Use for   Foxboro DM (sorry, not Foxview)
Application: `makeaatab`
Function: Synchronize `<lbug>AAtab` file in `/usr/fox/alarms` with existing `<lbug>AApan` file

Over the past few years, synchronizing the alarm buttons with the displays using the supplied utility 'FoxPanels' has been a hit or miss proposition. This utility either put too many things in the `<lbug>AAtab` file or not enough, or sometimes nothing at all. It may be fixed now in version 6.1.1 but this has worked so well we still use it and I haven't bothered to check.

The function of this utility is to provide a method of actually scanning the graphics as designated in the `<lbug>AApan` file and create a complete, accurate, functional and annotated `<lbug>AAtab` file.

It does so by using the results of the '`/usr/fox/wp/bin/tools/d_edit –dump`' command to produce workfiles from which it can gather alarm connections on each display and associate them with the proper LED on the Annunciator Keyboard. The association is currently made to the `ALMSTA` and `UNACK` parameters only. Should you need other associations, the source is provided for your pleasure.

Since this program utilizes the existing `<lbug>AApan` file as its input, it must be run on the AW or WP which will be the final destination of the new `<lbug>AAtab` file. The current `<lbug>AApan` file may be 'modified' in some situations to help get a cleaner output. More about this later.

The output consists of two files:

- The `<lbug>AAtab.new` file is the presumably finished product and is ready to replace the current `<lbug>AAtab` file on the current station. This file indicates, with comments, which connections have been assigned from which displays. A feature I've long wanted from Foxboro. It is not uncommon for more than one display to contain the same variable. This application handles this, normally unresolvable issue, by placing ALL of the connections it finds to ALL displays listed in the `<lbug>AApan` file. The FIRST instance of this type of variable is put in the file as an actual destination to light the LED. Subsequent instances are inserted in the file but are commented out (#). Should there be more than two instances, the third instance gets double comments (##), and so on.

- The `<lbug>AAtab.log` file is a record of duplicated variables. It indicates the display name, panel, and buttons that are duplicated. This file can then be used to determine which of the multiple displays indicated are actually to have the LED turn on. Simply comment out the one you ***don't*** want and remove the comments from the one you do.

Disclaimer:
The resultant output of this application must be reviewed prior to replacing the existing `<lbug>AAtab` file. While it attempts to ignore entries in the `<lbug>AApan` file that do not point to displays, it may still be possible to get unpredictable results. It is recommended that you make a backup copy of both the current `<lbug>AApan` file and the `<lbug>AAtab` file prior to running this application.

Helpful Hint:
If you know you have displays that you do not want to have entered into the `<lbug>AAtab` file, make a copy of the current `<lbug>AApan` file, then modify the current `<lbug>AApan` file to exclude those display paths. Don't forget to replace the original `<lbug>AApan` file.

To run this program:
1. Copy the '`makeaatab`' program into `/usr/fox/alarms` on the station you want to make a new `<lbug>AAtab` file.
2. Be sure that all of the displays noted in the `<lbug>AApan` file are resident on the selected station. This will NOT work for displays located on an AP/AW and accessed remotely.
3. Enter '`makeaatab <lbug>`' where `<lbug>` is the letterbug name of the station (i.e. W23A02).
4. Review the `<lbug>AAtab.log` file.
5. Modify the `<lbug>AAtab.new` file if necessary.
6. Replace the `<lbug>AAtab` file with the `<lbug>AAtab.new` file.

The Bad News:
Don't get in a hurry. Much sifting and sorting is going on in the background and it takes quite a while to complete. At least there is a lot of screen output during the operation to give you something to watch while you sip your coffee and wait.

Author: Ronald B. "Bo" Stear <stear@ppg.com>
          PPG Industries, Inc.
          Lake Charles, LA, US