MSE ASSIGNMENT 3  
Name : Zaina Hamid  
USN : 1PI10IS124

Qn1.

1. Copy Load

|  |  |
| --- | --- |
| mneumonic | n.copy |
| opcode | 0x |
| arguments | *greg1*   *greg2* |
| description | Load a value from a second register into a general purpose register. |
| example | n.copy SGP1,SGP2 |
| definition | *if( true )      greg1' = greg2       greg2' = greg2       pr' = pr + 1* |
| Traps | n/a |

1. Memory Direct Store

|  |  |
| --- | --- |
| mneumonic | n.store |
| Opcode | 0x |
| arguments | *greg*   *[n]* |
| description | Store a value to direct memory from a general purpose register. The memory address is limited to 15-bits (from local base). For larger addresses, use n.storei instead. |
| Example | n.store [0x7000],SGP1 |
| Definition | *if( n ≥ 0 )&&( n < MAXMEM )*   *memory[n  0x7fff]= greg*'   *greg' = greg*   *pr' = pr + 1*   *else*   *n.trap MemoryAddress* |
| Traps | MemoryAddress |

1. Bitwise Register And

|  |  |
| --- | --- |
| mneumonic | n.bitand |
| Opcode | 0x |
| arguments | *greg1*   *greg2* |
| description | Perform a bitwise AND between two general purpose register. The result is stored in the first register. |
| Example | n.bitand SGP1,SGP4 |
| definition | *if( true )*   *greg1*' = *greg1*  *greg2*       *greg2' = greg2*   *pr' = pr + 1* |
| Traps | n/a |

1. Bitwise Register And

|  |  |
| --- | --- |
| mneumonic | n.bitnot |
| opcode | 0x |
| arguments | *greg1* |
| description | Perform a bitwise NOT on a general purpose register. The result is stored in the register. |
| example | n.bitnot SGP1 |
| definition | *if( true )*   *greg1*' =  *greg1*   *pr' = pr + 1* |
| traps | n/a |

1. Bitwise Register OR

|  |  |
| --- | --- |
| mneumonic | n.bitor |
| opcode | 0x |
| arguments | *greg1*   *greg2* |
| description | Perform a bitwise OR between two general purpose register. The result is stored in the first register. |
| example | n.bitor SGP1,SGP3 |
| definition | *if( true )*   *greg1*' = *greg1*  *greg2*       *greg2' = greg2*   *pr' = pr + 1* |
| traps | n/a |

1. Bitwise Register Exclusive Or

|  |  |
| --- | --- |
| mneumonic | n.bitxor |
| opcode | 0x |
| arguments | *greg1*   *greg2* |
| description | Perform a bitwise XOR between two general purpose register. The result is stored in the first register. If *greg1* is *greg2* then the effect is to clear *greg1*. |
| example | n.bitor SGP1,SGP3 |
| definition | *if( true )*   *greg1*' = *greg1*  *greg2*       *greg2' = greg2*   *pr' = pr + 1* |
| traps | n/a |

1. Compare Equal

|  |  |
| --- | --- |
| mneumonic | n.cmpeq |
| Opcode | 0x |
| arguments | *greg1*   *greg2* |
| description | Perform an integer comparison between two general purpose register. The result is stored in the first register. |
| example | n.cmpeq SGP1,SGP3 |
| definition | *if( true )*   *greg1*' = *greg1* == *greg2*       *greg2' = greg2*   *pr' = pr + 1* |
| traps | n/a |

Qn2.

|  |  |
| --- | --- |
| **Cloud Computing** | **Mobile Computing** |
| Allows you to store your files and folders in a “cloud” area on the Internet. | Is taking a physical device with you. This could be a laptop or a mobile phone or some device which enables you to [telework](http://en.wikipedia.org/wiki/Telecommuting) – working wherever you go because of the small size of the device you’re using. |
| Allows access to all of your files and folders wherever you are in the world – but you do need a physical device with Internet access to access it. | Internet access is not mandatory |
| Files are synchronised between devices so wherever you go, you’ll always have access to your files. There is no need to carry round a laptop with you, you still need some physical device to access your service | There is always a necessity of a physical device where the files were stored in order to access them. |
| There is a lack of control over personal data .Insufficient information regarding how, where and by whom data is being processed | communications no longer run over a private network; some run over less-secure public carrier networks |
| Sensitive data is being stored on cloud which may be private and can be accessed by anybody. Extra password protection needed. | As Data is stored remotely, it leads to concerns that companies will use or sell this information as well as concerns that the information could be given to government agencies without the user’s permission |
| Key risk factor for all cloud computing services, particularly where the data to be transferred to the service provider is sensitive and is to be held offshore | There can be issues of data becoming locked in to a particular service. |

Qn3.

SenSay is an example of an application simulating an environment of context aware computing.

Sensay is a context-aware mobile phone that adapts todynamically changing environmental and physiological states. In addition to manipulating ringer volume, vibration, and phone alerts, SenSay can provide remote callers with the ability to communicate the urgency of their calls, make call suggestions to users when they are idle, and provide the caller with feedback on the current status of the Sensay user. A number of sensors including accelerometers, light, and microphones are mounted at various points on the body to provide data about the user’s context. A decision module uses a set of rules to analyse the sensor data and manage a state machine composed of uninterruptible, idle, active and normal states. Results from our threshold analyses show a clear delineation can be made among several user states by examining sensor data trends. SenSay augments its contextual knowledge by tapping into applications such as electronic calendars, address books, and task lists. The phone alleviates cognitive load on users by various methods including detecting when the user is uninterruptible and automatically turning the ringer off.

The light sensor was incorporated into the system. The light sensor was attached from the sensor box to the user’s phone by a long wire. By placing the light sensor on the phone, SenSay was able to determine whether the phone was in the user’s pocket or not. Throughout the test, the light sensor proved to be extremely effective and sensitive. At the beginning of the test, the phone was placed in the user’s pocket. Next the user removed the phone and placed it on a table. A substantial jump in the output can be seen at this juncture. After another few seconds the user picked up the phone and placed in his hand, to simulate talking over the phone, partially covering the light sensor. The light sensor fluctuates a certain degree; however the average remains over the threshold value throughout the test. Lastly, the phone is returned to the user’s pocket, causing the data values to drop. The light sensor is used in various ways by the platform. One use is when the phone is visible to light, it is highly likely that the user is in view of the phone, allowing the phone not to ring and instead just vibrate to get the user’s attention.