One-Wire examples in C using DS18B20 temperature sensors

one_wire.c contains the 1-wire routines specific to Propeller
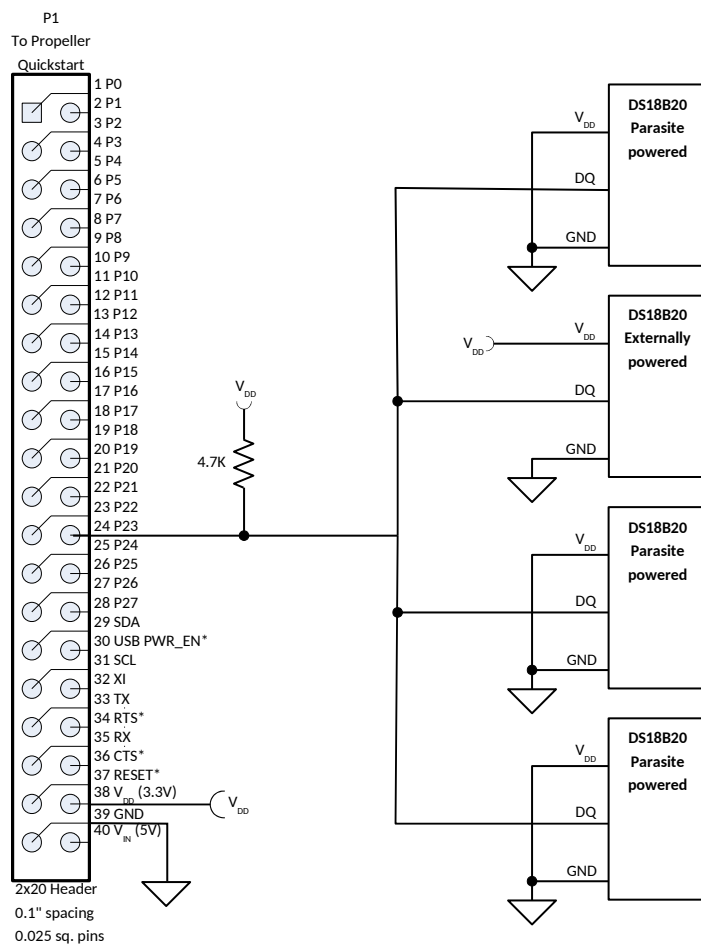
ownet.c contains 1-wire routines from Maxim that have been placed in the public domain. Some minor modifications have been made for Propeller

ownet.h contains function prototypes and general definitions from Maxim that have been placed in the public domain. Some minor modifications have been made for Propeller

ReadTemp.side and ReadTemp.c give an example of reading multiple temperature sensors with all the code running in the main cog. Note that the I/O line used for 1-wire communication is defined by the line:
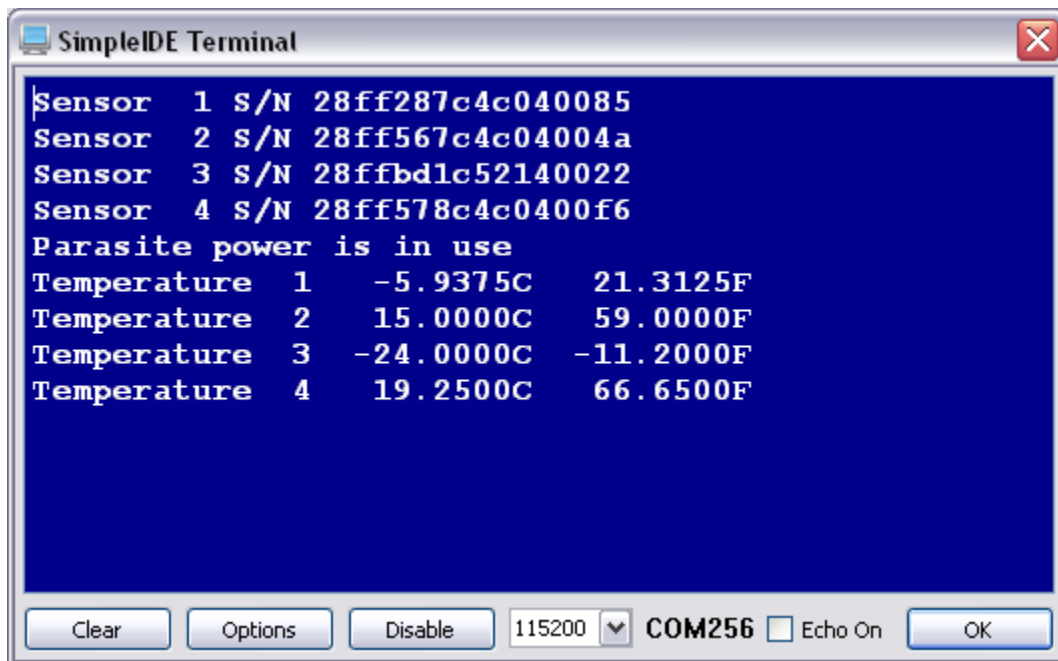int portnum=23; //the I/O number used for one wire communication

The schematic of the circuit used is shown below. A combination of externally powered and parasite powered devices has been used.

Single Bus Example

The terminal display is shown below. Freeze spray was used to lower the temperature of the devices.
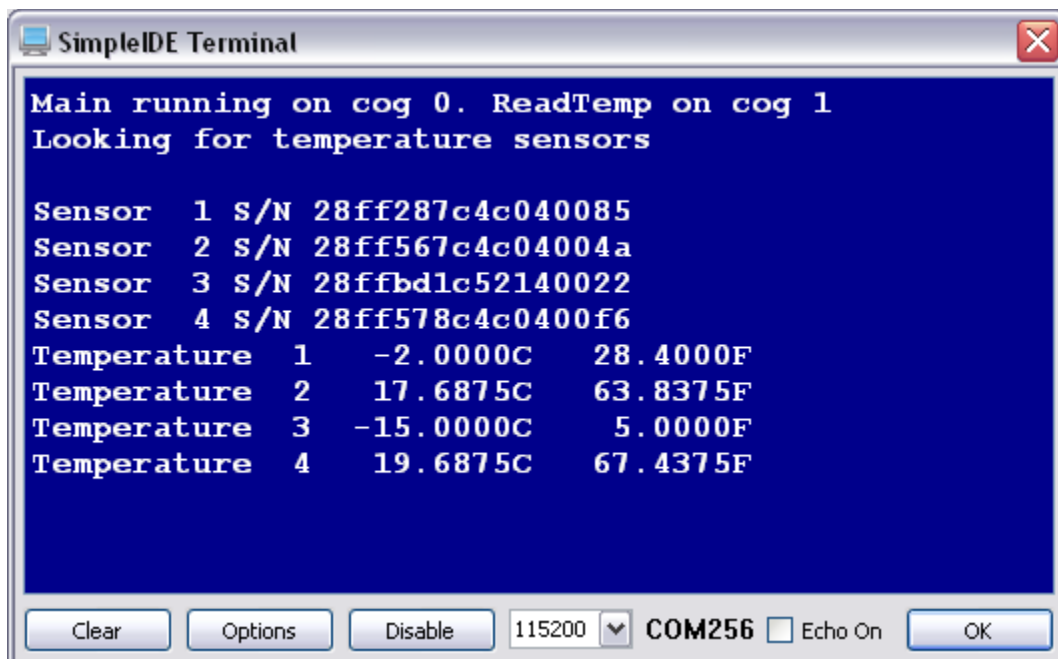
```
SimpleIDE Terminal                                              ✕

Sensor   1 S/N 28ff287c4c040085
Sensor   2 S/N 28ff567c4c04004a
Sensor   3 S/N 28ffbd1c52140022
Sensor   4 S/N 28ff578c4c0400f6
Parasite power is in use
Temperature   1    -5.9375C    21.3125F
Temperature   2    15.0000C    59.0000F
Temperature   3   -24.0000C   -11.2000F
Temperature   4    19.2500C    66.6500F




   Clear      Options      Disable    115200 ⌄ COM256 ☐ Echo On      OK
```

ReadTemp_cog.side and ReadTemp_cog.c give an example of reading multiple temperature sensors with the 1-wire communication code running in a separate cog. This would allow the code to do other tasks while waiting for a temperature measurement. Note that the I/O line used for 1-wire communication is defined by the line:
volatile int portNumber=23; //the I/O number used for one wire communication

The same schematic was used as in the first example and the terminal display is shown below.

```
SimpleIDE Terminal                                              ✕

Main running on cog 0. ReadTemp on cog 1
Looking for temperature sensors

Sensor   1 S/N 28ff287c4c040085
Sensor   2 S/N 28ff567c4c04004a
Sensor   3 S/N 28ffbd1c52140022
Sensor   4 S/N 28ff578c4c0400f6
Temperature   1    -2.0000C    28.4000F
Temperature   2    17.6875C    63.8375F
Temperature   3   -15.0000C     5.0000F
Temperature   4    19.6875C    67.4375F




   Clear      Options      Disable    115200 ⌄ COM256 ☐ Echo On      OK
```
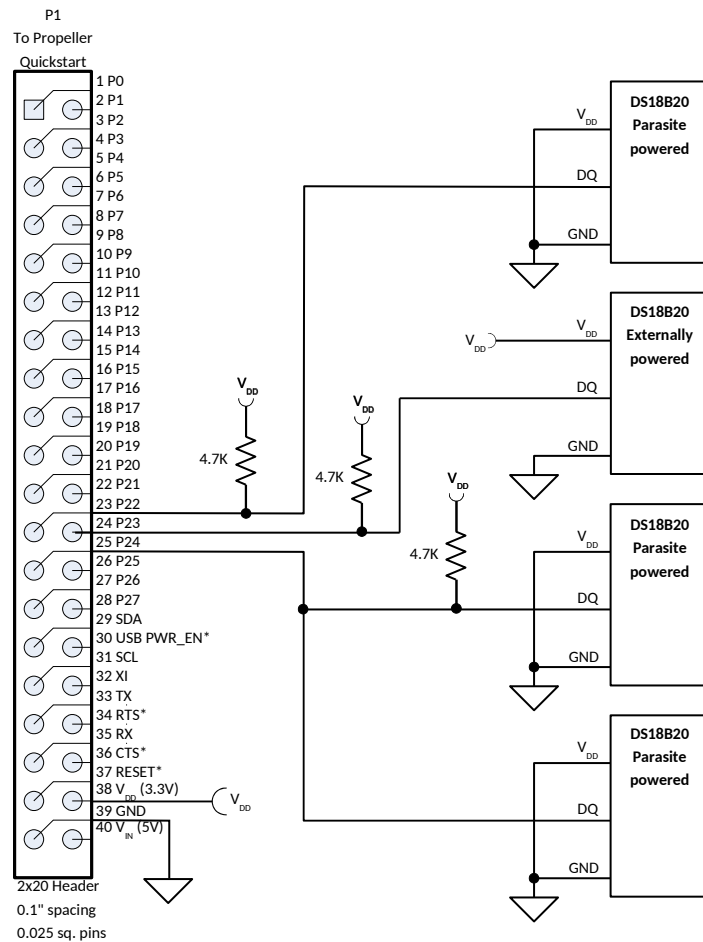
ReadTemp_multiple_cog.side and ReadTemp_multiple_cog.c give an example of using multiple 1-wire buses at the same time. Each bus is controlled by a separate cog. The number of buses (and number of I/O lines) used is defined by the line:

#define ONE_WIRE_BUSES 3 //the number of 1-wire buses we are using
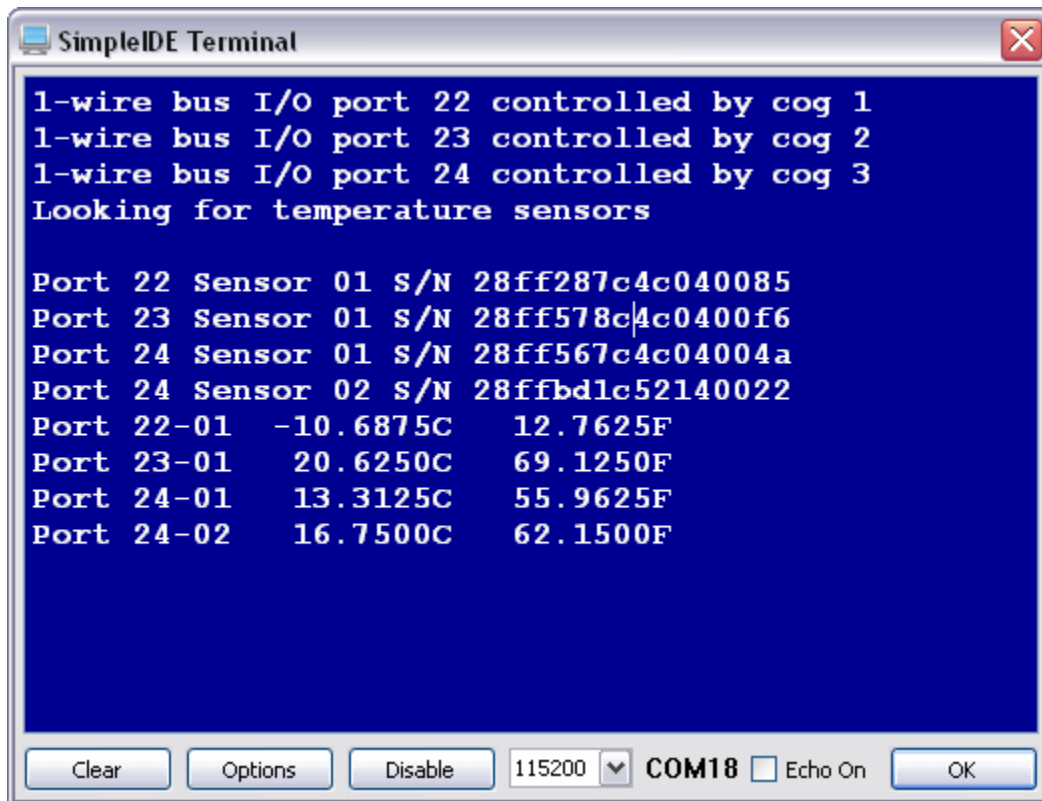
The I/O lines used for the buses is defined by the line:

   int portnum[ONE_WIRE_BUSES]={22,23,24}; //the I/O ports used for 1-wire communication

The schematic of the circuit used is shown below.

Multiple Bus Example

The terminal display is shown below.

```
SimpleIDE Terminal                                                    ✕

1-wire bus I/O port 22 controlled by cog 1
1-wire bus I/O port 23 controlled by cog 2
1-wire bus I/O port 24 controlled by cog 3
Looking for temperature sensors

Port 22 Sensor 01 S/N 28ff287c4c040085
Port 23 Sensor 01 S/N 28ff578c4c0400f6
Port 24 Sensor 01 S/N 28ff567c4c04004a
Port 24 Sensor 02 S/N 28ffbd1c52140022
Port 22-01  -10.6875C   12.7625F
Port 23-01   20.6250C   69.1250F
Port 24-01   13.3125C   55.9625F
Port 24-02   16.7500C   62.1500F




   Clear      Options     Disable   115200 ⌄ COM18 ☐ Echo On      OK
```