PFI Electronic BOX

Hardware

- **ADAM-6015** (x3) with RTD sensors (3x7)
  http://www.omega.com/pptst/RTD-830.html
  http://buy.advantech.com/Remote-I-O-Modules/Ethernet-I-O-Modules-Analog-IO-Modules/model-ADAM-6015-BE.htm?country=USA&token=636372260221909922&f=ATW&f=AUS
- **OMRON K7L-U/K7L-UD** (Liquid Leakage sensor)
  https://www.google.com.tw/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&ved=0ahUKEwjhkrHD9LfVAhVBHJQKHYxaDjQQFggIMAA&url=http%3A%2F%2Fwww.edata.omron.com.au%2FeData%2FLevel%2FF080-E1-01A.pdf&usg=AFQjCNHZAAxvnjY3vSkiMtRNaFqYF_ZOOw
- **Liquid Flowmeter**
  http://www.omega.com/pptst/FPR301_302_303_304.html
- **USB microphone with USB sound card**
  http://www.panasonic.com/in/consumer/tv-audio-video/accessories/microphones/rp-vc201.html
  http://www.galileo.com.tw/USB51A.html
- **Digital humidity sensor** (SHT75)
  https://www.sensirion.com/en/environmental-sensors/humidity-sensors/pintype-digital-humidity-sensors/
- **USB over fiber optic extension system**
  This device includes a PCIe card and a USB hub. Its hub must be power-on before we turn on the computer, check section Power-On sequence for details.
  http://www.adnaco.com/products/s3a/
- **Managed ethernet switch**
  http://www.galileo.com.tw/USB51A.html
- **Arduino ethernet board A**
  This board is used to control the power module with 13 outputs.
- **Arduino ethernet board B**
  This board is used to read the following devices:
  - Liquid leakage sensor
  - Liquid flowmeter
  - Digital humidity sensor
- **Arduino ethernet board C**
  This board is used to control the LED for fiber illumination

Ethernet Switch

The INS-8648(W) is a Managed Industrial Switch perfectly suited for industrial network applications which require managed devices that offer hassle-free fiber deployment and an ideal solution to deploy in automation systems. The switch's rugged IP30 aluminum case and hardened components withstand in operating temperatures from 0°C to 70°C (INS-8648) or wide operating temperature from -40°C to 75°C (INS-8648W).

The INS-8648(W) features with 4-slot Gigabit SFP which immune to moisture, static electricity, power surges and short circuits, plus 8 10/100/1000Base-T ports. Switch is also equipped with a variety of management functions that let you configure communication parameters as you desire and monitor the network behavior in number of different simple ways. In addition, the switch is built with dual redundant power inputs to ensure reliability and maximize network up time. Other integrated features of the switch such as Rate limitation, Port Isolation etc., optimizes your network performance and provide a secure network, offering a cost-effective solution in a small but powerful package.

IP: 10.1.120.30
MAC: 00:0b:04:13:1f:82
user: admin
pass: admin

Arduino board A

This board controls the power module. It controls the power of 13 devices. We have enabled the on-board watchdog function for all three Arduino boards in EBOX.

IP: 10.1.120.11
MAC: 90:a2:da:0f:87:05

This is current pin assignment for outputs. The default digital outputs is low for Arduino board. So for AG cameras, the power is off when we set the digital output to low. For other devices, the situation is reversed, the power is off when we set the output to high.

Bit	0(LSB)	1	2	3	4	5	6	7
Device	AG1	AG2	AG3	AG4	AG5	AG6	Leakage	ADAM x 3
Default	Off	Off	Off	Off	Off	Off	On	On
Bit	8	9	10	11	12(MSB)			
Device	Board B	Board C	USB hub1	USB hub2	Switch			
Default	On	On	On	On	On			

We can use telnet protocol to send commands: ‘Q’ for query, ‘P’ for pulse and ’S’ for set. For pulse command ‘P’, the format is ‘PXXXX’, ‘XXXX’ is the devices to cut off power for one second. This command won’t do any thing for a device in OFF state. For set command ’S', the format is ‘SXXXXYYYY’. ‘XXXX’ is the mask to specify which devices to set, ‘YYYY’ is the value to set. For query command ‘Q’, it returns current status.
There are also three commands for monitoring the network switch: ‘MONITORSET’ to set the switch IP, ‘MONITORON’ to turn on this function and ‘MONITOROFF’ to turn off this function. The default setting is OFF. If you turn on this function, once it failed to ping the network switch, it will turn off the network switch for one second in order to reboot this device.
Base on current design, there is only one telnet client can connectat the same time, other telnet clients must wait until available. Also, if a telnet client doesn’t issue any command for more than 10s, then it will be disconnected.

telnet 10.1.120.11
:*Q*
1FC0 # default value, all devices are power up except for AG cameras
:*X*
unknow
:*S1FFF1FFF* # *turn on all devices*
:*Q*
1FFF
:*S0FFF0000* # *turn off all devices except for the switch*
:*Q*
1000
:*S003F003F* # *turn on all AG cameras*
:*Q*
*10*3F

:*S0C000C00* *# turn on two USB hubs*
:*Q*
*1*C3F
:*S003F0000* *# turn off all AG cameras*
:*Q*
*1*C00
:*P1000* *# turn off network switch for a second*
:*Q*
*1*C00
:*MONITORSET10.1.120.1* *# set the switch ip, the default value is 10.1.120.30*
Set switch IP done
:*MONITORON* *# enable monitor function*
Switch monitor On
:*MONITOROFF* *# disable monitor function*
Switch monitor Off
:Bye # idle for 10 seconds
Connection closed by foreign host.

Arduino board B

This board connects to the humidity sensor, flow meter and leakage detector. We can program it to use DHCP or static IP.

IP: 10.1.120.12
MAC: 90:a2:da:0f:87:03

There are two ways to read the data:

- **Telnet protocol**

Only support ‘Q’ command for query.

telnet 10.1.120.12
:*Q*
Temperature = 25.80 C, Humidity = 68.61 %, Dewpoint = 19.59 C Flow = 0 Hz
Liquid leakage 1, disconnection 1
:*X*
unknow

- **SNMP protocol**

snmpwalk -c public -v 1 10.1.120.12 1.3.6.1.4.1.50399
SNMPv2-SMI::enterprises.50399.1.0 = STRING: "Subaru PFI telemmetry sensors"
SNMPv2-SMI::enterprises.50399.2.0 = STRING: "1.3.6.1.4.1.50399"
SNMPv2-SMI::enterprises.50399.3.0 = Timeticks: (6500) 0:01:05.00
SNMPv2-SMI::enterprises.50399.4.0 = STRING: "ChihYi Wen"
SNMPv2-SMI::enterprises.50399.5.0 = STRING: "Telemetry sensors"
SNMPv2-SMI::enterprises.50399.6.0 = STRING: “Subaru"
temperature (x100, Celsius) for SHT75
SNMPv2-SMI::enterprises.50399.7.0 = INTEGER: 2690
humidity (x100, %) for SHT75
SNMPv2-SMI::enterprises.50399.8.0 = INTEGER: 5848
dew point (x100, Celsius) for SHT75
SNMPv2-SMI::enterprises.50399.9.0 = INTEGER: 1806
flow meter (x100, Hz)
SNMPv2-SMI::enterprises.50399.10.0 = INTEGER: 0
leakage (0/1) for leakage sensor

SNMPv2-SMI::enterprises.50399.11.0 = INTEGER: 0
disconnection (0/1) for leakage sensor
SNMPv2-SMI::enterprises.50399.12.0 = INTEGER: 1
number of services
SNMPv2-SMI::enterprises.50399.13.0 = INTEGER: 12
End of MIB

snmpget -c public -v 1 10.1.120.12 1.3.6.1.4.1.50399.1.0
SNMPv2-SMI::enterprises.50399.1.0 = STRING: "Subaru PFI telemmetry sensors"

snmpgetnext -c public -v 1 10.1.120.12 1.3.6.1.4.1.50399.8.0
SNMPv2-SMI::enterprises.50399.9.0 = INTEGER: 1805

Arduino board C

This board controls the LED brightness for fiber illumination.

IP: 10.1.120.13
MAC: de:ad:be:ef:fe:ed

We use telnet to send commands to this board.

telnet 10.1.120.13

Send (a) or (b) to switch between two different LED modes
:*a* # turn on for 10.24us, turn off for 89.64us, period is 0.1ms
:*b* # turn on for 10.24ms, turn off for 89.60ms, period is 100ms

Send (q) to query current status
:*q* # query, (current, mode a, mode b)
100000,105,100,105,100000,105

Send (f) to setup mode (a) parameters
:*f010212345* # set period to 12345us, duty cycle=102/1024=10%

Send (g) to setup mode (b) parameters
:*g0306123* # set period to 123us, duty cycle=306/1024=30%

Send (q) to query current status
:*q* # query
100000,105,12345,102,123,306

Send (c) to turn off LED
:*c*

Send (z) to close telnet connection
:*z*

Adam 6015

The ADAM-6015 is a 16-bit, 7-channel RTD input module that provides programmable input ranges on all channels. It accepts various RTD inputs (PT100, PT1000, Balco 500 & Ni) and provides data to the host computer in engineering units (°C). In order to satisfy various

temperature requirements in one module, each analog channel is allowed to configure an individual range for several applications.

There are total three such modules inside EBox, so we have total 3x7=21 RTD sensors. This module supports Modbus/TCP Protocol and following is the function to read RTD sensors. A python module has been built to get the temperature readings. It doesn't support DHCP and SNMP protocols.

Function Code 03/04

The function code 03 or 04 is used to read the binary contents of input registers

Request message format for function code 03 or 04:

Command Body					
Station Address	Function Code	Start Address High Byte	Start Address Low Byte	Requested Number of Register High Byte	Requested Number of Register Low Byte

Example: Read Analog inputs #1 and #2 in addresses 40001 to 40002 as floating point value from ADAM-6017 module

01 04 00 01 00 02
Response message format for function code 03 or 04:

Example: Analog input #1 and #2 as floating point values where AI#1=100.0 and AI#2=55.32

01 04 08 42 C8 00 00 47 AE 42 5D

ADAM 6015 - 1

IP: 10.1.120.21
MAC: 00:d0:c9:f4:2a:78

RTD-1	RTD-2	RTD-3	RTD-4	RTD-5	RTD-6	RTD-7
AGC-4	AGC-3	AGC-2	AGC-1	AGC-6	AGC-5	UL Link-1

ADAM 6015 - 2

IP: 10.1.120.22
MAC: 00:d0:c9:f4:29:be

RTD-1	RTD-2	RTD-3	RTD-4	RTD-5	RTD-6	RTD-7
UL Link-2	UL Link-3	Positioner Frame	COB-1	COB-2	COB-3	COB-4

ADAM 6015 - 3

IP: 10.1.120.23
MAC: 00:d0:c9:f6:3f:60

RTD-1	RTD-2	RTD-3	RTD-4	RTD-5	RTD-6	RTD-7
COB-5	COB-6	EBox-1	EBox-2	EBox-3	Flow in	Flow out

USB microphone

This device can be used directly in Ubuntu 14.04. In the following we demonstrate how to use ALSA utility to record sound.

```
> lsusb
Bus 008 Device 004: ID 0d8c:0139 C-Media Electronics, Inc. Multimedia Headset [Gigaware by Ignition L.P.]

> cat /proc/bus/input/devices
I: Bus=0003 Vendor=0d8c Product=0139 Version=0100
N: Name="C-Media Electronics Inc.       USB PnP Sound Device"
P: Phys=usb-0000:03:00.0-2.1/input3
S: Sysfs=/devices/pci0000:00/0000:00:01.0/0000:01:00.0/0000:02:01.0/0000:03:00.0/
usb8/8-2/8-2.1/8-2.1:1.3/0003:0D8C:0139.0004/input/input8
U: Uniq=
H: Handlers=kbd event5
B: PROP=0
B: EV=13
B: KEY=1 0 0 e000000000000 0
B: MSC=10

> arecord —list-devices
**** List of CAPTURE Hardware Devices ****
card 1: Device [USB PnP Sound Device], device 0: USB Audio [USB Audio]
  Subdevices: 1/1
  Subdevice #0: subdevice #0

# record sound for 20s
> arecord -f cd -D hw:1,0 -c 1 -d 20 test.wav
```

Power-On sequence for USB devices in EBOX

1. Connect the power cable to EBOX
2. Wait for a while for the ethernet switch and board A to boot
3. Turn on AGC computer, during boot process, the USB hubs should be detected by kernel.
4. Now you can use the USB devices like AG cameras and USB microphone

Software

The python module to access those devices is under “ics_pebActor”. It includes the following functions:

- Read information from all telemetry sensors
- Power on/off function for all devices in power module
- LED brightness and on/off control

The following code is an example. Since this module is under development now, things may change in the near future.

```
>>> import peb

# check an set power status
>>> ppw = peb.PebPower()

# query ADAM6015 power status
>>> ppw.query_device(peb.POWER_ADAM)
False
# turn on ADAM6015
>>> ppw.adam_on()

# reset leakage detector
>>> ppw.leakage_pulse()

# SHT75 humidity sensor, flow meter and leakage sensor
>>> telemetry = peb.PebTelemetry()
# read current status
>>> telemetry.query()
{'Temperature': 21.08, 'LeakageDisconnection': 0, 'Humidity': 78.2, 'FlowMeter': 0.0,
'DewPoint': 17.13, 'Leakage': 0}

# ADAM5016
>>> adam = peb.PebAdam6015()
# read current status, total 3x7=21 RTD sensors
>>> adam.query()
[[19.977874418249783, 21.0582131685359, 21.59838254367895, 21.216906996261542,
21.0582131685359, 21.006332494087133, 22.166018158236056], [20.97886625467308,
21.088731212329293, 21.216906996261542, 21.216906996261542,
20.97886625467308, 21.216906996261542, 21.7265583276112], [21.10704203860533,
21.155870908674757, 21.31761654077974, 21.10704203860533, 21.216906996261542,
21.26878767071031, 22.31860837720302]]
```