From: **Reagan, Chuck** Chuck.Reagan@mksinst.com 📎 🚩
Subject: RE: Ophir 300PDR and Juno
Date: October 4, 2021 at 1:47 PM
To: Warner Miller wam@fau.edu
Cc: Joe Dodd doddj2018@fau.edu

CR

Warner,

Thank you very much for the reply and I would love visiting again. Do I need to quote you anything new, or do you already own what you need?

Chuck Reagan
Southeastern Manager
Ophir-Spiricon Products

⟲ **Ophir®**

MKS Instruments, Inc.
3050 North 300 West
North Logan, Utah 84341
Cell: 813-716-8972
Main: 435-753-3729
Sales: 435-755-5421
Fax: 435-753-5231

Chuck.Reagan@mksinst.com
www.ophiropt.com/photonics
www.ophiropt.com
http://www.mksinst.com

⠿mks

NEW! ARIEL - SMALL SIZE MEETS HIGH POWER
AN INDUSTRIAL ALL-IN-ONE SENSOR FOR HIGH POWER LASER MEASUREMENT
LEARN MORE

From: Warner Miller <wam@fau.edu>
Sent: Monday, October 4, 2021 12:10 PM
To: Reagan, Chuck <Chuck.Reagan@mksinst.com>
Cc: Joe Dodd <doddj2018@fau.edu>
Subject: [EXTERNAL] Re: Ophir 300PDR and Juno

Hi Chuck,
Perfect. I will read this over and place my order. We should get you back to FAU for another visit sometime (Winter is a great time!). I am Cc'ing my student (Joe Dodd).
Thank you

Thank you.
Best Wishes,
Warner

Dr. Warner A. Miller
Professor of Physics
Department of Physics
Florida Atlantic University
wam@fau.edu
http://www.physics.fau.edu/~wam


```
  ___
 {o,o}
 I)__)
—''—''—
```


On Oct 4, 2021, at 12:52 PM, Reagan, Chuck
<Chuck.Reagan@mksinst.com> wrote:

Warner,

One of my guys answered me with the following notes.   See also the attached files.

Linux and Raspberry PI **Linux and Raspberry PI**

Linux Project:

Attached is the Linux Eclipse project you can also download the package from:
https://drive.google.com/file/d/1gcIJBQ-tlhrrva64kSCoYOP22wKmVcZz/view?usp=sharing

This package enables customers to integrate communication with Ophir USB talking meters, or USB PC interfaces such as the Juno+, via our command set – see the attached pdf document.

In the main.cpp file there is sample code of how to work with the library.

There are two ways of receiving the measurement value:

There are two ways of receiving the measurement value.
- Command mode: e.g. "SP"- Send Power command (polling mode) via usbDriver.executeCommand
- Continuous mode (streaming measurement mode): via usbDriver.startMeasuring and then retrieve the measurement by calling to usbDriver.getReading

The following Ophir instruments are presently supported in Definition.h:

```
enum SupportedProducts {
    Nova2 = 0x333,
    Vega = 0x334,
    StarLite = 0x345,
    StarBright = 0x346,
    Juno = 0x777,
};
```

Compatibility with other meters and relatively newer meters and interfaces, the Linux code needs to be modified to add the PID's of the required instrument, and the streaming commands. The following tables indicates PIDs and which existing instrument to copy streaming commands from.

Instrument	PID	Streaming commands like
USBI	0222	NovaII/Vega
NovaII	0333	NovaII/Vega
Vega	0334	NovaII/Vega
StarLite	0345	StarBright
843-R	E345	StarBright
StarBright	0346	StarBright
1919-R	E346	StarBright
PM841PE	E501	Juno
PM844PE	E502	Juno
Juno	0777	Juno
Juno_Plus	0502	Juno
Centauri	0790	StarBright
x938	E794	StarBright

To support streaming mode, using usbDriver.startMeasuring, there is the following list in UsbDriver.cpp:

```
    mSupportedProductsSet[Nova2] = CsCommand("CS 1 1 3", "CS 0");
    mSupportedProductsSet[Vega] = CsCommand("CS 1 1 3", "CS 0");
    mSupportedProductsSet[Juno] = CsCommand("CS 3", "CS 1");
    mSupportedProductsSet[StarLite] = CsCommand("CS 2", "CS 1");
```

```
        mSupportedProductsSet[StarBright] = CsCommand("CS 2", "CS 1");
```

This also needs to be extended to support new instruments. The definitions should be copied from the existing definitions according to the table above.
Here are the new lines. You can amend for just a single instrument such as the Juno+, just make sure to do the same thing in both files.

In Definition.h:

```
enum SupportedProducts {
    Nova2 = 0x333,
    Vega = 0x334,
    StarLite = 0x345,
    StarBright = 0x346,
    Juno = 0x777,
    Pm843 = 0xE345,
    Pm1919 = 0xE346,
    JunoPlus = 0x502,
    Pm841 = 0xE501,
    Pm844 = 0xE502,
    Centauri = 0x790,
    X938 = 0xE794,
};
```

In UsbDriver.cpp:

```
    mSupportedProductsSet[Nova2] = CsCommand("CS 1 1 3", "CS 0");
    mSupportedProductsSet[Vega] = CsCommand("CS 1 1 3", "CS 0");
    mSupportedProductsSet[Juno] = CsCommand("CS 3", "CS 1");
    mSupportedProductsSet[StarLite] = CsCommand("CS 2", "CS 1");
    mSupportedProductsSet[Pm843] = CsCommand("CS 2", "CS 1");
    mSupportedProductsSet[Pm1919] = CsCommand("CS 2", "CS 1");
    mSupportedProductsSet[JunoPlus] = CsCommand("CS 3", "CS 1");
    mSupportedProductsSet[Pm841] = CsCommand("CS 3", "CS 1");
    mSupportedProductsSet[Pm844] = CsCommand("CS 3", "CS 1");
    mSupportedProductsSet[Centauri] = CsCommand("CS 2", "CS 1");
    mSupportedProductsSet[X938] = CsCommand("CS 2", "CS 1");
```

Raspberry Pi Info:

The following info was provided by a customer noting changes to the Linux project to work with Raspberry PI. Several customers have successfully used these instructions:

The basic steps

The basic steps

- sudo apt install libusb-1.0-0-dev

- rm include directory

- modify everywhere #include <libusb.h> to #include <libusb-1.0/libusb>

- added -lpthread to line 7 of Default/objects.mk

- execute make in Default directory

- execute binary as superuser (sudo)

Linux Project:

Attached is the Linux Eclipse project you can also download the package from:
https://drive.google.com/file/d/1gcIJBQ-tlhrrva64kSCoYOP22wKmVcZz/view?usp=sharing

This package enables customers to integrate communication with Ophir USB talking meters, or USB PC interfaces such as the Juno+, via our command set – see the attached pdf document.

In the main.cpp file there is sample code of how to work with the library.

There are two ways of receiving the measurement value:
- Command mode: e.g. "SP"- Send Power command (polling mode) via usbDriver.executeCommand
- Continuous mode (streaming measurement mode): via usbDriver.startMeasuring and then retrieve the measurement by calling to usbDriver.getReading

The following Ophir instruments are presently supported in Definition.h:

enum SupportedProducts {
    Nova2 = 0x333,
    Vega = 0x334,
    StarLite = 0x345,
    StarBright = 0x346,
    Juno = 0x777,
};

Compatibility with other meters and relatively newer meters and interfaces, the Linux code needs to be modified to add the PID's of the required instrument, and the streaming commands. The following tables indicates PIDs and which existing instrument to copy streaming commands from.

```
Instrument         PID        Streaming commands like
USBI     0222     NovaII/Vega
NovaII   0333     NovaII/Vega
Vega     0334      NovaII/Vega
StarLite 0345     StarBright
843-R    E345      StarBright
StarBright         0346       StarBright
1919-R  E346      StarBright
PM841PE          E501      Juno
PM844PE          E502      Juno
Juno     0777     Juno
Juno_Plus         0502      Juno
Centauri          0790       StarBright
x938     E794     StarBright
```

To support streaming mode, using usbDriver.startMeasuring, there is the following list in UsbDriver.cpp:

```
mSupportedProductsSet[Nova2] = CsCommand("CS 1 1 3", "CS 0");
mSupportedProductsSet[Vega] = CsCommand("CS 1 1 3", "CS 0");
mSupportedProductsSet[Juno] = CsCommand("CS 3", "CS 1");
mSupportedProductsSet[StarLite] = CsCommand("CS 2", "CS 1");
mSupportedProductsSet[StarBright] = CsCommand("CS 2", "CS 1");
```

This also needs to be extended to support new instruments. The definitions should be copied from the existing definitions according to the table above. Here are the new lines. You can amend for just a single instrument such as the Juno+, just make sure to do the same thing in both files.

In Definition.h:

```
enum SupportedProducts {
    Nova2 = 0x333,
    Vega = 0x334,
    StarLite = 0x345,
    StarBright = 0x346,
    Juno = 0x777,
    Pm843 = 0xE345,
    Pm1919 = 0xE346,
    JunoPlus = 0x502,
    Pm841 = 0xE501,
    Pm844 = 0xE502,
    Centauri = 0x790,
    X938 = 0xE794,
};
```

In UsbDriver.cpp:

```
mSupportedProductsSet[Nova2] = CsCommand("CS 1 1 3", "CS 0");
mSupportedProductsSet[Vega] = CsCommand("CS 1 1 3", "CS 0");
mSupportedProductsSet[Juno] = CsCommand("CS 3", "CS 1");
mSupportedProductsSet[StarLite] = CsCommand("CS 2", "CS 1");
mSupportedProductsSet[Pm843] = CsCommand("CS 2", "CS 1");
mSupportedProductsSet[Pm1919] = CsCommand("CS 2", "CS 1");
mSupportedProductsSet[JunoPlus] = CsCommand("CS 3", "CS 1");
mSupportedProductsSet[Pm841] = CsCommand("CS 3", "CS 1");
mSupportedProductsSet[Pm844] = CsCommand("CS 3", "CS 1");
mSupportedProductsSet[Centauri] = CsCommand("CS 2", "CS 1");
mSupportedProductsSet[X938] = CsCommand("CS 2", "CS 1");
```

Raspberry Pi Info:

The following info was provided by a customer noting changes to the Linux project to work with Raspberry PI. Several customers have successfully used these instructions:

The basic steps

- sudo apt install libusb-1.0-0-dev

- rm include directory

- modify everywhere #include <libusb.h> to #include <libusb-1.0/libusb>

- added -lpthread to line 7 of Default/objects.mk

- execute make in Default directory

- execute binary as superuser (sudo)

Chuck Reagan
Southeastern Manager
Ophir-Spiricon Products
<image001.png>
MKS Instruments, Inc.
3050 North 300 West
North Logan, Utah 84341
Cell: 813-716-8972
Main: 435-753-3729
Sales: 435-755-5421
Fax: 435-753-5231

Chuck.Reagan@mksinst.com
www.ophiropt.com/photonics

www.ophiropt.com
http://www.mksinst.com
<image007.jpg>

<image004.png>

From: Reagan, Chuck
Sent: Monday, October 4, 2021 9:49 AM
To: wam@fau.edu
Subject: Ophir 300PDR and Juno

Warner,

Thank you for your interest in our products. I have sent your request to my power meter guru. He will have to consult with the factory engineers to see if this will work.   Give me a few days to collect the answers. It has been many years since I last visited your campus.

Name:	Warner Miller
Company:	Florida Atlantic University
State:	Florida
Country:	United States
E-Mail:	wam@fau.edu
Message:	I am looking to put a compact power meter coupling into either an SMA or FC fiber for low power (laser pointer). It has to be light weight as I will be putting this on a gimbal. I would also like to interface this to a raspberry pi 4 computer running linux. I was thinking of the 300PDR connected to the Juno. What do you recommend? Can this run under linux? Thank you.
Page:	PD300R

Chuck Reagan
Southeastern Manager
Ophir-Spiricon Products
<image001.png>
MKS Instruments, Inc.
3050 North 300 West
North Logan, Utah 84341
Cell: 813-716-8972
Main: 435-753-3729

Main: 435-753-3729
Sales: 435-755-5421
Fax: 435-753-5231

Chuck.Reagan@mksinst.com
www.ophiropt.com/photonics
www.ophiropt.com
http://www.mksinst.com
<image008.jpg>

<image004.png>

<ophiropt.drivers.usb.linux.zip><Ophir User Commands.pdf>