

Beschreibung AC1 2010 Color-BWS mit CPLD

Stand 05.11.2011

Der neue AC1 Color-BWS ist ein neuer AC1 **BildWiederholSpeicher** (BWS) und soll den bisherigen ersetzen. Er kann auch parallel zum bisherigen BWS eingesetzt werden. Damit sind mehrere Monitore zugleich ansteuerbar.

Beim CPLD-BWS ist die gesamte Logik außer RAM, Zeichengenerator und Farbausgabe in einem CPLD enthalten. Die Leistungsstärke des CPLD XC95108 von XILINX erlaubt einige Erweiterungen, die jetzt bereits im CPLD-BWS integriert sind.

Merkmale des BWS:

- störungsfreier BWS Zugriff ohne WAIT-Schaltung ☺,
- Umschaltbarkeit zwischen BAS Modus (wie AC1 Original) oder TFT Modus !!,
- Erweiterung auf Farbausgabe einfach durch zusätzliche Bauelemente und Software,
- Einzelzeichen INVERS Modus mit 10H/11H,
- INVERS-Modus (via Software) für ganzes Bild,
- Taktfrequenzumschaltung für AC1 (via Software) zwischen 2/4Mhz integriert,
- rücklesbarer BWS Steuerport auf IO-Adresse F0H,
- Programmierbarkeit des Zeichengenerator (Einsatz 28C64 100ns notwendig),
- Unterstützung SCCH-Modul 3 durch Signal MEMDI-Mod1,
- in der Monochrom-Variante ist kein INIT für den BWS notwendig,

Der BWS stellt einen RAM im Adressbereich von 1000H..17FFH dar, welcher AC1-typisch rückwärts (17FFH ist links oben) ausgelesen und am Bildschirm angezeigt wird. Dabei werden immer 32 Zeilen mit je 64 Zeichen dargestellt.

Der Pegel an Pin 16 von JP3 bestimmt, ob das Signal als SVGA TFT-gerecht (H-Pegel) mit 800x600Pixeln und 60Hz oder als SVGA BAS-Signal (L-Pegel) ausgegeben wird.

In der Farbversion kommt zum normalen RAM (Text-RAM) ein weiterer RAM, der Farbspeicher (Farb-RAM), hinzu. Er liegt parallel zum Text-RAM von 1000H..17FFH und wird durch Setzen von Bit 3 im BWSPort statt dem Text-RAM für die CPU sichtbar. Bit 3 entscheidet also, welcher RAM gerade aktiv gelesen/geschrieben werden kann.

Für jedes Byte im Text-RAM gibt es an gleicher Adresse ein FarbByte, welches die Vordergrund- und Hintergrundfarbe bestimmt. Es sind damit 7 Farben + INTENS jeweils für Vorder- und Hintergrund möglich.

Bit 0	Zeichen ROT	Bit 4	Hintergrund ROT
Bit 1	Zeichen GRÜN	Bit 5	Hintergrund GRÜN
Bit 2	Zeichen BLAU	Bit 6	Hintergrund BLAU
Bit 3	Zeichen INTENSIV	Bit 7	Hintergrund INTENSIV

Für die Bilddarstellung werden Text-RAM und Farb-RAM parallel ausgelesen. Wird vom ZG ein „Hell“-Signal geliefert, werden Bit 0 - 3 ausgegeben, bei einem „Dunkel“-Signal werden die Bits 4 - 7 zur Anzeige geschaltet. Dieses Auslesen läuft für den Programmierer völlig transparent.

Beschreibung AC1 2010 Color-BWS mit CPLD

Stand 05.11.2011

BWSPort F0H:

Neu ist das vom AC1 **schreib und rücklesbare** Latch auf IO-Adresse F0H. Dieses 8bit breite Latch steuert das Verhalten des BWS und vom AC1 (Taktumschaltung). Es hat nach Reset den Wert 00H und wird nur von der CPU gelesen bzw. geschrieben. Der CPLD selbst verändert das Latch nicht.

Bit 0 schaltet den AC1-Takt zwischen 2Mhz (Bit=0) und 4Mhz (Bit=1),
Bit 1 schaltet zwischen NormalModus (Bit=0) und INVERS Modus (Bit=1),
Bit 2 schaltet zwischen Text-RAM (Bit=0) und Farb-RAM (Bit=1),

Die Bits3..7 sind für die Programmierung des Zeichengenerator ZG reserviert.

Bit 3 --> ZG Programmierung 0=Normal BWS, 1=ZG lesen/schreiben
Bit 4 --> ZG Programmierung A0
Bit 5 --> ZG Programmierung A1
Bit 6 --> ZG Programmierung A3
Bit 7 --> frei

Wird ein programmierbarer ZG verwendet, muss ein 28C64 (ATMEL) mit max. 100ns eingesetzt werden und JP 5 geschlossen werden.

Erläuterung Programmierung ZG:

Mit Setzen von Bit 3 im BWSPort wird Programmier-Modus geschaltet. D.h. der BWS wird dunkel und der ZG kann mit einem einfachen Lese- oder Schreibbefehl gelesen bzw. geschrieben werden. Beim Schreiben muss jedoch softwareseitig die Schreibzeit abgewartet werden, ehe die Daten im ZG gültig sind.

Da ein Zeichen auf dem BWS aus 8 Bytes im ZG besteht, müssen immer 8 Bytes gelesen oder geschrieben werden, um ein vollständiges Zeichen zu erhalten.

Adressiert wird der ZG wie folgt:

BWS Adresse 1000H + Offset akt. Zeichen = Basisadresse ZG (Zeile 1 vom Zeichen)

Die Zeilennummern des akt. Zeichens (A0..A2 ZG) müssen im BWSPort in den Bits 4..6 eingestellt und dort von der Programmerroutine hoch gezählt werden.

PIO1 Port B3 entspricht der A11 vom ZG und schaltet damit den Zeichensatz des ZG um.

D.h. der ZG wird für Zeichen 00H auf 1000H, 21H auf 1021, ... FFH auf 10FFH erreicht.

Soll der ZG programmiert werden, muss zuvor JP 4 in Position 2-3 gesteckt werden. Es empfiehlt sich nach dem Programmieren JP 4 wieder nach 1-2 zu setzen, damit eventuelle Programmabstürze nicht zu ungewollten Programmiersequenzen führen. Dies ist zwar im Testbetrieb nur 1x aufgetreten und da auch nur für einzelne Bytes, ratsam ist das Setzen des JP 4 in jedem Fall.

Beschreibung AC1 2010 Color-BWS mit CPLD

Stand 05.11.2011

So funktioniert die ZG Programmiersequenz:

1. richtigen ZG (SCCH oder ACC) einstellen (PIO1 Port B3)
2. Zeilennummer vom Zeichen nach BWS-Latch F0H Bit4..6, Bit7=0
3. Bit 3 auf 1, damit wird RAM deaktiviert und ZG_OE auf H (ZG ProgrammierMode)
4. Schreib oder Lesebefehl auf BWS Adresse 10xxH
5. beim Schreiben Schreibzeit mit Warteschleife abwarten
6. nächste ZG Zeile in IOF0H einstellen und weiter zu 3.
7. wenn alle 8 Zeilen durch ist das Zeichen fertig gelesen oder geschrieben,

FarbReset:

Der CPLD-BWS verhält sich nach dem Einschalten bzw. Reset wie ein ganz normaler BWS. Lediglich die Farbversion benötigt eine EINMALIGE Initialisierung des FarbRAM.

Dieser FarbReset sollte bereits im Monitor erfolgen und zwar VOR der Monitor Überschrift. In den vorliegenden Monitorversionen „10.2“ und „AC1 2010 11/2011“ ist dieser FarbReset bereits enthalten.

Um für weitere Software nicht jedes Mal den Test auf die Existenz des FarbBWS machen zu müssen, gibt es eine neu genutzte Speicherstelle „FarbByte“ an 181FH.

Der Wert wird von der FarbReset-Routine gesetzt und zwar mit dem beim FarbReset verwendeten FarbByte. Steht dort 00H gibt es KEINEN FarbBWS.

Damit ist eine leichte Erkennung ob ein FarbBWS existiert, von externer Software möglich.

Bedeutung Jumper:

- JP1 JTAG Adapter nach Xilinx, zur Programmierung des CPLD,
- JP2 RW-Mode 1-2 BWS kann nur beschrieben werden, 2-3 BWS beschreib/lesbar,
- JP3 Ausgabeport an Monitor, Pin 16 gleichzeitig Eingang TFT-Mode (H),
- JP4 ZG Schreibschutz, 1-2 ZG schreibgeschützt, 2-3 ZG beschreibbar,
- JP5 ZG EPROM/EEPROM, JP geschlossen ZG=EEPROM, JP offen ZG=EPROM
- JP6 Takt, JP geschlossen Takt für AC1 2010, JP offen alle anderen AC1

Wissenwertes:

Entstanden ist dieses Projekt aus meiner Idee, einen platzsparenden BWS zu bauen der gleichzeitig auch in der Lage ist, moderne TFT-Monitore anzusteuern.

Bei der Analyse der Schaltung der originalen Schaltung des BWS vom AC1 störte der erhebliche Bauelementeaufwand und so kam ich auf die Idee die gesamte Logik in einem programmierbaren IC unterzubringen.

Meine Wahl fiel auf den XILINX CPLD XC95108 weil er die meisten Ressourcen zur Verfügung stellt und noch nicht in SMD Bauweise, sondern im PLCC84 Gehäuse kam.

Recht schnell war klar, dass ich die bisherige Logik komplett im CPLD abbilden konnte und erste Test brachten auch schnell erste Erfolge, sodass der 1. Musteraufbau, damals noch mit einem Latch vor dem ZG, recht schnell auch funktionierte.

Beschreibung AC1 2010 Color-BWS mit CPLD

Stand 05.11.2011

Hier ein Foto vom 1. Testaufbau, damals noch mit 6116 als RAM.



Nach längerem probieren und wieder verwerfen, wurde dann der 6116 durch den DualPortRAM ersetzt und endlich die störenden BWS-Zugriffe beseitigt, ohne WAIT nutzen zu müssen.

Jetzt war die Zeit reif das Projekt zu veröffentlichen und Heiko Poppe erfuhr davon.

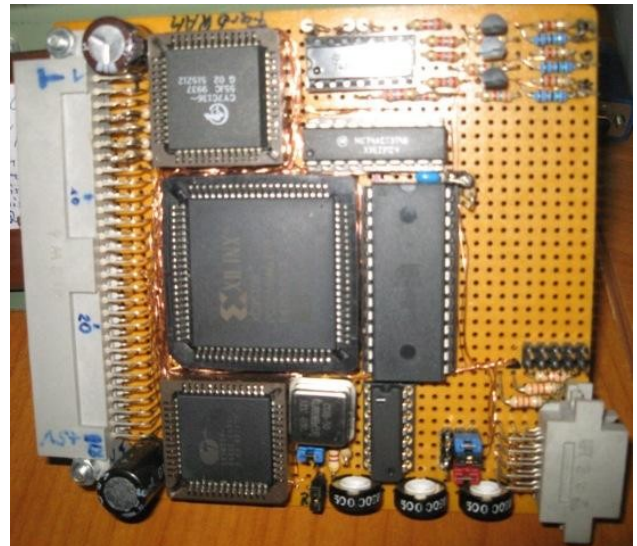
Seine Idee war es, den SW-BWS um die Farbausgabe zu erweitern.

Dabei hat er die damals verwendete Codebasis für den CPLD benutzt und angepasst.

Als sein Farb-BWS dann lief, hab ich das Projekt ebenfalls noch einmal nachgebaut und weiter am CPLD-Code gefeilt. Hier kamen dann die zusätzlichen Funktionen mit herein.

Aufgrund der breiten Resonanz der AC1 Gemeinde entwickelte Heiko dann die vorliegende LP für den AC1 2010.

Parallel dazu galt es die entstandene Software für die Farbausgabe zu optimieren und in die Monitore einzufügen.



Den Code für die Farbausgabe im HRDOS & HRCPM hat Heiko entwickelt, die Monitorroutinen stammen mir.

Heiko und Ralph wünschen viel Freude mit dem COLOR-BWS 😊

PS: Anregungen und Hinweise gern an ralphhaensel@gmx.de