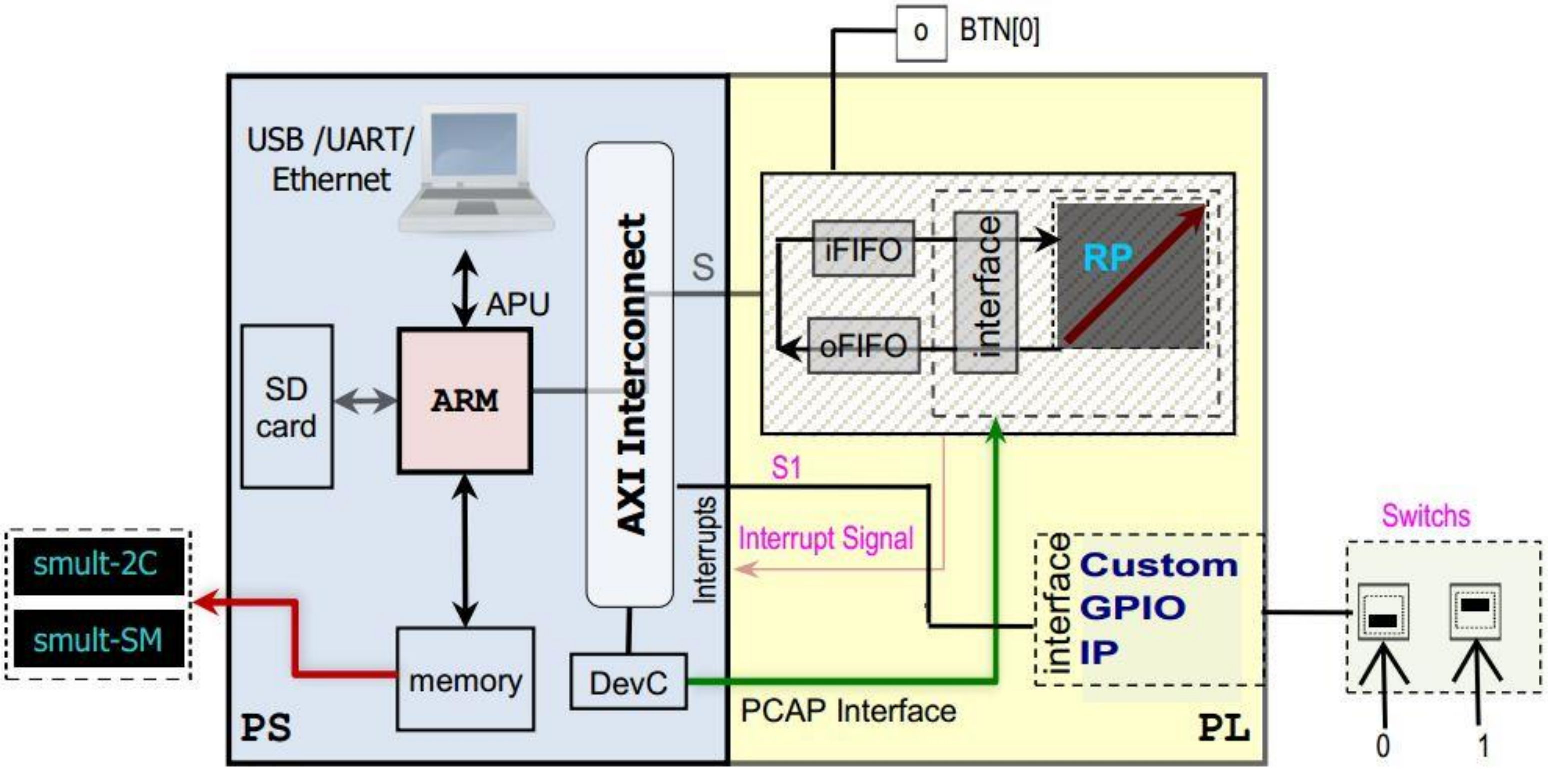
Reconfigurable Signed Multiplier (2C and SM) on Zynq-7000 PSOC

Andrew Meesseman and Hussein Alawsi
ECE 5736: Reconfigurable Computing
Professor D. Llamocca
Oakland University

Motivation

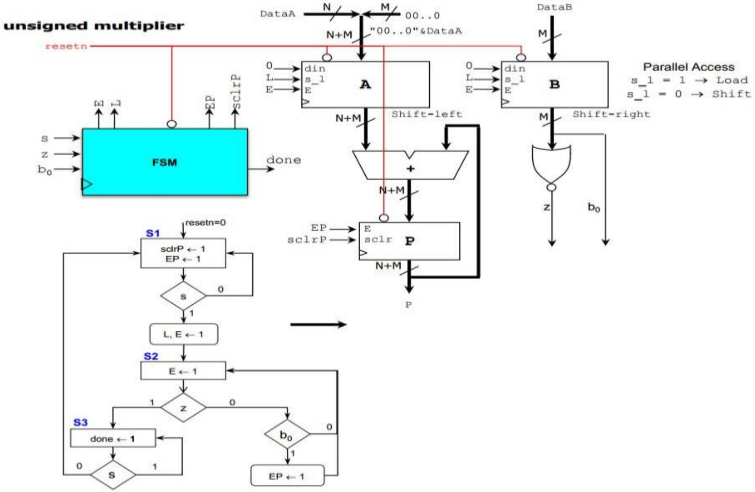
- Implement a system that makes use of dynamic partial reconfiguration (DPR) to show the benefits and possibilities with this feature
- Implement a PL to PS interrupt to show the possibilities available with this feature as well

High Level PS-PL Interfacing



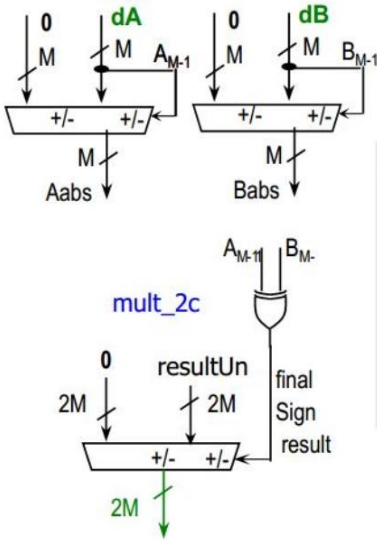
Signed Multiplier Components

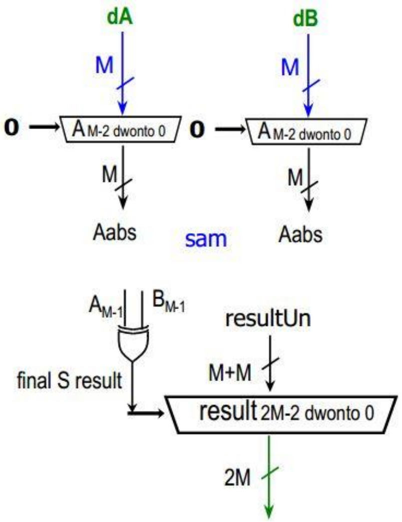
- Unsigned Multiplier
- 2C Logic
- SM Logic



2C

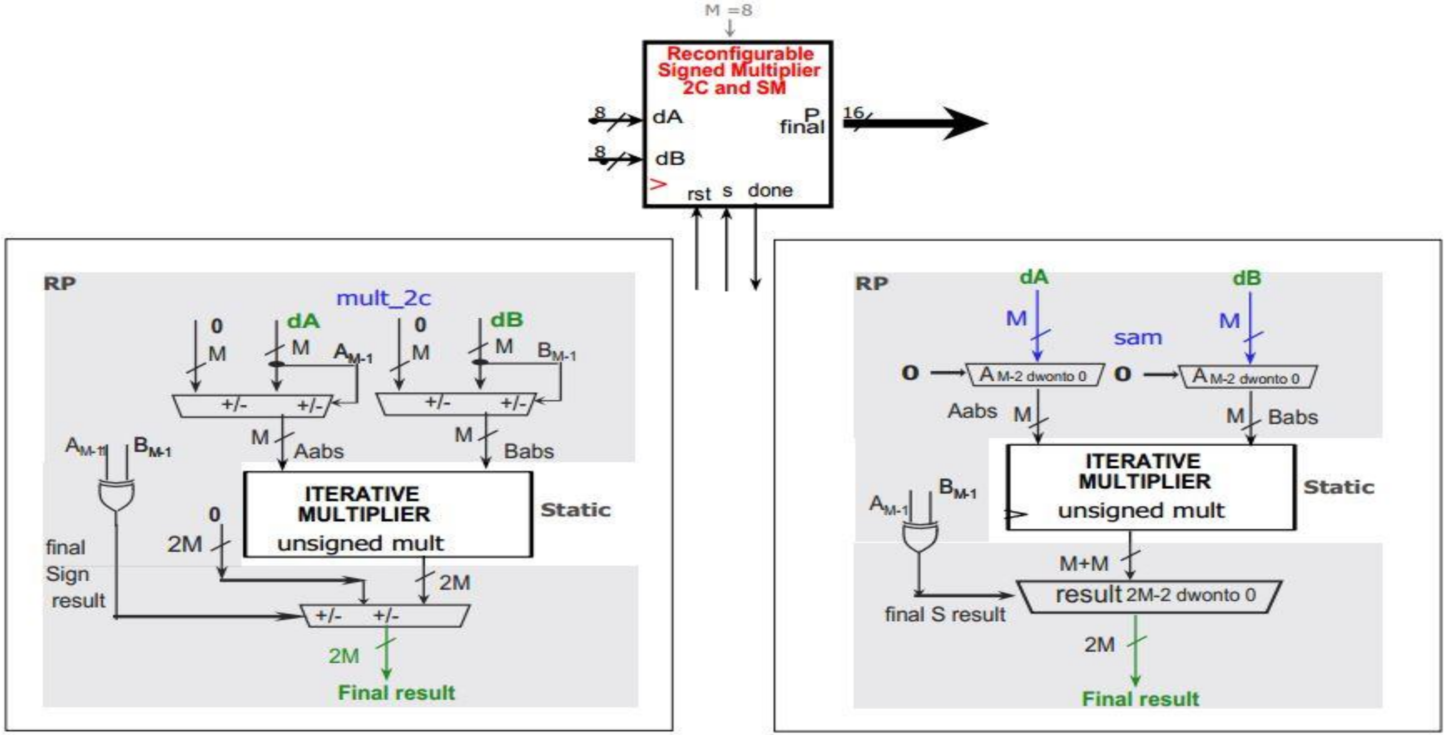
SM




dA
M
0
$A_{M-2 \text{ dwonto } 0}$
M
Aabs
dB
M
0
$A_{M-2 \text{ dwonto } 0}$
M
Aabs
sam
A_{M-1}
B_{M-1}
resultUn
M+M
final S result
$\text{result}_{2M-2 \text{ dwonto } 0}$
2M

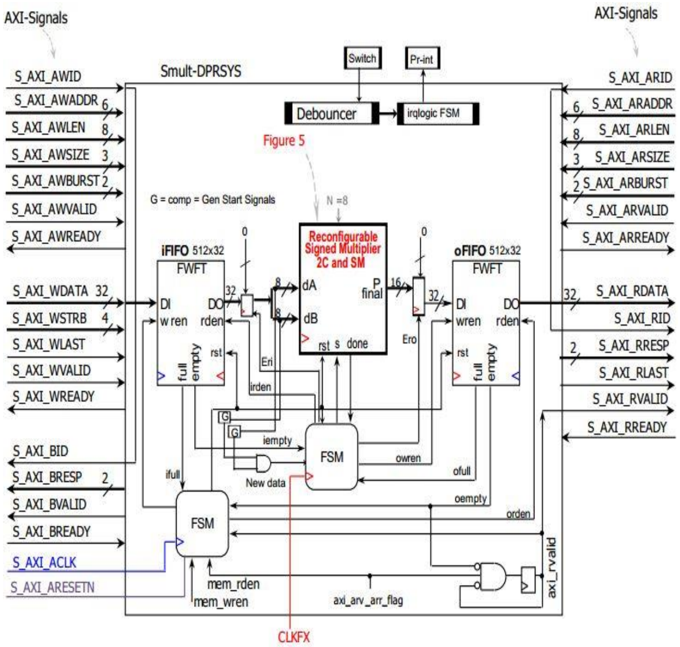
Static and RP Separation

- Unsigned multiplier is used in both circuits (which is why it is in static)
- All AXI4-Full logic and both FSM’s are in the static portion (as they do not change)



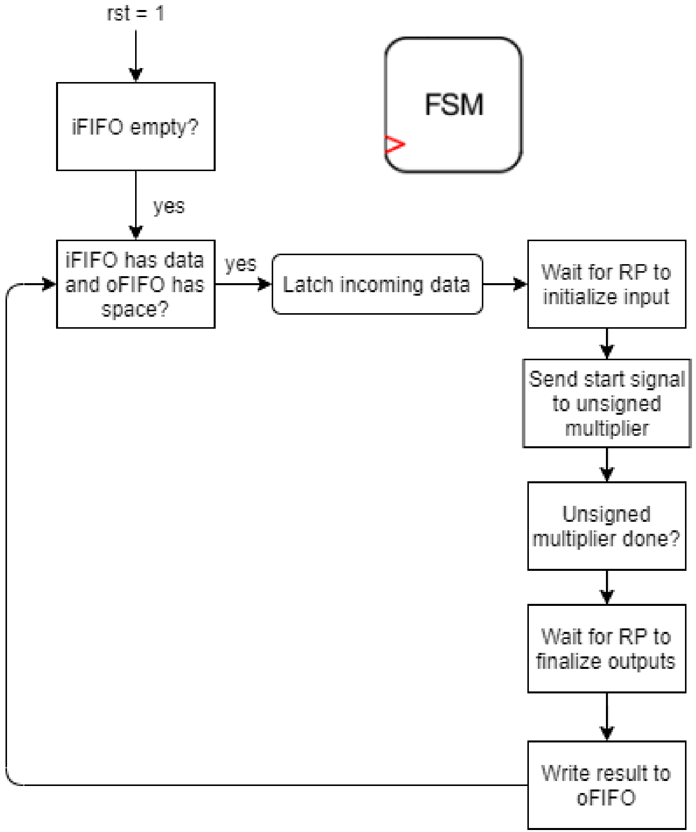
Signed Multiplier Implementation

- AXI4-Full
- Multiplies 8-bit by 8-bit input data and produces a 16-bit output based on the current circuit in the reconfigurable partition (RP)
- Also includes circuit to generate PL to PS interrupt



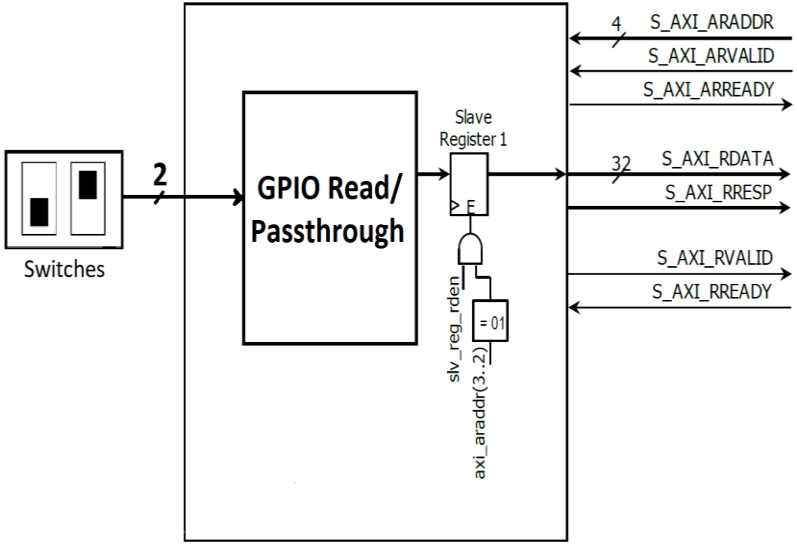
Signed Multiplier Application FSM

- Controls when input data is sent to circuit and when it is put into oFIFO


rst = 1
iFIFO empty?
FSM
yes
iFIFO has data and oFIFO has space?
yes
Latch incoming data
Wait for RP to initialize input
Send start signal to unsigned multiplier
Unsigned multiplier done?
Wait for RP to finalize outputs
Write result to oFIFO

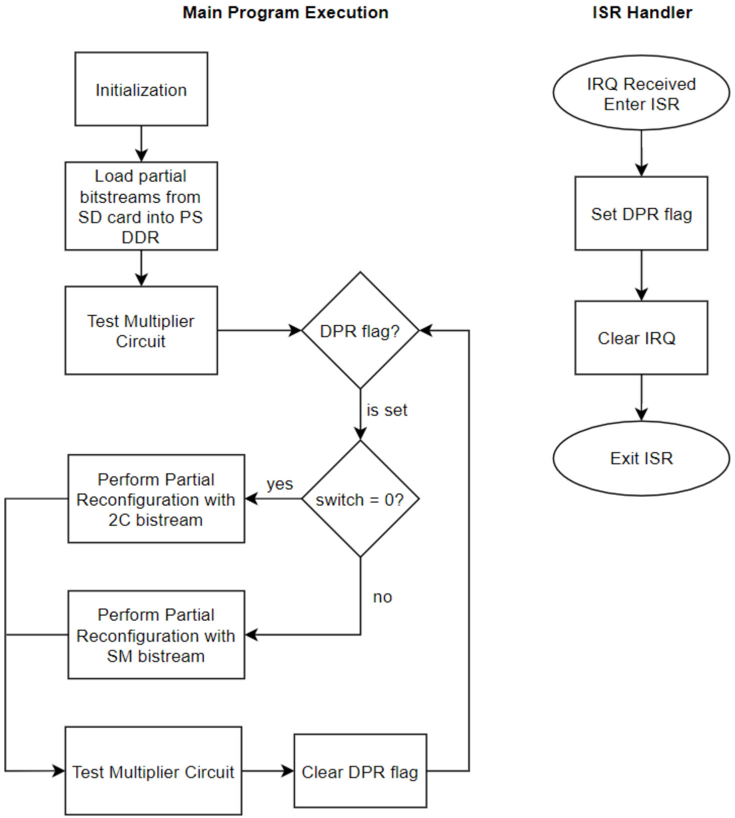
GPIO Passthrough IP

- AXI4-Lite
- Allows software to determine which bitstream to reconfigure during DPR



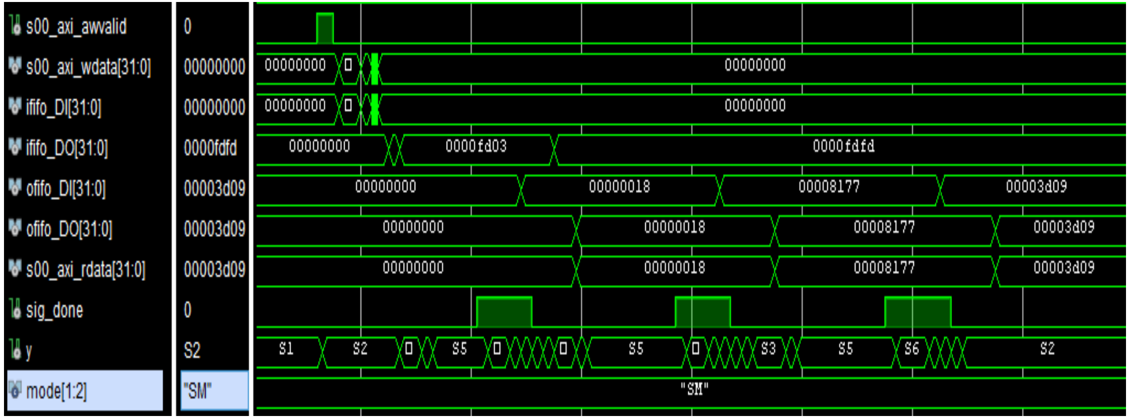
Software Functionality

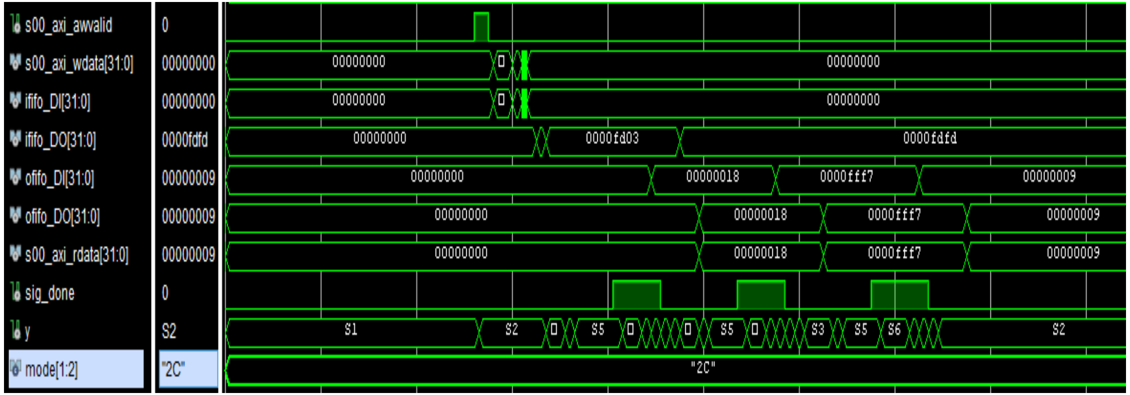
- Loading partial bitstreams into DDR
- Tx and Rx data from smult IP
- Handling ISR and performing DPR on smult IP
    - Includes determining which bitstream to reconfigure by checking switch value



Simulation Results

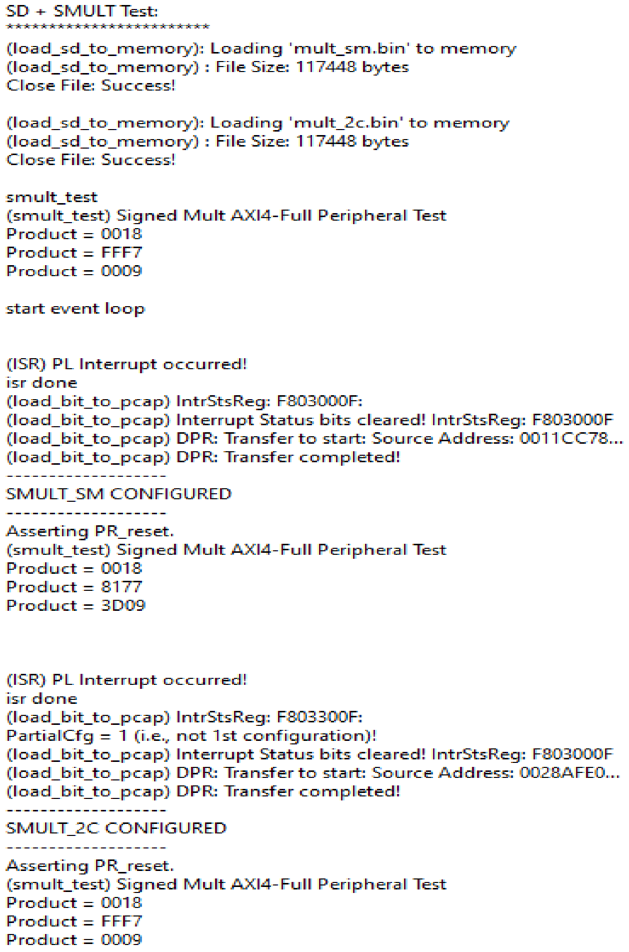
INPUT	OUTPUT	
	RP mode = "2C"	RP mode = "SM"
0x00000406	0x00000018	0x00000018
0x0000FD03	0x0000FFF7	0x00008177
0x0000FDFD	0x00000009	0x00003D09





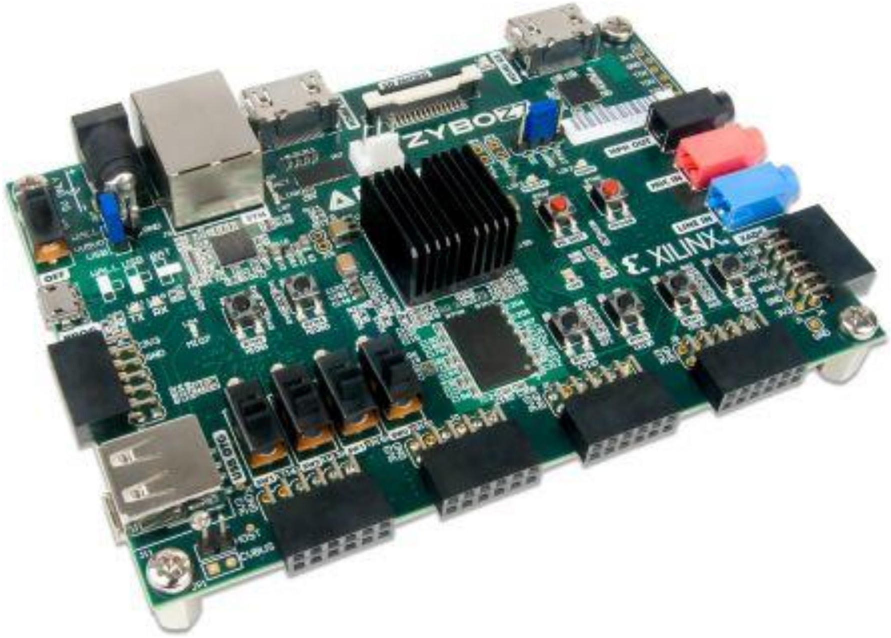
Implementation Results

- Performs as expected and meets all desired system requirements



```
SD + SMULT Test:
*************************
(load_sd_to_memory): Loading 'mult_sm.bin' to memory
(load_sd_to_memory) : File Size: 117448 bytes
Close File: Success!

(load_sd_to_memory): Loading 'mult_2c.bin' to memory
(load_sd_to_memory) : File Size: 117448 bytes
Close File: Success!

smult_test
(smult_test) Signed Mult AXI4-Full Peripheral Test
Product = 0018
Product = FFF7
Product = 0009

start event loop


(ISR) PL Interrupt occurred!
isr done
(load_bit_to_pcap) IntrStsReg: F803000F:
(load_bit_to_pcap) Interrupt Status bits cleared! IntrStsReg: F803000F
(load_bit_to_pcap) DPR: Transfer to start: Source Address: 0011CC78...
(load_bit_to_pcap) DPR: Transfer completed!
--------------------
SMULT_SM CONFIGURED
--------------------
Asserting PR_reset.
(smult_test) Signed Mult AXI4-Full Peripheral Test
Product = 0018
Product = 8177
Product = 3D09



(ISR) PL Interrupt occurred!
isr done
(load_bit_to_pcap) IntrStsReg: F803300F:
PartialCfg = 1 (i.e., not 1st configuration)!
(load_bit_to_pcap) Interrupt Status bits cleared! IntrStsReg: F803000F
(load_bit_to_pcap) DPR: Transfer to start: Source Address: 0028AFE0...
(load_bit_to_pcap) DPR: Transfer completed!
--------------------
SMULT_2C CONFIGURED
--------------------
Asserting PR_reset.
(smult_test) Signed Mult AXI4-Full Peripheral Test
Product = 0018
Product = FFF7
Product = 0009
```

Demo



Benefits of Reconfigurable Computing

- Device cost
- Power, energy, and size constraints
- One chip for multiple functionalities
    - The RP is capable of hosting however many reconfigurable modules (RM's) and thus many applications

Conclusions

Partial Reconfiguration along with the abilities of the Zynq-7000 chip and AXI allow for a very advanced system to be created. There are many applications beyond this that could host very advanced circuits (and very advanced RM's). Although the reconfigurable part of this circuit is rather simple, the most important part is showcasing the DPR functionality.

References

1. https://www.xilinx.com/support/documentation/sw_manuals/xilinx2019_1/ug909-vivado-partial-reconfiguration.pdf
2. https://www.xilinx.com/support/documentation/sw_manuals/xilinx2017_1/ug947-vivado-partial-reconfiguration-tutorial.pdf
3. D. Llamocca, “Reconfigurable Computing” Oakland University. 2020.

Questions?