**The Robotic Virtual Machine (RVM)**

It is now time for perhaps the second most exciting part for this initial version of the SCARL toolchain – an implementation of the Robotic Virtual Machine to execute Abstract Robot Control Language (ARCL) code!

The properties of the machine have been described in the ARCL document, but I have copied them here as well:

The properties of the abstract machine that ARCL code executes on

1. Has a 8 general purpose 16 bit registers.
2. Has a special implicit frame pointer register
3. Special register for returning function values called “FRR”
4. Has a stack with unbounded memory
5. Has unbounded heap space.
6. Has a space of read-only memory that is set before execution of the program. This is where program code is loaded into.
7. Each atomic object is a 2 byte word that can be interpreted in multiple different ways
8. Has a set of peripheral pins that may be set to input or output, and have power on, or read the power from it

The Robotic Virtual Machine is to realize these properties in a concrete way so that ARCL code can be directly executed.

As an exercise for myself, I will see if it is possible to write the RVM in the Ada programming language.

**Configuration of the “physical part” of the device**

To keep the process self-contained, we are to specify the physical devices of the tool through command line options. Simply specify the number of pins with the -pins option, or 10 pins is the default number.