TDT4258
Energy Efficient Computer Design
Exercise 1

Lundal, Per Thomas
perthol@stud.ntnu.no

Normann, Kristian
krinorm@stud.ntnu.no

Selvik, Andreas Løve
andrels@stud.ntnu.no

February 7, 2013

Abstract

JUST PLOTTING DOWN SOME DRAFT-IDEAS FOR
SUBJECTS TO MENTION
Energy-efficient computers, programming and simply awareness, have
over the last few decades become a more and more pressing issue.

In this assignment we go about the specified task first,

Contents

I   Introduction

Introduction goes here

This assignments's stated task was to create an assebly program that would turn on the central LED on a given microcontroller, and then allow a user to move the light either left or right. In a sense, this task was solved in three subsequent steps. The first step was to turn the lights on by hardcoding them. The second was to activate the buttons and connect them to the lights through simple(and inefficient) polling. The third step was to arrange for the buttons to work by way of interrupts and altering the code so that it would adhere to the assignment text in terms of keeping the LED-switching in the main-loop.

There was however, another, more holistic side to this task. Namely energy efficiency. This, aside from the fact that implementing the buttons via interrupts instead of polling is more energy-efficient and was part of the explicit assignment, is mostly due to actual name of the course: Energy Efficient Computer Design.

II   Description and methodology

Methodolololololology
The initial step in solving the task was to read up on the technical details on the AT32AP7000 microcontroller, the STK1000 development-board and the the AVR32 RISC instruction set in order to figure out how to actually go about solving the assignment. Then a plan of four subsequent parts was made.

1. Being able to program the lights in a functional manner by hard-coding a light-arrangement.

2. connect the buttons to their corresponding LED's, and putting this code in a polling loop such that pushing them will turn on those lights.

3. Making the button code interrupt based and putting all the code that altered the LED's in the main loop.

4. The optimization with respect to energy. That is, putting the microcontroller in sleep 1, whenever feasible and reducing the amount of code where some things could be done with fewr lines of code.

III   Results and tests

Results, wooohoo!

IV   Evaluation of assignment

Evaluation...

V   Conclusion

It all went well.

VI   References

Bibtex will handle this part.