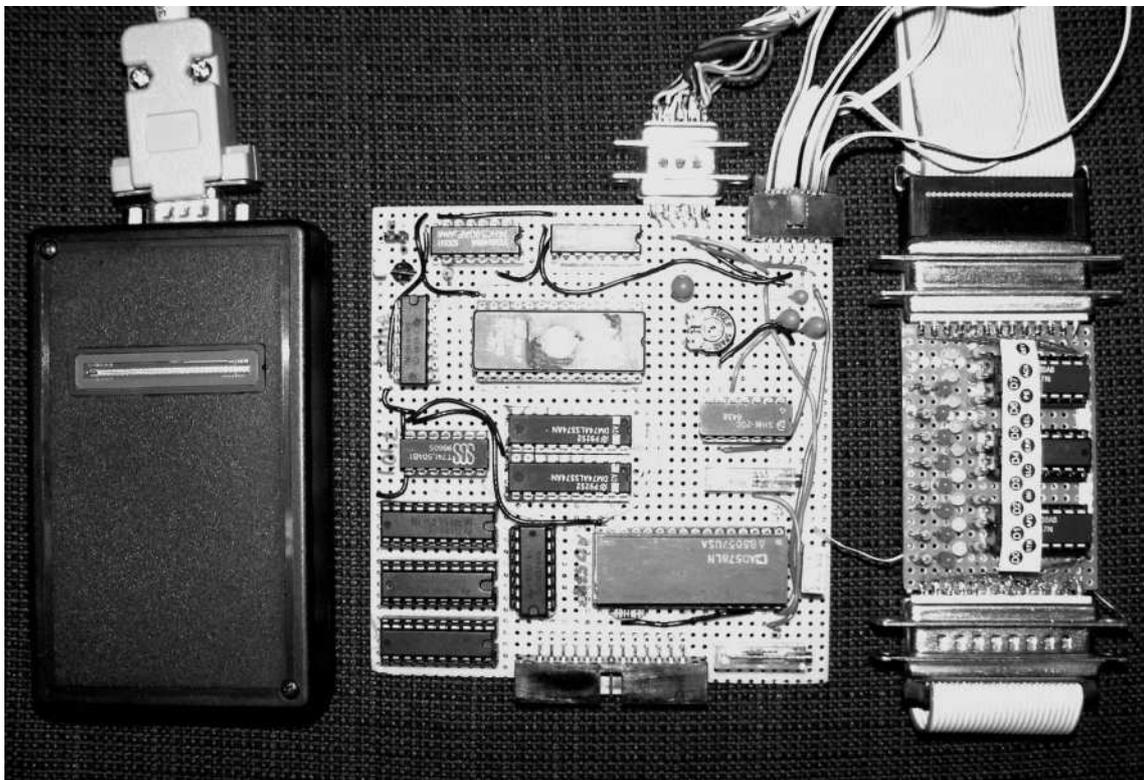


# ***Konstruktion einer Ansteuerung für Single-Line- CCD-Sensoren***



**Wettbewerb "Jugend Forscht" 2005**

**Sebastian Hellberg (16 Jahre)**

**Daniel Markx (15 Jahre)**

**Arbeitsgemeinschaft "Jugend Forscht"  
des Christian-Gymnasiums Hermannsburg  
Leitung: StD Thomas Biedermann**

# Inhaltsverzeichnis

1.	Einleitung	3
2.	Grundsätzliches zur Bilderfassung	3
2.1.	Analoge Fotografie	3
2.2.	Digitale Fotografie	3
3.	Von der optischen Abbildung zum Digitalbild im PC	4
3.1.	Einsatzgebiete von Single-Line-CCD's	4
3.2.	Probleme bei der zeilenweisen Abtastung von Bildern	4
3.3.	Funktion und Aufbau von Single-Line-CCD's	5
3.3.1.	Ladungstransport von der Photodiode in den Kondensator:	5
3.3.2.	Ladungstransport in der Eimerkette	6
3.4.	Spezifikation des CCD TCD1201D von Toshiba	6
3.5.	Anforderungen an die Schaltung um ein CCD herum	8
3.5.1.	Signalerzeugung	8
3.5.2.	Teil 2: Verstärker und Analog-Digital-Wandler	9
3.5.3.	Schnittstelleninterface zum PC	9
3.6.	Anforderungen an die PC-Software	10
4.	Konstruktionsbeschreibung	10
4.1.	Die Schaltung um unseren Bildsensor herum	10
4.1.1.	Signalerzeugung	10
4.1.2.	Analoge Datenverarbeitung	10
4.1.3.	Digitale Datenverarbeitung	11
4.1.4.	Der praktische Aufbau der Hardware	11
4.1.5.	Eine "nebenbei abgefallene" Schaltung - die Parallelport-Pegelanzeige	12
4.2.	Software	12
4.2.1.	Software zur Programmierung des Eproms	12
4.2.2.	Software zum Einlesen der CCD-Bilddaten	13
4.3.	Systemtest	14
5.	Messungen am CCD	14
6.	Probleme bei Schaltungsaufbau und Softwareentwicklung	14
7.	Ausblick	15
8.	Quellen:	15
9.	Danksagung	15
<b>Anhang:</b>		
	Timing-Diagramm des TCD1201D	16
	Benutzeroberfläche des Mustergenerators	16
	Q-Basic-Programm	17
	Schaltplan des Interfaces	18
	Schaltplan des Parallelport-Adapters	21

# 1. Einleitung

Eine Kamera mit CCD-Single-Line-Sensor? Das klingt beim ersten Lesen vielleicht komisch. Eine Kamera, die nur eine Bildzeile aufnimmt? Ist das wirklich sinnvoll?

Die Antwort ist: Ja, denn man kann mit ihr Bilder von bewegten Objekten aufnehmen. Genau damit haben wir uns schon in unserer vorherigen Arbeit befasst. Dort haben wir aus einem alten Hand-scanner eine Kamera gebaut, mit der wir mit Hilfe eines Teleskops Bilder von dem Mond, der sich durch die Erdrotation scheinbar bewegt, gemacht. Damit haben wir bewiesen, dass dieses Prinzip funktioniert. Jedoch war bei diesem Projekt die Aufnahmequalität der Kamera nicht besonders gut, sodass wir uns zu einer kompletten Neuentwicklung mit freundlicherweise von Toshiba gespendeten CCD-Bildsensoren entschieden haben.

Dieses Jahr haben wir uns in erster Linie mit der Technik um einen CCD-Line-Bildsensor herum befasst und eine komplette Schaltung sowie einen Prototyp des zugehörigen PC-Programms entwickelt.

## 2. Grundsätzliches zur Bilderfassung

Um ein Bild eines Gegenstandes festhalten zu können, braucht man mehrere Dinge:

- eine Optik, die aus dem virtuellen Bild des Gegenstandes ein reelles Bild erzeugt
- einen Film (analoge Fotografie) oder einen elektronischen Bildsensor (digitale Fotografie)

In den folgenden Abschnitten beschreiben wir die Theorie der analogen und der digitalen Aufnahme von Bildern.

### 2.1. Analoge Fotografie

Bei der analogen Fotografie wird mit Filmen gearbeitet, die aus lichtempfindlichen chemischen Substanzen, die auf einem Trägermaterial aufgebracht wurden, bestehen.

Mithilfe einer Optik wird ein reelles Bild eines Gegenstandes auf dem eben beschriebenen Film projiziert. In der Optik befindet sich ein so genannter Verschluss, der nur für eine bestimmte Zeit, die Belichtungszeit, geöffnet wird und Licht auf den Film fallen lässt. Während Licht auf den Film fällt, verändern sich durch die auftreffenden Photonen die chemischen Substanzen. Je nach Lichtmenge ist dieser Vorgang mehr oder weniger stark ausgeprägt. Das Abbild dieses Objektes ist nun auf dem Film gespeichert.

### 2.2. Digitale Fotografie

Bei der digitalen Fotografie wird hingegen mit einem elektronischen Flächenbildsensor gearbeitet, der viele Bildpunkte (Pixel) auf einer meist rechteckigen Fläche aufnimmt. Das auf einen Pixel des Sensors auftreffende Licht wird dort in eine elektrische Spannung umwandelt und nicht wie bei einem Film chemisch sondern elektrisch gespeichert. Hier ist ebenfalls eine Optik nötig, die ein reelles Bild des

Gegenstandes auf dem Bildsensor entstehen lässt. Die Länge der Belichtung ist auch hier für ein detailgetreues Bild sehr wichtig, im Gegensatz zum Film muss sie aber nicht zwingend durch einen Verschluss gesteuert werden, sondern ist bei einigen CCD's elektronisch einstellbar. Nachdem das CCD für eine bestimmte Zeit belichtet wurde, können die analogen Bilddaten anschließend mithilfe einer elektronischen Schaltung ausgelesen, in ein für den PC verarbeitbares Digitalsignal umgewandelt und auf einem Speichermedium gespeichert werden. Nach einem solchen Prinzip funktioniert zum Beispiel eine handelsübliche Digitalkamera.

### **3. Von der optischen Abbildung zum Digitalbild im PC**

Um ein Bild zu erfassen, muss für jeden Bildpunkt die Helligkeit bestimmt werden. Im einfachsten Fall könnte man dazu eine große Zahl von Photodioden nehmen und sie alle nacheinander auslesen. Der hierfür erforderliche Schaltungsaufwand wäre sehr hoch und wird deshalb in der Realität normalerweise nicht verwendet.

Die Anordnung der Bildpunkte in Zeilen mit jeweils der gleichen Anzahl von Bildpunkten (Pixeln) erlaubt eine Vereinfachung des Schaltungsaufwandes, wenn man sich des sogenannten Eimerkettenprinzips bedient. Dort werden die Helligkeiten der einzelnen Bildpunkte in Form von Ladungen in kleinen Kondensatoren gespeichert und durch einen gesteuerten Ladungstransport von Zelle zu Zelle weitergereicht. Jeweils an der letzten Zelle wird die Ladungsmenge gemessen und weiter verarbeitet. Dieses Prinzip findet man in den Single-Line-CCD's, die nur eine Bildzeile erfassen können.

Für ein flächiges Bild werden mehrere solcher Zeilen auf dem Chip übereinander angeordnet. Um nun das ganze Bild zu erhalten, wird eine Zeile nach der anderen wie beim Single-Line-CCD ausgelesen. Obwohl dieser Vorgang aufwändiger ist, bleibt das Prinzip jedoch insgesamt das gleiche.

Da wir uns mit der Entwicklung einer entsprechenden Schaltung befassen, haben wir uns zunächst auf Single-Line-CCD's konzentriert, da diese den einfachsten Fall einer Eimerkettenschaltung darstellen. Die dabei gewonnenen Erfahrungen sollen dann später für Flächensensoren weiter verwendet werden.

#### **3.1. Einsatzgebiete von Single-Line-CCD's**

Die Einsatzgebiete von CCD-Single-Line-Bildsensoren sind vielfältig: Sie werden im häuslichen Bereich zum Beispiel in Flachbett-, Hand- und Einzugsclannern oder in Faxgeräten, in der Industrie etwa für Dickenmessungen bei Walzstraßen verwendet. Alle für die Erfassung flächiger Bilder verwendeten Geräte haben dabei aber eines gemeinsam: Zum Erfassen des gesamten Bildes wird der Sensor entweder über das Dokument oder das einzulesende Dokument über den Sensor bewegt. Bei Dickenmessungen genügt dagegen die Information einer einzigen Zeile, wenn der Sensor genau senkrecht zur Bewegungsrichtung des zu messenden Objektes angeordnet wird.

#### **3.2. Probleme bei der zeilenweisen Abtastung von Bildern**

Bei einem Single-Line-CCD muss entweder der Sensor oder das auf ihn projizierte Bild bewegt werden. Bei Dokumenten ist das ja eigentlich kein Problem, da diese relativ leicht zu bewegen sind.

Schwieriger wird das aber bei feststehenden Objekten, wie zum Beispiel einem Baum: Er kann nicht einfach so wie ein Blatt Papier bewegt werden. Also bleibt nur noch die Bewegung des Sensors, um das gesamte Bild abtasten zu können. Man könnte zum Beispiel das CCD mit einem kleinen Schrittmotor immer um eine Zeilenbreite verschieben und eine Zeile aufnehmen, bis das gesamte Bild abgetastet ist. Dabei ergibt sich das Problem, dass für die Nachführung des Sensors eine sehr präzise Mechanik notwendig ist, da die Zeilenbreite im Bereich einiger Mikrometer liegt. Der Bau einer solchen Mechanik ist sehr schwierig und zeitaufwändig.

Man könnte aber auch einen beweglichen Oberflächenspiegel in den Strahlengang der Optik einbauen, um das Bild über den Sensor zu führen. Die Drehung könnte zum Beispiel mit einem Zeigerdrehspulinstrument bewerkstelligt werden. Bei dieser Variante ergeben sich dieselben Probleme wie bei der vorigen, jedoch ist dieser Aufbau zusätzlich sehr erschütterungsempfindlich.

Aufgrund dessen haben wir beschlossen, keine Sensornachführungseinrichtung zu konstruieren, sondern haben uns auf die Aufnahme bewegter Gegenstände, wie zum Beispiel ein langsam fahrendes Auto oder astronomische Objekte, beschränkt.

### 3.3. Funktion und Aufbau von Single-Line-CCD's

In unserer letzten Arbeit sind wir bereits ausführlich auf die Vorgänge in CCD's eingegangen, deshalb wiederholen wir hier nur kurz die wichtigsten Prinzipien, wie sie für unsere eigene Schaltungsentwicklung benötigt werden.

#### 3.3.1. Ladungstransport von der Photodiode in den Kondensator:

In Abbildung 1 (nach [2]) ist der Ladungstransport von der Photodiode in den zugehörigen Kondensator der Speicherkette zu sehen.

Im Zeitraum  $t_0 - t_1$  wird die Photodiode belichtet, dadurch werden in der Sperrschicht Elektronen freigesetzt.

Im Zeitraum  $t_2$  wird eine Spannung an das Shift-Gate (SH) angelegt und somit die Elektronenbarriere geöffnet. Jetzt können die freigesetzten Elektronen aus der Sperrschicht in den Kondensator fließen.

Im Zeitraum  $t_3$  wird die Spannung an SH wieder abgenommen. Es entsteht wieder eine Ladungsbarriere. Der Transfervorgang ist nun abgeschlossen. Man sieht, dass nicht alle Elektronen in den Kondensator abgeflossen sind. Das liegt daran, dass beim Anlegen einer Spannung an SH zwei Kon-

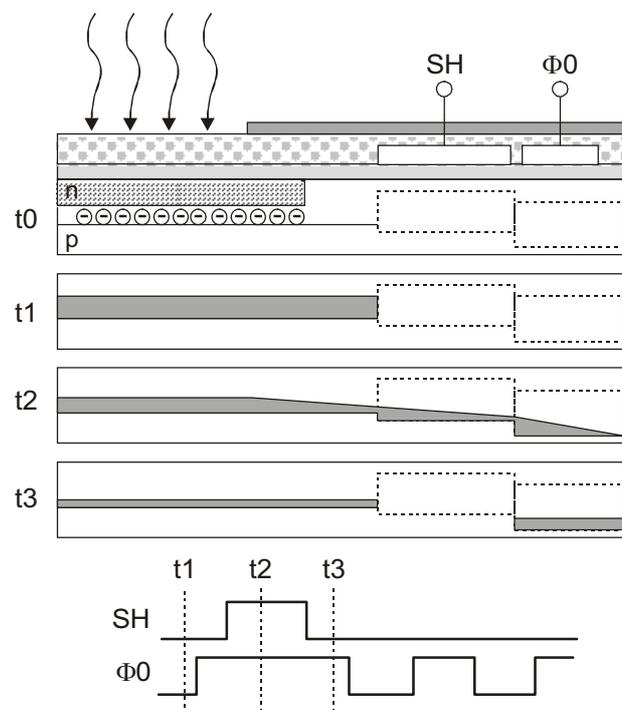


Abb. 1: Belichtung eines Pixels und anschließender Ladungstransport in die Eimerkette, unten ist das zugehörige Zeitdiagramm angegeben

ensoren parallelgeschaltet werden. Es findet also nur ein Ladungsausgleich statt. Die so übertragenen Ladungen können nun in der Eimerkette verschoben werden.

### 3.3.2. Ladungstransport in der Eimerkette

In Abbildung 2 ist der Ladungstransport von Kondensator zu Kondensator zu sehen.

Im Zeitraum  $t_0$  liegt nur an der Elektrode  $\Phi_1$  eine Spannung an, somit ist unter der Elektrode eine Ladungssenke. Unter den Elektroden  $\Phi_2$  und  $\Phi_3$  befindet sich eine Ladungsbarriere, da keine Spannung an ihnen anliegt,

Im Zeitraum  $t_1$  wird an Elektrode  $\Phi_2$  eine Spannung angelegt, somit entsteht unter ihr eine Ladungssenke. Die Elektronen verteilen sich in den Ladungssenken unter  $\Phi_1$  und  $\Phi_2$ .

Im Zeitraum  $t_2$  wird die Spannung an Elektrode  $\Phi_1$  langsam weggenommen, es entsteht eine Ladungsbarriere. Die Elektronen fließen in die Ladungssenke unter  $\Phi_2$ .

Im Zeitraum  $t_3$  ist die Spannung an Elektrode  $\Phi_1$  ganz weggenommen. Es befindet sich nur noch unter  $\Phi_2$  eine Ladungssenke, in der sich jetzt alle Elektronen befinden, die sich vorher unter  $\Phi_1$  befanden. Der Ladungstransport in die nächste Zelle ist nun vollständig.

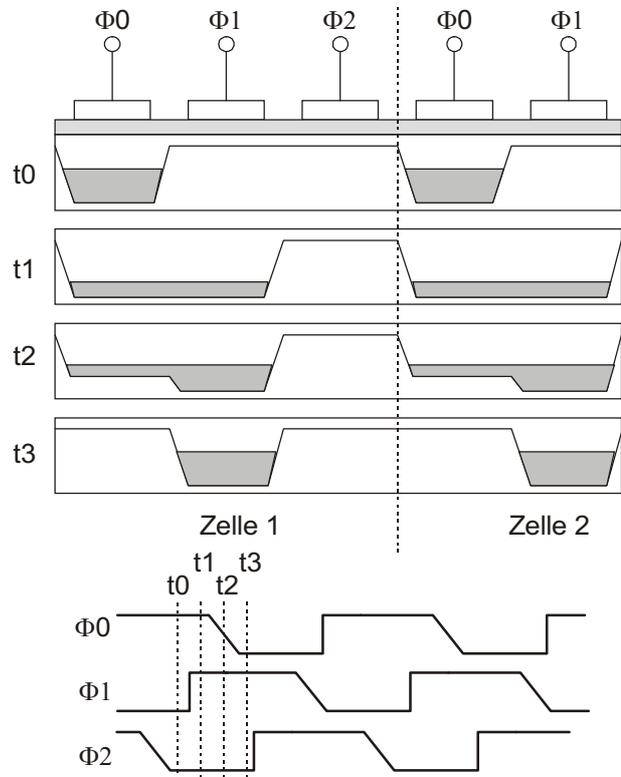


Abb. 2: Zeitlicher Ablauf des Ladungstransportes in der Eimerkette bei Verwendung eines Mehr-Phasentaktes

### 3.4. Spezifikation des CCD TCD1201D von Toshiba

Das TCD1201D ist ein hochempfindlicher Bildsensor mit einer Zeilenbreite von 2048 Pixeln, von denen jedes einzelne  $14 \times 200 \mu\text{m}$  groß ist. Dieser Sensor wird typischerweise in Barcode-scannern, zum Beispiel an der Supermarktkasse, eingesetzt. Dieser Baustein benötigt zum Betrieb im Gegensatz zu vielen anderen Sensoren nur eine einzige  $+5 \text{ V}$  Versorgungsspannung. Sein DIP-Gehäuse, welches aus Keramik mit einem eingesetzten Glasfenster besteht, besitzt 22 Anschlusspins.

CIRCUIT DIAGRAM

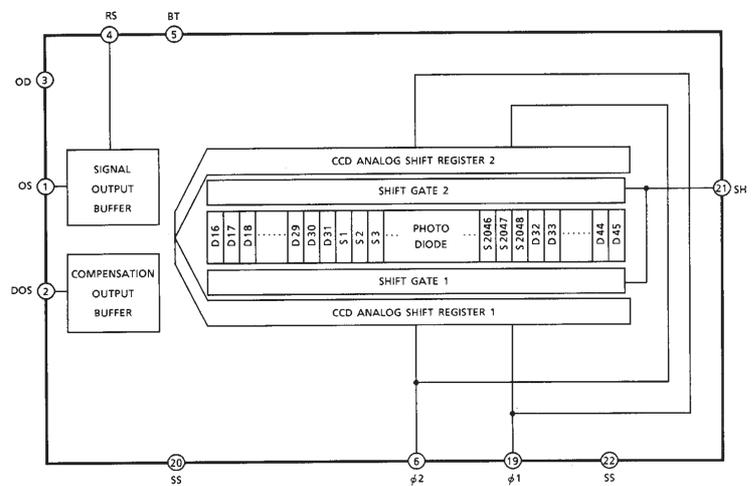


Abb. 3: Innenschaltung des TCD1201D [2, S. 93]

Die Lichtempfindlichkeit liegt nach Herstellerangaben gemittelt über alle sichtbaren Wellenlängen des Lichtes bei ca. 80 V/Lx·s.

In Abb. 3 ist der Innenaufbau des Sensors beschrieben. Das zentrale Element ist das Photodiodenarray mit den einzelnen Speicherkondensatoren und der zugehörigen Elektronik für die Eimerkettenschaltung, mit der die Speicherzellen sequentiell ausgelesen werden. Links im Bild sind die beiden Ausgangspuffer zu sehen, von denen der untere ein Kompensationssignal, der obere das eigentliche Bildsignal liefert. Die weiteren Anschlüsse gehen an die Photoelektronik und werden benötigt, um den Auslesevorgang zu steuern.

Die in Abb. 3 gekennzeichneten Anschlusspins haben dabei folgende Bedeutung:

Signal	Englische Bezeichnung	Deutsche Bezeichnung
$\Phi 1$	Clock (Phase 1)	Zeittakt (Phase 1)
$\Phi 2$	Clock (Phase 2)	Zeittakt (Phase 2)
BT	Boost Pulse	Internes Verstärker-Signal
RS	Reset Gate	Ausgangspuffer zurücksetzen
OS	Output Signal	Pixelsignal-Ausgang
DOS	Dark Output Signal	Dunkelsignal-Ausgang
OD	Power	Stromversorgung
SS	Ground	Masse

Das CCD benötigt einen 2-Phasen-Takt ( $\Phi 1$  und  $\Phi 2$ ) sowie einen Reset- und Boostpuls sowie ein Shift Gate - Ansteuersignal. Aus der Abb. 3 lässt sich außerdem entnehmen, dass es sich bei diesem Single-Line-CCD um eines mit 2 Shift Gates handelt, zwischen denen aber intern umgeschaltet wird, sodass dies von uns nicht weiter zu berücksichtigen ist. Die beiden Ausgänge OS und DOS stellen einen Differentialausgang dar, das heißt: das tatsächlich weiter zu verarbeitende Signal ergibt sich aus der Differenz der beiden Signale. Dies dient vor allem dazu, die von der Umgebungstemperatur abhängige Dunkelspannung der Pixel aus den Bilddaten eliminieren zu können.

Die Bedeutung und gegenseitige Abhängigkeit der Signale ist im Timing-Diagramm des CCD's (siehe Anhang, Abb. A1) definiert. Das Auslesen einer Bildzeile wird mit einer High-Impuls auf der Leitung SH gestartet. Synchron zu den beiden Taktsignalen  $\Phi 1$  und  $\Phi 2$ , die zueinander invertiert sind, müssen die Signale RS und BT erzeugt werden, die laut Spezifikation leicht gegeneinander verschoben sein müssen. Mit der Low-Flanke des  $\Phi 1$ -Signals werden dann die Daten an OS und DOS gültig und müssen nun verarbeitet werden, bevor RS wieder high wird. Die ersten 16 Bilddaten sind inhaltsleer, danach folgen 13 Daten von abgedunkelten Pixeln als Dunkelreferenz und wieder 3 Leerdaten (Dummy-Elemente). Nun folgen die eigentlichen Bildelemente (2048 an der Zahl), gefolgt von wiederum 14 Dummy-Elementen. Insgesamt werden also 2094 Bilddaten-Elemente produziert. Die Erzeugung der Signale  $\Phi 1$  und  $\Phi 2$ , SH, RS und BT muss von unserer Elektronik vorgenommen werden. Die zugehörigen Mindestzeiten sind in der Spezifikation angegeben, die entscheidende Angabe ist hier die maximale Clock Pulse Frequenz von 1 MHz, bei der 2 Pixel ausgelesen werden. Damit dauert das Belichten und Auslesen einer Zeile mindestens  $(2094 / 2) \cdot 1 \mu\text{s} = 1,047 \text{ ms}$ .

### 3.5. Anforderungen an die Schaltung um ein CCD herum

Leider ist es so, dass man einen digitalen Bildsensor, egal welchen Typs, nicht einfach per Kabel an einen Computer anschließen und Bilder aufnehmen kann. Es ist dafür eine elektronische Schaltung erforderlich, die im Allgemeinen aus drei Teilen besteht:

- Takt- und Signalerzeugung
- Analogdaten-Verarbeitung
- Digitale Datenschnittstelle

#### 3.5.1. Signalerzeugung

Wie bereits in 3.3 erwähnt, benötigt ein CCD-Single-Line-Bildsensor ein mehrphasiges digitales Takt-signal sowie weitere Steuersignale, damit die Bilddaten ausgelesen werden können. Die zugehörige Signalfolge ist vom Hersteller des Bildsensors vorgegeben und in Form des Timing-Diagramms im Datenblatt zu finden. Dieses Signal muss von der Elektronik erzeugt werden, damit der Sensor arbeiten kann. Die Erzeugung dieser Signale kann nach mehreren verschiedenen Prinzipien erfolgen:

##### **Verwendung eines vorgefertigten Bausteins eines CCD-Herstellers:**

Diese Methode ist die einfachste. Der Baustein des Herstellers muss nur an das CCD, die Versorgungsspannung und an einen Rechteck-Signalgenerator angeschlossen werden. Die Frequenz des Rechteck-Signals bestimmt die Auslesegeschwindigkeit und somit auch die Belichtungszeit. Das Problem hierbei ist, dass nicht alle Hersteller solche Bausteine im Programm haben und es für normale Menschen fast unmöglich ist, an diese Bausteine heranzukommen. Somit kommt diese Option für uns nicht in Frage.

##### **Erzeugung der Signale durch eine sequentielle digitale Logikschaltung:**

Diese Methode ist relativ kompliziert: Es muss eine komplette digitale Logikschaltung, bestehend zum Beispiel aus Digitalzählern, Invertern, Gattern oder Verzögerungsstufen, entworfen werden. Eine solche Schaltung ist schwierig zu entwickeln, erfüllt die Signalvorschriften des Sensors aber sehr zuverlässig und ist mit verhältnismäßig wenig Aufwand zu ändern. Wegen des hohen Entwicklungsaufwands haben wir diese Option nicht näher verfolgt.

##### **Erzeugung der Signale mithilfe eines EPROM's**

Das Timing-Diagramm zeigt, dass die Signale immer in der gleichen Abfolge erzeugt werden müssen. Zu jedem beliebigen Zeitpunkt ist der Status aller Signale also genau bekannt. Es genügt also, wenn man ein hinreichend feines Zeitraster über das Timing-Diagramm legt, das alle Spezifikationen erfüllt und dementsprechend den Zustand aller Signale zu jedem dieser Zeitpunkte auf geeignete Weise speichert. Das Abrufen der gespeicherten Zustände in zeitlicher richtiger Reihenfolge erzeugt dann den gewünschten Signalverlauf. Die Speicherung erfolgt wegen der kurzen Zugriffszeiten und der benötigten Nicht-Flüchtigkeit in einem EPROM, das ist ein durch UV-Licht löschbares (**E**rasable), mit einem geeigneten Gerät elektrisch programmierbares **R**ead-**O**nly-**M**emory.

Um die gespeicherte Signalfolge abzurufen, werden die Adressleitungen des EPROM's von einer digitalen Zähler-schaltung binär durchgezählt. An den einzelnen Datenausgängen liegen nun die gewünschten Signale an. Je nachdem, wie schnell die Adressen durchgezählt werden, ändert sich auch die Signalgeschwindigkeit am Ausgang und somit die Auslesegeschwindigkeit sowie die Belichtungszeit des Bildsensors. Wegen ihrer Einfachheit und der Möglichkeit, die Signalfolgen schnell anpassen zu können, haben wir uns für diese Methode entschieden; eine genauere Beschreibung dieser Schaltung und der Software zur EPROM-Dateierzeugung ist in Kapitel 4 zu finden.

### **3.5.2. Teil 2: Verstärker und Analog-Digital-Wandler**

Aus den beiden Signalausgang OS und DOS des CCD's kommt jeweils ein Analogsignal heraus, das je nach Helligkeit des entsprechenden Pixels - zwischen 0 und +5V liegt. Diese Ausgänge sind hochohmig und damit empfindlich gegen Störungen auf den Leitungsverbindungen. Eine Transistor-schaltung nimmt eine Impedanzwandlung vor, ohne die Signalhöhe zu verändern. Der nun niederohmige Ausgang ist nun wesentlich störungsempfindlicher.

Wenn die +5V Signalhöhe am Ausgang nicht ausreichen, um den nachfolgenden A/D-Wandler (ADC) voll auszusteuern, weil zum Beispiel sein Eingangspegel zur Vollaussteuerung +10V betragen muss, wird ein weiterer Verstärker, normalerweise ein Operationsverstärker, zwischen Impedanzwandler-Ausgang und ADC-Eingang geschaltet werden. Dieser hebt den Pegel um den Faktor 2 auf maximal +10V Signalhöhe an.

Der A/D-Wandler hat die Aufgabe, das für PCs "unverständliche" Analogsignal in ein digitales Signal umzuwandeln, das auch der PC verarbeiten kann.

### **3.5.3. Schnittstelleninterface zum PC**

Doch mit der reinen Umwandlung der analogen Bilddaten in digitale ist es noch nicht getan. Es muss nun noch ein PC-Interface her, das folgende Funktionen beherrschen muss:

- Zwischenspeichern der Bilddaten
- Übertragen der Bilddaten

#### **Zwischenspeichern der Bilddaten:**

Die Daten am Wandlerausgang liegen nur für einen sehr kurzen Augenblick am Ausgang an, da der Wandler, nachdem die Daten ausgegeben wurden, sofort mit dem nächsten Wandlungszyklus beginnt und dabei alle Ausgänge auf Null setzt. Deshalb muss ein Latch die Daten so lange festhalten, bis sie vom PC ausgelesen werden konnten.

#### **Übertragen der Bilddaten:**

PC und CCD-Elektronik laufen aufgrund verschiedener Taktfrequenzen asynchron zueinander. Deshalb muss beim Dateneinlesen ein so genannter Handshake erfolgen, damit ist gemeint, dass sich die Elektronik und der Computer gegenseitig verständigen müssen, wann die digitalen Daten eingelesen werden können und der Einlesevorgang beendet ist.

### 3.6. Anforderungen an die PC-Software

Nicht nur an die Elektronik um das CCD herum werden hohe Anforderungen gestellt, sondern auch an den PC mit seiner Software. Die Software muss folgende Punkte beherrschen, damit die Daten verlustlos eingelesen werden können und nachher ein hochwertiges Bild entsteht:

- Steuerung des Handshakes zum Datentransfer
- Übernahme und Speicherung der Rohdaten
- Umwandeln der Rohdaten in ein Standard-Bildformat

## 4. Konstruktionsbeschreibung

### 4.1. Die Schaltung um unseren Bildsensor herum

Wie vorher schon im Kap. 3 beschrieben, ist auch in unserer Schaltung eine Dreiteilung zu erkennen, die typisch für Schaltungen dieser Art ist. Zu unserer Schaltung haben wir einen Schaltplan erstellt, der im Anhang zu finden ist.

#### 4.1.1. Signalerzeugung

Dieser Teil der Schaltung hat die Funktion, die vom CCD Hersteller geforderte Signalfolge zu erzeugen. Wie schon erwähnt, erzeugen wir diese mithilfe eines EPROMs (in unserer Schaltung IC4). Im EPROM ist die mit dem PC- Takterzeugungsprogramm (siehe Kapitel 4.2.1) erzeugte zeitlich gesteuerte Signalfolge gespeichert. Zusätzlich werden noch die Signale "Sample" (SMPL), welches den ADC steuert, und das Signal "End of Line" (EOL) generiert, welches dem PC und dem Schnittstelleninterface mitteilt, dass die Bildzeile zu Ende ist.

Um die gespeicherte Signalfolge aus dem EPROM abzurufen, werden die Adressleitungen extern durchgezählt. Dazu benutzen wir zwei in Serie geschaltete 8-Bit-Binärzähler vom Typ 74LS590 (Im Schaltplan IC5 und IC6), die am Eingang CCKIN von einem externen Rechtecksignalgenerator ein Taktsignal bekommen, bei dem jede High-Flanke ein Erhöhen des Zählerwertes um 1 bewerkstelligt. Je höher die Frequenz dieses Signals ist, desto schneller läuft der Zähler. Aufgrund dessen erhöht sich auch die Frequenz der Signale, die am CCD ankommen, dadurch erhöht sich auch die Ausleserate des CCD und somit wird die Belichtungszeit kürzer. Wegen des von uns gewählten Zeitrasters muss diese Taktfrequenz um den Faktor 16 höher sein als die für das CCD gewünschte. Um die relativ stör anfälligen Signale am EPROM-Ausgang noch zu puffern und dadurch störunanfällig zu machen, werden sie über Inverter des Typs 74LS04 (IC7A-IC7F) invertiert. Deswegen sind die Signale im EPROM invertiert gespeichert.

#### 4.1.2. Analoge Datenverarbeitung

An die Ausgänge OS und DOS sind zwei diskret mit Transistoren aufgebaute Impedanzwandler angeschlossen (vergl. Kap. 3.5.2). Um die Signalhöhe von 5 auf 10 Volt zu erhöhen, folgt darauf ein als Differenzverstärker geschalteter Breitband-Operationsverstärker (U1, LF356N), an dessen Ausgang das bereits kompensierte Bildsignal SIGOUT abgenommen werden kann.

### 4.1.3. Digitale Datenverarbeitung

Nach der Spezifikation des TCD1201D liegt das gültige Bildsignal nur für eine kurze Zeit an. Um die Zeit zu verlängern, die dem nachfolgenden ADC zur Wandlung zur Verfügung steht, wird das analoge Signal zwischengespeichert. Dazu wird eine als Sample-and-Hold bezeichnete Schaltung verwendet, die mit einem integrierten Baustein (SH1, SHM-20C) realisiert wird. Dieser speichert in einem Kondensator mit vor- und nachgeschaltetem Pufferverstärker das anliegende Signal solange, bis sein Steuerungseingang Sample durch das EOC (End of Convert) vom ADC wieder freigegeben wird.

Der nachfolgende Baustein (ADC1, AD578LM) ist ein hoch integrierter High-Speed Analog-Digital-Wandler mit einer maximalen Auflösung von 12 Bit und einer Wandlerzeit von mindestens 3  $\mu$ s. Da er über eine justierbare interne Takterzeugung verfügt, ist seine Beschaltung sehr einfach. Der Wandlungsvorgang wird flankengesteuert über den Eingang START (Pin 21) gestartet, wenn vom EPROM das Signal /SMPL low wird. Solange der Wandlerprozess läuft, ist das Ausgangssignal EOC high, dies wird gleichzeitig dazu verwendet, das Sample-and-Hold gegen Veränderungen des analogen Ausgangssignals zu sperren, was andernfalls zu einem Digitalisierungsfehler führen würde. Mit der Low-Flanke von EOC werden die an den Ausgängen des ADC anliegenden Binärdaten in zwei nachgeschalteten Latches (IC 9 u. 10, 74LS574) zwischengespeichert.

Zur Übergabe der Daten an den angeschlossenen PC verwenden wir die Druckerschnittstelle. Diese stellt uns 12 Leitungen zum Lesen vom PC und 4 Steuerleitungen zum Lesen vom Interface zur Verfügung. Da mehr als 12 Signale vom Interface zum PC übertragen werden müssen, wird eine Buschaltung benötigt, die außerdem in der Lage ist, die hohen Ströme zu bewältigen, die notwendig sind, um die Datenleitungen des Parallelports gegen die Ausgabe vom PC auf die entsprechenden Pegel zu legen. Hierbei wird der erste Leitungstreiber (IC 12, 74LS645) verwendet, um den Status des Interfaces an den PC zu übermitteln. Der PC kann diesen Status abfragen, wenn er durch einen Low-Pegel an dem Signal INI dessen Ausgänge freischaltet. Der Status ergibt sich durch die Ausgangspegel der beiden Flipflops (IC 15A u. B, 74LS74), die ihrerseits durch die Signale /EOC und /EOL gesetzt und durch die Leitungen STR und SLC seitens des PC zurückgesetzt werden.

Wird an der Leitung INI vom PC ein High-Pegel angelegt, werden die anderen beiden Bustreiber (IC 13 u. 14, 74LS645) aktiviert, diese legen die unteren 8 Bit auf den Datenleitungen D0 .. D7 und die höheren 4 Bit an den Statusleitungen ACK, BUSY, PE und SEL an. Da das BUSY-Signal vom PC invertiert wird, ist vor den Eingang A4 von IC14 ein Inverter geschaltet, das erleichtert die numerische Auswertung der Daten im PC, da so keine weiteren Invertierungen oder Signalanpassungen vorgenommen werden müssen.

### 4.1.4. Der praktische Aufbau der Hardware

Unsere Elektronik ist auf einer kleinen und auf einer großen Lochrasterplatine in Fädertechnik aufgebaut.

Auf der kleinen Platine befindet sich das CCD mit der dazugehörigen Analogverstärkerschaltung. Sie ist in ein Gehäuse eingebaut, da an dieser Schaltung keine Veränderungen mehr vorgenommen werden

müssen. Für das CCD wurde in das Gehäuse ein Ausschnitt vorgenommen. Die beiden Platinen werden per Kabel mit SUB-D-9-Steckverbindern verbunden. Aufgrund der zwei Platinen, die mit einem Kabel miteinander verbunden sind, ist die Mobilität des Sensors gewährleistet.

Auf der großen Lochrasterplatine befinden sich beide Digitalteile der Schaltung, also Takterzeugung und das PC-Schnittstelleninterface, sowie Teile des Analogteils mit Sample-and-Hold und Analog-Digital-Wandler.

Der PC-Anschluss erfolgt über ein Kabel, welches mit einem Steckverbinder an der Platine angeschlossen ist. Im Foto auf der Titelseite sind in der Mitte die Hauptplatine, links die Sensorbox und rechts der Port-Adapter zu sehen.

#### **4.1.5. Eine "nebenbei abgefallene" Schaltung - die Parallelport-Pegelanzeige**

Eine große Hilfe beim Debuggen der Schaltung und der Software war die Parallelport-Pegelanzeige. Wir haben sie eigens für diesen Zweck aufgebaut. Diese Schaltung zeigt uns über 17 LEDs den Pegel jeder einzelnen Portleitung an. Das ist insofern sehr nützlich, da man alle Signale auf einen Blick sehen kann, und nicht wie bei einem Oszilloskop maximal 4 oder bei einem Voltmeter nur 1 Signal auf einmal darstellen kann. Außerdem muss man nicht immer umständlich irgendwelche Messspitzen anklemmen, denn diese kleine Schaltung wird direkt am Parallelport angeschlossen und an die Schaltung kann dann wieder das betreffende Gerät angeschlossen werden.

Da der Parallelport nicht genug Strom liefert, um die LEDs leuchten zu lassen, schalten wir sie über die Open-Collector-Treiber 7417. In einem einzelnen IC sind 6 Stück davon vorhanden, deshalb mussten wir 3 ICs verwenden. Der Schaltplan dieser Schaltung ist im Anhang zu finden.

## **4.2. Software**

### **4.2.1. Software zur Programmierung des Eproms**

Mit einem in Visual-Basic geschriebenen Programm ist es uns möglich, die einzelnen Signalfolgen, die zur Ansteuerung des CCD benötigt werden, zusammenzustellen und in einer Binärdatei zu speichern, die anschließend in das EPROM gebrannt werden kann. Dabei ist es möglich, auf einzelne Signalkomponenten zurückzugreifen, wodurch sich schnelle Änderungen einzelner Abschnitte leicht gestalten lassen. Für jede Komponente kann angegeben werden, in wieviele Taktraster sie aufgelöst werden und wieviele gleichartige Sequenzen hintereinander angelegt werden sollen. Der in jedem Rasterabschnitt gewünschte Signalpegel eines Signals kann durch Klicken mit der Maus geändert werden. Um eine vollständige Signalfolge zu erzeugen, werden die bereits erzeugten Module nacheinander geladen und zu einem Projekt zusammengestellt. Sowohl die Module einzeln als auch das jeweilige Projekt können als Datei abgespeichert werden.

Die Schaltfläche „Binärdatei erzeugen“ analysiert die einzelnen Module und erzeugt entsprechend den Vorgaben eine fortlaufende Binärdatei, die anschließend direkt in das EPROM gebrannt werden kann. Dazu steht uns eine entsprechendes Programmiergerät zur Verfügung.

Die Benutzeroberfläche dieses Programms findet sich im Anhang als Abb. A2.

#### 4.2.2. Software zum Einlesen der CCD-Bilddaten

Natürlich weiß der PC nicht von alleine, wie er die Daten einlesen und die Schaltung steuern muss. Deshalb mussten wir ein Programm schreiben, das dem PC mitteilt, was genau er zu tun hat. Aus Zeitmangel haben wir leider nur einen Prototyp des Programms in QBasic unter DOS schreiben können, welches nur sehr langsam läuft und auch nicht besonders viele Funktionen hat. Der Quellcode ist im Anhang unter A3 auf Seite 17 zu finden

Das Programm ist relativ einfach aufgebaut. In den Zeilen 10 – 30 werden die Konstanten für die Parallelportadressen definiert und die Hex-Werte für einzelne Bitmaskierungen festgesetzt. In Zeile 33 wird ein Array mit dem Namen „linedata“ dimensioniert, für das 2050 Stellen Speicherplatz reserviert werden. In dem Array werden im späteren Programmverlauf die Pixeldaten einer Zeile zwischengespeichert. Danach wird eine Datei angelegt, in der die Bildrohdaten aus dem Array später gespeichert werden. In den Zeilen 50 – 65 wird eine Abfrage gemacht, wie viele Bildzeilen eingelesen werden sollen. Außerdem wird auf einen Tastendruck gewartet, der die Datenabfrage startet. Die Hauptschleife des Programms beginnt in Zeile 67; diese FOR-Schleife läuft so oft durch, wie man zu Anfang des Programms Zeilen als Zahl eingegeben hat. Bei jedem Durchlauf dieser Schleife wird eine ganze Bildzeile aufgenommen. In Zeile 80 - 125 wird ein kurzer Low-Impuls auf der Strobe-Leitung des Parallelports ausgegeben, der die Elektronik veranlasst, einen Zeilendurchlauf zu starten. Danach wird eine Zählvariable für die Pixel der Zeile initialisiert. Danach beginnt eine DO LOOP UNTIL - Schleife, in der bei jedem Durchlauf ein Pixel eingelesen und im Array gespeichert wird. In dieser Schleife wird folgendes getan:

- Warte auf EOC- (End of Convert) an Leitung D0 oder EOL- (End of Line) – Signal an Leitung D1 des Parallelports: Wenn EOL da, gehe aus Schleife raus, wenn EOC da, mache weiter
- Setze INI-Ausgang auf High, um in Datenmodus umzuschalten
- Setze Datenport auf FF
- Lese Daten ein: untere 8 Bit vom Datenport sowie obere 4 Bit vom Statusport
- Füge die 4 und 8 Bit zusammen; shifte dabei die oberen 4 Bit, speichere den Wert im Array und erhöhe den Pixelzähler pixelcnt um 1
- Setze INI auf Low, um in Control-Modus zurückzuschalten
- bestätige Einlesen der Daten durch High-Impuls an SLCT-Leitung, dadurch wird das Flipflop IC5B in der Schaltung zurückgesetzt
- wiederhole diese Schleife so oft, bis der Pixelzähler 2050 erreicht hat.

In Zeile 360 und Folgenden werden dann die Daten des Arrays in einer Datei gespeichert, danach wird die Datei geschlossen, eine Meldung ausgegeben und die Zeilenaufnahme ist fertig.

Die Zeilenaufnahme wird aufgrund der ersten FOR-Schleife so oft wiederholt, wie es am Start des Programmes eingegeben wurde. Nachdem alle Zeilen eingelesen wurde, ist die Bildaufnahme fertiggestellt und das Programm wird beendet.

### 4.3. Systemtest

Nach dem Aufbau der Schaltung haben wir diese zunächst mit einem Durchgangsprüfer auf korrekte Verdrahtung geprüft. Danach haben wir die Elektronik an die Stromversorgung angeschlossen und uns mithilfe eines Oszilloskops vergewissert, dass alle Teile der Schaltung funktionieren. Als das sichergestellt war, haben wir den Sensor in seine Fassung gesteckt und uns auf dem Oszilloskop das Ausgangssignal des CCD's angesehen. Nachdem dies alles richtig war, haben wir die Schaltung an den PC angeschlossen, die Software gestartet und geprüft, ob das Zusammenspiel zwischen Hard- und Software gestimmt hat.

## 5. Messungen am CCD

Um die Lichtempfindlichkeit des CCD auszumessen, haben wir das vom Dunkelstrom bereinigte Analogsignal am Eingang des Sample and Hold mit einem Oszilloskop und die Beleuchtungsstärke mit einem Luxmeter gemessen. Der Sensor wurde zur Hälfte abgedeckt, um eine Hell-Dunkel-Referenz zu bekommen. Die Helligkeit wurde dann so eingestellt, dass das Signal bei maximaler Betriebsfrequenz des CCD von 1 MHz (das entspricht 16 MHz am Clockeingang der Zähler) gerade noch nicht die Aussteuergrenze erreichte. Diese war bei 6 Lux erreicht.

Außerdem haben wir den Sensor versucht, so dunkel wie möglich einzupacken, um ihn vor Umgebungslicht zu schützen. Dann haben wir in der gleichen Messanordnung die Frequenz so weit heruntergedreht, dass wieder das volle Ausgangssignal erreicht wurde. Dies war bei einer Zähler-Clockfrequenz von ca. 10 kHz (entsprechend ca. 625 Hz am CCD) erreicht. Eine Messung der „Helligkeit“ war nicht mehr möglich, da für unsere Begriffe der Sensor (eingepackt in einem doppellagigen schwarzen Rucksack) eigentlich völlig dunkel hätte sein müssen. Dennoch reichte nachweisbar das Ein- und Ausschalten einer Glühlampe, um das Signal nahezu völlig verschwinden zu lassen.

Bei mittlerer „Helligkeit“ konnten wir außerdem feststellen, dass eine Halbierung der Frequenz entsprechend einer Verdoppelung der Belichtungszeit eine Verdoppelung des Signals zur Folge hatte.

Nimmt man diese Messungen zur Grundlage, so erhält man:

6 Lx bei 1 MHz entsprechend einer Belichtungszeit von 1 ms

625 Hz, also eine Belichtungszeit von 1600 ms ergibt eine Empfindlichkeit von 3,75 mLux

Zum Vergleich: das Mondlicht soll etwa eine Beleuchtungsstärke von 3 Lux erzeugen, was uns zuversichtlich macht, dass wir mit diesem Sensor in der Lage sein werden, auch lichtschwache Objekte am Sternenhimmel abbilden zu können, womit wir wieder an unsere Arbeit vom Vorjahr anknüpfen können.

## 6. Probleme bei Schaltungsaufbau und Softwareentwicklung

Aufgrund der hohen Packungsdichte auf der Hauptplatine kam es dort manchmal zu Kurzschlüssen, weil manche Drähte nicht korrekt verlötet waren oder ihre Lackisolierung sich an manchen Stellen

gelöst hat. Außerdem ergab sich ein Problem mit einer unsauberen +12 V Versorgungsspannung, jedoch konnte dies durch den Einbau eines 4700  $\mu\text{F}$  großen Siebelkos zwischen +12 V und Masse behoben werden. Durch ein defektes IC (74LS07) wurde in einem Fall die gesamte Schaltung „lahmgelegt“. Dieser Fehler konnte jedoch relativ schnell durch den Einbau eines neuen IC's behoben werden. Auch mussten wir feststellen, dass die Ausgänge AFD, STR, INI und SLCT des Parallelports nur unter einer gewissen Belastung sauber arbeiten. Da aber ein TTL-IC die Leitungen nicht genug belastet, mussten diese Leitungen durch einen Widerstand gegen Masse zusätzlich belastet werden.

Aufgrund mehrerer Logikfehler im Programmcode musste auch die Software mehrmals umgeschrieben werden, was allerdings kein so schwerwiegendes Problem war.

## 7. Ausblick

Unsere jetzige QBasic-Software ist nur ein Prototyp. Sie hat noch nicht viele Funktionen und läuft sehr langsam. Deshalb wollen wir noch ein Programm in der Programmiersprache C schreiben, welches mehr Funktionen hat und schneller läuft.

Außerdem sollen noch genauere Lichtempfindlichkeitsmessungen in einem total dunklen Raum mit einer regelbaren Lichtquelle, dem Sensor vorgeschalteter Optik und variabler Belichtungszeit des Sensors durchgeführt werden.

## 8. Quellen:

- [1] Wilfried Kuhn: Physik Band I, Westermann Verlag, Braunschweig 1987
- [2] CCD Linear Image Sensor Databook, Toshiba Corporation Japan, 1997
- [3] Dieter Nährmann: Das Große Werkbuch Elektronik Teil B, Franzis Verlag, München 1989
- [4] Dan Gookin: C für Dummies, mitp Verlag, Bonn 2001
- [5] Wallace Wang: Programmieren für Dummies, mitp Verlag, Bonn 2000
- [6] Brian W. Kernighan / Dennis M. Ritchie: Programmieren in C, Carl Hanser Verlag, München, 1983

## 9. Danksagung

Wir bedanken uns bei unserem Betreuungslehrer, Herrn StD Thomas Biedermann, für seine große Geduld und Zeiteinsatz. Außerdem bedanken wir uns bei der gesamten Familie Biedermann, die uns an langen Forschungstagen ertragen und gut gepflegt hat. Unser Dank gilt auch der Firma Toshiba, die uns freundlicherweise 12 CCD's für weitere Experimente sowie ein umfangreiches Datenbuch kostenlos zur Verfügung gestellt hat, sowie der Firma e2v, die uns ebenfalls weitere Sensoren zur Verfügung gestellt hat, mit denen wir uns demnächst näher befassen werden. Last but not least bedanken wir uns bei unseren Eltern für ihre Geduld und Taxifahrerei.

DANKE!!!

# Anhang

TIMING CHART

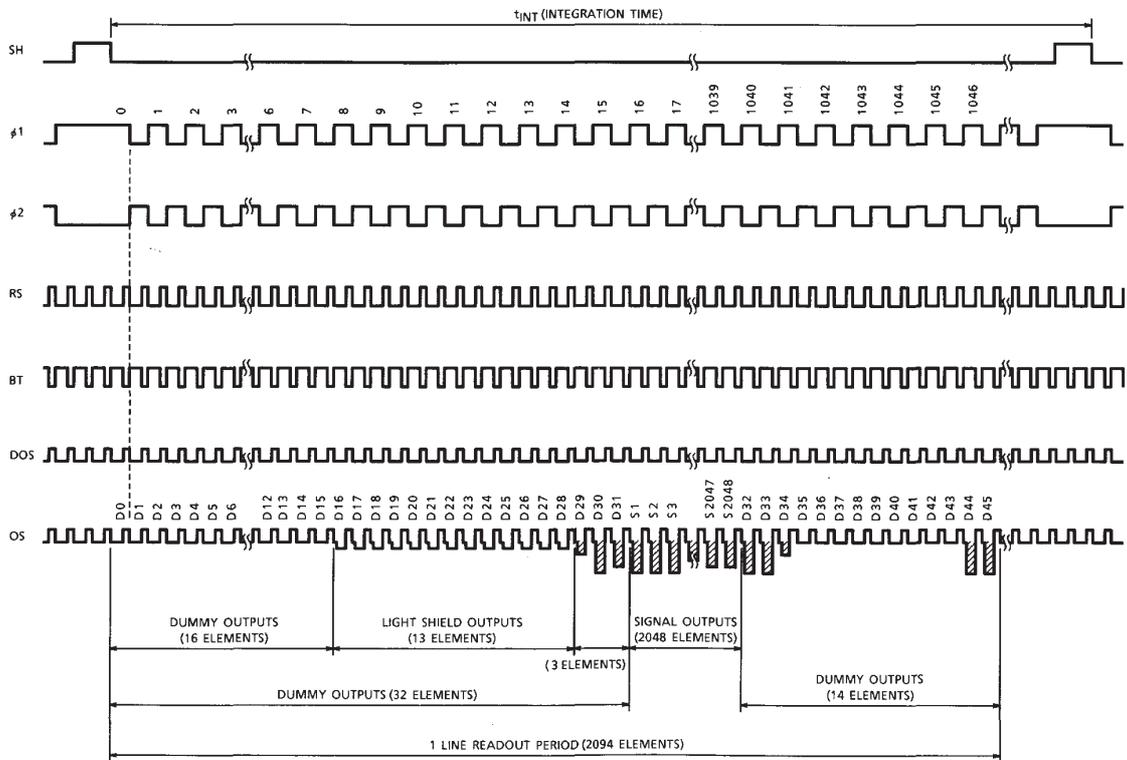


Abb. A1: Timing-Diagramm des TCD1201D [2, S. 97]

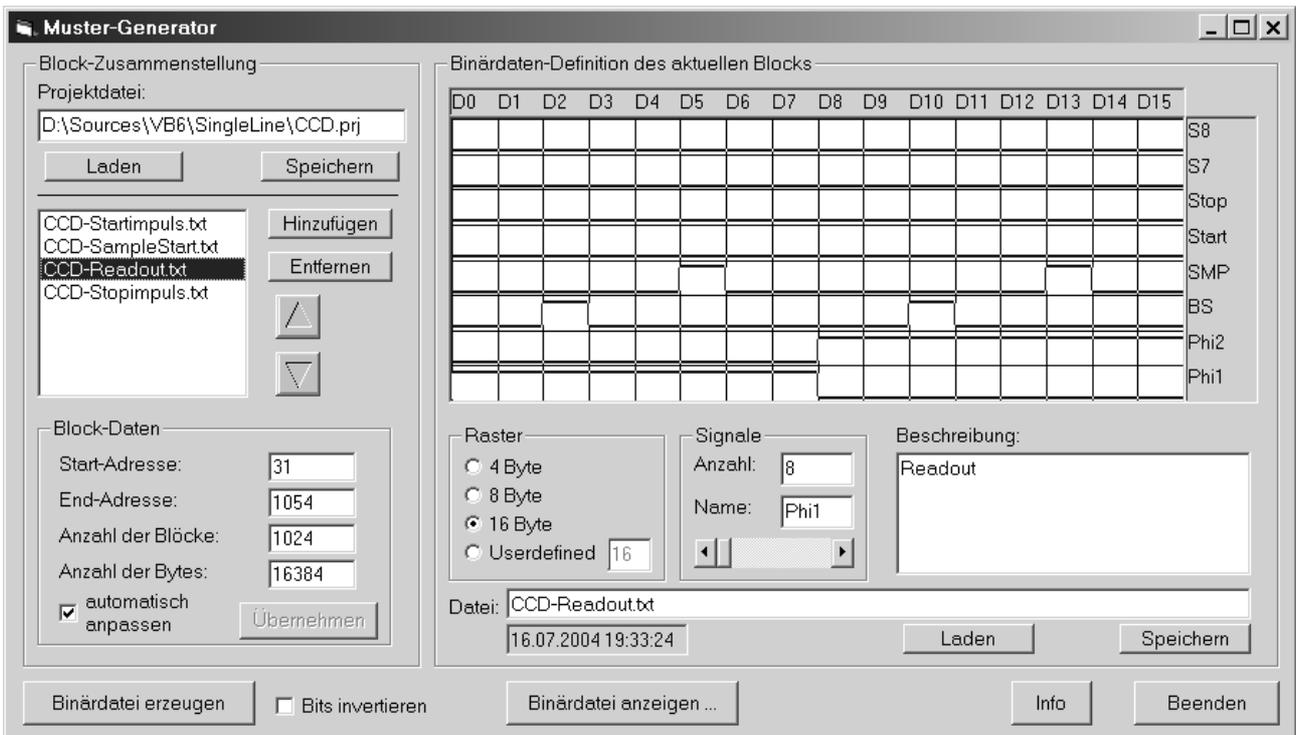
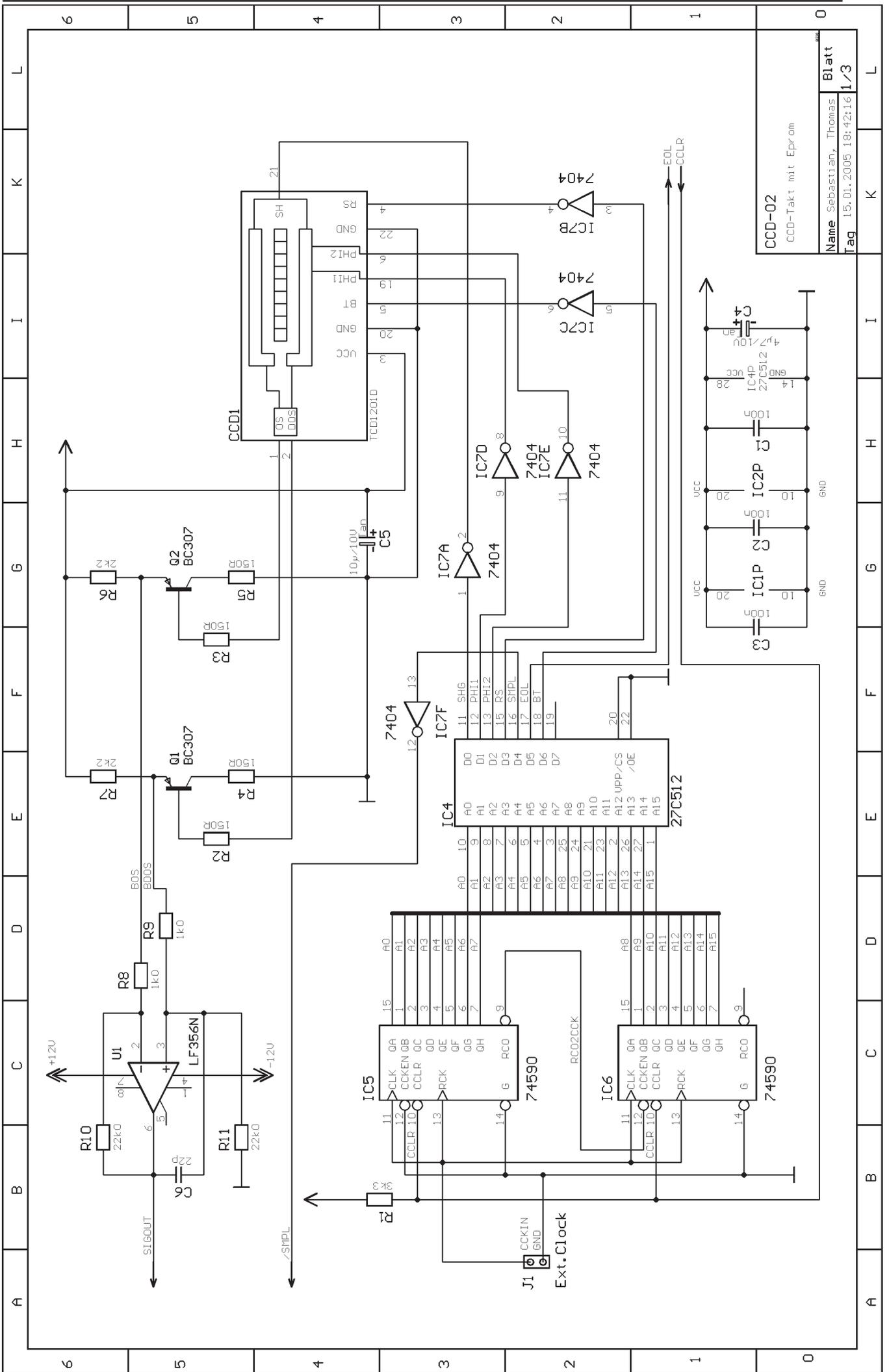
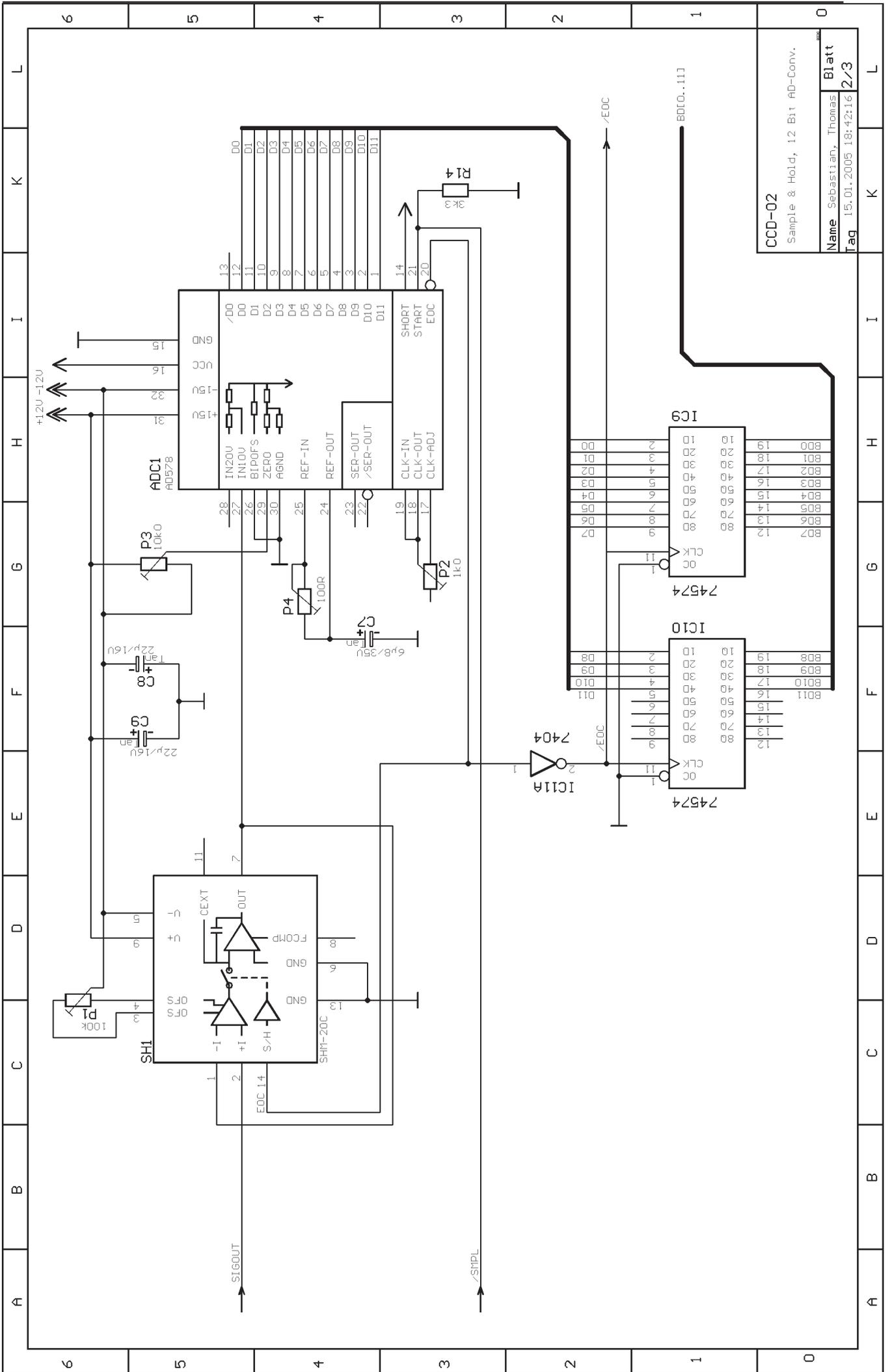


Abb. A2: Benutzeroberfläche des Mustergenerators zur Erzeugung der Signalfolge des EPROM's

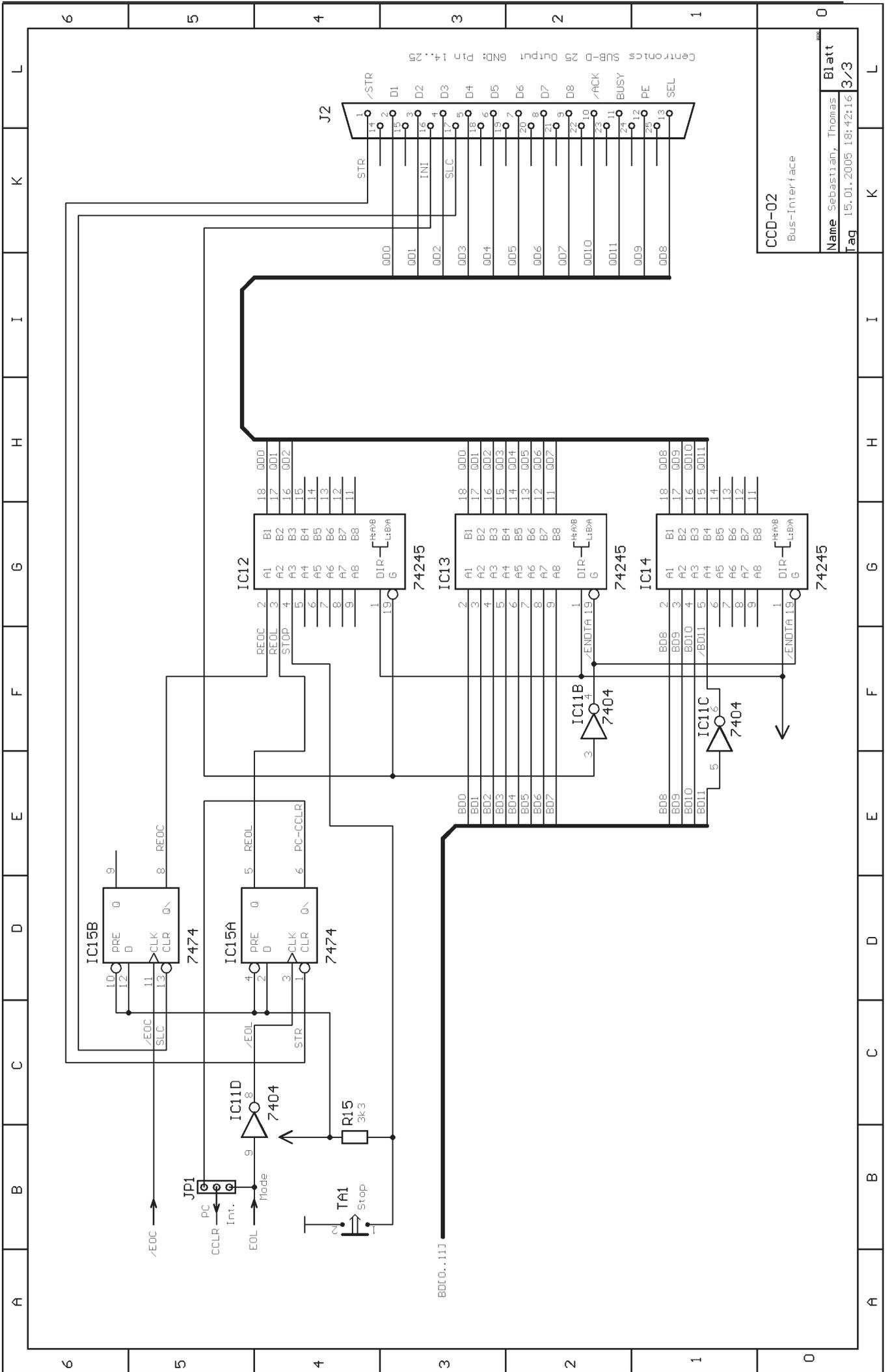
*A3: QBasic-Programm (Prototyp) zur Ansteuerung des Interfaces*

```
10 'Programm zur CCD-Datenerfassung
15 'Deklaration der Konstanten u. Variablen
20 CONST datport = &H378: CONST statport = &H379: CONST ctrlport = &H37A
25 CONST bit1 = &H1: CONST bit2 = &H2: CONST bit3 = &H4: CONST bit4 = &H8
30 CONST bit5 = &H10: CONST bit6 = &H20: CONST bit7 = &H40: CONST bit8 = &H80
33 DIM linedata(0 TO 2050) AS INTEGER: DIM pixeldata AS INTEGER: DIM d0 AS INTEGER: DIM
d1 AS INTEGER
35 pulslength = 10000
37 OPEN "C:\daten\jufo\2005\qb\bild.dat" FOR OUTPUT AS #1
40 'Programmstart
50 CLS
60 PRINT "CCD-Datenerfassung": PRINT : PRINT
62 INPUT "Anzahl der Bildzeilen eingeben: ", lines
65 PRINT "Taste...": WHILE INKEY$ = "": WEND
67 FOR i = 1 TO lines
68     'IF (INP(dataport) AND bit3) = 0 THEN END
70     'Initialisierung
80     OUT (ctrlport), 0
85     PRINT "Initialisiert"
90     'Strobe senden
100    OUT (ctrlport), 1
120    OUT (ctrlport), 0
125    pixelcnt = 0
130    'PRINT "Zeilendurchlauf gestartet"
135    DO
140        'PRINT "Warte auf EOC..."
150        DO WHILE (INP(datport) AND bit1) = 1
155            IF (INP(datport) AND bit2) = 2 THEN GOTO 340
157        LOOP
160        'PRINT "EOC bekommen"
170        OUT (ctrlport), bit3
180        'PRINT "In Data-Modus umgeschaltet"
190        OUT (datport), &HFF
200        d1 = ((INP(statport) AND &HF0) / 16) AND &HF
210        d0 = (INP(datport) AND &HFF)
220        pixeldata = d0 + d1 * 256
230        linedata(pixelcnt) = pixeldata
240        pixelcnt = pixelcnt + 1
245        LOCATE 20, 1: PRINT pixelcnt: LOCATE 20, 10: PRINT HEX$(pixeldata); "    "
250        'PRINT "Daten eingelesen und gespeichert"
260        OUT (ctrlport), 0
270        'PRINT "In Ctrl-Modus zurückgeschaltet"
280        OUT (ctrlport), bit4
300        OUT (ctrlport), 0
310        'PRINT "EOC-Flipflop zurückgesetzt"
330    LOOP UNTIL pixelcnt >= 2050
340    LOCATE 22, 20: PRINT "Line Ready"
350    LOCATE 22, 20: PRINT "          "
360    FOR j = 0 TO pixelcnt
370        PRINT #1, HEX$(linedata(j))
380    NEXT j
390    PRINT #1, "Line end"
395    WHILE (INP(datport) AND bit2) = 0: WEND
400 NEXT i
410 CLOSE #1
420 LOCATE 15, 10: PRINT "Bilderfassung vollst.,ndig"
430 END
```

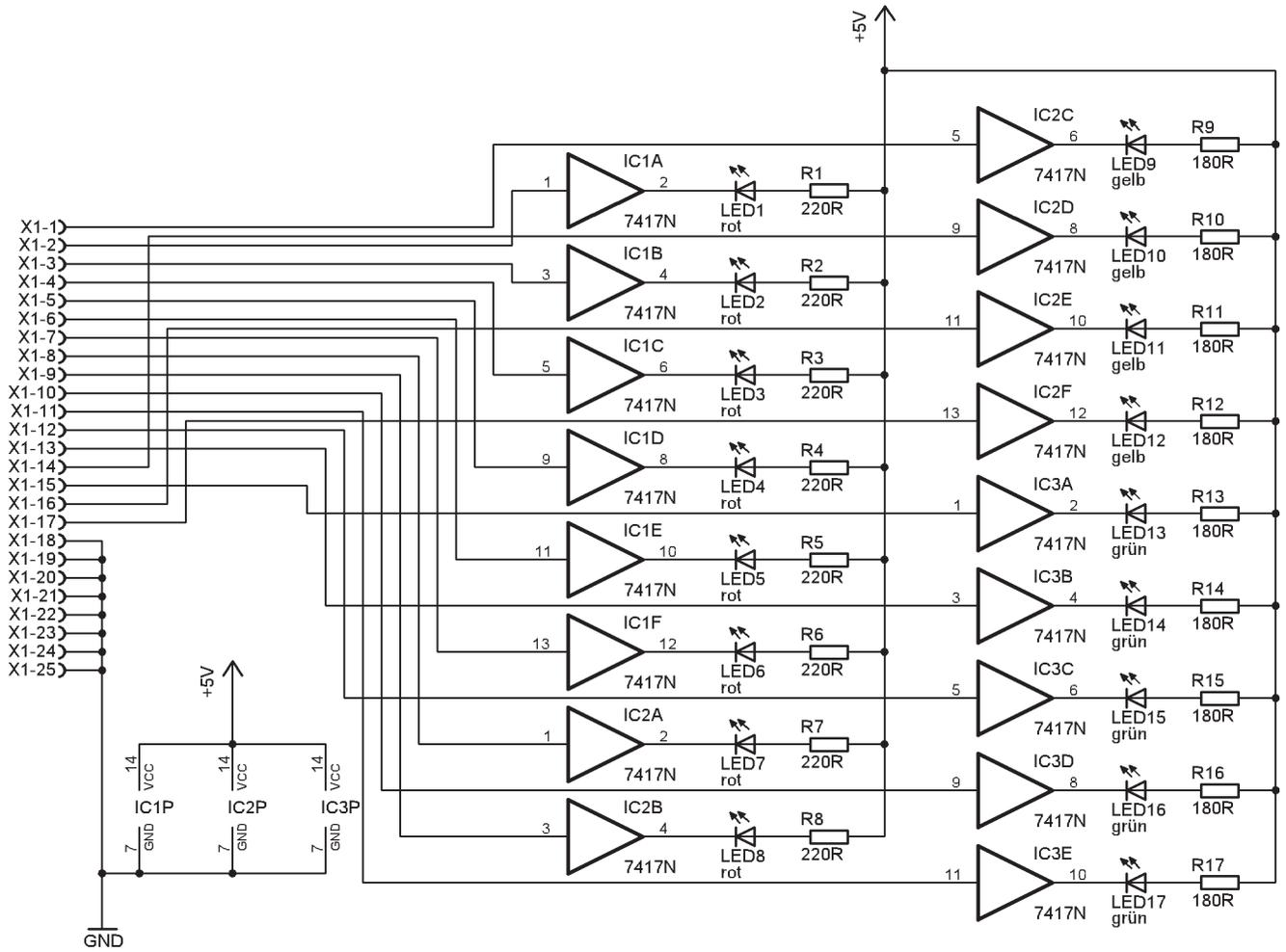




CCD-02  
 Sample & Hold, 12 Bit AD-Conv.  
 Name Sebastian, Thomas  
 Blatt  
 Tag 15.01.2005 18:42:16 2/3



**CCD-02**  
 Bus-Interface  
 Name Sebastian, Thomas  
 Tag 15.01.2005 18:42:16  
 Blatt 3/3



Schaltplan des Adapters zur Port-Pegelanzeige