



Pushing Performance



People | Power | Partnership

HARTING HAIC MICA Python 3.4 Container v1.1.0_r

1. Edition 2017, 07/17
Doc No 20 95 100 0003 / 99.01

© HARTING IT Software Development, Espelkamp

All rights reserved, including those of the translation.

No part of this manual may be reproduced in any form (print, photocopy, microfilm or any other process), processed, duplicated or distributed by means of electronic systems without the written permission of HARTING Electric GmbH & Co. KG, Espelkamp.

Subject to alterations without notice.

This application note explains how to use the Python 3.4 container v1.1.0_r.



Contents

Contents	3
1 Python 3.4 Container Basics	4
1.1 Overview	4
1.2 Installation	4
2 Use of the Python 3.4 Container	5
3 Version History	6

1 Python 3.4 Container Basics

1.1 Overview

The Python 3.4 Base Container is provided with Python 3.4 including pip based on Busybox v1.22.1.

Web Icons App.png and Header.png can be found under /META/web_icons. In case new Web Container Icons are desired one can simply replace these files.

The Python 3.4 Base Container is suited for productive purposes.

1.2 Installation

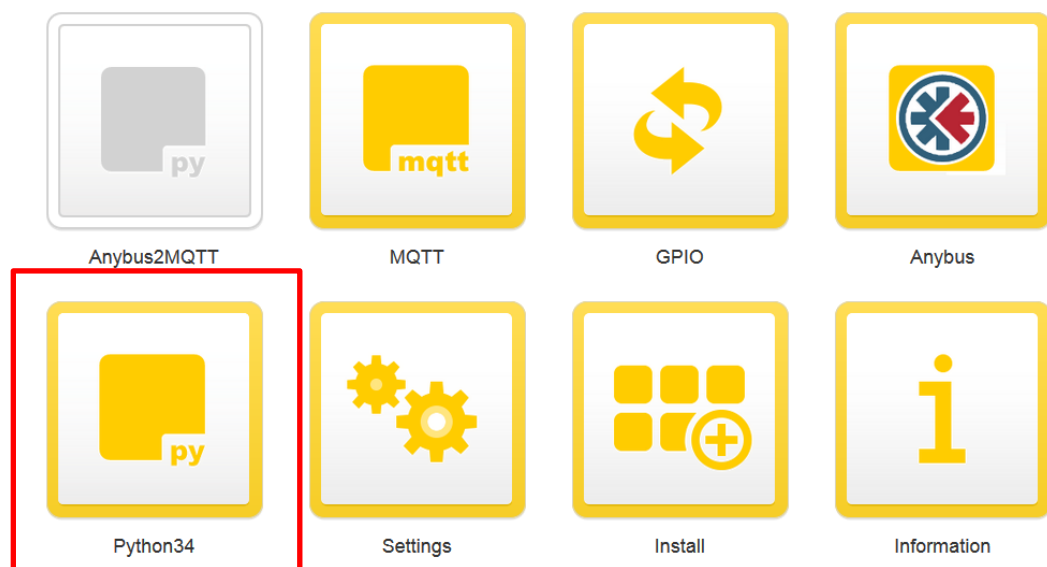


Figure 1: MICA Homescreen Including a Highlighted GPIO Container

The installation and configuration routine of the Python 3.4 Container follows the standard routine as provided by the IIC MICA and can be found in the “MICA Programming Guide”.

Connect your MICA to a webbrowser of your choice

Using Windows, you can access your MICA via <https://devicename>

If your host is running an mdns application, you can access your MICA also via <https://devicename.local>.

You have to accept a Security Certificate. After Container installation, you have to configure network settings via Firmware -> Settings -> Network in order to access the Container via SSH.

2 Use of the Python 3.4 Container

The Python container can be accessed over ssh using e.g. Putty. The user name and password is root\rroot. You can then use Python 3.4 and pip.

When you start the container WebGUI, you see the following:

```
Waiting for web server...
If no web server is configured for this container, please refer to the documentation on how to configure a web server.
```

A httpd webserver can be configured on the Python 2.7 container and the WebGUI can be accessed over https.

Access the busybox container via ssh. Under etc/init.d you find a start script for httpd. Create a softlink under rc.d by executing

```
cd /etc/rc.d
ln -s ../init.d/<script> <link_name>
```

The link_name should start with a number describing the place in line of start scripts where it is executed.
Add an index.html file to the directory /var/www and modify it to create a container WebGUI.

Restart the busybox container.

The WebGUI can be accessed from a client via https.

3 Version History

Version 1.1.0_r

- Idconfig added

Version 1.0.0