

**ERACE – Embedded Research Architectures for Co-Design Environments**  
**GEMS – Group of Embedded MicroSystems**

---

University of Ottawa  
School of Information Technology and Engineering  
TEXPO 2005 - October 13, 2005

# Run-Time Reconfigurable Digital Core Testing for Biomedical Instrumentation

---

Mohammad El-Kadri ([melkadri@site.uottawa.ca](mailto:melkadri@site.uottawa.ca))

Mohammed Elbadri ([melbadri@site.uottawa.ca](mailto:melbadri@site.uottawa.ca))

Rami Abielmona ([rabielfmo@site.uottawa.ca](mailto:rabielfmo@site.uottawa.ca))

Voicu Groza ([groza@site.uottawa.ca](mailto:groza@site.uottawa.ca))

---

# 1. Introduction

- What is reconfigurable computing ?
    - Adds the ability for the hardware to be changed or reconfigured during execution
  - What is run-time reconfiguration (RTR) ?
    - Combines characteristics of co-processors with those of reconfigurable computing
  - What is digital testing ?
    - Describes circuit behavior when faults occur
  - What is fault-injection testing ?
    - A fault model which uses Line Stuck-At (LSA) faults
-

## 2. Testing Strategies

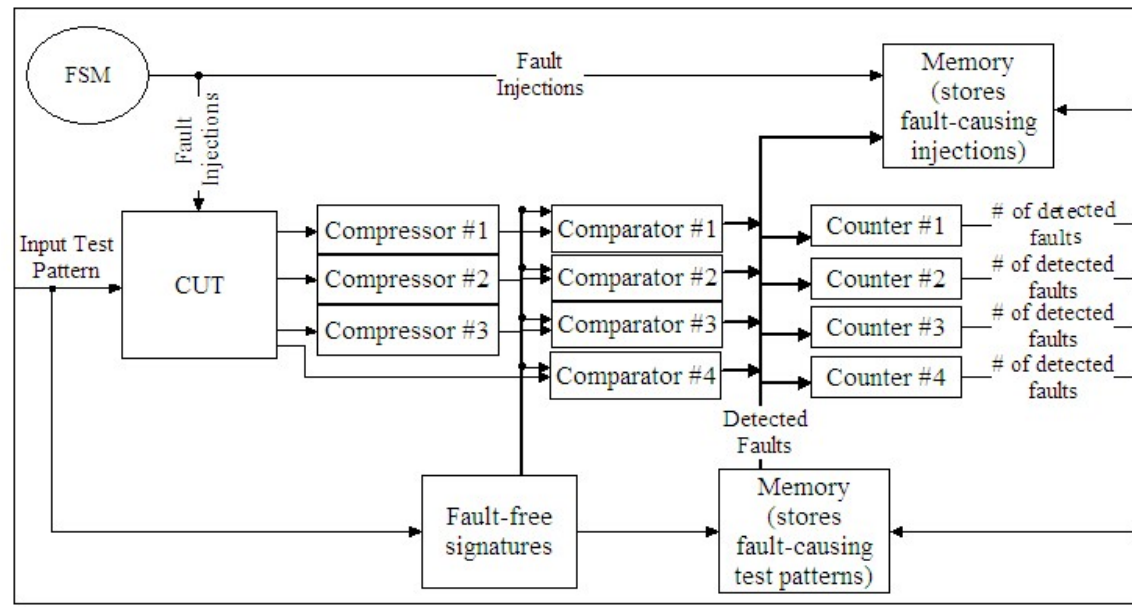


Figure 1. Hardware Test Architecture

### 1. Software

- Common method used to prototype fault models
- Does not pay much attention to real-time details

### 2. Sequential Compile Time Reconfiguration (CTR)

- Equivalent to the contemporary methods used to test circuits
- Performed after the synthesis and mapping steps

### 3. Parallel CTR

- Here, the FSM does not inject any faults into the CUT
- The faults are already synthesized into the circuit!
- Requires  $n * m * 2$  CUTs

### 4. Sequential RTR

- The CUT itself is reconfigured at run-time (hence no FIMs!)
- Innovative fault-injection technique!

### 3. Fault-Injection Multiplexers

- Attached to mutually exclusive wires
- Introduces a stuck-at-0 or a stuck-at-1 signal onto the line depending on the select signal
- Increases the area of the CUT, hence, should be avoided!

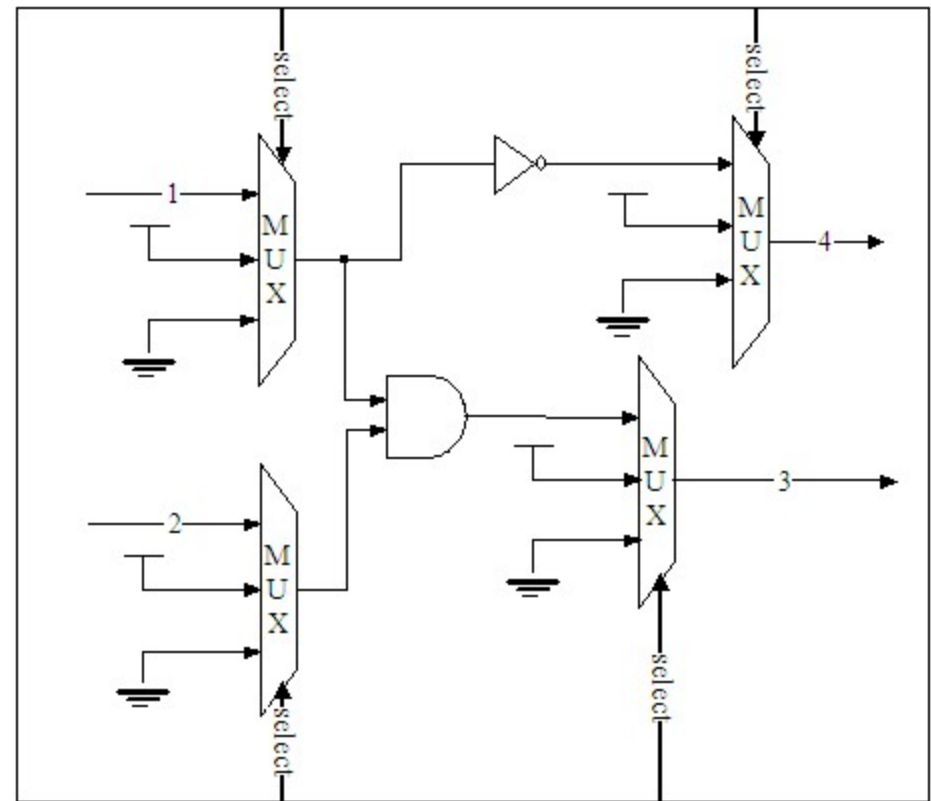


Figure 2. Fault Injection Scheme in Hardware

# 4. ERACE Initial System Architecture

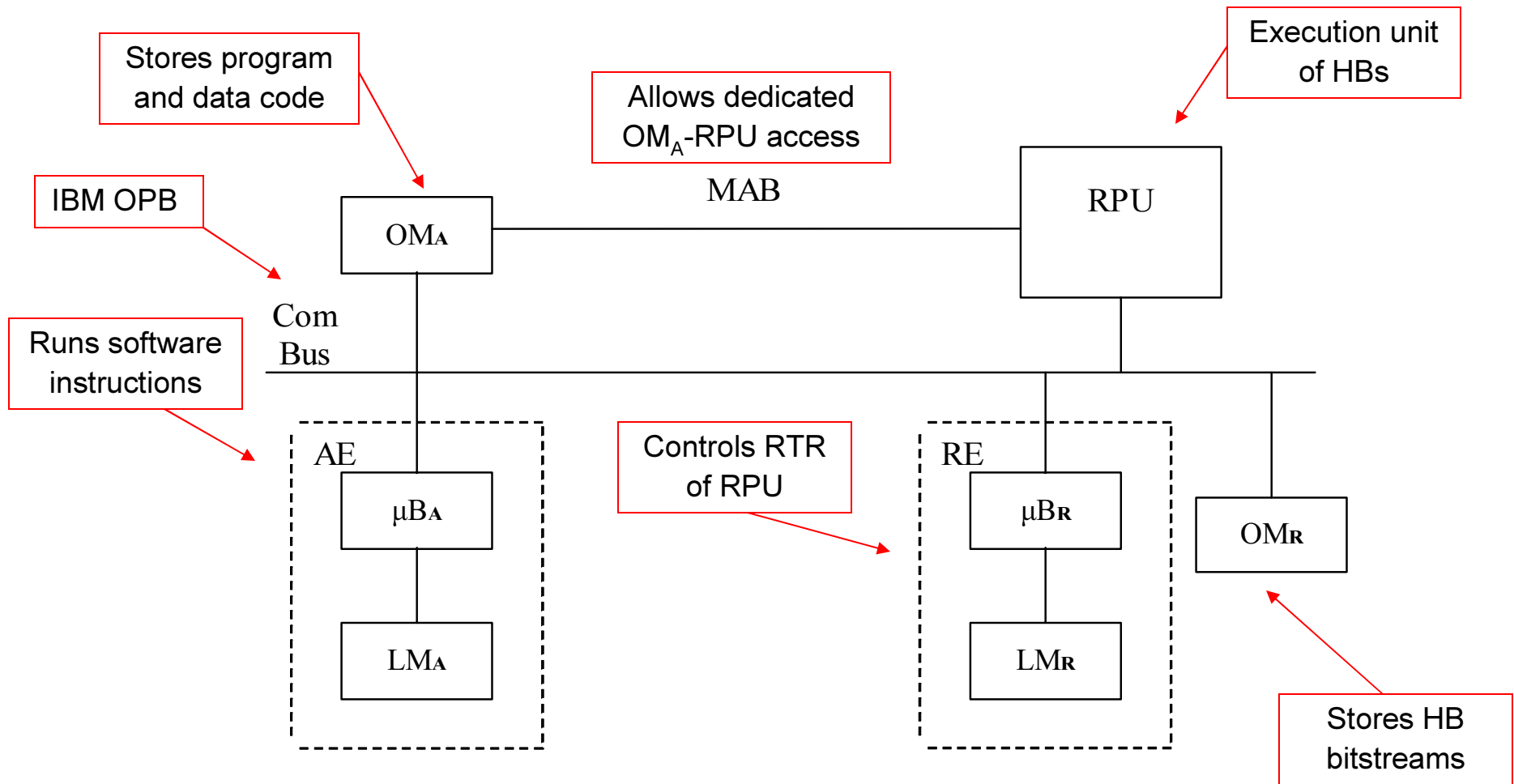


Figure 3. Initial Architecture



# 6. Just-In-Time Compiler

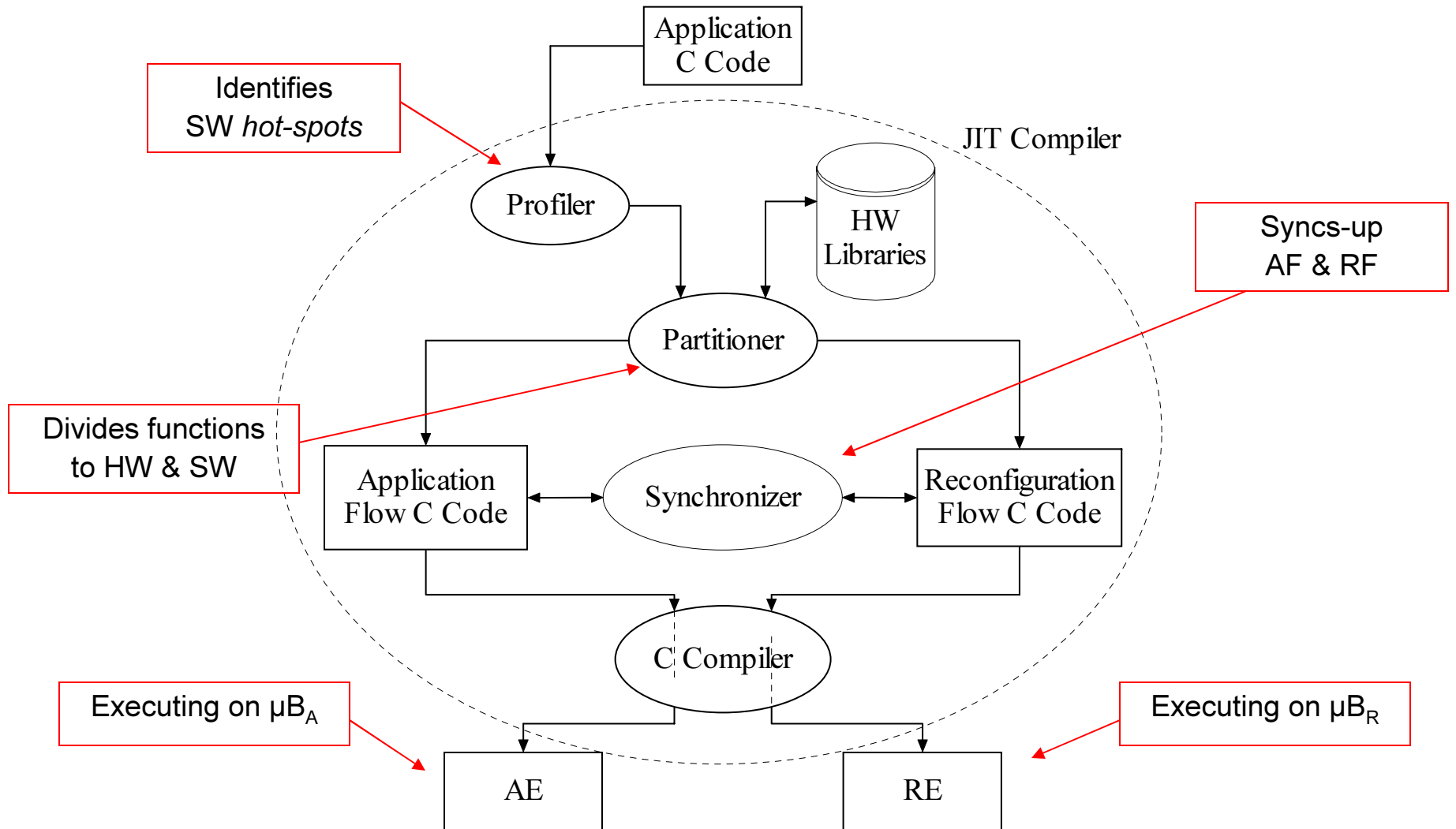


Figure 5. JIT Compiler

# 7. RPU Architecture

- Allows for simultaneous HB execution
- Utilizes its own bus in order to avoid a bottleneck on other buses
- FSL interfaces provide access to HB registers
- $LM_{RPU}$  for HB-to-HB communication

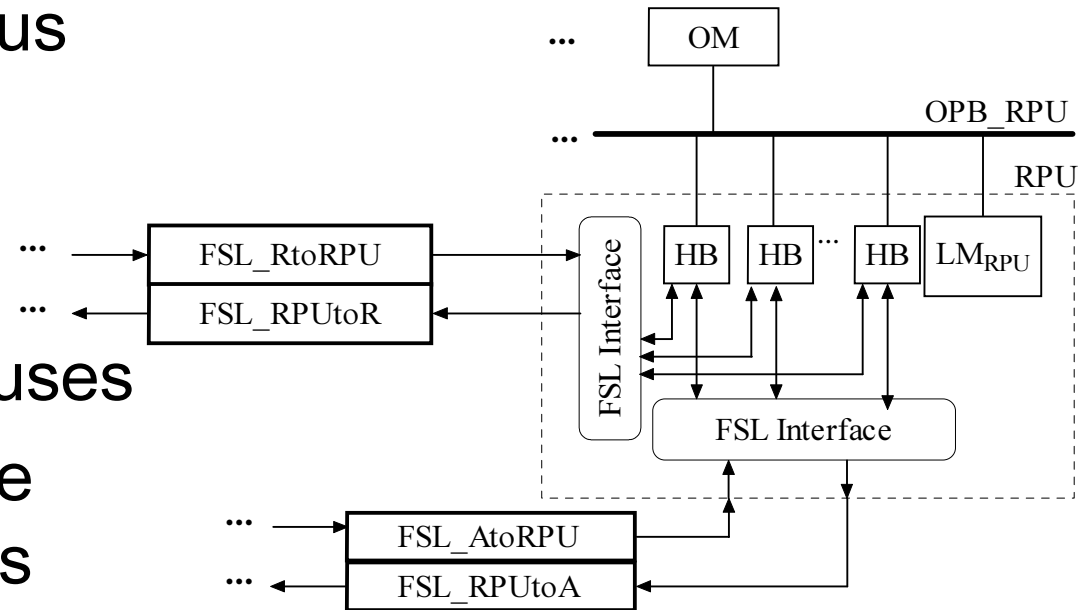


Figure 6. RPU Architecture



## 8. Hardware Block (HB) Architecture

**Input buffer (I-Buffer):** stores all incoming data that has been read from the OMA located in the AE

**Output buffer (O-Buffer):** stores all outgoing data received from the Packer unit

**Interface (I/F) module:** consists of several addressing registers; status and control registers (S/C) that hold all the status and control flags as well as a priority register

**Processing Element (PE):** processes the incoming data in order to fulfill a predefined task

**Packer:** assembles all data sent from the PEs prior to forwarding this information to the O-Buffer once the buffer-write signal is granted by the CU

**Dispatcher:** receives all the data from the I-Buffer and dispatches this information to various PEs for processing

**Control Unit (CU):** performs several operations to ensure proper synchronization amongst different units

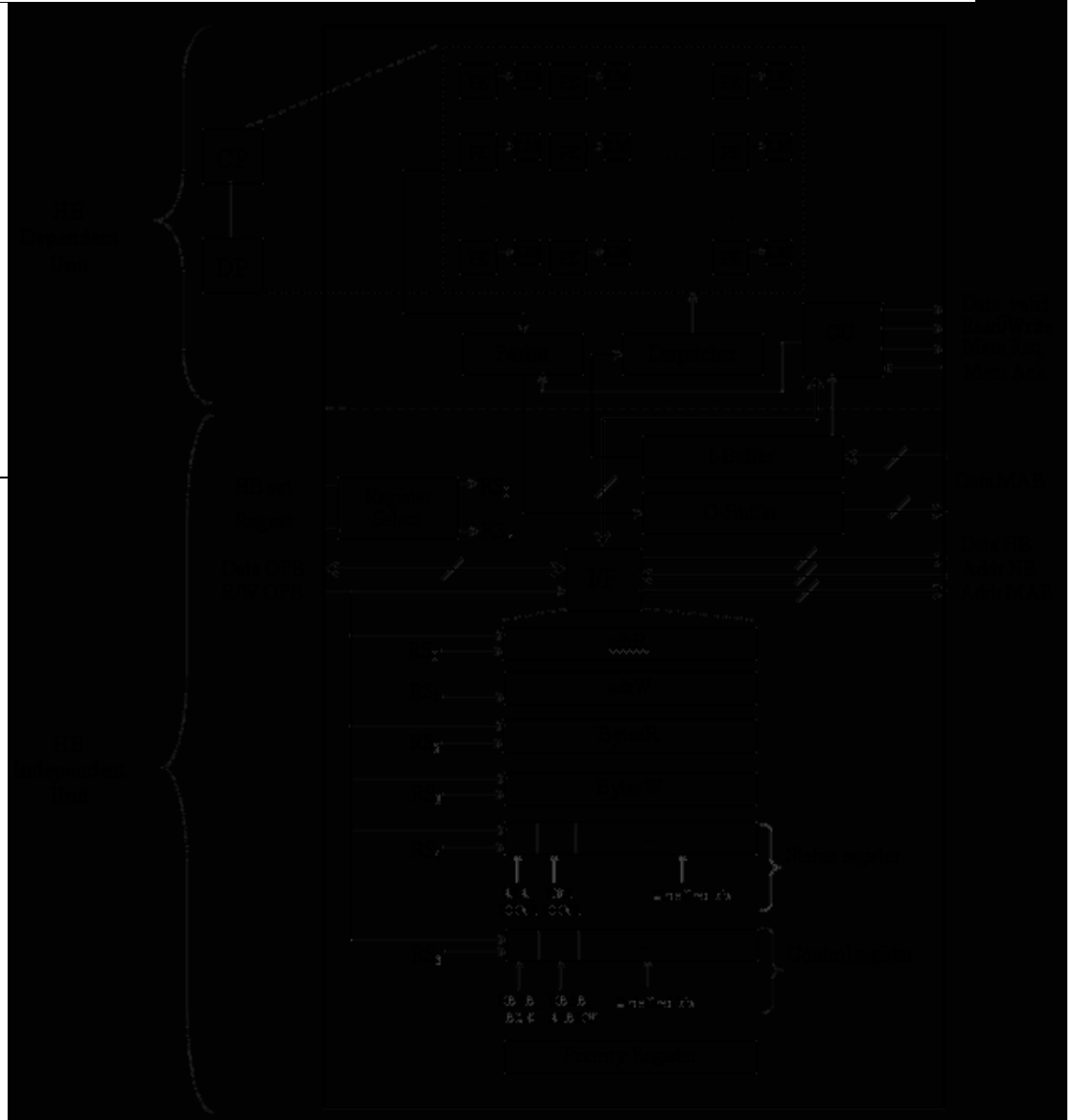


Figure 7. HB Architecture

---

## 9. User Fully Integrated Terminal (UFIT)

- Flagship product of BioSign Corporation
  - Data Acquisition System
    - Microcontroller
    - Pump (cuff)
    - Pressure Sensor
  - Interfaces to a centralized server where calculations are performed
    - Pulse rate, blood pressure, pulse pressure, mean arterial pressure, and can detect arrhythmia
    - Can detect if there is too much noise and interference to acquire accurate readings
    - Can store electronic health records to be used as an aid in diagnostic and disease management by doctors
    - The signal database can also be used in drug research and clinical studies, a market BioSign is targeting in the short-term
-

---

## 10. RTR and UFIT

- We are developing a digital core testing environment, starting with benchmark digital circuits and moving to more complex digital cores (such as those from BioSign)
  - A possible future application-scenario might be:
    - When the measuring process starts, the FPGA contains an HB that tests if the UFIT is operational
    - If UFIT is sane, the current HB is replaced by another one which checks the integrity of the acquired signal
    - If the patient's pulse can be measured (i.e., passed the test of signal integrity), another HB for the pulse acquisition can be uploaded (cleaning artifacts, filters, etc.)
-

# 11. Preliminary Results

- Clock cycle execution time comparison on MicroBlaze running Micrium's uC/OS-II
  - CUTTask: operation of a combinational circuit
  - CUTTaskCompress: operation of a compressor appended to CUTTask circuit
  - MULTTask: 10 by 10 matrix multiplication
  - FFTTask: 8 by 2 matrix FFT
  - LEDFlashTask: Flash LED
- Note that the software multiplication task consumes the most processing resources
  - Ideal for HB partitioning!

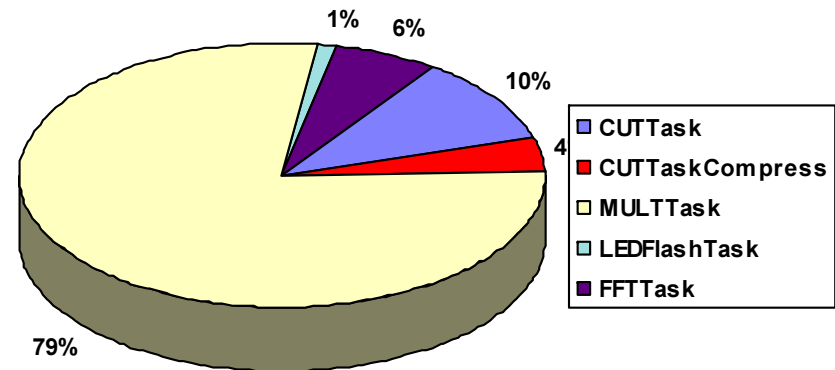


Figure 8. Clock Cycle Distribution

---

## 12. Conclusion

- A digital core testing environment capable of handling variant functions at run-time
  - Implemented on a system-on-chip (SoC) with
    - A soft-core microprocessor handling all real-time deadlines
    - A reconfigurable processing unit allowing for the parallel execution of multiple hardware functional units
  - Just-in-time compiler manages both application and reconfiguration flows
  - Used in testing complex digital circuits and simple cores
  - Mainstream integration into BioSign's UFIT next design
-

---

## 13. Team Member Contributions

- Voicu Groza
    - Supervising professor and **research lab director**
  - Rami Abielmona
    - Ph.D. student prime on **system integration**
  - Mohammed Elbadri
    - M.A.Sc. student prime on **RPU and HB architecture**
  - Mohammad El-Kadri
    - M.A.Sc. student prime on **JIT compiler**
-

---

## 14. Acknowledgements

- The authors gratefully acknowledge the assistance and contribution of the Canadian Microelectronics Corporation (CMC) in providing top-of-the-line workstations and prototyping platforms
- The authors would also like to acknowledge BioSign Corporation (<http://www.biosign.com/>) for their cooperation in providing details about the UFIT device

**THANK YOU!**

Comments/Questions ?

---

References provided upon request