11015_MS2

MPLAB® Simulators Advanced Stimulus

LAB 3

Circuit Breaker Stimulus Requirements Lab 3

- **A/D voltage and current values**:- Register Stimulus one file two data columns inject into AD1BUF0
  - AC voltage scaled A/D input :- Excel spread sheet
  - AC current scaled A/D input :- Excel spread sheet
- **Zero Crossing 60Hz line frequency clock** :- Clock Stimulus inject into IC1
- **Asynch Test button**:- Stimulus Controller pulse high RD3
- **Asynch Reset button**:- Stimulus Controller pulse high RD2
- **Solenoid Trip output**:- Watch window LATD [Bit 1]

Circuit Breaker Zero Crossing Lab 3a

- **Open MPLAB® IDE**
  - Select menu "*File>Open Workspace…*"
  - Select the "11015 MS2 / Lab3 / CircuitBreaker.mcw"
  - OR Select menu "*File>Recent Workspaces>CircuitBreaker*"
  - Build the project

- **Preparing stimulus**
  - Open the Excel spread sheet "VoltageCurrent.xls"
  - View the data and graph representations
  - Copy the 2 columns of data to be used as A/D readings
  - Within the MPLAB® IDE, open "*File>New*" and paste them into a new file
  - Save and name the file "xxxxxxxx.txt"

Circuit Breaker Zero Crossing Lab 3a

- **Open Stimulus and attach A/D file**
  - Select "*Debugger>Stimulus>New Workbook*"
  - Select the "Register Injection" tab at the top
  - Enter an optional label if desired
  - Select Register "AD1BUF0" to inject data
  - Select Trigger type "Demand"
  - Width will be "2" bytes
  - Add the data file name as specified in the first step
  - Select "Yes" for wrap
  - Select "Dec" for decimal data type
  - Add optional comment

Circuit Breaker Zero Crossing Lab 3a

- **Stimulus define ZC clock**
  - Select the “Clock Stimulus” tab at the top
  - Enter an optional label if desired
  - Select “IC1” from drop down under “Pin” Column for InputCapture 1
  - Select “Low” as “Initial” state from drop down
  - Set “Low Cycles” to “333333”. Set “High Cycles” to “333333” 60hz clock at 40 MIPS (six 3’s in each)
  - Select the “Begin” box. Leave at default “At Start”
  - Select the “End” box. Leave at default “Never”
  - Add optional comment

Circuit Breaker
Zero Crossing Lab 3a

- **Apply Synchronous Stimulus**
  - Select the “Apply” button at the bottom of the stimulus window
  - You are now ready to test

Circuit Breaker Zero Crossing Lab 3a

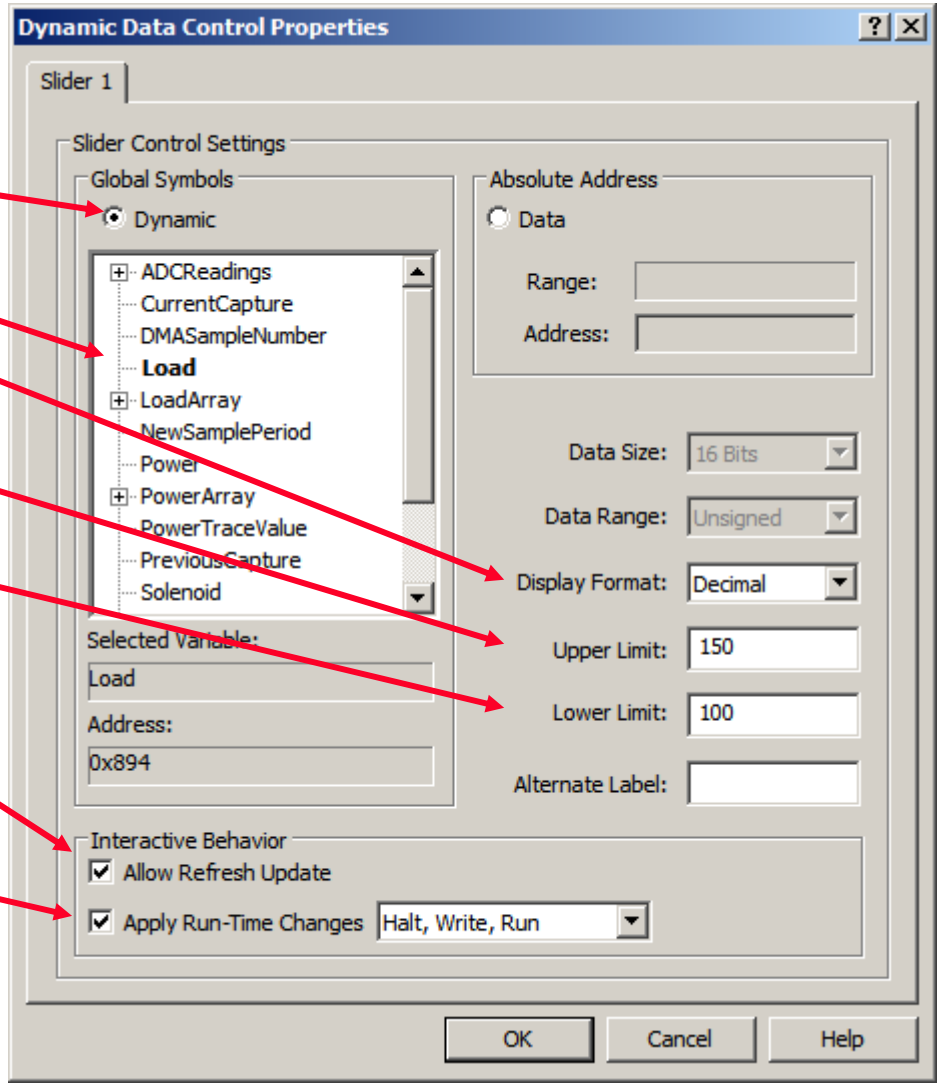
- **Testing. Execution**
  - Select “Reset” and then “Run”
  - Watch the variables in the watch window. Once the “Power” value has changed, stop the program
- **Testing. Verify Power**
  - Verify the Power value is equal to the Power value in the Excel spread sheet for the injected data. (One tab in the spread sheet for different test data)
  - View the File Register window at address 0x4780. Note the A/D data is placed here using the DMA and peripheral indirect address mode. (Handled totally by hardware within the silicon)

Circuit Breaker Zero Crossing Lab 3b

- **Verification using DMCI. Slider setup**
  - Select "*Tools>Data Monitor and Control Interface*"
  - Click "Tiled window view" button (bottom 4th button)
  - Adjust the tiles so you have 4 graphs and 1slider visible
  - Enable the slider by setting the check box in upper left
  - Right click in colored area of slider to bring up the configuration
  - Set the configuration up as displayed on next page

Circuit Breaker Zero Crossing Lab 3b

- **Slider configuration**
  - Dynamic selection
  - “Load” variable
  - Display format Decimal
  - Upper limit 150
  - Lower limit 100
  - Allow refresh
  - Apply Run-Time changes as “Halt, Write, Run”

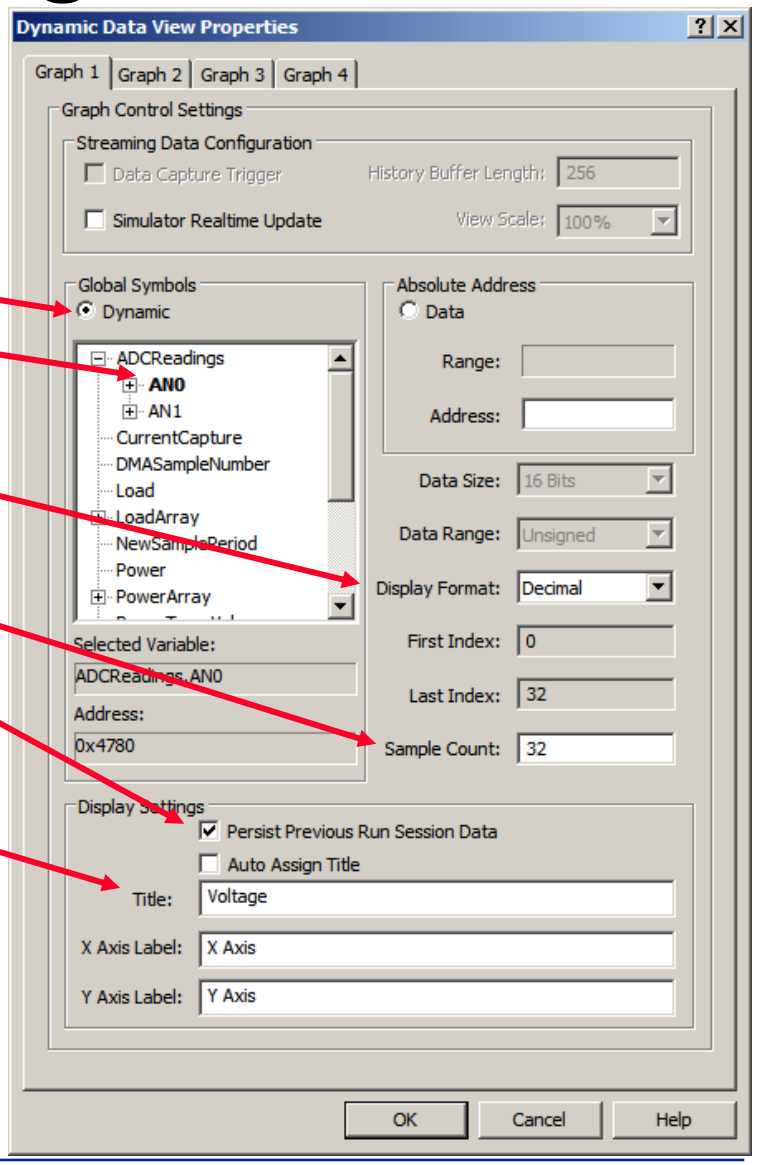


Circuit Breaker Zero Crossing Lab 3b

- **Verification using DMCI. Graph Setup**
  - Enable the 4 graphs by setting the check box in upper left of each.
  - Right click in the center of the first graph
  - Select the top item “Configure Data Source”
  - Go through each tab, one for each graph and set them up as shown on the next 4 pages

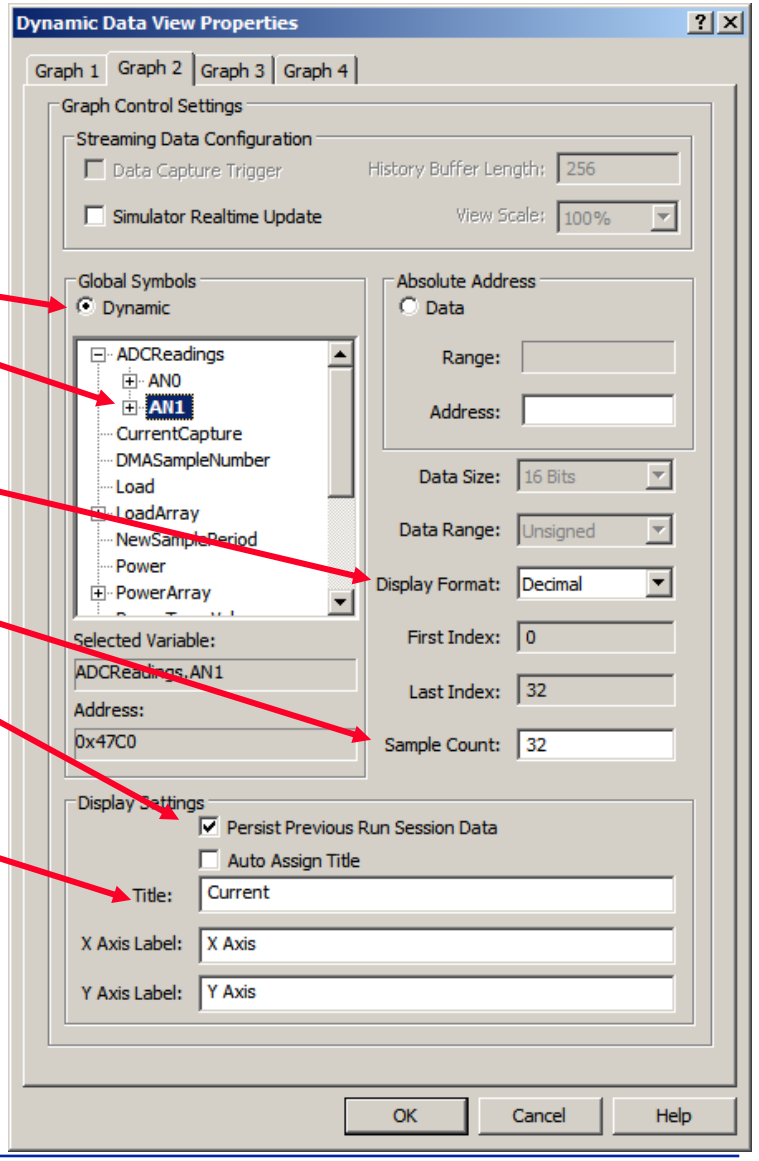
Circuit Breaker
Zero Crossing Lab 3b

- **Graph 1 configuration**
  - Dynamic selection
  - “ADCReadings.AN0” variable
  - Display format “Decimal”
  - Sample count “32”
  - Persist Previous Run data
  - Title “Voltage”



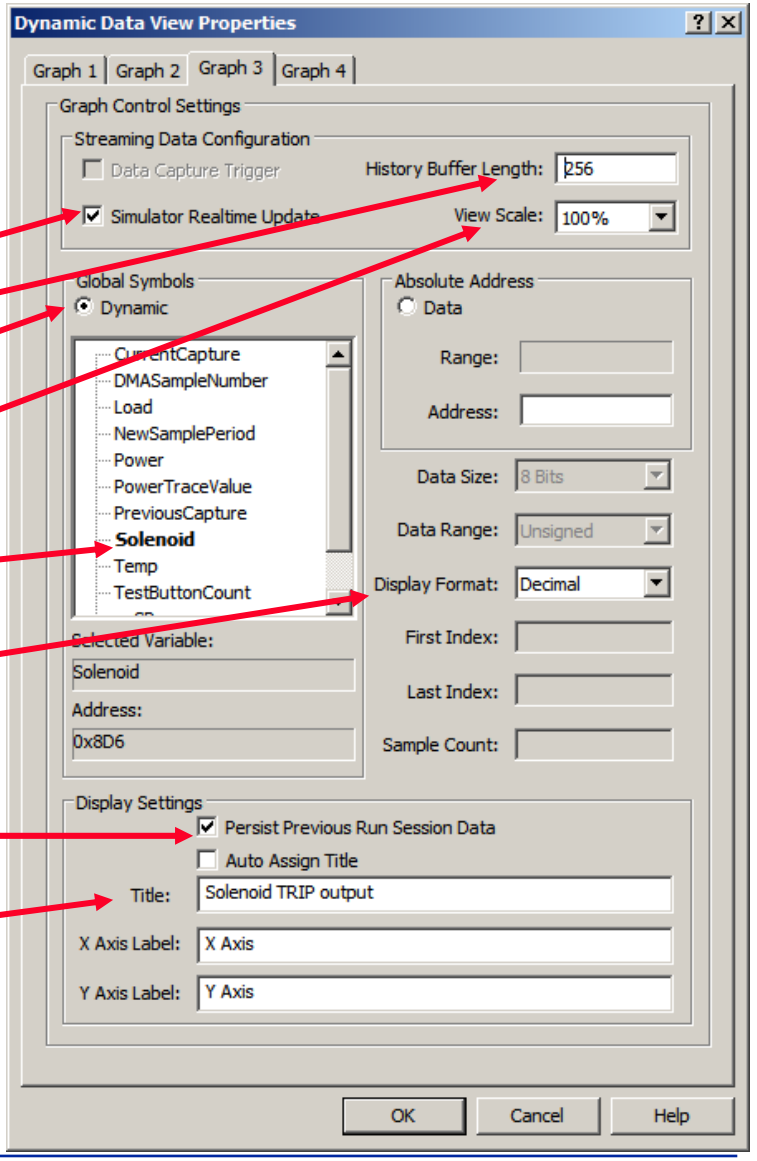
Circuit Breaker
Zero Crossing Lab 3b

- **Graph 2 configuration**
  - Dynamic selection
  - “ADCReadings.AN1” variable
  - Display format “Decimal”
  - Sample count “32”
  - Persist Previous Run data
  - Title “Current”



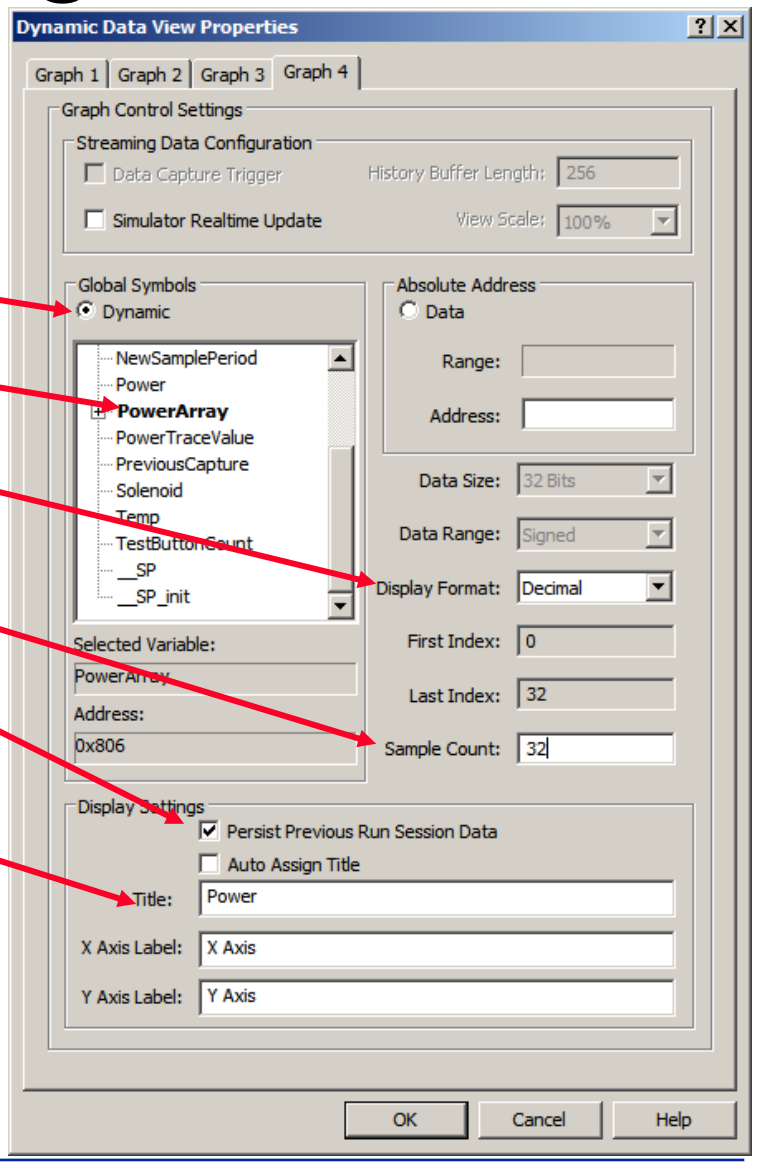
Circuit Breaker Zero Crossing Lab 3b

- **Graph 3 configuration**
  - Simulator Realtime Update
  - History Buffer “256”
  - Dynamic selection
  - View Scale “100%”
  - “Solenoid” variable
  - Display format “Decimal”
  - Persist Previous Run data
  - Title “Solenoid TRIP output”



Circuit Breaker
Zero Crossing Lab 3b

- **Graph 4 configuration**
  - Dynamic selection
  - “PowerArray” variable
  - Display format “Decimal”
  - Sample count “32”
  - Persist Previous Run data
  - Title “Power”



Circuit Breaker Zero Crossing Lab 3b

- **Testing using the DMCI**
  - Reset the application
  - Start execution
  - Select the slider control button with left mouse button
  - Adjust the slider keeping the mouse button down until you have the desired value.
  - Release the mouse and the selected value will be applied into the Load variable
  - When you raise the value above 116% the trip will occur
  - Set an Asynch stimulus to reset the breaker after you lower the Load percentage

Circuit Breaker *Extended reach* Lab 3b

- **Verify by tracing data**
  - Focus on Stimulus workbook
  - Select the “Register Trace” tab at the top
  - Enter an optional label if desired
  - Select Register “PowerTraceValue” to monitor
  - Trigger type “PC=” will be the default for data variables
  - Select the label “TracePower” for the PC value
  - Set width to “4” as the variable is a long (4 bytes)
  - Provide the file name to log the data into
  - Select “Dec” for decimal data type
  - Add optional comment

Circuit Breaker *Extended reach* Lab 3b

- **Apply updated Stimulus**
  - Select the “Apply” button at the bottom of the stimulus window

- **Testing. Execution**
  - Clear the “Power” value in the watch window
  - Select “Reset” and then “Run”
  - Watch the variables in the watch window. Once the “Power” value has changed, stop the program
  - Select the “Remove” button at the bottom of the stimulus window to allow the trace file to be closed

Circuit Breaker *Extended reach* Lab 3b

- Open the trace data file. “Select All” data within the file and “Copy”
- Open the Excel spread sheet “VoltageCurrent.xls”
- Highlight an empty cell in a free column next to the highest cell of calculated power values, that you will compare the data to
- “Paste” the data. This will fill the column adjacent to the column you are going to compare the data with
- Verify at each row (one set of A/D data) that the power traced out, matches the spread sheet calculations

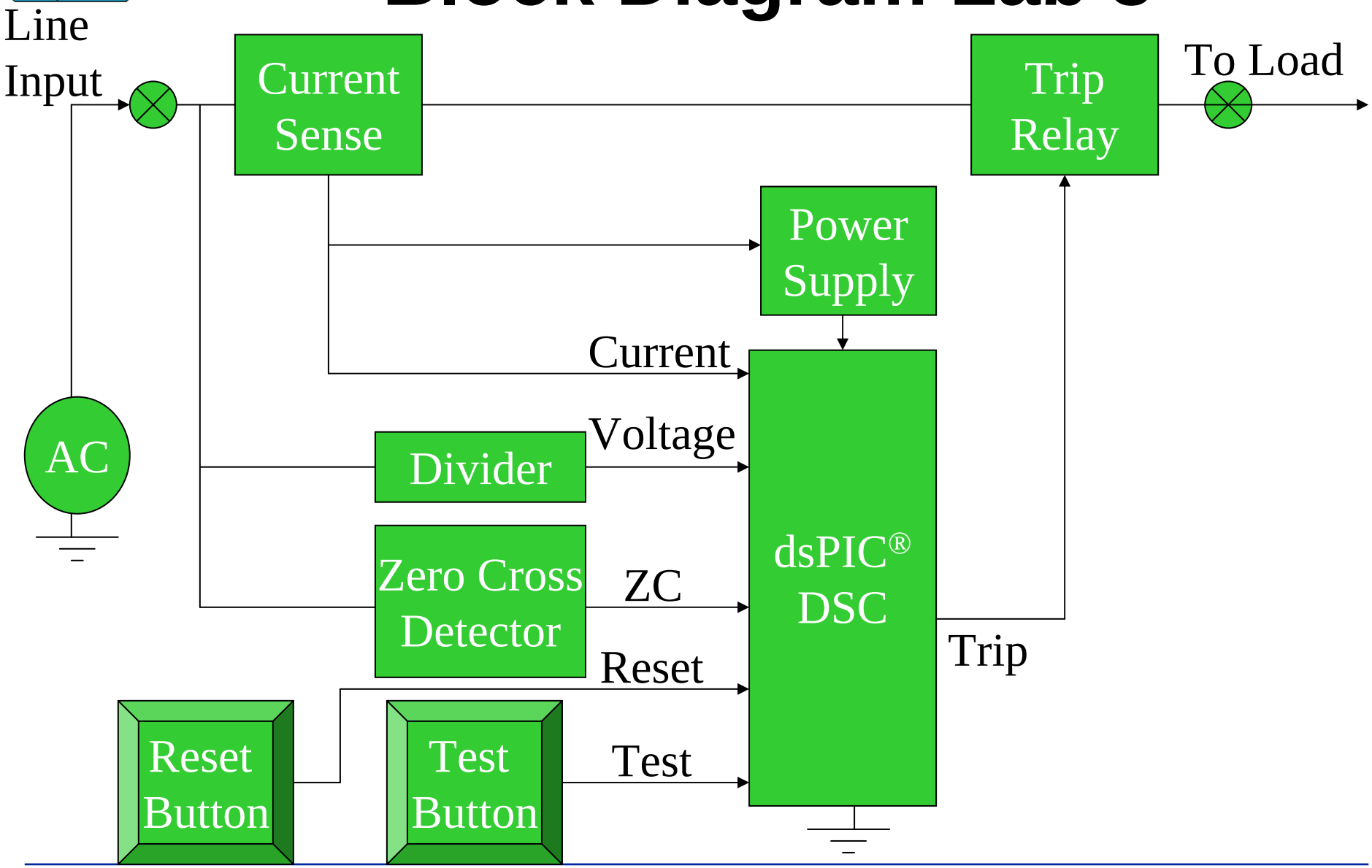
Circuit Breaker *Extended reach* Lab 3b

- **Additional Extra Objective**
  - Create different A/D input files using the different tabs within the EXCEL spread sheet, and test each set of data.
  - Use “Over Current” and “Over Voltage” and check if the “Trip” output is triggered.
  - The “Trip” pin is RD1, shown as “LATD [Bit 1]” in watch window
  - Create asynch button for “Test” (RD3) and test
  - Create asynch button for “Reset” (RD2) and test

Circuit Breaker

- **Following are block diagrams to explain how the application has been designed and how the peripherals are operating within the application**
  - Overall block diagram
  - ZC Input Capture block diagram
  - Timer 3 block diagram
  - ADC block diagram
  - DMAC block diagram

Circuit Breaker Block Diagram Lab 3

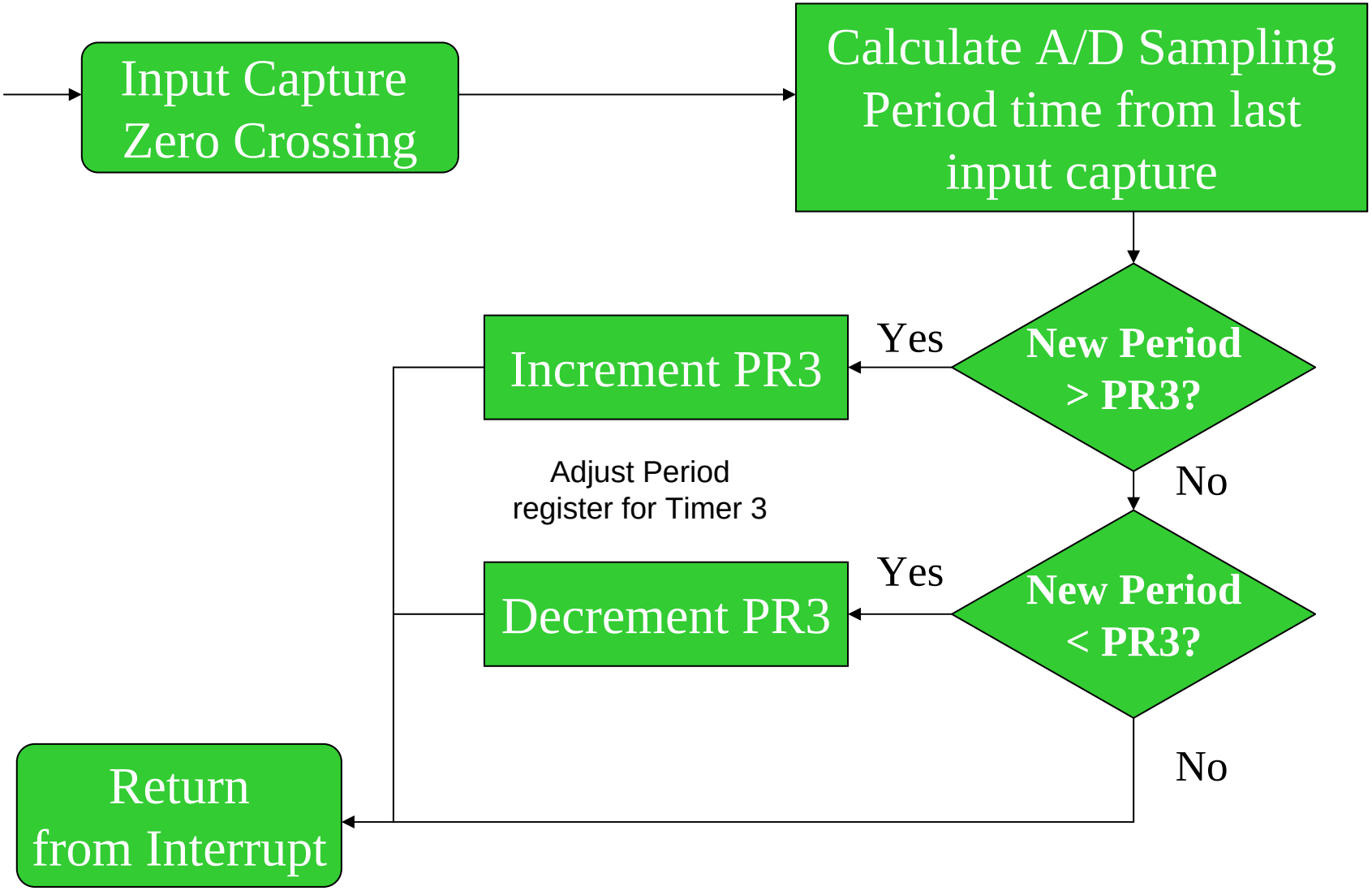


Circuit Breaker Hardware Configuration Lab 3

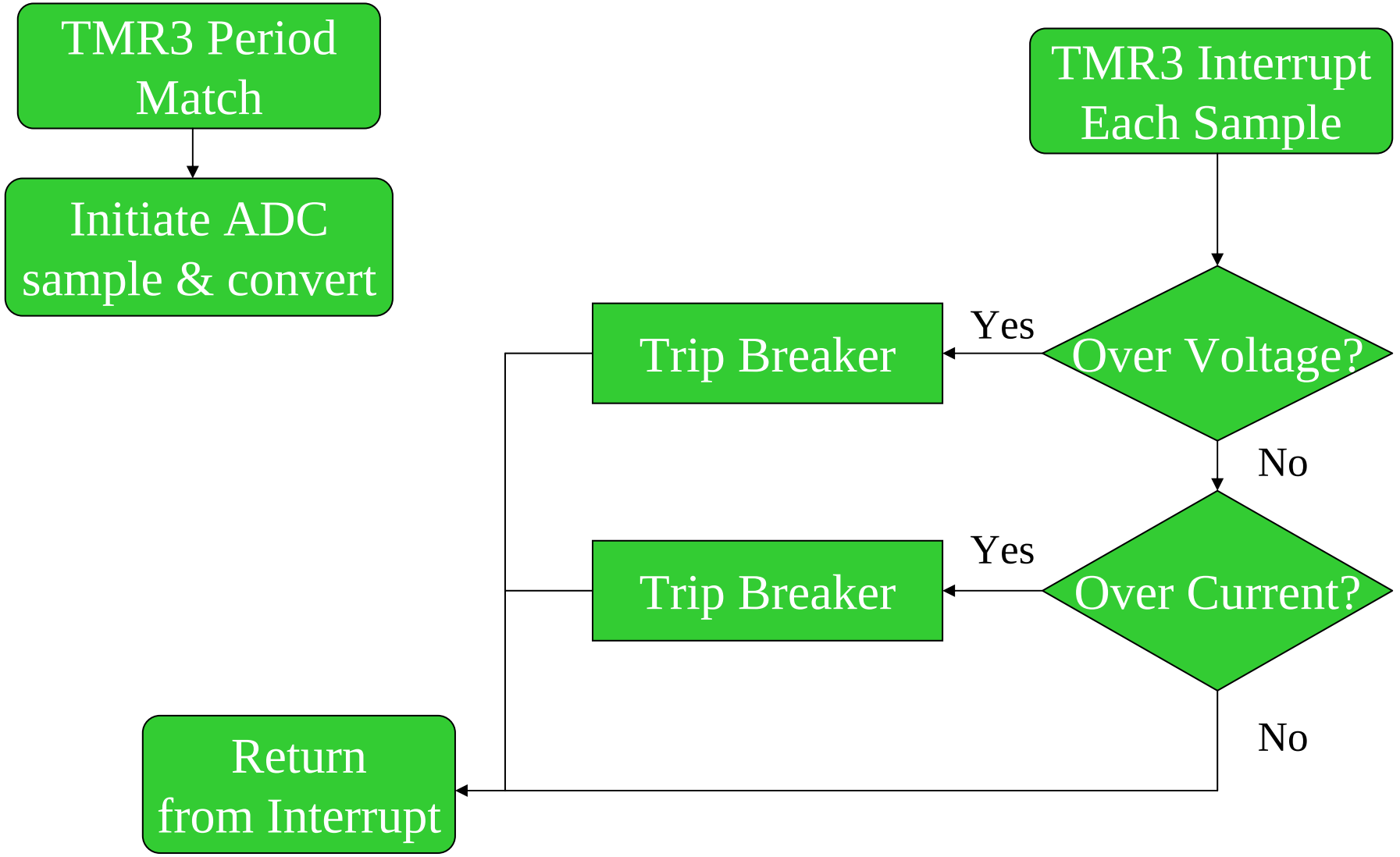
- **Input Capture**
  - AC Zero Crossing voltage triggers IC1 on rising edge
  - Uses TMR2 as time base, free running 16-bit mode period is 1/16 of TMR3 rate, no interrupts
  - IC1 Interrupts firmware to re-calculate A/D sampling period (TMR3 Period value)
  - Maintains phase lock with AC line

- **Timer 3 configuration**
  - Period is set to 1/32 of AC line period
  - Creates 32 identically spaced samples per line cycle
  - Period is adjusted by IC1 to compensate for Phase and line frequency shifts
  - Triggers ADC conversions for both voltage and current

Circuit Breaker Input Capture Interrupt Lab 3



Circuit Breaker TMR3 Interrupt Lab 3

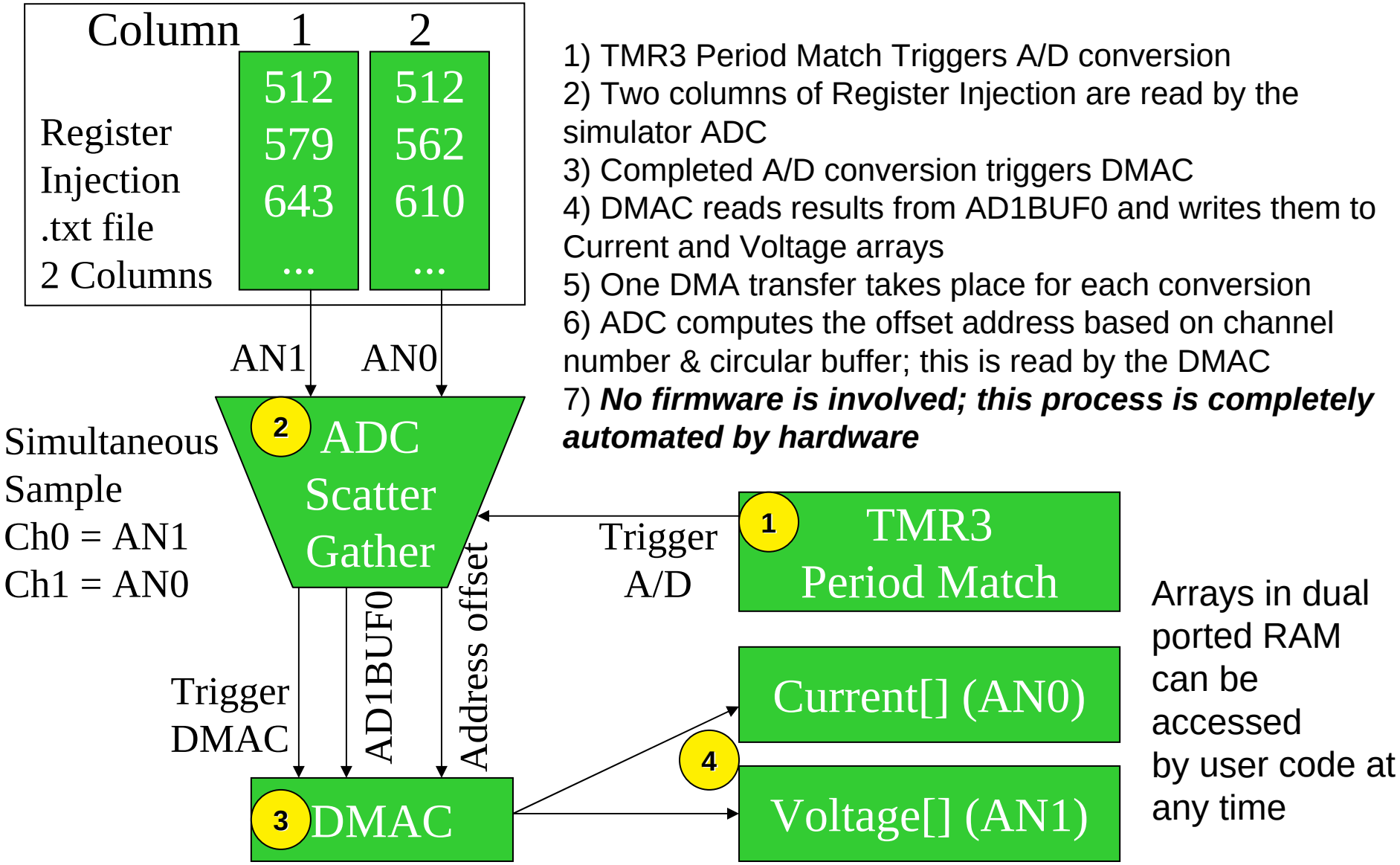


Circuit Breaker Hardware Configuration Lab 3

- **A/D configuration**
  - Simultaneous sampling CH0=AN1 and CH1=AN0
  - Conversion Triggered by TMR3 period match
  - Uses scatter / gather offset address generation for DMAC use, maintaining circular buffer computations
  - Interrupt detected and handled by DMAC hardware

- **DMAC configuration**
  - Services ADC conversion completion
  - Computes final destination address for ADC results
  - Moves data from AD1BUF0 to either Current[] or Voltage[] dual port RAM array
  - Interrupts firmware when both arrays are completely full with 32 A/D samples (64 transfers)

Circuit Breaker Simulator ADC Lab 3



1) TMR3 Period Match Triggers A/D conversion
2) Two columns of Register Injection are read by the simulator ADC
3) Completed A/D conversion triggers DMAC
4) DMAC reads results from AD1BUF0 and writes them to Current and Voltage arrays
5) One DMA transfer takes place for each conversion
6) ADC computes the offset address based on channel number & circular buffer; this is read by the DMAC
7) ***No firmware is involved; this process is completely automated by hardware***

Arrays in dual ported RAM can be accessed by user code at any time

Circuit Breaker DMAC Interrupt Lab 3