

THE ROLE OF THE SCAFFOLDING PROTEIN INAD IN LOCALIZATION OF  
SIGNALING COMPLEXES TO THE RHABDOMERES OF  
*DROSOPHILA* PHOTORECEPTORS

APPROVED BY SUPERVISORY COMMITTEE

Rama Ranganathan, M.D. Ph.D.

---

Paul Blount, Ph.D.

---

Alfred Gilman, M.D., Ph.D.

---

Ege Kavalali, Ph.D.

---

## DEDICATION

To my friends and family, to my son Preston,  
and to my husband, Randy.

THE ROLE OF THE SCAFFOLDING PROTEIN INAD IN LOCALIZATION OF  
SIGNALING COMPLEXES TO THE RHABDOMERES OF  
*DROSOPHILA* PHOTORECEPTORS

by

ADRIENNE R. HAHN

DISSERTATION OR THESIS

Presented to the Faculty of the Graduate School of Biomedical Sciences

The University of Texas Southwestern Medical Center at Dallas

In Partial Fulfillment of the Requirements

For the Degree of

DOCTOR OF PHILOSOPHY

The University of Texas Southwestern Medical Center at Dallas

Dallas, Texas

November, 2004

Copyright

by

Adrienne R. Hahn, 2004

All Rights Reserved

THE ROLE OF THE SCAFFOLDING PROTEIN INAD IN LOCALIZATION OF  
SIGNALING COMPLEXES TO THE RHABDOMERES OF  
*DROSOPHILA* PHOTORECEPTORS

Publication No. \_\_\_\_\_

Adrienne R. Hahn, Ph.D.

The University of Texas Southwestern Medical Center at Dallas, 2004

Supervising Professor: Rama Ranganathan, M.D., Ph.D.

Organization of proteins into macromolecular complexes is one way cells maximize the speed, specificity, and efficiency of signal transduction. In fruit fly photoreceptors InaD, a scaffolding protein containing 5 PDZ domains, organizes proteins involved in the visual signaling pathway into complexes within a microvillar stack of membranes known as the rhabdomere. Light activation of rhodopsin activates a signaling cascade via a Gq-coupled reaction that quickly opens  $\text{Ca}^{++}$ -selective TRP channels. Subsequent  $\text{Ca}^{++}$  influx activates eye protein kinase C (ePKC), and calmodulin, which in turn modulate the activity of other visual proteins. Mutants in which the InaD / TRP association has been disrupted (*inaD*<sup>215</sup>) phenocopy the delayed inactivation of mutants lacking ePKC, suggesting that one of the functions of InaD includes localizing ePKC to its downstream targets such as the TRP channel.

There are currently two different models for scaffolding proteins: the “beads on a string” or “tethering” model where the order of the binding domains and their respective binding partners is unimportant, and the “specific quaternary structure” model where the

specific stereochemical orientation of the domains is vital for proper signaling. The latter model also allows for allosteric regulation of binding.

We assess the “beads on a string” vs. the “specific quaternary structure” model for InaD, a scaffolding protein found in the photoreceptor cells of fruit flies, by analyzing the characteristics of the light response in flies expressing two InaD constructs where the order of the PDZ domains has been shuffled. Based on biochemical and electrophysiological data on these mutants, we conclude that the “specific quaternary structure” model applies best to InaD. In addition, we investigate whether the *inaD*<sup>215</sup> phenotype is due to displacement of ePKC from microdomains of calcium initiated by TRP channels, by calcium imaging of photoreceptors expressing visual proteins tagged with CaMgaroo, a Ca<sup>++</sup>-sensitive derivative of yellow fluorescent protein. After conducting these experiments, we conclude that this is not the case. Rather, the *inaD*<sup>215</sup> phenotype is most likely due to the inability of ePKC to phosphorylate TRP and attenuate its activity. These results also support the “specific quaternary structure” model for InaD.

## TABLE OF CONTENTS

Signature page .....	i
Dedication .....	ii
Title page .....	iii
Copyright .....	iv
Abstract .....	v
Table of Contents .....	vii
Prior Publications .....	x
List of Figures .....	xi
List of Tables .....	xiii
List of Appendices .....	xiv
List of Terms .....	xv
1. Introduction .....	16
1.1 Macromolecular organization of the Photoreceptor .....	16
1.1.1 The Role of Scaffolding Proteins in Cell Signaling .....	16
1.1.2 Two Models for Scaffolding Proteins .....	20
1.1.3 PDZ Domains.....	23
1.1.4 Invertebrate Phototransduction.....	26
1.2 The Role of Calcium in Feedback Regulation of the Light Response.....	38
1.2.1 InaD Complex-Independent Calcium-Mediated Feedback Regulation.....	40
1.2.2 InaD Complex-Dependent Calcium-Mediated Feedback Regulation.....	42

1.3 Motivations for this Dissertation .....	43
2. Functional Importance of Domain Order in InaD-Mediated Scaffolding of Visual Transduction.....	45
2.1 Analysis of InaD PDZ domain Rearrangement .....	45
2.1.1 Introduction.....	45
2.1.2 Generation of Flies Expressing InaD PDZ Domain Rearrangements.....	46
2.1.3 Stability of InaD Complex Over Time.....	47
2.1.4 Co-immunoprecipitation of InaD Complex from Fly Heads.....	49
2.1.5 Phenotype of Macroscopic Responses in <i>p34512</i> Photoreceptors.....	50
2.1.6 Phenotype of Macroscopic Responses in <i>p51234</i> Photoreceptors.....	56
2.2 Conclusions.....	58
3. Calcium Imaging of <i>Drosophila</i> Photoreceptors .....	61
3.1 Introduction .....	61
3.1.1 Calcium microdomains .....	61
3.1.2 Calcium Imaging of Fly Photoreceptors .....	64
3.2 Generation of Flies Expressing Camgaroo-Tagged Visual Proteins.....	66
3.2.1 Quantitative Western Blot Analysis of Camgaroo Fly Lines.....	67
3.2.2 Immunofluorescence Analysis of Camgaroo Flies.....	69
3.2.3 Fluorescence Imaging of Dissociated Camgaroo Photoreceptors.....	71
3.2.4 Electrophysiological analysis of Camgaroo Photoreceptors.....	71
3.2.5 <i>In vitro</i> Calcium Titration Curves.....	74
3.3 Fluorescence Measurements of Camgaroo Dissociated Ommatidia.....	75



3.3.1 Comparison of Flies Expressing Camga-Arr2, InaD-Camga, and cyto-Camga	75
3.3.2 Comparison of Flies Expressing InaD-Camga and InaD <sup>215</sup> -Camga	80
3.4 Conclusions	82
4. Conclusions and Recommendations	85
4.1 Analysis of InaD PDZ domain Rearrangement	85
4.1.1 Conclusions	85
4.1.2 Recommendations	86
4.2 Fluorescence Measurements of Camgaroo in Dissociated Ommatidia	87
4.2.1 Conclusions	87
4.2.2 Recommendations	88
5. Methodology	90
5.1 Molecular Biology and Biochemistry	90
5.2 Electrophysiology	97
5.3 Calcium Imaging	99
Appendix A: Light Response Acquisition program	101
Appendix B: Matlab M File Routines	146
Bibliography	154
Curriculum Vitae	161

## PRIOR PUBLICATIONS

“Purification, Characterization, Synthesis, and Cloning of the Lockjaw Peptide from *Conus purpurascens* Venom.” Ki-Joon Shoon, Michelle M. Grilley, Maren Marsh, Doju Yoshikami, Adrienne R. Hall [Hahn], Boris Kurz, William R. Gray, Julita S. Imperial, David R. Hillyard, and Baldomero M. Olivera. Biochemistry **34**:4913-4918, 1995.

## LIST OF FIGURES

Figure 1.1: Organization of the Post-Synaptic Density .....	19
Figure 1.2: Replacement of Yeast/Ste5/Kinase Interactions .....	22
Figure 1.3: Structure of PSD-95 (PDZ3) Bound to Peptide .....	24
Figure 1.4: Morphology of the <i>Drosophila</i> Compound Eye.....	29
Figure 1.5: The <i>Drosophila</i> Phototransduction Pathway.....	30
Figure 1.6: The Structure of the InaD Complex .....	34
Figure 1.7: Calcium-Mediated Feedback Regulation in the Fly Photoreceptor .....	39
Figure 2.1: Western Blot Analysis of InaD PDZ Domain Rearrangments.....	47
Figure 2.2: Western Blot Analysis of InaD Complex Stability .....	48
Figure 2.3: Western Blot Analysis of NorpA Stability in <i>p34512</i> Flies.....	49
Figure 2.4: Comparison of Wild Type ( <i>yw</i> ), <i>p34512</i> , and <i>inaD<sup>l</sup></i> Light Responses .....	51
Figure 2.5: Comparison of Wild Type ( <i>yw</i> ), <i>p34512</i> , <i>inaD<sup>l</sup></i> Light Response Parameters...	52
Figure 2.6: Light Intensity Response Curves of Wild Type, <i>p34512</i> , and <i>inaD<sup>l</sup></i> .....	54
Figure 2.7: Comparison of Wild Type ( <i>yw</i> ), and <i>p51234</i> Light Responses .....	56
Figure 2.8: Comparison of Wild Type ( <i>yw</i> ), and <i>p51234</i> Light Response Parameters .....	57
Figure 3.1: Western Blot Analysis of Camgaroo Expression.....	68
Figure 3.2: Quantitative Western Blot Analysis of Camgaroo Expression .....	69
Figure 3.3: Immunofluorescence Analysis of Camgaroo Fly Eyes.....	70
Figure 3.4: Fluorescence Imaging of Dissociated Camgaroo Photoreceptors.....	72
Figure 3.5: Patch Clamp Analysis of Camgaroo Flies.....	73
Figure 3.6: <i>In vitro</i> Calcium Titration Curve for Camgaroo Protein.....	74

Figure 3.7: Fluorescence Imaging of Camgaroo Dissociated Ommatidia .....	76
Figure 3.8: Time to Peak Fluorescence and Fold Change in Camgaroo Flies.....	77
Figure 3.9: <i>In vivo</i> Calcium Titration of Camgaroo Dissociated Ommatidia .....	79
Figure 3.10: Time to Peak Fluorescence and Fold Change in Wild Type vs. <i>inaD</i> <sup>215</sup> Flies...81	
Figure 4.1: Fluorescence Imaging of <i>camga-arr2</i> Photoreceptors in Living Flies.....	89
Figure 5.1: Design of Shuffled InaD Constructs .....	91

## LIST OF TABLES

Table 2.5: Comparison of Wild Type ( <i>yw</i> ), <i>p34512</i> , <i>inaD<sup>l</sup></i> Light Response Parameters ....	53
Table 2.8: Comparison of Wild Type ( <i>yw</i> ), and <i>p51234</i> Light Response Parameters.....	58
Table 3.8: Time to Peak Fluorescence and Fold Change in Camgaroo Flies .....	77
Table 3.10: Time to Peak Fluorescence and Fold Change in Wild Type vs. <i>inaD<sup>215</sup></i> Flies ..	82

## LIST OF APPENDICES

APPENDIX A: Light Response Acquisition Program .....	99
APPENDIX B: Matlab M File Routines.....	142

## LIST OF FLY GENOTYPES

*yw* – white eyed yellow body color variant

*inaD*<sup>215</sup> – point mutation in InaD which mislocalizes the TRP channel

*inaD*<sup>1</sup> – loss of function mutation of InaD

*inaD*<sup>2</sup> – point mutation in InaD which mislocalizes NorpA (PLC-β)

*inaC* – loss-of-function eye-specific Protein Kinase C mutant

*norpA*<sup>P41</sup> – loss of function phospholipase C β mutant

*p34512* - InaD mutant with PDZ domains in order 3-4-5-1-2

*p51234* - InaD mutant with PDZ domains in order 5-1-2-3-4

*camga-arr2* – Fly expressing Camgaroo fusion to N-terminus of Arrestin2

*camga-inaD* – Fly expressing Camgaroo fusion to N-terminus of InaD

*inaD-camga* – Fly expressing Camgaroo fusion to C-terminus of InaD

*inaD*<sup>215</sup>- *camga* – Fly expressing Camgaroo fusion to C-terminus of InaD<sup>215</sup>

*dgq-camga* – Fly expressing Camgaroo fusion to internal loop of dgq

*cyto-camga* – Fly expressing cytosolic Camgaroo

## **CHAPTER ONE**

### **Introduction**

#### **MACROMOLECULAR ORGANIZATION OF THE PHOTORECEPTOR**

##### The Role of Scaffolding Proteins in Cell Signaling

Organization of proteins into macromolecular complexes is one way cells such as neurons and epithelial cells maintain their polarity and maximize the efficiency, specificity, and sensitivity of signal transduction. The colocalization of proteins involved in a particular signaling pathway enhances the efficiency and speed of the process by reducing the distance of interaction between receptors and their downstream effectors; thus increasing the effective concentrations of enzymes and their substrates, and protecting unstable intermediates. This strategy also reduces non-specific interactions, and prevents cross-talk with other competing signaling pathways. Signaling pathways containing clustered receptors have also been shown to be more sensitive to stimulation than those containing unclustered receptors (Bray, Levin et al. 1998). There are numerous examples of macromolecular complexes of signaling proteins anchored by various scaffolding proteins, including the protein-rich post-synaptic density (PSD) in mammalian neurons, the apical membrane of epithelial cells, the mating and osmotic regulation pathways in yeast, and the visual transduction pathway of invertebrate photoreceptor cells.



### *The Role of PSD-95 in Mammalian Excitatory Neurons*

Neurons exhibit a distinct polarity, with dendritic structures that receive inputs from other neurons at one end, a long axon that relays signals, and synaptic buttons that transmit signals to neighboring neurons. The organization of neuronal proteins into specifically localized complexes aids in maintaining this polarity, as well as enhances the efficiency and speed of transmission of information. The synapses of vertebrates contain many types of scaffolding proteins that organize the post-synaptic density into a dense three-dimensional network of complexes containing receptors and their effectors.

One of the scaffolding proteins, PSD-95, is a member of the MAGUK (membrane-associated guanylate kinases) family, and contains three PDZ domains, an SH3 domain, and a guanylate kinase homology (GuK) domain (Harris and Lim 2001; Zhang and Wang 2003). PSD-95 interacts with the plasma membrane via a palmitoyl moiety at its N-terminus. The first and second PDZ domains bind the C-terminus of the NR2B subunit of the *N*-methyl-D-aspartic acid (NMDA) glutamate receptor as well as Stargazin, which interact with AMPA( $\alpha$ -amino-3-hydroxy-5-methyl-4-isoxazolepropionic acid)-type glutamate receptors (Harris and Lim 2001; Zhang and Wang 2003) (see Figure 1.1). The second PDZ domain also can form a heterodimeric interaction with a PDZ domain of neuronal nitric oxide synthase (nNOS) (Brenman, Chao et al. 1996). The binding of calcium/calmodulin activates nNOS, and its localization near a source of calcium influx (the NMDA receptor) may optimize its production of the second messenger NO in response to activation of NMDA receptors. The third PDZ domain of PSD-95 binds the cell adhesion protein Neuroligin or the cytoskeletal protein CRIPT. The GuK domain of PSD-95 interacts with other PSD

proteins, including kainate receptors, a microtubule-associated protein called MAP1A, and GKAP (guanylate kinase-associated protein), which binds a scaffolding protein called Shank. PSD-95 is thought to cluster these receptors at the synapse; however knock-outs of PSD-95 in mice do not show dramatic neurological or developmental defects, possibly because of overlapping functions of related MAGUK proteins (Migaud, Charlesworth et al. 1998).

GRIP is another scaffolding protein found in the PSD that anchors AMPA receptors, contains 7 PDZ domains, and also possibly interacts with Shank. Considered a “scaffold of scaffolds”, Shank localizes metabotropic glutamate receptors, kainate receptors, AMPA receptors, and NMDA receptors via direct or indirect interaction with their respective scaffolding proteins (Harris and Lim 2001; Zhang and Wang 2003). According to subcellular localization studies, Shank resides in a region in the PSD farther away from the plasma membrane than PSD-95 (Naisbitt, Kim et al. 1999). Shank contains an ankyrin repeat, an SH3 domain, a PDZ domain, a proline-rich domain, and a SAM (sterile  $\alpha$  motif) domain. Through its association with the cytoskeleton via its ankyrin repeat and other scaffolding proteins, Shank helps maintain the superstructure of the synapse and allows for cross-talk between members of the glutamate receptor family within the PSD.

Similar scaffolding arrangements occur in different classes of neurons as well as in epithelial cells of vertebrates. Neuronal and epithelial cells of invertebrates such as *C. elegans* and *Drosophila melanogaster* also contain macromolecular complexes anchored by scaffolding proteins. The presence of these complexes throughout the metazoan world suggests that scaffolding proteins evolved in order to maximize the speed and efficiency

**Figure 1.1: Organization of the Post-Synaptic Density**

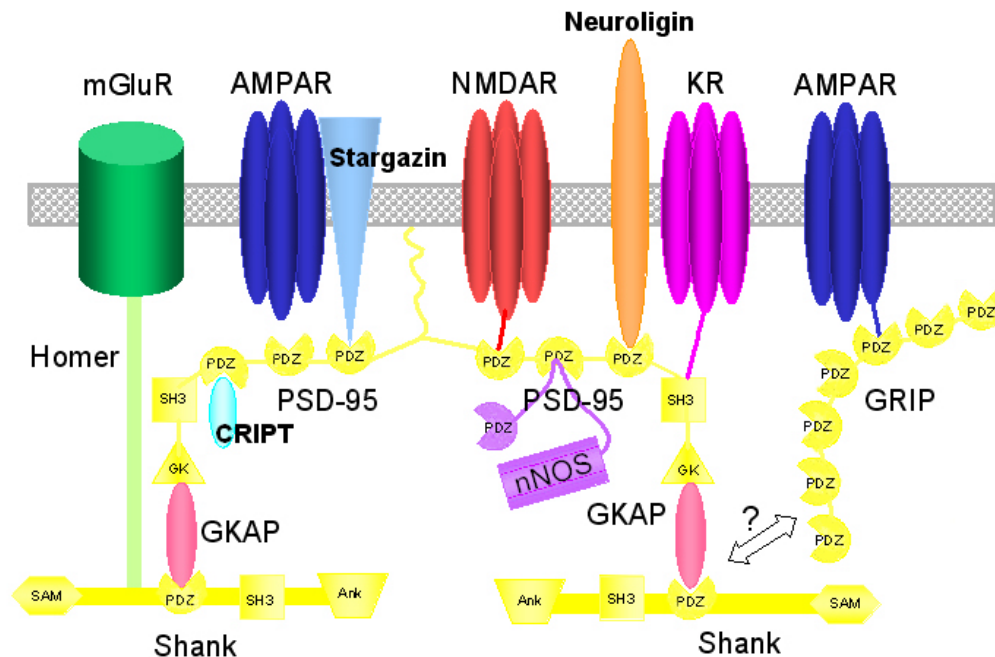


Figure 1.1: Organization of the post-synaptic density. The C-terminal domains of AMPA receptors (AMPA), NMDA receptors (NMDAR), Stargazin, Neurologin, GKAP, and CRIP1 bind the PDZ domains of scaffolding proteins (yellow) PSD-95, Shank, or GRIP. PSD-95 binds an internal domain of neuronal nitric oxide synthase (nNOS). Kainate receptors (KR) bind the SH3 domain of PSD-95, while the guanylate kinase homology domain (GK) binds GKAP. Homer binds metabotropic glutamate receptors (mGluR) and Shank at specific domains (not pictured). The interaction between GRIP and Shank is not well-characterized. The ankyrin domain (Ank) of Shank interacts with the cytoskeleton (not pictured).

of signaling in specialized cells. One of the major unanswered questions is whether these scaffolding proteins play a passive role in signal transduction by simply anchoring various components, or are they more actively involved in regulation of signaling?

## Two Models for Scaffolding Proteins

There are currently two different models for scaffolding proteins: the “beads on a string” or “tethering” model (Lim 2002) where the order of the binding domains and their respective binding partners is unimportant, and what we term the “specific quaternary structure” model where the precise organization of the domains is vital for proper signaling. The distinction between these two models is in the stereochemical specificity. The former model implies that scaffolding non-specifically localizes components to overcome diffusion limits to reaction speed and to enhance signaling selectivity. The latter model proposes a more detailed degree of order; scaffolding not only localizes, but also creates a specific quaternary complex that is required for proper signaling. Mechanistically, the quaternary complex could juxtapose enzymes and substrates, or permit allosteric regulation between scaffold components that requires a specific 3-D structure of the complex. For instance, it may be that the scaffolding protein allosterically regulates binding of component proteins based on binding site occupancy, and other factors such as phosphorylation state or the binding of calmodulin.

### *MAPK Pathways in *Saccharomyces cerevisiae**

The two above models were assessed in the MAPK (mitogen-activated protein kinase)-regulated mating pathways in yeast. Mutational ablation of several of the binding interactions of the scaffolding protein Ste5 resulted in a non-functional mating pathway and lack of diploid growth on appropriate media (Park, Zarrinpar et al. 2003). This phenotype was largely rescued by replacing the ablated interactions with heterologous

binding interactions, specifically the PDZ heterodimerization interaction of nNOS and syntrophin. Syntrophin was fused to the scaffolding protein Ste5 or one of its interacting proteins and nNOS was fused to another interacting protein in several different arrangements (see Figure 1.2A and B). While the mating pathway was restored, signaling efficiency, amplitude of response, and reaction kinetics were decreased compared to wild-type (see Figure 1.2C and D). Based on the incomplete rescue of the non-functional mating phenotype, Park et al. argue that their results support the “tethering” model in that precise ordering of Ste5 is not necessary for function. However, there may be a secondary factor that is fine-tuned through evolution to optimize speed and efficiency of signaling.

In addition, Park et al. showed that they could generate new signaling pathways in yeast by simply recombining portions of Ste5 with another scaffolding protein called Pbs2. They theorized that additional signaling pathways and phenotypes could have evolved in organisms through recombination or fusion events occurring in scaffolding proteins. In their view, scaffolding proteins are similar to promoters in that they are modular and flexible, and confer an amount of evolvability on the organism that may increase its fitness in response to evolutionary pressures.

High evolvability would most likely be a characteristic of scaffolding proteins that fit the “tethering” model, rather than the “specific quaternary structure” model. If a scaffolding protein simply tethered its components, its functionality would be more amenable to random mutation or the shuffling of the order of the binding domains. Because the mating response in yeast is a relatively slow and uncomplicated MAP kinase signaling pathway, the precise orientation of the signaling components appears to be

**Figure 1.2: Replacement of Yeast Ste5/Kinase Interactions with Heterologous PDZ Domain Interactions**

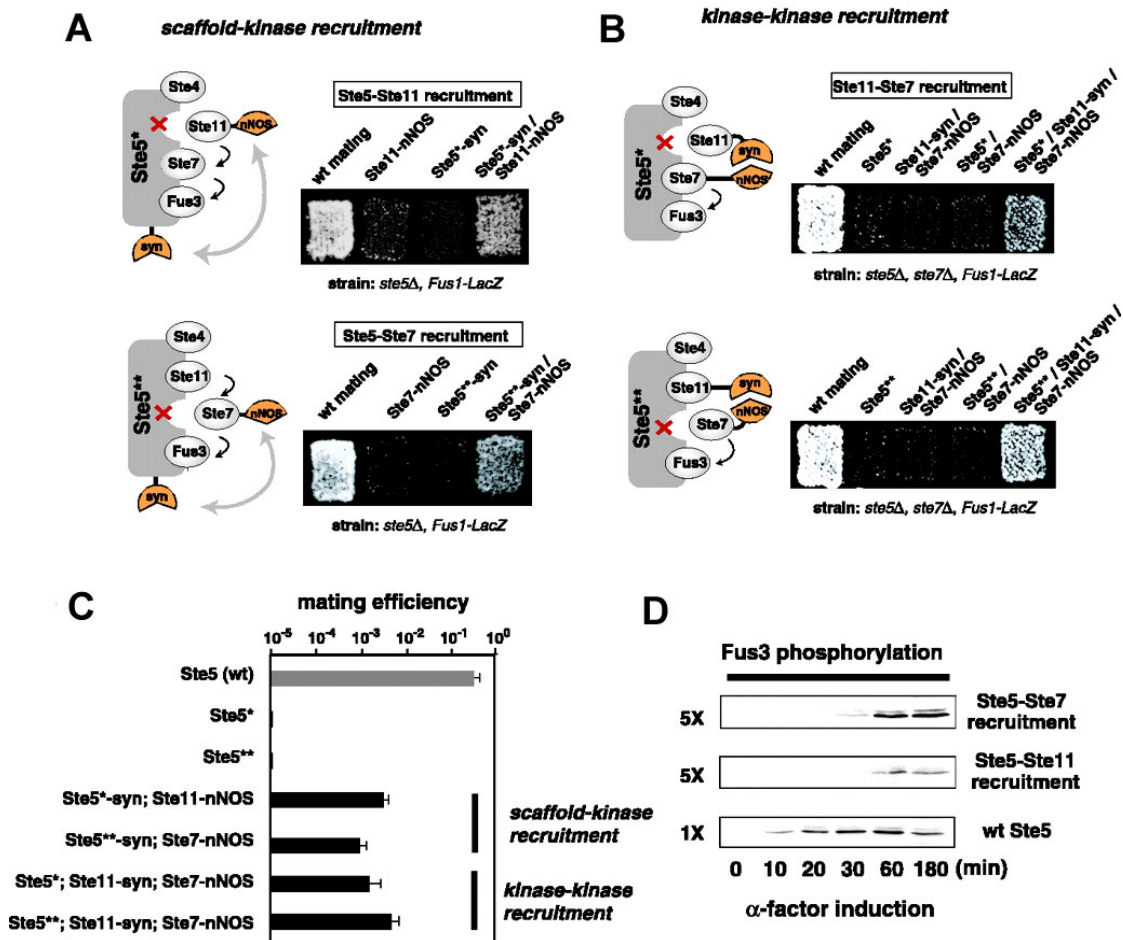


Figure 1.2: Replacement of yeast Ste5 /kinase protein-protein interactions with heterologous PDZ domain interactions. A) Complementation of mating deficiency in *ste5Δ* cells occurs when native kinase interactions are replaced with PDZ-mediated scaffold-kinase interactions as indicated by diploid growth on media. B) Complementation of mating deficiency in *ste5Δ* cells occurs when native kinase interactions are replaced with PDZ-mediated kinase-kinase interactions as indicated by diploid growth on media. C) Quantitative mating assays for rescued scaffolding pairs. D) Detection of phosphorylation of endogenous Fus3 protein conducted to assay for flux through mating MAPK cascade. [Courtesy of W. Lim]

unnecessary in generating the response. In cells that generate more complex and high speed signals, such as neurons, precise orientation and perhaps allostery of signaling

components may be required to generate a measurable signal. In chapter two of this dissertation, we test the veracity of these two models in the *Drosophila* visual system, focusing on the role of the multi-PDZ domain scaffolding protein InaD. Therefore, PDZ domains and *Drosophila* vision are discussed below in further detail.

### PDZ Domains

PDZ domains are one of the most common modular protein-interaction domains and are found primarily in metazoans (Harris and Lim 2001). More than 400 PDZ domains have been identified in the human genome alone (Zhang and Wang 2003). The name PDZ is derived from the first letter of three proteins that were first discovered to contain these domains: PSD-95, a protein found in the post-synaptic density of neurons; DLG, the *Drosophila* Discs Large protein; and ZO-1 or zona occludens 1 protein involved in formation of tight junctions of epithelial cells (Harris and Lim 2001; Hung and Sheng 2002; Zhang and Wang 2003). PDZ domains are about 90 residues in length and bind specific C-terminal 5-residue motifs or structurally related internal motifs (Harris and Lim 2001; Hung and Sheng 2002; Zhang and Wang 2003).

Structures of several PDZ domains solved by crystallography or NMR reveal a small  $\beta$ -sandwich fold consisting of 6  $\beta$ -strands ( $\beta$ A- $\beta$ F) and two  $\alpha$ -helices ( $\alpha$ A and  $\alpha$ B) (see Figure 1.3)(Doyle et al. 1996; Harris and Lim 2001; Zhang and Wang 2003).

Peptide ligands bind the PDZ domain between the  $\beta$ B strand and  $\alpha$ B helix by a mechanism known as  $\beta$ -strand addition (Harrison 1996). That is, the peptide acts as an extra  $\beta$ -strand that is added onto the edge of a  $\beta$ -sheet in the PDZ domain, and forms

**Figure 1.3: Structure of PSD-95 (PDZ3) Bound to Peptide**

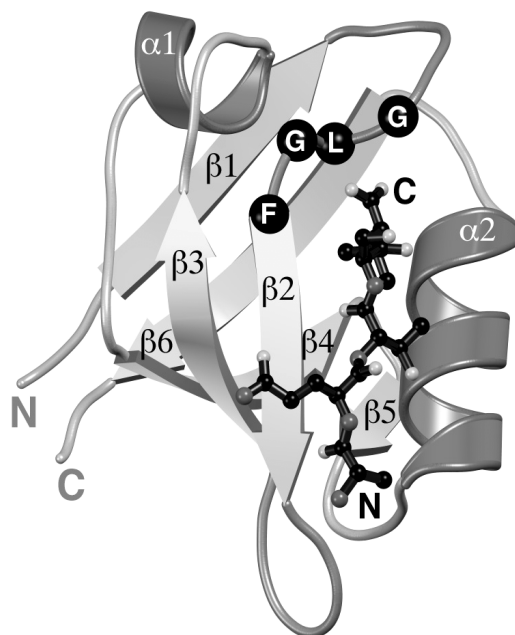


Figure 1.3: Structure of PSD-95 (PDZ3) bound to peptide. Pictured is a ribbon diagram of PSD-95 PDZ3 (residues 302-402) in complex with the peptide Thr-Lys-Asn-Tyr-Lys-Gln-Thr-Ser-Val (stick diagram). The original structure was solved by multiple isomorphous replacement to 1.8 Å (Doyle et al, 1996). The PDZ domain is a  $\beta$  sandwich consisting of 6  $\beta$  strands ( $\beta$ 1-6) and two  $\alpha$  helices ( $\alpha$ 1-2). The peptide binds in the groove between the  $\beta$ 5 strand and the  $\alpha$ 2 helix, and is orientated antiparallel to the  $\beta$ 5 strand. (PDB coordinates 1BE9).

hydrogen bonds similar to those observed between main chain carboxyl and amide groups in  $\beta$ -sheets.

The dissociation constant for PDZ domains falls in the low nanomolar to high micromolar range, depending on the method used (Harris and Lim 2001; Hung and Sheng 2002). The specificities of PDZ domains were determined by screens of peptide libraries (Songyang and Fanning 1997). By convention, the C-terminal residue of the peptide ligand is referred to as  $P_0$ , and the previous residues are termed  $P_{-1}$ ,  $P_{-2}$ , and so on. PDZ domains fall into four distinct binding classes based on their binding specificity for the  $P_0$  and  $P_{-2}$  residues of their ligands. Class I PDZ domains preferentially bind S/T-X- $\Phi$ -



COO<sup>-</sup> (where  $\Phi$  denotes a hydrophobic residue), Class II domains bind  $\Phi$ -X- $\Phi$ -COO<sup>-</sup>, and Class III domains bind X-X-C-COO<sup>-</sup>. PDZ domains that don't fall into the previous classes are classified as Class IV domains. PDZ domains that bind internal motifs that mimic the conformation of a chain terminus are also classified as Class IV domains (Harris and Lim 2001).

Proteins containing PDZ domains can also form heterodimers and homodimers. One of the best-studied examples is the heterodimerization of the PDZ domains of nNOS and PSD-95 or syntrophin (Harris and Lim 2001). The PDZ domain of nNOS contains a 30 residue C-terminal extension that forms a short two-stranded  $\beta$ -sheet. Upon binding to the PDZ domain of either PSD-95 or syntrophin, the  $\beta$ -sheet forms a rigid  $\beta$ -hairpin (or " $\beta$ -finger") that mimics the binding of a C-terminal peptide. An "E-T-T-F" motif interacts with the peptide-binding groove of the PDZ domain and the following hairpin turn replaces the carboxylate group. The PDZ domain of nNOS is still free to bind other partners and thus allows for multimerization of PSD-95, syntrophin, and nNOS. GRIP and a related scaffolding protein known as ABP (AMPA receptor binding protein) have also been shown to multimerize via their PDZ domains (Zhang and Wang 2003).

In PDZ-containing proteins, the PDZ domains are often grouped into tandem arrays of pairs and triplets. Although the significance of the groupings is not known, recent evidence suggests that adjacent PDZ domains may act in cooperation to bind their partners or induce folding of individual domains. In the case of syntenin-syndecan binding, although syndecans can bind either to PDZ1 or PDZ2 of syntenin, both PDZ domains are required for binding at either site (Grootjans, Reekmans et al. 2000). Likewise, a covalent bond between PDZ4 and PDZ5 of GRIP is required for proper

folding of both domains (Zhang, Fan et al. 2001). These data support the “specific quaternary structure model” for scaffolding proteins in that, at least in these examples, there is interdependence between adjacent PDZ domains of a protein.

### Invertebrate Phototransduction

In this dissertation, I utilize the photoreceptors of *Drosophila melanogaster* (commonly called fruit flies) as a system for studying the role of a scaffolding protein in a classic GPCR-mediated signaling pathway. InaD (inactivation no afterpotential D), a scaffolding protein containing 5 PDZ domains, organizes proteins involved in phototransduction into complexes within the photoreceptor cell. Light activation of rhodopsin activates a signaling cascade via a Gq (G protein subtype q) /PLC $\beta$  (phospholipase C subtype  $\beta$ ) reaction that quickly opens Ca<sup>++</sup>-selective TRP (transient receptor potential) channels. Subsequent Ca<sup>++</sup> influx activates eye protein kinase C (ePKC), and calmodulin, which in turn modulate the activity of other visual proteins. Well-established electrophysiological assays as well as numerous visual mutants make the fruit fly photoreceptor an ideal system for analysis of the importance of the formation of macromolecular complexes via scaffolding proteins.

The visual cycle of rhodopsin in *Drosophila* photoreceptor cells is similar to that in vertebrate photoreceptors with three important exceptions. First, unlike vertebrate rhodopsin *Drosophila* rhodopsin does not bleach, meaning the chromophore does not dissociate from the opsin moiety. Absorption of a second photon of a longer wavelength converts metarhodopsin back into the inactive R state of rhodopsin. Second, fly

rhodopsin is coupled to a Gq / PLC pathway, while vertebrate rhodopsin is coupled to a transducin / phosphodiesterase (PDE) pathway. Thus, in vertebrate photoreceptors cGMP channels close in response to light stimulation, while in fly photoreceptors TRP channels open. Lastly, rhodopsin phosphorylation is not required for arrestin binding to and inactivation of rhodopsin in fly photoreceptors. Flies that express a truncated form of rhodopsin in which the C-terminal phosphorylation sites are ablated appear to be no different from wild-type flies in electrophysiological assays (Vinos, Jalink et al. 1997).

The light responses of vertebrate and invertebrate photoreceptors also display many differences based on differences in cell morphology and their respective molecular pathways. Like vertebrate rod cells, *Drosophila* photoreceptors exhibit single photon sensitivity and the ability to “count” photons at low light intensities (Henderson, Reuss et al. 2000). Fruit fly photoreceptors have a greater range of sensitivity than vertebrate rods with an ability to handle up to  $10^6$  photons per second per cell, (Wu and Pak 1978; Howard, Blakeslee et al. 1987; Juusola and Hardie 2001) compared to approximately 500 photons per second in toad rod cells (Hardie 2001). The light response in flies is also 100 times faster than that of toads, and 10 times faster than that of mammals (Hardie 2001). The kinetics of the quantal light response (where one photon activates one rhodopsin) or quantum bump, are sharper for *Drosophila* photoreceptors than for toad rod cells. The increased speed and sharper kinetics of the fly light response may be due to the differences in the molecular pathway as well as the organization of the phototransduction machinery into signaling complexes within the microvilli via the scaffolding protein InaD. There is no known analogous scaffolding protein in vertebrate photoreceptor cells.

### *Cell Morphology*

The morphology of the invertebrate eye differs greatly from that of vertebrate eyes. The vertebrate eye consists of an organ that contains a corneal covering, iris, pupil, lens and retina with a layer of ciliary photoreceptors. Invertebrates, on the other hand, have compound eyes and microvillar photoreceptors (see Figure 1.4). The eye of *Drosophila melanogaster*, for example, is composed of 800 unit eyes called ommatidia. Each ommatidium consists of about 20 cells that form two layers: the cornea and the retina. The corneal layer contains a corneal lens cell, a pseudocone cell, pigment cells, and cone cells, while the retinal layer contains many pigment cells and 8 photoreceptor cells known as R1-R8. R1-6 cells are arranged radially with R7 and R8 stacked on top of each other in the center of the ommatidia (Montell 1999; Hardie and Raghu 2001).

The *Drosophila* photoreceptor cell contains two main parts: the cell body, which contains the nucleus, and the rhabdomere, a stack of microvillar extensions of the plasma membrane. The rhabdomere contains all the phototransduction machinery and serves as a photon collector. There are approximately 50,000 microvilli in cell R1-6 and 17,000 in cell R7 and R8. Each microvillus is 1-2  $\mu\text{m}$  long and 50 nm in diameter and contains a single actin filament extending the length of the microvillus. Cells R1-6 express one isoform of rhodopsin, Rh1, while Rh7-8 each express two different rhodopsin isoforms (Montell 1999; Hardie and Raghu 2001).

**Figure 1.4: Morphology of the *Drosophila* Compound Eye**

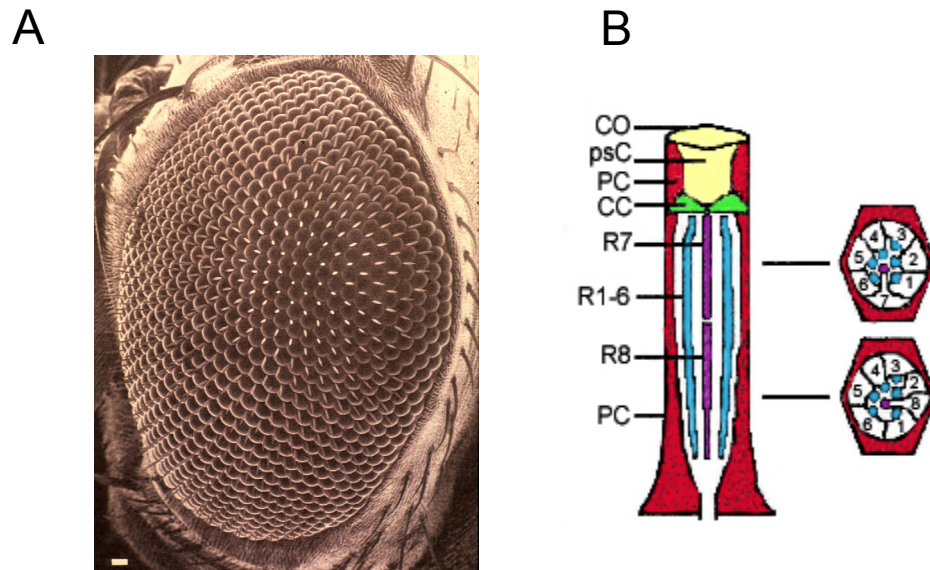


Figure 1.4: Morphology of *Drosophila* compound eye. A) Scanning electron micrograph of *Drosophila* compound eye with 800 unit eyes or ommatidia. B) Diagram of single ommatidium showing cluster of photoreceptor cells along with support cells and cross-section of ommatidium. CO = cornea and psC = pseudocone cell (yellow), PC = pigment cells (red), CC = cone cells (green), R1-6 = photoreceptor cells 1-6, R7 = photoreceptor cell 7, R8 = photoreceptor cell 8. Rhabdomeres of R1-6 shown in blue, rhabdomeres of R7 and R8 shown in purple.

### *The Molecular Pathway*

Rhodopsin (Rh) absorbs a photon, isomerizes its chromophore from 11-*cis*-3-hydroxy-retinal to all-*trans*-3-hydroxy-retinal, and converts to its active form, metarhodopsin (M). M then activates the heterotrimeric G-protein known as dgq in *Drosophila* by inducing exchange of GDP for GTP and dissociation of the  $\alpha$  subunit away from the  $\beta$  and  $\gamma$  subunits. The  $\alpha$  subunit of dgq remains active until it associates with and activates NorpA (no receptor potential A), an eye-specific phospholipase C- $\beta$ , which serves as a GAP (GTPase-activating protein) for dgq $\alpha$ . NorpA hydrolyzes PIP<sub>2</sub> (phosphatidylinositol-4,5-bisphosphate) into diacylglycerol (DAG) and inositol

triphosphate (IP<sub>3</sub>). This reaction causes calcium-selective TRP (transient receptor potential) channels and related non-selective cation TRPL channels to open, and the cell depolarizes as calcium and sodium ions flood the rhabdomere (see Figure 1.5)(Montell 1999; Hardie 2001).

**Figure 1.5: The *Drosophila* Phototransduction Pathway**

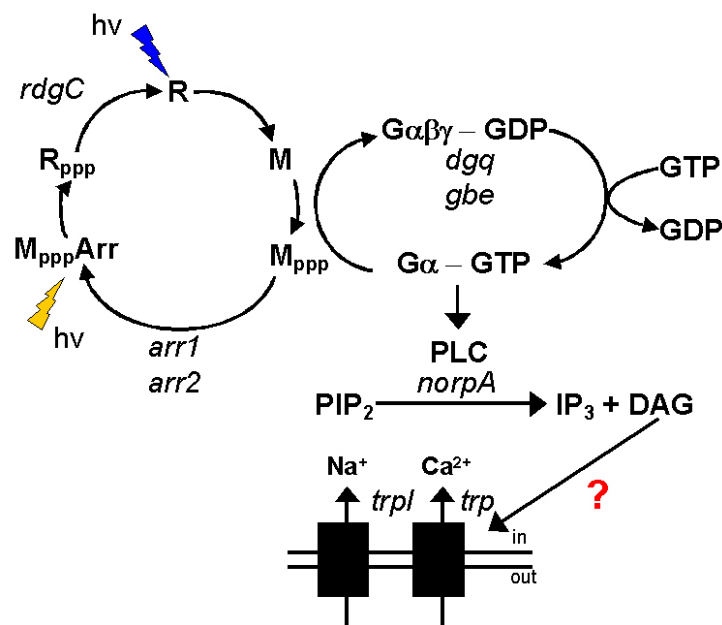


Figure 1.5: The *Drosophila* phototransduction pathway (see text for details). Absorption of a blue photon converts rhodopsin (R) into metarhodopsin (M).  $G\alpha$  exchanges GDP for GTP and then activates PLC. PLC hydrolyzes PIP<sub>2</sub> into IP<sub>3</sub> and DAG. TRP and TRPL channels then open and allow Ca<sup>2+</sup> and Na<sup>+</sup> to flow into the cell. M is phosphorylated multiple times and is inactivated by Arrestin (Arr). M can then be photoconverted back into R by absorbing an orange photon. Gene names shown in italics.

TRP and TRPL both belong to the TRP superfamily of cation channels that have been shown to play a role in calcium regulation in a wide variety of cell types. Many TRP-type channels are involved in sensory perception including heat and cold nociception, taste, smell, and mechanosensation (Harteneck and Plant 2000; Clapham

2003; Benham, Davis, et al. 2002; Voets, and Nilius 2003; Hardie 2003). *Drosophila* TRP is highly permeable to calcium ( $P_{Ca}:P_{Na}$  ca.100:1) and has a small conductance of only 8 pS, while TRP-like (TRPL) is less specific for calcium ( $P_{Ca}:P_{Na}$  ca.4:1) and has a larger conductance (40 pS) (Hardie 1991; Hardie and Minke 1992; Reuss, Mojet et al. 1997). TRP- $\gamma$ , a recently discovered putative third TRP homologue in *Drosophila* photoreceptors, may form heteromultimers with TRPL (Xu, Chien et al. 2000). As *trpl* and *trp $\gamma$*  mutants show very little defect in their light response, the functions of TRPL and TRP- $\gamma$  are as yet unclear (Reuss, Mojet et al. 1997).

The mechanism of gating of TRP and TRPL channels is not fully understood. Researchers initially believed that, as in most Gq / PLC pathways, the opening of TRP channels was dependent on inositol 1,4,5-trisphosphate ( $IP_3$ ). This is true in the case of another invertebrate, *Limulus polyphemus* or horseshoe crab (Fein 2003). Both  $IP_3$  production and the  $IP_3$  receptor appear to be unnecessary for phototransduction in *Drosophila* since caged  $IP_3$  release fails to activate the light response, and photoreceptor-specific knock-out of the  $IP_3$  receptor does not adversely affect the light response (Acharya and al. 1997; Hardie and Raghu 1998; Raghu and al. 2000).

Although DAG activates protein kinase C (PKC) in traditional PLC-mediated pathways, the activation phase of the fly light response is probably not mediated through PLC activation of PKC; this is based on mutants of an eye-specific PKC (ePKC) called InaC (Inactivation no afterpotential C), which have defects only in the deactivation and adaptation of the light response (Smith and al. 1991; Hardie and al. 1993). Rather, evidence indicates that DAG or a downstream derivative of DAG may cause opening of TRP channels, possibly through a direct interaction (Chyb, Raghu et al. 1999; Hofmann,

Obukhov et al. 1999). Also, a mutation in the *rdgA* gene, which encodes a DAG kinase responsible for producing phosphatidic acid from DAG, causes constitutive TRP channel activity and retinal degeneration (Masai, Okazaki et al. 1993; Raghu, Usher et al. 2000). These observations suggest DAG, not IP<sub>3</sub>, plays a role in the gating of TRP channels.

Other evidence supports a model for inhibition of TRP by PIP<sub>2</sub>. The hydrolysis of PIP<sub>2</sub> is thought to relieve inhibition of TRP channels caused by the binding of PIP<sub>2</sub> to the C-terminal domain of TRP, although no direct physical interaction between PIP<sub>2</sub> and TRP has been established (Hardie and al. 2001). Application of PIP<sub>2</sub> has been shown to inhibit heterologously expressed TRPL channels (Estacion, Sinkins et al. 2001), but TRPL is inactivated in *trp* mutants when PIP<sub>2</sub> has been depleted (Hardie and al. 2001). Under conditions of PIP<sub>2</sub> depletion, however, TRP channels are constitutively active which suggests that PIP<sub>2</sub> inhibition keeps TRP channels in their closed state (Hardie and al. 2001). For instance, since DAG kinase is also involved in the regeneration of PIP<sub>2</sub>, another interpretation of the *rdgA* mutant noted above is that PIP<sub>2</sub> depletion, not DAG accumulation, causes constitutive TRP activity.

### *The InaD Complex*

InaD (inactivation no afterpotential D) is the scaffolding protein located in the microvilli of the fly photoreceptor that organizes the phototransduction proteins into macromolecular complexes. It was discovered in 1970 in a mutant screen that pulled out a vision mutant known as *inaD*<sup>p215</sup> (Pak, Grossfield et al. 1970). Electrical recordings of this mutant revealed a defective inactivation of the light response, and the inability to maintain a light response known as a prolonged depolarizing afterpotential (PDA) after a



saturating light pulse. Identification of the gene in 1995 revealed a protein with homology to proteins containing PDZ domains (Shieh and Niemeyer 1995). InaD has 5 PDZ domains as well as a calmodulin-binding domain in between PDZ 1 and 2 (see Figure 1.6A).

Six different proteins involved in phototransduction have been shown to bind one or more of the 5 PDZ domains of InaD (see Figure 1.6B). The evidence for these interactions is stronger for some interactions than others. In the case of the PLC- $\beta$  (*norpA*), biochemical experiments suggest that both the first and fifth PDZ domains of InaD can bind either the C-terminal tail, or possibly an internal motif (Shieh, Zhu et al. 1997; van Huizen, Miller et al. 1998). Genetic evidence supports the binding of PLC- $\beta$  to PDZ5 of InaD, but not PDZ1. Flies that have a point mutation in PDZ5 of InaD (*inad*<sup>2</sup>) mislocalize PLC- $\beta$  and exhibit severe defects in activation and deactivation of the light response (Tsunoda and al. 1997; Scott and Zuker 1998). In addition, flies with a mutation in the C terminus of PLC- $\beta$  (*norpA*<sup>C1094S</sup>) display similar defects in activation and deactivation (Shieh, Zhu et al. 1997; van Huizen, Miller et al. 1998).

There has been some confusion regarding the importance of disulfide bonds and redox state in the proposed interaction of the C-terminus of PLC- $\beta$  with PDZ1. The original sequence of the last three residues of the C terminus of PLC- $\beta$  was published as F-C-A (Bloomquist, Shortridge et al. 1988), raising the possibility that perhaps the C terminus of PLC- $\beta$  formed a disulfide bond with PDZ1, which contains a cysteine near the binding pocket. Indeed, a group solved a structure of InaD PDZ1 bound to a peptide composed of the last 7 residues of PLC- $\beta$  (G-K-T-E-F-C-A), and the structure showed a

**Figure 1.6: The Structure of the InaD Complex**

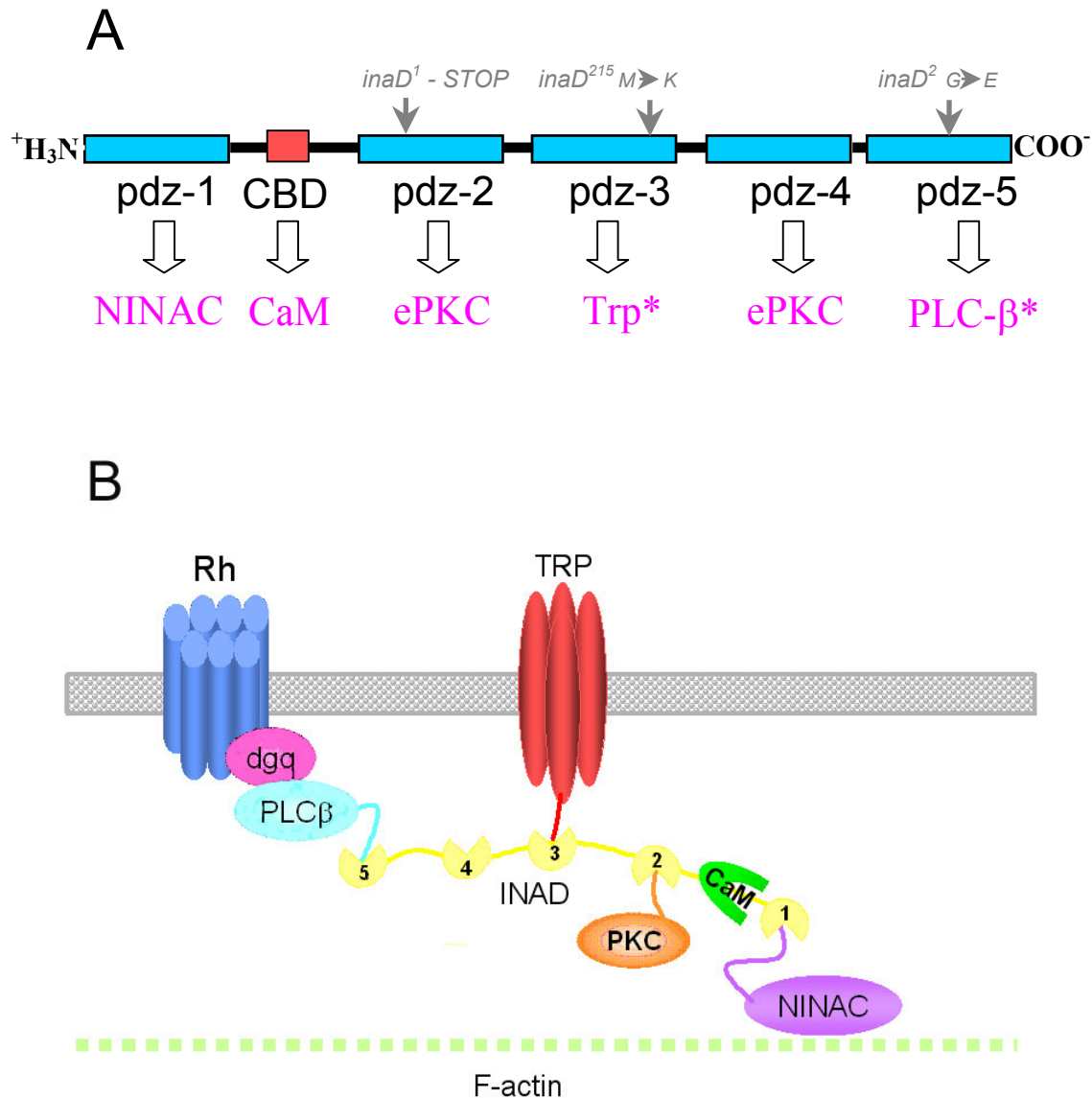


Figure 1.6: Structure of InaD complex. A) Schematic of InaD protein shown with proposed binding partners NINAC, calmodulin (CaM), eye-protein kinase C (PKC), TRP, and phospholipase C- $\beta$  (PLC- $\beta$ ) and their respective C-terminal residues. Also pictured are the approximate locations of the *inaD*<sup>1</sup>, *inaD*<sup>215</sup>, and *inaD*<sup>2</sup> mutations. Asterisk (\*) denotes genetic and biochemical evidence for interaction while no asterisk denotes biochemical evidence only. B) Schematic of InaD complex in the rhabdomere. InaD = PDZ domains 1-5, Rh = rhodopsin, PLC $\beta$  = phospholipase C- $\beta$ , PKC = protein kinase C, CaM = calmodulin.

disulfide bond between the Cys in the peptide and the Cys of PDZ1 (Kimple, Siderovski et al. 2001). Unfortunately, the sequence of *norpA* as published in the Fly Genome on Flybase and in Genbank, as well as established by sequencing of genomic *Drosophila* DNA in our lab, revealed the actual sequence of the C-terminus of PLC- $\beta$  to be G-K-T-E-F-Y-A (Prashant Mishra, personal communication). In light of this fact, one should consider the biochemical evidence showing an interaction between PDZ1 and PLC- $\beta$  as suspect, and abandon the possibility of dependence of the interaction on redox state.

Regulation of InaD interaction with PLC- $\beta$  by redox state may still be a possibility with regards to PDZ5, however. The crystal structure of InaD PDZ5 solved in our lab reveals an intermolecular disulfide bond that blocks the peptide binding site (Michael Socholich, personal communication). However, currently it is not known whether the disulfide bond is an artifact of crystallization conditions or if it is physiologically relevant.

The eye-specific protein kinase (ePKC) likewise has been shown to bind to InaD, and the interaction is required for proper localization and stability of ePKC. As with PLC- $\beta$ , the binding data from different research groups conflicts as some results showed ePKC binding PDZ 4 or PDZ3, while others showed binding to PDZ2 alone. (Huber, Sander et al. 1996; Adamski, Zhu et al. 1998; Xu, Choudhury et al. 1998; Kumar and Shieh 2001) The C-terminal sequence of ePKC is I-T-I-I and thus is classified as a PDZ Class I-binding motif. Mutation of the C-terminal Ile to an Asp (*inaC<sup>I700D</sup>*) renders ePKC unable to interact with INAD and mimics the phenotype of *inaC<sup>209</sup>* (Adamski, Zhu et al. 1998) The second PDZ domain is the only Class I PDZ domain in InaD (Kumar and Shieh 2001), but it is possible that ePKC may bind other types of PDZ domains or that

InaD may bind ePKC at an internal motif. Unfortunately, no mutants of InaD that specifically fail to bind ePKC are known at this time, so no genetic evidence supporting one interaction over another is available.

Both biochemical and genetic evidence support an interaction between PDZ3 of InaD and the C-terminus of TRP, which is required to localize the entire complex to the membrane (Huber, Sander et al. 1996; Shieh and Zhu 1996; Chevesich, Kreuz et al. 1997; Li and Montell 2000). The *inaD*<sup>p215</sup> mutant has a point mutation (M422K) in the third PDZ domain that prevents it from interacting with the C terminus of TRP, a Class IV PDZ binding motif (S-G-W-L), and causes mislocalization and eventual degradation of the entire InaD complex (Shieh and Zhu 1996) as well as a defect in response inactivation (Tsunoda and al. 1997). One explanation for the inactivation defect of the *inaD*<sup>215</sup> mutant is that ePKC is no longer in close enough proximity to phosphorylate TRP channels and possibly alter their gating properties. Alternatively, it is possible that ePKC requires a high local concentration of calcium for full activity and mislocalization of ePKC from the TRP channel decreases the local calcium concentration.

Biochemical experiments have shown that other molecules in the phototransduction pathway bind to InaD, including rhodopsin, an atypical myosin III (NinaC), TRPL, and calmodulin (Xu, Choudhury et al. 1998; Wes, Xu et al. 1999; Bahner, Sander, et al. 2002). NINAC is thought to bind PDZ1 via a PDZ class II binding motif (C-terminus (V-X-I)), but the interaction does not seem to be required for its localization to the rhabdomere (Wes, Xu et al. 1999). Calmodulin probably binds the CaM-binding motif of InaD between PDZ1 and PDZ2 (Xu and al. 1998). Some evidence has shown that TRPL binds either PDZ3 or PDZ4 (Xu, Choudhury et al. 1998), but the interaction

may be indirect via its interaction with TRP. Dgq $\alpha$  also interacts with the InaD complex indirectly via its interaction with PLC- $\beta$  (Bahner, Sander et al. 2000). The absence of InaD, as in the null allele *inaD<sup>l</sup>*, dramatically affects the light response by decreasing amplification of the photoisomerization event and broadening the latency distribution of quantum bumps which is reflected in a delay in activation and inactivation of the macroscopic response (Tsunoda and al. 1997; Scott and Zuker 1998).

Numerous biochemical and electrophysiological experiments indicate that InaD and TRP work together to localize signaling components to the microvilli. TRP tethers InaD to the membrane of the rhabdomere by its interaction with InaD, but InaD is required for retention rather than initial localization of TRP to the rhabdomere early in development (Chevesich, Kreuz et al. 1997). Likewise, InaD moves to the rhabdomere early in retinal development, but requires tethering to TRP for retention in the rhabdomere (Li and Montell 2000; Tsunoda, Sun et al. 2001). PLC- $\beta$  and e-PKC, however, require InaD interaction for initial localization and retention (Shieh, Zhu et al. 1997; vanHuizen, Miller et al. 1998). Since InaD, ePKC, TRP, and PLC- $\beta$  are present in equimolar amounts in photoreceptor cell membranes (Huber, Sander et al. 1996), InaD may also regulate the stoichiometry of signaling components in the complex.

Like PSD-95 in vertebrate neurons, some evidence suggests that InaD multimerizes to form large complexes of signaling components (Xu, Choudhury et al. 1998). Given that TRP probably forms tetramers *in vivo* based on sequence homology with other tetrameric channels (Harteneck and Plant 2000; Benham, Davis et al. 2002; Clapham 2003), at least four InaD proteins multimerize via their interaction with TRP. Each microvillus of the rhabdomere is estimated to contain 25-30 TRP channels (Huber,

Sander et al. 1996; Postma, Oberwinkler et al. 1999). Larger complexes may form via homo- and heterodimerization of the 3<sup>rd</sup> and 4<sup>th</sup> PDZ domains of InaD (Xu, Choudhury et al. 1998). At least 8 molecules of InaD have been shown to multimerize, but the evidence comes only from *in vitro* analyses and has not been proven to occur *in vivo*.

The question remains as to whether InaD simply localizes signaling components to the microvilli, or whether it plays a more a dynamic role in phototransduction. The order of the PDZ domains may aid in proper folding of the domains as well as allosteric regulation of binding partners. The excess of binding partners within the rhabdomere compared with the limited number of PDZ domains suggests that there may be a dynamic exchange of binding partners during phototransduction or as a result of adaptation. ePKC, one of the major players in adaptation, phosphorylates InaD possibly at multiple sites, which could be a mechanism for regulating the interactions between InaD and its numerous binding partners (Huber, Sander et al. 1996; Adamski, Zhu et al. 1998; Liu, Parker et al. 2000).

## **THE ROLE OF CALCIUM IN FEEDBACK REGULATION**

### **OF THE LIGHT RESPONSE**

Inactivation and adaptation of light responses in both invertebrates and vertebrates systems depend on calcium, which also influences the shape and amplitude of quantal and macroscopic currents through positive and negative feedback regulation (see Figure 1.7)(Ranganathan, Harris et al. 1991; Hardie and Minke 1994; Ranganathan, Bacsikai et al. 1994; Scott, Sun et al. 1997; Scott and Zuker 1997; Henderson, Reuss et al. 2000;

**Figure 1.7 – Calcium Mediated Feedback Regulation in the Fly Photoreceptor Cell**

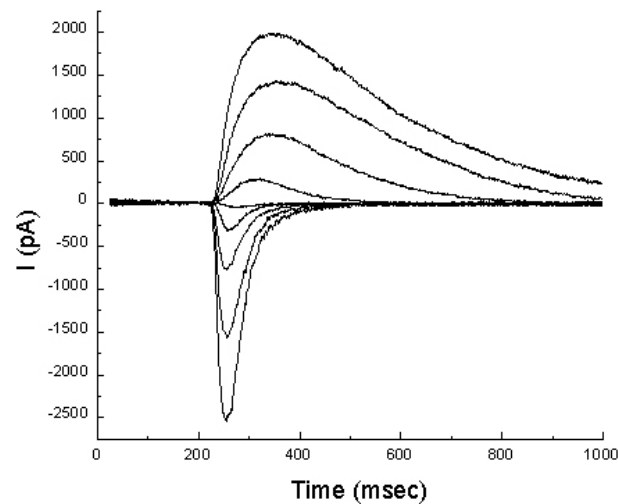


Figure 1.7: Calcium mediated feedback regulation in the fly photoreceptor cell. Pictured are inward (negative) and outward (positive) electrical currents elicited by stimulation with a single pulse of blue light. Each trace represents an different holding potential of the cell. Positive holding potentials elicit outward (positive) currents and negative holding potentials elicit inward (negative) currents. The resting potential of a fly photoreceptor cell is  $-40$  to  $-60$  mV. The increased inactivation and deactivation rates of the inward currents at negative holding potentials is due to calcium-mediated positive and negative feedback regulation that occurs inside the cell during activation of the phototransduction pathway.

Hardie and al. 2001). The levels of cytosolic calcium are thought to increase from submicromolar levels to hundreds of micromolar (Ranganathan, Bacskai et al. 1994; Postma, Oberwinkler et al. 1999; Oberwinkler and Stavenga 2000). The mechanism of positive feedback is not known, although it is possible that calcium may directly or indirectly alter the gating properties of TRP and TRPL. The role of calcium in negative feedback regulation is much better understood. Several calcium-sensitive proteins are known to be required for proper inactivation and adaptation of the light response, including ePKC, calmodulin (CaM), calcium/calmodulin-dependent kinase type II (CaMKII), RDGC (retinal degeneration C), and NINAC. InaD also plays a key role in calcium-mediated feedback regulation, since it binds TRP, ePKC, NINAC, and calmodulin.

### InaD Complex-Independent Calcium-Mediated Feedback Regulation

Some components of the negative feedback regulation involved in inactivation of the light response occur independent of the InaD complex. For example, *Drosophila* visual arrestin isoforms Arr1 and Arr2 bind to metarhodopsin and deactivate it by inhibiting interaction with Gq $\alpha$  (Dolph and al. 1993). Mutant flies lacking Arr2, the major isoform, show defects in light response inactivation (Dolph and al. 1993). Arr2 has been shown to be regulated by a calcium/calmodulin-dependent kinase, CaMKII. *Drosophila* CaMKII contains several CaM binding domains in its regulatory region and phosphorylates Arr2 at Ser366 in the C-terminus (Kiselev and al. 2000). The unphosphorylated tail of Arr2 interacts with clathrin, which mediates internalization of phosphorylated metarhodopsin, a mechanism implicated in light-dependent degeneration of the photoreceptor in *rdg* (retinal degeneration) mutants (Alloway, Howard et al. 2000; Kiselev and al. 2000).

Dephosphorylation of rhodopsin also seems to be required for inactivation of the light response. Fly *rdgC* (retinal degeneration C) mutants that lack a CaM-binding phosphatase that dephosphorylates the C terminus of metarhodopsin exhibit retinal degeneration and defects in inactivation (Vinos, Jalink et al. 1997; Lee and Montell 2001). In the absence of the RdgC phosphatase or in *rdgC* mutants that fail to bind CaM, the C-terminus of metarhodopsin remains phosphorylated and gradual retinal degeneration occurs via clathrin-mediated endocytosis (Alloway, Howard et al. 2000; Kiselev and al. 2000; Lee and Montell 2001). In wild-type photoreceptors apoptosis does not occur, as there is a balance between



calcium-mediated activity of CaMKII and RdgC, and possibly other kinases and phosphatases.

Since high levels of intracellular calcium can be toxic to a cell over an extended period, the photoreceptor must return the calcium back to baseline levels ( $\sim 100$  nM). During this period of calcium removal ( $\sim 100$  ms)(Oberwinkler and Stavenga 2000), the previously stimulated microvillus is temporarily refractory to further stimulation. Calcium exchangers in the microvillar membrane are the primary molecules involved in calcium removal. In *Drosophila* photoreceptors there are two calcium exchangers, called CalX and NCKX30C. CalX is a sodium/calcium exchanger that exchanges  $3 \text{ Na}^+ / 1 \text{ Ca}^{2+}$ , while NCKX30C is a potassium-dependent sodium/calcium exchanger that exchanges  $4 \text{ Na}^+ / 1 \text{ Ca}^{2+} / 1 \text{ K}^+$  (Haug-Collet and al. 1999). Although both exchangers are expressed in the photoreceptors of adult flies, the individual role of each type of exchanger in phototransduction is not yet clear. Some recent results, however, suggest that CalX localizes to TRP channels, and that *calX* mutants phenocopy *trp* mutants (Craig Montell, departmental lecture). In addition, overexpression of *calX* increases the amplitude of quantal responses and was shown to rescue mutants expressing a constitutively active TRP (*trp*<sup>P365</sup>) (Craig Montell, departmental lecture). These results suggest that phototransduction in *Drosophila* depends on a balance between calcium influx via TRP channels and calcium efflux via calcium exchangers CalX and NCKX30C.

### InaD Complex-Dependent Calcium-Mediated Feedback Regulation

Fly mutants lacking ePKC (*inaC<sup>209</sup>*), which is part of the InaD complex, exhibit a delay in inactivation and inability to adapt to constant light stimulus (Ranganathan, Harris et al. 1991; Smith and al. 1991; Hardie and al. 1993). Similarly, flies containing the *inaD<sup>215</sup>* mutation, which prevents the InaD from binding to TRP, have an almost identical phenotype (Shieh and Niemeyer 1995; Tsunoda and al. 1997). This suggests that localization of ePKC to TRP via the InaD complex is required for ePKC kinase activity. The kinase activity of ePKC is calcium dependent, and activated by the binding of calcium to its regulatory domain (Flybase)(Ron and Kazanietz 1999). In zero calcium, the light responses of *inaC<sup>209</sup>* mutants are indistinguishable from wild type due to the inactivity of wild-type ePKC in zero calcium (Ranganathan, Harris et al. 1991). PKC has been shown to phosphorylate several proteins in the InaD complex, including InaD and TRP (Huber, Sander et al. 1996; Liu, Parker et al. 2000), which is required for inactivation. The exact sites of phosphorylation of InaD and TRP are not known, but each protein contains several PKC phosphorylation motifs (Huber, Sander et al. 1996; Liu, Parker et al. 2000). PKC may also phosphorylate PLC- $\beta$  and modulate its activity, but currently there is no evidence to support that idea.

Besides involvement in arrestin-mediated deactivation of metarhodopsin, CaM may also facilitate inactivation by binding to CaM-binding domains found in members of the InaD complex, specifically InaD, TRP, TRPL, and NINAC. TRP and TRPL contain one and two putative CaM-binding domains respectively (Phillips, Bull et al. 1992; Niemeyer and al.

1996; Scott, Sun et al. 1997), and deletion of either CaM binding domains in TRPL causes a delay in inactivation of the macroscopic light response in *trp* mutants possibly due to an inability to inactivate the channels (Scott, Sun et al. 1997).

CaM plays a large role in negative feedback of the light response, and is maintained at a high concentration in the photoreceptor cell, about 0.5 mM (Porter, Yu et al. 1993). However, interaction with an unconventional myosin III known as NinaC (neither inactivation nor activation C) is required to maintain proper localization of CaM (Porter, Yu et al. 1993). The *ninaC* gene encodes two isoforms: p132, which is required for localization of CaM to the cell body and long-term adaptation via light-dependent trafficking of Arr2 to the rhabdomere, and p174, which is required for localization of CaM to the rhabdomere and inactivation of the light response (Porter, Yu et al. 1993; Wes, Xu et al. 1999).

### **MOTIVATIONS FOR THIS DISSERTATION**

Localization of the phototransduction components via binding to InaD as well as influx of high concentrations of calcium is known to be necessary for the proper activation, inactivation, and adaptation of the light response in *Drosophila*. The precise function of InaD in phototransduction is not known, however. Does InaD simply tether visual proteins together to increase their effective concentration or does InaD have a more complex role, such as assisting in the assembly of the complex and allosteric regulation of its binding partners?

The second chapter of this dissertation examines the “beads on a string” versus the “specific quaternary structure” models for InaD by analyzing the effect of shuffling the order of the PDZ domains in InaD. If the former model is correct, then shuffling the order of the PDZ domains should have little effect, assuming the mutant InaD binds all of its partners. We use biochemical assays to assess the expression levels of the mutant InaD protein as well as the ability to properly form the entire complex. We also utilize an electrophysiological assay to illustrate the effect of the mutant InaD on the macroscopic light response *in vivo* compared to wild-type flies as well as mutants that mislocalize different members of the INAD complex.

In the third chapter, we investigate whether one of the roles of InaD is to localize ePKC near its targets of phosphorylation, or if ePKC requires close proximity to the TRP channel for full kinase activity due to the presence of localized calcium transients. First, we conduct fluorescence imaging experiments of photoreceptors expressing fusions of several different visual proteins and Camgaroo, a  $\text{Ca}^{++}$ -sensitive derivative of yellow fluorescent protein (YFP), to determine whether discrete microdomains of calcium exist in the microvilli of the rhabdomere, or if the entire structure is simply flooded with calcium. If microdomains of calcium exist in the rhabdomere, Camgaroo fusions of proteins in the InaD complex will report higher levels of calcium compared to visual proteins that are not in the complex, e.g. Arr2. Finally, to discern whether the inactivation defect of *inaD*<sup>215</sup> (which cannot bind TRP) is due to displacement of ePKC from its target of phosphorylation (TRP) or a decrease in its kinase activity due to a lack of high calcium, we image photoreceptors expressing an InaD-Camgaroo construct in both wild type and *inaD*<sup>215</sup> backgrounds.

## **CHAPTER TWO**

### **Functional Importance of Domain Order in InaD-Mediated Scaffolding of Visual Transduction**

#### **ANALYSIS OF INAD PDZ DOMAIN REARRANGEMENT**

##### Introduction

As stated in the introduction, scaffolding proteins have been shown to be required for efficient signal transduction in various model systems, but what exactly do these proteins contribute to signaling? Two models have been proposed: 1) the “beads on a string” model where the binding domains of a scaffolding protein simply localize other proteins in general proximity to one another (Lim 2002), and 2) the “specific quaternary structure” model where precise orientation and interaction of the binding domains with their target proteins is required for proper signaling. The latter model is more complex and imposes great constraints on the design of the signaling system, but unlike the former model, it also allows for possible allosteric regulation of binding.

We investigated which of these two models applies to InaD, the scaffolding protein involved in *Drosophila* phototransduction, by rearranging the order of its PDZ domains and testing whether wild type light responses could be reconstituted. If the “beads on a string” model applies, then we expect that photoreceptors containing InaD with rearranged or “shuffled” PDZ domains would produce light response similar to wild type. Alternatively, if the “specific quaternary structure” model applies, then we expect that the light response

would be defective. We generated constructs of InaD PDZ domain rearrangements, and injected them into InaD deficient (*inaD<sup>l</sup>*) flies. We created homozygous lines of transgenic flies containing the shuffled InaD PDZ domains using traditional crossing methods, and verified expression of the mutant transgenes and stability of the complex. Finally, we characterized the macroscopic light responses of these transgenic flies.

### Generation of Flies Expressing InaD PDZ Domain Rearrangements

Traditional fly injection and crossing methods (see Chapter Five - Methodology) were used to create homozygous stocks of transgenic fly lines carrying one of four InaD PDZ domain rearrangements expressed behind the Rh1 promoter. The constructs were identified based on the order of their five PDZ domains: *p34512*, *p51234*, *p21354*, and *p45123*. As a positive control for rescue of InaD function in *inaD<sup>l</sup>* lines, transgenic wild type InaD (*p-inaD*) was also injected and crossed to create a homozygous stock. The light responses of these flies were analyzed previously by Jennifer Graves in our lab and were found to exhibit wild type characteristics (data not shown). Previous work done in another lab also showed rescue of *inaD<sup>l</sup>* using expression of wild type InaD driven by a heat-shock promoter (Shieh and Niemeyer 1995). Of the four constructs, only *p34512* and *p51234* were expressed at levels of InaD behind the Rh1 promoter similar to that in wild type cells (see Figure 2.1). Each of the lines of *p21354* and *p45123* showed little or no expression, which suggests that these constructs may not fold correctly.

**Figure 2.1: Western Blot Analysis of InaD PDZ Domain Rearrangements**

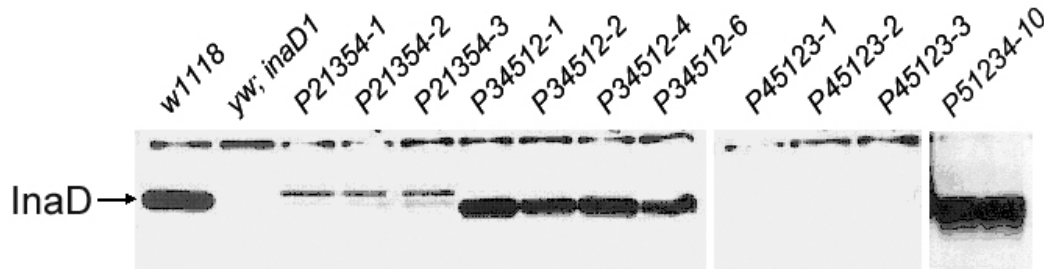


Figure 2.1: Western Blot Analysis of InaD Expression. All lines of *p34512* and *p51234* show nearly wild type (*w<sup>1118</sup>*) expression levels of InaD. The *inaD<sup>1</sup>* negative control and all lines of *p21354* and *p45123* show little or no expression.

### Stability of InaD Complex Over Time

Previous studies of fly visual proteins have shown a correlation between proper localization of InaD complex proteins (i.e. interaction with InaD) and their stability over time. In InaD null mutants or InaD mutants that cannot bind one or more of its corresponding proteins (*inaD<sup>215</sup>* and *inaD<sup>2</sup>*), western blot analysis has shown that the unbound protein or proteins are present at wild type levels in the photoreceptors upon eclosion, but that they are mislocalized and degrade over several days (Tsunoda and al. 1997). TRP has also been shown to randomly localize throughout the plasma membrane in *inaD<sup>215</sup>* flies (Tsunoda and al. 1997). We thus conducted western blot analysis of newly eclosed (virgin) and 10-day-old flies expressing *p34512* or *p51234* transgenic lines to test the *in vivo* stability of the InaD complex over time.

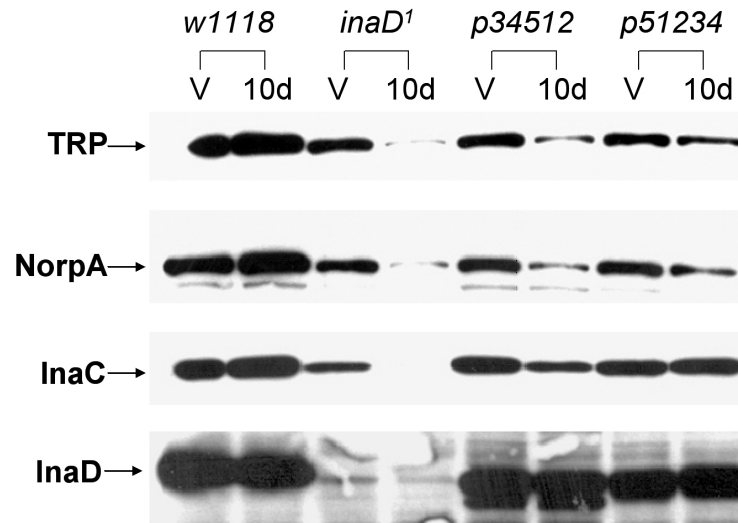
**Figure 2.2: Western Blot Analysis of InaD Complex Stability**

Figure 2.2: Western Blot Analysis of InaD complex stability at 0 (V) and 10 days (10d). *p34512* and *p51234* flies show the presence of InaC and InaD after 10 days similar to wild type (*w<sup>1118</sup>*). *p34512* flies show 50-60% decrease in levels of NorpA and TRP protein after 10 days. *p51234* shows only a slight decrease in levels of NorpA and TRP protein after 10 days. The *inaD<sup>1</sup>* negative control shows 90-100 % degradation of NorpA, InaC, and TRP after 10 days respectively.

In InaD deficient flies (*inaD<sup>1</sup>*), there is an almost complete decline in the amount of InaC, NorpA, and TRP proteins after 10 days. Similar to wild type flies, the level of InaC and NorpA remained the same after 10 days in *p51234* flies, while there was a 50-60 % decrease in NorpA in *p34512* flies (see Figures 2.2 and 2.3). The levels of TRP also remained about the same in wild type and *p51234* flies after 10 days, but there was a decrease of approximately 50-60% in the levels of TRP protein after 10 days in *p34512* flies.

These results suggest that the InaD complex is essentially intact in *p51234* flies, and at least partially intact in *p34512* flies. The decreasing levels of TRP in *p34512* flies may reflect a weaker binding affinity between *p34512*-InaD and TRP. Nevertheless, these results



motivate a further biochemical and also physiological analysis of these animals to understand how complex reorganization affects scaffolding function.

**Figure 2.3: Western Blot Analysis of NorpA Stability in *p34512* Flies**

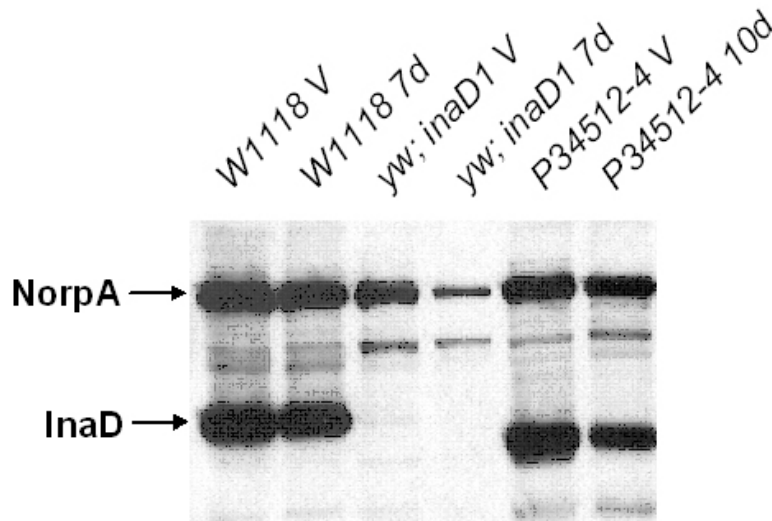


Figure 2.3: Western Blot Analysis of NorpA stability in *p34512* flies at 0 (V), 7 (7d), and 10 days (10d). *p34512* flies show a slight decrease in levels of NorpA compared to wild type (*w<sup>1118</sup>*) after 10 days. The *inaD<sup>1</sup>* negative control shows a significant loss of NorpA after 7 days and no InaD at either day 0 or 7 days.

### Co-immunoprecipitation of InaD Complex from Fly Heads

Coimmunoprecipitation experiments have also previously demonstrated interactions between InaD and its target proteins *in vitro* (Tsunoda and al. 1997). In addition, it was shown that NorpA and TRP cannot be coimmunoprecipitated with the rest of the complex in *inaD<sup>2</sup>* and *inaD<sup>215</sup>* flies respectively (Tsunoda and al. 1997). In recent work, we have developed conditions for efficient co-immunoprecipitation of wild type InaD complex.

Experiments are now underway to test whether p34512- and p51234-InaD proteins are also able to co-immunoprecipitate members of the InaD complex.

### Phenotype of Macroscopic Responses in *p34512* Photoreceptors

We investigated whether the p34512-InaD construct could 1) produce a light response, and 2) rescue the insensitivity to light of *inaD<sup>l</sup>* photoreceptors by comparing the light-induced currents (LIC) of *p34512* photoreceptors versus wild type (*yw*) and *inaD<sup>l</sup>* at 1.5 mM CaCl<sub>2</sub> and 0 CaCl<sub>2</sub>. The macroscopic light responses of *yw*, *inaD<sup>l</sup>*, and the *p34512* photoreceptors were analyzed with patch clamp analysis as described in Chapter Five – Methodology. Flies were dark-adapted overnight before dissection and age-matched to the eclosion stage. The mutant *p34512* displayed a slight delay in activation and a large delay in deactivation kinetics compared to wild type at 1.5 mM calcium (see Figure 2.4A). The delay in activation and deactivation was not as drastic compared to that of *inaD<sup>l</sup>*. However, at zero calcium, the activation and deactivation of wild type and *p34512* light responses are virtually indistinguishable as shown in Figure 2.4B. These results indicate a calcium-specific deactivation defect in *p34512* photoreceptors.

A comparison of the light response parameters better reveals the quantitative differences between the macroscopic light responses of wild type, *p34512*, and *inaD<sup>l</sup>* flies. Latency and time to peak are parameters that describe the activation phase of the light response. Latency is the time to the increase in inward current after the light stimulus is given, while time to peak is the time to the maximum amplitude of the response. There is

**Figure 2.4: Comparison of wild type (*yw*), *p34512*, and *inaD<sup>l</sup>* Light Responses**

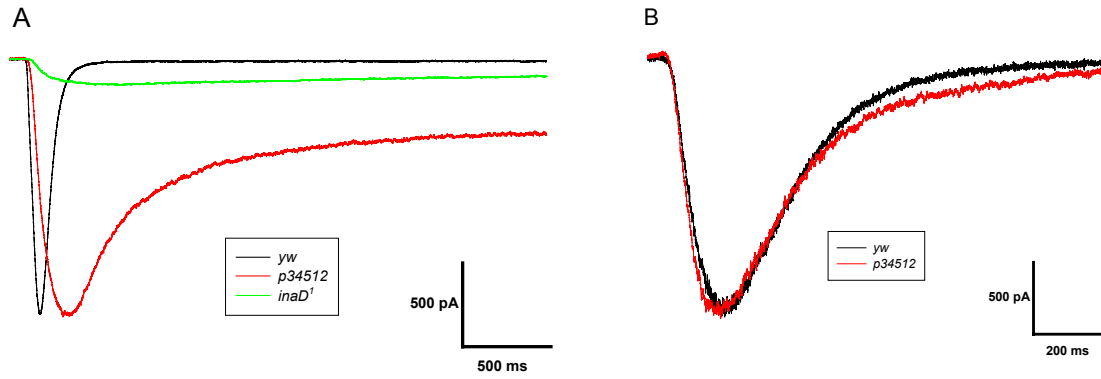


Figure 2.4: Comparison of wild type (*yw*, black), *p34512* (red), and *yw; inaD<sup>l</sup>* (green) phenotypes at 1.5 mM and 0 mM CaCl<sub>2</sub>. A) Overlay of normalized and averaged macroscopic responses at similar light intensities of eclosion stage photoreceptor cells at 1.5 mM CaCl<sub>2</sub> (*yw* N = 5, *p34512* N = 4). Single response at similar light intensity from *inaD<sup>l</sup>* (green) photoreceptor shown for comparison of difference in amplitude of light responses. B) Overlay of normalized and averaged macroscopic responses of eclosion stage photoreceptor cells at 0 mM CaCl<sub>2</sub> (*yw* N = 5, *p34512* N = 4). Light stimulus occurs at 25 ms and the holding potential for all measurements is -60 mV.

very little difference between the average latency of wild type and *p34512* light responses at both high and low calcium concentrations as shown in Figure 2.5A and Table 2.5, but there is some difference between the latency of wild type and *inaD<sup>l</sup>* light responses. The latter finding is consistent with previous reports (Shieh and Niemeyer 1995; Tsunoda and al. 1997). This difference in latency is probably underestimated, however, since latency is inversely related to stimulus intensity, and *inaD<sup>l</sup>* mutants are much less sensitive than wild type photoreceptors (see Figure 2.6). A comparison of the time to peak of wild type and *p34512* responses at 1.5 mM CaCl<sub>2</sub> shows some difference ( $p < 0.01$ ), but the difference between wild type is not as significant as that between wild type and *inaD<sup>l</sup>* photoreceptors ( $p$

< 0.001) as shown in Table 2.5. These results indicate that there is substantial rescue of activation in *p34512* versus *inaD<sup>1</sup>* photoreceptors.

**Figure 2.5: Comparison of wild type (*yw*), *p34512*, and *yw; inaD<sup>1</sup>***  
**Light Response Parameters**

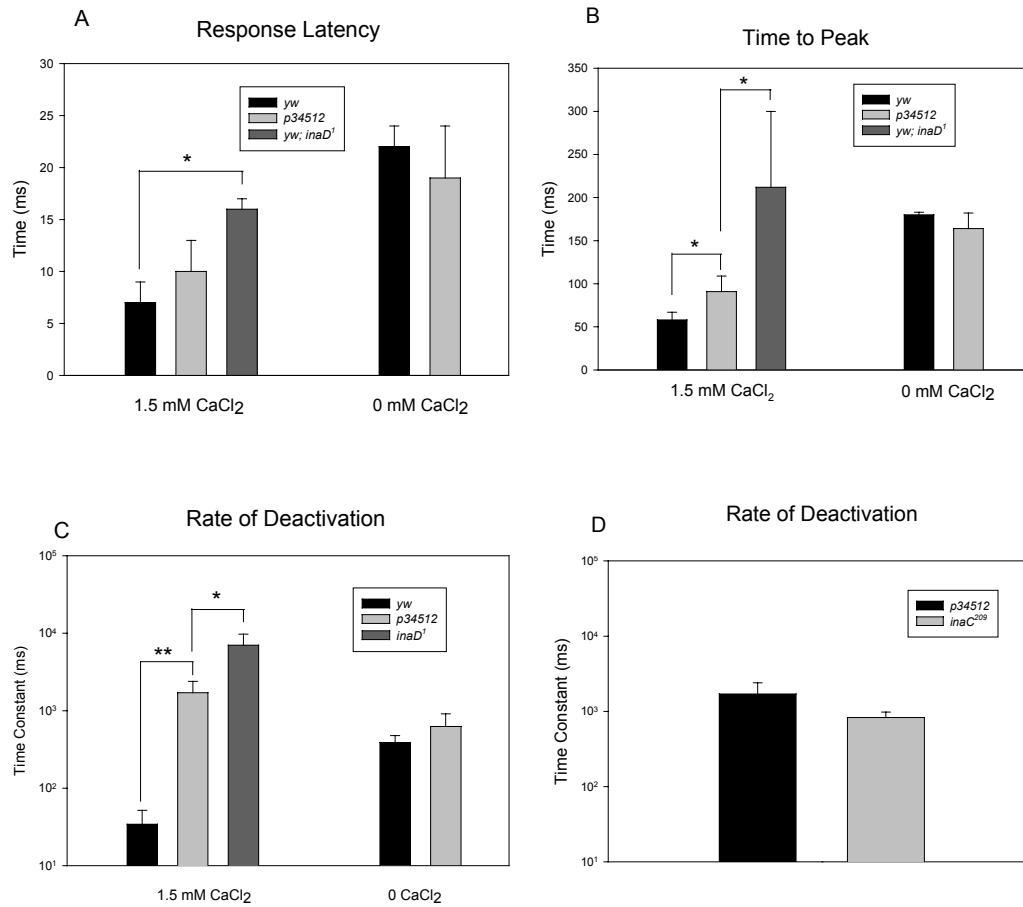


Figure 2.5: (A-C) Comparison of wild type (*yw*, black), *p34512* (light gray), and *yw; inaD<sup>1</sup>* (dark gray) light response parameters at 1.5 mM and 0 mM CaCl<sub>2</sub>. A) Average response latency at 1.5 mM CaCl<sub>2</sub> and 0 mM CaCl<sub>2</sub>. B) Average time to maximum peak at 1.5 mM CaCl<sub>2</sub>, and 0 mM CaCl<sub>2</sub>. C) Rate of deactivation as measured by the time constant,  $\tau$  (ms), was calculated by fitting a single exponential equation to the tail (final 20%) of the deactivation phase of light. D) Comparison of rate of deactivation of *p34512* (1.5 mM CaCl<sub>2</sub>, black) vs. *inaC<sup>209</sup>* (500  $\mu$ M CaCl<sub>2</sub>, gray) light responses. Student's non-paired t-test was performed on each pair of genotypes to determine statistical significance of difference in mean values. \*  $p < 0.01$ , \*\*  $p < 0.001$ .

**Table 2.5: Comparison of wild type (*yw*), *p34512*, and *yw; inaD<sup>l</sup>*****Light Response Parameters**

	<i>yw</i> 1.5 mM CaCl <sub>2</sub>	<i>p34512</i> 1.5 mM CaCl <sub>2</sub>	<i>inaD<sup>l</sup></i> 1.5 mM CaCl <sub>2</sub>	<i>inaC<sup>209</sup></i> 500 $\mu$ M CaCl <sub>2</sub>	<i>yw</i> 0 CaCl <sub>2</sub>	<i>p34512</i> 0 CaCl <sub>2</sub>
Latency	7 $\pm$ 2	10 $\pm$ 3	16 $\pm$ 1*	N/A	22 $\pm$ 2	19 $\pm$ 5
Time to peak	58 $\pm$ 9	91 $\pm$ 18*	212 $\pm$ 77**	N/A	180 $\pm$ 3	164 $\pm$ 18
Time constant ( $\tau$ )	34 $\pm$ 18	1700 $\pm$ 690**	7000 $\pm$ 2700**	828 $\pm$ 152	390 $\pm$ 90	630 $\pm$ 280

Figure 2.5: Comparison of wild type (N = 5), *p34512* (N = 4), and *yw; inaD<sup>l</sup>* (N = 4) light response parameters at 1.5 mM and 0 mM CaCl<sub>2</sub> and *inaC<sup>209</sup>* at 500  $\mu$ M CaCl<sub>2</sub> (N = 3). Rate of deactivation as measured by the time constant,  $\tau$  (ms), was calculated by fitting a single exponential equation to the tail (final 20%) of the deactivation phase of light. Student's non-paired t-test was performed on each pair of genotypes to determine statistical significance of difference in mean values. \*  $p < 0.01$ , \*\*  $p < 0.001$ .

We next investigated whether the *p34512*-InaD construct could rescue the insensitivity to light of *inaD<sup>l</sup>* photoreceptors by comparing the light sensitivity of *p34512* photoreceptors versus wild type (*yw*) and *inaD<sup>l</sup>* flies at 1.5 mM CaCl<sub>2</sub>. We recorded the LIC at increasing intensities of 492 nm light, and then graphed the normalized peak current values versus the log of light intensity for each phenotype as shown in Figure 2.6. The light sensitivity of *p34512* photoreceptors was similar to wild type, while the sensitivity of *inaD<sup>l</sup>* photoreceptors was one to two log orders lower due to the absence of InaD and the mislocalization of its component proteins.

These results show that sensitivity is rescued in *p34512* mutants, and suggests that the activity of NorpA, the primary mediator of response sensitivity is not hindered by the rearrangement of InaD. In contrast, *inaD<sup>l</sup>* as well as *inaD<sup>2</sup>* mutants that cannot bind NorpA exhibit marked insensitivity to light compared wild type (Shieh, Zhu et al. 1997).

**Figure 2.6: Light Intensity Response Curves of wild type (*yw*), *p34512*, and *yw; inaD<sup>l</sup>* Photoreceptors**

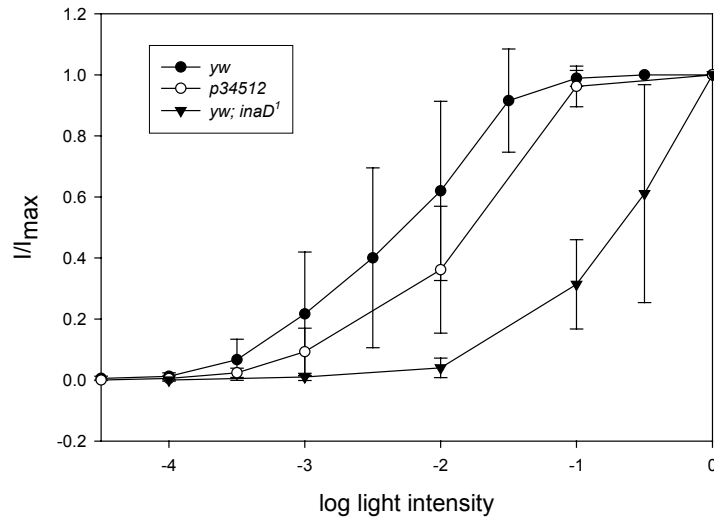


Figure 2.6: Light intensity response curves of wild type (*yw*) (closed circles), *p34512* (open circles), and *yw; inaD<sup>l</sup>* (closed triangles) photoreceptors at 1.5 mM CaCl<sub>2</sub>. Cells were dark-adapted overnight and then flashed with increasing intensity of a 3 ns flash of 492 nm light. Amplitudes of light-induced current were normalized to the maximal response and values were averaged (*yw* N = 5, *p34512* N = 3, *inaD<sup>l</sup>* N = 4). The holding potential for all experiments is -60V.

The average deactivation time constant ( $\tau$ ) is a measure of the rate of deactivation of the light response. It is calculated by fitting a single exponential equation to the tail end (last 20%) of the deactivation phase of the light response (Ranganathan and Stevens 1995). The average deactivation  $\tau$  of *p34512* and *inaD<sup>l</sup>* responses at 1.5 mM CaCl<sub>2</sub> is significantly higher than that of wild type at 1.5 mM CaCl<sub>2</sub> ( $p < 0.001$ ), and confirms that the deactivation kinetics of *p34512* are much more similar to *inaD<sup>l</sup>* than wild type (see Figure 2.5C and Table 2.5). However, there was some difference between the  $\tau$  of *p34512* and *inaD<sup>l</sup>* at 1.5 mM CaCl<sub>2</sub>, which indicates that the delay in deactivation is not as pronounced in the former versus the latter ( $p < 0.01$ ) as shown in Table 2.5. As with the latency and time to peak, there

was no difference between the  $\tau$  of wild type and *p34512* light responses at zero calcium. Interestingly, comparison of the  $\tau$  of *p34512* and *inaC*<sup>209</sup> at high calcium, even though the extracellular calcium concentration is lower in the latter, showed no significant difference and implies that the deactivation defect seen in *p34512* may involve the InaC kinase (see Figure 2.5D and Table 2.5). These results indicated that in *p34512* photoreceptors there is nearly complete loss of calcium-dependent negative feedback regulation, which is primarily mediated by the InaC kinase. The finding that this phenotype is absent in zero calcium argues that all of the *p34512* phenotype is downstream of calcium influx and therefore may be due to improper function of the InaC kinase.

Based on the above results, we concluded that *p34512* photoreceptors show almost normal activation, and thus calcium-mediated positive feedback regulation is largely unaffected. However, *p34512* photoreceptors seem unable to support proper calcium-mediated negative feedback regulation, the primary mechanism of deactivation of the light response. This may be due to a reduced affinity of p34512-InaD for one or more of its partners or improper stereochemical orientation of proteins such as InaC with its targets of phosphorylation. As shown above in Figure 2.2, all components of the InaD complex-- InaC, NorpA, and TRP-- are present at wild type levels in photoreceptors at eclosion, but their interaction with InaD may be weakened in *p34512* flies.

To better resolve this issue, we next studied *p51234* using patch clamp analysis. The p51234-InaD complex shows complete stability in aged animals, and thus any phenotype is unlikely to be a result of weakened protein-protein interactions.

### Phenotype of Macroscopic Responses in *p51234* Photoreceptors

The macroscopic light responses of *yw*, and the *p51234* InaD mutant were analyzed with patch clamp analysis in collaboration with Prashant Mishra, a graduate student in the lab, as described in Chapter Five – Methodology. Flies were dark-adapted overnight before dissection and age-matched to the eclosion stage. Electrophysiological analysis showed that *p51234* light response, unlike those of *p34512*, displayed essentially normal activation, a large delay in deactivation kinetics compared to wild type at 1 mM calcium as shown in Figure 2.7A. Preliminary recordings show that *p51234* responses are similar to wild type at zero calcium as shown in Figure 2.7B.

**Figure 2.7: Comparison of wild type (*yw*), and *p51234* Light Responses**

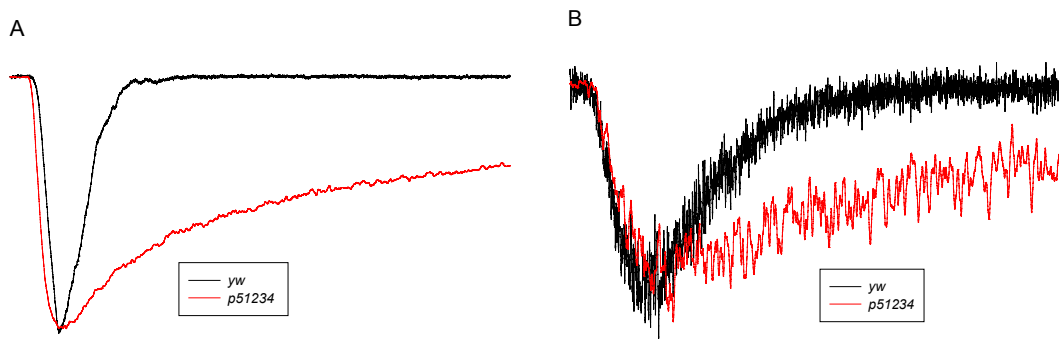


Figure 2.7: Comparison of wild type (*yw*, black), and *p51234* (red) electrical responses at A) 1 mM CaCl<sub>2</sub>. Overlay of normalized and averaged macroscopic responses of virgin stage photoreceptor cells. B) Single low light responses of *yw* (black) and *p51234* (red) at 0 CaCl<sub>2</sub>. Light stimulus occurs at 25 ms and holding potential for all measurements = -40 mV.



**Figure 2.8: Comparison of Wild Type (*yw*), and *p51234* Light Response Parameters**

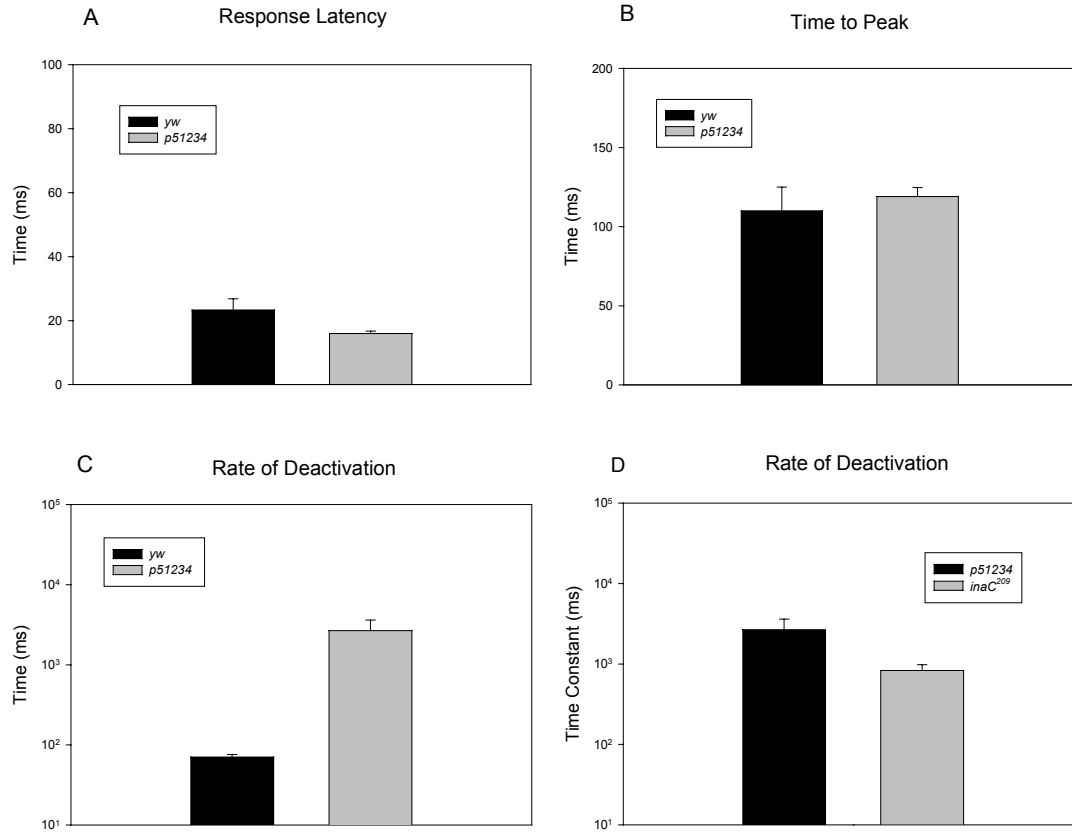


Figure 2.8: (A-C) Comparison of wild type (*yw*, N = 3), and *p51234* (N = 2) light response parameters at 1 mM and -40 mV holding potential. A) Average response latency at 1 mM  $\text{CaCl}_2$ . B) Average time to maximum peak at 1 mM  $\text{CaCl}_2$ . C) Rate of deactivation as measured by the time constant,  $\tau$  (ms), was calculated by fitting an exponential equation to the tail (final 20%) of the deactivation phase of light responses. D) Comparison of  $\tau$  of *p51234* (black) vs. *inaC<sup>209</sup>* (gray).

A comparison of the light response parameters show little difference between the latency and time to peak of wild type versus *p51234* flies as shown in Figure 2.8A, B (see also table inset). These results indicate that activation, and hence calcium-mediated feedback regulation is not affected by the rearrangement of InaD PDZ domains. However, the deactivation time constant ( $\tau$ ) is increased in *p51234* mutants by nearly two log orders as

shown in Figure 2.8C and Table 2.8. We infer from these results that as in *p34512* mutants, the calcium-mediated negative feedback (deactivation) is adversely affected by rearrangement of the PDZ domains of InaD. Comparison of  $\tau$  in *p51234* versus *inaC<sup>209</sup>* flies, as with *p34512* flies, shows no significant difference and suggests that the defect in *p51234* is linked with *inaC* kinase function. The finding that *p51234* largely recapitulates the phenotype of *p34512* mutants strongly supports our contention that precise stereochemical orientation of the PDZ domains of InaD is essential for calcium-dependent negative feedback regulation (deactivation), possibly due to the InaC kinase. Calcium-dependent positive feedback regulation (activation), however, appears not to depend on precise stereochemistry.

**Table 2.8: Comparison of Wild Type (*yw*) and *p51234* Light Response Parameters**

	<i>yw</i> (1 mM CaCl <sub>2</sub> )	<i>p51234</i> (1 mM CaCl <sub>2</sub> )	<i>inaC<sup>209</sup></i> (500 $\mu$ M CaCl <sub>2</sub> )
Latency	23 $\pm$ 4	16 $\pm$ 1	N/A
Time to Peak	110 $\pm$ 15	119 $\pm$ 6	N/A
Time constant ( $\tau$ )	71 $\pm$ 5	2700 $\pm$ 940	828 $\pm$ 152

---

Table 2.8: Comparison of wild type (*yw*, N = 3) and *p51234* (N = 2) light response parameters at 1 mM, and *inaC<sup>209</sup>* (N = 3) time constant ( $\tau$ ) at 500  $\mu$ M. Holding potential = -40 mV and -60 mV respectively.

## CONCLUSIONS

We have shown in the above experiments that two constructs of InaD in which the PDZ domains have been rearranged (p34512 and p51234) can be expressed in fly photoreceptors. Additionally, by western blot analysis we have shown that the NorpA, InaC, and TRP proteins remain stable after 10 days in *p51234* flies, suggesting that p51234-InaD retains its ability to bind these proteins despite the rearrangement of PDZ domains. Flies expressing p34512-InaD retain the ability to stabilize InaC and NorpA, but there is some loss of TRP after 10 days. These results suggest that perhaps p34512-InaD is unable to interact properly with TRP mislocalizing the InaD complex and resulting in a phenotype similar to *inaD*<sup>215</sup> or *inaC*.

Using patch clamp analysis, we have shown some aspects of the phenotype of *inaD*<sup>l</sup>, prolonged response latency, delayed activation, and insensitivity of the light response, is substantially rescued in both *p34512* and *p51234* photoreceptors. In contrast, inactivation of the light response is defective in both phenotypes. Thus normal activation kinetics as measured by latency and time to peak is largely rescued with both shuffled constructs, while deactivation as measured by the time constant of deactivation is similar to that of *inaD*<sup>l</sup>. In addition, by generating a light response curve we have shown photosensitivity is rescued in *inaD*<sup>l</sup> flies expressing p34512-InaD.

The rescue of activation and photosensitivity by expression of p34512-InaD and p51234-InaD, supports the “beads on a string” model for scaffolding proteins. That is, simple localization of InaD complex proteins to the rhabdomere seems necessary and

sufficient for wild type sensitivity and activation, but precise orientation of proteins in the InaD complex is not required.

Conversely, the inability to rescue deactivation of the light response in *inaD<sup>l</sup>* mutants with rearranged InaD constructs tends to support the “specific quaternary structure model” of scaffolding proteins. This is especially apparent in *p51234* flies, where the time constant of deactivation is delayed by almost 2 log orders despite the fact that all proteins in the InaD complex are shown to be stable after 10 days, and thus assumed to be localized to the rhabdomere. From these results we conclude that precise ordering of the PDZ domains of InaD is required for proper deactivation of the light response, and therefore we propose that the quaternary structure of InaD has been optimized for maximum signaling efficiency between the components involved in negative feedback regulation of the light response, such as the InaC kinase and TRP channels.

Future experiments should involve first quantifying the interactions of the shuffled InaD constructs with their target proteins, and pinpointing which components of negative feedback regulation (e.g. TRP, InaC, and calmodulin) are responsible for the deactivation defect observed in *p34512* and *p51234* mutants.

## CHAPTER THREE

### Calcium Imaging of *Drosophila* Photoreceptors

#### INTRODUCTION

##### Calcium microdomains

The vital role that calcium plays as a second messenger in signaling in many different cell types has sparked an interest in monitoring and understanding the complex quantitative, spatial, and temporal components of calcium signaling over the past two decades. Calcium imaging studies have shown that localized calcium transients or “microdomains” exist in excitable cells such neurons, and play a key role in memory and synaptic plasticity (Carafoli 2002; Goldberg, Tamas et al. 2003).

Because of the importance of calcium in the negative and positive feedback regulation of the light response in fly photoreceptors, calcium imaging has been important tool for understanding the temporal and spatial properties of calcium signaling in the fly eye. Several prior studies using calcium imaging have shown that transients of calcium occur in the rhabdomere and that these transients can reach up to hundreds of micromolar of calcium during the visual response (Peretz 1994; Ranganathan, Bacsikai et al. 1994; Postma, Oberwinkler et al. 1999; Oberwinkler and Stavenga 2000). Additionally, because the negative feedback regulation of the light response relies on high levels of extracellular calcium, there has been much interest in studying calcium signaling in flies with mutations in calcium-regulated visual proteins. For instance, calcium imaging experiments have shown

that in flies lacking InaC or eye-PKC (*inaC<sup>209</sup>*), there is a marked delay in the return to basal calcium levels compared to wild type (Ranganathan, Bacsikai et al. 1994).

The phenotype of the eye-specific PKC InaC null flies (*inaC<sup>209</sup>*) is remarkably similar to *inaD<sup>215</sup>* flies, which contain a mutation in the 3<sup>rd</sup> PDZ domain of InaD and are unable to bind TRP channels (Shieh and Niemeyer 1995). Both mutants display an increased delay in deactivation in both macroscopic and quantal responses, a phenotype that disappears in the absence of calcium (Henderson, Reuss et al. 2000). Because the phenotype of both mutants is epistatic to calcium influx, these flies appear to have defects in the calcium-regulated negative feedback mechanism of the light response. InaC, which requires high levels of calcium for its kinase activity, is known to be one of the primary mediators of negative feedback regulation of the light response. Therefore, it is also thought that the *inaD<sup>215</sup>* is due to a defect in InaC activity.

We propose in Chapter Two of this thesis that one of the primary functions of InaD is to colocalize the calcium-mediated negative feedback machinery into a precise quaternary structure to maximize the speed and efficiency of deactivation. The *inaC*-like phenotype of *inaD<sup>215</sup>* could be due to either 1) the inability of InaC to phosphorylate unassociated TRP (which is thought to be a primary target of deactivation), or 2) a decrease in the general kinase activity of InaC caused by its mislocalization away from the main source of calcium, TRP (Scott and Zuker 1997).

The mechanistic distinction between these models is significant. Both imply a primary role for InaD in organizing a microdomain of calcium signaling, but differ in their organizing principle. The first model requires a precise formation of a complex that properly

juxtaposes the TRP channel with the InaC kinase; no spatial gradients of calcium in the microvilli are necessary. The second model relies on the fundamental existence of local calcium gradients in the microvillus such that the InaC kinase only becomes active in the immediate vicinity of the TRP channel. The latter model has been proposed by several investigators, and is primarily driven by the idea that InaD acts simply to colocalize, rather than precisely order the signaling machinery (Scott and Zuker 1997; Montell 1999).

In Chapter Two of this dissertation, we have directly addressed this issue by establishing that the precise stereochemical orientation of the PDZ domains of InaD is required for calcium-mediated feedback regulation of the light response. In this chapter, we test the second model for the *inaD*<sup>215</sup> by directly imaging the rhabdomeres of fly photoreceptors to determine if discrete microdomains exist in the rhabdomere during phototransduction.

The first question to ask is if microdomains of calcium are even possible in the rhabdomere. Calcium microdomains as small as one micrometer have been resolved in the dendritic spines of mammalian interneurons using fluorescence imaging, which are about 0.4  $\mu\text{m}$  in diameter and tens of microns in length (Goldberg, Tamas et al. 2003). However, the dimensions of fly rhabdomeric microvilli are much smaller—only 50 nm in diameter and 1.5  $\mu\text{m}$  in length (Montell 1999; Hardie and Raghu 2001). Nevertheless, Stavenga and Oberwinkler have shown by modeling and imaging studies that a proper distribution of TRP channels and calcium exchangers make the existence of discrete microdomains localized near TRP channels a possibility (Oberwinkler and Stavenga 2000).

## Calcium Imaging of Fly Photoreceptors

Injectable or cell permeable calcium-sensitive fluorescent indicators have traditionally been used to study calcium signaling in neurons and photoreceptors, but in the past decade calcium-sensitive derivatives of fluorescent proteins from jellyfish have been developed for calcium imaging *in vivo*. These calcium “sensors” have several advantages over traditional fluorescent dyes in that they are genetically encodable, and can be specifically targeted to particular regions of a cell either with a cell localization tag (e.g. ER retention sequence) or by fusion to a particular protein. In addition, the temporal and spatial distribution can be controlled by the use of different types of promoters to drive expression. Localization of the calcium indicator is a particularly attractive trait in the imaging of fly photoreceptors because the small dimensions of the microvilli of the rhabdomere restrict diffusible dyes to the cell body. Also, the spatial differentiation of microdomains using diffusible dyes is limited by the resolution of the fluorescence microscope and the diffusibility of the dye.

As part of our investigation of the mechanisms of InaD function, we set out to determine whether we could resolve microdomains of calcium within the rhabdomeres of flies by using a calcium indicator that would localize specifically to the rhabdomere. Our experimental approach was to first show that a GFP-based calcium indicator can be targeted to the rhabdomeres. This led to the construction and study of untagged indicator and fusions of the indicator with Arr2, an abundant rhabdomeric protein. Next, we constructed N- and C-terminal fusions of the indicator with InaD; this should report levels of calcium in the



vicinity of the signaling complex. Finally, we constructed N- and C- terminal fusions of the indicator with InaD<sup>215</sup>, which is unable to localize TRP channels to the InaD complex.

Initially, we thought that we might be able to resolve calcium microdomains by tagging a protein in the InaD complex (InaD) and one that is not in the InaD complex but is still found in the rhabdomere (Arr2). We could then compare the level of total fluorescence normalized for protein expression, and determine if there was a notable difference in the amount of calcium reported by each indicator. After establishing that we could image fly rhabdomeres using Camgaroo-tagged visual proteins, our secondary goal was to test the hypothesis that the phenotype of *inaD*<sup>215</sup> is caused by the mislocalization of InaC away from high microdomains of calcium required for full kinase activity by comparing the calcium reported by flies expressing Camgaroo-tagged InaD and InaD<sup>215</sup> proteins.

In this chapter, we show that we have successfully tagged Arr2, InaD, InaD<sup>215</sup>, and dgq with a GFP-based calcium indicator, and measured fluorescence using these proteins. In addition, by fluorescence imaging of photoreceptors expressing Camgaroo-tagged InaD and InaD<sup>215</sup> we give evidence that no discrete calcium microdomains exist within the rhabdomeric microvilli. This suggests that the phenotype of *inaD*<sup>215</sup> is likely not due to a reduction in calcium-mediated InaC kinase activity, but instead is due to a loss of direct interaction between the InaC kinase and its target, the TRP channel.

## **GENERATION OF FLIES EXPRESSING CAMGAROO-TAGGED**

### **VISUAL PROTEINS**

In order to conduct calcium imaging of fly rhabdomeres, we tagged several visual proteins that localize specifically to the rhabdomere with a calcium-sensitive fluorescent protein known as Camgaroo that we obtained from the laboratory of Dr. Roger Tsien. Camgaroo is so named because it consists of the mutant *Aequorea* enhanced yellow fluorescent protein (EYFP), and has a *Xenopus* calmodulin domain inserted in its “pouch” at residue Y145 (Baird, Zacharias et al. 1999). The fluorescence of Camgaroo increases by up to 7-fold *in vitro* in the presence of calcium, and it senses calcium with a dynamic range from 1-100  $\mu\text{M}$  ( $K_d = 7 \mu\text{M}$ , see Figure 3.6), which is appropriate for the expected fluctuations in rhabdomeric calcium (Baird, Zacharias et al. 1999).

We generated Camgaroo-tagged constructs for expression in fly photoreceptor cells via the fly transformation vector YC4 driven by the Rh1 (rhodopsin 1) promoter as outlined in Chapter Five – Methodology. Traditional fly injection and crossing methods (see Chapter Five - Methodology) were used to create homozygous stocks of transgenic fly lines carrying one of several Camgaroo fusions. The constructs were identified based on the fly visual protein component, and the orientation of the Camgaroo component, e.g. Camgaroo-Arrestin2 (Camga-Arr2 or CA) denotes a fusion protein consisting of Arrestin2 with Camgaroo tagged to its N-terminus. We generated flies expressing Camgaroo fusions of Arr2, InaD, dgq $\alpha$ , and InaD<sup>215</sup>. We also generated flies expressing Camgaroo alone (cyto-Camga or CC) as a control for imaging of the cell body of the rhabdomere. All flies were

injected into a *yw* (wild type except for yellow body color and white eyes) background except for the *InaD*<sup>215</sup>-*camga* construct, which was injected into *yw; inaD*<sup>215</sup> flies.

As a control for levels of calcium in the dark state, we crossed all of the lines except *dgg-camga* with *norpA*<sup>P41</sup> flies that lack the phospholipase C NorpA and are functionally blind. We also crossed the *camga-arr2* and *inaD-camga* lines with *inaC*<sup>209</sup> flies in order to study the effects of the loss of InaC on our calcium measurements.

#### Quantitative Western Blot Analysis of Camgaroo Fly Lines

To confirm expression of the Camgaroo fusions in the transgenic fly lines obtained from injections, we conducted western blot analysis of disrupted fly heads for each line. Only those lines that expressed well were kept (see Figure 3.1). The amount of Camgaroo fusion protein expressed in the R1-6 photoreceptor cells was estimated using quantitative western blot analysis (see Figure 3.2).

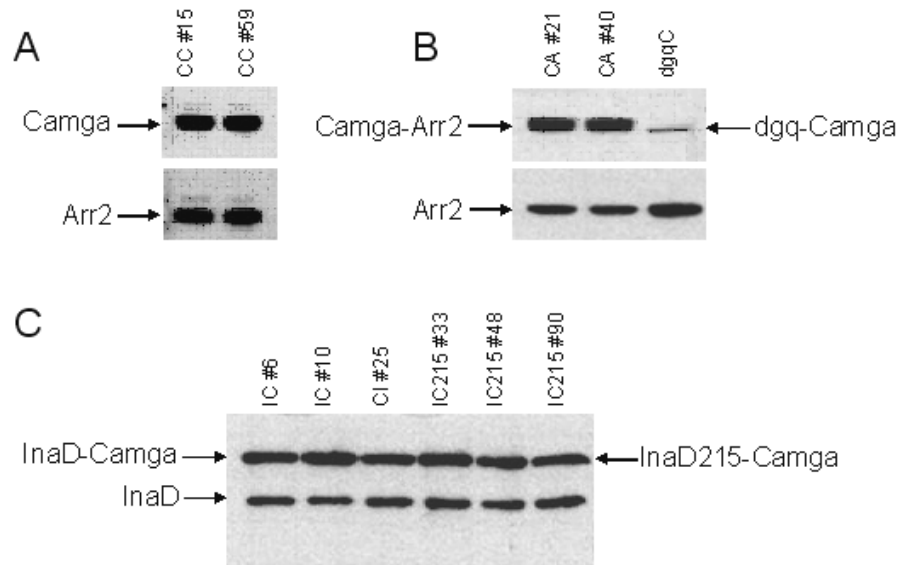
**Figure 3.1 – Western Blot Analysis of Camgaroo Expression**

Figure 3.1: Western blot analysis of expression of Camgaroo fusion proteins in various fly lines. A) Analysis of fly lines expressing cytoplasmic Camgaroo (CC) probed with GFP antibody (Clontech) diluted 1:100 (top), and Arrestin2 antibody diluted 1:2000 as a control for lane loading (bottom). B) Analysis of fly lines expressing Camgaroo-Arrestin2 (CA) and dgq-Camgaroo (dgqC) probed with GFP antibody (Clontech) diluted 1:100 (top), and Arrestin2 antibody diluted 1:2000 (bottom). C) Analysis of fly lines expressing InaD-Camgaroo (IC), Camgaroo-InaD (CI), and InaD215-Camgaroo (IC215).

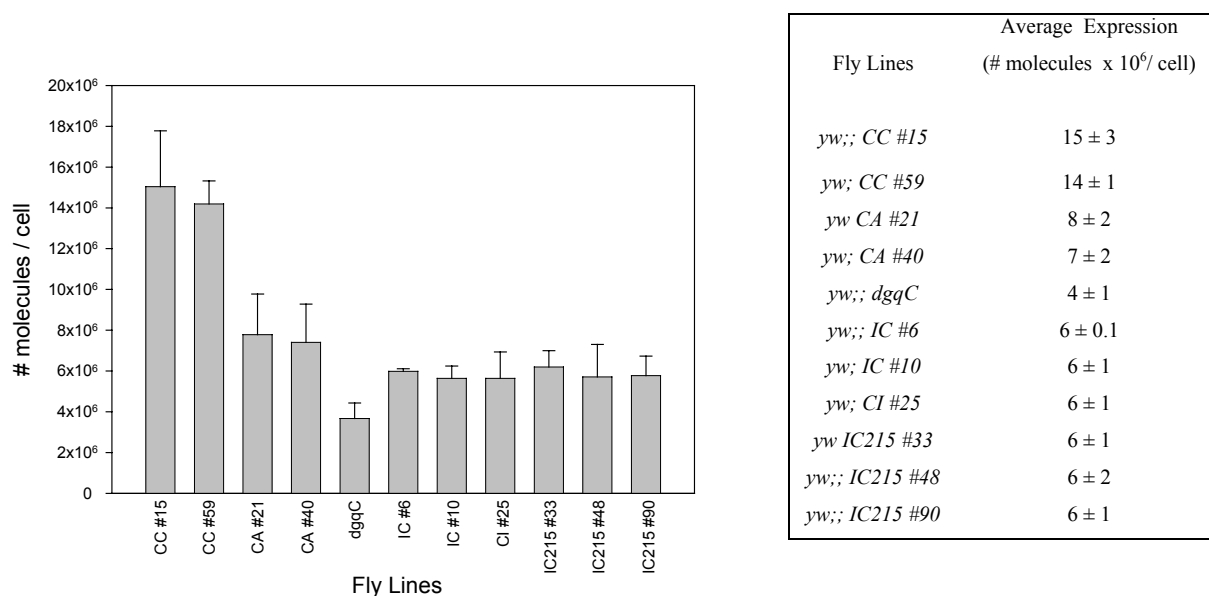
**Figure 3.2: Quantitative Western Blot Analysis of Camgaroo Expression**

Figure 3.2: Quantitative western blot analysis of mean expression of Camgaroo Fusion Proteins (N = 3). Four fly heads per line were sonicated in SDS head buffer and run on 10 % SDS-PAGE. Standards consisted of 1, 2, and 3 ng of purified Camgaroo protein. Gels were transferred to nitrocellulose and probed with GFP antibody (1:100 dilution, Clontech). Blot films were scanned, the intensity of bands was measured in Photoshop (Adobe, and a curve generated using the Camgaroo standards. CC = cytoplasmic Camgaroo, CA = Camgaroo-Arr2, dgqC = dgqα-Camgaroo, IC = InaD-Camgaroo, CI = Camga-InaD, IC215 = InaD-Camgaroo.

### Immunofluorescence Analysis of Camgaroo Flies

To confirm that the Camgaroo fusions were localized to the rhabdomeres of photoreceptor cells, we conducted immunofluorescence analysis of Camga-Arr2 and cyto-Camga fly eyes as outlined in Chapter Five – Methodology. The results showed that cyto-Camga localizes to the cytoplasm of the photoreceptors (see Figure 3.3A), while *yw* control flies showed no Camgaroo expression as expected (see Figure 3.3B). Similar to wild type Arr2, Camga-Arr2 exhibited light dependent movement from the cell body to the rhabdomere

(see Figure 3.3C and D)(Kiselev and al. 2000). These results confirm our ability to target Camgaroo to the rhabdomere.

**Figure 3.3 – Immunofluorescence Analysis of Camgaroo Fly Eyes**

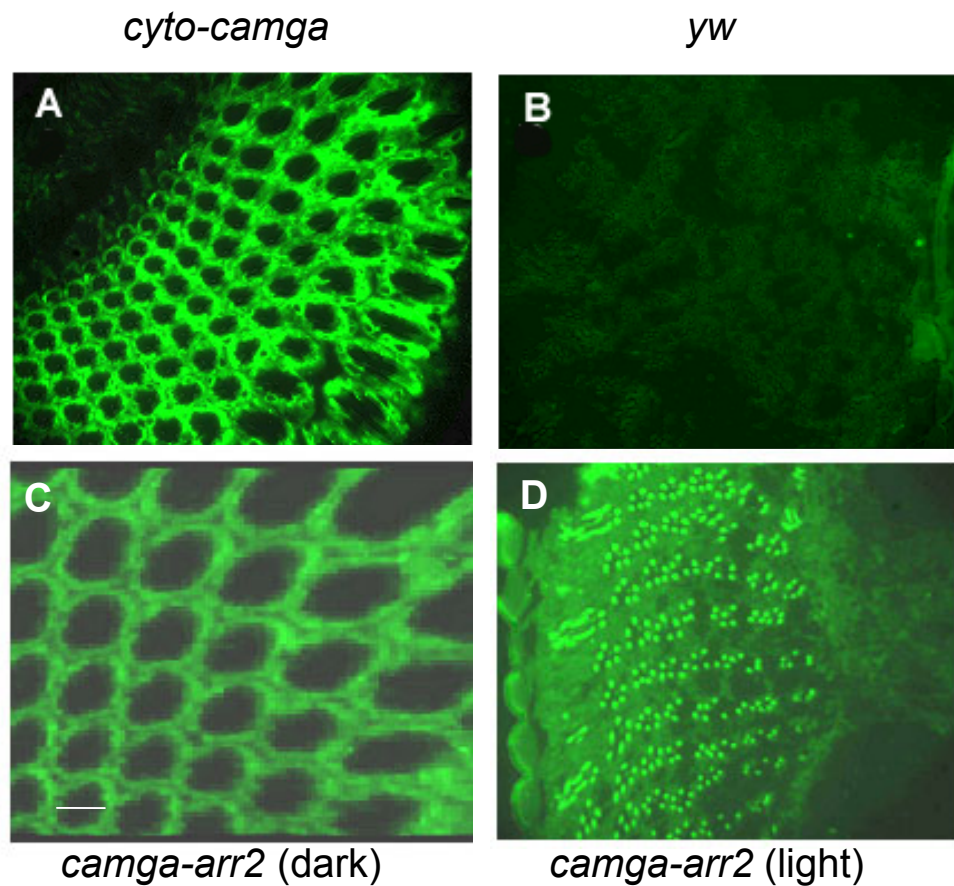


Figure 3.3: Immunofluorescence analysis of Camgaroo fly eyes. Cross sections (1 nm thickness) of A) *yw*; *cyto-Camga*, B) *yw* (control), C) light-reared *yw*; *camga-arr2*, and D) dark-reared *yw*; *camga-arr2* fly eyes stained with GFP polyclonal antibody diluted 1:10 (Clontech). Scale bar equals approximately 5 microns.

### Fluorescence Imaging of Dissociated Camgaroo Photoreceptors

We determined the localization of Camgaroo fusions in *camga-arr2*, *inaD-camga*, *camgaroo-inaD*, *dgq-camga*, and *inaD<sup>215</sup>-camga* photoreceptors by fluorescence imaging of dissociated photoreceptors with a CCD camera as outline in Chapter Five – Methodology. The Camgaroo-tagged protein nicely localized to the rhabdomeres in *camga-arr2* (see Figure 3.4A), *inaD-camga* (see Figure 3.4B), *camga-inaD* (not shown), and *inaD<sup>215</sup>-camga* (not shown) dissociated ommatidia. However as shown in Figure 3.4C, in *dgq-camga* photoreceptors the protein is dispersed throughout the rhabdomere and cell body because of light-dependent movement of dgq out of the rhabdomere (Kosloff and al. 2003). Based on the localization of the Camgaroo-tagged proteins to the rhabdomere, we concluded from these experiments that all of these mutants except for *dgq-camga* would be appropriate for rhabdomeric fluorescence imaging.

### Electrophysiological Analysis of Camgaroo Photoreceptors

To determine whether or not the Camgaroo fusion proteins adversely affect the light response, we conducted patch clamp analysis of several Camgaroo fly lines (see Figure 3.5). Light-induced inward currents (LIC) were recorded as outlined in Chapter Five – Methodology. In general, all Camgaroo photoreceptors showed no defects in kinetics and amplitude compared to wild type (see figure 3.5A). In *inaD-camga* LIC, we observed a

**Figure 3.4 – Fluorescence Imaging of Dissociated Camgaroo Photoreceptors**

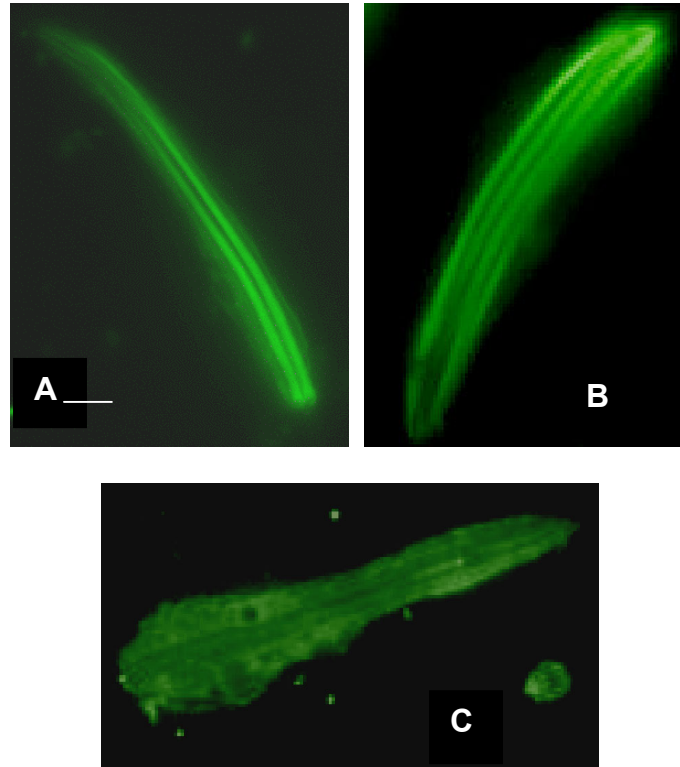


Figure 3.4: Fluorescence images of A) *yw; camga-arr2*, B) *yw; inaD-camga*, and C) *yw;;dgg-camga* dissociated ommatidia. The long structures in shown in green in A and B are individual rhabdomeres of photoreceptor cells. Scale bar equals approximately 5 microns.

slight delay in activation and deactivation compared to wild type and *camga-inaD* responses (see Figure 3.5B). However, when the InaD-Camgaroo indicator was crossed into either *inaD*<sup>215</sup> or *inaC* null backgrounds, phenotypes are observed that are similar to phenotypes of these mutants that do not express the InaD-Camgaroo indicator (see Figure 3.5C and D). These results indicated that proper visual physiology occurs in photoreceptors expressing



Figure 3.5 – Patch Clamp Analysis of Camgaroo Flies

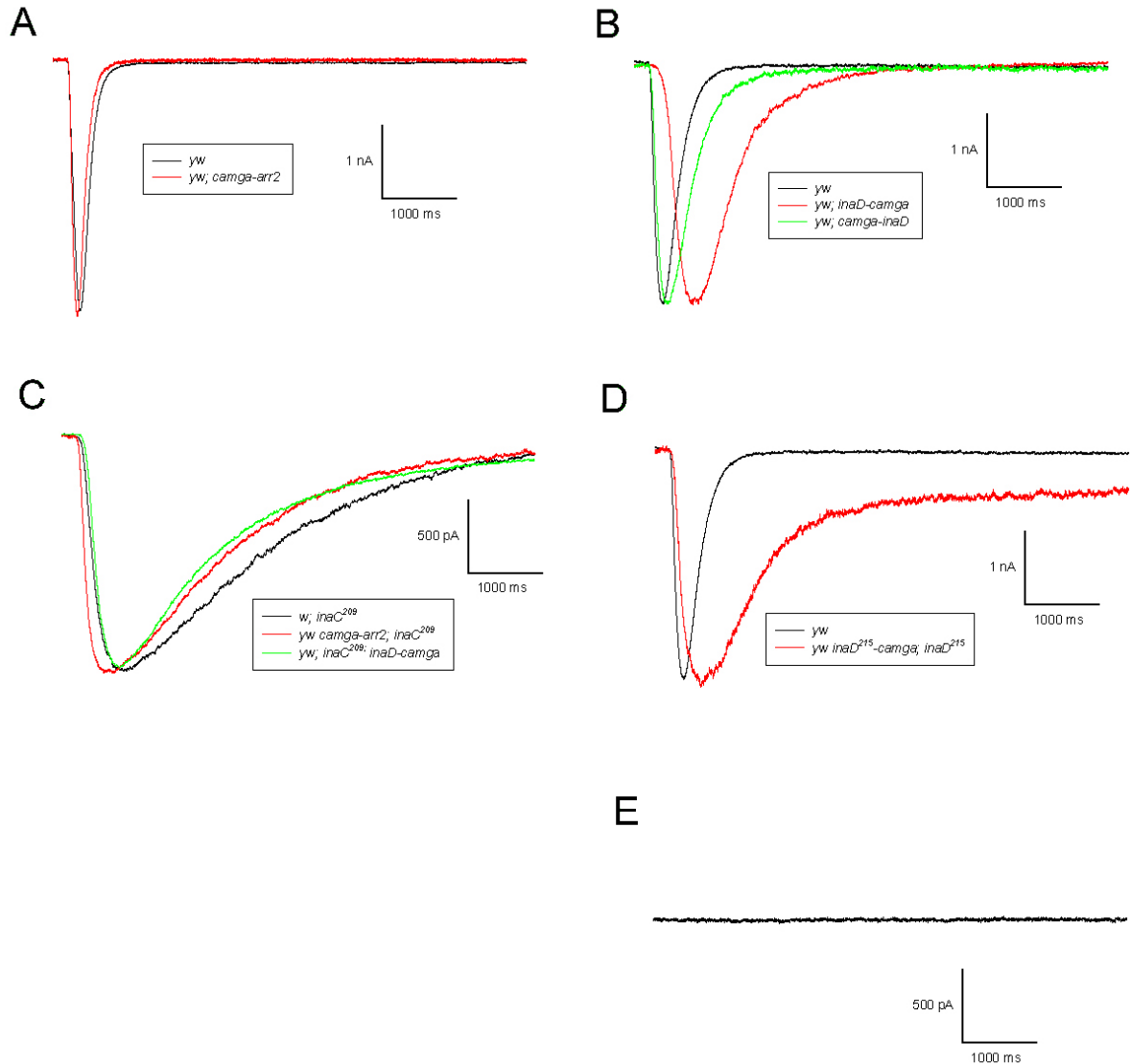


Figure 3.5: Patch clamp analysis of Camgaroo flies. A) Overlay of normalized and averaged *yw* (black, N = 5) and *yw; camga-arr2* (red, N = 4) light responses at 1.5 mM  $\text{CaCl}_2$ . B) Overlay of single normalized *yw* (black), *yw; inaD-camga* (red), and *yw; camga-inaD* (green) light responses at 500  $\mu\text{M}$   $\text{CaCl}_2$ . C) Overlay of normalized and averaged *w; inaC<sup>209</sup>* (black, N=2), *yw camga-arr2; inaC<sup>209</sup>* (red, N = 1), and *yw; inaC<sup>209</sup>; inaD-camga* (green, N = 3) light responses at 500  $\mu\text{M}$   $\text{CaCl}_2$ . D) Overlay of normalized and averaged *yw* (black, N=3), and *yw inaD<sup>215</sup>-camga* (red, N = 2) light responses at 500  $\mu\text{M}$   $\text{CaCl}_2$ . E) Averaged *yw norpA<sup>P41</sup>; inaD-camga* (N = 3) light responses at 500  $\mu\text{M}$   $\text{CaCl}_2$ . Light responses for *yw norpA<sup>P41</sup>* and *yw norpA<sup>P41</sup>; camga-inaD* look identical to E) but are not shown here for clarity. Light stimulus occurs at 25 ms and holding potential = -60mV for all recordings.

InaD-Camga. All Camgaroo fusions showed, as expected, no light responses when crossed with *norpA*<sup>P41</sup> flies that lack the NorpA PLC-β and are functionally blind (see Figure 3.5E).

### *In vitro* Calcium Titration Curves

To confirm the properties of Camgaroo as shown in the literature (Baird, Zacharias et al. 1999), we expressed cyto-Camga protein in *E. coli* and purified it for use in *in vitro* calcium titrations using a fluorescence spectrometer (see Chapter Five – Methodology). We calculated an apparent  $K_d$  of 7 μM and a Hill coefficient of 1.3 (see Figure 3.6), which is in agreement with published values (Baird, Zacharias et al. 1999).

**Figure 3.6 – *In vitro* Calcium Titration Curve for Camgaroo Protein**

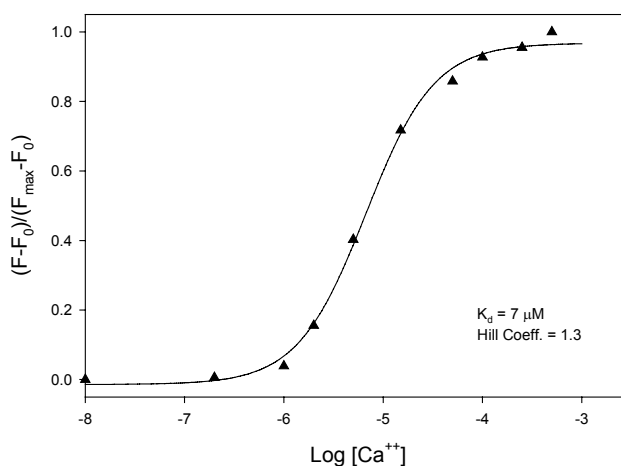


Figure 3.6: *In vitro* calcium titration curve for Camgaroo protein. Increasing amounts of 100 mM CaCl<sub>2</sub> were added to 10 nM purified Camgaroo protein at pH 7.5. Normalized fluorescence is graphed versus the log [Ca<sup>2+</sup>]. The Hill coefficient value of 1.3 was calculated by fitting the Hill equation to a graph of normalized fluorescence vs. [Ca<sup>2+</sup>]. Average of 3 experiments.

## **FLUORESCENCE MEASUREMENTS OF CAMGAROO**

### **DISSOCIATED OMMATIDIA**

#### **Comparison of Flies Expressing Camga-Arr2, InaD-Camga, and cyto-Camga**

To measure the fluorescence in rhabdomeres of photoreceptors expressing Camgaroo-tagged proteins, ommatidia were dissociated as for patch clamp experiments as outlined in Chapter Five (Methodology), and placed in bath solution containing a concentration of  $\text{CaCl}_2$  ranging from 0 to 1.5 mM. Fluorescence measurements were collected as outlined in Chapter Five (Methodology) using a photodiode that reported the levels of fluorescence at a sampling rate of 30 Hz ( $\lambda_{\text{excitation}} = 495 \text{ nm}$ ,  $\lambda_{\text{emission}} = 515 \text{ nm}$ ) for 1000 ms. Shown in Figure 3.7A is an overlay of photodiode traces from 5 different *yw; camga-arr2* cells in 1.5 mM  $\text{CaCl}_2$ . The traces are normalized to the basal fluorescence at the time of the first flash ( $t = 25 \text{ ms}$ ) and the average peak fluorescence. We note that since the calcium levels recorded for the first flash are well below the response time of each cell, the first time point provides an excellent measurement of the “dark state” calcium level. Using this protocol, data were also obtained for *inaD-camga*, *cyto-camga*, *norpA<sup>P41</sup>*; *inaD-camga*, *inaC<sup>209</sup>*; *inaD-camga*, and *inaD<sup>215</sup>-camga*; *inaD<sup>215</sup>* photoreceptors.

### Figure 3.7 – Fluorescence Imaging of Camgaroo Dissociated Ommatidia Using a Photodiode

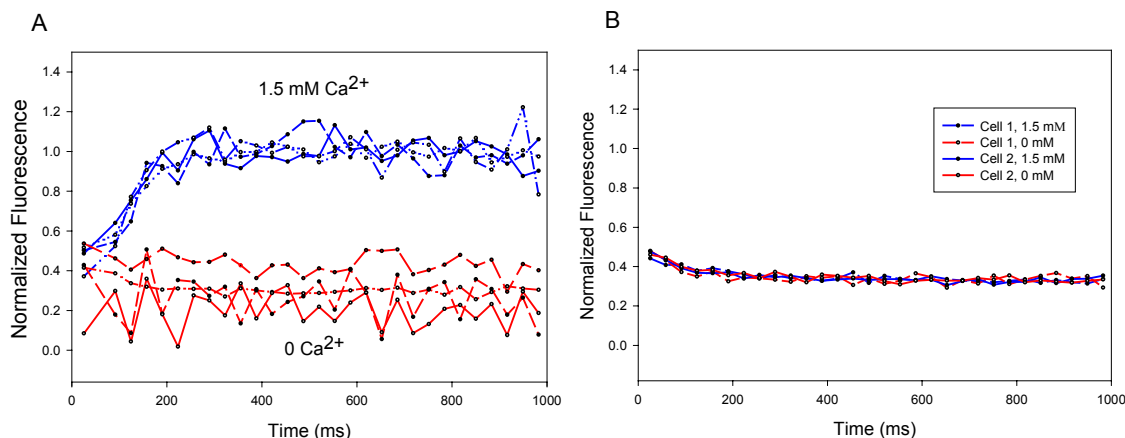


Figure 3.7: Overlay of representative photodiode traces recorded from fluorescence imaging of A) *yw; camga-inaD* ommatidia (N = 4) titrated in 1.5 mM  $\text{CaCl}_2$  (upper blue lines) and 0  $\text{CaCl}_2$  (lower red lines), and B) *norpA<sup>P41</sup>; inaD-camga* dissociated ommatidia (N = 2) titrated in 1.5 mM  $\text{CaCl}_2$  (blue) and 0  $\text{CaCl}_2$  (red). Laser pulse ( $\lambda = 495 \text{ nm}$ ) began at 25 ms and continued every 33.3 ms (black points). Top traces for A) were normalized by applying the formula  $(F - F_0)/(F_{\text{max}} - F_0)$  where  $F_0$  equals the basal fluorescence at  $t = 25 \text{ ms}$  and  $F_{\text{max}}$  equals the average peak fluorescence. Lower traces were normalized to average peak fluorescence of upper traces. Traces for B) were normalized by multiplying initial fluorescence value by fold fluorescence change (see Table 3.8).

At 1.5 mM  $\text{CaCl}_2$ , the rhabdomeres of *camga-arr2*, *inaD-camga*, and *inaC209; inaD-camga* photoreceptors showed an immediate increase in fluorescence in the rhabdomeres corresponding with the influx in calcium that occurs upon light stimulation. The time to peak fluorescence and the fold change in fluorescence were indistinguishable in *inaD-camga*, and *camga-arr2* photoreceptors (see Figure 3.8A and Table 3.8). The time to peak fluorescence in these flies is approximately 100 ms, which is in agreement with previous studies of fly photoreceptors done using fluorescent calcium dyes (Peretz 1994; Ranganathan, Bacsikai et al. 1994). The absence of significant differences between flies expressing Camga-Arr2 and InaD-Camga indicates that these proteins are reporting identical levels of calcium, and thus suggests that localized microdomains of calcium do not occur within the rhabdomere.

**Figure 3.8 – Time to Peak Fluorescence and Fold Change  
in Camgaroo Dissociated Ommatidia**

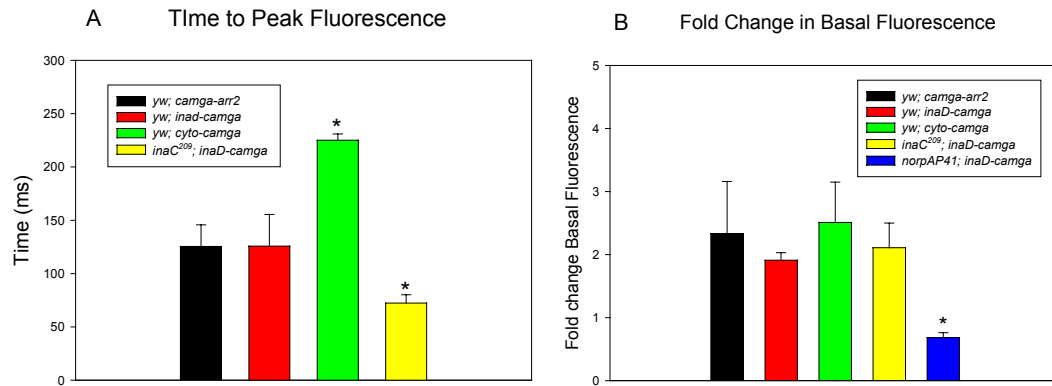


Figure 3.8: Comparison of A) mean time to peak fluorescence and B) mean fold change in basal fluorescence of *yw; camga-arr2* (black), *yw; inaD-camga* (red), *yw; cyto-camga* (green), *inaC<sup>209</sup>; inaD-camga* (yellow), and *norpA<sup>P41</sup>; inaD-camga* (blue) dissociated ommatidia in 1.5 mM CaCl<sub>2</sub>. Time to peak fluorescence calculated by measuring the rise time to the mean steady-state fluorescence value for each cell. Fold change is calculated by dividing mean peak fluorescence by initial fluorescence. Asterisk (\*) denotes significant difference in non-paired student's t-test with *yw; camga-arr2* ( $p < 0.005$ ).

**Table 3.8 – Time to Peak Fluorescence and Fold Change  
in Camgaroo Dissociated Ommatidia**

Genotype	<i>camga-arr2</i>	<i>inaD-camga</i>	<i>cyto-camga</i>	<i>inaC<sup>209</sup>; inaD-camga</i>	<i>norpA<sup>P41</sup>; IC</i>
Time to peak	125 ± 20	126 ± 30	225 ± 6*	72 ± 8*	N/A
Fold Δ	2.33 ± 0.83	1.91 ± 0.12	2.51 ± 0.64	2.11 ± 0.39	0.68 ± 0.08*

Table 3.8: Mean values for time to peak fluorescence and normalized fold change in fluorescence (norm. fold Δ) of *yw; camga-arr2* (N = 6), *yw; inaD-camga* (N = 3), *yw; cyto-camga* (N = 4), *inaC<sup>209</sup>; inaD-camga* (N = 5), and *norpA<sup>P41</sup>; IC* (N = 4) dissociated ommatidia at 1.5 mM CaCl<sub>2</sub>. Time to peak fluorescence calculated by measuring the rise time to the mean steady-state fluorescence value for each cell. Fold change is calculated by dividing mean peak fluorescence by initial fluorescence. Asterisk (\*) denotes significant difference in non-paired student's t-test with *yw; camga-arr2* ( $p < 0.005$ ).

Because the on-rate of the calmodulin component of Camgaroo ( $2 \times 10^8 \text{ M}^{-1} \text{ s}^{-1}$ ) is similar to Fluo-4 ( $10 \times 10^8 \text{ M}^{-1} \text{ s}^{-1}$ ), a fluorescent calcium dye commonly used to measure very fast calcium transients in neurons (Goldberg, Tamas et al. 2003), it is not likely that the kinetics of calcium influx are too fast for detection with our Camgaroo indicators.

In photoreceptors expressing cyto-Camga in the cell body, the time to peak fluorescence is significantly different from that seen in *camga-arr2* and *inaD-camga* photoreceptors. These results are not unexpected, and are most likely a reflection of the time delay of the diffusion of calcium from the rhabdomere to the cell body. There is also no significant difference in the fold change in fluorescence, which indicates that the maximum calcium reported in both the rhabdomere and the cell body is roughly the same.

Importantly, we show that the Camgaroo indicator is not being saturated during any part of the response profile in imaging experiments where single ommatidia are titrated with varying concentrations of calcium (see Chapter Five – Methods). The addition of  $1 \mu\text{M}$  ionomycin, a calcium ionophore, at the end of the titration series in high calcium (see Figure 3.9) shows a large increase in calcium compared to that measured at  $1.5 \text{ mM}$  calcium without ionomycin.

In the above genotypes at zero calcium (see Figure 3.9) and in *norpA<sup>P41</sup>*; *inaD-camga* photoreceptors the levels of fluorescence remained near baseline levels (see Figure 3.7B, Figure 3.8B, and Table 3.8). These results were also not unexpected because no increase in basal calcium should be observed in photoreceptors in zero extracellular calcium or in *norpA<sup>P41</sup>* flies that have lost the ability to open TRP channels. Taken together, these

experiments provide confidence that we are indeed measuring light-induced calcium responses within the rhabdomere.

**Figure 3.9 – *In Vivo* Calcium Titration of Camgaroo Dissociated Ommatidia**

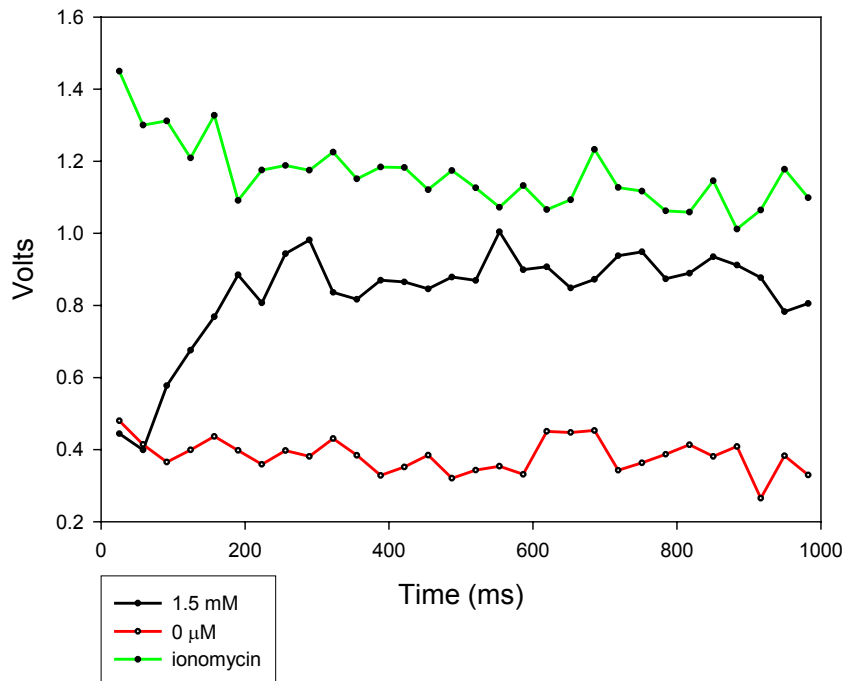


Figure 3.9: Overlay of representative photodiode traces recorded from calcium titration of single *yw; inaD-camga* dissociated ommatidia in 0 to 1.5 mM CaCl<sub>2</sub>. Laser pulse ( $\lambda = 495$  nm) began at 25 ms and continued every 33.3 ms (black points). Measurements began at 1.5 mM CaCl<sub>2</sub> and calcium levels were titrated down to 0 using a bath perfusion pump. CaCl<sub>2</sub> levels were then returned to 1.5 mM and 1  $\mu$ M ionomycin was added for the final trace in the series.

The significant decrease in the mean time to peak fluorescence in *inaC<sup>209</sup>; inaD-camga* flies compared to wild type *inaD-camga* flies is surprising, however (see Figure 3.8 and Table 3.8). It is approximately 50 ms faster than in wild-type flies. This may be explained by noting the corresponding time course of an electrical response. In wild type

flies, the electrical response has an average time to peak of approximately 30 ms (see Chapter Two) when TRP channels are opening, and at 100 ms TRP channels have already started to close and the inward current declines back to baseline levels. Since TRP is thought to be phosphorylated and deactivated by InaC (Hardie and al. 1993; Adamski, Zhu et al. 1998; Kumar and Shieh 2001), in the absence of InaC the TRP channels may stay open longer and thus cause a faster rise in intracellular calcium compared to wild type cells. Because the normalized fold fluorescence did not change in relation to wild type, it also seems likely that the rhabdomeres of *inaC* flies are not filled with higher amounts of calcium than wild type rhabdomeres.

#### Comparison of Flies Expressing InaD-Camga and InaD<sup>215</sup>-Camga

The main goal of our fluorescence measurements was to investigate whether the cause of the *inaD*<sup>215</sup> phenotype is the mislocalization of the entire InaD complex, including InaC, from the microgradients of calcium that exist near TRP. We thus performed fluorescence recordings on the dissociated ommatidia of both light- and dark-reared *inaD-camga* and *inaD*<sup>215</sup>-*camga* flies as outlined above, and compared the mean time to peak fluorescence and the mean fold fluorescence change. As shown in Figure 3.10 and Table 3.10, we observed no significant difference between the mean time to peak fluorescence and the mean fold fluorescence change of either light- or dark-reared *inaD-camga* and *inaD*<sup>215</sup>-*camga* flies.



The large standard deviation of the mean time to peak fluorescence of light-reared *inaD*<sup>215</sup>-*camga* reflects the high variability of this parameter in these flies ( $74 \pm 52$ ). In 5 different cells, some had values as low as 30 ms (closer to that of *inaC*) while others had values as high as 140 (near that of wild-type). These results may be due to variability in the health of these cells since the ommatidia of light-reared flies tend to look unhealthier than those of dark-reared flies.

**Figure 3.10 – Time to Peak Fluorescence and Fold Change in Light-Adapted vs. Dark-Adapted Wild Type and *inaD*<sup>215</sup> Ommatidia**

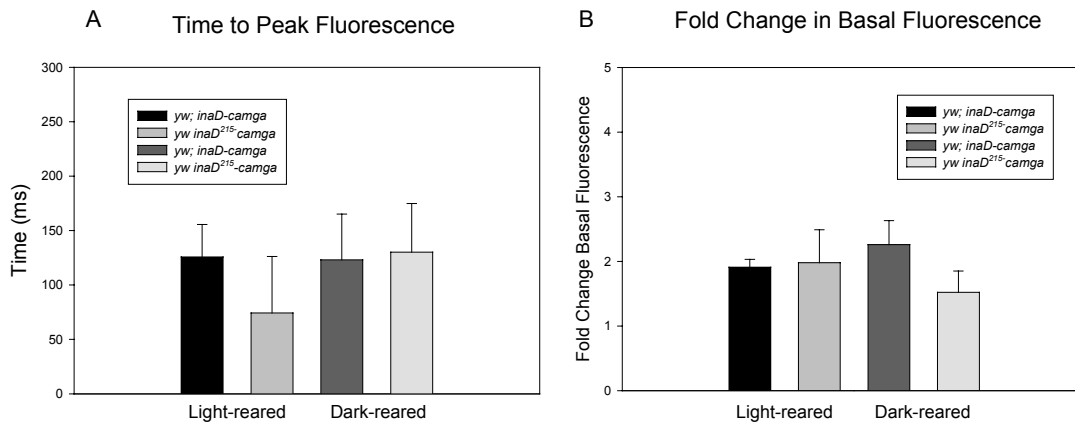


Figure 3.10: Comparison of A) mean time to peak fluorescence and B) mean fold change in fluorescence of light-reared and dark-reared *yw; inaD-camga* (black and charcoal), and *yw inaD<sup>215</sup>-camga* (dark and light gray) dissociated ommatidia. Time to peak fluorescence calculated by measuring the rise time to the mean steady-state fluorescence value for each cell. Fold change is calculated by dividing mean peak fluorescence by initial fluorescence. Non-paired student's t-test shows no significant difference between light- and dark-reared flies or between lines ( $p < 0.05$ ).

Taken together, these results indicate that InaD-Camga and InaD215- Camga are reporting similar levels of calcium, and thus that the phenotype of *inaD*<sup>215</sup> is not due to the reduced kinase activity of InaC caused by a low local concentration of calcium. Therefore, it is likely that the *inaD*<sup>215</sup> phenotype is due to the lack of formation of the InaD complex, and the inability of InaC to phosphorylate and shut off TRP.

**Table 3.10 – Time to Peak Fluorescence and Fold Change  
in Light-Reared vs. Dark-Reared Wild Type and *inaD*<sup>215</sup> Ommatidia**

Genotype	<i>yw; inaD-camga</i> (Light-reared)	<i>yw; inaD<sup>215</sup>-camga</i> (Light-reared)	<i>yw; inaD-camga</i> (Dark-reared)	<i>yw; inaD<sup>215</sup>-camga</i> (Dark-reared)
Time to Peak	126 ± 30	74 ± 52	123 ± 42	130 ± 45
Fold Δ	1.91 ± 0.11	1.98 ± 0.51	2.26 ± 0.37	1.52 ± 0.33

Table 3.10: Mean values for time to peak fluorescence and fold change in fluorescence of light reared and dark-reared *yw; inaD-camga* (N = 3, N = 5), and *yw inaD<sup>215</sup>-camga* (N = 5, N = 4) dissociated ommatidia. Time to peak fluorescence by measuring the rise time to the mean steady-state fluorescence value for each cell. Fold change is calculated by dividing mean peak fluorescence by initial fluorescence. Non-paired student's t-test shows no significant difference between light- and dark-reared flies or between lines ( $p < 0.05$ ).

## CONCLUSIONS

In this chapter, we demonstrate that we can measure calcium in the rhabdomeres of fly photoreceptors by measuring the fluorescence of photoreceptors expressing visual proteins tagged with Camgaroo, a calcium-sensitive derivative of yellow fluorescent protein. We also show that we are indeed measuring calcium by observing that there is no increase in

baseline fluorescence in zero extracellular calcium and in *norpA* null mutants. A comparison of fluorescence measurements of rhabdomeres expressing Camga-Arr2, a rhabdomeric protein not associated with the InaD complex, InaD-Camga, and InaD<sup>215</sup>-Camga indicates that discrete microdomains of calcium are likely not to exist within the rhabdomere. A simpler model is that the entire microvillus is homogenously flooded with calcium upon light stimulation. Micro-organization of signaling components is present in the rhabdomere, but it occurs instead through the precise assembly of target proteins into a macromolecular machine anchored by InaD.

We also show that free Camgaroo (cyto-Camga) reports a slower rate of increase of calcium and lower levels of peak calcium due to its localization to the cell body of the photoreceptor cell. This result is in agreement with previous calcium imaging experiments showing influx of calcium after light stimulation beginning in the rhabdomere, and then spreading to the cell body (Ranganathan, Bacsikai et al. 1994).

The most surprising result is our analysis of *inaC*<sup>209</sup>; *inaD-camga* photoreceptors, which exhibited an increased rate of calcium increase compared to wild type. Since TRP channels begin to close before the time of peak calcium levels in the rhabdomere (~100 ms) in wild type photoreceptors, this suggests that InaC phosphorylates and inactivates TRP channels as a part of calcium-mediated negative feedback regulation.

Lastly, we have shown that there is no significant difference between the rate of calcium increase or the peak calcium levels reported by InaD-camga in wild type and *inaD*<sup>215</sup> photoreceptors. Therefore, the phenotype of *inaD*<sup>215</sup> is not due to reduced kinase activity of InaC caused by insufficient levels of local calcium, but most likely it is a result of

mislocalization of InaC away from its primary target, TRP. The results of our study of calcium signaling in wild type versus *inaC*<sup>209</sup> and *inaD*<sup>215</sup> photoreceptors support our contention that the major function of the InaD complex is to localize all of the proteins involved in calcium-mediated negative feedback regulation into a well-ordered macromolecular complex.

## **CHAPTER FOUR**

### **Conclusions and Recommendations**

#### **ANALYSIS OF INAD PDZ DOMAIN REARRANGEMENT**

##### Conclusions

In Chapter Two we showed that the “specific quaternary structure” model, where precise orientation of the members of a signaling complex in relation to each other is key for efficient signaling, best applies to the *Drosophila* photoreceptor scaffolding protein InaD. We accomplished this by rearranging the PDZ domains of InaD and demonstrating through electrophysiological analysis that positive feedback regulation (activation), but not negative feedback regulation (deactivation) of the light response, is rescued in these mutants despite the stability of the components of the InaD complex over time. We also showed that the deactivation defect in light responses of these mutants is epistatic to calcium, which supports our theory that the primary function of InaD is to specifically localize the components involved in calcium-mediated negative feedback regulation in close proximity to one another in order to increase the speed and efficiency of deactivation of the light response.

## Recommendations

Although we have demonstrated through western blot analysis that most of the components of the InaD complex remain stable over several days in the shuffled InaD constructs, we have not yet demonstrated that the mutant constructs bind these components with an affinity similar to that of wild type InaD. Therefore, we suggest that further coimmunoprecipitation experiments be attempted in order to show whether or not the shuffled InaD constructs bind their partners *in vitro*.

Alternatively, one could try to express and purify wild type and shuffled InaD proteins, and conduct *in vitro* binding experiments with the peptide sequences for the C termini of TRP, NorpA, and InaC. However, our lab has already attempted to express stable wild type InaD in both *E. coli* and baculovirus with minimal success (Rohit Sharma and Jennifer Graves, unpublished). A different cell expression system would need to be tested for expression of wild type InaD, and even then expression of the shuffled constructs may be much more difficult due to the possibility increased instability.

While two out of the five PDZ domains of InaD have already been solved, the complete structure of full length InaD in complex with its binding partners would be ideal in visualizing how the components of the complex are arranged in proximity with one another. However, as with the *in vitro* binding experiments described, the issue of purifying large amounts of protein for use in x-ray crystallography or NMR experiments makes this suggestion impractical at this time.

However, further analysis of the light responses of the shuffled InaD mutants could be easily accomplished at this time. One might compare the quantal and adaptive responses in these mutants with wild type and other mutants defective in deactivation of the light response, such as *inaD*<sup>2</sup>, *inaD*<sup>215</sup>, and *inaC*<sup>209</sup>.

## **FLUORESCENCE MEASUREMENTS OF CAMGAROO**

### **IN DISSOCIATED OMMATIDIA**

#### **Conclusions**

In Chapter Three of this dissertation, we developed a method for calcium imaging of rhabdomeres of photoreceptors using a genetically encodable calcium indicator, Camgaroo, to tag rhabdomeric proteins. The results of our experiments utilizing this technique support the idea that easily resolvable microdomains of calcium are not created in the microvilli of the rhabdomere upon light stimulation, but rather that the entire length of the microvillus is quickly flooded with calcium. The results of our imaging studies of *inaC*<sup>209</sup> photoreceptors suggest that the primary role of InaC in negative feedback regulation is to phosphorylate and inactivate TRP channels. Likewise, the results of our imaging studies of *inaD*<sup>215</sup> photoreceptors support our claim that the primary function of InaD is to localize the main players in calcium-mediated feedback regulation, namely InaC and TRP, in close proximity to one another.

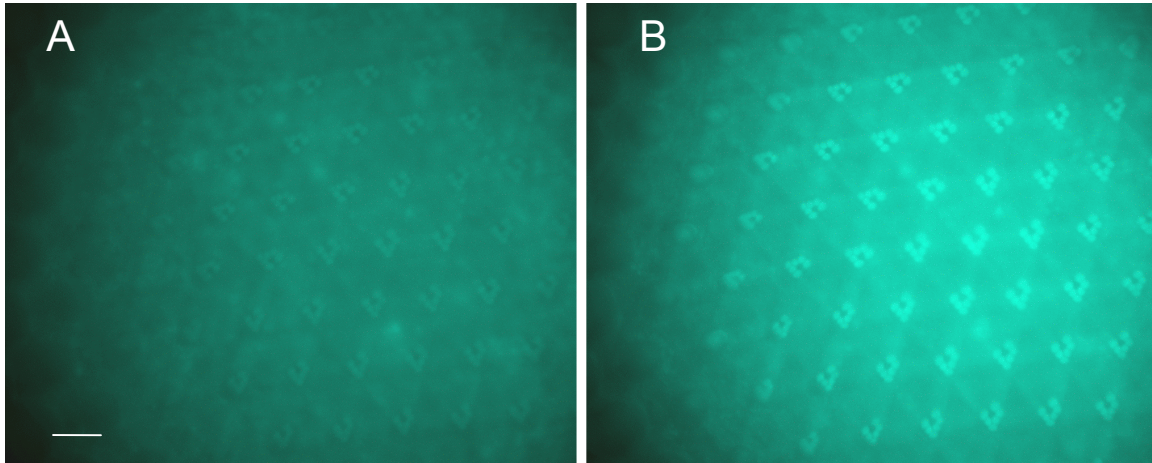
## Recommendations

Although we were not able to resolve microdomains of calcium using Camgaroo as an indicator, we did show that it could be used to measure calcium in parts of cells with very small dimensions. Conceivably, Camgaroo could be used in fluorescence imaging of other types of cells containing tiny substructures, such as the dendritic spines of neurons. Additionally, Camgaroo indicators could be used to analyze calcium dynamics in photoreceptors of other deactivation mutants, including calmodulin (*cam*) and *ninaC*.

Based on the unique structure of the compound eye of the fly, we were also able to conduct fluorescence imaging of the rhabdomeres of living flies (see Figure 4.1). The rhabdomere, because of its stacks of light sensitive microvilli, acts a single mode waveguide and conducts light along its axis. Different types of genetically encodable fluorescent indicators--calcium, pH, or redox--could be expressed in the rhabdomeres. One could apply various compounds to the eye of the fly and measure the effects on fluorescence. Alternatively, one could express these indicators in the rhabdomeres of blowflies, which are larger, and then test the effects of injection of various compounds into the eye on the fluorescence.



**Figure 4.1 – Fluorescence Imaging of *camga-arr2* Photoreceptors in Living Flies**



---

Figure 4.1: Fluorescence imaging of *camga-arr2* photoreceptors in a living fly. A) Light-adapted fly eye. B) Dark-adapted (~5 min.) fly eye. Scale bar equals approximately 5 microns.

**CHAPTER FIVE**  
**Methodology**  
**MOLECULAR BIOLOGY AND BIOCHEMISTRY**

Generation of Transgenic Flies

*Cloning of Shuffled InaD Constructs*

InaD shuffled constructs were generated by using PCR to copy adjacent PDZ domains of InaD and then they were spliced together using complementary ends in the arrangements indicated in Figure 5.1.

*Cloning of Camgaroo-Tagged Constructs*

Camgaroo fusion proteins were generated by adding complementary restriction ends to both the Camgaroo 2-3 SVQST gene (pRSET B vector) from Roger Tsien's lab (Baird, Zacharias et al. 1999) and either Arr2 (pet28T vector) or InaD (Bluescript KS vector) using the Polymerase Chain Reaction (PCR). The InaD<sup>215</sup> mutant was generated using internal primers containing the point mutation T1322A using PCR. The PCR fragments were first subcloned into the TA or TOPO TA vector (Invitrogen), sequenced, and then excised with the appropriate restriction enzyme. The Camgaroo fragment of the C-terminal Camgaroo fusions was subcloned into pRSETB vector via a 3' BamHI and a 5' EcoRI site. The Arr2 or InaD fragment was then subcloned into that vector at the 3' BamHI site. The Camgaroo fragment of the N-terminal Camgaroo fusions was subcloned into pRSETB vector or

Bluescript vectors via a 3' BamHI and 5' EcoRI sites. The Arr2 or InaD fragment was then subcloned into that vector at the 5' EcoRI site. The entire fusion was excised and ligated into the multiple cloning site of the modified P-element vector Yellow C4 containing the 3' and 5' UTR of the Rh1 promoter (RYC4) using the 3' SacI and 5' XhoI sites. For InaD-Camgaroo and InaD215-Camgaroo fusions, the Camgaroo and InaD fragments were generated separately by PCR, the PCR products were combined and the full-length fragment was generated in a second round of PCR. The fragment was cloned into TOPO TA vector, sequenced, excised, and ligated directly into the RYC4 vector using the 3' SacI and 5' XhoI sites.

**Figure 5.1: Design of InaD Shuffled Constructs**

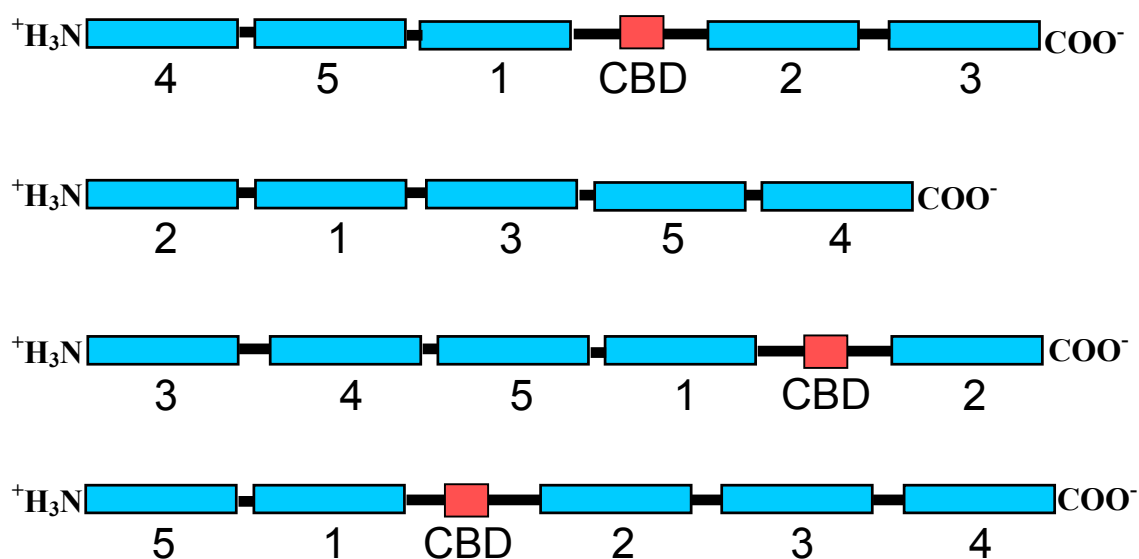


Figure 5.1: Design of InaD shuffled constructs. Ordered of numbered PDZ domains of InaD—p45123, p21354, p34512, p51234. CBD denotes the calmodulin-binding domain in between PDZ 1 and 2.

### *Fly injections*

Injection apparatus: Embryos from the *Drosophila* *yw* or *yw; inaD<sup>215</sup>* lines were injected using the Narashige IM 300 Microinjector (Injection parameters were adjusted to: Hold: 3.2; Balance: 2.5; Inject: 20; Fill: 60 psi). Protocol for fly injections obtained from Meridee Phistry of the laboratory of Helmut Kramer (UT Southwestern Medical Center).

DNA Preparation: P-element vector DNA was mixed with  $\Delta 2-3$  transposase DNA at a ratio of 3:1 by weight. The final volume of the DNA mixture was adjusted to 20  $\mu$ L in TE buffer at a final concentration of 1  $\mu$ g/ $\mu$ L. The DNA was centrifuged for 10 minutes at 13,000 rpm on a tabletop centrifuge prior to injection.

Fly Preparation: Flies were set up 3 days prior to injection in 3 cages (100-200 flies per cage) on 60x15 mm apple juice plates smeared with a dab of yeast paste. On the day of injections, the cages are transferred onto fresh plates thinly coated with yeast paste and changed hourly until ready for injection.

Embryo Injections: Fly embryos were collected from apple juice plates hourly, rinsed three times with wash buffer (0.7% NaCl, 0.1% Triton X-100), then two times with H<sub>2</sub>O, and lined up on a cover slip for injections. After air-drying for about 5 minutes, embryos were injected within one hour of being laid at 18 °C and 50% humidity.

Raising Transformed Flies: Injected embryos were kept at 25°C on 100 x 20 mm apple juice plates and greater than 50% humidity. Hatching larvae were collected, and placed into vials containing fly food and a dab of yeast paste. Adult flies hatching from the collected larvae were crossed with standard balancers and homozygous lines were obtained.

Apple juice plates: 24 g of Bacto Agar was added to 750 mL of H<sub>2</sub>O and autoclaved. The following were added: 26.4 g of sucrose, 1.6 g p-hydroxybenzoic acidmethyl ester and 250 mL of apple juice per liter of finished agar. Agar is then poured into 100 x 20 mm and 60 x 15 mm petri dishes.

### Co-immunoprecipitations

Flies were dark-adapted overnight, collected in the dark, and then flash frozen in liquid nitrogen in 50 ml conical tubes in aliquots of 1-2 g. The frozen flies were vortexed to remove the heads, which were separated in sieves. The heads were homogenized in 1 ml per gram flies of ice cold Head Homogenization Buffer (HHB) (250 mM sucrose, 120 mM KCl, 5 mM MgCl<sub>2</sub>, 10 mM MOPS pH 7.0, 1 mM DTT plus protease inhibitors). The homogenate was poured over a column containing a small amount of glass wool, and washed with 1-2 ml of HHB. The homogenate is then spun at 55,000 rpm in a Beckman Ultracentrifuge for 30 minutes at 4°C. The supernatant is discarded and the pellet is resuspended in HHB plus 0.5% CHAPS, 0.2 mM DTT, and protease inhibitors. The sample is incubated for 4 hours at 4°C on a nutator to solubilize the membranes. The insoluble particles are spun out at 15,000 rpm for 5 minutes in a tabletop microcentrifuge. 8 µl (~30 µg per gram of flies) of INAD-PDZ1 antibody are added to the sample and then the mixture is incubated overnight at 4°C. Protein A agarose beads (Pierce) are added to the supernatant (100 µl beads per 10 µg antibody), and the sample is incubated for 2 hours at 4 °C. The beads are spun down and the supernatant (or flow-thru, FT) is removed. The beads are washed six times in 0.5-1 ml HHB plus 0.5%

CHAPS and 0.2 mM DTT. The beads are washed a final time with 0.5 ml H<sub>2</sub>O. The immunocomplexes are eluted twice by adding 100 µl 2X SDS Head Buffer (60 mM Tris-HCl, [pH 6.8], 20% SDS, 0.0004% bromophenol blue, 10% β-mercaptoethanol, 20% glycerol) incubating 5 minutes at 95 °C, and then removing the eluant.

#### Western Blot Analysis of Co-Immunoprecipitations or Camgaroo Expression

For co-immunoprecipitations, samples (FT and eluant) are diluted 1:1 in 2X SDS Head Buffer. For analysis of Camgaroo expression, 4 fly heads are sonicated in 160 µl 1X SDS Head Buffer and incubated for 30 minutes at 37 °C. Samples (10 µl per lane) are fractionated by SDS-PAGE (5 or 10% gel) and electrotransferred to PVDF membrane (Biorad) for 1.5 to 2 hours at 100 V. The membrane was blocked with PBS (phosphate buffer solution pH 7.5) plus 0.5 % Tween-20 (T-PBS) plus 5 % nonfat dried milk for 20 minutes at room temperature and incubated overnight at 4 °C with antibody diluted in T-PBS plus 5 % milk. The membrane is then washed five times with T-PBS and incubated 2-4 hours at room temperature with anti-rabbit Ig horseradish peroxidase-linked whole antibody from Donkey (Amersham Biosciences) diluted 1:1000 in T-PBS plus 5 % milk. The blot was washed five times with T-PBS and incubated 1 min. with ECL kit reagents (Amersham Biosciences) mixed 1:1, and then diluted 1:1 with H<sub>2</sub>O. The blot was then exposed to film and developed. Expression of Camgaroo proteins was quantified by measurement of pixel intensity in Adobe Photoshop 7.0.

### Fluorescence Immunochemistry of Camgaroo Fly Eyes

3-4 fly heads were fixed by addition of 1 ml 4% formaldehyde in PBS (pH 7.5) and incubated for 4 hours at 4° C. Heads were washed two times with 1 ml PBS and then stored at 4° C in 2.3 M sucrose in PBS. Heads are frozen in liquid nitrogen and 0.5-1 µm thick sections of the eyes are cut using a microstat at -130°C. The sections were mounted on slides and covered with 2.3 M sucrose in PBS for storage at 4° C. The sections were gently rinsed with PBS and incubated for 30 minutes in PBS plus 0.1% saponin in Coplin jars. The sections were then incubated for one hour at room temperature with GFP polyclonal antibodies (Clontech) diluted 1:10 in 50 µl PBS plus 0.1% saponin. The sections were gently washed four times (at 3-4 minutes each) with PBS buffer. Then the sections were incubated with 50 µl anti-rabbit IgG F(ab')<sub>2</sub> fragment conjugated with fluorescein (Amersham Biosciences) diluted 1:8 in PBS plus 0.1% saponin for 1 hour in the dark at room temperature. The sections were again washed four times with PBS. A drop of anti-bleaching agent p-phenylenediamine (Sigma) in 90% glycerol was added to the slide. The sections were imaged with an inverted fluorescence microscope (Olympus IX70) and photographed using a Princeton Instruments CCD camera. Images were acquired and processed with Axon Imaging Workshop (Axon Instruments).

### Purification of Camgaroo Proteins

Camgaroo DNA was subcloned into pRSET B (Invitrogen) as an N-terminal His<sub>6</sub>-tagged construct. His<sub>6</sub>-Camgaroo was expressed by overnight IPTG (100  $\mu$ M) induction at 18°C in BL21(DE3) cells grown to an O.D.<sub>600</sub> of 1.6. in Terrific Broth (TB). Cells were harvested, and resuspended in Binding Buffer (10 mM Tris-HCl [pH 8], 0.5 M NaCl, 10 mM imidazole, 0.1% Tween-20, 1 mM PMSF, leupeptin [10  $\mu$ g/ml], and pepstatin [2  $\mu$ g/ml]). DNase I [1 mg/ml] was added at 0.5 ml per liter of cell culture and cells were lysed with sonication. Lysate was then spun at 20,000 rpm for 1 hour at 4°C. The supernatant was decanted and incubated with Ni-NTA beads (Qiagen) for one hour at 4°C. Protein was column purified with several washes of binding buffer plus Tween-20, and then with binding buffer without Tween-20. Protein was eluted with elution buffer (same as binding buffer except with 250 mM imidazole), and then dialyzed into protein buffer (50 mM Tris-HCl [pH 8.0], 100 mM NaCl, and 1 mM DTT) at 4°C. Protein was concentrated down to 5-10 mg/ml, aliquoted, and flash frozen in liquid nitrogen.

### Calcium Fluorescence Titrations:

In a 1 ml quartz cuvette, 20  $\mu$ l of purified Camgaroo protein (1.2 mg/ml) was added to 1 ml of buffer (100 mM KCl, 10 mM MOPS [pH 7.5], and 1 mM BAPTA)(Baird et al. 1999). Increasing amounts of 100 mM CaCl<sub>2</sub> were added to give a range of 2  $\mu$ M to 5 mM CaCl<sub>2</sub>, and the average fluorescence (excitation = 495 nm, emission = 515 nm) was measured



in a fluorescence spectrometer (Photon Technology International). Fractional conversion to the calcium-bound state was calculated using the equation  $\frac{(F - F_0)}{(F_{\max} - F_0)}$  where F is the average fluorescence measured,  $F_0$  is the fluorescence at zero calcium, and  $F_{\max}$  is the maximum fluorescence at the highest calcium concentration.

## **ELECTROPHYSIOLOGY**

### **Dissociation of Fly Ommatidia**

Fly stocks not used in fluorescence imaging experiments were dark adapted overnight prior to retinal dissection. All preparations for physiological recordings were conducted under dim red light. The heads of flies eclosing from the pupal case were removed and their retinas torn into multiple pieces with fine-tipped forceps. The pieces were triturated seven times before finally being placed on a glass coverslip in a recording dish containing 800  $\mu$ l bath solution. Ideal preparations were made by using 3-4 fly heads and two drops of dissection solution with a maximum of two heads per 10  $\mu$ l of solution.

### **Solutions**

All chemicals were obtained from Sigma.

*Dissection*

125mM CsCl<sub>2</sub>, 10mM Hepes, 30mM Sucrose, pH 7.15

*Bath Solution*

120 mM NaCl, 5 mM KCl, 10mM HEPES, 4mM MgCl<sub>2</sub>, 24 mM Proline, 5 mM Alanine, 1mM CaCl<sub>2</sub>, pH 7.15

*Electrode Solution*

140 mM Potassium Gluconate, 10 mM Hepes, 2 mM MgSO<sub>4</sub>, 1 mM NAD, 4 mM MgATP, 0.5 mM NaGTP, 0.5 mM EGTA, pH 7.15

Patch Clamp Recordings

Cell preparations were visualized in the dark through Hoffman interference contrast on an inverted light microscope (Olympus IX70). Dim red light was used for illumination. Whole cell recordings were made as reported previously (Ranganathan, Harris et al. 1991) using standard techniques. Cells were recorded at room temperature with fiber-filled borosilicate glass pipettes (O.D. 1.0mm, I.D. 0.5mm, 10cm length, Sutter Instruments, Novato, CA) pulled by a P97 Pipette Puller (Sutter Instruments) and fire-polished to a final tip resistance of 7-8 MΩ. Currents were amplified by an Axopatch 200B (Axon Instruments, Foster City, CA) and filtered at 10kHz. For light-induced currents over -500 nA, recordings were series resistance compensated to (80%), with series resistance varying between (10-25MΩ) and cell capacitance between (32-64pF). . These recordings were digitized at a rate of 5 or 2.5kHz on the PCI-605 2E data acquisition board (National Instruments).

Light responses were elicited by a 3ns pulse of 492 nm light from a dye stirred, VSL-337ND-S Nitrogen Laser (Laser Science, Inc., Newton, MA). Neutral density filters were used to control light intensity (Oriel, Stratford, CT). Power output of lasers was monitored by a dual channel joulemeter EPM2000 (Molelectron Detector, Inc., Portland, OR). Lasers were controlled through a 4 port digital board, PCI-DIO-32HS (National Instruments, Austin, TX).

Custom data acquisition programs and a data analysis program were written in VisualBasic.NET (Microsoft) by Prashant Mishra using the additional modular software package Measurement Studio (National Instruments). Code is detailed in Appendix A. Additional Matlab m files (written by Prashant Mishra and Rama Ranganathan) used to complete data analysis are listed in Appendix B.

## **CALCIUM IMAGING**

### *Fluorescence Imaging Using a CCD Camera*

Cell preparations were visualized in the dark through an inverted fluorescence microscope (Olympus IX70) using YFP filters (Chroma). Cells were illuminated with 495 nm light from Xenon light source (Photonics Corporation). Images were acquired and processed with a Micromax 5Hz CCD camera (Princeton Instruments) and either Axon Imaging Workbench version 2.2 (Axon Instruments, Foster City, CA) or Metamorph software version 6 (Universal Imaging Corporation).

*Fluorescence Imaging Using a Photodiode*

Calcium measurements were done with a photodiode (Photonics Corporation) at a rate of 30 Hz with a collection time of 1 to 1.2 seconds with excitation at 492 nm, and emitted light collected at 515 nm. For calcium titration experiments, calcium was titrated down from 1.5 mM to  $\sim 2 \mu\text{M}$   $\text{CaCl}_2$  by bath exchange with a perfusion pump model VC-6 (Warner Instruments corporation) with generally 4 minutes of perfusion between measurements. Then, the  $\text{CaCl}_2$  was increased back up to 1.5 mM by perfusion and  $1 \mu\text{M}$  of ionomycin was added. Imaging resumed 3 minutes after ionomycin was applied. Data acquisition program for imaging with photodiode was the same as used for electrophysiology experiments.

# APPENDIX A

## LIGHT RESPONSE ACQUISITION PROGRAM

Written by Prashant Mishra

```
Imports System
Imports System.Reflection
Imports System.Runtime.InteropServices

' General Information about an assembly is controlled through the following
' set of attributes. Change these attribute values to modify the information
' associated with an assembly.

' Review the values of the assembly attributes

<Assembly: AssemblyTitle("")>
<Assembly: AssemblyDescription("")>
<Assembly: AssemblyCompany("")>
<Assembly: AssemblyProduct("")>
<Assembly: AssemblyCopyright("")>
<Assembly: AssemblyTrademark("")>
<Assembly: CLSCompliant(True)>

'The following GUID is for the ID of the typelib if this project is exposed to COM
<Assembly: Guid("012A0A70-FE9C-44C5-8696-53BCAA1AAB9F")>

' Version information for an assembly consists of the following four values:
'
'     Major Version
'     Minor Version
'     Build Number
'     Revision
'
' You can specify all the values or you can default the Build and Revision Numbers
' by using the '*' as shown below:

<Assembly: AssemblyVersion("1.0.*")>

'-----
' <autogenerated>
'   This code was generated by Measurement Studio.
'   Runtime Version: 1.1.4322.573
'
'   Changes to this file may cause incorrect behavior and will be lost if
'   the code is regenerated.
' </autogenerated>
'-----

Option Strict Off
Option Explicit On

Imports NationalInstruments.DAQmx
Imports System

Public Class DAQmxTask1
    Inherits Task

    Public Sub New()
        MyBase.New("DAQmxTask1")
        Me.Configure
    End Sub

    Public Overridable Sub Configure()
        Me.AOChannels.CreateVoltageChannel("Dev1/ao0", "VoltageOut", -1, 1, AOVoltageUnits.Volts)
        Me.Timing.ConfigureSampleClock("", 1000, SampleClockActiveEdge.Rising, SampleQuantityMode.FiniteSamples, 2)
    End Sub
End Class

'-----
' <autogenerated>
'   This code was generated by Measurement Studio.
'   Runtime Version: 1.1.4322.573
'
'   Changes to this file may cause incorrect behavior and will be lost if
'   the code is regenerated.
' </autogenerated>
'-----

Option Strict Off
Option Explicit On

Imports NationalInstruments.DAQmx
Imports System
```

```

Public Class DAQmxTask2
    Inherits Task

    Public Sub New()
        MyBase.New("DAQmxTask2")
        Me.Configure
    End Sub

    Public Overridable Sub Configure()
        Me.AOChannels.CreateVoltageChannel("Dev1/ao0", "VoltageOut0", -1, 1, AOVoltageUnits.Volts)
        Me.AOChannels.CreateVoltageChannel("Dev1/ao1", "VoltageOut1", -1, 6, AOVoltageUnits.Volts)
        Me.Timing.ConfigureSampleClock("", 2000, SampleClockActiveEdge.Rising, SampleQuantityMode.ContinuousSamples)
    End Sub
End Class

'-----
' <autogenerated>
'   This code was generated by Measurement Studio.
'   Runtime Version: 1.1.4322.573
'
'   Changes to this file may cause incorrect behavior and will be lost if
'   the code is regenerated.
' </autogenerated>
'-----

Option Strict Off
Option Explicit On

Imports NationalInstruments.DAQmx
Imports System

Public Class DAQmxTask3
    Inherits Task

    Public Sub New()
        MyBase.New("DAQmxTask3")
        Me.Configure
    End Sub

    Public Overridable Sub Configure()
        Me.AIChannels.CreateVoltageChannel("Dev1/ai0", "Current", AITerminalConfiguration.Rse, -10, 10,
        AIVoltageUnits.Volts)
        Me.Timing.ConfigureSampleClock("", 4000, SampleClockActiveEdge.Rising, SampleQuantityMode.FiniteSamples, 1000)
        Me.Triggers.ReferenceTrigger.ConfigureDigitalEdgeTrigger("PFI1", DigitalEdgeReferenceTriggerEdge.Rising, 100)
    End Sub
End Class

'-----
' <autogenerated>
'   This code was generated by Measurement Studio.
'   Runtime Version: 1.1.4322.573
'
'   Changes to this file may cause incorrect behavior and will be lost if
'   the code is regenerated.
' </autogenerated>
'-----

Option Strict Off
Option Explicit On

Imports NationalInstruments.DAQmx
Imports System

Public Class DAQmxTask4
    Inherits Task

    Public Sub New()
        MyBase.New("DAQmxTask4")
        Me.Configure
    End Sub

    Public Overridable Sub Configure()
        Me.AIChannels.CreateVoltageChannel("Dev1/ai0", "Voltage0", AITerminalConfiguration.Rse, -10, 10,
        AIVoltageUnits.Volts)
        Me.AIChannels.CreateVoltageChannel("Dev1/ai1", "Voltage1", AITerminalConfiguration.Rse, 0.5, 500, "GainTable")
        Me.Timing.ConfigureSampleClock("", 100000, SampleClockActiveEdge.Rising, SampleQuantityMode.FiniteSamples, 10000)
        Me.Triggers.ReferenceTrigger.ConfigureAnalogEdgeTrigger("PFI0", AnalogEdgeReferenceTriggerSlope.Rising, 2.5, 2500)
    End Sub
End Class

'-----
' <autogenerated>
'   This code was generated by Measurement Studio.
'   Runtime Version: 1.1.4322.573
'
'   Changes to this file may cause incorrect behavior and will be lost if

```

```

'      the code is regenerated.
' </autogenerated>
'-----

Option Strict Off
Option Explicit On

Imports NationalInstruments.DAQmx
Imports System

Public Class DAQmxTask5
    Inherits Task

    Public Sub New()
        MyBase.New("DAQmxTask5")
        Me.Configure
    End Sub

    Public Overridable Sub Configure()
        Me.AIChannels.CreateVoltageChannel("Dev1/ai1", "Voltage", AITerminalConfiguration.Rse, 0.5, 500, "GainTable")
        Me.AIChannels.CreateVoltageChannel("Dev1/ai3", "Voltage0", AITerminalConfiguration.Rse, 1, 100, "freqTable")
        Me.AIChannels.CreateVoltageChannel("Dev1/ai5", "Voltage1", AITerminalConfiguration.Rse, -100, 100, "capacitance")
        Me.Timing.ConfigureSampleClock("", 1000, SampleClockActiveEdge.Rising, SampleQuantityMode.FiniteSamples, 2)
    End Sub
End Class

'-----
' <autogenerated>
'   This code was generated by Measurement Studio.
'   Runtime Version: 1.1.4322.573
'
'   Changes to this file may cause incorrect behavior and will be lost if
'   the code is regenerated.
' </autogenerated>
'-----

Option Strict Off
Option Explicit On

Imports NationalInstruments.DAQmx
Imports System

Public Class DAQmxTask6
    Inherits Task

    Public Sub New()
        MyBase.New("DAQmxTask6")
        Me.Configure
    End Sub

    Public Overridable Sub Configure()
        Dim ch As DOChannel = Me.DOChannels.CreateChannel("Dev1/port0", "DigitalOut",
ChannelLineGrouping.OneChannelForAllLines)
        ch.InvertLines = false
    End Sub
End Class

'-----
' <autogenerated>
'   This code was generated by Measurement Studio.
'   Runtime Version: 1.1.4322.573
'
'   Changes to this file may cause incorrect behavior and will be lost if
'   the code is regenerated.
' </autogenerated>
'-----

Option Strict Off
Option Explicit On

Imports NationalInstruments.DAQmx
Imports System

Public Class DAQmxTask7
    Inherits Task

    Public Sub New()
        MyBase.New("DAQmxTask7")
        Me.Configure
    End Sub

    Public Overridable Sub Configure()
        Me.AIChannels.CreateVoltageChannel("Dev1/ai0", "Voltage0", AITerminalConfiguration.Rse, -1, 5, AIVoltageUnits.Volts)
        Me.AIChannels.CreateVoltageChannel("Dev1/ai2", "Voltage1", AITerminalConfiguration.Rse, -1, 10, "laser1")
        Me.AIChannels.CreateVoltageChannel("Dev1/ai4", "Voltage2", AITerminalConfiguration.Rse, 0.76, 10, "laser2table")
        Me.Timing.ConfigureSampleClock("", 4000, SampleClockActiveEdge.Rising, SampleQuantityMode.ContinuousSamples)
        Me.Triggers.ReferenceTrigger.ConfigureDigitalEdgeTrigger("PFI0", DigitalEdgeReferenceTriggerEdge.Rising, 100)
    End Sub
End Class

```

```

End Sub
End Class

Imports System
Imports System.Collections
Imports System.ComponentModel
Imports System.Drawing
Imports System.Data
Imports System.Windows.Forms

Public Class DAQmxUserControl1
    Inherits System.Windows.Forms.UserControl
    Private components As System.ComponentModel.IContainer = Nothing
    Private daqmxTask As DAQmxTask1
    Private daqmxWriter As NationalInstruments.DAQmx.AnalogMultiChannelWriter

    Public Sub New()
        ' This call is required by the Windows.Forms Form Designer.
        InitializeComponent()
    End Sub 'New

    Public Sub New(ByVal initializeUI As Boolean)
        If initializeUI Then
            InitializeComponent()
        End If
        CreateTask()
    End Sub 'New

    Public Sub New(ByVal container As IContainer)
        Me.New()
        container.Add(Me)
        CreateTask()
    End Sub 'New

    Public Sub CreateTask()
        If Not (daqmxTask Is Nothing) Then
            Return
        End If
        daqmxTask = New DAQmxTask1
        daqmxWriter = New NationalInstruments.DAQmx.AnalogMultiChannelWriter(daqmxTask.Stream)
    End Sub 'CreateTask

    Protected Overloads Overrides Sub Dispose(ByVal disposing As Boolean)
        If disposing Then
            If Not (components Is Nothing) Then
                components.Dispose()
            End If
            If Not (daqmxTask Is Nothing) Then
                daqmxTask.Stop()
                daqmxTask.Dispose()
            End If
        End If
        daqmxTask = Nothing
        MyBase.Dispose(disposing)
    End Sub 'Dispose

    Private Sub InitializeComponent()
        '
        'DAQmxUserControl1
        '
        Me.Name = "DAQmxUserControl1"
        Me.Size = New System.Drawing.Size(16, 16)
    End Sub 'InitializeComponent

    Private Sub DAQmxUserControl1_Resize(ByVal sender As Object, ByVal e As System.EventArgs) Handles MyBase.Resize
        'waveformGraph1.Size = Me.Size
    End Sub 'DAQmxUserControl1_Resize

    Public Sub GenerateData(ByVal vm As Integer)
        If Not (daqmxTask Is Nothing) Then
            daqmxTask.Control(NationalInstruments.DAQmx.TaskAction.Verify)
            Dim numberOfSamples As Integer = 2
            Dim numChannels As Integer = daqmxTask.AOChannels.Count
            Dim data(numChannels - 1, numberOfSamples - 1) As Double
            Dim i As Integer
            For i = 0 To numChannels - 1
                'Dim channel As NationalInstruments.DAQmx.AOChannel = daqmxTask.AOChannels(i)
                'Dim slope As Double = (channel.Maximum - channel.Minimum) / numberOfSamples
            Dim j As Integer

```



```

        For j = 0 To numberOfSamples - 1
            data(i, j) = vm / 100
        Next j
    Next i
    Write(data)
End If
End Sub 'GenerateData

Public Sub Write(ByVal data(,) As Double)
    'If Not (waveformGraph1 Is Nothing) Then
    'waveformGraph1.PlotYAppendMultiple(data)
    'End If
    daqmxWriter.WriteMultiSample(True, data)
    daqmxTask.WaitUntilDone(-1)
    daqmxTask.Stop()
End Sub 'Write
End Class 'DAQmxUserControl1

Imports System
Imports System.Collections
Imports System.ComponentModel
Imports System.Drawing
Imports System.Data
Imports System.Windows.Forms

Public Class DAQmxUserControl2
    Inherits System.Windows.Forms.UserControl
    Private components As System.ComponentModel.IContainer = Nothing
    Private daqmxTask As DAQmxTask2
    Private daqmxWriter As NationalInstruments.DAQmx.AnalogMultiChannelWriter

    Public Sub New()
        ' This call is required by the Windows.Forms Form Designer.
        InitializeComponent()
    End Sub 'New

    Public Sub New(ByVal initializeUI As Boolean)
        If initializeUI Then
            InitializeComponent()
        End If
        CreateTask()
    End Sub 'New

    Public Sub New(ByVal container As IContainer)
        Me.New()
        container.Add(Me)
        'CreateTask()
    End Sub 'New

    Public Sub CreateTask()
        If Not (daqmxTask Is Nothing) Then
            Return
        End If
        daqmxTask = New DAQmxTask2

        daqmxWriter = New NationalInstruments.DAQmx.AnalogMultiChannelWriter(daqmxTask.Stream)
    End Sub 'CreateTask

    Protected Overloads Overrides Sub Dispose(ByVal disposing As Boolean)
        If disposing Then
            If Not (components Is Nothing) Then
                components.Dispose()
            End If
            If Not (daqmxTask Is Nothing) Then
                daqmxTask.Stop()
                daqmxTask.Dispose()
            End If
        End If
        MyBase.Dispose(disposing)
    End Sub 'Dispose

    Private Sub InitializeComponent()
        '
        'DAQmxUserControl2
        '
        Me.Name = "DAQmxUserControl2"
        Me.Size = New System.Drawing.Size(40, 24)
    End Sub 'InitializeComponent

    Private Sub DAQmxUserControl2_Resize(ByVal sender As Object, ByVal e As System.EventArgs) Handles MyBase.Resize
        'waveformGraph1.Size = Me.Size

```

```

End Sub 'DAQmxUserControl2_Resize

Public Sub GenerateData(ByVal vm As Integer, ByVal pulse As Integer)
    If Not (daqmxTask Is Nothing) Then
        daqmxTask.Stop()
        daqmxTask.Control(NationalInstruments.DAQmx.TaskAction.Verify)
        Dim numberOfSamples As Integer = 200
        Dim numChannels As Integer = daqmxTask.AOChannels.Count
        Dim data(numChannels - 1, numberOfSamples - 1) As Double
        'Dim i As Integer
        'For i = 0 To numChannels - 1
        'Dim channel As NationalInstruments.DAQmx.AOChannel = daqmxTask.AOChannels(i)
        'Dim slope As Double = (channel.Maximum - channel.Minimum) / numberOfSamples
        Dim j As Integer
        For j = 0 To 49
            data(0, j) = vm / 100
            data(1, j) = 0
        Next j
        For j = 50 To 149
            data(0, j) = (vm / 100) + (pulse / 100)
            data(1, j) = 5
        Next j
        For j = 150 To 199
            data(0, j) = vm / 100
            data(1, j) = 0
        Next j
        'Next i
        Write(data)
    End If
End Sub 'GenerateData

Public Sub Write(ByVal data(,) As Double)
    'If Not (waveformGraph1 Is Nothing) Then
    'waveformGraph1.PlotYMultiple(data)
    'End If
    daqmxWriter.BeginWriteMultiSample(True, data, Nothing, Nothing)
End Sub 'Write
End Class 'DAQmxUserControl2

Imports System
Imports System.Collections
Imports System.ComponentModel
Imports System.Drawing
Imports System.Data
Imports System.Windows.Forms

Public Class DAQmxUserControl3
    Inherits System.Windows.Forms.UserControl

    Private Class TaskEventArgs
        Inherits System.EventArgs
        Private data(,) As Double

        Public Sub New(ByVal d(,) As Double)
            data = d
        End Sub 'New

        Public Function GetData() As Double(,)
            Return data
        End Function
    End Class 'TaskEventArgs
End Class 'TaskEventArgs

Delegate Sub EventHandler(ByVal sender As Object, ByVal args As TaskEventArgs)

Public Event DataReady As EventHandler

Private components As System.ComponentModel.IContainer = Nothing

Private daqmxTask As DAQmxTask3
Private daqmxReader As NationalInstruments.DAQmx.AnalogMultiChannelReader
Private dataReadyHandler As System.AsyncCallback
Private stopped As Boolean = True
Private fileNumber As Integer
Private fileNumberFDU1 As Integer
Private fileNumberFDU2 As Integer
Private fileNumberlaserleft As Integer
Private fileNumberlaserright As Integer
Private FDU As Boolean
Private laseroutput As Boolean
Private samplingRate As Integer

Private currentGain As Double
Public pForm As Form2

```

```

Public Sub New()
    ' This call is required by the Windows.Forms Form Designer.
    InitializeComponent()
    AddHandler Me.DataReady, AddressOf Me.OnDataReady
End Sub 'New

Public Sub New(ByVal initializeUI As Boolean)
    If initializeUI Then
        ' This call is required by the Windows.Forms Form Designer.
        InitializeComponent()
    End If
    AddHandler Me.DataReady, AddressOf Me.OnDataReady
    CreateTask()
End Sub

Public Sub New(ByVal container As IContainer)
    Me.New()
    container.Add(Me)
    CreateTask()
End Sub 'New

Private Sub CreateTask()
    If Not (daqmxTask Is Nothing) Then
        Return
    End If
    If dataReadyHandler Is Nothing Then
        dataReadyHandler = New System.AsyncCallback(AddressOf DataReadyEventHandler)
    End If
    daqmxTask = New DAQmxTask3()
    daqmxReader = New NationalInstruments.DAQmx.AnalogMultiChannelReader(daqmxTask.Stream)
    daqmxReader.SynchronizingObject = Me
End Sub 'CreateTask

Protected Overloads Overrides Sub Dispose(ByVal disposing As Boolean)
    If disposing Then
        If Not (components Is Nothing) Then
            components.Dispose()
        End If
        If Not (daqmxTask Is Nothing) Then
            StopTask()
            daqmxTask.Dispose()
        End If
    End If
    MyBase.Dispose(disposing)
End Sub 'Dispose

Public WithEvents ScatterGraph1 As NationalInstruments.UI.WindowsForms.ScatterGraph
Friend WithEvents XAxis1 As NationalInstruments.UI.XAxis
Public WithEvents YAxis1 As NationalInstruments.UI.YAxis
Public WithEvents ScatterPlot1 As NationalInstruments.UI.ScatterPlot
Friend WithEvents Label1 As System.Windows.Forms.Label
Friend WithEvents Label2 As System.Windows.Forms.Label
Public WithEvents TextBox1 As System.Windows.Forms.TextBox
Public WithEvents TextBox2 As System.Windows.Forms.TextBox

Private Sub InitializeComponent()
    Me.ScatterGraph1 = New NationalInstruments.UI.WindowsForms.ScatterGraph
    Me.ScatterPlot1 = New NationalInstruments.UI.ScatterPlot
    Me.XAxis1 = New NationalInstruments.UI.XAxis
    Me.YAxis1 = New NationalInstruments.UI.YAxis
    Me.Label1 = New System.Windows.Forms.Label
    Me.Label2 = New System.Windows.Forms.Label
    Me.TextBox1 = New System.Windows.Forms.TextBox
    Me.TextBox2 = New System.Windows.Forms.TextBox
    CType(Me.ScatterGraph1, System.ComponentModel.ISupportInitialize).BeginInit()
    Me.SuspendLayout()
    '
    'ScatterGraph1
    '
    Me.ScatterGraph1.Location = New System.Drawing.Point(0, 0)
    Me.ScatterGraph1.Name = "ScatterGraph1"
    Me.ScatterGraph1.Plots.AddRange(New NationalInstruments.UI.ScatterPlot() {Me.ScatterPlot1})
    Me.ScatterGraph1.Size = New System.Drawing.Size(896, 344)
    Me.ScatterGraph1.TabIndex = 0
    Me.ScatterGraph1.XAxes.AddRange(New NationalInstruments.UI.XAxis() {Me.XAxis1})
    Me.ScatterGraph1.YAxes.AddRange(New NationalInstruments.UI.YAxis() {Me.YAxis1})
    '
    'ScatterPlot1
    '
    Me.ScatterPlot1.XAxis = Me.XAxis1
    Me.ScatterPlot1.YAxis = Me.YAxis1
    '
    'XAxis1
    '
    Me.XAxis1.Caption = "msec"
    Me.XAxis1.Mode = NationalInstruments.UI.AxisMode.AutoScaleExact
    '

```

```

'YAxis1
'
Me.YAxis1.Caption = "pA"
Me.YAxis1.Mode = NationalInstruments.UI.AxisMode.Fixed
'
'Label1
'
Me.Label1.Location = New System.Drawing.Point(0, 344)
Me.Label1.Name = "Label1"
Me.Label1.Size = New System.Drawing.Size(64, 16)
Me.Label1.TabIndex = 1
Me.Label1.Text = "DateStamp"
'
'Label2
'
Me.Label2.Location = New System.Drawing.Point(728, 344)
Me.Label2.Name = "Label2"
Me.Label2.Size = New System.Drawing.Size(32, 16)
Me.Label2.TabIndex = 2
Me.Label2.Text = "Gain"
'
'TextBox1
'
Me.TextBox1.Location = New System.Drawing.Point(64, 344)

Me.TextBox1.Name = "TextBox1"
Me.TextBox1.ReadOnly = True
Me.TextBox1.Size = New System.Drawing.Size(608, 20)
Me.TextBox1.TabIndex = 3
Me.TextBox1.Text = ""
'
'TextBox2
'
Me.TextBox2.Location = New System.Drawing.Point(760, 344)
Me.TextBox2.Name = "TextBox2"
Me.TextBox2.ReadOnly = True
Me.TextBox2.Size = New System.Drawing.Size(80, 20)
Me.TextBox2.TabIndex = 4
Me.TextBox2.Text = ""
'
'DAQmxUserControl3
'
Me.Controls.Add(Me.TextBox2)
Me.Controls.Add(Me.TextBox1)
Me.Controls.Add(Me.Label2)
Me.Controls.Add(Me.Label1)
Me.Controls.Add(Me.ScatterGraph1)
Me.Name = "DAQmxUserControl3"
Me.Size = New System.Drawing.Size(904, 368)
CType(Me.ScatterGraph1, System.ComponentModel.ISupportInitialize).EndInit()
Me.ResumeLayout(False)

End Sub 'InitializeComponent

Public Sub StartTask(ByVal samples As Integer)
    'If Not stopped Then
    'Return
    'End If
    'If daqmxTask Is Nothing Then
    'CreateTask()
    'End If
    daqmxTask.Stream.Timeout = -1           'NO TIMEOUTS!!!
    daqmxTask.Start()
    stopped = False
    daqmxReader.BeginReadMultiSample(samples, dataReadyHandler, Nothing)
End Sub 'StartTask

Public Sub StopTask()
    If Not (Me.daqmxTask Is Nothing) Then
        Me.daqmxTask.Stop()
    End If
    stopped = True
End Sub 'StopTask

Private Sub DAQmxUserControl3_Resize(ByVal sender As Object, ByVal e As System.EventArgs) Handles MyBase.Resize
    'WaveformGraph1.Size = Me.Size
End Sub 'DAQmxUserControl3_Resize

Public Sub DataReadyEventHandler(ByVal result As IAsyncResult)
    If stopped Then
        Return
    End If
    Dim data(,) As Double = daqmxReader.EndReadMultiSample(result)
    Dim args As New TaskEventArgs(data)
    RaiseEvent DataReady(Me, args)
End Sub 'DataReadyEventHandler

```

```

Public Overridable Sub OnDataReady(ByVal sender As Object, ByVal e As TaskEventArgs)
    Dim data(,) As Double = e.GetData()
    Dim sampleNumber As Integer
    sampleNumber = data.GetLength(1)
    Dim xData(sampleNumber - 1) As Double
    Dim yData(sampleNumber - 1) As Double
    Dim fdu1data(sampleNumber - 1) As Double
    Dim fdu2data(sampleNumber - 1) As Double

    Dim i As Integer
    For i = 0 To sampleNumber - 1
        xData(i) = (1000 * i) / samplingRate 'change sample# to msec
        yData(i) = data(0, i) * 1000 / currentGain 'convert to pA; adjust for gain
    Next i
    Me.YAxis1.Range = New NationalInstruments.UI.Range(-10000 / currentGain, 10000 / currentGain)
    Me.ScatterPlot1.PlotXY(xData, yData)
    Me.TextBox2.Text = CStr(currentGain)

    'Write data to file and close file
    WriteLine(fileName, currentGain) 'write gain
    WriteLine(fileName, sampleNumber) 'write number of samples
    For i = 0 To sampleNumber - 1
        WriteLine(fileName, yData(i)) 'write data
    Next i
    FileClose(fileName)

    Dim channelnumber As Integer
    If FDU = True Then
        If laseroutput Then
            channelnumber = 3
        Else
            channelnumber = 1
        End If
        WriteLine(fileNameFDU1, 0) 'write gain (placeholder)
        WriteLine(fileNameFDU2, 0) 'write gain (placeholder)
        For i = 0 To sampleNumber - 1
            fdu1data(i) = data(channelnumber, i)
            fdu2data(i) = data(channelnumber + 1, i)
            WriteLine(fileNameFDU1, data(channelnumber, i))
            WriteLine(fileNameFDU2, data(channelnumber + 1, i))
        Next i
        FileClose(fileNameFDU1)
        FileClose(fileNameFDU2)

        Me.pForm.ScatterGraph2.PlotXY(xData, fdu1data)
        Me.pForm.ScatterGraph3.PlotXY(xData, fdu2data)
    End If

    If laseroutput Then
        WriteLine(fileNameLaserLeft, 0) 'write gain (placeholder)
        WriteLine(fileNameLaserRight, 0) 'write gain (placeholder)
        For i = 0 To sampleNumber - 1
            WriteLine(fileNameLaserLeft, data(1, i))
            WriteLine(fileNameLaserRight, data(2, i))
        Next i
        FileClose(fileNameLaserLeft)
        FileClose(fileNameLaserRight)
    End If

    Me.daqmxTask.Stop()
    Me.pForm.Button4.Enabled = True

    'If Me.pForm.flashAgain = True Then
    'Me.pForm.currentFlash50form.PassData(xData, yData)
    'End If

    'Erase data, xData, yData, fdu1data, fdu2data 'Deallocate memory

    If Me.pForm.flashAgain = True Then
        'Me.pForm.Button15.PerformClick()
        Me.pForm.currentFlash50form.PassData(xData, yData)
    End If

    'If Not stopped Then
    'daqmxReader.BeginReadMultiSample(-1, dataReadyHandler, Nothing)
    'End If
End Sub 'OnDataReady

Public Sub Configure(ByVal sampleRate As Integer, ByVal numSamples As Integer, ByVal Laser As String, ByVal fileID As Integer, ByVal preTriggerSamples As Integer, ByVal gain As Double, ByVal fdu1 As Integer, ByVal fdu2 As Integer, ByVal laserleft As Integer, ByVal laserright As Integer)
    Me.daqmxTask.Dispose()
    Me.daqmxTask = Nothing
    Me.dataReadyHandler = Nothing
    Me.daqmxReader = Nothing
    Me.CreateTask()

```

```

'configure trigger
If Laser = "Right" Then
    Laser = "PFI2" 'trigger on right laser
ElseIf Laser = "Internal" Then
    Laser = "PFI3" 'internal trigger
Else
    Laser = "PFI1" 'default trigger on left laser
End If

'MsgBox(Laser)

FDU = False
laseroutput = False
If Me.pForm.CheckBox3.Checked = True Then
    laseroutput = True
    filenumberlaserleft = laserleft
    filenumberlaserright = laserright
    Me.daqmxTask.AIChannels.CreateVoltageChannel("Dev1/ai2", "LaserLeft", AITerminalConfiguration.Rse, -1, 10,
"laser1")
    Me.daqmxTask.AIChannels.CreateVoltageChannel("Dev1/ai4", "LaserRight", AITerminalConfiguration.Rse, 0.76, 10,
"laser2table")
    WriteLine(filenumberlaserleft, Me.pForm.NumericUpDown2.Value)
    WriteLine(filenumberlaserleft, numSamples)
    WriteLine(filenumberlaserleft, preTriggerSamples)
    WriteLine(filenumberlaserright, Me.pForm.NumericUpDown2.Value)
    WriteLine(filenumberlaserright, numSamples)
    WriteLine(filenumberlaserright, preTriggerSamples)

End If

If Me.pForm.CheckBox4.Checked = True Then
    FDU = True
    fileNumberFDU1 = fdu1
    fileNumberFDU2 = fdu2
    Me.daqmxTask.AIChannels.CreateVoltageChannel("Dev1/ai6", "FDU1", AITerminalConfiguration.Rse, -10, 10,
AIVoltageUnits.Volts)
    Me.daqmxTask.AIChannels.CreateVoltageChannel("Dev1/ai7", "FDU2", AITerminalConfiguration.Rse, -10, 10,
AIVoltageUnits.Volts)
    WriteLine(fileNumberFDU1, Me.pForm.NumericUpDown2.Value)
    WriteLine(fileNumberFDU1, numSamples)
    WriteLine(fileNumberFDU1, preTriggerSamples)
    WriteLine(fileNumberFDU2, Me.pForm.NumericUpDown2.Value)
    WriteLine(fileNumberFDU2, numSamples)
    WriteLine(fileNumberFDU2, preTriggerSamples)
End If

Me.daqmxTask.Timing.ConfigureSampleClock("", sampleRate, SampleClockActiveEdge.Rising,
SampleQuantityMode.FiniteSamples, numSamples)
Me.samplingRate = sampleRate
Me.daqmxTask.Triggers.ReferenceTrigger.ConfigureDigitalEdgeTrigger(Laser, DigitalEdgeReferenceTriggerEdge.Rising,
preTriggerSamples)
'MsgBox(Me.daqmxTask.Stream.ChannelsToRead())

fileNumber = fileID
currentGain = gain
WriteLine(fileNumber, Me.pForm.NumericUpDown2.Value) 'write sampling rate
WriteLine(fileNumber, numSamples) 'write total number of samples
WriteLine(fileNumber, preTriggerSamples) 'write number of preTrigger samples
Me.ScatterGraph1.ClearData() 'clear graphs
Me.pForm.ScatterGraph2.ClearData()
Me.pForm.ScatterGraph3.ClearData()
'Me.TextBox1.Text = ""
'Me.TextBox2.Text = ""
End Sub
End Class 'DAQmxUserControl3

Imports System
Imports System.Collections
Imports System.ComponentModel
Imports System.Drawing
Imports System.Data
Imports System.Windows.Forms

Public Class DAQmxUserControl4
    Inherits System.Windows.Forms.UserControl

    Public Class TaskEventArgs
        Inherits System.EventArgs
        Private data(,) As Double

    Public Sub New(ByVal d(,) As Double)
        data = d
    End Sub 'New

    Public Function GetData() As Double(,)

```

```

        Return data
    End Function
End Class 'TaskEventArgs

Delegate Sub EventHandler(ByVal sender As Object, ByVal args As TaskEventArgs)

Public Event DataReady As EventHandler

Private components As System.ComponentModel.IContainer = Nothing

Private daqmxTask As DAQmxTask4
Private daqmxReader As NationalInstruments.DAQmx.AnalogMultiChannelReader
Private dataReadyHandler As System.AsyncCallback
Private stopped As Boolean = True
Public pulseForm As Form3

Public Sub New()
    ' This call is required by the Windows.Forms Form Designer.
    InitializeComponent()
    AddHandler Me.DataReady, AddressOf Me.OnDataReady
End Sub 'New

Public Sub New(ByVal initializeUI As Boolean)
    If initializeUI Then
        ' This call is required by the Windows.Forms Form Designer.
        InitializeComponent()
    End If
    AddHandler Me.DataReady, AddressOf Me.OnDataReady
    CreateTask()
End Sub

Public Sub New(ByVal container As IContainer)
    Me.New()
    container.Add(Me)
    CreateTask()
End Sub 'New

Private Sub CreateTask()
    If Not (daqmxTask Is Nothing) Then
        Return
    End If
    If dataReadyHandler Is Nothing Then
        dataReadyHandler = New System.AsyncCallback(AddressOf DataReadyEventHandler)
    End If
    daqmxTask = New DAQmxTask4()
    daqmxReader = New NationalInstruments.DAQmx.AnalogMultiChannelReader(daqmxTask.Stream)

    daqmxReader.SynchronizingObject = Me
End Sub 'CreateTask

Protected Overrides Sub Dispose(ByVal disposing As Boolean)
    If disposing Then
        If Not (components Is Nothing) Then
            components.Dispose()
        End If
        If Not (daqmxTask Is Nothing) Then
            StopTask()
            daqmxTask.Dispose()
        End If
    End If
    MyBase.Dispose(disposing)
End Sub 'Dispose

Friend WithEvents ScatterGraph1 As NationalInstruments.UI.WindowsForms.ScatterGraph
Friend WithEvents XAxis2 As NationalInstruments.UI.XAxis
Friend WithEvents YAxis2 As NationalInstruments.UI.YAxis
Friend WithEvents ScatterPlot1 As NationalInstruments.UI.ScatterPlot
Friend WithEvents Label1 As System.Windows.Forms.Label
Friend WithEvents TextBox1 As System.Windows.Forms.TextBox
Friend WithEvents Label2 As System.Windows.Forms.Label
Friend WithEvents TextBox2 As System.Windows.Forms.TextBox
Friend WithEvents Button1 As System.Windows.Forms.Button
Friend WithEvents TextBox3 As System.Windows.Forms.TextBox
Friend WithEvents CheckBox1 As System.Windows.Forms.CheckBox
Friend WithEvents CheckBox2 As System.Windows.Forms.CheckBox
Friend WithEvents CheckBox3 As System.Windows.Forms.CheckBox
Friend WithEvents CheckBox4 As System.Windows.Forms.CheckBox

Private Sub InitializeComponent()
    Me.ScatterGraph1 = New NationalInstruments.UI.WindowsForms.ScatterGraph
    Me.ScatterPlot1 = New NationalInstruments.UI.ScatterPlot
    Me.XAxis2 = New NationalInstruments.UI.XAxis
    Me.YAxis2 = New NationalInstruments.UI.YAxis
    Me.Label1 = New System.Windows.Forms.Label
    Me.TextBox1 = New System.Windows.Forms.TextBox

```

```

Me.Label2 = New System.Windows.Forms.Label
Me.TextBox2 = New System.Windows.Forms.TextBox
Me.Button1 = New System.Windows.Forms.Button
Me.TextBox3 = New System.Windows.Forms.TextBox
Me.CheckBox1 = New System.Windows.Forms.CheckBox
Me.CheckBox2 = New System.Windows.Forms.CheckBox
Me.CheckBox3 = New System.Windows.Forms.CheckBox
Me.CheckBox4 = New System.Windows.Forms.CheckBox
CType(Me.ScatterGraph1, System.ComponentModel.ISupportInitialize).BeginInit()
Me.SuspendLayout()
'
'ScatterGraph1
'
Me.ScatterGraph1.Location = New System.Drawing.Point(0, 0)
Me.ScatterGraph1.Name = "ScatterGraph1"
Me.ScatterGraph1.Plots.AddRange(New NationalInstruments.UI.ScatterPlot() {Me.ScatterPlot1})
Me.ScatterGraph1.Size = New System.Drawing.Size(696, 480)
Me.ScatterGraph1.TabIndex = 1
Me.ScatterGraph1.XAxes.AddRange(New NationalInstruments.UI.XAxis() {Me.XAxis2})
Me.ScatterGraph1.YAxes.AddRange(New NationalInstruments.UI.YAxis() {Me.YAxis2})
'
'ScatterPlot1
'
Me.ScatterPlot1.XAxis = Me.XAxis2
Me.ScatterPlot1.YAxis = Me.YAxis2
'
'XAxis2
'
Me.XAxis2.Caption = "ms"
Me.XAxis2.Mode = NationalInstruments.UI.AxisMode.ScopeChart
Me.XAxis2.Range = New NationalInstruments.UI.Range(0, 100)
'
'YAxis2
'
Me.YAxis2.Caption = "pA"
Me.YAxis2.Mode = NationalInstruments.UI.AxisMode.Fixed
'
'Label1
'
Me.Label1.Location = New System.Drawing.Point(8, 480)
Me.Label1.Name = "Label1"
Me.Label1.Size = New System.Drawing.Size(32, 16)
Me.Label1.TabIndex = 2
Me.Label1.Text = "Gain:"
'
'TextBox1
'
Me.TextBox1.Location = New System.Drawing.Point(48, 480)
Me.TextBox1.Name = "TextBox1"
Me.TextBox1.ReadOnly = True
Me.TextBox1.Size = New System.Drawing.Size(48, 20)
Me.TextBox1.TabIndex = 3
Me.TextBox1.Text = ""
'
'Label2
'
Me.Label2.Location = New System.Drawing.Point(384, 480)
Me.Label2.Name = "Label2"
Me.Label2.Size = New System.Drawing.Size(136, 16)
Me.Label2.TabIndex = 4
Me.Label2.Text = "Seal Resistance (GOhms)"
'
'TextBox2
'
Me.TextBox2.Location = New System.Drawing.Point(528, 480)
Me.TextBox2.Name = "TextBox2"
Me.TextBox2.ReadOnly = True
Me.TextBox2.Size = New System.Drawing.Size(160, 20)
Me.TextBox2.TabIndex = 5
Me.TextBox2.Text = ""
'
'Button1
'
Me.Button1.Location = New System.Drawing.Point(376, 504)
Me.Button1.Name = "Button1"
Me.Button1.Size = New System.Drawing.Size(144, 24)
Me.Button1.TabIndex = 6
Me.Button1.Text = "DATA FOLDER"
'
'TextBox3
'
Me.TextBox3.Location = New System.Drawing.Point(528, 504)
Me.TextBox3.Name = "TextBox3"
Me.TextBox3.ReadOnly = True
Me.TextBox3.Size = New System.Drawing.Size(152, 20)
Me.TextBox3.TabIndex = 7
Me.TextBox3.Text = ""
'
'CheckBox1
'

```



```

Me.CheckBox1.Enabled = False
Me.CheckBox1.Location = New System.Drawing.Point(8, 504)
Me.CheckBox1.Name = "CheckBox1"
Me.CheckBox1.Size = New System.Drawing.Size(136, 24)
Me.CheckBox1.TabIndex = 8
Me.CheckBox1.Text = "Save Test Pulse???"
'
'CheckBox2
'
Me.CheckBox2.Location = New System.Drawing.Point(152, 480)
Me.CheckBox2.Name = "CheckBox2"
Me.CheckBox2.Size = New System.Drawing.Size(88, 24)
Me.CheckBox2.TabIndex = 9
Me.CheckBox2.Text = "Access Resistance"
'
'CheckBox3
'
Me.CheckBox3.Location = New System.Drawing.Point(248, 480)
Me.CheckBox3.Name = "CheckBox3"
Me.CheckBox3.Size = New System.Drawing.Size(104, 32)
Me.CheckBox3.TabIndex = 10
Me.CheckBox3.Text = "Seal Resistance"
'
'CheckBox4
'
Me.CheckBox4.Location = New System.Drawing.Point(152, 504)
Me.CheckBox4.Name = "CheckBox4"
Me.CheckBox4.Size = New System.Drawing.Size(88, 24)
Me.CheckBox4.TabIndex = 11
Me.CheckBox4.Text = "Whole Cell"
'
'DAQmxUserControl4
'
Me.Controls.Add(Me.CheckBox4)
Me.Controls.Add(Me.CheckBox3)
Me.Controls.Add(Me.CheckBox2)
Me.Controls.Add(Me.CheckBox1)
Me.Controls.Add(Me.TextBox3)
Me.Controls.Add(Me.Button1)
Me.Controls.Add(Me.TextBox2)
Me.Controls.Add(Me.Label2)
Me.Controls.Add(Me.TextBox1)
Me.Controls.Add(Me.Label1)
Me.Controls.Add(Me.ScatterGraph1)
Me.Name = "DAQmxUserControl4"
Me.Size = New System.Drawing.Size(840, 608)
CType(Me.ScatterGraph1, System.ComponentModel.ISupportInitialize).EndInit()
Me.ResumeLayout(False)

End Sub 'InitializeComponent

Public Sub StartTask()
    If Not stopped Then
        Return
    End If
    If daqmxTask Is Nothing Then
        CreateTask()
    End If
    daqmxTask.Start()
    stopped = False
    daqmxReader.BeginReadMultiSample(-1, dataReadyHandler, Nothing)
End Sub 'StartTask

Public Sub StopTask()
    If Not (Me.daqmxTask Is Nothing) Then
        Me.daqmxTask.Stop()
    End If
    stopped = True
End Sub 'StopTask

Private Sub DAQmxUserControl4_Resize(ByVal sender As Object, ByVal e As System.EventArgs) Handles MyBase.Resize
    'waveformGraph1.Size = Me.Size
End Sub 'DAQmxUserControl4_Resize

Public Sub DataReadyEventHandler(ByVal result As IAsyncResult)
    If stopped Then
        Return
    End If
    Dim data(,) As Double = daqmxReader.EndReadMultiSample(result)
    Dim args As New TaskEventArgs(data)
    RaiseEvent DataReady(Me, args)
End Sub 'DataReadyEventHandler

Public Overridable Sub OnDataReady(ByVal sender As Object, ByVal e As TaskEventArgs)

    Dim data(,) As Double = e.GetData()

```

```

Dim ydata(9999) As Double      'array for current data
Dim xdata(9999) As Double      'array for time data
Dim i As Integer
For i = 0 To 9999
    xdata(i) = i / 100          'convert sample number to milliseconds (assuming 2000Hz)
    ydata(i) = data(0, i) * 1000 / data(1, i) 'scale current for gain; convert to pA
Next i
'Calculate seal resistance
Dim preCurrent As Double

Dim postCurrent As Double
Dim resistance As Double
preCurrent = 0
postCurrent = 0
For i = 2301 To 2400
    preCurrent = preCurrent + ydata(i)
Next i
preCurrent = preCurrent / 100
For i = 7300 To 7400
    postCurrent = postCurrent + ydata(i)
Next i
postCurrent = postCurrent / 100
If (preCurrent = postCurrent) Then
    resistance = 9999999
Else
    resistance = CDbl(Me.pulseForm.NumericUpDown2.Value) / (postCurrent - preCurrent)
End If
If Not (ScatterGraph1 Is Nothing) Then
    'adjust range accordingly with gain
    Me.YAxis2.Range = New NationalInstruments.UI.Range(-10000 / data(1, 1), 10000 / data(1, 1))
    ScatterGraph1.PlotXY(xdata, ydata) 'plot data
    Me.TextBox1.Text = CStr(data(1, 1)) 'write gain to screen
    Me.TextBox2.Text = CStr(resistance) 'write seal resistance to screen
End If
If Me.CheckBox1.Checked = True Then
    Dim fileNumber As Integer
    fileNumber = FreeFile()
    Dim DateTime, theday, thetime As String
    theday = DateString
    thetime = TimeString
    DateTime = String.Concat(Me.TextBox3.Text, "\", Mid(theday, 7), Mid(theday, 1, 2), Mid(theday, 4, 2),
Mid(thetime, 1, 2), Mid(thetime, 4, 2), Mid(thetime, 7), "_testpulse.txt")
    FileOpen(fileNumber, DateTime, OpenMode.Output, OpenAccess.Write)
    If Me.CheckBox2.Checked Then
        WriteLine(fileNumber, "Access Resistance")
    End If
    If Me.CheckBox3.Checked Then
        WriteLine(fileNumber, "Seal Resistance")
    End If
    If Me.CheckBox4.Checked Then
        WriteLine(fileNumber, "Whole Cell")
    End If
    WriteLine(fileNumber, Me.pulseForm.NumericUpDown1.Value)
    WriteLine(fileNumber, Me.pulseForm.NumericUpDown2.Value)
    WriteLine(fileNumber, data(1, 1))
    WriteLine(fileNumber, resistance)
    For i = 0 To 9999
        WriteLine(fileNumber, ydata(i))
    Next i
    FileClose(fileNumber)
    MsgBox(DateTime, MsgBoxStyle.OKOnly, "Test Pulse Saved!!!")
    Me.CheckBox1.Checked = False
End If
Me.StopTask()
Me.StartTask()
'If Not stopped Then
'daqmxReader.BeginReadMultiSample(-1, dataReadyHandler, Nothing)
'End If
End Sub 'OnDataReady

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
    Dim dataFolder As New FolderBrowserDialog
    dataFolder.ShowDialog()
    Me.TextBox3.Text = dataFolder.SelectedPath
    Me.CheckBox1.Enabled = True
End Sub
End Class 'DAQmxUserControl4

Imports System
Imports System.Collections
Imports System.ComponentModel
Imports System.Drawing
Imports System.Data
Imports System.Windows.Forms

Public Class DAQmxUserControl5
    Inherits System.Windows.Forms.UserControl
    Private components As System.ComponentModel.IContainer = Nothing
    Private daqmxTask As DAQmxTask5

```

```

    Private daqmxReader As NationalInstruments.DAQmx.AnalogMultiChannelReader

Public Sub New()
    ' This call is required by the Windows.Forms Form Designer.
    InitializeComponent()
End Sub 'New

Public Sub New(ByVal initializeUI As Boolean)
    If initializeUI Then
        InitializeComponent()
    End If
    CreateTask()
End Sub 'New

Public Sub New(ByVal container As IContainer)
    Me.New()
    container.Add(Me)
    CreateTask()
End Sub 'New

Private Sub CreateTask()
    If Not (daqmxTask Is Nothing) Then
        Return
    End If
    daqmxTask = new DAQmxTask5()

    daqmxReader = New NationalInstruments.DAQmx.AnalogMultiChannelReader(daqmxTask.Stream)
End Sub 'CreateTask

Protected Overrides Sub Dispose(ByVal disposing As Boolean)
    If disposing Then
        If Not (components Is Nothing) Then
            components.Dispose()
        End If
        If Not (daqmxTask Is Nothing) Then
            daqmxTask.Stop()
            daqmxTask.Dispose()
        End If
    End If
    MyBase.Dispose(disposing)
End Sub 'Dispose

Friend WithEvents Label1 As System.Windows.Forms.Label
Friend WithEvents Label2 As System.Windows.Forms.Label
Friend WithEvents TextBox1 As System.Windows.Forms.TextBox
Friend WithEvents TextBox2 As System.Windows.Forms.TextBox
Friend WithEvents Label3 As System.Windows.Forms.Label
Friend WithEvents Label4 As System.Windows.Forms.Label

Private Sub InitializeComponent()
    Me.Label1 = New System.Windows.Forms.Label
    Me.Label2 = New System.Windows.Forms.Label
    Me.TextBox1 = New System.Windows.Forms.TextBox
    Me.TextBox2 = New System.Windows.Forms.TextBox
    Me.Label3 = New System.Windows.Forms.Label
    Me.Label4 = New System.Windows.Forms.Label
    Me.SuspendLayout()
    '
    'Label1
    '
    Me.Label1.Location = New System.Drawing.Point(8, 8)
    Me.Label1.Name = "Label1"
    Me.Label1.Size = New System.Drawing.Size(72, 16)
    Me.Label1.TabIndex = 0
    Me.Label1.Text = "Capacitance"
    '
    'Label2
    '
    Me.Label2.Location = New System.Drawing.Point(192, 8)
    Me.Label2.Name = "Label2"
    Me.Label2.Size = New System.Drawing.Size(72, 16)
    Me.Label2.TabIndex = 1
    Me.Label2.Text = "Filtering at..."
    '
    'TextBox1
    '
    Me.TextBox1.Location = New System.Drawing.Point(72, 8)
    Me.TextBox1.Name = "TextBox1"
    Me.TextBox1.ReadOnly = True
    Me.TextBox1.Size = New System.Drawing.Size(72, 20)
    Me.TextBox1.TabIndex = 2
    Me.TextBox1.Text = ""
    '
    'TextBox2
    '

```

```

Me.TextBox2.Location = New System.Drawing.Point(256, 8)
Me.TextBox2.Name = "TextBox2"
Me.TextBox2.ReadOnly = True
Me.TextBox2.Size = New System.Drawing.Size(40, 20)
Me.TextBox2.TabIndex = 3
Me.TextBox2.Text = ""
'
'Label3
'
Me.Label3.Location = New System.Drawing.Point(296, 8)
Me.Label3.Name = "Label3"
Me.Label3.Size = New System.Drawing.Size(24, 16)
Me.Label3.TabIndex = 4
Me.Label3.Text = "kHz"
'
'Label4
'
Me.Label4.Location = New System.Drawing.Point(144, 8)
Me.Label4.Name = "Label4"
Me.Label4.Size = New System.Drawing.Size(24, 16)
Me.Label4.TabIndex = 5
Me.Label4.Text = "pF"
'
'DAQmxUserControl5
'
Me.Controls.Add(Me.Label4)
Me.Controls.Add(Me.Label3)
Me.Controls.Add(Me.TextBox2)
Me.Controls.Add(Me.TextBox1)
Me.Controls.Add(Me.Label2)
Me.Controls.Add(Me.Label1)
Me.Name = "DAQmxUserControl5"
Me.Size = New System.Drawing.Size(320, 32)
Me.ResumeLayout(False)

End Sub 'InitializeComponent

Private Sub DAQmxUserControl5_Resize(ByVal sender As Object, ByVal e As System.EventArgs) Handles MyBase.Resize
    'waveformGraph1.Size = Me.Size
End Sub 'DAQmxUserControl5_Resize

Public Function Read() As Double(,)
    Dim data(,) As Double
    data = daqmxReader.ReadMultiSample(-1)
    'If Not (waveformGraph1 Is Nothing) Then
    'waveformGraph1.PlotYMultiple(data)
    'End If
    Return data
End Function 'Read
End Class 'DAQmxUserControl5

Imports System
Imports System.Collections
Imports System.ComponentModel
Imports System.Drawing
Imports System.Data
Imports System.Windows.Forms

Public Class DAQmxUserControl6
    Inherits System.Windows.Forms.UserControl
    Private components As System.ComponentModel.IContainer = Nothing
    Private daqmxTask As DAQmxTask6
    Private daqmxWriter As NationalInstruments.DAQmx.DigitalSingleChannelWriter

    Public Sub New()
        ' This call is required by the Windows.Forms Form Designer.
        InitializeComponent()
        CreateTask()
    End Sub 'New

    Public Sub New(initializeUI As Boolean)
        If initializeUI Then
            InitializeComponent()
        End If
        CreateTask()
    End Sub 'New

    Public Sub New(container As IContainer)
        Me.New()
        container.Add(Me)
        CreateTask()
    End Sub 'New

```

```

Private Sub CreateTask()
    If Not (daqmxTask Is Nothing) Then
        Return
    End If
    daqmxTask = new DAQmxTask6()
    daqmxWriter = New NationalInstruments.DAQmx.DigitalSingleChannelWriter(daqmxTask.Stream)
End Sub 'CreateTask

Protected Overloads Overrides Sub Dispose(disposing As Boolean)
    If disposing Then
        If Not (components Is Nothing) Then
            components.Dispose()
        End If
        If Not (daqmxTask Is Nothing) Then
            daqmxTask.Stop()
            daqmxTask.Dispose()
        End If
    End If
    MyBase.Dispose(disposing)
End Sub 'Dispose

Private Sub InitializeComponent()
    '
    'DAQmxUserControl6
    '
    Me.Name = "DAQmxUserControl6"
    Me.Size = New System.Drawing.Size(192, 80)
End Sub 'InitializeComponent

Public Sub GenerateData()
    Dim rand As New Random()
    Write(rand.Next(0, 100))
End Sub 'GenerateData

Public Sub Write(data As Integer)
    daqmxWriter.WriteSingleSamplePort(True, data)
End Sub 'Write
End Class 'DAQmxUserControl6

Imports System
Imports System.Collections
Imports System.ComponentModel
Imports System.Drawing
Imports System.Data
Imports System.Windows.Forms

Public Class DAQmxUserControl7
    Inherits System.Windows.Forms.UserControl

    Public Class TaskEventArgs
        Inherits System.EventArgs
        Private data(,) As Double

        Public Sub New(ByVal d(,) As Double)
            data = d
        End Sub 'New

        Public Function GetData() As Double(,)
            Return data
        End Function
    End Class 'TaskEventArgs

    Delegate Sub EventHandler(ByVal sender As Object, ByVal args As TaskEventArgs)

    Public Event DataReady As EventHandler

    Private components As System.ComponentModel.IContainer = Nothing

    Private daqmxTask As DAQmxTask7
    Private daqmxReader As NationalInstruments.DAQmx.AnalogMultiChannelReader
    Private dataReadyHandler As System.AsyncCallback
    Private stopped As Boolean = True
    Private fileNumber As Integer
    Private fileNumberLeft As Integer
    Private fileNumberRight As Integer
    Private samplingRate As Integer
    Private currentGain As Double
    Public pForm As Form2

    Public Sub New()

```

```

' This call is required by the Windows.Forms Form Designer.
InitializeComponent()
AddHandler Me.DataReady, AddressOf Me.OnDataReady
End Sub 'New

Public Sub New(ByVal initializeUI As Boolean)
    If initializeUI Then
        ' This call is required by the Windows.Forms Form Designer.
        InitializeComponent()
    End If
    AddHandler Me.DataReady, AddressOf Me.OnDataReady
    CreateTask()
End Sub

Public Sub New(ByVal container As IContainer)
    Me.New()
    container.Add(Me)
    CreateTask()
End Sub 'New

Private Sub CreateTask()
    If Not (daqmxTask Is Nothing) Then
        Return
    End If
    If dataReadyHandler Is Nothing Then
        dataReadyHandler = New System.AsyncCallback(AddressOf DataReadyEventHandler)
    End If
    daqmxTask = new DAQmxTask7()
    daqmxReader = New NationalInstruments.DAQmx.AnalogMultiChannelReader(daqmxTask.Stream)
    daqmxReader.SynchronizingObject = Me
End Sub 'CreateTask

Protected Overloads Overrides Sub Dispose(ByVal disposing As Boolean)
    If disposing Then
        If Not (components Is Nothing) Then
            components.Dispose()
        End If
        If Not (daqmxTask Is Nothing) Then
            StopTask()
            daqmxTask.Dispose()
        End If
    End If
    MyBase.Dispose(disposing)
End Sub 'Dispose

Private Sub InitializeComponent()
    '
    'DAQmxUserControl7
    '
    Me.Name = "DAQmxUserControl7"
    Me.Size = New System.Drawing.Size(48, 32)
End Sub 'InitializeComponent

Public Sub StartTask(ByVal samples As Integer)
    'If Not stopped Then
    'Return
    'End If
    'If daqmxTask Is Nothing Then
    'CreateTask()
    'End If
    daqmxTask.Stream.Timeout = -1 'NO TIMEOUTS!!
    daqmxTask.Start()
    stopped = False
    daqmxReader.BeginReadMultiSample(samples, dataReadyHandler, Nothing)
End Sub 'StartTask

Public Sub StopTask()
    If Not (Me.daqmxTask Is Nothing) Then
        Me.daqmxTask.Stop()
    End If
    stopped = True
End Sub 'StopTask

Private Sub DAQmxUserControl7_Resize(ByVal sender As Object, ByVal e As System.EventArgs) Handles MyBase.Resize
    waveformGraph1.Size = Me.Size
End Sub 'DAQmxUserControl7_Resize

Public Sub DataReadyEventHandler(ByVal result As IAsyncResult)
    If stopped Then
        Return
    End If

```

```

        Dim data(,) As Double = daqmxReader.EndReadMultiSample(result)
        Dim args As New TaskEventArgs(data)
        RaiseEvent DataReady(Me, args)
    End Sub 'DataReadyEventHandler

    Public Overridable Sub OnDataReady(ByVal sender As Object, ByVal e As TaskEventArgs)
        Dim data(,) As Double = e.GetData()
        Dim sampleNumber As Integer
        sampleNumber = data.GetLength(1)
        Dim xData(sampleNumber - 1) As Double
        Dim yData(sampleNumber - 1) As Double
        Dim i As Integer
        For i = 0 To sampleNumber - 1
            xData(i) = (1000 * i) / samplingRate 'change sample# to msec
            yData(i) = data(0, i) * 1000 / currentGain 'convert to pA; adjust for gain
        Next i
        Me.pForm.DaqmxUserControl31.YAxis1.Range = New NationalInstruments.UI.Range(-10000 / currentGain, 10000 /
currentGain)
        Me.pForm.DaqmxUserControl31.ScatterPlot1.PlotXY(xData, yData)
        Me.pForm.DaqmxUserControl31.TextBox2.Text = CStr(currentGain)

        'Write data to file and close file
        WriteLine(fileName, currentGain) 'write gain
        WriteLine(fileNameLeft, currentGain)
        WriteLine(fileNameRight, currentGain)
        'WriteLine(fileName, sampleNumber) 'write number of samples
        For i = 0 To sampleNumber - 1
            WriteLine(fileName, yData(i)) 'write data
            WriteLine(fileNameLeft, data(1, i))
            WriteLine(fileNameRight, data(2, i))
        Next i
        FileClose(fileName)
        FileClose(fileNameLeft)
        FileClose(fileNameRight)
        Erase data, xData, yData 'Deallocate memory

        Me.daqmxTask.Stop()
        Me.pForm.Button4.Enabled = True

        If Me.pForm.flashAgain = True Then
            Me.pForm.Button15.PerformClick()
        End If

        'If Not stopped Then
        'daqmxReader.BeginReadMultiSample(-1, dataReadyHandler, Nothing)
        'End If
    End Sub 'OnDataReady

    Public Sub Configure(ByVal sampleRate As Integer, ByVal numSamples As Integer, ByVal Laser As String, ByVal fileID As
Integer, ByVal fileIDLeft As Integer, ByVal fileIDRight As Integer, ByVal preTriggerSamples As Integer, ByVal gain As
Double)
        'configure timing, number of samples
        Me.daqmxTask.Timing.ConfigureSampleClock("", sampleRate, SampleClockActiveEdge.Rising,
SampleQuantityMode.FiniteSamples, numSamples)
        Me.samplingRate = sampleRate

        'configure trigger
        If Laser = "Right" Then
            Laser = "PFI2" 'trigger on right laser
        Else
            Laser = "PFI1" 'default trigger on left laser
        End If
        Me.daqmxTask.Triggers.ReferenceTrigger.ConfigureDigitalEdgeTrigger(Laser, DigitalEdgeReferenceTriggerEdge.Rising,
preTriggerSamples)

        fileName = fileID
        fileNameLeft = fileIDLeft
        fileNameRight = fileIDRight
        currentGain = gain
        WriteLine(fileName, numSamples) 'write total number of samples
        WriteLine(fileNameLeft, numSamples)
        WriteLine(fileNameRight, numSamples)
        WriteLine(fileName, preTriggerSamples) 'write number of preTrigger samples
        WriteLine(fileNameLeft, preTriggerSamples)
        WriteLine(fileNameRight, preTriggerSamples)
        Me.pForm.DaqmxUserControl31.ScatterGraph1.ClearData() 'clear graph
        'Me.TextBox1.Text = ""
        'Me.TextBox2.Text = ""
    End Sub
End Class 'DAQmxUserControl7

Public Class Form1
    Inherits System.Windows.Forms.Form

    #Region " Windows Form Designer generated code "

    Public Sub New()
        MyBase.New()
    End Sub

```

```

        'This call is required by the Windows Form Designer.
        InitializeComponent()

        'Add any initialization after the InitializeComponent() call

    End Sub

    'Form overrides dispose to clean up the component list.
    Protected Overloads Overrides Sub Dispose(ByVal disposing As Boolean)
        If disposing Then
            If Not (components Is Nothing) Then
                components.Dispose()
            End If
        End If
        MyBase.Dispose(disposing)
    End Sub

    'Required by the Windows Form Designer
    Private components As System.ComponentModel.IContainer

    'NOTE: The following procedure is required by the Windows Form Designer
    'It can be modified using the Windows Form Designer.
    'Do not modify it using the code editor.
    Friend WithEvents MainMenu1 As System.Windows.Forms.MainMenu
    Friend WithEvents MenuItem1 As System.Windows.Forms.MenuItem
    Friend WithEvents MenuItem2 As System.Windows.Forms.MenuItem
    Friend WithEvents MenuItem3 As System.Windows.Forms.MenuItem
    Friend WithEvents MenuItem4 As System.Windows.Forms.MenuItem
    Friend WithEvents MenuItem5 As System.Windows.Forms.MenuItem
    Friend WithEvents MenuItem6 As System.Windows.Forms.MenuItem
    <System.Diagnostics.DebuggerStepThrough()> Private Sub InitializeComponent()
        Me.MainMenu1 = New System.Windows.Forms.MainMenu
        Me.MenuItem1 = New System.Windows.Forms.MenuItem
        Me.MenuItem3 = New System.Windows.Forms.MenuItem
        Me.MenuItem4 = New System.Windows.Forms.MenuItem
        Me.MenuItem2 = New System.Windows.Forms.MenuItem
        Me.MenuItem5 = New System.Windows.Forms.MenuItem
        Me.MenuItem6 = New System.Windows.Forms.MenuItem
        '
        'MainMenu1
        '
        Me.MainMenu1.MenuItems.AddRange(New System.Windows.Forms.MenuItem() {Me.MenuItem1, Me.MenuItem2, Me.MenuItem5})
        '
        'MenuItem1
        '
        Me.MenuItem1.Index = 0
        Me.MenuItem1.MenuItems.AddRange(New System.Windows.Forms.MenuItem() {Me.MenuItem3, Me.MenuItem4})
        Me.MenuItem1.Text = "Trace"
        '
        'MenuItem3
        '
        Me.MenuItem3.Index = 0
        Me.MenuItem3.Text = "Acquire"
        '
        'MenuItem4
        '
        Me.MenuItem4.Index = 1
        Me.MenuItem4.Text = "Analyze"
        '
        'MenuItem2
        '
        Me.MenuItem2.Index = 1
        Me.MenuItem2.MdiList = True
        Me.MenuItem2.Text = "Window"
        '
        'MenuItem5
        '
        Me.MenuItem5.Index = 2
        Me.MenuItem5.MenuItems.AddRange(New System.Windows.Forms.MenuItem() {Me.MenuItem6})
        Me.MenuItem5.Text = "Protocol"
        '
        'MenuItem6
        '
        Me.MenuItem6.Index = 0
        Me.MenuItem6.Text = "Build"
        '
        'Form1
        '
        Me.AutoScaleBaseSize = New System.Drawing.Size(5, 13)
        Me.ClientSize = New System.Drawing.Size(968, 638)
        Me.IsMdiContainer = True
        Me.Menu = Me.MainMenu1
        Me.Name = "Form1"
        Me.Text = "ACQv3.5"
        Me.WindowState = System.Windows.Forms.FormWindowState.Maximized

    End Sub

#End Region

```



```

Private Sub MenuItem3_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MenuItem3.Click
    'This function opens an acquisition window/
    Dim NewMDIChild As New Form2
    NewMDIChild.MdiParent = Me 'Set the Parent Form of the Child window.
    NewMDIChild.Show() 'Display the new form.
End Sub
End Class

Public Class Form2
    Inherits System.Windows.Forms.Form

    Private choutString As String
    Public filterBuffer(1000) As Integer
    Public filtercount As Integer
    Public numfiltersamples As Integer
    Public flashAgain As Boolean
    Public flashCount As Integer
    Public flashFiftyFileNumber As Integer
    Public currentflash50form As Form4

#Region " Windows Form Designer generated code "

    Public Sub New()
        MyBase.New()

        'This call is required by the Windows Form Designer.
        InitializeComponent()

        'Add any initialization after the InitializeComponent() call

    End Sub

    'Form overrides dispose to clean up the component list.
    Protected Overrides Sub Dispose(ByVal disposing As Boolean)
        If disposing Then
            If Not (components Is Nothing) Then
                components.Dispose()
            End If
        End If
        MyBase.Dispose(disposing)
    End Sub

    'Required by the Windows Form Designer
    Private components As System.ComponentModel.IContainer

    'NOTE: The following procedure is required by the Windows Form Designer
    'It can be modified using the Windows Form Designer.
    'Do not modify it using the code editor.

    Friend WithEvents Button1 As System.Windows.Forms.Button
    Friend WithEvents Label1 As System.Windows.Forms.Label
    Friend WithEvents NumericUpDown1 As System.Windows.Forms.NumericUpDown
    Friend WithEvents AxCWD01 As AxCWDAQControlsLib.AxCWDO
    Friend WithEvents Button3 As System.Windows.Forms.Button
    Friend WithEvents Button2 As System.Windows.Forms.Button
    Friend WithEvents Button4 As System.Windows.Forms.Button
    Friend WithEvents Button5 As System.Windows.Forms.Button
    Friend WithEvents Button6 As System.Windows.Forms.Button
    Friend WithEvents TextBox2 As System.Windows.Forms.TextBox
    Friend WithEvents Label2 As System.Windows.Forms.Label
    Friend WithEvents TextBox3 As System.Windows.Forms.TextBox
    Friend WithEvents Label3 As System.Windows.Forms.Label
    Friend WithEvents TextBox4 As System.Windows.Forms.TextBox
    Friend WithEvents NumericUpDown2 As System.Windows.Forms.NumericUpDown
    Friend WithEvents Label5 As System.Windows.Forms.Label
    Friend WithEvents ListBox1 As System.Windows.Forms.ListBox
    Friend WithEvents Label6 As System.Windows.Forms.Label
    Friend WithEvents DaQmxUserControl31 As ACQv1.DAQmxUserControl3
    Friend WithEvents Label7 As System.Windows.Forms.Label
    Friend WithEvents NumericUpDown4 As System.Windows.Forms.NumericUpDown
    Friend WithEvents Label8 As System.Windows.Forms.Label
    Friend WithEvents NumericUpDown5 As System.Windows.Forms.NumericUpDown
    Friend WithEvents DaQmxUserControl11 As ACQv1.DAQmxUserControl1
    Friend WithEvents HScrollBar1 As System.Windows.Forms.HScrollBar
    Friend WithEvents Label9 As System.Windows.Forms.Label
    Friend WithEvents Label10 As System.Windows.Forms.Label
    Friend WithEvents NumericUpDown6 As System.Windows.Forms.NumericUpDown
    Friend WithEvents NumericUpDown7 As System.Windows.Forms.NumericUpDown
    Friend WithEvents ScatterGraph1 As NationalInstruments.UI.WindowsForms.ScatterGraph
    Friend WithEvents XAxis1 As NationalInstruments.UI.XAxis
    Friend WithEvents YAxis1 As NationalInstruments.UI.YAxis
    Friend WithEvents ScatterPlot1 As NationalInstruments.UI.ScatterPlot
    Friend WithEvents Label14 As System.Windows.Forms.Label
    Friend WithEvents Label11 As System.Windows.Forms.Label
    Friend WithEvents TextBox5 As System.Windows.Forms.TextBox
    Friend WithEvents Button7 As System.Windows.Forms.Button
    Friend WithEvents DaQmxUserControl51 As ACQv1.DAQmxUserControl5
    Friend WithEvents CheckBox1 As System.Windows.Forms.CheckBox

```

```

Friend WithEvents ListBox2 As System.Windows.Forms.ListBox
Friend WithEvents Button8 As System.Windows.Forms.Button
Friend WithEvents Button9 As System.Windows.Forms.Button
Friend WithEvents Button10 As System.Windows.Forms.Button
Friend WithEvents Button11 As System.Windows.Forms.Button
Friend WithEvents Button12 As System.Windows.Forms.Button
Friend WithEvents Button13 As System.Windows.Forms.Button
Friend WithEvents filterControl As ACQv1.VisaNSTask1
Friend WithEvents ListBox3 As System.Windows.Forms.ListBox
Friend WithEvents ListBox4 As System.Windows.Forms.ListBox
Friend WithEvents Label12 As System.Windows.Forms.Label
Friend WithEvents Label13 As System.Windows.Forms.Label
Friend WithEvents Label14 As System.Windows.Forms.Label
Friend WithEvents ListBox5 As System.Windows.Forms.ListBox
Friend WithEvents Label15 As System.Windows.Forms.Label
Friend WithEvents Label16 As System.Windows.Forms.Label
Friend WithEvents NumericUpDown3 As System.Windows.Forms.NumericUpDown
Friend WithEvents DaQmxUserControl61 As ACQv1.DAQmxUserControl6
Friend WithEvents Button14 As System.Windows.Forms.Button
Friend WithEvents TextBox1 As System.Windows.Forms.TextBox
Friend WithEvents CheckBox2 As System.Windows.Forms.CheckBox
Friend WithEvents Button15 As System.Windows.Forms.Button
Friend WithEvents CheckBox3 As System.Windows.Forms.CheckBox
Friend WithEvents DaQmxUserControl71 As ACQv1.DAQmxUserControl7
Friend WithEvents CheckBox4 As System.Windows.Forms.CheckBox
Friend WithEvents ScatterGraph2 As NationalInstruments.UI.WindowsForms.ScatterGraph
Friend WithEvents XAxis2 As NationalInstruments.UI.XAxis
Friend WithEvents YAxis2 As NationalInstruments.UI.YAxis
Friend WithEvents ScatterPlot2 As NationalInstruments.UI.ScatterPlot
Friend WithEvents ScatterGraph3 As NationalInstruments.UI.WindowsForms.ScatterGraph
Friend WithEvents XAxis3 As NationalInstruments.UI.XAxis
Friend WithEvents YAxis3 As NationalInstruments.UI.YAxis
Friend WithEvents ScatterPlot3 As NationalInstruments.UI.ScatterPlot
Friend WithEvents Label17 As System.Windows.Forms.Label
Friend WithEvents Label18 As System.Windows.Forms.Label
Friend WithEvents Button16 As System.Windows.Forms.Button
Friend WithEvents TextBox6 As System.Windows.Forms.TextBox

```

```

<System.Diagnostics.DebuggerStepThrough()> Private Sub InitializeComponent()
    Me.components = New System.ComponentModel.Container
    Dim resources As System.Resources.ResourceManager = New System.Resources.ResourceManager(GetType(Form2))
    Me.Button1 = New System.Windows.Forms.Button
    Me.Label1 = New System.Windows.Forms.Label
    Me.NumericUpDown1 = New System.Windows.Forms.NumericUpDown
    Me.AxCWDO1 = New AxCWDAQControlsLib.AxCWDO
    Me.Button3 = New System.Windows.Forms.Button
    Me.Button2 = New System.Windows.Forms.Button
    Me.Button4 = New System.Windows.Forms.Button
    Me.Button5 = New System.Windows.Forms.Button
    Me.Button6 = New System.Windows.Forms.Button
    Me.TextBox2 = New System.Windows.Forms.TextBox
    Me.Label2 = New System.Windows.Forms.Label
    Me.TextBox3 = New System.Windows.Forms.TextBox
    Me.Label3 = New System.Windows.Forms.Label
    Me.TextBox4 = New System.Windows.Forms.TextBox
    Me.NumericUpDown2 = New System.Windows.Forms.NumericUpDown
    Me.Label5 = New System.Windows.Forms.Label
    Me.ListBox1 = New System.Windows.Forms.ListBox
    Me.Label6 = New System.Windows.Forms.Label
    Me.DAQmxUserControl31 = New ACQv1.DAQmxUserControl3(Me.components)
    Me.Label7 = New System.Windows.Forms.Label
    Me.NumericUpDown4 = New System.Windows.Forms.NumericUpDown
    Me.Label8 = New System.Windows.Forms.Label
    Me.NumericUpDown5 = New System.Windows.Forms.NumericUpDown
    Me.DAQmxUserControl111 = New ACQv1.DAQmxUserControl11(Me.components)
    Me.HScrollBar1 = New System.Windows.Forms.HScrollBar
    Me.Label9 = New System.Windows.Forms.Label
    Me.Label10 = New System.Windows.Forms.Label
    Me.NumericUpDown6 = New System.Windows.Forms.NumericUpDown
    Me.NumericUpDown7 = New System.Windows.Forms.NumericUpDown
    Me.ScatterGraph1 = New NationalInstruments.UI.WindowsForms.ScatterGraph
    Me.ScatterPlot1 = New NationalInstruments.UI.ScatterPlot
    Me.XAxis1 = New NationalInstruments.UI.XAxis
    Me.YAxis1 = New NationalInstruments.UI.YAxis
    Me.Label4 = New System.Windows.Forms.Label
    Me.Label11 = New System.Windows.Forms.Label
    Me.TextBox5 = New System.Windows.Forms.TextBox
    Me.Button7 = New System.Windows.Forms.Button
    Me.DAQmxUserControl51 = New ACQv1.DAQmxUserControl5(Me.components)
    Me.CheckBox1 = New System.Windows.Forms.CheckBox
    Me.ListBox2 = New System.Windows.Forms.ListBox
    Me.Button8 = New System.Windows.Forms.Button
    Me.Button9 = New System.Windows.Forms.Button
    Me.Button10 = New System.Windows.Forms.Button
    Me.Button11 = New System.Windows.Forms.Button
    Me.Button12 = New System.Windows.Forms.Button
    Me.Button13 = New System.Windows.Forms.Button
    Me.ListBox3 = New System.Windows.Forms.ListBox
    Me.ListBox4 = New System.Windows.Forms.ListBox

```

```

Me.Label12 = New System.Windows.Forms.Label
Me.Label13 = New System.Windows.Forms.Label
Me.Label14 = New System.Windows.Forms.Label
Me.ListBox5 = New System.Windows.Forms.ListBox
Me.Label15 = New System.Windows.Forms.Label
Me.Label16 = New System.Windows.Forms.Label
Me.NumericUpDown3 = New System.Windows.Forms.NumericUpDown
Me.DaQmxUserControl161 = New ACQv1.DAQmxUserControl6(Me.components)
Me.Button14 = New System.Windows.Forms.Button
Me.TextBox1 = New System.Windows.Forms.TextBox
Me.CheckBox2 = New System.Windows.Forms.CheckBox
Me.Button15 = New System.Windows.Forms.Button
Me.CheckBox3 = New System.Windows.Forms.CheckBox
Me.DaQmxUserControl171 = New ACQv1.DAQmxUserControl7(Me.components)
Me.CheckBox4 = New System.Windows.Forms.CheckBox
Me.ScatterGraph2 = New NationalInstruments.UI.WindowsForms.ScatterGraph
Me.ScatterPlot2 = New NationalInstruments.UI.ScatterPlot
Me.XAxis2 = New NationalInstruments.UI.XAxis
Me.YAxis2 = New NationalInstruments.UI.YAxis
Me.ScatterGraph3 = New NationalInstruments.UI.WindowsForms.ScatterGraph
Me.ScatterPlot3 = New NationalInstruments.UI.ScatterPlot
Me.XAxis3 = New NationalInstruments.UI.XAxis
Me.YAxis3 = New NationalInstruments.UI.YAxis
Me.Label17 = New System.Windows.Forms.Label
Me.Label18 = New System.Windows.Forms.Label
Me.Button16 = New System.Windows.Forms.Button
Me.TextBox6 = New System.Windows.Forms.TextBox
CType(Me.NumericUpDown1, System.ComponentModel.ISupportInitialize).BeginInit()
CType(Me.AxCWD01, System.ComponentModel.ISupportInitialize).BeginInit()
CType(Me.NumericUpDown2, System.ComponentModel.ISupportInitialize).BeginInit()
CType(Me.NumericUpDown4, System.ComponentModel.ISupportInitialize).BeginInit()
CType(Me.NumericUpDown5, System.ComponentModel.ISupportInitialize).BeginInit()
CType(Me.NumericUpDown6, System.ComponentModel.ISupportInitialize).BeginInit()
CType(Me.NumericUpDown7, System.ComponentModel.ISupportInitialize).BeginInit()

CType(Me.ScatterGraph1, System.ComponentModel.ISupportInitialize).BeginInit()
CType(Me.NumericUpDown3, System.ComponentModel.ISupportInitialize).BeginInit()
CType(Me.ScatterGraph2, System.ComponentModel.ISupportInitialize).BeginInit()
CType(Me.ScatterGraph3, System.ComponentModel.ISupportInitialize).BeginInit()
Me.SuspendLayout()
'
'Button1
'
Me.Button1.Location = New System.Drawing.Point(1072, 16)
Me.Button1.Name = "Button1"
Me.Button1.Size = New System.Drawing.Size(144, 24)
Me.Button1.TabIndex = 0
Me.Button1.TabStop = False
Me.Button1.Text = "TestPulse"
'
'Label1
'
Me.Label1.Location = New System.Drawing.Point(560, 8)
Me.Label1.Name = "Label1"
Me.Label1.Size = New System.Drawing.Size(56, 16)
Me.Label1.TabIndex = 1
Me.Label1.Text = "Vm (mV)"
'
'NumericUpDown1
'
Me.NumericUpDown1.Increment = New Decimal(New Integer() {20, 0, 0, 0})
Me.NumericUpDown1.Location = New System.Drawing.Point(624, 8)
Me.NumericUpDown1.Minimum = New Decimal(New Integer() {100, 0, 0, -2147483648})
Me.NumericUpDown1.Name = "NumericUpDown1"
Me.NumericUpDown1.Size = New System.Drawing.Size(48, 20)
Me.NumericUpDown1.TabIndex = 2
'
'AxCWD01
'
Me.AxCWD01.Enabled = True
Me.AxCWD01.Location = New System.Drawing.Point(208, 0)
Me.AxCWD01.Name = "AxCWD01"
Me.AxCWD01.OcxState = CType(resources.GetObject("AxCWD01.OcxState"), System.Windows.Forms.AxHost.State)
Me.AxCWD01.Size = New System.Drawing.Size(32, 32)
Me.AxCWD01.TabIndex = 3
'
'Button3
'
Me.Button3.Location = New System.Drawing.Point(1072, 104)
Me.Button3.Name = "Button3"
Me.Button3.Size = New System.Drawing.Size(144, 40)
Me.Button3.TabIndex = 6
Me.Button3.Text = "FLASH RIGHT LASER"
'
'Button2
'
Me.Button2.Location = New System.Drawing.Point(1072, 48)
Me.Button2.Name = "Button2"
Me.Button2.Size = New System.Drawing.Size(144, 40)
Me.Button2.TabIndex = 7

```

```

Me.Button2.Text = "FLASH LEFT LASER"
'
'Button4
'
Me.Button4.Location = New System.Drawing.Point(1072, 216)
Me.Button4.Name = "Button4"
Me.Button4.Size = New System.Drawing.Size(144, 40)
Me.Button4.TabIndex = 8
Me.Button4.Text = "FLASH PROTOCOL"
'
'Button5
'
Me.Button5.Location = New System.Drawing.Point(1072, 272)
Me.Button5.Name = "Button5"
Me.Button5.Size = New System.Drawing.Size(144, 24)
Me.Button5.TabIndex = 9
Me.Button5.Text = "Open Flash Protocol"
'
'Button6
'
Me.Button6.Location = New System.Drawing.Point(1072, 464)
Me.Button6.Name = "Button6"
Me.Button6.Size = New System.Drawing.Size(144, 24)
Me.Button6.TabIndex = 11
Me.Button6.Text = "DATA FOLDER"
'
'TextBox2
'
Me.TextBox2.Location = New System.Drawing.Point(1072, 488)
Me.TextBox2.Name = "TextBox2"
Me.TextBox2.ReadOnly = True
Me.TextBox2.Size = New System.Drawing.Size(144, 20)
Me.TextBox2.TabIndex = 12
Me.TextBox2.Text = ""
'
'Label2
'
Me.Label2.Location = New System.Drawing.Point(16, 8)
Me.Label2.Name = "Label2"
Me.Label2.Size = New System.Drawing.Size(88, 32)
Me.Label2.TabIndex = 13
Me.Label2.Text = "Bessel Filter Frequency"
'
'TextBox3
'
Me.TextBox3.Location = New System.Drawing.Point(16, 40)
Me.TextBox3.Name = "TextBox3"
Me.TextBox3.Size = New System.Drawing.Size(88, 20)
Me.TextBox3.TabIndex = 14
Me.TextBox3.Text = ""
'
'Label3
'
Me.Label3.Location = New System.Drawing.Point(1072, 552)
Me.Label3.Name = "Label3"
Me.Label3.Size = New System.Drawing.Size(56, 16)
Me.Label3.TabIndex = 15
Me.Label3.Text = "NOTES"
'
'TextBox4
'
Me.TextBox4.AcceptsReturn = True
Me.TextBox4.AcceptsTab = True
Me.TextBox4.AutoSize = False
Me.TextBox4.Location = New System.Drawing.Point(1072, 568)
Me.TextBox4.Multiline = True
Me.TextBox4.Name = "TextBox4"
Me.TextBox4.Size = New System.Drawing.Size(144, 64)
Me.TextBox4.TabIndex = 16
Me.TextBox4.Text = ""
'
'NumericUpDown2
'
Me.NumericUpDown2.Increment = New Decimal(New Integer() {500, 0, 0, 0})
Me.NumericUpDown2.Location = New System.Drawing.Point(16, 96)
Me.NumericUpDown2.Maximum = New Decimal(New Integer() {300000, 0, 0, 0})
Me.NumericUpDown2.Minimum = New Decimal(New Integer() {500, 0, 0, 0})
Me.NumericUpDown2.Name = "NumericUpDown2"
Me.NumericUpDown2.Size = New System.Drawing.Size(80, 20)
Me.NumericUpDown2.TabIndex = 18
Me.NumericUpDown2.Value = New Decimal(New Integer() {10000, 0, 0, 0})
'
'Label5
'
Me.Label5.Location = New System.Drawing.Point(16, 80)
Me.Label5.Name = "Label5"
Me.Label5.Size = New System.Drawing.Size(104, 16)
Me.Label5.TabIndex = 19
Me.Label5.Text = "Sampling Rate (Hz)"
'

```

```

'ListBox1
,
Me.ListBox1.Items.AddRange(New Object() {"Left", "Right", "Internal"})
Me.ListBox1.Location = New System.Drawing.Point(16, 448)
Me.ListBox1.Name = "ListBox1"
Me.ListBox1.Size = New System.Drawing.Size(56, 43)
Me.ListBox1.TabIndex = 22
,
'Label6
,
Me.Label6.Location = New System.Drawing.Point(16, 416)
Me.Label6.Name = "Label6"
Me.Label6.Size = New System.Drawing.Size(112, 32)
Me.Label6.TabIndex = 23
Me.Label6.Text = "Trigger Acquisition from which laser?"
,
'DaQmxUserControl31
,
Me.DaQmxUserControl31.Location = New System.Drawing.Point(144, 56)
Me.DaQmxUserControl31.Name = "DaQmxUserControl31"
Me.DaQmxUserControl31.Size = New System.Drawing.Size(904, 368)
Me.DaQmxUserControl31.TabIndex = 24
,
'Label7
,
Me.Label7.Location = New System.Drawing.Point(16, 288)
Me.Label7.Name = "Label7"
Me.Label7.Size = New System.Drawing.Size(72, 32)
Me.Label7.TabIndex = 25
Me.Label7.Text = "Left Laser Lambda (nm)"
,
'NumericUpDown4
,
Me.NumericUpDown4.Location = New System.Drawing.Point(16, 320)
Me.NumericUpDown4.Maximum = New Decimal(New Integer() {800, 0, 0, 0})
Me.NumericUpDown4.Minimum = New Decimal(New Integer() {200, 0, 0, 0})
Me.NumericUpDown4.Name = "NumericUpDown4"
Me.NumericUpDown4.Size = New System.Drawing.Size(64, 20)
Me.NumericUpDown4.TabIndex = 26
Me.NumericUpDown4.Value = New Decimal(New Integer() {495, 0, 0, 0})
,
'Label8
,
Me.Label8.Location = New System.Drawing.Point(16, 344)
Me.Label8.Name = "Label8"
Me.Label8.Size = New System.Drawing.Size(72, 32)
Me.Label8.TabIndex = 27
Me.Label8.Text = "Right Laser Lambda (nm)"
,
'NumericUpDown5
,
Me.NumericUpDown5.Location = New System.Drawing.Point(16, 376)
Me.NumericUpDown5.Maximum = New Decimal(New Integer() {800, 0, 0, 0})
Me.NumericUpDown5.Minimum = New Decimal(New Integer() {200, 0, 0, 0})
Me.NumericUpDown5.Name = "NumericUpDown5"
Me.NumericUpDown5.Size = New System.Drawing.Size(64, 20)
Me.NumericUpDown5.TabIndex = 28
Me.NumericUpDown5.Value = New Decimal(New Integer() {580, 0, 0, 0})
,
'DaQmxUserControl11
,
Me.DaQmxUserControl11.Location = New System.Drawing.Point(176, 0)
Me.DaQmxUserControl11.Name = "DaQmxUserControl11"
Me.DaQmxUserControl11.Size = New System.Drawing.Size(16, 16)
Me.DaQmxUserControl11.TabIndex = 29
,
'HSScrollBar1
,
Me.HScrollBar1.LargeChange = 5
Me.HScrollBar1.Location = New System.Drawing.Point(144, 32)
Me.HScrollBar1.Minimum = -100
Me.HScrollBar1.Name = "HSScrollBar1"
Me.HScrollBar1.Size = New System.Drawing.Size(896, 17)
Me.HScrollBar1.SmallChange = 5
Me.HScrollBar1.TabIndex = 0
Me.HScrollBar1.TabStop = True
,
'Label9
,
Me.Label9.Location = New System.Drawing.Point(16, 128)
Me.Label9.Name = "Label9"
Me.Label9.Size = New System.Drawing.Size(96, 40)
Me.Label9.TabIndex = 30
Me.Label9.Text = "PreFlash Collection Time (msec)"
,
'Label10
,
Me.Label10.Location = New System.Drawing.Point(16, 208)
Me.Label10.Name = "Label10"
Me.Label10.Size = New System.Drawing.Size(96, 40)

```

```

Me.Label10.TabIndex = 31
Me.Label10.Text = "PostFlash Collection Time (msec)"
'
'NumericUpDown6
'
Me.NumericUpDown6.Location = New System.Drawing.Point(16, 168)
Me.NumericUpDown6.Maximum = New Decimal(New Integer() {1000, 0, 0, 0})
Me.NumericUpDown6.Minimum = New Decimal(New Integer() {1, 0, 0, 0})
Me.NumericUpDown6.Name = "NumericUpDown6"
Me.NumericUpDown6.Size = New System.Drawing.Size(88, 20)
Me.NumericUpDown6.TabIndex = 32
Me.NumericUpDown6.Value = New Decimal(New Integer() {25, 0, 0, 0})
'
'NumericUpDown7
'
Me.NumericUpDown7.Increment = New Decimal(New Integer() {1000, 0, 0, 0})
Me.NumericUpDown7.Location = New System.Drawing.Point(16, 248)
Me.NumericUpDown7.Maximum = New Decimal(New Integer() {60000, 0, 0, 0})
Me.NumericUpDown7.Minimum = New Decimal(New Integer() {1, 0, 0, 0})
Me.NumericUpDown7.Name = "NumericUpDown7"
Me.NumericUpDown7.Size = New System.Drawing.Size(96, 20)
Me.NumericUpDown7.TabIndex = 33
Me.NumericUpDown7.Value = New Decimal(New Integer() {975, 0, 0, 0})
'
'ScatterGraph1
'
Me.ScatterGraph1.Location = New System.Drawing.Point(144, 704)
Me.ScatterGraph1.Name = "ScatterGraph1"
Me.ScatterGraph1.Plots.AddRange(New NationalInstruments.UI.ScatterPlot() {Me.ScatterPlot1})
Me.ScatterGraph1.Size = New System.Drawing.Size(896, 128)
Me.ScatterGraph1.TabIndex = 34
Me.ScatterGraph1.XAxes.AddRange(New NationalInstruments.UI.XAxis() {Me.XAxis1})
Me.ScatterGraph1.YAxes.AddRange(New NationalInstruments.UI.YAxis() {Me.YAxis1})
'
'ScatterPlot1
'
Me.ScatterPlot1.XAxis = Me.XAxis1
Me.ScatterPlot1.YAxis = Me.YAxis1
'
'XAxis1
'
Me.XAxis1.Caption = "nsec"
Me.XAxis1.Mode = NationalInstruments.UI.AxisMode.Fixed
'
'YAxis1
'
Me.YAxis1.Caption = "Laser Triggers"
Me.YAxis1.Mode = NationalInstruments.UI.AxisMode.Fixed
Me.YAxis1.Range = New NationalInstruments.UI.Range(-0.5, 3)
'
'Label4
'
Me.Label4.Location = New System.Drawing.Point(944, 704)
Me.Label4.Name = "Label4"
Me.Label4.Size = New System.Drawing.Size(96, 16)
Me.Label4.TabIndex = 35
Me.Label4.Text = "Left Laser Trigger"
'
'Label11
'
Me.Label11.Location = New System.Drawing.Point(936, 776)
Me.Label11.Name = "Label11"
Me.Label11.Size = New System.Drawing.Size(104, 16)
Me.Label11.TabIndex = 36
Me.Label11.Text = "Right Laser Trigger"
'
'TextBox5
'
Me.TextBox5.Location = New System.Drawing.Point(1072, 664)
Me.TextBox5.Name = "TextBox5"
Me.TextBox5.Size = New System.Drawing.Size(144, 20)
Me.TextBox5.TabIndex = 37
Me.TextBox5.Text = ""
'
'Button7
'
Me.Button7.Location = New System.Drawing.Point(1072, 640)
Me.Button7.Name = "Button7"
Me.Button7.Size = New System.Drawing.Size(144, 24)
Me.Button7.TabIndex = 38
Me.Button7.Text = "MARK FILE"
'
'DaQmxUserControl51
'
Me.DaQmxUserControl51.Location = New System.Drawing.Point(712, 64)
Me.DaQmxUserControl51.Name = "DaQmxUserControl51"
Me.DaQmxUserControl51.Size = New System.Drawing.Size(320, 24)
Me.DaQmxUserControl51.TabIndex = 39
'
'CheckBox1

```

```

,
Me.CheckBox1.Location = New System.Drawing.Point(1072, 512)
Me.CheckBox1.Name = "CheckBox1"
Me.CheckBox1.Size = New System.Drawing.Size(144, 32)
Me.CheckBox1.TabIndex = 40
Me.CheckBox1.Text = "Append Protocol to File Name?"
,
'ListBox2
,
Me.ListBox2.HorizontalScrollbar = True
Me.ListBox2.Location = New System.Drawing.Point(1072, 296)
Me.ListBox2.Name = "ListBox2"
Me.ListBox2.Size = New System.Drawing.Size(144, 160)
Me.ListBox2.TabIndex = 41
,
'Button8
,
Me.Button8.Location = New System.Drawing.Point(16, 512)
Me.Button8.Name = "Button8"
Me.Button8.Size = New System.Drawing.Size(112, 24)
Me.Button8.TabIndex = 42
Me.Button8.Text = "Shutter 1 Open"
,
'Button9
,
Me.Button9.Location = New System.Drawing.Point(16, 544)
Me.Button9.Name = "Button9"
Me.Button9.Size = New System.Drawing.Size(112, 24)
Me.Button9.TabIndex = 43
Me.Button9.Text = "Shutter 1 Close"
,
'Button10
,
Me.Button10.Location = New System.Drawing.Point(16, 576)
Me.Button10.Name = "Button10"
Me.Button10.Size = New System.Drawing.Size(112, 24)
Me.Button10.TabIndex = 44
Me.Button10.Text = "Shutter 2 Open"
,
'Button11
,
Me.Button11.Location = New System.Drawing.Point(16, 608)
Me.Button11.Name = "Button11"
Me.Button11.Size = New System.Drawing.Size(112, 24)
Me.Button11.TabIndex = 45
Me.Button11.Text = "Shutter 2 Close"
,
'Button12
,
Me.Button12.Location = New System.Drawing.Point(16, 640)
Me.Button12.Name = "Button12"
Me.Button12.Size = New System.Drawing.Size(112, 32)
Me.Button12.TabIndex = 46
Me.Button12.Text = "Lambda-10 Shutter Open"
,
'Button13
,
Me.Button13.Location = New System.Drawing.Point(16, 680)
Me.Button13.Name = "Button13"
Me.Button13.Size = New System.Drawing.Size(112, 32)
Me.Button13.TabIndex = 47
Me.Button13.Text = "Lambda-10 Shutter Close"
,
'ListBox3
,
Me.ListBox3.AllowDrop = True
Me.ListBox3.Items.AddRange(New Object() {"0", "0.1", "0.2", "0.3", "0.4", "0.5", "4", "block"})
Me.ListBox3.Location = New System.Drawing.Point(0, 760)
Me.ListBox3.Name = "ListBox3"
Me.ListBox3.Size = New System.Drawing.Size(32, 108)
Me.ListBox3.TabIndex = 48
,
'ListBox4
,
Me.ListBox4.AllowDrop = True
Me.ListBox4.Items.AddRange(New Object() {"0", "0.5", "1", "2", "3", "4", "block"})
Me.ListBox4.Location = New System.Drawing.Point(112, 760)
Me.ListBox4.Name = "ListBox4"
Me.ListBox4.Size = New System.Drawing.Size(32, 95)
Me.ListBox4.TabIndex = 49
,
'Label12
,
Me.Label12.Location = New System.Drawing.Point(0, 744)
Me.Label12.Name = "Label12"
Me.Label12.Size = New System.Drawing.Size(48, 16)
Me.Label12.TabIndex = 50
Me.Label12.Text = "Wheel A"
,

```

```

'Label13
,
Me.Label13.Location = New System.Drawing.Point(96, 744)
Me.Label13.Name = "Label13"
Me.Label13.Size = New System.Drawing.Size(48, 16)
Me.Label13.TabIndex = 51
Me.Label13.Text = "Wheel B"
,
'Label14
,
Me.Label14.Location = New System.Drawing.Point(56, 768)
Me.Label14.Name = "Label14"
Me.Label14.Size = New System.Drawing.Size(32, 16)
Me.Label14.TabIndex = 52
Me.Label14.Text = "Inline"
,
'ListBox5
,
Me.ListBox5.Items.AddRange(New Object() {"0", "1.5", "4"})
Me.ListBox5.Location = New System.Drawing.Point(56, 784)
Me.ListBox5.Name = "ListBox5"
Me.ListBox5.Size = New System.Drawing.Size(24, 43)
Me.ListBox5.TabIndex = 53
,
'Label15
,
Me.Label15.Location = New System.Drawing.Point(32, 728)
Me.Label15.Name = "Label15"
Me.Label15.Size = New System.Drawing.Size(80, 16)
Me.Label15.TabIndex = 54
Me.Label15.Text = "Laser 1 Filter"
,
'Label16
,
Me.Label16.Location = New System.Drawing.Point(32, 872)
Me.Label16.Name = "Label16"
Me.Label16.Size = New System.Drawing.Size(72, 16)
Me.Label16.TabIndex = 55
Me.Label16.Text = "Laser 2 Filter"
,
'NumericUpDown3
,
Me.NumericUpDown3.DecimalPlaces = 1
Me.NumericUpDown3.Increment = New Decimal(New Integer() {1, 0, 0, 65536})
Me.NumericUpDown3.Location = New System.Drawing.Point(48, 888)
Me.NumericUpDown3.Maximum = New Decimal(New Integer() {85, 0, 0, 65536})
Me.NumericUpDown3.Name = "NumericUpDown3"
Me.NumericUpDown3.Size = New System.Drawing.Size(40, 20)
Me.NumericUpDown3.TabIndex = 56
,
'DaQmxUserControl61
,
Me.DaQmxUserControl61.Location = New System.Drawing.Point(272, 0)
Me.DaQmxUserControl61.Name = "DaQmxUserControl61"
Me.DaQmxUserControl61.Size = New System.Drawing.Size(104, 16)
Me.DaQmxUserControl61.TabIndex = 58
,
'Button14
,
Me.Button14.Location = New System.Drawing.Point(1072, 704)
Me.Button14.Name = "Button14"
Me.Button14.Size = New System.Drawing.Size(144, 23)
Me.Button14.TabIndex = 59
Me.Button14.Text = "Open Filter Protocol"
,
'TextBox1
,
Me.TextBox1.Location = New System.Drawing.Point(1072, 728)
Me.TextBox1.Name = "TextBox1"
Me.TextBox1.ReadOnly = True
Me.TextBox1.Size = New System.Drawing.Size(144, 20)
Me.TextBox1.TabIndex = 60
Me.TextBox1.Text = ""
,
'CheckBox2
,
Me.CheckBox2.Location = New System.Drawing.Point(1072, 744)
Me.CheckBox2.Name = "CheckBox2"
Me.CheckBox2.Size = New System.Drawing.Size(144, 24)
Me.CheckBox2.TabIndex = 61
Me.CheckBox2.Text = "Use Filter Protocol?"
,
'Button15
,
Me.Button15.Location = New System.Drawing.Point(1072, 784)
Me.Button15.Name = "Button15"
Me.Button15.Size = New System.Drawing.Size(152, 24)
Me.Button15.TabIndex = 62
Me.Button15.Text = "FLASH PROTOCOL 50X"
,

```



```

'CheckBox3
'
Me.CheckBox3.Location = New System.Drawing.Point(1072, 816)
Me.CheckBox3.Name = "CheckBox3"
Me.CheckBox3.Size = New System.Drawing.Size(152, 24)
Me.CheckBox3.TabIndex = 63
Me.CheckBox3.Text = "Record Laser Output??"
'
'DaQmxUserControl71
'
Me.DaQmxUserControl71.Location = New System.Drawing.Point(408, 8)
Me.DaQmxUserControl71.Name = "DaQmxUserControl71"
Me.DaQmxUserControl71.Size = New System.Drawing.Size(72, 16)
Me.DaQmxUserControl71.TabIndex = 64
'
'CheckBox4
'
Me.CheckBox4.Location = New System.Drawing.Point(1072, 848)
Me.CheckBox4.Name = "CheckBox4"
Me.CheckBox4.Size = New System.Drawing.Size(144, 24)
Me.CheckBox4.TabIndex = 65
Me.CheckBox4.Text = "Record FDU Channels?"
'
'ScatterGraph2
'
Me.ScatterGraph2.Location = New System.Drawing.Point(144, 424)
Me.ScatterGraph2.Name = "ScatterGraph2"
Me.ScatterGraph2.Plots.AddRange(New NationalInstruments.UI.ScatterPlot() {Me.ScatterPlot2})
Me.ScatterGraph2.Size = New System.Drawing.Size(896, 280)
Me.ScatterGraph2.TabIndex = 66
Me.ScatterGraph2.XAxes.AddRange(New NationalInstruments.UI.XAxis() {Me.XAxis2})
Me.ScatterGraph2.YAxes.AddRange(New NationalInstruments.UI.YAxis() {Me.YAxis2})
'
'ScatterPlot2
'
Me.ScatterPlot2.XAxis = Me.XAxis2
Me.ScatterPlot2.YAxis = Me.YAxis2
'
'ScatterGraph3
'
Me.ScatterGraph3.Location = New System.Drawing.Point(144, 832)
Me.ScatterGraph3.Name = "ScatterGraph3"

Me.ScatterGraph3.Plots.AddRange(New NationalInstruments.UI.ScatterPlot() {Me.ScatterPlot3})
Me.ScatterGraph3.Size = New System.Drawing.Size(896, 88)
Me.ScatterGraph3.TabIndex = 67
Me.ScatterGraph3.XAxes.AddRange(New NationalInstruments.UI.XAxis() {Me.XAxis3})
Me.ScatterGraph3.YAxes.AddRange(New NationalInstruments.UI.YAxis() {Me.YAxis3})
'
'ScatterPlot3
'
Me.ScatterPlot3.XAxis = Me.XAxis3
Me.ScatterPlot3.YAxis = Me.YAxis3
'
'Label17
'
Me.Label17.Location = New System.Drawing.Point(992, 440)
Me.Label17.Name = "Label17"
Me.Label17.Size = New System.Drawing.Size(40, 16)
Me.Label17.TabIndex = 68
Me.Label17.Text = "FDU1"
'
'Label18
'
Me.Label18.Location = New System.Drawing.Point(992, 840)
Me.Label18.Name = "Label18"
Me.Label18.Size = New System.Drawing.Size(40, 16)
Me.Label18.TabIndex = 69
Me.Label18.Text = "FDU2"
'
'Button16
'
Me.Button16.Location = New System.Drawing.Point(1072, 160)
Me.Button16.Name = "Button16"
Me.Button16.Size = New System.Drawing.Size(144, 40)
Me.Button16.TabIndex = 70
Me.Button16.Text = "FIRE INTERNAL TRIGGER"
'
'TextBox6
'
Me.TextBox6.Location = New System.Drawing.Point(1224, 784)
Me.TextBox6.Name = "TextBox6"
Me.TextBox6.ReadOnly = True
Me.TextBox6.Size = New System.Drawing.Size(32, 20)
Me.TextBox6.TabIndex = 71
Me.TextBox6.Text = ""
'
'Form2
'
Me.AutoScaleBaseSize = New System.Drawing.Size(5, 13)

```

```

Me.ClientSize = New System.Drawing.Size(1264, 926)
Me.Controls.Add(Me.TextBox6)
Me.Controls.Add(Me.Button16)
Me.Controls.Add(Me.Label18)
Me.Controls.Add(Me.Label17)
Me.Controls.Add(Me.ScatterGraph3)
Me.Controls.Add(Me.ScatterGraph2)
Me.Controls.Add(Me.CheckBox4)
Me.Controls.Add(Me.DaQmxUserControl71)
Me.Controls.Add(Me.CheckBox3)
Me.Controls.Add(Me.Button15)
Me.Controls.Add(Me.CheckBox2)
Me.Controls.Add(Me.TextBox1)
Me.Controls.Add(Me.Button14)
Me.Controls.Add(Me.DaQmxUserControl61)
Me.Controls.Add(Me.NumericUpDown3)
Me.Controls.Add(Me.Label16)
Me.Controls.Add(Me.Label15)
Me.Controls.Add(Me.ListBox5)
Me.Controls.Add(Me.Label14)
Me.Controls.Add(Me.Label13)
Me.Controls.Add(Me.Label12)
Me.Controls.Add(Me.ListBox4)
Me.Controls.Add(Me.ListBox3)
Me.Controls.Add(Me.Button13)
Me.Controls.Add(Me.Button12)
Me.Controls.Add(Me.Button11)
Me.Controls.Add(Me.Button10)
Me.Controls.Add(Me.Button9)
Me.Controls.Add(Me.Button8)
Me.Controls.Add(Me.ListBox2)
Me.Controls.Add(Me.CheckBox1)
Me.Controls.Add(Me.DaQmxUserControl51)
Me.Controls.Add(Me.Button7)
Me.Controls.Add(Me.TextBox5)
Me.Controls.Add(Me.TextBox4)
Me.Controls.Add(Me.TextBox3)
Me.Controls.Add(Me.TextBox2)
Me.Controls.Add(Me.Label11)
Me.Controls.Add(Me.Label4)
Me.Controls.Add(Me.ScatterGraph1)
Me.Controls.Add(Me.NumericUpDown7)
Me.Controls.Add(Me.NumericUpDown6)
Me.Controls.Add(Me.Label10)
Me.Controls.Add(Me.Label9)
Me.Controls.Add(Me.HScrollBar1)
Me.Controls.Add(Me.DaQmxUserControl11)
Me.Controls.Add(Me.NumericUpDown5)
Me.Controls.Add(Me.Label8)
Me.Controls.Add(Me.NumericUpDown4)
Me.Controls.Add(Me.Label7)
Me.Controls.Add(Me.DaQmxUserControl31)
Me.Controls.Add(Me.Label6)
Me.Controls.Add(Me.ListBox1)
Me.Controls.Add(Me.Label5)
Me.Controls.Add(Me.NumericUpDown2)
Me.Controls.Add(Me.Label3)
Me.Controls.Add(Me.Label2)
Me.Controls.Add(Me.Button6)
Me.Controls.Add(Me.Button5)
Me.Controls.Add(Me.Button4)
Me.Controls.Add(Me.Button2)
Me.Controls.Add(Me.Button3)
Me.Controls.Add(Me.AxCWD01)
Me.Controls.Add(Me.NumericUpDown1)
Me.Controls.Add(Me.Label1)
Me.Controls.Add(Me.Button1)
Me.Name = "Form2"
Me.Text = "DataAcquisition"
CType(Me.NumericUpDown1, System.ComponentModel.ISupportInitialize).EndInit()
CType(Me.AxCWD01, System.ComponentModel.ISupportInitialize).EndInit()
CType(Me.NumericUpDown2, System.ComponentModel.ISupportInitialize).EndInit()
CType(Me.NumericUpDown4, System.ComponentModel.ISupportInitialize).EndInit()
CType(Me.NumericUpDown5, System.ComponentModel.ISupportInitialize).EndInit()
CType(Me.NumericUpDown6, System.ComponentModel.ISupportInitialize).EndInit()
CType(Me.NumericUpDown7, System.ComponentModel.ISupportInitialize).EndInit()
CType(Me.ScatterGraph1, System.ComponentModel.ISupportInitialize).EndInit()
CType(Me.NumericUpDown3, System.ComponentModel.ISupportInitialize).EndInit()
CType(Me.ScatterGraph2, System.ComponentModel.ISupportInitialize).EndInit()
CType(Me.ScatterGraph3, System.ComponentModel.ISupportInitialize).EndInit()
Me.ResumeLayout(False)

End Sub

#End Region

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
    'This button opens up a test pulse window (Class Form3)
    Me.DaQmxUserControl11.Dispose()
    'disable the TEST PULSE and FLASH PROTOCOL buttons.

```

```

Me.Button1.Enabled = False
Me.Button4.Enabled = False
Dim NewMDIChild As New Form3
NewMDIChild.MdiParent = Me.MdiParent 'Set the Parent Form of the Child window.
NewMDIChild.AcqForm = Me 'Passes the parent acquisition form to the test pulse window.
NewMDIChild.NumericUpDown1.Value = Me.NumericUpDown1.Value 'Passes current holding potential to test pulse window.
NewMDIChild.Show() 'Display the new form.
End Sub

Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button3.Click
    'This button flashes Laser 2; no data acquisition occurs.
    Dim data(37) As Object
    Dim i As Integer
    For i = 0 To 35
        data(i) = 1 'Laser 1 burst high; laser 2 burst low. Laser 2 is charged for 35ms
    Next i
    data(36) = 33 'Laser 1 burst high; Laser 2 triggered
    data(37) = 17 'Laser 1 and Laser 2 burst high.
    Me.AxCWD01.Device = 2 'DIO-32HS board
    Me.AxCWD01.NPPatterns = 38 'This must be even.
    Me.AxCWD01.ChannelString = choutString 'Accessing port 0 on the DIO-32HS board.
    Me.AxCWD01.Continuous = False
    Me.AxCWD01.ProgressInterval = 0 'No progress events.
    Me.AxCWD01.UpdateClock.ClockSourceType = CWDAQControlsLib.CWDIOClockSources.cwdioCSInternalClock
    Me.AxCWD01.UpdateClock.Frequency = 1000 'Update rate is 1000 Hz.
    Me.AxCWD01.Configure() 'configure the board.
    Me.AxCWD01.Write(data) 'write the waveform to data buffer.
    Me.AxCWD01.Start() 'start the write process.
End Sub

Private Sub Button5_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button5.Click
    'This button loads a protocol into the protocol list (ListBox2)
    Dim openFile As New System.Windows.Forms.OpenFileDialog
    openFile.DefaultExt = ".txt" 'filter for text files (*.txt)
    openFile.Filter = "TextFiles (*.txt)|*.txt"
    openFile.InitialDirectory = "c:\\" 'set initial directory for dialog box
    openFile.ShowDialog()
    If openFile.FileNames.Length > 0 Then
        Dim filename As String
        For Each filename In openFile.FileNames
            Me.ListBox2.Items.Add(filename) 'add protocol file to list.
            Me.ListBox2.SelectedItem = filename 'Make current protocol the selected one.
        Next filename
    End If
End Sub

Private Sub Button6_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button6.Click
    Dim dataFolder As New FolderBrowserDialog
    dataFolder.ShowDialog()
    Me.TextBox2.Text = dataFolder.SelectedPath
End Sub

Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button4.Click
    'This button flashes the protocol selected in the protocol list (ListBox2), acquires data,
    'writes it to disk, and displays it on the graph.

    Me.Button4.Enabled = False 'Disable flash protocol button until acquisition complete.
    Me.DaQmxUserControl131.pForm = Me
    Me.DaQmxUserControl171.pForm = Me

    'Check if protocol file and data folder have been set...
    If Me.ListBox2.SelectedItem = "" Then
        MsgBox("No flash protocol specified!!", MsgBoxStyle.OKOnly, "Initialization Error")
        Me.Button4.Enabled = True
        Return
    End If
    If Me.TextBox2.Text = "" Then
        MsgBox("No data folder specified!!", MsgBoxStyle.OKOnly, "Initialization Error")
        Me.Button4.Enabled = True
        Return
    End If

    'Check if sampling rate is okay...
    If Me.NumericUpDown2.Value > 300000 Then
        MsgBox("Sampling Rate is too high!", MsgBoxStyle.OKOnly, "Sampling Rate Error")
        Me.Button4.Enabled = True
        Return
    End If
    If Me.CheckBox3.Checked And Me.NumericUpDown2.Value > 100000 Then
        MsgBox("Sampling Rate is too high!", MsgBoxStyle.OKOnly, "Sampling Rate Error")
        Me.Button4.Enabled = True
        Return
    End If
    If Me.CheckBox4.Checked And Me.NumericUpDown2.Value > 100000 Then
        MsgBox("sampling rate is too high!", MsgBoxStyle.OKOnly, "Sampling Rate Error")
        Me.Button4.Enabled = True
        Return
    End If
    If Me.CheckBox3.Checked And Me.CheckBox4.Checked And Me.NumericUpDown2.Value > 60000 Then
        MsgBox("Sampling Rate is too high!", MsgBoxStyle.OKOnly, "Sampling Rate Error")

```

```

        Me.Button4.Enabled = True
        Return
    End If

    'Opens flash protocol file from selected file in protocol list (ListBox2).
    Dim fileNumber As Integer
    fileNumber = FreeFile()
    FileOpen(fileNumber, Me.ListBox2.SelectedItem, OpenMode.Input, OpenAccess.Read)

    'Read in UpdateRate, NumPatterns. Checks to make sure number of patterns is even.
    'Checks to make sure UpdateRate is at least 1000Hz.
    Dim UpdateRate As Integer
    Dim NumPatterns As Integer
    UpdateRate = CInt(LineInput(fileNumber))
    NumPatterns = CInt(LineInput(fileNumber))
    If NumPatterns Mod 2 = 1 Then
        MsgBox("Number of Patterns is not even!!", MsgBoxStyle.OKOnly, "Flash Protocol Error")
        FileClose(fileNumber)
        Me.Button4.Enabled = True
        Return
    End If

    'check if update rate is too small; less than 1000Hz and you cannot accurately charge lasers
    If UpdateRate < 1000 Then
        MsgBox("Update rate is too small (<100Hz)", MsgBoxStyle.Question, "Flash Protocol Error")
        FileClose(fileNumber)
        Me.Button4.Enabled = True
        Return
    End If

    'Read in digital output buffer; close flash protocol file
    'Dim trueNumPatterns As Integer
    'If NumPatterns Mod 2 = 1 Then
    'trueNumPatterns = NumPatterns + 1
    'Else
    'trueNumPatterns = NumPatterns + 2
    'End If
    Dim data(NumPatterns - 1) As Object
    Dim i As Integer
    Dim currentCount As Integer
    currentCount = 0
    Dim numberSteps As Integer
    Dim value As Integer
    While currentCount < NumPatterns
        numberSteps = CInt(LineInput(fileNumber))
        value = CInt(LineInput(fileNumber))
        For i = 1 To numberSteps
            data(currentCount) = value
            currentCount = currentCount + 1
        Next i
    End While
    'If NumPatterns Mod 2 = 1 Then
    'data(NumPatterns) = 17
    'Else
    '    data(NumPatterns) = 17
    '    data(NumPatterns + 1) = 17
    'End If
    FileClose(fileNumber)

    'Figure out which lasers are flashing. Checks each decimal value to see if the
    'trigger pin for either laser is ever set high.
    'Dim leftLaser As Boolean
    'Dim rightLaser As Boolean
    'leftLaser = False
    'rightLaser = False
    'For i = 0 To NumPatterns - 1
    'If (data(i) - (data(i) Mod 2)) Mod 4 = 2 Then
    'leftLaser = True
    'End If
    'If (data(i) - (data(i) Mod 32)) Mod 64 = 32 Then
    'rightLaser = True
    'End If
    'Next i

    'Generate charge buffer: Based on which lasers are flashed, these burst for these lasers
    'are set low for 30ms.
    'Dim numCharge As Integer
    'numCharge = Int(0.03 * UpdateRate) + 1
    'If numCharge Mod 2 = 1 Then
    'numCharge = numCharge + 1
    'End If
    'Dim newData(numCharge + NumPatterns - 1) As Object
    'If leftLaser = True Then
    'If rightLaser = True Then
    'For i = 0 To numCharge - 1
    'newData(i) = 0 'charge both lasers
    'Next i
    'End If
    'If rightLaser = False Then
    'For i = 0 To numCharge - 1

```

```

'newData(i) = 16 'charge left laser
'Next i
'End If
'End If
'If leftLaser = False Then
'If rightLaser = True Then
'For i = 0 To numCharge - 1
'newData(i) = 1 'charge right laser
'Next i
'End If
'If rightLaser = False Then
'MsgBox("No lasers being fired???", MsgBoxStyle.OKOnly, "Flash Protocol Error")
'End If
'End If
'Append protocol to charge buffer.
'For i = numCharge To numCharge + NumPatterns - 1
'newData(i) = data(i - numCharge)
'Next i

'wait 1second
Dim start2, finish2 As Double
start2 = Microsoft.VisualBasic.DateAndTime.Timer
finish2 = start2 + 1.0
Do While Microsoft.VisualBasic.DateAndTime.Timer < finish2
Loop

'Open output file; write initial data
fileNumber = FreeFile()

Dim DateTime, theday, thetime, datetime3 As String
theday = DateString
thetime = TimeString

datetime3 = String.Concat(Mid(theday, 7), Mid(theday, 1, 2), Mid(theday, 4, 2), Mid(thetime, 1, 2), Mid(thetime, 4,
2), Mid(thetime, 7))
DateTime = String.Concat(Me.TextBox2.Text, "\", Mid(theday, 7), Mid(theday, 1, 2), Mid(theday, 4, 2), Mid(thetime,
1, 2), Mid(thetime, 4, 2), Mid(thetime, 7))
Dim DateTime2 As String
DateTime2 = DateTime
Dim t As Integer
Dim l As Integer
If (Me.CheckBox1.Checked) Then
Dim protocol As String
protocol = Me.ListBox2.SelectedItem
t = protocol.LastIndexOf("\")
l = protocol.LastIndexOf(".")
DateTime = String.Concat(DateTime, "_", Mid(protocol, t + 2, l - t - 1))
End If
FileOpen(fileNumber, String.Concat(DateTime, ".txt"), OpenMode.Output, OpenAccess.Write)
'if doing a 50flash trial, write filename to home file
If flashAgain = True Then
WriteLine(flashFiftyFileNumber, datetime3)
End If
WriteLine(fileNumber, Microsoft.VisualBasic.Right(DateTime2, 14)) 'Write DateStamp
WriteLine(fileNumber, Me.NumericUpDown1.Value) 'Write holding potential in mV
WriteLine(fileNumber, Me.ListBox2.SelectedItem) 'Write flash protocol filename
If (Me.CheckBox2.Checked) Then
WriteLine(fileNumber, Me.TextBox1.Text) 'Write filter protocol filename
Else
WriteLine(fileNumber, "None")
End If
WriteLine(fileNumber, Me.NumericUpDown4.Value) 'Write wavelength of Left Laser in nm
WriteLine(fileNumber, Me.NumericUpDown5.Value) 'Write wavelength of Right Laser in nm
WriteLine(fileNumber, Me.ListBox1.SelectedItem) 'Write Laser to trigger acquisition
Dim filter As Double 'write log filtering of laser 1
If Me.ListBox3.SelectedIndex = 7 Then
WriteLine(fileNumber, "block")
ElseIf Me.ListBox4.SelectedIndex = 6 Then
WriteLine(fileNumber, "block")
Else
filter = Cdbl(Me.ListBox3.SelectedItem) + Cdbl(Me.ListBox4.SelectedItem) + Cdbl(Me.ListBox5.SelectedItem)
WriteLine(fileNumber, filter)
End If
WriteLine(fileNumber, Me.NumericUpDown3.Value) 'Write log filtering of laser 2
WriteLine(fileNumber, Me.TextBox3.Text) 'Write Bessel Filter frequency
WriteLine(fileNumber, Me.NumericUpDown2.Value) 'Write Sampling rate in Hz
WriteLine(fileNumber, Me.TextBox4.Text) 'Write Notes to file
'WriteLine(fileNumber, "START") 'Flag for start of data

'If writing laser outputs, write headers of file...
Dim fileNumberLeft, fileNumberRight As Integer
If Me.CheckBox3.Checked = True Then
'open files
fileNumberLeft = FreeFile()
FileOpen(fileNumberLeft, String.Concat(DateTime, "_LaserLeft.txt"), OpenMode.Output, OpenAccess.Write)
fileNumberRight = FreeFile()
FileOpen(fileNumberRight, String.Concat(DateTime, "_LaserRight.txt"), OpenMode.Output, OpenAccess.Write)

WriteLine(fileNumberLeft, Microsoft.VisualBasic.Right(DateTime2, 14)) 'Write DateStamp
WriteLine(fileNumberRight, Microsoft.VisualBasic.Right(DateTime2, 14)) 'Write DateStamp

```

```

WriteLine(fileNumberLeft, Me.NumericUpDown1.Value) 'Write holding potential in mV
WriteLine(fileNumberRight, Me.NumericUpDown1.Value) 'Write holding potential in mV

WriteLine(fileNumberLeft, Me.ListBox2.SelectedItem) 'Write flash protocol filename
WriteLine(fileNumberRight, Me.ListBox2.SelectedItem) 'Write flash protocol filename

If (Me.CheckBox2.Checked) Then
    WriteLine(fileNumberLeft, Me.TextBox1.Text) 'Write filter protocol filename
    WriteLine(fileNumberRight, Me.TextBox1.Text) 'Write filter protocol filename
Else
    WriteLine(fileNumberLeft, "None")
    WriteLine(fileNumberRight, "None")
End If

WriteLine(fileNumberLeft, Me.NumericUpDown4.Value) 'Write wavelength of Left Laser in nm
WriteLine(fileNumberRight, Me.NumericUpDown4.Value) 'Write wavelength of Left Laser in nm

WriteLine(fileNumberLeft, Me.NumericUpDown5.Value) 'Write wavelength of Right Laser in nm
WriteLine(fileNumberRight, Me.NumericUpDown5.Value) 'Write wavelength of Right Laser in nm

WriteLine(fileNumberLeft, Me.ListBox1.SelectedItem) 'Write Laser to trigger acquisition
WriteLine(fileNumberRight, Me.ListBox1.SelectedItem) 'Write Laser to trigger acquisition

'write log filtering of left laser
If Me.ListBox3.SelectedIndex = 7 Then
    WriteLine(fileNumberLeft, "block")
    WriteLine(fileNumberRight, "block")
ElseIf Me.ListBox4.SelectedIndex = 6 Then
    WriteLine(fileNumberLeft, "block")
    WriteLine(fileNumberRight, "block")
Else
    filter = CDb1(Me.ListBox3.SelectedItem) + CDb1(Me.ListBox4.SelectedItem) + CDb1(Me.ListBox5.SelectedItem)
    WriteLine(fileNumberLeft, filter)
    WriteLine(fileNumberRight, filter)
End If

WriteLine(fileNumberLeft, Me.NumericUpDown3.Value) 'Write log filtering of laser 2
WriteLine(fileNumberRight, Me.NumericUpDown3.Value) 'Write log filtering of laser 2

WriteLine(fileNumberLeft, Me.TextBox3.Text) 'Write Bessel Filter frequency
WriteLine(fileNumberRight, Me.TextBox3.Text) 'Write Bessel Filter frequency

WriteLine(fileNumberLeft, Me.NumericUpDown2.Value) 'Write Sampling rate in Hz
WriteLine(fileNumberRight, Me.NumericUpDown2.Value) 'Write Sampling rate in Hz

WriteLine(fileNumberLeft, Me.TextBox4.Text) 'Write Notes to file
WriteLine(fileNumberRight, Me.TextBox4.Text) 'Write Notes to file

'WriteLine(fileNumberLeft, "START") 'Flag for start of data
'WriteLine(fileNumberRight, "START") 'Flag for start of data
End If

'If recording FDU ouputs, write headers of file....
Dim fileNumberFDU1, fileNumberFDU2 As Integer
If Me.CheckBox4.Checked = True Then
    'open files
    fileNumberFDU1 = FreeFile()
    FileOpen(fileNumberFDU1, String.Concat(DateTime, "_FDU1.txt"), OpenMode.Output, OpenAccess.Write)
    fileNumberFDU2 = FreeFile()
    FileOpen(fileNumberFDU2, String.Concat(DateTime, "_FDU2.txt"), OpenMode.Output, OpenAccess.Write)

    WriteLine(fileNumberFDU1, Microsoft.VisualBasic.Right(DateTime2, 14)) 'Write DateStamp
    WriteLine(fileNumberFDU2, Microsoft.VisualBasic.Right(DateTime2, 14)) 'Write DateStamp

    WriteLine(fileNumberFDU1, Me.NumericUpDown1.Value) 'Write holding potential in mV
    WriteLine(fileNumberFDU2, Me.NumericUpDown1.Value) 'Write holding potential in mV

    WriteLine(fileNumberFDU1, Me.ListBox2.SelectedItem) 'Write flash protocol filename
    WriteLine(fileNumberFDU2, Me.ListBox2.SelectedItem) 'Write flash protocol filename

    If (Me.CheckBox2.Checked) Then
        WriteLine(fileNumberFDU1, Me.TextBox1.Text) 'Write filter protocol filename
        WriteLine(fileNumberFDU2, Me.TextBox1.Text) 'Write filter protocol filename
    Else
        WriteLine(fileNumberFDU1, "None")
        WriteLine(fileNumberFDU2, "None")
    End If

    WriteLine(fileNumberFDU1, Me.NumericUpDown4.Value) 'Write wavelength of Left Laser in nm
    WriteLine(fileNumberFDU2, Me.NumericUpDown4.Value) 'Write wavelength of Left Laser in nm

    WriteLine(fileNumberFDU1, Me.NumericUpDown5.Value) 'Write wavelength of Right Laser in nm
    WriteLine(fileNumberFDU2, Me.NumericUpDown5.Value) 'Write wavelength of Right Laser in nm

    WriteLine(fileNumberFDU1, Me.ListBox1.SelectedItem) 'Write Laser to trigger acquisition
    WriteLine(fileNumberFDU2, Me.ListBox1.SelectedItem) 'Write Laser to trigger acquisition

    'write log filtering of left laser
    If Me.ListBox3.SelectedIndex = 7 Then

```

```

        WriteLine(fileNumberFDU1, "block")
        WriteLine(fileNumberFDU2, "block")
    ElseIf Me.ListBox4.SelectedIndex = 6 Then
        WriteLine(fileNumberFDU1, "block")
        WriteLine(fileNumberFDU2, "block")
    Else
        filter = CDb1(Me.ListBox3.SelectedItem) + CDb1(Me.ListBox4.SelectedItem) + CDb1(Me.ListBox5.SelectedItem)
        WriteLine(fileNumberFDU1, filter)
        WriteLine(fileNumberFDU2, filter)
    End If

    WriteLine(fileNumberFDU1, Me.NumericUpDown3.Value) 'Write log filtering of laser 2
    WriteLine(fileNumberFDU2, Me.NumericUpDown3.Value) 'Write log filtering of laser 2

    WriteLine(fileNumberFDU1, Me.TextBox3.Text) 'Write Bessel Filter frequency
    WriteLine(fileNumberFDU2, Me.TextBox3.Text) 'Write Bessel Filter frequency

    WriteLine(fileNumberFDU1, Me.NumericUpDown2.Value) 'Write Sampling rate in Hz
    WriteLine(fileNumberFDU2, Me.NumericUpDown2.Value) 'Write Sampling rate in Hz

    WriteLine(fileNumberFDU1, Me.TextBox4.Text) 'Write Notes to file
    WriteLine(fileNumberFDU2, Me.TextBox4.Text) 'Write Notes to file

    'WriteLine(fileNumberFDU1, "START") 'Flag for start of data
    'WriteLine(fileNumberFDU2, "START") 'Flag for start of data
End If

'Write TimeStamp to screen.
t = DateTime.LastIndexOf("\")
Me.DaQmxUserControl31.TextBox1.Text = Mid(DateTime, t + 2)

'Read in gain, filtering frequency, capacitance
Dim gain As Double
Dim gainData(,) As Double
gainData = Me.DaQmxUserControl51.Read
gain = gainData(0, 0)
Me.DaQmxUserControl51.TextBox1.Text = gainData(2, 0)
Me.DaQmxUserControl51.TextBox2.Text = gainData(1, 0)
WriteLine(fileNumber, String.Concat(CStr(gainData(1, 0)), "kHz")) 'write filtering frequency to file
WriteLine(fileNumber, String.Concat(CStr(gainData(2, 0)), "pF")) 'write cell capacitance to file
WriteLine(fileNumber, "START") 'Flag for start of Data
If Me.CheckBox3.Checked = True Then
    WriteLine(fileNumberLeft, String.Concat(CStr(gainData(1, 0)), "kHz"))
    WriteLine(fileNumberRight, String.Concat(CStr(gainData(1, 0)), "kHz"))
    WriteLine(fileNumberLeft, String.Concat(CStr(gainData(2, 0)), "pF"))
    WriteLine(fileNumberRight, String.Concat(CStr(gainData(2, 0)), "pF"))
    WriteLine(fileNumberLeft, "START") 'Flag for start of data
    WriteLine(fileNumberRight, "START") 'Flag for start of data
End If
If Me.CheckBox4.Checked = True Then
    WriteLine(fileNumberFDU1, String.Concat(CStr(gainData(1, 0)), "kHz"))
    WriteLine(fileNumberFDU2, String.Concat(CStr(gainData(1, 0)), "kHz"))
    WriteLine(fileNumberFDU1, String.Concat(CStr(gainData(2, 0)), "pF"))
    WriteLine(fileNumberFDU2, String.Concat(CStr(gainData(2, 0)), "pF"))
    WriteLine(fileNumberFDU1, "START") 'Flag for start of data
    WriteLine(fileNumberFDU2, "START") 'Flag for start of data
End If

'Configure Analog In Writer
Dim postSamples As Integer
Dim preSamples As Integer
Dim totalSamples As Integer
preSamples = Int(Me.NumericUpDown6.Value * Me.NumericUpDown2.Value / 1000) + 1
postSamples = Int(Me.NumericUpDown7.Value * Me.NumericUpDown2.Value / 1000) + 1
totalSamples = preSamples + postSamples
If Me.CheckBox3.Checked = True Then
    If Me.CheckBox4.Checked = False Then
        'configures for reading in current AND lasers
        Me.DaQmxUserControl31.Configure(Me.NumericUpDown2.Value, totalSamples, Me.ListBox1.SelectedItem, fileNumber,
preSamples, gain, 0, 0, fileNumberLeft, fileNumberRight)
    Else
        'configures for reading in current, lasers and FDU
        Me.DaQmxUserControl31.Configure(Me.NumericUpDown2.Value, totalSamples, Me.ListBox1.SelectedItem, fileNumber,
preSamples, gain, fileNumberFDU1, fileNumberFDU2, fileNumberLeft, fileNumberRight)
    End If
Else
    If Me.CheckBox4.Checked = True Then
        'configures for reading in current and FDU
        Me.DaQmxUserControl31.Configure(Me.NumericUpDown2.Value, totalSamples, Me.ListBox1.SelectedItem, fileNumber,
preSamples, gain, fileNumberFDU1, fileNumberFDU2, 0, 0)
    Else
        'configures for reading in JUST current
        Me.DaQmxUserControl31.Configure(Me.NumericUpDown2.Value, totalSamples, Me.ListBox1.SelectedItem, fileNumber,
preSamples, gain, 0, 0, 0, 0)
    End If
End If

'Configure Digital out writer

```

```

Me.AxCWD01.Device = 2
Me.AxCWD01.NPatterns = NumPatterns
Me.AxCWD01.ChannelString = choutString
Me.AxCWD01.Continuous = False
Me.AxCWD01.AutoSelectProgressInterval = False
Me.AxCWD01.ProgressInterval = 0
Me.AxCWD01.UpdateClock.ClockSourceType = CWDAQControlsLib.CWDIOClockSources.cwdioCSInternalClock
Me.AxCWD01.UpdateClock.Frequency = UpdateRate

'check if a filtering protocol is specified...
If Me.CheckBox2.Checked = True Then
    Dim filterFileNumber As Integer
    Dim proginterval As Integer
    filterFileNumber = FreeFile()
    FileOpen(filterFileNumber, Me.TextBox1.Text, OpenMode.Input, OpenAccess.Read)
    proginterval = Cint(LineInput(filterFileNumber))
    Me.AxCWD01.ProgressInterval = proginterval 'configure digital output to generate progress events
    If NumPatterns Mod proginterval <> 0 Then
        MsgBox("Progress Interval Must Divide Evenly Into Protocol Buffer Size", MsgBoxStyle.OKOnly, "Protocol
Error")
        FileClose(filterFileNumber)
        Me.Button4.Enabled = True
        Return
    End If
    numfiltersamples = Cint(LineInput(filterFileNumber))
    Dim value2 As Integer
    For i = 0 To numfiltersamples - 1
        value2 = Cint(LineInput(filterFileNumber))
        filterBuffer(i) = value2
    Next i
    FileClose(filterFileNumber)
    filtercount = 0
    Me.filterControl.Run(238) 'grab control of lambda-10
    Me.filterControl.Run(filterBuffer(filtercount)) 'write initial sample
    filtercount = filtercount + 1
    Me.filterControl.Run(filterBuffer(filtercount)) 'write initial sample
    filtercount = filtercount + 1
End If

'finish configuring Digital Out writer
Me.AxCWD01.Configure()

'Start Analog in Operation; it should wait for the digital trigger
'If Me.CheckBox3.Checked = True Then
Me.DaQmxUserControl31.StartTask(totalSamples) 'reads in current and laser outputs
'Else
Me.DaQmxUserControl31.StartTask(totalSamples) 'just reads in current
'End If

'wait for pre-trigger time
Dim start, finish As Double
start = Microsoft.VisualBasic.DateAndTime.Timer
finish = start + (Me.NumericUpDown6.Value / 1000) + 0.1
Do While Microsoft.VisualBasic.DateAndTime.Timer < finish
Loop

'Start Digital Out Operation (Flash protocol)
Me.AxCWD01.Write(data)
Me.AxCWD01.Start()

'Once analog input data has been acquired, data is displayed on screen, written to file. The file is then closed

End Sub

Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button2.Click
    'This button flashes the left laser; no data acquisition occurs.
    Dim data(37) As Object
    Dim i As Integer
    For i = 0 To 35
        data(i) = 16
    Next i
    data(36) = 18
    data(37) = 17
    Me.AxCWD01.Device = 2
    Me.AxCWD01.NPatterns = 38
    Me.AxCWD01.ChannelString = choutString
    Me.AxCWD01.Continuous = False
    Me.AxCWD01.ProgressInterval = 0
    Me.AxCWD01.UpdateClock.ClockSourceType = CWDAQControlsLib.CWDIOClockSources.cwdioCSInternalClock
    Me.AxCWD01.UpdateClock.Frequency = 1000
    Me.AxCWD01.Configure()
    Me.AxCWD01.Write(data)
    Me.AxCWD01.Start()
End Sub

Private Sub Form2_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
    Me.DaQmxUserControl11.GenerateData(Me.NumericUpDown1.Value) 'set holding potential
    Me.ListBox1.SelectedIndex = 0
    Me.filterControl = New ACQv1.VisaNSTask1 'create filter control VISA task
    Me.filterControl.Run(238) 'grab control of lambda-10

```



```

Me.filterControl.Run(80) 'set filter wheel A to pos.0; speed 5
Me.ListBox3.SelectedIndex = 0
Me.filterControl.Run(208) 'set filter wheel B to pos.0; speed 5
Me.ListBox4.SelectedIndex = 0
Me.ListBox5.SelectedIndex = 0
Me.CheckBox2.Enabled = False
choutString = "0" 'specify port/channel for digital output
flashAgain = False 'initialization
flashCount = 0 'initialization
End Sub

Private Sub NumericUpDown1_ValueChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
NumericUpDown1.ValueChanged
    Me.HScrollBar1.Value = Me.NumericUpDown1.Value 'sync scroll bar and numeric value
    Me.DaQmxUserControl11.GenerateData(Me.NumericUpDown1.Value) 'set holding potential
End Sub

Public Sub RestartVmControl()
    Me.DaQmxUserControl11.CreateTask()
End Sub

Private Sub HScrollBar1_Scroll(ByVal sender As System.Object, ByVal e As System.Windows.Forms.ScrollEventArgs) Handles
HScrollBar1.Scroll
    Me.NumericUpDown1.Value = Me.HScrollBar1.Value 'sync scroll bar and numeric value
End Sub

Private Sub Button7_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button7.Click
'rename file post-acquisition; used to mark files for selective analysis
If Me.TextBox5.Text = "" Then
    Return
End If
Dim oldName As String
Dim newName As String
oldName = String.Concat(Me.TextBox2.Text, "\", Me.DaQmxUserControl31.TextBox1.Text, ".txt")
newName = String.Concat(Me.TextBox2.Text, "\", Me.DaQmxUserControl31.TextBox1.Text, "_", Me.TextBox5.Text, ".txt")
If Not System.IO.File.Exists(oldName) Then
    MsgBox("File Not Found!", MsgBoxStyle.OKOnly, "File Error")
    Return
End If
Rename(oldName, newName)
Me.DaQmxUserControl31.TextBox1.Text = String.Concat(Me.DaQmxUserControl31.TextBox1.Text, "_", Me.TextBox5.Text)
End Sub

Private Sub ListBox2_SelectedIndexChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
ListBox2.SelectedIndexChanged
'read protocol traces...
'read protocol and write to graph
Dim fileNumber As Integer
fileNumber = FreeFile()
FileOpen(fileNumber, Me.ListBox2.SelectedItem, OpenMode.Input, OpenAccess.Read)

'read in update rate and total buffer size
Dim UpdateRate As Integer
Dim NumPatterns As Integer
UpdateRate = CInt(LineInput(fileNumber))

NumPatterns = CInt(LineInput(fileNumber))

'read in protocol
Dim data(NumPatterns - 1) As Integer
Dim currentcount As Integer
currentcount = 0
Dim value As Integer
Dim numbersteps As Integer
Dim i As Integer
While currentcount < NumPatterns
    numbersteps = CInt(LineInput(fileNumber))
    value = CInt(LineInput(fileNumber))
    For i = 1 To numbersteps
        data(currentcount) = value
        currentcount = currentcount + 1
    Next i
End While
FileClose(fileNumber)

'Set up axes
Dim highVal As Double
highVal = (Cdbl(NumPatterns) * 1000 * 1000 * 1000 / Cdbl(UpdateRate)) + 100
Me.XAxis1.Range = New NationalInstruments.UI.Range(-50, highVal)
Me.YAxis1.Range = New NationalInstruments.UI.Range(-0.5, 3)

Dim xdata(2 * NumPatterns) As Double
Dim ldata(1, 2 * NumPatterns) As Double
xdata(0) = -1
ldata(0, 0) = 1.5
ldata(1, 0) = 0
Dim timeStep As Double
timeStep = 1000 * 1000 * 1000 / Cdbl(UpdateRate)
'Dim stepNumber As Integer

```

```

'Dim flash As Boolean
Dim protocolValue As Integer

'stepNumber = 0
For i = 0 To NumPatterns - 1
    'flash = False
    'stepNumber = stepNumber + 1
    xdata(2 * i + 1) = timeStep * i
    xdata(2 * i + 2) = timeStep * (i + 1) - 1
    protocolValue = data(i)
    If ((protocolValue - protocolValue Mod 2) Mod 4) = 2 Then
        'flash = True
        ldata(0, 2 * i + 1) = 2.5
        ldata(0, 2 * i + 2) = 2.5
        'ldata(0, stepNumber + 2) = 2.5
        'ldata(0, stepNumber + 3) = 2.5
    Else
        ldata(0, 2 * i + 1) = 1.5
        ldata(0, 2 * i + 2) = 1.5
    End If
    If ((protocolValue - protocolValue Mod 32) Mod 64) = 32 Then
        'If Not flash Then
        'flash = True
        'ldata(0, stepNumber + 1) = 1.5
        'ldata(0, stepNumber + 2) = 1.5
        'ldata(0, stepNumber + 3) = 1.5
        'End If
        ldata(1, 2 * i + 1) = 1
        ldata(1, 2 * i + 2) = 1
        'ldata(1, stepNumber + 2) = 1
        'ldata(1, stepNumber + 3) = 1
    Else
        'ldata(1, stepNumber) = 0
        'If flash Then
        ldata(1, 2 * i + 1) = 0
        ldata(1, 2 * i + 2) = 0
        'ldata(1, stepNumber + 3) = 0
    End If
    'End If
    'If flash Then
    'xdata(stepNumber + 1) = xdata(stepNumber) + 1
    'xdata(stepNumber + 2) = xdata(stepNumber) + timeStep - 2
    'xdata(stepNumber + 3) = xdata(stepNumber) + timeStep - 1
    'stepNumber = stepNumber + 3
    'End If
Next i

'For i = stepNumber + 1 To 5 * NumPatterns
'xdata(i) = highVal
'ldata(0, i) = 1.5
'ldata(1, i) = 0
'Next i
Me.ScatterGraph1.PlotXYMultiple(xdata, ldata)
End Sub

Private Sub Button8_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button8.Click
    'This button opens shutter 1.
    'Dim data(3) As Object
    'data(0) = 1
    'data(1) = 1
    'data(2) = 1
    'data(3) = 1
    'Me.AxCWDO2.Device = 2
    'Me.AxCWDO2.NPatterns = 4

    'Me.AxCWDO2.ChannelString = shutterString
    'Me.AxCWDO2.Continuous = False
    'Me.AxCWDO2.ProgressInterval = 0
    'Me.AxCWDO2.UpdateClock.ClockSourceType = CWDAQControlsLib.CWDIOClockSources.cwdioCSInternalClock
    'Me.AxCWDO2.UpdateClock.Frequency = 4
    'Me.AxCWDO2.Configure()
    'Me.AxCWDO2.Write(data)
    'Me.AxCWDO2.Start()
    Me.DaQmxUserControl161.Write(0)
    Me.DaQmxUserControl161.Write(1)
End Sub

Private Sub Button9_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button9.Click
    'This button closes shutter 1.
    'Dim data(3) As Object
    'data(0) = 0
    'data(1) = 0
    'data(2) = 0
    'data(3) = 0
    'Me.AxCWDO2.Device = 2
    'Me.AxCWDO2.NPatterns = 4
    'Me.AxCWDO2.ChannelString = shutterString
    'Me.AxCWDO2.Continuous = False
    'Me.AxCWDO2.ProgressInterval = 0
    'Me.AxCWDO2.UpdateClock.ClockSourceType = CWDAQControlsLib.CWDIOClockSources.cwdioCSInternalClock

```

```

'Me.AxCWDO2.UpdateClock.Frequency = 4
'Me.AxCWDO2.Configure()
'Me.AxCWDO2.Write(data)
'Me.AxCWDO2.Start()
Me.DaQmxUserControl61.Write(0)
Me.DaQmxUserControl61.Write(2)
End Sub

Private Sub Button10_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button10.Click
    'This button opens shutter 2.
    'Dim data(3) As Object
    'data(0) = 0
    'data(1) = 4
    'data(2) = 0
    'data(3) = 0
    'Me.AxCWD01.Device = 2
    'Me.AxCWD01.NPatterns = 4
    'Me.AxCWD01.ChannelString = shutterString
    'Me.AxCWD01.Continuous = False
    'Me.AxCWD01.ProgressInterval = 0
    'Me.AxCWD01.UpdateClock.ClockSourceType = CWDAQControlsLib.CWDIOClockSources.cwdioCSInternalClock
    'Me.AxCWD01.UpdateClock.Frequency = 4
    'Me.AxCWD01.Configure()
    'Me.AxCWD01.Write(data)
    'Me.AxCWD01.Start()
    Me.DaQmxUserControl61.Write(0)
    Me.DaQmxUserControl61.Write(4)
End Sub

Private Sub Button11_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button11.Click
    'This button closes shutter 2.
    'Dim data(3) As Object
    'data(0) = 0
    'data(1) = 8
    'data(2) = 0
    'data(3) = 0
    'Me.AxCWD01.Device = 2
    'Me.AxCWD01.NPatterns = 4
    'Me.AxCWD01.ChannelString = shutterString
    'Me.AxCWD01.Continuous = False
    'Me.AxCWD01.ProgressInterval = 0
    'Me.AxCWD01.UpdateClock.ClockSourceType = CWDAQControlsLib.CWDIOClockSources.cwdioCSInternalClock
    'Me.AxCWD01.UpdateClock.Frequency = 4
    'Me.AxCWD01.Configure()
    'Me.AxCWD01.Write(data)
    'Me.AxCWD01.Start()\
    Me.DaQmxUserControl61.Write(0)
    Me.DaQmxUserControl61.Write(8)
End Sub

Private Sub Button12_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button12.Click
    Me.filterControl.Run(238) 'grab control of lambda-10
    Me.filterControl.Run(186) 'open shutter B
End Sub

Private Sub Button13_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button13.Click
    Me.filterControl.Run(238) 'grab control of lambda-10
    Me.filterControl.Run(188) 'close shutter B
End Sub

Private Sub ListBox3_SelectedIndexChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
ListBox3.SelectedIndexChanged
    Me.filterControl.Run(238) 'grab control of lambda-10
    If Me.ListBox3.SelectedIndex = 0 Then
        Me.filterControl.Run(80)
    ElseIf Me.ListBox3.SelectedIndex = 1 Then
        Me.filterControl.Run(81)
    ElseIf Me.ListBox3.SelectedIndex = 2 Then
        Me.filterControl.Run(82)
    ElseIf Me.ListBox3.SelectedIndex = 3 Then
        Me.filterControl.Run(83)
    ElseIf Me.ListBox3.SelectedIndex = 4 Then
        Me.filterControl.Run(86)
    ElseIf Me.ListBox3.SelectedIndex = 5 Then
        Me.filterControl.Run(85)
    ElseIf Me.ListBox3.SelectedIndex = 6 Then
        Me.filterControl.Run(87)
    ElseIf Me.ListBox3.SelectedIndex = 7 Then
        Me.filterControl.Run(84)
    End If
End Sub

Private Sub ListBox4_SelectedIndexChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
ListBox4.SelectedIndexChanged
    Me.filterControl.Run(238) 'grab control of lambda-10
    Me.filterControl.Run(CInt(208 + Me.ListBox4.SelectedIndex))
End Sub

Private Sub Button14_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button14.Click
    'This button loads a filter protocol.

```

```

Dim openFile As New System.Windows.Forms.OpenFileDialog
openFile.DefaultExt = ".txt" 'filter for text files (*.txt)
openFile.Filter = "TextFiles (*.txt)|*.txt"
'openFile.InitialDirectory = "c:\\" 'set initial directory for dialog box
openFile.ShowDialog()
If openFile.FileNames.Length > 0 Then
    Dim filename As String
    For Each filename In openFile.FileNames
        Me.TextBox1.Text = filename
        Me.CheckBox2.Enabled = True
    Next filename
End If
End Sub

Private Sub AxCWD01_Progress(ByVal sender As Object, ByVal e As AxCWDAQControlsLib._DCWDOEvents_ProgressEvent) Handles
AxCWD01.Progress
    If filtercount < (numfiltersamples - 1) Then
        Me.filterControl.Run(filterBuffer(filtercount))
        Me.filterControl.Run(filterBuffer(filtercount + 1))
        filtercount = filtercount + 2
    End If
End Sub

Private Sub Button15_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button15.Click
    If Me.flashAgain = False Then
        flashCount = 0

        'open home file (stores the filenames of the 50 traces)
        Dim filename As String
        Dim DateTime, theday, thetime As String
        theday = DateString
        thetime = TimeString
        filename = String.Concat(Me.TextBox2.Text, "\", Mid(theday, 7), Mid(theday, 1, 2), Mid(theday, 4, 2),
Mid(thetime, 1, 2), Mid(thetime, 4, 2), Mid(thetime, 7))
        filename = String.Concat(filename, "_flash50.txt")
        flashFiftyFileNumber = FreeFile()
        FileOpen(flashFiftyFileNumber, filename, OpenMode.Output, OpenAccess.Write)
        Dim NewMDIChild As New Form4
        NewMDIChild.MdiParent = Me.MdiParent 'Set the Parent Form of the Child window.
        NewMDIChild.HScrollBar1.Value = 1 'Initializes Hscrollbar.
        NewMDIChild.Text = Microsoft.VisualBasic.Right(filename, 26)
        NewMDIChild.pform = Me
        currentflash50form = NewMDIChild
        NewMDIChild.Show() 'Display the new form.
    End If
    Me.flashAgain = True
    flashCount = flashCount + 1
    If flashCount < 51 Then
        Me.TextBox6.Text = CStr(flashCount)
        Me.Button4.PerformClick()
    Else
        flashCount = 0
        Me.flashAgain = False
        FileClose(flashFiftyFileNumber)
    End If
End Sub

Private Sub Button16_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
    Me.DaQmxUserControl31.Size = New System.Drawing.Size(904, 632)
    Me.DaQmxUserControl31.Size = New System.Drawing.Size(1000, 700)
End Sub

Private Sub Button17_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
    Me.DaQmxUserControl31.Size = New System.Drawing.Size(904, 632)
End Sub

Private Sub Button16_Click_1(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button16.Click
    'This button fires the internal trigger; it does not trigger data acquisition.
    Dim data(5) As Object
    data(0) = 17 'Laser 1 burst high; laser 2 burst high.
    data(1) = 17 'Laser 1 burst high; laser 2 burst high.
    data(2) = 25 'Laser 1 burst high; laser 2 burst high; internal trigger high.
    data(3) = 17 'Laser 1 burst high; laser 2 burst high.
    data(4) = 17 'Laser 1 burst high; laser 2 burst high.
    data(5) = 17 'Laser 1 burst high; laser 2 burst high.
    Me.AxCWD01.Device = 2 'DIO-32HS board
    Me.AxCWD01.NPatterns = 6 'This must be even.
    Me.AxCWD01.ChannelString = choutString 'Accessing port 0 on the DIO-32HS board.
    Me.AxCWD01.Continuous = False
    Me.AxCWD01.ProgressInterval = 0 'No progress events.
    Me.AxCWD01.UpdateClock.ClockSourceType = CWDAQControlsLib.CWDIOClockSources.cwdioCSInternalClock
    Me.AxCWD01.UpdateClock.Frequency = 1000 'Update rate is 1000 Hz.
    Me.AxCWD01.Configure() 'configure the board.
    Me.AxCWD01.Write(data) 'write the waveform to data buffer.
    Me.AxCWD01.Start() 'start the write process.
End Sub

End Class

```

```

Public Class Form3
    Inherits System.Windows.Forms.Form

    #Region " Windows Form Designer generated code "

        Public Sub New()
            MyBase.New()

            'This call is required by the Windows Form Designer.
            InitializeComponent()

            'Add any initialization after the InitializeComponent() call

        End Sub

        'Form overrides dispose to clean up the component list.
        Protected Overrides Sub Dispose(ByVal disposing As Boolean)
            If disposing Then
                If Not (components Is Nothing) Then
                    components.Dispose()
                End If
            End If
            MyBase.Dispose(disposing)
        End Sub

        'Required by the Windows Form Designer
        Private components As System.ComponentModel.IContainer

        'NOTE: The following procedure is required by the Windows Form Designer
        'It can be modified using the Windows Form Designer.
        'Do not modify it using the code editor.
        Friend WithEvents Label1 As System.Windows.Forms.Label
        Friend WithEvents Label2 As System.Windows.Forms.Label
        Public WithEvents NumericUpDown2 As System.Windows.Forms.NumericUpDown
        Public WithEvents NumericUpDown1 As System.Windows.Forms.NumericUpDown
        Public AcqForm As Form2
        Friend WithEvents DaQmxUserControl21 As ACQv1.DAQmxUserControl2
        Friend WithEvents DaQmxUserControl41 As ACQv1.DAQmxUserControl4
        Friend WithEvents HScrollBar1 As System.Windows.Forms.HScrollBar
        <System.Diagnostics.DebuggerStepThrough()> Private Sub InitializeComponent()
            Me.components = New System.ComponentModel.Container
            Me.NumericUpDown1 = New System.Windows.Forms.NumericUpDown
            Me.Label1 = New System.Windows.Forms.Label
            Me.Label2 = New System.Windows.Forms.Label
            Me.NumericUpDown2 = New System.Windows.Forms.NumericUpDown
            Me.DaQmxUserControl21 = New ACQv1.DAQmxUserControl2(Me.components)
            Me.DaQmxUserControl41 = New ACQv1.DAQmxUserControl4(Me.components)
            Me.HScrollBar1 = New System.Windows.Forms.HScrollBar
            CType(Me.NumericUpDown1, System.ComponentModel.ISupportInitialize).BeginInit()
            CType(Me.NumericUpDown2, System.ComponentModel.ISupportInitialize).BeginInit()
            Me.SuspendLayout()
            ,
            'NumericUpDown1
            ,
            Me.NumericUpDown1.Increment = New Decimal(New Integer() {20, 0, 0, 0})
            Me.NumericUpDown1.Location = New System.Drawing.Point(360, 16)
            Me.NumericUpDown1.Minimum = New Decimal(New Integer() {100, 0, 0, -2147483648})
            Me.NumericUpDown1.Name = "NumericUpDown1"
            Me.NumericUpDown1.Size = New System.Drawing.Size(48, 20)
            Me.NumericUpDown1.TabIndex = 1
            Me.NumericUpDown1.TabStop = False
            ,
            'Label1
            ,
            Me.Label1.Location = New System.Drawing.Point(296, 16)
            Me.Label1.Name = "Label1"
            Me.Label1.Size = New System.Drawing.Size(56, 16)
            Me.Label1.TabIndex = 2
            Me.Label1.Text = "Vm (mV)"
            ,
            'Label2
            ,
            Me.Label2.Location = New System.Drawing.Point(736, 16)
            Me.Label2.Name = "Label2"
            Me.Label2.Size = New System.Drawing.Size(64, 16)
            Me.Label2.TabIndex = 3
            Me.Label2.Text = "Pulse (mV)"
            ,
            'NumericUpDown2
            ,
            Me.NumericUpDown2.Increment = New Decimal(New Integer() {10, 0, 0, 0})
            Me.NumericUpDown2.Location = New System.Drawing.Point(744, 40)
            Me.NumericUpDown2.Name = "NumericUpDown2"
            Me.NumericUpDown2.Size = New System.Drawing.Size(48, 20)
            Me.NumericUpDown2.TabIndex = 4
            Me.NumericUpDown2.Value = New Decimal(New Integer() {5, 0, 0, 0})
            ,
            'DaQmxUserControl21
            ,
            Me.DaQmxUserControl21.Location = New System.Drawing.Point(0, 0)

```

```

Me.DaQmxUserControl21.Name = "DaQmxUserControl21"
Me.DaQmxUserControl21.Size = New System.Drawing.Size(40, 24)
Me.DaQmxUserControl21.TabIndex = 10
'
'DaQmxUserControl41
'
Me.DaQmxUserControl41.Location = New System.Drawing.Point(40, 80)
Me.DaQmxUserControl41.Name = "DaQmxUserControl41"
Me.DaQmxUserControl41.Size = New System.Drawing.Size(736, 536)
Me.DaQmxUserControl41.TabIndex = 11
'
'HScrollBar1
'
Me.HScrollBar1.LargeChange = 5
Me.HScrollBar1.Location = New System.Drawing.Point(72, 40)
Me.HScrollBar1.Minimum = -100
Me.HScrollBar1.Name = "HScrollBar1"
Me.HScrollBar1.Size = New System.Drawing.Size(576, 24)
Me.HScrollBar1.SmallChange = 5
Me.HScrollBar1.TabIndex = 1
Me.HScrollBar1.TabStop = True
'
'Form3
'
Me.AutoScaleBaseSize = New System.Drawing.Size(5, 13)
Me.ClientSize = New System.Drawing.Size(800, 614)
Me.Controls.Add(Me.HScrollBar1)
Me.Controls.Add(Me.DaQmxUserControl41)
Me.Controls.Add(Me.DaQmxUserControl21)
Me.Controls.Add(Me.NumericUpDown2)
Me.Controls.Add(Me.Label2)
Me.Controls.Add(Me.Label1)
Me.Controls.Add(Me.NumericUpDown1)
Me.Name = "Form3"
Me.StartPosition = System.Windows.Forms.FormStartPosition.CenterScreen
Me.Text = "TestPulse"
CType(Me.NumericUpDown1, System.ComponentModel.ISupportInitialize).EndInit()
CType(Me.NumericUpDown2, System.ComponentModel.ISupportInitialize).EndInit()
Me.ResumeLayout(False)

End Sub

#End Region

Private Sub Form3_Closing(ByVal sender As System.Object, ByVal e As System.ComponentModel.CancelEventArgs) Handles
MyBase.Closing
'As we close the test pulse window, we pass control of Vm back to the acquisition window
'and update the Vm value. We call a GenerateData event in the case that the holding potential
'was never changed.
'We also reenale the FLASH PROTOCOL and TestPulse Buttons
Me.DaQmxUserControl21.Dispose()
Me.AcqForm.Button1.Enabled = True
Me.AcqForm.Button4.Enabled = True
Me.AcqForm.RestartVmControl()
Me.AcqForm.NumericUpDown1.Value = Me.NumericUpDown1.Value
Me.AcqForm.DaQmxUserControl11.GenerateData(Me.NumericUpDown1.Value)
End Sub

Private Sub NumericUpDown1_ValueChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
NumericUpDown1.ValueChanged
'Update holding potential
Me.HScrollBar1.Value = Me.NumericUpDown1.Value
Me.DaQmxUserControl21.GenerateData(Me.NumericUpDown1.Value, Me.NumericUpDown2.Value)
End Sub

Private Sub NumericUpDown2_ValueChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
NumericUpDown2.ValueChanged
'Update pulse amplitude
Me.DaQmxUserControl21.GenerateData(Me.NumericUpDown1.Value, Me.NumericUpDown2.Value)
End Sub

Private Sub Form3_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
Me.WindowState = FormWindowState.Normal
Me.DaQmxUserControl21.CreateTask()
Me.DaQmxUserControl21.GenerateData(Me.NumericUpDown1.Value, Me.NumericUpDown2.Value)
Me.DaQmxUserControl41.pulseForm = Me
Me.DaQmxUserControl41.StartTask() 'start analog input; analog output starts automatically
End Sub

Private Sub HScrollBar1_Scroll(ByVal sender As System.Object, ByVal e As System.Windows.Forms.ScrollEventArgs) Handles
HScrollBar1.Scroll
Me.NumericUpDown1.Value = Me.HScrollBar1.Value
End Sub

End Class

Public Class Form4
Inherits System.Windows.Forms.Form
Private plotcount As Integer
Public pform As Form2

```

```

#Region " Windows Form Designer generated code "

Public Sub New()
    MyBase.New()

    'This call is required by the Windows Form Designer.
    InitializeComponent()

    'Add any initialization after the InitializeComponent() call

End Sub

'Form overrides dispose to clean up the component list.
Protected Overloads Overrides Sub Dispose(ByVal disposing As Boolean)
    If disposing Then
        If Not (components Is Nothing) Then
            components.Dispose()
        End If
    End If
    MyBase.Dispose(disposing)
End Sub

'Required by the Windows Form Designer
Private components As System.ComponentModel.IContainer

'NOTE: The following procedure is required by the Windows Form Designer
'It can be modified using the Windows Form Designer.
'Do not modify it using the code editor.
Friend WithEvents ScatterGraph1 As NationalInstruments.UI.WindowsForms.ScatterGraph
Friend WithEvents XAxis1 As NationalInstruments.UI.XAxis
Friend WithEvents YAxis1 As NationalInstruments.UI.YAxis
Friend WithEvents ScatterPlot1 As NationalInstruments.UI.ScatterPlot
Friend WithEvents HScrollBar1 As System.Windows.Forms.HScrollBar
Friend WithEvents TextBox1 As System.Windows.Forms.TextBox
Friend WithEvents Button1 As System.Windows.Forms.Button
<System.Diagnostics.DebuggerStepThrough()> Private Sub InitializeComponent()
    Me.ScatterGraph1 = New NationalInstruments.UI.WindowsForms.ScatterGraph
    Me.ScatterPlot1 = New NationalInstruments.UI.ScatterPlot
    Me.XAxis1 = New NationalInstruments.UI.XAxis
    Me.YAxis1 = New NationalInstruments.UI.YAxis
    Me.HScrollBar1 = New System.Windows.Forms.HScrollBar
    Me.TextBox1 = New System.Windows.Forms.TextBox
    Me.Button1 = New System.Windows.Forms.Button
    CType(Me.ScatterGraph1, System.ComponentModel.ISupportInitialize).BeginInit()
    Me.SuspendLayout()
    'ScatterGraph1
    '
    Me.ScatterGraph1.Location = New System.Drawing.Point(24, 16)
    Me.ScatterGraph1.Name = "ScatterGraph1"
    Me.ScatterGraph1.Plots.AddRange(New NationalInstruments.UI.ScatterPlot() {Me.ScatterPlot1})
    Me.ScatterGraph1.Size = New System.Drawing.Size(784, 512)
    Me.ScatterGraph1.TabIndex = 1
    Me.ScatterGraph1.TabStop = False
    Me.ScatterGraph1.XAxes.AddRange(New NationalInstruments.UI.XAxis() {Me.XAxis1})
    Me.ScatterGraph1.YAxes.AddRange(New NationalInstruments.UI.YAxis() {Me.YAxis1})
    '
    'ScatterPlot1
    '
    Me.ScatterPlot1.XAxis = Me.XAxis1
    Me.ScatterPlot1.YAxis = Me.YAxis1
    '
    'XAxis1
    '
    Me.XAxis1.Caption = "msec"
    '
    'YAxis1
    '
    Me.YAxis1.Caption = "pA"
    '
    'HScrollBar1
    '
    Me.HScrollBar1.LargeChange = 1
    Me.HScrollBar1.Location = New System.Drawing.Point(24, 552)
    Me.HScrollBar1.Maximum = 50
    Me.HScrollBar1.Minimum = 1
    Me.HScrollBar1.Name = "HScrollBar1"
    Me.HScrollBar1.Size = New System.Drawing.Size(784, 24)
    Me.HScrollBar1.TabIndex = 0
    Me.HScrollBar1.TabStop = True
    Me.HScrollBar1.Value = 1
    '
    'TextBox1
    '
    Me.TextBox1.Location = New System.Drawing.Point(88, 584)
    Me.TextBox1.Name = "TextBox1"

```

```

Me.TextBox1.ReadOnly = True
Me.TextBox1.Size = New System.Drawing.Size(168, 20)
Me.TextBox1.TabIndex = 2
Me.TextBox1.TabStop = False
Me.TextBox1.Text = ""
'
'Button1
'
Me.Button1.Location = New System.Drawing.Point(624, 584)
Me.Button1.Name = "Button1"
Me.Button1.TabIndex = 3
Me.Button1.Text = "All"
'
'Form4
'
Me.AutoScaleBaseSize = New System.Drawing.Size(5, 13)
Me.ClientSize = New System.Drawing.Size(848, 622)
Me.Controls.Add(Me.Button1)
Me.Controls.Add(Me.TextBox1)
Me.Controls.Add(Me.HScrollBar1)
Me.Controls.Add(Me.ScatterGraph1)
Me.Name = "Form4"
Me.Text = "Form4"
CType(Me.ScatterGraph1, System.ComponentModel.ISupportInitialize).EndInit()
Me.ResumeLayout(False)

End Sub

#End Region

Private Sub HScrollBar1_Scroll(ByVal sender As System.Object, ByVal e As System.Windows.Forms.ScrollEventArgs) Handles
HScrollBar1.Scroll
    Me.TextBox1.Text = String.Concat(CStr(HScrollBar1.Value()), " of 50")
    Dim i As Integer
    For i = 0 To 49
        Me.ScatterGraph1.Plots(i).Visible = False
    Next i
    Me.ScatterGraph1.Plots(HScrollBar1.Value - 1).Visible = True
End Sub

Private Sub Form4_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
    Me.ScatterGraph1.ClearData()
    Dim theplot As ScatterPlot
    Dim i As Integer
    For i = 1 To 50
        theplot = New ScatterPlot
        Me.ScatterGraph1.Plots.Add(theplot)
    Next i
    plotcount = 0
    MsgBox(dataplots.Count)
End Sub

Public Sub PassData(ByVal xdata() As Double, ByVal ydata() As Double)
    MsgBox("hello")
    dataplots(plotcount).PlotXY(xdata, ydata)
    Me.ScatterGraph1.Plots(plotcount).PlotXY(xdata, ydata)
    MsgBox("hmmm")
    If plotcount > 0 Then
        Me.ScatterGraph1.Plots(plotcount - 1).Visible = False
    End If
    plotcount = plotcount + 1
    MsgBox("bye")
    Me.pform.Button15.PerformClick()
End Sub

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
    Dim i As Integer
    For i = 0 To 49
        Me.ScatterGraph1.Plots(i).Visible = True
    Next i
End Sub
End Class

'-----
' <autogenerated>
'   This code was generated by Measurement Studio.
'   Runtime Version: 1.1.4322.573
'
'   Changes to this file may cause incorrect behavior and will be lost if
'   the code is regenerated.
' </autogenerated>
'-----

Option Strict Off
Option Explicit On

Imports NationalInstruments.VisaNS
Imports System
Imports System.Collections

```



```

Imports System.Diagnostics
Imports System.Globalization
Imports System.IO
Imports System.Text

Public Class VisaNSTask1

    Private Shared Sub SetTermChar(ByVal resourceName As MessageBasedSession, ByVal terminationCharacter As Integer)
        'This utility method will set the termination character attribute of a
        'serial instrument. If resourceName is not serial, then this method
        'does nothing.
        If (resourceName.HardwareInterfaceType = HardwareInterfaceType.Serial) Then
            Dim useTerminationCharacter As Boolean = (terminationCharacter > 0)
            Dim ss As SerialSession = CType(resourceName, SerialSession)
            If useTerminationCharacter Then
                ss.ReadTermination = SerialTerminationMethod.TerminationCharacter
            Else
                ss.ReadTermination = SerialTerminationMethod.None
            End If
            resourceName.TerminationCharacterEnabled = useTerminationCharacter
            resourceName.TerminationCharacter = CType(terminationCharacter, Byte)
        End If
    End Sub

    Private Shared Sub Write(ByVal resourceName As MessageBasedSession, ByVal command As String, ByVal terminationCharacter
As Integer)
        'Does a VISA Write
        SetTermChar(resourceName, terminationCharacter)
        resourceName.Write(command)
    End Sub

    Public Overridable Sub Run(ByVal command As Integer)
        Dim resourceName As MessageBasedSession = Nothing
        Try
            'Instantiates an Instrument Session through VISA
            resourceName = CType(ResourceManager.GetLocalManager.Open("ASRL1::INSTR", CType(4, AccessModes), 2000),
MessageBasedSession)

            Write(resourceName, CStr(Chr(command)), 0)

        Finally
            If (Not (resourceName) Is Nothing) Then
                CType(resourceName, IDisposable).Dispose()
            End If
        End Try
    End Sub
End Sub
End Class

```

## APPENDIX B

### MATLAB M FILE ROUTINES

Written by Prashant Mishra and Rama Ranganathan

```
function [out]=ady_findpeaks(in_trace,method,threshold);
%This function takes in a trace of photodiode responses and returns the
%peak values and the time at which the peak occurs. Sample rate is assumed
%to always be 10kHz (10 times per msec).
```

```
if nargin==1
    method='sobel';
    threshold=[];
end

[x,y]=size(in_trace);
xsamples=[1:x];
[peak_samples]=edge(in_trace,method,threshold);
[nspikes,junk]=size(find(peak_samples==1));

figure_num=10;

xlims=xsamples(find(peak_samples==1));

ydata=in_trace(xlims(1):xlims(1)+20)';
xdata=[xlims(1):xlims(1)+20];
[peaks(1),index]=max(ydata);
xpeak(1)=xdata(index);

for i=2:nspikes
    if xlims(i)-xlims(i-1)>20
        ydata=in_trace(xlims(i):xlims(i)+20)';
        xdata=[xlims(i):xlims(i)+20];
        [peaks(i),index]=max(ydata);
        xpeak(i)=xdata(index);
    end
end

xpeak=xpeak(find(xpeak~=0));
peaks=peaks(find(peaks~=0));

figure(figure_num)
plot(xsamples./10,in_trace);hold on
plot(xpeak./10,peaks,'-ro');hold off
out=[xpeak'./10 peaks'];
```

---

```
function [out,averaged_tail]=ady_findpeaks(in_trace,method,threshold);
%This function takes in a trace of photodiode responses and returns the
%peak values and the time at which the peak occurs. Sample rate is assumed
%to always be 10kHz (10 times per msec).
```

```
if nargin==1
    method='sobel';
    threshold=[];
end

[x,y]=size(in_trace);
xsamples=[1:x];
in_trace=in_trace-mean(in_trace(1:30));
[peak_samples]=edge(in_trace,method,threshold);
[nspikes,junk]=size(find(peak_samples==1));

figure_num=10;

xlims=xsamples(find(peak_samples==1));

ydata=in_trace(xlims(1):xlims(1)+20)';
xdata=[xlims(1):xlims(1)+20];
[peaks(1),index]=max(ydata);
xpeak(1)=xdata(index);

for i=2:nspikes
    if xlims(i)-xlims(i-1)>20
        ydata=in_trace(xlims(i):xlims(i)+20)';
        xdata=[xlims(i):xlims(i)+20];
        [peaks(i),index]=max(ydata);
        xpeak(i)=xdata(index);
    end
end

end
```

```

xpeak=xpeak(find(xpeak~=0));
peaks=peaks(find(peaks~=0));

figure (figure_num)
plot (xsamples./10,in_trace);hold on
plot (xpeak./10,peaks,'-ro');hold off
out=[xpeak'./10 peaks'];

averaged_tail=mean(peaks(end-5:end));

function alltrace(path);
%path='c:\temp'
files=dir(path)
numFiles = size(files,1);
str(1)={'temp'};
y(1)=1;

currentFile=3;
currentFigure=1;
while currentFile<numFiles+1
    f=figure(currentFigure);

    path1=[path,'\ ',files(currentFile).name]
    fid=fopen(path1);
    h = axes('Position',[0 0 1 1],'Visible','off');
    str(:)=[];
    str(1)={path1};
    k=1;
    cline=fgetl(fid);
    while strcmp(cline,"START")==0
        k=k+1;
        str(k)={cline};
        cline=fgetl(fid);
    end
    k=k+1;
    cline=fgetl(fid);
    sampleRate=str2num(cline);
    str(k)={cline};
    k=k+1;
    cline=fgetl(fid);
    numSamples=str2num(cline);
    str(k)={cline};
    k=k+1;
    cline=fgetl(fid);
    str(k)={cline};
    k=k+1;
    cline=fgetl(fid);
    str(k)={cline};
    text(.01,.99,str,'FontSize',6,'VerticalAlignment','top');
    axes('Position',[0.125 .695 0.37 .25]);
    x=1000./sampleRate:1000./sampleRate:1000.*numSamples./sampleRate;
    y(:)=[];
    for i = 1:numSamples
        cline=fgetl(fid);
        y(i)=str2num(cline);
    end
    p=plot(x,y);
    xlabel('msec');
    ylabel('pA');
    fclose(fid);

    currentFile=currentFile+1;
    if currentFile<numFiles+1
        path1=[path,'\ ',files(currentFile).name]
        fid=fopen(path1);
        %h = axes('Position',[0 0 1 1],'Visible','off');
        set(f,'CurrentAxes',h);
        str(:)=[];
        str(1)={path1};
        k=1;
        cline=fgetl(fid);
        while strcmp(cline,"START")==0
            k=k+1;
            str(k)={cline};
            cline=fgetl(fid);
        end
        k=k+1;
        cline=fgetl(fid);
        sampleRate=str2num(cline);
        str(k)={cline};
        k=k+1;
        cline=fgetl(fid);
        numSamples=str2num(cline);
        str(k)={cline};
        k=k+1;
        cline=fgetl(fid);

```

```

str(k)={cline};
k=k+1;
cline=fgetl(fid);
str(k)={cline};
text(.51,.99,str,'FontSize',6,'VerticalAlignment','top');
axes('Position',[0.625 .695 0.37 .25]);
x=1000./sampleRate:1000./sampleRate:1000.*numSamples./sampleRate;
y(:)=[];
for i = 1:numSamples
    cline=fgetl(fid);
    y(i)=str2num(cline);
end
p=plot(x,y);
xlabel('msec');
ylabel('pA');
fclose(fid);
end

currentFile=currentFile+1;
if currentFile<numFiles+1
    path1=[path,'\ ',files(currentFile).name]
    fid=fopen(path1);
    %h = axes('Position',[0 0 1 1],'Visible','off');
    set(f,'CurrentAxes',h);
    str(:)=[];
    str(1)={path1};
    k=1;
    cline=fgetl(fid);
    while strcmp(cline,"START")==0
        k=k+1;
        str(k)={cline};
        cline=fgetl(fid);
    end
    k=k+1;
    cline=fgetl(fid);
    sampleRate=str2num(cline);
    str(k)={cline};
    k=k+1;
    cline=fgetl(fid);
    numSamples=str2num(cline);
    str(k)={cline};
    k=k+1;
    cline=fgetl(fid);
    str(k)={cline};
    text(.01,.66,str,'FontSize',6,'VerticalAlignment','top');
    axes('Position',[0.125 .365 0.37 .25]);
    x=1000./sampleRate:1000./sampleRate:1000.*numSamples./sampleRate;
    y(:)=[];
    for i = 1:numSamples
        cline=fgetl(fid);
        y(i)=str2num(cline);
    end
    p=plot(x,y);
    xlabel('msec');
    ylabel('pA');
    fclose(fid);
end

currentFile=currentFile+1;
if currentFile<numFiles+1
    path1=[path,'\ ',files(currentFile).name]
    fid=fopen(path1);
    %h = axes('Position',[0 0 1 1],'Visible','off');
    set(f,'CurrentAxes',h);
    str(:)=[];
    str(1)={path1};
    k=1;
    cline=fgetl(fid);
    while strcmp(cline,"START")==0
        k=k+1;
        str(k)={cline};
        cline=fgetl(fid);
    end
    k=k+1;
    cline=fgetl(fid);
    sampleRate=str2num(cline);
    str(k)={cline};
    k=k+1;
    cline=fgetl(fid);
    numSamples=str2num(cline);
    str(k)={cline};
    k=k+1;
    cline=fgetl(fid);
    str(k)={cline};
    k=k+1;
    cline=fgetl(fid);
    str(k)={cline};
end

```

```

text(.51,.66,str,'FontSize',6,'VerticalAlignment','top');
axes('Position',[0.625 .365 0.37 .25]);
x=1000./sampleRate:1000./sampleRate:1000.*numSamples./sampleRate;
y(:)=[];
for i = 1:numSamples
    cline=fgetl(fid);
    y(i)=str2num(cline);
end
p=plot(x,y);
xlabel('msec');
ylabel('pA');
fclose(fid);
end

currentFile=currentFile+1;
if currentFile<numFiles+1
    path1=[path,'\ ',files(currentFile).name]
    fid=fopen(path1);
    %h = axes('Position',[0 0 1 1],'Visible','off');
    set(f,'CurrentAxes',h);
    str(:)=[];
    str(1)={path1};
    k=1;
    cline=fgetl(fid);
    while strcmp(cline,"START")==0
        k=k+1;
        str(k)={cline};
        cline=fgetl(fid);
    end
    k=k+1;
    cline=fgetl(fid);
    sampleRate=str2num(cline);
    str(k)={cline};
    k=k+1;
    cline=fgetl(fid);
    numSamples=str2num(cline);
    str(k)={cline};
    k=k+1;
    cline=fgetl(fid);
    str(k)={cline};
    k=k+1;
    cline=fgetl(fid);
    str(k)={cline};
    text(.01,.32,str,'FontSize',6,'VerticalAlignment','top');
    axes('Position',[0.125 .033 0.37 .25]);
    x=1000./sampleRate:1000./sampleRate:1000.*numSamples./sampleRate;
    y(:)=[];
    for i = 1:numSamples
        cline=fgetl(fid);
        y(i)=str2num(cline);
    end
    p=plot(x,y);
    xlabel('msec');
    ylabel('pA');
    fclose(fid);
end

currentFile=currentFile+1;
if currentFile<numFiles+1
    path1=[path,'\ ',files(currentFile).name]
    fid=fopen(path1);
    %h = axes('Position',[0 0 1 1],'Visible','off');
    set(f,'CurrentAxes',h);
    str(:)=[];
    str(1)={path1};
    k=1;
    cline=fgetl(fid);
    while strcmp(cline,"START")==0
        k=k+1;
        str(k)={cline};
        cline=fgetl(fid);
    end
    k=k+1;
    cline=fgetl(fid);
    sampleRate=str2num(cline);
    str(k)={cline};
    k=k+1;
    cline=fgetl(fid);
    numSamples=str2num(cline);
    str(k)={cline};
    k=k+1;
    cline=fgetl(fid);
    str(k)={cline};
    k=k+1;
    cline=fgetl(fid);
    str(k)={cline};
    text(.51,.32,str,'FontSize',6,'VerticalAlignment','top');
    axes('Position',[0.625 .03 0.37 .25]);
    x=1000./sampleRate:1000./sampleRate:1000.*numSamples./sampleRate;

```

```

        y(:)=[];
        for i = 1:numSamples
            cline=fgetl(fid);
            y(i)=str2num(cline);
        end
        p=plot(x,y);
        xlabel('msec');
        ylabel('pA');
        fclose(fid);
    end

    set(f,'PaperPosition',[0.4 0.4 7.7 10.2]);
    set(f,'PaperOrientation','portrait');
    currentFigure=currentFigure+1;
    currentFile=currentFile+1;
end

function alltrace(path);
%path='c:\temp'
files=dir(path);
numFiles = size(files,1);
str(1)={'temp'};
strone(1)={'temp'};
strtwo(1)={'temp'};
strthree(1)={'temp'};
y(1)=1;

currentFile=1;
currentFigure=1;
while currentFile<numFiles+1
    f=figure(currentFigure);
    h = axes('Position',[0 0 1 1],'Visible','off');
    currentPosition=1;

    while currentPosition < 7
        if currentFile > numFiles
            break
        end
        if files(currentFile).isdir==0
            path1=[path,'\ ',files(currentFile).name]
            w=strfind(files(currentFile).name,'flash50');
            if w>0
                fid=fopen(path1);
                strone(:)=[];
                strtwo(:)=[];
                strthree(:)=[];
                strone(1)=path1;
                for k=2:21
                    cline=fgetl(fid);
                    strone(k)=cline;
                end
                strtwo(1)='';
                for k=2:21
                    cline=fgetl(fid);
                    strtwo(k)=cline;
                end
                strthree(1)='';
                for k=2:11
                    cline=fgetl(fid);
                    strthree(k)=cline;
                end

                if currentPosition==1
                    text(.01,.99,strone,'FontSize',6,'VerticalAlignment','top','Parent',h);
                    text(.11,.99,strtwo,'FontSize',6,'VerticalAlignment','top','Parent',h);
                    text(.21,.99,strthree,'FontSize',6,'VerticalAlignment','top','Parent',h);
                end
                if currentPosition==2
                    text(.51,.99,strone,'FontSize',6,'VerticalAlignment','top','Parent',h);
                    text(.61,.99,strtwo,'FontSize',6,'VerticalAlignment','top','Parent',h);
                    text(.71,.99,strthree,'FontSize',6,'VerticalAlignment','top','Parent',h);
                end
                if currentPosition==3
                    text(.01,.66,strone,'FontSize',6,'VerticalAlignment','top','Parent',h);
                    text(.11,.66,strtwo,'FontSize',6,'VerticalAlignment','top','Parent',h);
                    text(.21,.66,strthree,'FontSize',6,'VerticalAlignment','top','Parent',h);
                end
                if currentPosition==4
                    text(.51,.66,strone,'FontSize',6,'VerticalAlignment','top','Parent',h);
                    text(.61,.66,strtwo,'FontSize',6,'VerticalAlignment','top','Parent',h);
                    text(.71,.66,strthree,'FontSize',6,'VerticalAlignment','top','Parent',h);
                end
                if currentPosition==5
                    text(.01,.32,strone,'FontSize',6,'VerticalAlignment','top','Parent',h);
                    text(.11,.32,strtwo,'FontSize',6,'VerticalAlignment','top','Parent',h);
                    text(.21,.32,strthree,'FontSize',6,'VerticalAlignment','top','Parent',h);
                end
                if currentPosition==6

```

```

text(.51,.32,strone,'FontSize',6,'VerticalAlignment','top','Parent',h);
text(.61,.32,strtwo,'FontSize',6,'VerticalAlignment','top','Parent',h);
text(.71,.32,strthree,'FontSize',6,'VerticalAlignment','top','Parent',h);
end
else
fid=fopen(path1);
set(f,'CurrentAxes',h);
str(:)=[];
str(1)={path1};
k=1;
cline=fgetl(fid);
while strcmp(cline,"START")==0
    k=k+1;
    str(k)={cline};
    cline=fgetl(fid);
end
k=k+1;
cline=fgetl(fid);
sampleRate=str2num(cline);
str(k)={cline};
k=k+1;
cline=fgetl(fid);
numSamples=str2num(cline);
str(k)={cline};
k=k+1;
cline=fgetl(fid);
str(k)={cline};
k=k+1;
cline=fgetl(fid);
str(k)={cline};
k=k+1;
cline=fgetl(fid);
str(k)={cline};

if currentPosition==1
text(.01,.99,str,'FontSize',6,'VerticalAlignment','top');
axes('Position',[0.125 .695 0.37 .25]);
end
if currentPosition==2
text(.51,.99,str,'FontSize',6,'VerticalAlignment','top');
axes('Position',[0.625 .695 0.37 .25]);
end
if currentPosition==3
text(.01,.66,str,'FontSize',6,'VerticalAlignment','top');
axes('Position',[0.125 .365 0.37 .25]);
end
if currentPosition==4
text(.51,.66,str,'FontSize',6,'VerticalAlignment','top');
axes('Position',[0.625 .365 0.37 .25]);
end
if currentPosition==5
text(.01,.32,str,'FontSize',6,'VerticalAlignment','top');
axes('Position',[0.125 .03 0.37 .25]);
end
if currentPosition==6
text(.51,.32,str,'FontSize',6,'VerticalAlignment','top');
axes('Position',[0.625 .03 0.37 .25]);
end

x=1000./sampleRate:1000./sampleRate:1000.*numSamples./sampleRate;
y(:)=[];
for i = 1:numSamples
    cline=fgetl(fid);
    y(i)=str2num(cline);
end
p=plot(x,y);
xlabel('msec');
ylabel('pA');
fclose(fid);
end
currentPosition=currentPosition+1;
end
currentFile=currentFile+1;

end

set(f,'PaperPosition',[0.4 0.4 7.7 10.2]);
set(f,'PaperOrientation','portrait');
currentFigure=currentFigure+1;
end

function [peaks, trace] = peakread(filename)
% [peaks, trace] = peakread(filename)

trace = dlmread(filename, ' ',20,0);
peaks = trace(255:330:10000);

path='C:\Documents and Settings\Adrienne Hahn\My Documents\CaMga
Images\Photodiode\Titrations\080904\CA\1.5_Ca\20040809113916_left_cell 1.txt'
%files=dir(path);

```

```

numFiles = size(files,1);
str(1)={'temp'};
y(1)=1;

for i=3:3
    f=figure(i-2);
    fid=fopen(path);
    h = axes('Position',[0 0 1 1],'Visible','off');
    str(:)=[];
    str(1)={path};
    k=1;
    cline=fgetl(fid);
    while strcmp(cline,'START')==0
        k=k+1;
        str(k)={cline};
        cline=fgetl(fid);
    end
    k=k+1;
    cline=fgetl(fid);
    sampleRate=str2num(cline);
    str(k)={cline};
    k=k+1;
    cline=fgetl(fid);
    numSamples=str2num(cline);
    str(k)={cline};
    k=k+1;
    cline=fgetl(fid);

    str(k)={cline};
    k=k+1;
    cline=fgetl(fid);

    str(k)={cline};
    text(.015,.76,str,'FontSize',12);
    axes('Position',[.2 .1 .75 .71]);

    x=1000./sampleRate:1000./sampleRate:1000.*numSamples./sampleRate;
    y(:)=[];
    for i = 1:numSamples
        cline=fgetl(fid);
        y(i)=str2num(cline);
    end
    p=plot(x,y);
    xlabel('msec');
    ylabel('pA');
    fclose(fid);
    set(f,'PaperPosition',[0.4 0.4 10 7.5]);
    set(f,'PaperOrientation','landscape');

end

function [out,averaged_tail]=ady_findpeaks(in_trace,method,threshold);
%This function takes in a trace of photodiode responses and returns the
%peak values and the time at which the peak occurs. Sample rate is assumed
%to always be 10kHz (10 times per msec).

if nargin==1
    method='sobel';
    threshold=[];
end

[x,y]=size(in_trace);
xsamples=[1:x];
in_trace=in_trace-mean(in_trace(1:30));
[peak_samples]=edge(in_trace,method,threshold);
[nsamples,junk]=size(find(peak_samples==1));

figure_num=10;

xlims=xsamples(find(peak_samples==1));

ydata=in_trace(xlims(1):xlims(1)+20);
xdata=[xlims(1):xlims(1)+20];
[peaks(1),index]=max(ydata);
xpeak(1)=xdata(index);

for i=2:nsamples
    if xlims(i)-xlims(i-1)>20
        ydata=in_trace(xlims(i):xlims(i)+20);
        xdata=[xlims(i):xlims(i)+20];
        [peaks(i),index]=max(ydata);
        xpeak(i)=xdata(index);
    end
end
end

```



```
xpeak=xpeak(find(xpeak~=0));
peaks=peaks(find(peaks~=0));

figure (figure_num)
plot(xsamples./10,in_trace);hold on
plot(xpeak./10,peaks,'-ro');hold off
out=[xpeak'./10 peaks'];

averaged_tail=mean(peaks(end-5:end));
```

## BIBLIOGRAPHY

- Acharya, J. K. and e. al. (1997). "InsP<sub>3</sub> Receptor is Essential for Growth and Differentiation but Not for Vision in *Drosophila*." *Neuron* 18: 881-887.
- Adamski, F. M., M.-Y. Zhu, et al. (1998). "Interaction of Eye Protein Kinase C and INAD in *Drosophila*." *Journal of Biological Chemistry* 273(28): 17713-17719.
- Alloway, P. G., L. Howard, et al. (2000). "The Formation of Stable Rhodopsin-Arrestin Complexes Induces Apoptosis and Photoreceptor Cell Degeneration." *Neuron* 28: 129-138.
- Bahner, M., P. Sander, et al. (2000). "The Visual G Protein of Fly Photoreceptors Interacts with the PDZ Domain Assembled INAD Signaling Complex via Direct Binding of Activated G $\alpha$ q to Phospholipase C $\beta$ ." *JBC* 275(4): 2901-2904.
- Baird, G. S., D. A. Zacharias, et al. (1999). "Circular permutation and receptors insertion within green fluorescent proteins." *Proceedings of the National Academy of Sciences* 96: 11241-11246.
- Benham, C. D., J. B. Davis, et al. (2002). "Vanilloid and TRP channels: a family of lipid-gated cation channels." *Neuropharmacology* 42: 873-888.
- Bloomquist, B., R. Shortridge, et al. (1988). "Isolation of a putative phospholipase C gene of *Drosophila*, norpA, and its role in phototransduction." *Cell* 54(5): 723-33.
- Bray, D., M. D. Levin, et al. (1998). "Receptor clustering as a cellular mechanism to control sensitivity." *Nature* 393: 85-88.
- Brenman, J. E., D. S. Chao, et al. (1996). "Interaction of nitric oxide synthase with the postsynaptic density protein PSD-95 and  $\alpha$ 1-syntrophin mediated by PDZ domains." *Cell* 84: 757-767.
- Carafoli, E. (2002). "Calcium signaling: A tale for all seasons." *PNAS* 99(3): 1115-1122.
- Chevesich, J., A. J. Kreuz, et al. (1997). "Requirement for the PDZ Domain Protein, INAD, for Localization of the TRP Store-Operated Channel to a Signaling Complex." *Neuron* 18: 95-105.
- Chyb, S., P. Raghu, et al. (1999). "Polyunsaturated fatty acids activate the *Drosophila* light-sensitive channels TRP and TRPL." *Nature* 397: 255-259.

- Clapham, D. E. (2003). "TRP channels as cellular sensors." *Nature* 426: 517-524.
- Dolph, P. J. and E. al. (1993). "Arrestin Function in Inactivation of G Protein-Coupled Receptor Rhodopsin *In Vivo*." *Science* 260: 1910-1916.
- Estacion, M., W. G. Sinkins, et al. (2001). "Regulation of *Drosophila* transient receptor potential-like (TrpL) channels by phospholipase C-dependent mechanisms." *Journal of Physiology* 530(1): 1-19.
- Fein, A. (2003). "Inositol 1,4,5-trisphosphate-induced Calcium Release Is Necessary for Generating the Entire Light Response of *Limulus* Ventral Photoreceptors." *Journal of General Physiology* 121: 441-449.
- Goldberg, J. H., G. Tamas, et al. (2003). "Calcium microdomains in Aspiny Dendrites." *Neuron* 40: 807-821.
- Grootjans, J. J., G. Reekmans, et al. (2000). "Syntenin-Syndecan Binding Requires Syndecan-Syntenin and the Co-operation of Both PDZ Domains of Syntenin." *Journal of Biological Chemistry* 275(26): 19933-19941.
- Hardie, R. C. (1991). "Whole-Cell Recordings of the Light Induced Current in Dissociated *Drosophila* Photoreceptors: Evidence for Feedback by Calcium Permeating the Light-Sensitive channels." *Proceedings of the Royal Society of London : Biological Sciences* 245: 203-210.
- Hardie, R. C. (2001). "Phototransduction in *Drosophila melanogaster*." *The Journal of Experimental Biology* 204: 3403-3409.
- Hardie, R. C. (2003). "Regulation of TRP Channels Via Lipid Second Messengers." *Annual Review in Physiology* 65: 735-59.
- Hardie, R. C. and e. al. (1993). "Protein Kinase C is required for light adaptation in *Drosophila* photoreceptors." *Nature* 363: 634-637.
- Hardie, R. C. and e. al. (2001). "Calcium Influx via TRP Channels Is Required to Maintain PIP<sub>2</sub> Levels in *Drosophila*." *Neuron* 30: 149-159.
- Hardie, R. C. and B. Minke (1992). "The *trp* gene is essential for a light-activated Ca<sup>2+</sup> channel in *Drosophila* photoreceptors." *Neuron* 8(4): 643-51.
- Hardie, R. C. and B. Minke (1994). "Calcium-dependent Inactivation of Light-sensitive Channels in *Drosophila* Photoreceptors." *Journal of General Physiology* 103: 409-427.

- Hardie, R. C. and P. Raghu (1998). "Activation of heterologously expressed *Drosophila* TRPL channels:  $\text{Ca}^{2+}$  is not required and  $\text{InsP}_3$  is not sufficient." *Cell Calcium* 24(3): 153-163.
- Hardie, R. C. and P. Raghu (2001). "Visual Transduction in *Drosophila*." *Nature* 413: 186-193.
- Harris, B. Z. and W. A. Lim (2001). "Mechanism and role of PDZ domains in signaling complex assembly." *Journal of Cell Science* 114: 3219-3231.
- Harrison, S. C. (1996). "Peptide-surface association: the case of PDZ and PTB domains." *Cell* 86: 341-343.
- Harteneck, C. and T. D. Plant (2000). "From worm to man: three subfamilies of TRP channel." *Trends in Neuroscience* 23: 159-166.
- Haug-Collet, K. and e. al. (1999). "Cloning and Characterization of a Potassium-dependent Sodium/Calcium Exchanger in *Drosophila*." *J. Cell Biology* 147(3): 659-669.
- Henderson, S. R., H. Reuss, et al. (2000). "Single photon responses in *Drosophila* photoreceptors and their regulation by  $\text{Ca}^{2+}$ ." *Journal of Physiology* 524(Pt. 1): 179-194.
- Hofmann, T., A. G. Obukhov, et al. (1999). "Direct activation of human TRPC6 and TRPC3 channels by diacylglycerol." *Nature* 397: 259-263.
- Howard, J., B. Blakeslee, et al. (1987). "The Intracellular Pupil Mechanism and Photoreceptor Signal: Noise Ratios in the Fly *Lucilia cuprina*." *Proceedings of the Royal Society of London : Biological Sciences* 231(1265): 415-435.
- Huber, A., P. Sander, et al. (1996). "The transient receptor potential protein (Trp), a putative store-operated  $\text{Ca}^{2+}$  channel essential for phosphoinositide-mediated photoreception, forms a signaling complex with NorpA, InaC, and InaD." *EMBO* 15(24): 7036-7045.
- Huber, A., P. Sander, et al. (1996). "Phosphorylation of the *InaD* Gene Product, A Photoreceptor Membrane Protein Required for Recovery of Visual Excitation." *J. Biol. Chem.* 271(20): 11710-11717.
- Hung, A. L. and M. Sheng (2002). "PDZ Domains: Structural Modules for Protein Complex Assembly." *Journal of Biological Chemistry* 277(8): 5699-5702.

- Juusola, M. and R. C. Hardie (2001). "Light Adaptation in *Drosophila* Photoreceptors: I. Response Dynamics and Signaling Efficiency at 25 C." *Journal of General Physiology* 117(1): 3-25.
- Kimple, M. E., D. P. Siderovski, et al. (2001). "Functional relevance of the disulfide-linked complex of the N-terminal PDZ domain of InaD with NorpA." *EMBO Journal* 20(16): 4414-4422.
- Kiselev, A. and e. al. (2000). "A Molecular Pathway for Light-Dependent Photoreceptor Apoptosis in *Drosophila*." *Neuron* 28(1): 139-152.
- Kosloff, M. and e. al. (2003). "Regulation of light-dependent Gq $\alpha$  translocation and morphological changes in fly photoreceptors." *EMBO* 22(3): 459-468.
- Kumar, R. and B.-H. Shieh (2001). "The Second PDZ Domain of INAD Is a Type I Domain Involved in Binding to Eye Protein Kinase C." *J. Biol. Chem.* 276(27): 24971-24977.
- Lee, S.-J. and C. Montell (2001). "Regulation of the Rhodopsin Protein Phosphatase RDGC, through Interaction with Calmodulin." *Neuron* 32: 1097-1106.
- Li, H.-S. and C. Montell (2000). "TRP and the PDZ Protein, INAD, Form the Core Complex Required for Retention of the Signalplex in *Drosophila* Photoreceptors Cells." *Journal of Cell Biology* 150(6): 1411-1421.
- Lim, W. A. (2002). "The modular logic of signaling proteins: building allosteric switches from simple binding proteins." *Current Opinion in Structural Biology* 12: 61-68.
- Liu, M., L. L. Parker, et al. (2000). "Reversible Phosphorylation of the Signal Transduction Complex in *Drosophila* Photoreceptors." *J. Biol. Chem.* 275(16): 12194-12199.
- Masai, I., A. Okazaki, et al. (1993). "*Drosophila* retinal degeneration A gene encodes an eye-specific diacylglycerol kinase with cysteine-rich zinc-finger motifs and ankyrin repeats." *Proceedings of the National Academy of Sciences* 90: 11157-11161.
- Migaud, M., P. Charlesworth, et al. (1998). "Enhanced long-term potentiation and impaired learning in mice with mutant postsynaptic density-95 protein." *Nature* 396: 433-439.
- Montell, C. (1999). "Visual Transduction in *Drosophila*." *Ann. Rev. Cell Dev. Biol.* 15: 231-268.

- Naisbitt, S., E. Kim, et al. (1999). "Shank, a novel family of postsynaptic density proteins that binds to the NMDA receptor/PSD-95/GKAP complex and cortactin." *Neuron* 23: 569-582.
- Niemeyer, B. A. and e. al. (1996). "The *Drosophila* Light-Activated Conductance Is Composed of the Two Channels TRP and TRPL." *Cell* 85: 651-659.
- Oberwinkler, J. and D. G. Stavenga (2000). "Calcium imaging demonstrates colocalization of calcium influx and extrusion in fly photoreceptors." *Proceedings of the National Academy of Sciences* 97(15): 8578-8583.
- Oberwinkler, J. and D. G. Stavenga (2000). "Calcium Transients in the Rhabdomeres of Dark- and Light-Adapted Fly Photoreceptors." *J. Neurosci.* 20(5): 1701-1709.
- Pak, W. C., J. Grossfield et al. (1970). "Mutants of the Visual Pathway of *Drosophila melanogaster*." *Nature* 227(257): 518-520.
- Park, S., A. Zarrinpar, et al. (2003). "Rewiring MAP Kinase Pathways Using Alternative Mechanisms." *Science* 299: 1061-1064.
- Peretz, A. (1994). "Genetic Dissection of Light-induced Calcium Influx into *Drosophila* Photoreceptors." *J. Gen. Physiol.* 104: 1057-1077.
- Phillips, A. M., A. Bull, et al. (1992). "Identification of A *Drosophila* Gene Encoding a Calmodulin-Binding Protein with Homology to the *trp* Phototransduction Gene." *Neuron* 8: 631-642.
- Porter, J. A., M. Yu, et al. (1993). "Dependence of Calmodulin Localization in the Retina on the NINAC Unconventional Myosin." *Science* 262: 1038-1042.
- Postma, M., J. Oberwinkler, et al. (1999). "Does  $\text{Ca}^{2+}$  Reach Millimolar Concentrations after Single Photon Absorption in *Drosophila* Photoreceptor Microvilli?" *Biophys j* 77: 1811-1823.
- Raghu, P. and e. al. (2000). "Normal Phototransduction in *Drosophila* Photoreceptors Lacking an  $\text{InsP}_3$  Receptor Gene." *Molec. and Cell. Neurosci.* 15: 429-445.
- Raghu, P., K. Usher, et al. (2000). "Constitutive Activity of the Light -Sensitive Channels TRP and TRPL in the *Drosophila* Diacylglycerol Kinase Mutant, *rdgA*." *Neuron* 26: 169-179.
- Ranganathan, R., and C. F. Stevens (1995). "Arrestin Binding Determines the Rate of Inactivation of the G Protein-Coupled Receptor Rhodopsin *In Vivo*." *Cell* 81: 841-848.

- Ranganathan, R., B. J. Bacskaï, et al. (1994). "Cytosolic Calcium Transients: Spatial Localization and Role in *Drosophila* Photoreceptor Cell Function." *Neuron* 13: 837-848.
- Ranganathan, R., G. L. Harris, et al. (1991). "A *Drosophila* mutant defective in extracellular calcium-dependent photoreceptor deactivation and rapid desensitization." *Nature* 354: 230-232.
- Reuss, H., M. H. Mojet, et al. (1997). "*In Vivo* Analysis of the *Drosophila* Light-Sensitive Channels, TRP, and TRPL." *Neuron* 19: 1249-1259.
- Ron, D. and M. G. Kazanietz (1999). "New insights into the regulation of protein kinase C and novel phorbol ester receptors." *FASEB Journal* 13: 1658-1676.
- Scott, K., Y. Sun, et al. (1997). "Calmodulin Regulation of *Drosophila* Light-Activated Channels and Receptor Function Mediates Termination of the Light Response *In Vivo*." *Cell* 91: 375-383.
- Scott, K. and C. Zuker (1997). "Lights out: deactivation of the phototransduction cascade." *Trends in Biochemical Sciences* 22(9): 350-4.
- Scott, K. and C. Zuker (1998). "Assembly of the *Drosophila* phototransduction cascade into a signaling complex shapes elementary responses." *Nature* 395(6704): 805-808.
- Shieh, B.-H. and B. Niemeyer (1995). "A Novel Protein Encoded by the *InaD* Gene Regulates Recovery of Visual Transduction in *Drosophila*." *Neuron* 14: 201-210.
- Shieh, B.-H. and M.-Y. Zhu (1996). "Regulation of the TRP  $\text{Ca}^{2+}$  Channel by INAD in *Drosophila* Photoreceptors." *Neuron* 16: 991-998.
- Shieh, B.-H., M.-Y. Zhu, et al. (1997). "Association of INAD with NORPA is essential for controlled activation and deactivation of *Drosophila* phototransduction *in vivo*." *PNAS* 94: 12682-12687.
- Smith, D. P. and e. al. (1991). "Photoreceptor Deactivation and Retinal Degeneration Mediated by a Photoreceptor-Specific Protein Kinase C." *Science* 254: 1478-1484.
- Songyang, Z. and A. S. Fanning (1997). "Recognition of unique carboxy-terminal motifs by distinct PDZ domains." *Science* 275: 73-77.
- Tsunoda, S. and e. al. (1997). "A multivalent PDZ-domain protein assembles signalling complexes in a G-protein-coupled cascade." *Nature* 388: 243-249.

- Tsunoda, S., Y. Sun, et al. (2001). "Independent Anchoring and Assembly Mechanisms of INAD Signaling Complexes in *Drosophila* Photoreceptors." *J. Neurosci.* 21(1): 150-158.
- van Huizen, R., K. Miller, et al. (1998). "Two distantly positioned PDZ domains mediate multivalent INAD-phospholipase C interactions essential for G protein-coupled signaling." *EMBO* 17(8): 2285-2297.
- Vinos, J., K. Jalink, et al. (1997). "A G protein-coupled receptor phosphatase required for rhodopsin function." *Science* 277(5326): 687-90.
- Voets, T., and B. Nilius (2003). "TRPs Make Sense." *Journal of Membrane Biology* 192: 1-8.
- Wes, P. D., X.-Z. S. Xu, et al. (1999). "Termination of phototransduction requires binding of the NINAC myosin III and the PDZ protein INAD." *Nature Neuroscience* 2(5): 447-453.
- Wu, C.-F. and W. L. Pak (1978). "Light-Induced Voltage Noise in the Photoreceptor of *Drosophila melanogaster*." *Journal of General Physiology* 71: 249-268.
- Xu, X.-Z. S. and e. al. (1998). "Retinal Targets for Calmodulin Include Proteins Implicated in Synaptic Transmission." *J. Biol. Chem.* 273(47): 31297-31307.
- Xu, X.-Z. S., F. Chien, et al. (2000). "TRP $\gamma$ , a *Drosophila* TRP-Related Subunit, Forms a Regulated Cation Channel with TRPL." *Neuron* 26: 647-657.
- Xu, X.-Z. S., A. Choudhury, et al. (1998). "Coordination of an Array of Signaling Proteins through Homo- and Heteromeric Interactions Between PDZ Domains and Target Proteins." *Journal of Cell Biology* 142(2): 545-555.
- Zhang, M. and W. Wang (2003). "Organization of Signaling Complexes by PDZ-Domain Scaffold Proteins." *Accounts of Chemical Research* 36(7): 530-538.
- Zhang, Q., J. Fan, et al. (2001). "Interdomain Chaperoning between PDS-95, Dlg, and Zo-1 (PDZ) Domains of Glutamate Receptor-interacting Proteins." *Journal of Biological Chemistry* 276(46): 43216-43220.



## VITAE

Adrienne R. Hahn was born in Provo, Utah, on July 10, 1974, the daughter of Gayle Hildred Robinson and John Norman Hall. After completing her work at Layton High School in Layton, Utah in 1992, she entered the University of Utah at Salt Lake City, Utah where she received the degree of Bachelor of Science with a major in biology and minors in chemistry and creative writing in June, 1996. During her time as an undergraduate, she participated in the Howard Hughes Undergraduate Research Program and worked in the laboratories of Dr. Neil Bastien of the division of Infectious Diseases at the University of Utah Hospital and Dr. Doju Yoshikami of the department of Biology of the University of Utah from 1993 to 1995. She also worked as a fish biologist in the U.S. Forest Service during the summer of 1995. She worked as an assistant animal technician at the University of Utah Animal Resources Center for about a year until graduation in 1996. During the following two years after graduation, she worked as a laboratory technician at the University of Utah Biochemistry department in the laboratory of Dr. Venki Ramakrishnan. In June, 1998 she entered the Graduate School of Biomedical Sciences at the University of Texas Southwestern Health Sciences Center at Dallas, and joined the laboratory of Dr. Rama Ranganathan. She was awarded the degree of Doctor of Philosophy in Molecular Biophysics in December, 2004. In 1997, she married Randy Roy Hahn in Salt Lake City, Utah. She is expecting their first child, a son, in December, 2004.

Permanent Address: 420 Orchard Hill Dr.  
Cedar Hill, TX 75104