configuration of the programming environment (gdb, cross compiler) and getting familiar with basic functionality of ARM AT91SAM9263 processor and its input/output ports.

The control of the LED diodes and button operation servicing will be a final result of the task performed on the evaluation kit.

Task list

- Programming environment configuration.
- Testing of the example program (downloaded from Embedded System web page).
- Compilation of simple program.
- IO port configuration for USERLED1 diode.
- Turning on and turning off of the USERLED1 diode.
- Blinking of the USERLED1 diode in the main loop.
- Button LEFTCLIC (BP1) click servicing.
- USERLED1 turning on with the LEFTCLIC (BP1) button pressed.
- USERLED1 turning off with the RIGHTCLIC (BP2) button pressed.
- USERLED2 blinking.

Environment configuration

Create directory `lab1` and `openocd` within the main directory. Copy the following files from the directory `/opt/arm_user` to your laboratory account:

- `.bashrc` – to main user directory,
- `.gdbinit` – to main user directory,
- `openocd` – to `openocd` directory,
- `openocd.cfg` – to `openocd` directory.

Additionally from the subject webpage copy file ‘Makefile for C’ into `lab` directory.

The next step is running new console session in order to apply the settings from the `.bashrc` file. In order to check if this step is performed correctly type `arm-elf-gcc` and check if the compiler has been found. You should see `arm-elf-gcc: no input files` on the console.

Create the file `main.c` in the `lab1` directory for your first program. In this file insert an empty function void `dbgu_print_ascii(const char *a) {}` used when the program is compiled.

Example program

On the subject webpage there is an example file LED test (`led_blink.elf)` with the compiled program. It is recommended to upload this program first in order to check if the environment and hardware are configured properly. You should copy the file `led_blink.elf` into `lab1` directory.

Run the program `./openocd` located in `openocd` directory in one console. This program is for the communication with processor and must be open all the time.

With the second console upload the program to the microprocessor running the debugger with the command `arm-elf-gdb led_blink.elf`. When the program is loaded use the command `run` in the gdb console to run it.

To break the debugger operation press `ctrl-c` and type `quit` command in debugger console. If the program was running on the microprocessor you must confirm breaking its execution by pressing `y`.

LED diodes control

There are two LED diodes and two buttons directly connected to the processor pins on the evaluation board. The schematic is included in the board documentation available on the subject webpage.

In order to simplify the task perform the following steps.

Draw the schematic of the LED connection
Search the schematic of the evaluation board.

Draw the schematic of the button connection
Search the schematic of the evaluation board.

Answer the following questions
Look at the schematic above and analyze if the diodes and buttons are connected to ground or plus.

USERLED1 diode is connected to the pin	
USERLED2 diode is connected to the pin	
In order to turn on the diode the pin state must be	
RIGHTCLIC button is connected to the pin	
LEFTCLIC button is connected to the pin	
Depressing the button causes the pin state to be	

Write down the registers necessary to configure IO port
All the processor registers required for LED diodes and button operation are described in chapter 31 of the AT91SAM9263 processor manual available on the subject webpage. Information about clock is specifically included in chapters „31.4.8 Inputs” and “28. Power Management Controller (PMC)”.

Operation	**Registers**
Setting IO port in IO mode	
Turning on the clock for IO port	
Setting the direction of IO port	
Forcing the required state on port output	
Reading the port input state	

Program compilation and running

The command `make` compiles the program.

After successful compilation the program is uploaded to microprocessor when the debugger is run with `arm-elf-gdb led_blink.elf` and the command `run` is typed (as with the example program).

If the `openocd` from the directory `./openocd` is not run it is necessary to start it to keep the communication with the processor.