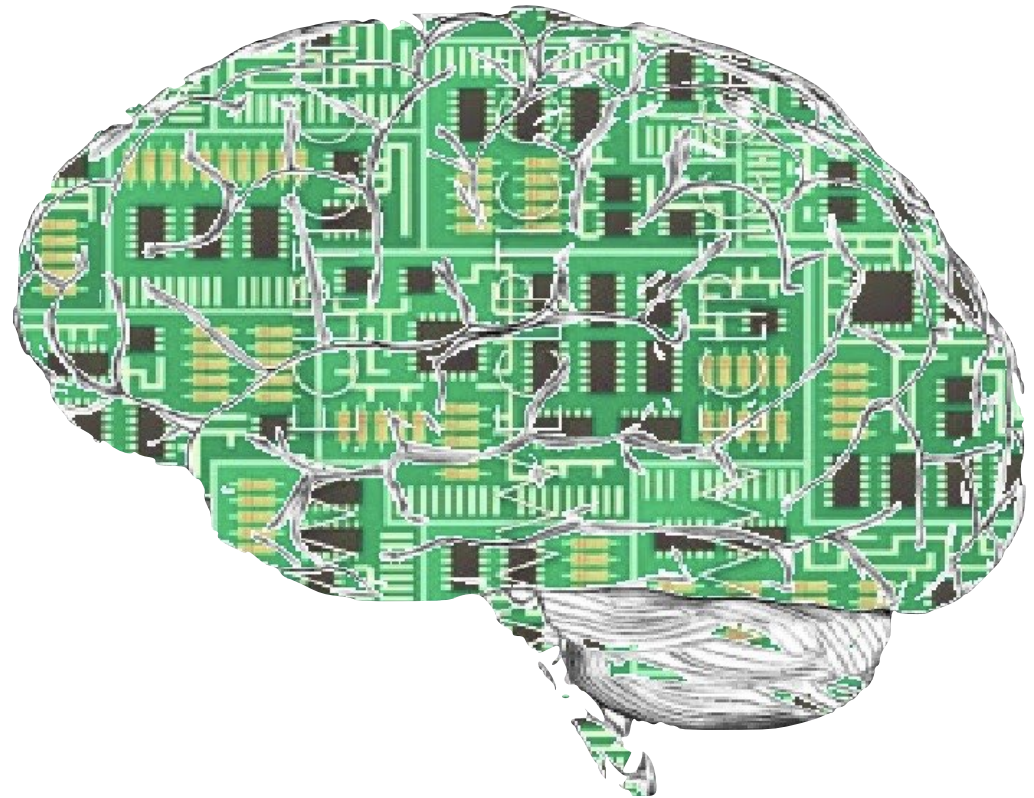


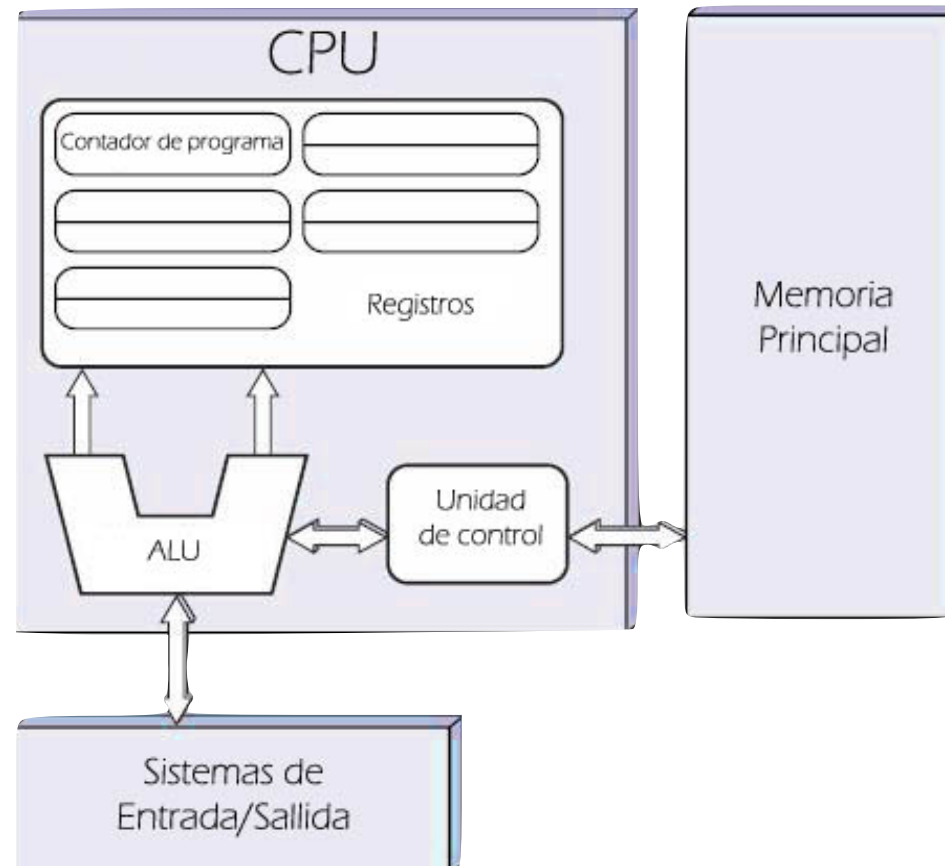
DEPARTAMENTO DE
COMPUTACIÓN
UNRC
2016

ORGANIZACIÓN DEL PROCESADOR



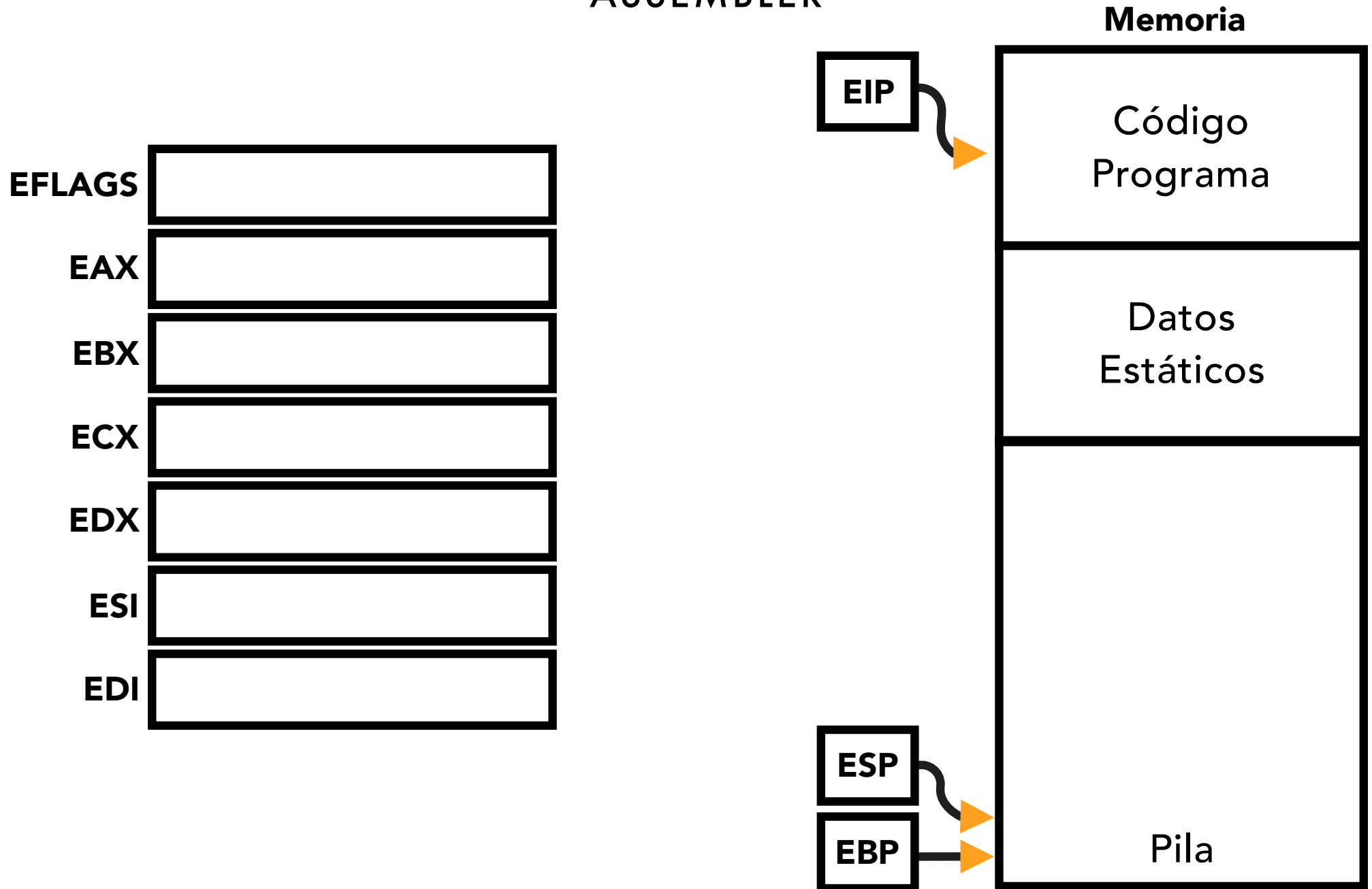
ORGANIZACIÓN DEL PROCESADOR

ASSEMBLER



ORGANIZACIÓN DEL PROCESADOR

ASSEMBLER



ORGANIZACIÓN DEL PROCESADOR

SUBROUTINAS

```
call Writeln      ;
Call NewLine      ;
mov dx, Msg6       ;
call Write        ;
mov cx, 10         ;
mov di, OpA        ;
Call NewLine      ;
mov dx, Msg7       ;
call Write        ;
mov cx, 10         ;
mov di, OpB        ;
Call NewLine      ;
mov si, OpA        ;
mov di, OpB        ;
mov bx, prd        ;
call Mult128Bits   ;
mov dx, Msg8       ;
call Write        ;
mov si, prd        ;
mov cx, 20         ;
mov bx, Res        ;

call Writeln128BitS ;
mov si, Res        ;
call WriteNum      ;
Call NewLine      ;
Call NewLine      ;
ret                ;
```

ORGANIZACIÓN DEL PROCESADOR

SUBROUTINAS

```
call Writeln      ;
Call NewLine      ;
mov dx, Msg6       ;
call Write        ;
mov cx, 10         ;
mov di, OpA        ;
Call NewLine      ;
mov dx, Msg7       ;
call Write        ;
mov cx, 10         ;
mov di, OpB        ;
Call NewLine      ;
mov si, OpA        ;
mov di, OpB        ;
mov bx, prd        ;
call Mult128Bits   ;
mov dx, Msg8       ;
call Write        ;
mov si, prd        ;
mov cx, 20         ;
mov bx, Res        ;

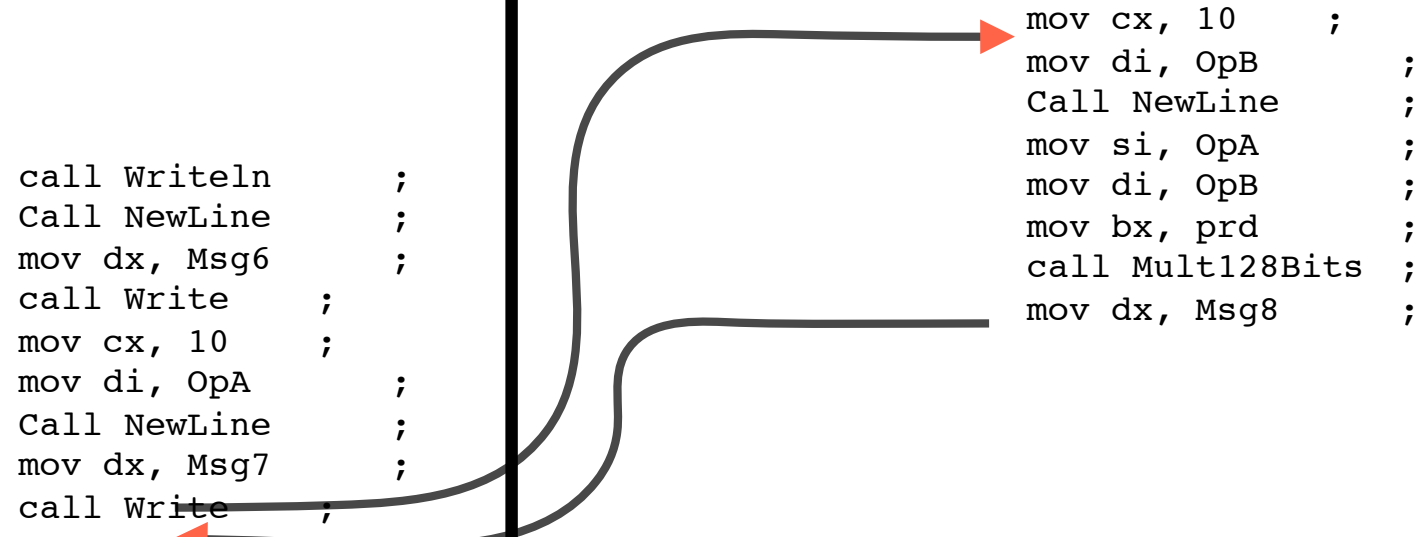
call Writeln128BitS ;
mov si, Res        ;
call WriteNum      ;
Call NewLine      ;
Call NewLine      ;
ret                ;
```

ORGANIZACIÓN DEL PROCESADOR

SUBROUTINAS

```
call WriteLn      ;
Call NewLine     ;
mov dx, Msg6      ;
call Write       ;
mov cx, 10        ;
mov di, OpA       ;
Call NewLine     ;
mov dx, Msg7      ;
call Write       ;
call Write       ;
mov si, prd       ;
mov cx, 20        ;
mov bx, Res       ;

call Write128Bits ;
mov si, Res       ;
call WriteNum     ;
Call NewLine     ;
Call NewLine     ;
ret              ;
```

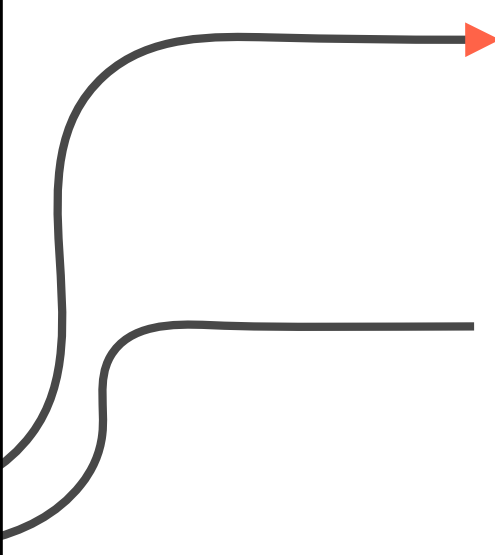


```
mov cx, 10      ;
mov di, OpB     ;
Call NewLine    ;
mov si, OpA     ;
mov di, OpB     ;
mov bx, prd     ;
call Mult128Bits ;
mov dx, Msg8    ;
```

ORGANIZACIÓN DEL PROCESADOR

SUBROUTINAS

```
call WriteLn      ;  
Call NewLine     ;  
mov dx, Msg6      ;  
call Write       ;  
mov cx, 10        ;  
mov di, OpA       ;  
Call NewLine     ;  
mov dx, Msg7      ;  
call Write       ;  
call Write       ;  
mov si, prd       ;  
mov cx, 20        ;  
mov bx, Res       ;  
  
call Write128Bits ;  
mov si, Res       ;  
call WriteNum     ;  
Call NewLine     ;  
Call NewLine     ;  
ret              ;
```

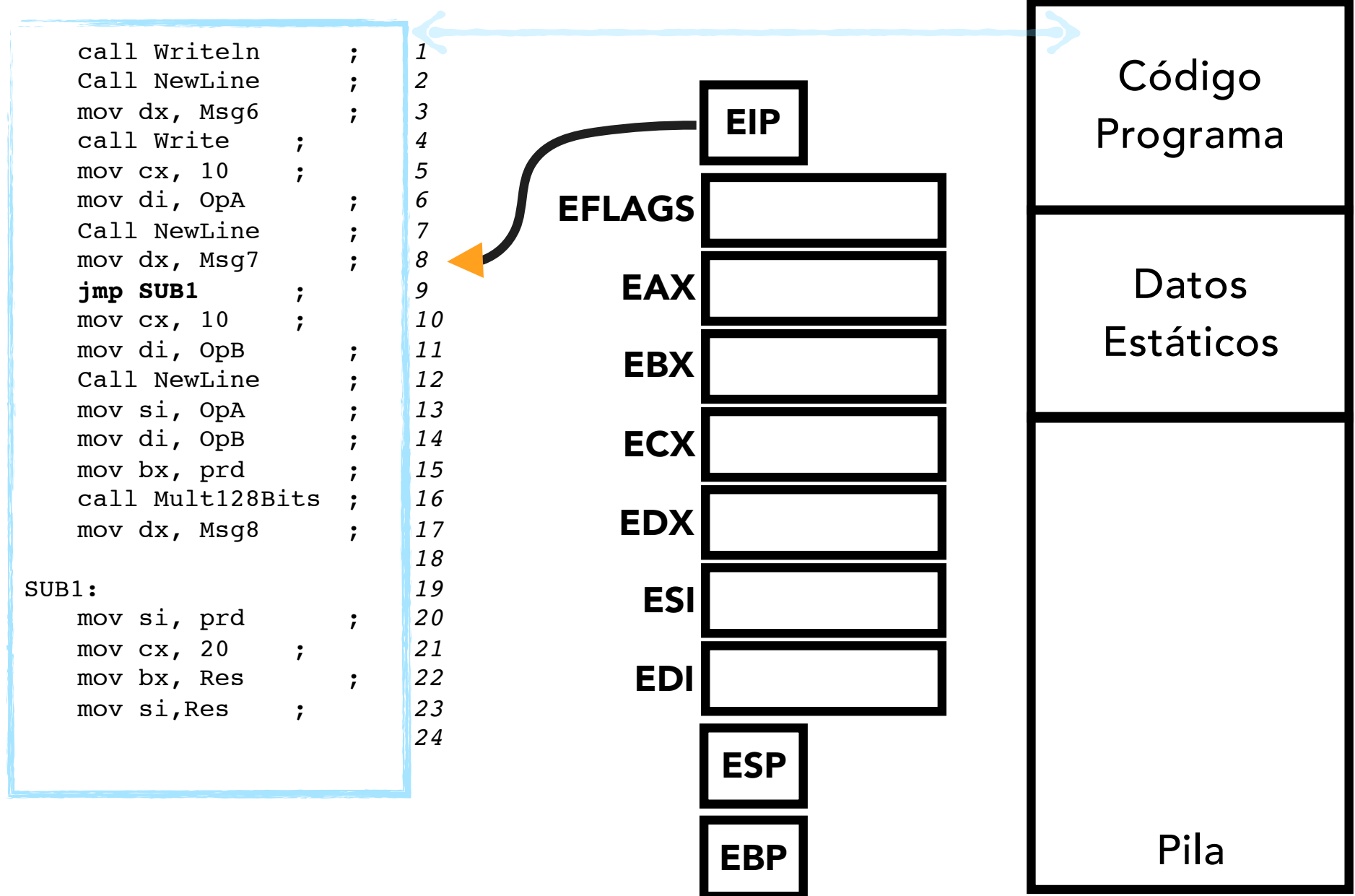


```
mov cx, 10      ;  
mov di, OpB     ;  
Call NewLine    ;  
mov si, OpA     ;  
mov di, OpB     ;  
mov bx, prd     ;  
call Mult128Bits ;  
mov dx, Msg8    ;
```

- Modularidad
- Reutilización

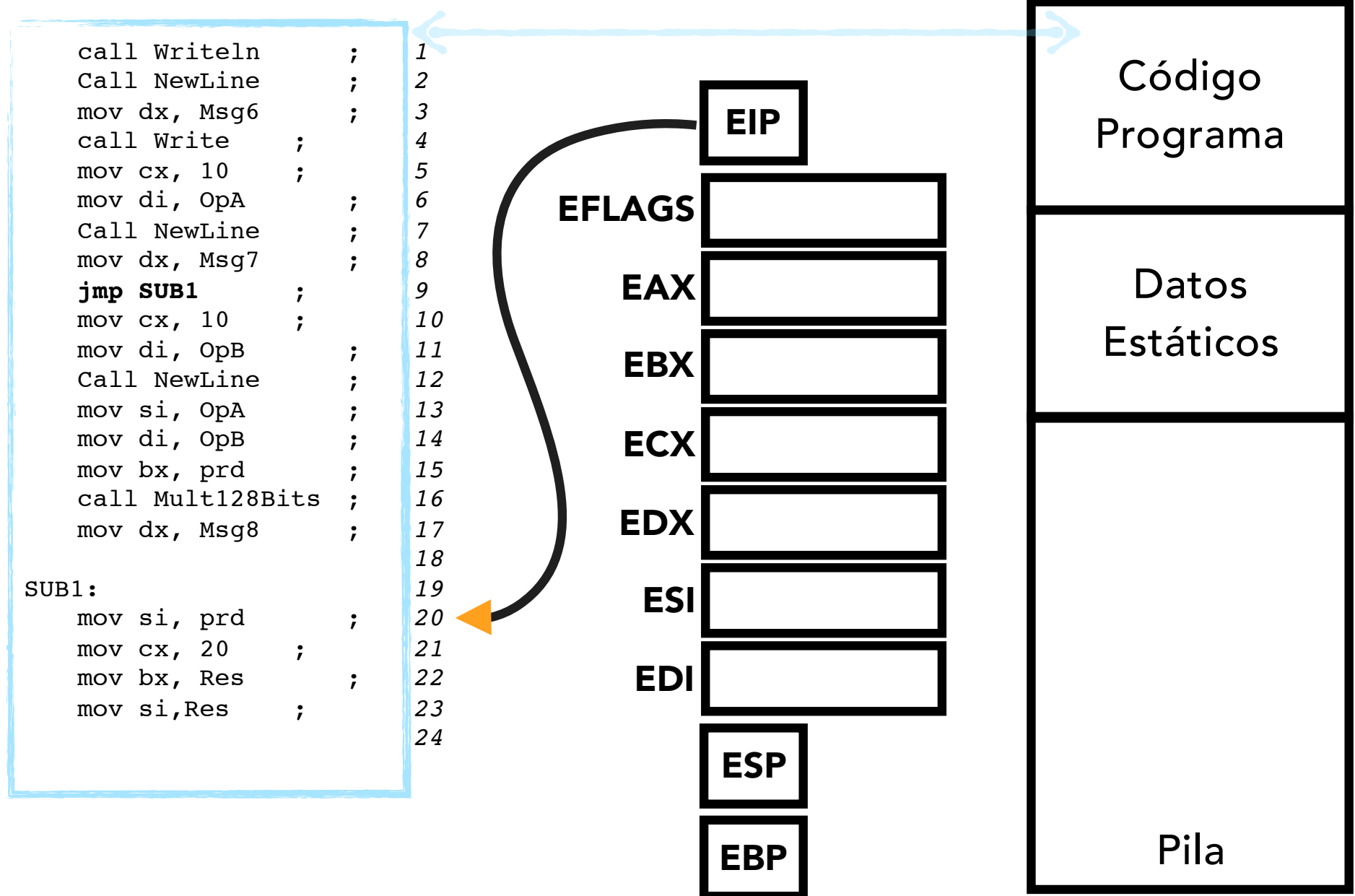
ORGANIZACIÓN DEL PROCESADOR

SUBROUTINAS - INTENTO MANUAL



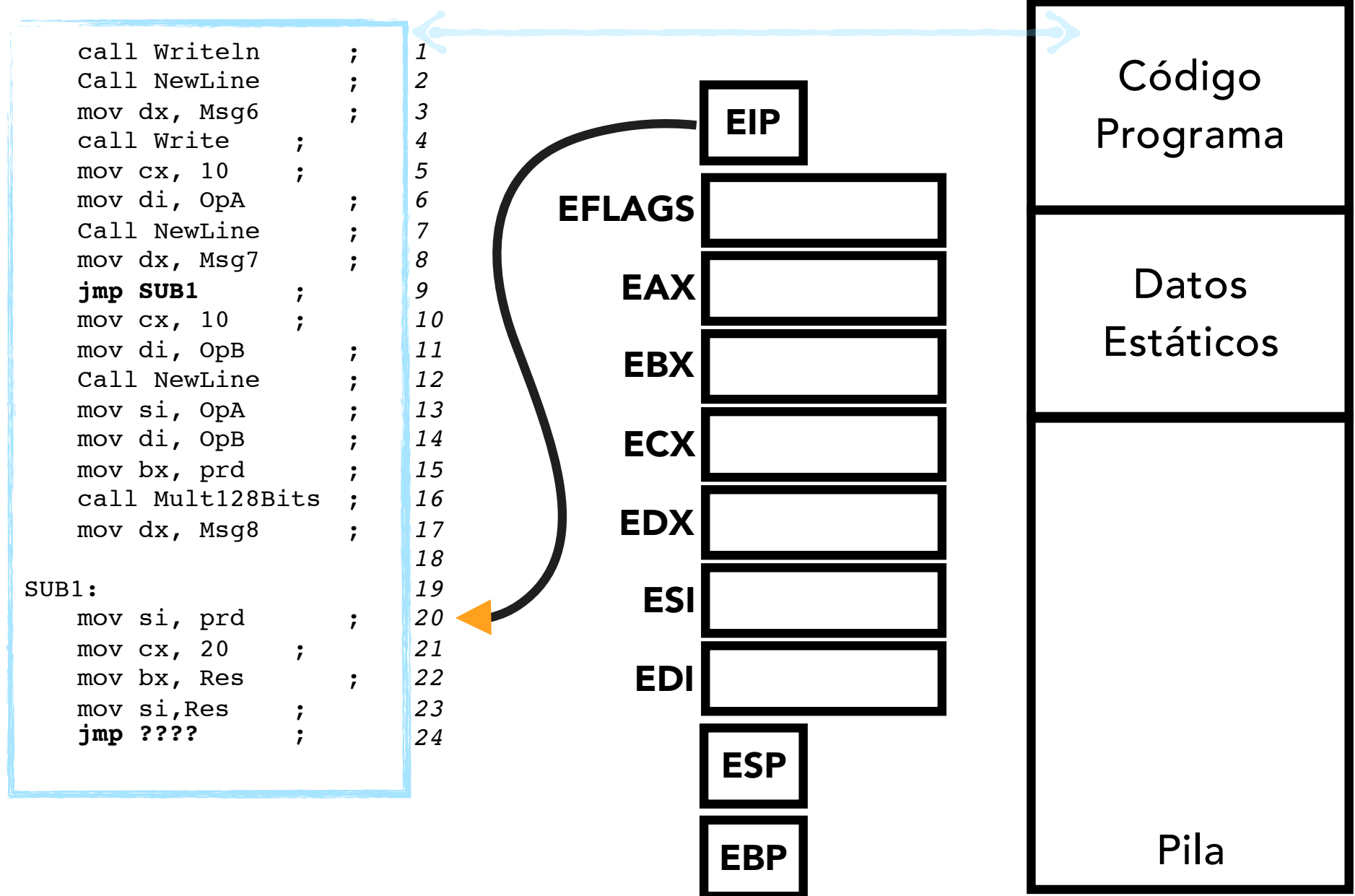
ORGANIZACIÓN DEL PROCESADOR

SUBROUTINAS - INTENTO MANUAL



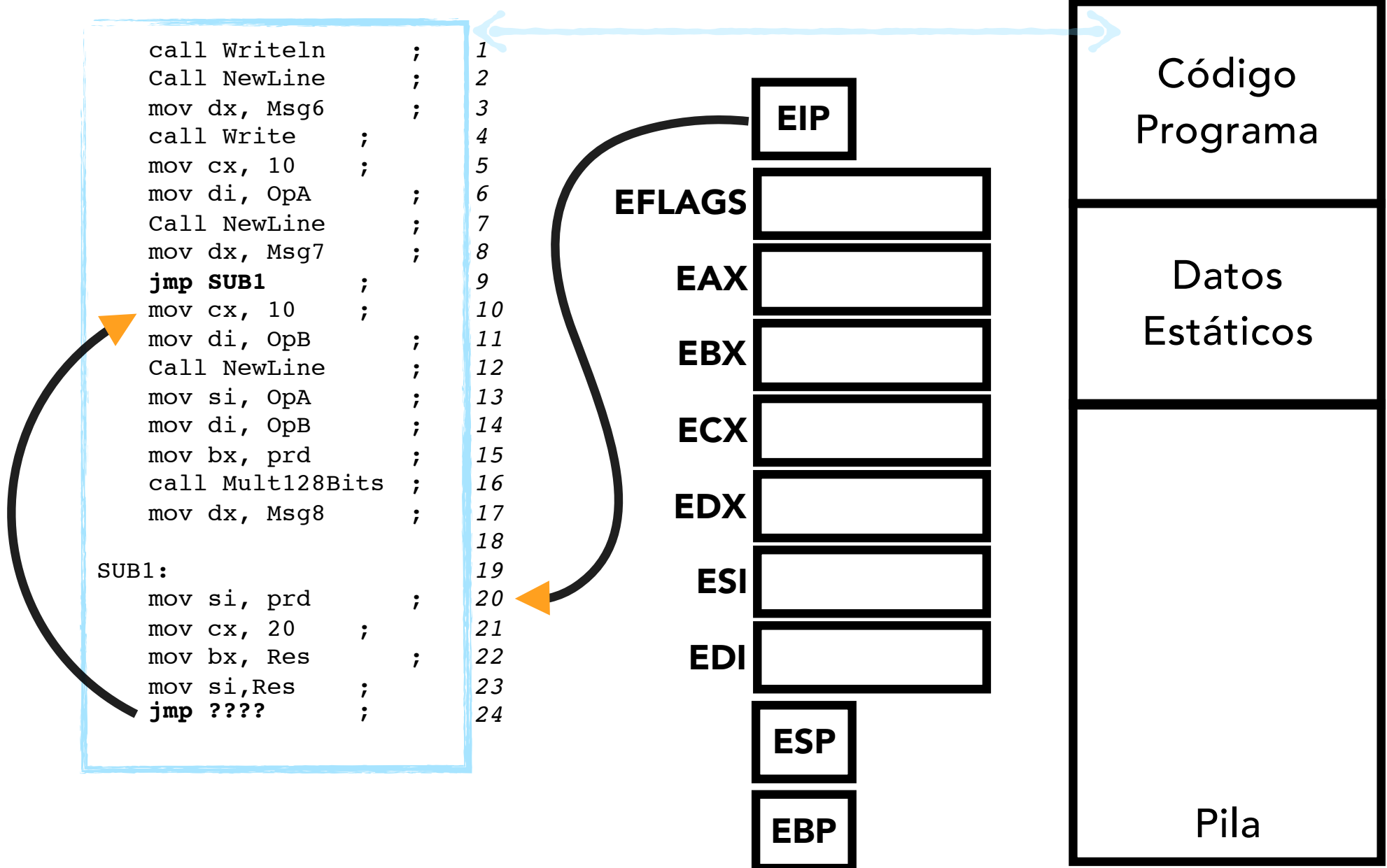
ORGANIZACIÓN DEL PROCESADOR

SUBROUTINAS - INTENTO MANUAL



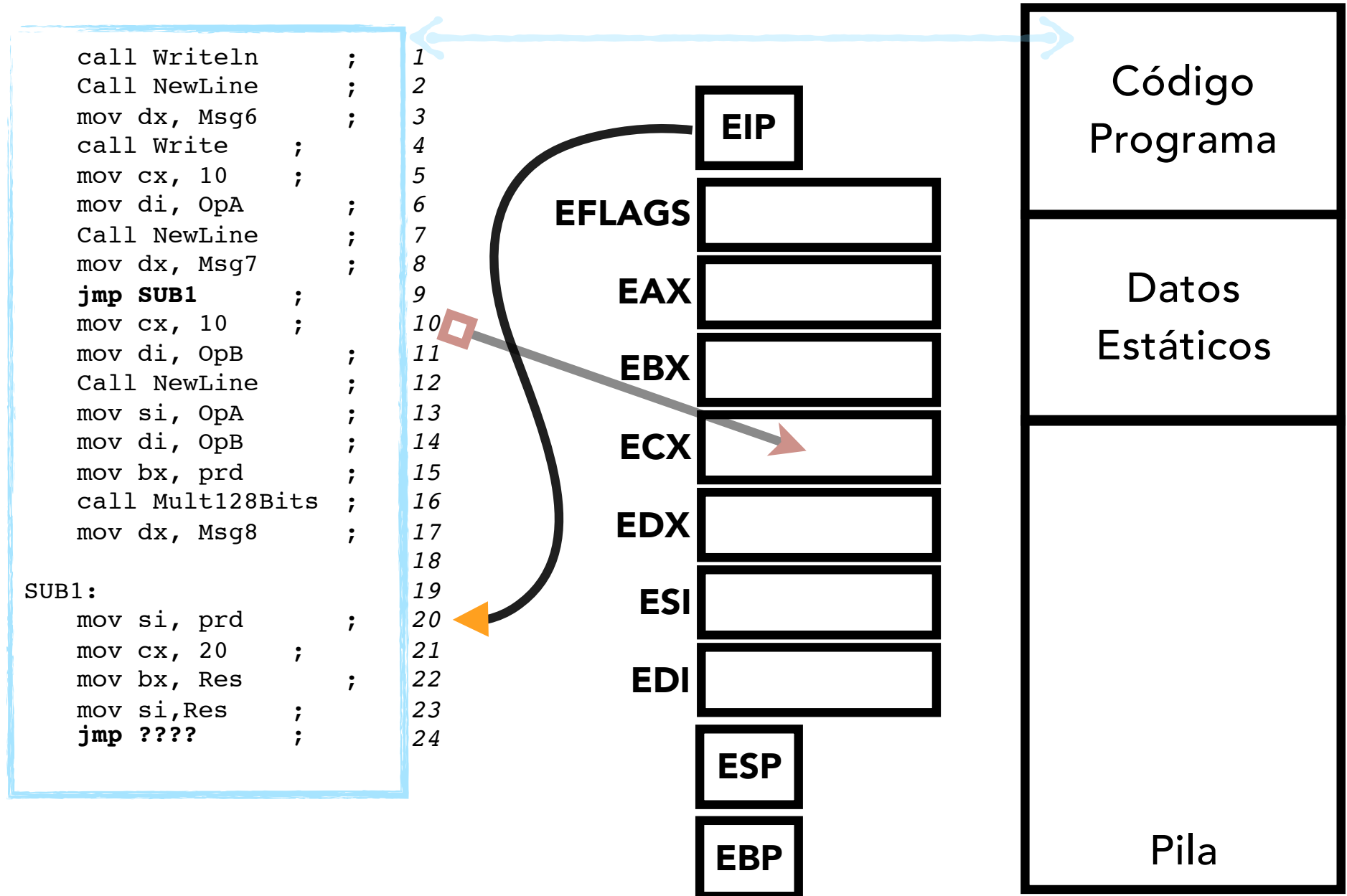
ORGANIZACIÓN DEL PROCESADOR

SUBROUTINAS - INTENTO MANUAL



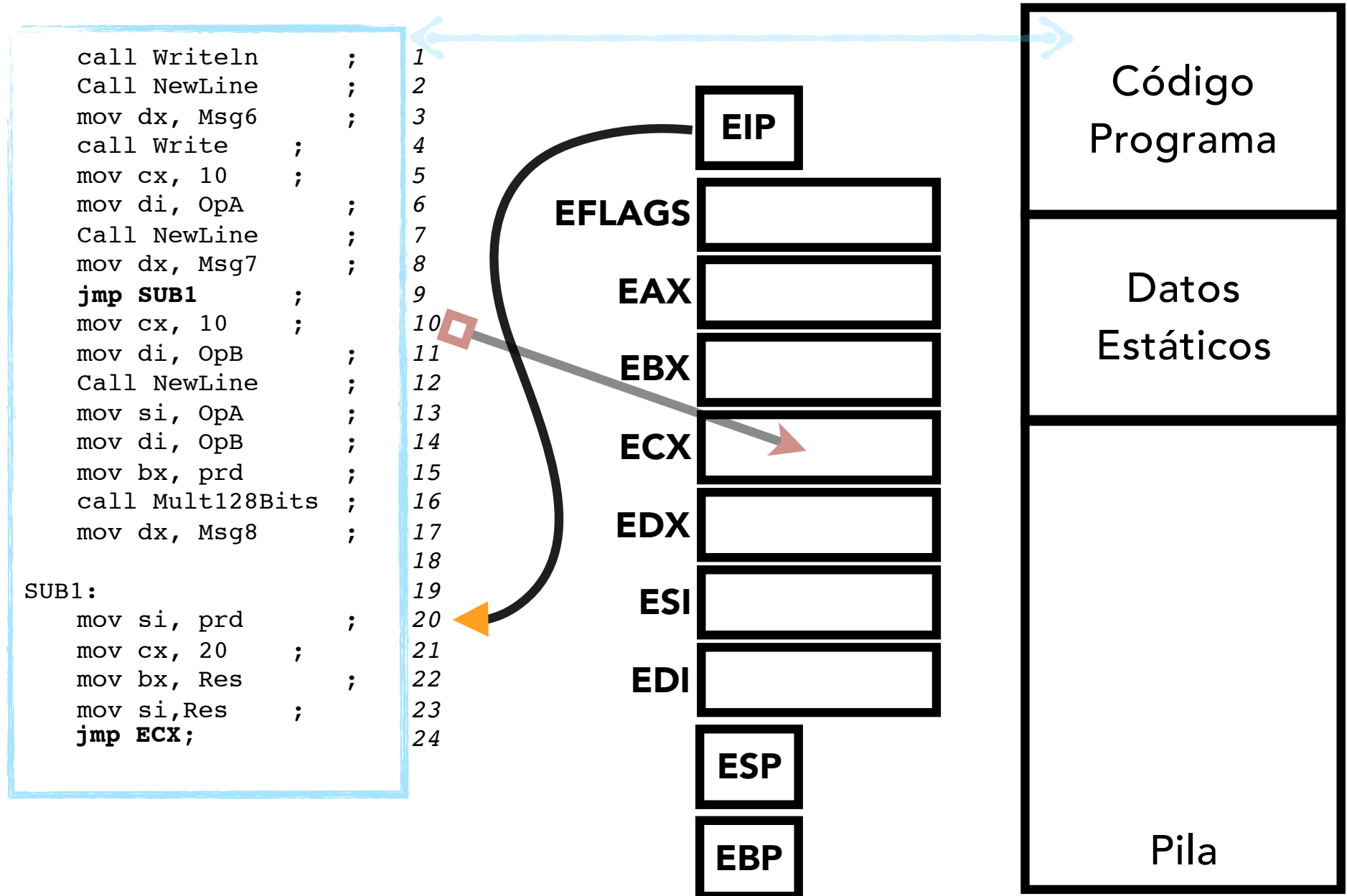
ORGANIZACIÓN DEL PROCESADOR

SUBROUTINAS - INTENTO MANUAL



ORGANIZACIÓN DEL PROCESADOR

SUBROUTINAS - INTENTO MANUAL



ORGANIZACIÓN DEL PROCESADOR

SUBROUTINAS - INTENTO MANUAL

Memoria

```
call WriteLn      ; 1
Call NewLine     ; 2
mov dx, Msg6      ; 3
call Write       ; 4
mov cx, 10        ; 5
mov di, OpA       ; 6
Call NewLine     ; 7
mov dx, Msg7      ; 8
jmp SUB1        ; 9
mov cx, 10        ; 10
mov di, OpB       ; 11
Call NewLine     ; 12
mov si, OpA       ; 13
mov di, OpB       ; 14
mov bx, prd       ; 15
call Mult128Bits ; 16
mov dx, Msg8      ; 17
SUB1:            ; 18
  mov si, prd     ; 19
  mov cx, 20      ; 20
  mov bx, Res     ; 21
  mov si, Res     ; 22
  jmp SUB1       ; 23
```

¿que sucede si tengo varias invocaciones anidadas?

EFLAGS

EAX

EBX

ECX

EDX

ESI

EDI

EIP

ESP

EBP

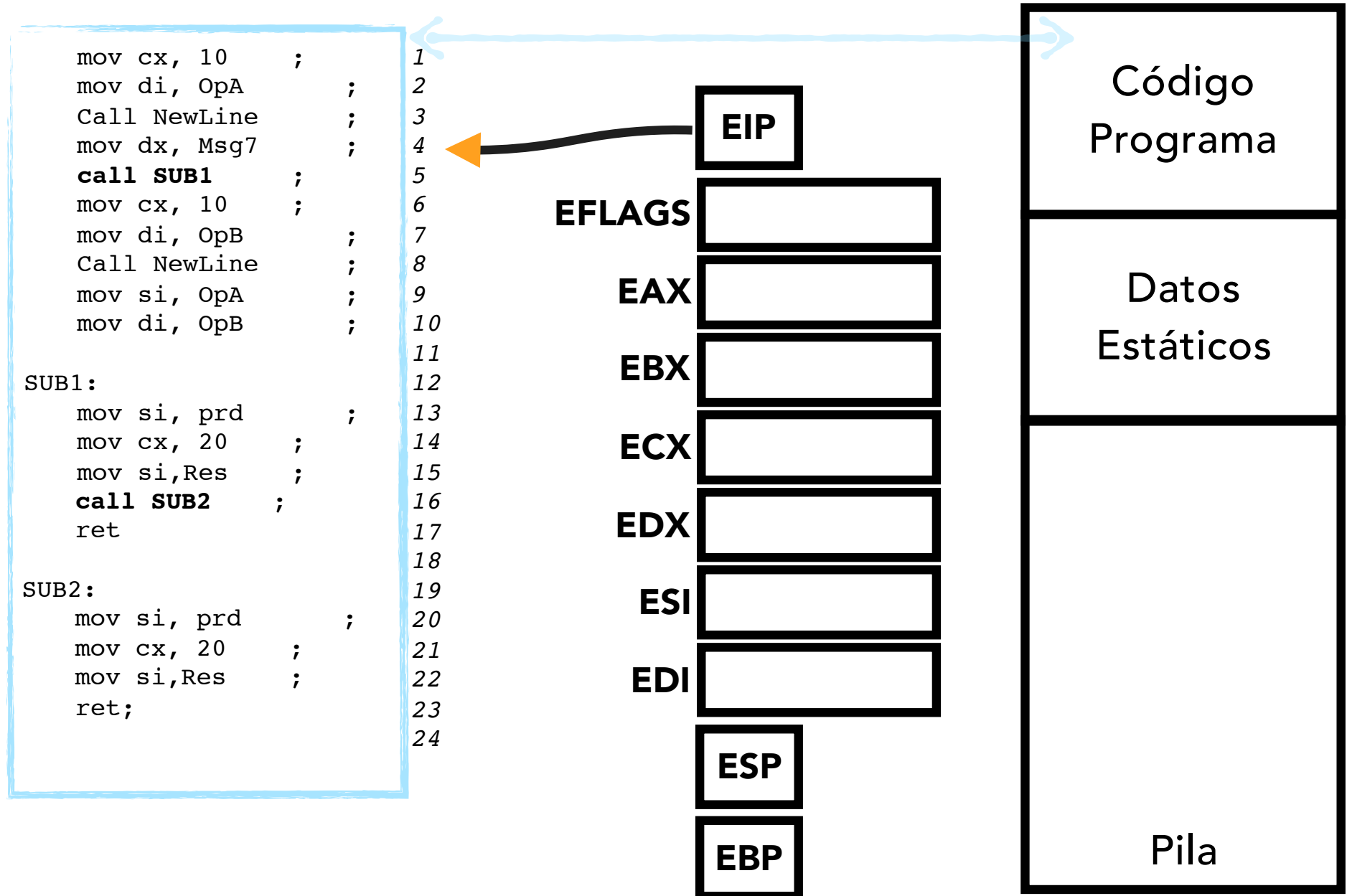
Código Programa

Datos Estáticos

Pila

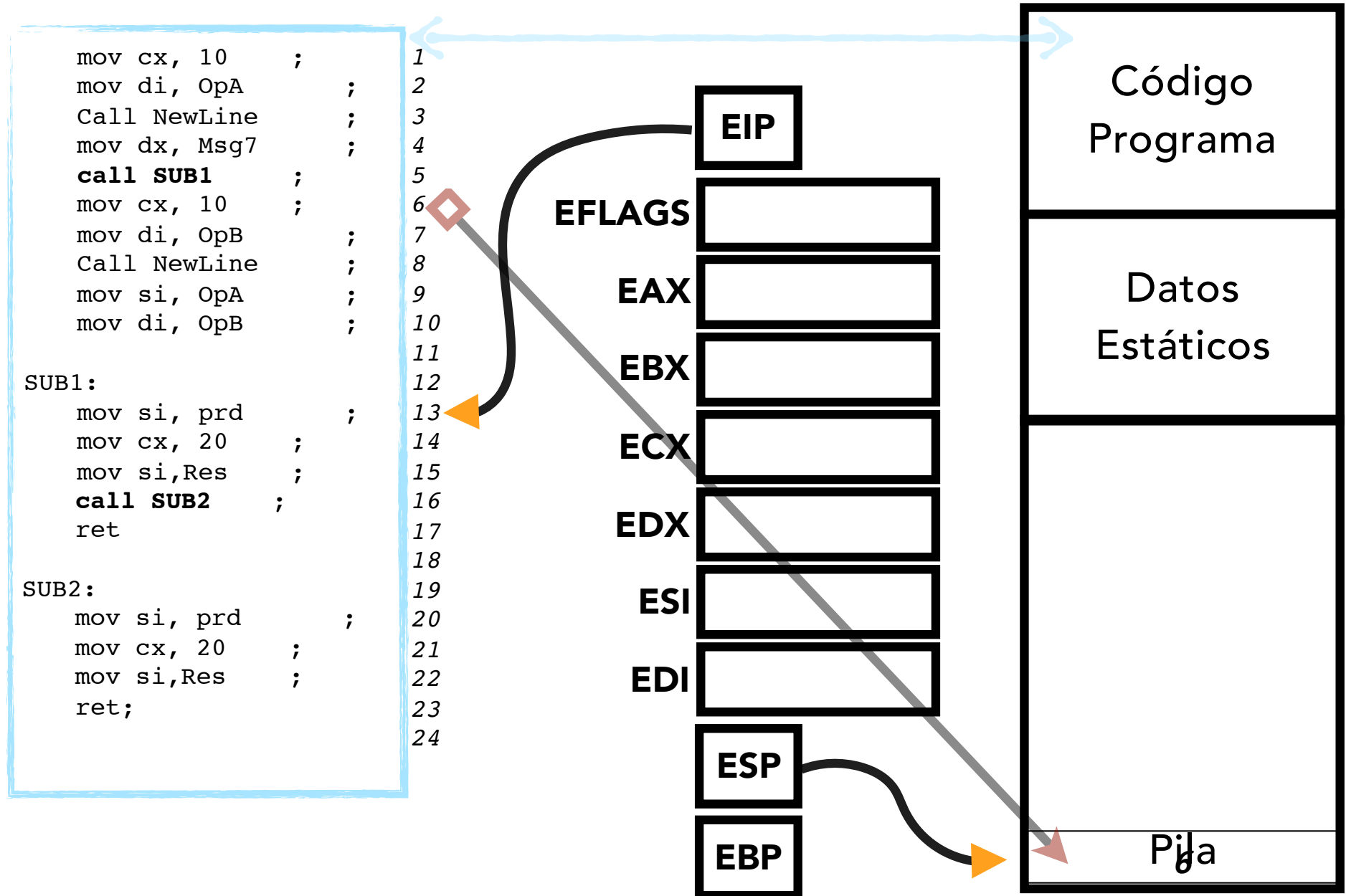
ORGANIZACIÓN DEL PROCESADOR

SUBROUTINAS - CALL & RET



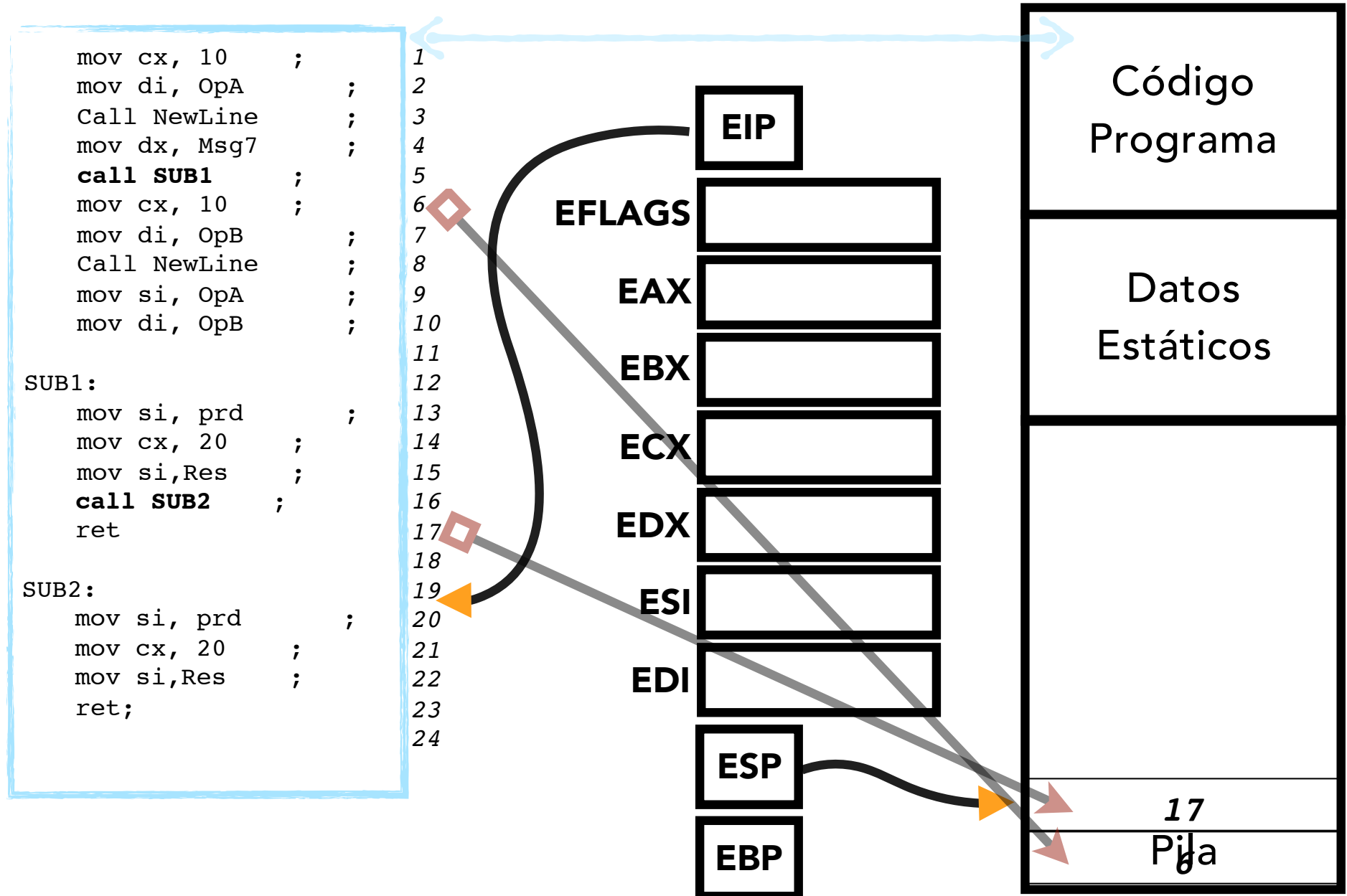
ORGANIZACIÓN DEL PROCESADOR

SUBROUTINES - CALL & RET



ORGANIZACIÓN DEL PROCESADOR

SUBROUTINAS - CALL & RET



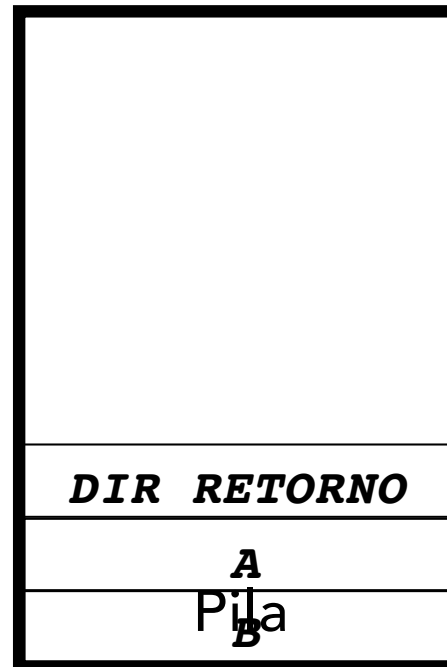
ORGANIZACIÓN DEL PROCESADOR

SUBROUTINAS - INTERCAMBIO DE INFORMACIÓN

...
Sub1 (a,b);
...

```
...  
mov di, OpA      ;  
Call NewLine     ;  
mov dx, Msg7     ;  
call SUB1       ;  
mov cx, 10       ;  
mov di, OpB      ;  
Call NewLine     ;  
mov si, OpA      ;  
...
```

```
SUB1:  
  mov si, prd      ;  
  mov cx, 20       ;  
  mov si, Res      ;  
  ret
```



ESP

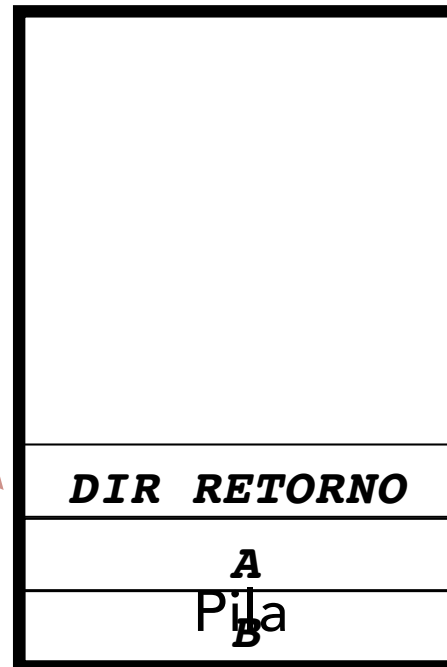
ORGANIZACIÓN DEL PROCESADOR

SUBROUTINAS - INTERCAMBIO DE INFORMACIÓN

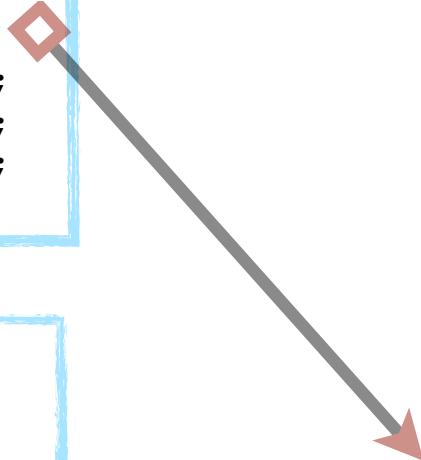
...
Sub1 (a,b);
...

```
...  
mov di, OpA      ;  
Call NewLine     ;  
mov dx, Msg7     ;  
call SUB1       ;  
mov cx, 10       ;  
mov di, OpB      ;  
Call NewLine     ;  
mov si, OpA      ;  
...
```

```
SUB1:  
  mov si, prd      ;  
  mov cx, 20       ;  
  mov si, Res      ;  
  ret
```



ESP



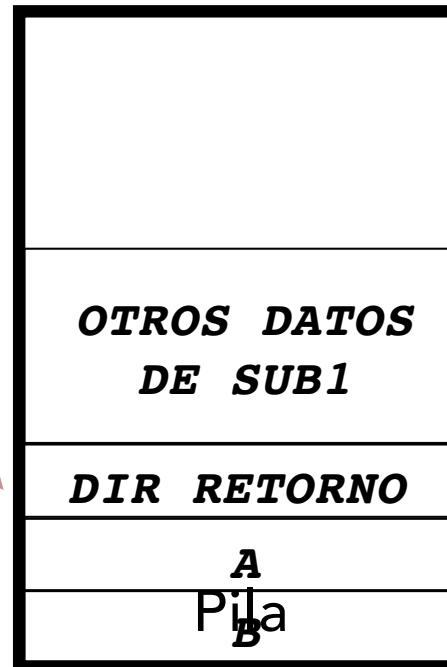
ORGANIZACIÓN DEL PROCESADOR

SUBROUTINAS - INTERCAMBIO DE INFORMACIÓN

...
Sub1 (a,b);
...

```
...  
mov di, OpA      ;  
Call NewLine    ;  
mov dx, Msg7     ;  
call SUB1      ;  
mov cx, 10       ;  
mov di, OpB      ;  
Call NewLine    ;  
mov si, OpA      ;  
...
```

```
SUB1:  
  mov si, prd      ;  
  mov cx, 20       ;  
  mov si, Res      ;  
  ret
```



ESP

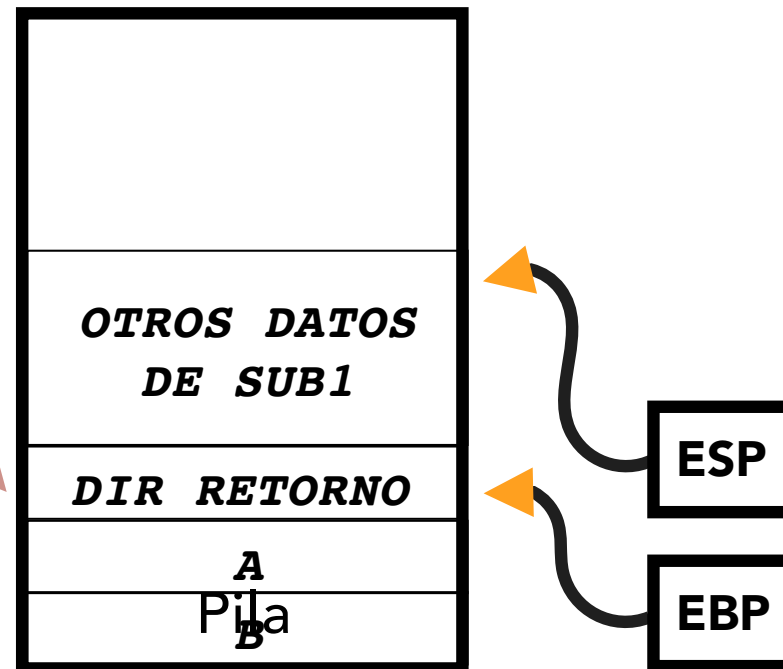
ORGANIZACIÓN DEL PROCESADOR

SUBROUTINAS - INTERCAMBIO DE INFORMACIÓN

...
Sub1 (a,b);
...

```
...  
mov di, OpA      ;  
Call NewLine     ;  
mov dx, Msg7     ;  
call SUB1       ;  
mov cx, 10       ;  
mov di, OpB      ;  
Call NewLine     ;  
mov si, OpA      ;  
...
```

```
SUB1:  
  mov si, prd      ;  
  mov cx, 20       ;  
  mov si, Res      ;  
  ret
```



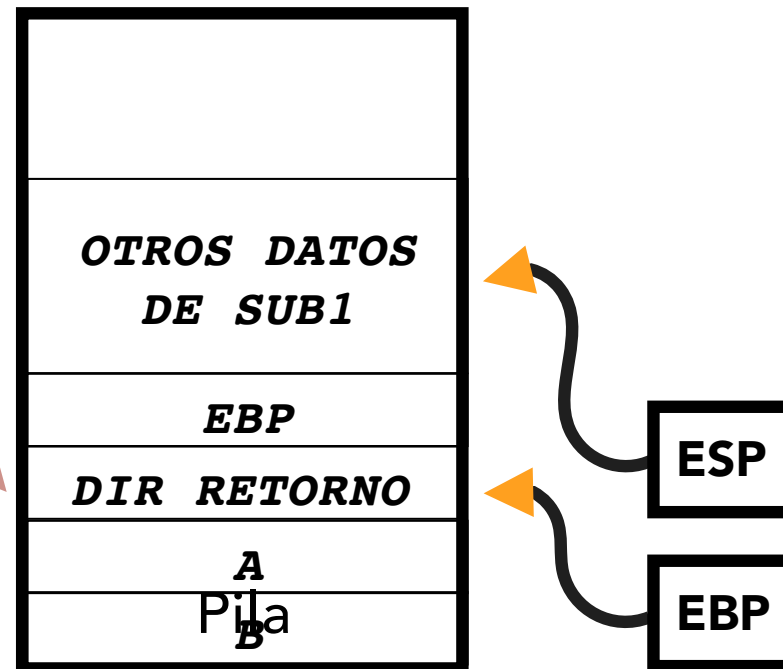
ORGANIZACIÓN DEL PROCESADOR

SUBROUTINAS - INTERCAMBIO DE INFORMACIÓN

```
...  
Sub1 (a,b);  
...
```

```
...  
mov di, OpA      ;  
Call NewLine    ;  
mov dx, Msg7     ;  
call SUB1      ;  
mov cx, 10       ;  
mov di, OpB      ;  
Call NewLine    ;  
mov si, OpA      ;  
...
```

```
SUB1:  
  mov si, prd      ;  
  mov cx, 20       ;  
  mov si, Res      ;  
  ret
```



ORGANIZACIÓN DEL PROCESADOR

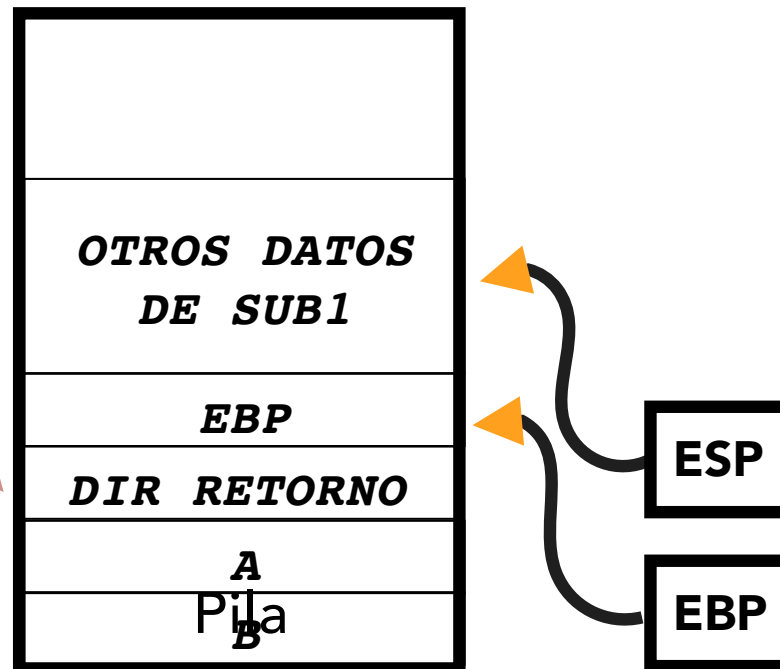
SUBROUTINAS - INTERCAMBIO DE INFORMACIÓN

```
...  
Sub1 (a,b);  
...
```

```
...  
mov di, OpA      ;  
Call NewLine    ;  
mov dx, Msg7     ;  
call SUB1      ;  
mov cx, 10       ;  
mov di, OpB      ;  
Call NewLine    ;  
mov si, OpA      ;  
...
```

```
SUB1:  
  push ebp  
  mov ebp esp  
  mov si, prd      ;  
  mov cx, 20       ;  
  mov si, Res      ;  
  pop ebp  
  ret
```

Toda subrutina comienza por poner el EBP en la Pila y mover al EBP el ESP. Al finalizar debe restaurar el EBP. Los parámetros estarán en: EBP+8,EBP+12...



ORGANIZACIÓN DEL PROCESADOR

SUBROUTINAS - VARIABLES LOCALES

```
...  
Sub1 (a,b);  
int x,y;  
...
```

```
...  
mov di, OpA      ;  
Call NewLine    ;  
mov dx, Msg7     ;  
call SUB1      ;  
mov cx, 10       ;  
mov di, OpB      ;  
Call NewLine    ;  
mov si, OpA      ;  
...
```

SUB1:

```
enter <tamaño>, 0  
mov si, prd      ;  
mov cx, 20       ;  
mov si, Res      ;  
leave  
ret
```

Las variables locales se
encontrarán en
EBP-4,EBP-8...

