LGPL-2.1 license

3 stars 2 forks 1 watching Branches Activity
Tags

Public repository

6 Branches 9 Tags Go to file t Go to file Add file Code

akashkapashia Update README.md 8e8f654 · 2 years ago

board-manager	release version 0.5	6 years ago
docs	improve unlock description	7 years ago
examples	add example code for HC-04 ultrason...	8 years ago
sduino	modify platform.txt for manual install	6 years ago
test	Merge branch 'flat' into development...	7 years ago
tools	bump stm8gal to new upstream versi...	6 years ago
.gitignore	add a basic regression test script	7 years ago
.gitmodules	automatic build of stm8flash and stm...	7 years ago
CHANGELOG.md	release version 0.5	6 years ago
LICENSE	Initial commit	9 years ago
README.md	Update README.md	2 years ago
mkdocs.yml	update docs, add info about cross-co...	7 years ago
package_sduino_stm8_index.js...	Update package_sduino_stm8_index.j...	2 years ago

Sduino: Small Devices Arduino

Getting started on the STM8 the easy way.

An Arduino-like programming API that can be used from within the Arduino IDE or for Makefile-controlled builds.

- Project website for more information on supported hardware and the programming API
- github issue tracker for bug reports and other issues tightly related to the repository content

- The STM8 board of the stm32duino forum: For general discussions and suggestions

Since this project is based on the SDCC Small Devices C compiler, I called it "Small Devices -uino" or "Small-duino". It is entirely based on free tools that are available for Linux, MacOS, and Windows: SDCC, make, and stm8flash.

This project is not supposed to be “better than Arduino”. It’s purpose is to give you a head start into a different CPU architecture if you happen to have a professional need or a private desire for it.

Installation

Starting with version 0.3.0 automatic IDE integration is supported via the Arduino Boards Manager. This is the recommanded way of installation now. For a manual non-IDE installation please check the manual installation instructions.

Start the Arduino-IDE. In *File->Preferences*, *Settings* tab, enter

> https://github.com/akashkapashia/stm8_arduino/raw/master/package_sduino_stm8_index.json

as an *Additional Boards Manager URL*.

- Open *Tools->Board:...->Boards Manager*
- Find Sduino by typing 'sd' into the search line
- Click on the list entry
- Click on *Install*.

Now you should find a new entry *STM8S Boards* in the list at *Tools->Board:...*

- Choose *STM8S103F3 Breakout Board* from the list
- open the standard Blink example from *File->Examples->01. Basics->Blink*
- compile it by hitting *Verify*

Known issues

The Arduino IDE version 1.8.7 is buggy, use 1.8.8 instead.

IDE error message **select upload port first** (IDE v1.8.7 only): Update to IDE 1.8.8 or apply this workaround.

Very slow compilation after using "save as" (IDE v1.8.7 only): After saving a file under a different name the CPU load maxes out for one CPU core and the IDE reacts extremly slow. Exit the Arduino IDE and re-open it. Update to IDE 1.8.8.

bash.exe: warning: could not find /tmp, please create!: Shows up on windows systems. Annoying, but harmless. Workaround This whole issue will be fixed in the upcoming 0.5 release by using busybox for all scripts on windows.

Included libraries

Most parts of the Arduino core system and some Arduino libraries are already ported to C-syntax. The resulting API is still very close to the C++ version and porting an existing application is not hard. Check out the migration guide for details.

Communication

- SPI: Real hardware-SPI up to 10MHz.
- Wire: Port of the stock Wire library for I2C communication (with improvements)

- I2C: Port of the I2C master library by Wayne Truchsess
- HardwareSerial: The standard serial interface.

Displays

- LiquidCrystal: HD44780 based text LCDs
- LiquidCrystal_I2C: HD44780 based text LCDs with I2C converter backpack
- LiquidCrystal_pcf2119: PCF2119 based text LCDs with I2C connection
- PCD8544: Monochrome graphical LCD based on the PCD8544 controller like the Nokia 5110 display. SPI mode only.
- Mini_SSD1306: SSD1306-based monochrome OLED displays with 128x64 pixels. I2C support only.

README LGPL-2.1 license

- EEPROM: Port of the stock EEPROM library for accessing the buildin EEPROM

Motor control

- Stepper: Stepper motors with 2, 4 or 5 phases.
- Servo: Up to 12 servos using only 1 timer.

Compatibility with the Arduino world

Since there is no free C++ for the STM8, it is impossible to do a full 1:1 port of the whole enviroment as is has been done for the STM32 and the ESP8266.

This is not a drop-in replacement for an AVR, but thanks to some C preprocessor magic the programming API is still very, very similar and it is often enough to just move over the opening bracket of the class instanciation statement and to replace the dot in a method call for an underscore. Check the migration guide for an overview.

Supported Systems:

Arduino IDE versions 1.8.8, 1.8.7, 1.8.5 and 1.6.13 are tested, but any version >=1.6.6 should work.

Version 1.8.7 might require a workaround if you see an error message "select upload port first".

- Linux 64 bit: Tested on Mint 19/Ubuntu 18.04
- Linux 32 bit: Tested on Mint 19/Ubuntu 18.04
- Windows: Tested on Windows 7. Sduino might work on XP (not tested), but the ST-Link/V2 driver is not available anymore for XP.
- MacOS: tested on 10.13.

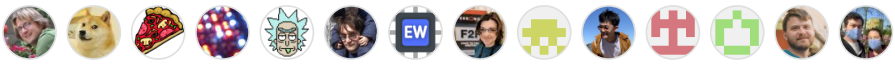
Releases 1

aa Latest
on Dec 17, 2023

Packages

No packages published

Contributors 24



+ 10 contributors

Deployments 1

github-pages 2 years ago

Languages

C 80.1% **C++** 9.6% **Makefile** 4.4% **HTML** 4.2% **Shell** 1.2% **POV-Ray SDL** 0.3% **Other** 0.2%