

# Test de interconexión de red Zybo

Jesús Rodríguez Heras

5 de abril de 2019

## **Resumen**

En este documento se desarrolla la explicación y uso de un pequeño script que se usará para el test de interconexión entre los dispositivos de la red Zybo.

# Índice

<b>1. Desarrollo</b>	<b>3</b>
<b>2. Script</b>	<b>3</b>

# 1. Desarrollo

Una vez se han conectado todos los dispositivos al switch como se indicó en el tutorial de Creación de una infraestructura de red de placas Zybo, y, estando encendidos todos los dispositivos, necesitamos comprobar que existe conexión entre todos ellos.

Para ello, se ha desarrollado un script en el que se pueden definir las tarjetas Zybo que se vayan a usar, con su alias y su dirección IP de red (previamente establecida).

El funcionamiento del script es el siguiente:

- Si los cuatro paquetes de ping de una tarjeta tienen éxito, indicará que dicha tarjeta está conectada.
- Si en el último de ellos obtenemos el mensaje de que se han perdido el 100% de los paquetes enviados, significará que la tarjeta está desconectada y habrá que revisar la conexión.

Además, el script cuenta con la opción `-v` para que podamos ver el lanzamiento de los paquetes de ping en tiempo real.

## 2. Script

```
1  #!/bin/bash
2
3  #Script para 4 placas Zybo
4
5  direcciones=(192.168.1.2 192.168.1.3 192.168.1.4 192.168.1.5)
6  tarjetas=(zybo1 zybo2 zybo3 zybo4)
7  a=0
8  for d in "${direcciones[@]}"
9  do
10   if [[ $1 = "-v" ]]; then
11     echo Comprobando la conexion con ${tarjetas[a]}.
12     ping -c 4 $d
13   else
14     echo Comprobando la conexion con ${tarjetas[a]}.
15     string=$(ping -c 4 $d)
16     if [[ $string == *100%\ packet\ loss* ]]; then
17       echo Tarjeta ${tarjetas[a]} desconectada. Compruebe la conexion.$'\n'
18     else
19       echo Tarjeta ${tarjetas[a]} conectada.$'\n'
20     fi
21   fi
22   let a=a+1
23 done
```