FBM Search

By **Tim Lowell**, Tosco Refining, Trainer, PA
Send e-mail questions to: tlowell@tosco.com

DISCLAIMER:

FBM Search is being distributed freely to all interested parties. Use at your own risk.

REQUIREMENTS

FBM Search has only been successfully run on AW51E and second-generation WP51D machines running Foxboro v.6.1 software on Solaris v.2.5.1. The `make_sub.sh` script file takes a very long time to run on WP51D first generation machines, and not all data is displayed on the `FBM_display.fdf` graphic. Your results may vary.

INSTALLATION

In order for **FBM Search** to work, all necessary files must be located on each AW51 or WP51 where you intend to run **FBM Search**. If you intend to run it on all WP51 being used as Operator Consoles, all the files described below must be on each WP51.

To install **FBM Search**, copy the compressed tar file, `fbm_search.tar.Z` to a directory that you normally use for shell scripts. The default is `/opt/tools`. If you choose another directory, you must edit the shell scripts `make_sub.sh`, `make_iolist.sh`, and `find_fbm.sh` to reflect this change. The line you must edit in each file looks like this:

```
        RUNPATH=/opt/tools
```

You must change `/opt/tools` to your chosen directory. You must also edit the line in `make_sub.sh` that reads:

```
        echo "subs:/opt/tools/FBM_display.fdf" > $RUNPATH/FBM_sublist
```

and replace `/opt/tools` with your chosen directory.

After copying the file to your chosen directory, in the VT-100 mode, uncompress it using the command:

```
        AW5101# uncompress fbm_search.tar.Z
```

Then, un-tar using the command:

```
        AW5101# tar xvf fbm_search.tar
```

This will place 5 files in your chosen directory:

1. `make_iolist.sh`: This is the shell script that should be run first to create a “master list” of all I/O Blocks in the system, along with their corresponding IOM_ID and PNT_NO. It supports AIN, AOUT, CIN, COUT, GDEV, and DPIDA blocks.
2. `SAMPLE_FBMS.fdf`: This is a FoxView graphic with a sample list of FBM’s to choose from, plus a Tag Name search field.
3. `FBM_display.fdf`: This is the target display file where the FBM loading information shows up.
4. `FBM_display`: This is a Display Manager version of `FBM_display.fdf`
5. `make_sub.sh`: This is the shell script file that creates the substitution list that is used by the target display file.

INSTRUCTIONS FOR USE

1. Run make_iolist.sh by typing in the command in the VT-100 mode:

    ```
    AW5101# make_iolist.sh &
    ```

    This script will take several minutes to complete. The first thing this script does is to create a list of all I/O blocks on the system using the getpars command. Then it uses the omgetimp command to look up the IOM_ID and PNT_NO parameters of each of these blocks. It then pastes the outputs together and sorts it alphabetically by PNT_NO. It will create several text files during the process of running, the last of which will be called cbp_all_sort.txt. When this file is created, the script is complete, and you can proceed.

2. Copy SAMPLE_FBMS.fdf to a directory that is accessible by a FoxView menu pick, or create a new Display Manager graphic that is accessible by a DM menu pick. Edit this graphic using any editing tool available to reflect the layout of your FBM cabinets. The sample graphic assumes a Rittal cabinet with six 1x8's. The fields on the sample are in the format "XXXXXX", representing a 6-character FBM letterbug. You can edit these text contents using the CTRL-T command in FoxDraw, or create a new file using Display Builder. In either case, each 6-character letterbug text field should be connected to the following "Execute Command" string:

    ```
    setglbl PICK1 @#1
    wp_applic /opt/tools/make_sub.sh $PICK1
    sleep 3
    dmcmd subslist /opt/tools/FBM_subslist
    ```

    NOTE: The "sleep 3" command gives the make_sub.sh shell script enough time to run before displaying on the target graphic. Increasing this sleep time may be necessary on your system.

    NOTE: If you choose to place the shell scripts in a directory other than /opt/tools, you must edit the two lines in these commands to reflect this change.

3. This step is optional, but can be very helpful. On the target substitution list FoxView graphic, FBM_display, there is a field for the FBM Type, <TYPE>. In order to make this field active, there are some files that need to be copied to the AW51 and concatenated. These files are in the directory /usr/fox/sp/ciofiles. In this directory is a list of all the CPs and other stations hosted by that AW51. In each file is a list of all the FBM's hosted by each station. The following command will create a file that can be used by make_sub.sh to determine the FBM type of any FBM on hosted by that AW51:

    ```
    AW5101# cat /usr/fox/sp/ciofiles/* > fbm_types_all.txt
    ```

    If you have more than one AW51 hosting CPs then you need to create a file for each AW51, and concatenate them all together into one file for the whole system, also called fbm_all_types.txt. For example, if you have an AW51 called AW5101, and another called AW5102, on AW5101, run the command:

    ```
    AW5101# cat /usr/fox/sp/ciofiles/* > fbm_types_AW5101.txt
    ```

    and on AW5102, run:

    ```
    AW5102# cat /usr/fox/sp/ciofiles/* > fbm_types_AW5102.txt
    ```

    Then copy the two files to /opt/tools on AW5101, and run:

    ```
    AW5101# cat fbm_types_AW5101.txt fbm_types_AW5102.txt > fbm_types_all.txt
    ```

And then copy `fbm_types_all.txt` to each AW and WP where you want to run **FBM Search**.

Test the substitution list file `FBM_display.fdf` or `FBM_display` by calling up `SAMPLE_FBMS.fdf` (or whatever you named your FBM layout graphic). Click on the FBM you want to display, and an hourglass should be displayed, showing that the shell script is running. After about 3 seconds, or whatever is in the "`sleep`" command, the `FBM_display` graphic should come up on the screen, with the block information filled in. On the `FBM_display` graphic there are 16 lines. Each line looks like this:

```
FBMXXX    X    COMPOUND:BLOCK       DESCRIPTION       VALUE
```

The `FBMXXX` is the FBM you are looking for. The field is configured as `<FBMx>_`, where x is 1-16

The `X` is a number, 1 through 16, representing the PNT_NO on the FBM. It is hard coded, and no connections exist.

`COMPOUND:BLOCK` is the Tag Name of each block on each point on that FBM. The field is configured as `<TAGx>__________________________`, where x is 1-16

`DESCRIPTION` is the 32-character text description for the block. The field is configured as `<DESx>_______________________________`, (32 characters) where is 1-16.

`VALUE` varies depending on the block type. For an AIN block, it is the PNT value. For an AOUT block, it is the OUT value. For a CIN block, it is the CIN value. For a COUT block, it is the COUT value. For a GDEV, it is the COUT_1 value. And for a PIDA block, it either the MEAS or OUT value, depending on the PNT_NO. The field is configured as `<VALx>_`, and it has a connection to a Real number, configured as `<VALx>`.

The `COMPOUND:BLOCK` is pickable. Picking on the `COMPOUND:BLOCK` field will call up a standard I/A Detail display for that Block. The connection configured for this is an "Execute Command" connection that looks like this:

```
dmcmd stddisp @#1
```

At the bottom of the screen in the FoxView version is a field configured as `<TYPE>`, which is filled in by the type of FBM if this feature is made active in Step 3.

4. The bottom of the screen on the FoxView `SAMPLE_FBMS.fdf` graphic also features a Tag Name search field. This allows you to find the FBM that a given tag name is on, if you know the tag name. To duplicate this on DM, use the following procedure:

    a. Create a "Normal Text" field, called "ENTER TAG HERE" or something else descriptive with enough characters for any tag name in your system. Also put some overstrike text to describe to the user what this field does.
    b. Configure the "Normal Text" field as `Contents> PICK1`
    c. Create a button with a text field of "GO" or something similar. Configure the button with an "Execute Command" statement of "`wp_applic /opt/tools/find_fbm.sh $PICK1`"

    After typing in the Tag Name in the "ENTER TAG HERE" field and pressing enter, and then clicking the "GO" button, the `find_fbm.sh` script will run locating the FBM where the tag is located, and will in turn call the `make_sub.sh` script with the correct FBM.

5. You may also optionally choose to put the `make_iolist.sh` shell script in `crontab` to keep the master I/O list updated on a regular basis.

TROUBLESHOOTING

Almost all problems are caused by files either not existing, or not being in the correct directory. Check the path name in all the scripts, and make sure it corresponds with the path name you chose during installation.

Make sure all the necessary files are in the AW51 or WP51 you are trying to run **FBM Search** from. These include `make_sub.sh`, `cbp_all_sort.txt`, `FBM_display.fdf` (`FBM_display` on DM) and `find_fbm.sh` and `fbm_types_all.txt` if you are using the FBM Type feature.

Also, if you choose not to make the FBM Type feature active, it would be best to comment out the lines in `make_sub.sh` that refer to this. They look like this:

```
FBM_TYPE=`cat $RUNPATH/fbm_types_all.txt|grep $1|awk '{print $2}'|cut -c3,4`
echo "<TYPE>="$FBM_TYPE>>$RUNPATH/FBM_subslist
```

Comment them out by placing a “`#`” in front of each line.

The “`sleep`” command is also very important. **FBM Search** has only been tested successfully on AW51E’s and second generation WP51D’s. Testing on first-generation WP51D’s revealed that the `make_sub.sh` script ran too slowly to fill in all necessary data on the `FBM_display` graphic. You can increase the sleep time to get better results.

Rev: 12/05/2000