Digitale Ansteuerung eines Ultraschallsenders auf Basis der *Virtual Workbench 300*

Vasco Jerinić
Professur Schaltkreis- und Systementwurf
TU Chemnitz

12. Oktober 2001

Inhaltsverzeichnis

1 Übersicht

Der vorliegende Aufbau dient zur Ansteuerung eines Ultraschall-Senders. Die Aufgabe besteht im Bereitstellen von vorgegebenen Bitfolgen auf einer Vielzahl von Kanälen mit einstellbaren Verzögerungen zueinander. Alle Steuerungsparameter sind über eine eingebaute EPP[1]-Schnittstelle programmierbar.

1.1 Basis

Die Steuerung wurde mit Hilfe einer Virtual Workbench 300, im weiteren als VW300 bezeichnet, implementiert. Dabei handelt es sich um eine Prototyping-Lösung der Firma *Virtual Computer Corporation*[2], welche von *Insight*[3] vertrieben wird. Basierend auf einem *XILINX XCV300*[4] stellt die VW300 neben der Umgebung zum Laden und Betrieb des FPGA

- Auswahlmöglichkeiten zu verschiedenen Betriebsarten des FPGA,
- einen Temperatursensor zur Überwachung des FPGA,
- verschiedene RAMs,
- externe programmierbare Taktgeneratoren,
- eine Anzahl von Schnittstellen und Steckerleisten,
- eine Anzahl von Anzeige- und Eingabemöglichkeiten und
- Platz für eigene Erweiterungen

zur Verfügung. Eine genaue Beschreibung findet sich in [1].

2 Steuerung

2.1 Laden des FPGA und Rücksetzen der Steuerung

Das Design liegt in zwei Formen vor:

1. als **.bit* Datei, welche mit Hilfe des Programms *xchecker* (*XILINX*) in den FPGA geladen werden kann und
2. als *mcs*-Datei, die zum Programmieren des auf der VW300 befindlichen Flash-ROM dient. In diesem Fall wird das Design bei entsprechender Konfiguration sofort nach dem Einschalten der VW300 in den FPGA geladen.

[1] Enhanced Parallel Port (*IEEE 1284*), Erweiterung der parallelen Schnittstelle
[2] www.vcc.com
[3] www.insight-electronics.com
[4] FPGA der Virtex-Serie des Hersteller XILINX

Während ein Design in den FPGA geladen wird, leuchtet *LED9.* Sie erlischt, wenn der Vorgang erfolgreich abgeschlossen wurde. Das vorliegende Design wurde so konzipiert, daß nach erfolgreichem Ladevorgang zusätzlich alle LEDs der auf der VW300 befindlichen LED-Zeile aufleuchten, um das Laden des richtigen Designs anzuzeigen.

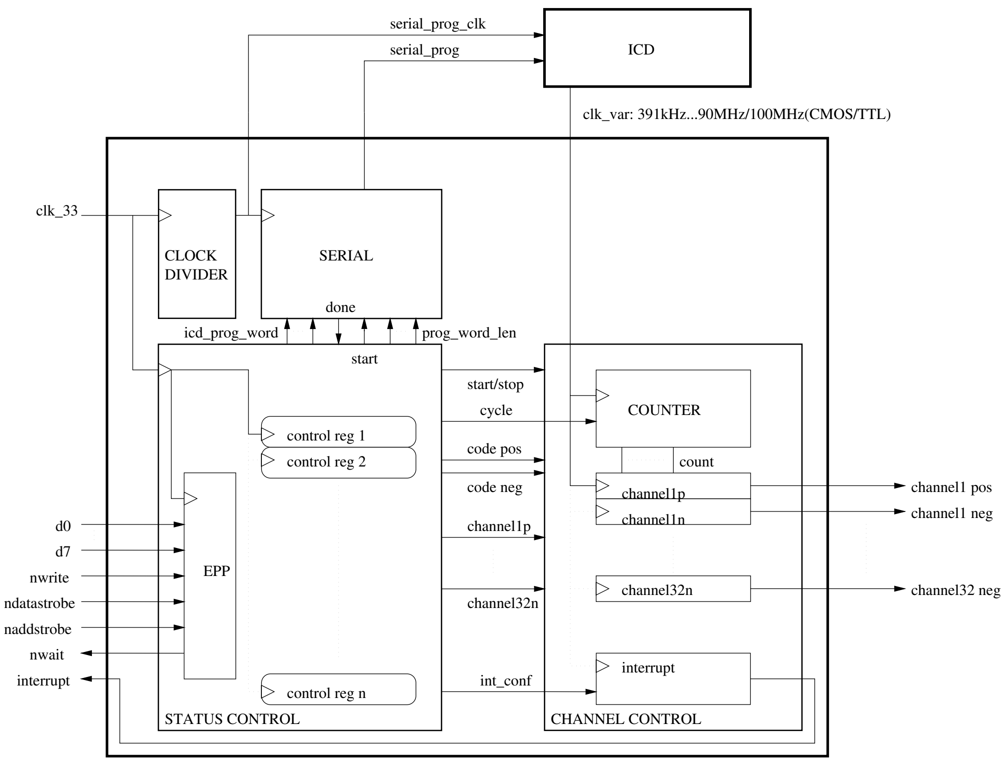
Es existieren zwei verschiedene Möglichkeiten des *RESET:*

- komplettes RESET (über Bestätigen des Schalters *SW5*) inklusive der FPGA-Programmierung., womit ein erneutes Laden des Designs in den FPGA notwendig wird
- RESET der FPGA-internen Register und Automaten (über Bestätigen des Schalters $SW1$), was eine erneute Programmierung der Steuerung via EPP-Schnittstelle erforderlich macht.

2.2 Aufbau und Funktion

Die Steuerung besteht aus 4 getrennten Funktionseinheiten:

- externer programmierbarer Taktgenerator (Bestandteil der VW300)
- Kanalsteuerung (im FPGA)
- Statuskontrolle (im FPGA)
- Komponente zum Programmieren des externen Taktgenerators (im FPGA)



2.2.1 Taktgenerator

Auf der VW300-Platine befindet sich ein programmierbarer Taktgenerator [2] vom Typ *ICD2053B* des Herstellers *Cypress*. Dieser kann über einen seriellen Bitstrom zur Erzeugung von Frequenzen im Bereich von *391 kHz* bis *100 MHz* (TTL) bzw. bis *90 MHz* (CMOS) konfiguriert werden. Dabei darf die Frequenz des seriellen Bitstroms *1 MHz* nicht überschreiten.

Die Länge des seriellen Bitstroms kann bis zu *32 bit* betragen, die einzustellenden Werte können mit Hilfe des Programms *BitCalc*[5], welches von *Cypress* zur Verfügung gestellt wird, berechnet werden.

Nach dem Programmieren des Generators verstreicht typischerweise eine Zeit von bis zu *10 ms* bis die Abweichung der erzeugten Frequenz zum Vorgabewert < *0,1%* ist. Diese Zeitdauer wird als *settle time* bezeichnet

2.2.2 Kanalkontrolle

Die Kanalsteuerung wird mit dem vom externen Taktgenerator erzeugten Takt betrieben, das heißt ihr Takt kann verändert werden. Ein Zähler von *24 bit* Breite dient als Zyklenzähler, beim Erreichen des vorgegebenen Maximalwertes (`cycle_length`) beginnt der Zyklus von vorn. Durch Ändern des externen Taktes kann die Frequenz der ausgegebenen Steuerimpulse angepaßt werden, durch Ändern der `cycle_length` kann die Anzahl der Taktperioden bis zur wiederholten Ausgabe der Steuerimpulse beeinflußt werden. Die Zeit bis zur Wiederholung der Steuersignale ermittelt sich zu

$$t_{cycle} = \frac{1}{f_{clk}} * cyclelength$$

wobei gilt:

f_{clk} eingestellte Frequenz des externen Taktgenerators

$cyclelength$ programmiertes Anzahl von Taktzyklen bevor die Ausgabe des Steuersignale von vorn beginnt

Es stehen 32 unterschiedliche Kanäle zur Verfügung, wobei jeder dieser Kanäle jeweils einen positiven und einen negativen Ausgang besitzt. Für jeden der Kanäle kann eine Verzögerung für den positiven und eine für den negativen Ausgang eingestellt werden. Diesen Wert muß der Zyklenzähler erreichen, bevor die Ausgabe der Steuersignale für den jeweiligen Ausgang gestartet wird. Die zeitliche Verzögerung berechnet sich zu

$$t_{delay} = \frac{1}{f_{clk}} * delay_{ch}$$

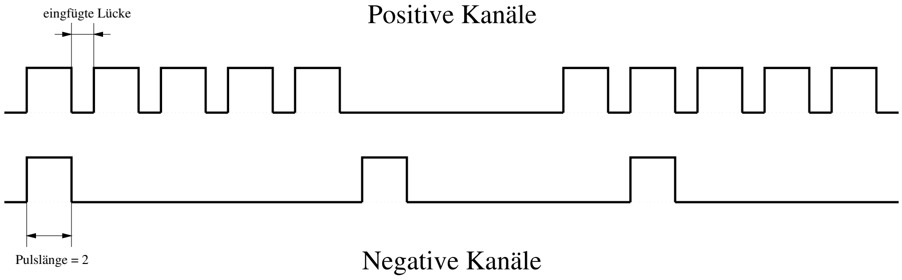
wobei gilt:

[5] Vorraussetzung: Microsoft Windows ab Version 3.1

f_{clk} eingestellte Frequenz des externen Taktgenerators

$delay_{ch}$ programmierte Anzahl von Taktzyklen bevor die Ausgabe des Steuersignals für diesen Kanal beginnt

Die Bitfolge, die als Steuersignal an den Kanalausgängen ausgegeben wird, besteht aus 13 Elementen. Sie ist für positive und negative Kanalausgänge getrennt einstellbar. Die Pulslänge der einzelnen Elemente ist wählbar im Bereich[6] $1...15$. Zwischen den Elementen des Steuersingals automatisch eine "Lücke" mit der Pulslänge *1* eingefügt.



Der positive und negative Ausgang eines jeden Kanals kann einzeln aktiviert bzw. deaktiviert werden.

Besonderheiten

Zykluslänge Es ist darauf zu achten, daß die Kanalverzögerungen immer kleiner sind als die Zykluslänge (`cycle_length`), da der Zykluszähler ansonsten den Wert zum Start des jeweiligen Kanals nie erreicht.

Interrupt Die Kanalsteuerung ist in der Lage, vor dem Beenden eines kompletten Zyklus' an der EPP-Schnittstelle einen Interrupt auszulösen. Der Zeitpunkt dieses Interrupts kann über das Register `int_conf` eingestellt werden, wobei die *23* niederwertigen Bitstellen die Differenz zur Zykluslänge (`cycle_length`) enthalten müssen und das höchstwertige Bit das Auslösen des Interrupts erlaubt bzw. verbietet. Beim Auswählen des Auslösezeitpunkts ist wiederrum zu beachten, daß dieser Wert nicht größer sein darf als die programmierte Zykluslänge.

2.2.3 Statuskontrolle

Die Statuskontrolle wird mit dem FPGA-Basistakt von *33 MHz* betrieben. Sie enthält die Konfigurationsregister, ist für die Kommunikation via EPP-Schnittstelle verantwortlich und verarbeitet die Befehle (Starten der Programmierung des externen Generators, Start/Stop der Kanalsteuerung). *Die EPP-Schnittstelle ist nur zum Schreiben auf die Steuerung ausgelegt, ein Auslesen der internen Konfigurationsregister ist nicht möglich.*

[6] Bei Programmierung des Wertes *0* wird eine Impulslänge von *1* verwendet.

Anhand unterschiedlicher Adreßbereiche entscheidet die Stauskontrolle, ob es sich um eine Register-Adresse handelt, der im nächsten Schreibzyklus ein Datum folgen muß, oder einen auszuführenden Befehl handelt.

Register	Adresse (*HEX*)
code_pos_low	00
code_pos_high	01
code_neg_low	02
code_neg_high	03
channel1p	04
channel1n	05
channel2p	06
channel2n	07
channel3p	08
channel3n	09
channel4p	0A
channel4n	0B
channel5p	0C
channel5n	0D
channel6p	0E
channel6n	0F
channel7p	10
channel7n	11
channel8p	12
channel8n	13
channel9p	14
channel9n	15
channel10p	16
channel10n	17
channel11p	18
channel11n	19
channel12p	1A
channel12n	1B
channel13p	1C
channel13n	1D
channel14p	1E
channel14n	1F
channel15p	20
channel15n	21
channel16p	22
channel16n	23
channel17p	24
channel17n	25
channel18p	26
channel18n	27

Register	Adresse (*HEX*)
channel19p	28
channel19n	29
channel20p	2A
channel20n	2B
channel21p	2C
channel21n	2D
channel22p	2E
channel22n	2F
channel23p	30
channel23n	31
channel24p	32
channel24n	33
channel25p	34
channel25n	35
channel26p	36
channel26n	37
channel27p	38
channel27n	39
channel28p	3A
channel28n	3B
channel29p	3C
channel29n	3D
channel30p	3E
channel30n	3F
channel31p	40
channel31n	41
channel32p	42
channel32n	43
serial1	44
serial2	45
serial3	46
serial4	47
serial_length	48
pulse_length	49
cycle_length1	4A
cycle_length2	4B
cycle_length3	4C
enable_pos1	4D
enable_pos2	4E
enable_pos3	4F
enable_pos4	50
enable_neg1	51
enable_neg2	52
enable_neg3	53

Register	Adresse (*HEX*)
`enable_neg4`	54
`int_conf1`	55
`int_conf2`	56
`int_conf3`	57

Befehl	Adresse (*HEX*)
`INSTR_PROGRAM_SERIAL`	C0
`INSTR_START`	C1
`INSTR_STOP`	C2

Über die EPP-Schnittstelle werden Daten mit einer Breite von *8 bit* übertragen. Konfigurationsregister, deren Breite diesen Wert übersteigt, werden in mehrere Teilbereiche untergliedert, welche über eine eigene Adresse verfügen, um jeden Bereich einzeln beschrieben zu können. Die Gliederung der Register ist in Abb. 1 dargestellt.

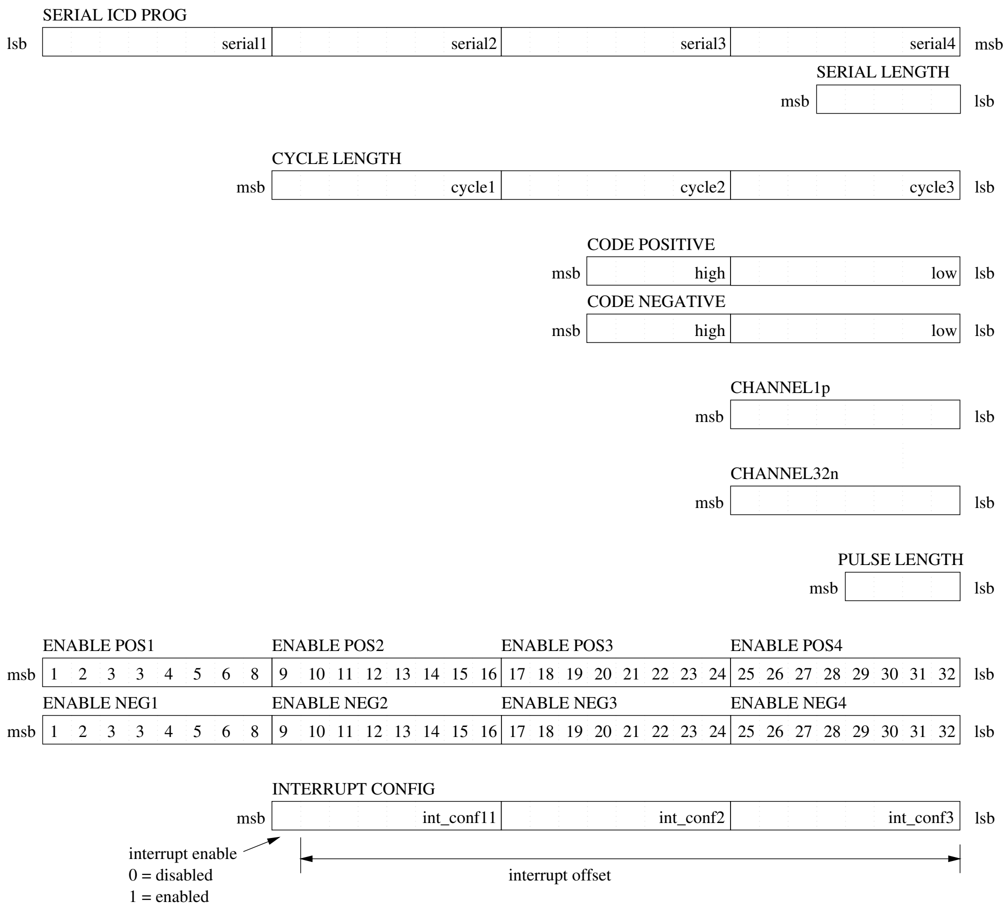


Abbildung 1: Fragmentierung der internen Konfigurationsregister

Nachdem alle Register mit den benötigten Werten programmiert wurden, kann der Befehl `INSTR_PROGRAM_SERIAL` ausgeführt werden. Dazu wird die Adresse `0xC0h` über die EPP-Schnitstelle geschrieben. Die Statuskontrolle gibt diesen Befehl an die Komponente zum Programmieren des externen Taktgenerators weiter und wartet auf deren Rückmeldung, daß die *settle time* (siehe Abschnitt 2.2.1) des Generators verstrichen ist, kann die Kanalsteuerung gestartet werden. Je nachdem, ob der Befehl `INSTR_START` bereits übergeben wurde , wird sofort gestartet oder bis zu Übergabe des Befehls gewartet.

Bevor eine erneute Programmierung des externen Taktgenerators erfolgen oder eines der Konfigurationsregister überschrieben werden kann, muß grundsätzlich der Befehl `INSTR_STOP` ausgeführt werden.

Überprüfen der Programmierung Die Programmierung der Steuerung wird durch ein Rechnerprogramm, das über die parallele Schnittstelle mit der VW300 kommuniziert, vereinfacht. Um Test und *Debugging* des zu erstellenden Programms (und der evtl. zusätzlich benötigten Betriebssystemtreiber) zu erleichtern, wird das jeweils zuletzt via EPP übertragenen Byte (Adresse oder Datum) mit Hilfe der LED-Zeile (LED0...LED7, LED7 = msb) angezeigt. Für jede übertragene unzulässige Adresse (Adresse ist keinem Register und keinem Befehl zugeordnet) wird ein interner Zähler (*EPP address error*) inkrementiert.

Nach abgeschlossener Programmierung kann mit Hilfe des 8bit-DIP-Schalters (*U31*, Anordnung in [1]) die Adresse jedes internen Registers eingestellt (8 = msb) und sein Inhalt durch Betätigen der Taste *SW4* (Anordnung in [1]) angezeigt werden. Wird eine Adresse größer als die höchste Registeradresse eingestellt, wird der Wert des Fehlerzählers (*EPP address error)* angezeigt. Bei korrekter Funktion sollte dieser Wert immer *0* sein.

2.2.4 Programmierung des externen Taktgenerators

Der Takt dieser Komponente wird aus dem FPGA-Basistakt mit Hilfe eines Taktteilers erzeugt. Er beträgt

$$f_{clk_{serial}} = \frac{33\,MHz}{1024} \approx 32\,kHz$$

Die Komponente verfügt intern über die nötigen Informationen zur Kommunikation mit dem Taktgenerator, wie die benötigten Protokoll- und Steuerworte (siehe [2]). Die Statuskontrolle liefert ihr die zu übergebenden Parameter, welche sie nach Senden des Befehls `INSTR_PROGRAM_SERIAL` an den Taktgenerator sendet. Ein interner Zähler ermittelt die die *settle time* (siehe Abschnitt 2.2.1) zu

$$t_{settle} = \frac{1}{f_{clk_{serial}}} * 512 = \frac{1024}{33\,MHz} * 512 \approx 16\,ms$$

Nach Ablauf dieser Zeit erhält die Statuskontrolle die Meldung über das Ende der Programmierphase.

3 Inbetriebnahme und Anschlußbelegung VW300

3.1 Inbetriebnahme

Die Jumperreihe *JP21* (bestehend aus *M0...M2*) dienen der Konfiguration des Ladeverhaltens des FPGA. Bei der ersten Inbetriebnahme sollte dieser so konfiguriert werden, daß er das im Flash-ROM gespeicherte Design lädt (*M0=closed; M1,M2=open*). Im Auslieferungszustand der VW300 wird ein Design zum Test des Prototyping-Systems geladen. Über die Einstellung des 8Bit-DIP-Schalters *U31* kann ausgewählt werden, ob ein Boot-Test durchgeführt wird oder ein Begrüßungstext auf dem vorhandenen Display erscheint.

Adresse (*HEX*) *U31*	Programm
0	Boot-Test
`64` oder `128`	Laufschrift

Nach erfolgreichem Abschließen der Tests kann

- der Flash-ROM durch einen anderen mit dem Design zur Ultraschall-Steuerung ausgetauscht werden oder
- durch Öffnen des des Jumpers *M0* die Nutzung des *xchecker*-Anschlusses ausgewählt werden, was das Laden des Designs vom Rechner aus ermöglicht.

3.2 Anschlüsse

Die Ein- und Ausgänge des FPGA-Designs sind über die um den FPGA plazierten Stiftleisten geführt, bevor sie zur entsprechenden Steckerleiste gelangen.

3.2.1 EPP-Schnittstelle

Zum Anschluß eines EPP-Kabels wurde eine Adapterplatine entworfen, die auf die Steckerleisten *JP11* und *JP12* aufgesteckt wird und eine SUBD-25-Buchse zum Anstecken einer Parallelverlängerung trägt. Die Signale können außerdem an der Stiftleiste *JP53* abgegriffen werden.

ACHTUNG: DIE ANSCHLÜSSE DER STECKERLEISTE *JP12* DÜRFEN AUF KEINEN FALL AUF DER ADAPTERPLATINE VERWENDET WERDEN, DA SIE GLEICHZEITIG ALS KANALAUSGÄNGE VERWENDET WERDEN (SH. ANSCHNITT 3.2.2).

Signal	FPGA	*JP53*	*JP11*
`nwrite`	`C2`	`3`	`3`
`ndatastrobe`	`E3`	`4`	`4`
`d0`	`F4`	`5`	`5`
`d1`	`C1`	`6`	`6`

Signal	FPGA	*JP53*	*JP11*
d2	G4	8	8
naddstrobe	F3	9	9
d3	F2	10	10
d4	G3	11	11
d5	G2	12	12
d6	F1	13	13
d7	J4	14	14
interrupt	H3	15	15
nwait	J2	16	16

3.2.2 Kanalausgänge

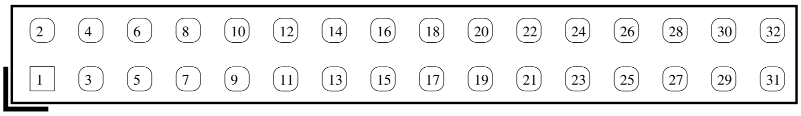
Die Ausgänge der Kanalsteuerung sind über die Steckerleisten *JP47, JP49, JP50* und *JP51* an die Buchsen *JP3, JP4* und *JP7* bzw. an die Steckerleiste *JP12* geführt.

Signal	FPGA	*JP47*	*JP49*	*JP50*	*JP51*	*JP3*	*JP4*	*JP7*	*JP12*
channel1p	C25	3				3			
channel1n	D24	4				4			
channel2p	E23	5				5			
channel2n	C26	6				6			
channel3p	D25	7				7			
channel3n	F23	8				8			
channel4p	F24	9				9			
channel4n	E25	10				10			
channel5p	F25	12				11			
channel5n	F26	13				12			
channel6p	H23	14				13			
channel6n	H24	15				14			
channel7p	G25	16				15			
channel7n	J23	17				16			
channel8p	H25	18				17			
channel8n	K23	19				18			
channel9p	J25	21					33		
channel9n	J26	22					34		
channel10p	L23	23					35		
channel10n	L24	24					36		
channel11p	K25	25					37		
channel11n	M23	26					38		
channel12p	M24	27					39		
channel12n	M25	28					40		
channel13p	N26		3				41		

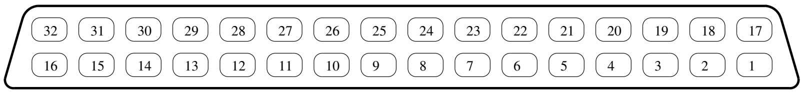
Signal	FPGA	*JP47*	*JP49*	*JP50*	*JP51*	*JP3*	*JP4*	*JP7*	*JP12*
channel13n	P23		4				42		
channel14p	P24		5				43		
channel14n	R26		6				44		
channel15p	R25		7				45		
channel15n	T26		8				46		
channel16p	T25		9				47		
channel16n	T24		10				48		
channel17p	AD6	3		11		3			20
channel17n	AE6	4		12		4			19
channel18p	AD7	5		13		5			18
channel18n	AE7	6		14		6			17
channel19p	AF6	7		15		7			16
channel19n	AC9	8		16		8			15
channel20p	AD8	9		17		9			14
channel20n	AF7	10		18		10			13
channel21p	AD9	12		19		11			12
channel21n	AE9	13		20		12			11
channel22p	AD10	14		21		13			10
channel22n	AF9	15		22		14			9
channel23p	AC11	16		23		15			8
channel23n	AE10	17		24		16			7
channel24p	AD11	18		25		17			6
channel24n	AE11	19		26		18			5
channel25p	AD12	21		27			33		4
channel25n	AE12	22		28			34		3
channel26p	AA26	23	20			35	35		
channel26n	Y25	24	21			36	36		
channel27p	Y24	25	22			37	37		
channel27n	AB25	26	23			38	38		
channel28p	AD14	27			3		39	35	
channel28n	AF15	28			4		40	36	
channel29p	AE15		3		5		41	37	
channel29n	AD15		4		6		42	38	
channel30p	AE16		5		7		43	21	
channel30n	AE17		6		8		44	43	
channel31p	AD16		7		9		45	40	
channel31n	AC16		8		10		46	45	
channel31p	AF18		9		11		47	39	
channel32n	AE18		10		12		48	26	

3.2.3 Steckerbelegung

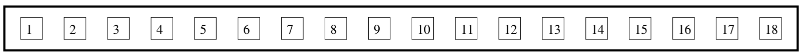
Steckerleisten *JP47, JP49, JP50, JP51:*



Buchsen *JP3, JP4,JP7:*



Steckerleiste *JP12:*



4 Ergebnisse

4.1 Timing

Die statische Timing-Analyse ermittelt eine maximale Taktfrequenz von $\approx$ *54 MHz* $(18,366\,ns)$ für die Statuskontrolle und $\approx$ *62 MHz* $(15,969\,ns)$ für die Kanalsteuerung (wobei in der Kanalsteuerung ein Anteil von mehr als *60%* der größten Pfadverzögerungen auf die Leitungen entfällt). Dieser Umstand legt nahe, daß mit einem größeren FPGA-Typ die Taktfrequenz wesentlich erhöht werden kann.

Das Analyse-Werkzeug geht immer vom ungünstigsten Fall aus, der durch die Konzeption des vorliegenden Designs aber teilweise nicht eintreten kann. Die komplette Implementierung wurde bis zu einem externen Takt von *80 MHz* für die Kanalkontrolle[7] getestet, wobei keine Probleme auftraten, woraus allerdings keine Garantie für die korrekte Funktion der Steuerung für ICD-Frequenzen *> 63 MHz* abgeleitet werden kann.

Die Timing-Analyse wurde auch für einen schnellen FPGA desselben Typs durchgeführt (speed class 6 anstelle von 4), was zu einer Erhöhung der maximal möglichen ICD-Taktfrequenz auf $\approx$ *83 MHz* $(11,983\,ns)$ führte.

Damit bietet sich zwei Möglichkeiten, für eine zukünftige Umsetzung der Steuerung eine höhere Taktfrequenz und damit eine feinere zeitliche Auflösung der Steuersignale zu erreichen:

[7] Im Gegensatz zu den ausführlichen Tests am zugrundeliegenden 16-Kanal-Design, wurden lediglich einige Probeläufe mit speziellen Programmierwerten für verschiedene Frequenzen des externen Taktgenerators durchgeführt.

- Verwendung eines größeren FPGA und/oder
- Verwendung eines schnelleren FPGA-Typs.

Diese Optionen stehen bei Verwendung der verwendeten Prototyping-Umgebung nicht zur Verfügung, da sich der enthaltenen FPGA praktisch nicht austauschen läßt. Nur bei einem komplett neuen Aufbau sollten diese Varianten in Erwägung gezogen werden.

4.2 Verbindungsleitungen

Besonderer Wert ist auf die Qualität von Verbindungen zwischen der VW300 und anderen Teilen des Gesamtaufbaus[8] und deren Abschirmungen zu legen, da es sonst je nach Taktfrequenz zu Laufzeitverzögerungen, Übersprechen zwischen den einzelnen Kanälen bzw. zwischen dem positiven und negativen Ausgang eines Kanals usw. kommen kann. Abb. 2 und zeigt das Übersprechen zwischen benachbarten Pins bei einer externen Taktfrequenz von *50 MHz*.

Erzeugte Steuersignale der Kanäle:

$$positiv = 1111100011111$$
$$negativ = 1000000000001$$

In Abb. 3 ist eine Messung für die gleichen Parameter dargestellt, wobei für die folgenden Meßsignale geschirmte Leitungen verwendet wurden:

- Kanal 14 positiv (*k14_pos*)
- Kanal 15 positiv (*k15_pos*)
- Kanal 16 positiv (*k16_pos*)
- Kanal 1 negativ (*k1_neg*)
- Kanal 2 negativ (*k2_neg*)

Neben dem unterdrückten Übersprechen sind die unterschiedlichen gemessenen Pulsbreiten deutlich zu sehen.

[8] das betrifft insbesondere die Ausgänge der Kanalsteuerung

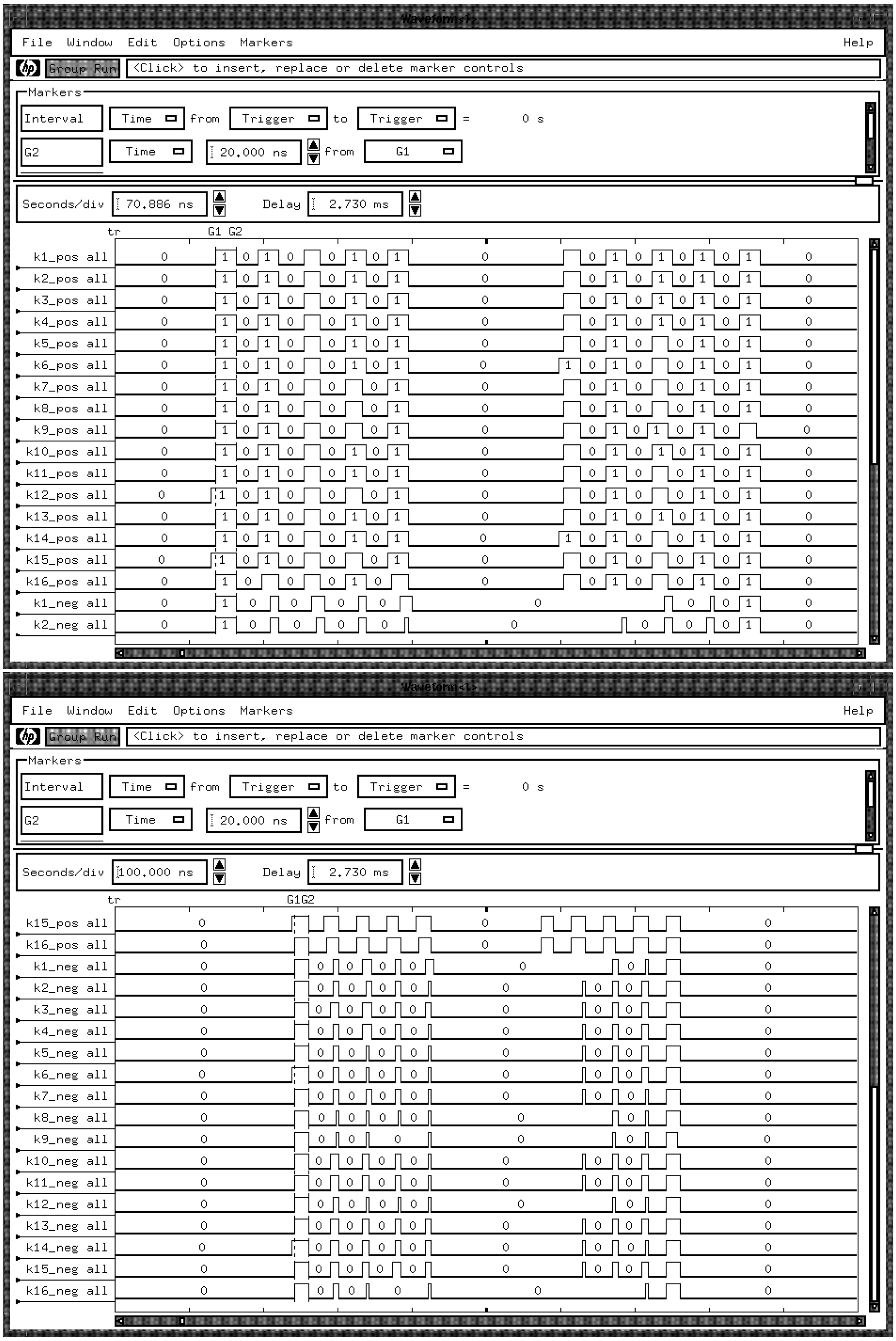


Abbildung 2: Übersprechen bei ungeschirmten Meßleitungen

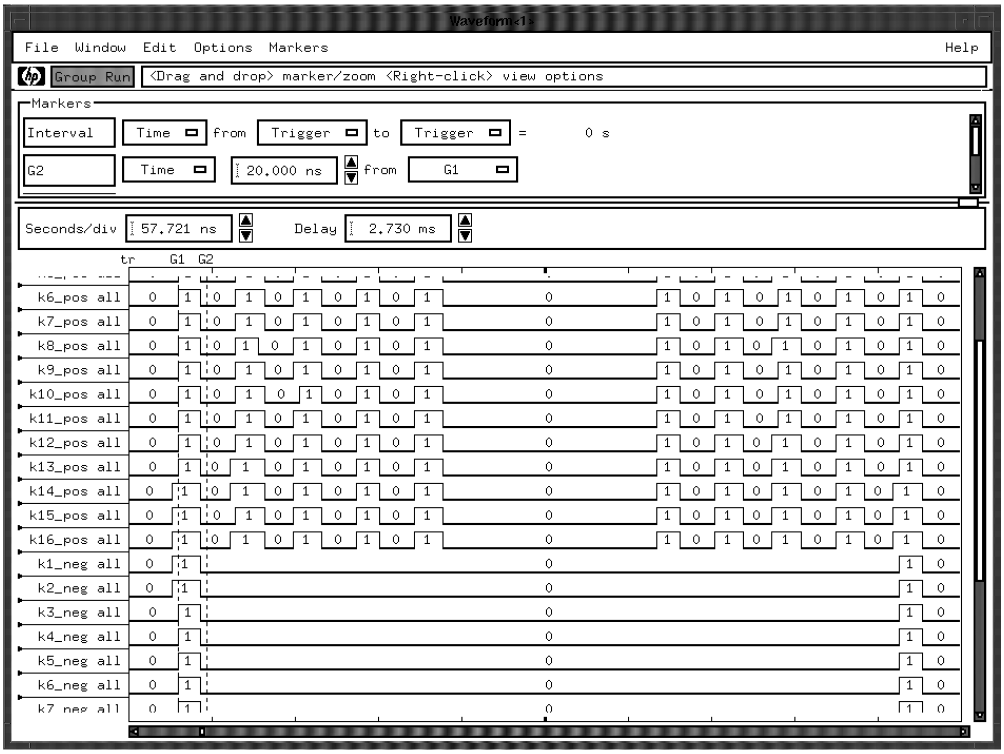


Abbildung 3: Verhindern des Übersprechens durch geschirmte Meßleitungen

Literatur

[1] *The Virtual Workbench Guide*.

[2] WWW.CYPRESS.COM: *ICD2053B Data Sheet*.