High Performance Embedded Systems

Assignment 1

Group #:	5
Final Grade:	42.50%
Additional comments	**Git repo was checked out at commit 1040f0c - Jun 26th 23:59:12**

Part I: QEMU

Item	Description	Total	Grade	Comments
Full boot deliverables	- images/ - u-boot-qemu.log - linux-qemu.log - qemu_boot.log - video.txt	15%	13.00%	-2%: qemu_boot.log is incomplete.
Full boot execution	Correct execution of uboot, kernel and file system on QEMU based on video and local testing.	25%	15.00%	-10%: video and instructions used are not functional - the kernel hangs after loading the image: **Why did the kernel hang?** Actually the kernel did not hang, well at least it did not at the begining. What happened is that you used the following command to set the boot args: setenv bootargs 'root=/dev/mmcblk0p2 rw console=${console} mem=512M vmalloc=256M earlyprintk' Look at the single quotes **'**. When you use single quotes on a u-boot variable it does not allow the variables to be expanded so console final value is: bootargs=root=/dev/mmcblk0p2 rw **console=${console}** mem=512M vmalloc=256M earlyprintk When you remove the quotes as follows: setenv bootargs root=/dev/mmcblk0p2 rw console=${console} mem=512M vmalloc=256M earlyprintk The final result is: bootargs=root=/dev/mmcblk0p2 rw **console=ttyAMA0**,38400n8 mem=512M vmalloc=256M earlyprintk So actually, in the first case it is not like the kernel is hanging (well it will hang at some point) but it is sending the output of the boot log to the wrong serial console so you can not see anything.
Bonus	Add boot commands to u-boot environment	10%		

Part II: Yocto & NEON

Item	Description	Total	Grade	Comments
Yocto meta-layer	Correct creation of the meta-tec layer and tec-image. Inherits from the console-image. Adds all new recipes into the tec-image and includes them in the final image.	5%	5.00%	- images directory usually goes at the root level, it does not require to be nested into another recipes-* directory.
Yocto recipes	Correct creation of the following recipes: - rgb2yuv_c - rgb2yuv_intrinsics This includes correct build and installation of the binaries into the file system. Also it includes the presence of the required PDF documentation.	10%	2.50%	-2.5%: Instrinsics recipe is not functional. -5%: C and Intrinsics documentation is missing
Autotools usage	Correct usage of autotools for the compilation of the programs.	10%	0.00%	- Autotools was not used at all.
Getopt implementation	Correct usage of getopt for the command line options	5%	0.00%	- This was so easy to earn :(
Application implementation and functionality	The application meets with the requirements proposed and is totally functional providing the correct image conversion.	25%	5.00%	- C application did not follow any of the requirements and it is totally hard coded. Just 5% for implementing some code there.

General

Item	Description	Total	Grade	Comments
Git control versioning	The delivery is correctly made using a git repository with the layout suggested and following the required work flow. User participation from all team members. Avoid using a unified commit or last hour commits (development must be continuous)	5%	2.00%	- Did not follow the work flow: usually only develop branch should merge to master (with exception of releases and hot-fixes). - Definitely there was a poor organization within the team, the contributions where not equal nor significant in some cases.