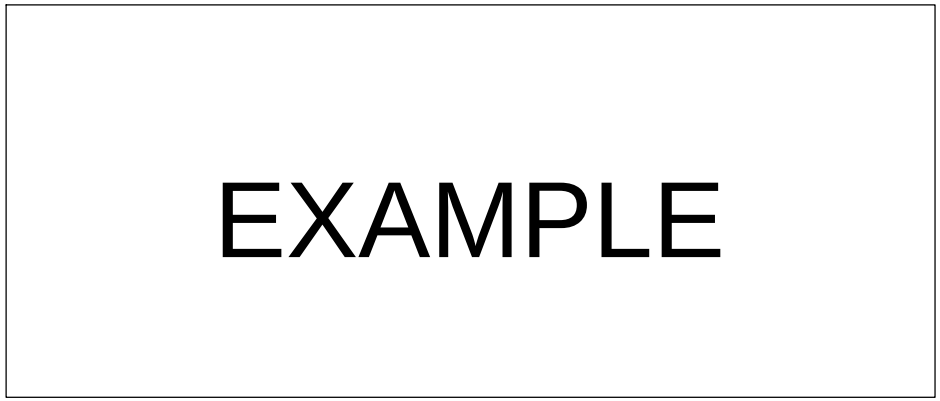

EXAMPLE

GitHub
GIT CHEAT SHEET

V 1.1.1

Git is the open source distributed version control system that facilitates GitHub activities on your laptop or desktop. This cheat sheet summarizes commonly used Git command line instructions for quick reference.

INSTALL GIT

GitHub provides desktop clients that include a graphical user interface for the most common repository actions and an automatically updating command line edition of Git for advanced scenarios.

GitHub for Windows
https://windows.github.com

GitHub for Mac
https://mac.github.com

Git distributions for Linux and POSIX systems are available on the official Git SCM web site.

Git for All Platforms
http://git-scm.com

CONFIGURE TOOLING

Configure user information for all local repositories

`$ git config --global user.name "[name]"`
Sets the name you want attached to your commit transactions
`$ git config --global user.email "[email address]"`
Sets the email you want attached to your commit transactions
`$ git config --global color.ui auto`
Enables helpful colorization of command line output

CREATE REPOSITORIES

Start a new repository or obtain one from an existing URL

`$ git init [project-name]`
Creates a new local repository with the specified name
`$ git clone [url]`
Downloads a project and its entire version history

MAKE CHANGES

Review edits and craft a commit transaction

`$ git status`
Lists all new or modified files to be committed
`$ git diff`
Shows file differences not yet staged
`$ git add [file]`
Snapshots the file in preparation for versioning
`$ git diff --staged`
Shows file differences between staging and the last file version
`$ git reset [file]`
Unstages the file, but preserve its contents
`$ git commit -m "[descriptive message]"`
Records file snapshots permanently in version history

GROUP CHANGES

Name a series of commits and combine completed efforts

`$ git branch`
Lists all local branches in the current repository
`$ git branch [branch-name]`
Creates a new branch
`$ git checkout [branch-name]`
Switches to the specified branch and updates the working directory
`$ git merge [branch]`
Combines the specified branch's history into the current branch
`$ git branch -d [branch-name]`
Deletes the specified branch

GIT CHEAT SHEET

REFACTOR FILENAMES
Relocate and remove versioned files

```
$ git rm [file]
```
Deletes the file from the working directory and stages the deletion

```
$ git rm --cached [file]
```
Removes the file from version control but preserves the file locally

```
$ git mv [file-original] [file-renamed]
```
Changes the file name and prepares it for commit

SUPPRESS TRACKING
Exclude temporary files and paths

```
*.log
build/
temp-*
```
A text file named `.gitignore` suppresses accidental versioning of files and paths matching the specified patterns

```
$ git ls-files --other --ignored --exclude-standard
```
Lists all ignored files in this project

SAVE FRAGMENTS
Shelve and restore incomplete changes

```
$ git stash
```
Temporarily stores all modified tracked files

```
$ git stash pop
```
Restores the most recently stashed files

```
$ git stash list
```
Lists all stashed changesets

```
$ git stash drop
```
Discards the most recently stashed changeset

REVIEW HISTORY
Browse and inspect the evolution of project files

```
$ git log
```
Lists version history for the current branch

```
$ git log --follow [file]
```
Lists version history for a file, including renames

```
$ git diff [first-branch]...[second-branch]
```
Shows content differences between two branches

```
$ git show [commit]
```
Outputs metadata and content changes of the specified commit

REDO COMMITS
Erase mistakes and craft replacement history

```
$ git reset [commit]
```
Undoes all commits after `[commit]`, preserving changes locally

```
$ git reset --hard [commit]
```
Discards all history and changes back to the specified commit

SYNCHRONIZE CHANGES
Register a repository bookmark and exchange version history

```
$ git fetch [bookmark]
```
Downloads all history from the repository bookmark

```
$ git merge [bookmark]/[branch]
```
Combines bookmark's branch into current local branch

```
$ git push [alias] [branch]
```
Uploads all local branch commits to GitHub

```
$ git pull
```
Downloads bookmark history and incorporates changes

OMB Number: 4040-0004
Expiration Date: 01/31/2009

Application for Federal Assistance SF-424 Version 02

9. Type of Applicant 1: Select Applicant Type:

Type of Applicant 2: Select Applicant Type:

Type of Applicant 3: Select Applicant Type:

* Other (specify):

*** 10. Name of Federal Agency:**

11. Catalog of Federal Domestic Assistance Number:

CFDA Title:

*** 12. Funding Opportunity Number:**

* Title:

13. Competition Identification Number:

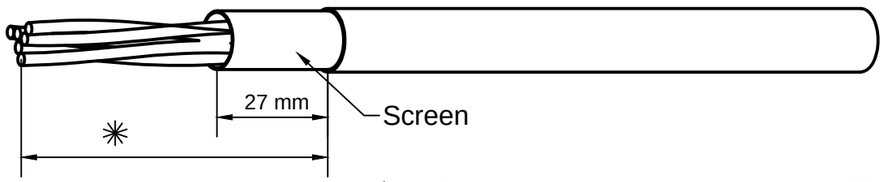
Title:

14. Areas Affected by Project (Cities, Counties, States, etc.):

*** 15. Descriptive Title of Applicant's Project:**

Attach supporting documents as specified in agency instructions.

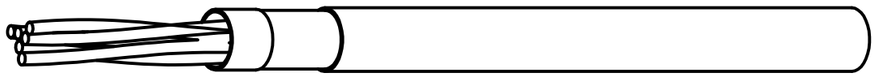
Add Attachments | Delete Attachments | View Attachments



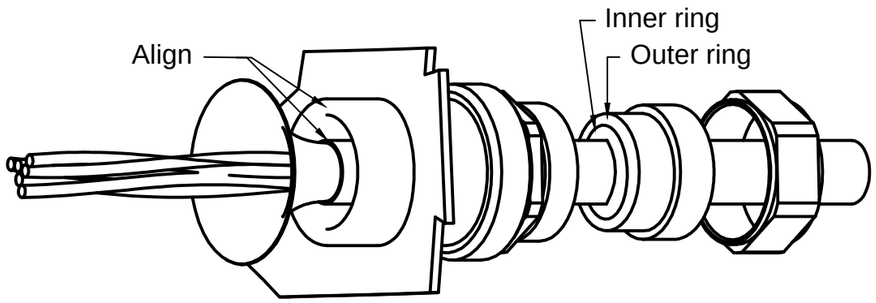
✳ Strip length on radar cable: 500mm

Strip length on temp.sensors cable: 800mm

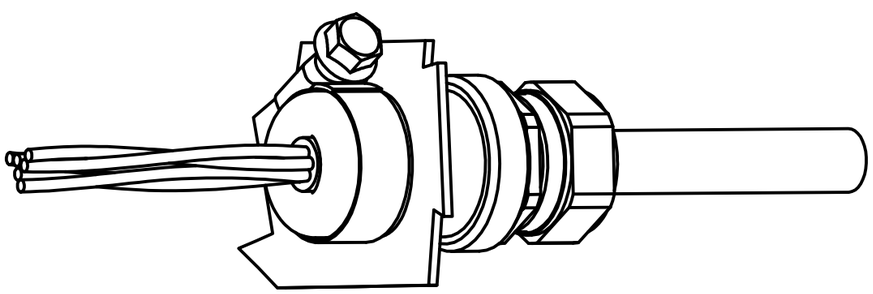
1. Strip the cable



2. Tape the screens before inserting the cable



3. If cable outside diameter is too big, remove the inner ring of the gland gasket. Insert the cable until the outer mantle is aligned with the inner edge of the gland. Remove the tape.

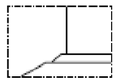


4. Tighten the gland nut so the gasket is tight, see note.
   Fold the cable screens back distributed evenly over the gland inner surface, and fasten the screen clamp.

Note:

Tighten the domed cap nut so far down to the insert in order to build a swelling of the rubber between the domed cap nut and the cable.

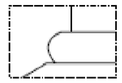
A



Optical inspection

A = correct
B = incorrect

B



REMARKS:

Gland sizes / torque

Thread (M)	Range (mm)	Torque (NM)
M20x1.5	8.0 - 15.0	8
M25x1.5	12.5 - 20.5	11
M32x1.5	17.0 - 25.5	15
M40x1.5	24.0 - 33.0	20
M50x1.5	33.0 - 42.0	30
M63x1.5	40.0 - 52.0	44

The above torques for the compression nuts are maximum values in the case of the largest cable in a normal environment. In order to ensure correct mounting under conditions differing from this, mounting should be terminated if the sealing insert forms a bead projecting slightly above the cap nut, even if the torque shown in the table has not yet been reached.