

Maksimum, minimum

- ① Znaleźć maksymalną wartość w tablicy 16-bitowych (*half*: 2B) liczb całkowitych w pamięci w zakresie 0x200 . . . 0x3FF. Liczby traktowane są jako *unsigned int*.

Wynik ma być zwracany w rejestrze R1.

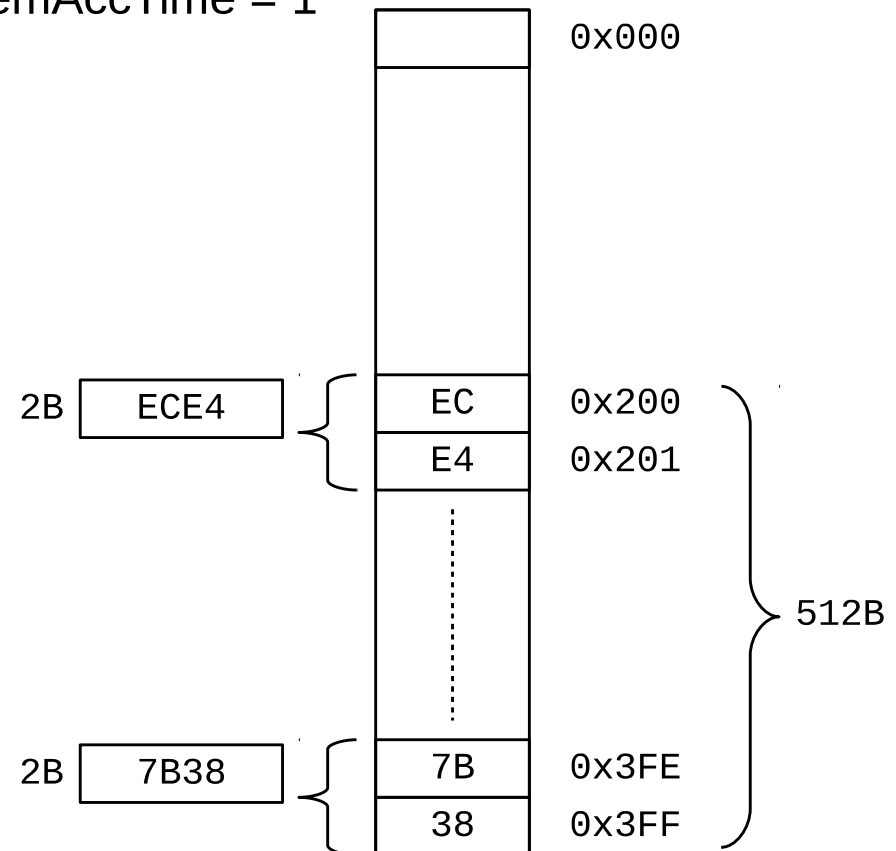
Zmierz czas działania programu przy MemAccTime = 1

Pliki znajdują się w katalogu lab1.

- wczytać konfigurację lab1.ecf
- wczytać projekt lab1.mpr:
 - lab1.mco
 - lab1.cod
 - random.dat

- ② Znaleźć maksymalną i minimalną wartość w tej samej tablicy.

Wyniki w rejestrach R1, R2.



Sortowanie przez wybieranie

- ③ Posortować (in-situ) tablicę 16-bitowych (*half*: 2B) liczb całkowitych w pamięci w zakresie $0x200 \dots 0x3FF$. Liczby traktowane są jako *unsigned int*.

Zmierz czas działania programu (MemAccTime = 1), oczekuj ponad 700,000 cykli.

Pliki znajdują się w katalogu lab1.

- wczytać konfigurację: lab1.ecf
- utwórz nowy projekt: sort.mpr

Proponowana struktura programu:

inicjowanie wskaźnika pętli głównej Rx=0x200

pętla główna
dekrementacja Rx

inicjowanie wskaźnika pętli wewnętrznej Ry=Rx
pętla wewnętrzna
dekrementacja Ry
sprawdzanie czy maksimum
jeśli Ry=0 to koniec

zamiana liczb w tablicy: a[Rx] i b[Rmax]

jeśli Rx=0 to koniec

