Wireless Connections:

The easiest way to connect the GRB controller to a computer (tablet) is using an USB connection. This is always a good starting point. Use the shortest cable possible and keep the USB cable away from the stepper cables.
If the USB connection works fine, you could try to connect to Bluetooth or Wi-Fi.
For the Bluetooth connection I use HC0X and SPP-C modules. The Bluetooth modules can be configured using an USB to serial adapter.
For the Wi-Fi connection I use an ESP8266 module (less then 3$ on eBay). This module needs to be flashed with firmware. This can be done by using the Arduino IDE and a cheap USB to Serial adapter. There is also an Arduino with build in ESP8266 module. This works really well and this is my preferred solution. It is supplied by https://robotdyn.com/ and also available on eBay (UNO WiFi R3 ATmega328P ESP8266, 32Mb flash, USB-TTL CH340G, Micro-USB).

Connecting to Bluetooth:

The serial port on the Arduino can be used to connect a Bluetooth module. I tried the cheap, less than 3$ on eBay, HC05, HC04 and the SPP-C modules. All work well but the HC-05 module is my favorite because its fast communication and easy to configuration. To connect the Bluetooth module to the Arduino you need a level converter. The Bluetooth communication can be tested using BT-serial.

Configuring Bluetooth:

These Bluetooth modules needs to be configured (Baud rate). This can be done using the Arduino and level converters, but it is easier to use a USB to serial adapter and a terminal program like CoolTerm.

Connecting to Wi-Fi:

Wi-Fi is super fast and in my 8m x 2m workshop works very reliable. I use an ESP8266 Wi-Fi module available on eBay for less than 3$. I flashed it with a small Arduino program I made (ESP8266Station). This program supports MDNS and LLNMR so it can be accessed by the HostName set in the program. Along with this program comes a small manual for flashing the ESP8266 using the Arduino IDE. See ESP8266Station.PDF.

Configuring Wi-Fi:

The ESP8266Station firmware works in "Transparent Mode", It simply passes the data. Configuration is done by adjusting a few settings in the program before flashing. After flashing the firmware the module can be accessed by the HostName you set in the program.

USB to Serial adapter:

For this I use a CP2102 or CH340 module you can buy for less than 2$ on eBay. These modules come with a 3V3 power supply you can use to power the ESP8266 or HC-0X module for configuration (only).
You can use 4 (Dupont) patch cables for the connection, but I made a breakout board for the CP2101 module having an IDC6 (flat cable) connector that fits the breakout boards I made for the HC0X and ESP8266 modules.

Level converters:

The HC0X and ESP8266 modules accept max 3.3 volts inputs. A straight connection to the Arduino will destroy them To connect you need a level converter on the RX input of the modules. Because a mistake is done easy, I also use a level converter on the TX output of the modules. For easy connection I made breakout boards with level converters and an IDC6 (flat cable connector)
If can not make your own boards, buy a level converter they are cheap (less then 1$ on eBay) and easy to use. Buy the BOB-12009 from SparkFun or look on eBay.

Breakout boards:

For all modules (HC0X, ESP8266 and CP2101) I designed a breakout board having a IDC6 (flat cable) connector, level converters on the input and output and a 3V3 power supply. All board have the same pin out on the IDC6 connector what makes it easy to change them.

Connecting to the Arduino:

For the Arduino I made a shield for connecting the stepper drivers and added an IDC6 connector for connecting a wireless (Bluetooth or Wi-Fi) module.

Boards and Schematics:

All boards and Schematic are included as Eagle files.