Zürcher Hochschule
für Angewandte Wissenschaften

School of Engineering

InES Institute of Embedded Systems

TSM_EmbHardw
Exercises Week 11 / Lecture 3

Exercise Set A

Project 2: Apply cache and/or scratchpad aware programming techniques to optimise performance of the Sobel algorithm.

Exercise Set B

Exercise 1: pre-fetch Instructions
Can a NIOS II custom instruction be used to implement a data prefetch instruction? Explain your answer.

Exercise 2: pre-fetch distance
Given the code below, an average memory latency (miss) of 100 clock cycles and a loop iteration time of 45 cycles modify the code below for this pre-fetch distance

```
fetch( &ip);
fetch( &a[0]);
fetch( &b[0]);

for (i = 0; i < N-4; i+=4){
      fetch( &a[i+4]);
      fetch( &b[i+4]);
      ip = ip + a[i]  *b[i];
      ip = ip + a[i+1]*b[i+1];
      ip = ip + a[i+2]*b[i+2];
      ip = ip + a[i+3]*b[i+3];
}
for ( ; i < N; i++)
      ip = ip + a[i]*b[i];
```

(d)

Exercise 3: NIOS II Cache
The NIOS implements four instructions for data cache initd(a) and flushd(a), considering and not considering the tag line – why?

Exercise 4: Instruction Cache
How can our Sobel code for the NIOS be optimised for instruction cache awareness? Present a solution.

Exercise 5: Tiling

Zürcher Hochschule
für Angewandte Wissenschaften

School of
Engineering

InES Institute of
Embedded Systems

TSM_EmbHardw
Exercises Week 11 / Lecture 3

Tile the following code

```
double a[m][n], b[m][n], c[m][n];
...
for (i=0; i < m; i++) {
   for (j=0; j < n; j++) {
      a[i][j] = b[i][j] + c[i][j];
   }
}
```

Exercise 6: SPM

Given the following overlay definition

```
SECTIONS {
     OVERLAY {
          function_1()
          function_3()
     }
     OVERLAY {
          function_2()
     }
}
```

Will the following code execution function? If not fix it.

```
for (i = 0; i<=5; i++) {
     function_1();
     function_3();
}
for (i = 0; i<=5; i++) {
     function_2();
}
```

Zürcher Hochschule
für Angewandte Wissenschaften

School of
Engineering

InES Institute of
Embedded Systems

TSM_EmbHardw
Exercises Week 11 / Lecture 3

Exercises: Exam Preparation

Exercise 1: Data Cache Aware Loop Unrolling

Assume the NIOS II has a cache line size of 4 bytes. Optimise the loop unrolling of the following code using this knowledge.

```
void rgb_to_grayscale( int width,
                       int height,
                       const unsigned int *rgb_source,
                       unsigned int *grayscale_destination) {
   int loop;
   unsigned int temp;
   unsigned int grayscale;

   for (loop = 0 ; loop < width*height ; loop++) {
      temp = rgb_source[loop]&0x3F; // red value
      grayscale = (temp*30)/100;
      temp = (rgb_source[loop]>>8)&0x3F; // green value
      grayscale += (temp*59)/100;
      temp = (rgb_source[loop]>>16)&0x3F; // blue value
      grayscale += (temp*11)/100;
      grayscale_destination[loop] = grayscale|
                                    (grayscale<<8)|
                                    (grayscale<<16);
   }
}
```

Exercise 2: Data Cache Aware Loop Unrolling

The NIOS II reads memory into cache in bursts, one clock cycle per line size i.e a 4-byte line is read in five clock cycles (including a setup-clock cycle). With this knowledge unroll the following loop and indicate where you would insert pre-fetch statements. Explain your reasoning.

Zürcher Hochschule
für Angewandte Wissenschaften

School of
Engineering

InES Institute of
Embedded Systems

TSM_EmbHardw
Exercises Week 11 / Lecture 3

```
void rgb_to_grayscale( int width,
                       int height,
                       const unsigned int *rgb_source,
                       unsigned int *grayscale_destination) {
   int loop;
   unsigned int temp;
   unsigned int grayscale;

   for (loop = 0 ; loop < width*height ; loop++) {
      temp = rgb_source[loop]&0x3F; // red value
      grayscale = (temp*30)/100;
      temp = (rgb_source[loop]>>8)&0x3F; // green value
      grayscale += (temp*59)/100;
      temp = (rgb_source[loop]>>16)&0x3F; // blue value
      grayscale += (temp*11)/100;
      grayscale_destination[loop] = grayscale|
                                    (grayscale<<8)|
                                    (grayscale<<16);
   }
}
```