

2013•2014
FACULTEIT INDUSTRIËLE INGENIEURSWETENSCHAPPEN
master in de industriële wetenschappen: energie

Masterproef

Ontwerp van een realtime triggersysteem toegepast op de fotografie van een ballistisch projectiel

Promotor :
dr. ir. Eric DEMEESTER

Promotor :
Prof. dr. ir. PIRLOT

Jurgen Wijnants

Proefschrift ingediend tot het behalen van de graad van master in de industriële wetenschappen: energie

Gezamenlijke opleiding Universiteit Hasselt en KU Leuven

2013•2014
Faculteit Industriële
ingenieurswetenschappen
master in de industriële wetenschappen: energie

Masterproef

Ontwerp van een realtime triggersysteem toegepast op de
fotografie van een ballistisch projectiel

Promotor :
dr. ir. Eric DEMEESTER

Promotor :
Prof. dr. ir. PIRLOT

Jurgen Wijnants

*Proefschrift ingediend tot het behalen van de graad van master in de industriële
wetenschappen: energie*

Woord vooraf

Deze masterproef vormt het sluitstuk van mijn opleiding industrieel ingenieur aan de U Hasselt / KU Leuven en is slechts tot stand kunnen komen dankzij de vele kennis en inzichten die ik heb kunnen opdoen gedurende de voorbije vier jaar.

Eerst en vooral zou ik mijn externe promotor, prof. Dr. Ir. Marc Pirlot, willen bedanken om mij de kans te geven om deze masterproef uit te voeren, en voor zijn ondersteuning en feedback gedurende de ganse periode. Ook wens ik mijn interne promotor, Dr. Ir. Eric Demeester, te bedanken voor zijn ondersteuning en bereidwilligheid om mij doorheen deze masterproef te loodsen.

Verder wil ik graag in het bijzonder adjudant Vanhove bedanken, omdat hij altijd beschikbaar was om mijn vragen te beantwoorden en om mij te helpen bij het uitvoeren van alle testen, zelfs wanneer het erg druk was. Ook het andere personeel van het labo heeft me steeds ondersteund bij het uitvoeren van dit eindwerk en ook hen wens ik dus van harte te bedanken.

Ten slotte zou ik nog een woord van dank willen richten aan mijn familie, vrienden en vriendin, die steeds voor mij klaar stonden gedurende de laatste vier jaar. Zonder hun motiverende (telefoon)gesprekken, noodzakelijke afleidingen en minutieus verbeterwerk zou ik vandaag een heel stuk minder ver gestaan hebben.

Ontwerp van een realtime triggersysteem toegepast op de fotografie van een ballistisch projectiel

Inhoudsopgave

Woord vooraf	1
Lijst van tabellen	5
Lijst van figuren	7
Verklarende woordenlijst	9
Abstract	11
Summary	13
1 Inleiding	15
1.1 Situering	15
1.2 Probleemstelling	15
1.3 Doelstellingen en eisen	16
1.4 Gevolgde methode	16
1.5 Algemene structuur	17
2 Testopstelling	19
2.1 Het labo wapensystemen & ballistiek	19
2.2 Gebruikte Hardware	20
2.2.1 Snelheidsmetingen	20
2.2.2 Chronometers	26
2.2.3 De PXI met geïntegreerde FPGA module	27
2.2.4 Camera's	30
2.2.5 Vergelijking met andere hedendaagse systemen en methodes	34
3 Software	37
3.1 LabVIEW	37
3.2 Hadland SVR2 Software	39
3.3 Photron Fastcam Viewer	41
4 Foutenanalyse uitgevoerd op de proefopstelling	45
4.1 Proefopstelling	45
4.2 Foutenanalyse	46
4.2.1 De afstand tussen de lichtschermen (k-waarde)	46
4.2.1 De afstand tussen het laatste lichtscherm en de camera (x-waarde)	47
4.2.3 Gemeten tijdsinterval Δt	48
4.2.4 Resultaten verificatie FPGA klok	49

Ontwerp van een realtime triggersysteem toegepast op de fotografie van een ballistisch projectiel

4.2.5 Hoek tussen de baan van het projectiel en het lichtscherm α	51
4.2.6 Hoek tussen baan van het projectiel en optische as van de camera β	52
4.2.7 Versnelling.....	53
4.3 Conclusie.....	54
5 Opzetten van de meetapparatuur	55
5.1 De meetbasis	55
5.2 De camera	56
6 Werking van het programma.....	59
6.1 FPGA Code.....	60
6.2 PC Code	62
6.3 Gedetailleerde bespreking code	63
6.3.1 Vertragingstijd van een projectiel met constante snelheid	63
6.3.2 Vertragingstijd van een projectiel onderhevig aan een versnelling.....	64
6.3.3 Verschillende soorten delingen bij een FPGA	65
6.3.4 Verhouding teller/noemer bij while loop subtraction.....	66
6.3.5 Berekening minimumafstand.....	67
7 Variatie van snelheid bij projectielen.....	69
8 Test van vuren schot per schot	71
8.1 Resultaten vergelijkende test	71
8.2 Bespreking resultaten.....	73
8.3 Problemen met de Hadland SVR2	74
9 Test met automatisch vuren.....	77
9.1 Inleiding	77
9.2 Resultaten en bespreking van de eerste test.....	77
9.2 Probleemoplossing.....	79
9.3 Resultaten en bespreking van de tweede test.....	81
10 Besluit en toekomstig werk	85
Literatuurlijst.....	87
Aanbevolen literatuur	89
Lijst met bijlagen.....	90

Lijst van tabellen

Tabel 1: ijking FPGA klok.....	49
Tabel 2: resultaten test.....	72

Ontwerp van een realtime triggersysteem toegepast op de fotografie van een ballistisch projectiel

Lijst van figuren

Figuur 1: schets van het grondplan van het labo ballistiek	19
Figuur 2: B481 HPI Doppler radar [3]	20
Figuur 3: waterfall diagram.....	21
Figuur 4: snelheidsdiagram gemeten met Doppler radar	22
Figuur 5: LS 19 i3 meetbasis van Drello Ballistics.....	24
Figuur 6: aansluitdiagram chronometers voor “Ls 19 i3”	26
Figuur 7: National Instruments PXI met een FPGA kaart in slot 3	28
Figuur 8: een SCB-68 break-out box	29
Figuur 9: bewegingsonscherpte bij een kermisattractie [8]	30
Figuur 10: Hadland SVR2 digitale camera	31
Figuur 11: Photron Fastcam SA5 digitale hogesnelheidsvideocamera [9]	33
Figuur 12: interface met compilatie instellingen.....	38
Figuur 13: de software om de Hadland SVR2 aan te sturen	39
Figuur 14: SVR2 software met “ <i>Camera Setup</i> ” weergegeven	40
Figuur 15: Photron FASTCAM Viewer software	41
Figuur 16: PFV met rechts het scherm om alle parameters in te stellen.....	42
Figuur 17: Photron FASTCAM Viewer software met kruisdraad en kalibratiescherm zichtbaar.....	43
Figuur 18: meetopstelling in het labo.....	45
Figuur 19: correcte opstelling van de meetbasis	55
Figuur 20: schets van opstelling met toestel (geel) dat twee onderling loodrechte lasers bevat, om de hoek tussen optische as van de camera en de kogelbaan correct in te stellen	56
Figuur 21: apparaat met twee loodrechte laserstralen	57
Figuur 22: doelenzone met twee samenvallende lasers	57
Figuur 23: de Photron camera met de opgelijnde laserlijn zichtbaar	58
Figuur 24: programmastructuur LabVIEW project met één VI op pc niveau, en één VI op FPGA niveau.	59
Figuur 25: Visuele voorstelling van de FPGA code	60
Figuur 26: overzicht pc code	62
Figuur 27: spreiding van de snelheid van 7,62 x 51 mm NATO	70
Figuur 28: Schot 1 @ 854 m/s.....	71
Figuur 29: Schot 5 aan 452 m/s.....	71
Figuur 30: signalen Hadland camera.....	75
Figuur 31: 1 ^e salvo MAG	78
Figuur 32: verstoorde signalen opgemeten met PXI ten gevolge van interferenties veroorzaakt door de mondingsvlam van een automatisch wapen	79
Figuur 33: correcte signalen opgemeten met PXI	81
Figuur 34: tweede salvo MAG	82

Ontwerp van een realtime triggersysteem toegepast op de fotografie van een ballistisch projectiel

Verklarende woordenlijst

Array: Engelstalige benaming voor een rij (of reeks). Dit is een reeks data die geïndexeerd kan worden met behulp van een integer.

Break-out box: een doosje met klemmenstroken om parallelle kabels eenvoudig te verbinden.

FPGA: Field Programmable Gate Array, een geïntegreerde schakeling bestaande uit logische componenten die geschakeld kunnen worden als logische poorten (AND,XOR,NAND,...).

IO: Input / Output oftewel ingang / uitgang. Benaming om het geheel van aansluitingen die functioneren als digitale of analoge in- of uitgang aan te duiden.

Motion blur: Leenwoord voor bewegingsonscherpte. Dit is in het verschijnsel dat objecten op een foto of op een video- of filmbeeld onscherp zijn als gevolg van beweging van het object en/of de camera. Dit wordt veroorzaakt doordat tijdens de beweging het object ook in de projectie in de camera beweegt en dus het fotografische vlak op meerdere plaatsen door hetzelfde object belicht wordt [1].

NI: National Instruments, Amerikaans bedrijf dat eigenaar is van de LabVIEW software en de bijbehorende hardware produceert.

PCI: Peripheral Component Interconnect. Dit is een standaard bus die gebruikt wordt om randapparatuur op het moederbord van een pc te bevestigen.

PXI: PCI eXtensions for Instrumentation : een standaard van National Instruments om meettoestellen via PCI aan te sluiten op een pc. Komt meestal voor onder de vorm van een kast met daarin plaats voor één pc en meerdere meettoestellen die dan automatisch verbonden worden met de pc via een rail in de achterzijde van de kast. De rail fungeert dus als een kabel tussen de PCI bus van de pc en het meettoestel.

Ricochet: projectiel dat terugkaatst bij impact in plaats van tot stilstand te komen in de doelenzone.

RTTS: Realtime trigger systeem, een systeem dat in realtime (dus terwijl de gebeurtenis zich nog afspeelt) metingen uitvoert en deze verwerkt om zo op het precieze moment een triggersignaal uit te sturen. Dit triggersignaal kan gebruikt worden voor toepassingen waar de timing zeer nauwkeurig moet zijn.

Sampling frequentie: Leenwoord voor bemonsteringsfrequentie. Dit is de frequentie waarmee een apparaat "*samples*" (= monsters) neemt. Een apparaat met een sampling frequentie van 1Mhz zal bijvoorbeeld 1 miljoen keer per seconde een meting nemen. Als een fenomeen zo snel verloopt dat het minder dan een miljoenste van een seconde waarneembaar is, zal het apparaat dit niet kunnen detecteren.

Ontwerp van een realtime triggersysteem toegepast op de fotografie van een ballistisch projectiel

Trigger: De trigger is een signaal (meestal een spanningspuls) dat aangeeft aan een camera wanneer deze laatste zijn opname moet starten. Eventueel kan een trigger ook gebruikt worden om een ander apparaat te starten.

Sluitertijd: de sluitertijd is de tijdsduur die beschrijft hoelang een lichtgevoelige plaat of ander lichtgevoelig element wordt blootgesteld aan een bepaalde hoeveelheid licht [2].

VI: Virtual Instrument, de naam voor een programma dat in LabVIEW geschreven is. Een VI bestaat steeds uit twee delen: een "*front panel*" dat knoppen, grafieken, status van bits, ... bevat en een blokdiagram waar de verwerking van data gebeurt aan de hand van blokken die verbonden worden met virtuele draden.

TTL: Transistor-Transistor Logica, dit is een standaard uit de digitale elektronica waarin bepaalde parameters zijn opgenomen zoals de voedingsspanning, de spanning die een 1 betekent (binair) en de spanning die een 0 betekent (binair).

Abstract

Het laboratorium van het departement “Wapensystemen & ballistiek” van de Koninklijke Militaire School te Brussel gebruikt digitale hogesnelheidscamera's om projectielen in vlucht te fotograferen. Het doel van deze masterproef is om een programma te schrijven dat in realtime de snelheid van een projectiel meet en zo op het juiste moment de camera activeert.

Het probleem is dat de snelheid van projectielen sterk varieert, zelfs in schijnbaar identieke omstandigheden, ondermeer omdat de verbranding van het kruit telkens anders verloopt. Hierdoor is het moeilijk om een projectiel te fotograferen omdat het exacte moment waarop het projectiel zich voor de camera bevindt niet gekend is. Een realtime FPGA (Field Programmable Gate Array) sturing lost dit probleem op door het exacte fototijdstip te berekenen in functie van de gemeten snelheid, om zo steeds een foto te nemen met het projectiel centraal in beeld.

De snelheidsmeting wordt uitgevoerd door twee infrarood lichtscheren. Verder wordt er gebruik gemaakt van een PXI van National Instruments die zowel een FPGA als een pc bevat. Beide worden geprogrammeerd onder LabVIEW: de FPGA voert in realtime de berekeningen uit terwijl de pc achteraf alle gegevens verwerkt en opslaat. De resultaten waren consequent positief, met een fout van maximaal 1 cm, zelfs wanneer de snelheid van de projectielen meer dan 400 m/s verschilde.

Ontwerp van een realtime triggersysteem toegepast op de fotografie van een ballistisch projectiel

Summary

The laboratory of the Department of weapon systems & ballistics of the Royal Military Academy in Brussels uses digital high-speed cameras to shoot images of projectiles in flight. The purpose of this master's thesis is to create software that can measure the velocity of the projectile in real-time, and using this measurement, trigger the camera at the right moment.

The challenge is that the velocity of projectiles varies greatly, even in seemingly identical circumstances, because the combustion of gunpowder is different every time. This makes it difficult to calculate the moment at which the projectile will be directly in front of the camera. A real-time FPGA (Field Programmable Gate Array) controller solves this problem by calculating the correct time using the measured speed of the projectile.

Two infrared light screens are used to measure the velocity. Furthermore, a PXI from National Instruments with a built-in FPGA and pc is used. Both devices are programmed using LabVIEW: the FPGA does all the calculations and communications in real-time while the pc is used to store all data. The results of all tests were very good: the RTTS achieved an accuracy of 1 cm, even when the speed difference between the projectiles was more than 400 m/s.

Ontwerp van een realtime triggersysteem toegepast op de fotografie van een ballistisch projectiel

1 Inleiding

1.1 Situering

Als laatstejaarsstudent industrieel ingenieur aan de UHasselt en KU Leuven voer ik mijn masterproef uit bij het labo ballistiek in de Koninklijke Militaire School (KMS) te Brussel. Dit is een federale universiteit waar ook heel wat onderzoek verricht wordt, onder meer op het vlak van ballistiek en wapensystemen.

Dit labo beschikt over een tunnel waarin projectielen afgevuurd worden om zo hun gedrag te bestuderen. Het is dus belangrijk om bepaalde parameters van het projectiel te meten: de snelheid, de vervorming, hoe het projectiel zich gedraagt bij impact, de druk in de loop, ... Vaak wordt er ook een foto genomen van het projectiel in vlucht, omdat uit deze foto ook heel wat informatie gehaald kan worden zoals de hoek tussen zijn baan en zijn lengteas.

1.2 Probleemstelling

Een eerste eenvoudige oplossing zou zijn om de snelheid van het projectiel op voorhand in te voeren in een computer zodat deze ruim de tijd heeft om het juiste triggermoment te berekenen. Met behulp van een eenvoudige akoestische sensor die opgesteld wordt langs de loop en het moment van afvuren detecteert zouden we dan eenvoudig en goedkoop op het juiste moment een foto van het projectiel kunnen maken. De snelheid van een bepaald projectiel is gemakkelijk op te zoeken (in functie van het kaliber, de lengte van de loop, etc...) in bepaalde databases. Deze oplossing werkt echter niet in de praktijk, omdat er een grote variatie zit in de snelheid van een bepaald type projectielen. Dit heeft alles te maken met de ontbranding van het kruit die bijna onmogelijk is om te voorspellen maar wel een grote invloed heeft op de snelheid van het projectiel. In hoofdstuk 7 geeft Tabel 2 dat dit snelheidsverschil kan oplopen tot 40 m/s voor identieke projectielen. Als een gevolg hiervan zou met bovenstaande methode het projectiel niet centraal op de foto staan, of zelfs gewoonweg niet op de foto staan.

Momenteel gebruikt men als oplossing hiervoor een videocamera in plaats van een fotocamera die meerdere beelden na elkaar kan opnemen. Hierdoor is er bijna altijd wel één goede foto die dan gebruikt kan worden. Het nadeel hiervan is dat de videocamera minder scherpe beelden maakt dan de fotocamera, en dat de videocamera een immense hoeveelheid licht nodig heeft gedurende de sequentie van 20 frames. De fotocamera heeft evenveel licht nodig maar dit kan met behulp van een flash in plaats van een continue lichtbron. Vandaar dus dat het labo graag software zou hebben die toelaat om de fotocamera op een nauwkeurige wijze te gebruiken in plaats van de videocamera die momenteel gebruikt wordt.

1.3 Doelstellingen en eisen

Het doel van deze masterproef is het ontwikkelen van een softwaresysteem dat in staat is om zeer nauwkeurig een foto te nemen van een ballistisch projectiel in vlucht. In de schietunnel wordt een projectiel afgevuurd met snelheden die variëren van 50 tot 1000 m/s. De snelheid van deze projectielen wordt gemeten(meestal aan de hand van 4 lichtschermen) en de FPGA (Field Programmable Gate Array) dient dan heel snel de juiste tijdsvertraging te berekenen opdat de fotocamera op het juiste moment getriggerd wordt en zo het projectiel (liefst) centraal op het beeld fotografeert. Vandaar dus ook de titel (Ontwerp van een realtime triggersysteem toegepast op de fotografie van een ballistisch projectiel) van dit eindwerk: het systeem moet in realtime een trigger berekenen en deze trigger wordt gebruikt om ballistische projectielen te fotograferen. De naam voor dit systeem is realtime trigger systeem (RTTS).

De eisen die gesteld worden zijn tamelijk eenvoudig: ten eerste moet het projectiel steeds centraal op de foto staan (de maximale toegestane afwijking bedraagt 2 cm) en ten tweede moet het systeem van de eerste keer erin slagen om een correcte foto te nemen. Om dit te verwezenlijken zal er behalve software, ook een handleiding voor het personeel gemaakt moeten worden met speciale aandacht voor een goede kalibratie van de opstelling. Hiervoor is een analyse van de gevoeligheid van de opstelling aan foutieve instellingen nodig.

We mogen ervan uit gaan dat de maximale snelheid die een projectiel kan bereiken 1000m/s is.

1.4 Gevolgde methode

De oplossing die het onderwerp uitmaakt van deze masterproef is de volgende: Het projectiel verlaat de loop met een onbekende willekeurige snelheid. We meten dan de snelheid van het projectiel met behulp van de infrarood lichtschermen. Vervolgens berekent de FPGA zeer snel de snelheid van het projectiel, en daaruit volgend het tijdstip waarop de camera getriggerd moet worden. De afstanden in onze meetopstelling tussen het punt waar de snelheidsmeting gebeurt en de positie van de fotocamera moeten dan natuurlijk gekend zijn. Op voorwaarde dat de FPGA module snel genoeg rekent zou deze oplossing nauwkeurig genoeg moeten zijn om keer op keer een correcte foto van het projectiel te nemen, zelfs als de snelheden van de projectielen grote variaties vertonen.

1.5 Algemene structuur

De algemene structuur van dit werk is als volgt opgebouwd: in hoofdstuk 2 wordt er een korte beschrijving van het labo gegeven waarna in hoofdstuk 2 en 3 de gebruikte hardware en software besproken wordt.

Vervolgens wordt in hoofdstuk 4 de opstelling besproken en wordt er geanalyseerd waar er overall fouten kunnen optreden, en wat de invloed van deze fouten is. Dit leidt tot hoofdstuk 5 waar een methode besproken wordt om de meetapparatuur zo op te stellen dat deze fouten geminimaliseerd worden.

In hoofdstuk 6 wordt de code besproken, vooral met het oog op mogelijke uitbreidingen of aanpassingen zodat het systeem ook voor andere, vergelijkbare toepassingen gebruikt kan worden.

Vanaf hoofdstuk 7 is ieder hoofdstuk afzonderlijk gewijd aan de resultaten van een bepaalde test: hoofdstuk 7 gaat over de spreiding van de snelheid van projectielen, hoofdstuk 8 over een vergelijkende test waar de werking van het systeem getest wordt voor allerlei verschillende kalibers en snelheden en in hoofdstuk 9 ten slotte worden de resultaten van een kleine uitbreiding besproken waar alle projectielen, afgevuurd door een machinegeweer, gefotografeerd werden.

2 Testopstelling

2.1 Het labo wapensystemen & ballistiek

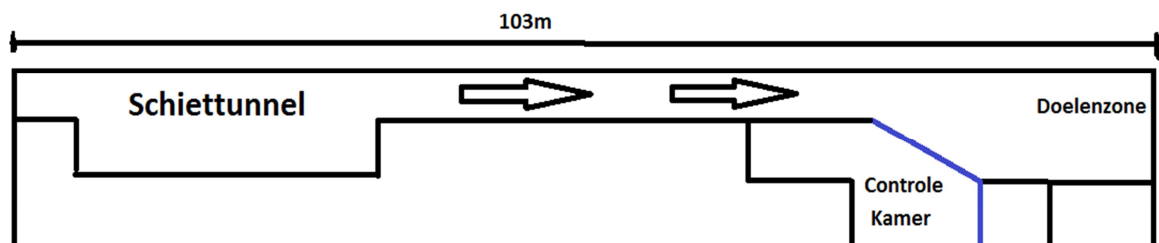
In het labo is een schietunnel aanwezig met een lengte van 103 meter: dit laat toe om te vuren tot op afstanden van maximaal 100 meter. De overige drie meter zijn voorbehouden voor de doelenruimte en de opstelling van het vuurwapen. Meestal wordt er echter gevuld vanop een afstand van ongeveer 10 à 15m, omdat dit de nauwkeurigheid ten goede komt en eenvoudiger is qua communicatie. Het labo beschikt over een collectie vuurwapens die allemaal gebruikt kunnen worden om mee te vuren. Meestal wordt er echter gebruik gemaakt van een modulair vuurwapen dat snel kan worden omgebouwd om met verschillende kalibers te vuren.

Het projectiel wordt op het einde van de tunnel afgeremd door een volume rubberen blokken. Indien een projectiel deze blokken zou missen of zelfs erdoor zou penetreren zijn er stalen platen onder een hoek van 45° graden opgesteld die het projectiel vernietigen en alle brokstukken naar onder toe projecteren. Achter deze stalen platen is er nog een betonnen muur en daarachter bevinden zich dan weer andere labo's. Het is dus belangrijk dat er zeker geen projectielen door de muur kunnen penetreren.

Op het einde van de schietunnel, langs de doelenruimte, is er een controlekamer van waaruit de doelen en de laatste 20 meter van de schietunnel zichtbaar zijn. De ramen in deze kamer zijn natuurlijk van kogelwerend glas zodat eventuele ricochets geen verwondingen kunnen veroorzaken. In deze controlekamer bevinden zich ook de meeste meettoestellen, behalve deze die in de schietunnel zelf aanwezig moeten zijn, en een aantal pc's om deze metingen te verwerken.

Tussen loop en doel wordt bijna altijd de snelheid van het projectiel gemeten: dit laat toe om bijvoorbeeld de energie van het projectiel te berekenen. Er zijn verschillende methodes om de snelheid te meten, die in de volgende paragraaf meer in detail zullen worden uitgelegd.

Op Figuur 1 is een schets te zien van het grondplan van het labo ballistiek.



Figuur 1: schets van het grondplan van het labo ballistiek

2.2 Gebruikte Hardware

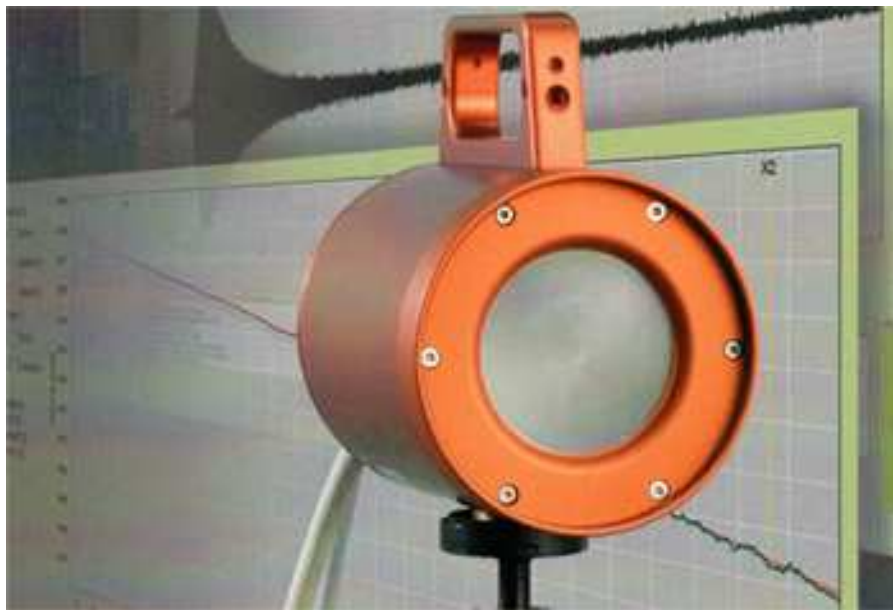
In dit hoofdstuk volgt er een bespreking van de gebruikte meettoestellen, de FPGA module en de fotocamera.

2.2.1 Snelheidsmetingen

In het labo worden er twee methodes gebruikt om de snelheid van het projectiel te bepalen: Een Doppler radar of infrarood lichtschermen. Deze laatste methode wordt veruit het meest gebruikt in het labo vanwege de nauwkeurigheid en de eenvoudige aansluiting. Hieronder volgt een meer uitgebreide beschrijving van deze methodes.

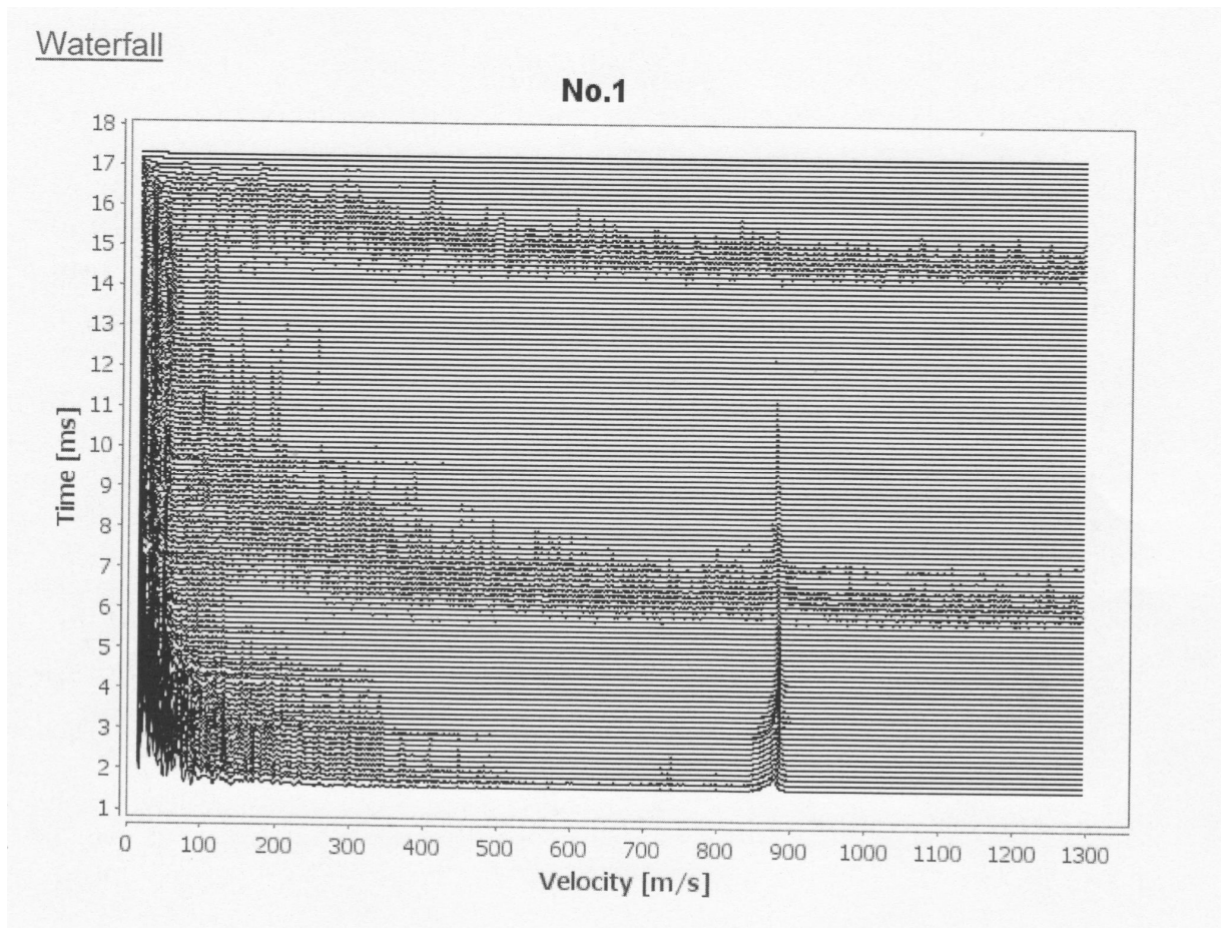
Doppler radar

Bij deze methode wordt er gebruik gemaakt van een kleine doppler radar (ontvanger + zender) die meestal langs de loop wordt opgesteld. Een voorbeeld hiervan is de B481 HPI Doppler radar zoals afgebeeld op Figuur 2.



Figuur 2: B481 HPI Doppler radar [3]

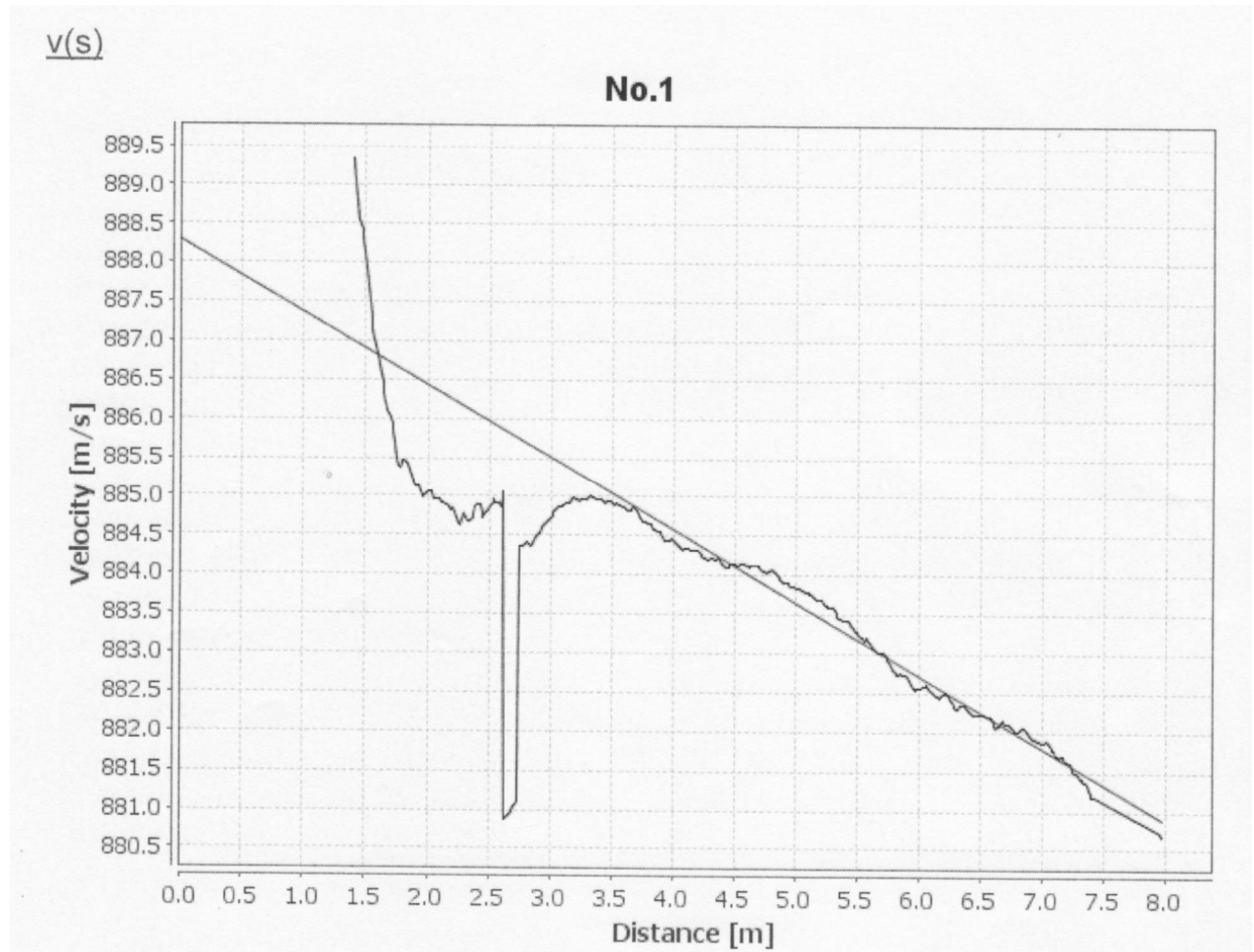
Deze radar zendt een golf uit met een bepaalde frequentie. De golf kaatst terug op het projectiel en verandert ook van frequentie ten gevolge van het Doppler effect omdat het projectiel zelf een bepaalde snelheid heeft. De radar vangt deze weerkaatste golf terug op en analyseert het frequentiespectrum ervan. De radar is gekoppeld aan een pc die dan een FFT (Fast Fourier Transformation) uitvoert op deze data. Vervolgens krijgen we een beeld zoals op Figuur 3, de scherpe piek zichtbaar bij ongeveer 900 m/s geeft de snelheid van het projectiel aan in functie van de tijd.



Figuur 3: waterfall diagram

We kunnen ook zien dat de piek lichtjes opschuift in functie van de tijd omdat het projectiel vertraagt nadat het de loop heeft verlaten. Na ongeveer 11 milliseconden valt het signaal plots weg omdat de Doppler radar het projectiel niet meer kan waarnemen.

Vervolgens kan hier ook nog een diagram van gemaakt worden dat de snelheid in functie van de afstand zichtbaar maakt. Dit wordt getoond in Figuur 4. De grillige lijn is de ruwe data die de doppler radar gemeten heeft. De verschillende pieken en dalen zijn te wijten aan reflecties en onnauwkeurigheden. De pc tekent ook een eerste orde interpolatiecurve, omdat de snelheid van een projectiel ongeveer lineair afneemt in functie van de afstand [4].



Figuur 4: snelheidsdiagram gemeten met Doppler radar

Voordelen

- Kan de snelheid van het projectiel meten vanaf een korte afstand nadat het de loop heeft verlaten tot het moment van impact.
- Neemt weinig plaats in, en kan langs de loop geplaatst worden zodat indien gewenst de loop tot vlakbij het doel geplaatst kan worden.

Nadelen

- Gevoelig aan storing, na ongeveer 10 meter is het resultaat meestal niet meer bruikbaar omdat de weerkaatsingen in de schietunnel teveel interferentie veroorzaken.
- Minder nauwkeurig dan de andere methodes [5].

Deze methode zal niet gebruikt worden om een snelheidsmeting voor de FPGA software uit te voeren. De reden hiervoor zijn de grotere complexiteit, kleinere nauwkeurigheid en de grotere rekenkracht die nodig is om deze data te verwerken [5]. Er wordt wel een aanzet gegeven naar een mogelijke uitbreiding waarbij de snelheidsmeting met een Doppler radar uitgevoerd wordt in de gevorderde handleiding die terug te vinden is in bijlage B.

Infrarood lichtschermen

Werking

Deze methode maakt gebruik van 2 of 4 infrarood lichtschermen die vastgemaakt zijn op een 'basis'. De basis is een stalen constructie waarvan de afmetingen exact gekend zijn en die regelmatig gekalibreerd wordt. Tegenover de LED's die infrarood licht uitzenden is een reflector opgesteld en wanneer het projectiel door de lichtschermen vliegt is er een stukje van de reflector dat geen licht meer zal ontvangen omdat het projectiel een schaduw werpt. Hierdoor wordt er een digitale puls gegenereerd in een rekeneenheid van het lichtscherm zelf. Deze puls wordt vervolgens, via een coaxiale kabel, doorgestuurd naar een meettoestel dat het tijdsinterval tussen deze twee pulsen zal opmeten. Meestal wordt hiervoor een chronometer gebruikt die automatisch deze tijd kan omrekenen naar een snelheid. Voor het RTTS zal de FPGA zelf het tijdsinterval opmeten omdat communicatie met de chronometer te traag zou zijn.

LS 19 i3

In het labo zijn meerdere meetbasissen aanwezig, met lengtes variërend van 0,5 tot 3 m. De meetbasis die gebruikt is om dit project te verwezenlijken is de "LS 19 i3" van Drello Ballistics. De datasheet van deze meetbasis is terug te vinden in bijlage D. Ook is deze afgebeeld in Figuur 5.



Figuur 5: LS 19 i3 meetbasis van Drello Ballistics

Deze basis beschikt over 4 infrarood lichtschermen, waarbij de afstand tussen de binnenste twee lichtschermen exact 2 meter bedraagt en de afstand tussen de buitenste twee lichtschermen 2,229 meter bedraagt.

Doordat er vier infraroodlichtschermen aanwezig zijn in plaats van twee kan de snelheidsmeting twee maal uitgevoerd worden. Zo kan een eventuele foutieve meting opgespoord worden door het resultaat van beide snelheidsmetingen te vergelijken. De meting van de buitenste twee lichtschermen wordt altijd als resultaat gebruikt omdat deze iets nauwkeuriger is door de langere afstand.

Voor- en nadelen

Een meetbasis geeft ons de mogelijkheid om de tijd te meten die een projectiel nodig heeft om de afstand tussen de twee lichtschermen af te leggen. We kunnen hier dus geen momentane snelheid mee meten. Omdat we de afstand tussen beide lichtschermen kennen, kunnen we natuurlijk wel de gemiddelde snelheid over het interval berekenen. Dit heeft ook als gevolg dat we de versnelling van het projectiel niet kunnen opmeten met behulp van een meetbasis.

Een ander nadeel is dat bij de verbranding van kruit in de kamer van een vuurwapen er ook infrarood licht ontstaat. Dit licht komt samen met het zichtbare licht uit de loop geschoten onder de vorm van een mondingsvlam. Wanneer de loop te dicht staat opgesteld bij de meetbasis is het mogelijk dat de meetbasis dit licht als een volgend projectiel detecteert. Afhankelijk van het type wapen en munitie moeten we dus steeds voldoende afstand respecteren tussen de loop van het wapen en de eerste lichtbasis.

Een groot voordeel echter is de nauwkeurigheid: omdat de afstand tussen de twee lichtbasissen zeer exact gekend is en omdat deze dan ook nog eens regelmatig geijkt worden is de fout op een snelheidsmeting minimaal. In de datasheet wordt de fout op maximaal 0,2% geschat. In de praktijk is deze fout bijna altijd minder dan 0,2% [4].

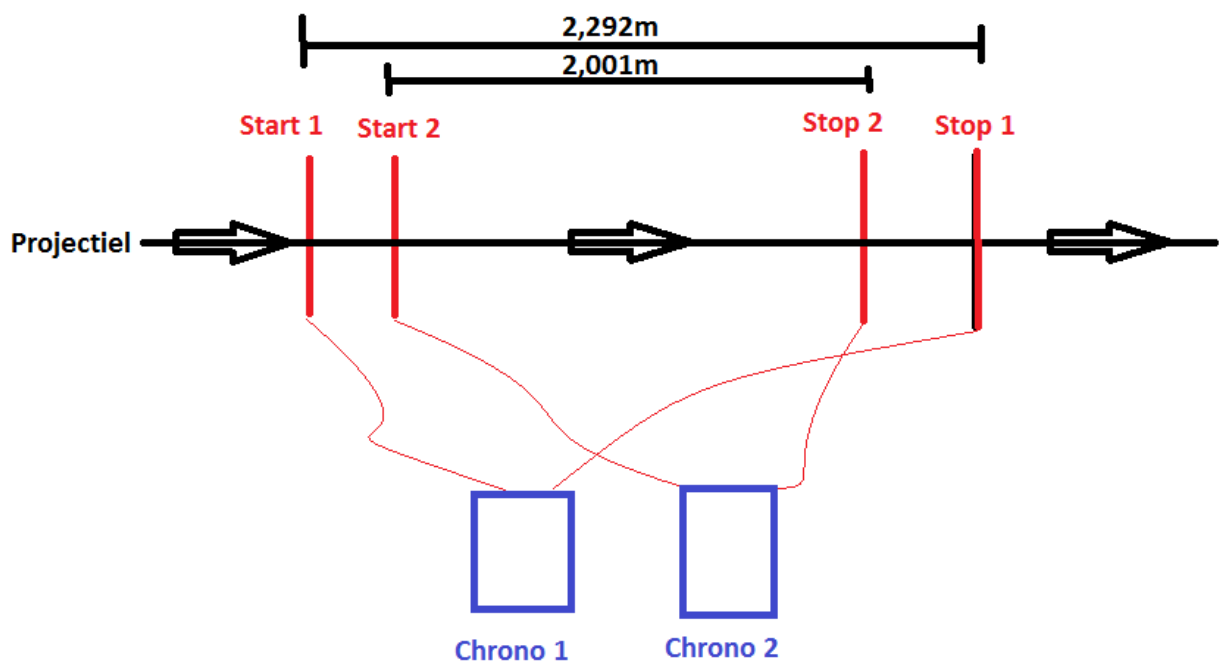
Verder is het ook zeer eenvoudig om deze meetbasis op te stellen en aan te sluiten: er is een rail in de grond waar de meetbasis in vastgezet kan worden en qua aansluiting volstaat het om twee coaxiale kabels te verbinden met het toestel dat het tijdsverschil gaat meten.

Conclusie

Omwille van de grote nauwkeurigheid en het gebruiksgemak van de lichtbasis gaat de FPGA software ook opgebouwd worden rond deze meetmethode. Zoals reeds aangehaald is steeds de “LS 19 i3” van Drello Ballistics gebruikt om snelheidsmetingen mee uit te voeren. Omdat alle meetbasissen gelijkaardig zijn opgebouwd is het wel mogelijk om elke andere meetbasis uit het labo aan te sluiten op de FPGA zonder aanpassingen aan de code zelf te moeten uitvoeren, alleen de parameters op het “*front panel*” moeten aangepast worden.

2.2.2 Chronometers

Als een lichtbasis gebruikt wordt om de snelheid van een projectiel te meten moet natuurlijk ook de tijd gemeten worden die het projectiel nodig heeft om de afstand tussen de twee lichtschermen af te leggen. Vervolgens kunnen we dan de afstand tussen de twee meetpunten (die normaal zeer precies gekend is) delen door de gemeten tijd om zo de snelheid van het projectiel te bekomen. Hiervoor maakt men gebruik van twee zeer nauwkeurige chronometers. Eén chronometer is natuurlijk voldoende maar meestal worden beide chronometers aangesloten (zoals getoond op Figuur 6) zodat er onderlinge controle mogelijk is. Als het resultaat tussen beide chrono's een grote afwijking vertoont weet men dat er iets niet correct is en kan men het probleem proberen op te lossen. Als er slechts één chrono gebruikt is, is er niets wat aantoont of de gemeten tijd/snelheid wel correct is (behalve de ervaring en intuïtie van de gebruiker). Beide chronometers bezitten ook een ingebouwde rekenfunctie en omdat meestal toch dezelfde lichtbasis gebruikt wordt, kunnen deze afstanden ingegeven worden in de chronometers (de zogenaamde 'k-waarde') zodat de chronometer onmiddellijk de snelheid weergeeft op het scherm in plaats van het gemeten tijdsinterval tussen de start- en stoppuls. Bij deze masterproef zullen de chronometers niet rechtstreeks gebruikt worden om de tijd te meten (de FPGA module voert deze meting zelf uit) maar ze laten wel toe om de bekomen resultaten te controleren.



Figuur 6: aansluitdiagram chronometers voor “Ls 19 i3”

2.2.3 De PXI met geïntegreerde FPGA module

Een Field Programmable Gate Array (FPGA) is een geïntegreerde schakeling bestaande uit programmeerbare logische blokken. Eén logische blok van een FPGA bestaat uit een flipflop en een Look Up Table (LUT). Afhankelijk van hoe de LUT geprogrammeerd is kan zo één blok verschillende functies vervullen: schuifregister, AND poort, geheugen, XOR poort, etc... De FPGA module zorgt er dan ook voor dat alle aparte logische blokken correct samenwerken (routing genaamd) om zo alle inputsignalen te verwerken tot het gewenste output signaal [6].

De hierboven beschreven functionaliteit is natuurlijk ook aanwezig in een pc of PLC, maar het grote verschil is dat een FPGA module veel sneller werkt dan een pc of PLC [7]. De reden hiervoor is dat de compiler het programma van de gebruiker 'vertaalt' naar hardware niveau en dan de logische blokken van de FPGA zo programmeert dat ze op hardware niveau doen wat de gebruiker in de software gevraagd heeft. Eenmaal geprogrammeerd werkt een FPGA dus volledig op hardware niveau, waardoor de relatief grote vertragingen die in de communicatie tussen hardware en software optreden weggewerkt worden. Het nadeel hiervan is wel dat FPGA's weinig rekenkracht bezitten, en ook vrij duur zijn. Wel kan een FPGA opnieuw en opnieuw geprogrammeerd worden, in tegenstelling tot een Application-Specific Integrated Circuit (ASIC) [6].

De FPGA module die opgesteld staat in het labo is een zogenaamde PXI (PCI Extensions for Instrumentation) van National Instruments. Dit is een grote, witte, metalen kast met links een embedded controller en daarlangs plaats voor maximaal achttien uitbreidingskaarten. Qua opbouw is dit vergelijkbaar met de opbouw van een PLC. De embedded controller is eigenlijk een gewone pc in een compacte behuizing die toelaat om de kaarten te programmeren en uit te lezen met behulp van LabVIEW, een programma van National Instruments.

De PXI in het labo, weergegeven op Figuur 7, bestaat uit één embedded controller NI PXIe-8133 (1.73 GHz Quad-Core PXI Express Controller) met daarlangs 1x NI PXI-5922 (24-Bit Flexible-Resolution Digitizer), één FPGA module NI PXI-7851R (R Series Multifunction RIO with Virtex-5 LX30 FPGA), 4x NI PXI-5122 (100 MS/s, 14-Bit Digitizer/Oscilloscope) en 1x NI PXIe-6368 (Simultaneous X Series Data Acquisition).



Figuur 7: National Instruments PXI met een FPGA kaart in slot 3

Voor deze masterproef wordt enkel de FPGA module gebruikt die geprogrammeerd wordt met de embedded controller (pc). Het hart van de FPGA kaart is een FPGA microprocessor van Xilinx. Deze FPGA module beschikt maximaal over 8 analoge ingangen, 8 analoge uitgangen en 96 digitale in- of uitgangen. De maximale sampling frequentie voor de digitale poorten is 40 MHz. De 7851R beschikt over 19200 flipflops om berekeningen uit te voeren. Andere details zijn terug te vinden in bijlage A waar de datasheet van de NI PXI-7851R hernomen is.

De FPGA NI PXI-7851R beschikt over 3 68-pins VHDCI (Very High Density Cable Interconnect) connectoren die verbonden kunnen worden met de bijgeleverde SCB-68 “break-out boxes”: dit zijn klemmenstroken onder de vorm van een doosje ervoor zorgen dat de 68 pinnen makkelijker te verbinden zijn met een kabel. Deze doosjes worden met een speciale kabel verbonden met de VHDCI connector en hierop kunnen we dan analoge en digitale in- of uitgangen aansluiten. Een afbeelding van een SCB-68 is hieronder weergegeven in Figuur 8.



Figuur 8: een SCB-68 break-out box

De digitale oscilloscoopkaarten die eveneens aanwezig zijn in de PXI kunnen gebruikt worden om tot acht signalen (analoog of digitaal) op te meten. Deze functionaliteit wordt niet gebruikt bij de gewone werking van het programma omdat de FPGA dan zelf alle berekeningen en communicatie voor zijn rekening neemt. In de ontwikkelings- of testfase zijn deze oscilloscoopkaarten wel zeer nuttig omdat ze gebruikt kunnen worden om de signalen op te meten die de FPGA moet verwerken of uitsturen. Zo is het mogelijk om de timing van de FPGA te controleren. Ook de amplitude en pulsbreedte van een signaal kunnen hiermee bepaald worden.

2.2.4 Camera's

Het uiteindelijke doel is om een foto te nemen van een projectiel in vlucht. Door de hoge snelheid van de projectielen (tot $1000 \text{ m/s} = 3600 \text{ km/h}$) zal een gewoon fototoestel natuurlijk niet voldoen. De reden hiervoor is dat de sluitertijd van de camera zeer klein moet zijn. De sluitertijd is de tijdsduur die beschrijft hoelang een lichtgevoelige plaat of ander lichtgevoelig element wordt blootgesteld aan een bepaalde hoeveelheid licht. Commerciële camera's hebben sluitertijden die variëren tussen 30 seconden en 250 microseconden. Op dit interval van $250 \mu\text{s}$ zal een projectiel met een snelheid van 1000 m/s dus 25 centimeter afleggen. Op de foto zal het projectiel dus als het ware uitgetrokken worden tot een lengte van meer dan 25 centimeter. Dit fenomeen noemt men de bewegingsonscherpte of motion blur in het Engels.

Figuur 9 toont een voorbeeld van bewegingsonscherpte.



Figuur 9: bewegingsonscherpte bij een kermisattractie [8]

Om de bewegingsonscherpte klein te houden moeten we gebruik maken van een camera met een heel kleine sluitertijd. Het RTTS is compatibel met alle camera's die een TTL puls als trigger aanvaarden. In het labo zijn verschillende camera's aanwezig die geschikt zijn om beelden met een korte sluitertijd te maken.

De twee camera's die getest zijn met betrekking tot het RTTS zijn de Hadland SVR2 camera en de Photron Fastcam SA5 hogesnelheidsvideocamera.

Hadland SVR2

De Hadland SVR2, weergegeven in figuur 10, is een digitale camera die speciaal ontworpen is voor toepassingen binnen de ballistiek, meer specifiek om projectielen tegen hoge snelheid te fotograferen. Deze camera kan één foto nemen waarna deze foto manueel moet opgeslagen worden voordat er opnieuw een foto genomen kan worden.

Het voordeel hiervan is dat er gebruikt gemaakt kan worden van flitsers in plaats van continue belichting. Deze flitsers kunnen opgeladen worden met behulp van condensatoren waardoor er ogenblikkelijk minder elektrische energie nodig is. Ook is de lichtopbrengst van een flitser veel hoger dan van een gewone lamp, waardoor er minder flitsers nodig zijn vergeleken met wanneer er lampen gebruikt zouden worden. Een ander voordeel is dat de sluitertijd (200 ns) korter is dan deze van de Photron (1 μ s) hogesnelheidsvideocamera en dat de resolutie van de Hadland (2048 x 2048 pixels) hoger is dan van de Photron (1024 x 1024 pixels). Er is geen datasheet meer beschikbaar van deze camera dus de meeste van deze parameters zijn mondeling overgeleverd.



Figuur 10: Hadland SVR2 digitale camera

Het nadeel is natuurlijk dat één enkele foto vrij weinig informatie biedt wanneer we een gebeurtenis willen vastleggen die een tijdje duurt (bijvoorbeeld de impact van een projectiel op een stalen plaat). Indien het projectiel niet op de foto zichtbaar is, is het ook onmogelijk om te achterhalen of de foto te vroeg of te laat genomen werd, omdat we maar over één frame beschikken.

De SVR2 heeft een ingebouwde pc die op Windows 95 draait of er is een laptop die aangesloten kan worden en zo de controle van de camera kan overnemen. Op deze laptop is het programma 'SVR' geïnstalleerd waarmee verschillende parameters zoals sluitertijd, opening van het diafragma, ... ingesteld kunnen worden. Meer informatie over dit programma is terug te vinden in hoofdstuk 3.2.

De SVR2 beschikt ook over een ingebouwde FPGA kaart die de tijdsvertraging van een projectiel kan berekenen. Hiervoor moet de geschatte snelheid van het projectiel op voorhand ingegeven worden samen met de afstand tussen het lichtscherm (dat de trigger moet geven aan de Hadland) en de camera zelf. Omdat deze berekening de snelheid van het projectiel niet in realtime bepaalt, zal er een vrij grote fout optreden waardoor het mogelijk is dat het projectiel niet centraal in beeld staat. Deze functionaliteit wordt niet gebruikt bij het RTTS.

Photron Fastcam SA5

De Photron Fastcam SA5, weergegeven in Figuur 11, is een digitale hogesnelheidsvideocamera. Om een vloeiende opname van een projectiel dat beweegt tegen hoge snelheid te maken, is een hoge framerate (aantal beelden per seconde) nodig. De Photron kan tot maximaal één miljoen beelden per seconde maken. De minimale sluitertijd per frame is $1\mu\text{s}$. De sluitertijd kan onafhankelijk van de framerate ingesteld worden, maar de sluitertijd moet natuurlijk kleiner of gelijk zijn aan de tijd tussen twee frames.

De resolutie van de Photron Fastcam SA5 is maximaal 1024×1024 pixels, maar deze neemt af als de framerate toeneemt. Bij een framerate van één miljoen beelden per seconde bedraagt de resolutie nog maar 64×16 pixels. Een volledig overzicht van de selecteerbare framerates met bijbehorende resoluties is terug te vinden in de datasheet die hernomen is in bijlage E.

Het voordeel van de Photron is de mogelijkheid om meerdere frames na elkaar op te nemen, waardoor de timing minder nauwkeurig moet zijn. Ook kan een gebeurtenis die enkele microseconden duurt volledig geregistreerd worden.

De nadelen zijn een hogere sluitertijd en lagere resolutie dan de Hadland SVR2, en de nood aan continue belichting in plaats van flitsers. Dit heeft tot gevolg dat er vaak een hele batterij aan lampen opgesteld moet worden, samen met een uitgebreide elektrische voorziening om deze van energie te voorzien.

De camera wordt extern aangestuurd met behulp van een pc. Bij de camera zit software meegeleverd: Photron Fastcam Viewer (PFV). Deze software biedt de mogelijkheid om alle parameters van de camera correct in te stellen, en kan ook (beperkt) gebruikt worden om de foto's na te bewerken. Meer informatie over dit softwarepakket is terug te vinden in paragraaf 4.3.



Figuur 11: Photron Fastcam SA5 digitale hogesnelheidsvideocamera [9]

2.2.5 Vergelijking met andere hedendaagse systemen en methodes

Inleiding

In deze sectie wordt een korte bespreking gegeven van moderne technologieën en systemen die ontwikkeld zijn, en die een vervanging of aanvulling kunnen zijn op het realtime triggersysteem dat het onderwerp uitmaakt van deze masterproef. Zo krijgt de lezer een beter beeld van waar de wetenschap vandaag staat op het gebied van hogesnelheidsfotografie van ballistische projectielen, en de aansturing van deze camera's.

Het gebruik van een state of the art hogesnelheidsvideocamera

Door een modernere versie van de Photron Fastcam Sa5 te gebruiken, bijvoorbeeld de Phantom V2010 van Vision [10], kan de sluitertijd nog kleiner gekozen worden, terwijl de resolutie hoger ligt. Deze oplossing zou de nood voor een FPGA sturing opheffen, maar de kostprijs zou vele malen hoger zijn, terwijl een hedendaagse hogesnelheidsfotocamera nog betere eigenschappen kan voorleggen [11].

Het gebruik van een state of the art framing camera

Een *framing* camera is een kruising tussen een fotocamera en een videocamera: de camera kan geen continue film maken, maar kan wel meerdere foto's na elkaar maken met een zeer kort interval tussen twee beelden. Dit zorgt ervoor dat, zolang de snelheid van de projectielen niet te veel varieert, er toch steeds een bruikbaar frame gemaakt zal worden. Een voorbeeld van zo'n camera is "*The Specialised Imaging Ultra Fast Framing Camera* [12]. Doordat de camera geen continue beelden moet opnemen zal de resolutie en sluitertijd beter zijn dan deze van een hogesnelheidsvideocamera.

Een nadeel hiervan is wel dat de nauwkeurigheid nooit zo groot kan worden als bij het gebruik van een videocamera of het RTTS omdat de tijd tussen twee frames een vaste waarde heeft: het is dus mogelijk dat op het eerste frame het projectiel nog niet volledig zichtbaar is, en op het tweede frame het projectiel al gedeeltelijk terug onzichtbaar is.

Het gebruik van een dubbele foto

Door twee foto's kort na elkaar te nemen is het mogelijk om bepaalde tijdsafhankelijke parameters zoals rolsnelheid, verandering van de hoek tussen de lengte as van het projectiel en zijn baan, etc te bepalen. Om toch de hoge resolutie en korte sluitertijd van een fotocamera te kunnen gebruiken, bestaan er nu speciale hogesnelheidsfotocamera's die twee foto's op korte tijd na elkaar nemen [11]. Hierdoor is het mogelijk om meer data uit deze afbeeldingen te halen.

Bij gebruik van dit systeem is het wel belangrijk dat de trigger om de eerste foto te nemen zeer nauwkeurig bepaald wordt, dus dan dringt een realtime triggersysteem zich opnieuw op als de resultaten van de eerste keer correct moeten zijn.

Conclusie

Bij het gebruik van een moderne hogesnelheidsvideo- of *framing* camera kan de kwaliteit van de foto's verbeteren ten opzichte van de huidige situatie in het labo, zonder gebruik van een RTTS. Het nadeel van deze oplossing is natuurlijk de zeer hoge kostprijs.

Door een nieuwer type hogesnelheidsfotocamera te gebruiken, kunnen we resultaten bekomen die nog beter zijn dan deze behaald met een moderne videocamera, maar dan blijft de noodzaak voor een RTTS bestaan natuurlijk. Het is ook daarom dat het RTTS niet alleen nuttig is om te gebruiken met de camera's aanwezig in het labo, maar mogelijk ook voor de camera's die in de toekomst aangekocht zullen worden.

3 Software

3.1 LabVIEW

Inleiding

Om de FPGA module te programmeren wordt Laboratory Virtual Instrument Engineering Workbench (LabVIEW) gebruikt, een programma geschreven door National Instruments. Deze software bevat een speciale FPGA programmeermodule die op maat gemaakt is voor de PXI/FPGA die in het labo staat. Deze software wordt ook bij de hardware geleverd.

LabVIEW is een grafische programmeertaal waar data de virtuele draden volgt en verwerkt wordt door virtuele functieblokken, die ieder hun eigen toepassing kennen. LabVIEW biedt een uitgebreide ondersteuning voor allerlei meettoestellen, zeker degene die gemaakt zijn door National Instruments.

VI

Een programma dat in LabVIEW geschreven wordt heet een Virtual Instrument (VI). Dit programma bestaat altijd uit twee delen, net zoals een echt meettoestel: enerzijds is er het ‘*front panel*’ waar alle knoppen en invoerschermen op te zien zijn, even als de resultaten en andere gegevens. Anderzijds is er het ‘*block diagram*’ waar de verwerking van alle data gebeurt aan de hand van virtuele draden en blokken die ieder een eigen specifieke functie hebben.

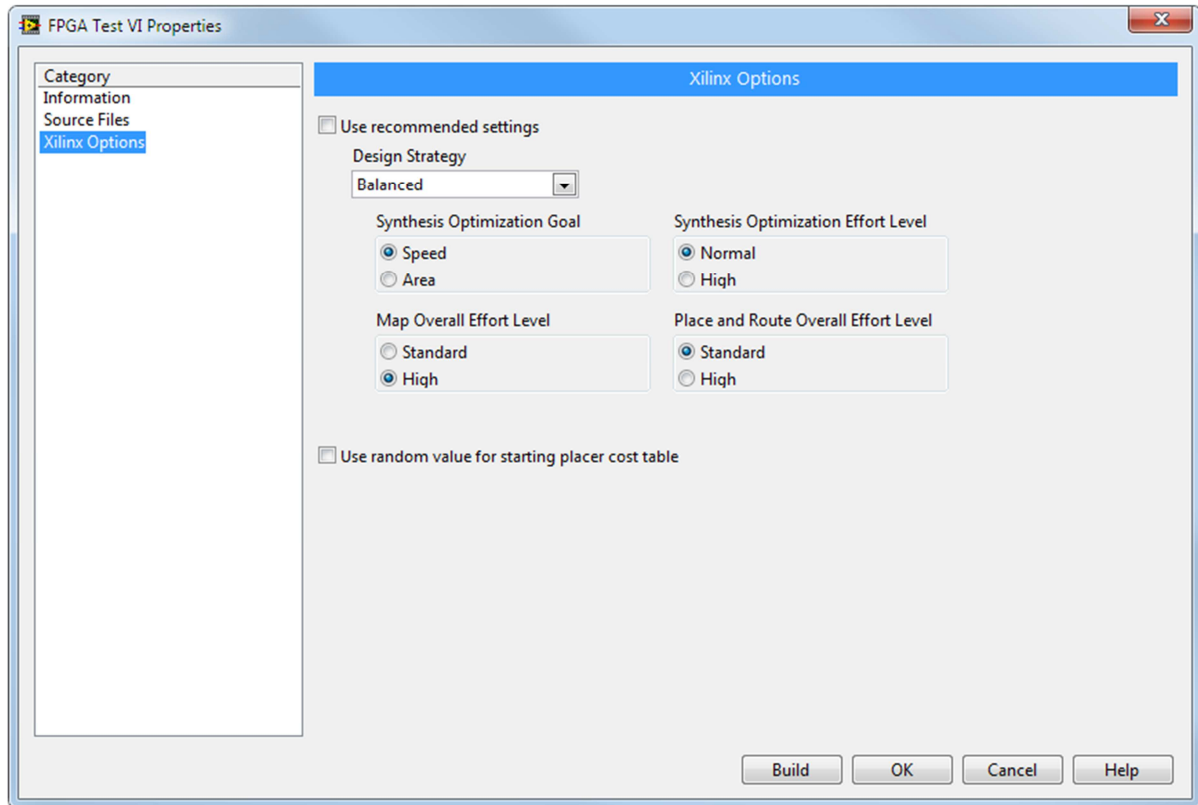
Compilatie

Alle code die geschreven wordt moet eerst gecompileerd worden voordat deze uitgevoerd kan worden. Eigenlijk vertaalt de compiler (het programma dat de compilatie uitvoert) de taal waarin de programmeur schrijft naar een taal die de computer begrijpt. De compilatie van eenvoudig LabVIEW programma op pc duurt hooguit enkele seconden. Als we daarentegen een programma in LabVIEW schrijven voor op de FPGA, kan dit al gauw 5 à 30 minuten duren om te compileren.

Deze lange compilatieduur is een gevolg van de “moeilijkere vertaling” die uitgevoerd moet worden [13]: zoals aangehaald in sectie 2.2.3, werkt een FPGA exclusief op hardware niveau met XOR, NAND, ... poorten. De compiler moet dus eigenlijk de beste (de gebruiker bepaalt waar de focus op ligt) manier vinden om de beschikbare poorten te schikken tot een optimaal werkend geheel. Dit proces noemt routing en maakt ook deel uit van het compilatieproces.

LabVIEW voorziet de mogelijkheid om de compilatie (tegen betaling) te laten uitvoeren door een “compilation farm”: dit is een cluster van computers die samen een heel grote rekenkracht bezitten en zo de compilatie veel sneller uitvoeren. Ook is het mogelijk om zelf een server vanop afstand te selecteren die de compilatie uitvoert.

Zoals net aangehaald kan de gebruiker zelf instellen bij “compilation settings” waar de focus op ligt: snelheid van uitvoeren, snelheid van compileren, zo weinig mogelijk poorten gebruiken,... Afhankelijk van de gekozen methode kan het compilatieproces iets langer duren. De interface waar deze instelling gekozen moet worden is weergegeven in Figuur 12.



Figuur 12: interface met compilatie instellingen

Bij de kleinste wijziging aan de code zal de FPGA code opnieuw gecompileerd moeten worden. Zelfs het uitlezen van één variabele zal er al voor zorgen dat het hele compilatieproces opnieuw moet uitgevoerd worden. Tijdens de ontwerpfase kan het daarom interessant zijn om de FPGA te simuleren op een gewone pc, omdat de lange compilatietijd dan wegvalt. Eenmaal de code gecompileerd is blijft deze wel aanwezig op de FPGA kaart, zelfs als de spanning wegvalt.

Conclusie

LabVIEW is propriëtaire software en omdat de controller en FPGA kaart beide speciaal gemaakt zijn om te functioneren met LabVIEW was de keuze snel gemaakt om in LabVIEW te programmeren. Ook was deze software al aanwezig in het labo waardoor de meeste laboranten al een goede kennis hadden opgebouwd over LabVIEW.

3.2 Hadland SVR2 Software

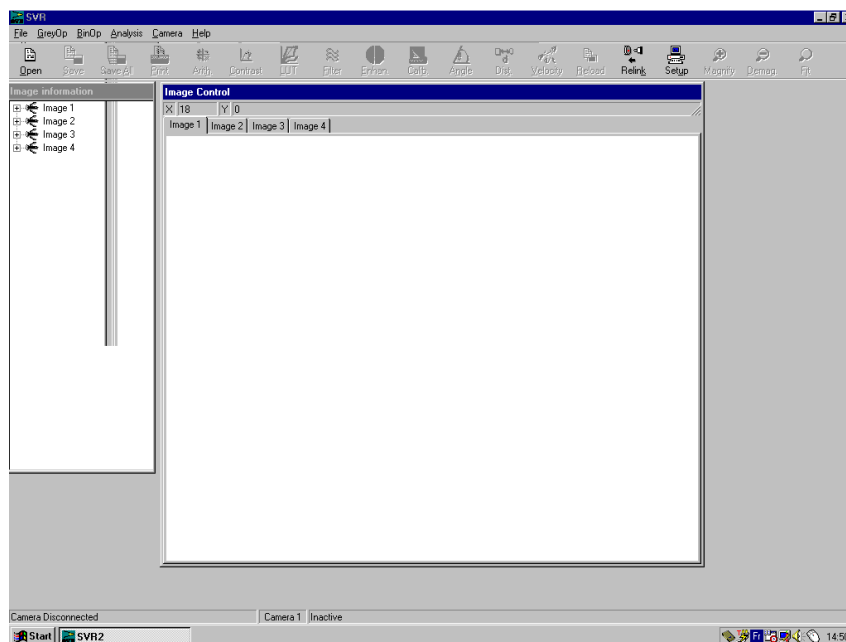
De Hadland SVR2 hogesnelheidsfotocamera heeft een ingebouwde pc die draait op Windows 95. Door de camera met een coaxiale kabel aan te sluiten op de USB-poort van een laptop (met behulp van een adapter) kan de controle overgenomen worden en gebeuren vanop de laptop. De coaxiale kabel die de camera verbindt met de laptop moet voorzien worden van een afsluitweerstand aan beide uiteinden van de kabel.

Als we de software van Hadland, die zichtbaar is in Figuur 13, opstarten neemt deze automatisch de controle over van de camera. We kunnen dan een aantal parameters instellen die een invloed hebben op de kwaliteit van de foto zoals exposure, sluitertijd, etc.

Exposure is een parameter die een maat geeft voor hoeveelheid licht die kan binnenvallen via het diafragma. De exposure van een camera wordt voorgesteld door een getal, en hoe kleiner dit getal hoe meer licht het diafragma zal laten binnenvallen op de sensor of lens.

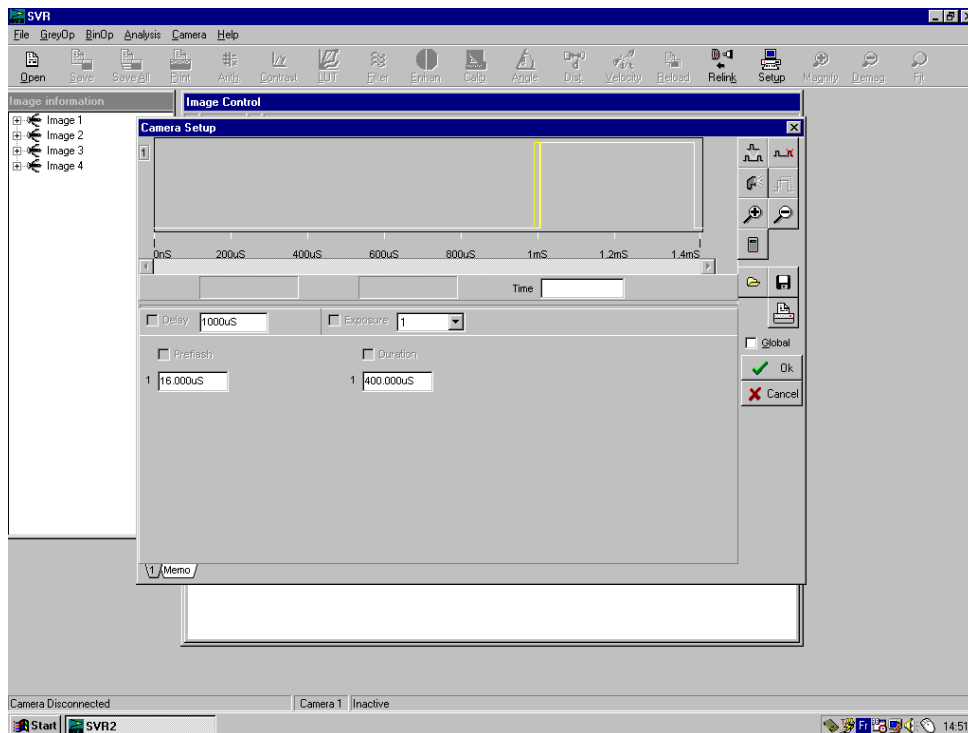
De sluitertijd is de tijd tussen het begin en einde van één foto. De sluiters moet namelijk even openen om het beeld vast te leggen op de sensor. Tijdens dit interval blijft het projectiel natuurlijk ook bewegen en daarom krijgen we bewegingsonscherpte (motion blur) in onze foto: het projectiel lijkt uitgerokken en vervormd. Het is dus belangrijk om de sluitertijd altijd zo klein mogelijk te kiezen, maar er moet natuurlijk genoeg licht zijn om de foto te belichten op deze korte tijd.

Vaak is het ook mogelijk om de focus, zoom en de opening van het diafragma in te stellen door te draaien aan de lens die gemonteerd staat op de camera. Dit is afhankelijk van het type lens dat gebruikt wordt.



Figuur 13: de software om de Hadland SVR2 aan te sturen

De Hadland camera beschikt zelf ook over een interne rekenmodule die bepaalde functies kan uitvoeren. Deze interne rekenmodule is niet uit te schakelen. Een groot nadeel hiervan is dat we altijd onze trigger een vast aantal microsecondes moeten sturen voor we onze foto willen nemen. Deze tijdsvertraging is ook in te stellen via deze software zoals te zien op Figuur 14 en moet groter zijn dan de tijd die de flitser(s) nodig hebben om hun maximale lichtintensiteit te bereiken. De tijd hangt af van flitser tot flitser en varieert ongeveer tussen de 5 en de 100 μs . Indien er geen flitser gebruikt wordt moet de vertragingstijd nog steeds minstens 1 μs bedragen, en niet 200 ns zoals de software van de Hadland suggereert [14].



Figuur 14: SVR2 software met “*Camera Setup*” weergegeven

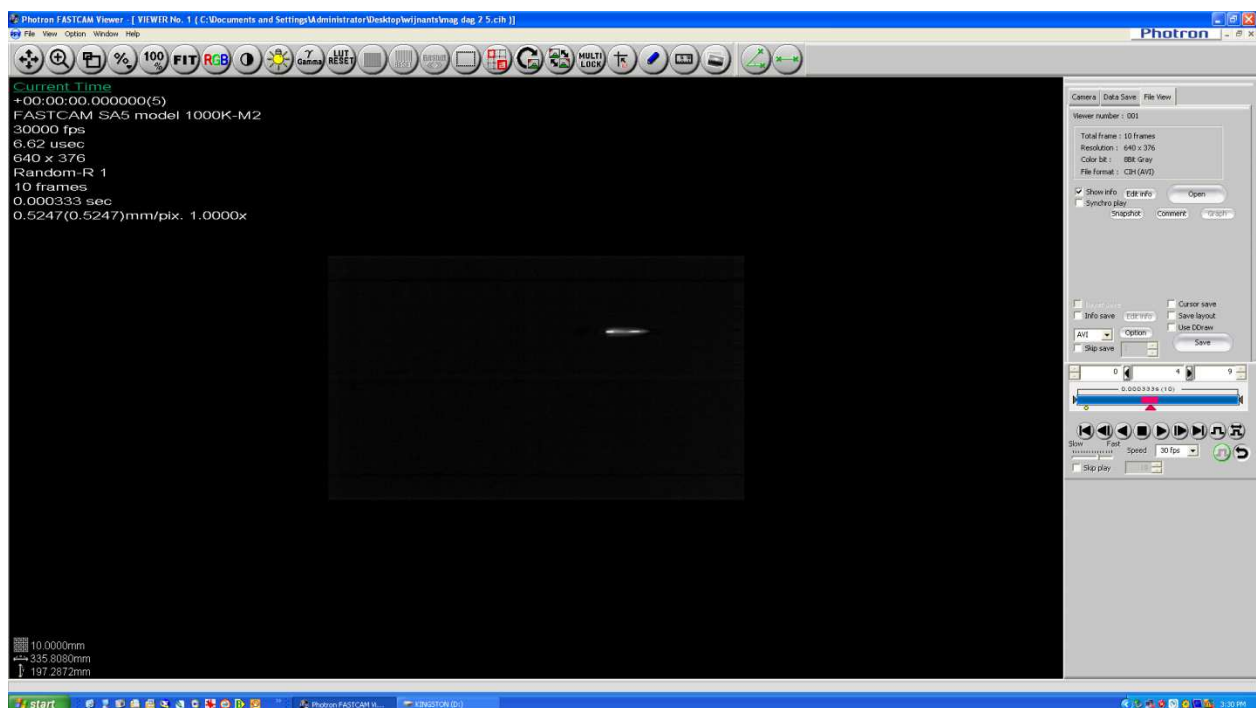
Verder geeft de software ook de mogelijkheid om de camera te wapenen, waarna de camera wacht op een triggersignaal om zijn foto te nemen. Het is ook mogelijk om een foto als test te nemen, dit is vooral handig om te testen of de belichting wel voldoende is voor de gekozen sluitertijd en exposure.

Na elke foto moet deze handmatig opgeslagen worden op de harde schijf van de computer. De software heeft ook ongeveer 20 seconden nodig voordat de foto verstuurd is van de camera naar de laptop en dus zichtbaar is. Dit heeft tot gevolg dat er minimaal ongeveer een minuut tussen elk schot zal moeten zijn voordat de verwerking en opslag van de vorige foto gelukt is.

De software beschikt ook over een basispalet aan nabewerkingsfunctionaliteiten, zoals het aanbrengen van een kruisdraad en het meten van afstanden, na kalibratie weliswaar. Deze functionaliteiten vallen buiten het bestek van deze thesis.

3.3 Photron Fastcam Viewer

De 'Photron FASTCAM Viewer' software die bij de Photron hogesnelheidsvideocamera is geleverd draait altijd op een aparte pc. In Figuur 15 is er een screenshot van deze software weergegeven. Om de camera te verbinden met deze pc moet er een ethernet kabel gebruikt worden. Omdat de Photron meerdere frames na elkaar kan opnemen, is het niet mogelijk om een flitser gebruiken. In plaats hiervan wordt er continue belichting gebruikt. Dit elimineert meteen ook het probleem van een trigger naar de flitser te moeten sturen die rekening houdt met de preflash tijd. Het nadeel hiervan is wel dat de sluitertijd nooit zo klein kan worden als bij de Hadland SVR2, omdat een continue lichtbron nooit de lichtintensiteit van een flitser kan evenaren.



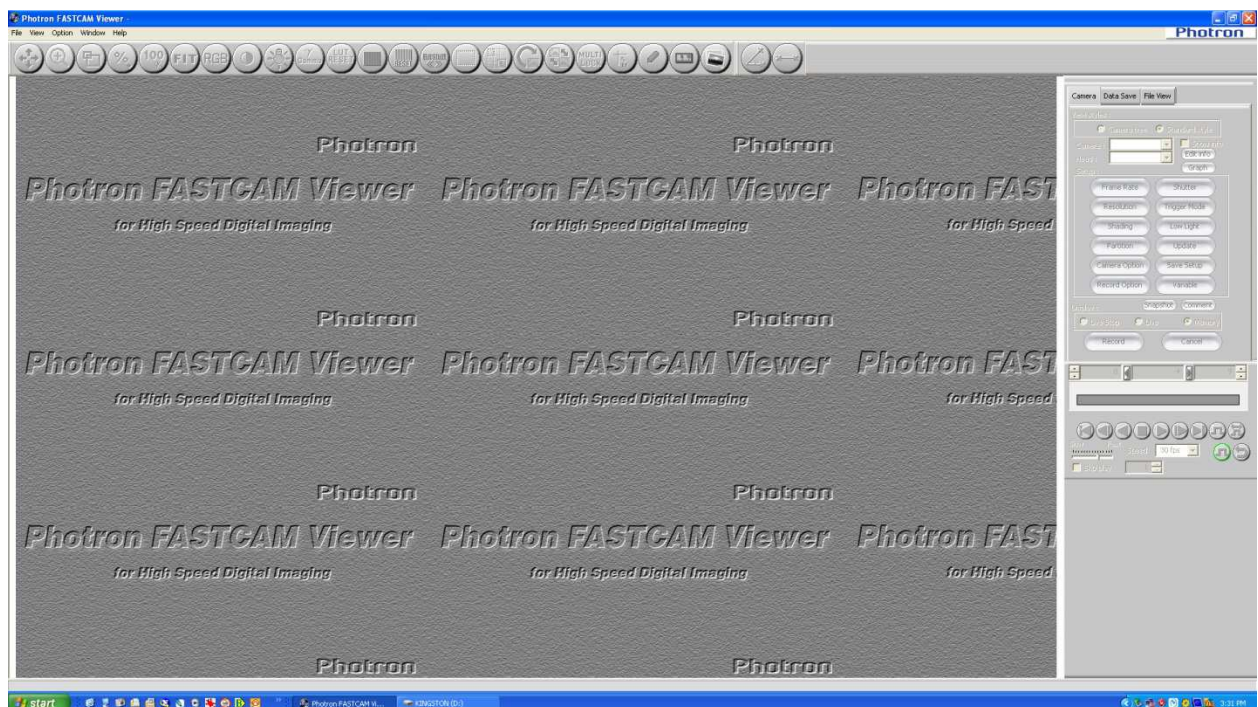
Figuur 15: Photron FASTCAM Viewer software

Eenmaal de software opgestart is, zal deze automatisch verbinding zoeken met de Photron camera. Deze verbinding wordt gemaakt aan de hand van een ethernet kabel die beide toestellen verbindt. Van zodra de connectie gelegd is, is het mogelijk om verschillende parameters van de videocamera in te stellen. Bij de Photron is het mogelijk om een framerate (aantal beelden / seconde) in te stellen, en deze parameter is een maat voor de sluitertijd: hoe hoger de framerate, hoe kleiner de sluitertijd en hoe minder motion blur er dus gaat optreden (maar er is dan wel meer licht nodig). Ook daalt de resolutie als de framerate stijgt. Het is dus best om de framerate zo hoog mogelijk te kiezen, in functie van de hoeveelheid licht en de gewenste resolutie.

Ook is het mogelijk om in te stellen hoeveel frames de Photron zal opslaan en waar de trigger ligt binnen dit interval van frames, uitgedrukt in procent. Het tijdsinterval waarin de camera dus filmt is gelijk aan het aantal frames gedeeld door het aantal frames per seconde. Indien het aantal frames op 1 ingesteld wordt, functioneert de Photron als een fotocamera. In de praktijk filmt de camera continu, en bepaalt de trigger slechts welk interval (of frame, in geval er slechts 1 frame gekozen is) hij opslaat en doorstuurt naar de pc. Als we bijvoorbeeld ervoor kiezen om 20000 frames te filmen en de trigger ligt op 50%, dan zullen 10000 frames vlak voor de trigger en 10000 frames vlak na de trigger opgeslagen en doorgestuurd worden.

Verder moet ook nog de triggermodus ingesteld worden op "*random reset*": van zodra de camera een puls krijgt, zal hij dan zijn interne klok resetten omdat de eerstvolgende stijgende flank dan sneller komt en zo er minder tijdsvertraging zit tussen de "*trigger in*" en het moment waarop de camera effectief getriggerd wordt. De andere trigger modi geven aanleiding tot een grotere vertraging.

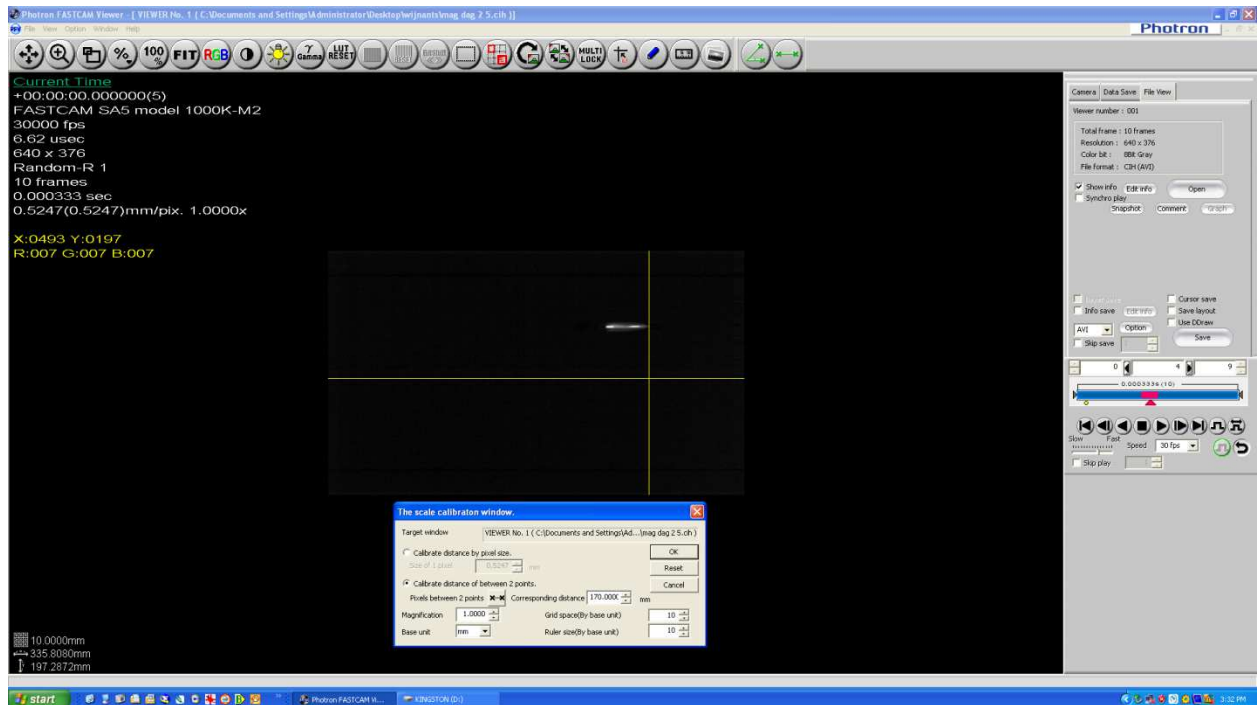
Eenmaal alle instellingen correct zijn ingesteld, kan men op "*Record*" drukken en zal de camera beginnen te lopen. Elke keer de camera een trigger krijgt zal hij nu één tijdsinterval opslaan, de lengte van dit tijdsinterval hangt af van wat door de gebruiker is gedefinieerd. Pas als de opname gestopt wordt zal hij zijn frames doorsturen naar de pc. De instellingen die zonet besproken zijn, zijn allemaal zichtbaar in Figuur 16.



Figuur 16: PFV met rechts het scherm om alle parameters in te stellen

Ontwerp van een realtime triggersysteem toegepast op de fotografie van een ballistisch projectiel

Verder beschikt deze software ook nog over de mogelijkheid om afstanden te meten (na kalibratie weliswaar) en om een kruisdraad in te stellen. Zo kan bijvoorbeeld de lengte van het projectiel in vlucht gemeten worden. De interface om een afstand te kalibreren is, samen met een kruisdraad, zichtbaar in Figuur 17. Ook kunnen twee foto's op elkaar gelegd worden, waardoor bijvoorbeeld de onderlinge afstand tussen beide projectielen gemeten kan worden. Deze functionaliteiten kunnen gebruikt worden om de correcte werking van het RTTS te controleren.



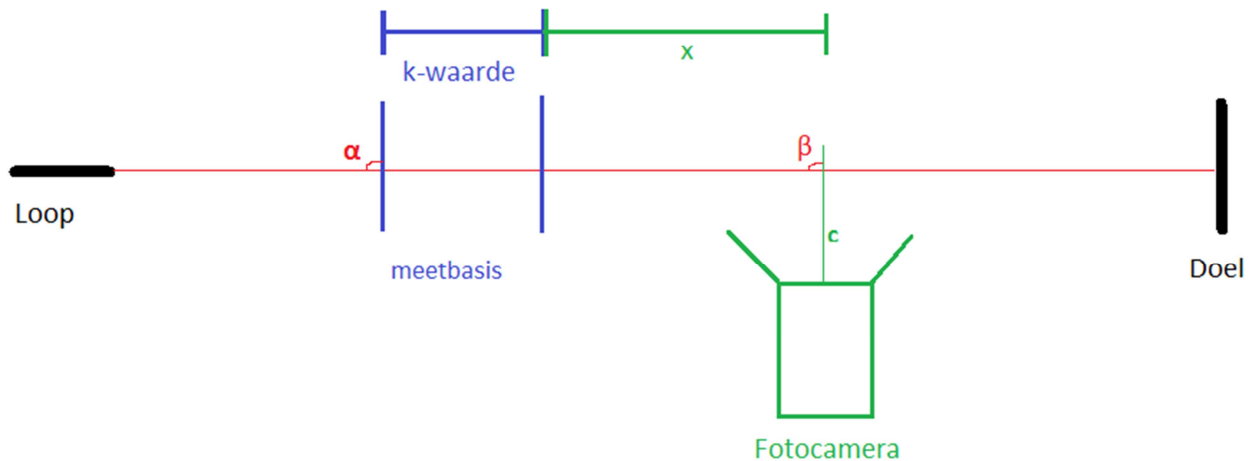
Figuur 17: Photron FASTCAM Viewer software met kruisdraad en kalibratiescherm zichtbaar.

De opgenomen data kan als een film (.avi) geëxporteerd worden of als een reeks foto's (.jpg of .png) afhankelijk van wat de gebruiker nodig heeft. Ook wordt er telkens een .cih file mee geëxporteerd, dit bevat alle parameters zoals sluitertijd, framerate, resolutie, trigger mode, etc... Het is belangrijk om op te merken dat de .cih en de .avi file dezelfde naam moeten hebben, anders kan de .cih niet meer refereren naar de .avi file en zal deze eerste niet meer werken.

4 Foutenanalyse uitgevoerd op de proefopstelling

4.1 Proefopstelling

In Figuur 18 is de opstelling van alle apparatuur in de schietunnel weergegeven zoals deze gebruikt is bij het RTTS.



Figuur 18: meetopstelling in het labo

Links is de loop van het (vuur)wapen zichtbaar in het zwart en rechts het doel, eveneens in het zwart. Daartussen zal het projectiel een ideale baan (rood) beschrijven die parallel aan de loop is bij het verlaten van de loop, en vanaf dan parabolisch verloopt ten gevolge van de zwaartekracht. Vanuit bovenaanzicht is deze baan (idealiter) een rechte.

De meetbasis is in het blauw geschetst, met de k-waarde als de afstand tussen beide basissen. Verder is de positie van de camera in het groen zichtbaar, met x de afstand tussen het laatste lichtscherm en de optische as van de camera.

Het is zo dat de baan van het projectiel niet exact loodrecht door de meetbasis gaat, en ook niet loodrecht op de optische as van de fotocamera staat. Daarom worden respectievelijk de hoeken α en β gedefinieerd als de hoek tussen de baan van het projectiel en de meetbasis en tussen de baan van het projectiel en de optische as van de fotocamera.

4.2 Foutenanalyse

In onderstaande secties wordt analytisch bepaald wat de invloed is van een fout op alle parameters die in bovenstaande sectie benoemd zijn (x-waarde, k-waarde, α , β). Hiervoor wordt er telkens één parameter per keer in rekening gebracht door de tijdsvertraging met én zonder de invloed van een fout op deze parameter te berekenen. De tijdsvertraging is de tijd tussen het moment waarop het projectiel het laatste lichtscherm passeert en het moment waarop het projectiel zich recht voor de camera bevindt.

De foutenanalyse wordt dit in formulevorm uitgewerkt, en vervolgens worden de parameters vervangen door een waarde die typisch in het labo gebruikt wordt. Zo kan bepaald worden hoe groot de invloed van een bepaalde fout is op ons eindresultaat. Ook is het mogelijk om zo de fouten onderling te vergelijken.

4.2.1 De afstand tussen de lichtschermen (k-waarde)

Noemen we k de afstand tussen beide maatbasissen en Δk het verschil tussen de reële k-waarde en de gemeten k-waarde. De snelheid v van het projectiel wordt gegeven door $k/\Delta t$, waarbij Δt de tijd is die het projectiel nodig heeft om de afstand tussen beide meetbasissen af te leggen. Dit wordt dan $(k+\Delta k)/\Delta t$ als we rekening houden met de fout. De tijdsvertraging die de FPGA gaat berekenen is gelijk aan:

$$Tijdsvertraging = \frac{x}{\frac{k+\Delta k}{\Delta t}}$$

Als we nu typische waarden invullen voor elk van de bovenstaande parameters:

$$X = 1 \text{ m}; k = 2 \text{ m}; \Delta t = 0,001 \text{ s}; \Delta k = 0,001 \text{ m}$$

$$Tijdsvertraging = \frac{x}{\frac{k+\Delta k}{\Delta t}} = \frac{1}{\frac{2+0,001}{0,001}} = 0,0004998 \text{ s}$$

$$\text{Terwijl de ideale tijdsvertraging gelijk zou zijn aan } \frac{x}{\frac{k}{\Delta t}} = \frac{1}{\frac{2}{0,001}} = 0,0005 \text{ s}$$

$$\text{Dit geeft dus een fout van } 0,0005 \text{ s} - 0,0004998 \text{ s} = 0,0000002 \text{ s} = 0,2 \text{ } \mu\text{s} \text{ of } \frac{0,0000002}{0,0005} = 0,04\%$$

In realiteit zal Δk waarschijnlijk nog veel minder dan 1 mm bedragen, omdat de basissen elke zes maanden nauwkeurig geijkt worden.

4.2.1 De afstand tussen het laatste lichtscherm en de camera (x-waarde)

De x-waarde is de afstand tussen de laatste meetbasis en de optische as van de fotocamera. Deze afstand is instelbaar en het is ook waarschijnlijk dat de afwijking op deze afstand enkele centimeters bedraagt.

De reële tijdsvertraging is gelijk aan $\frac{x+\Delta x}{\frac{k}{\Delta t}}$. Vullen we dan de typische waarden in zodat

$$x = 1 \text{ m} ; \Delta x = 0,01 \text{ m} ; k = 2 \text{ m} ; \Delta t = 0,001 \text{ s}$$

Dan bekomen we: $\frac{1+0,01}{\frac{2}{0,001 \text{ s}}} = 0,000505 \text{ s}$ terwijl de ideale tijdsvertraging weer $0,0005 \text{ s}$ zal

bedragen. Dit geeft een fout van $0,000005 \text{ s}$ oftewel $5 \mu\text{s}$. Dit komt overeen met een fout van $\frac{0,000005}{0,000500} = 1 \%$

We zien dat de invloed van deze parameter even groot is dan van de k-waarde, maar met het grote verschil dat de k-waarde zéér nauwkeurig gekend is, terwijl deze parameter door de gebruiker zelf gemeten moet worden, en dus ook een grotere foutmarge bezit. Het zal dan ook belangrijk zijn om deze waarde telkens zo nauwkeurig mogelijk te bepalen.

4.2.3 Gemeten tijdsinterval Δt

Het projectiel legt de afstand k tussen beide meetbasissen af in een bepaalde tijd Δt . Natuurlijk zal ook op deze gemeten waarde een fout zitten, die we δt zullen noemen. Deze Δt wordt door de FPGA gemeten, en deze samplet aan 40 MHz. We verwachten dus een fout van maximaal $1/(40 \text{ MHz}) = 0,025 \mu\text{s}$ ten gevolge van de beperkte sampling frequentie. Omdat we twee keer een meting doen is de fout dus $0,05 \mu\text{s}$. We moeten er natuurlijk ook mee rekening houden dat er een vertraging op de sensoren kan zitten, of op het netwerk waarover de data verstuurd worden, maar doordat we hier een Δt meten kunnen we ervan uitgaan dat deze fouten elkaar telkens zullen opheffen. Experimenten met de FPGA tonen aan dat de te verwachten fout in de praktijk eerder (maximaal) rond de $0,4 \mu\text{s}$ ligt in plaats van rond de $0,025 \mu\text{s}$. Dit experiment bestond erin om verschillende tijdsintervallen te meten met zowel de FPGA als de chronometers in parallel. De resultaten van dit experiment zijn hieronder weergegeven Omdat we weten dat de chronometers zéér nauwkeurig zijn en hun fout te verwaarlozen valt, kunnen we gemakkelijk het verschil in meting tussen FPGA en chronometer bepalen. Dit lag bijna altijd in de orde van $0,1 \mu\text{s}$.

De geschatte tijdsvertraging is gelijk aan $\frac{x}{\frac{k}{\Delta t + \delta t}}$. Vullen we in dat $x = 1 \text{ m}$; $k = 2 \text{ m}$; $\Delta t = 0,001 \text{ s}$; $\delta t = 0,4 \mu\text{s}$

Dan vinden we dat de geschatte tijdsvertraging $\frac{1}{\frac{2}{0,001\text{s} + 0,0000004\text{s}}} = 0,0005002 \text{ s}$ bedraagt, terwijl we een ideale tijdsvertraging van $0,0005\text{s}$ berekend hebben. Dit geeft ons dus een fout van $2 \mu\text{s}$ of van $0,4 \%$.

4.2.4 Resultaten verificatie FPGA klok

Om de nauwkeurigheid van de FPGA klok te bepalen lieten we deze een bepaald tijdsinterval opmeten, terwijl een geijkte chronometer hetzelfde interval ook opmat. De meetnauwkeurigheid die de chronometer heeft wordt nul gesteld omdat het hier een geijkt toestel betreft. Hieronder zijn in Tabel 1 de resultaten weergegeven.

Tabel 1: ijking FPGA klok

meting	tijd FPGA(klokpulsen)	tijd FPGA(ms)	correcte tijd(ms)	afwijking(ms)
1	100012	2,5003000	2,5003798	0,0000798
2	100012	2,5003000	2,5004297	0,0001297
3	100017	2,5004250	2,5005201	0,0000951
4	99977	2,4994250	2,4999947	0,0005697
5	100022	2,5005500	2,5006702	0,0001202
6	100022	2,5005500	2,5006498	0,0000998
7	100027	2,5006750	2,5007798	0,0001048
8	100032	2,5008000	2,5008379	0,0000379
9	100027	2,5006750	2,5007855	0,0001105
10	99997	2,4999250	2,5000583	0,0001333
11	99992	2,4998000	2,4999070	0,0001070
12	100002	2,5000500	2,5001863	0,0001363
13	99992	2,4998000	2,4999070	0,0001070
14	100022	2,5005500	2,5006209	0,0000709
15	100012	2,5003000	2,5002861	-0,0000139
16	100022	2,5005500	2,3454288	-0,1551212
17	100002	2,5000500	2,5001905	0,0001405
18	100007	2,5001750	2,5003561	0,0001811
19	99992	2,4998000	2,4999130	0,0001130
20	100007	2,5001750	2,5002674	0,0000924
21	100022	2,5005500	2,5006816	0,0001316
22	100022	2,5005500	2,5005598	0,0000098
23	99997	2,4999250	2,5000648	0,0001398
24	100027	2,5006750	2,5007639	0,0000889
25	100012	2,5003000	2,5003947	0,0000947
26	100002	2,5000500	2,5001941	0,0001441
27	100022	2,5005500	2,5006363	0,0000863
28	100017	2,5004250	2,5005805	0,0001555
29	100027	2,5006750	2,5008645	0,0001895
30	100007	2,5001750	2,5003233	0,0001483
gemiddelde	100011,6667	2,500291667	2,49524108	-0,005050587
standaardafwijking	13,28742095	0,000332186	0,027820947	0,0000948

In tabel A zien we in de eerste kolom het nummer van de meting. In totaal zijn er dertig metingen uitgevoerd teneinde een voldoende grote steekproef te bekomen. In de tweede kolom is de tijd genoteerd die de FPGA heeft opgemeten, uitgedrukt in klokpulsen, met een frequentie van 40 MHz. In de derde kolom is deze tijd omgerekend naar milliseconden. In de vierde kolom staat de tijd genoteerd die is opgemeten door de geijkte chronometer. In de laatste kolom ten slotte is het verschil berekend tussen de vierde en vijfde kolom: de fout die de FPGA gemaakt heeft ten opzicht van de geijkte chronometer.

Als tijdsinterval is 2500 microseconden genomen omdat dit een waarde is die we ook kunnen verwachten wanneer de FPGA de snelheid van een projectiel meet over een interval van 2,229 meter.

Een eerste opmerking is dat meting 16 niet consequent is met de andere resultaten en daarom niet meegenomen is in de berekeningen van gemiddelde en standaardafwijking. Waarschijnlijk is er een fout gebeurd bij het opmeten van het tijdsinterval door de chronometer.

Met behulp van Excel zijn dan vervolgens het gemiddelde en de standaardafwijking berekend. Alleen de statistische gegevens van de fout zijn relevant. We zien dat de FPGA gemiddeld ongeveer 0,12 microseconden meer heeft gemeten dan de chronometer. De standaardafwijking hierop bedraagt ongeveer 0,01 microseconden. We kunnen dan een 6 sigma interval berekenen dat 99,97% van alle metingen omvat. Dit interval is dan gelijk aan

$[gemiddelde - 3 * standaardafwijking ; gemiddelde + 3 * standaardafwijking]$.

Ingevuld is het interval dan $[-0,16 \mu s ; 0,4 \mu s]$. Dit zijn dus de waardes waartussen statistisch 99,97% van alle metingen zullen vallen. We zien ook dat ons interval nul bevat, wat erop wijst dat de FPGA geen consequente standfout heeft ten opzichte van het referentietoestel.

In de praktijk zal de FPGA dus maximaal een halve microseconde meer of minder meten dan de correcte tijd. Bij een projectiel dat een snelheid van 1000 m/s heeft betekent dit een fout van maximaal een halve millimeter.

4.2.5 Hoek tussen de baan van het projectiel en het lichtscherm α

De baan van het projectiel zou in theorie loodrecht door het lichtscherm moeten gaan, maar in de praktijk is het altijd mogelijk dat het lichtscherm of het vuurwapen lichtjes schuin staan en deze hierdoor niet meer loodrecht op elkaar staan. De hoek tussen de baan en het lichtscherm noemen we α , en in het ideale geval geldt dat $\alpha = 90^\circ$. De afstand k die het projectiel aflegt tussen beide meetbasissen zal in het niet ideale geval steeds meer dan de gekalibreerde k -waarde bedragen.

Indien de hoek α geen 90° bedraagt, is dit alsof de k -waarde met een afstand Δk langer geworden is.

De afstand die het projectiel dus in realiteit aflegt is dan $k + \Delta k$. Deze afstand kan uitgedrukt worden in functie van de hoek α : $k + \Delta k = \frac{k}{\cos(\alpha - 90^\circ)} = \frac{k}{\sin(\alpha)}$ en dus $\Delta k = \frac{k}{\sin(\alpha)} - k$

Als deze gegevens vervolgens ingevuld worden in de formule om de tijdsvertraging te berekenen samen met volgende waarden: $x = 1$ m; $k = 2$ m; $\Delta t = 0,002$ s; $\alpha = 91^\circ$, bekomen we onderstaand resultaat:

$$\text{Tijdsvertraging} = \frac{\frac{x}{k}}{\frac{\sin(\alpha)}{\Delta t}} = \frac{\frac{1\text{m}}{2}}{\frac{0,9998}{0,002}} = 0,000998 \text{ s}$$
 terwijl we in ideale omstandigheden een

tijdsvertraging van 0,001 s zouden hebben. Dit geeft dus een fout van $2 \mu\text{s}$ of van $\frac{0.000002}{0.001} = 0,2 \%$.

4.2.6 Hoek tussen baan van het projectiel en optische as van de camera β

In ideale omstandigheden zou de optische as van de lens van de camera loodrecht op de baan van het projectiel moeten staan. In de praktijk kan het natuurlijk voorvallen dat de lens van de camera niet exact uitgelijnd is met de rest van de meetopstelling en deze hoek β niet exact 90° bedraagt. De fout die hierdoor geïnduceerd zal worden is ook nog afhankelijk van een tweede parameter, namelijk de afstand tussen de lens van de camera en de baan van het projectiel. Deze afstand kan op zich geen fout veroorzaken in het meetsysteem, maar heeft dus wel een invloed indien de camera niet correct opgesteld staat. Deze afstand noemen we c .

De fout Δx die we maken is dan gelijk aan $\Delta x = c \tan(\beta - 90^\circ)$.

De afstand Δx is equivalent met een virtuele verplaatsing van onze camera volgens de baan van het projectiel. Afhankelijk van de richting van de fout zal de afstand x dus toe- of afnemen met een afstand Δx .

Indien we nu weer de formule voor de tijdsvertraging nemen en hierin volgende waarden invullen: $x = 1\text{m}$; $k = 2\text{m}$; $\Delta t = 0,002\text{s}$; $\beta = 88^\circ$; $c = 3\text{m}$ dan bekomen we:

Tijdsvertraging = $\frac{x+\Delta x}{\frac{k}{\Delta t}} = \frac{x+c \tan(\beta-90^\circ)}{\frac{k}{\Delta t}} = \frac{1+3 \tan(88^\circ-90^\circ)}{\frac{2}{0,002}} = 0,000895 \text{ s}$ terwijl de ideale tijdsvertraging $0,001\text{s}$ bedraagt. Dit betekent dat er een afwijking is van $105 \mu\text{s}$ ten opzichte van het reële resultaat. In percent uitgedrukt geeft dit een fout van $\frac{0,000105}{0,001} = 10,5 \%$

Er is nog een ander probleem indien de camera schuin opgesteld staat behalve de meetonnauwkeurigheid: Het projectiel gaat lichtjes vervormd op de foto staan waardoor het moeilijker wordt om data uit deze foto te halen (zoals de lengte van het projectiel). Daarom is het des te meer belangrijk dat de camera loodrecht ten opzichte van de baan van het projectiel opgesteld wordt.

4.2.7 Versnelling

Een projectiel is natuurlijk ook onderhevig aan een vertraging eenmaal het de loop verlaten heeft omwille van de luchtweerstand. Zoals al eerder aangehaald meet de meetbasis de gemiddelde snelheid over een interval, en niet de ogenblikkelijke snelheid.. Tussen de meetbasis en de fotocamera gaat het projectiel nog verder vertragen en dus in realiteit er langer over doen dan als berekend is. Anders gesteld kunnen we zeggen dat het projectiel nog niet zo ver is als berekend. Als we aannemen dat het projectiel de gemeten snelheid bezit die de meetbasis meet in het midden van de basis¹ dan kan de fout die hierdoor veroorzaakt wordt op de afgelegde weg tussen de meetbasis en de camera berekend worden.

Noem v de snelheid in het midden van de meetbasis; y de afstand tussen het midden van de meetbasis en de optische as van de camera, a de versnelling, en Δt de tijdsvertraging;

In dit vereenvoudigde geval is de afgelegde weg $= \Delta t \cdot v$

In het meer realistische geval is de afgelegde weg gelijk aan $v \cdot \Delta t + 0,5 a (\Delta t)^2$. De fout is dus gelijk aan $0,5 a (\Delta t)^2$.

Stel nu dat $v=1000\text{m/s}$; $y=2\text{m}$; $a= -800\text{m/s}^2$; $\Delta t = 0,002\text{s}$. Dan vinden we dat in het vereenvoudigde geval de afgelegde weg gelijk is aan 2m , wat ook de afstand is. In het realistische geval vinden we dat afgelegde weg gelijk is aan $2\text{m} - 0,0016\text{m} = 1,9984\text{m}$ oftewel een fout van $1,6\text{mm}$. Let ook op de grote versnelling van een projectiel, deze waarde is ongeveer de versnelling van een $5.56 \times 45\text{mm}$ NATO projectiel nadat het de loop verlaten heeft. Door de korte tijd tussen het meetinterval en de fotocamera heeft deze factor gelukkig niet zo'n grote invloed op het globale systeem.

Voor bepaalde soorten projectielen die minder aerodynamisch zijn vormgegeven en een lagere startsnelheid hebben kan de versnelling van het projectiel tijdens zijn vlucht wel een grote invloed hebben op het uiteindelijke resultaat [4]. Daarom is het mogelijk om in de FPGA code de versnelling in te geven waarna het programma hiervoor zal compenseren. De specifieke uitwerking hier van is te vinden in paragraaf 8.3.2.

¹ Dit is niet volledig correct omdat de versnelling niet helemaal lineair is.

4.3 Conclusie

De fouten veroorzaakt door de vertraging van het projectiel of door het niet nauwkeurig afregelen van de hoek β kunnen allebei zeer groot worden en ervoor zorgen dat het uiteindelijke resultaat te wensen overlaat. Daarom zijn deze twee parameters de belangrijkste aandachtspunten bij het opstellen van de meetopstelling. Aan de versnelling van het projectiel kan natuurlijk niks gedaan worden, maar aan de hoek β wel. Dit is ook zo hernomen in de handleiding die terug te vinden is in bijlage A.

De fouten veroorzaakt door de overige parameters zijn in normale omstandigheden van kleinere orde en zouden het uiteindelijke resultaat slechts minimaal mogen beïnvloeden als er geen excessieve fouten gemaakt zijn in de opstelling van alle apparatuur.

5 Opzetten van de meetapparatuur

In de vorige paragrafen zijn alle fouten die kunnen optreden ten gevolge van een niet ideale opstelling besproken. In deze paragraaf volgt een methode die toelaat om de meetbasis en camera zo op te stellen dat de fout minimaal is.

5.1 De meetbasis

De meetbasis moet opgesteld worden in de schietunnel, tussen de loop van het wapen en de camera. Er moet een minimumafstand tussen de basis en de camera van ongeveer 40 cm in acht genomen worden omdat de FPGA tijd nodig heeft om zijn berekeningen uit te voeren. De ideale afstand is ongeveer 1 meter. De minimumafstand hangt af van de snelheid en versnelling van het projectiel en kan ook exact berekend worden: de methode hiervoor wordt besproken in sectie 6.3.5.

De meetbasis moet parallel met de baan van het projectiel opgesteld worden, zowel in de lengte- als in de hoogterichting. Hiervoor kunnen de 4 schroeven gebruikt worden die vastgrijpen in de rail in de vloer zoals weergegeven in Figuur 19. Ook de loop van het wapen en het doel moeten dan op de rail opgelijnd zijn, zodat de baan van het projectiel (ongeveer) de rail in de tunnel volgt. Dit gebeurt meestal met behulp van een laser die op de loop gemonteerd is.



Figuur 19: correcte opstelling van de meetbasis

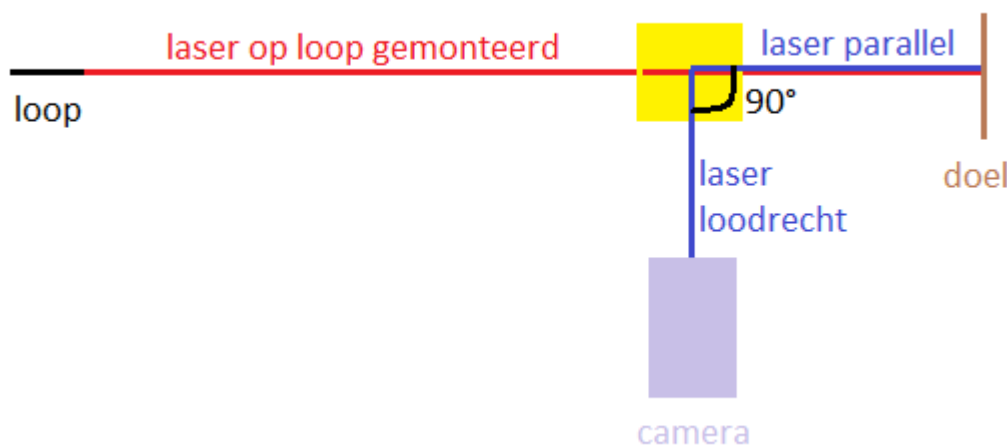
5.2 De camera

De opstelling van de camera is iets complexer dan die van de meetbasis. Het is namelijk belangrijk dat de hoek β tussen de optische as van de camera en de baan van het projectiel niet teveel afwijking vertoont, omdat deze een grote fout zou induceren in het uiteindelijke resultaat. De camera kan zowat overal geplaatst worden tussen de meetbasis en de doelenzone, maar hoe dichterbij de meetbasis hoe groter de nauwkeurigheid (dit zolang de afstand niet kleiner wordt dan de minimumafstand).

De afstand c van de lens tot de baan van het projectiel is vrij te kiezen in functie van de zoom van de lens en het gewenste resultaat. De hoogte van de lens is best dezelfde als de hoogte van de baan van het projectiel (deze kan zichtbaar gemaakt worden met behulp van een laser op de loop van het wapen) om zo de sferische vertekening te verkleinen. Eventueel kan de hoogte van de loop gemeten worden en de hoogte van de lens zo aangepast worden dat deze zich op dezelfde hoogte als de loop bevindt.

Om de hoek tussen de optische as en de baan van het projectiel nauwkeurig af te regelen op 90° , kan er gebruik gemaakt worden van een toestel dat uit twee lasers bestaat die onder een hoek van 90° gemonteerd zijn. Een schets van hoe deze lasers te oriënteren is weergegeven in Figuur 20.

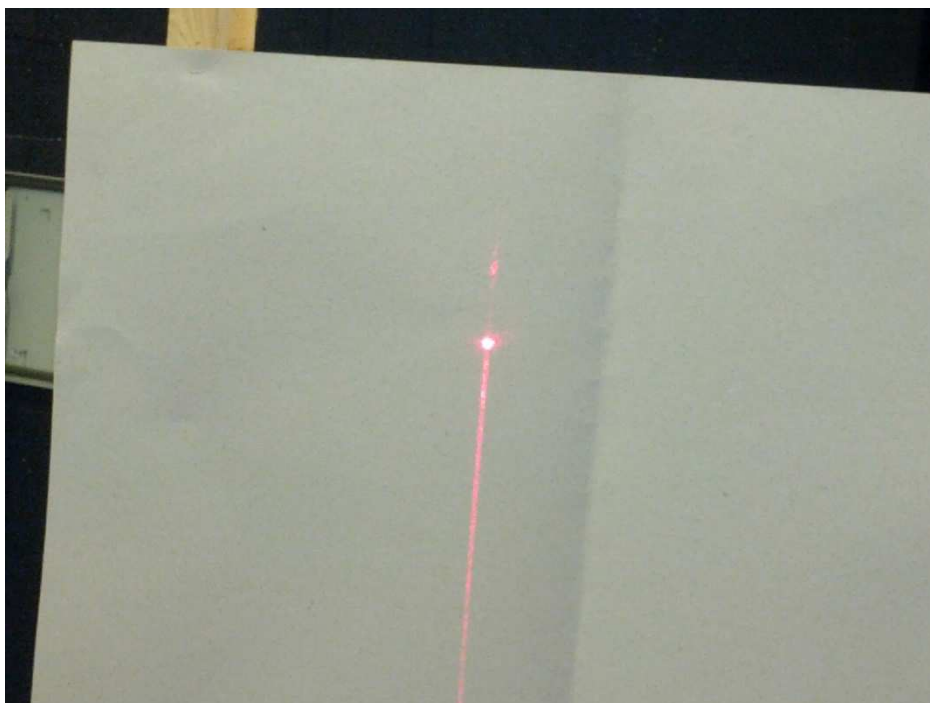
Eén laser kan dan opgelijnd worden met de laserstraal van de loop: links op Figuur 21 is te zien hoe de laser die op de loop van het wapen gemonteerd staat invalt op het gele apparaat. Eén van beide lasers ligt dan in het verlengde van de baan van het projectiel, terwijl de andere hier loodrecht opstaat. De laser die in het verlengde ligt van de laser die op de loop gemonteerd staat, moet dus ook verderop samenvallen met deze laatste zoals zichtbaar op Figuur 22. Vervolgens kunnen we de camera verschuiven totdat de laser die loodrecht op de baan van het projectiel staat midden op de lens invalt. Dit is weergegeven op Figuur 23.



Figuur 20: schets van opstelling met toestel (geel) dat twee onderling loodrechte lasers bevat, om de hoek tussen optische as van de camera en de kogelbaan correct in te stellen



Figuur 21: apparaat met twee loodrechte laserstralen



Figuur 22: doelenzone met twee samenvallende lasers



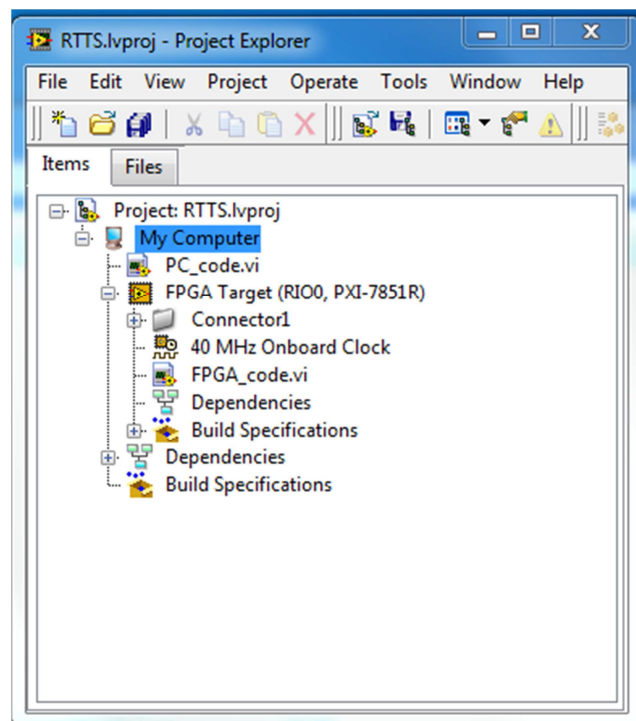
Figuur 23: de Photron camera met de opgelijnde laserlijn zichtbaar

We zien op Figuur 23 ook dat de laser opgelijnd is met de twee schroefjes boven en onder de lens. Op deze manier is de optische as van de camera opgelijnd met de laser en dus staat ze loodrecht op de baan van het projectiel. De afstand x ten slotte wordt gemeten met behulp van een rolmeter of laser afstandsmeter.

6 Werking van het programma

In dit hoofdstuk wordt de werking van het programma in zijn geheel besproken, en wordt er op sommige complexe of belangrijkere onderdelen dieper ingegaan.

Het programma bestaat uit twee VI's (programma's in LabVIEW) die op twee verschillende niveaus werken: ten eerste is er een VI die uitgevoerd wordt door de FPGA. Deze code wordt heel snel en nauwkeurig uitgevoerd, maar de berekeningen hier zijn zeer omslachtig en moeten beperkt gehouden worden in omvang. Een tweede VI draait op de pc waarmee de FPGA verbonden is: deze code wordt trager uitgevoerd, maar is eenvoudiger te schrijven en hier is de complexiteit van de code niet belangrijk omdat deze processor toch over immens veel rekenkracht beschikt. Deze structuur is ook duidelijk zichtbaar in Figuur 24, waar enerzijds een VI zichtbaar is op pc niveau, en anderzijds een VI op FPGA niveau.

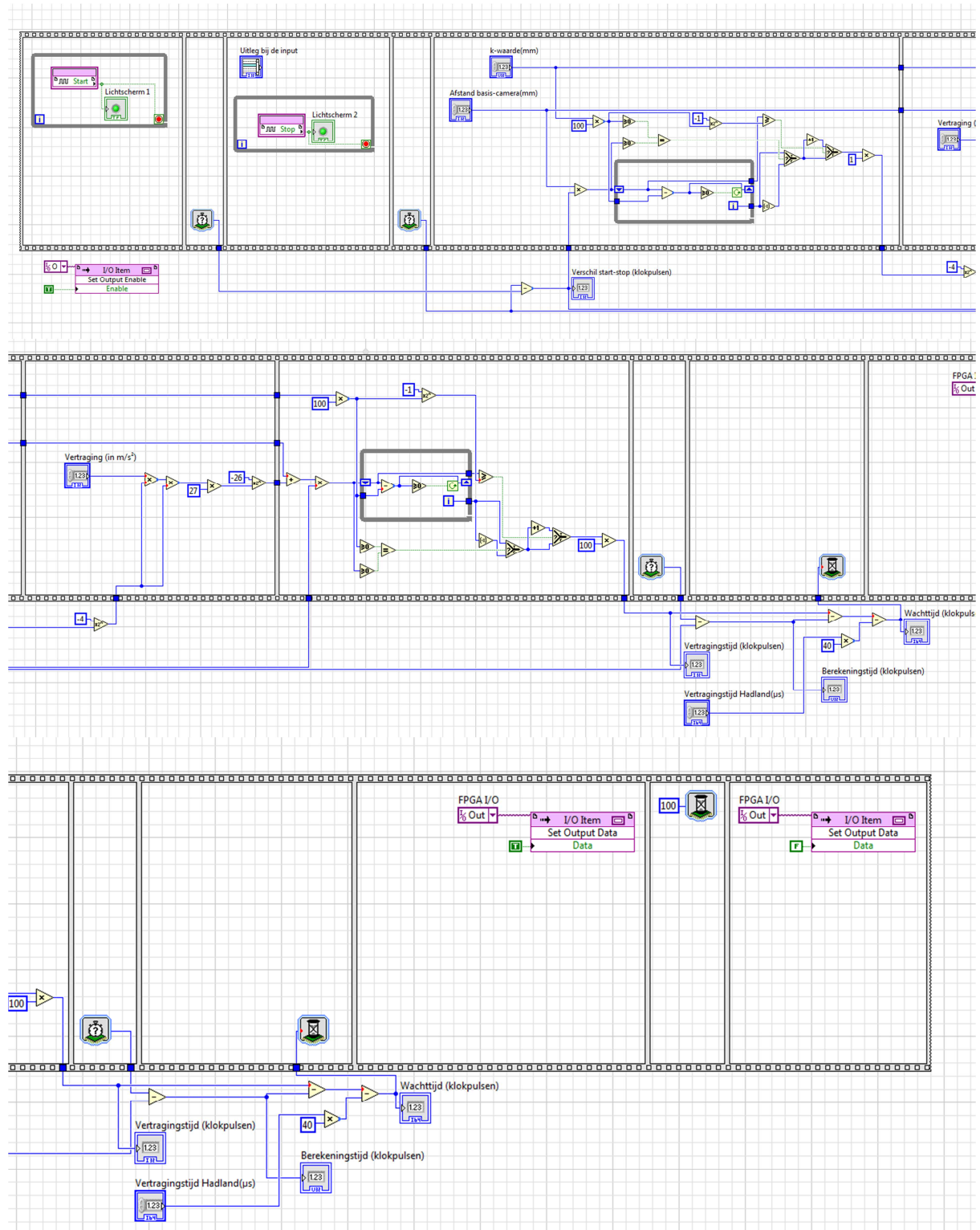


Figuur 24: programmastructuur LabVIEW project met één VI op pc niveau, en één VI op FPGA niveau.

Het spreekt dus voor zich dat de berekeningen die tijdens de vlucht van het projectiel uitgevoerd moeten worden door de FPGA afgehandeld worden, terwijl alle andere parameters door de pc berekend zullen worden.

6.1 FPGA Code

In Figuur 25 is de volledige code die op de FPGA draait weergegeven. Om de zichtbaarheid te verhogen is de code in drie delen verdeeld en zijn deze onder elkaar weergegeven.



Figuur 25: Visuele voorstelling van de FPGA code

Eerst wat uitleg over de code die door de FPGA wordt uitgevoerd: zoals al eerder aangehaald, wacht het programma op een eerste puls van een lichtbasis. Vervolgens meet de FPGA het tijdsverschil tussen deze puls en de tweede puls van de andere lichtbasis. Daarna berekent het de tijdsvertraging die het projectiel nodig heeft om de afstand tot aan de camera af te leggen. Ten slotte geeft het net op het juiste moment een puls aan de camera zodat deze een foto maakt van het projectiel. In de volgende alinea's wordt de werking stap per stap uitgelegd.

We zien dat het programma is opgebouwd in de vorm van een fotorolletje, zoals bij een analoog fototoestel. Dit is een sequentie waardoor LabVIEW gedwongen wordt om alles in de juiste volgorde af te werken, en pas naar de volgende stap te springen als de vorige stap al volledig uitgevoerd is. Als het programma bijvoorbeeld niet zou wachten tot het de tweede puls ontvangt, zou de timing helemaal mislopen en zou het projectiel nooit correct gefotografeerd worden.

In het eerste frame wacht LabVIEW op de eerste digitale ingangspuls. Vervolgens wordt de tweede sequentie uitgevoerd, waar een timer start. Deze timer telt in klokpulsen (1 klokpuls = 25 ns). De kloksnelheid staat vast ingesteld op 40Mhz. Onmiddellijk hierna begint stap 3, waar de FPGA wacht op de digitale puls van het tweede lichtscherf.

Van zodra deze puls ontvangen is, begint de uitvoering van stap 4 waar een tweede (andere) timer gestart wordt die eveneens in klokpulsen telt. Het verschil tussen deze twee timers (in klokpulsen) wordt dan berekend en deze waarde is de tijd die het projectiel nodig had om de afstand tussen lichtscherf 1 en lichtscherf 2 af te leggen.

In de 5^e stap wordt de daadwerkelijke wiskundige berekening uitgevoerd. Dit is een deling van twee getallen. Voor een meer gedetailleerde wiskundige kijk op dit probleem zijn er de twee volgende paragrafen waar de methode uitgebreid beschreven wordt die hier toegepast wordt.

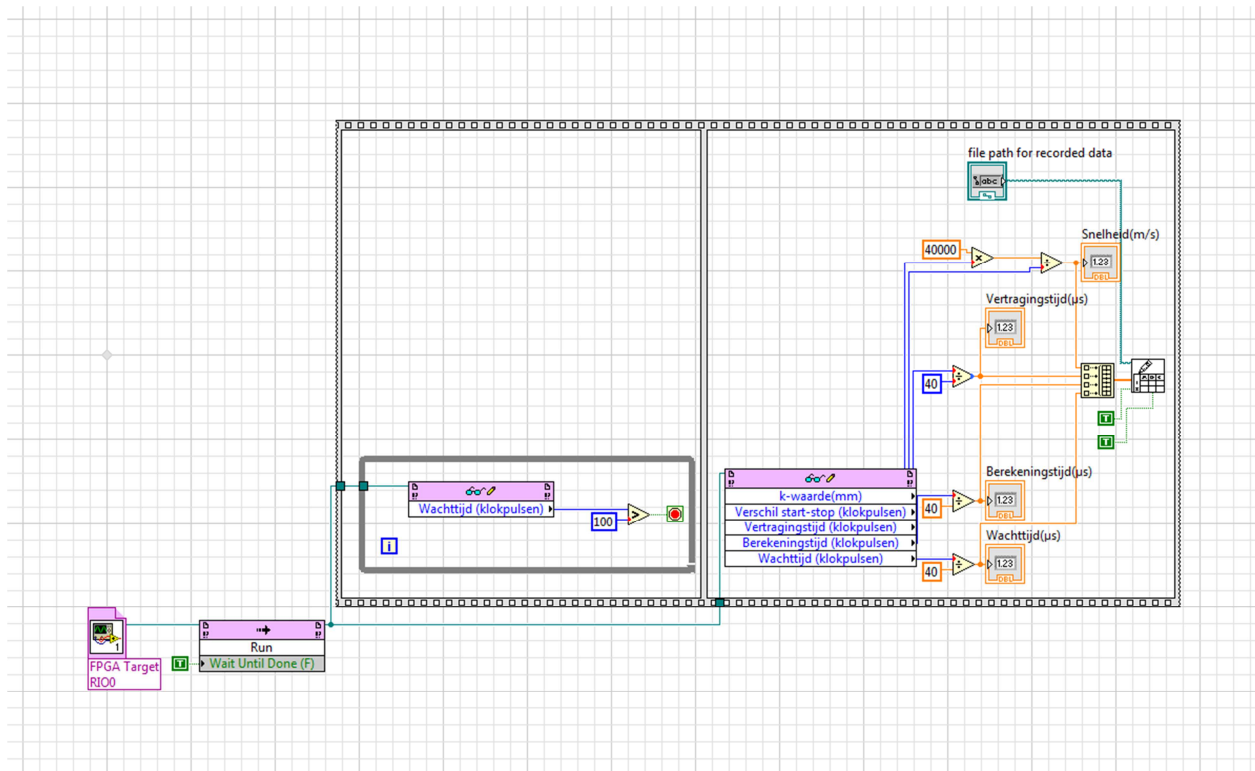
In het zesde frame wordt er een derde timer gestart (in klokpulsen). Onmiddellijk hierna wordt het verschil tussen timer 2 en timer 3 bepaald. Dit is de tijd die de module heeft nodig gehad om bovenstaande deling uit te voeren. Deze tijd trekken we dan af van ons resultaat omdat we eigenlijk al een tijdje gewacht hebben omdat de FPGA nog aan het rekenen was. Ook de eventuele preflash tijd trekken we af van dit resultaat.

Uiteindelijk bekomen we dan onze totale vertragingstijd (in klokpulsen) en we laten de module dan natuurlijk ook even lang wachten, dit is de zevende stap van het programma.

Na deze wachttijd springt het programma naar het achtste frame, waar hij de digitale uitgang hoog maakt zodat de camera een foto neemt. Vervolgens wacht hij 100 μ s (frame 9) en maakt ten slotte de uitgang weer terug laag (frame 10). Dit is het einde van het programma.

6.2 PC Code

Het tweede deel van de code bestaat uit een VI die op de pc ingebouwd in de PXI draait. Een overzicht van deze code is gegeven in Figuur 26.



Figuur 26: overzicht pc code

Linksonder op Figuur 26 is te zien hoe deze VI wacht tot de code op de FPGA volledig uitgevoerd is alvorens te starten. Omdat de code op de FPGA continu zijn ingangspoorten zit af te scannen (en dit dus niet eenmalig doet) is er in het linkse frame ook een stukje code voorzien dat ervoor zorgt dat het programma op pc alleen uitgevoerd wordt wanneer de wachttijd meer dan 100 klokpulsen bedraagt. In de praktijk zorgt dit ervoor dat de code op pc alleen wordt uitgevoerd wanneer de FPGA ook effectief een berekening heeft uitgevoerd.

In het rechtse frame zit de code die de functionaliteit van deze VI bevat: hier worden een aantal abstracte doch nuttige parameters uit de FPGA ingelezen en vertaald naar parameters die gebruikt kunnen worden voor de verdere verwerking. Zo wordt onder meer de berekenings-, vertraging- en wachttijd omgerekend van klokpulsen naar microseconden. Ook wordt in deze VI de snelheid in m/s berekend, omdat dit niet gebeurt bij de berekeningen van de FPGA.

Ten slotte worden geheel rechts al deze waarden gebundeld in een “array” en automatisch weggeschreven op een plaats in het geheugen van de pc die de gebruiker zelf kiest. Indien er meerdere metingen na elkaar gebeuren wordt de nieuwe data gewoon onder de oude data weggeschreven.

6.3 Gedetailleerde bespreking code

6.3.1 Vertragingstijd van een projectiel met constante snelheid

Het hoofddoel van de FPGA is om het tijdstip te berekenen waaarop het projectiel zich exact voor de camera bevindt. Eerst negeren we de vertraging a . Het projectiel legt dan dus een eenparig rechte lijnige beweging af bij benadering. Er zijn drie parameters die nodig zijn om de vertragingstijd in dat geval te berekenen: de afstand tussen de basis en de camera (x), de afstand tussen de basissen (k), en de tijd die het projectiel nodig heeft om de afstand tussen beide basissen af te leggen (t).

Met bovenstaande gegevens is het mogelijk om de snelheid van het projectiel te berekenen, en vervolgens de tijd die het projectiel nodig heeft om dan de afstand basis-camera af te leggen rekening houdend met zijn snelheid. Deze methode is het meest intuïtief maar niet ideaal omdat er dan twee delingen noodzakelijk zijn om ons resultaat te bekomen, en een deling is complex en traag om uit te voeren met een FPGA en dit willen we dus liefst zoveel mogelijk vermijden.

Een andere aanpak is om de verhouding tussen de x en k te beschouwen. Stel bijvoorbeeld dat $x = 2k$. Dan is het onmiddellijk duidelijk dat de FPGA tweemaal zolang moet wachten als de tijd die het projectiel nodig had om de afstand k af te leggen, omdat x nu eenmaal dubbel zo lang is als k . Meestal gaat de verhouding x/k natuurlijk geen natuurlijk getal zijn, maar op deze manier kan de vertragingstijd wel berekend worden met behulp van één deling in plaats van twee. De formule voor de vertragingstijd is dan $\frac{t * x}{k}$.

6.3.2 Vertragingstijd van een projectiel onderhevig aan een versnelling

Als we nu wel rekening houden met de vertraging a (= een negatieve versnelling) dan geldt niet meer dat de tijdsvertraging gelijk is aan $\frac{t*x}{k}$. Om een betere waarde voor de tijdsvertraging te vinden moeten we eerst de vertraging van het projectiel kennen. Omdat we over slechts twee infraroodlichtschermen beschikken, is het niet mogelijk om de vertraging in realtime te meten. Daarom moet de gebruiker deze zelf op voorhand ingeven zodat de FPGA hier rekening mee kan houden in zijn berekening. We stellen ook bij benadering dat de vertraging constant is en niet verandert in de tijd.

De invloed van de vertraging op de afgelegde weg is afhankelijk van de tijd die het projectiel nodig heeft om deze weg af te leggen. Omdat we deze tijd pas kennen nadat we de snelheid hebben gemeten in realtime, is het ook pas achteraf mogelijk om de versnelling in rekening te brengen. Daarom berekent de FPGA eerst de vertragingstijd zonder rekening te houden met de vertraging, en berekent hij dan achteraf de invloed van de vertraging, rekening houdend met de reeds berekende vertragingstijd.

Zoals reeds aangehaald berekent de FPGA dus eerst de vertragingstijd alsof $a = 0 \text{ m/s}^2$.

Vervolgens berekent hij de invloed van de versnelling op de afgelegde weg volgens de formule

$$x = a * \frac{t^2}{2}.$$

De afstand die we dan vinden zal steeds negatief zijn, omdat een projectiel altijd vertraagt nadat het de loop heeft verlaten. Deze afstand is dus als het ware een extra afstand die het projectiel (virtueel) moet afleggen ten gevolge van zijn vertraging voor het op het gewenste punt zal aankomen. Daarom dat we de absolute waarde van deze extra afstand optellen bij de afstand tussen de basis en camera. Vervolgens voeren we opnieuw de berekening uit dat de vertragingstijd gelijk is aan $\frac{t*x}{k}$, met als enige verschil dat nu x gelijk is aan de afstand basis camera plus de virtuele extra afstand ten gevolge van de vertraging. Uiteindelijk bekomen we dan een tweede, meer correcte, vertragingstijd die steeds groter of gelijk is aan de eerst berekende vertragingstijd. Het zou nauwkeuriger zijn om de tweedegraadsvergelijking rechtstreeks op te lossen, maar de extra tijd die hiervoor nodig is weegt niet op tegen de extra nauwkeurigheid.

6.3.3 Verschillende soorten delingen bij een FPGA

Om de vertragingstijd te berekenen zullen we één of meerdere delingen moeten uitvoeren.

Omdat een FPGA standaard met integers werkt, kan deze geen gewone deling uitvoeren zoals we die in de wiskunde kennen. Er zijn drie manieren waarop een FPGA wel een deling kan uitvoeren en alle drie hebben ze hun voor- en nadelen.

Het eerste soort deling kan uitgevoerd worden door een library te downloaden van de site van National Instruments [15]. Hierdoor emuleert de FPGA een deling van floating point getallen door complexe berekeningen met integers uit te voeren. Het voordeel van deze methode is de nauwkeurigheid en de eenvoudigheid om te implementeren. Het grote nadeel is de snelheid: om een grote nauwkeurigheid te halen en door de conversie zal het zeer lang duren om een deling uit te voeren. Daarom raadt National Instruments deze conversie alleen aan voor programma's waar de timing niet belangrijk is [16]. Dit type van deling is dus zeker niet geschikt voor deze toepassing.

Het tweede soort deling is een scalering met factor 2^x waarbij x een geheel getal is. Deze methode kan aangevuld worden met een vermenigvuldiging met een factor y , waarbij y een natuurlijk getal is. Op hardware niveau is een vermenigvuldiging niet zo heel complex om uit te voeren, omdat het eigenlijk gewoon y maal een optelling is die de FPGA moet uitvoeren. Het scaleren met een factor 2 wordt op hardware niveau vertaald door een shift left of shift right operatie en kan dus zeer snel uitgevoerd worden. Het grote voordeel van deze methode is dus de snelheid maar hier tegenover staat wel dat de nauwkeurigheid zeer slecht is omdat $y \cdot 2^x$ vaak een paar % afwijkt van de gewenste schaafactor. Er bestaat een keuzehulp voor y en x die vrij te downloaden is van de website van National Instruments [17]. Deze methode werd gebruikt om de extra tijdsvertraging ten gevolge van de vertraging van het projectiel om te zetten van microseconden naar klokpulsen.

Het derde soort deling is de 'while loop subtraction'. Deze methode voert een deling uit met integers door telkens de noemer af te trekken van de teller en bij te houden hoe vaak dit gebeurd is. Eenmaal de uitkomst kleiner of gelijk aan 0 is, stopt de lus en krijgen we het correcte, afgeronde resultaat. Het is ook mogelijk om indien nodig naar boven af te ronden door de helft van de teller te vergelijken met de rest. De code om deze deling uit te voeren is vrij te downloaden op de site van National instruments [18] en het is ook deze code die (weliswaar aangepast) gebruikt wordt om tweemaal de deling uit te voeren in de FPGA code. Om een maximale nauwkeurigheid te halen werken we natuurlijk met kleine eenheden zoals klokpulsen voor de tijd en millimeters voor de afstand.

Het voordeel van de 'while loop subtraction' is dat deze methode de snelheid van een scalering benadert en toch niet moet onderdoen qua nauwkeurigheid voor een floating point deling. Het nadeel is dat de argumenten die aangehaald werden bij de voordelen alleen maar gelden als de teller en noemer zich correct verhouden. Een verdere uitwerking van de invloed van deze verhouding is terug te vinden in de volgende paragraaf.

6.3.4 Verhouding teller/noemer bij while loop subtraction

Zoals reeds aangehaald in paragraaf 8.3.3 is een nadeel van while loop subtraction dat de verhouding van teller tot noemer ongeveer gekend en constant moet zijn. De reden hiervoor wordt in de volgende alinea's toegelicht.

Stel dat de teller kleiner is dan de noemer: een deling met integers zal dan steeds als resultaat 0 geven, terwijl de echte uitkomst bijvoorbeeld 0,85 is. Deze afwijking is onacceptabel voor bijna alle toepassingen en moet dus vermeden worden.

Stel nu bijvoorbeeld dat de teller twee miljoen keer groter is dan de noemer. In dat geval zal de FPGA dus twee miljoen keer de noemer moeten aftrekken van de teller voordat hij nul bereikt. Deze operatie zal dus ook (minstens) 2 miljoen klokpulsen duren. Zo een lange tijdsduur is opnieuw niet geschikt voor de meeste toepassingen, en zeker niet voor toepassingen zoals dit realtime trigger systeem.

Het is dus duidelijk dat de verhouding van teller tot noemer ongeveer constant moet zijn. Bij deze specifieke toepassing is dat het geval: de snelheid van een projectiel zal nooit lager dan 10 m/s zijn en nooit meer dan 1000 m/s. Tussen deze twee uitersten zit slechts een factor 100, waardoor we bijna zeker kunnen zijn dat de verhouding van teller tot noemer nooit absurde waarden zal aannemen.

Nu is het alleen nog een kwestie van de verhouding van teller tot noemer ongeveer goed te krijgen. De afstand tussen de twee lichtschermen en de afstand van de basis tot de camera ligt in dezelfde grootteorde (enkele meters). De afstand van basis tot camera staat in de teller en wordt vermenigvuldigd met de tijd die het projectiel nodig heeft om de afstand tussen beide lichtschermen af te leggen, uitgedrukt in klokpulsen. Deze tijd zal ongeveer schommelen tussen duizend en honderdduizend klokpulsen. Als een gevolg hiervan zal de teller dus ook duizend à honderdduizend keer groter zijn dan de noemer. Dit is teveel en zou ervoor zorgen dat de berekening veel te lang duurt.

Als oplossing voor dit probleem wordt de noemer vermenigvuldigd met een factor 100 voor de while loop subtraction uitgevoerd wordt. De teller zal dan maar 10 à 1000 keer groter zijn dan noemer. De maximale tijd die de FPGA nodig heeft om de deling uit te voeren is dan ook ongeveer 1000 klokpulsen = 25 μ s. Na de deling bekomen we dan een resultaat dat 100 keer te klein is. Daarom wordt dit resultaat ook met een factor 100 vermenigvuldigd zodat wiskundig alles klopt.

Het is ook mogelijk om deze factor aan te passen om zo een nauwkeurigere of snellere berekening uit te voeren. Door de factor bijvoorbeeld te veranderen van 100 naar 1000 zal de while loop subtraction tien keer sneller uitgevoerd worden, terwijl de nauwkeurigheid met een factor 10 daalt. De totale berekeningstijd van de FPGA zal ook sterk dalen, met niet met een factor tien omdat er nog andere stukken code uitgevoerd worden, wiens snelheid onafhankelijk is van de while loop subtraction.

Andersom is het ook mogelijk om deze factor te verkleinen om zo de nauwkeurigheid op te drijven, ten koste van de snelheid.

6.3.5 Berekening minimumafstand

Bij normale toepassingen doet de afstand tussen het tweede lichtscherm en de optische as van de camera er niet toe. Indien het nu belangrijk is om deze afstand zo klein mogelijk te maken omdat we bijvoorbeeld willen dat een projectiel met maximale snelheid een doel raakt, dan kan de minimale x-waarde berekend worden in plaats van uit te gaan van een standaard waarde van 40 centimeter.

De beste methode hiervoor is om eenmaal te vuren met het gewenste projectiel en de snelheid en berekeningstijd van de FPGA te noteren. Deze waarden worden weergegeven op het “front panel” van de pc code, en worden ook automatisch weggeschreven in een tekstbestand. De minimale afstand (in millimeter) kan berekend worden door de berekeningstijd van de FPGA (in microseconden) te vermenigvuldigen met de snelheid van het projectiel (in m/s) en vervolgens dit getal door 1000 te delen. Dit is de wiskundige minimale afstand die de FPGA nodig heeft om zijn berekeningen uit te voeren.

Omdat de snelheid van het volgende projectiel hoger kan liggen dan de snelheid van het projectiel waarmee de berekening gebeurd is, en omdat de FPGA nog enkele microseconden nodig heeft om zijn signaal door te sturen kan de minimale afstand best steeds vermenigvuldigd worden met een factor 1,2. Zo wordt de werking van het RTTS gegarandeerd.

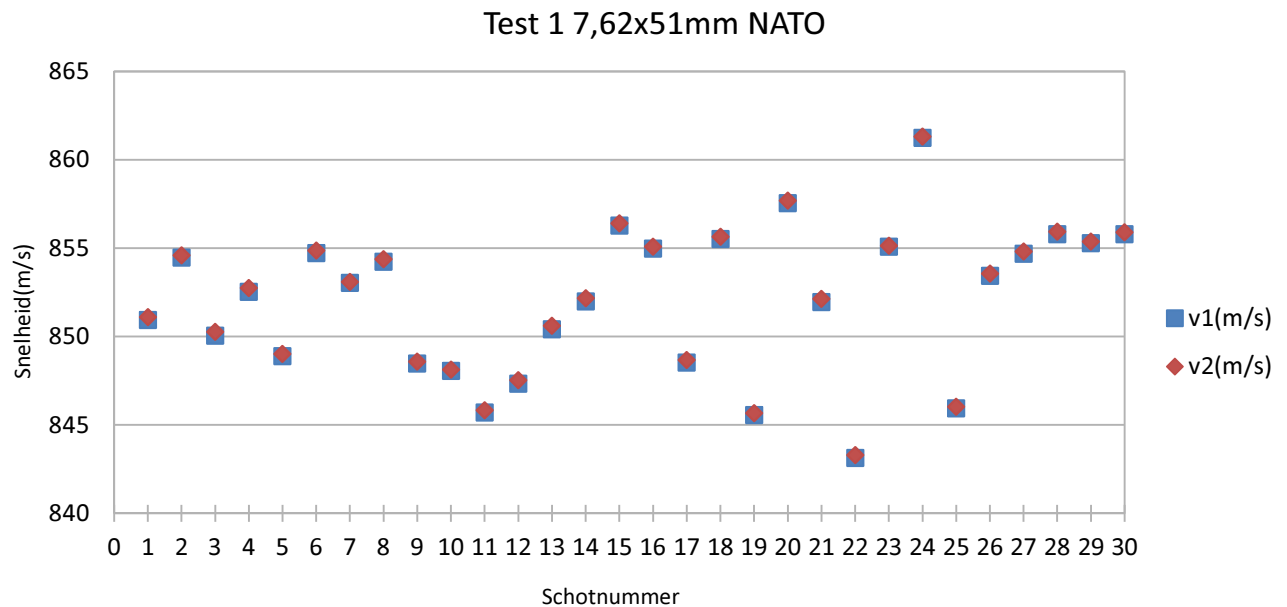
7 Variatie van snelheid bij projectielen

Projectielen die afgevuurd worden met een vuurwapen hebben een bepaalde snelheid als ze de loop verlaten, mondingssnelheid genoemd. Deze snelheid varieert natuurlijk sterk afhankelijk van het type projectiel en vuurwapen. Een 5,56 x 45mm NATO afgevuurd door een aanvalsgeweer heeft bijvoorbeeld een mondingssnelheid die veel hoger ligt dan een 9 x 18mm Parabellum afgevuurd met een pistool [4]. Dit heeft onder meer te maken met de vorm van het projectiel en de lengte van de loop van het gebruikte vuurwapen.

Er is echter ook een grote spreiding in de snelheid van projectielen die onder schijnbaar identieke omstandigheden worden afgevuurd. In onderstaande proef vuren we 30 patronen 7,62 x 51mm NATO af met hetzelfde vuurwapen. Alle 30 patronen zijn afkomstig van dezelfde fabrikant en ook andere omstandigheden zoals luchtdruk, luchtvochtigheid en temperatuur zijn identiek. We verwachten dus dat alle 30 patronen een bepaalde snelheid hebben die zeer dicht bij de snelheid van de andere patronen aanleunt. Maar de resultaten tonen aan dat de variatie op de snelheid vrij groot is. De reden hiervoor is dat deze projectielen aangedreven worden door de ontbranding van een lading kruit in de huls. De manier waarop dit kruit ontbrandt is een factor die een grote rol speelt op de snelheid van het projectiel maar het is bijna onmogelijk om te voorspellen hoe dit kruit exact gaat ontbranden, laat staan 'afregelen' zodat dit ongeveer identiek zou zijn aan de ontbranding van het vorige projectiel.

Dit is ook meteen de grote drijfveer voor deze masterproef, want door bovenstaand fenomeen kunnen we nooit op voorhand zeggen hoe snel een projectiel gaat gaan, en kunnen we dus ook nooit op voorhand bepalen wat de vertragingstijd gaat zijn om onze camera te triggeren.

Hieronder wordt in Figuur 27 de spreiding van 30 patronen 7,62 x 51 mm NATO weergegeven, afgevuurd in identieke omstandigheden. De onbewerkte data van dit experiment worden hernomen in Bijlage F onder de vorm van een tabel.



Figuur 27: spreiding van de snelheid van 7,62 x 51 mm NATO

Tussen het snelste en het traagste projectiel zit er een snelheidsverschil van ongeveer 20 m/s. De standaardafwijking bedraagt ongeveer 4,2 m/s en de gemiddelde snelheid is 852 m/s . Dit betekent dus dat 99,97 % van alle projectielen binnen het interval [839,4 m/s ; 864,6 m/s] vallen. Dit onderlinge snelheidsverschil zou er bijvoorbeeld voor kunnen zorgen dat een projectiel niet zichtbaar is op een foto als de fotocamera ingesteld is met een vaste vertragingstijd. Een simpele doch inefficiënte oplossing hiervoor is gewoon meerdere projectielen afvuren totdat er een projectiel is dat ongeveer de verwachte snelheid heeft en dus ook centraal zichtbaar is op de foto.

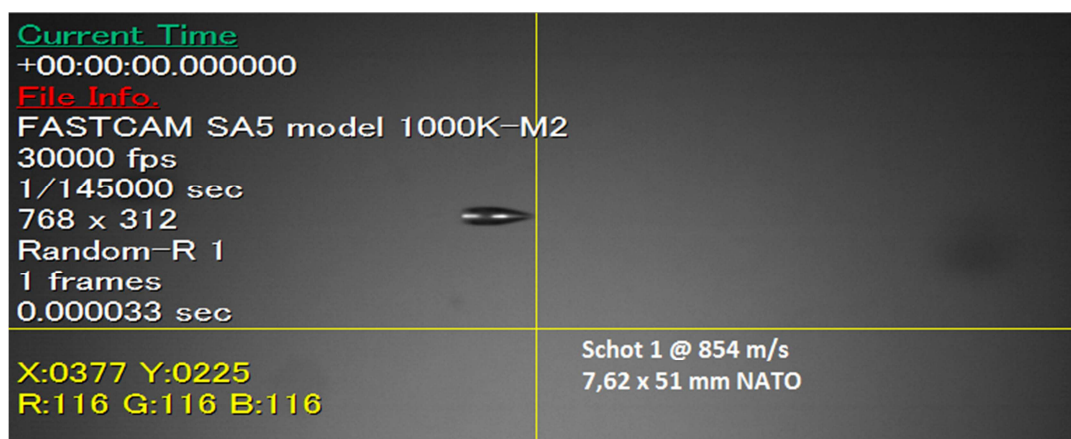
Dit fenomeen van een schijnbaar willekeurige snelheid en dus willekeurige vertragingstijd gaan we dus proberen te elimineren met behulp van een realtime FPGA sturing.

8 Test van vuren schot per schot

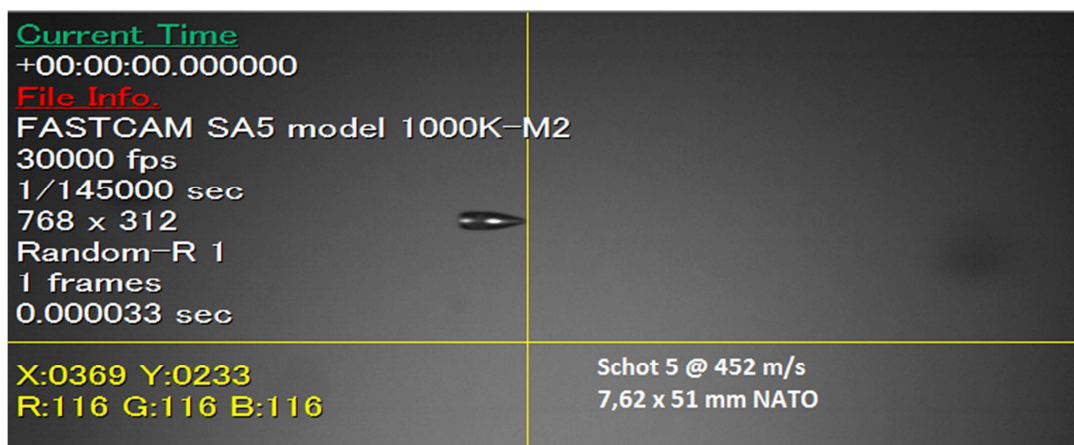
8.1 Resultaten vergelijkende test

Om de werking van het RTTS te testen werden er vijftientwintig patronen afgevuurd met hetzelfde vuurwapen, terwijl de instellingen van het RTTS niet veranderd werden (uitgezonderd de vertraging van het projectiel die per kaliber verschillend is). Er werden verschillende soorten patronen afgevuurd, en bij sommige werd de kruitlading veranderd waardoor de snelheid ook sterk veranderde, terwijl de vorm identiek bleef.

Op alle 25 foto's is dan achteraf de afstand gemeten die het projectiel had ten opzichte van de linkerkant van de foto. Omdat de camera niet bewogen heeft gedurende de hele test kunnen we dus zo de onderlinge positie van alle patronen bepalen. Hieronder op Figuur 28 en Figuur 29 is de methode van werken weergegeven, daaronder in Tabel 2 is een overzicht gegeven van de 25 schoten met het kaliber, de snelheid en de relatieve x-positie (uitgedrukt in pixels) van de punt van elk projectiel ten opzichte van de linkerkant van de foto. Alle 25 foto's zijn terug te vinden in bijlage G. Op voorhand was de camera ook gekalibreerd waardoor we weten dat 1 pixel gelijk is aan 0,6047 mm. De gebruikte camera was de Photron Fastcam SA5.



Figuur 28: Schot 1 @ 854 m/s



Figuur 29: Schot 5 aan 452 m/s

Tabel 2: resultaten test

Schot	Kaliber	Snelheid (m/s)	Relatieve x-positie (pixels)	Relatieve x-positie (mm)
1	7,62x51 mm NATO	854	377	228,0
2	7,62x51 mm NATO	853	376	227,4
3	7,62x51 mm NATO	844	379	229,2
4	7,62x51 mm NATO	844	379	229,2
5	7,62x51 mm NATO	452	369	223,1
6	7,62x51 mm NATO	845	378	228,6
7	7,62x51 mm NATO	851	378	228,6
8	7,62x51 mm NATO	856	378	228,6
9	7,62x51 mm NATO	843	378	228,6
10	7,62x51 mm NATO	839	378	228,6
11	7,62x51 mm NATO	854	378	228,6
12	7,62x51 mm NATO	853	378	228,6
13	7,62x51 mm NATO	848	378	228,6
14	7,62x51 mm NATO	845	378	228,6
15	7,62x51 mm NATO	855	378	228,6
16	9x19 mm Parabellum	412	353	213,5
17	9x19 mm Parabellum	416	354	214,1
18	9x19 mm Parabellum	396	354	214,1
19	9x19 mm Parabellum	406	352	212,9
20	9x19 mm Parabellum	404	352	212,9
21	9x19 mm Parabellum	414	352	212,9
22	9x19 mm Parabellum	402	352	212,9
23	9x19 mm Parabellum	385	352	212,9
24	9x19 mm Parabellum	382	352	212,9
25	9x19 mm Parabellum	381	352	212,9

8.2 Bespreking resultaten

De ideale x-coördinaat zou 384 pixels zijn, omdat dit exact het midden van de foto is. Nu is de afstand van het tweede lichtscherm tot de optische as met een rolmeter gemeten, waardoor er een kleine standfout gemaakt is die ervoor zorgt dat de projectielen in dit geval steeds iets meer naar links op de foto staan.

De reden waarom schot 5 een veel lagere snelheid heeft dan de andere schoten van hetzelfde kaliber is omdat dit projectiel een aangepaste (niet nominale) kruitlading had. Alle nominale schoten met 7,62 mm hebben een x-coördinaat (in pixels) tussen 376 en 378. De afwijking tussen deze projectielen onderling is dus maximaal twee pixels (= 1,2 mm) terwijl de snelheid een variatie van maximaal 17 m/s vertoont.

Dit geldt ook voor de projectielen met een kaliber van 9 mm: hoewel hun snelheden verschillen tot 35 m/s vertonen ten opzichte van elkaar varieert de x-coördinaat maximaal 2 pixels (= 1,2 mm) tussen de schoten onderling.

De maximale afwijking bij deze test voor identieke projectielen was dus ongeveer 1,5 mm, rekening houdend met de resolutie die 1 pixel oftewel 0,6 mm bedraagt.

Een fout van 1,5 millimeter (zonder de standfout) bij een nominaal projectiel 7,62 mm komt overeen met een fout van ongeveer 2 microseconden: deze omhelst zowel de onnauwkeurigheid van de FPGA klok, de onnauwkeurigheid van de FPGA berekening (afronden) als de vertragingen in kabels, klemmen, ...

Wel is er een grotere afwijking zichtbaar tussen de projectielen 7,62 mm en 9 mm, en tussen de nominale projectielen van 7,62 mm en onderladen projectiel van hetzelfde kaliber. De reden hiervoor is dat de versnelling die op voorhand ingegeven wordt in de FPGA slechts een schatting is. Deze waarde zal ook variëren van projectiel tot projectiel, net zoals de snelheid, maar de FPGA kan deze niet meten en dus niet corrigeren.

De doelstelling van deze thesis om de fout te beperken tot maximaal twee centimeter is hier zeker gehaald. Indien er steeds met dezelfde projectielen gevraagd wordt zal de fout zelfs nooit meer dan 2 mm bedragen. Ook is het eventueel nog mogelijk om het resultaat te verbeteren door de versnelling nauwkeuriger op te meten en in te voeren in de FPGA.

8.3 Problemen met de Hadland SVR2

De test die besproken is op de voorgaande pagina's is uitgevoerd met de Photron SA5. Het was beter geweest moest deze test uitgevoerd zijn met de Hadland SVR2 omdat deze camera een hogere resolutie heeft. Het grootste nadeel van deze camera (slechts één frame per opname) wordt ook gecorrigeerd door de nauwkeurigheid van het RTTS, omdat één frame voldoende is om het projectiel zichtbaar te maken.

De reden waarom de Hadland niet gebruikt werd is tweeledig. Ten eerste bleek de inwendige logica van de Hadland niet altijd correct te werken waardoor de timing foutief was en het projectiel niet zichtbaar was. Ten tweede werkte ook de flitser niet meer goed. Bij de eerste testen bleek al dat de condensator die de energie voor de lichtflits voorziet soms doorsloeg en daardoor ongewild de flitser activeerde. Ook kwam er soms een deel van deze elektrische energie op het net terecht, waardoor de zekering afsloeg of bepaalde apparaten beschadigd werden.

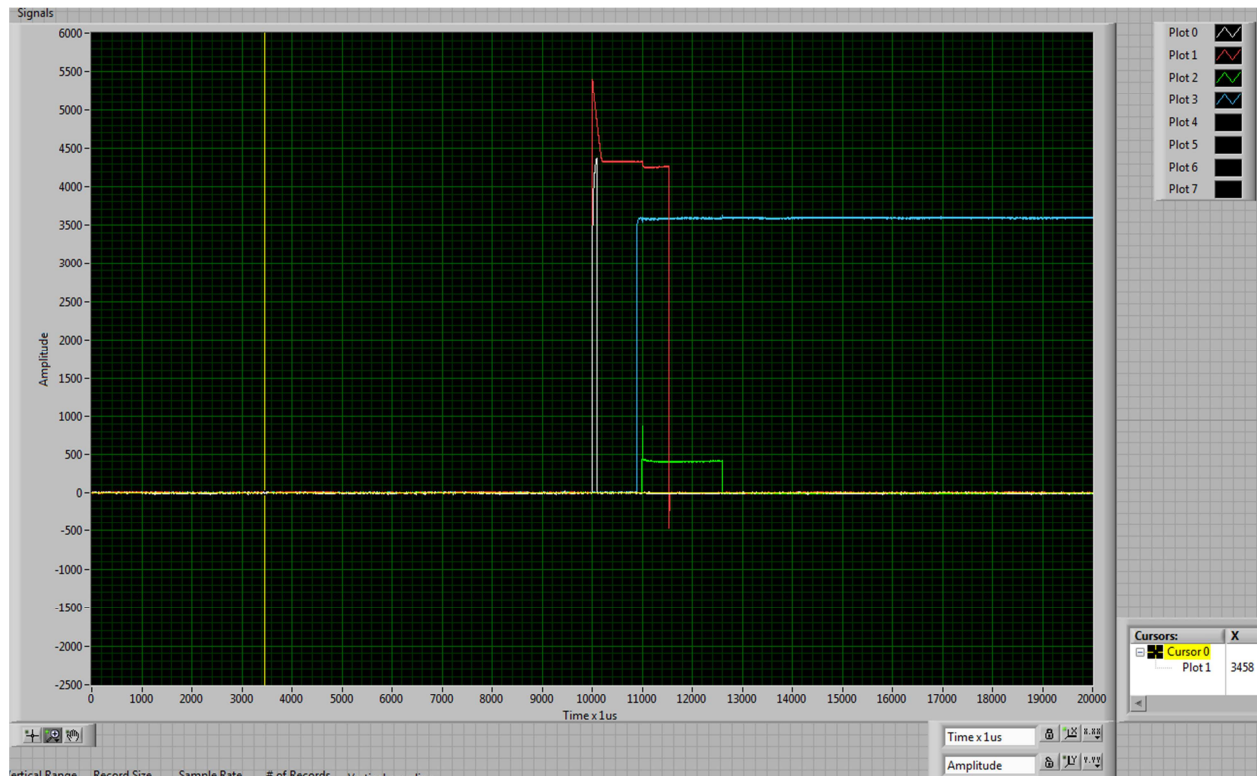
Als oplossing hiervoor zijn er nog een aantal testen uitgevoerd met een paar kleine flitsers maar deze konden niet genoeg licht voorzien om met een korte sluitertijd te werken. Vanaf een sluitertijd van 10 μ s werd het beeld al heel erg donker.

Om terug te komen op de problemen met de inwendige logica van de Hadland: zoals aangehaald in paragraaf 3.4.1 is er een kleine FPGA module ingebouwd in de Hadland die een vooraf ingestelde tijdsvertraging inbouwt tussen de '*trigger IN*' en het moment van opname. Het is onmogelijk om deze FPGA uit te schakelen. Daarom werd deze tijd ingesteld op 200 ns, zoals aangeraden door het programma zelf. De preflash time werd door de externe FPGA afgeregeld.

De resultaten met deze instellingen waren slecht: de foto was donker en het moment van opname was niet correct. Zowel de preflash tijd als de tijdsvertraging waren dus foutief. Omdat er geen datasheet beschikbaar is, en omdat Hadland Imaging alleen nog onder een andere vorm bestaat en geen ondersteuning meer kan bieden, was het zeer moeilijk om de oorzaak van dit probleem te vinden.

Het probleem van de verkeerde preflash tijd is uiteindelijk opgelost met "*trial and error*" door de tijd te laten variëren tot de foto belicht was. De preflash tijd was normaal 8 μ s, terwijl de preflash tijd die ingesteld moest worden 16 μ s bleek te zijn. De resultaten verbeterden ook sterk als de Hadland zelf de flitsers aanstuurde en dit niet door de externe FPGA gebeurde.

Om het probleem van de foutieve tijdsvertraging op te lossen werden alle in-en uitgangen van de Hadland opgemeten met behulp van een digitale oscilloscoop. Hieronder zijn de resultaten weergegeven in Figuur 30.



Figuur 30: signalen Hadland camera

In bovenstaande figuur zien we in het wit de puls die de FPGA stuurt naar de Hadland. Dit is het tijdstip 0 (10000 op de x-as van de figuur). Vervolgens zien we in het rood de *'Trigger out'*. Deze uitgang wordt niet gebruikt bij het RTTS. Vervolgens zien we in het blauw het signaal de flitser(s) moet activeren. Dit signaal wordt 16 μ s (de preflash tijd) voor het groene signaal uitgezonden: dit groene signaal is het signaal de camera uitstuurt om aan te geven dat er een foto gemaakt is. Het verschil tussen het witte en groene signaal is de tijdsvertraging die ingesteld kan worden. In dit geval is deze tijd ingesteld op 1000 μ s. Ten slotte kan de pulsbreedte van het groene signaal ook nog ingesteld worden, in dit geval is deze 1500 μ s.

Uiteindelijk heeft een ex-medewerker van Hadland Imaging het probleem van de foutieve tijdsvertraging kunnen oplossen. Brian Speyer: "In order to get the camera to trigger as soon as it receives an external trigger input then you should set the "delay" channel to its minimum setting which is 200 nanoseconds. In fact due to internal delays in the camera electronics the time from trigger input to the start of the exposure will be around 800 nanoseconds, this is the minimum time." [14]

Uiteindelijk bleek dus het probleem van foutieve timing veroorzaakt te zijn door een te kleine vertragingstijd. In bovenstaande simulatie blijkt ook dat bij een grotere tijdsvertraging de Hadland camera wel de opgelegde timings respecteert.

9 Test met automatisch vuren

9.1 Inleiding

De MAG (Mitrailleuse d'Appui Général) is een machinegeweer van Belgische makelij. Het kaliber van dit wapen is 7,62 x 51 mm NATO en de vuursnelheid is ongeveer 750 schoten per minuut. Het wapen wordt gevoed door middel van een patroonband [19].

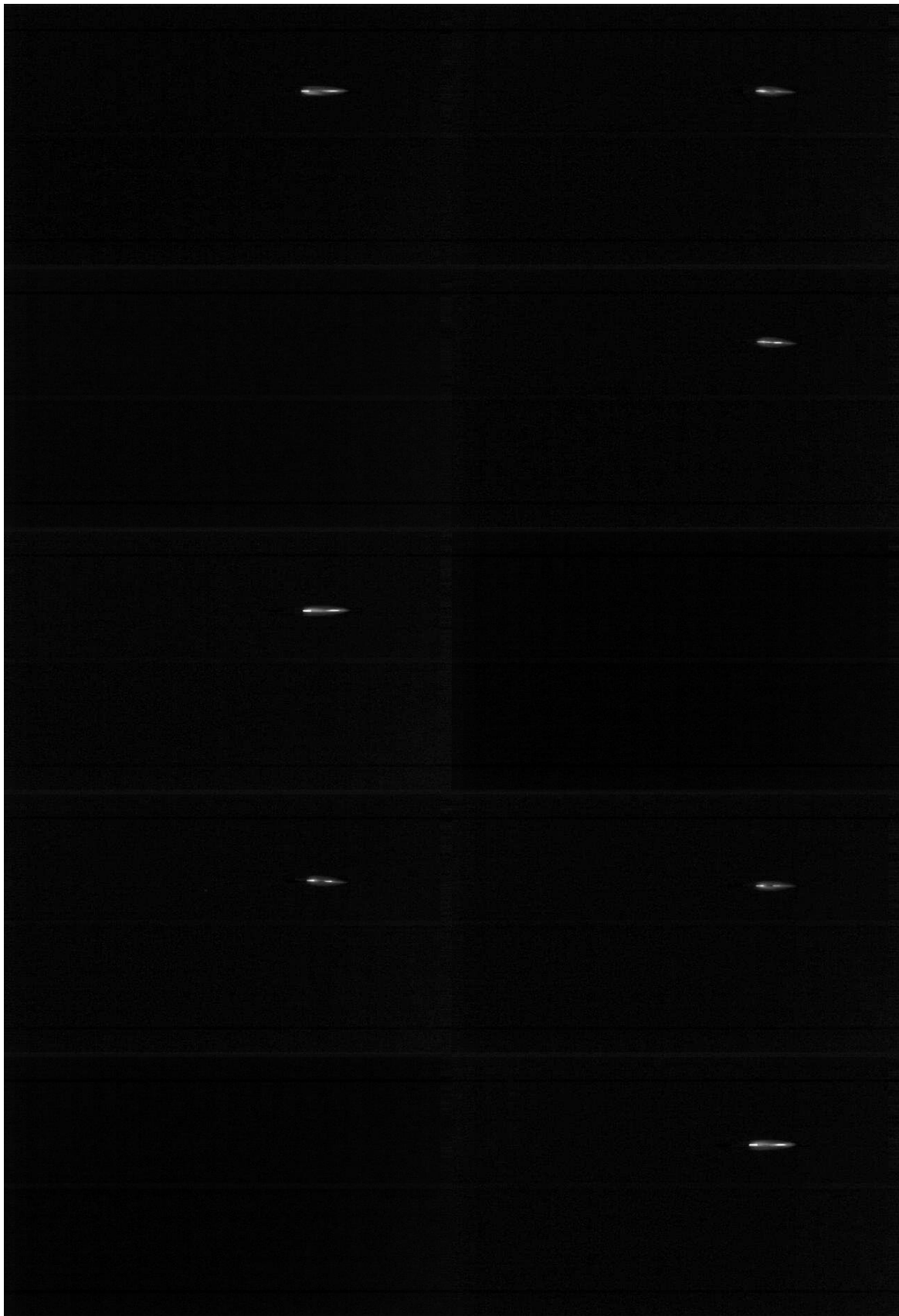
De bedoeling van dit experiment was om te testen of het RTTS ook projectielen kon fotograferen die zeer snel na elkaar werden afgevuurd. Ook moest getest worden of de camera zo snel na elkaar alle projectielen kon fotograferen en de data opslaan.

Om deze test uit te voeren werd het wapen dat normaal gebruikt wordt in het labo vervangen door een FN MAG waarmee telkens salvo's van 10 patronen werd afgevuurd. Het duurt ongeveer 0,8 seconden om een dergelijk salvo af te vuren en op deze tijd stuurt het RTTS dus tien triggersignalen naar de camera die op zijn beurt 10 verschillende opnames moet maken.

9.2 Resultaten en bespreking van de eerste test

Op de volgende pagina zijn de tien opnames van de camera weergegeven in Figuur 31.

De nauwkeurigheid is van ondergeschikt belang hier maar de afstand tussen het meest linkse en het meest rechtse projectiel is ongeveer 2 millimeter.



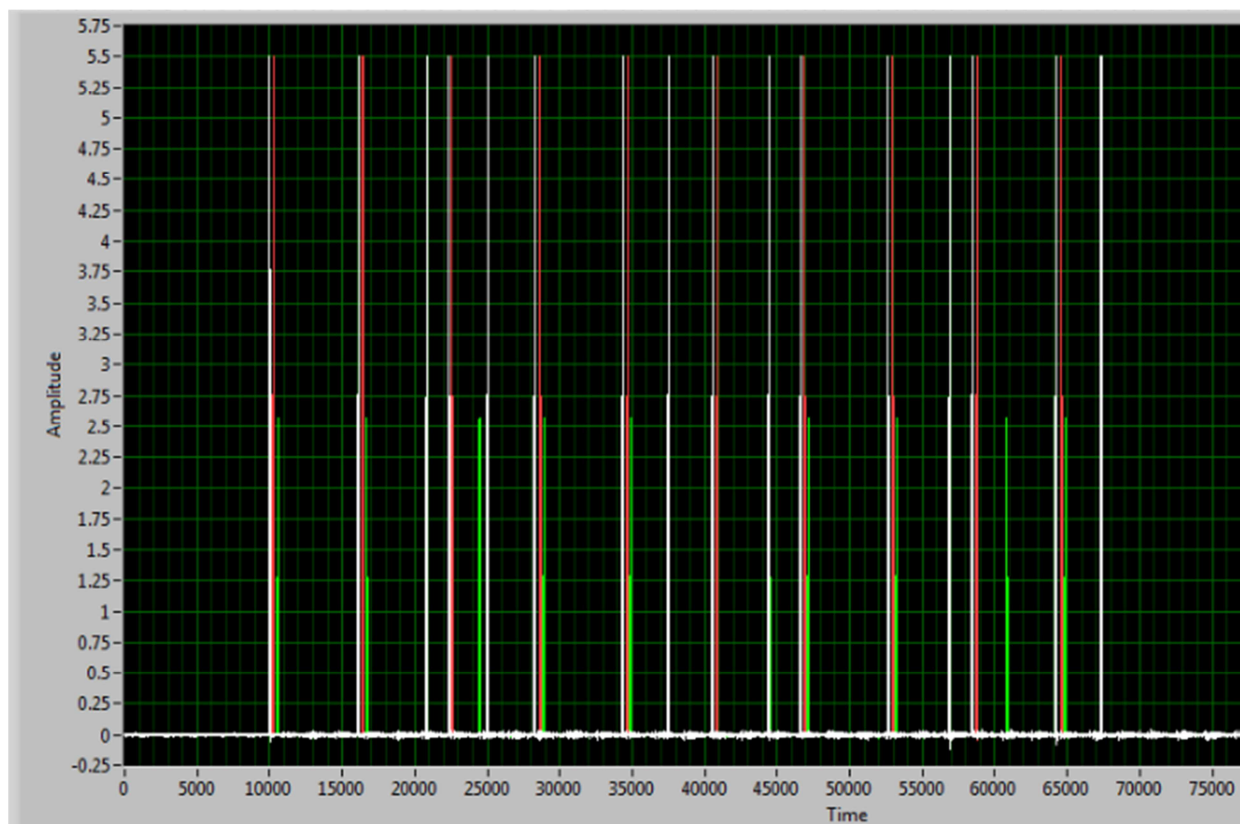
Figuur 31: 1^e salvo MAG

Het valt onmiddellijk op dat het tweede, zesde en negende frame leeg is. Ook bij een tweede test waren er telkens frames waar geen projectiel op zichtbaar was. Een mogelijkheid is dat dit probleem veroorzaakt wordt door tracer munitie: op de vijfde en tiende foto is te zien hoe het projectiel langer is dan de andere projectielen, de reden hiervoor is dat deze kogels een hoeveelheid fosfor bevatten dat ontbrandt in de vlucht om zo de baan van het projectiel zichtbaar te maken voor de schutter. De ontbranding van dit fosfor zou er voor kunnen zorgen dat de infraroodlichtschermen gestoord worden en foutieve resultaten doorgeven aan de FPGA.

Het probleem met deze theorie is echter dat een tracer zo gemaakt is dat hij pas na ongeveer 50 meter tot ontbranding komt, omdat anders de positie van de schutter verraden zou worden. Ook zou dan vlak na de tracer een leeg frame zichtbaar moeten zijn, terwijl dit niet het geval is. Waarschijnlijk is er dus nog een andere oorzaak voor dit probleem.

9.2 Probleemoplossing

Om te achterhalen wat het probleem nu is kunnen we de in- en uitgangsignalen van de FPGA module uitlezen en visualiseren met behulp van de digitale oscilloscoopkaarten die ook aanwezig zijn in de PXI. We krijgen dan een grafiek zoals te zien in Figuur 32. De witte lijnen stellen een ingangspuls voor van het eerste lichtscherm, de rode lijnen een ingangspuls van het tweede lichtscherm en de groene lijnen zijn het triggersignaal dat de FPGA naar de camera uitstuurt.



Figuur 32: verstoorde signalen opgemeten met PXI ten gevolge van interferenties veroorzaakt door de mondingsvlam van een automatisch wapen

Het valt onmiddellijk op dat de tijd tussen twee schoten groot is, vergeleken met de wachttijd van de FPGA of de tijd tussen de twee pulsen van de lichtschermen. Bij de eerste twee schoten is er nog een correcte opeenvolging van respectievelijk wit, rood en groen waarna er zich een lange pauze voordoet voordat dit patroon zich herhaalt. Na het tweede schot stuurt het lichtscherm echter eerst een foutieve puls door, en pas een hele tijd later de correcte puls van het projectiel dat het lichtscherm passeert. De FPGA houdt alleen rekening met de eerste puls en dus berekent deze een snelheid van ongeveer 80 m/s in plaats van ongeveer 840 m/s. De fout die hierdoor gemaakt wordt is zo groot dat het de camera pas een foto neemt wanneer het projectiel al ongeveer tien meter verder is. Vandaar dus dat het derde frame leeg is.

Dit probleem van foutieve pulsen uitgestuurd door het eerste lichtscherm herhaalt zich nog een paar keer waardoor ook het vijfde en negende frame leeg zijn.

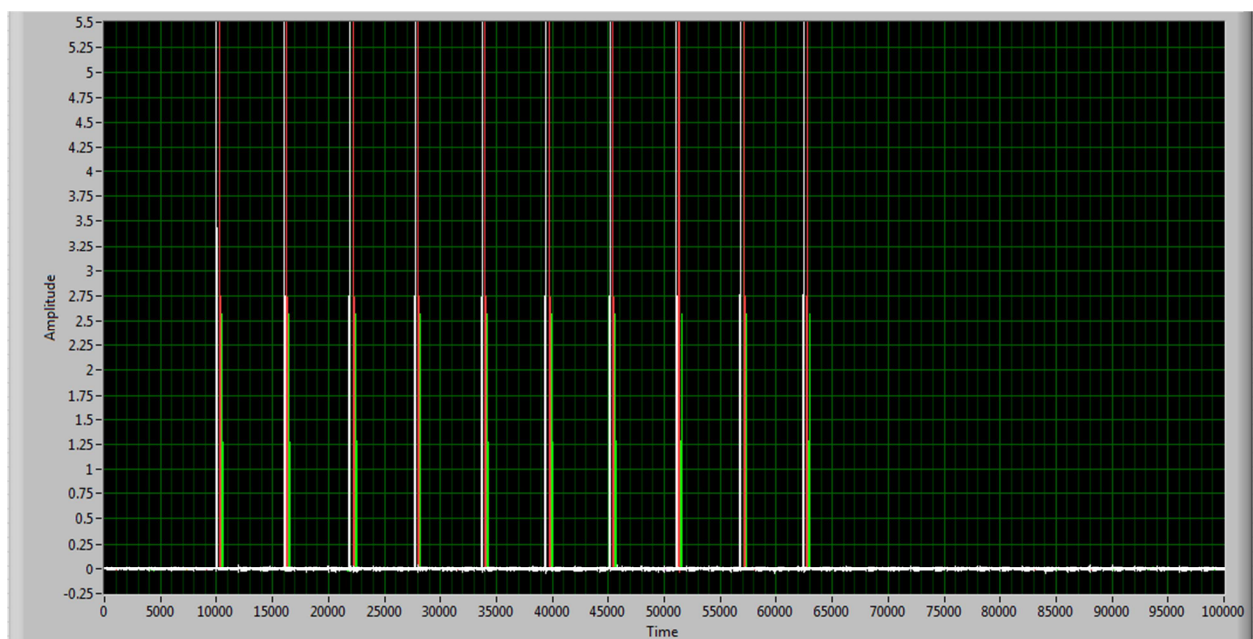
Er zijn een aantal conclusies die we uit deze data kunnen halen:

- De fout wordt enkel veroorzaakt door een foutieve puls, niet door de FPGA of camera.
- Alleen het eerste lichtscherm detecteert iets foutiefs, het tweede lichtscherm niet.
- Het fenomeen doet zich nooit voor bij het eerste schot.

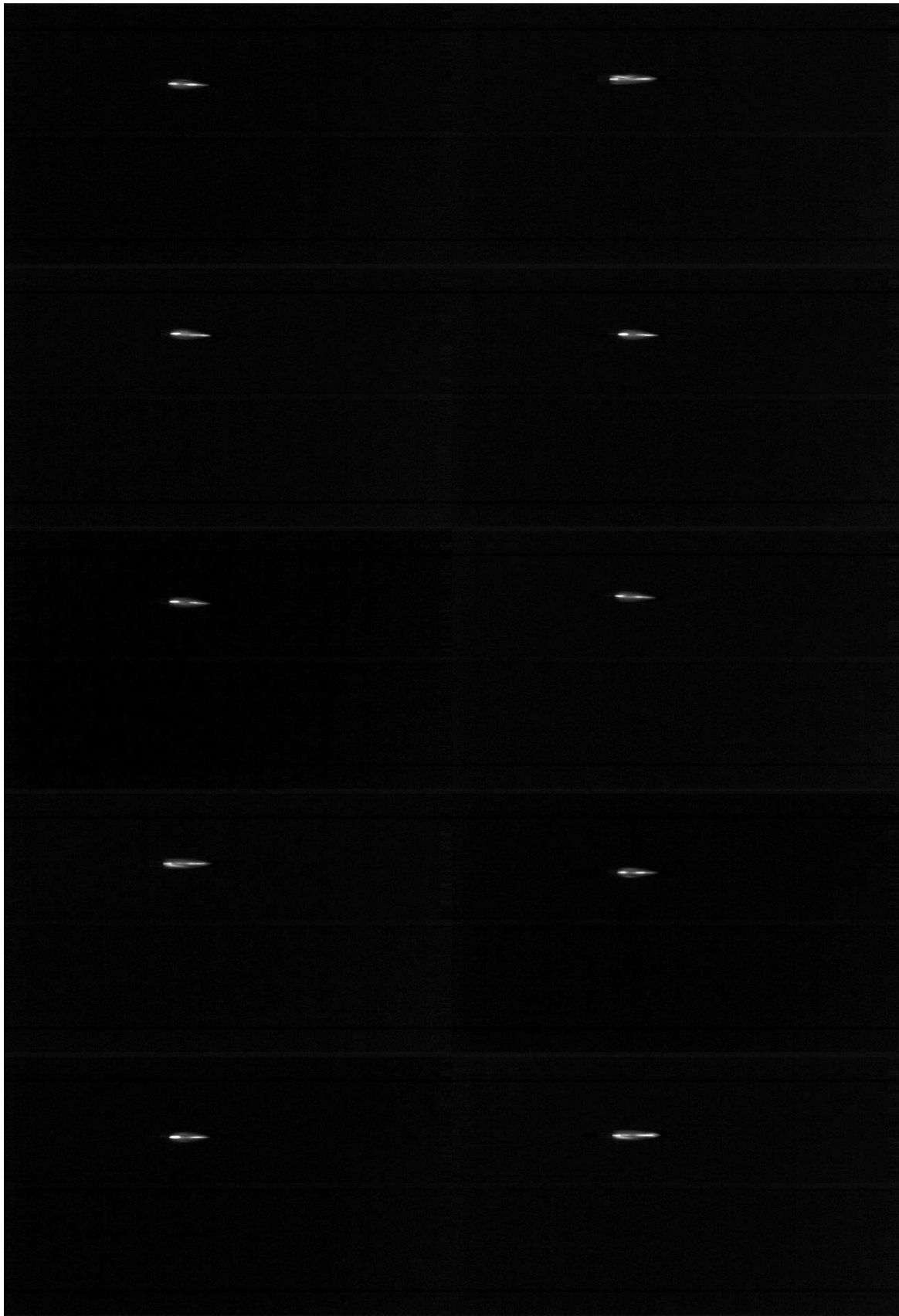
Een mogelijke verklaring is dat de lichtflits die uit de loop komt door het eerste lichtscherm gaat en hierdoor een foutieve detectie veroorzaakt: behalve zichtbaar licht is er immer ook infrarood licht dat uit de loop komt [20]. Om dit te onderzoeken kunnen we de afstand van de loop tot het eerste lichtscherm groter maken, zodat de lichtflits het lichtscherm niet meer kan bereiken.

9.3 Resultaten en bespreking van de tweede test

De resultaten van deze test waarbij de afstand tussen de loop van het wapen en het eerste lichtscherm is vergroot zijn hieronder weergegeven in Figuur 33, samen met de tien foto's die de camera gemaakt heeft in Figuur 34.



Figuur 33: correcte signalen opgemeten met PXI



Figuur 34: tweede salvo MAG

Ontwerp van een realtime triggersysteem toegepast op de fotografie van een ballistisch projectiel

Bij deze test is het RTTS samen met de camera er wel in geslaagd om de tien projectielen afzonderlijk te fotograferen. De nauwkeurigheid bij deze test was ongeveer 2 millimeter.

De reden waarom in eerste instantie een aantal projectielen niet zichtbaar waren was dus inderdaad omdat de mondingsvlam het eerste lichtscherm stoorde. Na ongeveer tien salvo's van tien patronen elk te hebben gevuurd met de MAG blijkt dat de minimumafstand om alle tien patronen te kunnen fotograferen ongeveer 2,5 meter is. Deze afstand hangt af van het type wapen, het kaliber waarmee gevuurd wordt en of het wapen voorzien is van een vlamdemper.

10 Besluit en toekomstig werk

De doelstelling van deze masterproef was om een RTTS te ontwerpen dat in staat is om een foto te maken met een nauwkeurigheid van 2 cm van ballistische projectielen. Deze projectielen hebben snelheden die variëren tussen de 50 en 1000 m/s. Ook hun vorm is volledig anders waardoor hun vertraging ook nog eens sterk kan variëren.

De opdracht is zeker gelukt gezien de resultaten. De FPGA code is efficiënt en snel genoeg om zelfs de snelste projectielen op tijd te fotograferen: de berekeningstijd schommelt tussen de 4000 en de 20000 klokpulsen, wat overeenkomt met 0,1 tot 0,5 μ s. Deze snelle berekeningstijd is het gevolg van een grote efficiëntie in de programmastructuur: er wordt enkel met integers gerekend en de bewerkingen zijn zo gekozen dat deze telkens snel uitgevoerd kunnen worden door een FPGA. Verder zijn er ook een aantal vereenvoudigingen toegepast die de berekening veel eenvoudiger maken zonder de nauwkeurigheid in het gedrang te brengen.

De nauwkeurigheid is veel beter dan 2 cm, zoals oorspronkelijk was vooropgesteld. Dit komt ondermeer door het programma dat nauwkeurig rekent, de opstellingsmethode die zo gedocumenteerd is dat er geen grote fouten kunnen optreden en de camera instellingen die ervoor zorgen dat de tijdsvertraging in de camera zelf minimaal blijft.

Oorspronkelijk was het ook de bedoeling om het RTTS te gebruiken samen met de Hadland SVR2 camera. Dit is niet gelukt omwille van een technisch probleem (defecte flitser) en omwille van de complexe inwendige logica van de Hadland die ervoor zorgde dat het er vaak problemen waren bij de testen. Uiteindelijk is het wel nog gelukt om deze logica te ontwarren en te documenteren in deze thesis waardoor in de toekomst het systeem wel zal kunnen functioneren met de Hadland camera van zodra de defecte flitser gerepareerd is.

Nadat er een test was uitgevoerd die de correcte werking van het programma aantoonde voor projectielen met zeer uiteenlopende snelheden en vertragingen is er ook een experiment gebeurd met een automatisch wapen om te testen of het systeem zeer snel en nauwkeurig meerdere foto's na elkaar kon nemen. Bij een eerste test waren de resultaten voldoende maar niet goed. Een tweede test heeft dan uitsluitsel gegeven over de oorzaak hiervan zodat we uiteindelijk een laatste test konden uitvoeren met quasi perfecte resultaten. Deze test toont ook aan dat het systeem eenvoudigweg geconfigureerd kan worden om andere, vergelijkbare taken uit te voeren.

Zowel ballistiek, hogesnelheidscamera's als FPGA's zijn uitgebreid gedocumenteerd in de literatuur. Het vernieuwende aspect van de masterproef is vooral de interactie tussen deze drie elementen en de problemen die dit met zich meebrengt: de mondingsvlam die de infrarood lichtschermen in de war stuurt, de extreem korte tijdsvertragingen en daaruit volgende de zeer hoge eisen die qua snelheid aan de FPGA gesteld werden, etc.

Omtrent de snelheid waarmee de FPGA moest rekenen valt het op dat in de literatuur er meestal over milliseconden of zelfs seconden gesproken wordt, terwijl bij deze toepassing zelfs een microseconde al een significant verschil was. Daartegenover staat wel dat de berekeningen hier veel eenvoudiger waren dan deze die aangehaald werden in de literatuur.

Ook de communicatie tussen een meer dan 15 jaar oude camera (Hadland SVR2) waar nog amper documentatie van beschikbaar is, en een moderne FPGA is een specifiek probleem waar weinig tot niets over te vinden was in de literatuur.

Tot slot is ook de algemene doelstelling van deze masterproef zeer actueel, zowel in de hedendaagse maatschappij als bij Defensie. Door een realtime sturing kan de efficiëntie verhoogd worden en moet er geen munitie of tijd verspild worden waardoor het mogelijk wordt om meer te doen met dezelfde middelen.

Het RTTS is ook geschikt om, mits kleine aanpassingen, gebruikt te worden voor een hoop andere doeleinden: Fast Fourier Transformatie van de signalen van een Doppler radar of het oplossen van een onderscheppingsprobleem bijvoorbeeld. Er wordt dieper ingegaan op deze mogelijke uitbreidingen in bijlage B.

Literatuurlijst

- [1] Wikimedia Foundation, Inc., „Wikipedia,” Wikimedia Foundation, Inc., 10 Augustus 2013. [Online]. Available: <http://nl.wikipedia.org/wiki/Bewegingsonscherpte>. [Geopend 12 November 2013].
- [2] Wikimedia Foundation, Inc., „Wikipedia,” Wikimedia Foundation, Inc., 8 maart 2013. [Online]. Available: <http://nl.wikipedia.org/wiki/Sluitertijd>. [Geopend 12 november 2013].
- [3] Ballistic Measurements, „Ballistic Measurements: External Ballistics,” Ballistic Measurements, 2014. [Online]. Available: <http://www.ballisticmeasurements.com/doppler-radar.php>. [Geopend 9 Maart 2014].
- [4] Cdt. Ir. Maldague, „Caractéristiques balistiques des Mun,” Ecole Royale Militaire, Brussels, 2007.
- [5] P. P. Toullos and K. D. Hartman, "Doppler-type projectile velocity measurement apparatus and method". Verenigde Staten Patent EP 0023365 A2, 31 Juli 1979.
- [6] R. Thoelen en N. Mentens, Digitale Elektronica, Pearson Custom Publications, 2011.
- [7] Z. Guo, W. Najjar, F. Vahid en K. Vissers, „A quantitative analysis of the speedup factors of FPGAs over processors,” *Proceedings of the 2004 ACM/SIGDA 12th international symposium on Field programmable gate arrays*, p. 170, 2004.
- [8] A. Dunn, „Wikmedia Commons,” 23 juni 2005. [Online]. Available: http://commons.wikimedia.org/wiki/File:Freak_Out,_Oblivion,_night.jpg. [Geopend 12 februari 2014].
- [9] Highspeedwizzard, „Wikimedia Commons,” 25 oktober 2013. [Online]. Available: http://commons.wikimedia.org/wiki/File:SA5_RV.jpg. [Geopend 18 oktober 2013].
- [10] Vision Research, „Phantom V2010,” Vision Research, [Online]. Available: <http://www.visionresearch.com/Products/High-Speed-Cameras/v2010/>. [Geopend 1 juni 2014].
- [11] Specialised Imaging, „SIR3 - Compact Range Camera,” Specialised Imaging, [Online]. Available: <http://specialised-imaging.com/products/sir3-compact-range-camera>. [Geopend 1 juni 2014].
- [12] Specialised Imaging, „SIMD - Ultra High Speed Framing Camera,” Specialised Imaging, [Online]. Available: <http://specialised-imaging.com/products/simd-ultra-high-speed-framing-camera>. [Geopend 1 juni 2014].

- [13] National Instruments, „National Instruments: Products and Services: White Papers,” National Instruments, 8 April 2014. [Online]. Available: <http://www.ni.com/white-paper/9381/en/>. [Geopend 1 Mei 2014].
- [14] B. Speyer, *Persoonlijke communicatie*, 2014.
- [15] National Instruments, „LabVIEW FPGA Compilation Process: From Run Button to Bitfile,” 2 mei 2012. [Online]. Available: <http://www.ni.com/white-paper/9381/en/>. [Geopend 19 september 2013].
- [16] National Instruments, „LabVIEW FPGA Floating-Point Data Type Support,” National Instruments, 25 oktober 2012. [Online]. Available: <http://www.ni.com/white-paper/14188/en/>. [Geopend 1 Juni 2014].
- [17] National Instruments, „LabVIEW FPGA Scaling and Division,” 13 januari 2010. [Online]. Available: <http://www.ni.com/example/29021/en/>. [Geopend 18 september 2013].
- [18] National Instruments, „Multiplying, Dividing and Scaling in LabVIEW FPGA,” National Instruments, 29 juni 2009. [Online]. Available: <http://www.ni.com/white-paper/3661/en/>. [Geopend 18 september 2013].
- [19] Defensie, „Defensie: Materiaal,” Defensie, [Online]. Available: <http://www.mil.be/nl/materiaal/mag-762-mm>. [Geopend 12 Mei 2014].
- [20] M. Kastek, R. Dulski, P. Trzaskawka, T. Piatkowski en H. Polakowski, „Spectral measurements of muzzle flash with multispectral and hyperspectral sensor,” Military University of Technology, Warschau, Polen, 2011.

Aanbevolen literatuur

- Kol. Dr. Ir. G. Dijkmans, „E-book in Ballistics Theory and Software”,
Available: <http://www.e-ballistics.com/>
- N. Weste and D. Harris, „Integrated Circuit Design, Fourth Edition”,
Pearson Education, 2011.
- T. Floyd, Digital Fundamentals, „Tenth Edition”, Pearson Education, 2013.

Lijst met bijlagen

Bijlage A: Handleiding RTTS

Bijlage B: Uitgebreide handleiding bij het RTTS

Bijlage C: Datasheet FPGA module 7851R

Bijlage D: Datasheet meetbasis Drello LS 19 i3

Bijlage E: Datasheet Photron Fastcam SA5

Bijlage F: Tabel snelheidsvariatie 7,62 x 51 mm NATO

Bijlage G: Foto's bij vergelijkende test

Bijlage A

Handleiding RTTS

KONINKLIJKE MILITAIRE SCHOOL

Handleiding bij het RTTS

Departement wapensystemen & ballistiek

Olt KBO Wijnants Jurgen

52 ISI

Handleiding bij het Realtime Trigger Systeem

Inhoudsopgave

1 Verklarende woordenlijst	4
2 Inleiding	6
3 Korte beschrijving van de meetopstelling	6
4 Opstellen van de meetapparatuur.....	7
4.1 De PXI.....	7
4.2 De meetbasis	8
4.3 De camera.....	9
5 Aansluiten van de meetopstelling	11
5.1 Hardware	11
5.2 Configureren van de software.....	13
5.2.1 LabVIEW code op FPGA en PC	13
5.2.2 Compileren.....	16
5.2.3 Software van de camera.....	17
6 Extra informatie	22
6.1 De minimale afstand tussen lichtscherm en camera bepalen.....	22
6.2 De minimale afstand tussen lichtscherm en camera verkleinen.....	23
6.3 Centreren van het projectiel op de foto	24
7 Probleemoplossing.....	24
7.1 Het systeem detecteert geen ingangspulsen.....	24
7.2 Het systeem berekent alles correct maar de camera wordt niet <i>getriggerd</i>	25
7.3 Het systeem voert wel een berekening uit maar de uitkomst is foutief.....	25
7.4 Het systeem neemt wel een foto maar de foto is donker.....	26

Handleiding bij het Realtime Trigger Systeem

1 Verklarende woordenlijst

Break-out box: een doosje met klemmenstroken dat helpt bij het aansluiten van bepaalde connectoren.

Exposure: Maat voor de hoeveelheid licht die de lens binnenlaat, een kleinere waarde duidt op meer licht.

FPGA: Field Programmable Gate Array, een geïntegreerde schakeling bestaande uit logische componenten die geschakeld kunnen worden als logische poorten (AND,XOR,NAND,...).

Framerate: getal dat aangeeft hoeveel beelden per seconde (fps, frames per second) een videocamera opneemt.

k-waarde: de afstand tussen twee infraroodlichtschermen die gemonteerd zijn op een meetbasis.

Motion Blur: leenwoord voor bewegingsonscherpte.

NI: National Instruments, Amerikaans bedrijf dat eigenaar is van de LabVIEW software en de bijbehorende hardware produceert.

Path: Engelstalige benaming voor een verwijzing naar een map op het geheugen van een computer.

PXI: PCI eXtensions for Instrumentation : een standaard van National Instruments om meettoestellen aan te sluiten via de PCI aansluiten van een PC. Komt meestal voor onder de vorm van een kast met daarin plaats voor één PC en meerdere meettoestellen die dan automatisch verbonden worden met de PC via een rail in de achterzijde van de kast.

Sluitertijd: De sensor van een digitale camera registreert het licht dat invalt in de lens voor een bepaalde tijd om een opname te maken. Deze tijd is de sluitertijd.

Trigger: Spanningspuls die het bevel geeft aan een camera om zijn opname te starten.

VI: Virtual Instrument, de naam voor een programma dat in LabVIEW geschreven is. Een VI bestaat steeds uit twee delen: een "*front panel*" dat knoppen, grafieken, status van bits, ... bevat en een blokdiagram waar de verwerking van data gebeurt aan de hand van blokken die verbonden worden met virtuele draden.

x-waarde: de afstand tussen het laatste lichtscherm en de optische as van de camera.

Handleiding bij het Realtime Trigger Systeem

2 Inleiding

Deze handleiding beschrijft de werking van het realtime triggersysteem (RTTS) : een meetsysteem ontworpen om in realtime (dus terwijl de gebeurtenis plaatsvindt) metingen te verrichten, deze meetdata te verwerken en uiteindelijk een signaal uit te sturen. Meer specifiek bij deze toepassing zullen we de snelheid van een projectiel meten tijdens zijn vlucht en hiermee een tijdsvertraging bepalen om zo op het correcte moment een triggersignaal uit te sturen dat gebruikt kan worden om bijvoorbeeld een camera te triggeren. Deze handleiding gaat niet dieper in op de werking van de code. Dit wordt verder besproken in de uitgebreide handleiding.

3 Korte beschrijving van de meetopstelling

De snelheid van het ballistisch projectiel wordt gemeten aan de hand van twee infraroodlichtschermen, die samen gemonteerd zijn op een meetbasis. De afstand tussen deze twee schermen is dan ook precies gekend. Wanneer het projectiel door het lichtscherm passeert zal de infrarooddetector dit opmerken en een signaal uitsenden.

De communicatie en rekenkracht worden voorzien door een FPGA (Field Programmable Gate Array), dit is een computer die op hardware niveau werkt om zo zeer snel berekeningen kunnen uit te voeren. Deze FPGA module is gemonteerd in de vorm van een kaart in een PXI (PCI eXtensions for Instrumentation) van National Instruments. In deze PXI zit ook een PC module die gebruikt wordt om de FPGA te programmeren. De FPGA beschikt verder over drie 68-pins connectoren die gebruikt kunnen worden om zowel analoge als digitale signalen in te lezen of uit te sturen. Om deze 68-pins connectoren aan te sluiten maken we gebruik van zogenaamde “*break-out boxes*”, dit zijn doosjes waar de 68 pinnen naar buiten gebracht worden onder de vorm van 68 klemmen die aangesloten kunnen worden.

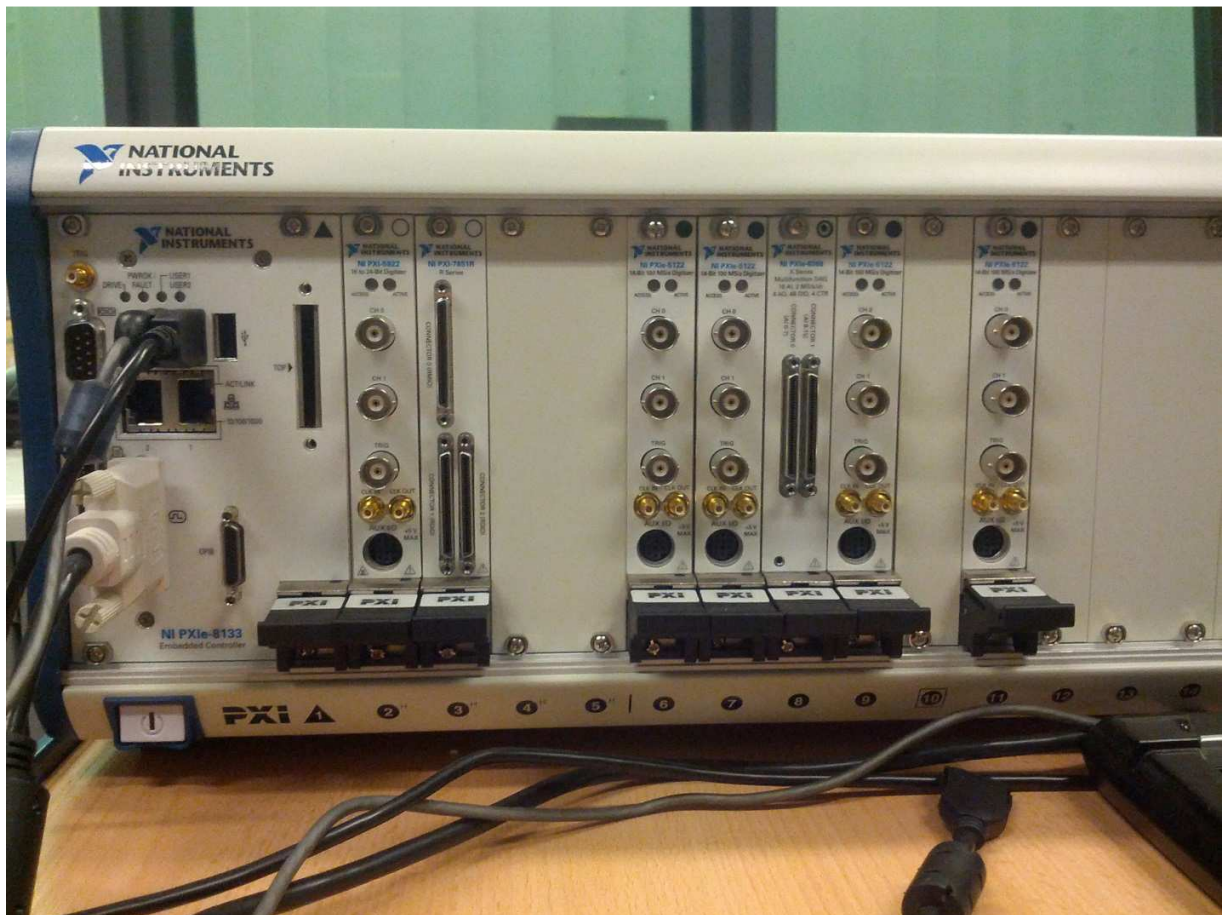
De twee infrarood lichtschermen zullen dus verbonden worden met de FPGA aan de hand van twee coaxiale kabels. Om het triggersignaal naar de camera te sturen wordt er ook een coaxiale kabel gebruikt.

4 Opstellen van de meetapparatuur

Er zijn drie grote onderdelen die moeten worden opgesteld: de PXI (met daarin de FPGA module en pc), de meetbasis met minstens 2 lichtschermen en de camera.

4.1 De PXI

De PXI, weergegeven in figuur 1, wordt best opgesteld in de controlekamer om deze te beschermen tegen ricochets en andere gevaren. Omdat deze verbonden moet worden met de apparatuur in de schietunnel is het best om deze in de buurt van een aansluitdoos op te stellen. Voor een normale werking zijn er twee stopcontacten nodig, indien het nodig is om te debuggen zijn er 5 stopcontacten nodig.



Figuur 1: PXI aanwezig in het labo

4.2 De meetbasis

De meetbasis, zichtbaar op figuur 2, moet opgesteld worden in de schietunnel, tussen de loop van het wapen en de camera. Er moet een minimumafstand van ongeveer 20 cm in acht genomen worden tussen de laatste meetbasis en de optische as van de camera omdat de FPGA tijd nodig heeft om zijn berekeningen uit te voeren. Ideaal is een afstand van ongeveer 1 meter. De meetbasis moet parallel met de baan van het projectiel opgesteld worden, zowel in de lengte- als in de hoogterichting. Hiervoor kunnen de 4 schroeven gebruikt worden die vastgrijpen in de rail in de vloer.



Figuur 2: de meetbasis van Drelo die gebruikt wordt in het labo ballistiek

4.3 De camera

De opstelling van de camera is iets complexer dan die van de meetbasis. Het is namelijk belangrijk dat de hoek tussen de optische as van de camera en de baan van het projectiel niet teveel afwijking vertoont, omdat deze een grote fout zou induceren in het uiteindelijke resultaat. De camera kan zowat overal geplaatst worden tussen de meetbasis en de doelenzone, maar hoe dichterbij de meetbasis hoe groter de nauwkeurigheid (dit zolang de afstand niet kleiner wordt dan de minimumafstand).

De afstand van de lens tot de baan van het projectiel is vrij te kiezen in functie van de zoom van de lens en het gewenste resultaat. De hoogte van de lens is best dezelfde als de hoogte van de baan van het projectiel (deze kan zichtbaar gemaakt worden met behulp van een laser op de loop van het wapen) om zo de sferische vertekening te verkleinen. Eventueel kan de hoogte van de loop gemeten worden en de hoogte van de lens zo aangepast worden dat deze zich op dezelfde hoogte als de loop bevindt.

Om de hoek tussen de optische as en de baan van het projectiel nauwkeurig af te regelen op 90° , kan er gebruikt gemaakt worden van een toestel dat uit twee lasers bestaat die onder een hoek van 90° gemonteerd zijn (figuur 3). Eén laser kan dan opgelijnd worden met de laserstraal van de loop (figuur 3), en vervolgens kunnen we de camera verschuiven totdat de andere laser midden op zijn lens invalt (figuur 4).



Figuur 3: apparaat met twee laserstralen die loodrecht ten opzichte van mekaar opgesteld staan



Figuur 4: de Photron camera met de laserlijn zichtbaar

We zien op figuur 4 ook dat de laser opgelijnd is met de twee schroefjes boven en onder de lens. Op deze manier is de optische as van de camera opgelijnd met de laser en dus staat ze loodrecht op de baan van het projectiel.

5 Aansluiten van de meetopstelling

5.1 Hardware

De eerste stap is om de PXI aan te sluiten. Deze heeft een beeldscherm, muis, en toetsenbord nodig qua randapparatuur om optimaal te functioneren.

Vervolgens moet er een break-out box, type SCB-68 (figuur 5) aangesloten worden op de FPGA module van de PXI met behulp van een 68-pins kabel (figuur 6). Er zijn drie connectoren: connector 0, connector 1 en connector 2. Deze zijn identiek qua opbouw maar connector 0 heeft een aantal pinnen die voorbehouden zijn voor analoge signalen, terwijl connector 1 en 2 alleen digitale signalen kunnen verwerken. De inputs en outputs zijn geconfigureerd om via connector 1 ingelezen te worden dus de break-out box moet ook op connector 1 aangesloten worden.



Figuur 5: SCB-68 breakout box



Figuur 6: 68-pins male to male kabel

In een tweede stap kunnen we dan de verschillende coaxiale kabels aansluiten aan de break-out box. In totaal hebben we drie kabels nodig: twee voor de input van de lichtscherms en één voor de output naar de camera. Deze drie IO's zijn respectievelijk geconfigureerd als DIO0, DIO1 en DIO2. In de bijgevoegde datasheet voor de FPGA module staat aan welke pinnen deze 3 kabels telkens aangesloten moeten worden. De positieve ader wordt respectievelijk aan pinnen 35,36 en 37 aangesloten terwijl de massa aan pinnen 0,1 en 2 aangesloten moet worden. Deze pinnen bevinden zich rechtsboven in de break-out box.

Handleiding bij het Realtime Trigger Systeem

Het is belangrijk om er voor te zorgen dat deze kabels goed aangesloten worden. Als één van de kabels fout aangesloten is zal het systeem niet werken, en er is geen automatische detectie om dit probleem te signaleren, waardoor het vaak vrij lang duurt om het probleem te ontdekken. Verder zal de FPGA indien één van zijn in- of uitgangen niet aangesloten is toch soms hier een signaal op detecteren, en door deze fout-positieve meting lijkt het natuurlijk alsof de kabel wél correct aangesloten is. Verder moet het raakoppervlak tussen de koper van de draad en de metalen klem zo groot mogelijk genomen worden, om de impedantie te verkleinen. Indien de impedantie te hoog is zal er een tijdsvertraging optreden waardoor het systeem minder nauwkeurig functioneert. Ook is het mogelijk dat de amplitude van de spanningspuls niet groot genoeg is waardoor de FPGA het signaal niet registreert. Een andere manier om de impedantie te verkleinen is door het uiteinde van de draad te solderen zodat het beter geleidt en beter contact maakt met de klem.

Tenslotte kunnen we deze coaxiale kabels nu aansluiten aan hun respectievelijke meettoestellen. De kabel die op DIO0 aangesloten is, moet aangesloten worden op het lichtscherm waar het projectiel het eerst zal doorvliegen. Het is ook belangrijk de coaxiale kabel op de digitale uitgang van het lichtscherm aan te sluiten, en niet op de analoge uitgang. Maak ook zeker niet de vergissing om de uitgang van de FPGA aan te sluiten op de uitgang van het lichtscherm, want de inwendige elektronica is storingsgevoelig en kan beschadigd worden door een uitgangspuls van de FPGA te ontvangen.

De kabel die op DIO1 is aangesloten, moet op een identieke manier aangesloten worden op het tweede lichtscherm.

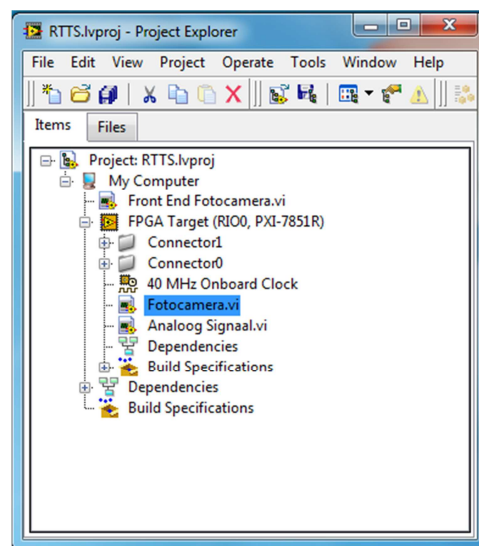
De kabel die tenslotte op DIO2 aangesloten is, moet naar de 'Trigger IN' van de camera. De naamgeving van deze ingang kan verschillen naargelang het type camera dat gebruikt wordt. Raadpleeg hiervoor de datasheet van de camera die gebruikt zal worden.

Indien nodig kunnen er eventueel nog op gelijkaardige wijze andere kabels aangesloten worden op de overige digitale of analoge in- en uitgangen. Dit moet dan wel geconfigureerd worden in de software.

5.2 Configureren van de software

5.2.1 LabVIEW code op FPGA en PC

Eenmaal alles correct is opgesteld en aangesloten is het tijd om de software correct te configureren. Start de PXI op en start LabVIEW2012. De software is niet compatibel met LabVIEW 2011. Open vervolgens het LabVIEW project *RTTS.lvproj* [figuur 7]. Dit is een LabVIEW project met daarin de verschillende VI's (virtual instruments, de naam die LabVIEW geeft aan elk stuk code dat in LabVIEW geschreven is) die nodig zijn om het RTTS te laten werken.



Figuur 7: overzicht van het LabVIEW project

Er bevinden zich 2 VI's in dit project: 1 vi die op de pc draait (*pc_code.vi* genoemd) en 1 bijhorende vi (*FPGA_code.vi* genoemd) die op de FPGA draait. De reden hiervoor is simpel: de code die op de FPGA draait is geschreven om zo snel mogelijk de nodige berekeningen te kunnen uitvoeren. Als een gevolg hiervan werkt deze code met nogal abstracte eenheden en berekent deze geen gegevens die nuttig kunnen zijn voor de verwerking achteraf. Deze gegevens worden pas achteraf door de pc berekend omdat deze trager werkt maar wel veel eenvoudiger te programmeren is. Deze 2 VI's zullen dus samenwerken om enerzijds heel snel de trigger naar de camera te sturen en anderzijds om de gebruiker een overzicht te geven van allerlei nuttige parameters die de pc heeft berekend op basis van de metingen van de FPGA.

Deze VI's zijn compatibel met twee soorten camera's: ten eerste een hogesnelheidsvideocamera type Photron die een trigger verwacht op het moment dat hij zijn opname moet maken. Er is geen interne tijdsvertraging meer en er is ook geen probleem met een flitser omdat deze met continue belichting werkt.

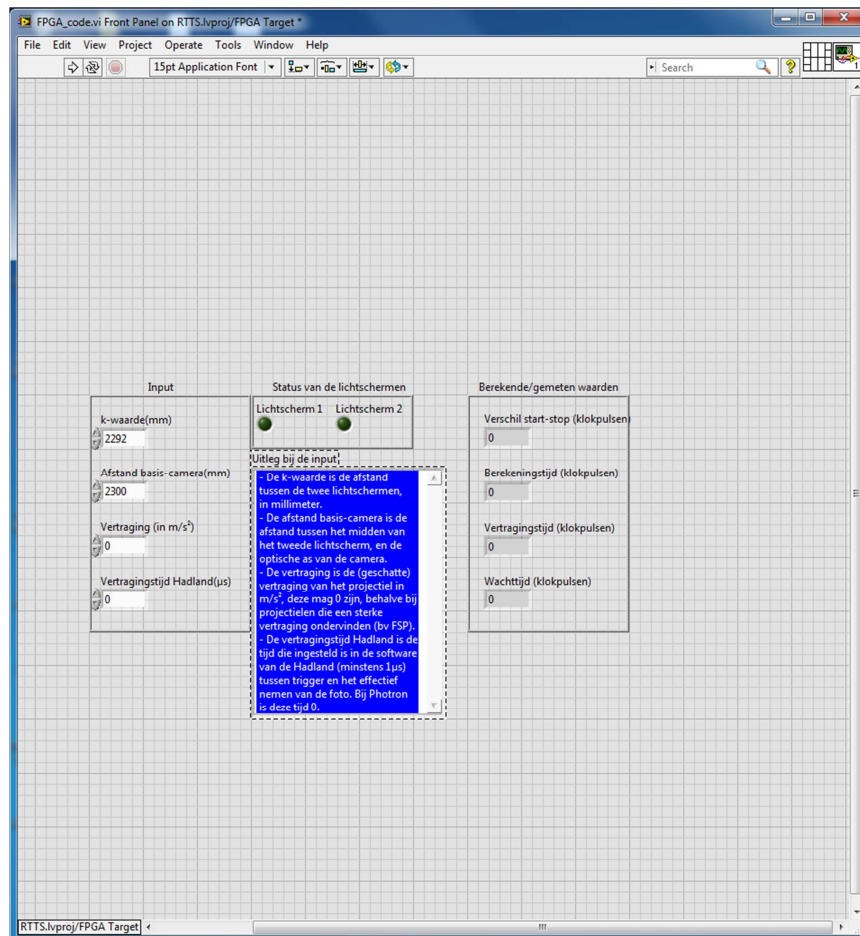
Ten tweede een type hogesnelheidsfotocamera type Hadland, waarbij we de camera zelf de flitser laten triggeren. Dit houdt in dat we een vaste, instelbare tijdsvertraging in de camera zelf hebben, waardoor we onze trigger dus ook vroeger moeten sturen dan het moment waarop we een foto willen hebben.

Deze programma's hebben ook een component die rekening houdt met de vertraging van het projectiel. Een projectiel dat door de lucht vliegt zal afgeremd worden door de luchtweerstand die het ondervindt. Bij snelle,

Handleiding bij het Realtime Trigger Systeem

aerodynamische projectielen is deze vertraging te verwaarlozen, maar bij trage, niet aerodynamische projectielen (bijvoorbeeld een Fragment Simulating Projectile) kan deze vertraging een grote fout veroorzaken.

Open dus de twee VI's en stel de parameters correct in [figuur 8]. De vier parameters die nog ingesteld moeten worden zijn de afstand tussen de lichtschermen (k-waarde), de afstand tussen de tweede basis en de optische as van de camera (x-waarde), de vertraging van het projectiel in m/s^2 (indien van toepassing) en ten slotte ook nog de vertragingstijd die ingesteld is in de software van de Hadland.



Figuur 8: de interface van de FPGA code

Handleiding bij het Realtime Trigger Systeem

De k-waarde wordt bepaald door de producent van de meetbasissen (Drehlo) en is zéér nauwkeurig opgemeten. Deze staan genoteerd voor alle meetbasissen onder de referentiechronometer die te vinden is in de controlekamer van het labo. De waarde staat daar genoteerd in meter, maar in de vi moet deze ingegeven worden in millimeter.

De x-waarde moet zelf opgemeten worden, bijvoorbeeld met behulp van een laserafstandsmeter of een rolmeter. De correcte afstand is vanaf het midden van de reflector van de tweede meetbasis tot de optische as die (virtueel) uit het midden van de lens komt. Deze afstand moet ook ingevoerd worden in millimeter.

De versnelling ten slotte moet, indien van toepassing, worden ingegeven in m/s^2 en is slechts een benadering, omdat dit van projectiel tot projectiel verschilt. In bijlage is een tabel met de vertraging van de meest gangbare kalibers in m/s^2 toegevoegd. Voor andere kalibers kan deze experimenteel bepaald worden of opgezocht worden op internet. Indien er 0 m/s^2 ingevuld wordt zal de FPGA de vertraging gewoon negeren.

Het is ook mogelijk om op voorhand een *path* en documentnaam te specificeren waar de meetgegevens opgeslaan zullen worden. Indien dit niet op voorhand wordt ingegeven zal er automatisch een dialoogvenster openen na de eerste meting zodat de gebruiker kan kiezen waar hij zijn meetgegevens wilt opslaan. De gegevens kunnen achteraf terug worden geopend met Notepad of Excel. De gegevens die geëxporteerd worden zijn respectievelijk: de snelheid in m/s , de totale vertragingstijd in microseconden, de berekeningstijd in microseconden en de wachttijd in microseconden [figuur 9]

The screenshot shows a graphical user interface for a PC application. It features a grid background. At the top left, there is a text box labeled 'file path for recorded data' containing the path 'C:\Users\PXI-FPGA\Desktop\Wijnants\Output VI\test.txt'. Below this, there is a section titled 'Data to write' which contains four input fields, each with a label and a value:

Label	Value
Snelheid(m/s)	1000.1
Vertragingstijd(μs)	1000
Berekeningstijd(μs)	100.6
Wachttijd(μs)	899.4

Figuur 9: interface van de PC code, met de 4 parameters die weggeschreven zullen worden naar het daarboven gedefinieerde path.

Handleiding bij het Realtime Trigger Systeem

Indien de totale vertragingstijd kleiner is dan de berekeningstijd zal de wachttijd negatief worden en zal de FPGA dus niet functioneren. Dit is te wijten aan een te korte afstand tussen het tweede lichtscherm en de camera. In zo een situatie is het ook mogelijk dat de VI's niet meer functioneren tenzij ze heropgestart worden. Dit om ervoor te zorgen dat de gebruiker zich bewust is van het probleem.

Door op 'Run' te drukken ten slotte zal de VI eenmaal uitgevoerd worden. Indien we op 'Run continuously' drukken wordt de VI continu uitgevoerd. In dit geval kan er keer op keer gevraagd worden zonder dat het nodig is om in te grijpen in LabVIEW. Op het einde van de reeks testen kan het tekstbestand, dat chronologisch alle parameters bevat van de projectielen, gewoon uitgelezen worden. Wel is het nodig dat af en toe gecontroleerd wordt of alles nog functioneert, anders is het mogelijk om de data van meerdere proeven te verliezen indien zou blijken dat de FPGA geen signalen meer registreert door bijvoorbeeld een kabelbreuk.

5.2.2 Compileren

Als we eenmaal de twee VI's correct hebben ingesteld, kunnen we deze nu opstarten. Voor de VI die op de pc draait zal dit geen enkel probleem geven en slechts enkele seconden duren. Voor de VI op de FPGA daarentegen zal er eerst een compilatie uitgevoerd moeten worden. Dit proces, waarbij de code vertaald wordt naar een schakeling van logische poorten, duurt 5 à 20 minuten. Telkens er iets verandert aan de VI, moet deze compilatie opnieuw uitgevoerd worden. Zolang er geen andere VI gecompileerd wordt, zal de vorige steeds in het geheugen van de FPGA blijven, zelfs na het heropstarten van het systeem.

Bij het submenu 'compilation settings' is het mogelijk om de methode van compileren te kiezen. Standaard staat deze ingesteld om zo snel mogelijk te compileren, wat handig is indien de code vaak veranderd wordt. Deze kan ook ingesteld worden om bijvoorbeeld zo weinig mogelijk logische poorten te gebruiken voor in het geval dat het programma maar net in het geheugen van de FPGA past. Een derde optie is gefocust erop om de applicatie zo snel mogelijk te laten runnen. Deze optie is aan te raden voor het RTTS tenzij er om specifieke redenen een andere keuze gewenst is.

Tijdens het compileren kunnen er 5 verschillende rapporten geraadpleegd worden, die informatie geven over de geschatte duur van de compilatie en het gebruik van de logische poorten, lookup tables en schuifregisters. Deze kunnen een goede indicatie geven van hoe zwaar het programma is, en of het eventueel nodig is om de code efficiënter te schrijven.

5.2.3 Software van de camera

Om de digitale beelden van elke camera uit te lezen bestaat er specifieke software die meegeleverd wordt door de fabrikant. Hieronder de software van de Hadland en de Photron camera's besproken worden. Voor andere camera's zullen de instellingen vergelijkbaar doch licht verschillend zijn.

Hadland

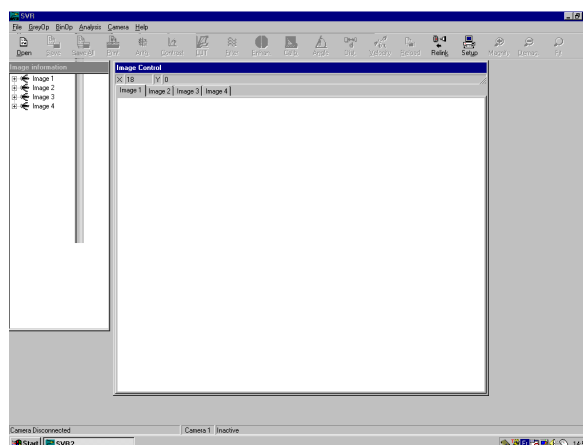
De Hadland SVR2 hogesnelheidsfotocamera heeft een ingebouwde PC die draait op Windows 95. Door de camera met een coaxiale kabel aan te sluiten op de usb poort van een laptop (met behulp van een adapter) kan de controle overgenomen worden en gebeuren vanop de laptop. De coaxiale kabel die de camera verbindt met de laptop moet voorzien worden van twee afsluitweerstand aan beide uiteinden van de kabel.

Als we de software van Hadland[figuur 10] opstarten neemt deze automatisch de controle over van de camera. We kunnen dan een aantal parameters instellen die een invloed hebben op de kwaliteit van de foto zoals exposure, sluitertijd, etc...

Exposure is een parameter die een maat geeft voor hoeveelheid licht die kan binnenvallen via het diafragma. Deze waarde wordt best zo klein mogelijk genomen (= meeste licht).

De sluitertijd is de tijd tussen het begin en einde van één foto. De sluiters moet namelijk even openen om het beeld vast te leggen op de sensor. Tijdens dit interval blijft het projectiel natuurlijk ook bewegen, en daarom krijgen we bewegingsonscherpte (= motion blur) in onze foto: het projectiel lijkt uitgerokken en vervormd. Het is dus belangrijk om de sluitertijd altijd zo klein mogelijk te kiezen, maar er moet natuurlijk genoeg licht zijn om de foto te belichten op deze korte tijd.

Vaak is het ook mogelijk om de focus, zoom en de opening van het diafragma in te stellen door te draaien aan de lens die gemonteerd staat op de camera.

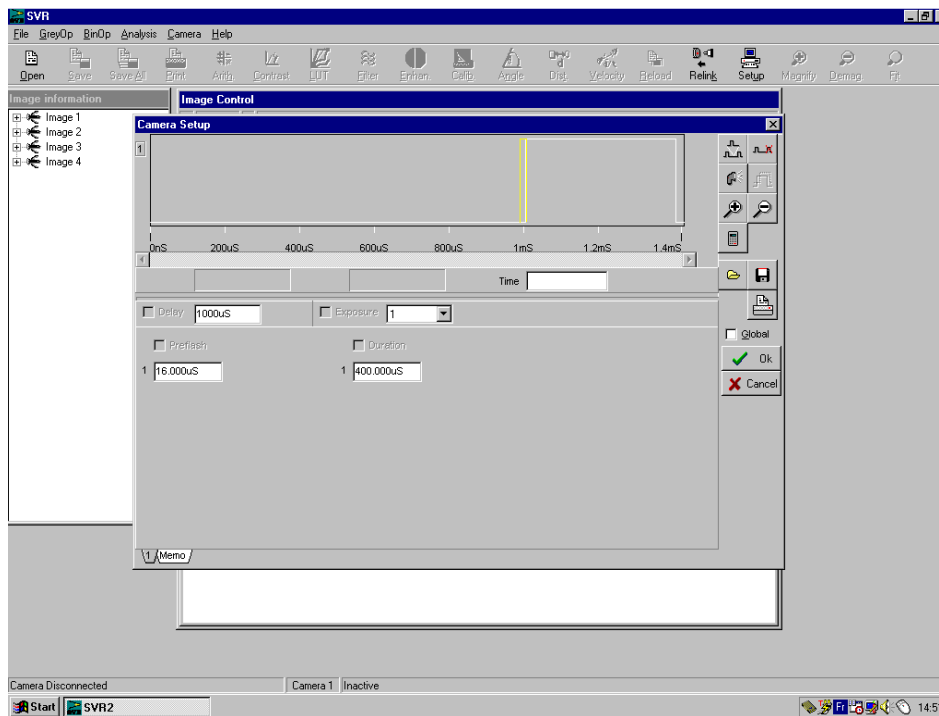


Figuur 10: de software om de Hadland SVR2 aan te sturen

De Hadland camera beschikt zelf ook over een interne rekenmodule die bepaalde functies kan uitvoeren. Deze interne rekenmodule is niet uit te schakelen. Een groot nadeel hiervan is dat we altijd onze trigger een

Handleiding bij het Realtime Trigger Systeem

vast aantal microsecondes moeten sturen voor we onze foto willen nemen. Deze tijdsvertraging is ook in te stellen via deze software [figuur 11]. Deze tijdsvertraging moet groter zijn dan de tijd die de flitser(s) nodig hebben om hun maximale lichtintensiteit te bereiken. Deze tijd hangt af van flitser tot flitser, en varieert ongeveer tussen de 5 en de 100 μs . Indien er geen flitser gebruikt wordt moet de vertragingstijd nog steeds minstens 1 μs bedragen, en niet 200 ns zoals de software van de Hadland suggereert¹.



Figuur 11: het scherm om de vertragingstijd, preflash tijd, en pulsbreedte van de Hadland in te stellen.

Verder geeft de software ook de mogelijkheid om de camera te wapenen, waarna de camera wacht op een triggersignaal om zijn foto te nemen. Het is ook mogelijk om een foto als test te nemen, dit is vooral handig om te testen of de belichting wel voldoende is voor de gekozen sluitertijd en exposure.

Na elke foto moet deze handmatig opgeslaan worden op de harde schijf van de computer. De software heeft ook ongeveer 20 seconden nodig voordat de foto verstuurd is van de camera naar de laptop en dus zichtbaar is. Dit heeft tot gevolg dat er minimaal ongeveer een minuut tussen elk schot zal moeten zijn voordat de verwerking en opslag van de vorige foto gelukt is.

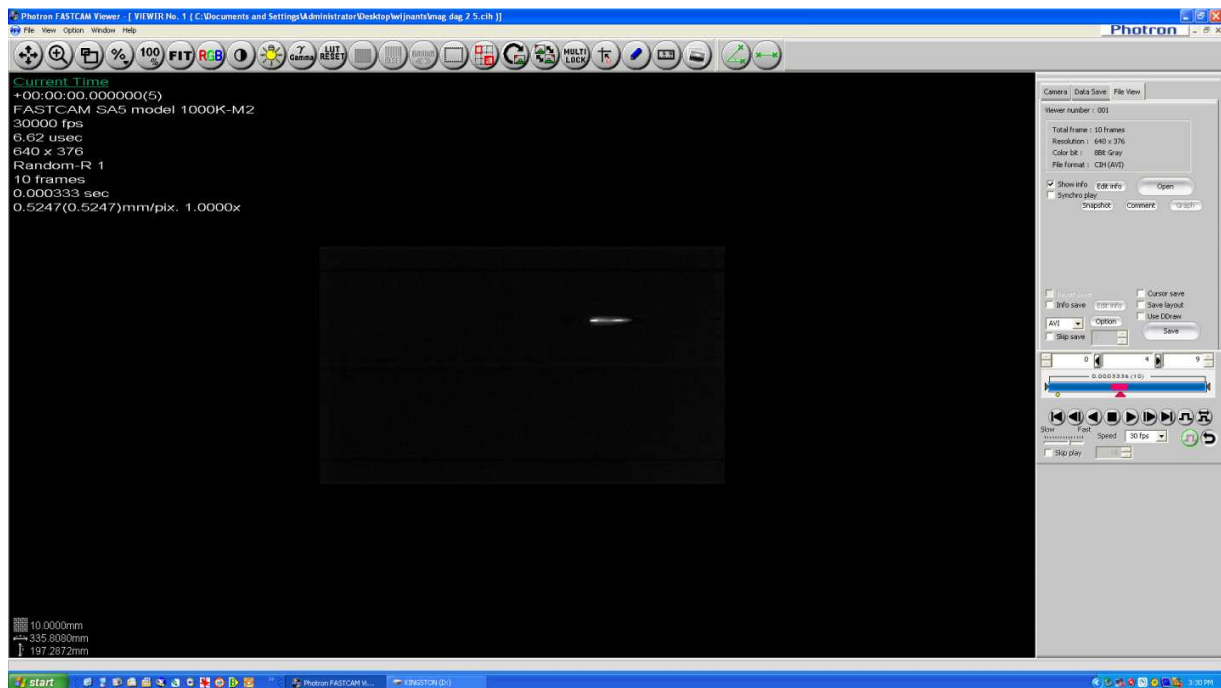
De software beschikt ook over een basispalet aan nabewerkingsfunctionaliteiten, zoals het aanbrengen van een kruisdraad en het meten van afstanden, na kalibratie weliswaar. Deze functionaliteiten vallen buiten het bestek van deze handleiding.

Photron

¹ Brian Speyer, persoonlijke communicatie, 14 april 2014

Handleiding bij het Realtime Trigger Systeem

De 'Photron FASTCAM Viewer' software die bij de Photron hogesnelheidsvideocamera is geleverd [figuur 12] draait altijd op een aparte PC. Om de camera te verbinden met deze PC moet er een ethernet kabel gebruikt worden. Omdat de Photron meerdere frames na mekaar kan opnemen, is het niet mogelijk om een flitser gebruiken. In de plaats van een flitser wordt er continue belichting gebruikt. Dit elimineert meteen ook het probleem van een trigger naar de flitser te moeten sturen die rekening houdt met de preflash time. Het nadeel hiervan is wel dat de sluitertijd nooit zo klein kan worden als bij de Hadland SVR2, omdat een continue lichtbron nooit de lichtintensiteit van een flitser kan evenaren.



Figuur 12: Photron FASTCAM Viewer software

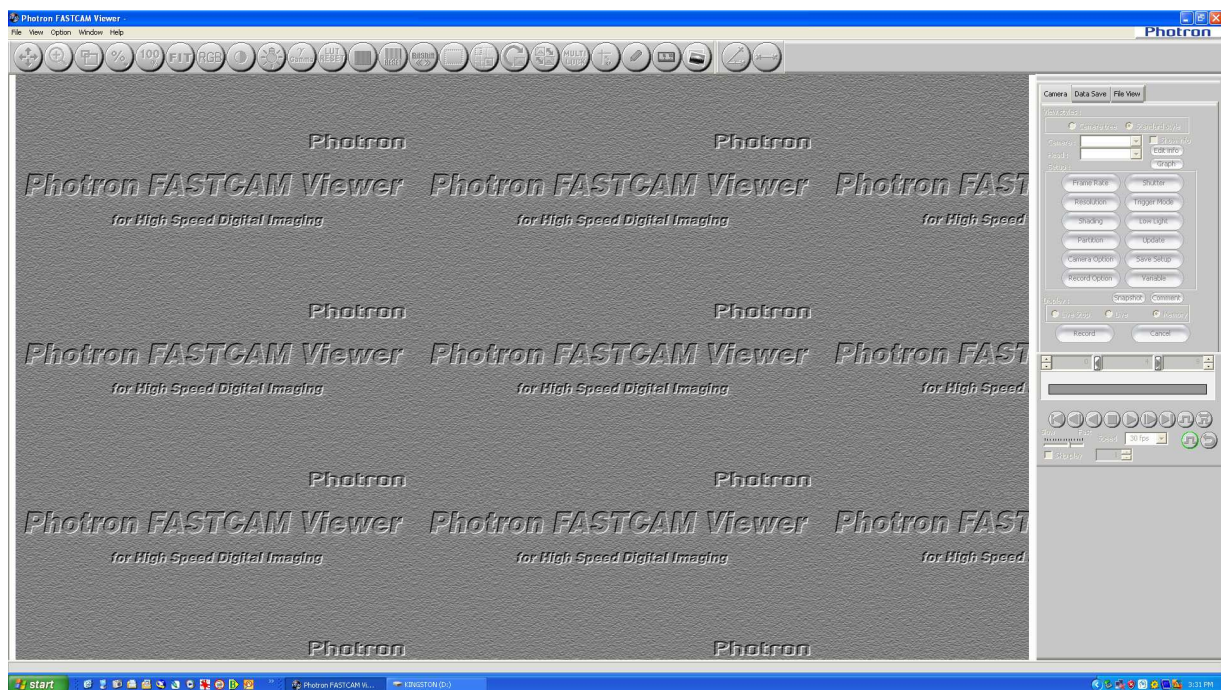
Eenmaal de software opgestart is, zal deze automatisch verbinding zoeken met de Photron camera. Deze verbinding wordt gemaakt aan de hand van een ethernet kabel die beide toestellen verbindt. Van zodra de connectie gelegd is, is het mogelijk om verschillende parameters van de videocamera in te stellen. Bij de Photron is het mogelijk om een framerate (aantal beelden / seconde) in te stellen, en deze parameter is een maat voor de sluitertijd: hoe hoger de framerate, hoe kleiner de sluitertijd en hoe minder *motion blur* er dus gaat optreden (Er is dan wel meer licht nodig). Ook daalt de resolutie als de framerate stijgt. Het is dus best om de framerate zo hoog mogelijk te kiezen, in functie van de hoeveelheid licht en de gewenste resolutie.

Handleiding bij het Realtime Trigger Systeem

Ook is het mogelijk om in te stellen hoeveel frames de Photron zal opslaan, en waar de trigger ligt binnen dit interval van frames, uitgedrukt in procent. Het tijdsinterval waarin de camera dus filmt is gelijk aan het aantal frames gedeeld door het aantal frames per seconde. Indien het aantal frames op 1 ingesteld wordt, functioneert de Photron als een fotocamera. In de praktijk filmt de camera continu, en de trigger bepaalt slechts welk interval (of frame, in geval er slechts 1 frame gekozen is) hij opslaat en doorstuurt naar de PC. Als we bijvoorbeeld ervoor kiezen om 20000 frames te filmen en de trigger ligt op 50%, dan zullen 10000 frames vlak voor de trigger en 10000 frames vlak na de trigger opgeslaan en doorgestuurd worden.

Verder moet ook nog de trigger modus ingesteld worden op *"random reset"*: van zodra de camera een puls krijgt, zal hij dan zijn interne klok resetten omdat de eerstvolgende stijgende flank dan sneller komt, en zo er minder tijdsvertraging zit tussen de *"trigger in"* en het moment waarop de camera effectief getriggerd wordt.

Eenmaal alle instellingen correct zijn ingesteld, kan men op *"record"* drukken en zal de camera beginnen te lopen. Elke keer als de camera een trigger krijgt zal hij nu één tijdsinterval opslaan, de lengte van dit tijdsinterval hangt af van wat door de gebruiker is gedefinieerd. Pas als de opname gestopt wordt zal hij zijn frames doorsturen naar de PC. De instellingen die zonet besproken zijn, zijn allemaal zichtbaar op figuur 13.

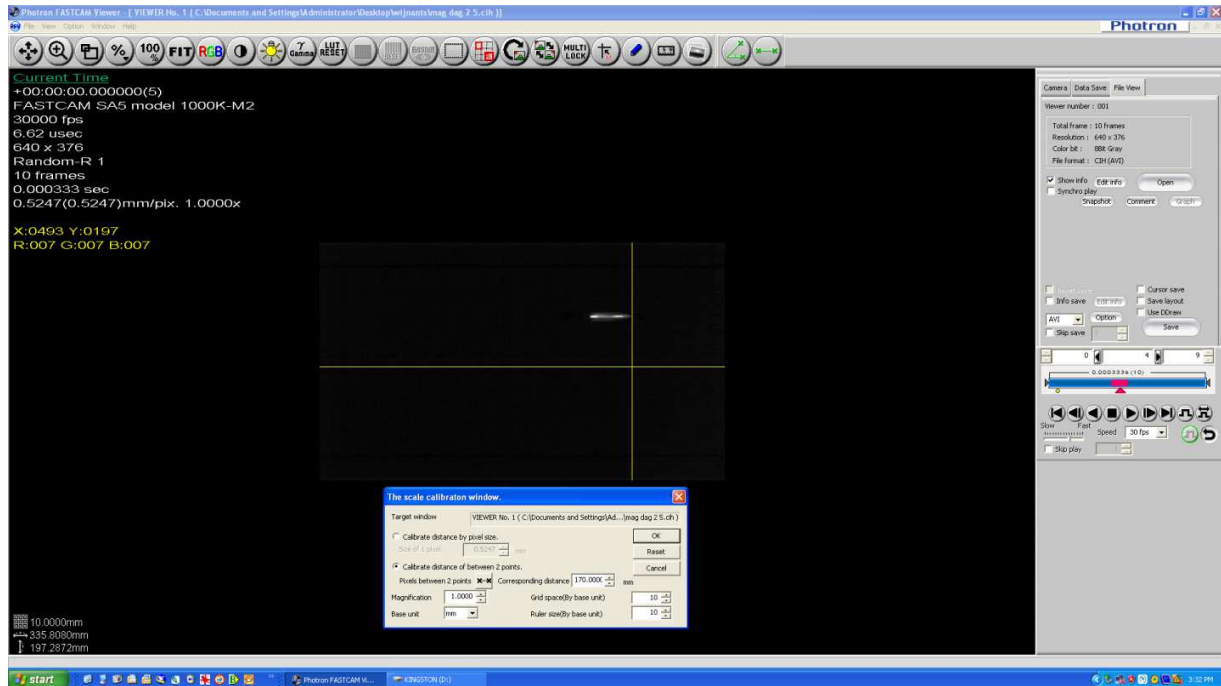


Figuur 13: Photron FASTCAM Viewer software met rechts het scherm om alle parameters in te stellen

Handleiding bij het Realtime Trigger Systeem

Het is dus mogelijk om 30 projectielen na elkaar af te vuren zonder in de tussentijd te moeten ingrijpen op de Photron hogesnelheidsvideocamera.

Verder beschikt deze software ook nog over de mogelijkheid om afstanden te meten (na kalibratie weliswaar) en om een kruisdraad in te stellen. Zo kan bijvoorbeeld de lengte van het projectiel in vlucht gemeten worden. [figuur 14]



Figuur 14: Photron FASTCAM Viewer software met kruisdraad en kalibratiescherm zichtbaar.

De opgenomen data kan als een film (.avi) geëxporteerd worden of als een reeks foto's (.jpg of .png) afhankelijk van wat de gebruiker nodig heeft. Ook wordt er telkens een .cih file mee geëxporteerd, dit bevat alle parameters zoals sluitertijd, framerate, resolutie, trigger mode, etc... Het is belangrijk om op te merken dat de .cih en de .avi file dezelfde naam moeten hebben, anders kan de .cih niet meer refereren naar de .avi file en zal deze niet meer werken.

6 Extra informatie

6.1 De minimale afstand tussen lichtscherm en camera bepalen

Indien het nodig is om de camera zo dicht mogelijk bij het lichtscherm te plaatsen, is het best om de minimale afstand zelf te berekenen in functie van de snelheid van het projectiel en de berekeningstijd.

Hiervoor is het best om eenmaal te vuren terwijl de FPGA al loopt, en daarna de berekeningstijd in microseconden af te lezen uit de software [figuur 15]. Door vervolgens deze tijd te vermenigvuldigen met de snelheid in m/s en te delen door 1000 bekomen we de minimale afstand in millimeter.

Om zeker te zijn is het best om een marge van ongeveer 20% te nemen. Indien we met projectielen met verschillende snelheden gaan vuren, moet deze berekening uitgevoerd worden voor het projectiel met de hoogste snelheid.

In de praktijk zal de minimale afstand ongeveer rond de 20cm bedragen voor een projectiel dat 800m/s vliegt. Bij tragere projectielen zal de afstand natuurlijk nog verkorten.

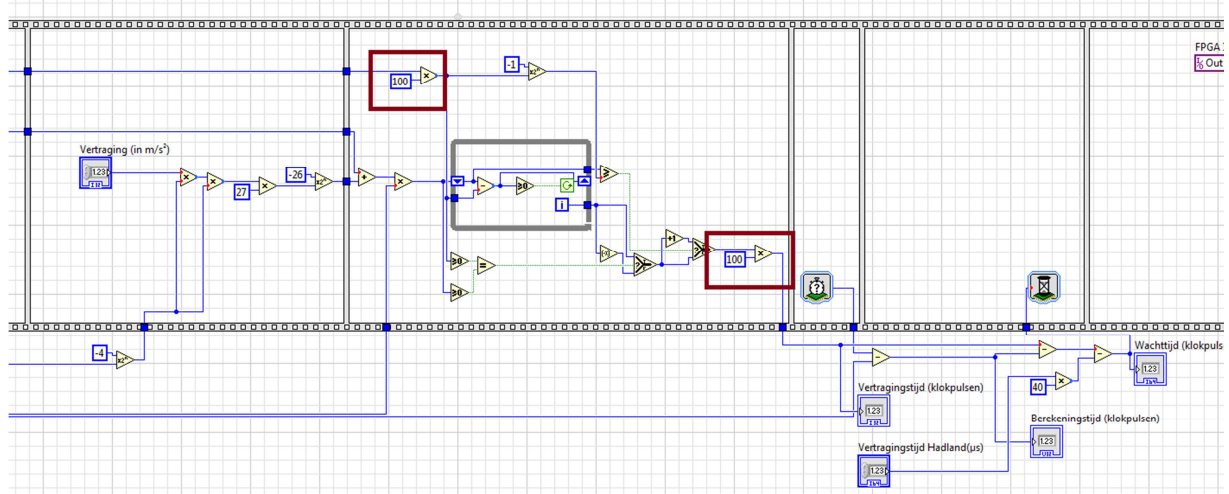


Figuur 15: weergave van de berekeningstijd in de PC software van LabVIEW

6.2 De minimale afstand tussen lichtscherm en camera verkleinen

Omdat een FPGA alleen met integers kan werken, kan deze geen eenvoudige deling uitvoeren. In de software is dit opgelost door met een ‘while loop subtraction’ te werken. Om deze loop sneller te laten uitvoeren, wordt de noemer met een factor 100 vermenigvuldigd voor de deling, en de uitkomst met een factor 100 na de deling. Zo heeft dit geen invloed op het uiteindelijke resultaat, maar wel op de snelheid van de berekening. Hiertegenover staat wel dat het een negatieve invloed heeft op de nauwkeurigheid.

Deze factor 100 kan vrij veranderd worden, zolang het getal voor en na de deling maar identiek is. Zo kan er gekozen worden voor een snellere, minder nauwkeurige berekening of voor een trage maar meer nauwkeurige berekening. Het interval waartussen het RTTS goed werkt is ongeveer tussen een factor 10 en een factor 1000. Hoe groter de factor, hoe sneller de berekening. Op deze manier kan de minimale afstand nog met ongeveer de helft verminderd worden, maar deze reductie hangt ook af van andere factoren, die niet beïnvloedbaar zijn. Deze factor is twee keer vermeldt in de volledige FPGA code en moet dus op beide plaatsen veranderd worden. De locatie van deze factor is gemarkeerd met rood in figuur 16.



Figuur 16: de locatie van de te veranderen getallen in de FPGA code

6.3 Centreren van het projectiel op de foto

Zelfs als alle afstanden en hoeken nauwkeurig afgemeten zijn, is het nog steeds mogelijk dat er een zogenaamde standfout optreedt: deze fout is altijd aanwezig en zal niet veranderen, zelfs als andere parameters zoals de snelheid veranderen. De oorzaak van deze fout is meestal een slecht opgemeten afstand tussen het lichtscherm en de camera.

Omdat het moeilijk is om de camera of lichtscherm nog te verplaatsen nadat alles opgemeten is, is het eenvoudiger om de afstand die ingegeven is in de FPGA te veranderen. Hiervoor moet er eerst een foto genomen worden en hierop moet dan de standfout bepaald worden. Met de Photron software kan dit als er vooraf een kalibratie uitgevoerd is (de software weet dan hoeveel micrometer één pixel is), maar dit kan ook manueel indien we de lengte van het projectiel kennen.

Als we zien dat het projectiel zich teveel naar links bevindt op de foto, dan moet de ingegeven x-waarde groter worden. Indien we zien dat het projectiel zich rechts van het centrum bevindt, moeten we de x-waarde verkleinen.

Het is wel belangrijk om op te merken dat deze methode ENKEL gebruikt mag worden om een standfout (meetfout) te corrigeren. Indien blijkt dat de fout veranderlijk is in functie van de snelheid van het projectiel, ligt de oorzaak ergens anders en zal deze oplossing ook geen soelaas brengen.

7 Probleemoplossing

7.1 Het systeem detecteert geen ingangspulsen

Meestal is dit probleem te wijten aan een slechte verbinding. Een eerste oplossing is om te kijken of alles correct aangesloten is:

- Zijn alle coaxiale kabels correct aangesloten?
- Steken alle stopcontacten in (PXI, lichtschermen, videocamera,...) ?
- Maken de uiteinden van de coaxiale kabels die vastgeschroefd zijn in de break-out box contact?

Als bovenstaande oplossingen het probleem niet kunnen verhelpen, kan het helpen om de draden die in break-out box vastgeschroefd zijn los te maken en opnieuw aan te sluiten. Zorg er ook zeker voor dat de schroeven goed vastgedraaid zijn.

Ten slotte kan met behulp van een trigger box een virtueel ingangssignaal aangelegd worden. Indien blijkt dat het systeem deze wel detecteert dan ligt de fout bij de meetschermen. Mogelijk is de interne zekering gesmolten of is er een ander technisch defect.

Indien de FPGA de ingangspulsen van de trigger box ook niet kan detecteren, ligt het defect waarschijnlijk aan een kapotte kabel of slechte aansluiting.

7.2 Het systeem berekent alles correct maar de camera wordt niet *getriggerd*

Meestal is dit probleem te wijten aan een slechte aansluiting van de coaxiale kabel die de uitgangspuls moet doorgeven. Controleer zeker de aansluiting in de break-out box, en schroef desnoods de draden los en verbind ze opnieuw om zeker te zijn dat ze goed contact maken. Het is ook cruciaal dat de coaxiale kabel aan de juiste ingang van de camera aangesloten is. Bij de Photron is dit '*TTL Trigger IN*' en bij de Hadland is dit '*Trigger in*'.

Indien het probleem nog altijd niet opgelost is na bovenstaande stappen, is het raadzaam om de coaxiale kabel te verbinden met de coaxiale aansluitingen op de PXI en deze ingangen binnen te lezen met de '8 kanaalsdrukmeting_2012' VI die aanwezig is op de PXI. Als deze een puls detecteert op het uitgangskanaal zit de fout bij de instellingen of aansluiting van de camera. Als deze geen puls detecteert is er waarschijnlijk kabelbreuk opgetreden of is de software fout geconfigureerd waardoor deze geen uitgangspuls geeft: dit gebeurt bijvoorbeeld als de afstand van lichtscherm tot camera kleiner is dan de minimale afstand.

7.3 Het systeem voert wel een berekening uit maar de uitkomst is foutief

Indien het systeem bijvoorbeeld 0 of een negatieve waarde teruggeeft voor de wachttijd is het een eerste zaak om te kijken of alle kabels correct aangesloten zijn, zie ook paragrafen 4.1 en 6.2. Indien het probleem hiermee niet verholpen is, is het waarschijnlijk zo dat de FPGA buiten zijn bereik gaat: de FPGA rekent met *64 bits signed integers*, en van zodra de waarde van een integer groter dan 9.223.372.036.854.775.807 wordt, wordt deze negatief.

Een gevolg van de while loop subtraction is dat de vi alleen goed werkt binnen een beperkt bereik. In de praktijk komt het er op neer dat de snelheid van het projectiel ongeveer tussen de 10m/s en de 10000m/s moet zitten. Normaal is dit natuurlijk geen probleem, maar als één lichtscherm bijvoorbeeld een foute puls geeft waardoor het tijdsinterval 42 seconden is in plaats van enkele milliseconden, is het mogelijk dat het systeem foute berekeningen uitvoert. Ook een aanpassing aan de code van de vi kan dit als oorzaak hebben.

De beste oplossing is om met behulp van 'probes' te kijken hoeveel de waarde van elke parameter is op elk punt van de berekening. Zo kan vaak snel ontdekt worden waar de fout zich bevindt: als we twee grote positieve getallen met elkaar vermenigvuldigen maar een negatief getal uitkomen is het meteen duidelijk waar de fout zich bevindt.

7.4 Het systeem neemt wel een foto maar de foto maar de foto is donker

Photron

Indien bij de Photron de foto donker is, is het makkelijk op te lossen door de framerate te verlagen. Hierdoor zal de sluitertijd vergroten en zal de lens meer tijd hebben om licht op te vangen waardoor de foto helderder zal worden. Meer lichtbronnen plaatsen is natuurlijk ook altijd een oplossing.

In uitzonderlijke gevallen kan een lichtbron zelfs zo opgesteld worden om recht in de lens te schijnen, maar dan moet er wel een blad kalkpapier voor de lichtbron aangebracht worden om het licht diffuus te maken.

Hadland SVR2

Er zijn twee grote oorzaken mogelijk waarom de foto donker is: ofwel is er niet genoeg licht aanwezig, ofwel is de timing van de flitser slecht gekozen.

Om zeker te zijn wat exact het probleem is, is het best om eerst de sluitertijd heel groot kiezen, en nog een foto maken. Indien de foto niet meer donker is, is het probleem opgelost en moet de sluitertijd gewoon groter gekozen worden of moeten er extra lichtbronnen worden bijgeplaatst.

Als de foto dan nog steeds te donker is, is de timing van de flitser waarschijnlijk fout gekozen. Het is belangrijk om de juiste preflash time in te stellen, ofwel op de FPGA ofwel in de software van de SVR2 zelf (afhankelijk van de gekozen methode). De preflash time hangt af van model tot model flitser en staat vaak vermeld in de datasheet.

Als we de FPGA de preflash trigger laten geven, kan het ook voorkomen dat de flitser toch niet correct afgaat, zelfs indien de preflash tijd gerespecteerd is. De enige mogelijke oplossing dan is om toch de Hadland camera de sturing van de flitser te laten doen.

Handleiding bij het Realtime Trigger Systeem

Bijlage B

Uitgebreide handleiding bij het RTTS

Uitgebreide handleiding bij het RTTS

Departement wapensystemen & ballistiek

Wijnants Jurgen

Olt KBO

Inhoudsopgave

1 Lijst met figuren	3
2 Verklarende woordenlijst	4
3 Inleiding	6
4 Bespreking code	6
4.1 De FPGA code	7
4.1.1 Bespreking FPGA code	7
4.1.3 FPGA in- en uitgangen	11
4.2 PC code	11
5 Gedetailleerde bespreking code.....	13
5.1 Vertragingstijd van een projectiel met constante snelheid	13
5.2 Vertragingstijd van een projectiel onderhevig aan een versnelling.....	14
5.3 Verschillende soorten delingen bij een FPGA	15
5.4 Verhouding teller/noemer bij while loop subtraction.....	16
6 Mogelijke uitbreidingen	17
6.1 Analoge signalen	17
6.2 Doppler radar	17
6.3 Onderscheppingsprobleem	17

1 Verklarende woordenlijst

Array: Engelstalige benaming voor een rij (of reeks). Dit is een reeks data die geïndexeerd kan worden.

FPGA: Field Programmable Gate Array, een geïntegreerde schakeling bestaande uit logische componenten die geschakeld kunnen worden als logische poorten (AND,XOR,NAND,...).

IO: Input / Output oftewel ingang / uitgang. Benaming om het geheel van aansluitingen die functioneren als digitale of analoge in- of uitgang aan te duiden.

NI: National Instruments, Amerikaans bedrijf dat eigenaar is van de LabVIEW software en bijbehorende hardware fabriceert.

PXI: PCI eXtensions for Instrumentation : een standaard van National Instruments om meettoestellen aan te sluiten via de PCI aansluiten van een PC. Komt meestal voor onder de vorm van een kast met daarin plaats voor één PC en meerdere meettoestellen die dan automatisch verbonden worden met de PC via een rail in de achterzijde van de kast.

Realtime Trigger Systeem (RTTS) : een meetsysteem ontworpen om in realtime (dus terwijl de gebeurtenis plaatsvindt) metingen te verrichten, deze meetdata te verwerken en uiteindelijk een signaal uit te sturen.

Trigger: Spanningspuls die het bevel geeft aan een camera om zijn opname te starten.

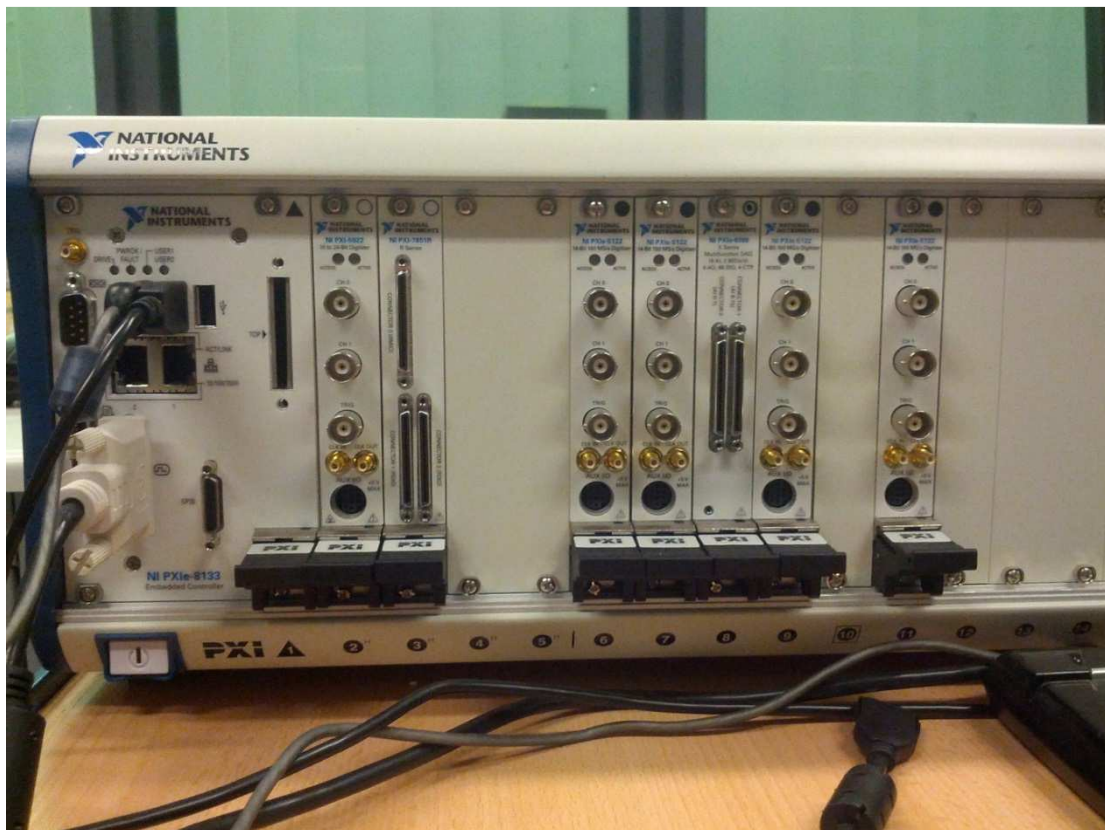
VI: Virtual Instrument, de naam voor een programma dat in LabVIEW geschreven is. Een VI bestaat steeds uit twee delen: een *“front panel”* dat knoppen, grafieken, status van bits, ... bevat en een blokdiagram waar de verwerking van data gebeurt aan de hand van blokken die verbonden worden met virtuele draden.

2 Inleiding

Deze handleiding heeft als doel de de achterliggende code van het ‘Realtime Trigger System’ (RTTS) uit te leggen om zo aanpassingen en uitbreidingen mogelijk te maken. In eerste instantie is het aan te raden om de handleiding bij ‘Werking van het RTTS’ te lezen. Deze beschrijft de hardware componenten die samen het RTTS vormen, en beschrijft de uitwendige werking van de software. Deze handleiding is een uitbreiding hierop en beschrijft meer in detail de LabVIEW code en hoe deze aangepast kan worden om ook andere functionaliteiten te voorzien.

3 Bespreking code

In deze paragraaf wordt de code besproken om zo de werking van het systeem beter te begrijpen. De code is gestructureerd op twee niveau's: een stuk code (onder de vorm van een VI in LabVIEW FPGA module) dat op de FPGA draait en een ander stuk dat op de PC draait (ook onder de vorm van een VI in LabVIEW). Deze twee hardwarecomponenten (PC en FPGA) zitten samen vervat in één PXI [figuur 1] (eXtensions for Instrumentation) van National Instruments. De PC dient ook als programmeertoestel van de FPGA. Er wordt geprogrammeerd met LabVIEW dat ook de compilatie voor zijn rekening neemt.

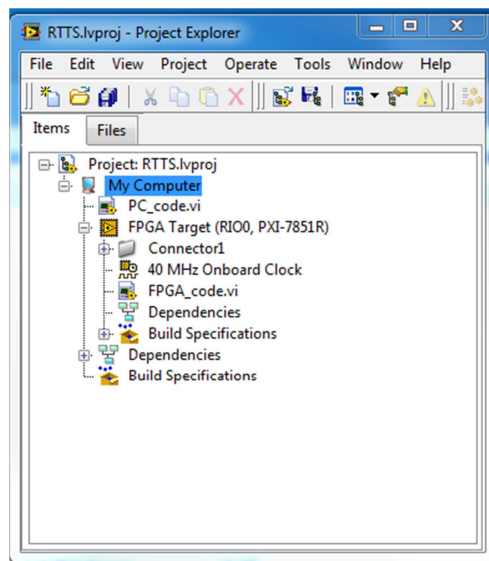


Figuur 1: de PXI met in slot 1 (geheel links) de PC module en in slot 3 de FPGA module.

De software is op twee verschillende niveau's geschreven omdat deze aan twee verschillende eisen moest voldoen. Allereerst was het zeer belangrijk dat de software razendsnel werkt, en daarom is het absoluut noodzakelijke deel op de FPGA geprogrammeerd om zo een maximale snelheid te bekomen.

Anderzijds moest de software ook gebruiksvriendelijk en bijvoorbeeld bepaalde parameters berekenen en wegschrijven in een tekstbestand. Dit stuk van de code is geprogrammeerd op de PC omdat deze hier beter mee overweg kan, en omdat snelheid toch niet zo belangrijk is dan. Het is wel zo dat de twee delen continu met elkaar in verbinding staan. Indien gewenst kan het deel software op de FPGA wel zelfstandig werken.

Figuur 2 geeft de programmastructuur visueel weer: enerzijds is er een VI die op de PC draait, en anderzijds is er een VI die op de FPGA draait. Ook de verschillende in- en uitgangen (IO's) zijn hier weergegeven.



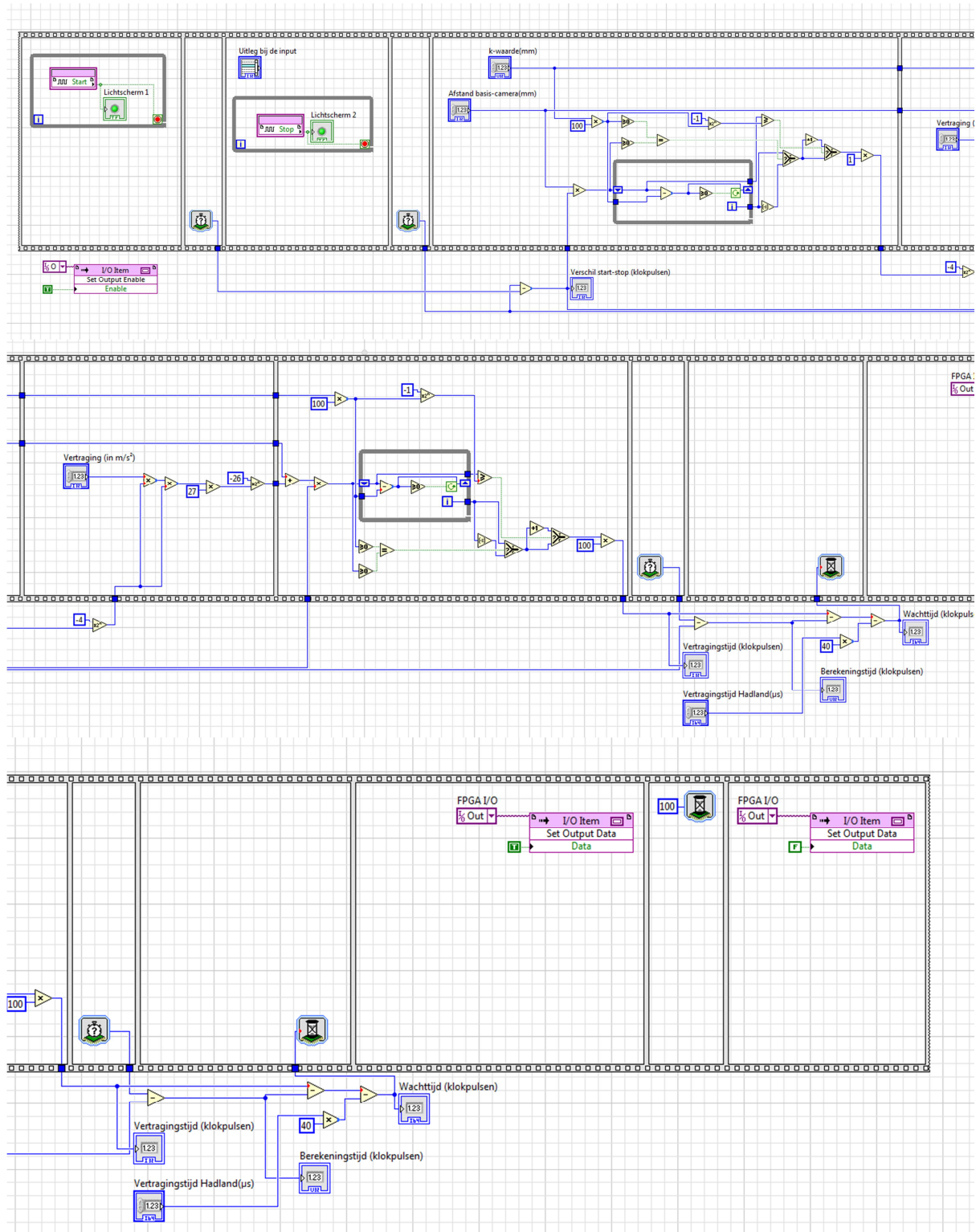
Figuur 2: programmastructuur in LabVIEW

3.1 De FPGA code

3.1.1 Bespreking FPGA code

De hoofdstructuur van de FPGA code bestaat uit een lange “*flat sequence structure*” (visueel weergegeven door een afgerold fotorolletje) die ervoor zorgt dat elk “*frame*” van het fotorolletje chronologisch wordt uitgevoerd. Als de uitvoering van het laatste frame gedaan is stopt de VI of begint hij opnieuw van voor af aan, indien de VI in de mode “*Run Continuously*” staat. Hieronder wordt de code weergegeven in figuur 3. Het lange lint is in drie delen gekapt om de code beter zichtbaar te maken.

Uitgebreide handleiding bij het Realtime Trigger System



Figuur 3: visuele voorstelling van de FPGA code in LabVIEW

In het eerste frame wordt de digitale ingang 0 (DIO0) uitgelezen. De ingang wordt omgeven door een while loop zodat de FPGA continu controleert of de ingang al dan niet hoog is (sampling frequentie = 40Mhz). Met deze ingang is het eerste lichtscherm waar het projectiel door passeert verbonden. Van zodra hij een puls detecteert stopt de while loop en springt de FPGA naar frame 2.

Onder het eerste frame is een klein stukje code te zien waar we DIO2 als output 'enablen'. Dit is nodig omdat alle 34 digitale kanalen als in- of uitgang gebruikt kunnen worden ,maar standaard als ingang geconfigureerd zijn. DIO0 en DIO1 kunnen we dus gewoon standaard als ingang laten staan, maar DIO2 moet als uitgang geconfigureerd worden om correct te werken.

In het tweede frame wordt een timer gestart (deze telt in klokpulsen) en één klokpuls later begint de FPGA met de uitvoering van de code in het derde frame.

Het derde frame is identiek aan het eerste frame, behalve dan dat hier DIO1 ingelezen wordt in plaats van DIO0. DIO1 is verbonden met het tweede lichtscherm waar het projectiel door passeert.

In het vierde frame wordt een (andere) timer (opnieuw in klokpulsen) gestart en begint de FPGA met de uitvoering van frame 5. Ook berekent hij in tussentijd het verschil tussen beide klokken. Dit getal, uitgedrukt in klokpulsen, is de tijd tussen de twee triggers die de FPGA ontvangen heeft, en dus ook de tijd die het projectiel nodig had om de gekende afstand tussen lichtscherm 1 en lichtscherm 2 af te leggen.

In het vijfde frame wordt dan vervolgens het product van de tijd tussen beide lichtschermen en de afstand basis-camera gedeeld door de afstand van beide meetbasissen. Dit geeft ons dan de tijd, in klokpulsen, totdat het projectiel zich voor de camera bevindt. De wiskundige verklaring hiervoor is terug te vinden in paragraaf 5.1. Meer uitleg over deze deling onder de vorm van een while loop subtraction is terug te vinden in paragraaf 5.3.

Frame 6: de tijd die we nu gevonden hebben zou de tijd zijn die het projectiel nodig heeft indien het niet zou vertragen. Natuurlijk zijn alle projectielen onderhevig aan de luchtweerstand, en zullen ze dus een vertraging ondervinden. Daarom berekenen we de afstand die het projectiel minder zal afleggen ten gevolge van deze vertraging met de formule $x = 0,5 \cdot a \cdot t^2$ waarbij x de afgelegde weg is, a de vertraging en t de tijd voor hoelang het projectiel deze vertraging ondergaat. We bekomen dus een (virtuele) extra afstand die het projectiel moet afleggen om zijn vertraging te compenseren.

In frame 7 wordt opnieuw dezelfde deling uitgevoerd als in frame 5, alleen vervangen we de afstand basis-camera door de som van deze afstand met de extra (virtuele) afstand veroorzaakt door de vertraging. Uiteindelijk komen we dan opnieuw een tijdsvertraging uit, uitgedrukt in klokpulsen, die groter of gelijk is aan de tijdsvertraging die we in frame 5 berekend hadden.

In frame 8 start opnieuw een teller, in klokpulsen, die meteen afgetrokken wordt van de teller die in frame 4 gestart werd, vlak voor de twee delingen uitgevoerd werden. Het verschil tussen deze twee tellers is de berekeningstijd in klokpulsen. Deze tijd wordt dan nog eens afgetrokken van de berekende vertragingstijd, om zo de tijd die de FPGA nog moet wachten te bekomen. Indien ingevoerd door de gebruiker wordt er ook nog eens rekening gehouden met de interne vertragingstijd van de hadland, die ook van de totale vertragingstijd wordt afgetrokken (zie ook de handleiding bij het gebruik van het RTTS).

In frame 9 is er een dan een “*wait*” functie die exact het aantal klokpulsen wacht dat zonet berekend is door de FPGA.

Vervolgens wordt in frame 10 DIO2 hoog gemaakt, dit is de uitgang die de trigger voorziet naar de camera.

In frame 11 wacht de FPGA 100 microseconden (niet klokpulsen) en in frame 12 wordt DIO2 tenslotte weer laag gemaakt. De 100 microseconden van frame 11 is dus eigenlijk de pulsbreedte.

Hierna stopt het programma en wacht op instructies van de gebruiker.

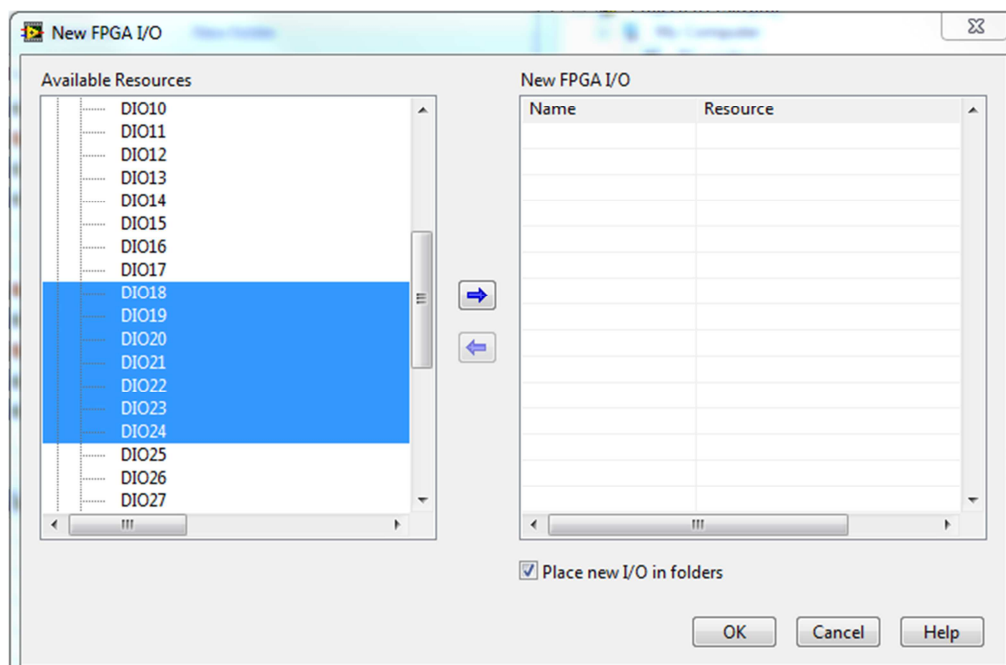
3.1.3 FPGA in- en uitgangen

De FPGA module beschikt over drie 68-pins connectoren waarlangs alle communicatie verloopt. Op elk van deze drie connectoren kan een SCB-68 Breakout box worden aangesloten. In deze breakout box zijn telkens 68 klemmen aanwezig zodat draden eenvoudig aangesloten kunnen worden.

Hoewel alle connectoren er identiek uitzien, is er toch een verschil. Connector 0 beschikt over 15 digitale in- of uitgangen (zelf te kiezen door de gebruiker), 7 analoge ingangen en 7 analoge uitgangen. Connector 1 en 2 daarentegen beschikken ieder over 33 digitale in- of uitgangen (eveneens zelf te kiezen door de gebruiker). Ook de nummering / aansluitvolgorde verschilt dus tussen de verschillende connectoren onderling. Daarom is het noodzakelijk om de datasheet te raadplagen, die vervat zit als bijlage bij deze thesis.

Om analoge of digitale kanalen te gebruiken in een FPGA programma is het eerst noodzakelijk om deze eerst toe te voegen aan het FPGA project. Dit kan door rechts te klikken op de projectnaam in het overzichtsscherm en te klikken op “*Configure FPGA IO*” [figuur 4]. Individuele IO's kunnen dan toegevoegd worden aan het LabVIEW project zodat het mogelijk wordt om ze op te roepen in de VI's geprogrammeerd op de FPGA.

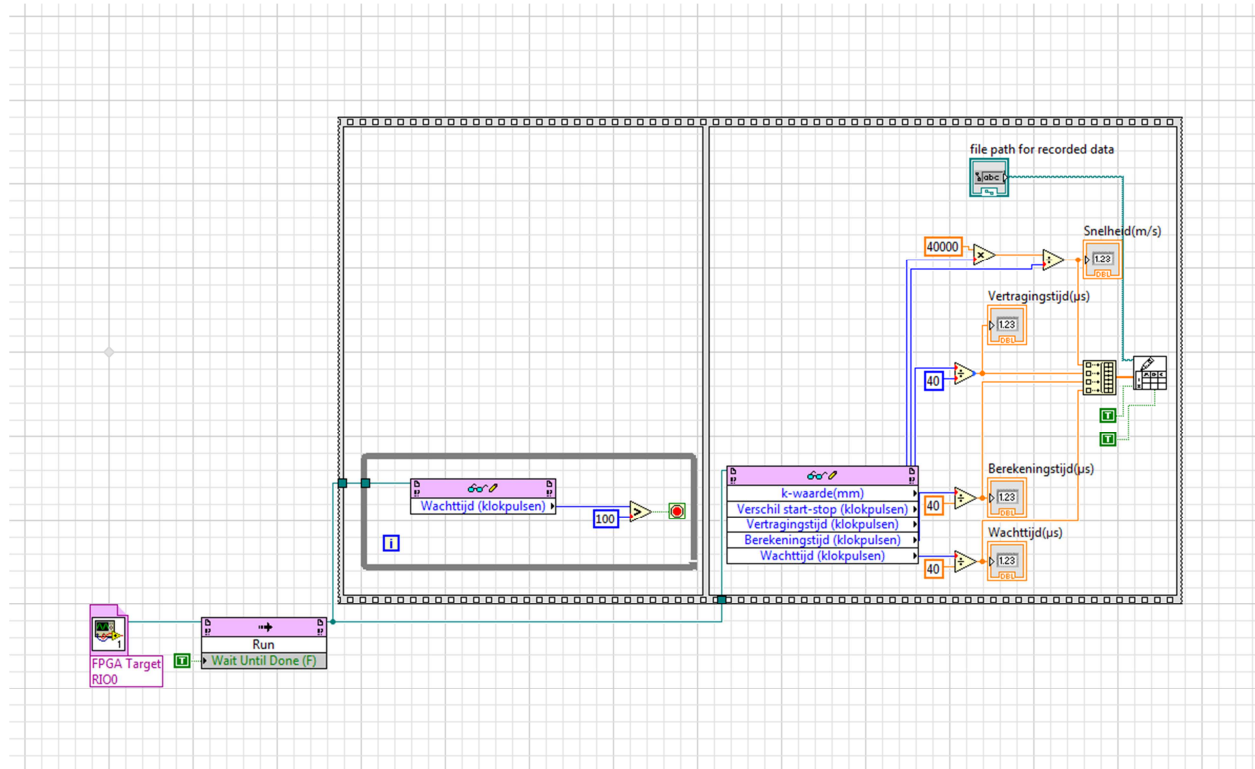
De analoge kanalen zijn op voorhand reeds gedefinieerd als zijnde in- of uitgangskanaal. Het is dus belangrijk om de correcte analoge kanalen toe te voegen aan het LabVIEW project. Alle digitale kanalen zijn configureerbaar door de gebruiker als in- of uitgang. Standaard zijn deze als ingang geconfigureerd. Indien de gebruiker een digitaal uitgangskanaal tot zijn beschikking wilt hebben, moet hij deze eerst als zodanig configureren door het blokje “*Configure FPGA IO*” op te roepen en daar “*Set output Enable*” te verbinden met een boolean die hoog is.



Figuur 4: interface om de FPGA IO's te configureren

3.2 PC code

Het tweede deel van de code bestaat uit een VI die op de PC draait die ingebouwd zit in de PXI. Een overzicht van deze code is gegeven in figuur 5.



Figuur 5: overzicht PC code

Linksonder op figuur 5 is te zien hoe deze VI wacht tot de code op de FPGA volledig uitgevoerd is alvorens te starten. Omdat de code op de FPGA continue zijn ingangspoorten zit af te scannen (en dit dus niet eenmalig doet) is er in het links frame ook een stukje code voorzien dat ervoor zorgt dat het programma op PC alleen uitgevoerd wordt wanneer de wachttijd meer dan 100 klokpulsen bedraagt. In de praktijk zorgt dit ervoor dat de code op PC alleen wordt uitgevoerd wanneer de FPGA ook effectief een berekening heeft uitgevoerd.

In het rechtse frame zit de code die de functionaliteit van deze VI bevat: hier worden een aantal abstracte doch nuttige parameters uit de FPGA ingelezen en vertaald naar parameters die gebruikt kunnen worden voor de verdere verwerking. Zo wordt ondermeer de berekenings-, vertraging- en wachttijd omgerekend van klokpulsen naar microseconden. Ook wordt in deze VI de snelheid in m/s berekend, omdat dit niet gebeurt bij de berekeningen van de FPGA.

Ten slotte worden geheel rechts al deze waarden gebundeld in een “array” en automatisch weggeschreven op een plaats in het geheugen van de PC die de gebruiker zelf kiest. Indien er meerdere metingen na mekaar gebeuren wordt de nieuwe data gewoon onder de oude data weggeschreven.

4 Gedetailleerde bespreking van de code

4.1 Vertragingstijd van een projectiel met constante snelheid

Het hoofddoel van de FPGA is om de tijdsvertraging te berekenen wanneer het projectiel zich exact voor de camera bevindt. Eerst negeren we de vertraging a . Het projectiel legt dan dus een eenparig rechtlijnige beweging af bij benadering. Er zijn drie parameters die nodig zijn om de vertragingstijd t te berekenen: de afstand tussen de basis en de camera(x), de afstand tussen de basissen(k), en de tijd die het projectiel nodig heeft om de afstand tussen beide basissen af te leggen(t).

Met bovenstaande gegevens is het mogelijk om de snelheid van het projectiel te berekenen, en vervolgens de tijd die het projectiel nodig heeft om dan de afstand basis-camera af te leggen rekening houdend met zijn snelheid. Deze methode is het meest intuïtief maar niet ideaal omdat er dan twee delingen noodzakelijk zijn om ons resultaat te bekomen, en een deling is complex en traag om uit te voeren met een FPGA en dit willen we dus liefst zoveel mogelijk vermijden.

Een andere aanpak is om de verhouding tussen de x en k te beschouwen. Stel bijvoorbeeld dat $x = 2k$. Dan is het onmiddellijk duidelijk dat de FPGA tweemaal zolang moet wachten als de tijd die het projectiel nodig had om de afstand k af te leggen, omdat x nu eenmaal dubbel zo lang is als k . Meestal gaat de verhouding x/k natuurlijk geen natuurlijk getal zijn, maar op deze manier kan de vertragingstijd wel berekend worden met behulp van één deling in plaats van twee. De formule voor de vertragingstijd is dan $\frac{t*x}{k}$.

4.2 Vertragingstijd van een projectiel onderhevig aan een versnelling

Als we nu wel rekening houden met de vertraging a (= een negatieve versnelling) dan geldt niet meer dat de tijdsvertraging gelijk is aan $\frac{t*x}{k}$. Om een betere waarde voor de tijdsvertraging te vinden moeten we eerst de vertraging van het projectiel kennen. Omdat we over slechts twee infraroodlichtschermen beschikken, is het niet mogelijk om de vertraging in realtime te meten. Daarom moet de gebruiker deze zelf op voorhand ingeven zodat de FPGA hier rekening mee kan houden in zijn berekening. We stellen ook bij benadering dat de vertraging constant is en niet verandert in de tijd.

De invloed van de vertraging op de afgelegde weg is afhankelijk van de tijd die het projectiel nodig heeft om deze weg af te leggen. Omdat we deze tijd pas kennen nadat we de snelheid hebben gemeten in realtime, is het ook pas achteraf mogelijk om de versnelling in rekening te brengen. Daarom berekent de FPGA eerst de vertragingstijd zonder rekening te houden met de vertraging, en berekent hij dan achteraf de invloed van de vertraging, rekening houdend met de reeds berekende vertragingstijd.

Zoals reeds aangehaald berekent de FPGA dus eerst de vertragingstijd alsof $a=0\text{m/s}^2$. Vervolgens berekent hij de invloed van de versnelling op de afgelegde weg volgens de formule $x = a * \frac{t^2}{2}$.

De afstand die we dan vinden zal steeds negatief zijn, omdat een projectiel altijd vertraagt nadat het de loop heeft verlaten. Deze afstand is dus als het ware een extra afstand die het projectiel (virtueel) moet afleggen ten gevolge van zijn vertraging voor het op het gewenste punt zal aankomen. Daarom dat we de absolute waarde van deze extra afstand optellen bij de afstand tussen de basis en camera. Vervolgens voeren we opnieuw de berekening uit dat de vertragingstijd gelijk is aan $\frac{t*x}{k}$, met als enige verschil dat nu x gelijk is aan de afstand basis camera plus de virtuele extra afstand ten gevolge van de vertraging. Uiteindelijk bekomen we dan een tweede, meer correcte, vertragingstijd die steeds groter of gelijk is aan de eerst berekende vertragingstijd.

4.3 Verschillende soorten delingen bij een FPGA

Om de vertragingstijd te berekenen zullen we één of meerdere delingen moeten uitvoeren. Omdat een FPGA standaard met integers werkt, kan deze geen gewone deling uitvoeren zoals we die in de wiskunde kennen. Er zijn drie manieren waarop een FPGA wel een deling kan uitvoeren en alle drie hebben ze hun voor- en nadelen.

Het eerste soort deling kan uitgevoerd worden door een library te downloaden van de site van National Instruments¹. Hierdoor emuleert de FPGA een deling van floating point getallen door complexe berekeningen met integers uit te voeren. Het voordeel van deze methode is de nauwkeurigheid en de eenvoudigheid om te implementeren. Het grote nadeel is de snelheid: om een grote nauwkeurigheid te halen en door de conversie zal het zeer lang duren om een deling uit te voeren. Daarom is deze methode niet geschikt voor de FPGA module.

Het tweede soort deling is een scalering met factor 2^x waarbij x een geheel getal is. Deze methode kan aangevuld worden met een vermenigvuldiging met een factor y , waarbij y een natuurlijk getal is. Op hardware niveau is een vermenigvuldiging niet zo heel complex om uit te voeren, omdat het eigenlijk gewoon y maal een optelling is die de FPGA moet uitvoeren. Het scaleren met een factor 2^x wordt op hardware niveau vertaald door een shift left of shift right operatie en kan dus razendsnel uitgevoerd worden. Het grote voordeel van deze methode is dus de snelheid maar hier tegenover staat wel dat de nauwkeurigheid zeer slecht is omdat $y \cdot 2^x$ vaak een paar % afwijkt van de gewenste schaafactor. Er bestaat een keuzehulp voor y en x die vrij te downloaden is van de website van National Instruments². Deze methode werd gebruikt om de extra tijdsvertraging ten gevolge van de vertraging van het projectiel om te zetten van microseconden naar klokpulsen.

Het derde soort deling is de 'while loop subtraction'. Deze methode voert een deling uit met integers door telkens de noemer af te trekken van de teller en bij te houden hoe vaak dit gebeurd is. Eenmaal de uitkomst kleiner of gelijk aan 0 is, stopt de lus en krijgen we het correcte, afgeronde resultaat. Het is ook mogelijk om indien nodig naar boven af te ronden door de helft van de teller te vergelijken met de rest. De code om deze deling uit te voeren is vrij te downloaden op de site van National instruments³ en het is ook deze code die (weliswaar aangepast) gebruikt wordt om tweemaal de deling uit te voeren in de FPGA code.

Het voordeel van de 'while loop subtraction' is dat deze methode de snelheid van een scalering benadert en toch niet moet onderdoen qua nauwkeurigheid voor een floating point deling. Het nadeel is dat de argumenten die aangehaald werden bij de voordelen alleen maar gelden als de teller en noemer zich correct verhouden. Een verdere uitwerking van de invloed van deze verhouding is terug te vinden in de volgende paragraaf (5.4).

¹ <http://www.ni.com/white-paper/7781/en/>

² <http://www.ni.com/example/29021/en/>

³ <http://www.ni.com/white-paper/3661/en/>

4.4 Verhouding teller/noemer bij while loop subtraction

Zoals reeds aangehaald in paragraaf 5.3 is een nadeel van while loop subtraction dat de verhouding teller noemer ongeveer gekend en constant moet zijn. De reden hiervoor wordt in de volgende alinea's toegelicht.

Stel dat de teller kleiner is dan de noemer: een deling met integers zal dan steeds als resultaat 0 geven, terwijl de echte uitkomst bijvoorbeeld 0,85 is. Deze afwijking is onacceptabel voor bijna alle toepassingen en moet dus vermeden worden.

Stel nu bijvoorbeeld dat de teller twee miljoen keer groter is dan de noemer. In dat geval zal de FPGA dus twee miljoen keer de noemer moeten aftrekken van de teller voordat hij nul bereikt. Deze operatie zal dus ook (mintens) 2 miljoen klokpulsen duren. Zo een lange tijdsduur is opnieuw niet geschikt voor de meeste toepassingen, en zeker niet voor toepassingen zoals dit Realtime Trigger System.

Het is dus duidelijk dat de verhouding teller/noemer ongeveer constant moet zijn. Bij deze specifieke toepassing is dat het geval: de snelheid van een projectiel zal nooit lager dan 10m/s zijn en nooit meer dan 1000m/s. Tussen deze twee uitersten zit slechts een factor 100, waardoor we bijna zeker kunnen zijn dat de verhouding teller/noemer nooit absurde waarden zal aannemen.

Nu is het alleen nog een kwestie van de verhouding teller noemer ongeveer goed te krijgen. De afstand tussen de twee lichtschermen en de afstand van de basis tot de camera ligt in dezelfde grootteorde (enkele meters). De afstand basis camera staat in de teller en wordt vermenigvuldigd met de tijd die het projectiel nodig heeft om de afstand tussen beide lichtschermen af te leggen, uitgedrukt in klokpulsen. Deze tijd zal schommelen tussen duizend en honderdduizend klokpulsen ongeveer. Als een gevolg hiervan zal de teller dus ook duizend à honderdduizend keer groter zijn dan de noemer. Dit is teveel en zou ervoor zorgen dat de berekening veel te lang duurt.

Als oplossing voor dit probleem wordt de noemer vermenigvuldigd met een factor 100 voor de while loop subtraction uitgevoerd wordt. De teller zal dan maar 10 à 1000 keer groter zijn dan noemer. De maximale tijd die de FPGA nodig heeft om de deling uit te voeren is dan ook ongeveer 1000 klokpulsen = 25µs. Na de deling bekomen we dan een resultaat dat 100 keer te klein is. Daarom wordt dit resultaat ook met een factor 100 vermenigvuldigd zodat wiskundig alles nog steeds klopt.

Het is ook mogelijk om deze factor aan te passen om zo een nauwkeurigere of snellere berekening uit te voeren. Door de factor bijvoorbeeld te veranderen van 100 naar 1000 zal de while loop subtraction tien keer sneller uitgevoerd worden, terwijl de nauwkeurigheid met een factor 10 daalt. De totale berekeningstijds van de FPGA zal ook sterk dalen, met niet met een factor tien omdat er nog andere stukken code uitgevoerd worden, wiens snelheid onafhankelijk is van de while loop subtraction.

Andersom is het ook mogelijk om deze factor te verkleinen om zo de nauwkeurigheid op te drijven, ten kosten van de snelheid.

5 Mogelijke uitbreidingen

In volgende paragrafen wordt de aanzet gegeven voor een aantal mogelijke uitbreidingen die gerealiseerd kunnen worden op basis van het huidige, reeds bestaande Realtime Trigger System.

5.1 Analoge signalen

Zoals reeds aangehaald in paragraaf x is het mogelijk om ook digitale signalen in te lezen en / of uit te sturen met de FPGA. Hierdoor is het dus mogelijk om bijvoorbeeld een hoekopnemer, drukmeting, etc... uit te voeren met de FPGA. Door de rekenkracht van de FPGA is het dus ook mogelijk om deze gegevens in realtime te gebruiken om bepaalde processen aan / bij te sturen.

5.2 Doppler radar

Door de uitgang van een doppler radar aan te sluiten op een analoge ingang van de FPGA module, is het mogelijk om deze signalen in realtime binnen te lezen en te gebruiken. Hiervoor beschikt LabVIEW standaard over een Fast Fourier Transformatie (FFT) functie. Deze functie is ook hernomen in de FPGA module van LabVIEW en geoptimaliseerd voor het gebruik met een FPGA.

Mogelijke toepassingen hiervan zijn bijvoorbeeld een studie over actieve beschermingssystemen voor voertuigen, waarbij rekenkracht en rekensnelheid een primordiale rol spelen.

Ook kan eventueel de snelheidsmeting van het projectiel, die momenteel gebeurt aan de hand van twee infraroodlichtschermen, vervangen worden door een snelheidsmeting aan de hand van een doppler radar. Dit zou als voordeel hebben dat ook de vertraging meetbaar is, en dat er eventueel gecompenseerd kan worden voor een afwijking op de baan van het projectiel. Deze methode gaat wel aan nauwkeurigheid inboeten ten opzichte van de methode met de infraroodlichtschermen.

5.3 Onderscheppingsprobleem

Door een kleine aanpassing in de code kan het RTTS ervoor zorgen dat een projectiel in zijn vlucht geraakt wordt door een ander voorwerp, bijvoorbeeld een slinger die zich in een labiele positie bevindt. Deze toepassing leunt ook opnieuw naar een actief beschermingssysteem, waarbij het de bedoeling is om een projectiel (meestal een anti-tank geleide raket) te raken met een stalen plaat die gelanceerd wordt met behulp van een springlading. Dit kan eventueel ook gesimuleerd worden, bijvoorbeeld door een stalen plaat en een veer (in opgespannen toestand) langs de baan van het projectiel op te stellen en de veer los te laten op het juiste moment om zo het projectiel van zijn baan te doen afwijken.

Bijlage C

Datasheet FPGA module 7581R

Last Revised: 2011-03-08 11:32:31.0

NI R Series Multifunction RIO

Integrated Analog and Digital I/O with FPGA Technology



- Onboard FPGA chip, programmable with the LabVIEW FPGA Module
- User-defined triggering, timing, and decision making in hardware with 25 ns resolution
- Up to 8 analog inputs, independent sampling rates up to 750 kHz, 16-bit resolution
- Up to 8 analog outputs, independent update rates up to 1 MHz, 16-bit resolution
- Up to 160 digital lines configurable as inputs, outputs, counters, or custom logic at rates up to 40 MHz
- Direct memory access (DMA) channels for high-speed data streaming
- Implement custom control logic, inline signal processing, and digital communication protocols

Overview

National Instruments reconfigurable I/O (RIO) technology gives you the ability to define your own custom measurement hardware circuitry using reconfigurable field-programmable gate array (FPGA) chips and NI LabVIEW graphical development tools. NI R Series multifunction RIO devices offer the best combination of value and performance by integrating this FPGA technology with eight analog inputs, eight analog outputs, and 96 digital I/O lines, all into a single device that is offered on standard PC form factors such as PCI, PCI Express, and PXI/CompactPCI. Using the LabVIEW FPGA Module, you can create your own hardware personalities for custom data acquisition, high-speed control, digital communications protocols, sensor simulation, hardware-in-the-loop test, and onboard signal processing without in-depth knowledge of hardware description languages.

[Back to Top](#)

Requirements and Compatibility

OS Information

Windows 2000/XP
Windows Vista x64/x86

Driver Information

NI-RIO

Software Compatibility

LabVIEW FPGA Module

[Back to Top](#)

Comparison Tables

Product	Bus/Form Factor	FPGA	Analog Inputs (16-bit)	Max Sampling Rate per Channel (kS/s)	Analog Outputs (16-bit)	Max Update Rate per Channel (MS/s)	Digital I/O
Multifunction R Series							
NI 7851R	PCI Express, PCI	Virtex-5 LX30	8	750	8	1	96
NI 7852R	PCI Express, PXI	Virtex-5 LX50	8	750	8	1	96
NI 7853R	PXI	Virtex-5 LX85	8	750	8	1	96
NI 7854R	PXI	Virtex-5 LX110	8	750	8	1	96
NI 7841R	PCI Express, PXI	Virtex-5 LX30	8	200	8	1	96

Product	Bus/Form Factor	FPGA	Analog Inputs (16-bit)	Max Sampling Rate per Channel (kS/s)	Analog Outputs (16-bit)	Max Update Rate per Channel (MS/s)	Digital I/O
NI 7842R	PCI Express, PXI	Virtex-5 LX50	8	200	8	1	96
NI 7830R	PCI, PXI	Virtex-II 1M Gates	4	200	4	1	56
NI 7831R	PCI, PXI	Virtex-II 1M Gates	8	200	8	1	96
NI 7833R	PCI, PXI	Virtex-II 3M Gates	8	200	8	1	96
Digital R Series							
NI 7811R	PCI, PXI	Virtex-II 1M Gates	-	-	-	-	160
NI 7813R	PCI, PXI	Virtex-II 3M Gates	-	-	-	-	160

[Back to Top](#)

Application and Technology

Graphical Programming with LabVIEW FPGA

The LabVIEW FPGA Module uses LabVIEW embedded technology to extend LabVIEW graphical development and target FPGAs on NI RIO hardware. LabVIEW is distinctly suited for FPGA programming because it clearly represents parallelism and data flow. With LabVIEW FPGA, you can create custom measurement and control hardware without low-level hardware description languages or board-level design. You can use this custom hardware for unique timing and triggering routines, ultrahigh-speed control, interfacing to digital protocols, digital signal processing (DSP), and many other applications requiring high-speed hardware reliability and tight determinism.

Low-Cost Signal Conditioning and Channel Expansion

The NI 9151 R Series expansion chassis connects directly to any digital connector on R Series devices and houses up to four C Series I/O modules for industrial signal conditioning, I/O channel expansion, and direct sensor connectivity.

New Virtex-5 FPGAs

The new NI 784xR and NI 785xR devices use new Virtex-5 FPGAs with improved optimization capabilities that provide faster code execution and increased code capacity. These Virtex-5 FPGAs feature a new six-input lookup table (LUT) architecture for substantially improved resource utilization as well as DSP48 slices that make it possible for you to implement more complex digital signal processing at faster rates. Previous-generation Virtex-II FPGAs use four-input LUTs for up to 16 combinations of digital logic values. The new Virtex-5 FPGAs use six-input LUTs for up to 64 combinations, increasing the amount of logic that you can implement per slice. In addition, the slices themselves are placed in closer proximity to each other to reduce the propagation delay of electrons and increase overall execution rates. The single-cycle timed loop structure in LabVIEW FPGA takes full advantage of six-input LUTs for substantially improved resource utilization. This means you can optimize more LabVIEW FPGA code to fit within Virtex-5 FPGAs and perform more operations per clock cycle.

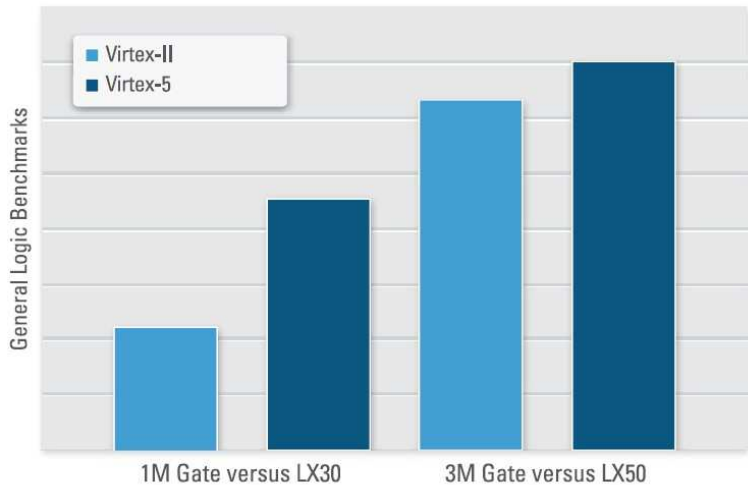


Figure 1. General logic benchmarks show that Virtex-5 FPGAs offer larger sizes when compared to Virtex-II FPGAs.

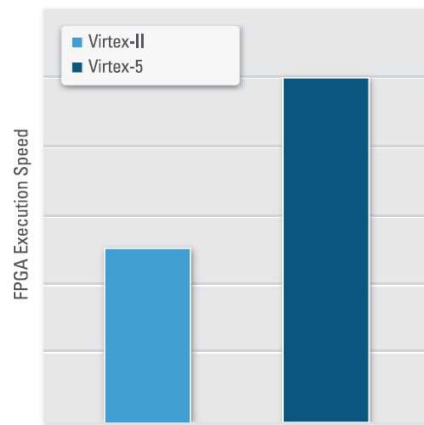


Figure 2. Execution speed benchmarks show that Virtex-5 FPGAs feature faster processing capabilities when compared to Virtex-II FPGAs.

For more information on LabVIEW FPGA benchmarks for Virtex-5 FPGAs, visit ni.com/info and enter **lvfpgabenchmarks**.

Required Software for R Series Multifunction RIO

NI 781xR and 783xR devices require the LabVIEW FPGA Module 7.1 or later and NI-RIO 1.3 or later driver software.
 PXI-7841R/42R/51R/52R modules require the LabVIEW FPGA Module 8.5.1 or later and NI-RIO 2.4 or later.
 PXI-7853R/54R modules require the LabVIEW FPGA Module 8.6 or later and NI-RIO 3.0 or later.
 NI PCIe-7841R/42R/51R/52R boards require the LabVIEW FPGA Module 8.5.1 or later and NI-RIO 3.1 or later.

Recommended Accessories

High Performance

SHC68-68-RMIO - High-performance shielded 68-conductor cable terminated with a VHDCI 68-pin male connector at one end and a 68-pin female 0.050 D-type connector at the other end that has been specifically designed for the multifunction I/O connector on R Series multifunction RIO devices.

1 m189588-01
 2 m189588-02

SHC68-68-RDIO - High-performance shielded 68-conductor cable terminated with a VHDCI 68-pin male connector at one end and a 68-pin female 0.050 D-type connector at the other end that has been specifically designed for the digital I/O connector on R Series multifunction RIO devices.

1 m191667-01

SCB-68 - Shielded I/O connector block for rugged, very low-noise signal termination for connecting to 68-pin devices. The SCB-68 also includes two general-purpose breadboard areas. Dimensions - 19.5 by 15.2 by 4.5 cm (7.7 by 6.0 by 1.8 in.)

SCB-68776844-01

Low Cost

SH68-C68-S - General-purpose shielded cable that connects any type of R Series connector to 68-pin connector blocks.

0.5 m.....186381-0R5
 1 m186381-01
 2 m186381-02

Custom Cabling

SHC68-NT-S - Shielded 68-conductor cable terminated with a 68-pin male VHDCI connector at one end and unterminated bare wires at the other. Use this cable, ideal for OEM applications, to create custom cabling solutions for R Series devices.

2 m189041-02

NSC68-262650 - Shielded cable terminated with a VHDCI 68-pin male connector at one end and two 26-pin ribbon connectors and one 50-pin ribbon connector on the other; designed to connect the R Series RMIO connector to standard ribbon cable accessories.

1 m189151-01

NSC68-5050 - Shielded cable terminated with a VHDCI 68-pin male connector at one end and two 50-pin ribbon connectors on the other; designed to connect R Series RDIO connectors to standard ribbon cable accessories.

1 m189152-01

[Back to Top](#)

Ordering Information

For a complete list of accessories, visit the product page on ni.com.

Products	Part Number	Recommended Accessories	Part Number
NI PXI-7811R			
NI PXI-7811R Requires: 1 Cables , 1 Connector Blocks ;	778800-01	Connector 0: Cables: Shielded - SHC68-68-RDIO Cable (1m) **Also Available: [Unshielded]	191667-01

		Connector Blocks: Spring-Screw_Terminals - SCB-68A	782536-01
		Connector 1:	
		Cables: Shielded - SHC68-68-RDIO Cable (1m) <i>**Also Available: [Unshielded]</i>	191667-01
		Connector Blocks: Spring-Screw_Terminals - SCB-68A	782536-01
		Connector 2:	
		Cables: - <i>**Also Available: [Unshielded]</i>	
		Connector Blocks: -	
		Connector 3:	
		Cables: - <i>**Also Available: [Unshielded]</i>	
		Connector Blocks: -	
NI PXI-7813R			
NI PXI-7813R	779362-01		
Requires: 1 Cables , 1 Connector Blocks ;		Connector 0:	
		Cables: Shielded - SHC68-68-RDIO Cable (1m) <i>**Also Available: [Unshielded]</i>	191667-01
		Connector Blocks: Spring-Screw_Terminals - SCB-68A	782536-01
		Connector 1:	
		Cables: Shielded - SHC68-68-RDIO Cable (1m) <i>**Also Available: [Unshielded]</i>	191667-01
		Connector Blocks: Spring-Screw_Terminals - SCB-68A	782536-01
		Connector 2:	
		Cables: - <i>**Also Available: [Unshielded]</i>	
		Connector Blocks: -	
		Connector 3:	
		Cables: - <i>**Also Available: [Unshielded]</i>	
		Connector Blocks: -	
NI PCI-7830R			
NI PCI-7830R	779361-01		
Requires: 2 Cable , 2 Connector Block ;		Connector 0:	
		Cable: Shielded - SHC68-68-RMIO Cable (2m) <i>**Also Available: [Unshielded]</i>	189588-02
		Connector Block: Spring-Screw_Terminals - SCB-68A	782536-01
		Connector 1:	
		Cable: Shielded - SHC68-68-RMIO Cable (2m) <i>**Also Available: [Unshielded]</i>	189588-02
		Connector Block: Spring-Screw_Terminals - SCB-68A	782536-01
NI PXI-7830R			
NI PXI-7830R	779364-01		
Requires: 1 Cable , 1 Connector Block ;		Connector 0:	
		Cable: Shielded - SHC68-68-RMIO Cable (2m) <i>**Also Available: [Unshielded]</i>	189588-02
		Connector Block: Spring-Screw_Terminals - SCB-68A	782536-01
		Connector 1:	
		Cable: Shielded - SHC68-68-RMIO Cable (2m) <i>**Also Available: [Unshielded]</i>	189588-02
		Connector Block: Spring-Screw_Terminals - SCB-68A	782536-01
NI PCI-7831R			
NI PCI-7831R	778797-01		
Requires: 3 Cables , 3 Connector Block ;		Connector 0:	
		Cables: Shielded - SHC68-68-RMIO Cable (2m)	189588-02
		Connector Block: Spring-Screw_Terminals - SCB-68A	782536-01
		Connector 1:	
		Cables: Shielded - SHC68-68-RMIO Cable (2m)	189588-02
		Connector Block: Spring-Screw_Terminals - SCB-68A	782536-01
		Connector 2:	

		Cables: Shielded - SHC68-68-RMIO Cable (2m)	189588-02
		Connector Block: Spring-Screw_Terminals - SCB-68A	782536-01
NI PXI-7831R			
NI PXI-7831R	778668-01		
Requires: 1 Cables , 1 Connector Block ;		Connector 0:	
		Cables: Shielded - SHC68-68-RMIO Cable (2m)	189588-02
		**Also Available: [Unshielded]	
		Connector Block: Spring-Screw_Terminals - SCB-68A	782536-01
		Connector 1:	
		Cables: Shielded - SHC68-68-RMIO Cable (2m)	189588-02
		**Also Available: [Unshielded]	
		Connector Block: Spring-Screw_Terminals - SCB-68A	782536-01
		Connector 2:	
		Cables: -	
		**Also Available: [Unshielded]	
		Connector Block: -	
NI PCI-7833R			
NI PCI-7833R	779359-01		
Requires: 3 Cables , 3 Connector Blocks ;		Connector 0:	
		Cables: Shielded - SHC68-68-RMIO Cable (2m)	189588-02
		**Also Available: [Unshielded]	
		Connector Blocks: Spring-Screw_Terminals - SCB-68A	782536-01
		Connector 1:	
		Cables: Shielded - SHC68-68-RMIO Cable (2m)	189588-02
		**Also Available: [Unshielded]	
		Connector Blocks: Spring-Screw_Terminals - SCB-68A	782536-01
		Connector 2:	
		Cables: Shielded - SHC68-68-RMIO Cable (2m)	189588-02
		**Also Available: [Unshielded]	
		Connector Blocks: Spring-Screw_Terminals - SCB-68A	782536-01
NI PXI-7833R			
NI PXI-7833R	779360-01		
Requires: 1 Cables , 1 Connector Blocks ;		Connector 0:	
		Cables: Shielded - SHC68-68-RMIO Cable (2m)	189588-02
		**Also Available: [Unshielded]	
		Connector Blocks: Spring-Screw_Terminals - SCB-68A	782536-01
		Connector 1:	
		Cables: Shielded - SHC68-68-RMIO Cable (2m)	189588-02
		**Also Available: [Unshielded]	
		Connector Blocks: Spring-Screw_Terminals - SCB-68A	782536-01
		Connector 2:	
		Cables: -	
		**Also Available: [Unshielded]	
		Connector Blocks: -	
NI PXI-7841R			
NI PXI-7841R	780337-01		
Requires: 1 Cable , 1 Connector Block ;		Connector 0:	
		Cable: Shielded - SHC68-68-RMIO Cable (2m)	189588-02
		**Also Available: [Unshielded]	
		Connector Block: Spring-Screw_Terminals - SCB-68A	782536-01
		Connector 1:	
		Cable: Shielded - SHC68-68-RMIO Cable (2m)	189588-02
		**Also Available: [Unshielded]	
		Connector Block: Spring-Screw_Terminals - SCB-68A	782536-01
		Connector 2:	
		Cable: -	
		**Also Available: [Unshielded]	
		Connector Block: -	
NI PXI-7842R			
NI PXI-7842R	780338-01		

Requires: 1 Cable , 1 Connector Block ;		Connector 0: Cable: Shielded - SHC68-68-RMIO Cable (2m) **Also Available: <i>[Unshielded]</i> Connector Block: Spring-Screw_Terminals - SCB-68A	189588-02 782536-01
		Connector 1: Cable: Shielded - SHC68-68-RMIO Cable (2m) **Also Available: <i>[Unshielded]</i> Connector Block: Spring-Screw_Terminals - SCB-68A	189588-02 782536-01
		Connector 2: Cable: - **Also Available: <i>[Unshielded]</i> Connector Block: -	
NI PXI-7851R			
NI PXI-7851R	780339-01		
Requires: 1 Cable , 1 Connector Block ;		Connector 0: Cable: Shielded - SHC68-68-RMIO Cable (2m) **Also Available: <i>[Unshielded]</i> Connector Block: Spring-Screw_Terminals - SCB-68A	189588-02 782536-01
		Connector 1: Cable: Shielded - SHC68-68-RMIO Cable (2m) **Also Available: <i>[Unshielded]</i> Connector Block: Spring-Screw_Terminals - SCB-68A	189588-02 782536-01
		Connector 2: Cable: - **Also Available: <i>[Unshielded]</i> Connector Block: -	
NI PXI-7852R			
NI PXI-7852R	780340-01		
Requires: 1 Cable , 1 Connector Block ;		Connector 0: Cable: Shielded - SHC68-68-RMIO Cable (2m) **Also Available: <i>[Unshielded]</i> Connector Block: Spring-Screw_Terminals - SCB-68A	189588-02 782536-01
		Connector 1: Cable: Shielded - SHC68-68-RMIO Cable (2m) **Also Available: <i>[Unshielded]</i> Connector Block: Spring-Screw_Terminals - SCB-68A	189588-02 782536-01
		Connector 2: Cable: - **Also Available: <i>[Unshielded]</i> Connector Block: -	
NI PXI-7853R			
NI PXI-7853R	780341-01		
Requires: 1 Cables , 1 Connector Block ;		Connector 0: Cables: Shielded - SHC68-68-RMIO Cable (2m) **Also Available: <i>[Unshielded]</i> Connector Block: Spring-Screw_Terminals - SCB-68A	189588-02 782536-01
		Connector 1: Cables: Shielded - SHC68-68-RMIO Cable (2m) **Also Available: <i>[Unshielded]</i> Connector Block: Spring-Screw_Terminals - SCB-68A	189588-02 782536-01
		Connector 2: Cables: - **Also Available: <i>[Unshielded]</i> Connector Block: -	
NI PXI-7854R			
NI PXI-7854R	780342-01		
Requires: 1 Cables , 1 Connector Block ;		Connector 0: Cables: Shielded - SHC68-68-RMIO Cable (2m) **Also Available: <i>[Unshielded]</i> Connector Block: Spring-Screw_Terminals - SCB-68A	189588-02 782536-01

Connector 1:**Cables:** Shielded - SHC68-68-RMIO Cable (2m)

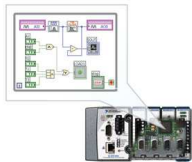
189588-02

****Also Available:** [Unshielded]**Connector Block:** Spring-Screw_Terminals - SCB-68A782536-01

Connector 2:**Cables:** -****Also Available:** [Unshielded]**Connector Block:** -[Back to Top](#)

Software Recommendations

NI LabVIEW FPGA Module



Create your own I/O hardware without VHDL coding or board design
Graphically configure FPGAs on NI reconfigurable I/O (RIO) hardware targets
Define your own control algorithms with loop rates up to 300 MHz
Execute multiple tasks simultaneously and deterministically
Implement custom timing and triggering logic, digital protocols, and DSP algorithms
Incorporate existing HDL code and third-party IP including Xilinx CORE Generator functions

[Back to Top](#)

Support and Services

System Assurance Programs

NI system assurance programs are designed to make it even easier for you to own an NI system. These programs include configuration and deployment services for your NI PXI, CompactRIO, or Compact FieldPoint system. The NI Basic System Assurance Program provides a simple integration test and ensures that your system is delivered completely assembled in one box. When you configure your system with the NI Standard System Assurance Program, you can select from available NI system driver sets and application development environments to create customized, reorderable software configurations. Your system arrives fully assembled and tested in one box with your software preinstalled. When you order your system with the standard program, you also receive system-specific documentation including a bill of materials, an integration test report, a recommended maintenance plan, and frequently asked question documents. Finally, the standard program reduces the total cost of owning an NI system by providing three years of warranty coverage and calibration service. Use the online product advisors at ni.com/advisor to find a system assurance program to meet your needs.

Calibration

NI measurement hardware is calibrated to ensure measurement accuracy and verify that the device meets its published specifications. To ensure the ongoing accuracy of your measurement hardware, NI offers basic or detailed recalibration service that provides ongoing ISO 9001 audit compliance and confidence in your measurements. To learn more about NI calibration services or to locate a qualified service center near you, contact your local sales office or visit ni.com/calibration.

Technical Support

Get answers to your technical questions using the following National Instruments resources.

Support - Visit ni.com/support to access the NI KnowledgeBase, example programs, and tutorials or to contact our applications engineers who are located in NI sales offices around the world and speak the local language.

Discussion Forums - Visit forums.ni.com for a diverse set of discussion boards on topics you care about.

Online Community - Visit community.ni.com to find, contribute, or collaborate on customer-contributed technical content with users like you.

Repair

While you may never need your hardware repaired, NI understands that unexpected events may lead to necessary repairs. NI offers repair services performed by highly trained technicians who quickly return your device with the guarantee that it will perform to factory specifications. For more information, visit ni.com/repair.

Training and Certifications

The NI training and certification program delivers the fastest, most certain route to increased proficiency and productivity using NI software and hardware. Training builds the skills to more efficiently develop robust, maintainable applications, while certification validates your knowledge and ability.

Classroom training in cities worldwide - the most comprehensive hands-on training taught by engineers.

On-site training at your facility - an excellent option to train multiple employees at the same time.

Online instructor-led training - lower-cost, remote training if classroom or on-site courses are not possible.

Course kits - lowest-cost, self-paced training that you can use as reference guides.

Training memberships and training credits - to buy now and schedule training later.

Visit ni.com/training for more information.

Extended Warranty

NI offers options for extending the standard product warranty to meet the life-cycle requirements of your project. In addition, because NI understands that your requirements may change, the extended warranty is flexible in length and easily renewed. For more information, visit ni.com/warranty.

OEM

NI offers design-in consulting and product integration assistance if you need NI products for OEM applications. For information about special pricing and services for OEM customers, visit ni.com/oem.

Alliance

Our Professional Services Team is comprised of NI applications engineers, NI Consulting Services, and a worldwide National Instruments Alliance Partner program of more than 700 independent consultants and integrators. Services range from start-up assistance to turnkey system integration. Visit ni.com/alliance.

[Back to Top](#)

Detailed Specifications

This document lists the specifications of the NI 781xR/783xR/784xR/785xR. These specifications are typical at 25 °C unless otherwise noted.

Analog Input (NI 783x R/784x R/785x R Only)

Input Characteristics

Number of channels		
NI 7830R		4
NI 7831R/7833R/7841R/7842R/7851R/7852R/7853R/7854R		8
Input modes		DIFF, RSE, NRSE (software-selectable; selection applies to all channels)
Type of ADC		Successive approximation
Resolution		16 bits, 1 in 65,536
Conversion time		
NI 783xR/NI 784xR		4 µs
NI 785xR		1 µs
Maximum sampling rate		
NI 783xR/784xR		200 kS/s (per channel)
NI 785xR		750 kS/s (per channel)
Input impedance		
Powered on		10 GΩ in parallel with 100 pF
Powered off/overload		4.0 kΩ min
Input signal range		±10 V
Input bias current		
NI 783xR		±2 nA
NI 784xR/785xR		±5 nA
Input offset current		
NI 783xR		±1 nA
NI 784xR/785xR		±5 nA
Input coupling		DC
Maximum working voltage (signal + common mode)		Inputs should remain within ±12 V of ground
Overvoltage protection		
Powered on		±42 V
Powered off		±35 V

Accuracy Information

NI 783xR										
Nominal Range (V)		Absolute Accuracy							Relative Accuracy	
		% of Reading		Offset (μ V)	Noise + Quantization (μ V)		Temp Drift (%/ °C)	Absolute Accuracy at Full Scale ($\pm$ mV)	Resolution (μ V)	
Positive Full Scale	Negative Full Scale	24 Hours	1 Year		Single Point	Averaged			Single Point	Averaged
10.0	–10.0	0.0496	0.0507	2,542	1,779	165	0.0005	7.78	2,170	217
 Note Accuracies are valid for measurements following an internal calibration. Averaged numbers assume dithering and averaging of 100 single-channel readings. Measurement accuracies are listed for operational temperatures within ± 1 °C of internal calibration temperature and ± 10 °C of external or factory-calibration temperature.										

NI 784xR/NI 785xR										
Nominal Range (V)		Absolute Accuracy							Relative Accuracy	
		% of Reading		Offset (μ V)	Noise + Quantization (μ V)		Temp Drift (%/ °C)	Absolute Accuracy at Full Scale ($\pm$ mV)	Resolution (μ V)	
Positive Full Scale	Negative Full Scale	24 Hours	1 Year		Single Point	Averaged			Single Point	Averaged
10.0	–10.0	0.0186	0.0228	1,591	1,029	91.6	0.0005	3.97	1,205	121
 Note Accuracies are valid for measurements following an internal calibration. Averaged numbers assume dithering and averaging of 100 single-channel readings. Measurement accuracies are listed for operational temperatures within ± 1 °C of internal calibration temperature and ± 10 °C of external or factory-calibration temperature.										

DC Transfer Characteristics

INL

NI 783xR	± 3 LSB typ, ± 6 LSB max
NI 784xR/785xR	± 1 LSB typ, ± 3 LSB max

DNL

NI 783xR	± 1.0 to ± 2.0 LSB max
NI 784xR/785xR	± 0.4 LSB typ, ± 0.9 LSB max

No missing codes

NI 783xR	16 bits typ, 15 bits min
NI 784xR/785xR	16 bits guaranteed

CMRR, DC to 60 Hz	–86 dB
-------------------	--------

Dynamic Characteristics

Bandwidth

NI 783xR	
Small signal (– 3 dB)	650 kHz
Large signal (1% of THD)	55 kHz
NI 784xR/NI 785xR	
Small signal (– 3 dB)	1 MHz
Large signal (1% of THD)	500 kHz

Settling Time

Device	Step Size	Accuracy		
		± 16 LSB	± 4 LSB	± 2 LSB
NI 783xR	± 20.0 V	7.5 μ s	10.3 μ s	40 μ s
	± 2.0 V	2.7 μ s	4.1 μ s	5.1 μ s
	± 0.2 V	1.7 μ s	2.9 μ s	3.6 μ s
NI 784xR/ 785xR	± 20.0 V	2.1 μ s	4.2 μ s	8 μ s
	± 2.0 V	1.3 μ s	1.6 μ s	1.8 μ s
	± 0.2 V	0.8 μ s	1.1 μ s	1.2 μ s

Crosstalk

–80 dB, DC to 100 kHz

Analog Output (NI 783x R/784x R/785x R Only)

Output Characteristics

Output type	Single-ended, voltage output
Number of channels	
NI 7830R	4
NI 7831R/7833R/7841R/7842R/7851R/7852R/7853R/7854R	8
Resolution	16 bits, 1 in 65,536
Update time	1.0 μ s
Maximum update rate	1 MS/s
Type of DAC	Enhanced R-2R

Accuracy Information

Nominal Range (V)		Absolute Accuracy				Absolute Accuracy at Full Scale ($\pm$ mV)
		% of Reading		Offset (μ V)	Temp Drift (%/ $^{\circ}$ C)	
Positive Full Scale	Negative Full Scale	24 Hours	1 Year			
10.0	−10.0	0.0335	0.0351	2366	0.0005	5.88
 Note Accuracies are valid for analog output following an internal calibration. Analog output accuracies are listed for operation temperatures within ± 1 $^{\circ}$ C of internal calibration temperature and ± 10 $^{\circ}$ C of external or factory calibration temperature. Temp Drift applies only if ambient is greater than ± 10 $^{\circ}$ C of previous external calibration.						

DC Transfer Characteristics

INL	± 0.5 LSB typ, ± 4.0 LSB max
DNL	± 0.5 LSB typ, ± 1 LSB max
Monotonicity	16 bits, guaranteed

Voltage Output

Range	± 10 V
Output coupling	DC
Output impedance	
NI 783xR	1.25 Ω
NI 784xR/785xR	0.5 Ω
Current drive	± 2.5 mA
Protection	Short-circuit to ground
Power-on state	User configurable

Dynamic Characteristics

Settling time			
Step Size	Accuracy		
	± 16 LSB	± 4 LSB	± 2 LSB
± 20.0 V	6.0 μ s	6.2 μ s	7.2 μ s
± 2.0 V	2.2 μ s	2.9 μ s	3.8 μ s
± 0.2 V	1.5 μ s	2.6 μ s	3.6 μ s

Slew rate	10 V/ μ s
Noise	150 μ V _{rms} , DC to 1 MHz
Glitch energy at midscale transition	± 200 mV for 3 μ s

Digital I/O

Number of channels

NI 781xR	160
NI 7830R	56
NI 7831R/7833R/7841R/7842R/7851R/7852R/7853R/7854R	96
Compatibility	TTL

Digital logic levels		
Level	Min	Max
Input low voltage (V_{IL})	0.0 V	0.8 V
Input high voltage (V_{IH})	2.0 V	5.5 V
Output low voltage (V_{OL}), where $I_{OUT} = -4$ mA	0 V	0.4 V
Output high voltage (V_{OH}), where $I_{OUT} = 4$ mA	2.4 V	3.3 V

Output current	
Source	4.0 mA
Sink	4.0 mA
Input leakage current	± 10 μ A
Power-on state	Programmable, by line
Protection	
Input	
NI 781xR/783xR	-0.5 to 7.0 V, single line
NI 784xR/785xR	-20.0 to 20.0 V, single line
Output	Short-circuit (up to eight lines may be shorted at a time)
Minimum pulse width	
Input	25 ns
Output	12.5 ns
Minimum sampling period	5 ns

Reconfigurable FPGA

NI 7811R/7830R/7831R	
FPGA type	Virtex-II V1000
Number of flip-flops	10,240
Number of 4-input LUTs	10,240
Number of 18 $\times$ 18 multipliers	40
Embedded block RAM	720 kbits
NI 7813R/7833R	
FPGA type	Virtex-II V3000
Number of flip-flops	28,672
Number of 4-input LUTs	28,672
Number of 18 $\times$ 18 multipliers	96
Embedded block RAM	1,728 kbits
NI 7841R/7851R	
FPGA type	Virtex-5 LX30
Number of flip-flops	19,200
Number of 6-input LUTs	19,200
Number of DSP48 slices (25 $\times$ 18 multipliers)	32
Embedded block RAM	1,152 kbits

NI 7842R/7852R	
FPGA type	Virtex-5 LX50
Number of flip-flops	28,800
Number of 6-input LUTs	28,800
Number of DSP48 slices (25 × 18 multipliers)	48
Embedded block RAM	1,728 kbits
NI 7853R	
FPGA type	Virtex-5 LX85
Number of flip-flops	51,840
Number of 6-input LUTs	51,840
Number of DSP48 slices (25 × 18 multipliers)	48
Embedded block RAM	3,456 kbits
NI 7854R	
FPGA type	Virtex-5 LX110
Number of flip-flops	69,120
Number of 6-input LUTs	69,120
Number of DSP48 slices (25 × 18 multipliers)	64
Embedded block RAM	4,608 kbits
Timebase	40, 80, 120, 160, or 200 MHz
Timebase reference sources	
NI PCI-781xR/783xR	Onboard clock only
NI PCIe-784xR/785xR	Onboard clock only
NI PXI-78xxR	Onboard clock, phase-locked to PXI 10 MHz clock
Timebase accuracy, onboard clock	±100 ppm, 250 ps peak-to-peak jitter
Phase locked to PXI 10 MHz Clock (NI PXI-78xxR only)	Adds 350 ps peak-to-peak jitter
Additional frequency-dependent peak-to-peak jitter	
NI 781xR/783xR	
40 MHz	None
80 MHz	400 ps
120 MHz	720 ps
160 MHz	710 ps
200 MHz	700 ps
NI 784xR/785xR	
40 MHz	None
80 MHz	460 ps
120 MHz	172 ps
160 MHz	172 ps
200 MHz	152 ps
Calibration (NI 783x R/784x R/785x R Only)	
Recommended warm-up time	15 minutes
Calibration interval	1 year
Onboard calibration reference	
DC level	5.000 V (±3.5 mV) (actual value stored in Flash memory)
Temperature coefficient	±5 ppm/°C max
Long-term stability	±20 ppm/ $\sqrt{1,000 \text{ h}}$



Note Refer to *Calibration Certificates* at ni.com/calibration to generate a calibration certificate for the NI 78xxR.

Bus Interface		
PCI/PCIe/PXI		Master, slave
Data transfers		DMA, interrupts, programmed I/O
Number of DMA channels		3
Power Requirement		
+5 VDC ($\pm 5\%$) ¹		
NI 781xR		9 mA typ
NI 7830R/7831R		330 mA typ
NI 7833R		364 mA typ
NI PXI-7841R/7851R		125 mA typ
NI PXI-7842R/7852R		136 mA typ
NI 7853R		460 mA typ
NI 7854R		484 mA typ
+3.3 VDC ($\pm 5\%$) ²		
NI 7811R		650 mA typ
NI 7813R		850 mA typ
NI 7830R/7831R		462 mA typ
NI 7833R		727 mA typ
NI PCIe-7841R/7851R		847 mA typ
NI PCIe-7842R/7852R		984 mA typ
NI PXI-7841R/7851R		525 mA typ
NI PXI-7842R/7852R		604 mA typ
NI 7853R		640 mA typ
NI 7854R		843 mA typ
+12 V		
NI 784xR/785xR		0.5 A
−12 V		
NI PXI-784xR/785xR		0.25 A
+5V terminal		
Connector 0		0.5 A max ³
Connector 1		0.5 A max ³
Connector 2		0.5 A max ³
All connectors		1.5 A max ³ , ⁴
To calculate the total current sourced by the digital outputs, use the following equation:		
$\sum_{i=1}^j$ ⁱ = 1 current sourced on channel <i>i</i>		
Power available at I/O connectors		4.50 to 5.25 VDC at 1 A total, 250 mA per I/O connector pin
Physical		
Dimensions (not including connectors)		
NI PCI-781xR/783xR		17 cm by 11 cm (6.7 in. by 4.3 in.)
NI PCIe-784xR/785xR		17 cm by 11 cm (6.7 in. by 4.3 in.)
NI PXI-78xxR		16 cm by 10 cm (6.3 in. by 3.9 in.)
Weight		
NI PCI-781xR/783xR		112 g

NI PCIe-784xR/785xR	127 g
NI PXI-78xxR	152 g
I/O connectors	
NI 781xR	Four 68-pin female high-density VHDCI type
NI 7830R	Two 68-pin female high-density VHDCI type
NI 783xR/784xR/785xR	Three 68-pin female high-density VHDCI type
Disk drive power connector (PCIe devices)	Standard ATX peripheral connector (not serial ATA)

Maximum Working Voltage (NI 783xR/784xR/785xR Only)

Maximum working voltage refers to the signal voltage plus the common-mode voltage.

Channel-to-earth ± 12 V, Measurement Category I

Channel-to-channel ± 24 V, Measurement Category I



Caution Do *not* use the NI 783xR/784xR/785xR for connection to signals in Measurement Categories II, III, or IV.

Environmental

The NI 78xxR is intended for indoor use only.

Operating Environment

NI 781xR	0 °C to 55 °C, tested in accordance with IEC-60068-2-1 and IEC-60068-2-2.
NI 7830R, NI 7831R	
40 MHz or 80 MHz timebase	0 °C to 55 °C, tested in accordance with IEC-60068-2-1 and IEC-60068-2-2.
NI PCI/PXI-7833R	
40 MHz timebase	0 °C to 55 °C, tested in accordance with IEC-60068-2-1 and IEC-60068-2-2.
80 MHz timebase	0 °C to 55 °C except the following: 0 °C to 45 °C when installed in an NI PXI-1000/B or NI PXI-101X, tested in accordance with IEC-60068-2-1 and IEC-60068-2-2.
NI PXI-7841R/7842R/7851R/7852R/7853R/7854R	
40 MHz timebase	0 °C to 55 °C, tested in accordance with IEC-60068-2-1 and IEC-60068-2-2.
80 MHz timebase	0 °C to 55 °C except the following: 0 °C to 45 °C when installed in an NI PXI-1000/B or NI PXI-101X, tested in accordance with IEC-60068-2-1 and IEC-60068-2-2.
NI PCIe-7841R/7842R/7851R/7852R	
40 MHz or 80 MHz timebase	0 °C to 40 °C, tested in accordance with IEC-60068-2-1 and IEC-60068-2-2.
Relative humidity range	10% to 90%, noncondensing, tested in accordance with IEC-60068-2-56.
Altitude	2,000 m at 25 °C ambient temperature
Storage Environment	
NI PCI/PXI-781xR/783xR	–20 °C to 70 °C, tested in accordance with IEC-60068-2-1 and IEC-60068-2-2.
NI PCIe-784xR/785xR	–20 °C to 70 °C, tested in accordance with IEC-60068-2-1 and IEC-60068-2-2.
NI PXI-784xR/785xR	–40 °C to 70 °C, tested in accordance with IEC-60068-2-1 and IEC-60068-2-2.
Ambient temperature range	–20 °C to 70 °C, tested in accordance with IEC-60068-2-1 and IEC-60068-2-2.
Relative humidity range	5% to 95%, noncondensing, tested in accordance with IEC-60068-2-56.



Note Clean the device with a soft, non-metallic brush. Make sure that the device is completely dry and free from contaminants before returning it to service.

Shock and Vibration (for NI PXI-78xxR Only)

Operational shock	30 g peak, half-sine, 11 ms pulse; tested in accordance with IEC-60068-2-27. Test profile developed in accordance with MIL-PRF-28800F.
Random vibration	
Operating	5 Hz to 500 Hz, 0.3 g _{rms}
Nonoperating	5 Hz to 500 Hz, 2.4 g _{rms} , tested in accordance with IEC-60068-2-64. Nonoperating test profile exceeds the requirements of MIL-PRF-28800F, Class 3.

Safety

This product is designed to meet the requirements of the following standards of safety for electrical equipment for measurement, control, and laboratory use:

IEC 61010-1, EN 61010-1

UL 61010-1, CSA 61010-1



Note For UL and other safety certifications, refer to the product label or the *Online Product Certification* section.

Electromagnetic Compatibility

This product is designed to meet the requirements of the following standards of EMC for electrical equipment for measurement, control, and laboratory use:

EN 61326 (IEC 61326): Class A emissions; Basic immunity
EN 55011 (CISPR 11): Group 1, Class A emissions
AS/NZS CISPR 11: Group 1, Class A emissions
FCC 47 CFR Part 15B: Class A emissions
ICES-001: Class A emissions



Note For the standards applied to assess the EMC of this product, refer to the [Electromagnetic Compatibility](#) section.



Note For EMC compliance, operate this device with shielded cabling.

CE Compliance

This product meets the essential requirements of applicable European Directives, as amended for CE marking, as follows:

2006/95/EC; Low-Voltage Directive (safety)
2004/108/EC; Electromagnetic Compatibility Directive (EMC)



Note For the standards applied to assess the EMC of this product, refer to the *Online Product Certification* section.

Online Product Certification

Refer to the product Declaration of Conformity (DoC) for additional regulatory compliance information. To obtain product certifications and the DoC for this product, visit ni.com/certification, search by module number or product line, and click the appropriate link in the Certification column.

Environmental Management

National Instruments is committed to designing and manufacturing products in an environmentally responsible manner. NI recognizes that eliminating certain hazardous substances from our products is beneficial not only to the environment but also to NI customers.

For additional environmental information, refer to the *NI and the Environment* Web page at ni.com/environment. This page contains the environmental regulations and directives with which NI complies, as well as other environmental information not included in this document.

Waste Electrical and Electronic Equipment (WEEE)



EU Customers At the end of their life cycle, all products *must* be sent to a WEEE recycling center. For more information about WEEE recycling centers and National Instruments WEEE initiatives, visit ni.com/environment/weee.htm.

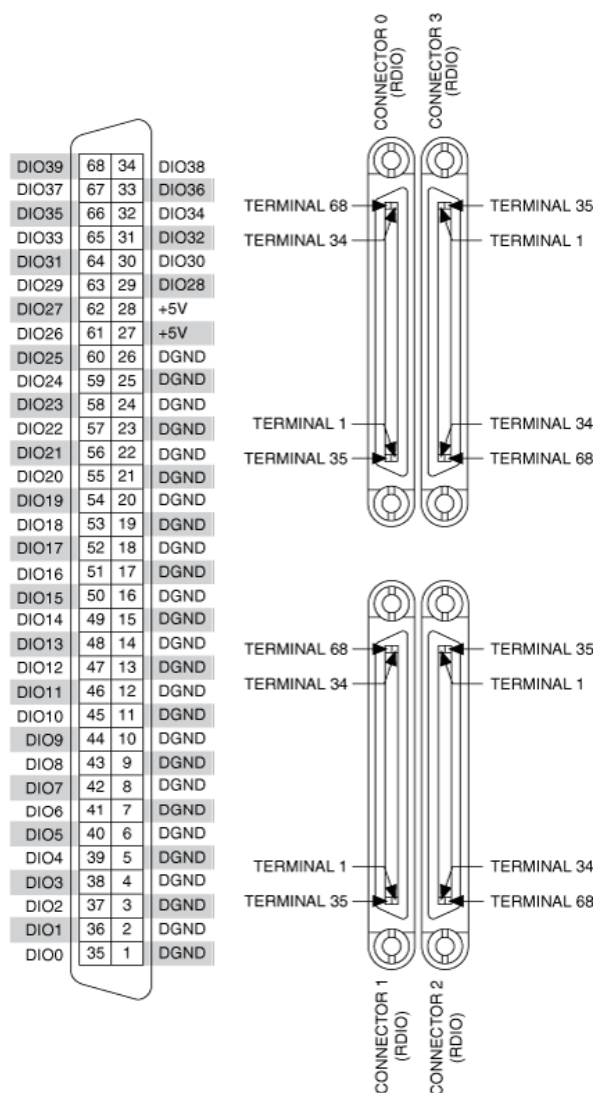
¹ Does not include current drawn from the +5 V line on the I/O connectors.

² Does not include current sourced by the digital outputs.

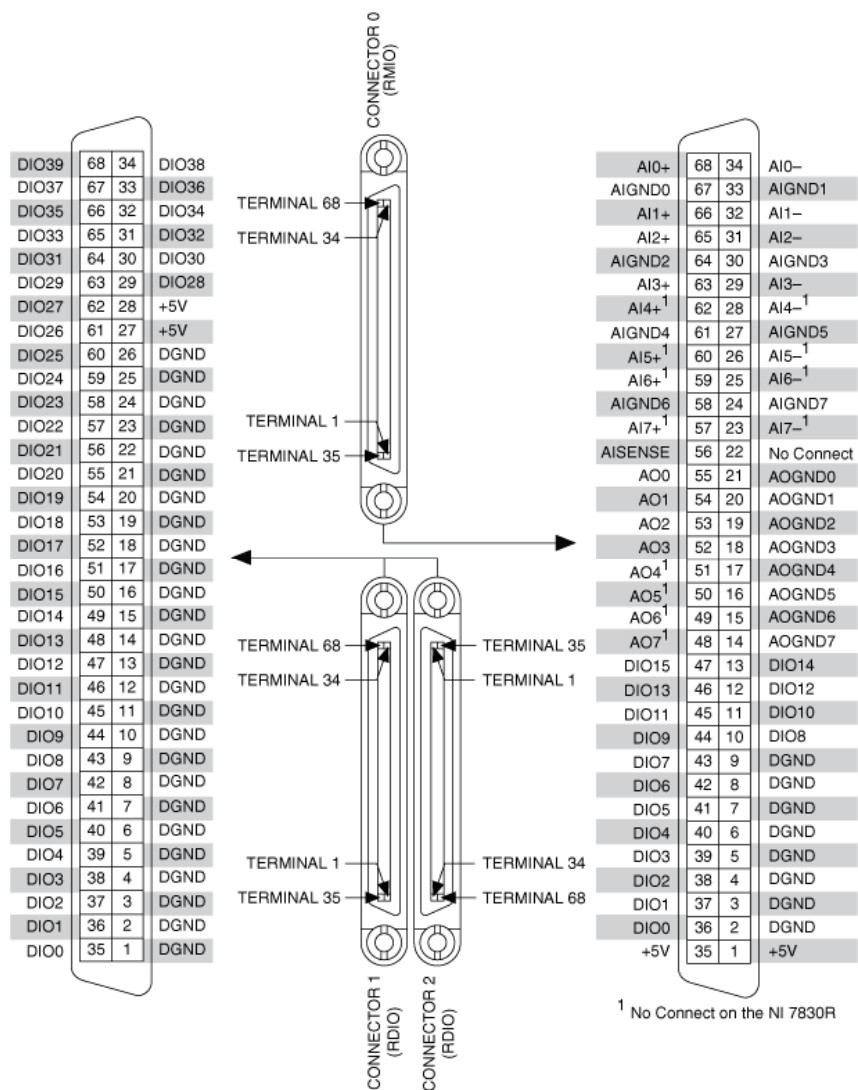
³ **(NI PCIe-78xxR only)** Total maximum terminal current for all connectors is 100 mA unless disk drive connector is attached.

⁴ **(NI 784xR/785xR only)** The NI 784xR/785xR has a user-replaceable socketed fuse that opens when current exceeds the current specification. Refer to the *NI R Series Multifunction RIO User Manual*, available at ni.com/manuals, for information about fuse replacement.

Pinouts/Front Panel Connections



NI 781xR Connector Pin Assignments and Locations



NI 783xR/784xR/785xR Connector Pin Assignments and Locations

[Back to Top](#)

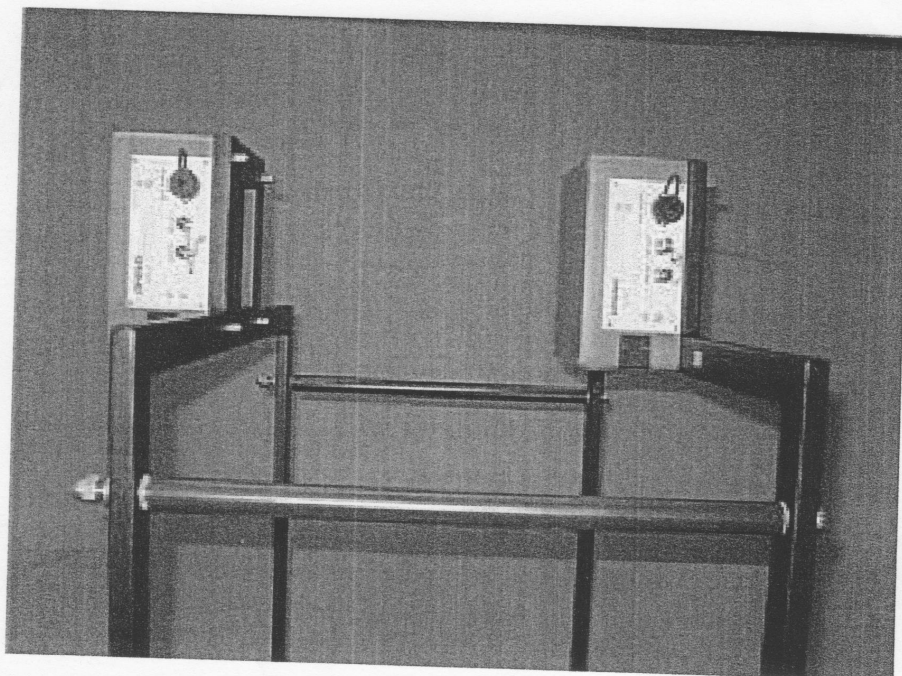
©2009 National Instruments. All rights reserved. CompactRIO, FieldPoint, LabVIEW, National Instruments, National Instruments Alliance Partner, NI, and ni.com are trademarks of National Instruments. Other product and company names listed are trademarks or trade names of their respective companies. A National Instruments Alliance Partner is a business entity independent from National Instruments and has no agency, partnership, or joint-venture relationship with National Instruments.

[My Profile](#) | [RSS](#) | [Privacy](#) | [Legal](#) | [Contact NI](#) © 2012 National Instruments Corporation. All rights reserved.

Bijlage D

Datasheet meetbasis Drello LS 19 i3

**INFRARED LIGHT SCREEN
LS 19 i3**



DRELO

1.0 General Comments

With this system the projectile velocity "v" is measured in that way that the flight time "t" of the projectile is determined for a certain flight distance "s".

$$v = s/t$$

The light screen LS 1913 is mounted on a frame with an inner dimension of 840 x 750 mm and with a trapezoidal sensitive area (sketch 1). Opposite to the measuring head a reflector is mounted on the frame. The frames either can be mounted individually on supports or at the wall by means of a special hinged holding. In addition they can also be used in a special frame make Mero with a 2 m measuring reference or with double measurement with a 2 m and 4 m or 4 m and 6 m measuring reference. Furtheron there is the possibility to form a fixed 2 m measuring reference with the LS 1913 or with double measurement a 2 m and 3 m measuring reference by using respective spacing bars (see illustration on front cover). Principally, the test assembly can be manufactured custom configured.

For the measurement of the flight time "t" serve our chronometers, velocity computers as well as computer aided measuring equipments which are started by a pulse released at the beginning of the measuring reference and are stopped by a pulse released at the end of the measuring reference.

The light screen LS 1913 can be used for calibers from 4 to 40 mm and under certain conditions also for caliber 105 mm. First of all the light screen is destined for the use in closed shooting channels but can also be used in the open air. In the later case the sky light serves as light source. When mounting the light screens on supports velocity measurements are also possible under elevation (option).

The uncertainty in measurement is with all calibers $\leq 0.2\%$.

1.1 Measuring method

The design of the LS 1913 is such that emitter and receiver are incorporated in one measuring head. The emitter consists of 3 infrared light emitting diodes. These diodes emit the light in a wave range of 780 nm. By means of a cylindrical lens this infrared light is concentrated to a small light band in one direction. In the other direction (transvers to the firing direction) it has an opening angle of 35°. Opposite to the measuring head a Scotchlith foil is placed. This foil reflects the light back to the measuring head.

The reflected light band falls through a gap of 0.7 mm width in the measuring head. At this gap the light band has a width of approx. 1.2 mm. Behind the gap are the receiving diodes.

When the projectile passes the light band the receiving diodes are shaded. This shading must be minimum 2 %. The receiving diodes thereby release a minimum pulse which is amplified 500 times in an amplification stage. The amplified pulse is converted into a digital signal first in a trigger stage and then in an end stage.

DRELO

INFRARED LIGHT SCREEN LS 19i 3

List of Contents

Chapter		Page
1.0	General comments	1
1.1	Measuring method	1
2.0	Mounting of the measuring reference	2
2.1	Mounting with supports	2
2.2	Mounting with spacing bars	2
2.3	Mounting with frame make "Mero" (sketch 5)	3
2.4	Function of the operating elements	3
3.0	Adjustment of the measuring head	5
3.1	Adjustment of the LED current	5
3.2	Adjustment of the analog amplification	5
3.3	Adjustment of the trigger threshold	5
4.0	Measurement of tracer ammunition	7
5.0	Technical data	9
	Sketch 1	10
	Sketch 2	11
	Sketch 3	12
	Sketch 4	13
	Sketch 5	14

3.52.2e Bi*975

2.0 Mounting of the measuring reference

The LS 19i3 should be mounted such that the measuring reference s is minimum 2000 mm. The larger the measuring reference is the smaller enters the uncertainty in measurement of the light screens and of the set-up into the calculation. With a measuring reference of e.g. 1000 mm an error in the measuring base of 2 mm corresponds to 0.2 %. On the other hand an error in the measuring base of 2 mm at a measuring reference of 2000 mm corresponds to only 0.1 %.

The start light screen should be placed from the muzzle of the weapon so far that the exhausted muzzle flash and the powder steam cannot reach the start light screen.

For the admeasurement of the measuring reference you should use a calibrated measuring tape with mm-division. For this purpose the light screen LS 19i3 is provided with marks right and left next to the optics on the measuring head and on the reflective foil.

For mounting of the LS 19i3 you have different possibilities:

2.1 Mounting with Supports

The light screens LS 19i3 are mounted on the provided supports (sketch 2). The supports with the light screens are placed in the trajectory such that the trajectory is in the center of the free opening of the light screen frame. Thereby the light screen marked with A1 is used for the start and that one marked with A2 for the stop. The measuring heads always face the butt. Then the light screens are measured with the measuring tape at all four corners and aligned. Also here the principal is valid the smaller the measuring reference is the more precisely it must be aligned.

With a double measuring reference the two outer and two inner light screens form a measuring reference in each case. Thereby the light screens designated with A, B, C, etc. are a measuring reference in each case. 1 (A1, B1, C1) always is the start light screen, 2 (A2, B2, C2) always the stop light screen. You have to take care for that the outer measuring reference is measured before the inner measuring reference is placed in its position.

2.2 Mounting with Spacing Bars

With the aid of spacing bars the light screens LS 19i3 can be used as a fixed single or double measuring reference (sketches 3 and 4). The lengths of the measuring bases will be supplied custom configured. The standard length with a single measuring reference is 2000 mm and with a double measuring reference 2000 and 3000 mm. The measuring reference is mounted such that the measuring heads face the butt. The light screens designated with A, B, C, etc. form a measuring reference in each case. The light screens designated with 1 are used for the start and those designated with 2 are used for the stop.

With a single measuring reference (sketch 3) first the two upper spacing bars are screwed between the light screens. Then in each case two support holdings are shoved over the lower spacing bars and are screwed between the light screens. The support holdings are shoved to the extreme end of the spacing bars and the complete measuring reference is put on the supports.

DRELO

With a double measuring reference (sketch 4) first the inner spacing bars (2000 mm) and the shorter outer spacing bars (500 mm) are screwed on the inner measuring reference. Now the four support holdings are shoven over the lower outer spacing bars. Next the outer measuring reference is screwed at the spacing bars. The complete double measuring reference now is put on the four supports and fixed by screws.

The single or double measuring reference is placed in the trajectory of the projectiles such that the trajectory is in the center of the free opening of the light screens.

2.3 Mounting with Frame make "MERO" (sketch 5)

The Mero frame is available in different dimensions as single measuring reference with 2 m or 4 m measuring base and as double measuring reference with 2 m and 3 m or 4 m and 6 m measuring base.

After assembly of the Mero frame (see special description) the holding frames with the spring bolts are inserted in direction of the butt. The light screens are screwed on the spring bolts with the measuring heads in direction to the butt. Now the measuring references are measured and adjusted.

The Mero frame is placed in the trajectory of the projectile such that the trajectory is in the center of the free opening of the light screens.

2.4 Function of the Operating Elements

1. "IMP."
This LED lights up for approx. 3 seconds as soon as a pulse is triggered. This is also the case if you strike with your hand through the sensitive area for testing purposes.
2. "ANALOG"
An oscilloscope for checking of the analog signal can be connected to this socket. It is however also possible to connect a counter with positive analog input for the purpose of time measurement. Output impedance 100 Ohm, offset about +2 V.
3. "IMP.OUT"
To these output sockets (either BNC socket or 6 pole MS socket) you can connect our chronometers or velocity computers. The output pulse is a positive needle pulse of about 12 V.

DRELO

4. "IMP. OUT"

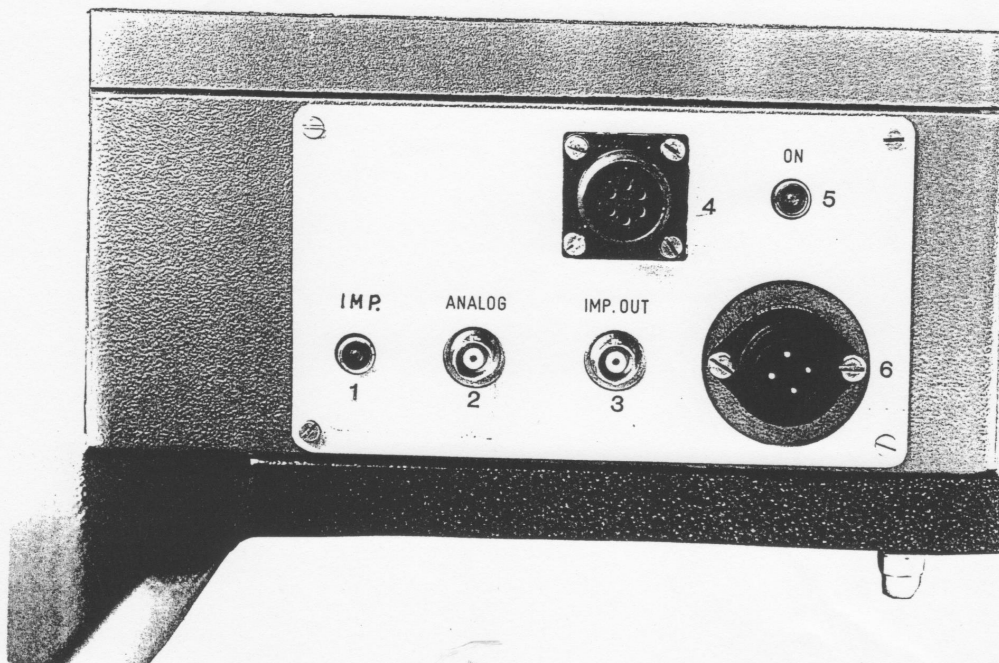
This 6 pole MS socket fulfills two requirements:

- a) the positive output pulse for starting or stopping of the chronometers or velocity computers is available at PIN A of this socket
- b) the measuring head can be fed separately from the chronometers, velocity computers or a battery via this socket. For this purpose a direct current of minimum 18 V to maximum 24 V is necessary. To PIN F + and to PIN E 0 V are connected.

5. "ON"

This LED indicates the readiness for measurement of the LS 19i.

6. This input socket serves for the connection of the mains voltage 220 V 40 - 60 Hz.



DRELO

3.0 Adjustment of the Measuring Head

When adjusting the measuring head pay attention to that no light emitted from lamps running on alternating current will fall into the measuring head with unscrewed cover. Furtheron it is important that the adjustments on the potentiometer are always made from the left to the right, i.e. clockwise direction.

3.1 Adjustment of the LED Current on the emitter board

Draw off the jumper S 1 and connect an ampere meter having 300 mm Amp. range between both pins. The current may be 195 - 200 mA and is adjusted with the potentiometer R11 (LED I). If this is no more possible the LED's have to be replaced. Then the ampere meter is removed and the jumper slipped on again.

3.2 Adjustment of the Analog Amplification

Connect an oscilloscope to the output 2 "ANALOG" and measure the share of the direct current of the analog signal. The share of the direct current must be 2 V. This is adjusted with the potentiometer R 9 (AN-V).

3.3 Adjustment of the Trigger Threshold

Connect a digital voltmeter between the pins 2 and 3 of IC 2 (IM 311). At this place the trigger threshold is adjusted with the potentiometer R 17 (TR) to a value of 2 V. The digital voltmeter consequently must indicate 2.000 V. Deviations of + 3 mV are permissible. Thereby the positive pole is connected with Pin 3 and the negative pole with Pin 2.

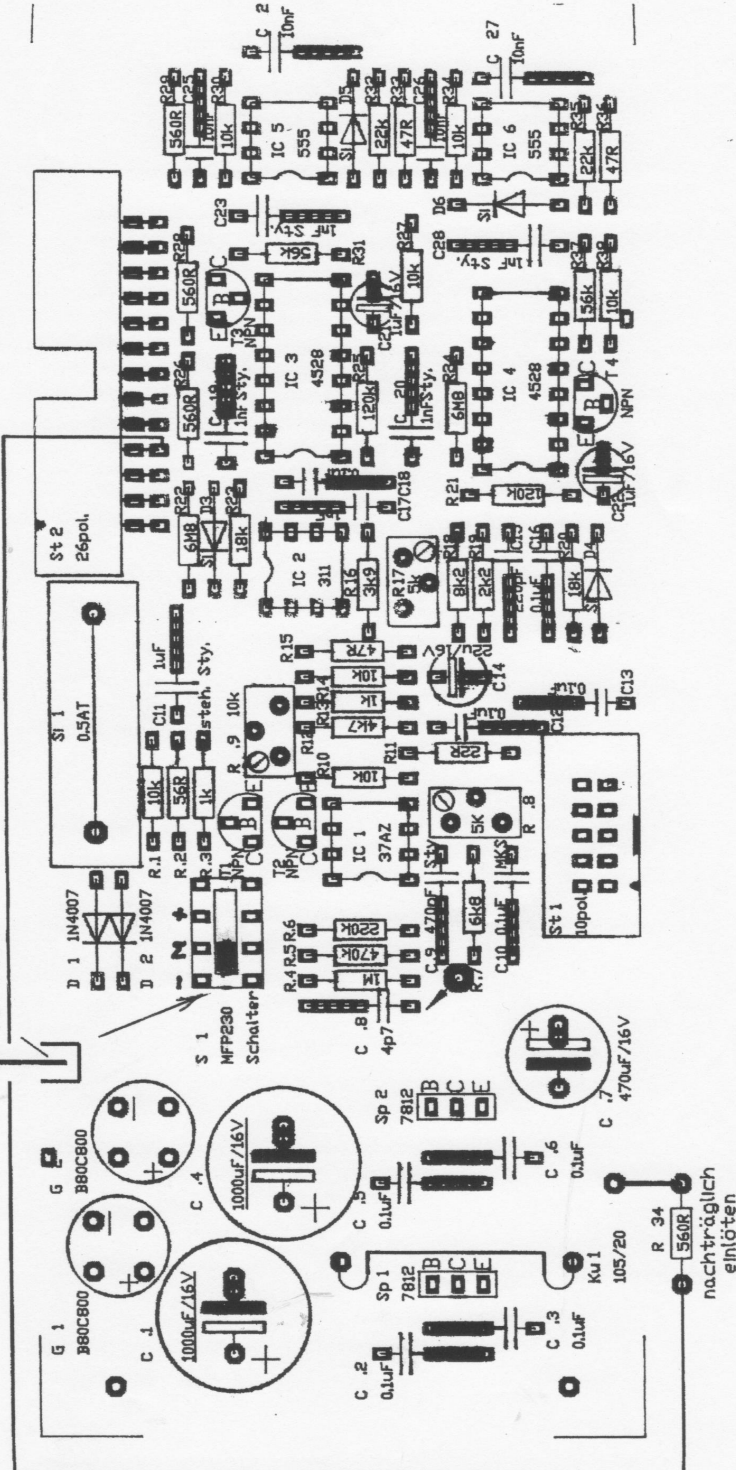
Erfa.
Ratk.

Type: LS 191 3

Ing. Paul Drewell
4050 Mönchengladbach 5 (Rheinland)
Fernruf 02161/59563 Telex 852361 drelo d

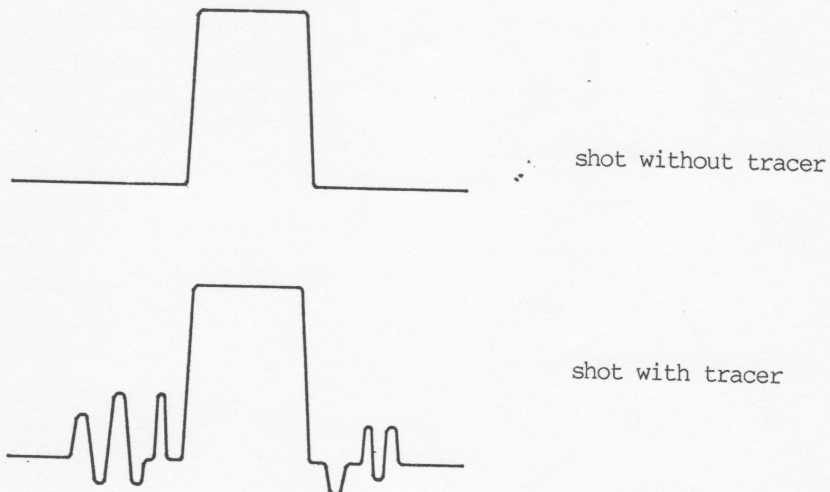
Art. Nr.: 2.0019.50002

Masse
Lötpunkt
freimachen
vom Lack



4.0 Measurement of Tracer Ammunition

With the velocity measurement of tracer ammunition limiting cases can occur through strong reflection where the measuring accuracy is no more guaranteed. In these cases the light screen LS 19i3 can be adjusted in that way that an unobjectionable measurement can be carried out. For the adjustment a storage oscilloscope is needed. The oscilloscope is connected to output 2 "ANALOG" and is operated in the AC range with a resolution of $5 \mu\text{s}/\text{cm}$. The height of the amplitude is adjusted to a value of $2 \text{ V}/\text{cm}$. Now the shot is fired and the analog curve is picked up. The record could look as follows:



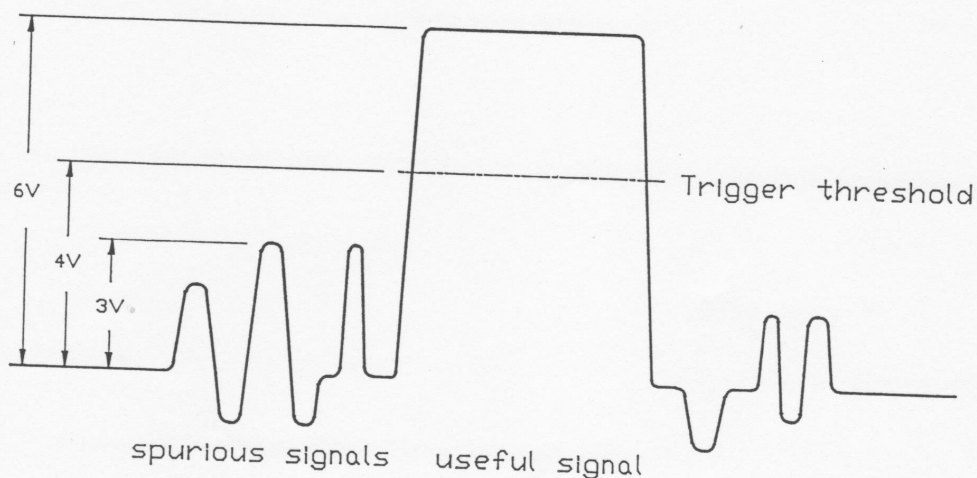
As can be seen from the pictures, the basic line in the first picture is very quiet and the useful signal is transmitted unobjectionable. With the second shot, however, severe oscillations can be seen before the useful signal on the base line. These are created by reflections of the tracer on walls and objects close to the light screens. It is therefore advisable to paint all objects close to the light screens black including walls, ceiling and ground. The oscillations before the useful signal thereby will be reduced considerably.

After termination of the record the pulses on the analog curve before the useful signal must be measured. They must not exceed the 2 V trigger threshold and should be minimum 0.5 V below the trigger threshold. If this should not be the case there are different possibilities to achieve this.

DRELO

1. Supposed the interference pulses are between 3 and 3.5 V and the useful signal of the shot is between 6 and 8 V the trigger threshold then is increased to 4.5 V as described under 3.3.

The trigger threshold must be adjusted such that it is minimum 0.5 V above the interference signals and minimum 0.5 V below the level of the useful signal.



2. If the adjustment as described under item 1. is not sufficient because the interference signals are above 4 V there is the possibility to reduce the amplification by the built-in jumper. For this purpose the jumper is set from "middle" to "low". Now a further record of a shot must be made and then proceed further as described under 1.

Should these two possibilities not be sufficient, it is advisable to contact our company and possibly send us a record of the analog curve.

4.1 Measurement of extreme bright tracer ammunition

During firing tests with extreme bright tracer ammunition the interference signals can have the same level as the useful signal. It is then no longer possible to separate these two signals. To be able to make a measurement also with this ammunition the LS 19i3 can be modified. For this purpose a special illuminating equipment LS 19i3/BEL with supply unit LS 19iL/NG and shields for the measuring heads are available (Option). With the use of this illuminating equipment the tracer is outshined by the light so that the projectile can now be detected by the light screen. The light screen thereby becomes very sensitive against movements of the air so that sound motions of the air will be registered. This means that subsonic projectiles like pistol ammunition will produce incorrect measurements. For this purpose, reflectors are delivered with the equipment which then are screwed on the illuminating equipment which then has to be switched off.

3.52.2e Bi*975/92

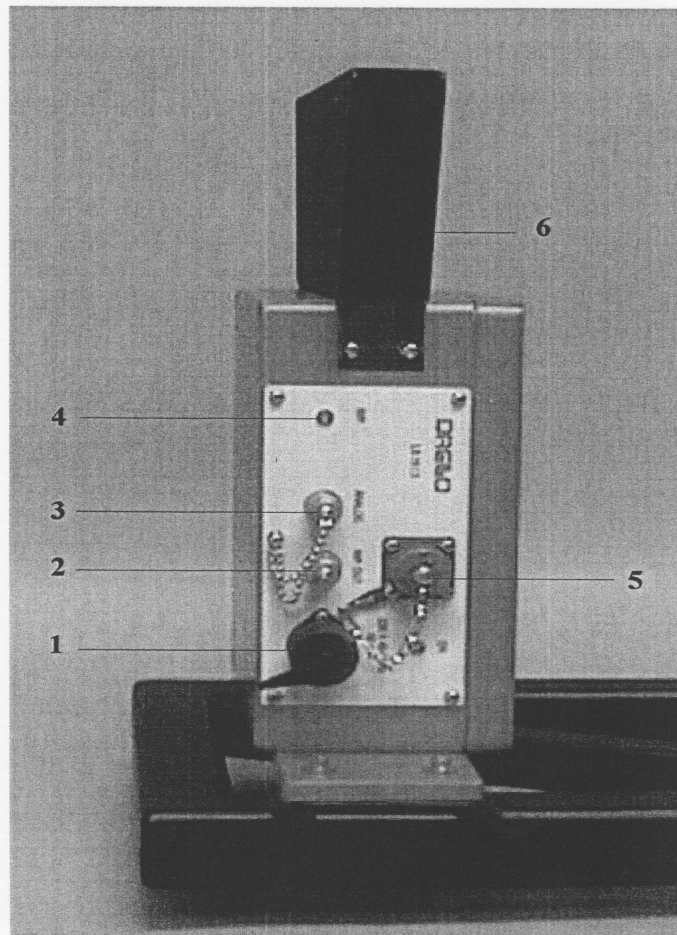
DRELO

5.0 Technical Data

Supply:	230 V 50 – 60 Hz or 18 – 24 V DC
Current Consumption:	40 mA with AC 300 mA with DC
Fuses:	63 mA slow (5 x 20 mm) for AC 500 mA slow (5 x 20 mm) for DC
Opening angle:	35° active
Inside dimension of frame:	750 mm x 840 mm
Shading:	minimum 2 %
Output pulse:	+12 V 30 µs
Output impedance:	120 Ohm
Dimensions:	830 mm 1110 mm x 135 mm
Weight:	12 kg
Accessories:	1 power cable, 3.5 m long

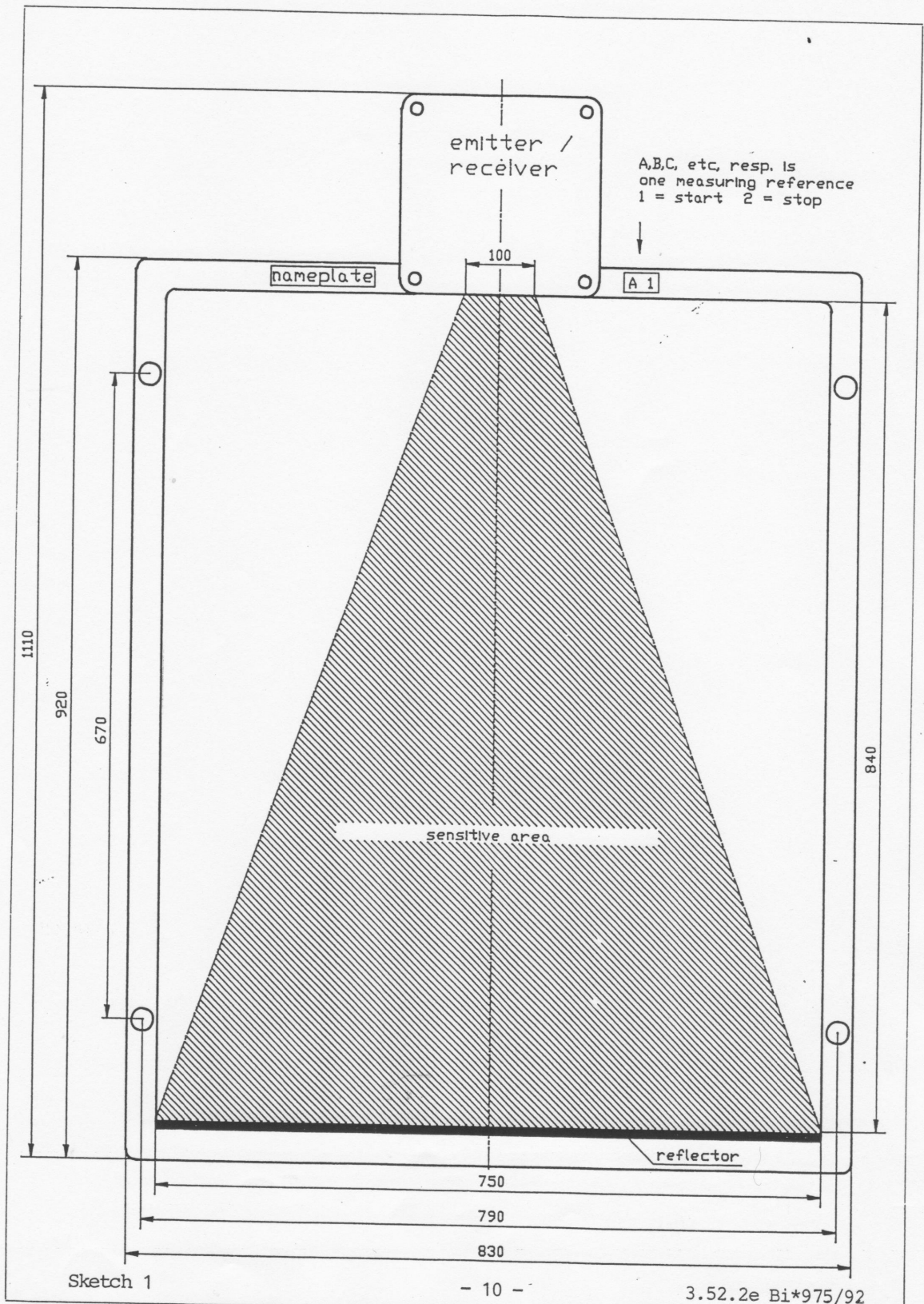
3.52.2e Bi*975/

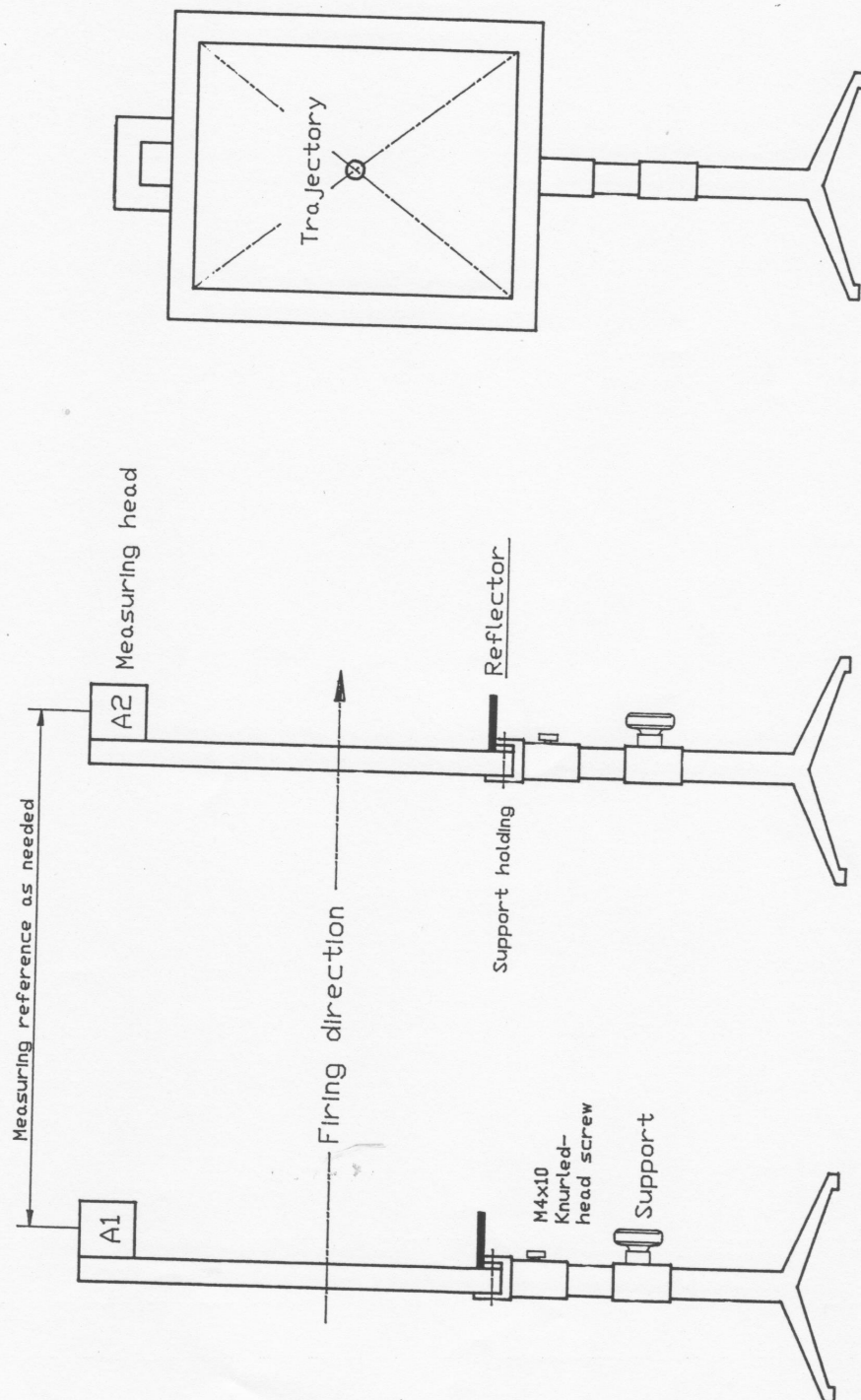
Sensor Head LS19i3



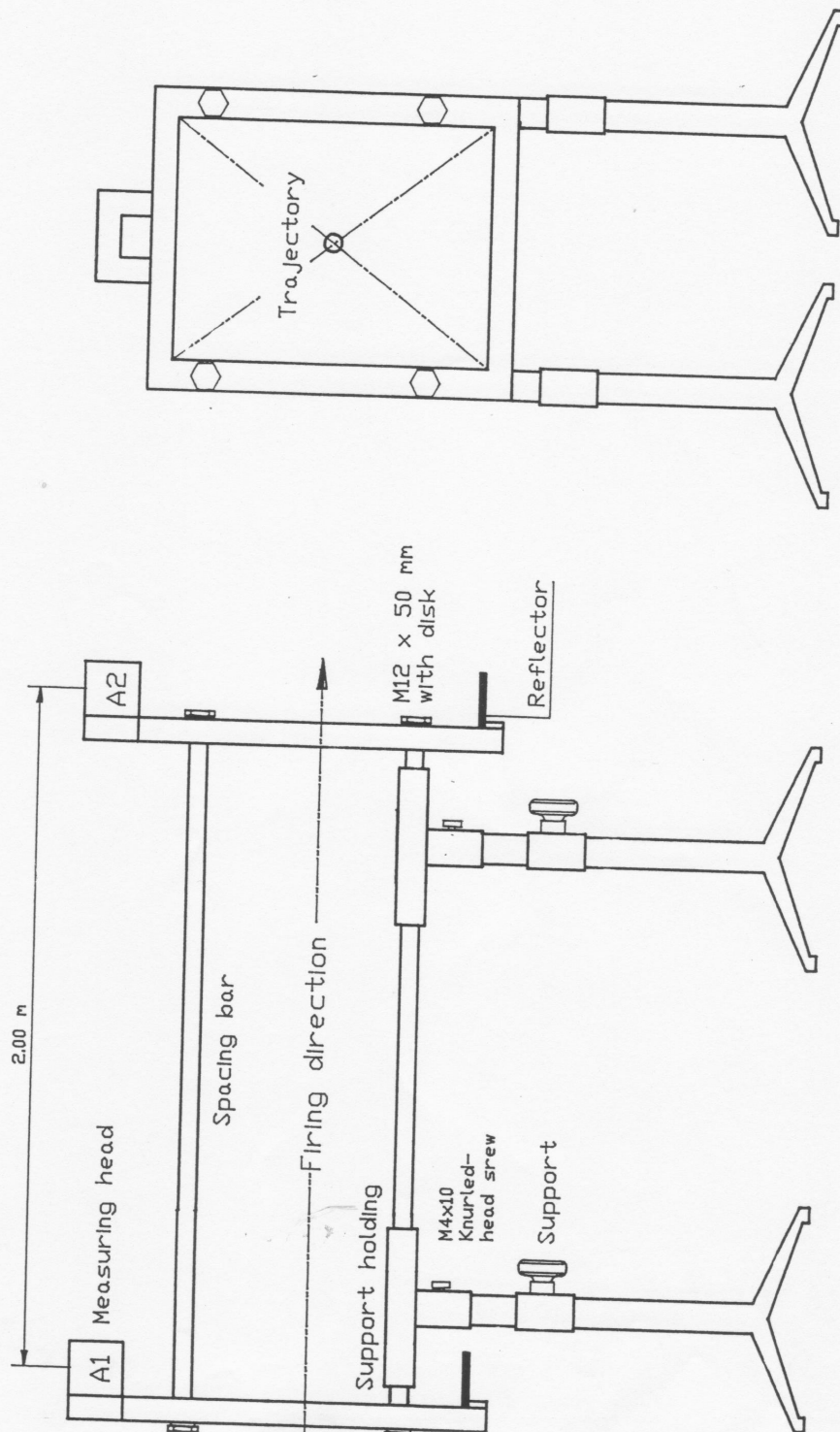
- 1 = Mains 230 V (ON)
- 2 = Digital Output
- 3 = Analog Output
- 4 = LED Signal Indicator
- 5 = Power Supply from: (Option)
 - Battery
 - DRELO Counters and INLOC 2040 with digital output
- 6 = Telecentric shield (Option)

DRELO
Ing. Paul Drewell





Sketch 2



Sketch 3

Bijlage E

Datasheet Photron Fastcam SA5

FASTCAM SA5

ULTRA HIGH-SPEED VIDEO SYSTEM



High-Speed Video System

Next generation CMOS sensor technology providing 7,500 fps, 1K x 1K pixels

The Photron FASTCAM SA5 will meet the requirements of the most demanding applications in research and development due to its unrivaled sensitivity, frame rate and resolution.

This high performance will permit the SA5 to be applied to areas of research once dismissed as unsuitable for digital high-speed imaging.

Building on the success of Photron's Emmy award winning high speed cameras our sensor design improves sensitivity, image quality and color reproduction.

The FASTCAM SA5 delivers mega-pixel resolution at 7,500 fps, an impressive maximum frame rate of 775,000 fps and a 1 microsecond exposure time. Optional one million fps and 369 ns shutter capabilities are available subject to export restrictions.

Target applications include:

- Materials research
- Ballistics
- Aerospace
- PIV
- Combustion
- Cavitation
- Fluid dynamics

Benefits

- Performance examples:
 - 1,024 x 1,000 pixels @ 7,500 fps
 - 512 x 512 pixels @ 25,000 fps
 - 256 x 256 pixels @ 87,500 fps
 - 128 x 128 pixels @ 262,500 fps
 - 128 x 24 pixels @ 775,000 fps
- Variable Region of Interest (ROI)
- Capture 12-bit uncompressed data
- 20µm pixels ensure best light sensitivity for demanding high-speed or low light applications
- Equivalent ISO light sensitivity 10,000 (monochrome), 4,000 (color) measured to ISO12232 S sat
- Phase lock to IRIG/GPS
- Composite and SDI video output for real time monitoring during set up, recording and playback
- Optional remote keypad control with integrated viewfinder
- 8GB, 16GB, 32GB or 64GB memory options
- Gigabit Ethernet interface

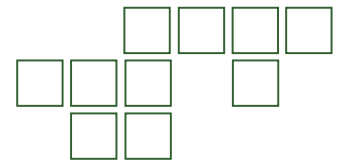


Photron

www.photron.com

FASTCAM SA5

ULTRA HIGH-SPEED VIDEO SYSTEM



Specifications: Partial Frame Rate / Recording Duration Table

FRAME RATE (fps)	MAXIMUM RESOLUTION Horizontal Vertical		MAXIMUM SHUTTER SPEED	RECORD DURATION (12-BIT)							
				TIME (Sec.)				FRAMES			
				8GB	16GB	32GB	64GB	8GB	16GB	32GB	64GB
1,000	1,024	1,024	1 μ s 1/1,000,000 sec	5.46	10.92	21.84	43.68	5,457	10,918	21,841	43,686
2,000	1,024	1,024		2.73	5.46	10.92	21.84	5,457	10,918	21,841	43,686
4,000	1,024	1,024		1.36	2.73	5.46	10.92	5,457	10,918	21,841	43,686
5,000	1,024	1,024		1.09	2.18	4.37	8.73	5,457	10,918	21,841	43,686
7,000	1,024	1,024		0.78	1.56	3.12	6.24	5,457	10,918	21,841	43,686
7,500	1,024	1,000		0.75	1.49	2.98	5.96	5,588	11,180	22,365	44,735
9,300	1,024	800		0.75	1.50	3.01	6.01	6,985	13,975	27,956	55,918
10,000	1,024	744		0.75	1.50	3.01	6.01	7,511	15,027	30,061	60,127
15,000	960	528		0.75	1.51	3.01	6.02	11,289	22,587	45,182	90,374
20,000	832	448		0.77	1.54	3.07	6.14	15,352	30,716	61,443	122,898
30,000	768	320		0.78	1.55	3.11	6.21	23,284	46,586	93,189	186,396
50,000	512	272		0.82	1.64	3.29	6.57	41,090	82,211	164,452	328,934
75,000	320	264		0.90	1.81	3.61	7.22	67,737	135,523	271,097	542,244
100,000	320	192		0.93	1.86	3.73	7.45	93,138	186,345	372,758	745,585
150,000	256	144		1.03	2.07	4.14	8.28	155,230	310,575	621,264	1,242,642
300,000	256	64		1.16	2.33	4.66	9.31	349,269	698,794	1,397,845	2,795,946
420,000	128	64		1.66	3.33	6.66	13.31	698,538	1,397,589	2,795,690	5,591,893
525,000	128	48		1.77	3.55	7.10	14.20	931,384	1,863,452	3,727,587	7,455,857
775,000	128	24		2.40	4.81	9.62	19.24	1,862,769	3,726,904	7,455,175	14,911,715
930,000	128	16	369 ns	3.00	6.01	12.02	24.05	2,794,154	5,590,357	11,182,762	22,367,573
1,000,000	64	16	1/2,712,000 sec	5.59	11.18	22.37	44.73	5,588,309	11,180,714	22,365,525	44,735,146

OPTION SUBJECT TO EXPORT LICENSE CONTROL RESTRICTIONS WHERE APPLICABLE

Sensor	12-bit ADC (Bayer system color, single sensor) with 20 μ m pixel	Event Markers	Ten user entered event markers mark specific events within the image sequence in real time. Immediately accessible through software
Shutter	Global electronic shutter from 16.7ms to 1 μ s independent of frame rate	Dual Speed Recording	Enables the recording speed to be changed up or down by a factor of 2, 4 or 8 during a recording
Lens Mount	Interchangeable F-mount and C-mount using supplied adapters	Trigger Modes	Start, End, Center, Manual, Random, Random Reset, Random Center, Random Manual and Duals Speed Recording
Extended Dynamic Range	Selectable in twenty steps (0 to 95% in 5% increments) to prevent pixel over-exposure	Saved Image Formats	JPEG, AVI, TIFF, BMP, RAW, PNG, MOV and FTIF. Images can be saved with or without image or comment data
Memory	8GB (standard: 5,457 frames @ maximum resolution) 16GB (option: 10,913 frames @ maximum resolution) 32GB (option: 21,841 frames @ maximum resolution) 64GB (option: 43,686 frames @ maximum resolution)	Data Display	Frame Rate, Shutter Speed, Trigger Mode, Date or Time, Status (Playback/Record), Real Time, Frame Count and Resolution
Video Output 1	NTSC/PAL composite VBS (BNC). Ability to zoom, pan and tilt within image via keypad. Live video during recording	Partitioning	Up to 64 memory segments for multiple recording in memory
Video Output 2	HD-SDI: HD-SDI 2 channel (BNC) digital output	Data Acquisition	Supports Photron MCDL and DAQ
Camera Control	Through optional keypad with integrated viewfinder and Gigabit Ethernet or RS-422	Cooling	Actively cooled
User Preset Switches	Four user selectable camera function controls mounted on the camera's rear panel	Operating Temperature	0 - 40 degrees C (32 - 104 degree F)
Low Light Mode	Low light mode drops the frame rate and shutter time to their maximum values, while maintaining other set parameters, to enable users to position and focus the camera	Mounting	1 x 1/4 - 20 UNC, 1 x 3/8 - 16 UNC, 6 x M6
Triggering	Selectable positive or negative TTL 5Vp-p or switch closure	Dimensions	165mm (6.50")H x 153mm (6.02")W x 242.5mm (9.55")D *excluding protrusions
Trigger Delay	Programmable delay on selected input and output triggers, 100ns resolution	Weight	6.2 kg (13.67 lbs)
Timing	Internal clock or external source	Power Requirements	100V - 240V AC ~ 1.5A, 50-60Hz DC operation 18 - 36 V DC, 100VA
Phase Lock	Enables cameras to be synchronized precisely together to a master camera or external source, such as IRIG/GPS time codes		

Specifications subject to change without notice

PHOTRON USA, INC.
9520 Padgett Street, Suite 110
San Diego, CA 92126-4446
USA
Tel: 858.684.3555 or 800.585.2129
Fax: 858.684.3558
Email: image@photron.com
www.photron.com

PHOTRON (EUROPE) LIMITED
The Barn, Bottom Road
West Wycombe, Bucks, HP14 4BS
United Kingdom
Tel: +44 (0) 1494 481011
Fax: +44 (0) 1494 487011
Email: image@photron.com
www.photron.com

PHOTRON LIMITED
Fujimi 1-1-8
Chiyoda-Ku, Tokyo 102-0071
Japan
Tel: +81 (0) 3 3238 2107
Fax: +81 (0) 3 3238 2109
Email: image@photron.co.jp
www.photron.co.jp

Photron

SLOW MOTION IMAGING SOLUTIONS

Bijlage F

Tabel snelheidsvariatie 7,62 x 51 mm NATO

Nummer schot	v1(m/s)	v2(m/s)	Δv (m/s)
1	850.9473	851.1047	0.1574
2	854.4642	854.5931	0.1289
3	850.0462	850.2715	0.2253
4	852.5372	852.7513	0.2141
5	848.8839	849.0212	0.1373
6	854.7194	854.8685	0.1491
7	853.0495	853.1034	0.0539
8	854.2347	854.3893	0.1546
9	848.4801	848.5861	0.106
10	848.0543	848.1264	0.0721
11	845.7115	845.8368	0.1253
12	847.3378	847.5324	0.1946
13	850.4175	850.6104	0.1929
14	851.9816	852.1682	0.1866
15	856.2797	856.4057	0.126
16	854.9886	855.0913	0.1027
17	848.5259	848.6834	0.1575
18	855.5260	855.6464	0.1204
19	845.5562	845.6620	0.1058
20	857.5483	857.6949	0.1466
21	851.9473	852.1442	0.1969
22	843.1253	843.2905	0.1652
23	855.0876	855.1443	0.0567
24	861.2571	861.3237	0.0666
25	845.9321	846.0330	0.1009
26	853.4383	853.5662	0.1279
27	854.6882	854.8255	0.1373
28	855.7905	855.9330	0.1425
29	855.2885	855.3804	0.0919
30	855.7923	855.9052	0.1129
Gemiddelde	852.05457	852.189767	0.13519667
Mediaan(Q2)	852.79335	852.92735	0.1331
Standaardafwijking	4.1930405	4.18262007	0.04494349
Q1	848.6154	848.76785	0.10585
Q3	855.06285	855.13105	0.157475
minimum(Q0)	843.1253	843.2905	0.0539
maximum(Q4)	861.2571	861.3237	0.2253

Bijlage G

Foto's bij vergelijkende test

Deze bijlage herneemt de 25 foto's waarvan de data hernomen is in de tabel bij het hoofdstuk 'resultaten' in de thesis.

<u>Current Time</u> +00:00:00.000000 <u>File Info.</u> FASTCAM SA5 model 1000K-M2 30000 fps 1/145000 sec 768 x 312 Random-R 1 1 frames 0.000033 sec	
X:0377 Y:0225 R:116 G:116 B:116	Schot 1 @ 854 m/s 7,62 x 51 mm NATO
<u>Current Time</u> +00:00:00.000000 <u>File Info.</u> FASTCAM SA5 model 1000K-M2 30000 fps 1/145000 sec 768 x 312 Random-R 1 3 frames 0.0001 sec	
X:0376 Y:0196 R:117 G:117 B:117	schot 2 @ 853 m/s 7,62 x 51 mm NATO
<u>Current Time</u> +00:00:00.000000 <u>File Info.</u> FASTCAM SA5 model 1000K-M2 30000 fps 1/145000 sec 768 x 312 Random-R 1 3 frames 0.0001 sec	
X:0379 Y:0189 R:112 G:112 B:112	schot 3 @ 844 m/s 7,62 x 51 mm NATO

<u>Current Time</u> +00:00:00.000000 <u>File Info.</u> FASTCAM SA5 model 1000K-M2 30000 fps 1/145000 sec 768 x 312 Random-R 1 3 frames 0.0001 sec X:0379 Y:0201 R:111 G:111 B:111	 shot 4 @ 844 m/s 7,62 x 51 mm NATO
<u>Current Time</u> +00:00:00.000000 <u>File Info.</u> FASTCAM SA5 model 1000K-M2 30000 fps 1/145000 sec 768 x 312 Random-R 1 1 frames 0.000033 sec X:0369 Y:0233 R:116 G:116 B:116	 Schot 5 @ 452 m/s 7,62 x 51 mm NATO
<u>Current Time</u> +00:00:00.000000 <u>File Info.</u> FASTCAM SA5 model 1000K-M2 30000 fps 1/145000 sec 768 x 312 Random-R 1 10 frames 0.000333 sec X:0378 Y:0221 R:114 G:114 B:114	 shot 6 @ 845 m/s 7,62 x 51 mm NATO

Current Time

+00:00:00.000000

File Info.

FASTCAM SA5 model 1000K-M2

30000 fps

1/145000 sec

768 x 312

Random-R 1

10 frames

0.000333 sec

X:0378 Y:0221

R:114 G:114 B:114

shot 7 @ 851 m/s

7,62 x 51 mm NATO

Current Time

+00:00:00.000000

File Info.

FASTCAM SA5 model 1000K-M2

30000 fps

1/145000 sec

768 x 312

Random-R 1

10 frames

0.000333 sec

X:0378 Y:0221

R:113 G:113 B:113

shot 8 @ 856 m/s

7,62 x 51 mm NATO

Current Time

+00:00:00.000000

File Info.

FASTCAM SA5 model 1000K-M2

30000 fps

1/145000 sec

768 x 312

Random-R 1

10 frames

0.000333 sec

X:0378 Y:0221

R:112 G:112 B:112

shot 9 @ 843 m/s

7,62 x 51 mm NATO

Current Time

+00:00:00.000000

File Info.

FASTCAM SA5 model 1000K-M2

30000 fps

1/145000 sec

768 x 312

Random-R 1

10 frames

0.000333 sec

X:0378 Y:0221

R:113 G:113 B:113

schot 10 @ 839 m/s

7,62 x 51 mm NATO

Current Time

+00:00:00.000000

File Info.

FASTCAM SA5 model 1000K-M2

30000 fps

1/145000 sec

768 x 312

Random-R 1

10 frames

0.000333 sec

X:0378 Y:0221

R:115 G:115 B:115

schot 11 @ 854 m/s

7,62 x 51 mm NATO

Current Time

+00:00:00.000000

File Info.

FASTCAM SA5 model 1000K-M2

30000 fps

1/145000 sec

768 x 312

Random-R 1

10 frames

0.000333 sec

X:0378 Y:0221

R:114 G:114 B:114

schot 12 @ 853 m/s

7,62 x 51 mm NATO

<u>Current Time</u> +00:00:00.000000 <u>File Info.</u> FASTCAM SA5 model 1000K-M2 30000 fps 1/145000 sec 768 x 312 Random-R 1 10 frames 0.000333 sec	
X:0378 Y:0221 R:114 G:114 B:114	shot 13 @ 848 m/s 7,62 x 51 mm NATO
<u>Current Time</u> +00:00:00.000000 <u>File Info.</u> FASTCAM SA5 model 1000K-M2 30000 fps 1/145000 sec 768 x 312 Random-R 1 10 frames 0.000333 sec	
X:0378 Y:0221 R:116 G:116 B:116	shot 14 @ 845 m/s 7,62 x 51 mm NATO
<u>Current Time</u> +00:00:00.000000 <u>File Info.</u> FASTCAM SA5 model 1000K-M2 30000 fps 1/145000 sec 768 x 312 Random-R 1 10 frames 0.000333 sec	
X:0378 Y:0221 R:113 G:113 B:113	Schot 15 @ 855 m/s 7,62 x 51 mm NATO

<u>Current Time</u> +00:00:00.000000 <u>File Info.</u> FASTCAM SA5 model 1000K-M2 30000 fps 1/145000 sec 768 x 312 Random-R 1 1 frames 0.000033 sec	
X:0353 Y:0231 R:112 G:112 B:112	Schot 16 @ 412 m/s 9 x 19 mm Parabellum
<u>Current Time</u> +00:00:00.000000 <u>File Info.</u> FASTCAM SA5 model 1000K-M2 30000 fps 1/145000 sec 768 x 312 Random-R 1 9 frames 0.0003 sec	
X:0354 Y:0197 R:112 G:112 B:112	Schot 17 @ 416 m/s 9 x 19 mm Parabellum
<u>Current Time</u> +00:00:00.000000 <u>File Info.</u> FASTCAM SA5 model 1000K-M2 30000 fps 1/145000 sec 768 x 312 Random-R 1 9 frames 0.0003 sec	
X:0354 Y:0197 R:112 G:112 B:112	Schot 18 @ 396 m/s 9 x 19 mm Parabellum

<u>Current Time</u> +00:00:00.000000 <u>File Info.</u> FASTCAM SA5 model 1000K-M2 30000 fps 1/145000 sec 768 x 312 Random-R 1 9 frames 0.0003 sec	
X:0352 Y:0232 R:119 G:119 B:119	Schot 19 @ 406 m/s 9 x 19 mm Parabellum
<u>Current Time</u> +00:00:00.000000 <u>File Info.</u> FASTCAM SA5 model 1000K-M2 30000 fps 1/145000 sec 768 x 312 Random-R 1 9 frames 0.0003 sec	
X:0352 Y:0232 R:116 G:116 B:116	Schot 20 @ 404 m/s 9 x 19 mm Parabellum
<u>Current Time</u> +00:00:00.000000 <u>File Info.</u> FASTCAM SA5 model 1000K-M2 30000 fps 1/145000 sec 768 x 312 Random-R 1 9 frames 0.0003 sec	
X:0352 Y:0232 R:117 G:117 B:117	Schot 21 @ 414 m/s 9 x 19 mm Parabellum

Current Time

+00:00:00.000000

File Info.

FASTCAM SA5 model 1000K-M2

30000 fps

1/145000 sec

768 x 312

Random-R 1

9 frames

0.0003 sec

X:0352 Y:0232

R:115 G:115 B:115

Schot 22 @ 402 m/s

9 x 19 mm Parabellum

Current Time

+00:00:00.000000

File Info.

FASTCAM SA5 model 1000K-M2

30000 fps

1/145000 sec

768 x 312

Random-R 1

9 frames

0.0003 sec

X:0352 Y:0232

R:117 G:117 B:117

Schot 23 @ 385 m/s

9 x 19 mm Parabellum

Current Time

+00:00:00.000000

File Info.

FASTCAM SA5 model 1000K-M2

30000 fps

1/145000 sec

768 x 312

Random-R 1

9 frames

0.0003 sec

X:0352 Y:0232

R:117 G:117 B:117

Schot 24 @ 382 m/s

9 x 19 mm Parabellum

Current Time

+00:00:00.000000

File Info.

FASTCAM SA5 model 1000K-M2

30000 fps

1/145000 sec

768 x 312

Random-R 1

9 frames

0.0003 sec

X:0352 Y:0232

R:118 G:118 B:118

Schot 25 @ 381 m/s

9 x 19 mm Parabellum

Auteursrechtelijke overeenkomst

Ik/wij verlenen het wereldwijde auteursrecht voor de ingediende eindverhandeling:

Ontwerp van een realtime triggersysteem toegepast op de fotografie van een ballistisch projectiel

Richting: **master in de industriële wetenschappen: energie-automatisering**

Jaar: **2014**

in alle mogelijke mediaformaten, - bestaande en in de toekomst te ontwikkelen - , aan de Universiteit Hasselt.

Niet tegenstaand deze toekenning van het auteursrecht aan de Universiteit Hasselt behoud ik als auteur het recht om de eindverhandeling, - in zijn geheel of gedeeltelijk -, vrij te reproduceren, (her)publiceren of distribueren zonder de toelating te moeten verkrijgen van de Universiteit Hasselt.

Ik bevestig dat de eindverhandeling mijn origineel werk is, en dat ik het recht heb om de rechten te verlenen die in deze overeenkomst worden beschreven. Ik verklaar tevens dat de eindverhandeling, naar mijn weten, het auteursrecht van anderen niet overtreedt.

Ik verklaar tevens dat ik voor het materiaal in de eindverhandeling dat beschermd wordt door het auteursrecht, de nodige toelatingen heb verkregen zodat ik deze ook aan de Universiteit Hasselt kan overdragen en dat dit duidelijk in de tekst en inhoud van de eindverhandeling werd genotificeerd.

Universiteit Hasselt zal mij als auteur(s) van de eindverhandeling identificeren en zal geen wijzigingen aanbrengen aan de eindverhandeling, uitgezonderd deze toegelaten door deze overeenkomst.

Voor akkoord,

Wijnants, Jurgien

Datum: **6/06/2014**