Searduino - - C/C++, simulator, stubs

January 2012

Table of Contents

1 Background

The authors of Searduino loves developing code for Arduino. We love using Arduino and we believe that developing code for Arduino has been made significantly easier for not-so-experienced-developers with the Arduino IDE. However, for some of us it is easier to develop code in our favorite editors and build and upload via the command line.

Searduino was initially created to make it possible to automate the building of your arduino program which has to be done outside of the Arduino IDE. Once we had the build and linking up and running we quickly noticed that it wouldn't take that much to make it possible to turn your Arduino program into a program executing on your local computer. The writes and reads on pins in your Arduino program were 'translated' in to reads and writes on stdin/stdout, which we used to create a simple simulator communicating via a pipe. We saw the potential of the simulator and decided to write a "proper" API for it instead. To make the simulator more usable for quick checks and for people preferring GUIs we started to write a simulator GUI in Python, so we added a Python extension to the simulator API. After a while we started hacking on a GUI frontend in Java. And here we are right now....

Hope you like it! And feel free to join us!

2 Abbreviations

- Arduino program - a program written for the Arduino board. Uses only the Arduio and avr APIs.
- stub - Type of board. When building the software to run locally on your computer and not building for real Arduino boards we use the word stub we use this word. A better name would perhaps have been sim or simulaor but stub it is.
- Faked Arduino - library implementing the Arduino and avr APIs
- Streamed input/output - instead of a fullblown simulator GUI Searduino provides you with a stdin/stdout interface. This can be used to script (bash, Python..) your test cases. Using programs such as netcat you can also run the Arduino program on one PC and the test on another PC.
- Java interface - All of the simulation features are offered by a C API s well as via a Java API.
- local computer - the computer you're developing your code on
- Jearduino - The grphical simulator, written in Java.
- Pearduino - The grphical simulator, written in Python/gtk. Obsoleted.
- Python interface - Most of the simulation features are offered by a C API s well as via a Python API. This API is deprecated.

3 Layers

3.1 Faked Arduino layer - the Arduino stub

3.2 Communication layer -