In [ ]:
```
%autosave 0
```

In [ ]:
```
import subprocess
from os import listdir
from os.path import isfile, join
```

Tagfuse File System Feature Test

This file contains a set of tests that verify the features of the Tag Fuse Filesystem.

Prerequisites

- Raspberry Pi with latest software (See `01_RPi_Build.md` for details on contructing and installing a new RPi Disk Image). Make sure that the lates TagNet Basestation Software is installed, too.
- Tag with latest software (see `?` for details on constructing and installing a new Tag Image)

Verified Features

- Polling for tags
  - tested with two
- Dblk read
  - using `tagdump`
- Dlbk special files
  - sizes reflect offsets of underlying record status

To be Tested

- Panic verification
  - Panic byte file read is working
  - need to figure out if/how to erase
- GPS XYZ value
- GPS cmd
  - has been used extensively, just need to add test
- Poll count verification
- Reboot using sys/{Active,Golden,Nib,Backup,Running}

Failed Tests

- getattr
  - rtc
  - panic/{count, byte}
  - test/{sum,zeros,ones}
  - sys/{golden,nib}

Missing Features

- Image filesize and timestamp metadata
- Not all file timestamps are being set
- Poll event identifiers
- Date on GPS (designates acquisition time)

Initial State

The following variables provide user environment control.

In [116]:
```
nid='658bc8e5205c'
#nid='a0b696b4fb29'
```

In [117]:
```
TREE_BASE       = '/home/pi/tags'
TAG_ID          = TREE_BASE + '/' + nid
TAG_BASE        = TAG_ID + '/tag'
TAG_IMAGE_STORE = TAG_BASE + '/sd/0/img'
DBLK_BASE       = TAG_BASE + '/sd/0/dblk'
SYS_BASE        = TAG_BASE + '/sys'
POLL_BASE       = TAG_BASE + '/poll'
SPARSE_STORE    = '/home/pi/Desktop'
BS_IMAGE_STORE  = '/mnt/neptune/tagbin'
```

NOTES / ISSUES

- writing image when no room available gets stuck in a loop - very bad

Install Tagfuse Driver

First, get the tagfuse driver running as a daemon. TagFuse can be started separately, just comment out these invocation below.

In [ ]:
```
#!python ../tagfuse/tagfuse -b -s $SPARSE_STORE $TREE_BASE
#!tagfuse -b -s $SPARSE_STORE $TREE_BASE
```

Now look at the entire tree of information available in the tag

In [ ]:
```
!tree -a $TREE_BASE
#!tree -aJsD $TREE_BASE
```

Poll for Tags

In [207]:
```
!ls $TREE_BASE/.poll
```

```
658bc8e5205c
```

In [ ]:
```
!ls -al $TREE_BASE
```

Examine the Dblk Area

In [ ]:
```
#STOP
```

In [ ]:
```
!ls -al $DBLK_BASE
```

In [ ]:
```
!echo `stat -c '%s' $DBLK_BASE/.last_rec`
```

In [ ]:
```
!tagdump -n 10 --net -j `stat -c '%s' $DBLK_BASE/.last_sync` $DBLK_BASE/byte
```

In [ ]:
```
# !tagdump -j `stat -c '%s' $DBLK_BASE/.last_rec` $DBLK_BASE/byte
```

Write a note into the Dblk Area

In [ ]:
```
!echo 'this is a note' > $DBLK_BASE/note
```

In [ ]:
```
ls -l $DBLK_BASE/note
```

In [ ]:
```
!echo 'this is a second note' > $DBLK_BASE/note
```

In [ ]:
```
ls -l $DBLK_BASE/note
```

Extract a Panic Event

Get GPS Position

In [ ]:
```
STOP
```

Send Commands to control the GPS

Examine Pending Events on the Tag

In [201]:
```
!ls -l $POLL_BASE
```

```
total 0
-r--r--r-- 1 pi pi 0 Dec 31  1969 cnt
-r--r--r-- 1 pi pi 0 Dec 31  1969 ev
```

Show the state of software currently installed on the tag

In [259]:
```
!tree $SYS_BASE
```

```
/home/pi/tags/658bc8e5205c/tag/sys
├── active
|   └── 0.4.14
├── backup
|   └── 0.4.13
├── golden
|   └── 0.4.7
├── nib
|   └── 0.4.14
├── rtc
└── running
    └── 0.4.7

5 directories, 6 files
```

Show the currently active version

In [246]:
```
!ls -l $SYS_BASE/active
```

```
total 0
-rw-rw-r-- 1 pi pi 0 Jul 23 03:33 0.4.14
```

Show list of Images stored on Tag

In [247]:
```
!ls -l $TAG_IMAGE_STORE
```

```
total 0
-rw-rw-r-- 1 pi pi 0 Jul 23 03:33 0.4.14
```

Show list of Images stored on the Basestation

In [248]:
```
!ls -l $BS_IMAGE_STORE
```

```
total 384
-rwxrwxrwx 1 pi devgrp 96252 Jul 22 22:56 0.4.12
-rwxrwxrwx 1 pi devgrp 96252 Jul 22 22:56 0.4.13
-rwxrwxrwx 1 pi devgrp 96252 Jul 22 22:57 0.4.14
-rwxrwxrwx 1 pi devgrp 96252 Jul 23 01:35 0.4.15
```

In [249]:
```
cmd='ls ' + BS_IMAGE_STORE + '/*'
basestation_image_list= [ f for f in subprocess.check_output(cmd, shell=
True).split('\n') if f]
basestation_image_list
```

Out[249]:
```
['/mnt/neptune/tagbin/0.4.12',
 '/mnt/neptune/tagbin/0.4.13',
 '/mnt/neptune/tagbin/0.4.14',
 '/mnt/neptune/tagbin/0.4.15']
```

Exercise the Image Storage and Activation Logic

Perform the following steps:

1. Get Tag's currently active version and save it as original_vers
2. Get list of available images to load (stored on basestation)
3. Remove all images from Tag (should't be able to remove active image)
4. Load three of the available images
5. Set first of the newly loaded images as Active
6. Set second of the newly loaded images as Backup
7. Set third of the newly loaded images as Active
8. Set orignal_vers image as Active

1. Get currently active version

In [250]:
```
cmd='ls '+SYS_BASE+'/active'
original_vers=subprocess.check_output(cmd, shell=True)[:-1]
original_vers
```

Out[250]:
```
'0.4.14'
```

2. Get list of available images to load (stored on basestation)

In [251]:
```
basestation_image_list = [f for f in listdir(BS_IMAGE_STORE) if isfile(j
oin(BS_IMAGE_STORE, f))
                          and not f.startswith('.') and not f == origina
l_vers]
basestation_image_list
```

Out[251]:
```
['0.4.12', '0.4.13', '0.4.15']
```

3. Remove all images from Tag

Note: should't be able to remove active image.

In [252]:
```
!rm $TAG_IMAGE_STORE/*
!ls -l $TAG_IMAGE_STORE
```

```
total 0
-rw-rw-r-- 1 pi pi 0 Jul 23 03:34 0.4.14
```

4. Load three of the available images

Use `cp` to copy one of the software images in the IMAGE_STORE to the tag Monitor progress using pv

In [253]:
```
import os
import struct as pystruct
IMAGE_INFO_SIG = 0x33275401
IMAGE_META_OFFSET = 0x140
IMAGE_INFO_DEFAULT = [IMAGE_INFO_SIG, 0x20000, (0x140*2)+0x1c,
                      9999, 2, 0,
                      0,
                      99, 77,
                      '\00'  * 10]
IMAGE_INFO_LEN = 2
#
# Struct created for accessing image info (little indian)
# sig, image_start, imagelength, vector_chk, image_chk, im_build, im_min
or, im_major, main_tree, aux_tree, build_time, im_rev, im_model = image_
info
#
IMB_FIELDS = '<LLLHBBLBB10s'
image_info_struct = pystruct.Struct(IMB_FIELDS)
IMAGE_MIN_SIZE  =  (IMAGE_META_OFFSET + image_info_struct.size)
```

In [254]:
```
def info_check(filename):
    with open(filename, 'rb') as infile:
        if not infile:
            return (NONE, NONE, NONE)
        infile.seek(0, 2) # seek to the end
        file_size = infile.tell()
        if file_size < IMAGE_MIN_SIZE: raise RadioLoadException("input f
ile too short")
        infile.seek(0, 0)    # seek to the beginnnig
        # get image info from input file and sanity check
        infile.seek(IMAGE_META_OFFSET) # seek to location of image info
        image_info = image_info_struct.unpack(infile.read(image_info_str
uct.size))
        print("file information")
        sig, image_start, imagelength, im_build, im_minor, im_major, ima
ge_chk, \
             im_rev, im_model, pad = image_info
        pstr = "  signature: 0x{:x}, start: 0x{:x}, length: 0x{:x}, imag
e_chk: 0x{:x}"
        print(pstr.format(sig, image_start, imagelength, image_chk))
        pstr = "  version: ({}.{}.{}(0x{:x})), rev: {}, model: {}"
        print(pstr.format(im_major, im_minor, im_build, im_build, im_rev
, im_model))
        if sig != IMAGE_INFO_SIG: raise RadioLoadException("image metada
ta is invalid")
    return ((im_major, im_minor, im_build), imagelength)
```

In [255]:
```
for i in range(3 if len(basestation_image_list) >= 3
               else len(basestation_image_list)):
    version = basestation_image_list[i]
    print(version)
    info_check(os.path.join(BS_IMAGE_STORE, version))
    !dd if=$BS_IMAGE_STORE/$version of=$TAG_IMAGE_STORE/$version status=
progress
```

```
0.4.12
file information
  signature: 0x33275401, start: 0x20000, length: 0x1769c, image_chk: 0x
0
  version: (0.4.12(0xc)), rev: 1, model: 1
95744 bytes (96 kB, 94 KiB) copied, 318.286 s, 0.3 kB/s
187+1 records in
187+1 records out
96252 bytes (96 kB, 94 KiB) copied, 320.081 s, 0.3 kB/s
0.4.13
file information
  signature: 0x33275401, start: 0x20000, length: 0x1769c, image_chk: 0x
0
  version: (0.4.13(0xd)), rev: 1, model: 1
95744 bytes (96 kB, 94 KiB) copied, 320.7 s, 0.3 kB/s
187+1 records in
187+1 records out
96252 bytes (96 kB, 94 KiB) copied, 322.424 s, 0.3 kB/s
0.4.15
file information
  signature: 0x33275401, start: 0x20000, length: 0x1769c, image_chk: 0x
0
  version: (0.4.15(0xf)), rev: 1, model: 1
95744 bytes (96 kB, 94 KiB) copied, 254.396 s, 0.4 kB/s
187+1 records in
187+1 records out
96252 bytes (96 kB, 94 KiB) copied, 256.214 s, 0.4 kB/s
```

5. Set first of the newly loaded images as Active

Set the active version NOTE: REBOOTS TAG

In [260]:
```
version=basestation_image_list[0]
!ln $TAG_IMAGE_STORE/$version $SYS_BASE/active/$version
```

In [261]:
```
!ls -l $SYS_BASE/active
```

```
total 0
-rw-rw-r-- 1 pi pi 0 Dec 31  1969 0.4.12
```

Try again with same version should fail

In [198]:
```
version=basestation_image_list[2]
!ln $TAG_IMAGE_STORE/$version $SYS_BASE/active/$version
```

```
ln: failed to create hard link '/home/pi/tags/658bc8e5205c/tag/sys/acti
ve/0.4.14': File exists
```

5. Set second of the newly loaded images as Backup

In [256]:
```
!ls -l $SYS_BASE/backup
```

```
total 0
```

In [257]:
```
version=basestation_image_list[1]
!ln $TAG_IMAGE_STORE/$version $SYS_BASE/backup/$version
```

```
ln: failed to create hard link '/home/pi/tags/658bc8e5205c/tag/sys/back
up/0.4.13' => '/home/pi/tags/658bc8e5205c/tag/sd/0/img/0.4.13': No such
file or directory
```

In [258]:
```
!ls -l $SYS_BASE/backup
```

```
total 0
-rw-rw-r-- 1 pi pi 0 Jul 23 04:10 0.4.13
```

5. Set third of the newly loaded images as Active

6. Set orignal_vers image as Active

Set the backup version

In [ ]:
```
!ln $IMAGE_BASE/$original_vers $SYS_BASE/backup/$original_vers
```

In [262]:
```
!tree $SYS_BASE
```

```
/home/pi/tags/658bc8e5205c/tag/sys
├── active
|   └── 0.4.12
├── backup
|   └── 0.4.14
├── golden
|   └── 0.4.7
├── nib
|   └── 0.4.12
├── rtc
└── running
    └── 0.4.12

5 directories, 6 files
```

Reboot into the NIB

Reboot into Golden

Load new Image onto Tag

In [ ]:
```
#STOP
```

Possible software images ready for loading are in the IMAGE_STORE directory

In [ ]:
```
!ls -l $IMAGE_STORE/*
```

See what is currently in the tag's Image storage

In [ ]:
```
!ls -l $IMAGE_BASE
```

Now remove all images currently on the tag. NOTE: the active and backup images cannot be deleted (see later steps for how to find out which version is active)

In [ ]:
```
!rm $IMAGE_BASE/*
```

Check the tag's Image storage again to verify

In [ ]:
```
!ls -l $IMAGE_BASE
```

Use `cp` to copy one of the software images in the IMAGE_STORE to the tag Monitor progress using pv

In [ ]:
```
version = '0.4.1'
!dd if=$IMAGE_STORE/$version of=$IMAGE_BASE/$version status=progress
```

Check to see that image is now on the tag

In [ ]:
```
!ls -l $IMAGE_BASE
```

Load another image to the tag, this time using `dd`

In [ ]:
```
!dd if=$IMAGE_STORE/main442.bin of=$IMAGE_BASE/0.2.442
```

In [ ]:
```
!tree $IMAGE_BASE
```

In [ ]:
```
!dd if=$IMAGE_STORE/main443.bin of=$IMAGE_BASE/0.2.443
```

In [ ]:
```
!ls -l $IMAGE_BASE
```

In [ ]:
```
!dd if==%IMAGE_STORE/blink464.bin of=$IMAGE_BASE/0.2.464
```

In [ ]:
```
!ls -l $IMAGE_BASE
```

Check for memory full error condition

In [ ]:
```
!dd if=$IMAGE_STORE/main496.bin of=$IMAGE_BASE/0.2.496
```

In [ ]:
```
!ls -l $IMAGE_BASE
```

Stop the Fuse Driver

In [ ]:
```
!fusermount -u ~/tags
```

In [ ]:
```
from time import time
```

In [ ]:
```
time()
```