

MARKT ONDERZOEK FPGA EN CAMERA

Hogeschool van Arnhem en Nijmegen, 22 november 2013

Projectgroup: Ramon Kleiss <ramonkleiss@gmail.com>
Niek van der Straaten <niekvds@gmail.com>

Begeleider: Hugo Arends <hugo.arends@han.nl>

Inhoudsopgave

| | | |
|----------|-------------------------------------|----------|
| 1 | Doel marktonderzoek | 3 |
| 2 | Eisen en wensen | 4 |
| 2.1 | Resolutie camera | 4 |
| 2.2 | RAM calculatie | 4 |
| 2.3 | Camera interface | 4 |
| 2.4 | Microcontroller interface | 5 |
| 2.5 | Ontwikkelingsbord | 5 |
| 3 | Onderzoek | 6 |
| 3.1 | Vision ontwikkelingskits | 6 |
| 3.2 | Camera modules | 7 |
| 3.3 | FPGA ontwikkelingsborden | 8 |
| 4 | Onderzoeksresultaten | 9 |

Hoofdstuk 1

Doel marktonderzoek

Het systeem wordt gebouwd op basis van een camera waarvan een FPGA de beelden af haalt. De FPGA voert er vervolgens een serie beeld verbeteringen op los om deze vervolgens in een dual-port RAM te zetten die vanuit de buitenkant te benaderen is. Om het doel realistisch te houden is er een goede basis nodig. In dit marktonderzoek wordt er gekeken naar diverse mogelijkheden en hun voor/nadelen.

Hoofdstuk 2

Eisen en wensen

Om een goed werkend systeem te kunnen maken zijn er een aantal minimale eisen waaraan een geselecteerd systeem aan moet voldoen. Iedere verhoging heeft zo zijn consequenties verder in het traject. De eisen en wensen staan op volgorde, wordt voor het bovenste item een krachtigere oplossing gekozen dan moeten de onderstaande items dat ook ondersteunen.

- Resolutie camera, zie sectie 2.1;
- RAM geheugen FPGA, zie sectie 2.2;
- Camera interface, zie sectie 2.3;
- Microcontroller interface, zie sectie 2.4;
- Ontwikkelingsbord, zie sectie 2.5;

2.1 Resolutie camera

De resolutie van de camera bepaald hoe goed een object gedetecteerd kan worden. Ook bepaald deze hoe snel het systeem 1 frame kan verwerken. Om niet te snelle hardware nodig te hebben om een acceptabele frame-rate te halen moet de resolutie niet te hoog liggen. HD of hoger is bijvoorbeeld helemaal niet nodig. Als minimale resolutie wordt een $640 * 480$ geaccepteerd.

2.2 RAM calculatie

Met de onderstaande calculatie is de minimaal benodigde ruimte te berekenen.

$$C = A * D * R * X * Y \quad (2.1)$$

Waarbij Y staat voor het aantal pixels dat 1 afbeelding hoog is; X het aantal pixels dat 1 afbeelding breed is; R het aantal bytes per kleur van 1 pixel; D het aantal kleuren in 1 pixel; A het aantal plaatjes data het RAM moet kunnen opslaan; en C staat voor de hoeveelheid ruimte in bytes.

Om een goede inschatting te kunnen maken is het aan te raden om uit te gaan van 4 plaatjes in 8-bit RGB formaat. Op deze manier moet er altijd voldoende ruimte zijn.

2.3 Camera interface

Een camera interface heeft als enige eis dat hij zo eenvoudig mogelijk dient te zijn. Dit i.v.m. de implementatie tijd. De focus van het project ligt op de vision bewerking, niet op de acquisitie van beelden. Denk bij interfaces aan I²C, SPI en/of parallele interfaces. Houdt hierbij rekening met het benodigde aantal I/O's. Deze moeten wel voldoende aanwezig zijn op het ontwikkelingsbord.

2.4 Microcontroller interface

Als microcontroller interface is het verstandig om gebruik te maken van een zogenaamde FSMC-bus. Althans, zo heet dit bij ST microcontrollers. Texas instruments heeft ook een dergelijk concept, de EMIF-bus. Bij NXP is dat dan weer EMC-bus.

De FSMC-bus maakt gebruik van een 16-bit data bus en daarbij ondersteunt hij 26-bits adressen. Dat is een totaal van maximaal $2 * 2^{26} = 134217728bytes$.

Daarnaast maakt de bus gebruik van enkele bedieningslijnen, doorgaans 4 stuks.

Bij een $640 \times 480 pixel$ camera is het nodig om rekening te houden met $\log_2 \left(\frac{640 \times 480 \times 3}{2} \right) \equiv 18,813 \approx 19 - bits$. Dat betekent dat we voor de FSMC-bus minimaal $20 + 16 + 4 \equiv 40lijnen$ nodig zijn.

2.5 Ontwikkelingsbord

Het juiste FPGA ontwikkelingsbord wordt gekozen op basis van de onderstaande criteria.

- Bij voorkeur een ontwikkelingskit voor camera applicaties;
- FPGA bord moet het aantal I/O's kunnen aansturen;
- er dient voldoende RAM aanwezig te zijn, zoals gezegd in sectie 2.2;
- de FPGA dient genoeg LUTs en flip-flops te hebben;
- FPGA moet voldoende snelheid hebben om een frame te verwerken;

De laatste 2 punten zijn wat vaag. Daarvoor geldt, des te hoger, des te beter!

Hoofdstuk 3

Onderzoek

Hieronder is een lijst van gevonden producten te vinden. Ze zijn geordend op basis van type.

- Vision ontwikkelingskit;
- camera modules;
- FPGA ontwikkelingsborden.

3.1 Vision ontwikkelingskits

| Avnet Spartan-6 FPGA Industrial Video Processing Kit | | |
|--|-------------------------|--------|
| Type | Vision ontwikkelingskit | |
| Type camera | Omnivision OV9715 | sensor |
| Maximale resolutie camera | 1280 x 800 | pixels |
| Configuratie interface camera | SCCB | |
| Communicatie interface camera | DVP | |
| Type FPGA | Xilinx XC6SLX150T | |
| FPGA LUTs | 92.152 | |
| FPGA RAM | 4,824 | Mbyte |
| FPGA snelheid | 100 | MHz |
| Aantal I/O's | 136 | |
| Kosten | 2.695 | \$ |

Dit is een complete kit voor het ontwikkelen van een industrieel vision systeem. Waarschijnlijk wat overdreven voor de applicatie maar voldoet zeker!

3.2 Camera modules

| Raspberry Pi camera module | | |
|-------------------------------|-------------------|--------|
| Type | Camera module | |
| Type camera | Omnivision OV5647 | sensor |
| Maximale resolutie camera | 2592 x 1944 | pixels |
| Configuratie interface camera | SCCB | |
| Communicatie interface camera | DVP | |
| Kosten | 42,99 | € |

Een camera met hoge resolutie. Deze kan ook terug gehaald worden naar VGA (640x480) resolutie. Daarnaast is hij compatible met Raspberry Pi, dus mocht de acquisitie niet lukken met de FPGA dan kunnen we dat nog aan de Raspberry Pi geven. Een groot nadeel is dat er niet snel veel informatie over de bussen te vinden is. SCCB is een standaard (I²C-bus gebaseerde) communicatie bus die door OmniVision is ontwikkeld. Zodoende werken vrijwel alle camera's er mee. We kunnen daarom eenvoudig een andere camera pakken met bijvoorbeeld een lagere resolutie om de camera beter te laten presteren in situaties met weinig licht.

3.3 FPGA ontwikkelingsborden

| Altium Nanoboard 3000 | | |
|-----------------------|--|-------|
| Type | FPGA ontwikkelingsbord | |
| Type FPGA | Xilinx XC3S1400AN | |
| FPGA LUTs | 11.264 | |
| FPGA RAM | 576 | Kbyte |
| FPGA snelheid | 6-200 | MHz |
| Aantal I/O's | 36 (headers) + 50 (uitbreidings connector) | |
| Kosten | Aanwezig op school | |

Het bord is prima geschikt, vooral wegens zijn grote externe geheugen. Het grote nadeel is de Altium omgeving, al is het bord direct met een binary file die de Xilinx ISE genereert te programmeren. Matig tot slecht gedocumenteerd.

| Digilent Nexis 4 | | |
|------------------|------------------------|-------|
| Type | FPGA ontwikkelingsbord | |
| Type FPGA | Xilinx XC7A100T | |
| FPGA LUTs | 63.400 | |
| FPGA RAM | 4,860 | Mbyte |
| FPGA snelheid | 100 | MHz |
| Aantal I/O's | 32 (headers) | |
| Kosten | 159,00 | \$ |

Bord is matig geschikt. Beschikt over een goede kloksnelheid, veel intern RAM. Goed gedocumenteerd! De grote nadelen zijn dat het bord niet direct tot de beschikking is en de lange hoeveelheid I/O's.

| Digilent Nexis 3 | | |
|------------------|--|-------|
| Type | FPGA ontwikkelingsbord | |
| Type FPGA | Xilinx XC6LX16 | |
| FPGA LUTs | 9.112 | |
| FPGA RAM | 576 | Kbyte |
| FPGA snelheid | 100 | MHz |
| Aantal I/O's | 32 (headers) + 40 (uitbreidings connector) | |
| Kosten | 119,00 | \$ |

Bord is prima geschikt. Helaas heeft deze weinig intern RAM, dat is op te lossen middels het 16Mbyte externe RAM. Het bord is goed gedocumenteerd en heeft voldoende I/O's. Helaas is het bord niet direct beschikbaar.

| Digilent Atlys | | |
|----------------|---|-------|
| Type | FPGA ontwikkelingsbord | |
| Type FPGA | Xilinx XC6SLX45 | |
| FPGA LUTs | 27.288 | |
| FPGA RAM | 2,100 | Mbyte |
| FPGA snelheid | 100 | MHz |
| Aantal I/O's | 8 (headers) + 40 (uitbreidings connector) | |
| Kosten | 219,00 | \$ |

Dit bord is prima geschikt wegens zijn grote interne geheugen en redelijke hoeveelheid I/O's. Helaas geldt ook voor dit bord, het is niet direct beschikbaar.

Hoofdstuk 4

Onderzoeksresultaten

Uit het onderzoek is gebleken dat vooral het aanbod van camera modules erg klein is. De enige bruikbare module is de Raspberry Pi camera module die gebruik maakt van een gepatenteerd protocol (Camera-Serial-Interface, of CSI) ontwikkeld door MIPI. Om toegang te krijgen tot de specificaties van dit protocol is het nodig lidmaatschap tot het MIPI aan te vragen wat 4000 dollar per jaar kost¹. Dit is voor een eenmalig project niet te betalen dus de optie om alleen de Raspberry Pi camera module te gebruiken in combinatie met een FPGA valt af.

Wat wel een optie is, is het gebruiken van een Raspberry Pi als acquisitie in combinatie met een bestaand of zelf ontwikkeld protocol om te communiceren met de FPGA die in dit model zal dienen als filter. Een andere optie is om geheel af te stappen van een digitale camera module en gebruik te maken van een camera module die een analoog signaal (bijvoorbeeld PAL of NTSC) uitstuurt. Door bijvoorbeeld een MAX9526² te gebruiken kan dit signaal gedigitaliseerd worden naar een signaal wat de FPGA vervolgens kan uitlezen. Een bijkomend voordeel is dat een PAL camera goed verkrijgbaar is en er dus veel keus is tussen kwaliteit (lichtgevoeligheid, etc.).

Verder is het mogelijk om een USB stack te implementeren op een FPGA, echter is dit geen wenselijke keuze omdat er dan veel werk gaat zitten in het correct implementeren van deze stack in plaats van beeldverwerking (waar dit project om zou moeten draaien). Daarnaast is het nog onbekend of de camera ook geïnitieerd dient te worden. Als dat het geval is, dan is het nagenoeg onmogelijk om dit te implementeren in de FPGA.

Nog een andere optie is het geheel weglaten van de FPGA en een Raspberry Pi gebruiken met de bijbehorende camera module. Een van de dingen waar dan wel rekening mee moet worden gehouden is het feit dat de Raspberry Pi niet snel is wat problemen kan veroorzaken.

Verder zou het ook een optie kunnen zijn om het CSI protocol te *reverse engineeren* om er gebruik van te maken, ook dit is niet wenselijk, omdat hier veel tijd in zou gaan zitten.

¹<http://mipi.org/join-mipi/membership-model>

²<http://datasheets.maxim-ic.com/en/ds/MAX9526.pdf>