



Open Communication Interface for Road Traffic Control Systems  
Offene Schnittstellen für die Straßenverkehrstechnik

# **OCIT-Outstations Lichtsignalsteuergeräte Version 2.0**

## **Testspezifikation 6 AP-Werte**

OCIT-O V2.0 Testspezifikation 6 V1.0\_A01

**OCIT Developer Group (ODG)**

OCIT® ist eine registrierte Marke der Firmen Dambach, Siemens, Signalbau Huber, Stoye und Stührenberg

# **OCIT-Outstations Lichtsignalsteuergeräte Version 2.0**

## **Testspezifikation 6 AP-Werte**

Dokument: OCIT-O V2.0 Testspezifikation 6 V1.0\_A01

Herausgeber: OCIT Developer Group (ODG)

Kontakt: [www.ocit.org](http://www.ocit.org)

Copyright © 2013 ODG. Änderungen vorbehalten. Dokumente mit Versions- oder Ausgabestände neueren Datums ersetzen alle Inhalte vorhergehender Versionen.

## Inhaltsverzeichnis

Dokumentation .....	4
1 Allgemeines .....	5
2 Testaufbau, Hilfsmittel .....	5
3 Liste der Testfälle .....	5
3.1 Detaillierte Testfälle .....	6
3.1.1 AP-Wert Abfrage mit InstanceInfo.....	6
3.1.2 Beauftragung eines nicht existierenden AP-Wertes.....	7
3.1.3 Beauftragung eines existierenden AP-Wertes .....	7
3.1.4 Abfrage des aktuellen Wertes eines nicht existierenden AP-Wertes.....	8
3.1.5 Abfrage des aktuellen Wertes eines existierenden AP-Wertes .....	9
3.1.6 Abfrage ob ein nicht existierender AP-Wert von außen verändert werden kann .....	9
3.1.7 Abfrage ob ein existierender AP-Wert von außen verändert werden kann...	10
3.1.8 Verändern eines nicht existierenden AP-Wertes von außen.....	10
3.1.9 Verändern eines existierenden, nicht veränderbaren AP-Wertes von außen.	11
3.1.10 Verändern eines existierenden, veränderbaren AP-Wertes von außen .....	11
3.1.11 Beauftragung eines AEAPVektor-Elementes .....	12

## Dokumentenstand

Version Zustand	Verteilerkreis	Datum	Kommentar
V1.0_A01	Public	31.01.2013	Neu

## Dokumentation

Es gilt jeweils das Dokument und die Datenspezifikation mit dem höchsten Ausgabestand!

Dokumente	Titel
OCIT-O V2.0 Testsuite MAT Handbuch	Handbuch zum Testen von OCIT-Outstations Version 2.0 für Lichtsignalsteuergeräte mit der OCIT-O V2.0 Testsuite MAT.
OCIT-O V2.0 Testspezifikation 1	Kommunikation Profil 1 und 2
OCIT-O V2.0 Testspezifikation 2	Systemfunktionen
OCIT-O V2.0 Testspezifikation 3	Lichtsignalsteuergerät Schalten
OCIT-O V2.0 Testspezifikation 4	Meldungsverhalten durch Ereignisse am Lichtsignalsteuergerät
OCIT-O V2.0 Testspezifikation 5	Erfassung von Detektor- und Visualisierungsdaten
OCIT-O V2.0 Testspezifikation 6	AP-Werte
OCIT-O V2.0 Testspezifikation 7	Versorgungstransaktionen
OCIT-O V2.0 Testspezifikation 8	Versorgen und Rücklesen Block VT-Grunddaten / Festzeit
OCIT-O V2.0 Testspezifikation 9	Versorgen und Rücklesen Block VT-Daten mit Netzbezug
OCIT-O V2.0 Testspezifikation 10	Versionierung Herstellerspezifisch
OCIT-O V2.0 Testspezifikation 11	Erweiterte Detektorwerte
OCIT-O V2.0 Testversorgung	Beschreibung der Testversorgung
Knoten_ODG_Test.xml	OCIT-O Testversorgung als XML-Datei. Geeignet zur Versorgung der Lichtsignalsteuergeräte über einen OCIT-I Versorgungsdatenserver.

# 1 Allgemeines

Nach Absprache im 14. Workshop OCIT-O Lstg V2.0 wird der Test der AP-Werte in drei Ausbaustufen durchgeführt.

- Stufe 1 entspricht einem Konformitätstest, bei dem mit den drei standardisierten AP-Werten (TX,PH,UE) getestet wird.
- Stufe 2 entspricht einem Interoperabilitätstest und bezieht auch die nicht standardisierten AP-Werte mit ein. Dabei muss vor dem Test konfiguriert werden, welche AP-Werte vom Gerät (bzw. vom Steuerungsverfahren) erwartet werden, welche von diesen schreibbar oder nur lesbar sind und in welchem Zyklus sie erwartet werden.
- Stufe 3 prüft optionale Parameter wie z.B. die APWertGroup

## 2 Testaufbau, Hilfsmittel

Siehe Testfälle.

## 3 Liste der Testfälle

### Legende:

<b>MAT</b>	Teil des Minimal Acceptance Tests
<b>Std</b>	Teil des Standardtests
<b>Opt</b>	Optionaler Test
<b>HS</b>	Ergänzende Angaben zu Std oder Opt
<b>P1</b>	Profil 1
<b>P2</b>	Profil 2
<b>V2</b>	Nur ab OCIT-O Version V2.0

ID	Testfälle	MAT	Std	Opt	HS
6.1	AP-Wert Abfrage mit InstanceInfo	x			
6.2	Beauftragung eines nicht existierenden AP-Wertes	x			
6.3	Beauftragung eines existierenden AP-Wertes	x			
6.4	Abfrage des aktuellen Wertes eines nicht existierenden AP-Wertes	x			

ID	Testfälle	MAT	Std	Opt	HS
6.5	Abfrage des aktuellen Wertes eines existierenden AP-Wertes	x			
6.6	Abfrage ob ein nicht existierender AP-Wert von außen verändert werden kann		V2		
6.7	Abfrage ob ein existierender AP-Wert von außen verändert werden kann		V2		
6.8	Verändern eines nicht existierenden AP-Wertes von außen		x		
6.9	Verändern eines existierenden, nicht veränderbaren AP-Wertes von außen		x		
6.10	Verändern eines existierenden, veränderbaren AP-Wertes von außen			x	
6.11	Beauftragung eines AEAPWertVektor-Elementes			x	

### 3.1 Detaillierte Testfälle

#### 3.1.1 AP-Wert Abfrage mit InstanceInfo

ID	6.1	
<b>Beschreibung</b>	Die Liste der im Gerät vorhandenen AP-Werte wird über die Methode InstanceInfo abgefragt. Alternativ oder zusätzlich kann auch ExtendedInstanceInfo benutzt werden.	
<b>Ausgangszustand</b>	Die BTPPL-Kommunikation funktioniert grundsätzlich.	
<b>Testablauf</b>	Automatisch (Vorausgesetzt, die erwarteten AP-Werte sind definiert)	
<b>Testhilfsmittel</b>	-	
<b>Aktion 1</b>	Aufruf von SystemobjektFeldgeraet::InstanceInfo	0:815.104 (1:505)
<b>Erwartetes Verhalten</b>	RetCode == OK  Liste der versorgten AP-Werte wird geliefert. Die 3 standardisierten AP-Werte (TX,UH,UE) müssen immer geliefert werden.	

### 3.1.2 Beauftragung eines nicht existierenden AP-Wertes

ID	6.2
<b>Beschreibung</b>	Ein nicht existierender AP-Wert soll im Archiv beauftragt werden. Erwartet wird dabei eine Fehlermeldung. <i>Hinweis: Für den Test wird die Liste 35 benutzt (theoretisch wären auch die Listen 31-34 möglich).</i>
<b>Ausgangszustand</b>	Die BTPPL-Kommunikation funktioniert grundsätzlich.
<b>Testablauf</b>	Automatisch.
<b>Testhilfsmittel</b>	-
<b>Aktion 1</b>	Aufruf von Liste::AddAuftrag (1:407) 0:400.108
<b>Erwartetes Verhalten</b>	RetCode == OK Rückgabe der neuen Auftragsnummer
<b>Aktion 2</b>	Aufruf von MWAuftragAbtastAenderung::AddElement mit AEAPWert (1:434) 1:407.120
<b>Erwartetes Verhalten</b>	RetCode == OK Rückgabe der neuen Elementnummer
<b>Aktion 3</b>	Aufruf von AEAPWert::SetAP <u>mit nicht existierendem</u> AP-Wert 1:434.153 (AP-Wert-Kennung)
<b>Erwartetes Verhalten</b>	RetCode == PARAM_INVALID

### 3.1.3 Beauftragung eines existierenden AP-Wertes

ID	6.3
<b>Beschreibung</b>	Ein existierender AP-Wert soll im Archiv beauftragt werden. Erwartet wird dabei eine OK-Meldung.
<b>Ausgangszustand</b>	Die BTPPL-Kommunikation funktioniert grundsätzlich.
<b>Testablauf</b>	Automatisch

<b>Testhilfsmittel</b>	-
<b>Aktion 1</b>	Aufruf von Liste::AddAuftrag (1:407) 0:400.108
<b>Erwartetes Verhalten</b>	RetCode == OK Rückgabe der neuen Auftragsnummer
<b>Aktion 2</b>	Aufruf von MWAuftragAbtastAenderung::AddElement mit AEAPWert (1:434) 1:407.120
<b>Erwartetes Verhalten</b>	RetCode == OK Rückgabe der neuen Elementnummer
<b>Aktion 3</b>	Aufruf von AEAPWert::SetAP <u>mit existierendem</u> AP-Wert 1:434.153 (AP-Wert-Kennung von TX)
<b>Erwartetes Verhalten</b>	RetCode == OK

### 3.1.4 Abfrage des aktuellen Wertes eines nicht existierenden AP-Wertes

<b>ID</b>	<b>6.4</b>
<b>Beschreibung</b>	Der aktuelle Wert eines <u>nicht existierenden</u> AP-Wertes wird abgefragt. Hier am Beispiel eines <u>APWertUshort</u> .
<b>Ausgangszustand</b>	Die BTPPL-Kommunikation funktioniert grundsätzlich.
<b>Testablauf</b>	Automatisch
<b>Testhilfsmittel</b>	-
<b>Aktion 1</b>	Aufruf von ApWertUshort::GetWert 1:506.16 (ApWertName)
<b>Erwartetes Verhalten</b>	RetCode == ERR_PATH_VAL



### 3.1.5 Abfrage des aktuellen Wertes eines existierenden AP-Wertes

ID	6.5
<b>Beschreibung</b>	Der aktuelle Wert eines <u>existierenden</u> AP-Wertes wird abgefragt.
<b>Ausgangszustand</b>	Die BTPPL-Kommunikation funktioniert grundsätzlich.
<b>Testablauf</b>	Automatisch
<b>Testhilfsmittel</b>	-
<b>Aktion 1</b>	Aufruf von ApWertUshort::GetWert 1:506.16 (ApWertName)
<b>Erwartetes Verhalten</b>	RetCode == OK Rückgabe des AP-Wertes

### 3.1.6 Abfrage ob ein nicht existierender AP-Wert von außen verändert werden kann

ID	6.6
<b>Beschreibung</b>	Prüft, ob ein <u>nicht existierender</u> AP-Werte von außen verändert werden kann.
<b>Ausgangszustand</b>	Die BTPPL-Kommunikation funktioniert grundsätzlich.
<b>Testablauf</b>	Automatisch
<b>Testhilfsmittel</b>	-
<b>Aktion 1</b>	Aufruf von ApWertUshort::IsWritable 1:506.18 (ApWertName)
<b>Erwartetes Verhalten</b>	RetCode == ERR_PATH_VAL

### 3.1.7 Abfrage ob ein existierender AP-Wert von außen verändert werden kann

ID	6.7
<b>Beschreibung</b>	Prüft, ob ein <u>existierender</u> AP-Werte von außen verändert werden kann.
<b>Ausgangszustand</b>	Die BTPPL-Kommunikation funktioniert grundsätzlich.
<b>Testablauf</b>	Automatisch
<b>Testhilfsmittel</b>	-
<b>Aktion 1</b>	Aufruf von ApWertUshort::IsWritable 1:506.18 (ApWertName)
<b>Erwartetes Verhalten</b>	RetCode == OK  WriteableFlag (Ob „true“ oder „false“ ist für den Testerfolg nicht entscheidend)

### 3.1.8 Verändern eines nicht existierenden AP-Wertes von außen

ID	6.8
<b>Beschreibung</b>	Der Wert eines <u>nicht existierenden</u> AP-Wertes soll von außen verändert werden.
<b>Ausgangszustand</b>	Die BTPPL-Kommunikation funktioniert grundsätzlich.
<b>Testablauf</b>	Automatisch
<b>Testhilfsmittel</b>	-
<b>Aktion 1</b>	Aufruf von ApWertUshort::SetWert 1:506.17 (ApWertName,Wert)
<b>Erwartetes Verhalten</b>	RetCode == ERR_PATH_VAL

### 3.1.9 Verändern eines existierenden, nicht veränderbaren AP-Wertes von außen

ID	6.9
<b>Beschreibung</b>	Der Wert eines <u>existierenden, aber nicht veränderbaren</u> AP-Wertes soll von außen verändert werden. Hier am Beispiel eines <u>APWertUshort</u> .
<b>Ausgangszustand</b>	Die BTPPL-Kommunikation funktioniert grundsätzlich.
<b>Testablauf</b>	Automatisch, wenn zuvor bekannt, dass Wert nicht von außen veränderbar ist.
<b>Testhilfsmittel</b>	-
<b>Aktion 1</b>	Aufruf von ApWertUshort::SetWert 1:506.18 (ApWertName,Wert)
<b>Erwartetes Verhalten</b>	RetCode == NOT_POSSIBLE

Hinweis: In der Ausbaustufe 2 sollte hier der Test erweitert werden um zu überprüfen, dass der Wert auch wirklich nicht verändert wurde. Dazu muss ein AP-Wert ausgewählt werden, dessen Wert während der Testzeit nicht von innen verändert wird. Dann würde der Test folgendermaßen erweitert:

- Wert auslesen und merken
- Verändern des Wertes durch Aufruf von SetWert 1:506.18 (ApWertName,Wert)
- Mit GetWert prüfen, dass der AP-Wert noch auf dem alten Wert steht und nicht verändert wurde.

### 3.1.10 Verändern eines existierenden, veränderbaren AP-Wertes von außen

ID	6.10
<b>Beschreibung</b>	Der Wert eines <u>existierenden, veränderbaren</u> AP-Wertes soll von außen verändert werden.  Es wird ein AP-Wert bestimmt, dessen Wert während der Testzeit <u>ausschließlich</u> durch die Testsuite beschrieben wird!  <i>Hinweis: Dieser Test kann erst in Ausbaustufe 2 getestet werden, da die standardisierten AP-Werte (TX,PH,UE) nicht von außen veränderbar sind.</i>
<b>Ausgangszustand</b>	Die BTPPL-Kommunikation funktioniert grundsätzlich.
<b>Testablauf</b>	Automatisch, wenn zuvor bekannt, dass Wert nicht von außen veränderbar ist.

<b>Testhilfsmittel</b>	-
<b>Aktion 1</b>	Aufruf von ApWertUshort::GetWert 1:506.16 (ApWertName)
<b>Erwartetes Verhalten</b>	RetCode == OK Rückgabe des AP-Wertes (Ist-Wert merken!)
<b>Aktion 2</b>	Aufruf von ApWertUshort::SetWert mit Wert != Ist-Wert 1:506.17 (ApWertName, Wert)
<b>Erwartetes Verhalten</b>	RetCode == OK
<b>Aktion 3</b>	Aufruf von ApWertUshort::GetWert 1:506.16 (ApWertName)
<b>Erwartetes Verhalten</b>	RetCode == OK Rückgabe des AP-Wertes. Der hier gelesene Wert muss mit dem geschriebenen Wert übereinstimmen.

### 3.1.11 Beauftragung eines AEAPVektor-Elementes

<b>ID</b>	<b>6.11</b>
<b>Beschreibung</b>	Die drei Standard-ApWerte sollen in einem AEAPWertVektor beauftragt werden.
<b>Ausgangszustand</b>	Die BTPPL-Kommunikation funktioniert grundsätzlich.
<b>Testablauf</b>	Automatisch
<b>Testhilfsmittel</b>	-
<b>Aktion 1</b>	Aufruf von Liste::AddAuftrag 0:400.108 (1:407)
<b>Erwartetes Verhalten</b>	RetCode == OK Rückgabe der neuen Auftragsnummer
<b>Aktion 2</b>	Aufruf von MWAuftragAbtastAenderung::AddElement mit AEAPWertVektor 1:407.120 (1:437)
<b>Erwartetes Verhalten</b>	RetCode == OK Rückgabe der neuen Elementnummer

<b>Aktion 3</b>	Aufruf von AEAPWert::SetAPListe; die Liste beinhaltet die drei Standard-ApWerte „TX“, „PH“ und „UE“ 1:437.156 (AP-Wert-Kennung von TX,PH,UE)
<b>Erwartetes Verhalten</b>	RetCode == OK

OCIT-O V2.0 Testspezifikation 6 V1.0\_A01

Copyright © 2013 ODG

---