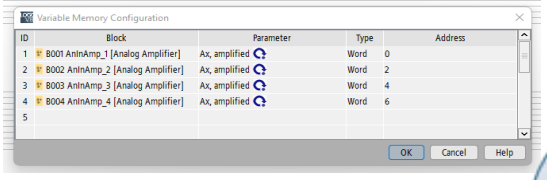
LOGO! 8 snap7 address mapping **0BA8**

Digital Input	VM Address in LOGO! 8		Python sample to read from VM
I1	V1024.0	Bit	print("DI1: " + str(plc.read("V1024.0")))
I2	V1024.1	Bit	print("DI2: " + str(plc.read("V1024.1")))
I3	V1024.2	Bit	print("DI3: " + str(plc.read("V1024.2")))
I4	V1024.3	Bit	print("DI4: " + str(plc.read("V1024.3")))
I5	V1024.4	Bit	…
I6	V1024.5	Bit	…
I7	V1024.6	Bit	…
I8	V1024.7	Bit	…
I9	V1025.0	Bit	print("DI9: " + str(plc.read("V1025.0")))
I10	V1025.1	Bit	print("DI10: " + str(plc.read("V1025.1")))
I11	V1025.2	Bit	…
I12	V1025.3	Bit	…
I13	V1025.4	Bit	…
I14	V1025.5	Bit	…
I15	V1025.6	Bit	…
I16	V1025.7	Bit	…
I17	V1026.0	Bit	print("DI17: " + str(plc.read("V1026.0")))
I18	V1026.1	Bit	print("DI18: " + str(plc.read("V1026.1")))
I19	V1026.2	Bit	…
I20	V1026.3	Bit	…
I21	V1026.4	Bit	…
I22	V1026.5	Bit	…
I23	V1026.6	Bit	…
I24	V1026.7	Bit	…

In Logo Soft Comfort können 24 digitale Eingänge konfiguriert werden, obwohl der von Siemens angegebene Addressbereich wesentlich weiter reicht, von 1024 bis 1031 (8 Bytes).
Quelle: Siemens Industrie Support

V	Bereichskennung	Bsp.:	Zugriff auf Bit: V1026.3
B	Byte		Zugriff auf Byte: VB1026
W	Word		Zugriff auf Word: VW1032
D	Double Word		Zugriff auf Double Word: VD1072



Für die Konfiguration und den Zugriff auf den **"variablen Speicher" / "Variable Memory Configuration"**, siehe bitte die letzte Seite dieses Dokuments

LOGO! 8 snap7 address mapping **0BA8**

Analog Input	VM Address in LOGO! 8		Python sample to read from VM
AI1	VW1032	2 Byte	print("AI1: " + str(plc.read("VW1032")))
AI2	VW1034	2 Byte	print("AI2: " + str(plc.read("VW1034")))
AI3	VW1036	2 Byte	print("AI3: " + str(plc.read("VW1036")))
AI4	VW1038	2 Byte	…
AI5	VW1040	2 Byte	…
AI6	VW1042	2 Byte	…
AI7	VW1044	2 Byte	…
AI8	VW1046	2 Byte	…

In Logo Soft Comfort können 8 analoge Eingänge konfiguriert werden, obwohl der von Siemens angegebene Addressbereich wesentlich weiter reicht, von 1032 bis 1063 (32 Bytes).

Digital Output	VM Address in LOGO! 8		Python sample to read from VM
DO1	V1064.0	Bit	plc.write("V1064.0", True)
DO2	V1064.1	Bit	plc.write("V1064.1", False)
DO3	V1064.2	Bit	…
DO4	V1064.3	Bit	…
DO5	V1064.4	Bit	…
DO6	V1064.5	Bit	…
DO7	V1064.6	Bit	…
DO8	V1064.7	Bit	…
DO9	V1065.0	Bit	print("DO9: " + str(plc.read("V1065.0")))
DO10	V1065.1	Bit	print("DO10: " + str(plc.read("V1065.1")))
DO11	V1065.2	Bit	…
DO12	V1065.3	Bit	…
DO13	V1065.4	Bit	…
DO14	V1065.5	Bit	…
DO15	V1065.6	Bit	…
DO16	V1065.7	Bit	…
DO17	V1066.0	Bit	…
DO18	V1066.1	Bit	…
DO19	V1066.2	Bit	…
DO20	V1066.3	Bit	…

In Logo Soft Comfort können 20 digitale Ausgänge konfiguriert werden, obwohl der von Siemens angegebene Addressbereich wesentlich weiter reicht, von 1064 bis 1071 (8 Bytes).

Auf digitale Ausgänge kann auch lesend zugegriffen werden, um zu prüfen, welcher aktuelle Wert anliegt (True, False):

```
plc.write("V1065.0", True)
print("DO9: " + str(plc.read("V1065.0")))
```

LOGO! 8 snap7 address mapping **0BA8**

Analog Output	VM Address in LOGO! 8		Python sample to read from VM
AO1	VW1072	2 Byte	plc.write("VW1072", value)
AO2	VW1074	2 Byte	plc.write("VW1074", value)
AO3	VW1076	2 Byte	…
AO4	VW1078	2 Byte	…
AO5	VW1080	2 Byte	…
AO6	VW1082	2 Byte	…
AO7	VW1084	2 Byte	…
AO8	VW1086	2 Byte	…

In Logo Soft Comfort können 8 analoge Ausgänge konfiguriert werden, obwohl der von Siemens angegebene Addressbereich wesentlich weiter reicht, von 1072 bis 1103 (32 Bytes).

Auf analoge Ausgänge kann auch lesend zugegriffen werden, um zu prüfen, welcher aktuelle Wert anliegt:

```
plc.write("VW1072", 8000)
print("AO1: " + str(plc.read("VW1072")))
```

Digital Memory	VM Address in LOGO! 8		Python sample to read from VM
M1	V1104.0	Bit	print("M1: " + str(plc.read("V1104.0")))
M2	V1104.1	Bit	print("M2: " + str(plc.read("V1104.1")))
M3	V1104.2	Bit	…
M4	V1104.3	Bit	…
M5	V1104.4	Bit	…
M6	V1104.5	Bit	…
M7	V1104.6	Bit	plc.write("V1104.6", True)
M8	V1104.7	Bit	plc.write("V1104.7", False)
M9	V1105.0	Bit	…
M10	V1105.1	Bit	…
M11	V1105.2	Bit	…
M12	V1105.3	Bit	…
M13	V1105.4	Bit	…
M14	V1105.5	Bit	…
M15	V1105.6	Bit	…
M16	V1105.7	Bit	…
M17	V1106.0	Bit	…
M18	V1106.1	Bit	…
M19	V1106.2	Bit	…
M20	V1106.3	Bit	…
M21	V1106.4	Bit	…
M22	V1106.5	Bit	…

LOGO! 8 snap7 address mapping **0BA8**

M23	V1106.6	Bit	…
M24	V1106.7	Bit	…
M25	V1107.0	Bit	…
M26	V1107.1	Bit	…
M27	V1107.2	Bit	…
M28	V1107.3	Bit	…
M29	V1107.4	Bit	…
M30	V1107.5	Bit	…
M31	V1107.6	Bit	…
M32	V1107.7	Bit	…
M33	V1108.0	Bit	…
M34	V1108.1	Bit	…
M35	V1108.2	Bit	…
M36	V1108.3	Bit	…
M37	V1108.4	Bit	…
M38	V1108.5	Bit	…
M39	V1108.6	Bit	…
M40	V1108.7	Bit	…
M41	V1109.0	Bit	…
M42	V1109.1	Bit	…
M43	V1109.2	Bit	…
M44	V1109.3	Bit	…
M45	V1109.4	Bit	…
M46	V1109.5	Bit	…
M47	V1109.6	Bit	…
M48	V1109.7	Bit	…
M49	V1110.0	Bit	…
M50	V1110.1	Bit	…
M51	V1110.2	Bit	…
M52	V1110.3	Bit	…
M53	V1110.4	Bit	…
M54	V1110.5	Bit	…
M55	V1110.6	Bit	…
M56	V1110.7	Bit	…
M57	V1111.0	Bit	…
M58	V1111.1	Bit	…
M59	V1111.2	Bit	…
M60	V1111.3	Bit	…
M61	V1111.4	Bit	…
M62	V1111.5	Bit	…
M63	V1111.6	Bit	…
M64	V1111.7	Bit	…

In Logo Soft Comfort können 64 digitale Merker (Flags) konfiguriert werden, obwohl der von Sie
angegebene Addressbereich wesentlich weiter reicht, von 1104 bis 1117 (14 Bytes).

LOGO! 8 snap7 address mapping

0BA8

Analog Memory	VM Address in LOGO! 8		Python sample to read from VM
AM1	VW1118	2 Byte	print("AM1: " + str(plc.read("VW1118")))
AM2	VW1120	2 Byte	print("AM2: " + str(plc.read("VW1120")))
AM3	VW1122	2 Byte	...
AM4	VW1124	2 Byte	...
AM5	VW1126	2 Byte	...
AM6	VW1128	2 Byte	...
AM7	VW1130	2 Byte	...
AM8	VW1132	2 Byte	...
AM9	VW1134	2 Byte	...
AM10	VW1136	2 Byte	...
AM11	VW1138	2 Byte	...
AM12	VW1140	2 Byte	plc.write("VW1140", 500)
AM13	VW1142	2 Byte	plc.write("VW1142", 1500)
AM14	VW1144	2 Byte	...
AM15	VW1146	2 Byte	...
AM16	VW1148	2 Byte	...
AM17	VW1150	2 Byte	...
AM18	VW1152	2 Byte	...
AM19	VW1154	2 Byte	...
AM20	VW1156	2 Byte	...
AM21	VW1158	2 Byte	...
AM22	VW1160	2 Byte	...
AM23	VW1162	2 Byte	...
AM24	VW1164	2 Byte	...
AM25	VW1166	2 Byte	...
AM26	VW1168	2 Byte	...
AM27	VW1170	2 Byte	...
AM28	VW1172	2 Byte	...
AM29	VW1174	2 Byte	...
AM30	VW1176	2 Byte	...
AM31	VW1178	2 Byte	...
AM32	VW1180	2 Byte	...
AM33	VW1182	2 Byte	...
AM34	VW1184	2 Byte	...
AM35	VW1186	2 Byte	...
AM36	VW1188	2 Byte	...
AM37	VW1190	2 Byte	...
AM38	VW1192	2 Byte	...

AM39	VW1194	2 Byte	…
AM40	VW1196	2 Byte	…
AM41	VW1198	2 Byte	…
AM42	VW1200	2 Byte	…
AM43	VW1202	2 Byte	…
AM44	VW1204	2 Byte	…
AM45	VW1206	2 Byte	…
AM46	VW1208	2 Byte	…
AM47	VW1210	2 Byte	…
AM48	VW1212	2 Byte	…
AM49	VW1214	2 Byte	…
AM50	VW1216	2 Byte	…
AM51	VW1218	2 Byte	…
AM52	VW1220	2 Byte	…
AM53	VW1222	2 Byte	…
AM54	VW1224	2 Byte	…
AM55	VW1226	2 Byte	…
AM56	VW1228	2 Byte	…
AM57	VW1230	2 Byte	…
AM58	VW1232	2 Byte	…
AM59	VW1234	2 Byte	…
AM60	VW1236	2 Byte	…
AM61	VW1238	2 Byte	…
AM62	VW1240	2 Byte	…
AM63	VW1242	2 Byte	…
AM64	VW1244	2 Byte	…

LOGO! 8 snap7 address mapping

0BA8

Network Input	VM Address in LOGO! 8		Python sample to read from VM
NI1	V1246.0	Bit	siehe Digitale Eingänge
NI2	V1246.1	Bit	…
NI3	V1246.2	Bit	…
NI4	V1246.3	Bit	…
NI5	V1246.4	Bit	…
NI6	V1246.5	Bit	…
NI7	V1246.6	Bit	…
NI8	V1246.7	Bit	…
NI9	V1247.0	Bit	…
NI10	V1247.1	Bit	…
NI11	V1247.2	Bit	…
NI12	V1247.3	Bit	…
NI13	V1247.4	Bit	…
NI14	V1247.5	Bit	…
NI15	V1247.6	Bit	…
NI16	V1247.7	Bit	…
NI17	V1248.0	Bit	…
NI18	V1248.1	Bit	…
NI19	V1248.2	Bit	…
NI20	V1248.3	Bit	…
NI21	V1248.4	Bit	…
NI22	V1248.5	Bit	…
NI23	V1248.6	Bit	…
NI24	V1248.7	Bit	…
NI25	V1249.0	Bit	…
NI26	V1249.1	Bit	…
NI27	V1249.2	Bit	…
NI28	V1249.3	Bit	…
NI29	V1249.4	Bit	…
NI30	V1249.5	Bit	…
NI31	V1249.6	Bit	…
NI32	V1249.7	Bit	…
NI33	V1250.0	Bit	…
NI34	V1250.1	Bit	…
NI35	V1250.2	Bit	…
NI36	V1250.3	Bit	…
NI37	V1250.4	Bit	…
NI38	V1250.5	Bit	…

NI39	V1250.6	Bit	…
NI40	V1250.7	Bit	…
NI41	V1251.0	Bit	…
NI42	V1251.1	Bit	…
NI43	V1251.2	Bit	…
NI44	V1251.3	Bit	…
NI45	V1251.4	Bit	…
NI46	V1251.5	Bit	…
NI47	V1251.6	Bit	…
NI48	V1251.7	Bit	…
NI49	V1252.0	Bit	…
NI50	V1252.1	Bit	…
NI51	V1252.2	Bit	…
NI52	V1252.3	Bit	…
NI53	V1252.4	Bit	…
NI54	V1252.5	Bit	…
NI55	V1252.6	Bit	…
NI56	V1252.7	Bit	…
NI57	V1253.0	Bit	…
NI58	V1253.1	Bit	…
NI59	V1253.2	Bit	…
NI60	V1253.3	Bit	…
NI61	V1253.4	Bit	…
NI62	V1253.5	Bit	…
NI63	V1253.6	Bit	…
NI64	V1253.7	Bit	…

LOGO! 8 snap7 address mapping 0BA8

Network A-Input	VM Address in LOGO! 8		Python sample to read from VM
NAI1	VW1262	2 Byte	siehe Analoge Eingänge
NAI2	VW1264	2 Byte	…
NAI3	VW1266	2 Byte	…
NAI4	VW1268	2 Byte	…
NAI5	VW1270	2 Byte	…
NAI6	VW1272	2 Byte	…
NAI7	VW1274	2 Byte	…
NAI8	VW1276	2 Byte	…
NAI9	VW1278	2 Byte	…
NAI10	VW1280	2 Byte	…
NAI11	VW1282	2 Byte	…
NAI12	VW1284	2 Byte	…
NAI13	VW1286	2 Byte	…
NAI14	VW1288	2 Byte	…
NAI15	VW1290	2 Byte	…
NAI16	VW1292	2 Byte	…
NAI17	VW1294	2 Byte	…
NAI18	VW1296	2 Byte	…
NAI19	VW1298	2 Byte	…
NAI20	VW1300	2 Byte	…
NAI21	VW1302	2 Byte	…
NAI22	VW1304	2 Byte	…
NAI23	VW1306	2 Byte	…
NAI24	VW1308	2 Byte	…
NAI25	VW1310	2 Byte	…
NAI26	VW1312	2 Byte	…
NAI27	VW1314	2 Byte	…
NAI28	VW1316	2 Byte	…
NAI29	VW1318	2 Byte	…
NAI30	VW1320	2 Byte	…
NAI31	VW1322	2 Byte	…
NAI32	VW1324	2 Byte	…

LOGO! 8 snap7 address mapping **0BA8**

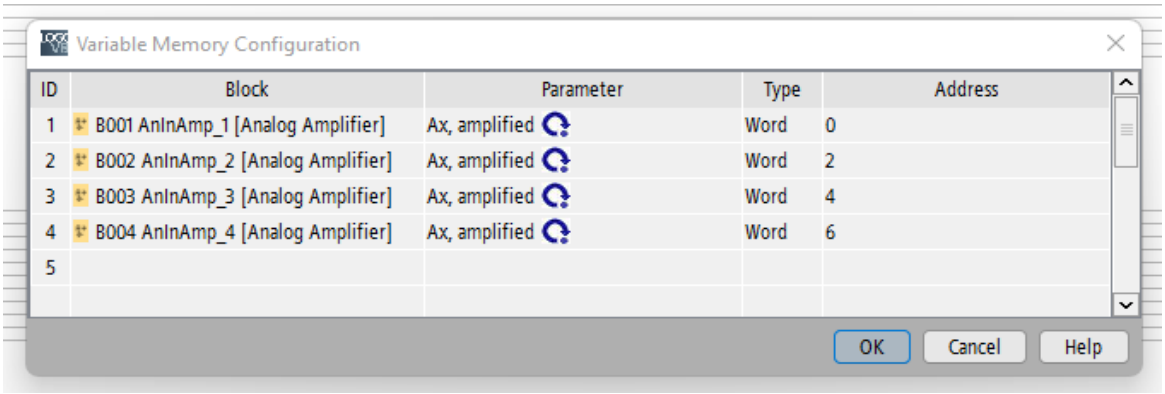
Network Output	VM Address in LOGO! 8		Python sample to read from VM
NO1	V1390.0	Bit	siehe Digitale Ausgänge
NO2	V1390.1	Bit	…
NO3	V1390.2	Bit	…
NO4	V1390.3	Bit	…
NO5	V1390.4	Bit	…
NO6	V1390.5	Bit	…
NO7	V1390.6	Bit	…
NO8	V1390.7	Bit	…
NO9	V1391.0	Bit	…
NO10	V1391.1	Bit	…
NO11	V1391.2	Bit	…
NO12	V1391.3	Bit	…
NO13	V1391.4	Bit	…
NO14	V1391.5	Bit	…
NO15	V1391.6	Bit	…
NO16	V1391.7	Bit	…
NO17	V1392.0	Bit	…
NO18	V1392.1	Bit	…
NO19	V1392.2	Bit	…
NO20	V1392.3	Bit	…
NO21	V1392.4	Bit	…
NO22	V1392.5	Bit	…
NO23	V1392.6	Bit	…
NO24	V1392.7	Bit	…
NO25	V1393.0	Bit	…
NO26	V1393.1	Bit	…
NO27	V1393.2	Bit	…
NO28	V1393.3	Bit	…
NO29	V1393.4	Bit	…
NO30	V1393.5	Bit	…
NO31	V1393.6	Bit	…
NO32	V1393.7	Bit	…
NO33	V1394.0	Bit	…
NO34	V1394.1	Bit	…
NO35	V1394.2	Bit	…
NO36	V1394.3	Bit	…
NO37	V1394.4	Bit	…
NO38	V1394.5	Bit	…

NO39	V1394.6	Bit	…
NO40	V1394.7	Bit	…
NO41	V1395.0	Bit	…
NO42	V1395.1	Bit	…
NO43	V1395.2	Bit	…
NO44	V1395.3	Bit	…
NO45	V1395.4	Bit	…
NO46	V1395.5	Bit	…
NO47	V1395.6	Bit	…
NO48	V1395.7	Bit	…
NO49	V1396.0	Bit	…
NO50	V1396.1	Bit	…
NO51	V1396.2	Bit	…
NO52	V1396.3	Bit	…
NO53	V1396.4	Bit	…
NO54	V1396.5	Bit	…
NO55	V1396.6	Bit	…
NO56	V1396.7	Bit	…
NO57	V1397.0	Bit	…
NO58	V1397.1	Bit	…
NO59	V1397.2	Bit	…
NO60	V1397.3	Bit	…
NO61	V1397.4	Bit	…
NO62	V1397.5	Bit	…
NO63	V1397.6	Bit	…
NO64	V1397.7	Bit	…

LOGO! 8 snap7 address mapping **0BA8**

Network A-Outpu	VM Address in LOGO! 8		Python sample to read from VM
NAO1	VW1406	2 Byte	siehe Analoge Ausgänge
NAO2	VW1408	2 Byte	…
NAO3	VW1410	2 Byte	…
NAO4	VW1412	2 Byte	…
NAO5	VW1414	2 Byte	…
NAO6	VW1416	2 Byte	…
NAO7	VW1418	2 Byte	…
NAO8	VW1420	2 Byte	…
NAO9	VW1422	2 Byte	…
NAO10	VW1424	2 Byte	…
NAO11	VW1426	2 Byte	…
NAO12	VW1428	2 Byte	…
NAO13	VW1430	2 Byte	…
NAO14	VW1432	2 Byte	…
NAO15	VW1434	2 Byte	…
NAO16	VW1436	2 Byte	…

Variable Memory	VM Address in LOGO! 8		Python sample to read from VM



Auf den variablen Speicher wird analog zu den anderen Speicherbereichen zugegriffen. Der Variable Speicher (auch "Variable Memory Configuration") muss jedoch vor Zugriff in Logo Soft Comfort konfiguriert werden.

VMC-ID1	VW0	2 Byte	print("VMC-ID1: " + str(plc.read("VW0")))
VMC-ID2	VW2	2 Byte	print("VMC-ID2: " + str(plc.read("VW2")))
VMC-ID3	VW4	2 Byte	…
VMC-ID4	VW6	2 Byte	…