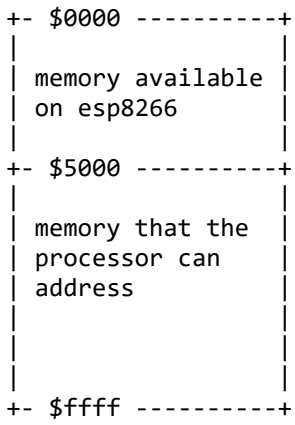
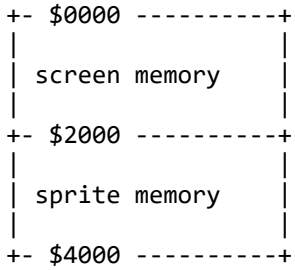
Description of the ESP Little Game Engine virtual machine

Memory map

RAM



SCREEN



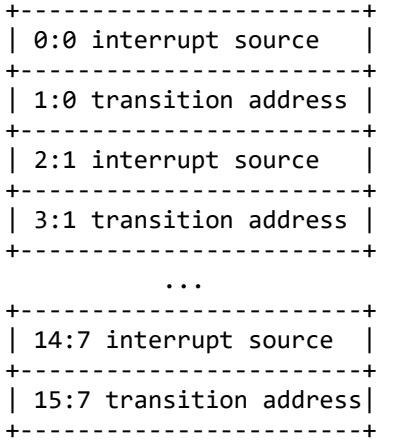
SPRITES

Sprite memory structure	Sprite attributes	
int16_t x;	S_X	0
int16_t y;	S_Y	1
int16_t angle;	S_ANGLE	6
uint16_t address;		
uint16_t oncollision;	S_ON_COLLISION	11
uint16_t onexitscreen;	S_ON_EXIT_SCREEN	12
int8_t speedx;	S_SPEEDX	2
int8_t speedy;	S_SPEEDY	3
int8_t lives;	S_LIVES	7
int8_t collision;	S_COLLISION	8
int8_t gravity;	S_GRAVITY	10
uint8_t width;	S_WIDTH	4
uint8_t height;	S_HEIGHT	5
uint8_t flags;		
flags		
0		
0		
0		
0		
fliphorizontal	S_FLIP_HORIZONTAL	15
isonebit	S_IS_ONEBIT	14
scrolled	S_IS_SCROLLED	13
solid	S_SOLID	9

The screen refreshes every 50 milliseconds. Update order:
1. Clearing sprite memory
2. Drawing sprites in sprite memory
3. Moving sprites
4. Collision Check
5. Drawing particles in the memory of sprites
6. Redrawing the changed areas of the screen
7. Setting the redraw flag to 1

Processor registers
R0: stack register
R1-R15: general purpose registers
PC: command counter
C: carry
Z: zero
N: negative
I: interrupt
R: redraw
tX: the current carriage position in the line
tY: the current line in which the carriage is located
color: brush color
bgcolor:background color

Interrupt buffer



Opcodes
Rx,Ry,Rz - register R1 - R16
XX - 8bit number
XXXX - 16bit number

assembler mnemonics	binary representation	description of instructions
LDI Rx,int	01 0R XXXX	Load 16 bit number into register Rx. Changes flags.
LDI Rx,(Ry)	02 RR	Load a 16-bit number into the Rx register at the address located in the Ry register. Changes flags.
LDI Rx,(adr)	03 0R XXXX	Load a 16-bit number into the Rx register at the address XXXX. Changes flags.
LDI Rx,(adr+Ry)	04 RR XXXX	Load a 16-bit number into the Rx register at the address located in XXXX and the offset in the Ry register. Changes flags.
STI (Rx),Ry	05 RR	Save 16 bit number from Ry register to address located in Rx register.
STI (adr),Rx	06 R0 XXXX	Save 16 bit number from Rx register to address XXXX.

STI (adr+Rx),Ry	06 RR XXXX	Save 16 bit number from Ry register to address XXXX and offset located in Rx register.
MOV Rx,Ry	07 RR	Copy contents of register Ry to register Rx
LDIAL Rx,(adr+Ry*2)	08 RR XXXX	Load a 16-bit number into the Rx register at the address located in XXXX and the offset in the Ry register multiplied by two Changes flags.
STIAL (adr+Rx*2),Ry	09 RR XXXX	Save 16 bit number from Ry register to address XXXX and offset located in Rx register multiplied by two.
LDC Rx,char	1R XX	Load 8bit number into Rx register. Changes flags.
LDC Rx,(Ry)	20 RR	Load a 8-bit number into the Rx register at the address located in the Ry register. Changes flags.
LDC Rx,(Ry+Rz)	2R RR	Load a 8-bit number into the Rx register at the address located in register Ry and offset Rz. Changes flags.
LDC Rx,(adr+Ry)	30 RR XXXX	Load a 8-bit number into the Rx register at the address located in XXXX and offset Ry. Changes flags.
LDC Rx,(adr)	31 0R XXXX	Load a 8-bit number into the Rx register at the address XXXX. Changes flags.
STC (adr),Rx	32 0R XXXX	Save 8 bit number from Rx register to address XXXX
STC (adr+Rx),Ry	33 RR XXXX	Save 8 bit number from Ry register to address XXXX and offset located in Rx register
STC (Rx),Ry	40 RR	Save 8 bit number from Ry register to address located in Rx register
STC (Rx+Ry),Rz	4R RR	Save the 8 bit value of the register Rz at the address located in the register Rx and the offset Ry
HLT	50 00	Abort a program
STIMER Rx,Ry	51 RR	Write the value of register Ry to the timer with the number Rx
GTIMER Rx	52 0R	Get the timer value with the number written in the Rx register. Changes flags.
SETLED Rx	53 0R	Write color in R5G6B5 format to the LED (if available)
LOADRT Rx,Ry	54 RR	Download from the Rx register the address of the string containing the melody in rtttl format. If the Ru register is not zero, the melody will be played in a loop.
PLAYRT	55 00	Play rtttl.
PAUSERT	55 01	Pause rtttl.
STOPRT	55 02	Stop rtttl. Play position is reset.

PLAYTN Rx,Ry	56 RR	Generates a square wave of the specified frequency. Rx: the frequency of the tone in hertz. Ry: the duration of the tone in milliseconds
LDATA Rx	57 0R	Loads data into an array with an address in Rx.Returns the number of bytes loaded in Rx
NDATA Rx	57 1R	Sets the address of a string with a save name
SDATA Rx,Ry	58 RR	Saves an array with an address in Rx. Ry contains the number of elements in the array. Returns the number of saved items or the number of free blocks if saving failed.
LDI Rx,(Ry+Rz)	6R RR	Load a 16-bit number into the Rx register at the address located in register Ry and offset Rz. Changes flags.
STI (Rx+Ry),Rz	7R RR	Save the 16-bit value of the register Rz at the address located in the register Rx and the offset Ry
POP Rx	80 0R	Loads two bytes from the top of the stack into the Rx register. Increases the value of register R0 by two.
POPN Rx	81 0R	Loads data from the top of the stack into registers, starting with R1 and ending with the value written to Rx. Increases R0 by Rx * 2
PUSH Rx	82 0R	Stores the value of the Rx register at the top of the stack. Decreases the value of register R0 by two
PUSHN Rx	83 0R	Saves the values of the registers from R1 to the number indicated in the register Rx. Decreases the value of register R0 by Rx * 2
JMP adr	90 00 XXXX	Go to XXXX
JNZ adr	91 00 XXXX	Go to XXXX if the "zero" flag is not zero
JZ adr	92 00 XXXX	Go to XXXX if the "zero" flag is zero
JNP adr	93 00 XXXX	Go to XXXX if the "negative" flag is zero
JP adr	94 00 XXXX	Go to XXXX if the "negative" flag is not zero
JNC adr	95 00 XXXX	Go to XXXX if the "cary" flag is zero
JC adr	96 00 XXXX	Go to XXXX if the "cary" flag is not zero
JZR Rx,adr	97 0R XXXX	Go to XXXX if the value of register Rx is zero
JNZR Rx,adr	98 0R XXXX	Go to XXXX if the value of register Rx is not zero
CALL adr	99 00 XXXX	Jump to address XXXX and save the return address to the top of the stack. The value of R0 decreases by two
RET	9A 00	Go to the address on the top of the stack.

		The value of R0 is increased by two.
ADD Rx,Ry	A0 RR	Add the contents of register Rx and register Ry, place the result in Rx. Changes flags.
ADC Rx,Ry	A1 RR	Add the contents of the Rx register and the Ry register with the carry flag, place the result in Rx. Changes flags.
SUB Rx,Ry	A2 RR	Subtract the contents of register Ry from register Rx, place the result in Rx. Changes flags.
SBC Rx,Ry	A3 RR	Subtract the contents of the register Ry and the carry flag from the register Rx, place the result in Rx. Changes flags.
MUL Rx,Ry	A4 RR	Multiply the contents of the register Rx by Ry, put the result in Rx. Changes flags.
DIV Rx,Ry	A5 RR	Divide the contents of the Rx register by Ry, place the result in Rx. Changes flags.
AND Rx,Ry	A6 RR	Logical And between Rx and Ry, put the result in Rx. Changes flags.
OR Rx,Ry	A7 RR	Logical OR between Rx and Ry, put the result in Rx. Changes flags.
INC Rx	A8 0R	Increment the contents of the Rx register. Changes flags.
INC adr	A8 10 XXXX	Increment the value at XXXX. Changes flags.
INC Rx,n	A8 nR	Add the number n to the contents of the register Rx. Changes flags.
DEC Rx	A9 0R	Decrement the contents of the Rx register. Changes flags.
DEC adr	A9 10 XXXX	Decrement the value at XXXX. Changes flags.
DEC Rx,n	A9 nR	Subtract n from the contents of register Rx Changes flags.
XOR Rx,Ry	AA RR	Logical XOR between Rx and Ry, put the result in Rx. Changes flags.
SHL Rx,Ry	AB RR	Rotate the bits of the contents of the Rx register to the left by the number contained in the Ry register. Changes flags
SHR Rx,Ry	AC RR	Rotate the bits of the contents of the Rx register to the right by the number contained in the Ry register. Changes flags
RAND Rx	AD 0R	Returns a random number from 0 to the contents of the Rx register in the Rx register. Changes flags.
SQRT Rx	AD 1R	Returns the square root of the contents of the Rx register to the Rx register. Changes flags.
ANDL Rx,Ry	AE RR	Returns to the register Rx 1 if the register Rx is not equal to zero and the register Ry is not equal to zero. Changes flags.

ORL Rx,Ry	AF RR	Returns to the register Rx 1 if the register Rx is not equal to zero or the register Ry is not equal to zero. Changes flags.
CMP Rx,CHR	BR XX	Compares the Rx register with an 8-bit value of XX. Changes flags.
CMP Rx,INT	C0 R0 XXXX	Compares the Rx register with an 16-bit value of XXXX. Changes flags.
CMP Rx,Ry	C1 RR	Compares the value of register Rx with register Ry. Changes flags.
LDF Rx,F	C2 RF	Loads the value of the flag F in the register Rx. F can be: 0 - transfer, 1 - zero, 2 - negative, 3 - positive, 4 - not positive, 5 - not zero, 6 - redraw After requesting the redraw flag, the flag is set to 0
CLS	D0 00	Clears the screen.
GSPRXY Rx,Ry	D0 RR	Get the number of the sprite that is in position X (Rx) and Y (Ry), or 0 if there is no sprite in position.
PUTC Rx	D1 0R	Print the character contained in register Rx at position tX in line tY.
PUTS Rx	D1 1R	Print the string contained in register Rx at position tX in line tY.
PUTN Rx	D1 2R	Print an unsigned number contained in register Rx at position tX in line tY.
SETX Rx	D1 3R	Set the value of tX to the value of the register Rx.
SETY Rx	D1 4R	Set the value of tY to the value of the register Rx.
GETK Rx	D2 0R	Get Rx character from UART or virtual keyboard.
GETJ Rx	D2 1R	Get the Rx byte into the register, each bit of which corresponds to the value of one of the eight buttons.
PPIX Rx,Ry	D3 RR	Draw a point with the current color at positions X (Rx) and Y (Ry).
DRWIM Rx	D4 0R	Draw an image. The register Rx indicates a piece of memory in which the height, width, y, x, and address are located in series.
SFCLR Rx	D4 1R	Set the brush color to the value contained in the Rx register.
SBCLR Rx	D4 2R	Set the background color to the value contained in the Rx register.
GFCLR Rx	D4 3R	Get current brush color to Rx register.
GBCLR Rx	D4 4R	Get current background color in Rx register
ISIZE Rx	D4 5R	Set pixel size when drawing picture.

DLINE Rx	D4 6R	Draw a line. The register Rx indicates a portion of memory in which y1, x1, y, x are located sequentially.
DRWRLE Rx	D4 7R	Draw an image encoded in RLE format. The register Rx indicates a piece of memory in which the height, width, y, x, and address are located in series.
LDTILE Rx	D4 8R	Download tile information. The Rx register points to a section of memory in which height, width, imageheight, imagewidth, the address of the array of tiles are located.
SPRSDS Rx	D4 9R	Set the speed and direction of movement of the sprite.Register Rx indicates a piece of memory in which direction, speed, sprite number are located in series.
DRWBIT Rx	D4 AR	Draw an 1bit image. The register Rx indicates a piece of memory in which the height, width, y, x, and address are located in series.
LDSPRT Rx,Ry	D5 RR	Loads the image address from the Ry register into the sprite with the number Rx
SPALET Rx,Ry	D6 RR	Changes the color in the palette under the number contained in the Rx register to the color contained in the Ry register in the format R5G6B5.
SPART Rx	D7 0R	Changes particle settings. The Rx register indicates a piece of memory in which the particle count, particle lifetime, particle gravity are located in series.
SEMIT Rx	D7 1R	Changes the emitter settings. The Rx register indicates a piece of memory in which speed, direction2, direction1, the emitter lifetime are located in series.
DPART Rx	D7 2R	Draws particles. The register Rx indicates a portion of memory in which the color of the particles, y, x.
DISTPP	D7 5R	Returns the distance between two points in the Rx register. The Rx register indicates a memory location in which y2, x2, y1, x1 are located sequentially.
SCROLL Rx	D8 0R	Scroll the screen. The Rx register indicates the direction of the scroll. Scrolling vertically occurs by one pixel, horizontally by two pixels.
GETPIX Rx,Ry	D9 RR	Returns the color index of the pixel at point X (Rx) and Y (Ry)
DRTILE Rx,Ry	DA RR	Sets the values of X (Rx) and Y (Ry) for tiles and redraws them.
SPRGET Rx,Ry	DC RR	Gets the value of the sprite parameter. The register Rx contains the number of the sprite; the register Ry contains the number of the value.
AGBSPR Rx,Ry	DE RR	Returns the angle between sprites in the Rx register. The number of the first sprite is

		indicated in the register Rx, the number of the second in the register Ry.
GTILEXY Rx,Ry	DF RR	Returns the address of the tile at position X (Rx) and Y (Ry) in the register Rx.
DRSPRT Rx,Ry,Rz	ER RR	Sets the sprite with the number specified in the Rx register to the position X (Ry) and Y (Rz) and sets its parameter S_LIVES to one if it is less than zero.
SSPRTV Rx,Ry,Rz	FR RR	Sets the Ry property of the sprite with the number specified in the Rx register to the value specified in the Rz register.