02

Atari 2600: Workshop

The O.G. Game Console....well... one of them...

02

External

External Appearance

Original VCS

2600 - 'Vader'

Atari 2600 Jr.

2800 - same thing

Attack of the clones

Coleco Gemini

Funvision

Rinco

Dactar & Dactar II

03

Controls

Controls


paddle


ATARI


PRO-LINE
ATARI
SPACE AGE
JOYSTICK
UNIQUE AND FUN TO USE!
QUICK-ON-THE-TRIGGER ARCADE FUN!


ATARI
VIDEO COMPUTER SYSTEM
VIDEO TOUCH PAD
STAR RAIDERS
ATARI
CX21


Keyboard Controller
1 2 3
4 5 6
7 8 9
* 0 #

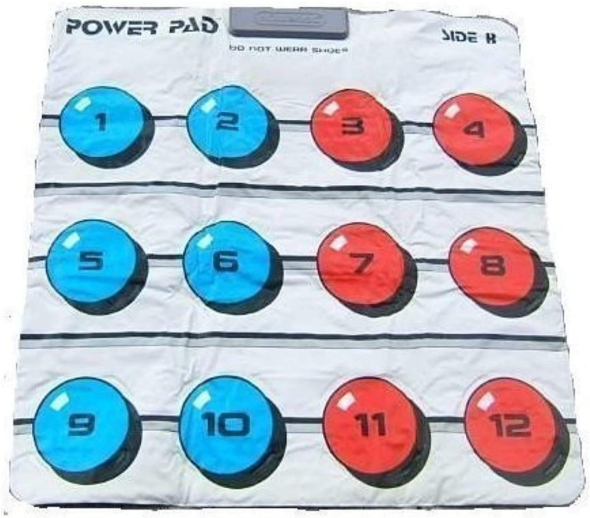
Controls

Joyboard (Amiga)

Foot Craz (Exus)

Controls

Influenced future controls like the Wii Fit Balance Board & the NES Power Pad



Controls


Turn Your ATARI 2600 into POWER ARCADE with
COSMIC COMMANDER
Control
ACTION-PACKED VIDEO GAME CARTRIDGE INCLUDED
Cosmic Commander
RETRO RAVEN


Turn Your ATARI 2600 into POWER ARCADE with
FLIGHT COMMANDER
Control

Controls


Cosmic Commander

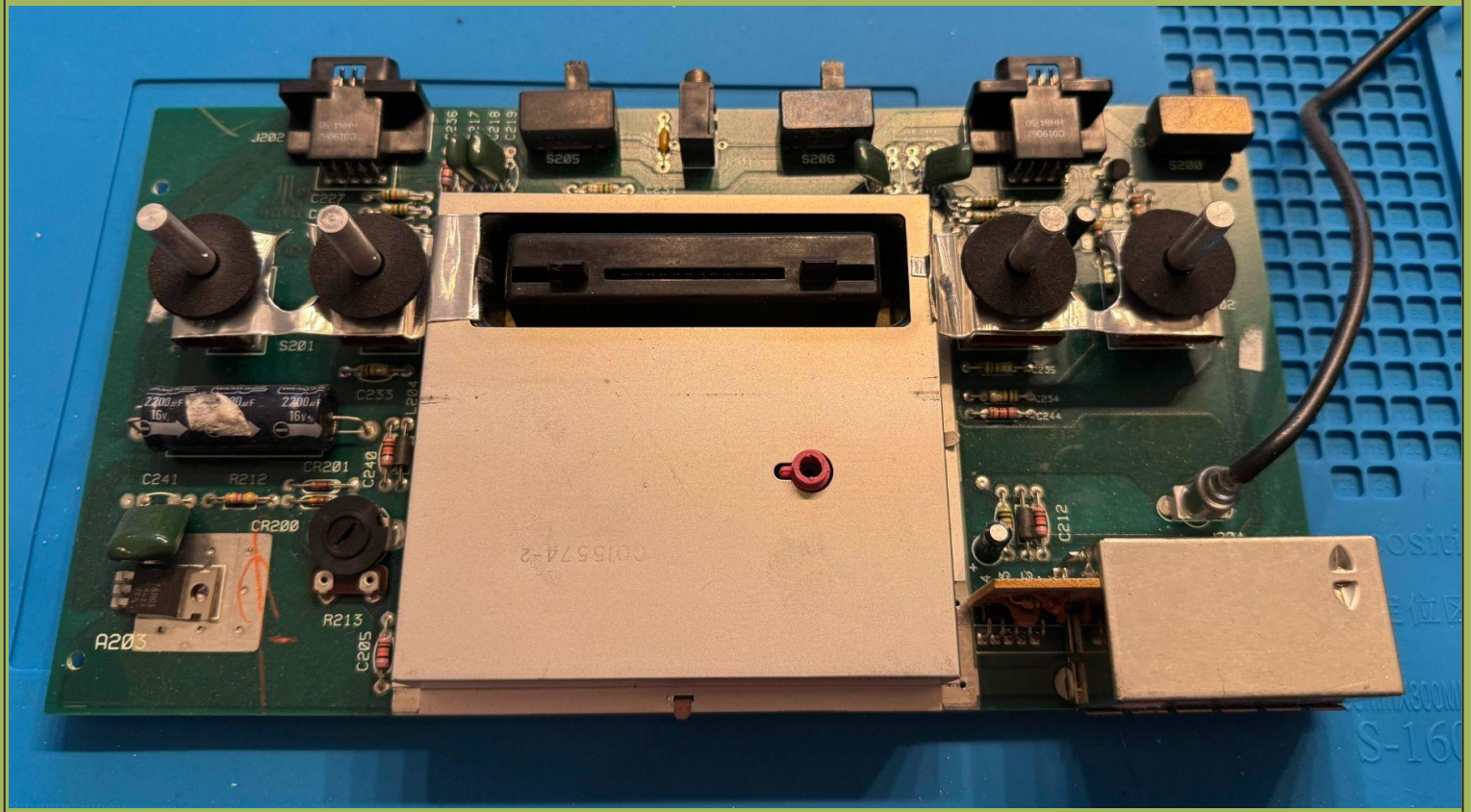

MILTON BRADLEY COMPANY
SPITFIRE ATTACK
SPITFIRE ATTACK
COMMANDER

Controls

04

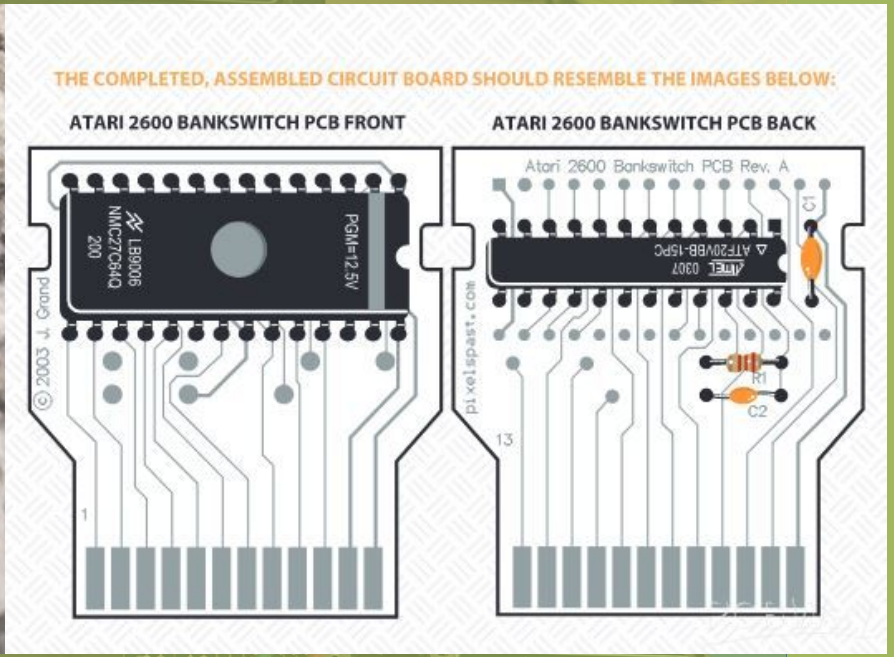
Internal

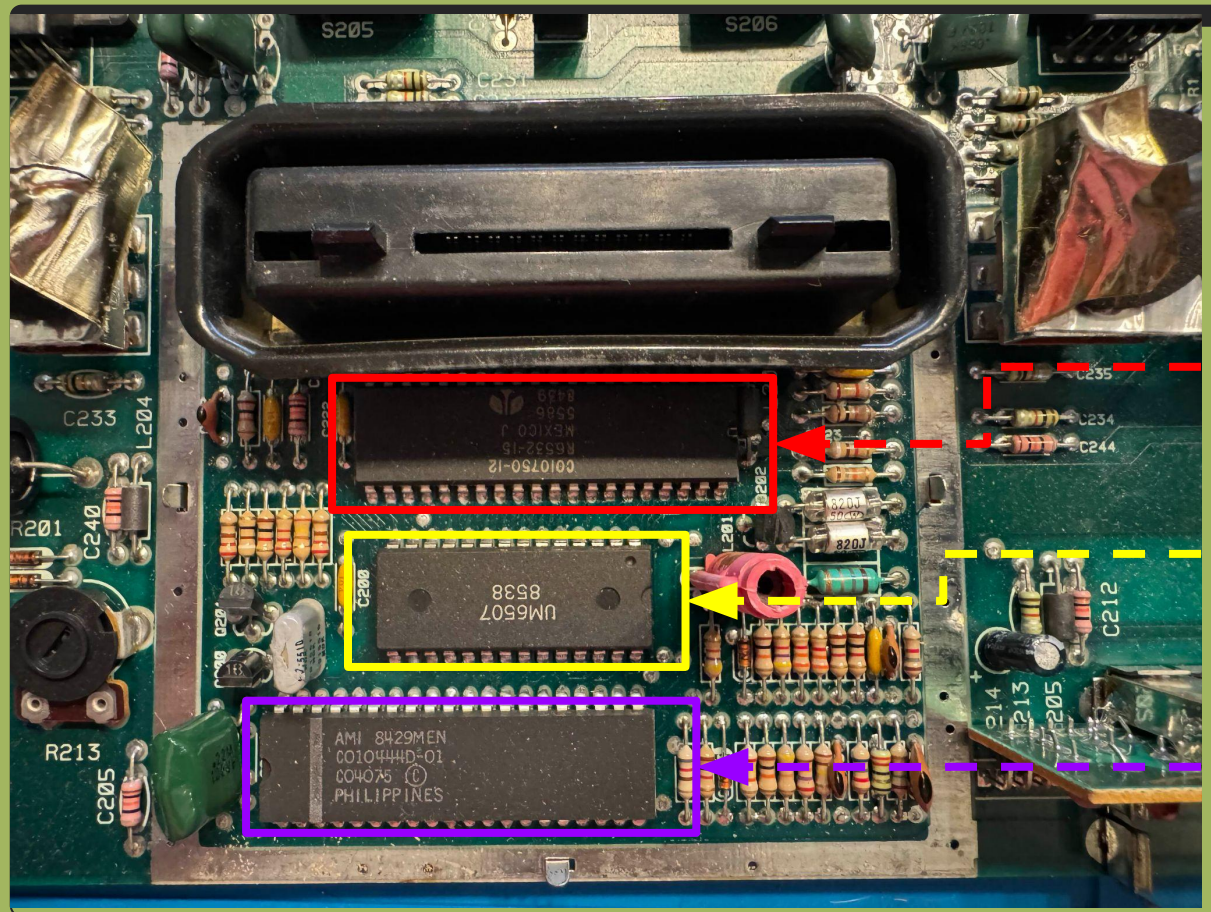

VIDEO COMPUTER SYSTEM™
POWER
ON
OFF
COLOR
B•W
GAME SELECT
GAME RESET
ATARI® 2600
14 PCS DISMANTLE


J202
S205
S206
S200
S201
C233
C241
R212
CR201
C240
CR200
R213
C205
A203
C212

04 - Internal


game program
ASTEROIDS
Use with Joystick Controllers
game program
STREET RACER
Use with Paddle Controllers
game program
PAC-MAN
Use with Joystick Controllers
game program
DEFENDER
Use with Joystick Controllers
game program
STAR RAIDERS
ACTIVISION
CHOPPER COMMAND
ACTIVISION
FREEWAY
game program
OUTLAW
Use with Joystick Controllers
game program
MISSILE COMMAND
Use with Joystick Controllers
game program
NIGHT DRIVER
Use with Paddle Controllers
game program
combat
CX-2601
1-5 TANK
6-9 TANK-PONG
10-14 INVISIBLE-TANK
15-20 BIPLANE
21-27 JET-FIGHTER
Use with Joystick Controllers
Q*bert
ACTIVISION
KABOOM!
ATARI 2600
CENTIPEDE
Use with Joystick Controller
game program
COMBAT
Use with Joystick Controllers
DONKEY KONG
Nintendo
ACTIVISION
SKY JINKS
game program
CIRCUS ATARI
Use with Paddle Controllers

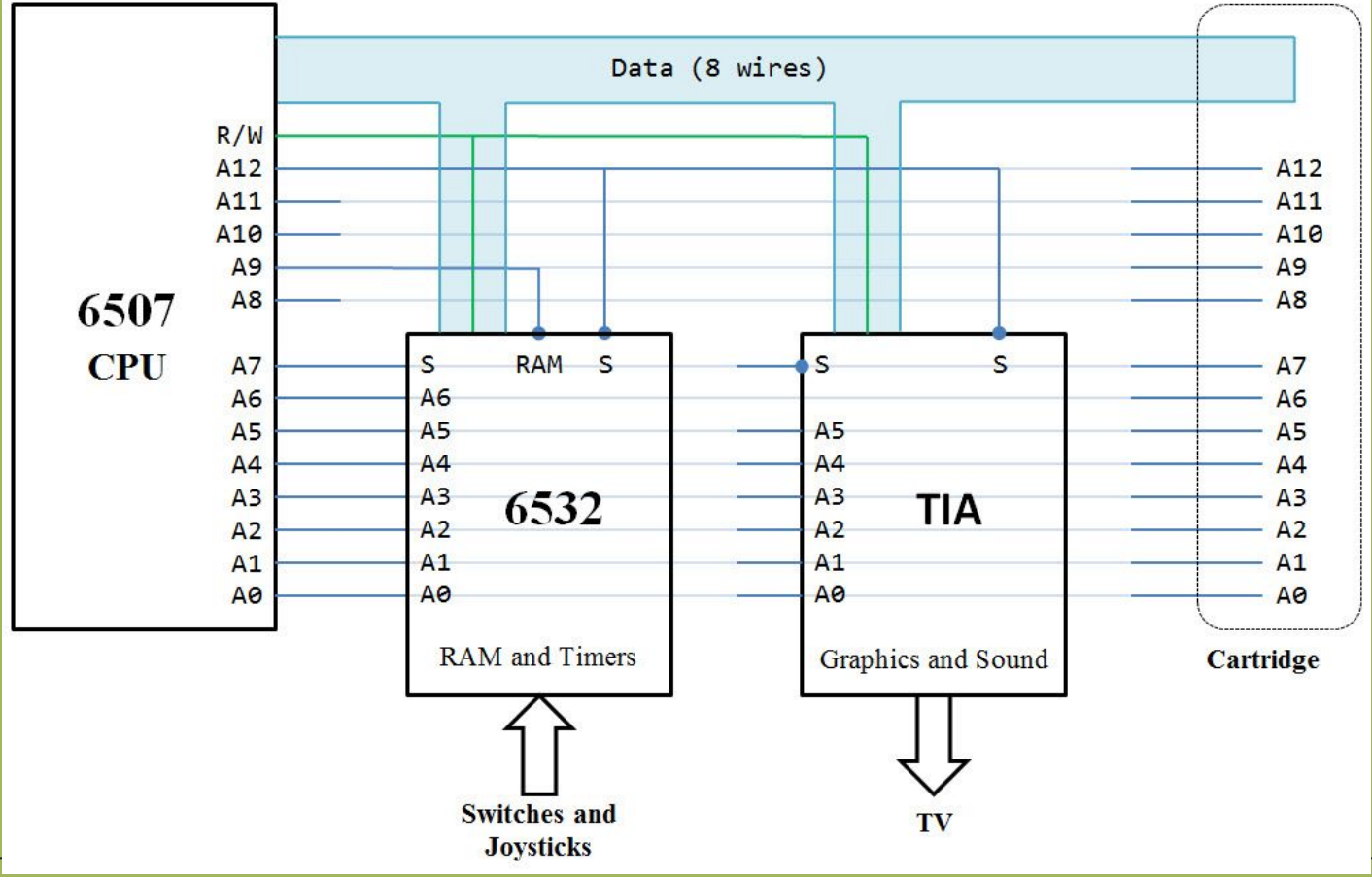

THE COMPLETED, ASSEMBLED CIRCUIT BOARD SHOULD RESEMBLE THE IMAGES BELOW:
ATARI 2600 BANKSWITCH PCB FRONT
ATARI 2600 BANKSWITCH PCB BACK
LB9006
NMC27C64Q
200
PGM=12.5V
© 2003 J. Grand
1
Atari 2600 Bankswitch PCB Rev. A
C1
R1
C2
pixelspast.com
13



RIOT
(6532)

CPU
6507
(6502
compatible)

TIA
Television
Interface
Adapter


Data (8 wires)
6507
CPU
R/W
A12
A11
A10
A9
A8
A7
A6
A5
A4
A3
A2
A1
A0
S
RAM
S
6532
RAM and Timers
S
S
TIA
Graphics and Sound
Cartridge
Switches and Joysticks
TV

05

Software Development

NO OPERATING SYSTEM

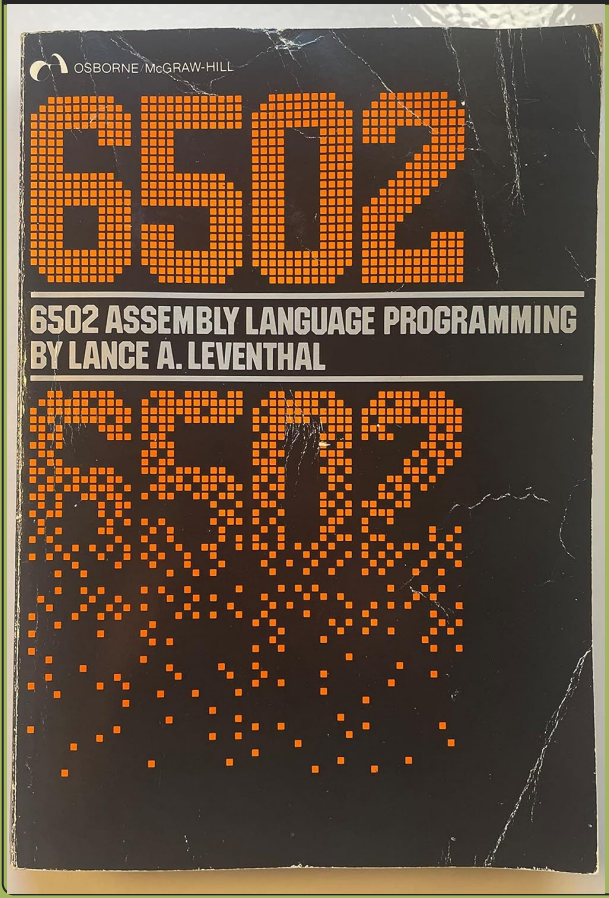
NO OPERATING SYSTEM

NO OPERATING SYSTEM

...

how do you dev?

...



https://skilldrick.github.io/easy6502/

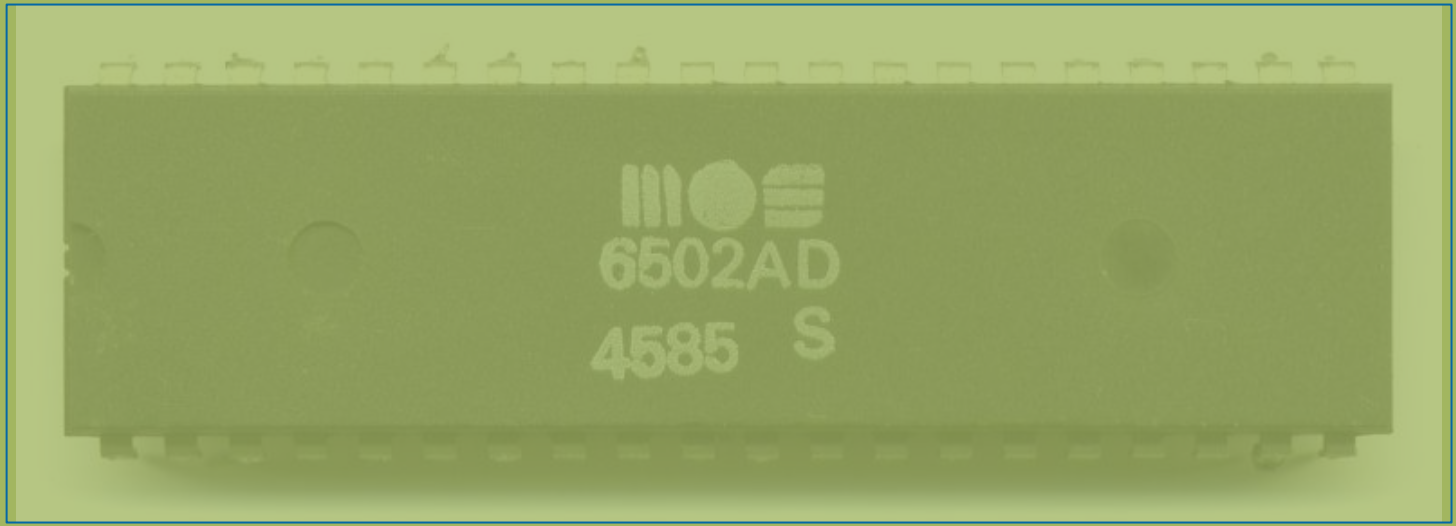
https://codeburst.io/an-introduction-to-6502-assembly-and-low-level-programming-7c11fa6b9cb9

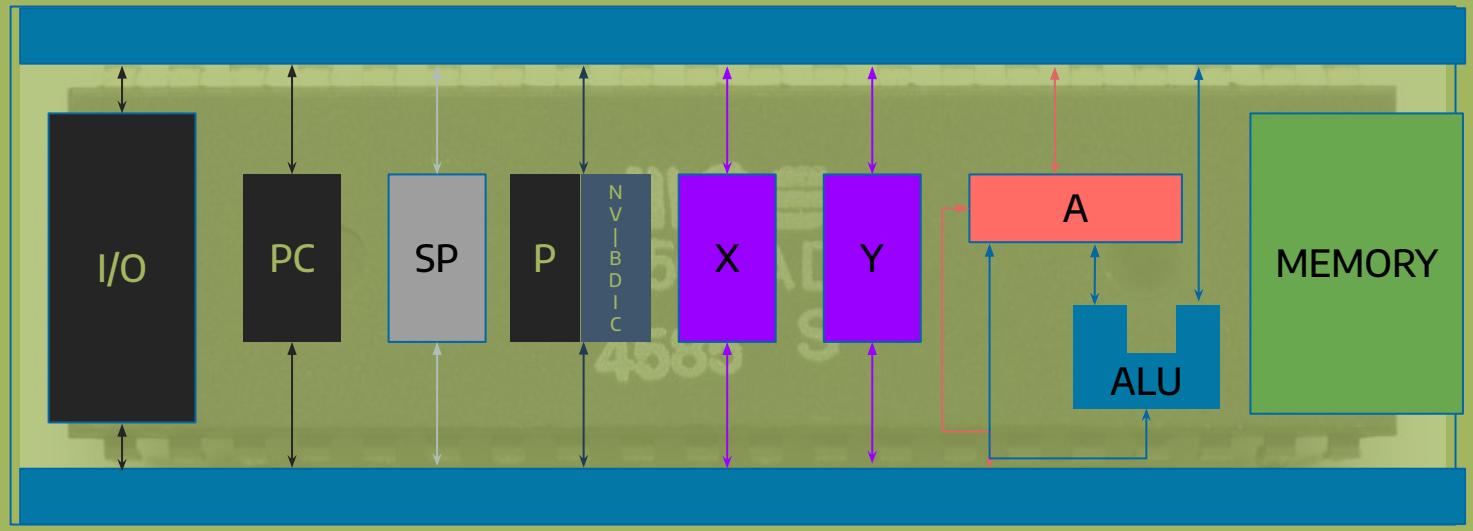
https://en.wikibooks.org/wiki/6502_Assembly

https://github.com/LukeMcCann/Atari_Assembly Projects

https://www.youtube.com/watch?v=PxZGoiWvA4A


MOS
6502AD
4585 S


6502AD
4585 S



PC - Program counter
which line of code is being executed right now

```
LDA #$01
STA $0200
LDA #$05
STA $0201
LDA #$08
STA $0202
```

https://skilldrick.github.io/easy6502/

```
LDA #$01
STA $0200
LDA #$05
STA $0201
LDA #$08
STA $0202
```

LOAD Hex value 01 into the 'A' register

LOAD Hex value 05 into the 'A' register

LOAD Hex value 08 into the 'A' register

https://skilldrick.github.io/easy6502/

```
LDA #$01
STA $0200
LDA #$05
STA $0201
LDA #$08
STA $0202
```

STORE the value in the 'A' register at memory location 0200

STORE the value in the 'A' register at memory location 0201

STORE the value in the 'A' register at memory location 0202

https://skilldrick.github.io/easy6502/

$0201 $0202

$0200

```
LDA #$01
STA $0200
LDA #$05
STA $0201
LDA #$08
STA $0202
```

Memory Locations: $0200 to $05ff

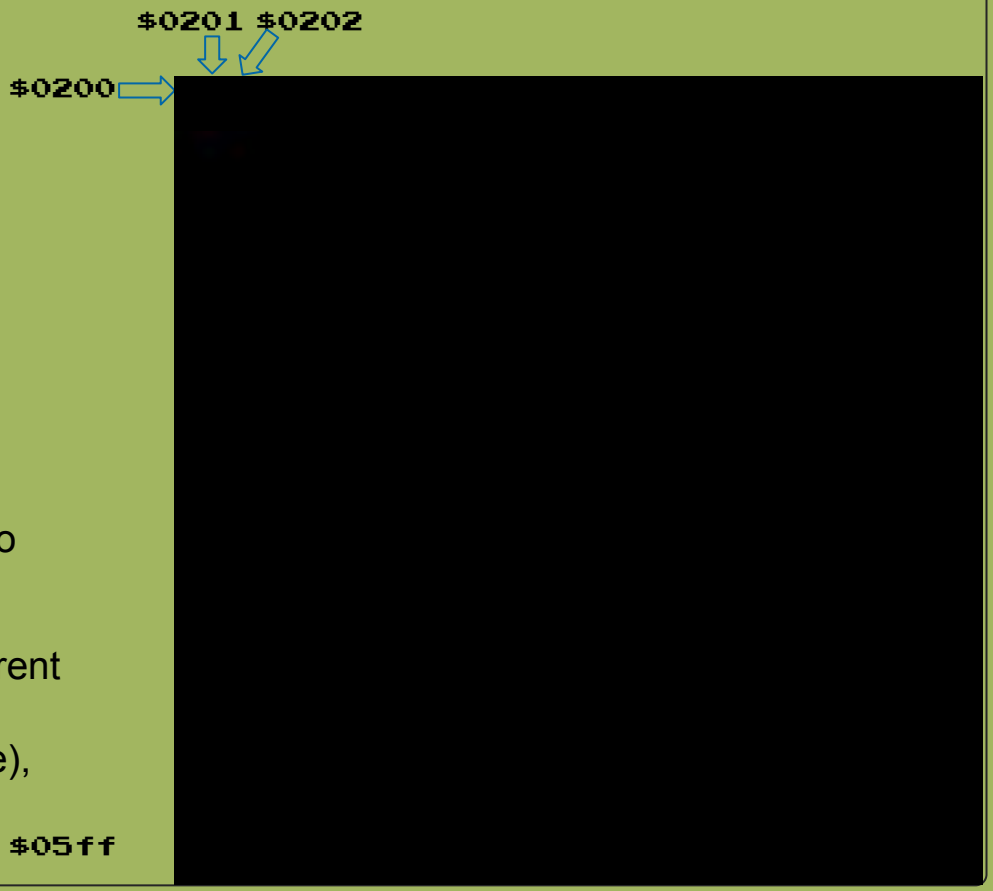
$00 to $0f represent 16 different colours
($00 is black and $01 is white),

$05ff

```
LDA #$01
STA $0200
LDA #$05
STA $0201
LDA #$08
STA $0202
```

Memory Locations: $0200 to $05ff

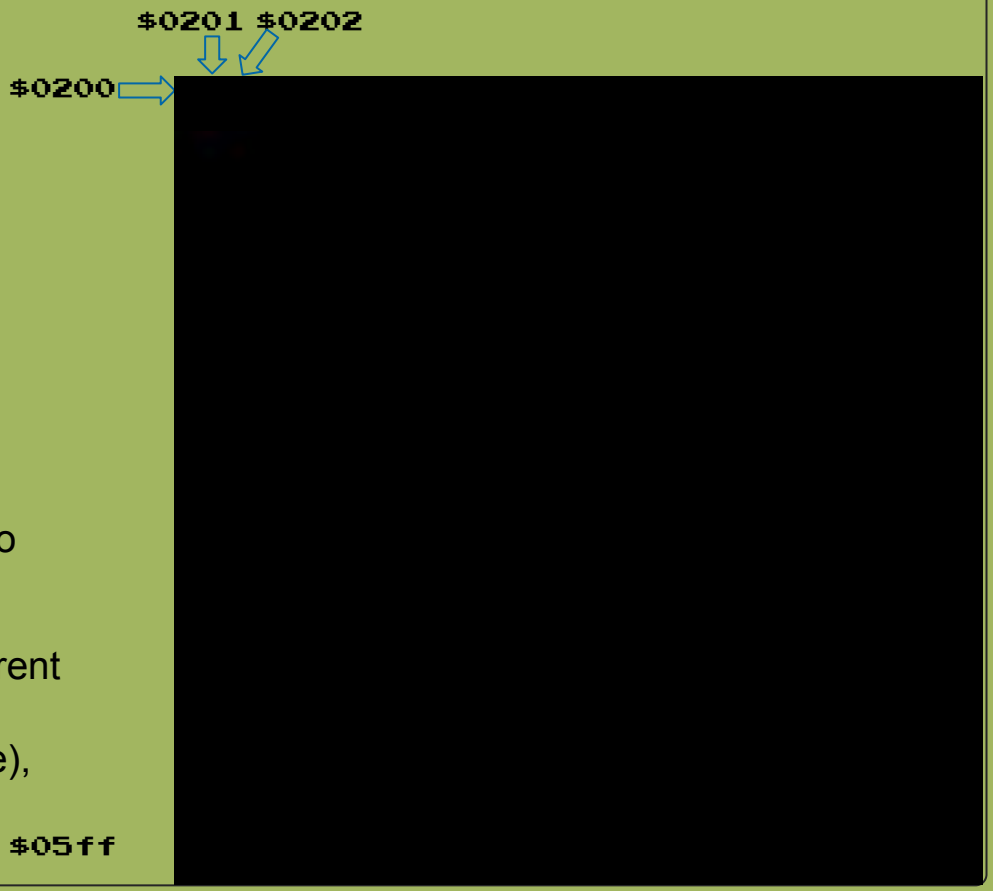
$00 to $0f represent 16 different colours
($00 is black and $01 is white),



```
LDA #$01
STA $0200
LDA #$05
STA $0201
LDA #$08
STA $0202
```

Memory Locations: $0200 to $05ff

$00 to $0f represent 16 different colours
($00 is black and $01 is white),



$0201 $0202

$0200

```
LDA #$01
STA $0200
LDA #$05
STA $0201
LDA #$08
STA $0202
```

Memory Locations: $0200 to $05ff

$00 to $0f represent 16 different colours
($00 is black and $01 is white),

$05ff

$0201 $0202

$0200

```
LDA #$01
STA $0200
LDA #$05
STA $0201
LDA #$08
STA $0202
```

Memory Locations: $0200 to $05ff

$00 to $0f represent 16 different colours
($00 is black and $01 is white),

$05ff

$0201 $0202

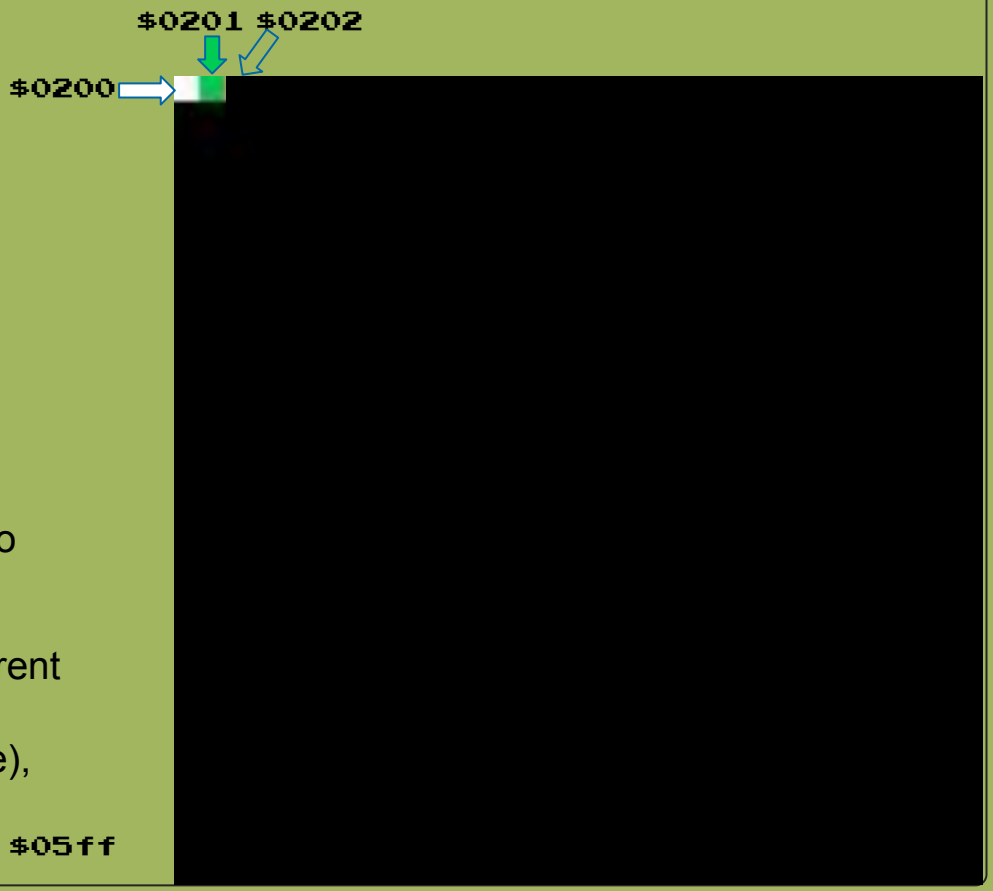
$0200

```
LDA #$01
STA $0200
LDA #$05
STA $0201
LDA #$08
STA $0202
```

Memory Locations: $0200 to $05ff

$00 to $0f represent 16 different colours
($00 is black and $01 is white),

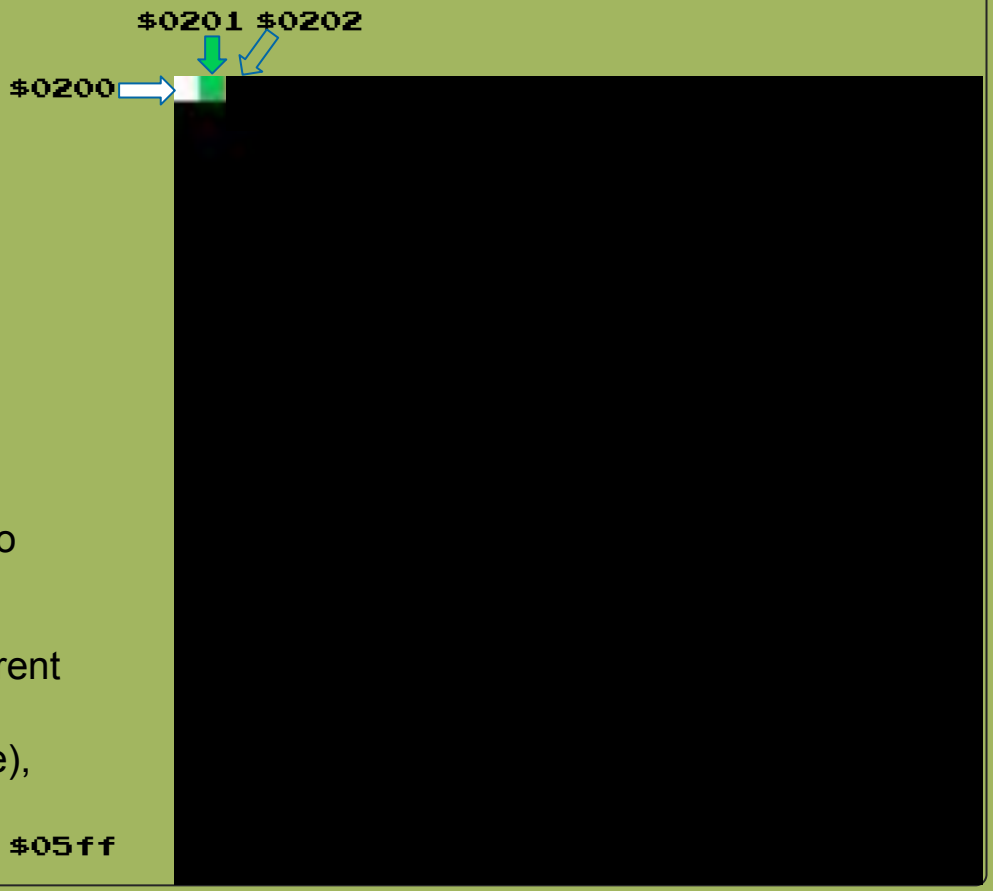
$05ff




```
LDA #$01
STA $0200
LDA #$05
STA $0201
LDA #$08
STA $0202
```

Memory Locations: $0200 to $05ff

$00 to $0f represent 16 different colours
($00 is black and $01 is white),




```
LDA #$01
STA $0200
LDA #$05
STA $0201
LDA #$08
STA $0202
```

Memory Locations: $0200 to $05ff

$00 to $0f represent 16 different colours
($00 is black and $01 is white),


```
LDA #$01
STA $0200
LDA #$05
STA $0201
LDA #$08
STA $0202
```

Memory Locations: $0200 to $05ff

$00 to $0f represent 16 different colours
($00 is black and $01 is white),


```
LDA #$01
STA $0200
LDA #$05
STA $0201
LDA #$08
STA $0202
```

Memory Locations: $0200 to $05ff

$00 to $0f represent 16 different colours
($00 is black and $01 is white),

WHY?

No...
really..
WHY?

6502 was used in lots of systems


Acorn Atom


Acorn Electron


Apple I


Apple II


Apple IIe


Atari 2600


Atari 5200


Atari 7800


Atari 800


Atari Lynx


BBC Master


Baby! 1


BBC Micro


Commodore PET


Commodore VIC-20

6502 was used in lots of systems

Commodore 64

Commodore 128

Family Computer (Famicom)

Nintendo Entertainment System

Ohio Scientific Challenger 4P

Orao

Oric-1

Oric Atmos

TurboGrafx-16

6502 was used in lots of systems

Family Computer (Famicom)

Nintendo Entertainment System

Ohio S
Challe

6502 was used in lots of systems

06

BYO-Atari 2600 Game

Assembly is *simple* but <u>confusing</u>!

Luckily...
We have a saviour

Stella

A multi-platform Atari 2600 VCS emulator

https://stella-emu.github.io/

batari Basic

https://github.com/batari-Basic/batari-Basic

https://www.randomterrain.com/atari-2600-memories-batari-basic-commands.html

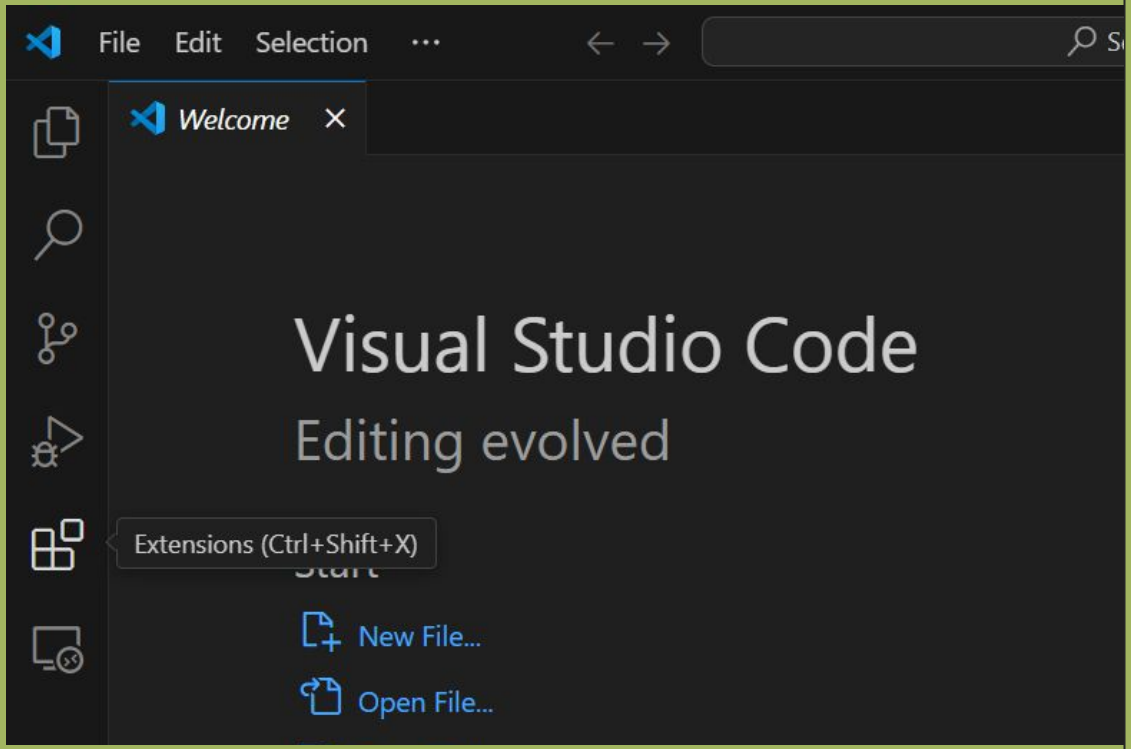
https://alienbill.com/2600/basic/home.html

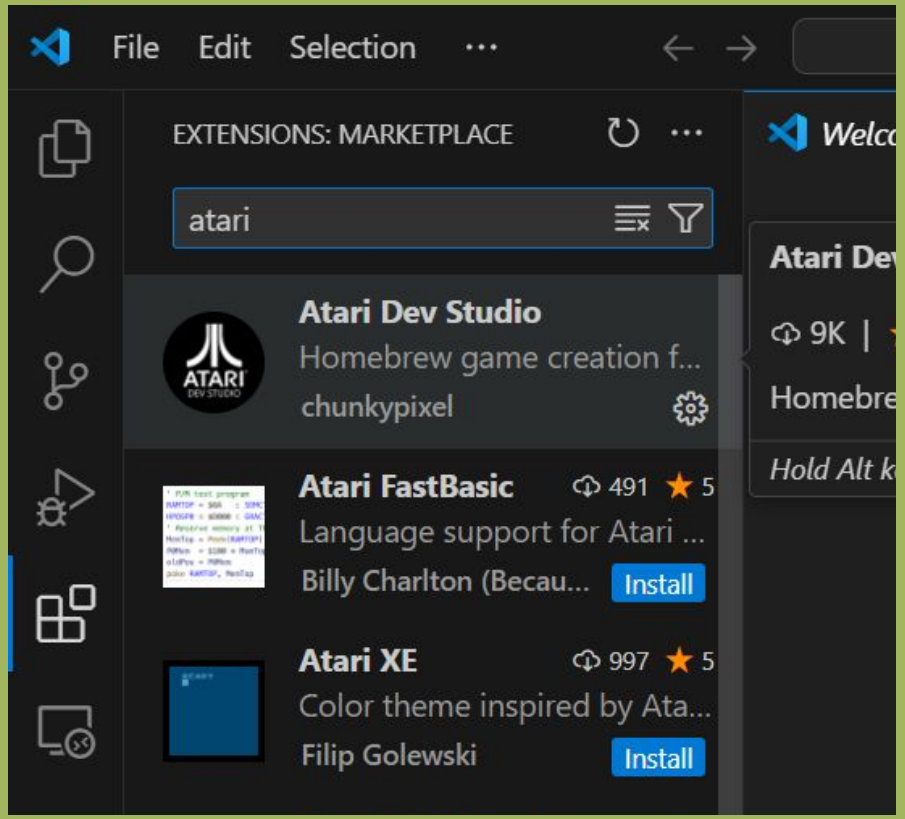
Even more lucky...

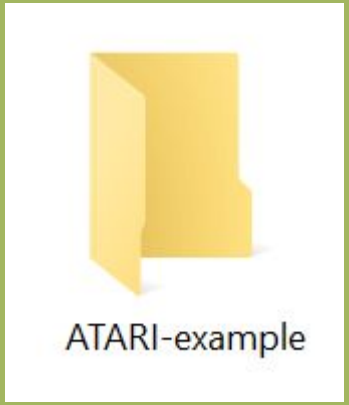
We have a Development Environment available

Install if you don't have it

https://code.visualstudio.com/


File
Edit
Selection
Welcome
Visual Studio Code
Editing evolved
Extensions (Ctrl+Shift+X)
New File...
Open File...


File
Edit
Selection
EXTENSIONS: MARKETPLACE
atari
Atari Dev Studio
Homebrew game creation f...
chunkypixel
Atari FastBasic
491
5
Language support for Atari ...
Billy Charlton (Becau...
Install
Atari XE
997
5
Color theme inspired by Ata...
Filip Golewski
Install
9K
Homebre
Hold Alt k

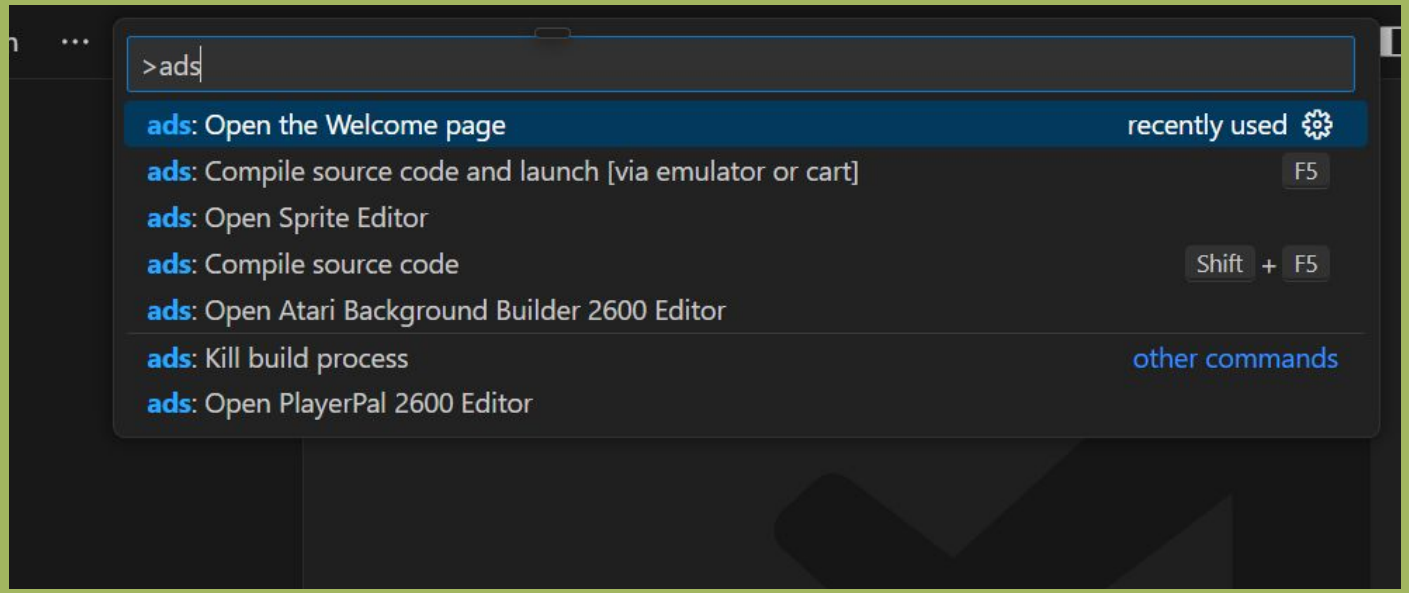


Create a new empty folder somewhere

And open the folder in VS-Code

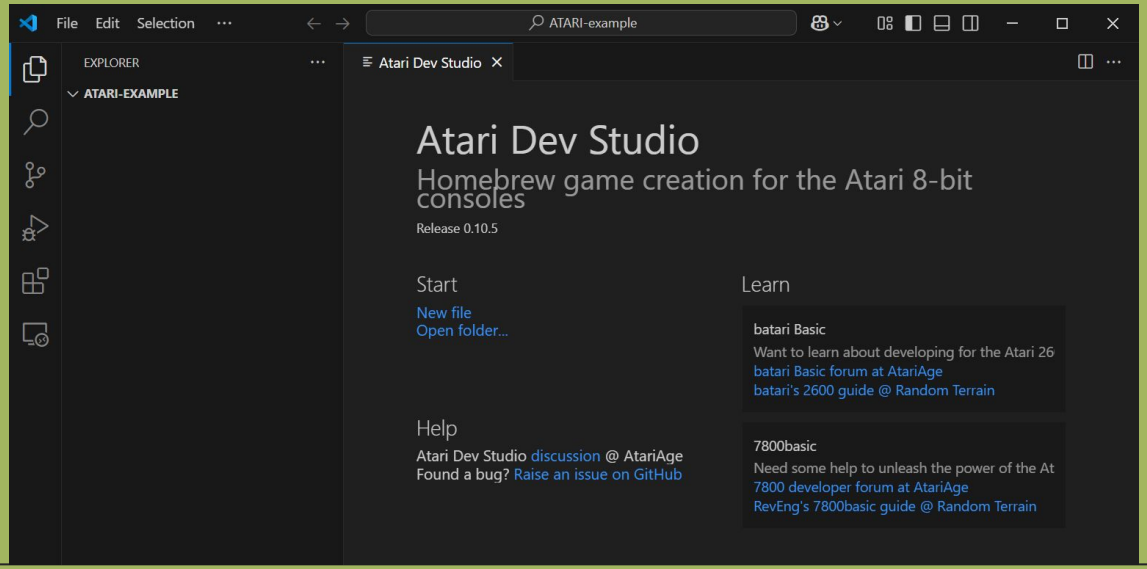
Ctrl-Shift-P

Look for 'ads' select Open the Welcome Page

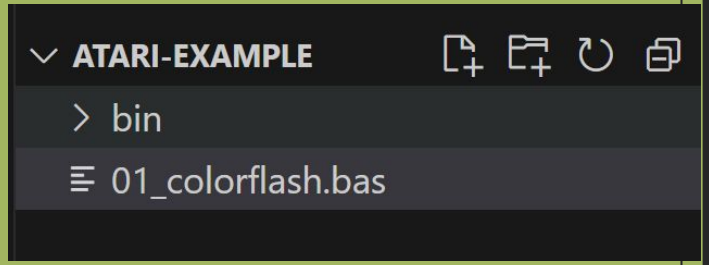


Ctrl-Shift-P

Look for 'ads' select Open the Welcome Page



Create a new file ending with ".bas"



Write this code -->

Compile & Run (Press F5)

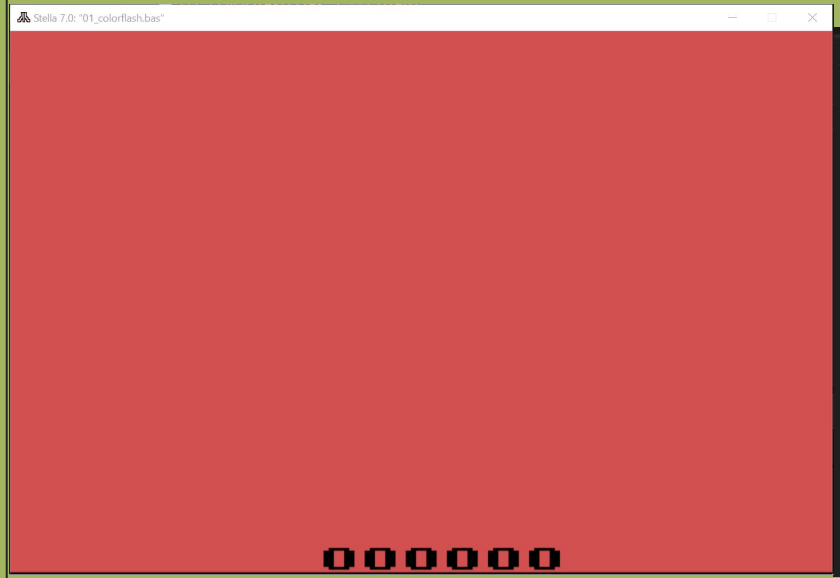
01_colorflash.bas > ...

```
main
    COLUBK=a
    a = a+1
    drawscreen
    goto main
```

Create a new file ending with ".bas"

ATARI-EXAMPLE

- bin
- 01_colorflash.bas



01_colorflash.bas > ...

```
main
    COLUBK=a
    a = a+1
    drawscreen
    goto main
```

Colors!

Interactive tool:

https://www.randomterrain.com/atari-2600-memories-tia-color-charts.html#ntsc_pal_color_conversion

NTSC (128 unique colors)

Color Value: $00

	0	2	4	6	8	A	C	E
0								
1								
2								
3								
4								
5								
6								
7								
8								
9								
A								
B								
C								
D								
E								
F								

PAL (104 unique colors)

Color Value: $00

	0	2	4	6	8	A	C	E
0								
1								
2								
3								
4								
5								
6								
7								
8								
9								
A								
B								
C								
D								
E								
F								

Let's make a sprite!

Set Color before Drawing

(like a paint program, select color then draw)

Atari Sprites max: 8x192 px
Pixels are binary (on/off)

2 Player Sprites only
player0 and player1

```
player0:
    %00000000
    %00000000
    %01100110
    %01000010
end
```

0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	1	1	0	0	1	1	0
0	1	0	0	0	0	1	0

```
main
    COLUBK=$02
    COLUP0=$28

    player0x=50
    player0y=80
    drawscreen
    goto main
```

Set BG colour

```
player0:
    %00000000
    %00000000
    %01100110
    %01000010
    %01100010
    %00100100
    %00100100
    %00100100
    %01011010
    %01111110
    %01100110
    %01100110
    %01100110
    %01100110
    %01111110
    %00011000
end
```

Create your own sprite!

```
main
    COLUBK=$02
    COLUP0=$28

    player0x=50
    player0y=80
    drawscreen
    goto main
```

Set Player0 colour

```
player0:
    %00000000
    %00000000
    %01100110
    %01000010
    %01100010
    %00100100
    %00100100
    %00100100
    %01011010
    %01111110
    %01100110
    %01100110
    %01100110
    %01100110
    %01111110
    %00011000
end
```

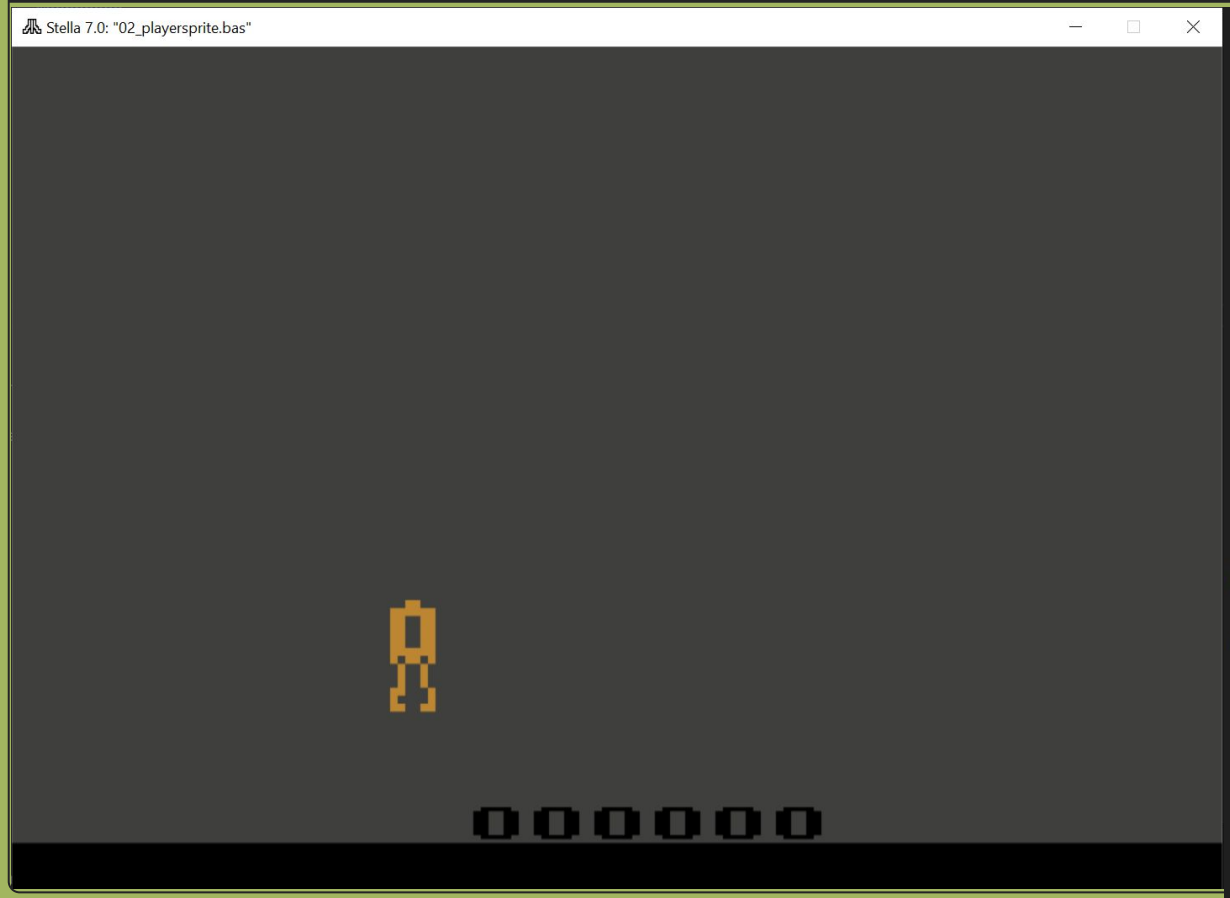
```
main
    COLUBK=$02
    COLUP0=$28

    player0x=50
    player0y=80
    drawscreen
    goto main
```

Set Player0 x-pos

Set Player0 y-pos

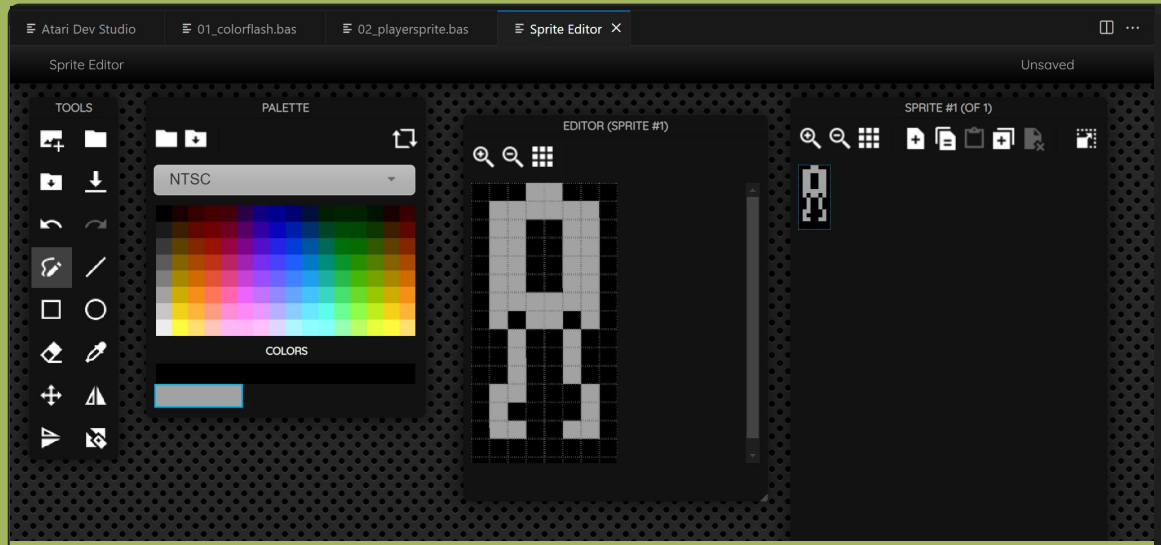
```
player0:
    %00000000
    %00000000
    %01100110
    %01000010
    %01100010
    %00100100
    %00100100
    %00100100
    %01011010
    %01111110
    %01100110
    %01100110
    %01100110
    %01100110
    %01111110
    %00011000
end
```




```
player0:
    %00000000
    %00000000
    %01100110
    %01000010
    %01100010
    %00100100
    %00100100
    %00100100
    %01011010
    %01111110
    %01100110
    %01100110
    %01100110
    %01100110
    %01111110
    %00011000
end

main
    COLUBK=$02
    COLUP0=$28

    player0x=50
    player0y=80
    drawscreen
    goto main
```




```
player0:
    %00000000
    %00000000
    %01100110
    %01000010
    %01100010
    %00100100
    %00100100
    %00100100
    %01011010
    %01111110
    %01100110
    %01100110
    %01100110
    %01100110
    %01111110
    %00011000
end
```

Sprite Editor

EXPORT

Export 2600 sprites

Select the source language format required then copy or export to file the generated code.

- (●) Selected sprite

  Export the selected sprite currently chosen in the Editor window.

- ( ) All sprites

  Export all sprites in your project into a single package.

batari Basic

```
rem Created using Atari Dev Studio
rem batari Basic format (bottom to top)

sprite1:
%00000000
%00000000
%01100110
%01000010
%01100010
%00100100
%00100100
%00100100
%01011010
%01111110
%01100110
%01100110
%01100110
```

Copy to clipbaord

ACTION

ATARI DEV STUDIO

ACTIONS:

Create your own Sprite

- what happens if you make too many columns?
- How big can you make a sprite?
- Can you find the origin of the (x,y) coordinates of the player?
  - Test by placing it at various locations