# Sysfs is deprecated

|  |  |
| --- | --- |
| **Say this** | **Show this** |
| sysfs is being deprecated.  Which is to say, if you’ve been using /sys/class/gpio/ with bash, it may stop working with an upcoming update of Raspbian. | Code:  # set up pushbuttons  btnVolUp=15 # BCM numbering scheme  echo $btnVolUp > /sys/class/gpio/export  echo "in" > "/sys/class/gpio/gpio${btnVolUp}/direction" |
| Instead, you’ll start using something called libgpiod. Libgpiod comes with some really useful tools that will make your development easier | gpioget $chipName $btnVolUp |
| Instead - let’s take a look at libgpiod and how it improves on sysfs | <https://manpages.debian.org/experimental/gpiod/> |
| **Insert opener here** |  |
| For now, you’ll need to install libgpiod. This will change when it’s installed as part of Raspbian. Fortunately, this is easy | Sudo apt-get update  sudo apt-get install gpiod libgpiod-dev libgpiod-doc |
| Now you have a selection of tools:  All of them work from a Bash shell, so I’ll demonstrate them, then finally I’ll show how to use them in a Bash Script | gpiodetect, gpioinfo, gpioget, gpioset, gpiofind and gpiomon |
|  | #!/bin/bash  # demonstration of libgiod  # install libgpiod with  Sudo apt-get update  sudo apt-get install gpiod libgpiod-dev libgpiod-doc  # now you have new commands...  ##############  gpioget 0 14 # returns value of gpio14  gpioget gpiochip0 14 # same thing  gpioget pinctrl-bcm2835 14 # also same thing  ##############  gpiodetect # returns available gpio banks  ##############  gpioinfo # returns state and names of pins in a chipset  ##############  gpiofind TXD1 # returns chipname and pin number for named pin  ##############  gpioget $(gpiofind TXD1) # combine gpiofind with gpioget . COOL!  ##############  gpioset -m wait gpiochip0 24=1 # do this in terminal #2  gpioinfo # do this in terminal #1, look for GPIO24  # terminal 2, hit enter  ##############  gpiomon $(gpiofind TXD1) # shows status of input pin. Watch for float |
| So here’s a sample of how to use it in a bash script | #!/bin/bash  # Don't forget to install libgpiod  # sudo apt-get install gpiod libgpiod-dev libgpiod-doc  # documentation: https://manpages.debian.org/experimental/gpiod/  ##################  # config  btnAlpha=15 # BCM numbering scheme  btnGamma=14 # BCM numbering scheme  # either of these work  chipName=$(awk -F"[][]" '/gpiochip0/ { print $2 }' <( gpiodetect ))  chipName=gpiochip0  ###################  # working loop  while :  do  if [[ $(gpioget $chipName $btnAlpha) == "0" ]]  then  echo "Alpha"  fi  if [[ $(gpioget $chipName $btnGamma) == "0" ]]  then  echo "Gamma"  fi    done |
| **Insert closer here** |  |

<https://manpages.debian.org/experimental/gpiod/>

<https://microhobby.com.br/blog/2020/02/02/new-linux-kernel-5-5-new-interfaces-in-gpiolib/>

<https://git.kernel.org/pub/scm/libs/libgpiod/libgpiod.git/about/>

<https://www.beyondlogic.org/an-introduction-to-chardev-gpio-and-libgpiod-on-the-raspberry-pi/>

|  |  |
| --- | --- |
| gpioget | gpioget gpiochip0 14  Possible to do…  Watch -n 1 gpioget gpiochip0 14  But there’s something better coming up  <https://manpages.debian.org/experimental/gpiod/gpioget.1.en.html>  Returns value of one or more input as 0 or 1  Note: the -l option sets the input to active-low. If the original state was active-high, -l will reverse the value |
| gpiodetect | <https://manpages.debian.org/experimental/gpiod/gpiodetect.1.en.html>  Lists the gpiochips (0 and 1)  Returns the chip name |
| Gpioset | <https://manpages.debian.org/experimental/gpiod/gpioset.1.en.html>  Set GPIO line values of a GPIO chip and maintain the state until the process exits  gpioset --mode=time --sec=1 pinctrl-bcm2835 25=1 |
|  | Gpioset -m wait gpiochip0 24=1  Then use gpioinfo (in a different terminal) to find the status of that pin  “Enter” in original terminal to terminate the gpioset |
| gpiofind | <https://manpages.debian.org/experimental/gpiod/gpiofind.1.en.html>  Gpiofind TXD1  Returns chipName and GPIO pin  Usable with gpioget or gpioset |
| gpiomon | <https://manpages.debian.org/experimental/gpiod/gpiomon.1.en.html>  Gpiomon $(gpiofind TXD1)  Run this, then press button GPIO14.  Result…  Event: FALLING/RISING EDGE offset: 14 timestamp… |
| gpioinfo | <https://manpages.debian.org/experimental/gpiod/gpioinfo.1.en.html>  Lists all gpio pins, in use by, input/output , active-high or low |