Low Level C-programming and microcomputer architecture

Task 10 C-code optimization

**Versionshistorik**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Datum** | **Ansvarig** | **Beskrivning** |
| 0.0 | 2013-03-26 | LL | Preliminär version |
| 0.1 | 2013-04-22 | ML | Reviderad version |
| 0.2 | 2014-10-30 | LL | Tips på resultattabell |
| 0.3 | 2016-08-15 | ML | Uppdaterad |

Innehåll

[1 Inledning 1](#_Toc352083579)

[2 Kravspecifikation 1](#_Toc352083580)

# 1 Inledning

Ingenjören får ett uppdrag från en kund att visa på hur koden kan optimeras genom att skriva den på ett mera optimalt sätt. Ge kunden tre exempel, resultatet ska vara storlek (visa assembler), tiden att exekvera och en kort analys. Exemplen visas på Nios economy.

# 2 Kravspecifikation

Följande ska genomföras för att uppgiften ska anses slutförd:

Tabell 1 Kravspecifikation

|  |  |
| --- | --- |
| **Krav** | **Beskrivning** |
| ***Förstudie*** | |
| Krav\_001 | Gå igenom teoribokens kapitel **“C code optimization”**. Skriv eventuella förbättringskommentarer och/eller nya exempel, vilka ska ingå i rapporten enligt krav\_005. |
| ***Funktionskrav*** | |
| Krav\_002 | Beskriv tre exempel kod från teoribokens kapitel **“C code optimization”**. eller skriv egna exempel som inte finns med i boken. Beskriv två fall för varje exempel. Före C-kods optimering och efter. Använd Nios Economy, **ingen kod optimering** och all kod i internt FPGA RAM. |
| Krav\_003 | Mät tid, visa assemblern och kodstorlek för varje kod exempel före och efter. Dokumentera resultaten i en tabell. |
| Krav\_004 (frivilligt) | Kort analys av de tre fallen och förklara varför det blev samma eller olika resultat. I ett separat kapitel ”Analys av resultat”. |
| ***Dokumentationskrav*** | |
| Krav\_005 | Sammanfoga det till en kortfattad standard rapport. |
| ***Leveranskrav*** | |
| Krav\_006 | Leveransen ska ske till plattformen Itslearning. Leveransen ska vara en rapport. Namnet på filen ska vara ”förnamn\_efternamn\_C\_task\_10”. Sista leveransdag se kursschema (för VG). |

# 3 Tips Testresultattabell

Tabell Exempel testresultat

|  |  |  |
| --- | --- | --- |
|  | **Före optimering** | **Efter Optimering** |
| **Kodexempel** | int test(int I) {  int sum = 0;  sum = sum + I + 8 + 10 + 45 + (300 \* I);  return sum; | int test(const int I) {  return I + 63 + (300 \* I);  } |
| **Storlek på kod** | ??? bytes | ??? bytes |
| **Tid för exekvering** | ???clock cykles / ???us | ???cc / ???us |
| **Assembler**  **(NIOS Economy)** | int test(int I) {  **addi sp,sp,-20**  **stw ra,16(sp)**  **stw fp,12(sp)**  **stw r16,8(sp)**  **addi fp,sp,8**  **stw r4,-4(fp)**  int sum = 0;  **stw zero,-8(fp)**  sum = sum + I + 8 + 10 + 45 + (300 \* I);  **ldw r3,-8(fp)**  **ldw r2,-4(fp)**  **add r2,r3,r2**  **addi r16,r2,63**  **ldw r4,-4(fp)**  **movi r5,300**  **call 405c0 <\_\_mulsi3>**  **add r2,r16,r2**  **stw r2,-8(fp)**  return sum;  **ldw r2,-8(fp)**  } | int test(const int I) {  **addi sp,sp,-16**  **stw ra,12(sp)**  **stw fp,8(sp)**  **stw r16,4(sp)**  **addi fp,sp,4**  **stw r4,-4(fp)**  return I + 63 + (300 \* I);  **ldw r2,-4(fp)**  **addi r16,r2,63**  **ldw r4,-4(fp)**  **movi r5,300**  **call 405ac <\_\_mulsi3>**  **add r2,r16,r2**  } |

Analys

Resultatet visade att storleken …. Det beror på att xxx . Prestanda ökade/minskade…. Det beror på att xxx.

För den intresserade, räkna fram antal cc genom teoretiskt räkna fram dem med assemblerinstruktionerna (se Nios-manualen).