



Low Level C-programming and microcomputer architecture

Task 7 Code reduction

Versionshistorik

Version	Datum	Ansvarig	Beskrivning
0.0	2013-03-14	LL	Preliminär version
0.1	2013-04-22	ML	Reviderad version
0.2	2014-10-19	LL	Tog bort "exceptions"
0.2.1	2015-09-21	LL	Bytt plats på vecka 6 och 7, nu har uppgiften nr 7. Tog bort led funktionen.
0.3	2016-08-15	ML	Uppdaterad

Innehåll

1 Inledning.....	1
2 Kravspecifikation.....	1
Bilagor	3
Bilaga A Tabell som ska fyllas i	3
Bilaga B Programkod	4
Bilaga C Minnestyper	5

1 Inledning

En av företagets kunder vill kostnadseffektivisera genom att reducera mängden minne med vissa optioner som kan väljas i Alteras verktyg. Chefen har gett ingenjören jobbet att genomföra detta och leverera resultatet till kunden. Om något är oklart kan frågor ställas till chefen.

Konstruktionen behöver inte valideras utan syftet är att ta reda på hur stor koden blir före nerladdning.

2 Kravspecifikation

Följande ska genomföras för att uppgiften ska anses slutförd:

Tabell 1 Kravspecifikation

Krav	Beskrivning	Utfört Ja/nej
Förstudie		
Krav_001	Skriv en kort sammanfattning av teorilektion 7 (Theory_7) vilken ska ingå i rapporten enligt krav_006.	
Funktionskrav		
Krav_002	Skapa ett applikationsprojekt med C-programmet i bilaga 2.	
Krav_003	Genomför följande steg och fyll i tabellen som finns i bilaga 1.	

	<p>Steg 5 har alla optimeringarna, alltså bygg på optimeringarna i steg.</p> <p>Ej reducerad kod</p> <p>Ingen optimering, utgå ifrån "hello world".</p> <p>Steg 1</p> <p>I Nios II BSP Editorn/Main/Settings/Advanced aktivering av enable "<i>reduced_device_drivers</i>".</p> <p>Steg 2</p> <p>I Nios II BSP Editorn/Main/Settings/Advanced aktivering av <i>enable_small_c_library</i></p> <p>Steg 3</p> <p>I Nios II BSP Editorn/Main/Settings/Advanced aktivering av <i>enable_lightweight_device_driver_api</i> deaktivering av <i>enable_c_plus_plus</i>, <i>enable_clean_exit</i> och <i>enable_exit</i>.</p> <p>Steg 4</p> <p><i>printf</i> i C-koden ersattes med <i>alt_printf</i></p> <p>Steg 5</p> <p>I Nios II BSP Propertyts sätts <i>Optimization Level</i> till <i>Size</i>.</p>	
Krav_004	Beskriv minnestyperna i ett separat kapitel i rapporten (krav_007).	
Krav_005	Redovisa resultaten från stegen ovan i ett separat kapitel i rapporten (krav_007). Frågeställning: Varför "minskar"/"minskar inte" kodmängden? Gör en kort analys.	
Dokumentationskrav		
Krav_006	Sammanfoga dokumentationen från krav_001 till krav_005 till en läsbar rapport. Framsida med titel, en kort sammanfattning, innehållsförteckning och separata kapitel enligt krav_001 till krav_005 ovan. Lägg även till eventuella slutsatser och referenser.	
Leveranskrav		
Krav_007	Leveransen ska ske till plattformen Itslearning. Leveransen ska vara en kort rapport. Namnet på filen ska vara "förnamn_efternamn_C_task_7". Sista leveransdag se kursschema (för VG).	

Bilagor

Bilaga A Tabell som ska fyllas i

Den minnesmängd som behövs efter varje steg har hämtats (xxx.objdump).

Tabell 2 Minnesmängd (exempel, kan minskas ner med minnestyper)

Minnestyp	Ej reducerad kod	steg 1	steg 2	Steg 3	Steg 4	Steg 5
.entry + .exceptions						
.text						
.rodata						
.rwdata						
.bss						
Total summa						
Reducering [%] OBS, det är hex!	100%					



Bilaga B Programkod

```
#include <stdio.h>
#include <io.h>
#include <alt_types.h>
#include <system.h>
// #include "altera_avalon_pio_regs.h"

void delay_function(int time_delay)
{
    int i;
    for(i=1; i<time_delay;i++);
    for(i=1; i<time_delay;i++);
}

int main()
{
    Alt_32 led_value;
    // if you get error, please look into the file "alt_types.h"

    // loop for ever
    while(1)
    {
        led_value = 0;
        printf("Led: %x\n", led_value);

        // IOWR_ALTERA_AVALON_PIO_DATA(PIO_OUT_LED_BASE, led_value);

        delay_function(500000);
        led_value = 0xF;
        printf("Led: %x\n", led_value);

        // IOWR_ALTERA_AVALON_PIO_DATA(PIO_OUT_LED_BASE, led_value);
    }
    return 0;
}
```

Tabell 3 printf syntax (ref <http://www.cplusplus.com>)

Specifier	Description	Characters extracted
i, u	Integer	Any number of digits, optionally preceded by a sign (+ or -). decimal digits assumed by default (0-9), but a 0 prefix introduces octal digits (0-7), and 0x hexadecimal digits (0-f).
d	Decimal integer	Any number of decimal digits (0-9), optionally preceded by a sign (+ or -).
x	Hexadecimal integer	Any number of hexadecimal digits (0-9, a-f, A-F), optionally preceded by 0x or 0X, and all optionally preceded by a sign (+ or -)



Bilaga C Minnestyper

Minnesåtgången delas upp i följande minnestyper. För mer ingående beskrivning se [Nios II Development Handbook](#).

.entry och .exceptions

.entry och .exceptions är specialfall och förändras inte i storlek vid reducering av koden. De här två minnestyperna innehåller Nios II-processorns reset och exceptions adresser. Nios II verktygen genererar en 32-byte .entry sektion som startar med reset adressen och är exklusivt reserverad för hanteringen av reset.

På liknande sätt genererar verktyget en sektion för .exceptions som startar med exceptions adressen.

.rodata

Innehåller konstanter. Read-Only data.

.rwddata

Innehåller variabler. Read-Write data.

.bss (Block Started by Symbol)

Data som ska initieras till noll. Zero initialized data.

.text

Innehåller all övrig kod. Applikations C-kod.