

Embedded Systems with uCLinux based on FPGA

by

Siddharth Kamath

February 17th , 2010

An Embedded System based on FPGA

- The Hardware architecture
 - Design and selection of hardware platform
 - uCLinux porting of hardware
 - Application: Designing an mpeg2-player
-

FPGA's so far...

- Configured by the customer
 - Low NRE as compared to an ASIC
 - Supports high speed computing
 - Compatible with most modern I/O's
 - Processors can be implemented in soft-cores
-

Hardware architecture

- SOPC (System on a Programmable Chip) used to design the hardware
- SOPC – A software by Altera which automates connecting software – hardware components
- Peripherals designed :
 - Ethernet controller
 - SD card controller
 - LCD controller
 - USB controller

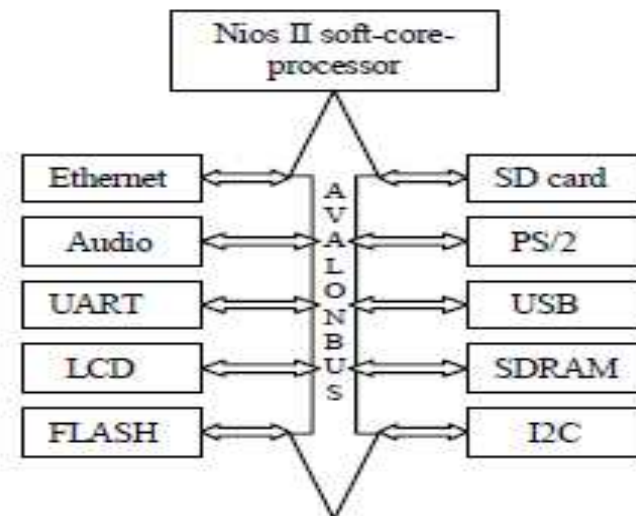


Fig.1 Block diagram of hardware architecture

Hardware Platform

- Development board: Altera DE2-70
- Core: Cyclone II EP2C70 FPGA
 - 68416 logic elements
 - 250 M4K RAM blocks (1152000 bits)
- Other on-board peripherals:
 - FLASH and SDRAM
 - Ethernet
 - USB
 - Audio card
 - RS232
 - PS/2

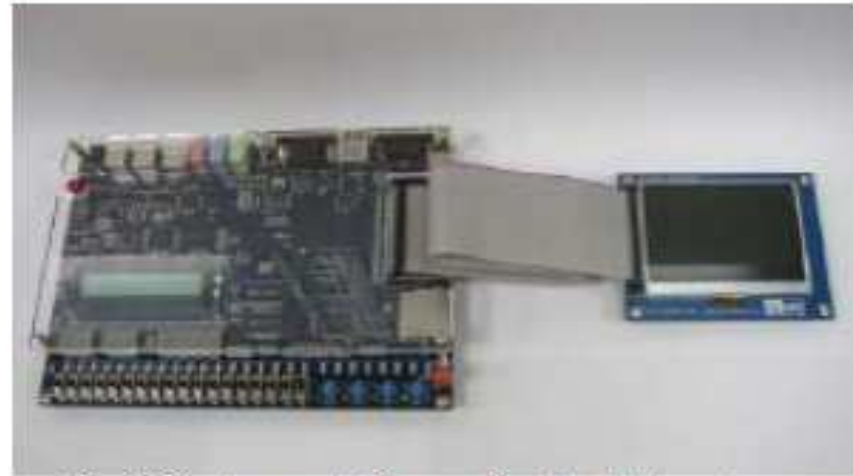


Fig.2 Hardware platform of embedded system

Hardware design using SOPC builder

cpu	Nios II Processor	sys_clk	0x00000000
flash_tristate_bridge	Avalon-MM Tristate Bridge	sys_clk	
ext_flash	Flash Memory (CFI)	sys_clk	0x00000000
sdram	SDRAM Controller	sys_clk	0x02000000
sys_clk_timer	Interval Timer	sys_clk	0x00001800
ISP1362	EP1362_IF	multiple	multiple
led_i2c_scl	PIO (Parallel IO)	sys_clk	0x00001800
led_i2c_en	PIO (Parallel IO)	sys_clk	0x00001900
led_i2c_sdat	PIO (Parallel IO)	sys_clk	0x00001920
ps2_0	PS2 Serial Port	sys_clk	0x00000010
dm9000	DM9000A_IF	sys_clk	0x00000000
uart	UART (RS-232 Serial Port)	sys_clk	0x00001880
led_sgdma	Scatter-Gather DMA Controller	sys_clk	0x00001400
led_ta_sgdma_to_fifo	Avalon-ST Timing Adapter	sys_clk	
led_pixel_fifo	On-Chip FIFO Memory	multiple	
led_ta_fifo_to_dfa	Avalon-ST Timing Adapter	sys_clk	
led_64_to_32_bits_dfa	Avalon-ST Data Format Adapter	sys_clk	
led_pixel_converter	Pixel Converter (BGR0 -> BGR)	sys_clk	

Fig.3 Design of hardware of FPGA

Software Design with uCLinux

- uClinux incorporated as an OS on the NIOS II processor
 - uClinux originally designed for Motorola 68000 chip
 - Why uClinux ?
 - uClinux is designed for processors without MMU
 - Open source
 - Powerful network functions
 - Efficient and robust file system
 - Kernel version 2.6.26 used included in nios2-linux-20080619.tar
 - Custom ports can be added
-

The uClinux architecture

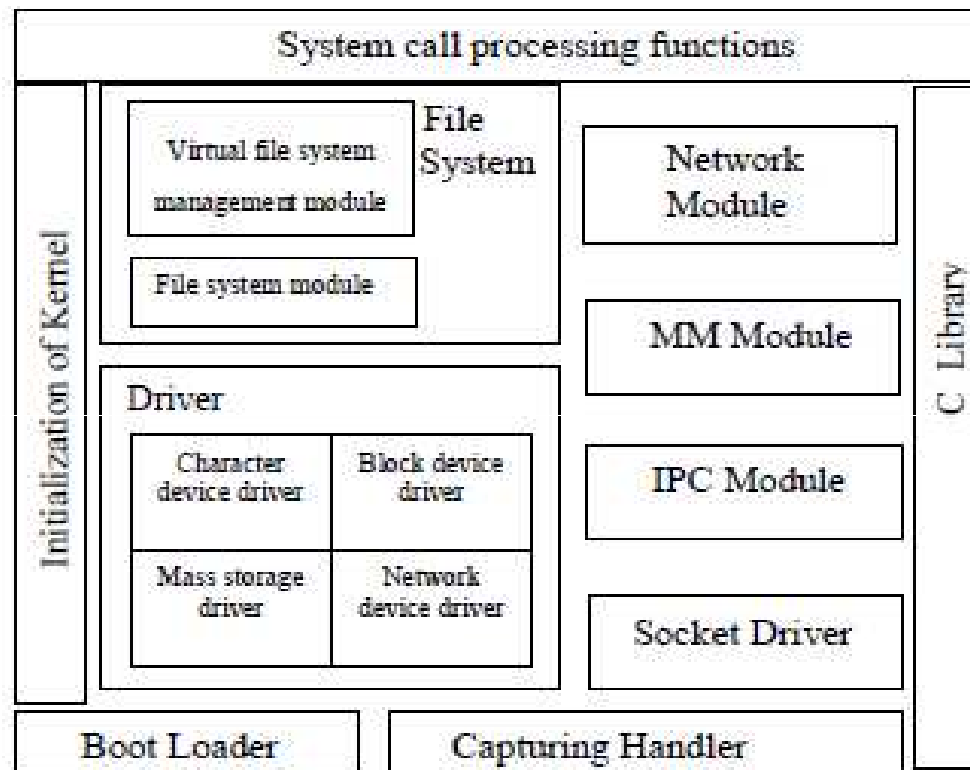


Fig.4 Architecture of uClinux

- All devices listed under the OS can be used
- Example: SD card as the root device, USB as keyboard, PS/2 as the mouse

Applications

- Nano-X : An open source GUI for embedded systems
- SDL : The Simple Direct Media Layer which is a cross platform multimedia library
- Links : A fast text and graphical web browser
- BusyBox: A set of many common UNIX utilities as a single executable

In short....Its an on-board computer system on FPGA

Design of an MPEG2-player

- Design based on libmpeg2- a stream decoder
 - Audio device needed to play MPEG2 files
 - The on-board audio device on DE2-70 used
 - Challenge: No audio drivers under the NIOS II platform
-

Control of the audio device

- Method 1: Writing device drivers
 - Disadvantage: Time consuming
High rate of error
 - Method 2: Directly read/write data using the port address
 - Advantage: Hassle free
Easy to debug
-

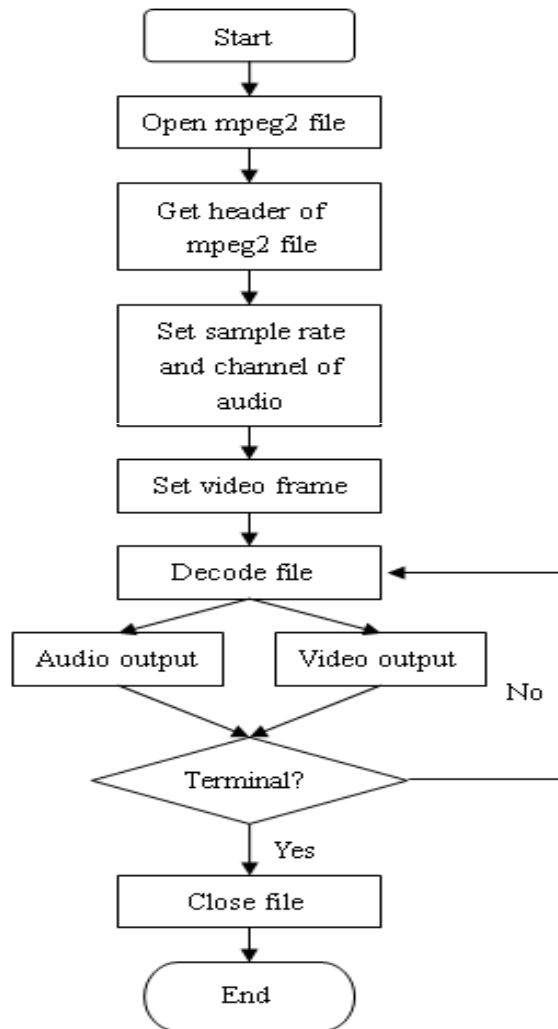
Initializing the audio device

- Only write access required in an audio device
 - Initiate audio device by initializing the I2C port
 - Set the sample rate and audio channel
 - Write data to audio device
 - Control FIFO of audio components
-

Code excerpt to initialize the I2C port

```
int AUDIO_Init(void){
    int complete = 1;
    printf(("AUDIO_Init...\r\n"));
    if(complete)
        complete = I2CWrite(15, 0x0000);
    if(complete)
        complete = I2CWrite(9, 0x0000);
    if(complete)
        complete = I2CWrite(0, 0x0017);
    if(complete)
        complete = I2CWrite(1, 0x0017);
    if(complete)
        complete = I2CWrite(2, 0x005B);
    if(complete)
        complete = I2CWrite(3, 0x005B);
    if (complete)
        complete = I2CWrite(4, 0x0015 | 0x20 | 0x08 | 0x01);
    if (complete)
        complete = I2CWrite(5, 0x0000);
    if (complete)
        complete = I2CWrite(6, 0);
    if (complete)
        complete = I2CWrite(7, 0x0042);
    if (Complete)
        Complete = I2CWrite(8, 0x0002);
    if (Complete)
        Complete = I2CWrite(9, 0x0001);
    printf("audio_Init %s\r\n",complete?"sucess":"fail");
    return complete;
}
```

MPEG2-player design flowchart



1. Start
2. Open file in read/write mode (mpeg_open)
3. Read the header
4. Sample rate for audio (mpeg_has_audio)
5. Frame rate for video (mpeg_has_video)
6. Decode files and send to their respective audio or video devices
7. Close the mpeg2 file

Performance and Conclusion

- System is highly reconfigurable
 - Limited processor performance
 - Soft-core processors have more flexibility
 - High scalability – Peripherals can be easily added
 - Multicore softcore processors can be designed to improve performance
-

References

- [1] Zongqing Lu, Xiong Zhang, Chuiliang Sun, "An Embedded System with uClinux Based on FPGA," pacia, vol. 2, pp.691-694, 2008 IEEE Pacific-Asia Workshop on Computational Intelligence and Industrial Application, 2008
- [2] Tyson S. Hall, James O. Hambien (2007). "Using an FPGA Processor Core and Embedded Linux for Senior Design Projects". 2007 IEEE International Conference on Microelectronic Systems Education (MSE'07).
- [3] Xilinx, Inc., <http://www.xilinx.com>, 2008.
- [4] Altera Corp., <http://www.altera.com>, 2008.
- [5] Libmpeg2, <http://libmpeg2.sourceforge.net>, 2008.
- [6] uClinux distribution, <http://www.uclinux.org>, 2008.
- [7] InstallNios2Linux,
<http://nioswiki.jot.com/WikiHome/InstallNios2Linux>, 2008.
- [8] Nano-X, <http://www.microwindows.org>, 2008.