

BridgerSoc1

Getting Started Guide

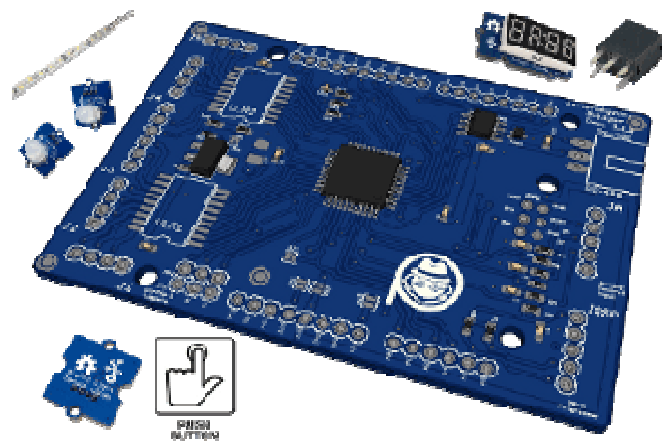


Copyright 2011, 2012, 2013
www.patternagents.com

<http://www.patternagents.com/products.html>

Revision 3.1.11
PCD #272A

BridgerSoc1



Contents :	
Overview	2
Getting Started	2
Operating Modes	6
Example Project	12
Schematic and Layout	14
PSoC Pin Assignment	20
BridgerSoc1 PCB Pin Assignment	21
Custom Configuration Jumpers	22
Frequently Asked Questions (FAQ)	24

Congratulations on your purchase of the “BridgerSoc1” board!

The PatternAgents “BridgerSoc1” board has the following features :

- System-on-Chip Standalone Mode (Switches/Analog Inputs Control the Outputs)
- Arduino Uno/Leonardo/Mega/Due Compatible Shield
- Selectable 3.3Volt-5.0Volt Operation
- Sixteen (16) high power darlington control outputs
- Seven (7) analog inputs (sensors, potentiometers)
- Five (5) digital inputs (switches, sensors)
- UART, I2C, SPI, RS-485/DMX Interfaces
- Selectable mode of operation (Manual or Interface/Shield controlled)

Overview :

BridgerSoc1 has sixteen (16) programmable digital blocks, and twelve (12) programmable analog blocks that come pre-programmed for a variety of uses, so it is ready to use right out of the box. It also includes a Command Line Interface (CLI) that gives you the ability to control it right from a terminal or the programming language of your choice, making it very easy to get started.

Using a set of configuration jumpers, you can interface to the BridgerSoc1 using simple switches or sensors, so anyone can use it without having to learn how to program. If you want to control Bridger from a program you can use the UART (CLI), I2C/Two-Wire Bus, or the SPI interface - giving you the maximum flexibility to configure your other shields without I/O conflicts.

Stack-Thru Models (optional request):

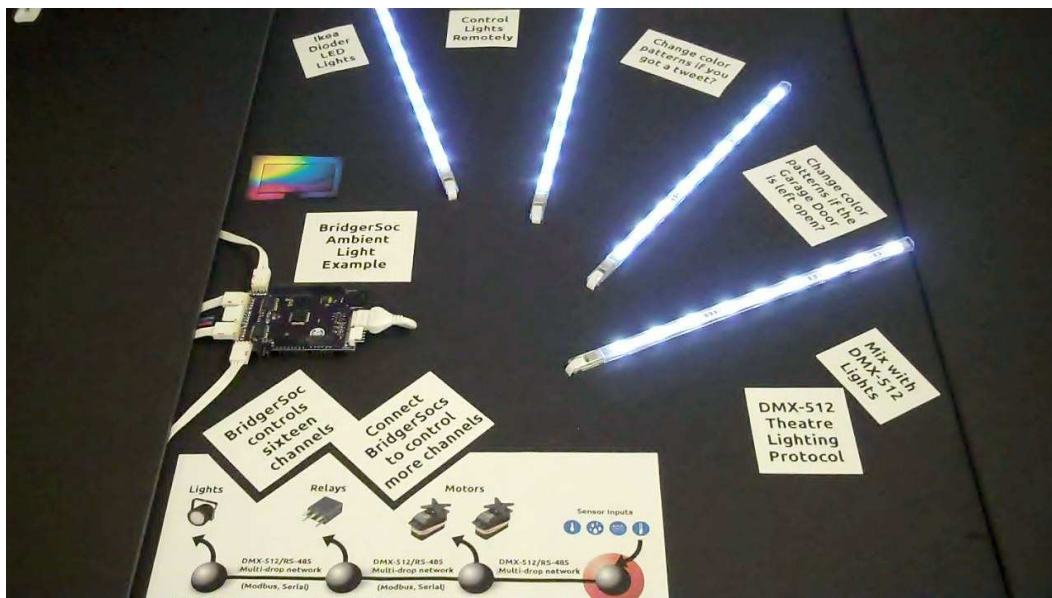
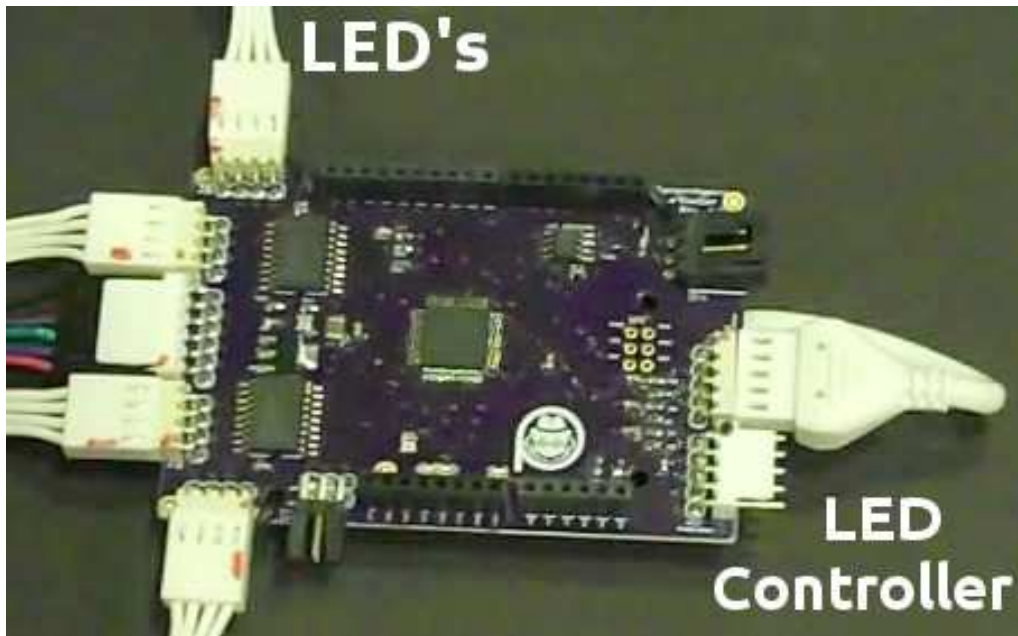
As an option, PatternAgents “BridgerSoc1” boards can be populated with “Stack-Thru” connectors, which allow Arduino Shields to be added to the top or bottom of the Arduino board stack.

NOTE: *Always make sure the stacking pins don't contact a metal surface (like a metal table).*

Getting Started (StandAlone):

To test the PatternAgents “BridgerSoc1” board as shipped (in Standalone Mode), connect four (4) Color LED lights to J1, J2, J5, J6 and connect to a power outlet. After a moment, the lights will begin changing colors and shifting automatically.

NOTE: *Always make sure the power is OFF, before adding or removing connections!*



Getting Started (Arduino):

To connect the PatternAgents “BridgerSoc1” board to an Arduino CPU, carefully align the pins between the two boards and gently press together to connect.

NOTE: *Always make sure the power is OFF, before adding or removing connections!*

Arduino Serial Mode (Mode 0):

Make sure that all three (3) jumpers are connected between JP1 and JP2 to select Mode 0. Connect the Arduino USB interface to your host computer and open a “Hyper-terminal” or other Terminal emulation program to the serial port for your Arduino (COMx:). Set the Terminal emulation program to 9600 Baud , N, 8, 1 (which is usually the default). Set the Terminal emulation program to “Echo” so that you can see the characters typed.

The Sixteen Lighting Channels reside from a default DMX address of 40 to 55 (decimal). So the User would set the lighting channels all to maximum brightness using :

```
> <DMX> 40 255<CR>
> <DMX> 41 255<CR>
> <DMX> 42 255<CR>
> ...
> <DMX> 55 255<CR>
```

To read the setting of a DMX channel, just use the question mark for data :

```
> <DMX> 40 ?<CR>
```

and the BridgerSoc1 will respond :

```
> <DMX> 40 255<CR>
```

CLI Help : (enter ?)

```
> ?
Help Topics
-----
...
```

BridgerSoc1 Operating Modes :

The BridgerSoc1 has three (3) Configuration Jumpers which select one (1) of eight (8) operating modes, namely :

- 0 : 000 : UART CLI Mode
- 1 : 001 : I2C Arduino Mode
- 2 : 010 : I2C Leonardo Mode
- 3 : 011 : SPI Mode
- 4 : 100 : Test Mode
- 5 : 101 : DMX-512 Slave Mode
- 6 : 110 : Bike Light Mode : (Run/Stop/Left-Turn/Right-Turn)
- 7 : 111 : Standalone Mode - Dioder Manual Control

The Configuration Jumpers are labeled JP1/JP2, pins 1 thru 3.
The BridgerSoc1 is shipped in “Test Mode (4)” so that it automatically Starts changing the lighting colors in a shifting pattern.

BridgerSoc1 Operating Mode O : UART CLI Mode :

```
# -----
# Mode o - UART CLI Mode
# -----
# In this mode the BridgerSoc1 communicates with a host using a
# Command Line Interface via a UART serial interface,
# using Arduino Pins Do (TXD) & D1 (RXD)
#
# The Arduino default of 9600 Baud, 8 Bits, No Parity, 1 Stop Bit is used.
# Open a “HyperTerminal” or other program interface to the Arduino serial Port.
#
# The Sixteen Lighting Channels reside from a default DMX address of 40 to 55 (decimal).
# So the User would set the lighting channels all to maximum brightness using :
#
#     > <DMX> 40 255<CR>
#     > <DMX> 41 255<CR>
#     > <DMX> 42 255<CR>
#     > ...
#     > <DMX> 55 255<CR>
#
# To read the setting of a DMX channel, just use the question mark for data :
#     > <DMX> 40 ?<CR>
#
# and the BridgerSoc1 will respond :
#     > <DMX> 40 255<CR>
#
# CLI Help : (enter ?)
#
#     > ?
#     Help Topics
#     -----
#     ...
#
```

BridgerSoc1 Operating Mode 1 : I2C Arduino Mode :

```
# -----
# Mode 1 - I2C Arduino Mode
# -----
# In this mode the BridgerSoc1 communicates with an Arduino host using
# the original I2C Pins for the Arduino Uno, A4 & A5.
# #
# The Arduino default of 400K speed I2C Bus is used.
# Open a “HyperTerminal” or other program interface to the Arduino serial Port.
#
# The Sixteen Lighting Channels reside from a default I2C address of 40 to 55 (decimal).
# So the User would set the lighting channels all to maximum brightness using :
#
#       > <I2C_Write> 40 255<CR>
#       > <I2C_Write> 41 255<CR>
#       > <I2C_Write> 42 255<CR>
#       > ...
#       > <I2C_Write> 55 255<CR>
#
```

BridgerSoc1 Operating Mode 2 : I2C Leonardo Mode :

```
# -----
# Mode 2 - I2C Leonardo Mode
# -----
# In this mode the BridgerSoc1 communicates with an Arduino host using
# the I2C Pins for the Arduino Leonardo, SCL & SDA.
# #
# The Arduino default of 400K speed I2C Bus is used.
# Open a “HyperTerminal” or other program interface to the Arduino serial Port.
#
# The Sixteen Lighting Channels reside from a default I2C address of 40 to 55 (decimal).
# So the User would set the lighting channels all to maximum brightness using :
#
#       > <I2C_Write> 40 255<CR>
#       > <I2C_Write> 41 255<CR>
#       > <I2C_Write> 42 255<CR>
#       > ...
#       > <I2C_Write> 55 255<CR>
#
```


BridgerSoc1 Operating Mode 3 : SPI Mode :

```
# -----  
# Mode 3 - SPI Mode  
# -----  
#  
# In this mode the BridgerSoc1 communicates with an Arduino host using  
# the SPI Pins for the Arduino Leonardo, P13, P12, P11, & P10.  
#  
# The Sixteen Lighting Channels reside from a default SPI address of 40 to 55 (decimal).  
# So the User would set the lighting channels all to maximum brightness using :  
#  
#     > <SPI_Write> 40 255<CR>  
#     > <SPI_Write> 41 255<CR>  
#     > <SPI_Write> 42 255<CR>  
#     > ...  
#     > <SPI_Write> 55 255<CR>  
#
```

BridgerSoc1 Operating Mode 4 : Test Mode :

```
# -----  
# Mode 4 - Test Mode  
# -----  
#  
# In this mode the BridgerSoc1 operates standalone,  
# and cycles random colors through all of the lighting channels.  
#  
# The BridgerSoc1 will ignore all other inputs in this mode.  
#  
#  
#  
#  
#  
#  
#  
#  
#  
#  
#  
#
```

BridgerSoc1 Operating Mode 5 : DMX-512 Slave Mode :

```
# -----  
# Mode 5 - DMX-512 Slave Mode  
# -----  
#  
# In this mode the BridgerSoc1 communicates with a DMX-512 Universe/Host using  
# the DMX-512/RS-485 connector, labeled "DMX-512".  
#  
# The DMX-512 default of 250k Baud, 8 Bits, No Parity, 1 Stop Bit is used.  
#  
# The Sixteen Lighting Channels reside from a default DMX address of 40 to 55 (decimal).  
# So the User would set the lighting channels all to maximum brightness using :  
#  
#       > <DMX> 40 255<CR>  
#       > <DMX> 41 255<CR>  
#       > <DMX> 42 255<CR>  
#       > ...  
#       > <DMX> 55 255<CR>  
#  
#
```

BridgerSoc1 Operating Mode 6 : DMX-512 Slave Mode :

```
# -----
# Mode 6 - Bike Light Controller Mode
# -----
# In this mode the BridgerSoc1 reads a number of input switches,
# located on the handlebars or the steering wheel column of a
# Light Electric Vehicle (LEV),
# via the D12/D11/D10/A3/A2/A1/A0 pin inputs.
#
# The control inputs have internal pullup resistors,
# so that a switch closed (on) should connect them to ground.
#
# Handlebar Switch mapping is as follows :
#
#      Arduino   Pin Function      PSoC
#      Pin
#
#      D13      : Arduino LED (input)  P1_3
#      D12      : Run Light/On Switch  P4_1
#      D11      : Horn Pushbutton      P2_1
#      D10      : Mode Pushbutton      P2_7
#      A3       : Right Turn Switch    Po_2
#      A2       : Left Turn Switch     Po_3
#      A1       : Brake Switch         Po_5
#      A0       : Throttle/ADC Input   Po_4
#
#      LED Light output Mapping is as follows :
#
#      J1       : Left Rear Running/Turn LED
#      J2       : Left Front Running/Turn LED
#      J3       : Front Headlight/Body LEDs
#      J4       : Ikea Color Wheel Controller Input
#      J5       : Right Front Running/Turn LED
#      J6       : Right Rear Running/Turn LED
#
# In this mode, all other inputs are ignored, the physical switches control the lights.
#
# -----
```

BridgerSoc1 Operating Mode 7 : Manual Controller Mode :

```
# -----
# Mode 7 - Manual Controller Mode
# -----
# In this mode the BridgerSoc1 is controlled by the Ikea Dioder Manual Control/Box,
# which include it's built-in color changing programs.
#
```

BridgerSoc1— Example Project :

(Coming...)

BridgerSoc1— Example Project :

(Coming...)

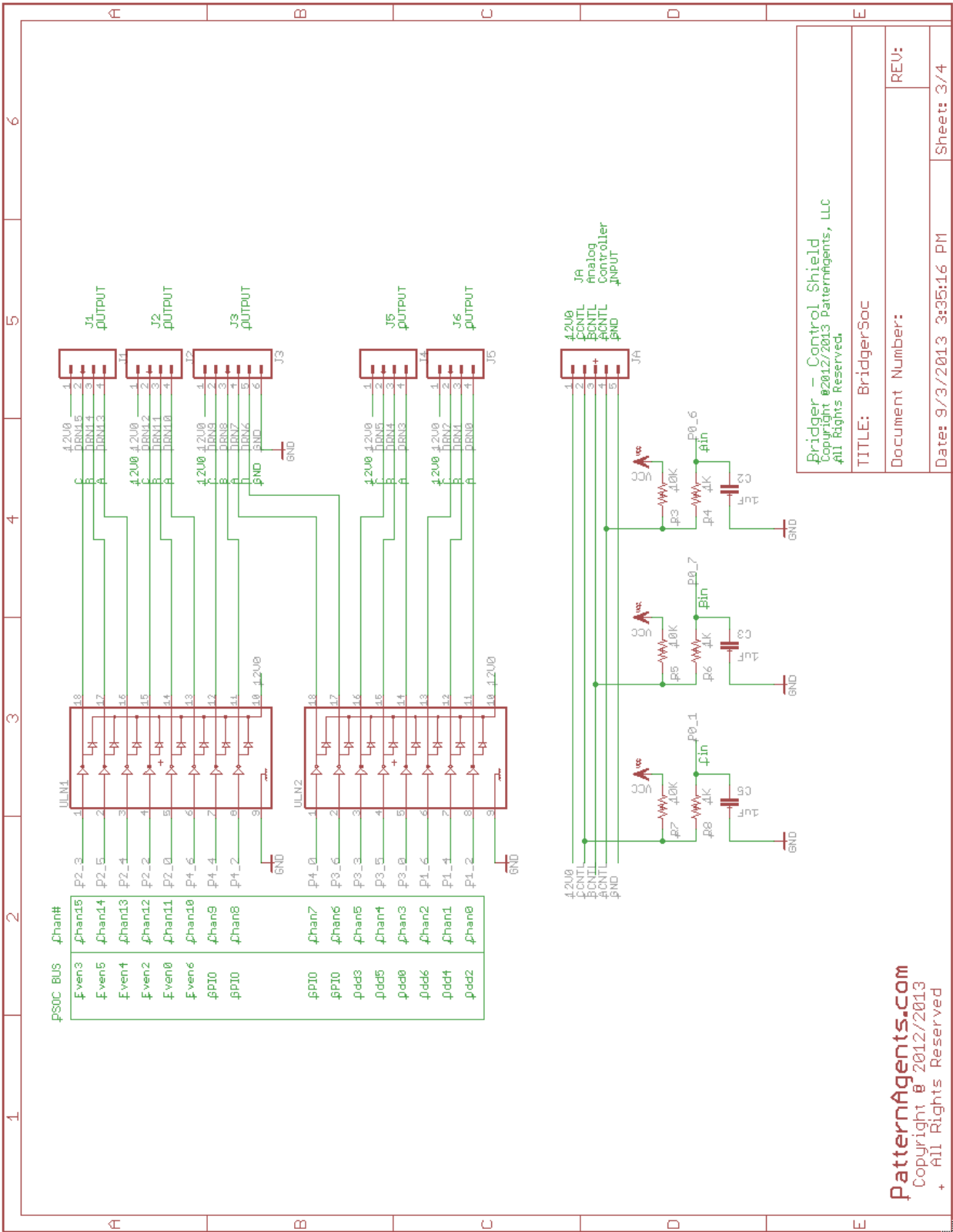
www.patternagents.com



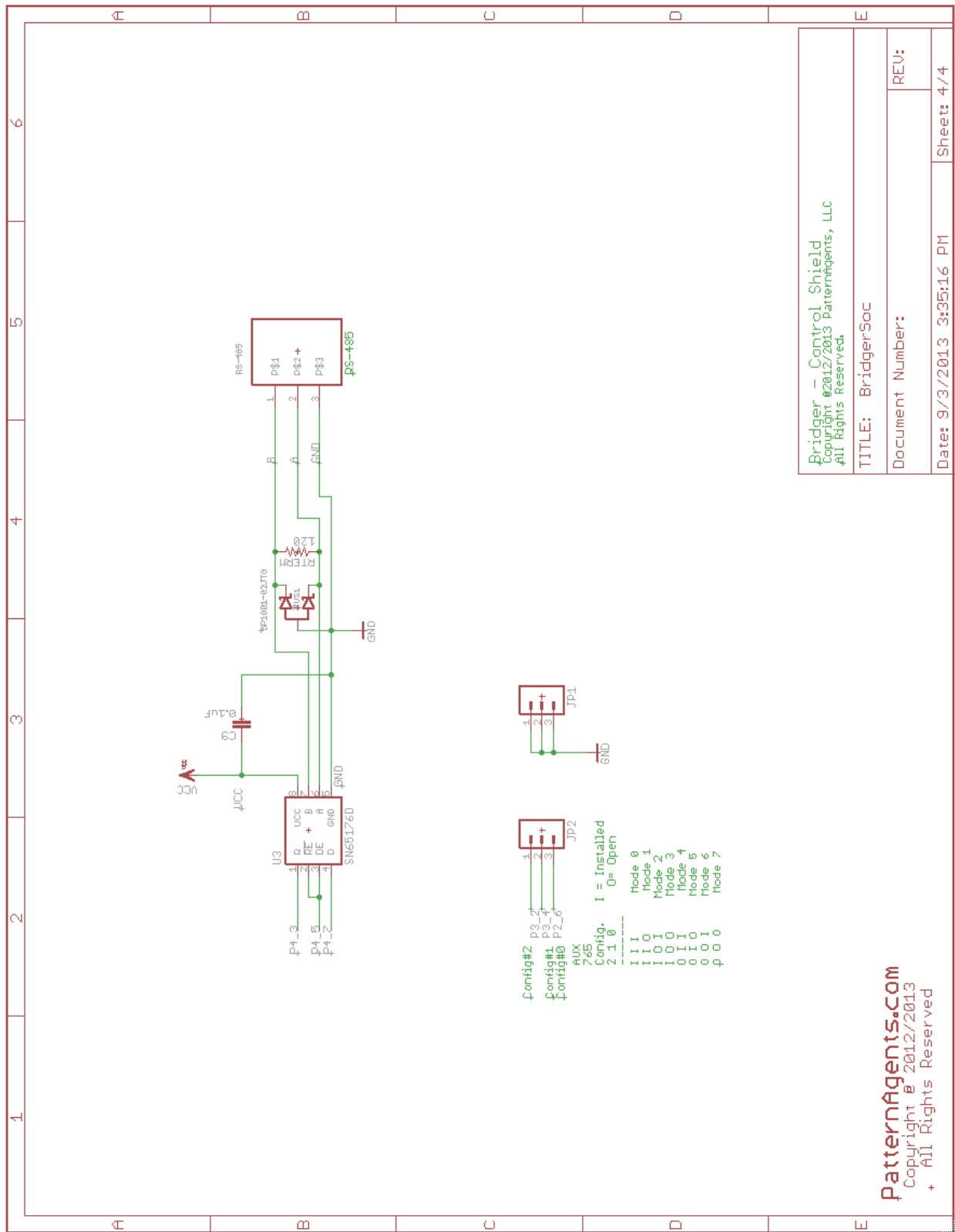
www.patternagents.com



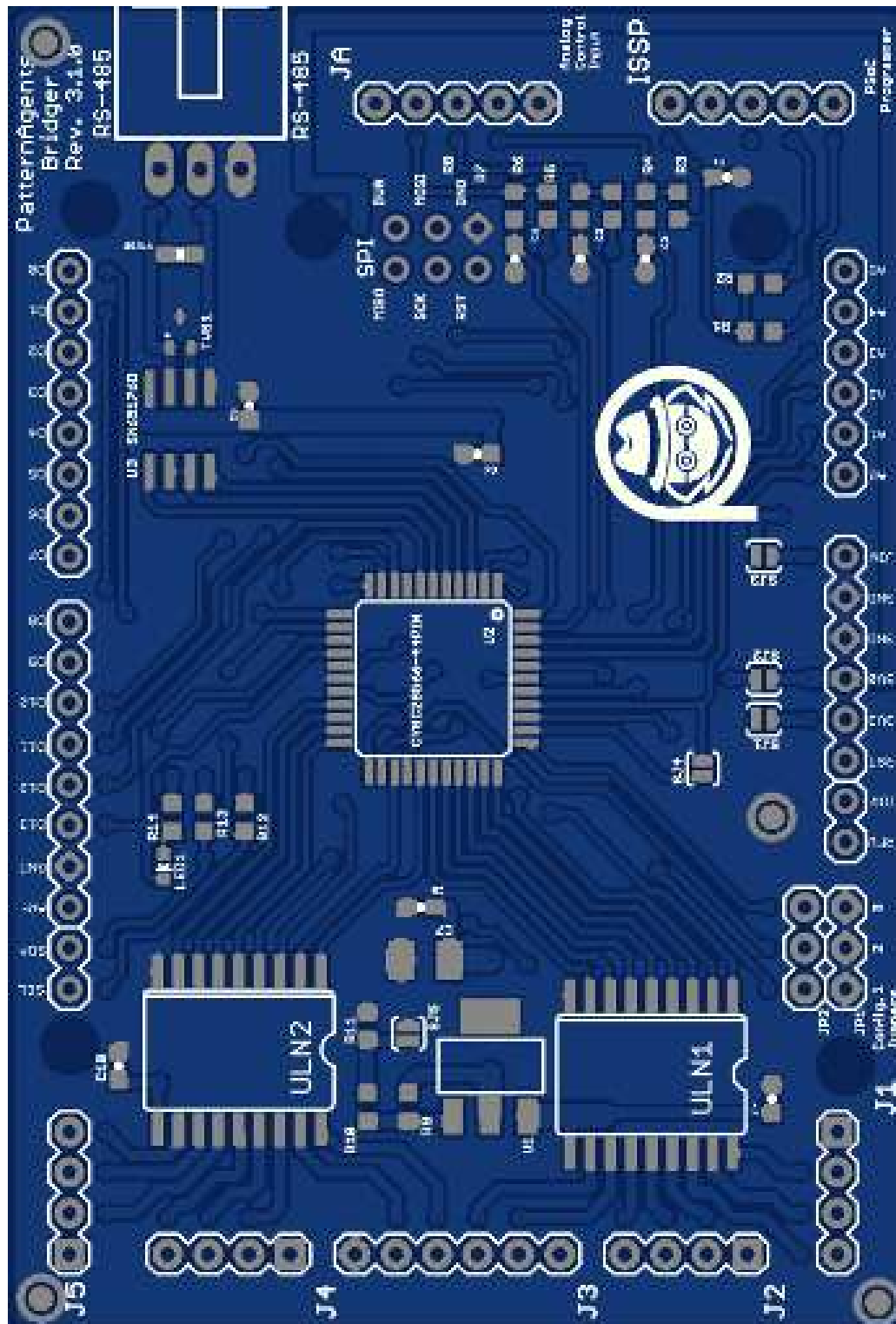
BridgerSoc1—Schematic Drawings



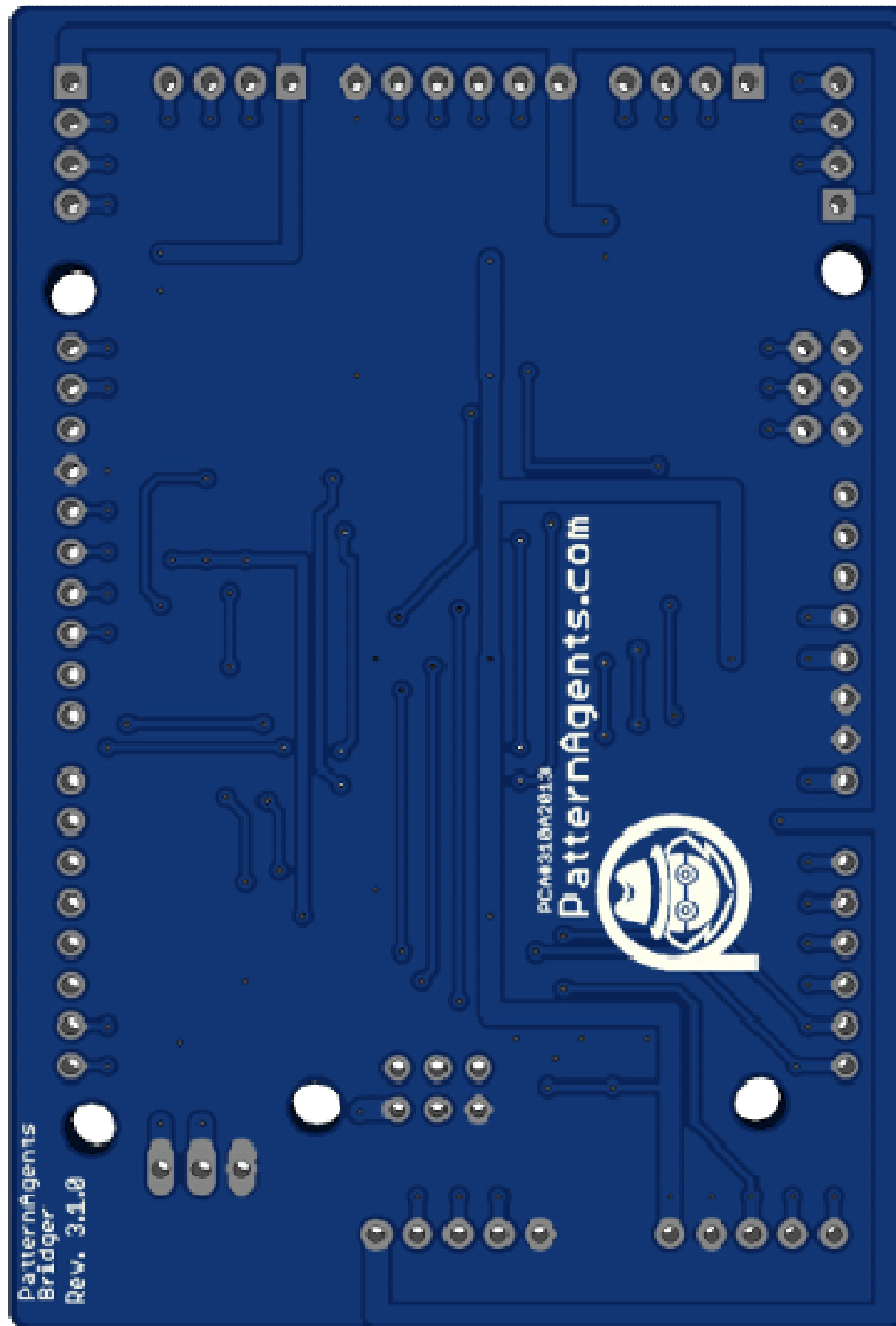
BridgerSoc1—Schematic Drawings



BridgerSoc1—PCB Top View



BridgerSoc1—PCB Bottom View



BridgerSoc1— PSoC I/O Pin Assignment :

Pin Assignments :

Pin Po.0 (0) - SJ4/Po_4 Feedback (RFU)

Pin Po.1 (1) - Green Manual Control

Pin Po.2 (2) - A3

Pin Po.3 (3) - A2

Pin Po.4 (4) - A0

Pin Po.5 (5) - A1

Pin Po.6 (6) - Red Manual Control

Pin Po.7 (7) - Blue Manual Control

Pin P1.0 (8) - ISSP/I2C-SDA/A4

Pin P1.1 (9) - ISSP/I2C-SCL/A5

Pin P1.2 (10) - LED0/SSDM0

Pin P1.3 (11) - D13/LED_D13

Pin P1.4 (12) - LED1/SSDM1

Pin P1.5 (13) - I2C-SDA

Pin P1.6 (14) - LED2/SSDM2

Pin P1.7 (15) - I2C-SCL

#

Pin P2.0 (16) - LED11/SSDM11

Pin P2.1 (17) - D11

Pin P2.2 (18) - LED12/SSDM12

Pin P2.3 (19) - LED15/SSDM15

Pin P2.4 (20) - LED13/SSDM13

Pin P2.5 (21) - LED14/SSDM14

Pin P2.6 (22) - CFG3

Pin P2.7 (23) - D10

#

Pin P3.0 (24) - LED3/SSDM3

Pin P3.1 (25) - Do/UART1-TXD

Pin P3.2 (26) - CFG1

Pin P3.3 (27) - LED5/SSDM5

Pin P3.4 (28) - CFG2

Pin P3.5 (29) - LED4/SSDM4

Pin P3.6 (30) - LED6

Pin P3.7 (31) - D1/UART1-RXD

#

Pin P4.0 (32) - LED7

Pin P4.1 (33) - D12

Pin P4.2 (34) - LED8

Pin P4.3 (35) - DMX-TXD

Pin P4.4 (36) - LED9

Pin P4.5 (37) - DMX-DIR

Pin P4.6 (38) - LED10/SSDM10

Pin P4.7 (39) - DMX-RXD

#

BridgerSoc1— PCB Pin Assignment :

J1/J2/J4/J5	Pin#	1—VIN (12 VDC) 2—Green LED Output 3— Blue LED Output 4—Red LED Output
J3	Pin#	1—VIN (12 VDC) 2—Green LED Output 3— Blue LED Output 4—Red LED Output 5—Yellow/White LED Output 6—Ground
JA	Pin#	1—VIN (12 VDC) 2—Green LED In 3— Blue LED In 4—Red LED In 5—Ground
RS-485/DMX-512	Pin#	1— B 2—A 3—Ground

I/O Configuration :

BridgerSoc1 I/O Voltage Configuration :

The BridgerSoc1 boards have the ability to configure I/O signals for different voltage levels, and can be configured for either 3.3V or 5.0V volt levels. It is important to determine if your Arduino requires 3.3V or 5.0V volt levels, and configure your BridgerSoc1 board correctly. Refer to the manual for your Arduino board for the correct voltage settings.

Solder Jumper J5 Installed—3.3V operation

Solder Jumper J5 Removed—5.0V operation (factory default setting)

NOTE: *Always make sure voltages are set correctly for your Arduino shield or you can damage it.*

External VIN/5V0/3V3 Power Configuration :

The BridgerSoc1 can either get its power from an Arduino CPU, or have its own power supplied independently. This allows the BridgerSoc1 power input to be different than the Arduino VIN. The default setting is that no power is connected to the Arduino (they are independently powered). In order to change those settings, install solder jumpers as follows :

Solder Jumper J3 Installed—Arduino and BridgerSoc1 share VIN

Solder Jumper J3 Removed—Arduino and BridgerSoc1 have separate VIN

Solder Jumper J2 Installed—Arduino and BridgerSoc1 share 5.0V power

Solder Jumper J2 Removed—Arduino and BridgerSoc1 have separate 5.0V power

Solder Jumper J1 Installed—Arduino and BridgerSoc1 share 3.3V power

Solder Jumper J1 Removed—Arduino and BridgerSoc1 have separate 3.3V power

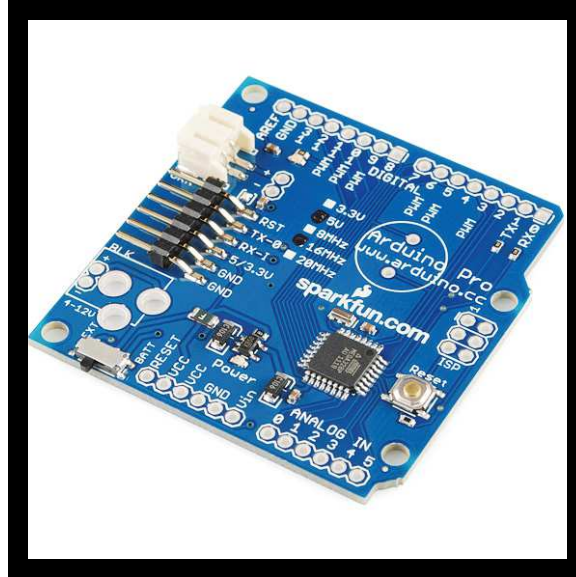
NOTE: *Install either J1 or J2 , but not both, that would short 3.3v and 5V together.*

(This page intentionally left blank)

FAQ :

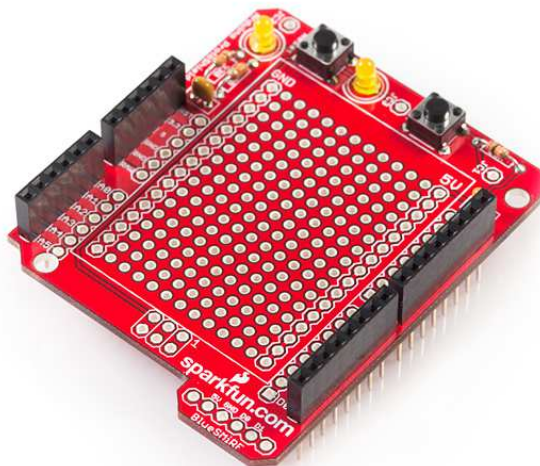
- 1) Can I use the “BridgerSoc1” with an “Arduino” CPU Board?

Yes. Any “Host” Arduino CPU board can be connected on the top/bottom side of the “BridgerSoc1” Board depending on what connector you install (i.e. female only, or stack-through connectors). This allows for multi-CPU/multi-processor/multiple architecture (i.e. big/little) configurations.



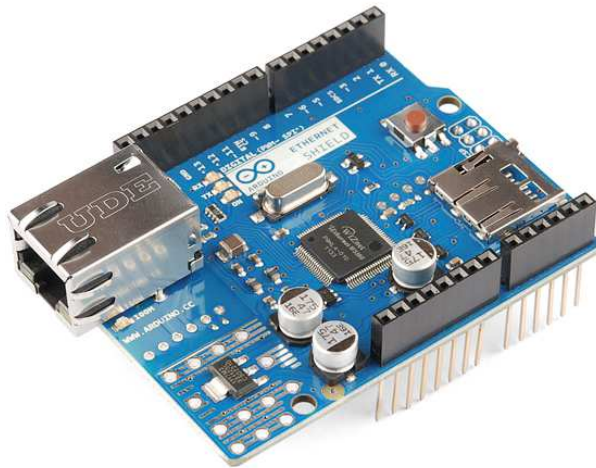
- 2) Can I use the “BridgerSoc1” with “Arduino” prototyping supplies ?

Yes. Any Arduino prototyping board can be connected on the top/bottom side of the “CPM-S to Arduino Shield Adaptor” Board depending on what connectors you install (i.e. male/female only, or stack- through).



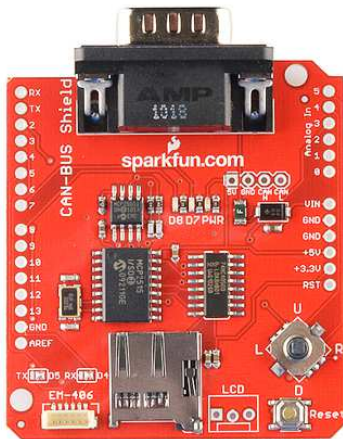
5) Can I use the “BridgerSoc1” with other “Arduino” shields ?

Yes. (Almost) any Arduino shield board can be connected on the top/bottom side of the “BridgerSoc1” Board depending on what connectors you install (i.e. male/female only, or stack-through)



Arduino Ethernet Shield :

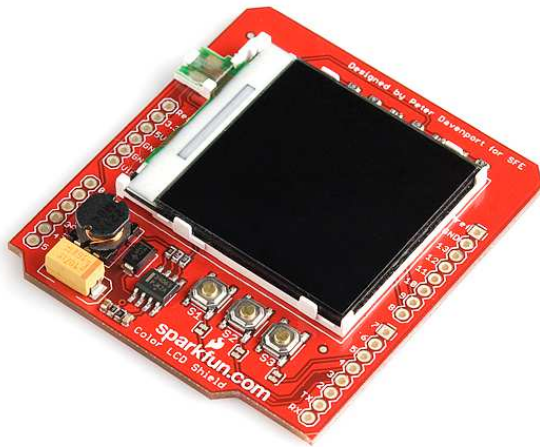
Image from :
<https://www.sparkfun.com/products/9026>



Arduino CAN-BUS Shield :

Image from :
<https://www.sparkfun.com/products/10039>

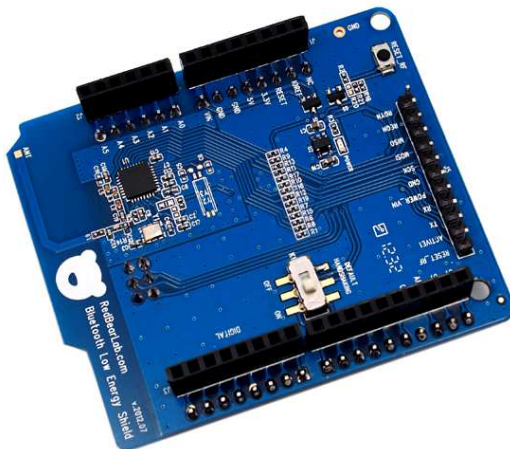




Arduino Color LCD Shield :

Image from :
<https://www.sparkfun.com/products/9363>

provides an easy method of connecting the popular [Nokia 6100 LCD](#) to your PSoC.



Arduino BlueTooth LE Shield :

Image from :
<https://www.seedstudio.com>

PatternAgents provides community and commerce for electronics designers so that they can collaborate using shared design patterns, practices, and resources; connect with others in the industry, conceptualize and create new ideas, commercialize their designs, and keep up on the latest technologies and concepts.



Copyright 2011, 2012, 2013
www.patternagents.com