Question 2 (20 points): A new processor, designed to be used in the toy industry, works with a 12-bit address. You need to determine the configuration of a cache after a certain number of memory references by this processor.

The cache that you are studying is a two-way set associative cache with 16-byte blocks and a capacity to store 256 bytes.

a. (**10 points**) How many bits will be used for tag, index, and offset for this cache design?

$$
\begin{aligned}
16\frac{\text{byte}}{\text{block}} &\Rightarrow \text{offset} = \log_2(16) = 4 \text{ bits} \\
\frac{\#\text{ bytes}}{\text{cache line}} &= \frac{16 \text{ bytes}}{\text{block}} \times \frac{2 \text{ blocks}}{\text{line}} = 32\frac{\text{bytes}}{\text{cache line}} \\
\#\text{ cache lines} &= \frac{256 \text{ bytes}}{32\frac{\text{bytes}}{\text{cache line}}} = 8 \text{ cache lines} \\
\text{index} &= \log_2(8) = 3 \text{ bits} \\
\text{Tag} &= \#\text{ address bits} - (\#\text{ offset bits} + \#\text{ index bits}) = 12 - (4+3) = 5 \text{ bits}
\end{aligned}
$$

b. (**10 points**) In the table below indicate the state of the cache after the following sequence of memory references:

```
operation      address
load            0x060
load            0x064
load            0x068
load            0x0A4
store           0x064
load            0x120
```

Assume that before the references above all entries are invalid, *i.e.* all the valid bits (**V**) in the cache are 0. And all the least-recently-used (**LRU**) bits are equal 0.

You must indicate the value of the valid bit and the LRU bits for all entries in the cache. For entries that have a valid bit V=1 you must also list the value of the dirty bit (**D**) and the value of the Tag.

Index	**V**	**D**	**Tag**	**Data**	**LRU**	**V**	**D**	**Tag**	**Data**
0	0				0	0			
1	0				0	0			
2	1	0	00001		0	1	0	00010	
3	0				0	0			
4	0				0	0			
5	0				0	0			
6	1	1	00000		1	0			
7	0				0	0			

operation	address	tag	index	offset
load	0x060	00000	110	0000
load	0x064	00000	110	0100
load	0x068	00000	110	1000
load	0x0A4	00001	010	0100
store	0x064	00000	110	0100
load	0x120	00010	010	0000