José Antonio Ruiz-Domínguez
Cambridge (UK)
ja_rd@outlook.com / **(+44) 07741188600**

KEY SKILLS

Digital design

- Can design digital logic and map that design onto FPGA or ASIC technology using vendor-agnostic VHDL, Verilog or System Verilog.
- Can work from existing designs, existing specifications or functional requirements.

Digital verification

- Habitual builder of designer-friendly block-level test benches in System Verilog (**not UVM**).
- Can develop small custom ad-hoc SW tools when required. Models for hardware devices in C, support data conversion Python scripts...
- Experience in ISA modelling with ISS (*including auto-generation of ISS from pseudocode*).

Embedded programming in C and assembler

- Knowledgeable in very low-level C programming.
- Knowledgeable in assembly programming for several CPUs, can pick up new ones quickly.
- Some experience with MISRA-C standard, can adapt to tighter rule sets and methodologies.
- Very little experience in embedded Linux or RTOS programming – junior level at best.

SAMPLES OF CODE AND DOCUMENTATION

Some of my open-hardware pet projects can be found online:

https://github.com/jaruiz/light52	Intel 8051 core
https://github.com/jaruiz/light8080	Intel 8080 core
https://github.com/jaruiz	Other assorted pet projects

Some of those CPU cores are quite old but they're functional CPUs and might be useful as code samples. I can set up live demos of any of my projects if requested to.

Documentation samples are also available, like this datasheet for my 8051 core:

https://github.com/jaruiz/light52/blob/master/doc/light52_ds.pdf

EDUCATION

BS in Telecommunications Engineering (Esp. Electronic Systems).

Final year project: Hopfield neural network on FPGA.
(Fixed point vector computation with sparse matrices on Altera Stratix).

RECENT EXPERIENCE IN DIGITAL DESIGN

Digital Design Contractor Arm

Part of a team that's building a new generation of ISP cores. I'm another pair of hands on deck dealing with mundane digital design chores: adaption and repackaging of legacy RTL, creation of new subunits, block level dev TBs, trials of new flows...
System Verilog, VHDL and all the glue tools you can expect (Python, makefiles etc.)

Digital Design Engineer Agile Analog

Assisted in the creation of a gigital-on-top front-end integration flow for mixed signal cores.
In other words, I wrote Python code that interacts with EDA tools and helps put together a composite core out of independent subunits. Python coding and EDA documentation.

Digital Design Contractor Huawei Apr/22 - present

Hired to contribute to a new version of an existing display control chip.

Digital Design Contractor Onsemi Sep/21- Apr/22

I designed and implemented a few tiny digital modules (GPIO, wakeup interrupt controller...).
Included block-level directed test bench. All in System Verilog.

Digital Design Engineer CSR/Qualcomm Sep/13 - Oct/17 + May/19 – Sep/21

I designed and implemented IP blocks to be integrated in larger subsystems, in Verilog or System Verilog.
I also did a moderate amount of testing in System Verilog and C and/or assembler for our DSP, including the development of several test utilities for our DSP in Python/C/Verilog.

Projects and roles

A necessarily brief summary of my 5+ years in CSR/Qualcomm:

1. Project lead in power management module for IoT SoC just successfully taped out.
    a) Always-On block manages power state of the rest of the SoC.
    b) Design, implementation (System Verilog), documentation.
2. Built several small blocks for audio processing chip (successfully taped out in 2016).
    a) Decryption module for Quad SPI flash data stream.
    b) Configurable arbitration module for the same.
    c) Design, implementation, documentation, some testing, SW bring-up support.
3. Part of a 2-person team that implemented the 'applications subsystem' in that chip.
    a) Integration of several IP blocks and two proprietary 32-bit DSPs.
    b) Memory interconnect fabric (proprietary bus), arbiters, interrupt logic...
    c) Subsystem-level tests (DSP SW cooperating with test RTL).
4. Developed a proprietary *random instruction sequence* test scheme for our in-house DSP.
    a) Generic: can be retargeted to any common ISA within certain constraints.
    b) ISA described in custom DSL: including all encoding and semantic information.
    c) Auto-generated instruction set simulator in C from ISA description (bit-perfect golden model).
    d) Constrained random sequence generator also auto-generated from ISA description.
    e) Strong complement to directed test suite.
    f) Mostly Python, C and auto-generated C.
    g) A one-person effort that yielded a small system that was largely state of the art.

I've also worked in the maintenance of our proprietary 32-bit fixed point DSP Kalimba.

I left the company in late 2017 only to re-join in mid-2019 to perform a similar role.

FPGA Engineer	**Kaleao/Bamboo**	**Nov/18 – May/19**

I helped maintain, develop and refactor an existing FPGA project (Vivado on Zynq).

I had time to write a single IP block (IP/TCP on-the-fly packet checksum insertion) in Verilog along with its test bench in System Verilog. I also reviewed and refactored existing code in Verilog and VHDL.

My day to day was spent mostly writing TCL scripts for Vivado and System Verilog test bench code.

COMPLETE EMPLOYMENT HISTORY

Jobs not related to digital design or not recent or interesting enough have been excluded from this summary. You can find a complete chronological employment history in my LinkedIn profile:

www.linkedin.com/in/josé-a-ruiz-domínguez-5978185a

A more exhaustive version of this CV is available on demand too.