

## Entwurf einer Software für ein Femtosekunden-Lasermikroskop und Realisierung mit LabVIEW™ [Fachartikel VIP 2003]

Matthias Rother, Prof. Dr.-Ing. Jörg Müller  
 Fachhochschule Jena Fachbereich Elektrotechnik / Informationstechnik

### Kurzfassung

Bilderzeugende Untersuchungen an Gewebe ohne invasiven Eingriff spielen in der heutigen Medizin eine immer größer werdende Rolle. Dies ist mit einem Femtosekunden-Lasermikroskop möglich. Mit dem Femtosekunden-Lasermikroskop ist es ebenfalls möglich gezielte Schnitte an Chromosomen oder Mitochondrien mit einer Schnittbreite von 110 nm durchzuführen.

Die Steuerungssoftware wurde in UML 2.0 modelliert. Hierzu wurde eine CASE-Tool eingesetzt, mit dem verschiedene Arten von Diagrammen erstellt wurden (Klassendiagramm, Aktivitätendiagramm, usw.).

Zur Realisierung dieses Systems wurde für die Steuerungssoftware die Programmiersprache G von National Instruments™ eingesetzt. Dies wurde getan, weil sich mit LabVIEW™ messtechnische Software relativ leicht und schnell erstellen lässt.

Durch die Realisierung mit LabVIEW™ verkürzt sich die Entwicklungszeit. LabVIEW™ bietet eine modulare Struktur was sich als vorteilhaft für kommende Änderungen bzw. Erweiterungen der Software auswirkt.

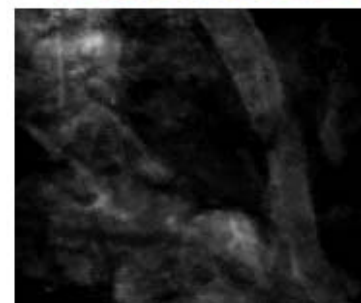
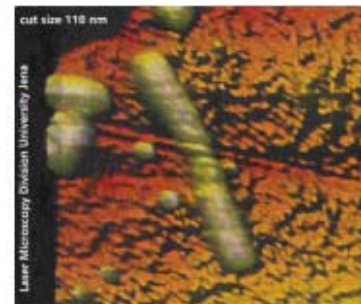
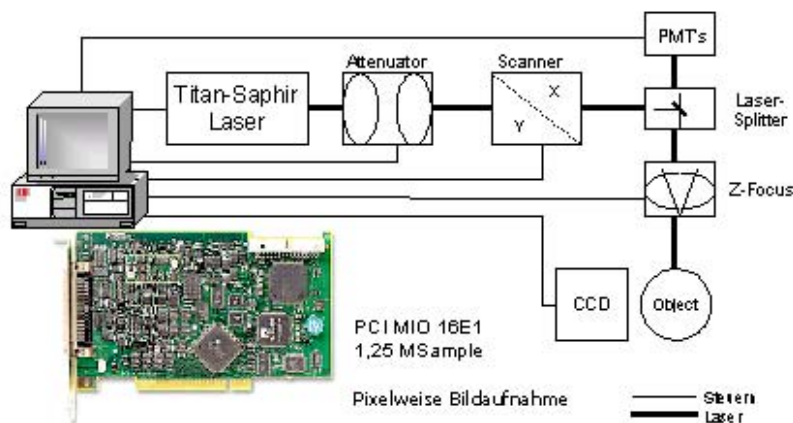


Bild 1: Prinzipdarstellung Lasermikroskop

## **abstract**

Picture producing examinations to human tissue without invasive intervention play a role getting bigger and bigger in today's medicine. This is possible with a femto-seconds laser microscope. It is also possibly specific slice of 110 nm with the femto-seconds laser microscope to chromosomes or mitochondria with a cut breadth to carry out.

The control software was modelled in UML 2.0. To it a CASE tool with which different manners were made by diagrams was used (class diagram, activity diagram, etc.).

For the realization of this system the programming language G of National Instruments was used for the control software. This was done because can be made metrological software relatively easily and fast with LabVIEW™.

The development time is shortened by the realization with LabVIEW™ and what has an effect as advantageous for coming changes or expansions of the software offers a modular structure.

## **Aufgabe**

Die Aufgabe bei diesem Projekt bestand darin eine Software zur Ansteuerung eines Lasermikroskops zu entwickeln. Die Software sollte jeweils die einzelnen Komponenten des Lasermikroskops steuern, die aufgezeichneten Daten in Form eines Graustufen-Bildes darstellen und diese Bilder ebenfalls archivieren können.

Sie sollte so flexibel sein, dass Anpassungen, die sich aus Änderungen am Gerät ergeben können, einfach umgesetzt werden können.

## **Entwurf**

Durch die Forderung der flexibeln Gestaltung des Gerätes wurde die Software in Form von Objekten zu realisieren. Dadurch ist eine leichte Austauschbarkeit, z.B. für einen neuen Hardwaretreiber, gegeben.

Durch die Analyse der Kommunikationsbeziehungen der einzelnen Objekte untereinander ergab sich folgendes Objektnetz.

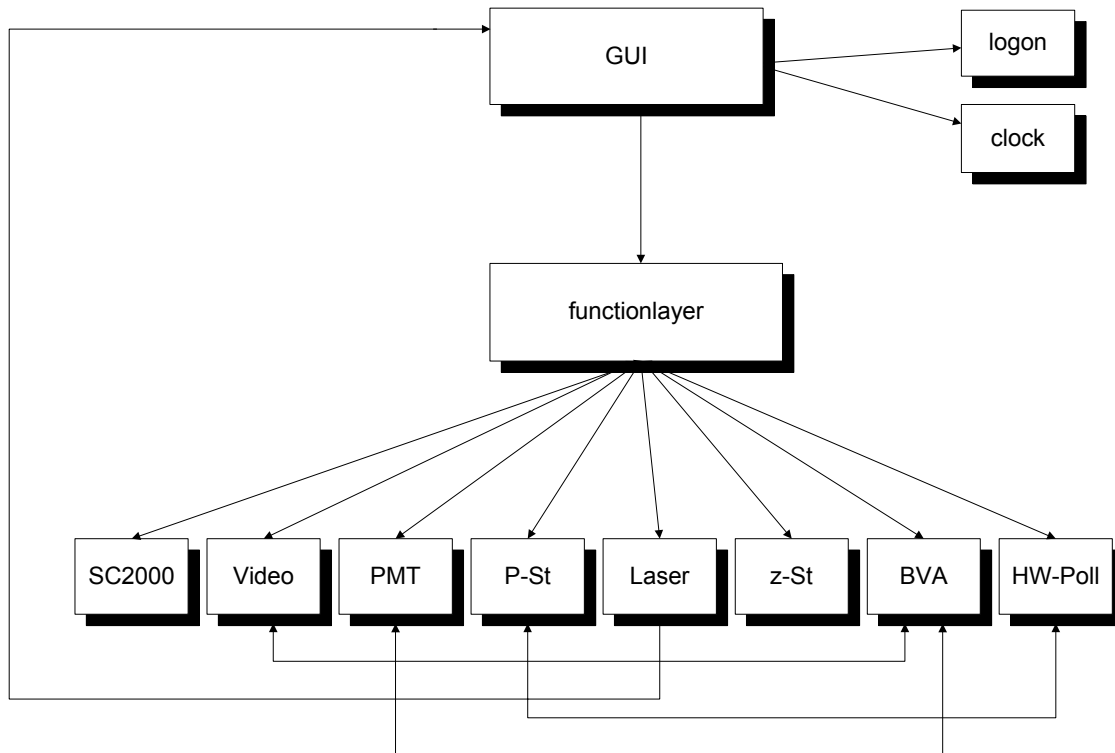


Bild 2: Kommunikationsbeziehungen der einzelnen Objekte

Aus dieser Analyse ergab sich eine Struktur in Form eines Drei-Schichten-Modells.

Die einzelnen Objekte wurden in Form von Zustandsmaschinen realisiert. Diese sollten ereignisorientiert kommunizieren.

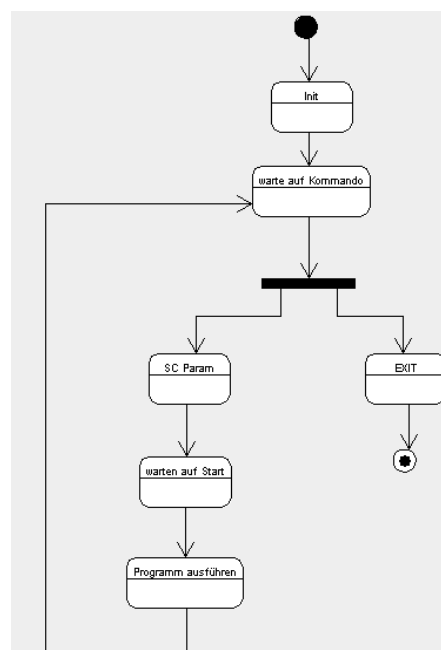


Bild 3: Zustandsmaschine Scanner

Wie bereits erwähnt wurde die Software in UML modelliert, hierzu wurde das Tool „Poseidon for UML“ genutzt. Mit diesem wurden die Aktivitätendiagramme, Sequenzdiagramme, Use-Case, usw. erstellt.

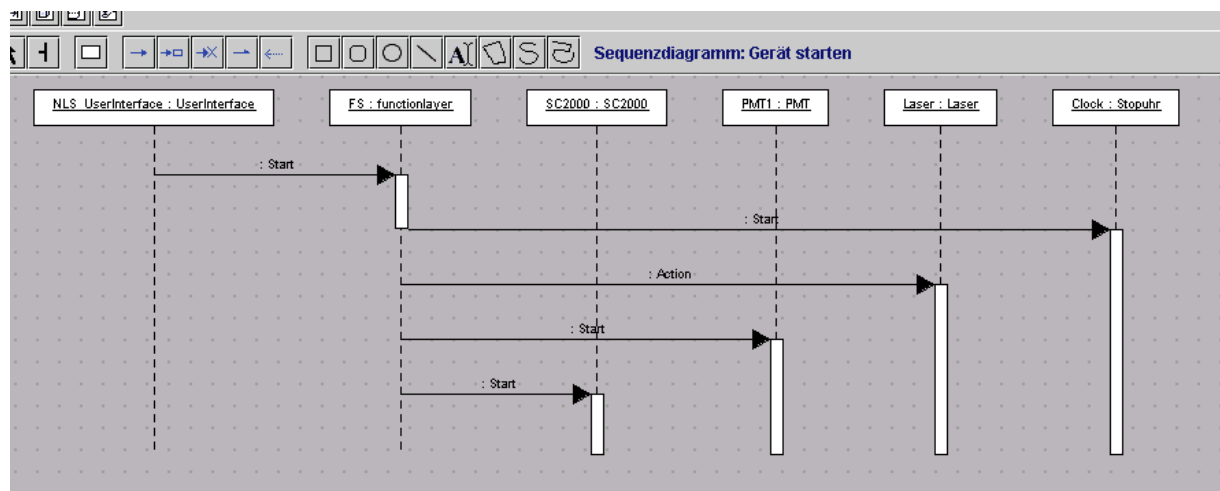


Bild 4: Sequenzdiagramm mit Poseidon for UML

## Umsetzung

Zur Umsetzung des Programms mit LabVIEW™ 6.0 wurde zunächst ein Objektnetz mit den einzelnen SubVIs erstellt. Die Kommunikation der einzelnen Objekte sollte mittels Notifications und Messages geschehen, ebenfalls sollten die Objekte in den einzelnen Schichten als Zustandsmaschinen realisiert werden, welche ereignisorientiertes Verhalten zeigen. Um diese Punkte zu erfüllen wurde das Toolset ObjectVIEW (Object Net Toolset) der Firma Vogel Automatisierungstechnik GmbH eingesetzt. Durch dieses Toolset ist es möglich unter der Programmiersprache LabVIEW, die auf dem Datenflussprinzip basiert, objektorientiert zu programmieren. Insbesondere lassen sich bei diesem speziellen Toolset hierarchische Objektnetze zu erstellen.

Dadurch ist es nun möglich die bestehende Softwarestruktur unter LabVIEW zu realisieren. Ports ermöglichen es den Objekten ereignisgesteuert mit ihrer Umgebung zu kommunizieren.

Weiterhin bietet dieses Toolset die Möglichkeit diese Ports grafisch zu verbinden. Die Anzahl der Ports eines Objektes hängt ab von der Anzahl der Objekte, mit denen kommuniziert werden soll. Die Ports werden aber unidirektional genutzt, sodass für jede Richtung eine eigener Port(in/out) angelegt werden muss.

Die erstellten Objekte werden anschließend an den Ports über Connector - Objekte verbunden. Durch die Connector - Objekte wird den Ports automatisch eine Notification bzw. Message zugewiesen, an dem das Objekt hören soll bzw. schreiben kann. Dadurch vereinfacht sich der Programmcode, weil „nur noch“ auf Events am Eingangsport gewartet wird, dann einmalig die Verarbeitung erfolgt und die Antworten auf den Ausgangsport geschrieben werden. Darauf reagiert dann das an diesem Port angeschlossene Objekt.

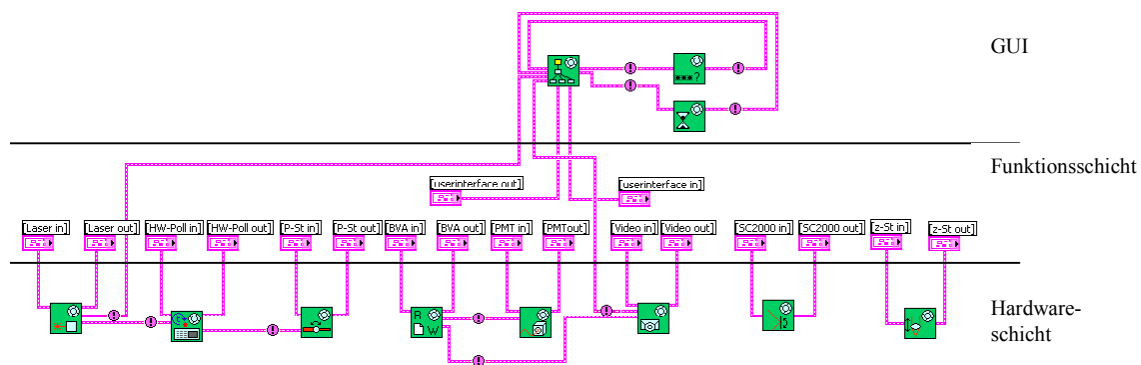


Bild 5: Objektnetz in LabVIEW™

Zur Bilddarstellung wurde das Toolset Imaq Vision von NI eingesetzt. Dadurch war es einfach Daten in ein Bild schreiben, sowie dieses anzeigen und abspeichern zu können. Die Erfassung der Daten fand mittels der Multi I/O Karte PCI-MIO16E-1 statt. Das Datenakquisitionsobjekt ist in der Lage Daten mit 1,25 Ms/s aufzunehmen, zu verarbeiten und diese als Bilder online darzustellen. Mit dieser Karte steht eine sehr schnelle Karte zur Verfügung. Diese Karte wurde ebenfalls dafür genutzt um verschiedene Signale an die Hardwarekomponenten zu übermitteln.

## Literatur

- [HBON] [www.objectview.de](http://www.objectview.de) - Einführung in Objekt- Netze / 09.2001
- [HBOVG] [www.objectview.de](http://www.objectview.de) - ObjectVIEW - Handbuch / 15.03.2002
- [Oe01] Objektorientierte Softwareentwicklung Analyse und Design mit der Unified Modeling Language, Oldenburg, Bernd Oestereich 5. völlig überarbeitete Auflage, 2001