|  |
| --- |
| TEIS AB |
| VGA\_IP |
|  |
|  |
| **Andrey Yunin** |
| **2021-05-06** |

|  |
| --- |
| Sammanfattning: Den rapporten innehåller ett resultat av konstruering en återanvändbar VGA IP-komponent till Intels system. En VGA-kontroller i VHDL återvänds från ett tidigare projekt liksom drivrutiner från C-kursen. HW, SW och verktyg integrerades, för att skapa en komplett återanvändbar komponent. Det finns testfallen som visar hur VGA\_IP fungerar med flera testfunktioner som exempel. |

# 

INNEHÅLLSFÖRTECKNING

[1. KRAVSPECIFIKATION 3](#_Toc71712980)

[2. AKTIVITETS - OCH TIDPLAN 6](#_Toc71712981)

[3. VAlidering 7](#_Toc71712982)

# KRAVSPECIFIKATION

Följande ska genomföras för att uppgiften ska anses slutförd:

Tabell 1: Kravspecifikationen

|  |  |  |
| --- | --- | --- |
| **Krav** | **Beskrivning** | **Utfört** |
|  | | ***Ja/nej*** |
| 001 | Kundprojektets arbetsnamn är VGA\_IP | ja |
| 002 | Skapa en IP komponent med VGA-komponenten och återanvänd tidigare VHDL- och C-projekt. Alltså VHDL koden från VHDL kursen och C koden från C kursen (VGA uppgiften). | ja |
| 003 | Krav på konstruktionskoden:  1)”filhuvud” i C-filen.  2) Namn på variabler med mera ska vara tydliga.  3) Rikligt med kommentarer i koden.  4) *Frivilligt:* Kontroll av gränsvärden ska göras, till exempel att värdet inte ligger utanför skärmen. Om detta inträffar ska ett felmeddelande skrivas ut till systemterminalen (Nios II ”console”). | ja |
| ***Funktionskrav*** | |  |
| 004 | **Följande drivrutiner** sparas under HAL och ska vara uppdaterade i VGA\_IP\_SW\_sw.tcl filen). Drivrutinerna kan kopieras från tidigare utfört projekt i C-kursen eller från ett äldre projekt. Frivilligt att lägga på flera drivrutiner.  **Funktion:** print\_pix(unsigned int x,unsigned int y,unsigned int rgb); **Funktionsbeskrivning:** Skriver en pixel med färgen *rgb* på koordinaten (x, y). Alternativ lösning för print\_pix; om funktionen behöver snabbas upp, kan den istället göras om till ett macro.  **Funktion:** print\_hline(unsigned int x\_start, unsigned int y\_start, unsigned int len, unsigned int RGB); **Funktionsbeskrivning:** Skriver en horisontell linje med färgen *rgb* och med längden *len* vilken startar på koordinaten (x\_start, y\_start).  **Funktion:** print\_vline(unsigned int x\_start, unsigned int y\_start, unsigned int len, unsigned int RGB); **Funktionsbeskrivning:** Skriver en vertikal linje med färgen *rgb* och med längden *len* vilken startar på koordinaten (x\_start, y\_start).  **Funktion:** print\_char(unsigned int x,unsigned int y,unsigned int rgb,unsigned int BG\_RGB,char Character); **Funktionsbeskrivning:** Skriver tecknet “Character” med färgen *rgb* och med bakgrundsfärgen ”BG\_RGB” på koordinaten (x, y). Lägg till de ASCI som behövs för att skriva det som krävs. Det går att bestämma hur tecknen ska visas på skärmen, se bilaga, äldre projekt och på webben.  **Funktion:** read\_pixel\_ram\_int(unsigned int x\_start, unsigned int y\_start); return: unsigned pixel\_data (obs! enbart de sista tre bitarna är intressanta). **Funktionsbeskrivning:** Läser pixel\_data från pixel RAM (3 bitar, RGB) från adress x och y (beräknat från x och y).  **Funktion**: clear\_screen (int rgb);  **Funktionsbeskrivning**: Denna funktion rensar skärmen genom att RGB-värdet ”000”(svart) skrivs till alla pixlar på skärmen.  **Funktion:** print\_circle(unsigned int radie, unsigned int x\_centrum, unsigned int y\_centrum, unsigned int rgb); **Funktionsbeskrivning:** Skriver en cirkel med radien “radie” och färgen” rgb” på mittkoordinaten (x, y). Cirkeln ska fyllas med samma färg. | **ja** |
| **Valideringskrav** | |  |
| 005 | Testprotokollet kan vara ofullständigt, men ska innehålla minst tre testfall (HW/SW design report). | ja |
| 006 | Skriv ditt namn på skärmen, fotografera skärmen och lägg in fotot som figur i rapporten (HW/SW design report). | ja |
| 007 | Rapporten ska också innehålla kostnaden för konsultuppdraget. Företaget konstukören är anställd på fakturerar 600 SEK/timme för projektet. Totalt antal timmar \* 600 SEK utgör den totala kostnaden. | ja |
| ***Struktur på leveransen och dokumentationskrav*** | | |
| 008 | Leverera följande struktur.   * VGA\_IP (top library and the sub library)   + VGA\_IP\_HW\_hw.tcl   + VGA\_IP\_SW\_sw.tcl   + hdl (HW),     - ….vhd (VHDL code)   + HAL (SW),     - Inc (library for h files)     - src (library for c files)   + doc (documentations),     - VGA\_datasheet     - Software manual for the programmers     - SW/HW Design Report (standard report)   + Examples     - Ett exempelsystem som skriver ut ditt namn på skärmen     - SOF och sopcinfo filen     - Arkiverade projektet     - C filen (main)   + Projects (HW)     - VGA\_IP (Quartus) | ja |
| ***Leveranskrav*** | |  |
| 009 | Leveransen ska ske till plattformen Itslearning. Leveransen ska vara en IP-komponent enligt strukturen från krav 004. Namnet på filen ska vara ”fornamn\_efternamn\_HWSW\_task\_3.zip”. Sista leveransdag se Itslearning (för VG). | ja |

# AKTIVITETS - OCH TIDPLAN

Projektet var tänkt att börja den 03 maj och hålla på till 07 maj. Projektet ska ta cirka 5 dagar att genomföra, 40 timmar.

Tabell 2 innehåller aktiviteter och tidplan som planeras utföra under ingenjörsjobb:

Tabell 2: Tidplan

|  |  |
| --- | --- |
| **Aktiviteten** | **Tid** |
| Skapa ett HW designprojekt (16 timmar) | 03.05.2021 – 04.05.2021 |
| Skapa ett SW designprojekt (8 timmar) | 5.05.2021 |
| Analys, validering, och rapport (16 timmar) | 06.05.2021 – 07.05.2021 |

Plan kan redigeras i processen av jobbutförande.

Projektet har tagit 40 timmar att genomföra, med en kostnad på 600 SEK/tim och ger en total kostnad på 24000 kr.

# VAlidering

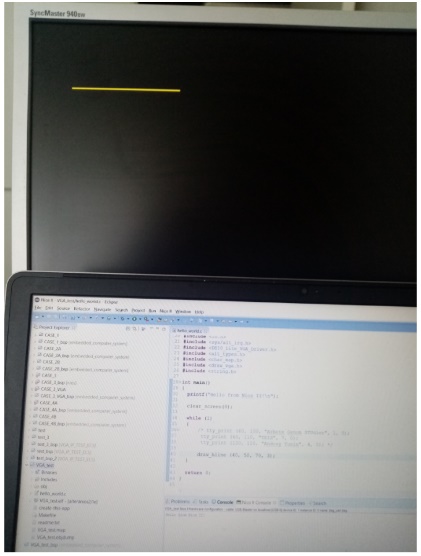
Det här testprotokoll som presenteras nedan visar testfallen som används i validering, se tabell

Protokollet används för validering, se tabell 3

Tabell 3: Test validering protokoll

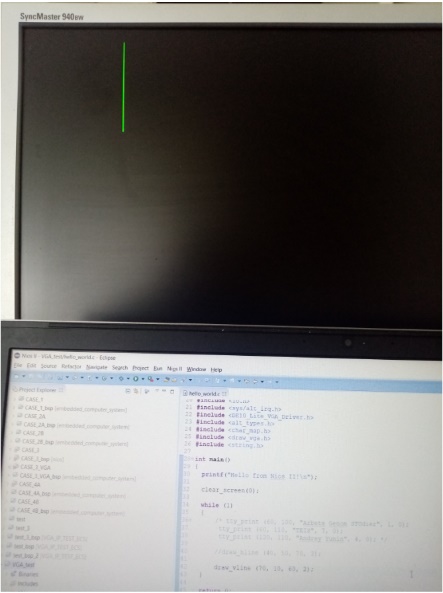
|  |  |  |  |
| --- | --- | --- | --- |
| **Testfall** | **Initiala testvillkor (test funktioner)** | **Testfall Beskrivning** | **Validering** |
| 1 | **draw\_hline** (40, 50, 70, 3) | Skärmen visar en 70 pixlar lång horisontell gul linje som startar på koordinaten (40,50) | ok |
| 2 | **draw\_vline** (70, 10, 60, 2) | Skärmen visar en 60 pixlar lång vertikal grön linje som startar på koordinaten (70,10) | ok |
| 3 | **tty\_print** (60, 100, "Arbete Genom STUdier", 1, 0);  **tty\_print** (60, 110, "TEIS", 7, 0); **tty\_print** (120, 110, "Andrey Yunin", 4, 0); | Skärmen visar två linjen text | ok |

Figur 7 visar resultat av testfall 1



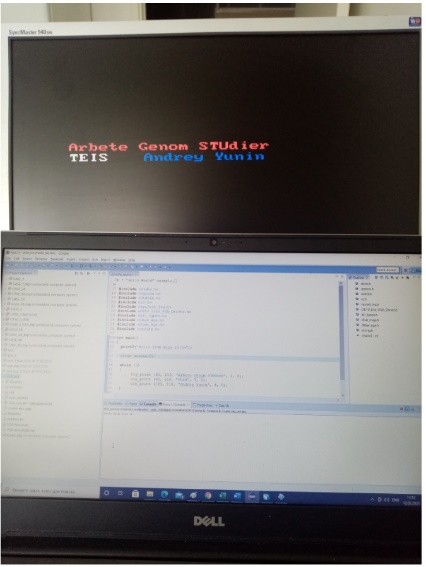
Figur 1 : resultat av testfall 1

Figur 7 visar resultat av testfall 2



Figur 2: resultat av testfall 2

Figur 7 visar resultat av testfall 3



Figur 3: resultat av testfall 3