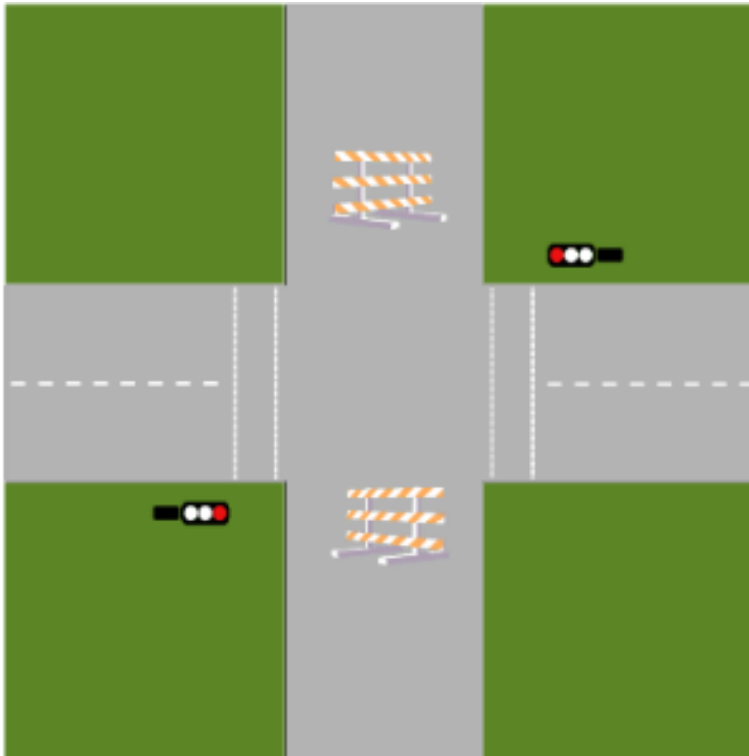


Ampelsteuerung



Verfasser: Marc Weintz

Kurz-Info:

Informatischer Inhalt: Zustandsdiagramme, Modellierung von Schaltungen, Implementierung eines Mikrocontrollers (z.T. generativ)

Jahrgangsstufe: (Ober- und) Mittelstufe

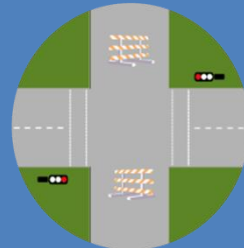
Vorwissen: erste Erfahrungen in der Programmierung (beliebige Programmiersprache)

KURZINFORMATION FÜR DIE LEHRKRAFT

Titel: *Ampelsteuerung*

Schulstufe: *(Ober- und) Mittelstufe*

optimale Jahrgangsstufe: *Klasse 8-EF*



Themenbereich: *Zustandsdiagramme, Modellierung und Implementierung von Ampelschaltungen*

EINORDNUNG IN GESETZLICHE RAHMENBEDINGUNGEN

Bildungsstandards der GI: Algorithmen, Sprachen und Automaten, Modellieren und Implementieren, Kommunizieren und Kooperieren

Kernlehrplan für die gymnasiale Oberstufe: Modellieren; Implementieren; Darstellen und Interpretieren; Kommunizieren und Kooperieren; Algorithmen; Formale Sprachen und Automaten; Informatiksysteme; Informatik, Mensch und Gesellschaft

Vorgaben zum Zentralabitur: Endliche Automaten und formale Sprachen

Einbindung in den Unterricht: Dieses Modul bietet verschiedene Anknüpfungspunkte, die sich für eine Einbindung in eine Unterrichtsreihe anbieten. Zum einen kann das Modul als Exkurs zu einer in die Programmierung einführenden Unterrichtsreihe genutzt werden. Durch den starken Alltagsbezug und die Programmierung des Mikrocontrollers wird die Motivation der Schülerinnen und Schüler gefördert und die Relevanz der Informatik in der Gesellschaft verdeutlicht. Das Modul kann außerdem in der Oberstufe zur Einführung von endlichen Automaten dienen und eine Anwendungsmöglichkeit derselben aufzeigen.

Vorkenntnisse: erste Programmiererfahrung (beliebige Programmiersprache)

Dauer: *4 Stunden*

Inhaltsbeschreibung: Das Modul ist in drei Phasen unterteilt. Die erste Phase dient der Erarbeitung von Zustandsdiagrammen und erfolgt mit Hilfe einer Präsentation im Unterrichtsgespräch. In der zweiten Phase entwickeln die Schülerinnen und Schüler eine einfache Ampelsteuerung weiter. Dazu gibt es verschiedene Stationen, in denen die Situation jeweils zuerst in einem Zustandsdiagramm modelliert und anschließend mit einem Arduino Mikrocontroller implementiert wird. Nachdem die Lernenden sich auf diese Weise ein Grundwissen angeeignet haben, können sie in der letzten Phase eine komplexere Situation auswählen und diese modellieren und implementieren. Zum Schluss werden die Ergebnisse dem Plenum präsentiert.

INHALT

Kurzinformation für die Lehrkraft.....	2
Lernziele	5
Fachliche Analyse	5
Einordnung in Gesetzliche Rahmenbedingungen.....	6
Bildungsstandards der GI	6
Lehrplan Informatik.....	6
Vorgaben zum Zentralabitur	6
Benötigte unterRichtliche Voraussetzungen	7
Einbettung in den Schulunterricht	7
Didaktische/Methodische Schwerpunktsetzung.....	7
Verlaufsplan des Moduls.....	10
Quellenverzeichnis	11
Abbilddungsverzeichnis	11
Anhang	12

LERNZIELE

Die Schülerinnen und Schüler..

- verstehen das Konzept des Zustandsdiagramms.
- wenden das Konzept des Zustandsdiagramms an.
- vertiefen ihr Verständnis und Wissen über Variablen, if-Abfragen.
- kommunizieren über informatische Inhalte.

FACHLICHE ANALYSE

- Einer der inhaltlichen Schwerpunkte dieses Moduls ist die Modellierung von Zustandsdiagrammen. Diese beschreiben alle möglichen Zustände des Systems zur Laufzeit und die Übergänge zwischen diesen. Grundsätzlich gibt es in solchen Diagrammen zwei verschiedene Arten von Elementen, die hier relevant sind: Zustände und Transitionen. Transitionen beschreiben dabei meistens den Übergang von einem Zustand in einen anderen. In einigen Fällen kann eine Transition allerdings auch zum Ausgangszustand zurückführen. Zustandsübergänge können mit einer Verhaltensspezifikation, also einer Aktivität verbunden sein, die eintritt, wenn der Übergang aktiv wird. Darüber hinaus können sie an Bedingungen geknüpft sein, die erfüllt sein müssen, damit sie durchlaufen werden können. Ein Zustand modelliert eine bestimmte Situation und ist mit einem Namen beschriftet. Auch ein Zustand kann Aktivitäten beinhalten, die ausgeführt werden können, wenn sich ein Objekt in diesem Zustand befindet. Neben den normalen Zuständen existieren noch sogenannte Pseudo-Zustände, in ihnen existiert keine Wertbelegung. Hier sind vor allem die Start- und Endzustände relevant. Außerdem können Gabelungen und Entscheidungen benötigt werden.

Zur Implementierung wird die opensource Physical-Computing Plattform Arduino benutzt. Sie besteht aus Soft- und Hardware. Die Hardware besteht aus einem Mikrocontroller mit analogen und digitalen Anschlüssen. Die Entwicklungsumgebung ist übersichtlich gestaltet und eröffnet so auf einfachem Wege auch Anfängern den ProgrammierEinstieg mit Mikrocontrollern. Die Programmierung erfolgt in C bzw. in C++ (ähnlich zu Java). Dabei unterstützen umfangreiche Libraries und auch zahlreiche Beispiele. Um ein funktionsfähiges Programm, genannt Sketch, zu implementieren genügt es zwei Methoden zu definieren. Die setup()- Methode wird beim Start einmalig aufgerufen und dient für Einstellungen zum Beispiel zur Definition von Pins. Die loop()-Methode wird durchgängig immer wieder durchlaufen, solange das Arduino-Board eingeschaltet ist.

- Möglichkeiten zur weiteren Einarbeitung ins Thema:
 - DAVE: <https://eleed.campussource.de/archive/2/201>
 - https://de.wikipedia.org/wiki/Zustandsdiagramm_%28UML%29
 - www.uml.ac.at/wp-content/uploads/teaching/04_Zustandsdiagramm_Folien.pdf
 - <https://www.arduino.cc/>

BILDUNGSSTANDARDS DER GI

- Im Vordergrund dieses Moduls ist mit Sicherheit das Prozessfeld „Modellieren und Implementieren“ zu sehen, da die Schülerinnen und Schüler genau das tun. Sie modellieren und implementieren verschiedene Situationen einer Ampelsteuerung. Zur Modellierung derselben verwenden sie Zustandsdiagramme, die auf endlichen Automaten basieren. Damit erstellen sie einen Algorithmus zur Lösung der jeweiligen Problematik innerhalb der einzelnen Situationen. Zur Implementierung verwenden sie eine renommierte Programmiersprache, um den Mikrocontroller zu steuern. Da das gesamte Modul mindestens in Partnerarbeit bewältigt wird, schulen die Schülerinnen und Schüler außerdem ihre Kooperationsfähigkeiten. Sie müssen zusammenarbeiten und sich gegenseitig unterstützen um die umfangreichen Situationen zu bewältigen. Dabei wird natürlich auch die Kommunikation über informatische Inhalte weiterentwickelt und geübt.

LEHRPLAN INFORMATIK

- Modellieren
Ein Schwerpunkt dieses Moduls liegt auf der Erstellung von Zustandsdiagrammen, die die Situationen an Kreuzungen modellieren.
- Implementieren
Die Ampelsteuerungen für die verschiedenen Situationen werden von den Lernenden mit Hilfe der Programmieroberfläche Arduino implementiert und können über den Mikrocontroller direkt getestet werden.
- Kommunizieren und Kooperieren
Die Bearbeitung der Phasen 2 und 3 erfolgt in Teamarbeit. Die Lernenden müssen also zusammenarbeiten und über ihr Vorgehen und die informatischen Inhalte kommunizieren, damit sie die Probleme lösen können.
- Algorithmen
Durch die Modellierung der einzelnen Situationen in Zustandsdiagrammen entwickeln die Schülerinnen und Schüler Algorithmen, die sie anschließend mit dem Mikrocontroller implementieren.
- Formale Sprachen und Automaten
Die Lernenden modellieren die verschiedenen Situationen an einer Kreuzung als Zustandsdiagramm und lernen damit ein Beispiel endlicher Automaten kennen und anzuwenden.
- Informatik, Mensch und Gesellschaft
Durch die Thematik des Moduls „Ampelsteuerung“ wird ein klarer Lebensweltbezug für die Schülerinnen und Schüler geschaffen, der die Relevanz der Informatik für die Gesellschaft verdeutlicht.

VORGABEN ZUM ZENTRALABITUR

- Endliche Automaten und formale Sprachen
Die Schülerinnen und Schüler modellieren kontextbezogene Problemstellungen indem sie Zustandsdiagramme zu den einzelnen Situationen erstellen.

BENÖTIGTE UNTERRICHTLICHE VORAUSSETZUNGEN

- Die Schülerinnen und Schüler sollen erste Programmiererfahrungen gesammelt haben und ein Grundverständnis von Variablen und Kontrollstrukturen entwickelt haben.

EINBETTUNG IN DEN SCHULUNTERRICHT

- Dieses Modul kann genutzt werden um den Schülerinnen und Schüler die Strukturierung eines größeren Projekts näher zu bringen. Dabei liegt ein Fokus auf der Erstellung von Zustandsdiagrammen. Diese können genutzt werden um endliche Automaten einzuführen und einen Anwendungsbezug derselben vorzustellen. Außerdem kann der Aufbau und die Funktionsweise von Mikrocontrollern an einem alltagsnahen Beispiel kennen gelernt und erprobt werden. Dadurch wird die Programmierung in Bezug zur Lebenswelt der Schülerinnen und Schüler gesetzt. Dies fördert die Motivation der Lernenden und stellt die Bedeutung der Informatik für die moderne Gesellschaft heraus.

DIDAKTISCHE/METHODISCHE SCHWERPUNKTSETZUNG

- Beschreibung des Modulablaufs

Das Modul „Ampelsteuerung“ ist in 3 Phasen eingeteilt. Die erste Phase findet im Plenum statt und nutzt eine interaktive Präsentation zum thematischen Einstieg. In dieser Präsentation wird anhand der Situation an einem Bahnübergang ein informelles Zustandsdiagramm erstellt.

Die Schülerinnen und Schüler diskutieren im Plenum über mögliche Zustände und Zustandsänderungen und erweitern so Schritt für Schritt das Zustandsdiagramm bis die Situation vollständig erfasst ist.

In der nun folgenden ersten Entwicklungsphase greifen verschiedene Medien und Methoden ineinander. Die Schülerinnen und Schüler sollen in Zweierteams fünf verschiedene Stationen durchlaufen, die alle aufeinander aufbauen. In Station 1 wird zunächst eine überschaubare Situation an einer Straßenkreuzung beschrieben, die in den darauf folgenden Stationen weiterentwickelt wird und so immer komplexere Strukturen annimmt. In den einzelnen Stationen durchlaufen die Schülerinnen und Schüler einzelne Phasen, in denen sie unterschiedliche Medien benutzen müssen, um die Aufgaben zu lösen. Diese unterschiedlichen Medien und Programme werden im Einzelnen durch Merkblätter eingeführt und erläutert. Der erste Schritt besteht daraus die beschriebene Situation aus Holzmodellen nachzubauen und diese anschließend mit dem Arduino Mikrocontroller zu verbinden. Nachdem diese Rahmenbedingungen geschaffen wurden, wird in dem grafischen Editor „DAVE“ das entsprechende Zustandsdiagramm modelliert, das die gegebene Situation beschreibt. Als Abschluss der Station fehlt jetzt noch die Implementierung des Zustandsdiagramms. In den ersten Stationen bekommen die SuS ein Grundgerüst gestellt und sollen dieses dann an Hand ihres Zustandsdiagramms erweitern. Wenn die Stationen komplexer werden, werden die Schülerinnen und Schüler mit einem Merkblatt angeleitet einen Generator (UML2Arduino) zu verwenden, der aus dem Zustandsdiagramm Code erzeugt.

Nach diesen relativ stark angeleiteten Stationen sollen die Schülerinnen und Schüler ihr gelerntes und erprobtes Wissen anwenden. Dazu bekommen die Zweierteams nun unterschiedliche Situationen, die sie umsetzen sollen. Die Beschreibungen dieser Situationen sind wesentlich undetaillierter und offener gestaltet, sodass die Schülerinnen und Schüler die Möglichkeit bekommen, ihre eigenen Ideen mit einzubringen.

Im Anschluss an diese zweite Erarbeitungsphase findet dann ein gemeinsamer Rundgang statt, bei dem die Gruppen ihre Aufgabe und die Ergebnisse präsentieren sollen.

- Begründung des Methoden- und Medieneinsatzes:

Im InfoSphere Schülerlabor für Informatik ist eines der wichtigsten Prinzipien, dass die Schülerinnen und Schüler selbstständig arbeiten um informatische Probleme zu lösen. Dabei sollen sie in Teams möglichst ohne direkte Anleitung durch eine Lehrperson arbeiten und sich gegenseitig unterstützen. Um dies zu gewährleisten wurden für dieses Modul Arbeitsblätter entwickelt, die die Schülerinnen und Schüler durch die einzelnen Phasen führen sollen. Der Einsatz der verschiedenen Programme wird durch Merkblätter angeleitet und eingeübt, sodass die Schülerinnen und Schüler sich den Umgang mit diesen selber erarbeiten können. Dabei ist die Rolle der Betreuerinnen und Betreuer eine rein beratende. Um allerdings trotzdem den zeitlichen Rahmen und vor allem das Erfolgserlebnis der Teilnehmerinnen und Teilnehmer gewährleisten zu können, wurden die letzten Situationen entwickelt. Die SuS können hier relativ schnell erste Ergebnisse erzielen, die dann allerdings beliebig weiter entwickelt werden können.

Um ein Verständnis der Theorie der Zustandsdiagramme in einem übersichtlichen Zeitrahmen zu ermöglichen, findet die Erarbeitung im Plenum an Hand eines anschaulichen Beispiels statt. Dies bietet den Schülerinnen und Schüler außerdem eine erste Gelegenheit über die informatischen Inhalte zu kommunizieren. Diese Situation ist ähnlich zu der gewohnten schulischen, sodass die Schülerinnen und Schüler sich, trotz der neuen Umgebung, erst einmal auf das inhaltliche konzentrieren können, bevor sie sich mit der vielleicht ungewohnten Situation des eigenständigen Experimentierens auseinandersetzen müssen.

Der später verwendete grafische Editor „DAVE“ ist (an der Universität Dortmund) entwickelt worden, um die formale Beschreibung des Verhaltens von Software zu verstehen und einzuüben. Obwohl er ursprünglich für die Benutzung durch Studierende entwickelt wurde, ist er sehr übersichtlich und intuitiv entwickelt worden, sodass sich die Nutzung auch für Schüler und Schülerinnen anbietet.

Mit Hilfe der Mikrocontrollerprogrammierung können die Schülerinnen und Schüler direkt die Ergebnisse ihrer Arbeit sehen. Sie können in kurzen Intervallen ihren Code testen. Dies beschert zum einen häufige Erfolgserlebnisse und zum anderen hält es die Frustrationsgrenze relativ niedrig, da der mögliche Fehler immer auf einen kleinen Bereich einzugrenzen ist.

Die Verwendung der Software UML2Arduino bietet die Möglichkeit, trotz komplexerer Aufgabenstellungen und begrenzter Zeit, tolle Ergebnisse zu erlangen. Sie wird erst gegen Ende des Moduls eingeführt, da die Schülerinnen und Schüler zuvor das entsprechende „Handwerkszeug“ erst einmal selber erlernen und üben sollen. Dies gibt auch einen Ausblick auf professionelle Systementwicklung, da auch dort oft schon vorhandene Werkzeuge genutzt oder weiterentwickelt werden.

Die Präsentation am Schluss des Moduls dient dazu, dass die Schülerinnen und Schüler einmal üben können, ihre Ergebnisse vor Publikum zu präsentieren, aber auch um ihr Vorgehen und ihre möglichen Probleme zu erläutern und damit ihre Kommunikationsfähigkeiten über informatische Inhalte erweitern können. Der Applaus der Mitschülerinnen und Mitschüler, falls nötig durch Betreuende initialisiert, steigert zudem das Selbstbewusstsein und auch das Selbstwertgefühl der Präsentierenden.

- Didaktische Prinzipien nach Baumann und Hubwieser

Das Modul „Ampelsteuerung“ ist unter der Berücksichtigung der didaktischen Prinzipien von Baumann und Hubwieser entwickelt worden. Es ist nicht immer möglich alle diese Prinzipien gleichmäßig umzusetzen. Hier soll ein Überblick über die in diesem Modul im Wesentlichen berücksichtigten Prinzipien gegeben werden.

Schon der Titel dieses Moduls verweist auf einen starken Bezug zur Alltagswelt der Schülerinnen und Schüler. In der moderne Welt hat jeder schon einmal an einer Kreuzung gestanden und darauf gewartet, dass die Ampel grün wird, oder eine Bedarfsampel benutzt um eine vielbefahrene Straße zu überqueren. Doch wie funktioniert das überhaupt, dass an einer großen Kreuzung die Ampeln korrekt arbeiten und keine Unfälle passieren (Prinzip der Lebensnähe und Aktualität)? Zu Beginn jedes Moduls findet im Schülerlabor „InfoSphere“ eine kurze Frontalphase statt, in der sich die Betreuerinnen und Betreuer kurz vorstellen, einige organisatorische Aspekte ansprechen und außerdem einen Überblick über das jeweilige Modul geben. Dabei wird der Ablauf des Tages besprochen um die Struktur, aber auch die gemeinsamen Ziele zu verdeutlichen (Prinzip der Zielvorstellung). Im Anschluss daran finden die verschiedenen Phasen des Moduls statt, in denen sowohl die verwendeten Medien, als auch die Arbeitsformen wechseln (Variabilität und Flexibilität). Dabei arbeiten die Lernenden größtenteils selbstständig in Zweierteams und erarbeiten sich ihr Wissen mit Hilfe der Merkblätter um es anschließend in den Arbeitsaufträgen anzuwenden (Prinzip des aktiven Lernens). In dieser Erarbeitungsphase arbeiten die Schülerinnen und Schüler nicht nur am Rechner, sondern bauen ihre Modelle und Schaltungen der Ampelsteuerung erst nach bevor sie mit den Zustandsdiagrammen und der eigentlichen Implementierung beginnen (Operatives Prinzip). Durch die Einteilung in einzelne Stationen und die Unterstützung der Betreuerinnen und Betreuer können die Lernenden ihr Wissen erweitern und einüben (Übung). Außerdem können sie so den Erfolg ihrer Arbeit immer zwischendrin überprüfen und erleben dadurch viele Teilerfolge (Motivierung). Durch die Erweiterung der Erarbeitungsphase um die Zusatzsituation, können die Schülerinnen und Schüler ihre Ampelsteuerung freier gestalten und ihre eigenen Ideen einbringen (Motivierung, Kreativitätsförderung). Die Offenheit der Aufgabe ermöglicht es den Lernenden außerdem die Bearbeitung nach ihren Fertigkeiten und in ihrem eigenen Tempo zu erledigen (Prinzip des individuellen Lerntempos, Differenzierung).

VERLAUFSPLAN DES MODULS

Zeit	Phase	Inhalt	Medium	Sozialform
0:00	Begrüßung	<i>Organisatorisches und informierender Einstieg</i>		LV
0:05	Teil 1 - Einstieg	<i>Erarbeitung der Zustandsautomaten anhand der Präsentation</i>	Smartboard	UG
0:30	Teil 2 - Station 1	<i>Modellierung und Implementierung der Station anhand der ABs und Merkblätter</i>	PC, Modelllampeln, Grundplatte, ABs, Merkblätter, rote Folie, Zustandskärtchen, Folienstift	PA
1:10	Teil 2 - Station 2			
1:40	Teil 2 - Station 3			
2:00	Teil 2 - Station 4			
2:30	Teil 2 - Station 5			
2:50	Pause	<i>Pause</i>	Pause	Pause
3:00	Teil 3	<i>Modellierung und Implementierung der gewählten Situation</i>	Arbeitsblatt, Kreuzungsskizze, Modelllampeln, PC	GA (je 4)

Auf eine explizite Einteilung der Pausen im zweiten Teil wird an dieser Stelle verzichtet, da die SuS eigenständig arbeiten und dadurch die Pausen nach eigenem Ermesse einrichten können, ohne dass gesonderte Unterbrechungen eingeplant werden müssen.

QUELLENVERZEICHNIS

- „Schülerlabor-Modul zur Zustandsmodellierung einer Ampelanlage unter Einsatz von Arduino-Mikrocontrollern“, Schriftliche Hausarbeit im Rahmen der Ersten Staatsprüfung dem Landesprüfungsamt für Erste Staatsprüfungen für Lehrämter an Schulen vorgelegt von: Marc Weintz, Aachen, den 4. Mai 2012
- Hubwieser, P. (2013): Didaktik der Informatik: Grundlagen, Konzepte, Beispiele, Springer-Verlag Berlin Heidelberg
- Baumann, R. (1996): Didaktik der Informatik – Klett

ABBILDUNGSVERZEICHNIS

- Alle verwendeten Bilder sind lizenzfrei veröffentlicht (CC0) oder vom InfoSphere-Team erstellt. Die genauen Nachweise sind den einzelnen Abbildungen zu entnehmen.

ANHANG

- Arbeitsblätter
- Lösungsdokumente für Betreuer