#### 7.3.2.2 TCI‑CD provided

This clause specifies the operations the TM shall provide to the TE.

##### 7.3.2.2.1 decode

|  |  |  |
| --- | --- | --- |
| Signature | Value decode(in TriMessageType message,  in Type decodingHypothesis) | |
| In Parameters | message | The encoded message to be decoded. |
| decodingHypothesis | The hypothesis the decoding can be based on. |
| Return Value | Returns the decoded value, if the value is of a compatible type as the decodingHypothesis, else the distinct value null. | |
| Constraint | This operation shall be called whenever the TE has to decode an encoded value. The TE might decode immediately after reception of an encoded value, or might for performance considerations postpone the decoding until the actual access of the encoded value. In case of an explicit codec call (using predefined functions decvalue and decvalue\_unichar), it is called if the decoding\_info parameter contains the default value (an empty string). | |
| Effect | This operation decodes message according to the encoding rules and returns a TTCN‑3 value. The decodingHypothesis shall be used to determine whether the encoded value can be decoded. If an encoding rule is not self‑sufficient, i.e. if the encoded message does not inherently contain its type decodingHypothesis shall be used. If the encoded value can be decoded without the decoding hypothesis, the distinct null value shall be returned if the type determined from the encoded message is not compatible with the decoding hypothesis. | |

##### 7.3.2.2.2 encode

|  |  |  |
| --- | --- | --- |
| Signature | TriMessageType encode(in Value value) | |
| In Parameters | value | The value to be encoded. |
| Return Value | Returns an encoded TriMessage for the specified encoding rule. | |
| Constraint | This operation shall be called whenever the TE has to encode a Value. In case of an explicit codec call (using predefined functions encvalue and encvalue\_unichar), it is called if the encoding\_info parameter contains the default value (an empty string). | |
| Effect | Returns an encoded TriMessage according to the encoding rules. | |

##### 7.3.2.2.1 decodeParam

|  |  |  |
| --- | --- | --- |
| Signature | Value decode(in TriMessageType message,  in Type decodingHypothesis,  in TString encodingInfo) | |
| In Parameters | message | The encoded message to be decoded. |
| decodingHypothesis | The hypothesis the decoding can be based on. |
| decodingInfo | Dynamic decoding parameters |
| Return Value | Returns the decoded value, if the value is of a compatible type as the decodingHypothesis, else the distinct value null. | |
| Constraint | This operation shall be called whenever the TE invokes the predefined functions decvalue or decvalue\_unichar in case the decoding\_info parameter contains a value that is different from the default value. | |
| Effect | This operation decodes message according to the encoding rules and returns a TTCN‑3 value. The decodingHypothesis shall be used to determine whether the encoded value can be decoded. If an encoding rule is not self‑sufficient, i.e. if the encoded message does not inherently contain its type decodingHypothesis shall be used. If the encoded value can be decoded without the decoding hypothesis, the distinct null value shall be returned if the type determined from the encoded message is not compatible with the decoding hypothesis. | |

##### 7.3.2.2.2 encodeParam

|  |  |  |
| --- | --- | --- |
| Signature | TriMessageType encode(in Value value, in TString encodingInfo) | |
| In Parameters | value | The value to be encoded. |
| encodingInfo | Dynamic encoding parameters |
| Return Value | Returns an encoded TriMessage for the specified encoding rule. | |
| Constraint | This operation shall be called whenever the TE invokes the predefined functions decvalue or decvalue\_unichar in case the encoding\_info parameter contains a value that is different from the default value. | |
| Effect | Returns an encoded TriMessage according to the encoding rules. | |

#### 8.5.2.1 TCI‑CD provided

The TCI‑CD Provided interface is mapped to the following interface:

// TCI‑CD

// TE ‑> CD

package org.etsi.ttcn.tci;

public interface TciCDProvided {

public Value decode (TriMessage message, Type decodingHypothesis );

public TriMessage encode (Value value);

public Value decodeParam (TriMessage message, Type decodingHypothesis,  
 String decodingInfo );

public TriMessage encodeParam (Value value, String encodingInfo);

}

#### 9.4.2.1 TCI‑CD provided

The TCI‑CD Provided interface is mapped to the following interface:

Value tciDecode (BinaryString message, Type decHypothesis)

BinaryString tciEncode(Value value)

Value tciDecodeParam (BinaryString message, Type decHypothesis, String decodingInfo)

BinaryString tciEncodeParam(Value value, string encodingInfo)

NOTE: BinaryString type reused from TRI.

#### 10.6.2.2 TciCdProvided

This class defines the TCI\_CD provided interface. It is mapped to the following interface:

//Destructor

virtual ~TciCdProvided ();

//This operation is called whenever the TE has to decode and encode value

virtual TciValue \* decode (const TriMessage \*p\_message, const TciType \*p\_decodingHypothesis)=0;

//This operation is called whenever the TE has to encode a Value

virtual TriMessage \* encode (const TciValue \*p\_value)=0;

//This operation is called whenever the TE uses decvalue with a dynamic decoding parameter

virtual TciValue \* decodeParam (const TriMessage \*p\_message, const TciType \*p\_decodingHypothesis,

const TuniversalString & decodingInfo)=0;

//This operation is called whenever the TE uses encvalue with a dynamic encoding parameter

virtual TriMessage \* encodeParam (const TciValue \*p\_value, const TuniversalString & encodingInfo)=0;

#### 12.5.2.1 TCI-CD provided

The **TCI-CD provided** interface is mapped to the following interface:

public interface ITciCDProvided {  
 ITciValue Decode(ITriMessage message,  
 ITciType decodingHypothesis);  
 ITriMessage Encode(ITciValue value);  
 ITciValue DecodeParam(ITriMessage message,  
 ITciType decodingHypothesis,

string decodingInfo);  
 ITriMessage EncodeParam(ITciValue value,

string encodingInfo);  
}