

Exercice 1 Adressage de 32 bits

Si l'adresse de la mémoire physique est de 32 bits, si la cache (à correspondance directe) est de 256 KIB (1 KIB = 2¹⁰ octets), et si les blocs sont de 128 octets, quel est le nombre de bits des champs « étiquette », « index » et « offset » de l'adresse mémoire ?



Adressage 32 bits

Cache size = 256 KIB $\Rightarrow 2^{18}$ B

Block size = 128 B $\Rightarrow 2^7$ B

bits offset = $\log_2 2^7 \Rightarrow 7$ bits

sets/index: $\frac{2^{18}}{2^7} = 2^{11}$ blocs dans le cache

bits index: $\log_2 2^{11} = 11$ bits

bits tag: $32 - (11 + 7) = 14$ bits

tag: 14 bits
index: 11 bits
offset: 7 bits

tag	index	offset
14 bits	11 bits	7 bits
32 bits		

Cache size = 128 B $\Rightarrow 2^7$

12 bits addressing

block size = 32 B $\Rightarrow 2^5$

offset: $\log_2 2^5 = 5$ bits

index: $\log_2 \left(\frac{2^7}{2^5} \right) = \log_2 2^2 = 2$ bits

tag: $12 - (5 + 2) = 5$ bits

tag	index	offset
5	2	5



0x070



Cache		
00		
01		
10	0000	0000
11	1000	1100
index	offset	tag

cache blocks $\frac{2^7}{2^5} = 2^2 = 4$ blocks

Exercice 2

On a une mémoire dont le temps de latence, i.e. le temps moyen pour trouver le premier mot, est de 40 ns ; le temps de transfert des mots suivants est 10 ns chacun.

- a) Quel est le temps (espéré) de chargement d'une ligne (le temps pour copier une ligne dans le cache) si le cache a des lignes de 16 mots ?
- b) Quel est le temps de chargement si les lignes sont de 8 mots ?

latency 40ns

10ns

$$a) 1 \times 40 + 15 \times 10 = 190 \text{ ns}$$

$$b) 1 \times 40 + 7 \times 10 = 110 \text{ ns}$$

Exercice 3

Un programme consiste en deux blocs indicés, soit la structure générale suivante :

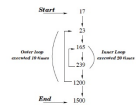


FIGURE 1 - Structure générale d'un certain programme avec deux blocs indicés

Les adresses mémoire sont en décimal. À l'exécution des deux blocs aux adresses 220 et 1200, toutes les instructions utilisent le *straight-line sequencing* et chaque adresse contient une instruction. On exécute le programme sur un ordinateur avec la performance suivante :

Taille de la mémoire principale	64 K
Taille de la mémoire cache	1 K
Taille d'un bloc	128

Un cycle dans la mémoire principale prends 10 ns et un cycle dans la mémoire cache prends 1 ns (10^{-9} seconde).

- Déterminer le nombre de bits dans les champs TAG, BLOCK et WORD (TAG, INDEX et OFFSET) d'une adresse mémoire.
- Calculer le nombre de temps total nécessaire pour l'accès aux instructions (*instruction fetching*) durant l'exécution du programme.

o) *Conversion direct*
 m.p. : 64 Kb $\Rightarrow 2^6 \times 2^{10} \Rightarrow 2^{16}$ B
 k.c. : 1 Kb $\Rightarrow 2^0 \times 2^{10} \Rightarrow 2^{10}$ B
 b.c. : 128 B $\Rightarrow 2^7$ B

bits offset : $\log_2 2^7 \Rightarrow 7$ bits
 # cache block : $\frac{2^{10}}{2^7} = 2^3 = 8$ blocs
 # bits index : $\log_2 8 \Rightarrow 3$ bits

Memory block : $\frac{2^{16}}{2^7} \Rightarrow 2^9$ blocs
 # bits TAG : $\log_2 2^9 = 9$ bits

bits TAG = TAG - C.I.
 = 9 - 3
 = 6 bits

Mémoire	
0	
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	
20	
21	
22	
23	
24	
25	
26	
27	
28	
29	
30	
31	
32	
33	
34	
35	
36	
37	
38	
39	
40	
41	
42	
43	
44	
45	
46	
47	
48	
49	
50	
51	
52	
53	
54	
55	
56	
57	
58	
59	
60	
61	
62	
63	
64	
65	
66	
67	
68	
69	
70	
71	
72	
73	
74	
75	
76	
77	
78	
79	
80	
81	
82	
83	
84	
85	
86	
87	
88	
89	
90	
91	
92	
93	
94	
95	
96	
97	
98	
99	
100	
101	
102	
103	
104	
105	
106	
107	
108	
109	
110	
111	
112	
113	
114	
115	
116	
117	
118	
119	
120	
121	
122	
123	
124	
125	
126	
127	
128	
129	
130	
131	
132	
133	
134	
135	
136	
137	
138	
139	
140	
141	
142	
143	
144	
145	
146	
147	
148	
149	
150	
151	
152	
153	
154	
155	
156	
157	
158	
159	
160	
161	
162	
163	
164	
165	
166	
167	
168	
169	
170	
171	
172	
173	
174	
175	
176	
177	
178	
179	
180	
181	
182	
183	
184	
185	
186	
187	
188	
189	
190	
191	
192	
193	
194	
195	
196	
197	
198	
199	
200	
201	
202	
203	
204	
205	
206	
207	
208	
209	
210	
211	
212	
213	
214	
215	
216	
217	
218	
219	
220	
221	
222	
223	
224	
225	
226	
227	
228	
229	
230	
231	
232	
233	
234	
235	
236	
237	
238	
239	
240	
241	
242	
243	
244	
245	
246	
247	
248	
249	
250	
251	
252	
253	
254	
255	
256	
257	
258	
259	
260	
261	
262	
263	
264	
265	
266	
267	
268	
269	
270	
271	
272	
273	
274	
275	
276	
277	
278	
279	
280	
281	
282	
283	
284	
285	
286	
287	
288	
289	
290	
291	
292	
293	
294	
295	
296	
297	
298	
299	
300	
301	
302	
303	
304	
305	
306	
307	
308	
309	
310	
311	
312	
313	
314	
315	
316	
317	
318	
319	
320	
321	
322	
323	
324	
325	
326	
327	
328	
329	
330	
331	
332	
333	
334	
335	
336	
337	
338	
339	
340	
341	
342	
343	
344	
345	
346	
347	
348	
349	
350	
351	
352	
353	
354	
355	
356	
357	
358	
359	
360	
361	
362	
363	
364	
365	
366	
367	
368	
369	
370	
371	
372	
373	
374	
375	
376	
377	
378	
379	
380	
381	
382	
383	
384	
385	
386	
387	
388	
389	
390	
391	
392	
393	
394	
395	
396	
397	
398	
399	
400	
401	
402	
403	
404	
405	
406	
407	
408	
409	
410	
411	
412	
413	
414	
415	
416	
417	
418	
419	
420	
421	
422	
423	
424	
425	
426	
427	
428	
429	
430	
431	
432	
433	
434	
435	
436	
437	
438	
439	
440	
441	
442	
443	
444	
445	
446	
447	
448	
449	
450	
451	
452	
453	
454	
455	
456	
457	
458	
459	
460	
461	
462	
463	
464	
465	
466	
467	
468	
469	
470	
471	
472	
473	
474	
475	
476	
477	
478	
479	
480	
481	
482	
483	
484	
485	
486	
487	
488	
489	
490	
491	
492	
493	
494	
495	
496	
497	
498	
499	
500	
501	
502	
503	
504	
505	
506	
507	
508	
509	
510	
511	
512	
513	
514	
515	
516	
517	
518	
519	
520	
521	
522	
523	
524	
525	
526	
527	
528	
529	
530	
531	
532	
533	
534	
535	
536	
537	
538	
539	
540	
541	
542	