DEPARTMENT OF MATHEMATICS
AND COMPUTER SCIENCE

Computer Architecture Project

Gabriel Howard Jadderson : gajad16@student.sdu.dk
<EMPTY> : <EMPTY>
<EMPTY> : <EMPTY>

DM548
November 9, 2017

Contents

List of Figures

1 Introduction

2 Design

3 Implementation

3.1 Information restructuring

```
i ← 1
while i < length(A)
    j ← i
    while j > 0 and A[j-1] >
A[j]
        swap A[j] and A[j-1]
        j ← j - 1
    end while
    i ← i + 1
end while
```

```
1  .global _start
2
3  .text
4  _start:
5      movl  $4, %eax
6      movl  $1, %ebx
7      movl  $msg, %ecx
8      movl  $len, %edx
9      int   $0x80
10
11     movl  $1, %eax
12     movl  $0, %ebx
13     int   $0x80
14 .data
15 msg:
16     .ascii  "Hello, world!\n"
17     len =   . - msg
```

`network_layer_allowed_to_send`
: Signals the network layer that it can send a piece of data. Here the network layer should make sure the `from_network_layer_queue` contains at least one element, after which it can signal `network_layer_ready`, which means that the network layer has prepared at least one element in the queue.

`network_layer_ready`
: This signal signals to the link layer that the network layer has prepared an element to be sent in the `from_network_layer_queue` queue and that it should be sent now.

`data_for_network_layer`
: This signal signals to the network layer that the `for_network_layer_queue` contains a data

element for it to take care of. (a data element has been received)

4 Testing

5 Conclusion