Timekeeper

- one volunteer keeps the time spent *programming*
- one volunteer keeps the time spent *executing* the program

programmer

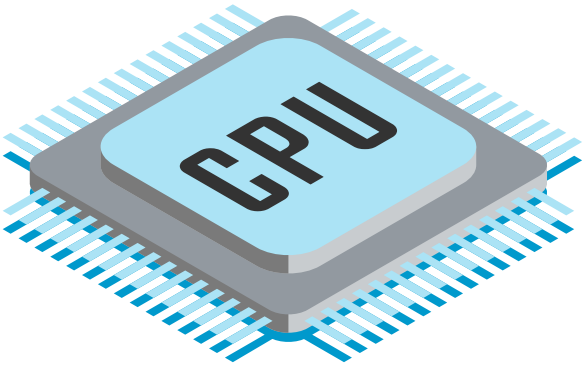
- can use the whiteboard
- can and should speak and think out loud and ask for **help**

```
41
42  # programmer function
43  def I_can_talk(volunteer, words, thoughts):
44      '''
45      your voice will be heard
46      let it be meaningful
47      '''
48      if volunteer:
49          for i in words:
50              print(i)
51      if 'listen' in words or 'Listen' in words:
52          print('Oh boy this is not working!')
53      if 'disaster' in words  or 'Disaster'in words:
54          print('I should bite the bullet!')
55      if 'Give up' in words or 'give up' in words or 'gi
56          print('Game Over! \n [u wish]')
57      if 'smart' in thoughts or 'Smart' in thoughts:
58          print('Time to show off!')
59
```

CPU

- only understand the instructions:
  - **fetch** a value from a memory address into register N ➔ returns 0 if succeded else 1
  - **push** the value from register N to a memory address ➔ returns 0 if succeded else 1
  - **compare** var0 and var1 ➔ returns 0 if var0 ≥ var1 else 1

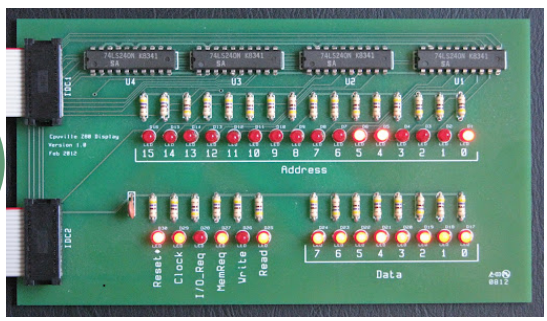
Fetch, push, compare



CPU register

- a value fetched from memory is kept in short-term memory by the registers
- the result value of an operation is stored in one register

R1



Timekeeper

- one volunteer keeps the time spent *programming*
- one volunteer keeps the time spent *executing* the program

programmer

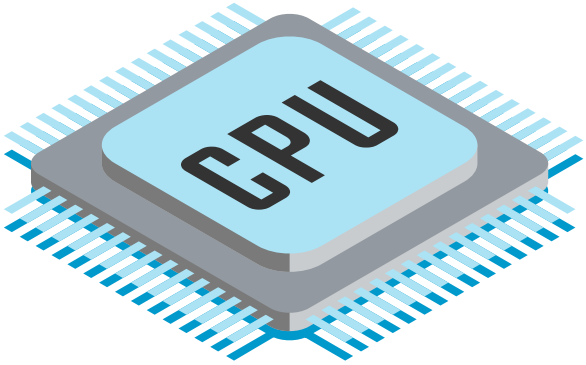
- can use the whiteboard
- can and should speak and think out loud and ask for **help**

```
41
42  # programmer function
43  def I_can_talk(volunteer, words, thoughts):
44      '''
45      your voice will be heard
46      let it be meaningful
47      '''
48      if volunteer:
49          for i in words:
50              print(i)
51      if 'listen' in words or 'Listen' in words:
52          print('Oh boy this is not working!')
53      if 'disaster' in words  or 'Disaster'in words:
54          print('I should bite the bullet!')
55      if 'Give up' in words or 'give up' in words or 'gi
56          print('Game Over! \n [u wish]')
57      if 'smart' in thoughts or 'Smart' in thoughts:
58          print('Time to show off!')
59
```

CPU

- only understand the instructions:
  - **fetch** a value from a memory address into register N ➔ returns 0 if succeded else 1
  - **push** the value from register N to a memory address ➔ returns 0 if succeded else 1
  - **compare** var0 and var1 ➔ returns 0 if var0 ≥ var1 else 1

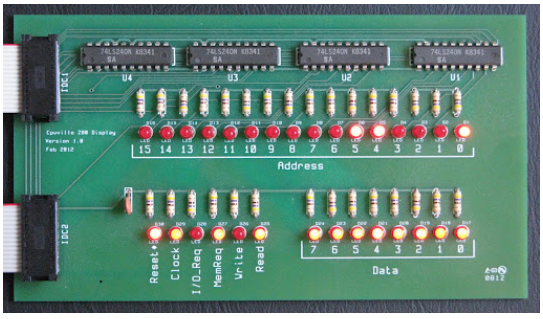
fetch, push, compare



CPU register

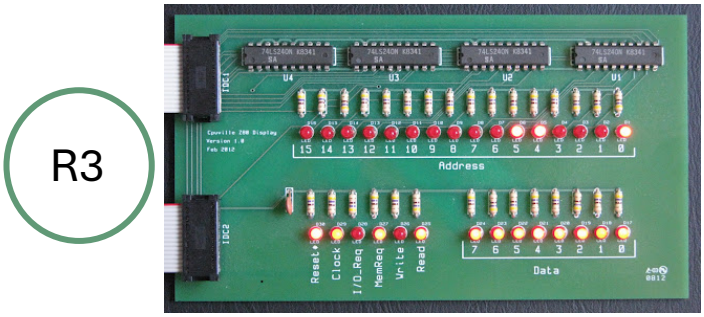
- a value fetched from memory is kept in short-term memory by the registers
- the result value of an operation is stored in one register

R2



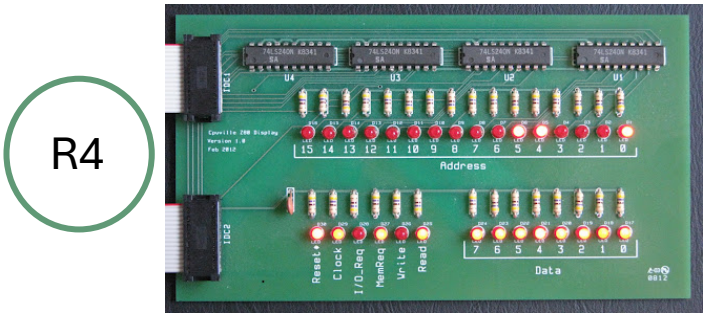
CPU register

- a value fetched from memory is kept in short-term memory by the registers
- the result value of an operation is stored in one register



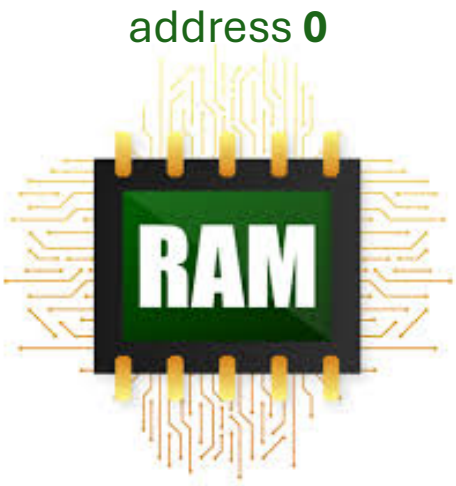
CPU register

- each register has a tag: R1, R2, R3, R4
- a value fetched from memory is kept in short-term memory by the registers
- the result value of an operation is stored in one register



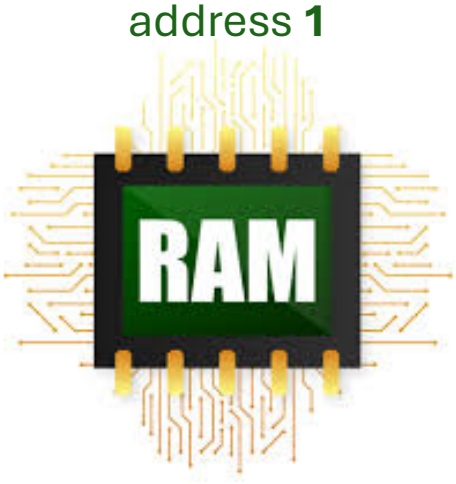
RAM

- own a *value*, i.e. they hold on a tarot card
- have an address based on their seating order: 0th seat, 1st seat, 2nd seat, 3rd seat, 4th seat, etc...
- when *fetched*, walk to the corresponding register and hand in their *value* (card)
- when *pushed*, walk to the corresponding register and fetch their new *value* (card)



RAM

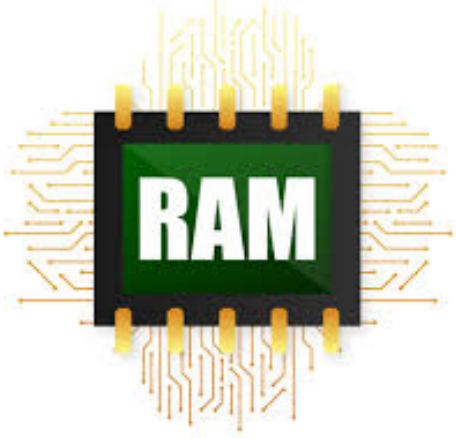
- own a *value*, i.e. they hold on a tarot card
- have an address based on their seating order: 0th seat, 1st seat, 2nd seat, 3rd seat, 4th seat, etc...
- when *fetched*, walk to the corresponding register and hand in their *value* (card)
- when *pushed*, walk to the corresponding register and fetch their new *value* (card)



RAM

- own a *value*, i.e. they hold on a tarot card
- have an address based on their seating order: 0th seat, 1st seat, 2nd seat, 3rd seat, 4th seat, etc...
- when *fetched*, walk to the corresponding register and hand in their *value* (card)
- when *pushed*, walk to corresponding register and fetch their new *value* (card)

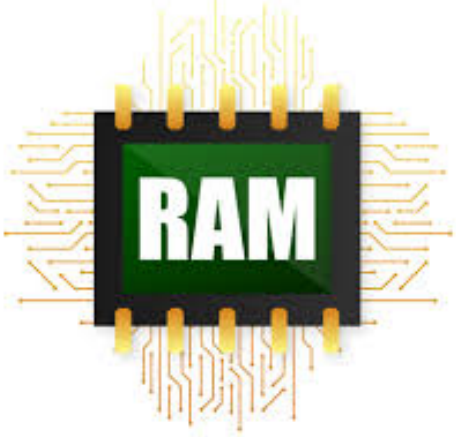
address **2**



RAM

- own a *value*, i.e. they hold on a tarot card
- have an address based on their seating order: 0th seat, 1st seat, 2nd seat, 3rd seat, 4th seat, etc...
- when *fetched*, walk to the corresponding register and hand in their *value* (card)
- when *pushed*, walk to corresponding register and fetch their new *value* (card)

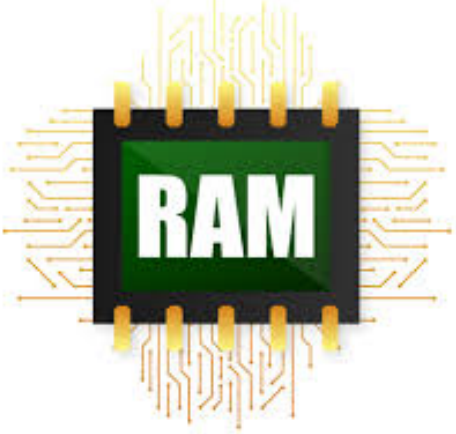
address **3**



RAM

- own a *value*, i.e. they hold on a tarot card
- have an address based on their seating order: 0th seat, 1st seat, 2nd seat, 3rd seat, 4th seat, etc...
- when *fetched*, walk to the corresponding register and hand in their *value* (card)
- when *pushed*, walk to corresponding register and fetch their new *value* (card)

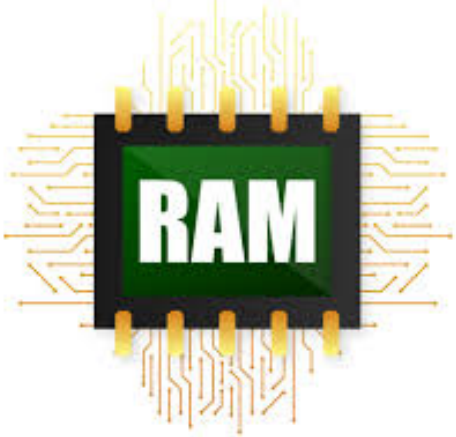
address **4**



RAM

- own a *value*, i.e. they hold on a tarot card
- have an address based on their seating order: 0th seat, 1st seat, 2nd seat, 3rd seat, 4th seat, etc...
- when *fetched*, walk to the corresponding register and hand in their *value* (card)
- when *pushed*, walk to corresponding register and fetch their new *value* (card)

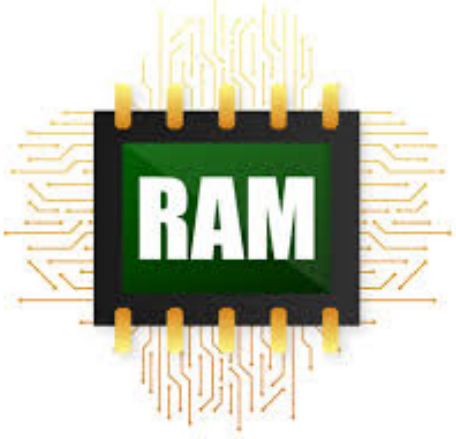
address **5**



RAM

- own a *value*, i.e. they hold on a tarot card
- have an address based on their seating order: 0th seat, 1st seat, 2nd seat, 3rd seat, 4th seat, etc...
- when *fetched*, walk to the corresponding register and hand in their *value* (card)
- when *pushed*, walk to corresponding register and fetch their new *value* (card)

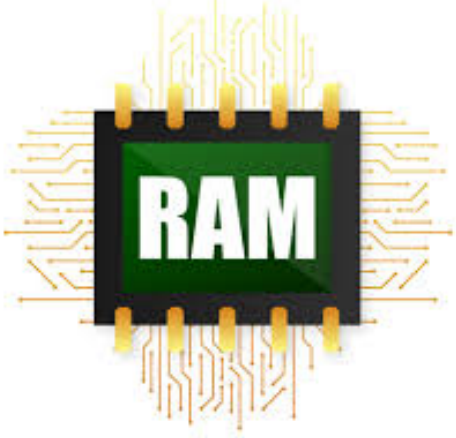
address **6**



RAM

- own a *value*, i.e. they hold on a tarot card
- have an address based on their seating order: 0th seat, 1st seat, 2nd seat, 3rd seat, 4th seat, etc...
- when *fetched*, walk to the corresponding register and hand in their *value* (card)
- when *pushed*, walk to corresponding register and fetch their new *value* (card)

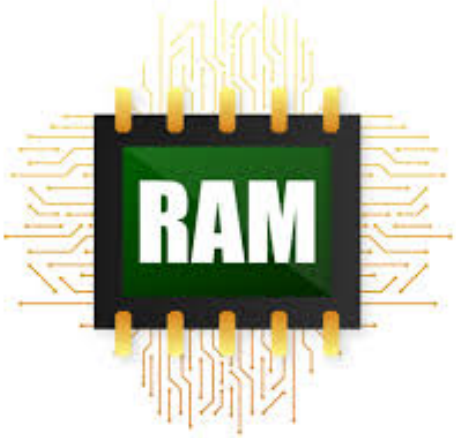
address **7**



RAM

- own a *value*, i.e. they hold on a tarot card
- have an address based on their seating order: 0th seat, 1st seat, 2nd seat, 3rd seat, 4th seat, etc...
- when *fetched*, walk to the corresponding register and hand in their *value* (card)
- when *pushed*, walk to corresponding register and fetch their new *value* (card)

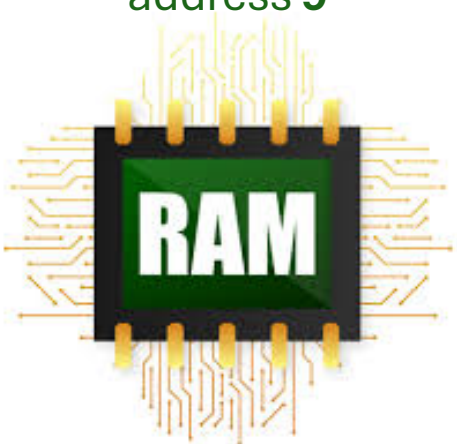
address **8**



RAM

- own a *value*, i.e. they hold on a tarot card
- have an address based on their seating order: 0th seat, 1st seat, 2nd seat, 3rd seat, 4th seat, etc...
- when *fetched*, walk to the corresponding register and hand in their *value* (card)
- when *pushed*, walk to corresponding register and fetch their new *value* (card)

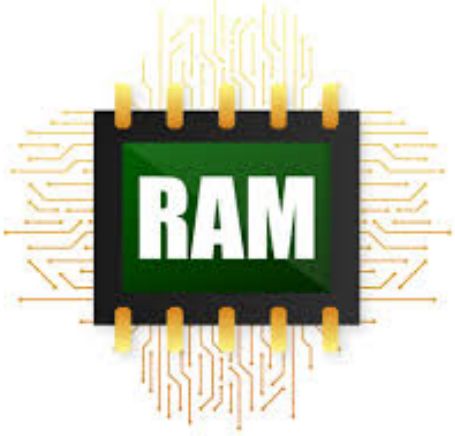
address **9**



RAM

- own a *value*, i.e. they hold on a tarot card
- have an address based on their seating order: 0th seat, 1st seat, 2nd seat, 3rd seat, 4th seat, etc...
- when *fetched*, walk to the corresponding register and hand in their *value* (card)
- when *pushed*, walk to corresponding register and fetch their new *value* (card)

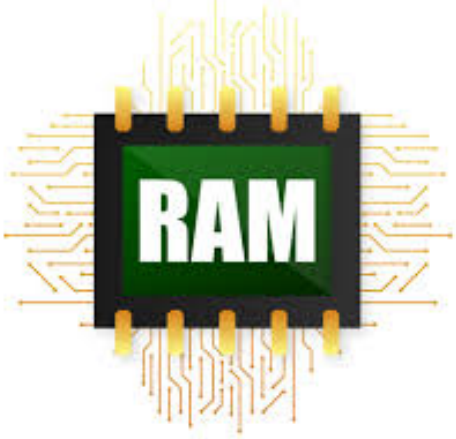
address **10**



RAM

- own a *value*, i.e. they hold on a tarot card
- have an address based on their seating order: 0th seat, 1st seat, 2nd seat, 3rd seat, 4th seat, etc...
- when *fetched*, walk to the corresponding register and hand in their *value* (card)
- when *pushed*, walk to corresponding register and fetch their new *value* (card)

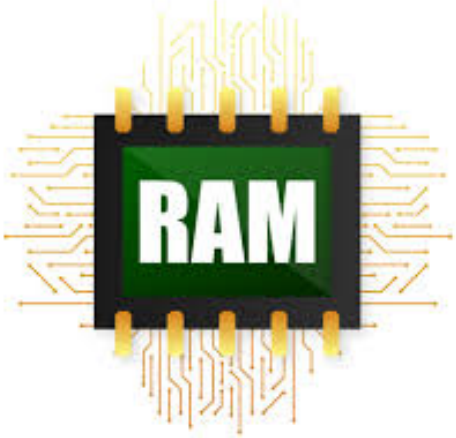
address **11**



RAM

- own a *value*, i.e. they hold on a tarot card
- have an address based on their seating order: 0th seat, 1st seat, 2nd seat, 3rd seat, 4th seat, etc...
- when *fetched*, walk to the corresponding register and hand in their *value* (card)
- when *pushed*, walk to corresponding register and fetch their new *value* (card)

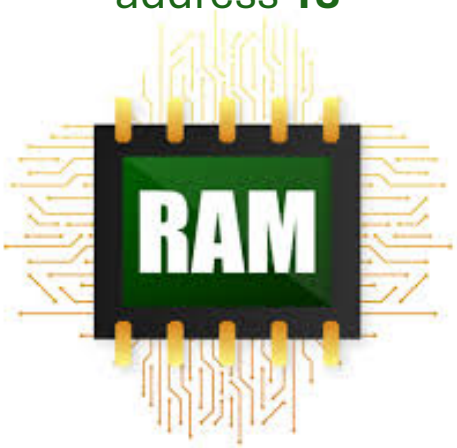
address **12**



RAM

- own a *value*, i.e. they hold on a tarot card
- have an address based on their seating order: 0th seat, 1st seat, 2nd seat, 3rd seat, 4th seat, etc...
- when *fetched*, walk to the corresponding register and hand in their *value* (card)
- when *pushed*, walk to corresponding register and fetch their new *value* (card)

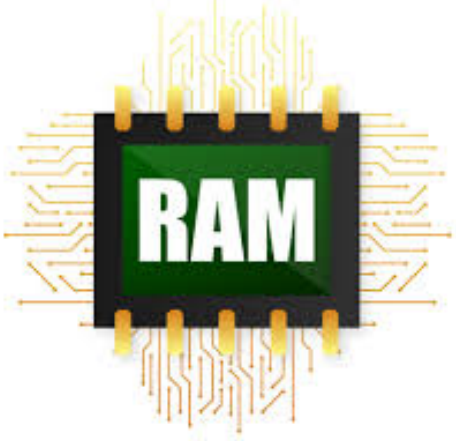
address **13**



RAM

- own a *value*, i.e. they hold on a tarot card
- have an address based on their seating order: 0th seat, 1st seat, 2nd seat, 3rd seat, 4th seat, etc...
- when *fetched*, walk to the corresponding register and hand in their *value* (card)
- when *pushed*, walk to corresponding register and fetch their new *value* (card)

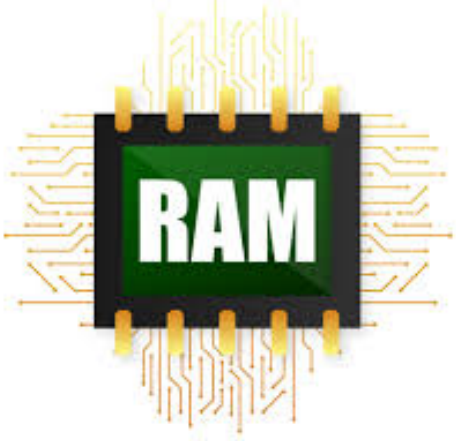
address **14**



RAM

- own a *value*, i.e. they hold on a tarot card
- have an address based on their seating order: 0th seat, 1st seat, 2nd seat, 3rd seat, 4th seat, etc...
- when *fetched*, walk to the corresponding register and hand in their *value* (card)
- when *pushed*, walk to corresponding register and fetch their new *value* (card)

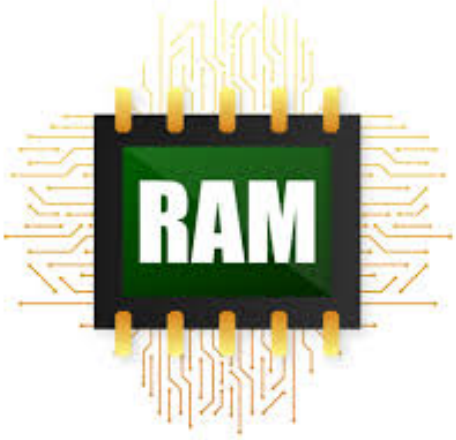
address **15**



RAM

- own a *value*, i.e. they hold on a tarot card
- have an address based on their seating order: 0th seat, 1st seat, 2nd seat, 3rd seat, 4th seat, etc...
- when *fetched*, walk to the corresponding register and hand in their *value* (card)
- when *pushed*, walk to corresponding register and fetch their new *value* (card)

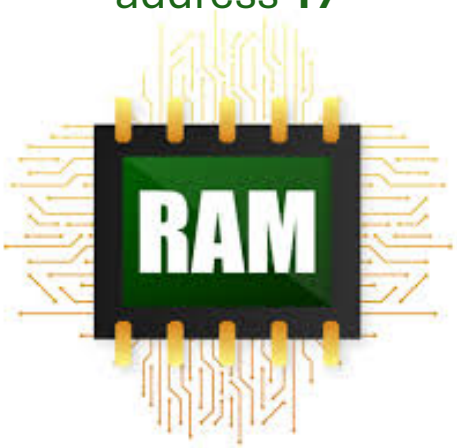
address **16**



RAM

- own a *value*, i.e. they hold on a tarot card
- have an address based on their seating order: 0th seat, 1st seat, 2nd seat, 3rd seat, 4th seat, etc...
- when *fetched*, walk to the corresponding register and hand in their *value* (card)
- when *pushed*, walk to corresponding register and fetch their new *value* (card)

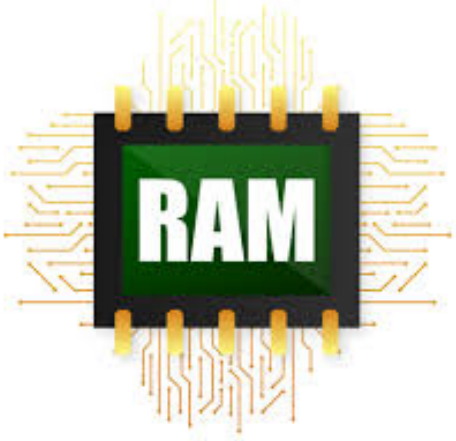
address **17**



RAM

- own a *value*, i.e. they hold on a tarot card
- have an address based on their seating order: 0th seat, 1st seat, 2nd seat, 3rd seat, 4th seat, etc...
- when *fetched*, walk to the corresponding register and hand in their *value* (card)
- when *pushed*, walk to corresponding register and fetch their new *value* (card)

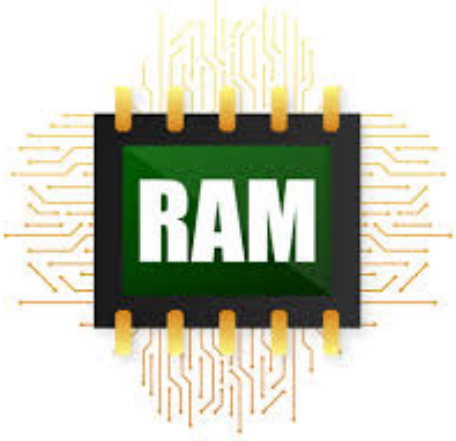
address **18**



RAM

- own a *value*, i.e. they hold on a tarot card
- have an address based on their seating order: 0th seat, 1st seat, 2nd seat, 3rd seat, 4th seat, etc...
- when *fetched*, walk to the corresponding register and hand in their *value* (card)
- when *pushed*, walk to corresponding register and fetch their new *value* (card)

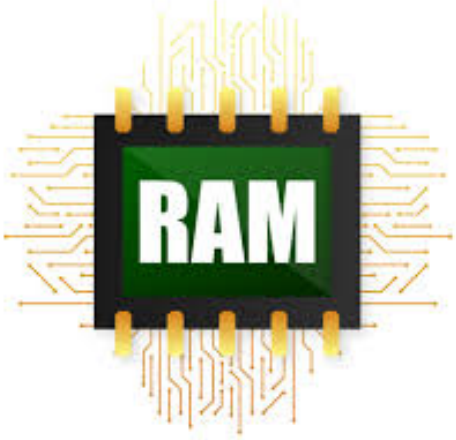
address **19**



RAM

- own a *value*, i.e. they hold on a tarot card
- have an address based on their seating order: 0th seat, 1st seat, 2nd seat, 3rd seat, 4th seat, etc...
- when *fetched*, walk to the corresponding register and hand in their *value* (card)
- when *pushed*, walk to corresponding register and fetch their new *value* (card)

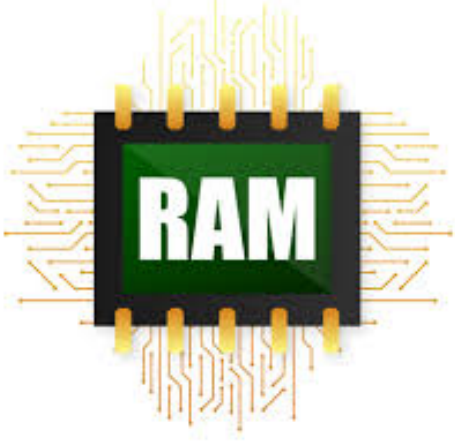
address **20**



RAM

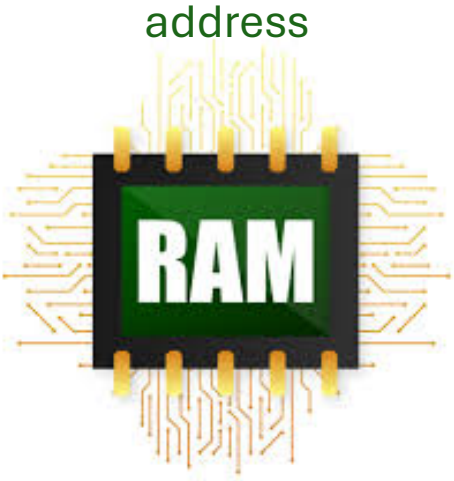
- own a *value*, i.e. they hold on a tarot card
- have an address based on their seating order: 0th seat, 1st seat, 2nd seat, 3rd seat, 4th seat, etc...
- when *fetched*, walk to the corresponding register and hand in their *value* (card)
- when *pushed*, walk to corresponding register and fetch their new *value* (card)

address



RAM

- own a *value*, i.e. they hold on a tarot card
- have an address based on their seating order: 0th seat, 1st seat, 2nd seat, 3rd seat, 4th seat, etc...
- when *fetched*, walk to the corresponding register and hand in their *value* (card)
- when *pushed*, walk to corresponding register and fetch their new *value* (card)



RAM

- own a *value*, i.e. they hold on a tarot card
- have an address based on their seating order: 0th seat, 1st seat, 2nd seat, 3rd seat, 4th seat, etc...
- when *fetched*, walk to the corresponding register and hand in their *value* (card)
- when *pushed*, walk to corresponding register and fetch their new *value* (card)