9.2C - Computer Systems

The RAM.hdl, Screen.hdl, and Keyboard.hdl chips are the three main chips used in the construction of the Memory.hdl file included in the Nand2tetris resources. The Memory.hdl chip, like other chips, is divided into two portions. Section heading and part content. The chip's name and the names of its inputs are listed in the header section. The Memory.hdl code and the necessary justification are attached below.

Code:
```
CHIP Memory {
IN in[16], load, address[15];
OUT out[16];

PARTS: DMux4Way(in=load, sel=address[13..14], a=loadram1, b=loadram2, c=loadscreen,
d=loadkbd);
Or(a=loadram1, b=loadram2, out=loadram); RAM16K(in=in, load=loadram,
address=address[0..13], out=ramout);
Screen(in=in, load=loadscreen, address=address[0..12], out=scrout);
Keyboard(out=kbout); Mux4Way16(a=ramout, b=ramout, c=scrout, d=kbout,
sel=address[13..14], out=out); }
```

The memory chip has three inputs: in (16-bit), load (1-bit), and address (15-bit), as can be seen in the header section. A 16-bit data value is provided by the input, a 15-bit memory location is provided by the address, and data writing is managed by the load. A 16-bit value is returned from memory via the out output. Six chips are included in the parts section; the keyboard and screen are pre-built. The single load input is demultiplexed into 4 outputs (loadram1, loadram2, loadscreen, and loadkbd) via the DMux4Way chip. In accordance with the address, this sends the load signal to the proper memory region. Four combinations (00, 01, 10, 11) are available in the address range [13–14] to control the four DMux outputs. For RAM writing, for instance, 00 would activate loadram1. Depending on the circumstance, loadram1 and loadram2 may target different RAM memory locations. The loading to the screen and keyboard areas is controlled by loadscreen and loadkbd, respectively. To make RAM writing easier, the Or chip combines loadram1 and loadram2 into a single loadram signal. With 14 address bits on the RAM16K chip, 16,384 (2^14) memory addresses with 16 bits each are possible. It uses address and load RAM to write, and it uses addressing to give read/write access. The Screen chip has a 8,192 pixel output display and utilises 12 address bits for it. Keyboard input is handled instantly without storage, therefore it just produces output. Based on the [13–14] address bits, the Mux4Way16 chooses whether of the following to output: ramout, ramout, scrout, or kbout.
This enables the CPU to access sensed and stored data from various memory locations as needed. The general architecture uses loading and addressing to control the RAM, screen, and keyboard I/O.