

# **Amiga Frame Grabber Dokumentation**

Helge Böhme

COLLABORATORS
---------------

	TITLE : Amiga Frame Graber Dokumentation		
ACTION	NAME	DATE	SIGNATURE
WRITTEN BY	Helge Böhme	August 22, 2024	

REVISION HISTORY
------------------

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>Amiga Frame Grabber Dokumentation</b>	<b>1</b>
1.1	Amiga Frame Grabber . . . . .	1
1.2	installation . . . . .	2
1.3	einleitung . . . . .	3
1.4	hardware . . . . .	4
1.5	netzteil . . . . .	4
1.6	analog . . . . .	4
1.7	synchronisation . . . . .	5
1.8	abtastfrequenz . . . . .	6
1.9	digital . . . . .	6
1.10	rams . . . . .	7
1.11	software . . . . .	8
1.12	bildverarbeitung . . . . .	8
1.13	filter . . . . .	9
1.14	farbe . . . . .	10
1.15	bedienung . . . . .	11
1.16	einstellungen . . . . .	12
1.17	Der ARexx-Port . . . . .	14
1.18	Was ist noch zu tun? . . . . .	15
1.19	fehler . . . . .	15
1.20	literatur . . . . .	16
1.21	dank . . . . .	16
1.22	versionen . . . . .	16

## Chapter 1

# Amiga Frame Grabber Dokumentation

## 1.1 Amiga Frame Grabber

Amiga Frame Grabber

Inhaltsverzeichnis

Der Autor	
Erklärung	
-1 Installation	
0 Einleitung - das Konzept	
1 Die Hardware	
1.1 Anschlüsse	
1.2 Das Netzteil	
1.3 Der Analogteil	
1.3.1 Synchronisation	
1.3.2 Zur Abtastfrequenz	
1.4 Der Digitalteil	
1.4.1 Die RAMs	
2 Die Software	
2.1 Schwarzweiß - der einfache Teil	
2.1.1 Die Bildverarbeitungsschritte im einzelnen	
2.2 Digitale Filterung	
2.3 Farbe - der schwierige Teil	
3 Die Bedienungsanleitung	
3.1 Die Einstellungen	
3.2 Der ARexx-Port	
3.3 Was ist noch zu tun?	
3.4 Fehler	
3.4.1 Bekannte Fehler	
Literatur	
Dank	
A Versionen	
B Die Schaltpläne 1 2	
C Das Platinenlayout (Postscript, gespiegelt) oben unten	
D Der Bestückungsplan	
E Die Bauteileliste	

Eigenschaften:

- Digitalisiert einzelne Video-Vollbilder mit 768x575 Pixeln in Farbe

- Wird mittels Scart oder Cinch mit jeder beliebigen FBAS-Quelle verbunden
- Arbeitet (theoretisch) Videonorm-unabhängig
- Signalaufbereitung mittels Software im Amiga
- Preview mit 6 Sekunden / Frame auf der Workbench

Die Hardware:

- Eine Platine mit 7x16cm<sup>2</sup> zum Anschluß an den Parallel-Port
- Digitalisiert wird mit dem TDA8708A mit 16MHz
- Gespeichert wird in zwei Video-Frame-DRAMS HM530281R mit je 32KByte
- Netzteil integriert (softwaregesteuert)
- Schaltplan und PCB-Layout als Postscript erhältlich

Die Software:

- Geschrieben in AmigaE (von Wouter van Oortmerssen)
- Benutzerinterface in EasyGUI (von Wouter/Jason R. Hulance):
- Preview paßt sich automatisch dem Workbench-Screen an
- Zahlreiche Optionen
- Zur Zeit unterstützte Videonormen: PAL
- Adapter Dreizeilenkammfilter
- Speichert Rohe Daten, Bildschirm und Echtfarben
- Lokalisiert
- ARexx-Port
- AppIcon und AppWindow
- Rauschfilter und Farbflankenversteilerung
- Unterstützt OCS bis AGA, 68k
- Läuft ab AmigaOS 2.1
- Erfolgreich getestet auf:
  - Amiga4000LC040 KS39.106(WB3.0)AGA 2C16F1.7HD
  - Amiga500 KS37.175(WB2.1)OCS 0.5C2.3F0HD

Werft einen Blick auf meine Homepage:

<http://www.tu-bs.de/~y0001729>

## 1.2 installation

Installation:

- kopiere "Env/GrabPrefs" nach "Env:" (nicht unbedingt erforderlich)
- kopiere das Verzeichnis "FGrab" wohin auch immer
- kopiere "Catalogs/Deutsch/FrameGrab.catalog" nach "Locale:Catalogs/Deutsch" (nicht unbedingt erforderlich)

Software, die nicht unbedingt installiert ist:

- Reqtools.library (Nico François, Magnus Holmgren)
- gadgets/tabs.gadget (wird umgangen, wenn nicht vorhanden)
- Post.library zum Anzeigen der Schaltpläne und des Platinenlayouts (am besten HWGPost (Heinz Wrobel) und PostV2 (Christian Eibl)
  - beides in Aminet:text/print)
- das PaTeX-Paket zum Anzeigen und Ausdrucken der Doku (in Aminet:text/tex)

Ausprobieren:

---

- starte "ZonePlateStream" (speichern, wohin auch immer)
- starte "FrameGrab" und lade eben gespeicherte Datei

## 1.3 einleitung

### 0 Einleitung – das Konzept

Wie bekomme ich Fernseh- (oder Video-) Bilder in meinen Amiga?

Diese Frage hat mich lange Zeit beschäftigt. Zuhilfe kam mir die Tatsache, daß ich als Student der Elektrotechnik ein paar passende Vorlesungen besuchen durfte:

- Fernsehtechnik I
- Computer-Sehen (Bildverarbeitung und -analyse)
- Fernsehtechnik II

Echtzeit oder nicht oder wie?

Prinzipiell gibt es drei mögliche Arten von Video-Digitizern:

Slow-Scan-Digitizer: Digitalisiert einen (oder wenige Pixel) pro Videozeile ( $64\ \mu\text{s}$ ). Dazu muß ein Standbild im RGB-Format vorliegen. ←  
Wahrlich

keine Echtzeit, ist am besten mit dem Photographieren eines Motivs und anschließenden Einscannen des Bildes zu vergleichen.

Echtzeit-Digitizer: Diese erledigen den gesamten Signalverarbeitenden Teil mittels Hardware. Auch die Umformatierung der Daten auf Amigaspezifische Formate muß onboard geschehen. Das ganze muß dann mittels DMA ins Chip-Memory geschrieben werden. Eine Aufwendige Hardware für den Prozessor- oder Zorro-Bus.

Frame-Grabber: Diese sind in der Lage, Videosignale in Echtzeit einzulesen und in Nicht-Echtzeit zu Verarbeiten, d.h. sie liefern Standbilder aus laufenden Videobildern.

Nach Abwägung der Vor- und Nachteile dieser Typen stand das Konzept für meinen Frame-Grabber ...

Die Hardware wird aus Kosten- und Aufwandsgründen möglichst einfach gehalten. Dies hat zudem noch den Vorteil kleiner Abmessungen der Schaltung. Konzipiert wurde sie Schließlich als eine externe Schaltung zum Anschluß an den Parallelport des Amiga (darum kann sie prinzipiell mit nur kleinen Änderungen auch an anderen Plattformen betrieben werden). Die Aufgaben der Hardware beschränken sich auf:

- Auswahl und Verstärkung des Videosignals
- der Synchronisation
- der Digitalisierung
- der Zwischenspeicherung der Daten
- der Übertragung zum Amiga

Die Software hat nun die große Aufgabe die rohen Videosignale zu verarbeiten. Dieses umfaßt im wesentlichen die Filterung, also die Trennung des Helligkeitssignals (der Luminanz) von dem Farbsignal (der Chrominanz) sowie die Farbddekodierung (das sgn. RGB-Splitting) der entsprechenden Norm (PAL, NTSC,

SECAM). Die Signalverarbeitung per Software hat nun den großen Vorteil der größeren Flexibilität, der nahezu unbegrenzten Erweiterbarkeit und der größeren Präzision bei der Berechnung. Dazu kommt dann nur noch die Darstellung auf dem Bildschirm sowie das Speichern der Bilder.

## 1.4 hardware

### 1 Die Hardware

Möchte man alles verstehen, was hier beschrieben ist, setze ich neben Grundlagen der Elektronik auch die Kenntnis vom Aufbau eines Videosignals voraus. Letzteres ist in der angegebenen Literatur beschrieben. Das Verständnis ist zum Nachbau allerdings nicht erforderlich, nur mit dem Lötkolben sollte man umgehen können (die Platine ist zweiseitig, SMD-Bestückt und hat über 100 Durchkontaktierungen).

#### 1.1 Anschlüsse

Bild: Der Framegrabber im Strapu-Gehäuse

## 1.5 netzteil

### 1.2 Das Netzteil

Das Netzteil ist konventionell aufgebaut: Kabelklemme, Sicherung, Trafo, Gleichrichter, Kondensator, Spannungsregler (5V), Kondensator. Und doch nicht so konventionell: noch ein Spannungsregler (5V) und noch ein Kondensator. Die zweifache Ausführung deshalb, um die analoge Spannungsversorgung von der digitalen zu entkoppeln. Der Eingangskondensator muß mindestens  $2200\,\mu\text{F}$  ←

haben, sonst brummt's d.h. des Videosignal ist alle 10ms eingedellt.

Eine zusätzliche Eigenschaft ist die softwaregesteuerte primärseitige Abschaltung der Schaltung. Ein Opto-Triac wird mittels der /BUSY- (und der 5V-Pullup-)Leitung des Parallelports über einen PNP-Transistor angesteuert. Der verwendete Opto-Triac vom Typ TLP3042 hat einen Zero-Crossing-Detector integriert, so daß beim Schalten keine Spannungssprünge auftreten.

Um den Triac vor Überspannungen zu schützen sind nun noch zwei Schutzelemente vorgesehen: Ein 300V Varistor parallel zu den Eingangsklemmen schützt ihn vor Bursts aus den Netzleitungen (z.B. durch Blitzschlag) und eine Transiltdiode parallel zu den Trafoklemmen vor dem Ausschaltimpuls der Induktivität des Trafos.

## 1.6 analog

### 1.3 Der Analogteil

Das Herzstück, der Analog-Digital-Wandler TDA8708A, stellt gleichzeitig die

Brücke zum Digitalteil dar. Es handelt sich um einen Spezialwandler für Videosignale mit einer maximalen Wandlungsfrequenz von 32MHz. Zur optimalen Aussteuerung hat er einen selbstregelnden Eingangsverstärker (er kann sowohl den Pegel als auch den Offset variieren). Es ist möglich bis zu drei Videoquellen anzuschließen, die sich per Software auswählen lassen. Beschaltet ist dieser gemäß des Datenblattes.

Aber der Reihe nach: Das Videosignal von der Scartbuchse oder einer der Cinchbuchsen gelangt nach der obligatorischen Leitungsanpassung und kapazitiver Koppelung an die Eingangspins des TDA. Einer der Pins wird mittels zweier Steuerleitungen ausgewählt und das jeweilige Signal gelangt an den Verstärker. Dort wird es gemäß der Ladungen zweier Kondensatoren (also der Spannung über diese) Verstärkt und mit einem Gleichspannungsoffset versehen. So und nur so wird gewährleistet, daß der A/D-Wandler immer richtig angesteuert wird (schwarz ist schwarz - Wert 64 und weiß ist weiß - Wert ca. 213). Die Ladungen auf den Kondensatoren werden von dem bereits digitalisierten Videosignal abgeleitet, dazu hat der TDA ein paar Komparatoren eingebaut. Das Prinzip ist einfach: Ist das Signal zu hell oder zu dunkel -> variiere die Ladung auf dem einen Kondensator, hat das Bild zuviel oder zuwenig Kontrast -> variiere die Ladung auf dem anderen Kondensator.

Nun passiert das Videosignal erst noch einen passiven Filter, um hochfrequente Anteile herauszufiltern damit das Abtasttheorem eingehalten wird (die Abtastfrequenz ist im übrigen 16MHz - warum, dazu später). Vor dem Filter wird das Signal abgegriffen, das für den Syncseparator (LM1881) bestimmt ist. Das digitalisierte Signal gelangt nun auf den Eingangsdatenbus.

## 1.7 synchronisation

### 1.3.1 Synchronisation

Ein endloser Datenstrom nützt wenig, wenn man nicht weiß, wann welche Daten anliegen, um nun alles auf die Bilderfolge zu synchronisieren werden dem analogen Videosignal weitere Daten entnommen. Der Syncseparator LM1881 entnimmt nun dem Videosignal die Synchronisationsimpulse, die in dem sgn. Bereich "schwärzer als Schwarz" (digital "0") vorhanden sind. Um den LM an seinem CVBS-Eingang zu versorgen wird das Videosignal stromverstärkt und mit dem spezifizierten Widerstand von 750hm einem Tiefpaß zugeführt. Ganz zufrieden bin ich mit dem LM1881 nicht, da er nicht in der Lage ist aus verrauschtem Signal die Synchronimpulse sauber abzutrennen (es gibt Vergleichstypen der Firma élantec - EL1881, EL1882, EL4581, die sollen besser sein). Der LM gibt folgende digitalen Signale ab:

CSYNC Composite Sync, die blanken Synchronimpulse (werden zusammen mit BURST vom TDA benötigt).

VSYNC Vertical Sync, markiert den Anfang eines Bildes (50Hz-Takt), setzt die Speicher auf den Anfang zurück.

ODD/EVEN Zeigt, ob es sich bei dem aktuellen Halbbild um ein gerades oder ungerades handelt, selektiert jeweils einen der beiden Speicher.

BURST Markiert die Position des Farbträger-Bursts in der hinteren Schwarzschiene der horizontalen Austastlücke (ist eigentlich nur das Signal eines durch die hintere Flanke von CSYNC getriggerten Monoflops). BURST gelangt



wie CSYNC an den TDA (invertiert, und BURST zusätzlich verzögert -> zwei einander folgende, sich nicht berührende High-Impulse). Die Signale werden benötigt, um die Verstärkerregelung zu steuern.

## 1.8 abtastfrequenz

Zur Abtastfrequenz

Bei der Digitalisierung von Videosignalen gibt es folgende Möglichkeiten zu Wahl des Abtasttaktes:

Auf die Zeilenfrequenz bezogener Takt: Pro Videozeile (beginnend mit der horizontalen Austastlücke) werden eine feste Zahl von Werten entnommen. Das sind z.B. bei 15625Hz Zeilenfrequenz und 13,5MHz Abtasttakt dann genau 864 Pixel. Damit das immer genau paßt, muß man den Taktgenerator auf das Videosignal synchronisieren -> PLL erforderlich.

Auf den Farbträger bezogener Takt: Es ist auch Sinnvoll mit dem vierfachen des Farbträgers von 4,43MHz abzutasten, da sich dann die Farbe sehr einfach demodulieren läßt (auch hierfür ist eine PLL erforderlich). Der Nachteil hierbei ist außerdem die schlechte Anpaßbarkeit auf verschiedene Fernsehnormen (so ist der Farbträger bei NTSC bei 3,58MHz).

Freilaufender Takt: Der Takt ist völlig unabhängig vom Videosignal. Trotzdem ist es ratsam, ein ganzes Teilverhältnis zur Zeile zu wählen, da das Raster dann annähernd orthogonal ist.

Ich wählte aus Aufwands- und Praktikabilitätsgründen letzteres mit einer Abtastfrequenz von 16MHz, es ergeben sich dann genau 1024 Abtastwerte pro Zeile (ein Videobild hat bei mir also genau 625KByte). Da dieser Takt nicht ganz genau mit dem Videosignal synchron ist, kann es passieren, daß das Bild etwas schräg auf dem Bildschirm erscheint.

## 1.9 digital

### 1.4 Der Digitalteil

Gesteuert wird die ganze Schaltung durch den Parallelport (es gibt also keine Schalter und Tasten zur Betätigung). Zunächst wird die SEL-Leitung dazu benutzt, um von der Amiga-Seite aus die Datenrichtung festzulegen (SEL="1": Amiga -> Framegrabber, SEL="0" umgekehrt). Empfangene Daten landen dabei immer in den Eingangsregistern (IC9, IC10), gesendet werden die Daten direkt von den RAMs. Es wird das Standard-Handshakeverfahren der Parallelschnittstelle verwendet, so werden die Daten mit dem /STROBE-Impuls übergeben bzw. angefordert. Der /ACK-Impuls wird direkt aus /STROBE gewonnen, indem er invertiert wird - die Schaltung arbeitet schnell genug, um die Daten zu senden/empfangen. Ich möchte hierbei darauf hinweisen, daß es unbedingt erforderlich ist, eine Leitungsanpassung für /STROBE zu schaffen und /ACK nicht direkt mit den RCK-Eingängen der RAMs zu verbinden. Normalerweise wird der Framegrabber mit einem Kabel am Port angeschlossen, dort gibt es dann Laufzeiten und Reflexionen der Signale, das führt dann zum mehrmaligen Triggern der RAMs und es gehen Daten verloren.

Was wird nun in den Registern gespeichert:

Bit	Name	Aufgabe
D0	ReadReset	setzt die Leseadresse der RAMs zurück
D1	ReadOdd/Even	wählt das RAM, aus dem gelesen werden soll
D2	I0	
D3	I1	legen den Videoeingangskanal des TDA8708A fest
D4	/Freeze	legt fest, ob neue digitalisierte Daten in die RAMs geschrieben werden sollen

Das Signal /Freeze wird durch ein weiteres FlipFlop geführt, welches von dem ODD/EVEN-Signal getriggert wird. Dies gewährleistet, daß das Bild nicht mittendrin eingefrohren wird.

Es bleibt noch zu erwähnen, daß es einen einfachen Resetgenerator (bestehend aus einer R-C-Kombination und einem Inverter) gibt, der die FFs und die RAMs nach dem Einschalten definiert zurücksetzt.

## 1.10 rams

### 1.4.1 Die RAMs

Bei den HM530281Rs handelt es sich um 331776-word x 8-bit Frame DRAMs. Es sind speziell für digitale Videoanwendungen konzipierte Chips. Sie haben zwei voneinander vollkommen unabhängige Datenports (einen zum schreiben, einen zum lesen), zwei interne Adreßzähler (es gibt also keinen Adreßbus) und einen integrierten Refreshcontroller. Man braucht sich also um nichts weiteres zu kümmern, als um die Daten, die rein und raus sollen, die entsprechenden Schreib- und Lesetakte und die Resetimpulse, die die internen Zähler zurücksetzen. Das klingt gut und funktioniert auch wunderbar.

Ich möchte aber doch auf den Nachteil dieser Chips hinweisen: sie sind extrem schwierig zu besorgen, bei den üblichen (Versand-)Händlern bekommt man sie nicht. Also verweise ich im allgemeinen auf die Hitachi-Website, wo man eine Liste der Distributoren erhält, die Hitachi führen, und im speziellen auf den einen Distributor, der sie mir verschafft hat (zu 40DM/Stück):

CED Ditronic GmbH  
 Julius-Hölder-Str.42, D-70597 Stuttgart  
 Postfach 700159, D-70571 Stuttgart  
 Tel.: +49 (0)711/720001-0, Fax.: +49 (0)711/7289780

Vertriebsniederlassung Hannover  
 - Herrn Aschenberg -  
 Karl Wichert Allee 66, D-30625 Hannover  
 Tel.: +49 (0)511/54269-18, Fax.: +49 (0)511/64269-20

Die Firma Neumüller Fenner Elektronik GmbH hat sich um die Chips bemüht (zu 17,60DM/Stück), es gelang ihr allerdings nicht sie zu besorgen, die Firma MSC Vertriebs GmbH räumte sich eine Lieferzeit von 7 Monaten(!) ein. Weitere (europäische) Distributoren findet man hier:  
<http://www.hitachi-eu.com/hel/ecg/dist.htm>

Bild: Der Framegrabber, geöffnet (die Änderungen sind mittlerweile in das Layout integriert)

## 1.11 software

### 2 Die Software

Bei der Beschreibung der Software möchte ich mich auf die Erklärung der Signalverarbeitenden Komponenten beschränken.

#### 2.1 Schwarzweiß - der einfache Teil

Es gibt zwei Anzeigemodi:

1. die rohe Ausgabe der Daten (sie werden genau wie sie von der Hardware kommen dargestellt - evtl. gedithert auf die vorhandenen Grauwerte),
2. die auf den gültigen Bildbereich beschränkte, auf 4:3 skalierte, schwarzwertgeklemmte, kontrastgeregelte und evtl. gefilterte Ausgabe.

Die rohe Ausgabe zeigt dabei auch die Informationen des Fernsehbildes, die sonst unsichtbar sind. Dies sind im einzelnen:

- die Synchronimpulse (der Bereich "schwärzer als Schwarz") inklusive der sgn. Vor- und Nachtrabanten der vertikalen Austastlücke
- die vordere und hintere Schwarzschiene der horizontalen Austastlücke - dieser Bereich ist jetzt nicht schwarz sondern dunkelgrau.
- der Farbträgerburst in der hinteren Schwarzschiene
- Zusatzsignale in der vertikalen Austastlücke (Videotext, VPS und ein paar Meßsignale)

Bild: Die Austastlücke

## 1.12 bildverarbeitung

### 2.1.1 Die Bildverarbeitungsschritte im einzelnen

Zunächst besteht das Bild aus zwei Halbbildern, die nacheinander eingelesen und auch nacheinander verarbeitet werden:

**Zentrierung:** oder auch Synchronisation; der Bildausschnitt wird richtig positioniert, die 896x575 Pixel des Bildes werden aus den 1024x625 Pixeln ausgeschnitten.

**Filterung:** Handelt es sich um ein Farbbild, kann man mittels digitaler Filter den Farbträger entfernen (siehe dazu das nächste Kapitel).

**Skalierung:** Das Bild wird auf 4:3 (768x575) entzerrt.

**Schwarzwertklemmung:** Der hinteren Schwarzschiene wird der Pegel für Schwarz entnommen (in der Regel 64), dieser wird von den sichtbaren Pixeln abgezogen.

**Kontrastregelung:** Der Schwarzwert entspricht 300mV (bezogen auf die

Synchronsignale) des Videosignals. Weiß entspricht dann genau 1V, der Digitalwert für Weiß wird aus dem Schwarzwert und diesem Verhältnis gebildet (in der Regel 213). Die Pixel werden dementsprechend auf den Bereich von 0 bis 255 umgerechnet.

Rauschfilterung: Optional wird ein verrauschtes Bild mittels eines IIR-Tiefpasses geglättet. Damit das Bild dabei nicht zu stark verwischt, wird die Filterung bei Erkennung von Konturen ausgesetzt. (Diese Vorgehensweise hat sich als effektiver und einfacher herausgestellt, als ein Median-Filter.)

## 1.13 filter

### 2.2 Digitale Filterung

Es gibt zwei Möglichkeiten, den Farbträger (4433643,75Hz bei PAL – das entspricht einer Periode von ~3,6 Pixeln) aus dem Signal herauszufiltern:

1. Eindimensional – der übliche Weg bei einfachen Fernsehgeräten
2. Zweidimensional – dieser Weg ist nur mittels digitaler Signalverarbeitung mit vernünftigen Aufwand zu realisieren

Eindimensionale Filterung im digitalen Videobereich wird in der Regel mittels der FIR – (finite impulse response) – Filter realisiert. Dabei werden in einer Zeile mehrere benachbarte Pixel mit einem Faktor versehen und addiert. Die Anzahl der zugrundegelegten Pixel bestimmt die Ordnung des Filters.

Beispiel: Einen Tiefpaß realisiert man dabei mit den für alle  $n$  Pixel gleichen Faktor  $1/n$  (das entspricht der Durchschnittsbildung)

Zweidimensionale Filter werden auch Kammfilter genannt. Der Name Kammfilter kommt von dem kammförmigen Spektrum, das genau dem Spektrum des Farbträgers angepaßt ist. Ein zweidimensionaler Filter berücksichtigt auch die Pixel der benachbarten Zeilen (so ist der Farbträger bei PAL gegenüber der vorletzten und der übernächsten Zeile genau um  $180^\circ$  verschoben). Da die Zeilen dafür gespeichert werden müssen, erklärt sich damit die Notwendigkeit der digitalen Signalverarbeitung. Treten nun Farb- oder Helligkeitssprünge von Zeile zu Zeile auf, so werden diese jetzt verwischt, dieses zu erkennen und auszugleichen ist die Eigenschaft der sgn. adaptiven Kammfilter, welche von drei Zeilen die nicht berücksichtigt, die vom Mittelwert allzu stark abweicht.

Bild: Das Testbild, zweidimensional gefiltert

Das Testbild (die ".video"-Datei dafür wurde mittels des Programms "ZonePlateStream(.e)" erzeugt) enthält neben den obligatorischen Grauwert- und Farbbalken die sgn. Zonenplatte. Die Zonenplatte hat nun die Eigenschaft, alle in einem Videosignal möglichen Frequenzen zu enthalten, die derart angeordnet sind, daß die 2d-Fouriertransformierte wieder eine Zonenplatte ergibt. Die Abbildung zeigt, welche Frequenzen der Kammfilter entfernt. Der Vorteil ist hierbei, daß senkrechte hochfrequente Strukturen erhalten bleiben, das Bild wird schärfer.

Es bleibt noch zu erwähnen, daß das Schwarzweißbild (die Luminanz) sich aus der Differenz von dem rohen Videosignal und dem herausgefilterten Farbartsignal (der Chrominanz) ergibt.

## 1.14 farbe

### 2.3 Farbe - der schwierige Teil

Um es vorwegzunehmen, die Qualität der Farbberechnung hängt zu einem Großteil von der Qualität der Filter ab. Ist der Filter nicht perfekt, werden Luminaz und Chrominaz nicht sauber voneinander getrennt und es kommt zu Übersprecheffekten. Abgesehen davon gliedert sich die Farbddekodierung wie folgt:

**Rückgewinnung der Farbträgerphase:** Das Modulationsverfahren ist die Quadraturamplitudenmodulation, das Signal enthält zwei Komponenten (diese sind bei PAL U und V), die um  $90^\circ$  versetzt sind. Zur Demodulation ist die Kenntnis über die Phase nötig. Dazu wird in der horizontalen Austastlücke eine Referenz gesendet (der Farbträgerburst - ca. 10 Schwingungen mit  $135^\circ$ ), mit der sich der (digitale) Oszillator synchronisieren läßt. Praktisch wird dazu der Burst normal demoduliert (s.u.) und dann der Phasenfehler ausgeglichen.

**Synchrondemodulation:** Die Farbkomponenten U und V werden aus der Chrominaz C errechnet:

$$U = s \cdot C \cdot \cos \beta$$

$$V = s \cdot C \cdot (+/-) \sin \beta$$

s ist dabei der Faktor, der die Farbsättigung angibt. Er errechnet sich aus der Amplitude des Farbträgerbursts, dieser ist zu 75% gesättigt und hat in der Regel eine Amplitude von 20% des Weißwerts (gegebüber Schwarz), also etwa 140mV. Daß die Phase von V von Zeile zu Zeile umgekehrt wird, ist PAL-Eigenschaft und wird unten erläutert. beta ist nun der Wert des digitalen Oszillator-Inkrementes. Praktisch realisiert wurde dieses mit einer Sinustabelle mit 4096 Einträgen. Das Inkrement ist bei einer Abtastfrequenz von 16MHz dann genau  $1135^\circ$ .

**Tiefpaßfilterung der Farbkomponenten:** U und V sind jetzt noch mit dem  $\sin^2$ -förmigen Farbträger versehen, dieser wird mittels eines (FIR-) Tiefpaßfilters entfernt.

**PAL-Kompensation:** Um mögliche (prinzipbedingte) Phasenfehler (und somit Farbverfälschungen) ausgleichen zu können, ist die Phase der V-Komponente Zeilenweise um  $180^\circ$  verschoben. Hebt man diese Phasenverschiebung beim Dekodieren auf (invertieren jeder zweiten Zeile) und mittelt die U- und V-Komponenten Zeilenweise, so kompensieren sich gleiche Phasenfehler in den Zeilen. Da dadurch die Farbauflösung in der Vertikalen halbiert wird, kann man einen Schwellwert setzen, über dem dieser Mechanismus abgeschaltet wird.

**Farbflankenversteilerung (Optional):** Da das Farbsignal prinzipbedingt eine geringere Bandbreite hat, verwischen starke Farbkontraste. Diese können erkannt und verstärkt werden, indem benachbarte Pixel verschoben werden. Wählt man den Faktor der Verschiebung zu groß, werden allerdings auch Farbflanken verstärkt, die von vornherein nicht steil waren (z.B. in unscharfen Bildbereichen).



Dematrizierung: Aus der Luminanz Y und den Komponenten U, V errechnen sich R,G,B wie folgt:

$$\begin{array}{rcl} |R| & | & 1 \quad 0 \quad 1.14 \quad | \quad |Y| \\ |G| = & | & 1 \quad -0.37 \quad -0.57 \quad | \quad \cdot \quad |U| \\ |B| & | & 1 \quad 2.04 \quad 0 \quad | \quad |V| \end{array}$$

## 1.15 bedienung

### 3 Die Bedienungsanleitung

Nach dem Start des Programms "FrameGrab" wird die Hardware eingeschaltet. Reagiert diese nicht, wird das erkannt, es können immer aber noch Dateien geladen werden. Läuft bereits ein "FrameGrab", so stehen dem zweiten Programm weder die Hardware noch der ARexx-Port zur Verfügung. Das Programm kann von der Workbench oder aus der Kommandozeile gestartet werden:

"FrameGrab FILE,ICONIFIED/S,NOMESSAGES/S,INTERRUPT/S,PUBSCREEN/K"

FILE:           Dateiname einer zu ladenden ".video"-Datei  
 ICONIFIED:     Startet das Programm als AppIcon  
 NOMESSAGES:   Unterbindet einen Großteil der Warnrequester  
                   Die beiden letzten Parameter erleichtern den Start aus einem  
                   ARexx-Script heraus.  
 INTERRUPT:    Es wird zur Paralelportansteuerung eine Interruptroutine  
                   installiert. Diese ist langsamer, aber evtl. sicherer.  
 PUBSCREEN:    Gibt den Namen des Public-Screens an, auf dem das Benutzerinterface  
                   geöffnet werden soll (normalerweise der Default-PubScreen).

Das Programm präsentiert sich mit der Gadtools-Basierten Oberfläche EasyGUI (von Wouter van Oortmerssen und Jason R. Hulance):

Bild: Das FrameGrab Hauptfenster

Im Hauptfenster befinden sich der Preview-Bereich, der das laufende Videobild mit verminderter Qualität anzeigt (ein Refresh dauert auf meinem A4000LC040 immerhin ca. 6 Sekunden) sowie ein paar Action-Buttons. Diese sind fast alle "sticky" d.h. sie bleiben nach Anwahl eingedrückt und zeigen somit auch den Status des Programms an. Die Funktionen der Buttons im einzelnen:

Oeffnen(o) öffnet (oder schließt) eine Videodatei zur Anzeige und Bildverarbeitung. Eine geöffnete Datei verhält sich genau so, wie ein Bild aus dem Framegrabber (abgesehen von der Synchronisation).

Speichern(s) speichert die aktuellen Daten (vom Grabber oder aus einer Datei). Es stehen drei Möglichkeiten zur Auswahl:

1. Der rohe Videodatenstrom: das Videobild wird 1:1 in eine Datei geschrieben (625KByte), die später mit "Oeffnen" wieder eingeladen werden kann.
2. Render-Screen: Der Render-Screen wird genau so, wie er ist in eine IFF-ILBM-Datei geschrieben.
3. Echtfarbig: Es werden sämtliche (ausgewählten) Bildverarbeitungsschritte durchgeführt und die 24-Bit-Daten als IFF-ILBM24-Datei gespeichert. Um ein berechnetes Halbbild dabei zwischenspeichern, wird etwas RAM angefordert (max. 961536 Bytes bei Farbe mit Synchronimpulsen).

Optionen(p): Das Einstellungsfenster wird geöffnet (oder geschlossen).

Halt(f): Ist die Hardware aktiv, so kann man das laufende Videobild einfrieren (wird beim Speichern oder Rendern automatisch aktiv).

Rendern(r) startet alle (ausgewählten) Bildverarbeitungsschritte und stellt das Bild auf dem bestmöglichen Screenmode dar.

Ende(q): No Comment.

Es gibt dann noch ein kleines Menü:

Über: Ein kleiner Informationsrequester öffnet sich.

Ikonifizieren: Das Fenster wird geschlossen und durch ein AppIcon ersetzt. Man kann ".video"-Dateien laden, indem man deren Icons auf das Icon (oder auch auf das Hauptfenster) fallenläßt (diese Vorgehensweise dürfte aber wohl bekannt sein).

Ende: s.o.

## 1.16 einstellungen

### 3.1 Die Einstellungen

Das Einstellungsfenster gliedert sich in sechs Untergruppen, die untere Buttonreihe ist dabei immer präsent:

Speichern: Die Einstellungen werden dauerhaft in "ENVARC:GrabPrefs" gespeichert, übernommen und das Einstellungsfenster geschlossen.

Benutzen: Die Einstellungen werden temporär in "ENV:GrabPrefs" gespeichert, übernommen und das Fenster geschlossen.

Zurück: Alle Änderungen seit dem Öffnen des Fensters werden rückgängig gemacht.

Abbruch: Wie Zurück, das Fenster wird nun aber geschlossen (das Close-Gadget hat die gleiche Wirkung).

Bild: Das Allgemein-Einstellungsfenster

Wähle Grafikmodus: Ein Screenmoderequester wird geöffnet, der Bildschirmmodus für den Renderscreen wird ausgewählt und angezeigt.

Benutze HAM: Öffne einen HAM-Screen zum rendern, falls erforderlich.

Benutze FS-Dither: Benutze zum Rendern das Floyd-Steinberg Dithering.

Fenster befestigen: Speichert die augenblickliche Position der Fenster als Standard.

Bild: Das Eingabe-Einstellungsfenster

---

Eingabekanal: Wählt einen der drei Video-Eingabeports (Scart, Cinch1 oder Cinch2).

Dateien Puffern: Dateien werden im RAM zwischengespeichert, das erhöht die Geschwindigkeit, benötigt aber weitere 625KByte RAM.

Synchronisation per Software: Es wird versucht, nicht vollständig übereinanderliegende Halbbilder (verursacht durch verrauschte Videosignale) Zeilenweise zu synchronisieren.

Bild: Das Ausgabe-Einstellungsfenster

Speichern als legt den Speichermodus fest (Video-Frame, Render-Screen, True-Color oder Fragen).

Dateiendungen aktiviert das automatische Anhängen der passenden Endungen (".video", ".ilbm", ".ilbm24") an die Dateinamen (kann im Speichen-Requester immer noch geändert werden).

Dateien nummerieren dto. für die Nummerierung der Dateien.

Bild: Das Vorschau-Einstellungsfenster

Floyd-Steinberg-Dither für das Vorschaufenster,

Glätten für das Vorschaufenster, aktiviert die Tiefpaßfilterung,

Halbbild wählt das gerade oder das ungerade Halbbild für das Vorschaufenster.

Bild: Das Luminanz-Einstellungsfenster

zeige Synchronimpulse aktiviert die Anzeige und das Speichern auch der Synchronimpulse und der Austastlücken samt Zusatzsignale. Deaktiviert den Farbträgerfilter im Schwarzweißmodus.

Farbträger filtern schaltet die Farbträgerfalle für die Luminanz an.

Kammfilter aktiviert den Kammfilter (0 ist aus), der Schwellwert für die Adaptivität läßt sich einstellen (ein guter Wert ist 4).

Rauschunterdrückung erlaubt die Aktivierung und Einstellung der Intensität der Rauschunterdrückung (IIR-Tiefpaß). Wirkt auch auf die Chrominanz.

Bildsignal legt den Pegel des Videosignals fest (hängt vom Sender ab - normal sind 700mV gegenüber Schwarz), wirkt wie ein Kontrastregler.

Berechne ermittelt aus den Bilddaten den Maximalwert und stellt den Bildsignalpegel entsprechend ein.

Bild: Das Chrominanz-Einstellungsfenster

Farbdekoder wählt den Farbdekoder. Gegenwärtig nur "keiner" (Schwarz-Weiß) und PAL ("Vierquadrantenmultiplizierer" wie in Kap. 2.3).

PAL-Kompensation aktiviert die zeilenweise Mittelung der Chrominanz und legt den Schwellwert dafür fest (gute Schwellwerte sind zwischen 4 und 6).



Chroma-Pegel legt die bekannte Farbsättigung fest (gute Werte sind zwischen 40 und 50).

Farbflankenversteilerung aktiviert schließlich dieselbe. Werte größer als zwei sind mit Vorsicht zu verwenden, da auch unscharf gewünschte Farbflanken verstärkt werden.

## 1.17 Der ARexx-Port

### 3.2 Der ARexx-Port

Beim Programmstart wird ein ARexx-Port namens "FGRAB" eingerichtet. Folgende Kommandos werden unterstützt (in [] optional, in <> gefordert):

FREEZE Schaltet zwischen laufendem Videobild und Standbild um. Hat nur bei aktivierter Hardware (keine Datei geladen) Wirkung.

RENDER Das aktuelle Videobild wird gezeichnet.

QUIT ...

ICON (De-)Ikonifiziert.

SAVE [file] Speichert das Bild im aktuellen Modus. Ein Neuer Dateiname kann übergeben werden.

OPEN [file] Öffnet die angegebene (oder letztgenannte) ".video"-Datei.

CLOSE Die (geöffnete) Datei wird geschlossen und wieder auf den Parallelport umgeschaltet.

CALCPICLEVEL Berechnet den Bildsignalpegel neu (entspricht genau dem Button in den Luminanz-Einstellungen).

SET <var> <value> Setzt die angegebene Variable auf den Wert.

QUERY <var> Liest die angegebene Variable aus und übergibt den Wert der 'RESULT'-Variable von ARexx.

#### 3.2.1 Die Variablen

Variable	Werte	Bemerkung
MODEID	integer	der rohe Grafikmodus
HAM	TRUE FALSE	
FLOYD	TRUE FALSE	
INPCHANNEL	SCART CINCH1 CINCH2	
FILEBUF	TRUE FALSE	
SOFTSYNC	TRUE FALSE	
SAVEAS	ASK VIDEO SCREEN TRUECOLOR	
FILEEXT	TRUE FALSE	
FILENUM	TRUE FALSE	
PREVIEWFS	TRUE FALSE	
PREVIEWSM	TRUE FALSE	

PREVIEWFR	EVEN ODD	
SYNCS	TRUE FALSE	
CHRFILTER	TRUE FALSE	
COMBFILTER	integer[0-10]	
NOISERED	integer[0-4]	
PICLEVEL	integer[50-90]	entspr. 500mV bis 900mV
CHROMADEC	NONE PAL	
PALCOMP	integer[0-10]	
CHROMALEVEL	integer[0-100]	
COLEDGEENH	integer[0-4]	
BUSY	TRUE FALSE	READONLY Preview wird gezeichnet
FROZEN	TRUE FALSE	READONLY Bild eingefrohren
ICONIFIED	TRUE FALSE	READONLY
DISPMESSAGES	TRUE FALSE	Warnings ausgeben
FILENAME	string	

Ein Beispielscript

Es eine Bildsequenz gespeichert.

## 1.18 Was ist noch zu tun?

### 3.3 Was ist noch zu tun?

Dies ist die nicht vollständige Liste der noch nicht verwirklichten Möglichkeiten der Software:

- Fehler beseitigen
- Farbfilter verbessern (evtl. im Frequenzbereich?)
- HAM-Modus optimieren
- Synchronisation verbessern
- Graduations-Entzerrung (Gamma-Regler)
- Meßtechnik: Oszilo- und Vektorskop, Spectrum-Analyser
- Auslesen der Zusatzinformation in der Austastlücke (Videotext, VPS, Zeile 23)
- PALplus (inklusive 16:9-Screen)
- NTSC, SECAM, MAC, ...
- Descrambler
- Geschwindigkeitsoptimierungen
- ...

## 1.19 fehler

### 3.4 Fehler

Fehlerberichte erreichen mich am besten per EMail: [h.boehme@tu-bs.de](mailto:h.boehme@tu-bs.de)

Wenn ich weitere Videonormen (NTCS oder SECAM) in die Software integrieren soll, schickt mir gepackte ".video"-Files. Wenn ich Grafikkarten unterstützten soll, schickt mir eine Grafikkarte (oder zumindest deren Dokumentation).

#### 3.4.1 Bekannte Fehler

- Die SEL-Leitung ist Amigaseitig mit RI der seriellen Schnittstelle

verbunden. Das kann zu Problemen mit Modemverbindungen führen -> Kurzfristige Abhilfe: Den Framegrabber abklemmen, wenn man das Modem benutzen will, langfristig ist die Hardware zu modifizieren.

## 1.20 literatur

### Bibliothek

- Fernsehtechnik und Bildübertragung (Teil I & II), Prof. Dr.-Ing. U. Reimers, Vorlesungsscripte, Institut für Nachrichtentechnik, Technische Universität Braunschweig
- Digitale Signalverarbeitung in Fernsehgeräten, Paschko, Verlag Technik
- Von PAL zu PALplus, Dobler, Verlag Technik
- Digitale Filter in der Videotechnik, Schönfelder (Hrsg.), Drei R
- Computer-Sehen, Prof. Dr. Friedrich M. Wahl, Vorlesungsscript, Institut für Robotik und Prozeßinformatik, Technische Universität Braunschweig
- Digitale Bildsignalverarbeitung, Prof. Dr. Friedrich M. Wahl, Springer
- Datenblatt TDA8708A, Phillips IC02,  
<http://www.semiconductors.philips.com/acrobat/2031.pdf>
- Datenblatt HM530281R, Hitachi,  
[http://www.hitachi-eu.com/hel/ecg/pdf/01\\_063.pdf](http://www.hitachi-eu.com/hel/ecg/pdf/01_063.pdf)
- Datenblatt LM1881, National  
<http://www.national.com/search/search.cgi/design?keywords=lm1881>
- High-Speed CMOS Data, Motorola

## 1.21 dank

Dank geht an:

AmigaE & EasyGUI: Wouter van Oortmerssen & Jason R. Hulance

Betatester: Björn Haake, Lars Unger

## 1.22 versionen

### Versionen

#### Hardware

#### Revision 1.0

#### Erster Prototyp

#### Revision 1.1

BUG Videosignal des Filterausgangs darf nicht direkt mit SCART verbunden werden -> Verstärker und Leitungsanpassung erforderlich!

BUG Böse Leitungseffekte, wenn die Schaltung mit einem Kabel vom nur 1,5m mit dem Amiga verbunden ist -> Leitungsabschluß an Strobe (mit Tiefpaß).

Außerdem dürfen die Clock-Eingänge nicht direkt mit Acknowledge verbunden werden (Rücklaufende Welle läßt nochmal Triggern!) -> Zweiter Inverter erzeugt das Signal Separat!

BUG Der LM1881-Syncseparator hat große Probleme mit verrauschtem Videosignal, er arbeitet auch sonst etwas unregelmäßig. -> Tiefpaß einfügen!

BUG Zu Dumm - Der TDA8708 verlangt als Eingabe für GateA und GateB Active High Signale, CSync und Burst vom LM1881 sind aber Active Low. Folge: Der TDA ist die ganze Zeit im Controlmode 1 und die digitalisierten Bilder sind in der Regel etwas übersteuert! -> Inverter einfügen! Außerdem muß GateB gegenüber GateA verzögert werden (um sagen wir mal eine Mikrosekunde), dann (und nur dann) geht's.

NEU PNP-Transistorschaltung mit BUSY verbunden -> Softwaremäßiges Ein- und Ausschalten ist jetzt möglich.

Software

Version 1.0

Erste Veröffentlichung

Version 1.1

NEU Lokalisation

NEU AppWindow und AppIcon

NEU Kommandozeile und Menü

NEU ARexx-Port

Version 1.2

VERBESSERT Parallelport: Die Interruptroutine wurde (optional) ersetzt durch eine Burstroutine. -> Preview ist jetzt doppelt so schnell.

NEU Rauschfilter

NEU Farbflankenversteilerung

BUG Glätten der Vorschau war fehlerhaft

VERBESSERT Kammfilter: kommt jetzt mit vertikalen Farbkontrasten besser zurecht.