

# Retrocomputing with RISC-V as a Learning Plataform

---

Jecel Assumpção Jr  
Mario Gazziro

June 14, 2024

# Motivation

Robotics



Videogames



IoT



AI



**Students  
trying  
our  
console  
here,  
in  
RISC-V  
Eldorado**

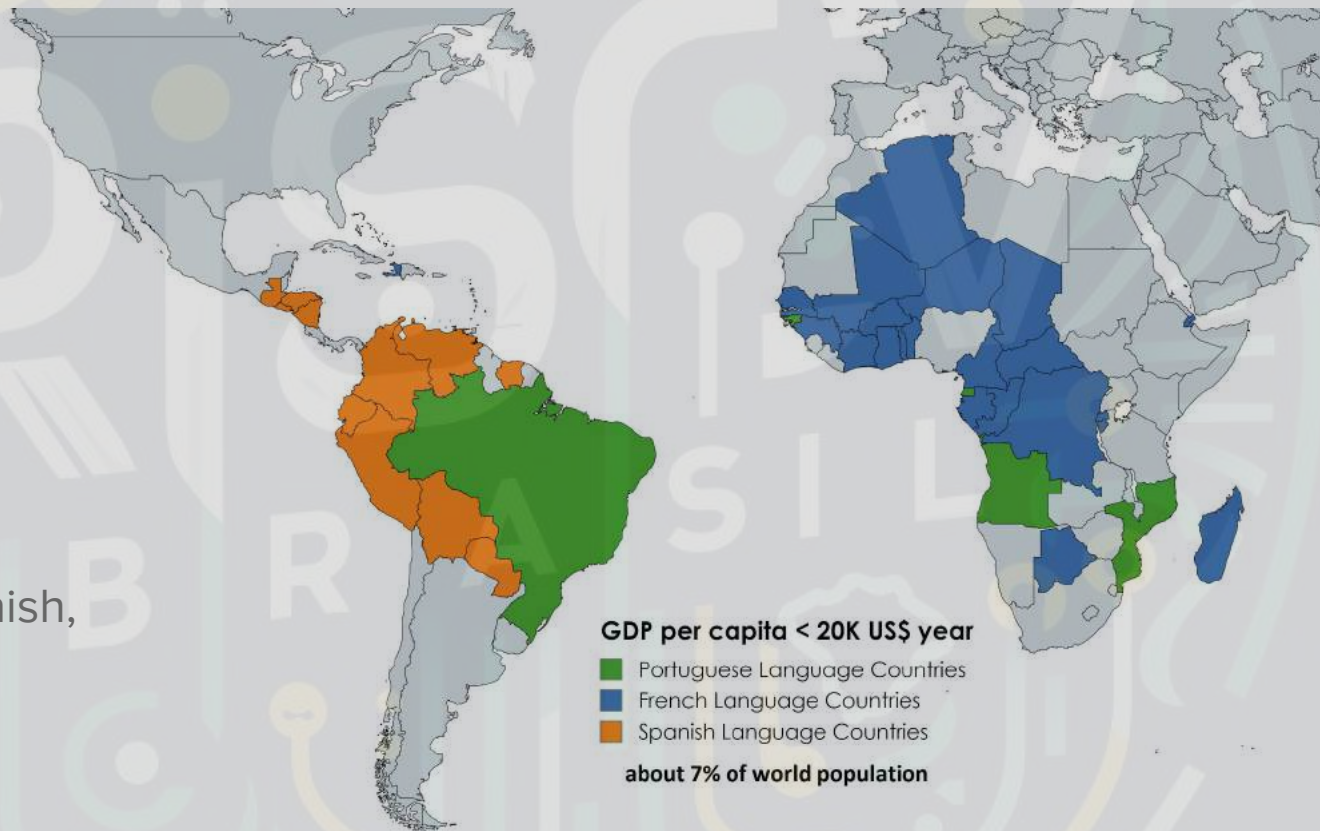




# Target Audience



Portuguese, Spanish,  
French, English



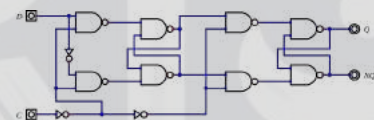
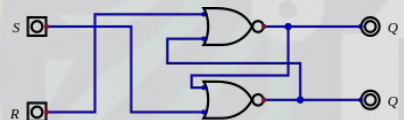
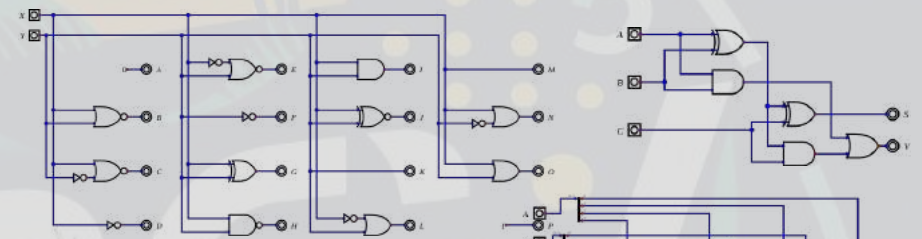
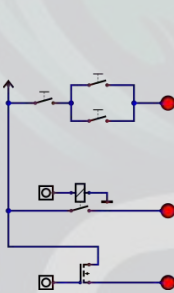
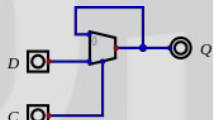
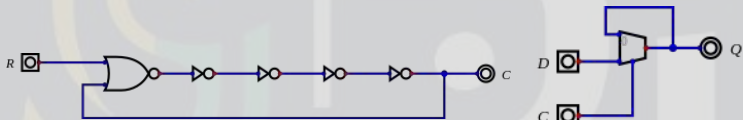
**GDP per capita < 20K US\$ year**

- Portuguese Language Countries
- French Language Countries
- Spanish Language Countries

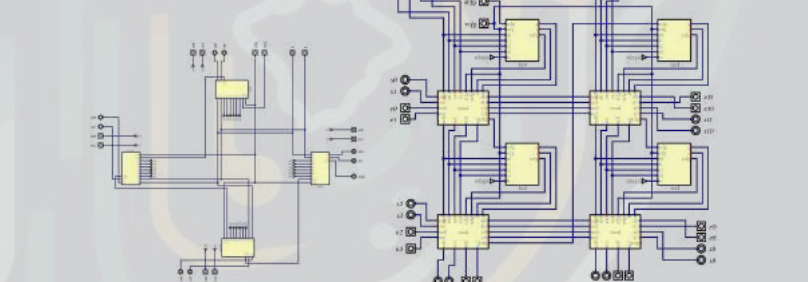
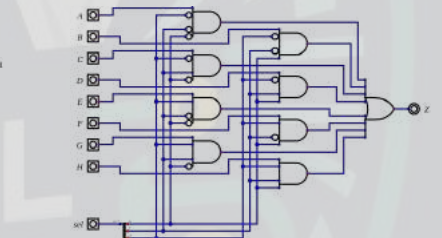
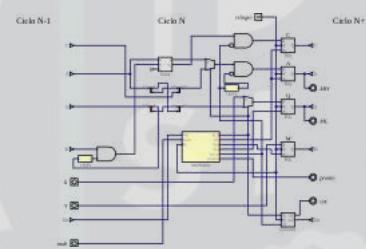
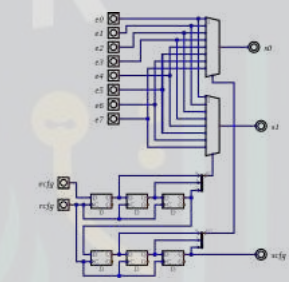
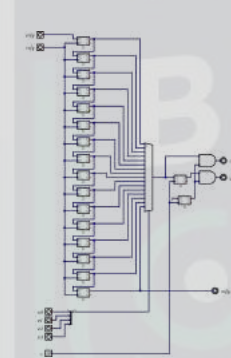
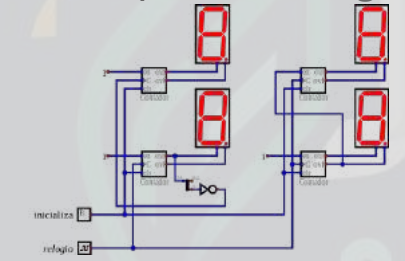
**about 7% of world population**

# Building Blocks

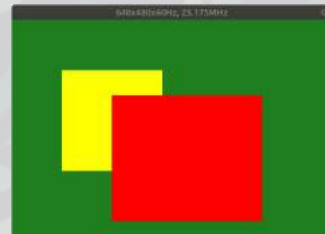
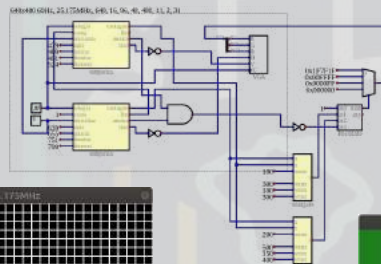
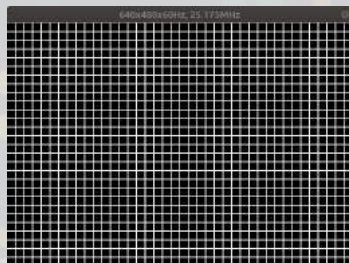
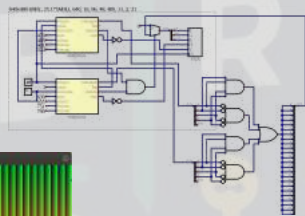
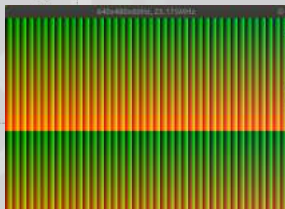
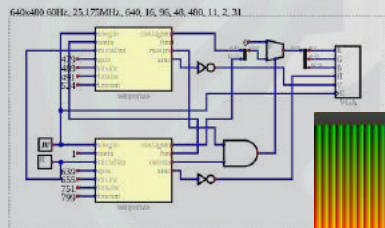
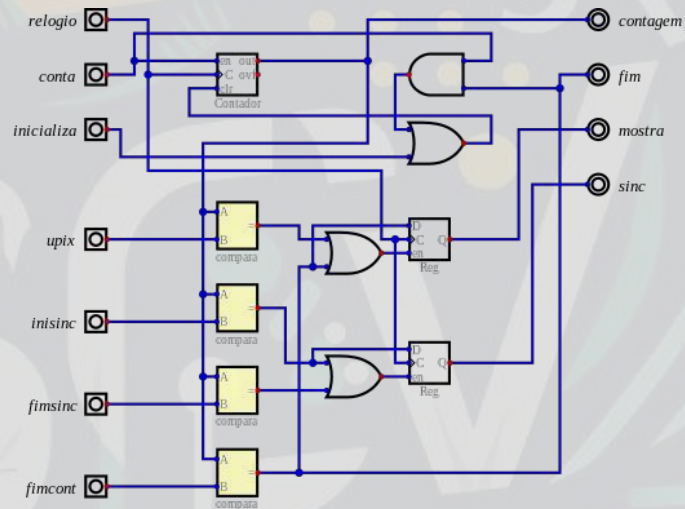
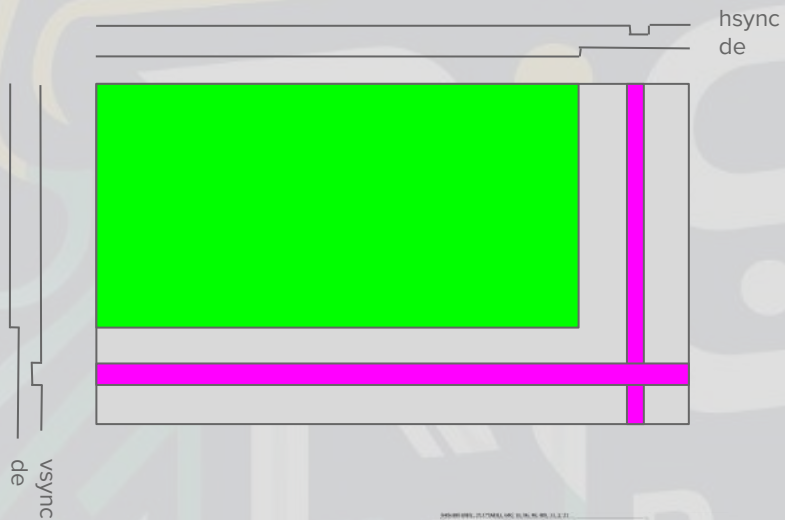
## Combinational Logic Circuits



## Sequential Logic Circuits

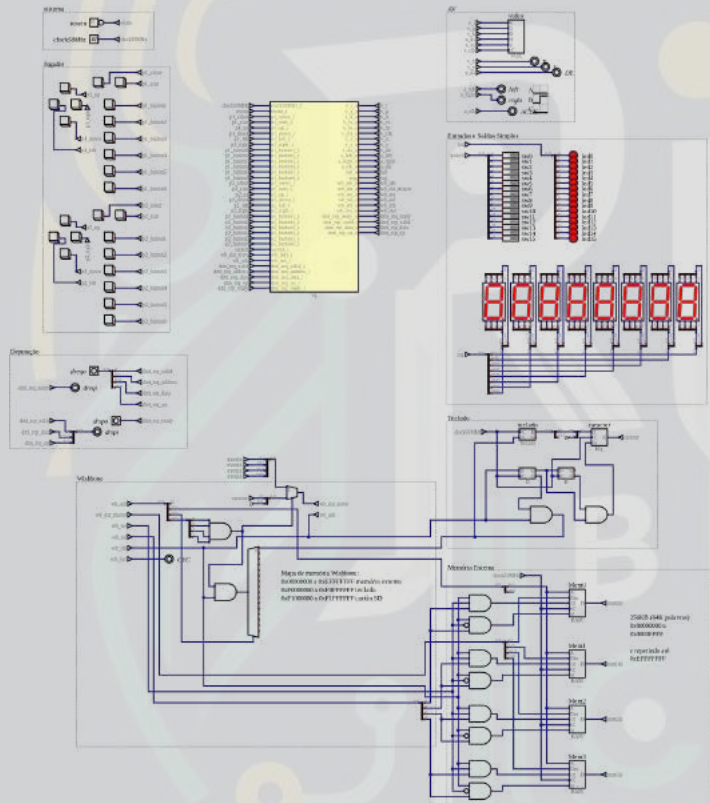


# Video

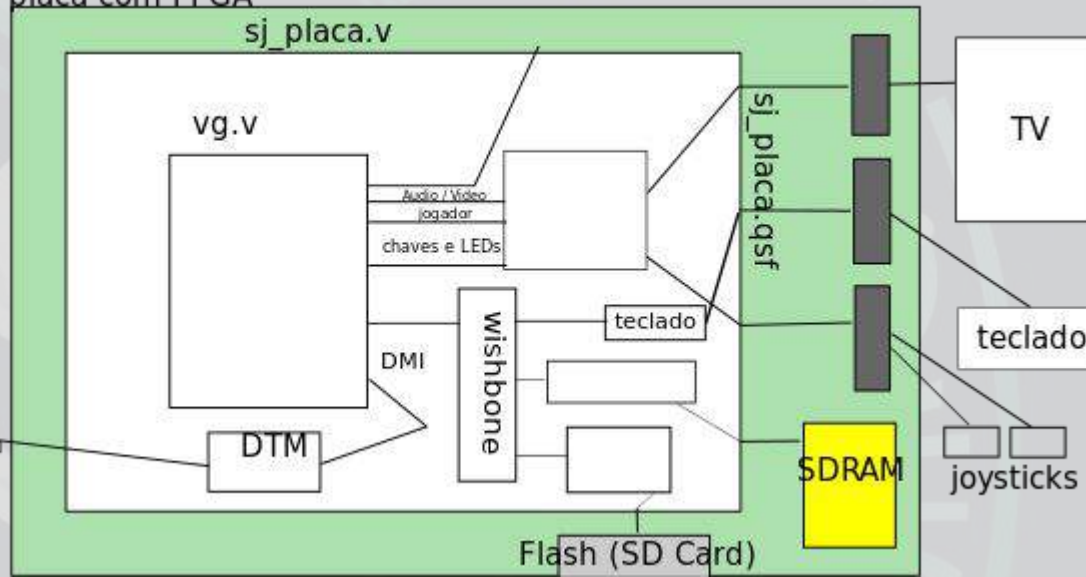




# Shin JAMMA



placa com FPGA



Speed Tang Nano 20K



DIGILENT

Arty A7-35T



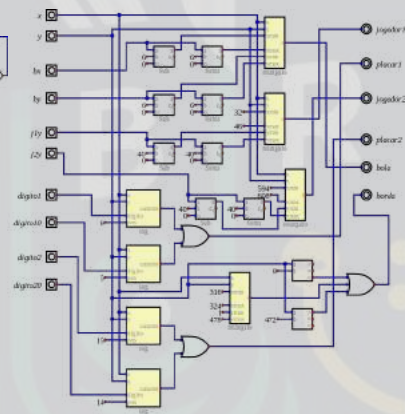
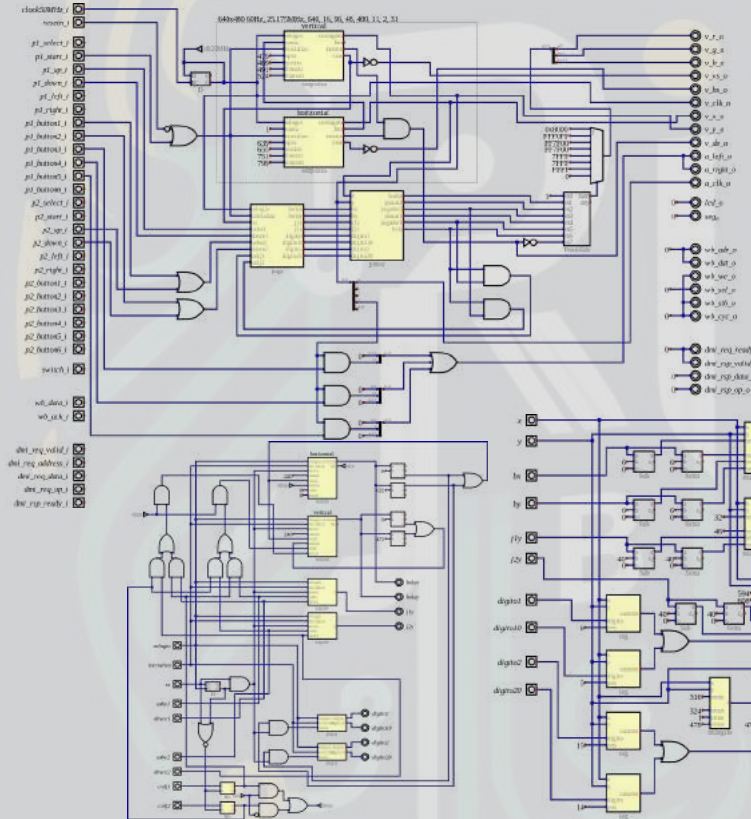
Terasic DE0



Terasic DE0-Nano



# First Generation (Retro)





# First Generation (Vintage)

Manavox 1972



Atari Home Pong 1975



Nintendo TV-Game 1977



Coleco Telstar 1976



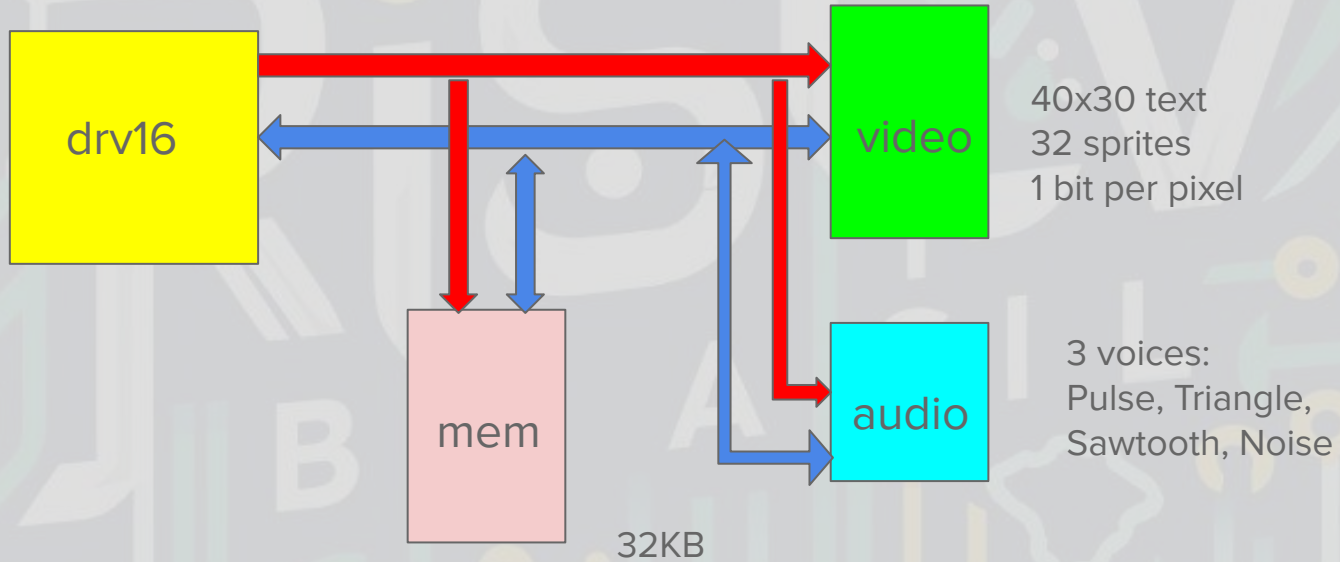
Philco-Ford Telejogo (1977)



Philco-Ford Telejogo II (1979)



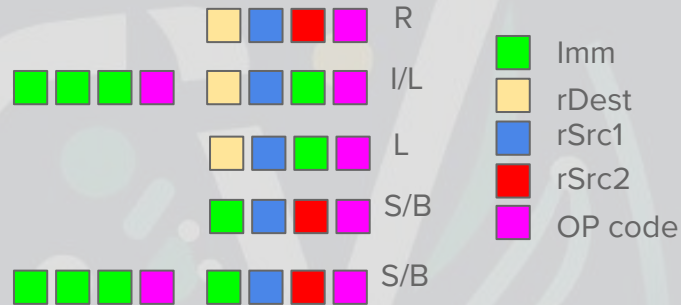
# Second Generation (Retro)



# drv16 and drv32I processors



state



|      | xx00       | xx01        | xx10        | xx11        |
|------|------------|-------------|-------------|-------------|
| 00xx |            | AND<br>ANDI | OR<br>ORI   | XOR<br>XORI |
| 01xx | <b>JAL</b> | ADD<br>ADDI | SUB<br>SUBI | SLT<br>SLTI |
| 10xx | JALR       | LH          | LB          | LBU         |
| 11xx | <b>SH</b>  | <b>SB</b>   | BEQ<br>BNE  | BLT<br>BGE  |

OP codes



# Second Generation (Vintage)

Fairchild  
Channel F 1976  
F8 1.79MHz 2KB



Atari  
2600 1977 (1983)  
6507 1.19MHz 128



Bally  
Astrocade 1978  
Z80 1.79MHz 4KB



Philips/Magnavox  
Odyssey<sup>2</sup> 1978 (1983)  
8048 1.79MHz 192



Mattel  
Intellivision 1980  
(1984)  
CP1610 2MHz 1.4KB



Emerson  
Arcadia 1982  
2650 3.58MHz 512



Coleco  
ColecoVision 1982  
Z80 3.58MHz 17KB



GCE/Milton Bradley  
Vectrex 1982  
6809 1.5MHz 1KB



Sharp  
HotBit (1985)  
Z80 3.58MHz 80KB



Gradiente  
Expert (1985)  
Z80 3.58MHz 80KB



# 8 Bit Workshop


8bitworkshop.com/v3.11.0/?file=chase%2Fgame.c&platform=nes#

Shirur's Chase Game

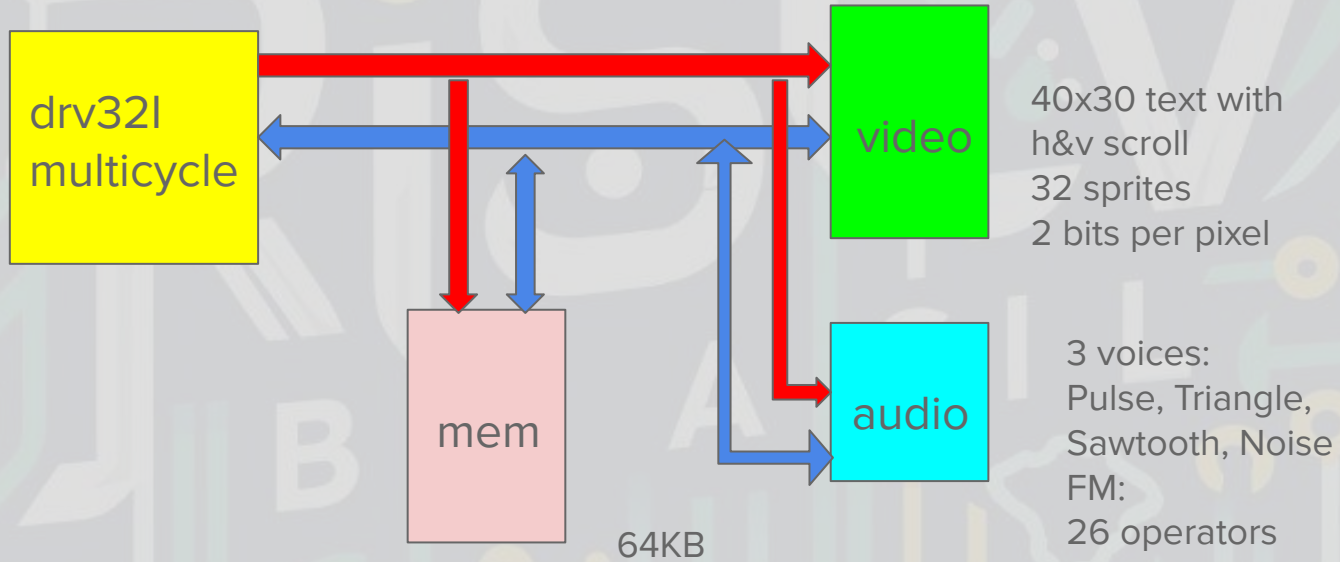
game.c  
neslib.h  
title\_nam.h  
level\_nam.h  
gameover\_nam.h  
welldone\_nam.h  
level1\_nam.h  
level2\_nam.h  
level3\_nam.h  
level4\_nam.h  
level5\_nam.h  
tileset.chr  
famitone2.s  
music.s  
sounds.s  
tileset.s  
famitone2.lst  
music.lst  
sounds.lst  
tileset.lst  
game.lst  
Disassembly  
Memory Browser  
Memory Map  
VRAM Browser  
Memory Probe  
CRT Probe

```
481 //the main gameplay code
482
483 void game_loop(void)
484 {
485     oam_clear();
486
487     i=game_level-1;
488
489     vram_addr(NAMETABLE_A);
490     vram_unrle(levelList[i]); //unpack
491
492     vram_addr(NAMETABLE_A+0x0042);
493     vram_write((unsigned char*)statsStr,27); //
494
495     pal_bg(levelList[i+1]); //set up
496     pal_spr(palGamespr); //set up
497
498     player_all=0;
499     items_count=0;
500     items_collected=0;
501
502     //this loop reads the level nametable back f
503     //constructs game map, removes spawn points
504     //and writes back to the VRAM
505
506     i16=NAMETABLE_A+0x0080;
507     ptr=0;
508     wait=0;
509
510     for(i=2; i<MAP_HGT+2; ++i)
511     {
512         vram_addr(i16);
513         vram_read(nameRow, 32);
514         vram_addr(i16);
515
516         for(j=0; j<MAP_WDT<<1; j+=2)
517         {
518             spr=nameRow[j];
```

LEVEL: 1 GEMS: 004/020 LIVES: 3



# Third Generation (Retro)





# Third Generation (Vintage)

Sega  
SG-1000 1983  
NEC 780C 3.58MHz  
9918, 76489  
3KB



Nintendo  
Famicom 1983/  
NES 1985 (1985)  
2A03 1.79MHz  
PPU  
4.3KB



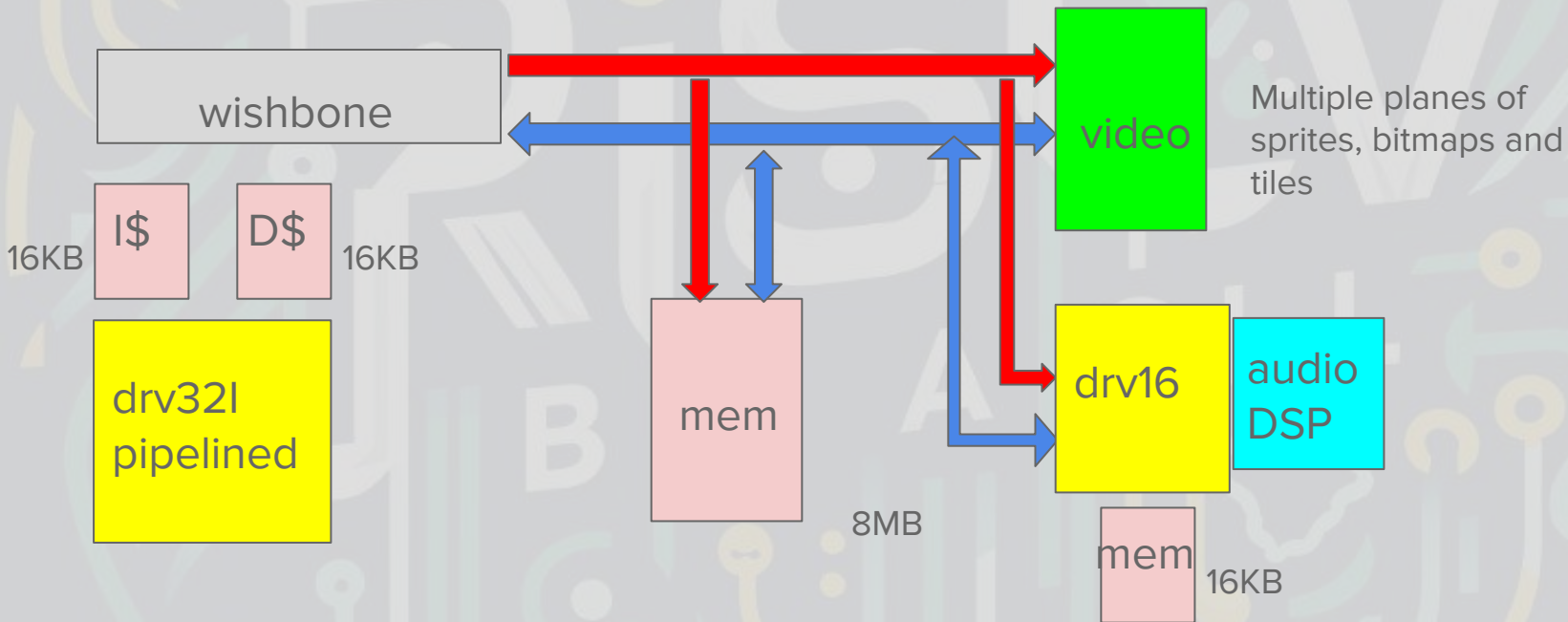
Sega  
Mark III 1985/  
Master System  
1986 (1989)  
Z80 3.58MHz  
YM2602, 76496  
24KB



Atari  
7800 1986  
6502C 1.79MHz  
MARIA  
4KB



# Fourth Generation (Retro)



# Fourth Generation (Vintage)

Plus  
CD-ROM!

NEC  
PC-Engine 1987  
HuC6280A  
1.79MHz/7.16MHz  
HuC6260+HuC6270A,  
HuC6280A  
72KB



NEC  
SuperGrafx 1989  
HuC6280A  
1.79MHz/7.16MHz  
HuC6260+2xHuC6270A,  
HuC6280A  
160KB



Sega  
Mega Drive 1988/  
Genesis 1989  
68000 7.67MHz,  
Z80 3.58MHz  
YM7101, YM2612  
136KB



**Nintendo**  
**Super Famicom 1990/  
Super Nintendo 1991**  
**5A55 3.58MHz**  
**PPU1+PPU2, APU**  
**256KB**

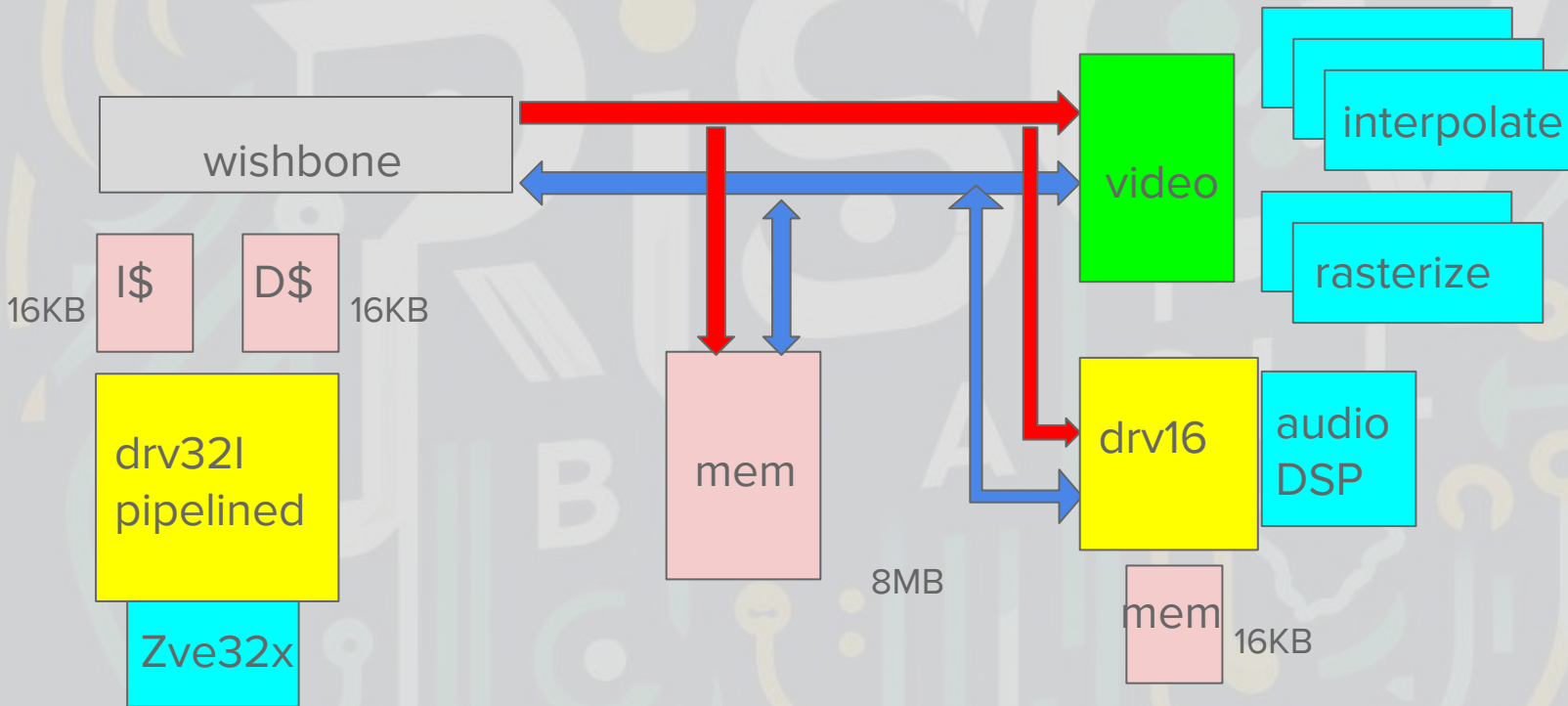


SNK  
Neo Geo 1991  
68000 12MHz,  
Z80 4MHz  
LSPC2-A2+  
PRO-B0,  
YM2610  
140KB





# Fifth Generation (Retro) ??



# Fifth Generation (Vintage)

3DO  
various 1994  
ARM60  
12.5MHz  
2x Video  
Coprocessors,  
DSP  
3MB



Atari  
Jaguar 1993  
68000 13.3MHz,  
TOM 26.59MHz,  
Jerry 26.59MHz  
2MB



Sega  
Saturn 1994  
2x SH-2  
28.63MHz,  
SH-1 20MHz,  
68EC000 11.3MHz  
VDP1+VDP2,  
YMF292  
4.5MB



Sony  
PlayStation  
1994  
MIPS R3051  
33.87MHz  
GPU,  
SPU  
3.6MB



Nintendo  
Nintendo 64  
1996  
MIPS VR4300  
93.75MHz  
Reality  
Coprocessor  
4MB



# Thanks!

Online copy of the book and project sources:

<https://github.com/jeceljlr/LivroComputadoresEVideojogos>

