Biblioteki systemowe Action!

cz. 2

Opracował Atari Fan ,atari8.eu, 2020

wersja 230119

Print	InputB	Graphics	SCompare	Rand
PrintE	InputC	SetColor	SCopy	Break
PrintD	InputI	Plot	SCopyS	Error
PrintDE	InputBD	DrawTo	SAssign	Peek
PrintB	InputCD	Fill		PeekC
PrintBD	InputID	Position	StrB	Poke
PrintBDE	InputS	Locate	StrC	PokeC
PrintC	InputSD	Sound	StrI	Zero
PrintCE	InputMD	SndRst		SetBlock
PrintCD	GetD	Paddle	ValB	MoveBlock
PrintCDE		PTrig	ValC	
PrintI	Open	Stick	ValI	color
PrintIE	Close	STrig		device
PrintID	XIO			Trace
PrintIDE	Note			List
PrintF	Point			EOF()
Put				
PutE				
PutD				
PutDE				

PrintH – procedura nie opisana w oficjalnych podręcznikach OSS Action!

Zmienne zdefiniowane w Action!:

BYTE color adres zmiennej: color=$02FD, wartość koloru dla operacji np. PLOT, DRAWTO, FILL, etc.
BYTE device adres zmiennej: device=$B7, numer kanału dla operacji np. PRINT, PUT, etc.
BYTE Trace dyrektywa dla monitora, włącza/wyłącza tryb wyświetlania wejść do procedur
BYTE List adres zmiennej: list=$49A, włącza/wyłącza listowanie źródła w czasie kompilacji
BYTE ARRAY **EOF**(8) 8 bajtowa tablica zwraca status EOF we wskazanym kanale CIO

SYS.ACT

```
;SYS.ACT (c) 1983,1984 ACS Copyright (c) 1983,1984 by Action Computer Services (ACS) All rights reserved. No part of this work may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording or by any information storage or retrieval system without permission in writing from ACS. No portion of this software may be incorporated within or included with another computer program that is to be sold, reproduced, or transfered to other parties in any manner without the express written permission of Action Computer Services or its agent, OSS, Inc.
; version 1.4
; last modified March 27, 1984

DEFINE STRING="CHAR ARRAY"
DEFINE EOL="$9B"
DEFINE OpenBuf =  "$0500"
DEFINE OpenBufL = "$00"
DEFINE OpenBufH = "$05"
STRING copy_right(0) = "(c)1983 Action Computer Services"

                    ; Primitive IO routines

PROC Clos=*(BYTE d)
[$FFA2$A686$CA0$AD0]

PROC Output=*(BYTE d, STRING s)
[$A684$BA0$4D0]

PROC In=*(BYTE d, STRING s)
[$A684$5A0$A586$A2$0$A386]

PROC XIOstr=*(BYTE d,x,c,a1,a2,STRING s)
[$A0A$A0A$98AA$9D$342$A3A5$AF0$9D$34A$A4A5$9D$34B$A9$0$9DA8$349$A5B1
$9D$348$12F0$18$A5A5$169$9D$344$A6A5$69$0$9D$345$4C$E456$60]

PROC Opn=*(BYTE d,STRING s,BYTE m,o)
[$A586$A684$3A0$4C XIOstr]

PROC Prt=*(BYTE d,STRING s)
[$A586$A684$A2$0$A386$9A0$20 XIOstr $AD0$BA9$9D$342$9BA9$4C$E456$60]

PROC Error (BYTE err)
[$6C$A$0$1113$8301]

PROC Break=*()
[$BA$8E$4C1$80A0$98$4C Error]

                    ; math library routines

PROC Lshift=*()
[$84A4$AF0$8586$A$8526$88$FAD0$85A6$60]

PROC Rshift=*()
[$84A4$AF0$8586$8546$6A$88$FAD0$85A6$60]

PROC SetSign=*()
[$D3A4$1010]
```

```
PROC SS1=*()
[$8685$8786$38$A9$0$86E5$A8$A9$0$87E5$AA$98$60]
```

```
PROC SMOps=*()
[$D386$E0$0$310$20 SS1 $8285$8386$85A5$E10$AA$D345$D385$84A5$20SS1$8485
$8586$A9$0$8785$60]
```

```
PROC MultB=*()
[$1BF0$CA$C786$AA$15F0$C686$A9$0$8A2$A$C606$290$C765$CA$F6D0$18$8765
$8785$86A5$87A6$60]
```

```
PROC MultI=*()
[$20 SMOps $82A6$1BF0$C686$84A6$15F0$CA$C786$8A2$A$8726$C606$690$C765$290
$87E6$CA$F0D0$8685$82A5$85A6$20 MultB $83A5$84A6$20 MultB $4C SetSign]
```

```
PROC DivI=*()
[$20 SMOps $85A5$27F0$8A2$8226$8326$8726$38$83A5$84E5$A8$87A5$85E5$490$8785
$8384$CA$E7D0$82A5$2A$A2$0$83A4$8684$4C SetSign $10A2$8226$8326$2A$4B0$84C5
$390$84E5$38$CA$EFD0$8226$8326$8685$82A5$83A6$4C SetSign]
```

```
PROC RemI=*()
[$20 DivI $86A5$87A6$60]
```

```
PROC Sargs=*()
[$A085$A186$A284$18$68$8485$369$A8$68$8585$69$0$48$98$48$1A0$84B1$8285$C8$84
B1$8385$C8$84B1$A8$B9$A0$0$8291$88$F810$11A5$FD0$11E6$4C Break $6308$1109
$1819$2113$3323$60]
```

```
SET $4E4=Lshift
SET $4E6=Rshift
SET $4E8=MultI
SET $4EA=DivI
SET $4EC=RemI
SET $4EE=Sargs
```

```
PROC ChkErr=*(BYTE r,b,eC)
[$1610$88C0$8F0$98$80C0$12F0$4C Error $8A$4A4A$4A4A$98AA$9D EOF $60]
```

```
PROC Break1=*(BYTE err)
[$1A2$1186$48$20 Break $68$A8$60]
```

```
PROC Open=*(BYTE d,STRING f,BYTE m,a2)
[$48$A186$A284$A8$A9$0$99 EOF $A8$A1B1$8D OpenBuf $A8$C8$9BA9$2D0$A1B1
$99 OpenBuf $88$F8D0$68$A2 OpenBufL $A0 OpenBufH $20 Opn $4C ChkErr]
```

```
PROC PrintE=*(STRING s)
[$A186$AA$A1A4$A5 device]
```

```
PROC PrintDE=*(BYTE d,STRING s)
[$20 Prt $4C ChkErr]
```

```
PROC Close=*(BYTE d)
[$20 Clos $4C ChkErr]
```

```
PROC Print=*(STRING s)
[$A186$AA$A1A4$A5 device]
```

```
PROC PrintD=*(BYTE d,STRING s)
```

[$20 **Output** $4C **ChkErr**]
PROC InS=*() [$20 **In** $A084$BD$348$3F0$38$1E9$A0$0$A591$A0A4$60]
PROC InputS=*(STRING s) [$A286$AA$A2A4$A5 **device**]
PROC InputSD=*(BYTE d,STRING s) [48FFA9$A385$68]
PROC InputMD=*(BYTE d,STRING s,BYTE m) [48A186$A284$A0$0$A3A5$A191$68$A2A4]
PROC **InputD**=*(BYTE d,STRING s) [$20 **InS** $4C **ChkErr**]
CHAR FUNC **GetD**=*(BYTE d) [$7A2]
PROC **CCIO**=*() [$A386$A0A$A0A$AA$A3A5$9D342A9$0$9D$348$9D$349$98$20$E456$A085$4C **ChkErr**]
PROC **PutE**=*() [$A9$9B]
PROC Put=*(CHAR c) [AAA5 **device**]
PROC **PutD**=*(BYTE d,CHAR c) [$A186$A1A4]
PROC **PutD1**=*() [$BA2$4C **CCIO**]
PROC **PutDE**=*(BYTE dev) [$A0$9B$F7D0]
PROC **XIO**=*(BYTE d,f,c,a1,a2,STRING s) [$20 **XIOstr** $4C **ChkErr**]
PROC **CToStr**=*() [$D485$D586$20$D9AA20D8E6$FFA0$A2$0$C8$E8$F3B1$9D$550$F710$8049$9D$550 $8E$550$60]
PROC PrintB=*(BYTE n) [$A2$0]
PROC **PrintC**=*(CARD n) [$20 **CToStr** $A5 **device**]
PROC **Pnum**=*() [$50A2$5A0$20 **Output** $4C **ChkErr**]
PROC PrintBE=*(BYTE n) [$A2$0]
PROC PrintCE=*(CARD n) [$20 **PrintC** $4C **PutE**]

PROC PrintBD=*(BYTE d, n)
[$A0$0]

PROC **PrintCD**=*(BYTE d, CARD n)
[$A085$8A$A284$A2A6$20 **CToStr** $A0A5$4C **Pnum**]

PROC PrintBDE=*(BYTE d, n)
[$A0$0]

PROC PrintCDE=*(BYTE d,CARD n)
[$20 **PrintCD** $A0A5$4C **PutDE**]

PROC **PrintI**=*(INT n)
[$A286$AA$A2A4$A5 **device**]

PROC **PrintID**=*(BYTE d,INT n)
[$C0$0$1610$48$A186$A284$2DA0$20 **PutD1** 38A9$0$A1E5AAA9$0$A2E5$A8$68 $4C **PrintCD**]

PROC PrintIE=*(INT n)
[$20 **PrintI** $4C **PutE**]

PROC PrintIDE=*(BYTE d,INT n)
[$20 **PrintID** $A0A5$4C **PutDE**]

PROC StrB=*(BYTE n, STRING s)
[$A286$A384$A2$0$A2A4]

PROC StrC=*(CARD n, STRING s)
[$A284$20 **CToStr** $C8$B9$550$A291$88$F810$60]

PROC StrI=*(INT n, STRING s)
[$E0$0$ED10$A085$A186$A284$38$A9$0$A0E5$A8$A9$0$A1E5AA98$20 **CToStr** $E8 $8A$A8$B9$54F$A291$88$F8D0$8A$A291$C8$2DA9$A291$60]

BYTE FUNC InputB=*()

CARD FUNC InputC=*()

INT FUNC InputI=*()
[$A5 **device**]

BYTE FUNC InputBD=*(BYTE d)

CARD FUNC InputCD=*(BYTE d)

INT FUNC InputID=*(BYTE d)
[$13A2$8E$550$50A2$5A0$20 **InputD** $50A9$5A2]

BYTE FUNC ValB=*(STRING s)

CARD FUNC ValC=*(STRING s)

INT FUNC ValI=*(STRING s)
[$A485$A586$A0$0$A084$A184$A284$A4B1$A385$A3E6$20A9$C8$A4D1$5D0$C8$A3C4 $F730$A4B1$2DC9$3D0$A285$C8$A3C4$3610$A4B1$30C9$3030$3AC9$2C10$38$30E9 AAA1A5$48$A0A5AA126AA126$18$A065$A085$68$A165$A185$A006$A126$18 $8A$A065$A085$290$A1E6$C8$A3C4$CA30$A2A5$DF0$38$A9$0$A0E5$A085$A9$0 $A1E5$A185$60]

PROC **PrintH**=*(CARD n)

[$A485$A586$4A9$A685$24A9$20 **Put** $A9$0$4A2$A406$A526$2ACAF8D0$3069$3AC9 $230$669$20 **Put** $A6C6$E5D0$60]

PROC PrintF=*(STRING f, CARD a1,a2,a3,a4,a5)
[$C085$C186$8C$5F0$A0$0$C0B1$C285$C2E6$DA2$A2B5$9D$5F0$CA$F8D0$8B86 $8A86]

PROC **PF2**=*()
[$8AE6$8AA4$C2C4$DAB0$C0B1$25C9$FD0$8AE6$C8$C0B1$25C9$6F0$45C9$8D0$9BA9 $20 **Put** $4C **PF2**$8BA4$8BE6$8BE6$A085$B9$5F0$BE$5F1$A0A4$43C0$E6F0$53C0$6D0 $20 **Print** $4C **PF2** $49C0$6D0$20 **PrintI** $4C **PF2** $48C0$6D0$20 **PrintH** $4C **PF2** $20 **PrintC** $4C **PF2**]

PROC Note=*(BYTE d,CARD POINTER s,BYTE POINTER o)
[$A186$A284$A0A$A0AAA26A9$9D$342$20$E456$20 **ChkErr** $A0$0BD34E$A391 BD34C$A191BD34D$C8$A191$60]

PROC Point=*(BYTE d,CARD s,BYTE o)
[$A186$A0A$A0A$98AA$9D$34D$A1A5$9D$34C$A3A5$9D$34E$25A9$9D$342$20$E456 $4C **ChkErr**]

; graphics routines

STRING **dev_S**="S:", **dev_E**="E:"

PROC Graphics=*(BYTE m)
[48A9$0$20 **Close** $CA9$A385$A9$0$AE **dev_E** $AC **dev_E** +1$20 **Open** $6A9$20 **Close** 68A485$3029$1C49$A385$6A9$AE **dev_S** $AC **dev_S**+1$4C **Open**]

PROC **Position**=*(CARD c,BYTE r)
[$5B85$5C86$5A84]

PROC **Pos1**=*()
[$5585$5686$5484$60]

PROC **GrIO**=*()
[$20 **Pos1** AD2FD$8D$2FB$AD **dev_S** $A585$AD **dev_S** +1$A685$A9$0$A385$A485$6A9 $60]

PROC DrawTo=*(CARD c,BYTE r)
[$20 **GrIO** $11A0$4C **XIO**]

BYTE FUNC Locate=*(CARD c,BYTE r)
[$20 **Position** $6A9$4C **GetD**]

PROC Plot=*(CARD c,BYTE r)
[$20 **Pos1** $6A9$AE$2FD$4C **PutD**]

PROC SetColor=*(BYTE reg,hue,lum)
[$5C9$1610$A085$98$F29$A285$8A$A0A$A0A$A205$A0A6$9D$2C4$9D$D016$60]

PROC Fill=*(CARD c,BYTE r)
[$20 **GrIO** $12A0$4C **XIO**]

BYTE FUNC Rand=*(BYTE r)
[AED20A$C9$0$9F0$8486$A2$0$8586$20 **MultI** $A086$60]

PROC Sound=*(BYTE v, p, d, vol)
[AA284$A8$7C9$530$64A0$20 **Error** $998A$D200$A2A5$A0A$A0A$A305$99$D201$60]

PROC SndRst=*()
[AD232$EF29$8D$232$8D$D20F$A9$0$8A2$9D$D200CAFA10$60]

BYTE FUNC Paddle=*(BYTE p)
[$BDAA$270$A085$60]

BYTE FUNC PTrig=*(BYTE p)
[$A2$0$4C9$330$E8$329$A8$BD$D300$39*+5$A085$60$804$8040]

BYTE FUNC Stick=*(BYTE p)
[$A2$0$2C9$330$E8$129$BDA8$D300$88$4D0$4A4A$4A4A$F29$A085$60]

BYTE FUNC STrig=*(BYTE p)
[$BDAA$D010$A085$60]

BYTE FUNC Peek=*(CARD a)

CARD FUNC PeekC=*(CARD a)
[$A285$A386$A0$0$A2B1$A085$C8$A2B1$A185$60]

PROC **Poke**=*(CARD a,BYTE v)
[$A085$A186$A098$9100$60A0]

PROC PokeC=*(CARD a,v)
[$20 **Poke** $A5C8$91A3$60A0]

PROC Zero=*(BYTE POINTER a,CARD s)
[48A9$0$A485$68]

PROC SetBlock=*(BYTE POINTER a,CARD s,BYTE v)
[$A085$A186$A284$A0$0$A4A5$A3A6$10F0$A091$C8$FBD0$A1E6$A3C6$F5D0$3F0
$A091$C8$A2C4$F9D0$60]

PROC MoveBlock=*(BYTE POINTER d,s,CARD sz)
[$A085$A186$A284$A0$0$A5A5$16F0$A2B1$A091$C8$F9D0$A1E6$A3E6$A5C6$F1D0
$5F0$A2B1$A091$C8$A4C4$F7D0$60]

INT FUNC SCompare=*(STRING a,b)
[$A485$A586$A284$A0$0$A084$A184$A4B1$A2D1$3F0$20*+21$C9$0$1D0$60$A685$C8
$A4B1$A2D1$5D0$A6C4$F590$60$FFA2$A086$390$A2B1$E8$A186$60]

PROC **SCopy**=*(STRING d,s)
[$A085$A186$A284$A0$0$A2B1$A091$8F0$A8$A2B1$A091$88$F9D0$60]

PROC SCopyS=*(STRING d,s,BYTE b,e)
[$A085$A186$A284$A0$0$A2B1$A5C5$2B0$A585$A4C6$18$A2A5
$A465$A285$290$A3E6$38$A5A5$A4E5$2B0$A9$0$4C **SCopy** +10]

PROC SAssign=*(STRING d,s,BYTE b,e)
[$A085$A186$A284$A0$0$A2B1$DF0$A685$A4C6$38$A5A5$A4E5$2F0$1B0$AA60$A6C5
$890$18$A6A5$AA$A465$A585$A5A5$A0D1$390$A091$18$A0A5$A465$A085$290$A1E6
$4C8A **SCopy** +14]