Team 38 Strength Points

**Direct Memory Access**

* Modularity
  + All modules that in data sheet is implemented
  + Support reusability and readable comments
  + Easy to test and debug
* Supports program condition mode from processor to DMA by addresses in 8237 datasheet or by using software (Assembly instructions)
* Supports reading internal registers, current mode and status registers in DMA
* Supports IO to Memory, Memory to IO, Memory to Memory modes
* Supports transferring data up to 64k byte in one request
* Supports Memory mapped devices and memory up to 64k byte
* Supports 4 channels to connect up to 4 IO devices (Keyboard, Printer and Hard Disk, etc)
* Supports handling more than 4 IO devices by cascade mode
* Supports single transfer mode to transfer one time till number of words changes to zero, block transfer mode to transfer unknown number of transfers till the IO sends external EOP (end of operation) to DMA.
* Handling priority to the devices which connected to channels in priority module
* Supports disable or enable any of channels in DMA by mask register
* Supports Auto initialization to set the operation to the default mode to restart the transfer if channel in verify transfer mode
* Supports resetting the DMA, address increment or decrement and DREQ and DACK signal to be read as active low or high
* Supports status register to be read by processor to be aware the current status of channels in DMA
* Supports set request by software (Assembly instruction)
* Supports mode register programing to define every channel parameter as read or write transfer

**Processor**

* Supports arithmetic operations (xor, or, add, addi, etc).
* Communicates with RAM and IO Devices through the Bus through an assembly instructions “LW” $register $address and “SW “ $register $address “
* Supports assembly instructions to access with DMA internal registers
* Executes arithmetic instructions during DMA Transfers
* Support Holding the Execution of the Bus’s Instructions if the DMA uses the Bus.

**BUS**

* Consist of DataBus, AddressBus, control, IReady and TReady
* Supports acknowledgement for both master and slaves through IReady and TReady Control lines
* Read and Write Operations take one clock cycle

**GUI**

* Could browse any assembly file located in the system
* Supports User Interface to generate assembly instructions that can access the Verilog code to initiate DMA operations
* Checks registers ranges to match the allowed range & detect if negative number is specified showing error message
* Three edit boxes to specify source, destination & count of bytes to transfer in memory to memory mode
* Six edit boxes to specify DMA registers in (Program DMA) mode (not completed).

**Simulation GUI**

* Animation Coloring for BUSs as indication for using the bus.
* In Write Bus Transfer the Address Bus, Data Bus and Control Bus get colored with the Master’s Color, and in Read Transfer the Data Bus get colored with the Active Slave Color, and Control and Address Bus get colored with Master’s color.
* Visualizes the current program counter and current executed instruction.
* Supports visualization of requesting the bus from DMA or CPU.
* Shows the waiting states of both CPU and DMA if one of them waits bus.
* User can control the clock cycles forward and backward.
* Supports User Interface Buttons and Progress Bar to control the cycles, show the progress of the clock cycles and a play-stop button for auto-play animation.
* Supports Data acquisition for values of data bus, address bus, control bus.

**Website**

* Home page has 3 main buttons highlighting a brief description about our project, our GUI, intel 8237, contribution of each member:
  + Check code button that takes you to out GitHub repository
  + Try our GUI now Button that launch our GUI once you click on it
* Help page:
  + A graphical step by step carousel walk-through of the GUI
  + GitHub repo Download button
  + Button that yields more details on our design choices

**GitHub Repos:**

GUI Repo: <https://github.com/AmrElsersy/DMA_GUI>

DMA Repo: <https://github.com/AmrElsersy/DMA>