

基於FPGA之16位元生醫訊號處理 器實現

指導教授 闕河鳴

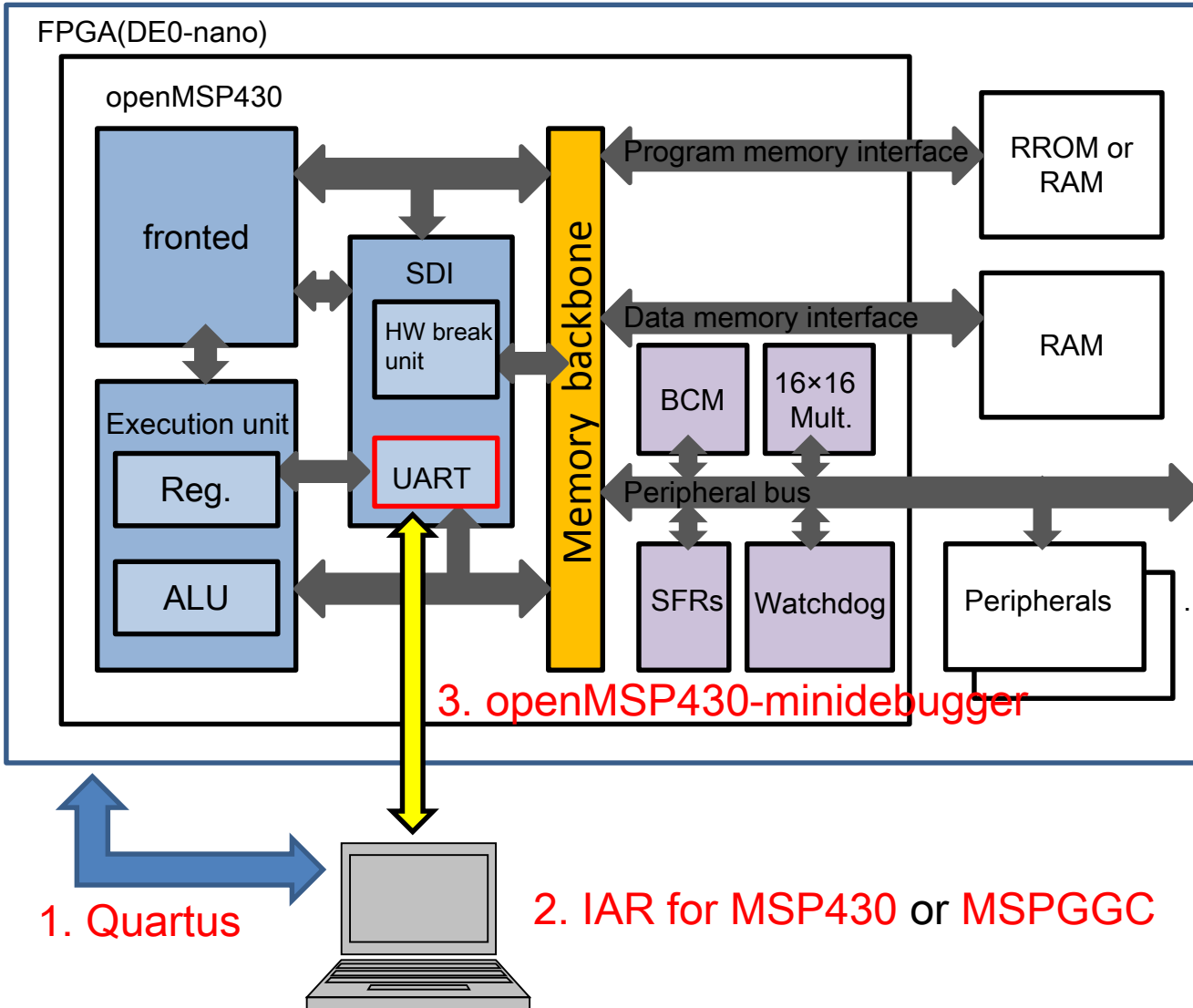
學生 吳嘉偉

學號 9923075

簡介

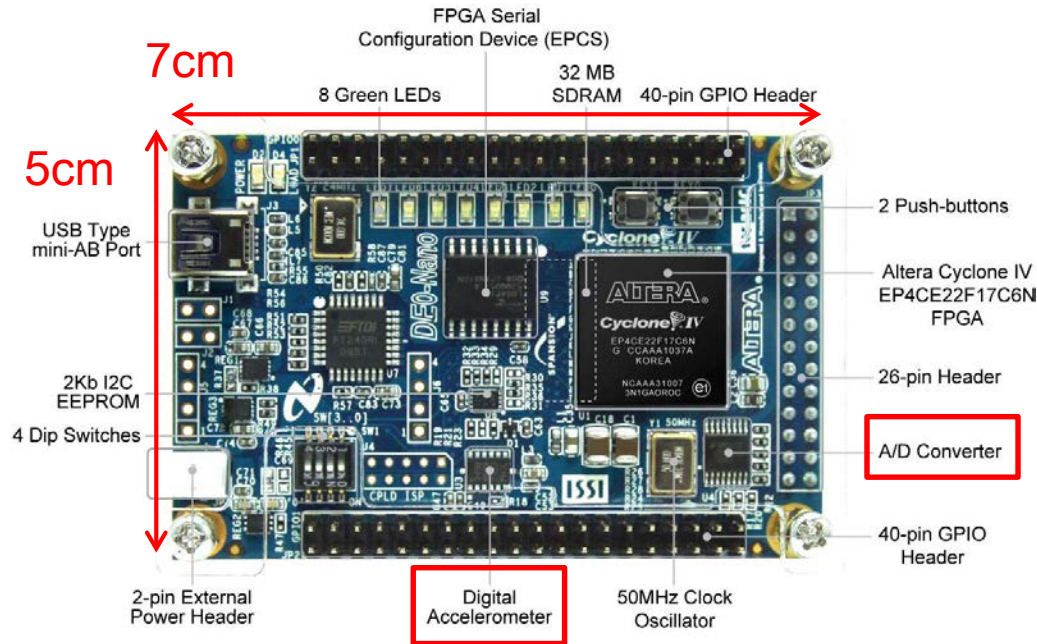
- 近年來，積體電路在個人醫療中扮演著重要的角色。考量生理訊號擷取的方便性及設計的彈性，我們以FPGA為基礎；利用openMSP430的架構，以達到低功率高效能的需求。
- 設計動機：
 - 以FPGA設計可依需求做調整，設計專門處理生醫訊號之處理器發展平台
 - MSP430具有低功率高效能的特徵，符合醫療應用的需求，使用openMSP430使得FPGA達到接近MSP430的功能
 - 網路上已有MSP430編譯器相關資源，可直接應用於openMSP430，方便軟體開發

系統架構



1. 使用Quartus 燒錄 openMSP430至 DE0-nano上
2. 使用MSP430 compiler編輯程式，並將軟體編譯，產生物件檔
3. openMSP430-minidebugger 透過UART將物件檔裝載於 openMSP430上執行所設計程式

硬體設備



- 使用7cm*5cm方便攜帶的ALTERA的DE0-nano
- openMSP430所用到的元件(GPIO、RAM、ROM)已對應至DE0-nano上，並將系統做flash PROM的燒錄，關閉電源之後不用再重新燒錄
- 目前正在開發將DE0-nano上的元件(ADC、G-sensor)加入openMSP430中，擴大系統的功能

實際操作

```
C:\Windows\system32\cmd.exe

C:\msp430\bin>msp430-gcc test.cpp -o test.elf
c:/msp430/bin/./lib/gcc/msp430/4.6.3/./../../../../
linker script file memory.x: No such file or direct
collect2: ld returned 1 exit status

C:\msp430\bin>msp430-gcc test.cpp -o test.elf
c:/msp430/bin/./lib/gcc/msp430/4.6.3/./../../../../
linker script file periph.x: No such file or direct
collect2: ld returned 1 exit status

C:\msp430\bin>msp430-ld test.elf memory.x final.elf
msp430-ld: cannot open linker script file periph.x:
msp430-ld: cannot open linker script file periph.x:

C:\msp430\bin>msp430-gcc test.cpp -o test.elf
c:/msp430/bin/./lib/gcc/msp430/4.6.3/./../../../../
linker script file periph.x: No such file or direct
collect2: ld returned 1 exit status

C:\msp430\bin>msp430-gcc test.cpp -o test.elf
```

使用MSPGCC進行編譯

The screenshot shows the openMSP430 mini debugger interface. The top panel displays device selection options (Core 0-3) and connection settings (Device Port: COM6, Speed: 115200). The CPU is connected and running. The ELF file path is C:/msp430/bin/test.elf. The CPU control panel shows buttons for Reset, Stop, and Step, with the CPU status set to Running. The registers table is visible, showing the PC register at 0xf06c. The disassembly window on the right shows the assembly code for the main function, with the instruction at address 0xf06c highlighted: `0xf06c: b4 90 80 07 cmp #2000, -8(r4); #0x07d0, 0xffff(r4)`.

r0 (pc):	Address	Data
r1 (sp):	0x0200	0x5a28
r2 (sr):	0x0202	0x0000
r3:	0x0204	0x0000
r4:	0x0206	0x0000
r5:	0x0208	0x0000
r6:	0x020a	0x0000
r7:	0x020c	0x0000
r8:	0x020e	0x0000
r9:	0x0210	0x0000
r10:	0x0212	0x0000
r11:	0x0214	0x0000
r12:	0x0216	0x0000
r13:	0x0218	0x0000
r14:	0x021a	0x0000
r15:	0x021c	0x0000
r16:	0x021e	0x0000
r17:	0x0220	0x0000
r18:	0x0222	0x0000
r19:	0x0224	0x0000
r20:	0x0226	0x0000
r21:	0x0228	0x0000
r22:	0x022a	0x0000
r23:	0x022c	0x0000
r24:	0x022e	0x0000
r25:	0x0230	0x0000
r26:	0x0232	0x0000
r27:	0x0234	0x0000
r28:	0x0236	0x0000
r29:	0x0238	0x0000
r30:	0x023a	0x0000
r31:	0x023c	0x0000

使用openMSP430-mini debugger
成功裝載程式於openMSP430上
並可監看程式碼進行除錯

總結及未來展望

- 使用Quartus編寫verilog完成openMSP430對應至FPGA上腳位的設定，並做flash PROM的燒錄
- 可使用MSP430的編譯器產生物件檔，透過openMSP430-minidebugger裝載程式於openMSP430上，並進行驗證，成功執行所設計之程式
- 未來希望加入ADC於openMSP430的架構中，實現低功率混合訊號處理器