variablesAPI

Net and Light

2023 - 2024

Résumé

Cette documentation décrit un fichier Python qui initialise un dictionnaire stockant des métadonnées sur les variables PLC (Automate Programmable Industriel). Chaque entrée comprend le type de données de la variable et son adresse logique dans la mémoire du PLC, nécessaire pour les tâches de manipulation des données.

Table des matières

1 Introduction

Ce document présente un script Python utilisé pour définir un dictionnaire appelé `variables`, qui stocke les variables du PLC et leurs propriétés telles que le type de données et l'adresse logique. Ce dictionnaire est essentiel pour les systèmes d'automatisation nécessitant la lecture et l'écriture des données dans un PLC.

2 Contenu du Script Python

2.1 Structure Générale

Le script initialise un dictionnaire avec des variables de type entier et booléen, chacune étant associée à une adresse logique spécifique dans le PLC.

2.2 Détail du Script

Le script Python est présenté ci-dessous avec des explications pour chaque partie importante.

```
# -*- coding: utf-8 -*-
""" Define a dictionary called 'variables' to store PLC
   variables and their properties """
variables = {
    # Integer type variables with their logical addresses in
       the PLC
    "Mw_API_CVEntree": {"Data Type": "Int", "Logical Address"
       : "%MW506"},
    "Mw_API_CVSortieStandard": {"Data Type": "Int", "Logical
       Address": "%MW508"},
    "Mw_API_CVSortieSecours": {"Data Type": "Int", "Logical
       Address": "%MW510"},
    "Mw_master_nb_billes_entree": {"Data Type": "Int", "
       Logical Address": "%MW500"},
    "Mw_master_nb_billes_sortie_normal": {"Data Type": "Int",
        "Logical Address": "%MW502"},
    "Mw_master_nb_billes_sortie_secours": {"Data Type": "Int"
       , "Logical Address": "%MW504"},

    # Boolean type variables for control, indication, and
       alarms with their specific bit addresses in the PLC
    "Mx_master_start": {"Data Type": "Bool", "Logical Address
       ": "%M320.1"},
    "Mx_master_pause": {"Data Type": "Bool", "Logical Address
       ": "%M320.0"},
    "Mx_master_multi_possible": {"Data Type": "Bool", "
       Logical Address": "%M320.2"},
    "Mx_master_multi+de3": {"Data Type": "Bool", "Logical
       Address": "%326.2"},
```

```
17      "Mx_master_demande_mode_multi": {"Data Type": "Bool", "
          Logical Address": "%M301.6"},
18      "Mx_master_alarme_urgence": {"Data Type": "Bool", "
          Logical Address": "%M325.0"},
19      "Mx_master_quittance_obstruation": {"Data Type": "Bool",
          "Logical Address": "%M325.3"},
20      "Mx_master_quittance_urgence": {"Data Type": "Bool", "
          Logical Address": "%M325.1"},
21      "Mx_master_connecte": {"Data Type": "Bool", "Logical
          Address": "%M310.0"},
22      "Mx_master_alarme_obstruation": {"Data Type": "Bool", "
          Logical Address": "%M325.3"},
23      "Mx_master_quittance_distribution": {"Data Type": "Bool",
           "Logical Address": "%M325.5"},
24      "Mx_master_quittance_sortie_secours": {"Data Type": "Bool
          ", "Logical Address": "%M325.7"},
25      "Mx_master_alarme_surveillance_bille": {"Data Type": "
          Bool", "Logical Address": "%M326.0"},
26      "Mx_master_quittance_surveillance_bille": {"Data Type": "
          Bool", "Logical Address": "%M326.1"},
27
28      # Boolean type variables for specific alerts and warnings
            for control areas with their specific bit addresses
29      "Mx_API_C1_attention": {"Data Type": "Bool", "Logical
          Address": "%M135.4"},
30      "Mx_API_C1_alerte": {"Data Type": "Bool", "Logical
          Address": "%M135.0"},
31      "Mx_API_C2_attention": {"Data Type": "Bool", "Logical
          Address": "%M135.5"},
32      "Mx_API_C2_alerte": {"Data Type": "Bool", "Logical
          Address": "%M135.1"},
33      "Mx_API_C3_attention": {"Data Type": "Bool", "Logical
          Address": "%M135.6"},
34      "Mx_API_C3_alerte": {"Data Type": "Bool", "Logical
          Address": "%M135.2"},
35      "Mx_API_C4_attention": {"Data Type": "Bool", "Logical
          Address": "%M135.7"},
36      "Mx_API_C4_alerte": {"Data Type": "Bool", "Logical
          Address": "%M135.3"},
37  }
38
39  """
40  This script initializes a dictionary storing metadata about
        PLC variables which can be used
41  for reading from and writing to the PLC in automation systems
        . Each entry includes the variable's data type and
42  its logical address in the memory of the PLC, necessary for
        data manipulation tasks.
```

```
"""
```

Listing 1 – Contenu du fichier Python

2.2.1 Définition du Dictionnaire `variables`

Le dictionnaire `variables` est utilisé pour stocker les métadonnées des variables PLC. Chaque entrée du dictionnaire contient le type de données de la variable (`Int` pour entier et `Bool` pour booléen) ainsi que son adresse logique dans la mémoire du PLC.

```
variables = {
    "Mw_API_CVEntree": {"Data Type": "Int", "Logical Address": "%MW506"},
    "Mw_API_CVSortieStandard": {"Data Type": "Int", "Logical Address": "%MW508"},
    "Mw_API_CVSortieSecours": {"Data Type": "Int", "Logical Address": "%MW510"},
    "Mw_master_nb_billes_entree": {"Data Type": "Int", "Logical Address": "%MW500"},
    "Mw_master_nb_billes_sortie_normal": {"Data Type": "Int", "Logical Address": "%MW502"},
    "Mw_master_nb_billes_sortie_secours": {"Data Type": "Int", "Logical Address": "%MW504"},
    "Mx_master_start": {"Data Type": "Bool", "Logical Address": "%M320.1"},
    "Mx_master_pause": {"Data Type": "Bool", "Logical Address": "%M320.0"},
    "Mx_master_multi_possible": {"Data Type": "Bool", "Logical Address": "%M320.2"},
    "Mx_master_multi+de3": {"Data Type": "Bool", "Logical Address": "%326.2"},
    "Mx_master_demande_mode_multi": {"Data Type": "Bool", "Logical Address": "%M301.6"},
    "Mx_master_alarme_urgence": {"Data Type": "Bool", "Logical Address": "%M325.0"},
    "Mx_master_quittance_obstruation": {"Data Type": "Bool", "Logical Address": "%M325.3"},
    "Mx_master_quittance_urgence": {"Data Type": "Bool", "Logical Address": "%M325.1"},
    "Mx_master_connecte": {"Data Type": "Bool", "Logical Address": "%M310.0"},
    "Mx_master_alarme_obstruation": {"Data Type": "Bool", "Logical Address": "%M325.3"},
    "Mx_master_quittance_distribution": {"Data Type": "Bool", "Logical Address": "%M325.5"},
    "Mx_master_quittance_sortie_secours": {"Data Type": "Bool", "Logical Address": "%M325.7"},
    "Mx_master_alarme_surveillance_bille": {"Data Type": "Bool", "Logical Address": "%M326.0"},
```

```
    "Mx_master_quittance_surveillance_bille": {"Data Type": "
       Bool", "Logical Address": "%M326.1"},
    "Mx_API_C1_attention": {"Data Type": "Bool", "Logical
       Address": "%M135.4"},
    "Mx_API_C1_alerte": {"Data Type": "Bool", "Logical
       Address": "%M135.0"},
    "Mx_API_C2_attention": {"Data Type": "Bool", "Logical
       Address": "%M135.5"},
    "Mx_API_C2_alerte": {"Data Type": "Bool", "Logical
       Address": "%M135.1"},
    "Mx_API_C3_attention": {"Data Type": "Bool", "Logical
       Address": "%M135.6"},
    "Mx_API_C3_alerte": {"Data Type": "Bool", "Logical
       Address": "%M135.2"},
    "Mx_API_C4_attention": {"Data Type": "Bool", "Logical
       Address": "%M135.7"},
    "Mx_API_C4_alerte": {"Data Type": "Bool", "Logical
       Address": "%M135.3"},
}
```

2.2.2 Commentaires

Les commentaires dans le script expliquent le but de chaque section et fournissent des informations sur l'utilisation des variables dans les systèmes d'automatisation PLC.

```
"""
This script initializes a dictionary storing metadata about
   PLC variables which can be used
for reading from and writing to the PLC in automation systems
   . Each entry includes the variable's data type and
its logical address in the memory of the PLC, necessary for
   data manipulation tasks.
"""
```

3 Conclusion

Ce script Python est essentiel pour les systèmes d'automatisation nécessitant l'interaction avec un PLC. En définissant un dictionnaire de variables avec leurs types de données et adresses logiques, il permet de simplifier et de structurer les tâches de manipulation des données dans les systèmes d'automatisation.

Références

[1] Python Documentation, `https://docs.python.org/3/`.

[2] PLC Programming Documentation, `https://www.plcdev.com/book/export/html/216`.