WiringPi C To C++ Library Port Summary

Authorship

Gordon Henderson is the author of the original WiringPi library set. Aaron Abassi made these further listed alterations to the original work in order to produce the C++ (2014) version as well as this project summary document. The C++ version is derived directly from the original work of Gordon Henderson and I claim no copyright on the derived work though it is a derived work, leaving the derived work strictly under the same copyright license terms as Gordon Henderson's original work.

Goal

To port the WiringPi library and tools to the C++ (2014) programming language in order to simplify the addition of future trivial, I2C & SPI compliant extensions and fast look-up tables. To identify and implement straightforward opportunities to make use of templates to replace multiple functions and/or variables. To maintain linker compatibility with the original C programming language library version in order to continue to serve users of other programming languages.

Conclusions

The result is a library which is linker compatible, and therefore the original examples provided with the WiringPi library which remained in the C programming language link against the new C++ library versions without modification. A total of 17 function definitions are replaced with 4 function templates. A total of 13 (16 conditionally) variable definitions are replaced with 5 variable templates.

Observations

What follows is a summary of all the changes which were made to the source distribution. The C++ WiringPi source files used in this demonstration are based on WiringPi source code dated June 8th 2017 using the URL "git://git.drogon.net/wiringPi". All source files with the extension ".c" were renamed to the extension ".cpp" except those source files in the "examples" directory. Likewise all "Makefile" were modified except the one in the "examples" directory, which will continue to make use of "gcc" (the GNU C programming language compiler). In all other instances the "Makefile" were converted to C++ format and use "g++" (the GNU C++ programming language compiler).

WiringPi Library (wiringPi directory)

---

rht03.c → rht03.cpp

myAnalogRead → Renamed local identifier “try” to “attempt”

---

ds18b20.c → ds18b20.cpp

ds18b20Setup → Type Cast Return Of "malloc"

---

softPwm.c → softPwm.cpp

softPwmCreate → Type Cast Return Of "malloc"

WiringPi Library (wiringPi directory)

---

wiringPi.c → wiringPi.cpp

gpio_t → Introduced Type Alias

pinToGpioRx → Introduced Variable Template
pinToGpioR1 → Refactored As Array Reference
pinToGpioR2 → Refactored As Array Reference

physToGpioRx → Introduced Variable Template
physToGpioR1 → Refactored As Array Reference
physToGpioR2 → Refactored As Array Reference

gpioToLowHi → Introduced Variable Template
gpioToGPSET → Refactored As Constant Pointer
gpioToGPCLR → Refactored As Constant Pointer
gpioToGPLEV → Refactored As Constant Pointer
(conditional) gpioToEDS → Refactored As Constant Pointer
(conditional) gpioToREN → Refactored As Constant Pointer
(conditional) gpioToFEN → Refactored As Constant Pointer
gpioToPUDCLK → Refactored As Constant Pointer

gpioToPwm → Introduced Variable Template
gpioToPwmALT → Refactored As Constant Pointer
gpioToPwmPort → Refactored As Constant Pointer

gpioToClk → Introduced Variable Template
gpioToGpClkALT0 → Refactored As Constant Pointer
gpioToClkCon → Refactored As Constant Pointer
gpioToClkDiv → Refactored As Constant Pointer

wpiExtensions.c → wpiExtensions.cpp

do_extension_t → Introduced Type Alias
trivial_setup_t → Introduced Type Alias
i2c_setup_t → Introduced Type Alias
spi_setup_t → Introduced Type Alias
spi_port_setup_t → Introduced Type Alias
extensionFunctionStruct → Simplified Member “function” Using Type Alias
extractStr → Type Cast Return Of "calloc"
doExtensionTrivial → Introduced Function Template
doExtensionI2C → Introduced Function Template
doExtensionSPI → Introduced Function Template
doExtensionSPIPort → Introduced Function Template
doExtensionMcp23008 → Replaced By Function Template Instance
doExtensionMcp23016 → Replaced By Function Template Instance
doExtensionMcp23017 → Replaced By Function Template Instance
doExtensionMcp23s08 → Replaced By Function Template Instance
doExtensionMcp23s17 → Replaced By Function Template Instance
doExtensionPcf8574 → Replaced By Function Template Instance
doExtensionAds1115 → Replaced By Function Template Instance
doExtensionPcf8591 → Replaced By Function Template Instance
doExtensionPseudoPins → Replaced By Function Template Instance
doExtensionBmp180 → Replaced By Function Template Instance
doExtensionHtu21d → Replaced By Function Template Instance
doExtensionMax31855 → Replaced By Function Template Instance
doExtensionMcp3002 → Replaced By Function Template Instance
doExtensionMcp3004 → Replaced By Function Template Instance
doExtensionMax5322 → Replaced By Function Template Instance
doExtensionMcp4802 → Replaced By Function Template Instance
doExtensionSn3218 → Replaced By Function Template Instance

WiringPi Library (wiringPi directory)

Makefile

CC → CXX
gcc → g++
CFLAGS → CXXFLAGS
CXXFLAGS → Appended -std=c++14
wiringPi.c → wiringPi.cpp
wiringSerial.c → wiringSerial.cpp
wiringShift.c → wiringShift.cpp
piHiPri.c → piHiPri.cpp
piThread.c → piThread.cpp
wiringPiSPI.c → wiringPiSPI.cpp
wiringPiI2C.c → wiringPiI2C.cpp
softPwm.c → softPwm.cpp
softTone.c → softTone.cpp
mcp23008.c → mcp23008.cpp
mcp23016.c → mcp23016.cpp
mcp23017.c → mcp23017.cpp
mcp23s08.c → mcp23s08.cpp
mcp23s17.c → mcp23s17.cpp
sr595.c → sr595.cpp
pcf8574.c → pcf8574.cpp
pcf8591.c → pcf8591.cpp
mcp3002.c → mcp3002.cpp
mcp3004.c → mcp3004.cpp
mcp4802.c → mcp4802.cpp
mcp3422.c → mcp3422.cpp
max31855.c → max31855.cpp
max5322.c → max5322.cpp
ads1115.c → ads1115.cpp
sn3218.c → sn3218.cpp
bmp180.c → bmp180.cpp
htu21d.c → htu21d.cpp
ds18b20.c → ds18b20.cpp
rht03.c → rht03.cpp
drcSerial.c → drcSerial.cpp
drcNet.c → drcNet.cpp
pseudoPins.c → pseudoPins.cpp
wpiExtensions.c → wpiExtensions.cpp
$(SRC:.c=.o) → $(SRC:.cpp=.o)

Device Library (devLib directory)

---

lcd128x64.h

Introduced → Conditional extern “C” block (missing)

Makefile

CC → CXX
gcc → g++
CFLAGS → CXXFLAGS
CXXFLAGS → Appended -std=c++14
ds1302.c → ds1302.cpp
maxdetect.c → maxdetect.cpp
piNes.c → piNes.cpp
gertboard.c → gertboard.cpp
piFace.c → piFace.cpp
lcd128x64.c → lcd128x64.cpp
lcd.c → lcd.cpp
scrollPhat.c → scrollPhat.cpp
piGlow.c → piGlow.cpp
$(SRC:.c=.o)→ $(SRC:.cpp=.o)

GPIO Application (gpio directory)

---

gpio.c → gpio.cpp

usage → Declared Constant
findExecutable → Type Cast Return Of "malloc"
changeOwner → Parameters “cmd” and “file” Declared Constant
moduleLoaded → Parameter “modName” Declared Constant
doLoad → Locals “module1”, “module2”, “file1” and “file2” Declared Constant
doI2Cdetect → Type Cast Return Of "malloc"

---

readall.c → readall.cpp

alts → Declared Constant
physNames → Declared Constant
abReadAll → Local “type” Declared Constant

Makefile

CC → CXX
gcc → g++
CFLAGS → CXXFLAGS
CXXFLAGS → Appended -std=c++14
gpio.c → gpio.cpp
readall.c → readall.cpp
pins.c → pins.cpp
$(SRC:.c=.o) → $(SRC:.cpp=.o)

WiringPi Daemon (wiringPiD directory)

---

drcNetCmd.h

drcNetComStruct → Removed Orphaned “comDat” Identifier

---

network.c → network.cpp

getResponse → Type Cast Return Of "malloc"

---

wiringpid.c → wiringpid.cpp

main → Type Cast Return Of "malloc"

Makefile

CC → CXX
gcc → g++
CFLAGS → CXXFLAGS
CXXFLAGS → Appended -std=c++14
wiringpid.c → wiringpid.cpp
network.c → network.cpp
runRemote.c → runRemote.cpp
daemonise.c → daemonise.cpp
$(SRC:.c=.o) → $(SRC:.cpp=.o)