



Kurzübersicht

Das Remotelab GOLDi (Grid of Online Lab Devices Ilmenau) ist seit vielen Jahren ein integraler Bestandteil der Lehre. Es wird an der TU Ilmenau für Lehrveranstaltungen, praktische Arbeiten sowie Online-Demonstrationen genutzt. Darüber hinaus wird es im Rahmen von Kooperationen mit nationalen und internationalen Projektpartnern gemeinsam eingesetzt. Nicht erst mit der Pandemie-Situation des Jahres 2020 wurden die Vorteile von digitalen Laborlösungen deutlich. Die Aufbereitung der Lerninhalte in Form von digitalen, interaktiven Simulationen, Videoaufzeichnungen oder über einen Remote-Zugriff eröffnet mit der permanenten Verfügbarkeit eine dem individuellen Arbeitsrhythmus und Kenntnisstand entsprechende flexible Zeiteinteilung.

Die Komplexität digitaler Systeme nimmt jedoch stetig zu. Um mit diesem Trend Schritt zu halten, entwickelt sich auch das didaktische Konzept unseres Remotelabs ständig weiter. GOLDi stößt derzeit an seine Grenzen bezüglich Komplexität und Variabilität. Die Sensoren aller aktuellen Modelle liefern ausschließlich binäre Status-Informationen (z.B. der Aufzug hat die obere Etage erreicht), die von der gewählten Steuereinheit (Mikrocontroller, FPGA, SPS) ausgewertet werden. Die Steuereinheit liefert daraufhin ihre binären Aktor-Signale (z. B. Fahrkorbmotor abwärts ein).

Viele dieser elektromechanischen Hardwaremodelle könnten durch zusätzliche Sensoren (z.B. Ultraschall, I2C, SPI) und Aktoren (z.B. PWM, I2C, SPI, USART) mit komplexeren Protokollen erweitert werden. Diese Arten von Sensoren und Aktoren können vom aktuellen GOLDi-System nicht angesteuert werden, da die Latenzzeit innerhalb eines verteilten Systems es nicht erlaubt, das Zeitverhalten dieser Protokolle zu reproduzieren. Das wiederum bedeutet, dass die Vielfalt und Komplexität der Experimente begrenzt ist.

Deshalb haben wir uns entschlossen, die Architektur von GOLDi unter Beibehaltung der bestehenden Funktionalität zu überarbeiten und zu erweitern. Die Schlüsselkomponente des neuen **GOLDi 2.0** Systems ist eine *Elektrische Schnittstellen Emulation*. Das bedeutet, dass die gewählte Steuereinheit (z.B. ein Mikrocontroller) mit unserem System genauso interagieren kann, wie sie es mit den realen Sensoren und Aktoren am entfernten Hardwaremodell über das entsprechende Protokoll tun würde. In der Realität ist es jedoch direkt mit einer Schnittstelleneinheit verdrahtet, die den realen Sensor/Aktor emuliert. Der interne Zustand der Schnittstelle und des Modells wird dabei im Hintergrund ständig synchronisiert.

Angenommen, ein Student möchte ein Wasserstands-Experiment mit einem Mikrocontroller steuern, um den Wasserverlust durch einen Verbraucher auszugleichen. Er kann den Wasserstand mit Hilfe eines Ultraschall-Abstandssensors ermitteln, um anschließend eine Pumpe zu steuern. Eine Kommunikationsschnittstelle für einen solchen Sensor setzt sich aus einem Trigger- und einem Antwortsignal zusammen. Der Mikrocontroller sendet einen Impuls auf der Trigger-Leitung und empfängt einen Impuls auf der Rückantwortleitung. Die Zeitverzögerung zwischen Trigger und Echo ist direkt proportional zum Wasserstand. Das ist überhaupt kein Problem, solange er es im praktischen Labor macht, wo die Steuereinheit fest mit dem physikalischen System verdrahtet ist (wie beispielsweise in unserem klassischen Schaltsysteme-Praktikumsversuch „Füllstands-Regelung“). Allerdings ist die Steuerung eines solchen Ultraschallsensors in der aktuellen Remotelab-Architektur aufgrund des Latenzproblems, das die Zeitverzögerungsmessung stört, nicht möglich. Dies wird in der neuen GOLDi 2.0 Architektur durch die Emulation der Sensoren und Aktoren auf der Controllerseite gelöst. Effektiv können wir so die direkte Kopplung zwischen Mikrocontroller und Wasserstands-Modell auflösen und beide Teile im Remotelab transparent synchronisieren.

Zusätzlich zu den Sensoren können auch Aktoren über spezielle Protokolle angesteuert werden. So können die Studierenden z.B. einen Servo-Motor eines elektromechanischen Hardwaremodells



ansteuern, indem sie die erforderliche PWM-Signalfolge (Puls Weiten Modulation) auf dem Mikrocontroller erzeugen. Diese Signale werden von der Steuereinheit so aufgearbeitet, dass das Interface der Steuereinheit das erzeugte PWM-Signal nachbilden kann, um den angeschlossenen Servo zu betreiben. Auf diese Weise können Sensoren und Aktuatoren mit beliebiger digitaler Schnittstelle (I2C, 1-wire, SPI, PWM, USART, etc.) in GOLDi 2.0 angesprochen werden.

1. Aufgabenstellungen in der Grundausbildung „Technische Informatik“, „Rechnerorganisation“ und „Schaltssysteme“

Mit dem GOLDi-Remotelab werden Themengebiete der Technischen Informatik unterstützt, wie sie im Grundstudium der TU Ilmenau für alle ingenieurwissenschaftlichen Studiengänge angeboten werden, in denen die Grundlagen für das Verständnis, den systematischen Entwurf und die formale Verifikation von digitalen Systemen gelegt werden.

Lernziel der Lehrveranstaltungen „Technische Informatik“, „Rechnerorganisation“ und „Schaltssysteme“ ist es, die Studierenden zu befähigen, selbstständig digitale Steuerungen zu entwerfen, zu bauen und deren Korrektheit nachzuweisen. Die Studierenden sollen die Fähigkeit erwerben, mit Hilfe digitaler Schaltungen Sensorwerte zu erfassen und so zu verarbeiten, dass Aktoren in einer in der Aufgabenstellung vorgegebenen Weise reagieren, z.B. die Steuerung eines Aufzugs oder eines einfachen 3-Achs-Portals.

Um dieses Ziel zu erreichen, werden zunächst die mathematischen Grundlagen der Booleschen Algebra vermittelt. Es folgt die systematische, formal überprüfbare Funktionsbeschreibung digitaler Steuerungen, wie z. B. die Formulierung von Abhängigkeiten von Sensorwerten in Form von Booleschen Gleichungen. Ein weiteres theoretisches Konzept, das insbesondere für den Aufbau sequentieller Schaltungen wichtig ist, sind endliche digitale Automaten (FSM – Finite State Machines). Darauf aufbauend wird gezeigt, wie diese Beschreibungen in digitale Schaltungen umgesetzt werden können. Um die Korrektheit ihres Verhaltens zu beweisen, ist ein praktischer Aufbau durch Verschaltung elementarer Grundsaltungen, die in ihrer Funktion den Grundelementen der Booleschen Algebra entsprechen, notwendig. Dies geschieht üblicherweise in einem Laborexperiment. Im Folgenden wird beschrieben, wie dieses Lehrkonzept mit Hilfe unseres hybriden Online-Labors GOLDi begleitet und unterstützt wird.

Integraler Bestandteil des Remotelabs sind verschiedene Online-Tools, sogenannte Interaktive Lernobjekte, die auf den Ausbildungsprozess zugeschnitten sind und den Studierenden die Möglichkeit geben, das in den Vorlesungen vermittelte Wissen durch internetbasierte Experimente zu nutzen. Beispielsweise können Gesetzmäßigkeiten der Booleschen Algebra mit Hilfe des Tools „**SANE**“ (Schaltssysteme Arbeitsblätter im Netz) interaktiv erforscht werden. Abbildung 1 zeigt einen Screenshot des SANE-Tools, der die Anwendung der Booleschen Algebra auf die Minimierung digitaler Schaltungen illustriert. Studierende können mit vorgegebenen Aufgaben oder frei mit beliebigen Ausdrücken der Algebra experimentieren und so die Zusammenhänge der algebraischen Gesetzmäßigkeiten im Zusammenspiel erarbeiten und verstehen. Dazu sind die Werkzeuge so konzipiert, dass sie Berechnungen in Echtzeit durchführen und die Ergebnisse in verschiedenen frei wählbaren Ansichten und Kontexten darstellen - z.B. boolesche Ausdrücke in verschiedenen Normalformen oder deren Wertverlauf als Wertetabelle.



The screenshot shows the SANE Tool interface. On the left, under 'All Functions', the 'Normal form' is set to 'KDNF'. Below, 'Boolean expressions' are listed, including a complex expression for f_0 . On the right, the 'Karnaugh Map' section shows the 'Output function' as y_0 and the 'Highest x-index' as x_3 . The 'Normal form' is set to 'DNF'. There are options for 'Auto-minimization' and 'Enable block mode'. Below these, two simplified Boolean expressions are shown: $f_0 = (\overline{x_2} \overline{x_1}) \vee (x_2 \overline{x_0}) \vee (x_2 x_1)$ and $f_0 = (\overline{x_2} \overline{x_1}) \vee (\overline{x_1} \overline{x_0}) \vee (x_2 x_1)$. A Karnaugh map grid is displayed with variables x_3, x_2, x_1, x_0 and values 0 and 1.

Abbildung 1. Minimierung Boolescher Ausdrücke mit dem SANE Tool

Für sequentielle Schaltungen und Steuerungsalgorithmen, die systematisch auf der theoretischen Basis von endlichen Zustandsautomaten entworfen werden, steht das Werkzeug „GIFT“ (Graphical Interactive FSM-Tool) zur Verfügung. Mit Hilfe eines grafischen Editors können hier Steueralgorithmen als Automatengraphen entworfen und analysiert werden. Die Analyse erfolgt über wählbare animierte Kurvenformen der Ein- und Ausgangssignale des Entwurfs oder durch den Export des Entwurfs in das GOLDi Remotelab und anschließenden interaktiven Experimentierens damit (siehe Abbildung 2).

The image shows two screenshots from the GIFT tool. The top-left screenshot shows a menu with 'Export to GOLDi' highlighted. A large orange arrow points from this menu to the bottom-right screenshot. The bottom-right screenshot shows a 'Control Unit' window with a dialog box titled 'Please Name the Machine to be imported:'. The machine is named 'A0'. Below, there are sections for linking variables to sensors and actuators. The sensor variables are x_6 (Z-Axis: Crane at position up), x_3 (User button), x_0 (X-Axis: Crane position right), and x_1 (X-Axis: Crane position left). The actuator variables are y_0 (Crane drive +X (right)) and y_1 (Crane drive -X (left)).

Abbildung 2. Export der Entwürfe aus dem GIFT Tool direkt ins GOLDi Lab

Die Verbindung zwischen Boolescher Algebra, schaltalgebraischen Ausdrücken und deren schaltungstechnischer Umsetzung stellt das Online-Tool „BEAST“ (Block Diagram Editing and Simulation Tool) her. In diesem Tool können die Studenten digitale Schaltungen virtuell erstellen und ihr Verhalten in Echtzeit beobachten, indem sie Wellenformen und farbkodierte Verbindungen verwenden, die aktive und inaktive virtuelle Leitungen



symbolisieren. Abbildung 3 zeigt einen Screenshot dieses Tools. Damit lassen sich sowohl einfache kombinatorische Schaltungen als auch komplexere sequentielle Schaltungen als Regelalgorithmen entwickeln.

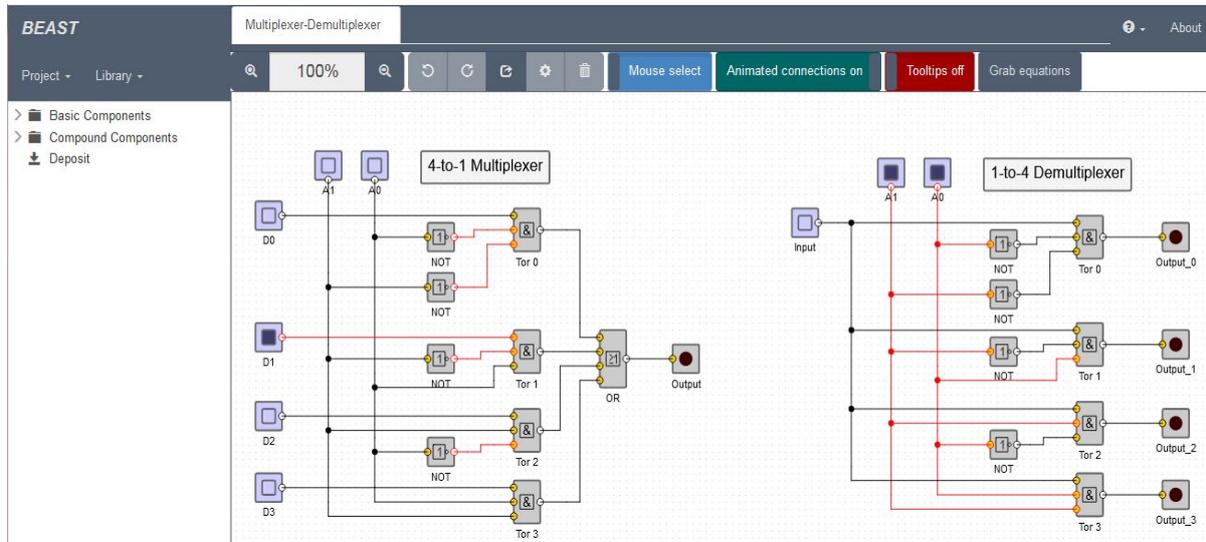


Abbildung 3. Multiplexer/Demultiplexer Entwürfe mit dem BEAST Tool

Die vorgestellten Werkzeuge SANE, GIFT und BEAST dienen dem praktischen Experimentieren auf der Basis der in den einzelnen Abschnitten des Ausbildungskonzeptes erworbenen Kenntnisse. Damit können die Studenten auf dem jeweiligen Wissensstand problemorientierte Teilaufgaben interaktiv lösen. Da die Lösungen dieser Teilaufgaben auch wesentliche Bestandteile von Steueralgorithmen sind, können die Werkzeuge auch bei der Vorbereitung von Laborexperimenten im GOLDi-Labor eingesetzt werden.

Der Schwierigkeitsgrad der Aufgaben kann an den jeweiligen Wissensstand im Semester angepasst werden. Zu Beginn des Semesters sind z.B. nur einfache Aufgaben wie das Arbeiten mit Booleschen Konstanten und Variablen möglich, die aber sofort im Online-Labor erprobt werden können, indem Variablen (Aktoren) auf konstante Werte gesetzt werden und eine sofortige Reaktion im Labor ausgelöst wird. Entsprechende Schutzvorrichtungen im Labor und Rückmeldungen verhindern, dass die Laborgeräte schädliche Operationen durchführen (z. B. Motoren über bestimmte Endschalter hinaus steuern). Im nächsten Trainingsschritt folgen Aufgaben zu Booleschen Ausdrücken, mit deren Hilfe komplexere Abhängigkeiten von Sensorsignalen bei der Steuerung von Aktoren berücksichtigt werden können. Zur Steuerung von sequentiellen Prozessen sind schließlich Kompetenzen zum Entwurf von sequentiellen Schaltungen bzw. Digitalen Automaten (FSM) erforderlich, die ebenfalls mit den oben beschriebenen Online-Tools bearbeitet und in das GOLDi-Labor exportiert werden können.

In der Grundausbildung werden zunächst virtuelle Interpreter als Steuergeräte eingesetzt, die mit den in der Theorie erworbenen Fähigkeiten leicht zu bedienen sind. Die Variablen der Booleschen Ausdrücke werden mit den entsprechenden Sensoren oder Aktoren im Labor gekoppelt und taktbasiert berechnet. Das Verhalten des physikalischen Systems und die Belegung der Eingangs- und Ausgangsvariablen, d.h. die aktuellen digitalen Sensor- und Aktorwerte, können über eine Webkamera beobachtet werden.

In höheren Semestern werden dann Mikrocontroller, programmierbare Schaltkreise (FPGAs) oder industrielle speicherprogrammierbare Steuerungen (SPS) als Steuergeräte eingesetzt. Die notwendigen Kenntnisse zur Programmierung in höheren Programmiersprachen (z.B. C++) oder Hardwarebeschreibungssprachen (z.B. VHDL), wie sie für die Programmierung von FPGAs erforderlich sind, werden in separaten Kursen erworben. Anspruchsvolle Aufgaben, die mit diesen Kenntnissen gelöst werden können, sind z. B. parallele Steuerungsalgorithmen für eine Aufzugssteuerung einschließlich der Ansteuerung der Bedienfelder innerhalb und außerhalb des Aufzugsmodells. Auch der Vergleich zwischen einer Softwarelösung und einer Hardwarelösung entweder über ein FPGA oder eine SPS ist für die Ausbildung in höheren Semestern interessant. Für die Bearbeitung von Quellcodes im Browser und deren Kompilierung stehen professionelle Entwurfswerkzeuge als Cloud-Service zur Verfügung, die über die Weboberfläche WIDE (Web-Integrated Design



Environment) aufgerufen werden können, die eine einheitliche Benutzeroberfläche für alle Programmiersprachen bietet.

Die Weiterentwicklung des Labors erfolgt stets unter Einbindung von Studierenden höherer Semester im Rahmen von Softwareprojekten und jährlichen Studienarbeiten sowie in Bachelor- und Masterarbeiten. Insbesondere werden Kompetenzen für die Herausforderungen entwickelt, die das Konzept der Industrie 4.0 an zukünftige Absolventen stellt. Um die Vielfalt solcher Aufgabenstellungen zu skizzieren, wird im folgenden Abschnitt die Architektur des GOLDi-Labors beschrieben.

2. GOLDi Lab Architektur

Unser GOLDi-Labor realisiert flexibel alle Varianten von Remote-Laboren: Remote-, Virtual- und Hybrid-Labore. Damit ist es möglich, alle Experimente entweder komplett virtuell oder auf realen Geräten oder in einer Kombination aus beidem durchzuführen.

2.1. Hardware

Der Hardware-Kern des hybriden Online-Labors GOLDi besteht aus einem Labor-Server, parallel betreibbaren physikalischen Systemen (elektromechanische Modelle wie z. B. ein Aufzug, ein Hochregallager, eine Fertigungszelle mit Förderbändern) und Steuereinheiten (z. B. Mikrocontroller, FPGA, SPS) – siehe Abbildung 4..

Physikalische Systeme und Steuergeräte sind über das Internet verbunden und können für Experimente beliebig gekoppelt werden, wobei jedes aus einer beliebigen Anzahl von Steuergeräten der physikalischen Systeme besteht. Diese flexible Architektur ermöglicht eine einfache Erweiterung der verfügbaren physikalischen Systeme und Steuereinheiten, ohne dass Hardwareänderungen an der Gesamtarchitektur vorgenommen werden müssen.

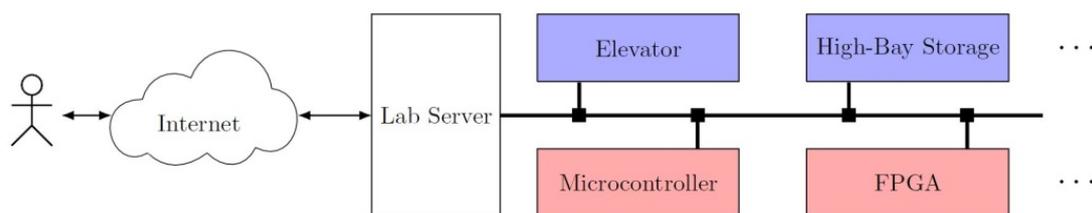


Abbildung 4. Hardware Infrastruktur des GOLDi Labs

2.2. Software

Die Software des GOLDi-Labs wird über eine zentrale Datenbank (im Rechenzentrum der TU Ilmenau) verwaltet. In ihr werden die Werkzeuge (u.a. die Interaktiven Lernobjekte) und die Verwaltungssoftware für Benutzer und Experimente sowie Dokumente gespeichert. Die Benutzeroberflächen der Dienste werden über eine Cloud verwaltet. Abbildung 5 zeigt die Softwarestruktur des GOLDi-Labs. Derzeit sind neben Ilmenau noch 10 weitere GOLDi-Infrastrukturen in Armenien, Australien, Deutschland, Georgien und der Ukraine in Betrieb.

Die cloudbasierte Softwarearchitektur ermöglicht eine effiziente Fernwartung und Aktualisierung der Laborsoftware. Alle Änderungen in der Cloud stehen allen Partnern sofort zur Verfügung, ohne dass Installationen oder Updates vor Ort notwendig sind. Steuergeräte und physikalische Systeme unterscheiden sich an den einzelnen Standorten und werden an die lokalen Anforderungen des Unterrichts angepasst. Sie stehen allen Nutzern der GOLDi Cloud nach Maßgabe der verfügbaren Kapazitäten zur Verfügung.

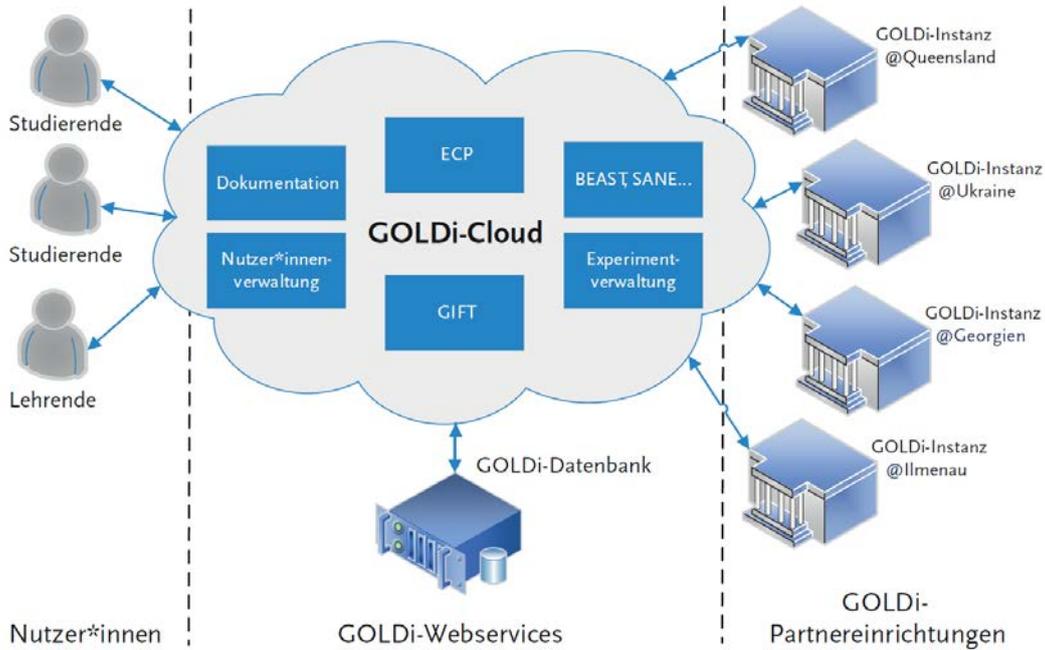


Abbildung 5. Software Infrastruktur des GOLDi Labs

3. Experimente mit dem GOLDi Remotelab

Die Architektur des GOLDi-Labors (siehe Abbildung 6) erlaubt eine beliebige Kombination von realen und virtuellen Geräten. Die Steuereinheit (CU, z. B. Mikrocontroller, FPGA, SPS) empfängt Sensorsignale vom physikalischen System (PS, Aufzug, 3-Achsen-Portal, Hochregallager) und wandelt sie entsprechend dem Regelalgorithmus in Aktorsignale um, die an das physikalische System gesendet werden. Während virtuelle Steuereinheiten und physikalische Systeme im Browser des Anwenders laufen, sind die realen Geräte als Hardware im Labor vorhanden.

Im neuen GOLDi 2.0 Ansatz erfolgt in den Interface-Einheiten (CU-I und PS-I) die eingangs erwähnte elektrische Interface Emulation, so daß sich damit in Zukunft Sensoren und Aktuatoren mit beliebiger digitaler Schnittstelle (I2C, SPI, etc.) für Versuchsaufgaben heute üblicher komplexer eingebetteter Systeme ansteuern lassen.

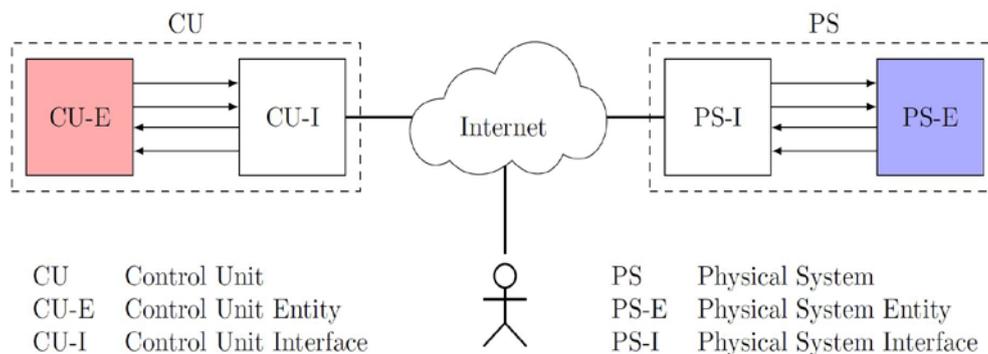


Abbildung 6. GOLDi Infrastruktur

Die virtuelle CU interpretiert und simuliert eine abstrakte Beschreibung des Steuerungsalgorithmus auf Basis eines endlichen Zustandsautomaten. Diese abstrakte Form des Steuerungsalgorithmus ist unabhängig von einer konkreten Implementierung. Wie diese Flexibilität des Hybridlabors in verschiedenen Lernszenarien genutzt



wird, soll Gegenstand des nächsten Abschnitts sein. Hier wird zunächst eine allgemeine Beschreibung des Experimentierens im GOLDi-Labor gegeben.

Virtuelle Geräte sind als JAVA Script realisiert und werden in einem Browser ausgeführt. Virtuelle physikalische Systeme bestehen aus einem Animationsteil, der für die Visualisierung der Bewegungen der physikalischen Systeme zuständig ist und einem Simulationsteil, der das Bewegungsverhalten des realen physikalischen Systems und dessen Sensor/Aktor-Reaktionen simuliert. Die virtuelle Steuereinheit ist ein Interpreter für FSM-Gleichungen, der aus den Gleichungen und aktuellen Sensorwerten zyklisch die Werte der Steuersignale für die Aktoren berechnet.

Die realen Geräte befinden sich im Labor und sind über ein lokales Ethernet (LAN) mit dem Labor-Server verbunden. Dieser realisiert die Kommunikation mit den Browsern auf den Client-Geräten (PCs, Laptops oder mobile Geräte) über Web-Sockets.

Durch die Konfiguration des Experiments entscheiden die Studierenden zwischen vier Arten von Experimenten:

- (A) Virtuelle Experimente (PS virtuell, CU virtuell),
- (B) Abstrakte Experimente (PS real, CU virtuell),
- (C) Reale Remote-Experimente (PS real, CU real) und
- (D) Implementierungstest-Experimente (PS virtuell, CU real).

Im Folgenden soll auf die Verwendung der verschiedenen Konfigurationen in Lernszenarien für unsere Studierenden eingegangen werden.

4. Einsatz für aktuelle und zukünftige Lehrmethoden

Die Aufgabe in einem Experiment besteht darin, ein physikalisches System mit einem selbst entworfenen Algorithmus so zu steuern, dass es einen vorgegebenen Bewegungsablauf ausführt. Die Verwendung des Digital-Twin-Konzepts ermöglicht die Erforschung von Seiteneffekten, die bei fehlerhaften Entwürfen bereits in der Entwurfsphase auftreten können. Dabei können Teilkonzepte des Regelalgorithmus unabhängig von der späteren Implementierung an einer virtuellen Emulation eines physikalischen Systems ausprobiert werden. Sobald der Algorithmus auf dem digitalen Zwilling funktioniert, wird er auch in der realen Umgebung funktionieren.

In den folgenden Beispielen wird diskutiert, welche der oben beschriebenen Experimentierkonfigurationen (A) bis (D) auf welches Lernszenario sinnvoll angewendet werden kann.

4.1 Hörsaal

Zur Demonstration des Entwurfsprozesses während der Vorlesungen eignet sich Konfiguration (B), weil mit abstrakten Experimenten die schrittweise Erarbeitung FSM-basierter Steueralgorithmen gezeigt werden kann. Beginnend mit Konzepten der Booleschen Konstanten, Variablen und Ausdrücke wird schrittweise das Verständnis für das Konzept der endlichen Zustandsmaschinen geschaffen. Es wird gezeigt, wie ein virtuelles Steuergerät unter Verwendung grundlegender Elemente von Steuerungsalgorithmen programmiert werden kann. Das Setzen von Aktoren y auf konstante Boolesche Werte setzt diese beispielsweise in Abhängigkeit von Sensorwerten x und deren logischer Verknüpfung in Bewegung oder stoppt die Bewegung wieder. Auf diese Weise können die Auswirkungen auf reale physikalische Systeme direkt im Experiment während der Vorlesung demonstriert werden. Für Studienanfänger ist es sehr motivierend, auf diese Weise eine Anwendung dessen zu sehen, was ihnen in der Theorie vermittelt wird.

4.2 Reflexion / „Flipped Classroom“

Virtuelle Experimente (Konfiguration A) sind sowohl für das Selbststudium als auch für Flipped-Classroom-Szenarien von Vorteil. Diese Arten von Experimenten laufen offline im Browser, sobald sie konfiguriert und gestartet wurden. So können die Studierenden ihre Experimente unabhängig von der Internetverbindung durchführen und Fragen für die Diskussion mit dem Betreuer in den Seminaren vorbereiten. Sie können



verschiedene Varianten von Regelalgorithmen oder Teile davon ausprobieren und die Unterschiede untersuchen oder die während der Vorlesung gezeigten Experimente wiederholen. Da die virtuellen Experimente vollständig auf Client-Rechnern laufen, können viele Studenten gleichzeitig Experimente durchführen. In Verbindung mit einem Lernmanagementsystem (LMS) können die Kursinhalte direkt mit den eingangs beschriebenen Interaktiven Lernobjekten (SANE, BEAST, GIFT) und virtuellen Experimenten gekoppelt werden, so dass das Gelesene sofort interaktiv ausprobiert werden kann. Die Interaktiven Lernobjekte im LMS werden auch für Übungen und Tests genutzt - allerdings bisher nur mit wenigen Beispielen.

4.3 Angeleitete Entwürfe

In Workshops zur beruflichen Bildung mit 10 bis 20 Personen setzen die Lehrenden zunächst virtuelle Experimente ein (Konfiguration (A)) und lassen die Lernenden die jeweils demonstrierten Schritte am eigenen Computer nachvollziehen. Dies geschieht zunächst unabhängig von einer späteren Umsetzung auf der abstrakten Ebene der digitalen Automaten, die ebenfalls in der Theorie vermittelt wurde. Auf diese Weise werden die Lernenden mit dem Entwurfsprozess und der Bedienung des GOLDi-Labs vertraut gemacht. Anschließend wird ihnen eine ähnliche Aufgabe gestellt, die sie selbständig lösen sollen. Verschiedene von den Lernenden vorgeschlagene Lösungen werden als Grundlage für die Diskussion an einer Präsentationstafel und für die Auswahl der besten Lösung verwendet. Dafür wird jetzt Konfiguration (B) verwendet, um das reale physikalische System damit zu steuern. Die Erfahrung zeigt, dass dies die Lernenden motiviert und sie anspornt, im Wettbewerb mit den anderen ihr Bestes zu geben.

Im weiteren Verlauf des Workshops werden die Regelalgorithmen in Hardware und Software implementiert. Um sicher zu gehen, dass die Umsetzung gelungen ist, können die Lernenden zunächst mit virtuellen physikalischen Systemen in der Testkonfiguration (D) arbeiten. Schließlich wird das reale Experiment in der Konfiguration (C) eingesetzt.

4.4 Laborübungen

Um reale Experimente durch die Nutzung von Online-Laboren zu ersetzen, ist Konfiguration (C) die geeignete Wahl. Die Architektur des Labors erlaubt den Anschluss mehrerer identischer physikalischer Systeme und Steuergeräte, ohne dies bei der Konfiguration des Experiments berücksichtigen zu müssen. Bei der Reservierung von Experimenten wird dynamisch auf die verfügbaren Geräte zugegriffen. Sind z. B. drei Instanzen desselben physikalischen Systems installiert, so ist dies für den Benutzer transparent. Sie sehen nicht, welches konkrete Gerät mit dem Experiment verbunden ist. Erst wenn alle Geräte belegt sind, müssen sie nach einem anderen Zeitfenster suchen. Für die Reservierung gibt es verschiedene Prioritätsklassen, so dass gewährleistet ist, dass die Dozenten/Seminarleiter die notwendigen Reservierungen für ihren Kurs vornehmen können.