

Videotimeinserter



Sven Andersson

24. NOVEMBER 2008

Inhaltsverzeichnis

1	GPS-Videotimeinserter (VTI)	3
1.1	Verwendungszweck	3
1.2	Allgemeines	3
1.3	Funktion	4
2	Zeitbestimmung	4
2.1	Genauigkeit	6
2.2	Zeitsynchronisation des PCs	6
3	Software	7
3.1	Programmupdate	7
3.2	Erweiterungen	8
4	Hardware	9
4.1	Anschlußbelegung	9
4.2	Hardwarefunktion	9
5	Aufbau	10
5.1	Schaltung	12
5.2	Daten- und Programmieradapter	13
5.3	Bauteilliste	13
	Abbildungsverzeichnis	15
	Tabellenverzeichnis	15

1 GPS-Videotimeinserter (VTI)

1.1 Verwendungszweck

Der Videotimeinserter dient der Einblendung der genauen Zeit(UTC) in ein Videobild. **Es ist ein freies Projekt!** Eine kommerzielle Nutzung ist nicht erlaubt!!! Dieser VTI kann von jedem aufgebaut und nach eigenen Wünschen modifiziert werden. Modifikationen an Hard- und Software müssen wieder veröffentlicht werden! Somit kommen Erweiterungen und Verbesserungen allen Nutzern zu Gute.

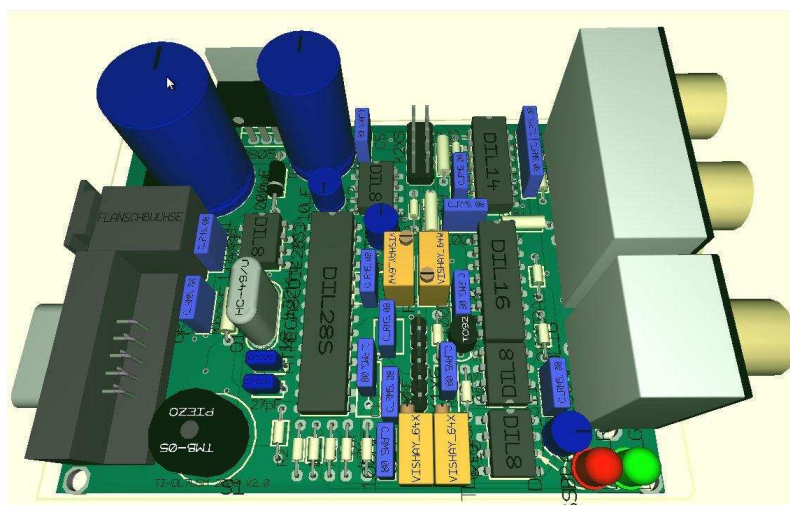


Abbildung 1: Ansicht der Platine

1.2 Allgemeines

Das Grundkonzept der Hard- und Software stammt von Gari Dion (N4TXI)[1] Das hier vorgestellte Konzept lässt sich, ohne Softwareänderungen, entsprechend abrüsten. Die meisten Hard- und Softwareerweiterungen erfolgten auf Anregung von Wolfgang Rothe und Eckehard Rothenberg - Vielen Dank dafür! Ein Dankeschön geht auch an Niko Wünsche für die mechanischen Arbeiten. Die Kosten betragen für die Bauelemente ca. €35,- und für die Platine ca. €13,-. Ein Warenkorb bei Reichelt Elektronik[7] ist verfügbar. Die Stromversorgungsbuchse und den MAX4278EPA+ bekommt man z.B. bei Segor Electronics[8]. Der in der Version 1 verwendete SSM2165 ist inzwischen vom Hersteller abgekündigt und nicht mehr lieferbar. Ich habe die Schaltung des Mikrofonverstärkers

geändert, diese arbeitet jetzt mit einem normalen Operationsverstärker, allerdings nun ohne Dynamikkompressor.

1.3 Funktion

Der VTI wird zwischen Kamera und Videorecorder geschaltet. Er wird mit 12V (9-20V möglich) betrieben. Ein Verpolschutz ist eingebaut. An den VTI wird ein GPS-Empfänger Garmin GPS18LVC angeschlossen. Bei diesem GPS-Empfänger ist garantiert, daß mit der Flanke des PPS-Signals¹ die aktuelle Zeit auf der seriellen Schnittstelle des Empfängers ausgegeben wird. Es wird die UTC in das Bild eingeblendet! Bei einer CCD-Videocamera ist das im Video sichtbare Halbbild aber 20ms vorher aufgenommen - siehe Zeitbestimmung!!! Optional kann ein PC-Mikrofon, zum Aufsprechen von Kommentaren angeschlossen werden. Da beim Videorecorder zwei Tonspuren vorhanden sind, kann auf die zweite Tonspur ein Ton im Takte des PPS-Signals aufgezeichnet werden. Eine grüne Leuchtdiode(LED) blinkt im Sekundentakt des PPS-Signales. Wenn kein PPS anliegt, leuchtet die rote LED 1s, bzw. ist 1s aus. Dieses Blinken der roten Leuchtdiode zeigt das Laufen der Notuhr an. Liegt PPS an leuchtet die rote LED ständig, oder ist ständig aus. Ein sporadisches an/aus der roten LED kommt vor, ist aber ohne Bedeutung. Folgende Zeilen werden ins Videosignal eingeblendet:

- Titelzeile - Kann ein- und ausgeschaltet werden. Hier kann z.B. der Sternwartenname stehen.
- Standortanzeige (WGS84) - Kann ein- und ausgeschaltet werden.
- Zeitzeile - Die Schriftgröße kann umgeschaltet und die vertikale Position stufenlos eingestellt werden.

2 Zeitbestimmung

Ein komplettes Videobild besteht aus zwei Halbbildern. Das erste Halbbild enthält die ungeraden (1,3,5,...) Zeilen, das zweite Halbbild die geraden (2,4,6,...) Zeilen. Ein Halbbild dauert 20ms. Bei CCD-Kameras wird immer ein komplettes Halbbild belichtet und anschließend auf Video ausgegeben. Wir sehen im Video also ein Halbbild, welches 20ms früher belichtet wurde! In jedes Halbbild wird nun die genaue Zeit (UTC) eingeblendet. Die Aufnahmezeit des Halbbildes ergibt sich in diesem Fall aus der eingeblendeten Zeit minus 20ms. Die Belichtungszeit beträgt nicht ganz 20ms, da die Kamera Zeit für das

¹Puls pro Sekunde

Löschen des CCD-Chips benötigt. Das Löschen erfolgt nach dem Bildsynchronimpuls[4]. Wenn mehrere Halbbilder in der Kamera (Mintron/Watec) aufaddiert werden, ist der Belichtungsbeginn des Bildes zu errechnen! Werden n Halbbilder addiert, ist der Belichtungsbeginn $n \cdot 20\text{ms}$ früher! Die Anzahl der aufaddierten Bilder kann man gut durch betrachten der einzelnen Halbbilder bestimmen - alle n -Bilder ändert sich das Rauschen im Bild. Wir benötigen eine genaue Zeitbasis. In der Vergangenheit stand uns nur der Zeitzeichensender DCF77 zur Verfügung. Hier haben wir folgende Probleme:

- Laufzeit des Signals von Frankfurt/M zum Empfänger (Berlin) - ca. 300km ca. 1ms
- Laufzeit des Signals im Empfänger - nicht konstant u. Empfängerabhängig ca.10-80ms !
- nicht weltweit verfügbar - abhängig von Ausbreitung und Antennenaufwand

Heute stehen uns GPS-Empfänger mit einem PPS-Signal (Puls Per Second) zur Verfügung. Die beim DCF77-Zeitsignal genannten Nachteile sind hier nicht vorhanden. Das PPS-Signal hat beim GPS18LVC eine Genauigkeit von $1\mu\text{s}$, also sehr viel besser als es die Videozeitauflösung hergibt!

2.1 Genauigkeit

Die Genauigkeit der Zeiteinblendung ist von der zeitlichen Präzision des PPS-Signals und der Verarbeitungszeit im Mikrocontroller abhängig. Der GPS-Empfänger GPS18LVC liefert das PPS-Signal mit einer Genauigkeit von $\pm 1\mu\text{s}$. Der Mikrocontroller benötigt für die Verarbeitung eines Befehls 1 bis 4 Takte (bei 18,432MHz also von 54,2ns bis 217ns). Vom PPS-Impuls bis zum Setzen des Millisekundentimers vergehen $\approx 975\text{ns}$. Diese Verarbeitungszeit ist immer eine Verzögerung. Wenn der PPS-Impuls während einer Interruptroutine aktiv wird, ist eine zusätzliche Verzögerung möglich. Diese Verzögerung beträgt maximal $\approx 60\mu\text{s}$.

Somit ist: $t_0 = \pm t_{\text{gps}} + t_{\text{soft}} + t_{\text{inter}} = \pm 1\mu\text{s} + 0,975\mu\text{s} + 0 \dots 60\mu\text{s} = -0,25\mu\text{s} \dots 62\mu\text{s}$
Ein Halbbild dauert allerdings 20ms, somit kann diese Verzögerung vernachlässigt werden.

2.2 Zeitsynchronisation des PCs

Den GPS-Empfänger kann man auch zur Zeitsynchronisation des PCs benutzen. Hierzu wird der PC über das Daten- und Programmieradapter mit dem VTI verbunden. Der VTI dient dann nur zur Stromversorgung des GPS18LVC. Der VTI kann aber auch zeitgleich

verwendet werden. Auf dem PC muß ein NTP-Client laufen. Bei Linux ist dieser standardmäßig im System vorhanden. Für Windows ist ein fertig kompilierter NTP-Client kostenlos erhältlich[9]. Als Treiber ist der Generic NMEA GPS Receiver (127.127.20.x) einzustellen. Das x steht für die verwendete COM-Schnittstelle (0-4). Auf jeden Fall die Dokumentation studieren! Der GPS18LVC ist auf eine Baudrate von 4800Baud einzustellen, bei gleichzeitiger Nutzung des VTI ist dieser auch auf 4800 Baud zu programmieren².

3 Software

Die Software des Mikrocontrollers ist in der Programmiersprache C geschrieben. Als Programmieroberfläche wurde AVR-Studio[5] und Winavr[6] verwendet. Das Programm des ATMega8-Mikrocontrollers besteht aus der Hauptschleife und mehreren Interruptroutinen.

- Hauptschleife - Einlesen der NMEA-Daten des GPS-Empfängers
- Interrupt0 - durch Zeilensynchronimpuls ausgelöst, Einblendung der Zeichen in das Videobild [1],
- Interrupt1 - PPS, setzt den Timer auf 0, schaltet ■ ein, Zeit um 1s erhöhen
- TimerCaptureInterrupt - durch Bildsynchronimpuls ausgelöst, holt den Timerinhalt (=ms), setzt Zeilenzähler auf 0, Zeit fixieren, ■ ausschalten, E/O setzen
- TimerOverflowInterrupt - wenn kein PPS anliegt, läuft der Timer über, Zeit um 1s erhöhen

3.1 Programmupdate

Ein Update der Timeinsertersoftware ist ohne Programmiergerät möglich. Lediglich der Programmieradapter (Seite 13) ist nötig. Dieses wird statt dem GPS-Empfänger an den Timeinserter gesteckt. Wenn der Timeinserter dann eingeschaltet wird, startet der sogenannte Bootloader. Nun kann über ein serielles (RS232) Kabel und einer Programmiersoftware eine neue Software geladen werden. Als Software ist AVR-Dude[2] und natürlich auch das AVR-Studio[5] geeignet.

²GPS18LVC und VTI arbeiten sonst mit 38400 Baud

3.2 Erweiterungen

- 2006 12 31 - Hardwareerweiterung - Schwarzwertklemmung
- 2007 02 11 - Softwareupdate - sporadisches springen der eingeblendeten Zeilen beseitigt
- 2007 02 23 - Softwareupdate - PPS-Anzeige von P auf ■ umgestellt
- 2007 02 24 - Bootloader funktioniert, Hardwareerweiterung folgt. Somit kann jeder ,der kein Programmiergerät hat, über die serielle Schnittstelle updaten.
- 2007 03 02 - Hardwareerweiterung - 2 Drähte werden vom ATmega8 zur SUB-D-Buchse gelötet, dann kann der Bootloader genutzt werden, nicht ganz Schnittstellenkonform, spart aber einen Schaltkreis...
- 2007 03 18 - Software erweitert - kann jetzt auch für ATmega88 und 168 kompiliert werden
- 2007 08 01 - Softwareupdate - nach dem Update vom 11.02.07 kam es vor, das in 2 Halbbildern die gleichen ms angezeigt wurden. Software wieder auf den Stand vor dem 11.02.07 gebracht. Ergab wieder das Zeilenspringen. Mit dem Winavr-Update (GCC4.12) ist der Fehler verschwunden.
- 2007 08 02 - Softwareupdate - Videoausgaberoutine überarbeitet
- 2007 12 03 - Softwareupdate - Zeichensatz überarbeitet, Doppelpunkt wurde als Hochpunkt angezeigt

4 Hardware

4.1 Anschlußbelegung

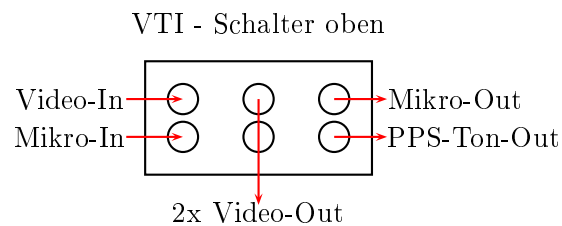


Abbildung 3: Belegung der Buchsen

4.2 Hardwarefunktion

Kernstück des VTI ist ein ATMega8-Mikrocontroller. Ein ATMega168-Mikrocontroller funktioniert ebenfalls und hat den Vorteil bis 20MHz spezifiziert zu sein. Der ATMega8 wird in dieser Schaltung übertaktet (spezifiziert bis 16MHz, betrieben mit 18,432MHz). Solange der EEPROM-Speicher nicht genutzt wird ist es aber unkritisch. Aus dem Videosignal separiert ein LM1881 den Vertikal- und Horizontalsynchronimpuls und das E/O-Signal, welches anzeigt ob es sich bei dem Halbbild um ein Even- oder Odd-Field handelt. Diese Signale liegen am ATMega8 an. Das PPS-Signal und die seriellen NMEA-Daten vom GPS18LVC liegen ebenfalls direkt am Mikrocontroller an. Die SPI-Schnittstelle des ATMega8 gibt das Videosignal aus.

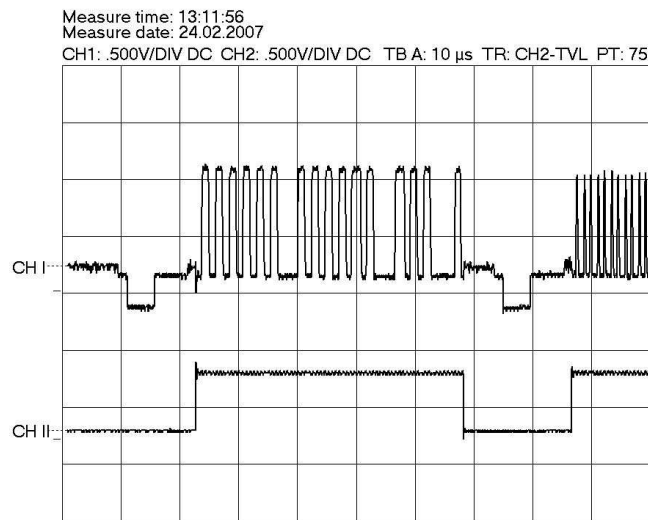


Abbildung 4: Videozeile mit Zeicheneinblendung

Dazu wird der Multiplexer (74HCT4053) umgeschaltet. Dann ist das Eingangsvideosignal weggeschaltet und der Hintergrund dunkel. Nach dem Multiplexer folgt noch ein Videoverstärker (MAX4278), welcher die beiden Videoausgänge treibt. Das PPS-Signal steuert noch einen Tongenerator und die, über ein Gatter getriebene, grüne Leuchtdiode. Der Ausgang vom Tongenerator liegt an einer Chinchbuchse an, der Pegel kann über ein Potentiometer eingestellt werden. Der Mikrofoneingang stellt eine Speisung für PC-Mikrofone zur Verfügung. Das Mikrofonsignal geht über einen Verstärker zur Ausgangsbuchse. Der Ausgangspegel kann ebenfalls über ein Potentiometer eingestellt werden.

5 Aufbau

Der Aufbau des VTI ist unproblematisch. Alle Schaltkreise sollten auf Fassungen gesetzt werden. Die Platine wird, bis auf die ICs, bestückt. Anschließend wird die Betriebsspannung von 12V angelegt. Die Stromaufnahme sollte jetzt 4mA betragen. Am Ausgang des 7805-Spannungsreglers müssen 5V anliegen. Es ist empfehlenswert den Spannungsregler zur Inbetriebnahme mit einem Kühlblech zu versehen. Nach dem Zusammenbau wird das Gehäuse zur Kühlung benutzt. Wenn dies der Fall ist wird der 7660 bestückt. Die Stromaufnahme liegt immernoch bei 4mA. Jetzt sind die -5V zu kontrollieren. War auch diese

Messung erfolgreich, werden die restlichen Schaltkreise bestückt. Empfehlenswert ist immer einen Schaltkreis in seine Fassung zu stecken und anschließend die Stromaufnahme zu messen. Der Mikrokontroller muß schon programmiert sein.

Es ergeben sich folgende Ströme:

- LM1881 10mA
- TL081 12mA
- MAX4278 28mA
- 74HCT4053 28mA
- 74AC00 66mA
- ATmega8 90-110mA (rote LED blinkt)
- GPS18LVC 110-130mA

Nun können Kamera und Monitor angeschlossen werden. Mit dem Oszilloskop wird der Videopegel der eingeblendeten Zeichen auf kleiner $1V_{ss}$ eingestellt. Anschließend werden der Mikrofonausgang und der PPS-Tonausgang auf den gewünschten Pegel eingestellt.

5.2 Daten- und Programmieradapter

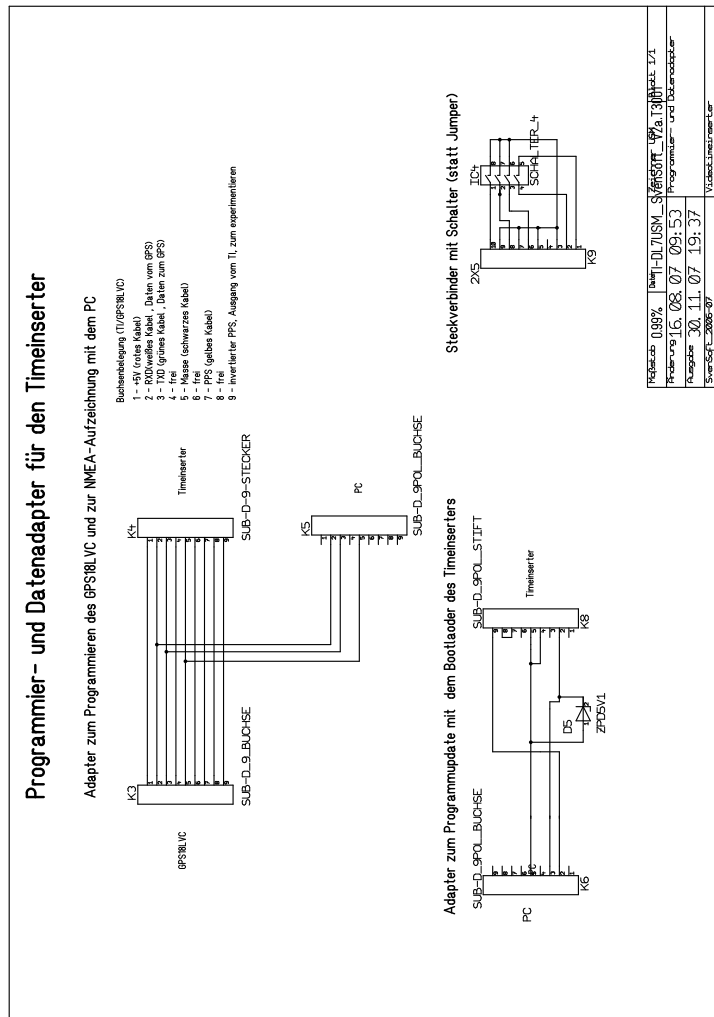


Abbildung 6: Schaltung des Daten- und Programmieradapters

5.3 Bauteilliste

teilleiste]

Nr Name Wert

Gehäuse

ARTICLE REICHEL

5 Aufbau

1	BR2	Teko70x100	TEKO70X100	TEKO B3
2	Bu1	+UB	FLANSCHBUCHSE	Segor Elektronik
3	Bu3	CINCH-ANSCHLUSSPLATTE	4-POLIG	CBP 4
4	Bu4	CINCH-ANSCHLUSSPLATTE	2-POLIG	CBP 2
5	C1	27pF	2,54X5,08	RM5,08
6	C2	27pF	2,54X5,08	RM5,08
7	C3	100NF	C RM5,08	Z5U-5 100n
8	C4	100NF	C RM5,08	Z5U-5 100n
9	C5	100NF	C RM5,08	Z5U-5 100n
10	C6	470µF	C ELKO RM5,08	DM13 RAD 470/63
11	C7	47nF	5X6R2,54	
12	C8	1000µF	C ELKO RM7,62	DM16,5 RAD 1.000/63
13	C9	0,47µF	C RM5,08	Z5U-5 470n
14	C10	100nF	C RM5,08	Z5U-5 100n
15	C11	100nF	C RM5,08	Z5U-5 100n
16	C12	10µF	C ELKO RM2,54 DM6	RAD 105 10/63
17	C13	100nF	2,54X5,08 RM2,54	Z5U-2,5 100n
18	C14	100n	C2	
19	C15	100n	C2	
20	C16	10µF	ELKO1	
21	C17	10µF	ELKO1	
22	C18	100NF	Z5U-2,5	100n
23	C19	100NF	C RM5,08	Z5U-5 100n
24	C20	100NF	C RM5,08	Z5U-5 100n
25	C21	100NF	C RM5,08	Z5U-5 100n
26	C22	100NF	C RM5,08	Z5U-5 100n
27	C23	C-KERAMIK-5,08	50V 10	10NF 6X6R5,08
28	D1	LED ROT	LED RM2,54	LED 5MM RT
29	D2	LED GRUEN	LED RM2,54	LED 5MM GN
30	D3	1N4001	D RM12,7 DM3	1N 4001
31	D4	1N4007	D4	
32	D5	ZPD5V1		
42	FP1	D-SUB9 STANDARD	D-SUB9 STANDARD	
43	IC1	ATMEGA8(DIL28S)	DIL28S	ATMEGA 8-16 DIP
44	IC2	7805	TO220 µA 7805	
45	IC3	LM1881	DIL 8	
46	IC4	SCHALTER 4	DIL8-SCHALTER	
47	IC6	TL081CN	DIL8	TL 081 DIP
49	IC7	ICL7660	DIL8	
50	IC8	74HC00	DIL14	74HC 00
51	IC12	74HCT4053	DIL16	
52	K1	GPS-RX SUB-D	9POL BUCHSE LIEGEND	
53	K2	K1X6 1X06		
54	K3	SUB-D 9-BUCHSE		
55	K4	SUB-D-9-STECKER		
56	K5	SUB-D 9POL BUCHSE		
57	K6	SUB-D 9POL BUCHSE		
58	K7	K2X5	2X05	SL 2X10G 2,54
59	K8	SUB-D 9POL STIFT		
60	K9	2X5	LEITERPLATTENVERBINDER 2X5	LPV 10
61	Q1	18,432MHz	HC-49/U	
62	R1	680K	0207	Metall 6,8K
63	R2	150	0204	
64	R3	4,7K	0204	
65	R4	100	0204	
66	R5	1K	VISHAY 64W	64W-1,0K
67	R6	100	0204	
68	R7	TRIMMER 64W 10-KOHM	TRIMMER 64W	
69	R8	10K	R3	
70	R9	220R	R3	
71	R10	1K	TRIMMER 74X	
72	R11	75R	R3	
73	R12	75R	R3	
74	R13	10K	TRIMMER 74X	
75	R14	22K	R3	
76	R15	R7,62	R3	
77	R16	1,1K	0204	METALL 1,1K
78	R17	47K	0204	
79	R18	75R	0207	Metall 10K
80	R19	10K	0207	Metall 10K
81	R20	47K	0204	
82	R22	10K	0207	Metall 10K

Literatur

- [1] Homepage von Gari Dion, <http://gardion.com>
- [2] AVRDUde <http://www.nongnu.org/avrdude/>
- [3] AVRDUde-GUI <http://sourceforge.net/projects/avrdude-gui/>
- [4] Thomas Flatrès, Jean Jacques Sacré, Société d'Astronomie de Rennes: Technical audit on a video camera WAT LCL 902K (ESOP2007)
- [5] <http://www.atmel.com>
- [6] <http://sourceforge.net/projects/winavr/>
- [7] <http://www.reichelt.de>
- [8] <http://www.segor.de>
- [9] <http://www.meinberg.de>

Abbildungsverzeichnis

1	Ansicht der Platine	3
2	Videobild mit Zeiteinblendung	5
3	Belegung der Buchsen	9
4	Videozeile mit Zeicheneinblendung	10
5	Schaltung des Videotimeinserters	12
6	Schaltung des Daten- und Programmieradapters	13

Tabellenverzeichnis

1	Codierung der Zeitzeile	5
2	Codierung der Positionszeilen	5