

Example from yesterday

- subtract smallest element from all elements in an array ending with 0; all elements > 0

• text

• global -start

```

-start:  movia  r16, A
        ldw   r17, 0(r16)

```

findmin:

```

ldw   r18, 0(r16)

```

```

beq   r18, r0, subtract

```

```

blt   r17, r18, skipmin

```

```

mov   r17, r18

```

skipmin:

```

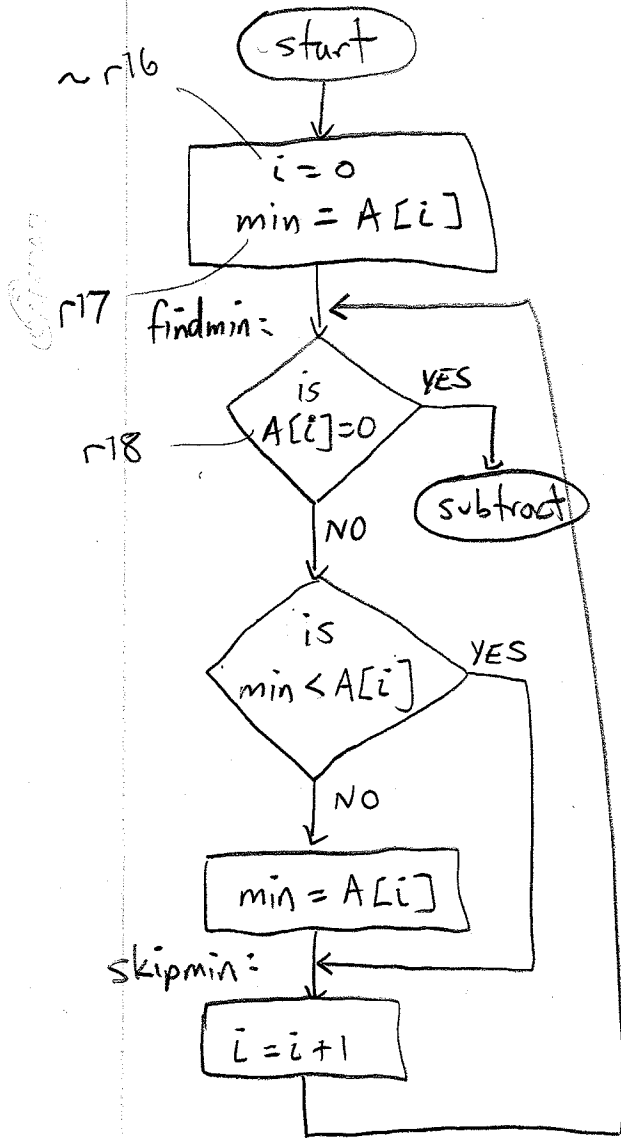
addi  r16, r16, 4

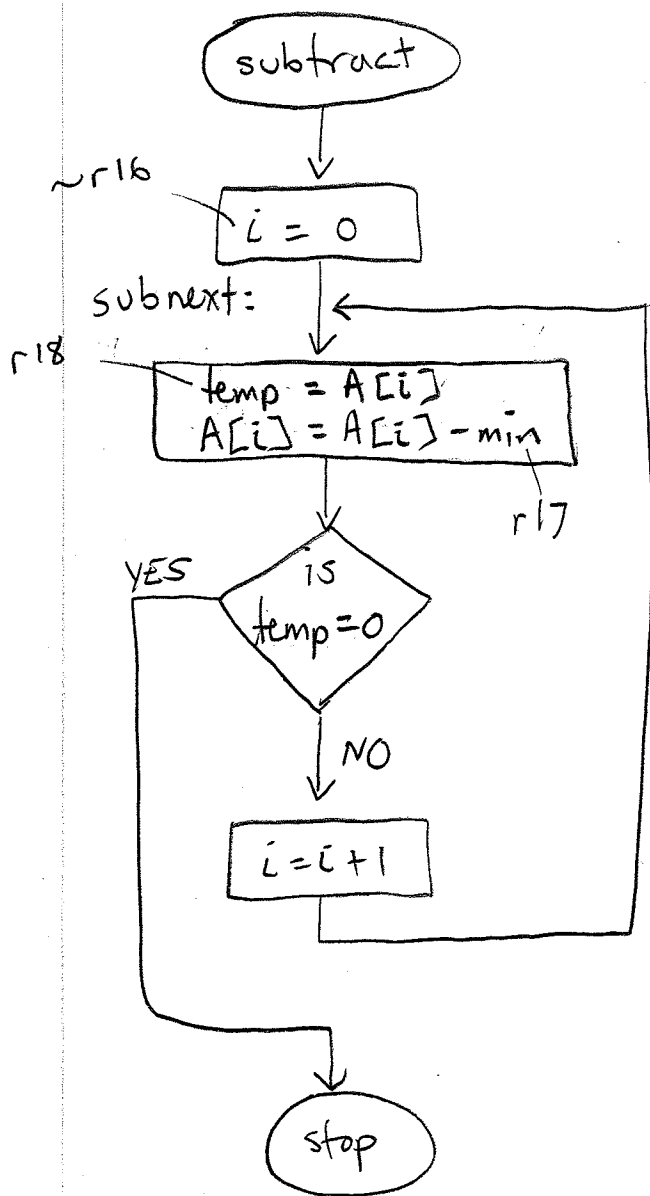
```

```

br    findmin

```





subtract:

```
movia r16, A
```

subnext:

```
ldw r18, 0(r16)
```

```
sub r19, r17, r18
```

```
stw r19, 0(r16)
```

```
beq r18, r0, STOP
```

```
addi r16, r16, 4
```

```
br subnext
```

```
STOP: br STOP
```

NOTE: the error
(when $A[0] = 0$)
is fixed!

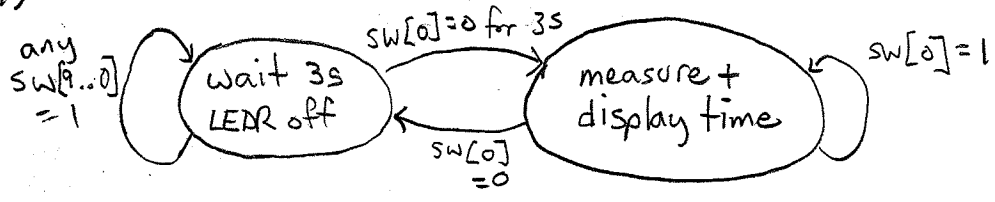
```

.data
A:
.word 14, 8, 3, 16, 120, 13, 0
.end
  
```

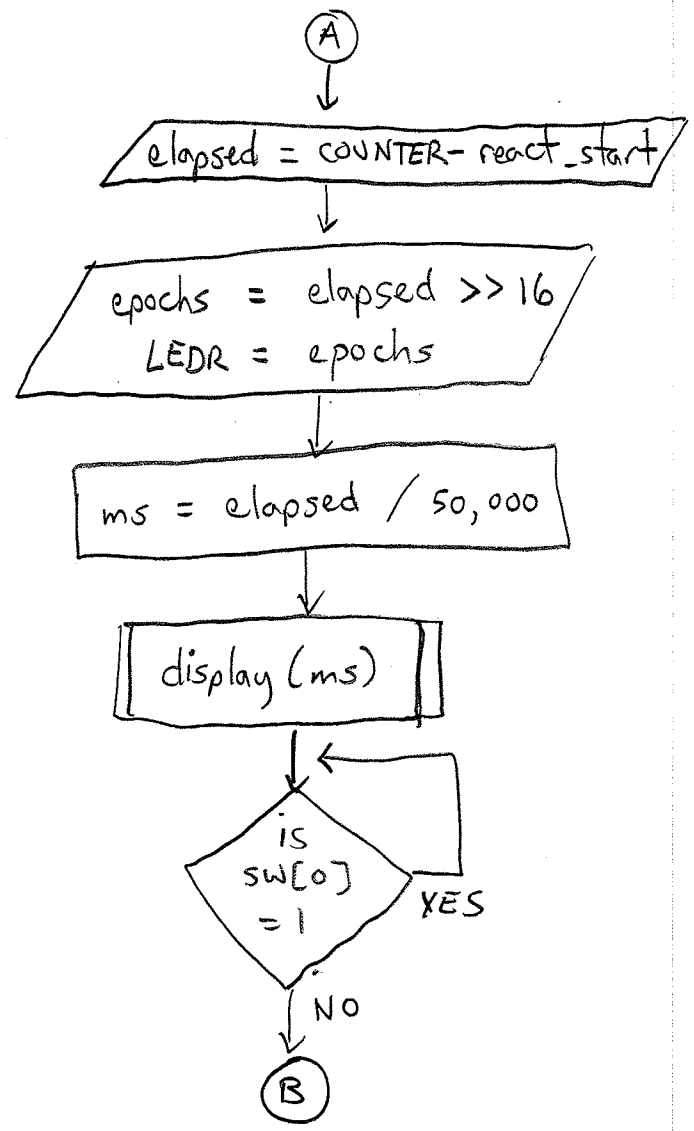
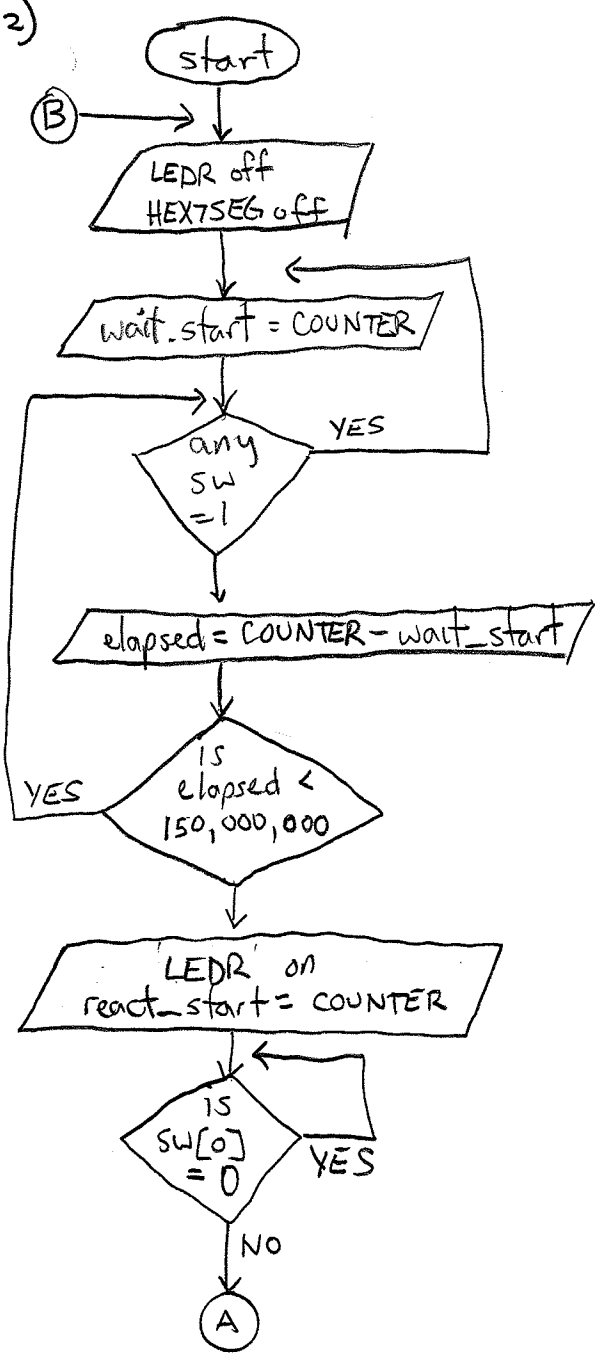
Example : practical assignment 2 reaction timer

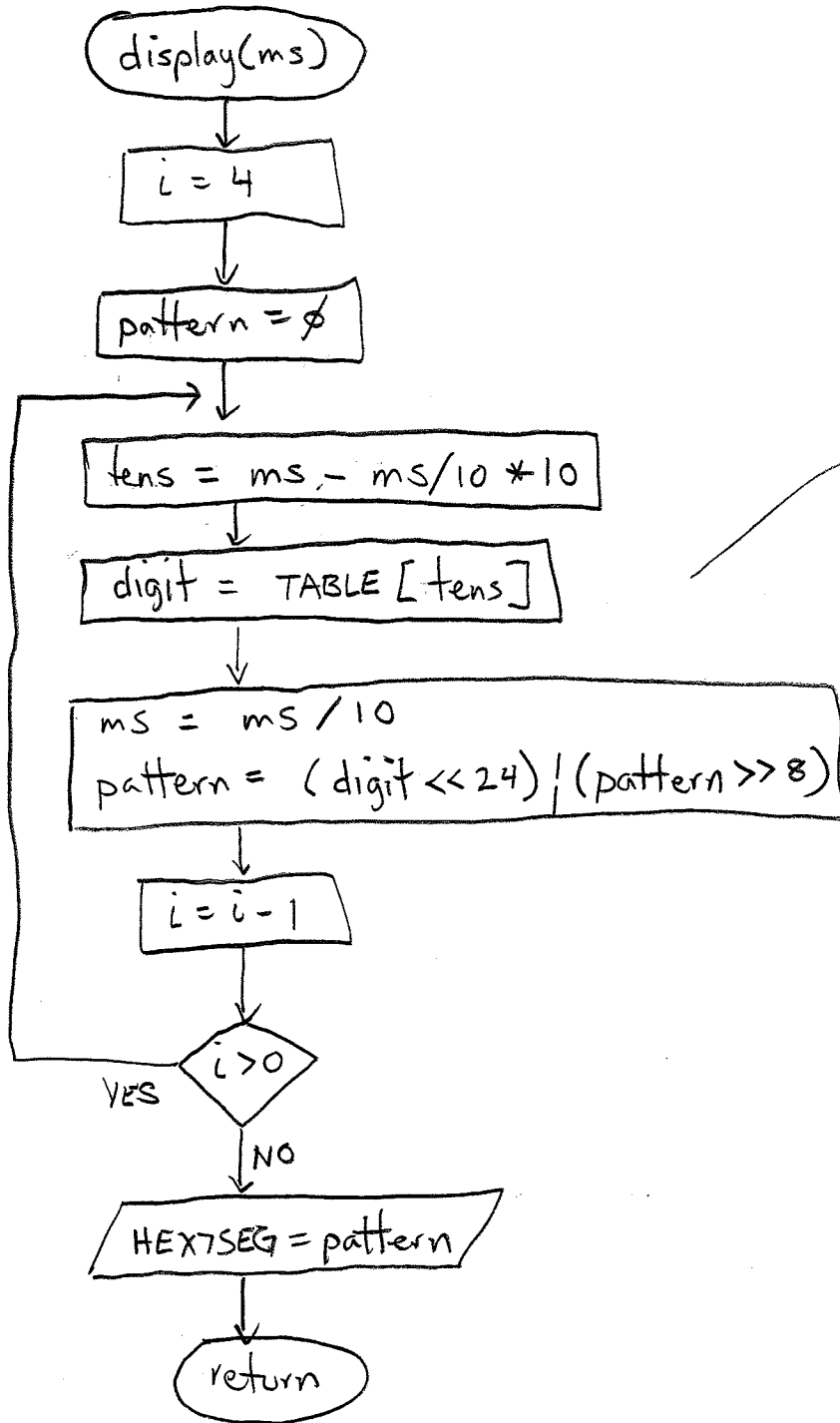
- approach to solution :
- 1) state transition diagram
 - 2) flow chart
 - 3) assembly code (not given)

1)



2)





TABLE[0] = DIGIT0
[1] = DIGIT1
⋮
[9] = DIGIT9