

Multi Player Animator#

by Peter Finzel

Table of Contents

- [Multi Player Animator](#)
- [Atari Magazin Article 6/87 Page 36/37 \(click on Image for full resolution\)](#)
- [Atari Magazin Article 1/88 Page 62-65 \(click on Image for full resolution\)](#)
- [MPA in Assembler](#)
- [MPA in ACTION!](#)
- [PDF and Disk Images](#)

Atari Magazin Article 1/88 Page 62-65 (click on Image for full resolution)<#>

MPA in Assembler#

```
*****
* Animation von Multicolor-Players
* die mit
* MULTIPLAYER-ANIMATOR
* gezeichnet wurden
*
* P. Finzel 1987 Assembler: ATMAS-II
*****
```

```
HPOSP0 EQU $D000 Hor.-Position
GRACTL EQU $D01D Graphik-Kontrollreg.
PMBASE EQU $D407 PM-Basisadresse
SDMCTL EQU $22F DMA-Kontrollreg.
GPRIOR EQU $26F Prioritaeten
SETVBV EQU $E45C Interrupt einfuegen
XITVBV EQU $E462 Ende des Interrupts
```

```
*
* ORG $600 in PAGE 6
```

```
*
* Einsprung-Tabelle
```

```
*
* JMP USER Werte uebergeben
* JMP START Animation starten
```

```
*
* Uebergabevariable
```

```
*
X1POS DFB 0 X-Positionen
X2POS DFB 0
Y1POS DFB 0 Y-Positionen
Y2POS DFB 0
ANF1 DFB 0 Anfangs-Shapes
ANF2 DFB 0
END1 DFB 0 End-Shapes
END2 DFB 0
GES1 DFB 0 Animations-Geschw.
GES2 DFB 0
SHPAG1 DFB 0 Page-Adresse der
SHPAG2 DFB 0 Shapetabellen
```

```
*
* Offsets zum Eintragen (fuer USR)
```

```
*
TAB DFB 0,1,4,5,8,9
```

```
* Interne Variablen:
```

```
PMPAG1 DFB 0 Page-Adresse Player 1
PMPAG2 DFB 0 Page-Adresse Player 2
SHP1 DFB 0 aktuelle Shapes
SHP2 DFB 0
VERZ1 DFB 0 Zaehler fuer Verzoegerung
VERZ2 DFB 0
Y1ALT DFB 0 alte Y-Positionen
Y2ALT DFB 0
```

```
*
* Zero-Page Register
```

```
*
ZREG1 EQU $CC
```

ZREG2 EQU \$CE

*-----
* Routine zur Uebergabe der Parameter
*-----

```
USER      PLA
          PLA      welche Parameter
          PLA      werden uebergeben?
          TAX      Ort des Eintrages
          LDA TAB,X aus Tabelle
          TAX
          PLA      ersten Parameter
          PLA      eintragen
          STA X1POS,X
          PLA      und zweiten Parameter
          PLA
          STA Y1POS,X
          RTS
```

*-----
* Vorbereitungsroutine
* - loescht PM-Bereich
* - schaltet PM-Graphik ein
* - aktiviert VBI
*-----

```
START     CLD      zur Sicherheit
          PLA
          PLA      wo soll PM-Bereich
          PLA      liegen?
          STA PMBASE
          CLC      Berechnung der Player
          ADC #4   Adressen
          STA PMPAG1
          STA ZREG1+1
          ADC #2
          STA PMPAG2
```

*
* PM-Bereich loeschen
*

```
          LDA #0
          STA ZREG1
          LDX #4   vier Pages loeschen
          TAY
PMCLR     STA (ZREG1),Y
          DEY      Schleife zum Loeschen
          BNE PMCLR
          INC ZREG1+1
          DEX
          BNE PMCLR
```

*
* PM-Graphik einschalten
*

```
          LDA #$11   Multicolor-Option
          STA GPRIOR aktivieren
          LDA #$3A   Player-DMA ein-
          STA SDMCTL schalten
          LDA #2     Player Darstellung
          STA GRCTL  einschalten
```

*

```

LDX #1      Interne Variable
VORBER LDA GES1,X   vorbereiten:
STA VERZ1,X Verzoeigerung
LDA ANF1,X   und Anfangs-Shape
STA SHP1,X
DEX
BPL VORBER

*

LDY #PMVBI:L VBI einrichten
LDX #PMVBI:H
LDA #7      hier: deferred
JSR SETVBV  VB-Interrupt
RTS        fertig!

*-----
* Interrupt-Routine fuer PM-Graphik
*-----
PMVBI  LDX #0      Multiplayer 1
      JSR PMCOPY   bearbeiten
      STA HPOSP0   Horizontal-Werte
      STA HPOSP0+1 eintragen
      INX          dann kommt Multi-
      JSR PMCOPY   Player 2 dran
      STA HPOSP0+2 X-Wert eintragen
      STA HPOSP0+3
VBIENDE JMP XITVBV Interrupt beenden

*-----
* Animation eines Multiplayers
*   Eingabe: <X> 0:M.-Player 1 bearb.
*           1:M.-Player 2 bearb.
*   Ausgabe: <A>: X-Position
*-----
PMCOPY DEC VERZ1,X  neues Shape?
      BNE PMC2     nein -->
PMC1  LDA GES1,X   Verzoeigerung
      STA VERZ1,X  neu einrichten
      INC SHP1,X   naechstes Shape
      LDA END1,X   schon uebers
      CMP SHP1,X   Ende des Bereiches?
      BCS PMC2     nein -->
      LDA ANF1,X   Anfangsshape
      STA SHP1,X   in Shape-Zaehler

PMC2  LDA SHP1,X   Relative Adresse
      ASL          ;des Shapes berechnen
      ASL
      ASL
      ASL
      STA ZREG1    in Zeropage-Register
      LDA SHPAG1,X fuer Daten-Quelle
      STA ZREG1+1  eintragen
      LDA PMPAG1,X Ziel ist der PM-
      STA ZREG2+1 Bereich

      JSR MOVE     Datentransfer Player 1

      LDA ZREG1    Adressen fuer
      ORA #128     Ziel und Quelle

```

```

        STA ZREG1      von Player 2
        INC ZREG2+1

        JSR MOVE
        LDA Y1POS,X   Alte Position
        STA Y1ALT,X   merken
        LDA X1POS,X   X-Position zurueck-
        RTS           geben
*
*-----
*  Datentransfer vom Shapespeicher
*  in den PM-Bereich
*  <X>: Nummer des M.-Players
*-----
MOVE      LDA Y1ALT,X   zuerst wird das
          STA ZREG2     alte Shape
          LDA #0        geloescht
          LDY #15       Laenge 16 Bytes
LOESCH   STA (ZREG2),Y
          DEY
          BPL LOESCH
          LDA Y1POS,X   Neue Position wird
          STA ZREG2     und neues Shape
          LDY #15       kopieren
COPY     LDA (ZREG1),Y
          STA (ZREG2),Y
          DEY
          BPL COPY
          RTS

```

MPA in ACTION!#

- [Multi Player Animation](#)

PDF and Disk Images#

[Multi Player Animator/lf8-188a.atr](#) [Multi Player Animator/lf8-188b.atr](#) [Multi Player Animator/lf8-687b.atr](#) [Multi Player Animator/mpa.PDF](#) [Multi Player Animator/multiplayeranimator.pdf](#)