Demo Day Activities

1) Download the first example from internet (you can try the other two later)

There are some simple examples to begin with

   1) This example is a squarewave output on PP7, with the time delay implemented as a loop.
   http://users.ece.utexas.edu/~valvano/Starterfiles/Simple_DP512asm.zip (if DP512)
   http://users.ece.utexas.edu/~valvano/Starterfiles/Simple_DG128asm.zip (if DG128)
   2) This example is a not gate, with PT7 input, PP7 output
   http://users.ece.utexas.edu/~valvano/Starterfiles/NotGate_DP512asm.zip
   http://users.ece.utexas.edu/~valvano/Starterfiles/NotGate_DG128asm.zip
   3) This example implementing a squarewave, with the TCNT timer delay
   http://users.ece.utexas.edu/~valvano/Starterfiles/Square_DP512asm.zip
   http://users.ece.utexas.edu/~valvano/Starterfiles/Square_DG128asm.zip

2) Cut out sheet of paper

3) Plug board into protoboard, lining the paper up with the pins (see pins 1,25,26,50)

4) Connect RS232 cable between the 9S12 and the PC (this is COM1 in ACA1.106)

5) Switch in LOAD mode

6) Power applied to 9S12 (notice the polarized power plug)

DG128/DP512		
1	H1	50
PS4		Gnd
PS5		Gnd
PS6		PS0
PS7		+5
PS1		PE1
PT7		PE0
PT6		Reset*
PT5		PE7
PT4		PH0
PT3		PH1
PT2		PH2
PT1		PH3
PT0		PH4
PP7		PH5
PP6		PH6
PP5		PH7
PP4		PS2
PP3		PE4
PP2		PS3
PP1		VRL
PP0		VRH
PAD0		PAD4
PAD1		PAD5
PAD2		PAD6
PAD3		PAD7
25		26

7) Start TExaS

8) Open the file **SimpleDG128.UC** (if DP512 use **SimpleDP512.UC**)

9) Place the system in *Simulation* mode, assemble, and run. Observe the Scope and IO.

10) Place the system in *Real 9S12* mode. Is your board connected? The IO and Scope windows
can be closed or hidden; they are not needs in *Real 9S12* mode

11) Assemble the program, notice that it also downloads code to 9S12DG128

12) Run the program and notice the lights flash (click on the UC window and type **F5** to update)

13) Experiment with single stepping features

>   **Reset**
>
>   **Single step** until Register Y decrements twice
>
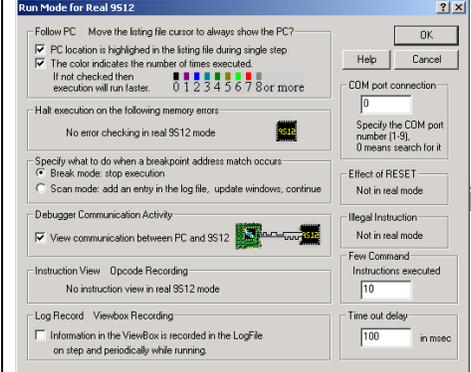>   Notice that **BreakMode** is selected in the **Mode** menu
>
>   In the list file, first left click on the **eora** instruction, then right click **RuntoCursor**

This **Delay** subroutine takes over 150,000 instructions to execute, so it will not be appropriate
to use **StepOver** or **StepOut** commands for this example. However, for most subroutines, the
StepOver command will single step through, executing the entire subroutine. The **StepOut**
command can be used once a subroutine is started to single step through the remaining
instructions of the subroutine. (**StepOut** actually executes until the SP gets bigger).

14) Explain the **Mode->RunMode…** settings
       Click **Mode** menu, roll mouse over **RunMode**… command and type **F1**
       Click the green **Real 9S12** link in the first line

	- COM port defines the serial port on the PC
	   COM=0 will search all COM ports for a 9S12
	   In ACA1.106, COM=0 or COM=1 can be used
	   Some USB to serial links use COM=3
	- PC location is highlighted and colored
	   Highlighting is good for single step
	- Breakpoint/Scanpoint mode selection
	- View communication shows low-level messages
	- Few Command is F11
	- Time out delay should be 100 msec

15) Experiment with *Breakpoints*
       Reset (Recall that **BreakMode** is selected in the **Mode** menu)
       In the list file, first left click on the **eora** instruction, then right click **BreakatCursor**
       Run and observe the program halts at or near the **eora** instruction
16) Experiment with *Scanpoints*
       Reset (Recall that **Breakpoint** is set at the **eora** instruction)
       Execute **Mode->BreakMode** and deselect **BreakMode** (this is *Scanpoint* mode)
       Run and observe the program dumps the ViewBox at or near the **eora** instruction
17) Experiment with *Screen Update Mode*
       Reset
       Remove the breakpoints (click RemoveAll in the Break/Scan box)
       Execute **Mode->ScreenUpdatePeriod** and select 1 second updates
       Run and observe the program dumps the ViewBox every 1 second
To run in embedded mode
0) Disconnect power to the 9S12DG128 board
1) Disconnect the serial cable because it is not needed
2) Set the Run/Load switch to Run mode,
3) Apply power to the 9S12DG128 board
       The 9S12DG128 runs at 8 MHz if you do not modify the PLL.
       You can adjust the E clock rate by configuring the PLL.
Things you can do with the power on
       Hit the reset button
       Move the Run/Load switch
       Touch/release buttons on the board or protoboard
       Connect/disconnect voltmeter to system
       Plug and unplug the RS232 cable
Things you should not do with the power on (the $100 learning experience)
       Plug/unplug the board into/from the protoboard
       Connect/disconnect cables to external devices
       Insert/remove wires resistors or chips from the protoboard
       Touch bare wires or pins with your fingers