APPLICATION NOTE

Understanding IEEE 1641—Implementing Synchros

Describing tests and measurements using synchros and resolvers in IEEE 1641

Ashley Hulme

Application Note 1201/ Issue 1, January 2012

Introduction

Looking at synchros and resolvers generally and descriptions of these in test programs in particular, it is clear that some explanation is required as to how these should be described in IEEE 1641. In traditional test requirements, the description of synchro and resolver related tests usually relates directly to the instrumentation being used. In ATLAS, the synchro and resolver nouns are not fully described and much is left to be assumed by the ATLAS programmer and the implementer. There are many types of synchro and resolver, and although the ATLAS nouns may be able to cover many of them, it is not clear how. This is because most implementations are only used to program specific instruments or implementations and do not describe the *signals* required.

IEEE 1641 can deal with all the different types of synchro and resolver but needs more than one TSF to cover all the cases. In this application note, the most common cases of synchros and resolvers are described, and the other versions may then be deduced from that starting point, and suitable models defined for each.

Sample TSF in IEEE 1641

The sample synchro TSF provided with the 1641 standard is configured as a source, with the reference signal defined by attributes. This represents a synchro with an internal source for the reference signal. The sample TSF also has attributes for zero offset, transformer ratio, and angle rate, which may not be needed in all cases, but these may be left at their default value of zero if not used. It is important to remember that this TSF is only an example, like all the TSFs in the standard, and should be considered the "correct" TSF for a synchro.

Synchro and resolver measurements

Synchro and resolver measurements are prime examples of the type that should be implemented using the *Measure As* facility in IEEE 1641. This is because most Automated Test Systems (ATS) that test synchros and resolvers use specific instruments designed for that purpose. That does not mean that the measurements cannot be performed by using the simple sensor BSCs and intrinsic measurement, but if generic measurement is best for the situation, then use it. To define a measurement TSF for the synchro, it is necessary to define the reference *As* signal, which is the expected signal. This could be the synchro TSF as provided with the 1641 standard, a modified version that also outputs the reference signal, or version of the TSF defined as a transformation TSF.

The basic CG control synchro

The synchro as a transformation TSF

Modelling the TSF

The best starting point is the basic control CG synchro (see Figure 1), which is initially modelled here as a transformation TSF. This represents a real synchro with a reference signal being input to the rotor terminals, the shaft angle being specified as an attribute value, and the stator outputs provided as the output signal. In the examples in this application note, the symbols R1 and R2 are used for the reference (rotor) terminals and the symbols S1, S2 and S3 for the three phase stator terminals. These may (of course) be replaced by whatever convention is used in the particular ATS implementation being used.

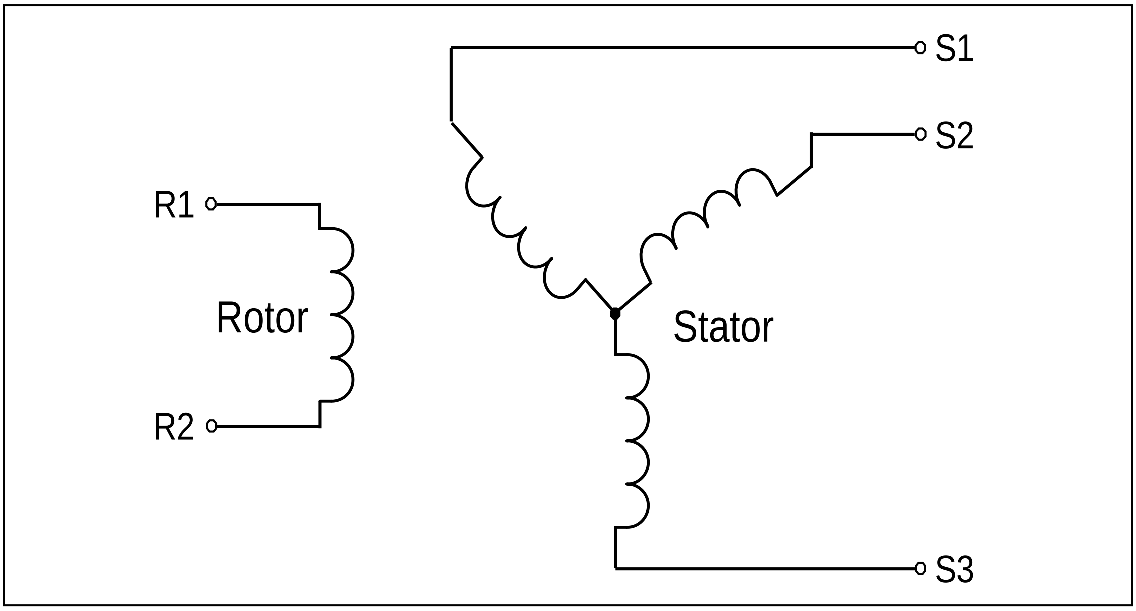


Figure 1—The basic CG synchro

The TSF "Synchro_CG" shown pictorially in Figure 2 represents the basic CG synchro in which the reference signal is provided as an input and the shaft angle is provided as an attribute. This TSF also allows for a shaft offset to be specified and a rotational speed for a moving shaft – these two attributes are defaulted to zero.

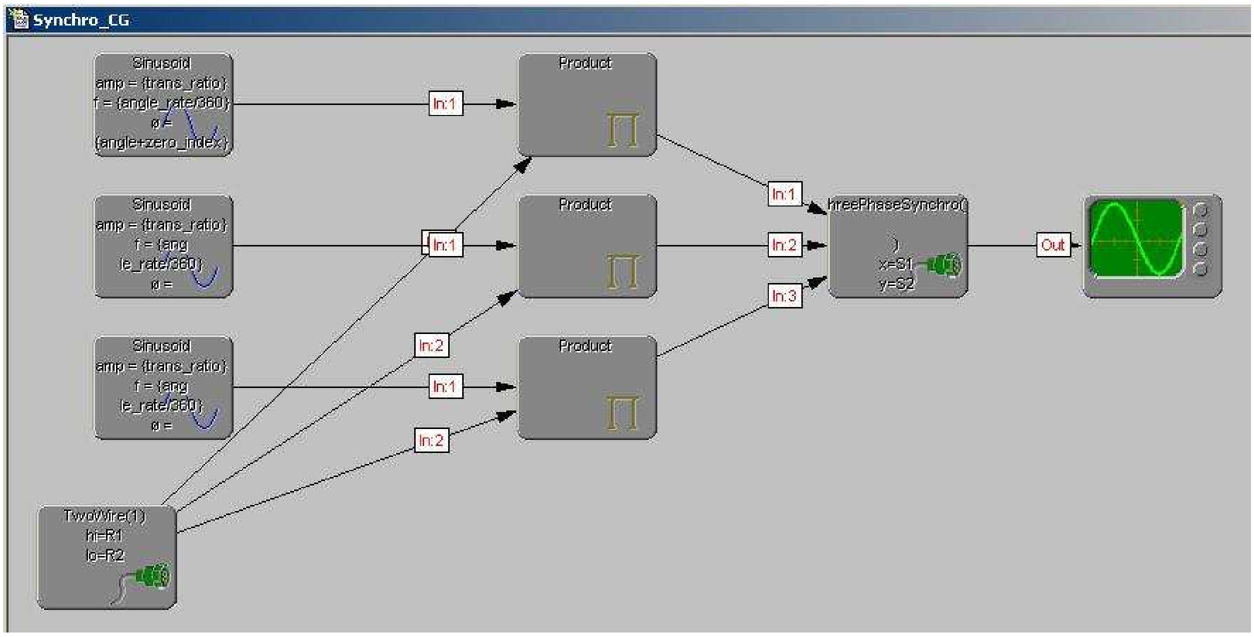


Figure 2—CG synchro TSF

In this transformation TSF, the reference input is via the TwoWire BSC (lower left in the figure) and this input signal is mixed with the shaft information to provide the output signal. The TSF interface attributes comprise shaft_angle (the position of the shaft in degrees away from the null position), zero_index (to allow for a rotor offset in degrees), angle_rate (to allow for the speed of a rotating shaft to be specified in deg/s), and trans_ratio (to allow the transformer ratio to be specified). An XML representation of this TSF is shown in Figure 3.

```xml
<?xml version="1.0" ?>
<!--  generated with newWaveX v4.0.0 (http://www.newWaveX.com)   -->
<tsf:TSF name="Synchro_CG" uuid="{23783E46-717F-4A04-B058-D4C6371B04AF}"
  xmlns:tsf="urn:IEEE-1641:2010:STDTSF" xmlns:std="urn:IEEE-1641:2010:STDBSC" xmlns="
  urn:IEEE-1641:2010:STDBSC">
 <tsf:interface>
  <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
   <xs:element name="Synchro_CG">
    <xs:annotation>
     <xs:documentation>Transmitter type Synchro with reference signal input,</xs:documentation>
    </xs:annotation>
    <xs:complexType>
     <xs:complexContent>
      <xs:extension base="SignalFunctionType">
       <xs:attribute name="shaft_angle" type="PlaneAngle" default="0 deg">
        <xs:annotation>
         <xs:documentation>The angle of the shaft from the null postion</xs:documentation>
        </xs:annotation>
       </xs:attribute>
      <xs:attribute name="zero_index" type="PlaneAngle" default="0 deg">
       <xs:annotation>
        <xs:documentation>Shaft offset angle</xs:documentation>
       </xs:annotation>
      </xs:attribute>
      <xs:attribute name="angle_rate" type="Frequency" default="0 Hz">
       <xs:annotation>
        <xs:documentation>The speed of rotation of the shaft (if moving) in deg/s</xs:documentation>
       </xs:annotation>
      </xs:attribute>
      <xs:attribute name="trans_ratio" type="double" default="1.0">
       <xs:annotation>
        <xs:documentation>Transformer Ratio (TR) usually between 0.1.and 1.0</xs:documentation>
       </xs:annotation>
      </xs:attribute>
     </xs:extension>
    </xs:complexContent>
   </xs:complexType>
  </xs:element>
 </xs:schema>
 </tsf:interface>
 <tsf:model>
  <Signal Out="Stator_pins">
   <Sinusoid name="Sinusoid__4194" amplitude="{trans_ratio}" frequency="{angle_rate/360}"
     phase="{angle+zero_index}" />
   <TwoWire name="Rotor_pins" lo="R2" hi="R1" channelWidth="1" />
   <Sinusoid name="Sinusoid__4198" amplitude="{trans_ratio}" frequency="{angle_rate/360}"
     phase="{angle+120+zero_index}" />
   <Sinusoid name="Sinusoid__4202" amplitude="{trans_ratio}" frequency="{angle_rate/360}"
     phase="{angle+240+zero_index}" />
   <Product name="Signal_S1" In="Sinusoid__4194 Rotor_pins" />
   <Product name="Signal_S2" In="Sinusoid__4198 Rotor_pins" />
   <Product name="Stator_S3" In="Sinusoid__4202 Rotor_pins" />
   <ThreePhaseSynchro name="Stator_pins" x="S1" y="S2" z="S3" channelWidth="3"
     In="Signal_S1 Signal_S2 Signal_S3" />
  </Signal>
 </tsf:model>
</tsf:TSF>
```

Figure 3—CG synchro TSF XML

Improving the model

There is a shortcoming in this initial example (Figure 2 & Figure 3) in that the UUT pin names are fixed (R1, R2, S1, S2, & S3). This has been done to make the input and outputs clear. It is usually better to omit the Connector BSCs and use the "In" BSC for the input. This allows the connections to be defined externally to the TSF and make the TSF more reusable. This is shown pictorially in Figure 4 and the model section of the XML definition is shown in Figure 5, (the interface section being identical).

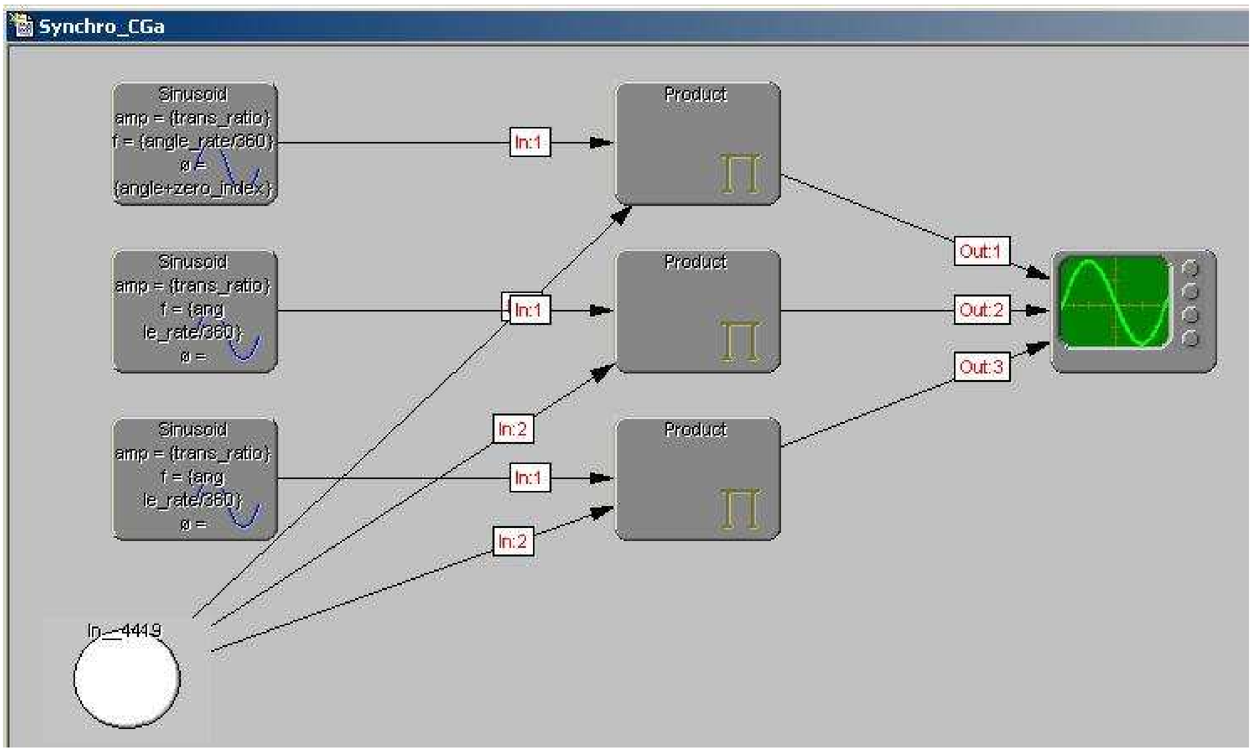


Figure 4—CG synchro TSF (without connectors)

```
<tsf:model>
  <Signal Out="Signal_S1 Signal_S2 Signal_S3" In="In__4419">
    <Sinusoid name="Sinusoid__4401" amplitude="{trans_ratio}"
       frequency="{angle_rate/360}" phase="{angle+zero_index}" />
    <Sinusoid name="Sinusoid__4405" amplitude="{trans_ratio}"
       frequency="{angle_rate/360}" phase="{angle+120+zero_index}" />
    <Sinusoid name="Sinusoid__4409" amplitude="{trans_ratio}"
       frequency="{angle_rate/360}" phase="{angle+240+zero_index}" />
    <Product name="Signal_S1" In="Sinusoid__4401 In__4419" />
    <Product name="Signal_S2" In="Sinusoid__4405 In__4419" />
    <Product name="Signal_S3" In="Sinusoid__4409 In__4419" />
  </Signal>
</tsf:model>
```

Figure 5—XML model section for TSF without connectors

Using the TSF

This TSF may be used in a situation where a reference signal is available from the UUT or a separate reference supply is available (as shown in the signal illustrated in Figure 6).

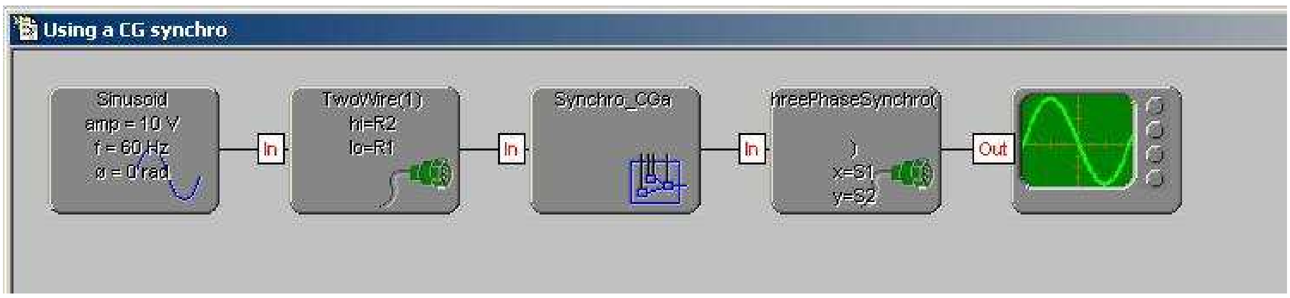


Figure 6—Using a CG synchro

The XML version of this signal definition is shown in Figure 7.

```xml
<?xml version="1.0" ?>
<!--  generated with newWaveX v4.0.0 (http://www.newWaveX.com)   -->
<Signal Out="ThreePhaseSynchro_4433" xmlns="urn:IEEE-1641:2010:STDBSC"
    xmlns:this="MyTSFLib" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="MyTSFLib MyTSFLib.xsd">
  <Sinusoid name="Sinusoid_4305" amplitude="10 V" frequency="60 Hz" />
  <TwoWire name="TwoWire_4432" lo="R1" hi="R2" channelWidth="1" In="Sinusoid_4305" />
  <this:Synchro_CGa name="Synchro_CGa_4425" In="TwoWire_4432" />
  <ThreePhaseSynchro name="ThreePhaseSynchro_4433" x="S1" y="S2" z="S3"
    channelWidth="3" In="Synchro_CGa_4425" />
</Signal>
```

Figure 7—Using a CG synchro (XML version)

This is one of many 1641 signals that can now make use of the synchro TSF, each using different pin names (i.e. different connections).

Synchro as a source TSF

It is possible to present the synchro TSF as a source TSF with an internal reference supply. Although this can obviously be used as a synchro signal source, it is also very useful as an input to a generic measurement TSF – more of this later.

Figure 8 shows the CG synchro with an internal reference. This is a variation of the version that is provided in the IEEE 1641 standard, with the angles expressed in degrees rather than radians, and it more closely follows the model in the Synchro_CG transformation TSF. Also the reference signal has been made available externally via the TwoWire connector.

This TSF is shown with an internal programmable supply for the reference winding. Although this is not truly representative of a normal physical synchro, it does provide a good example of how the synchro performs using programmed values rather than an input signal.

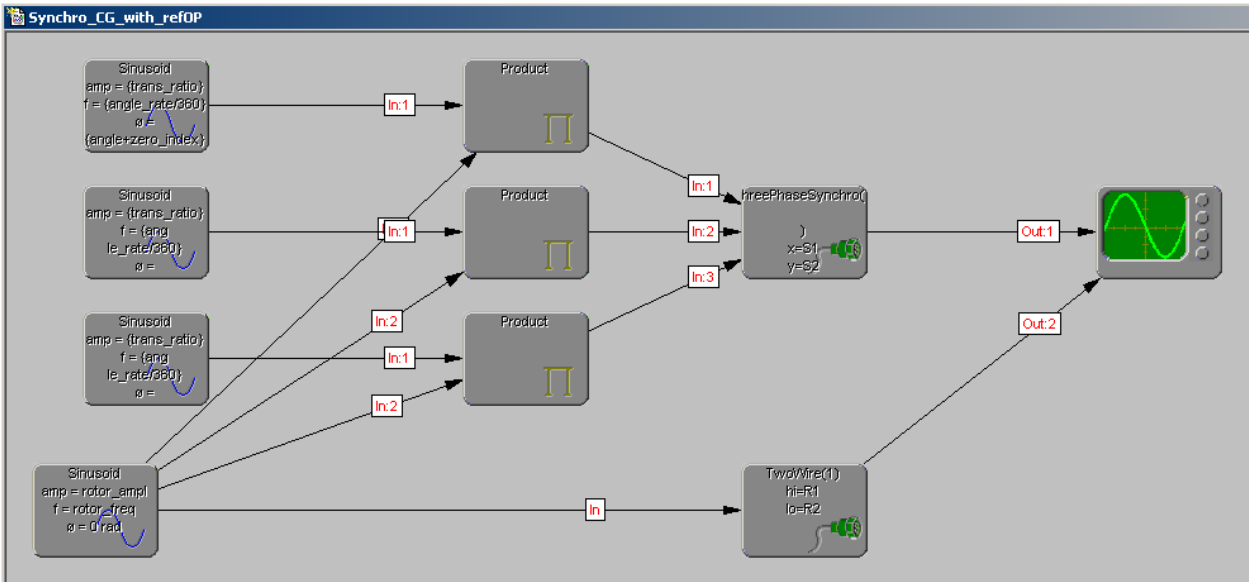


Figure 8—CG synchro with internal reference

Again, this example with built in connectors and fixed pin names is not ideal, but does show the design well. The more useful version does not include the connectors and is illustrated in Figure 9.

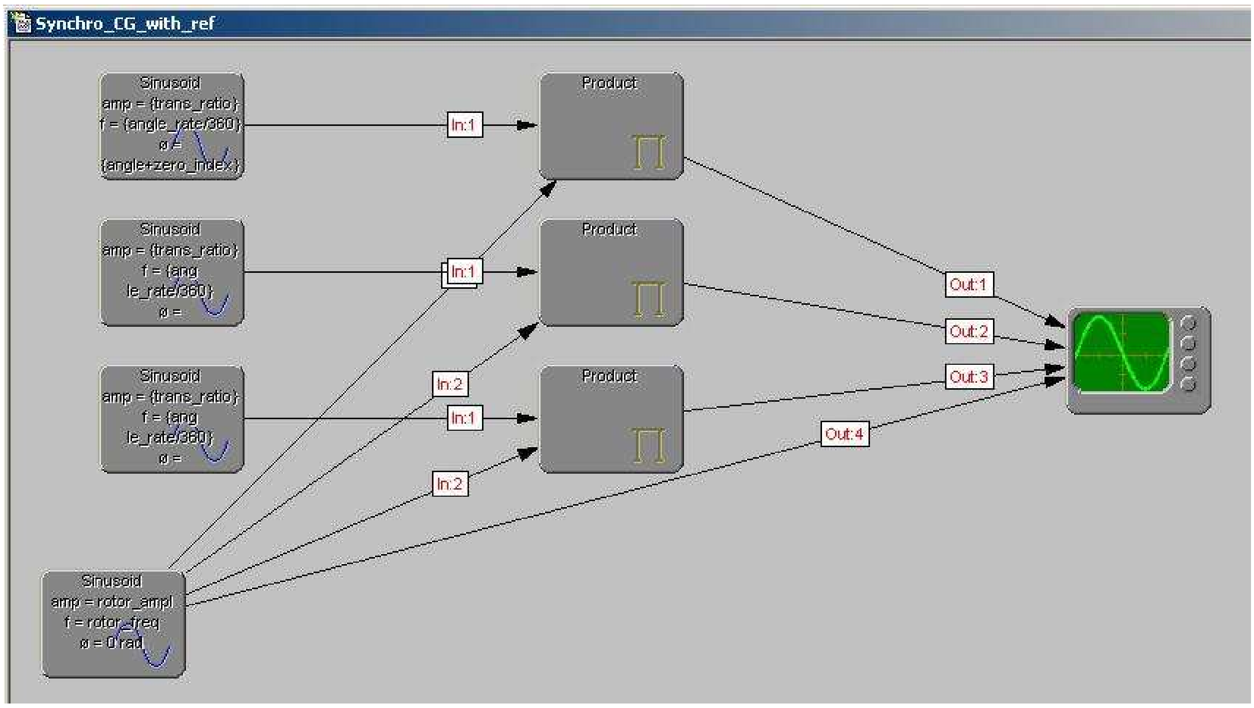
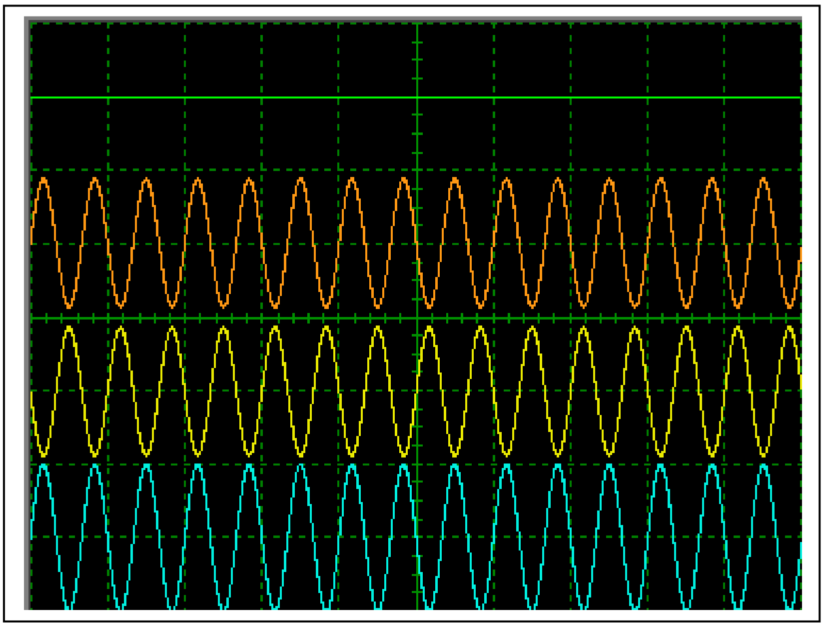


Figure 9—CG synchro with internal reference (no connectors)

The XML version of the "Synchro_CG_with_ref" model is shown in Figure 11. The interface part of the TSF is similar to the Synchro_CG model, with the attributes shaft_angle, zero_index, angle_rate, and trans_ratio, but also includes attributes for describing the reference signal; i.e. rotor_ampl and rotor_freq.

This source TSF allows a synchro signal to be fully described. The output of the TSF model may be simulated to show the exact signal defined. Figure 10 shows the simulated output for a synchro with a static rotor and the rotor in the null position. The reference (rotor signal) is shown in blue. The rms values of the other three signals indicate the rotor position. As the signals are unchanging, the rotor is stationary.



Attribute values used

shaft_angle = 0 deg (initial)
zero_index = 0 deg
angle_rate = 0 deg/s
trans_ratio = 1.0
rotor_ampl = 1 V
rotor_freq = 15 Hz

Trace information

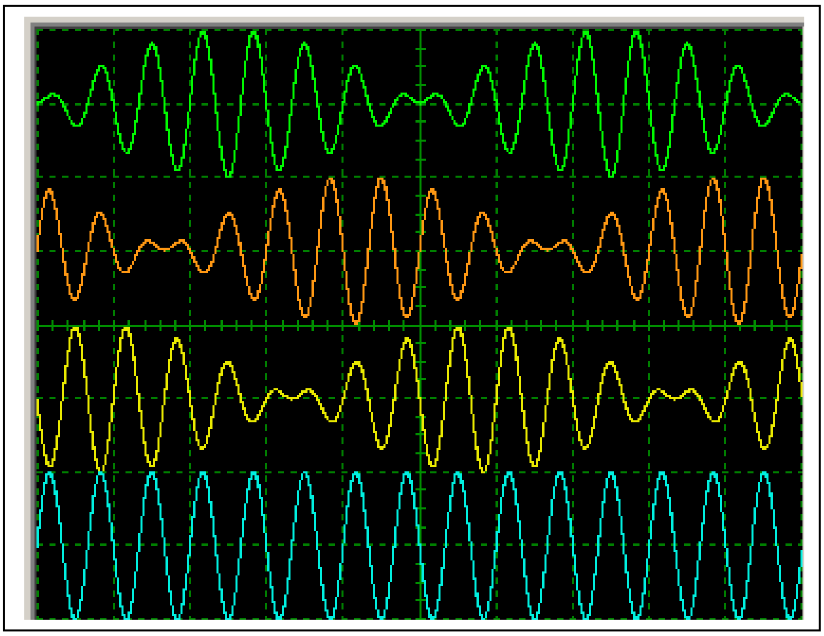
Green = S1 output
Orange = S2 output
Yellow = S3 output
Blue = R1 (R2) signal

Figure 10—Simulated output from synchro with stationary rotor

```xml
<?xml version="1.0" ?>
<!--  generated with newWaveX v4.0.0 (http://www.newWaveX.com)   -->
<tsf:TSF name="Synchro_CG_with_ref" uuid="{4FF3D801-908A-418D-A3FF-7CE3F6463BDF}"
     xmlns:tsf="urn:IEEE-1641:2010:STDTSF" xmlns:std="urn:IEEE-1641:2010:STDBSC"
     xmlns="urn:IEEE-1641:2010:STDBSC">
  <tsf:interface>
    <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
      <xs:element name="Synchro_CG_with_ref">
        <xs:annotation>
          <xs:documentation>Transmitter type Synchro with internal reference signal</xs:documentation>
        </xs:annotation>
        <xs:complexType>
          <xs:complexContent>
            <xs:extension base="SignalFunctionType">
              <xs:attribute name="shaft_angle" type="PlaneAngle" default="0 deg">
                <xs:annotation>
                  <xs:documentation>Angle of shaft from the null position</xs:documentation>
                </xs:annotation>
             </xs:attribute>
             <xs:attribute name="zero_index" type="PlaneAngle" default="0 deg">
                <xs:annotation>
                  <xs:documentation>Shaft angle offset</xs:documentation>
                </xs:annotation>
              </xs:attribute>
              <xs:attribute name="angle_rate" type="Frequency" default="0 Hz">
                <xs:annotation>
                  <xs:documentation>Speed of rotation of shaft (if moving) in deg/s</xs:documentation>
                </xs:annotation>
              </xs:attribute>
              <xs:attribute name="trans_ratio" type="double" default="1.0">
                <xs:annotation>
                  <xs:documentation>Transformer Ratio (usually in range of 0.1 to 1.0)</xs:documentation>
                </xs:annotation>
              </xs:attribute>
              <xs:attribute name="rotor_ampl" type="Voltage" default="1 V">
                <xs:annotation>
                  <xs:documentation>Reference (rotor) signal voltage</xs:documentation>
                </xs:annotation>
              </xs:attribute>
              <xs:attribute name="rotor_freq" type="Frequency" default="60 Hz">
                <xs:annotation>
                  <xs:documentation>Reference (rotor) signal frequency</xs:documentation>
                </xs:annotation>
              </xs:attribute>
            </xs:extension>
          </xs:complexContent>
        </xs:complexType>
      </xs:element>
    </xs:schema>
  </tsf:interface>
  <tsf:model>
    <Signal Out="Output_S1 Output_S2 Output_S3 Rotor_Ref">
      <Sinusoid name="Stator_winding_1" amplitude="{trans_ratio}" frequency="{angle_rate/360}"
         phase="{angle+zero_index}" />
      <Sinusoid name="Rotor_Ref" amplitude="rotor_ampl" frequency="rotor_freq" />
      <Sinusoid name="Stator_winding_2" amplitude="{trans_ratio}" frequency="{angle_rate/360}"
         phase="{angle+120+zero_index}" />
      <Sinusoid name="Stator_winding_3" amplitude="{trans_ratio}" frequency="{angle_rate/360}"
         phase="{angle+240+zero_index}" />
      <Product name="Output_S1" In="Stator_winding_1 Rotor_Ref" />
      <Product name="Output_S2" In="Stator_winding_2 Rotor_Ref" />
      <Product name="Output_S3" In="Stator_winding_3 Rotor_Ref" />
    </Signal>
  </tsf:model>
</tsf:TSF>
```

Figure 11—XML model of synchro TSF with internal reference

If the rotor is moving the output voltages would be changing with time. This can also be simulated as shown in Figure 12. The reference signal (the blue trace) is, of course, unchanging.



Attribute values used

shaft_angle = 0 deg
zero_index = 0 deg
angle_rate = 360 deg/s
trans_ratio = 1.0
rotor_ampl = 1 V
rotor_freq = 15 Hz

Trace information

Green = S1 output
Orange = S2 output
Yellow = S3 output
Blue = R1 (R2) signal

Figure 12—Simulated output from synchro with moving rotor

The receiver synchro

The receiver synchro (CR) is electrically the same as the CG synchro but operates in the opposite sense; i.e. the electrical signals are applied to the synchro and the shaft is rotated to the appropriate position. This is the synchro function most likely to be considered as a "measurement" in terms of testing; i.e. the electrical signals are applied and the "measurement" is the resultant angular position of the rotor. This is where the Measure BSC and the *As* input are used. The generic measurement facility allows any attribute of an incoming signal to be measured by referencing it to another (the expected or reference) signal. This is illustrated in Figure 13 where the input signal is passed to a Measure BSC with the reference signal connected to the *As* input.

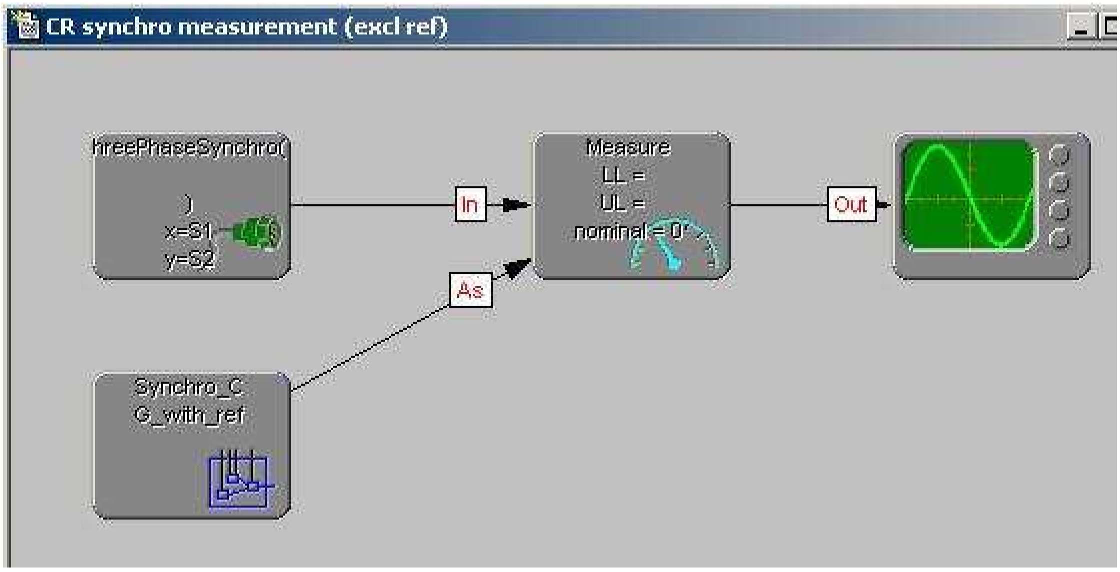


Figure 13—Measurement of a synchro signal

This example only shows the principle of the generic measurement as it is only using the Stator connections. The reference signal is defined with all its relevant attributes and describes the signal that is expected on the input connector. The measurement attribute can be any of the attributes of the reference signal. That is, the input signal is measured ***as if*** it is the reference signal.

Synchro measurement using synchro source TSF

The full solution requires all the stator and rotor terminals to be connected. This measurement TSF includes two connectors; one for the three phase stator connections and the other for the rotor connections. The measurement TSF is illustrated in Figure 14 and follows the same pattern as the simple example. The interface information includes all the attributes required to define the expected or reference signal, i.e. the attributes relevant to the synchro source TSF. It also has connection attributes to allow the measurement to be used on the actual UUT connection pins; although connection BSCs are included in the model, the pin identifiers are provided as variables (attributes) allowing the UUT pins to be specified (see the interface section in Figure 15). The measured attribute may be any of the attributes of the incoming signal, including the rotor signal amplitude and frequency.

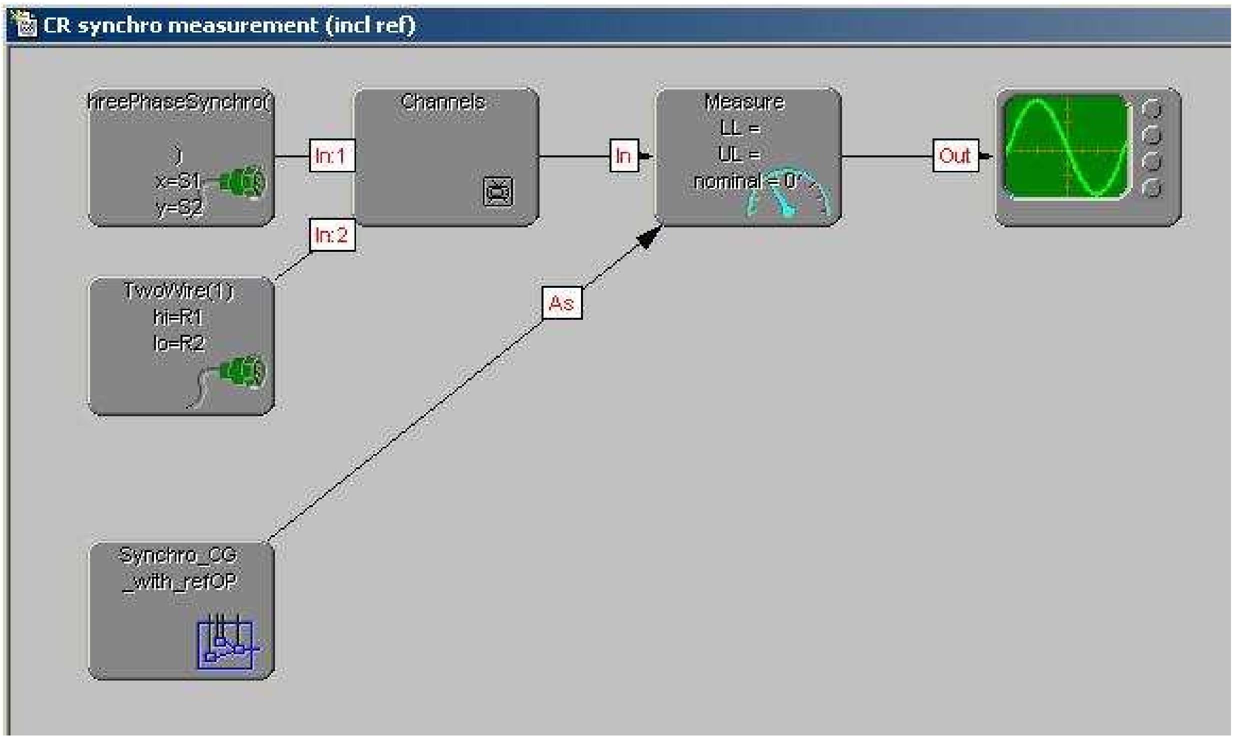


Figure 14—Measurement of a complete synchro signal

In this model, the outputs from the two connection BSCs have to be combined before they can be passed to the measure BSC input. To do this a Channels BSC is used. This combines the signals but preserves the four separate channels (S1, S2, S3, & R1/R2).

The XML for this measurement TSF is shown in Figure 15. The interface section shows all the attributes defining the expected signal but the values assigned to each are not always significant – they are there just to ensure that the expected signal is adequately described rather than *accurately* described. (There is an exception to this rule when no measurement attribute is defined but this is outside of the scope of this application note).

The attribute being measured is specified model section as one of the Measure BSC attributes (see <Measure name="**Measure__4238**" . . . />). In this example attribute="**shaft_angle**" specifies that the shaft angle is being measured.

```xml
<?xml version="1.0" ?>
<!--  generated with newWaveX v4.0.0 (http://www.newWaveX.com)  -->
<tsf:TSF name="CR synchro measurement" uuid="{34E95189-7B62-4A76-BC11-EB557252E54E}"
      xmlns:tsf="urn:IEEE-1641:2010:STDTSF" xmlns:std="urn:IEEE-1641:2010:STDBSC"
      xmlns="urn:IEEE-1641:2010:STDBSC">
  <tsf:interface>
    <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
      <xs:element name="CR synchro measurement">
        <xs:annotation>
          <xs:documentation>CR synchro measurement</xs:documentation>
        </xs:annotation>
        <xs:complexType>
          <xs:complexContent>
            <xs:extension base="SignalFunctionType">
              <xs:attribute name="shaft_angle" type="PlaneAngle" default="0 deg">
                <xs:annotation>
                  <xs:documentation>Expected shaft offset</xs:documentation>
                </xs:annotation>
              </xs:attribute>
              <xs:attribute name="zero_index" type="PlaneAngle" default="0 deg">
                <xs:annotation>
                  <xs:documentation>Expected shaft offset</xs:documentation>
                </xs:annotation>
              </xs:attribute>
              <xs:attribute name="angle_rate" type="Frequency" default="0 Hz">
                <xs:annotation>
                  <xs:documentation>Expected shaft rotation speed (in deg/s)</xs:documentation>
                </xs:annotation>
              </xs:attribute>
              <xs:attribute name="trans_ratio" type="double" default="1.0">
                <xs:annotation>
                  <xs:documentation>Expected transformer ratio</xs:documentation>
                </xs:annotation>
              </xs:attribute>
              <xs:attribute name="rotor_ampl" type="Voltage" default="115 V">
                <xs:annotation>
                  <xs:documentation>Expected Rotor voltage</xs:documentation>
                </xs:annotation>
              </xs:attribute>
              <xs:attribute name="rotor_freq" type="Frequency" default="400 Hz">
                <xs:annotation>
                  <xs:documentation>Expected Rotor frequency</xs:documentation>
                </xs:annotation>
              </xs:attribute>
              <xs:attribute name="S1" type="string">
                <xs:annotation>
                  <xs:documentation>UUT Stator connection S1</xs:documentation>
                </xs:annotation>
              </xs:attribute>
              <xs:attribute name="S2" type="string">
                <xs:annotation>
                  <xs:documentation>UUT Stator connection S2</xs:documentation>
                </xs:annotation>
              </xs:attribute>
              <xs:attribute name="S3" type="string">
                <xs:annotation>
                  <xs:documentation>UUT Stator connection S3</xs:documentation>
                </xs:annotation>
              </xs:attribute>
              <xs:attribute name="R1" type="string">
                <xs:annotation>
                  <xs:documentation>UUT Rotor connection R1</xs:documentation>
                </xs:annotation>
              </xs:attribute>
              <xs:attribute name="R2" type="string">
                <xs:annotation>
                  <xs:documentation>UUT Rotor connection R2</xs:documentation>
                </xs:annotation>
              </xs:attribute>
            </xs:extension>
          </xs:complexContent>
        </xs:complexType>
      </xs:element>
    </xs:schema>
  </tsf:interface>
```

```
  <tsf:model>
    <Signal Out="Measure__4238" xmlns:this="MyTSFLib"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:schemaLocation="MyTSFLib MyTSFLib.xsd">
      <ThreePhaseSynchro name="ThreePhaseSynchro__4242" x="S1" y="S2" z="S3" channelWidth="3" />
      <TwoWire name="TwoWire__4243" lo="R2" hi="R1" channelWidth="1" />
      <Channels name="Channels__4244" In="ThreePhaseSynchro__4242 TwoWire__4243" />
      <this:Synchro_CG_with_ref name="Synchro_CG_with_ref__4434" shaft_angle="0 deg" />
      <Measure name="Measure__4238" As="Synchro_CG_with_ref__4434" attribute="shaft_angle"
          In="Channels__4244" />
    </Signal>
  </tsf:model>
</tsf:TSF>
```

Figure 15—XML version of the measurement TSF

Using this TSF to make a measurement is simple (see Figure 16). The measurement attribute is set in the TSF (for example using a drop-down box as shown in Figure 17 – this saves editing the XML directly).

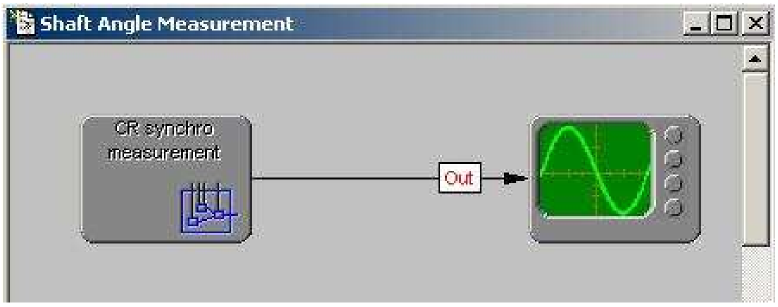


Figure 16—Using the measurement TSF

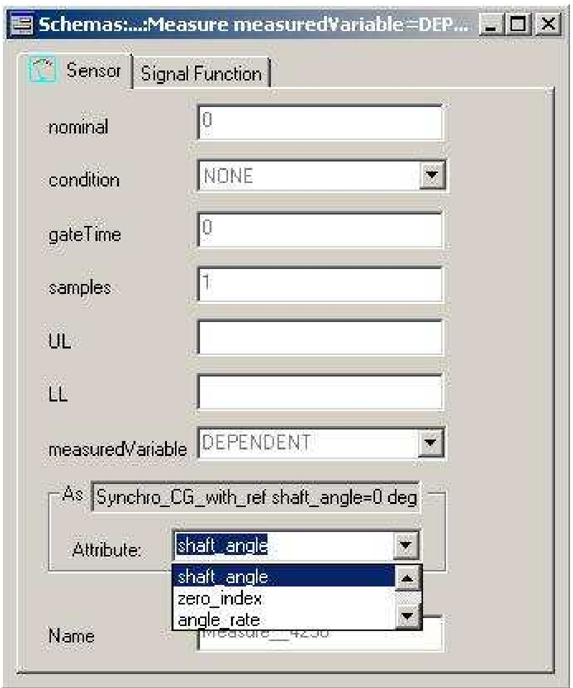


Figure 17—Defining the measurement attribute

Then the attributes for the *As* reference signal and the connection pins are set up in the measurement signal. This can be done directly in XML, or more simply using an interface GUI as shown in Figure 18. Attributes not specifically entered will take the default.

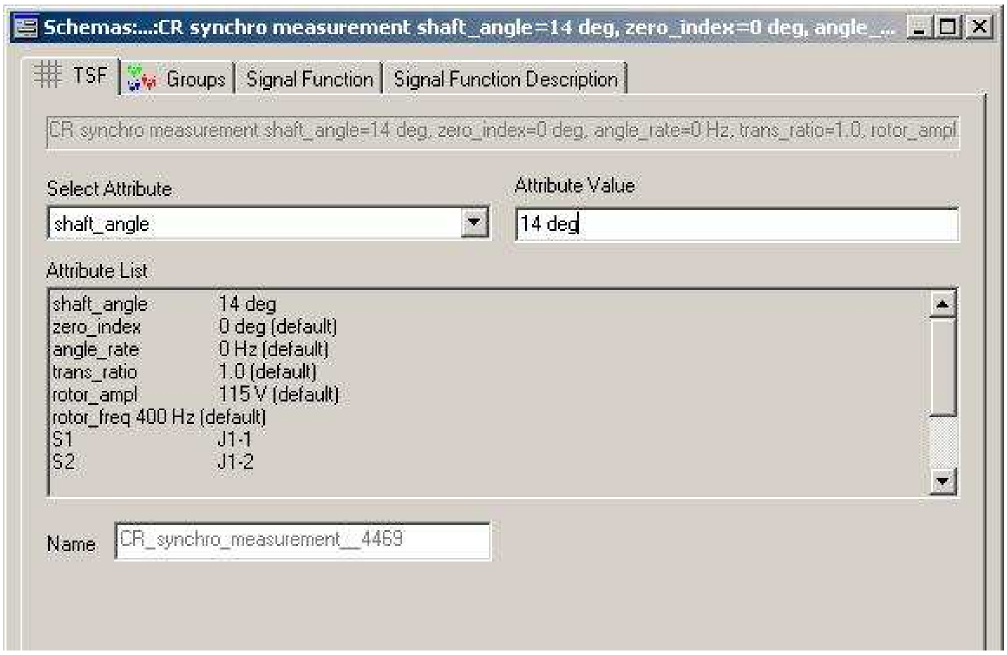


Figure 18—Specifying the measurement attributes

The resultant measurement XML is shown in Figure 19.

```
<?xml version="1.0" ?>
<!--  generated with newWaveX v4.0.0 (http://www.newWaveX.com)   -->
<Signal Out="CR_synchro_measurement__4469" xmlns="urn:IEEE-1641:2010:STDBSC"
    xmlns:this="MyTSFLib" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="MyTSFLib MyTSFLib.xsd">
  <this:CR_synchro_measurement name="CR_synchro_measurement__4469"
       shaft_angle="14 deg" S1="J1-1" S2="J1-2" S3="J1-3" R1="J2-7" R2="J2-9" />
</Signal>
```

Figure 19—XML version of the actual measurement (using a TSF)

The attribute values are used in this measurement are shown below. The specified (non-default) values may be identified in the measurement XML (see Figure 19) and the default values are taken from the measurement TSF interface section (see Figure 15).

Attribute values used

shaft_angle:	14 deg	(expected value)
zero_index:	0 deg	(default)
angle_rate:	0 deg/s	(default)
trans_ratio:	1.0	(default)
rotor_ampl:	115 V	(default)
rotor_freq:	400 Hz	(default)

Measurement information

As signal:	Synchro_CG_with_ref__4434
Measured attribute:	shaft_angle
Input signal connections:	
Stator:	J1-1, J1-2, J1-3
Rotor:	J2-7, J2-8

Of course, it is not essential to use the measurement TSF. It is possible to perform the measurement directly using a measurement *signal* rather than a *TSF*, although this still uses the reference signal TSF. This is more flexible as it allows any attribute to be measured – the TSF method requires a different TSF for each different attribute measurement. However, the complete *As* signal has to be defined every time unless the defaults are used – with the TSF method, the *As* signal only has to be defined once. The measurement (in XML) would appear as shown in Figure 20.

```
<?xml version="1.0" ?>
<!--  generated with newWaveX v4.0.0 (http://www.newWaveX.com)   -->
<Signal Out="Measure__4570" xmlns="urn:IEEE-1641:2010:STDBSC"
    xmlns:this="MyTSFLib" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="MyTSFLib MyTSFLib.xsd">
  <ThreePhaseSynchro name="ThreePhaseSynchro__4561" x="13" y="14" z="15"
       channelWidth="3" />
  <TwoWire name="TwoWire__4563" lo="11" hi="12" channelWidth="1" />
  <Channels name="Channels__4562" In="ThreePhaseSynchro__4561 TwoWire__4563" />
  <this:Synchro_CG_with_ref name="Synchro_CG_with_ref__4564" shaft_angle="0 deg"
       zero_index="0 deg" angle_rate="0 deg/s" rotor_ampl="115 V" rotor_freq="60 Hz" />
  <Measure name="Measure__4570" As="Synchro_CG_with_ref__4564"
attribute="shaft_angle" In="Channels__4562" />
</Signal>
```

Figure 20—XML version of the actual measurement (using a signal)

Important

- these measurement TSFs only obtain a single measurement value (samples = 1)
- make no judgement as to whether the result is a pass or fail. The limit fields (UL and LL) have been left blank
- the values have all been specified without a qualifier, which means that all values have been specified as instantaneous values. To specify rms values, the trms qualifier will need to be added (synchros normally use rms values), e.g., the XML describing the rotor voltage should actually be: rotor_ampl="**trms 115 V**"
- the examples shown above have included most if not all of the parameters relevant to the use of synchros. It is always possible to create simple TSFs that only include the attributes required for a specific task.

Resolvers

A resolver is very similar to a CG synchro, with the difference that the output signal comes from two stator windings displaced at 90°, whereas the synchro has three windings displaced at 120°. Figure 21 shows a typical resolver. This may also be modelled using the same techniques that were used with the CG synchro, i.e. either as a transformation TSF or a source TSF with an internal reference. The approach taken depends on the requirement of the tests being described. Note that the stator windings are completely separate in a resolver.

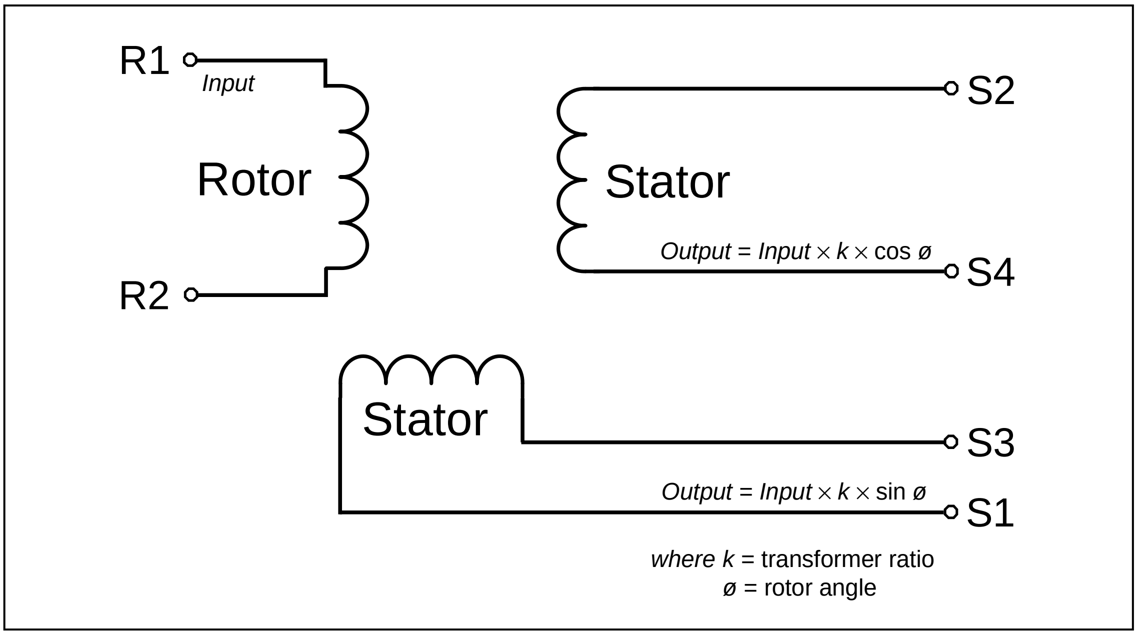


Figure 21—Basic resolver schematic

<u>The resolver as a transformation TSF</u>

Figure 22 shows a graphical view of a transformation TSF for a resolver and Figure 23 shows the more useful version without fixed connections. This allows an input reference rotor signal to be applied to the TSF. The output signals on S1-S3 and S2-S4 give the position of the rotor via their proportional values. The shaft information (its position and whether it is rotating or not) is provided as attributes.

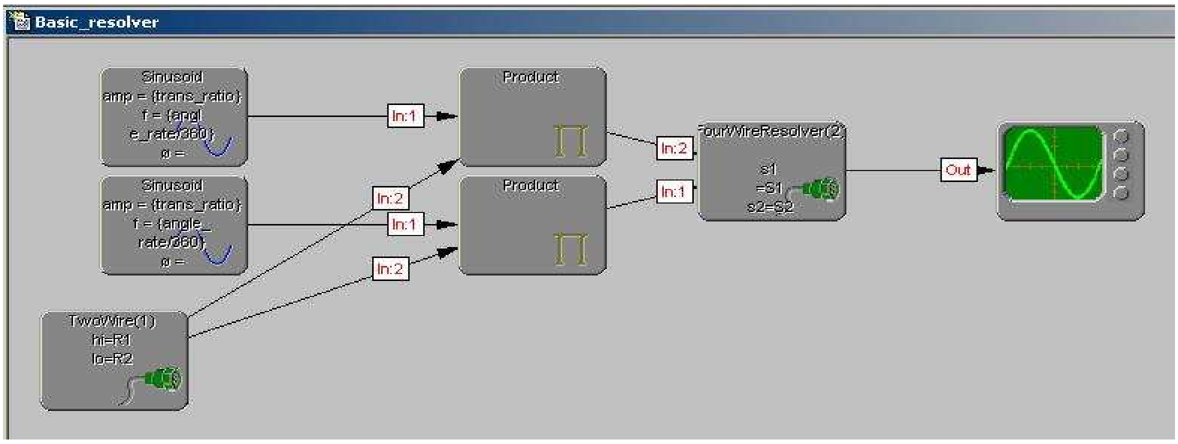


Figure 22—Resolver TSF with fixed connections

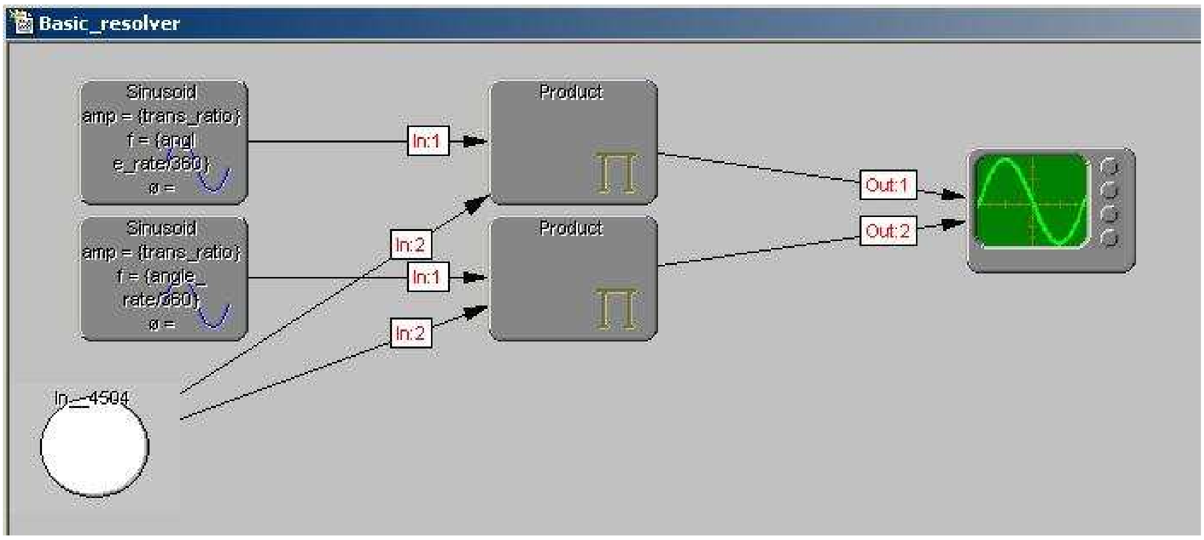


Figure 23—Resolver TSF without fixed connections

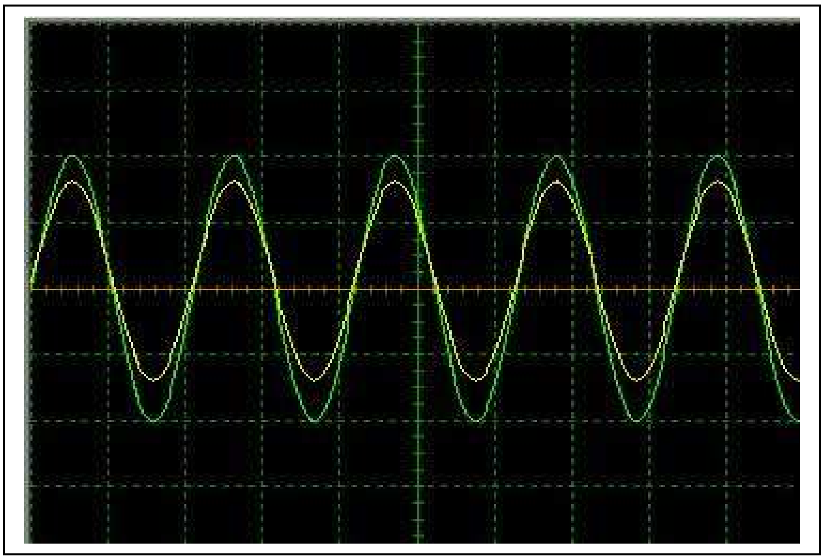
Figure 24 gives the main parts of the XML version of the TSF. Due to the similarity of the TSF to the synchro version, the less important commentary parts of the TSF have been stripped out. The attributes are the same as those for the synchro TSF illustrated in Figure 3.

```
<?xml version="1.0" ?>
<!--  generated with newWaveX v4.0.0 (http://www.newWaveX.com)  -->
<tsf:TSF name="Basic_resolver" uuid="{CD392CA8-4BA7-4F4F-BCDD-52CEC2A608A1}"
     xmlns:tsf="urn:IEEE-1641:2010:STDTSF" xmlns:std="urn:IEEE-1641:2010:STDBSC"
     xmlns="urn:IEEE-1641:2010:STDBSC">
  <tsf:interface>
    <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
      <xs:element name="Basic_resolver">
        <xs:complexType>
          <xs:complexContent>
            <xs:extension base="SignalFunctionType">
              <xs:attribute name="shaft_angle" type="PlaneAngle" default="0 deg">
              </xs:attribute>
              <xs:attribute name="zero_index" type="PlaneAngle" default="0 deg">
              </xs:attribute>
              <xs:attribute name="angle_rate" type="Frequency" default="0 Hz">
              </xs:attribute>
              <xs:attribute name="trans_ratio" type="double" default="1.0">
              </xs:attribute>
            </xs:extension>
          </xs:complexContent>
        </xs:complexType>
      </xs:element>
    </xs:schema>
  </tsf:interface>
- <tsf:model>
    <Signal Out="Signal_S24 Signal_S_13" In="In_4504">
      <Sinusoid name="Sinusoid_4350" amplitude="{trans_ratio}" frequency="{angle_rate/360}"
          phase="{shaft_angle+zero_index}" />
      <Sinusoid name="Sinusoid_4354" amplitude="{trans_ratio}" frequency="{angle_rate/360}"
          phase="{shaft_angle+90+zero_index}" />
      <Product name="Signal_S24" In="Sinusoid_4350 In_4504" />
      <Product name="Signal_S13" In="Sinusoid_4354 In_4504" />
    </Signal>
  </tsf:model>
</tsf:TSF>
```

Figure 24—Resolver TSF XML

This TSF may be used in a very similar way to the CG_Synchro TSF (see *Using the TSF*). Therefore this is not described in detail here.

The following two figures show the simulated output of the TSF for different parameter values. Figure 25 shows a stationary resolver with the shaft angle at zero. The green trace shows the input signal; 10 V ac at 60 Hz. The yellow trace shows the output (S2 & S4 in Figure 21) with a transformer ratio of 0.75, i.e. ¾ of the amplitude of the input signal. The orange trace shows the quadrature signal and remains at 0 V in this example. Note that the shaft position is indicated by the ratio of the two output voltages; i.e. $ø = \tan^{-1} (V_{S2S4}/V_{S1S3})$, where ø is the shaft angle, V_{s2s4} is the voltage from the in-phase winding and V_{S1S3} is the voltage from the quadrature winding.



Attribute values used

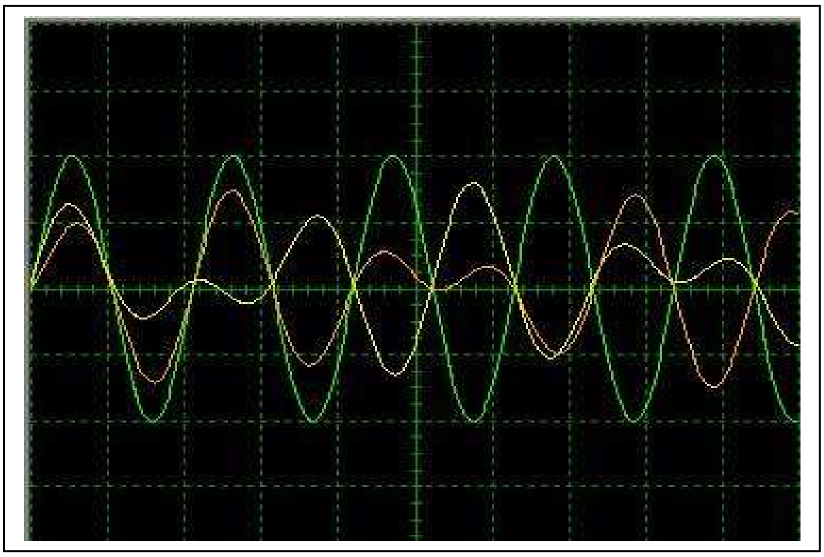
shaft_angle = 0 deg
zero_index = 0 deg
angle_rate = 0 deg/s
trans_ratio = 0.75
rotor_ampl = 10 V
rotor_freq = 60 Hz

Trace information

Green = R1 (R2) signal
Yellow = S2 (S4) output
Orange = S1 (S3) output

Figure 25—Simulated output from resolver with static rotor

Figure 26 shows a resolver with the shaft rotating at 3600 deg/s. This figure, which represents 10 revolutions per second, has been chosen so that the changes in the waveforms show in the simulated 'scope screen. The green trace shows the input signal; 10 V ac at 60 Hz. The yellow trace shows the output (S2 & S4 in Figure 21) with a transformer ratio of 0.75, i.e. ¾ of the amplitude of the input signal. The orange trace shows the quadrature signal. The waveform shows the initial starting point of 25 deg, but the change in the voltage ratio may easily be seen on the trace as the shaft moves away from the 25 deg position.



Attribute values used

shaft_angle = 25 deg (initial)
zero_index = 0 deg
angle_rate = 3600 deg/s
trans_ratio = 0.75
rotor_ampl = 10 V
rotor_freq = 60 Hz

Trace information

Green = R1 (R2) signal
Yellow = S2 (S4) output
Orange = S1 (S3) output

Figure 26—Simulated output from resolver with moving rotor

The resolver as a source TSF

It is also possible to present the resolver TSF as a source TSF with an internal reference supply, in a very similar way that was done with the synchro. This makes more sense with a resolver as the phase of the reference signal is often not important because the shaft position is determined by the ratio of the outputs and not their phase relationship. Figure 27 shows the resolver with an internal reference.

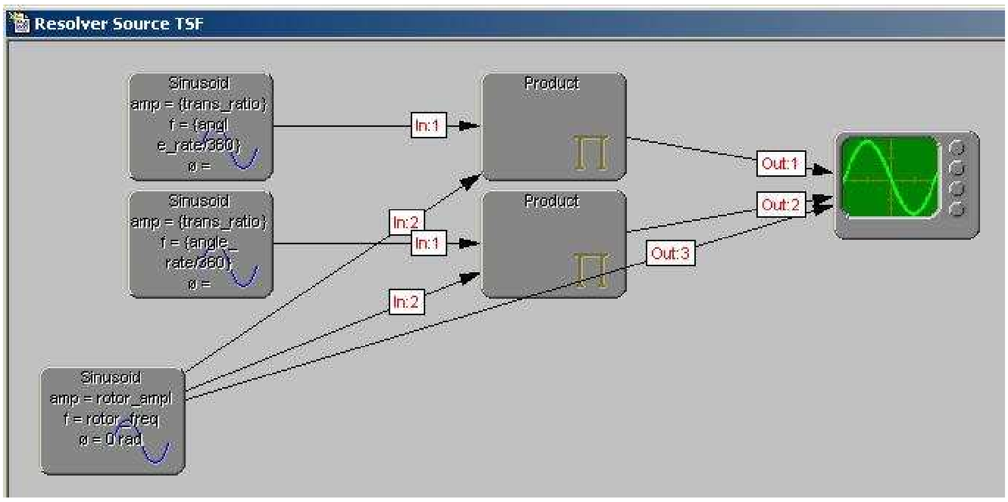


Figure 27—Resolver with internal reference

As this TSF is very similar to the synchro version there is little to be gained by including the XML, which may be easily generated manually or automatically. The use of this TSF in a generic measurement is also very similar. In addition to the basic CS resolver shown above, there are also receiver resolvers (that work in the opposite sense to the CS resolver) and differential resolvers.

Intrinsic Measurement

This application note has focused on representing the operation of synchros and resolvers using IEEE 1641 techniques, and the use of generic measurement. This is because in most test scenarios, synchro/resolver specific instruments are used. The specification of the test is still not instrument specific or even instrument type specific, but does convert easily to real measurement situations. It is, of course, possible to perform measurements as intrinsic measurements. Rather than using a reference signal, and the *As* input to a measurement BSC, it is possible to use RMS BSCs and calculate the result from those measurements. For example, Figure 28 represents such a measurement.

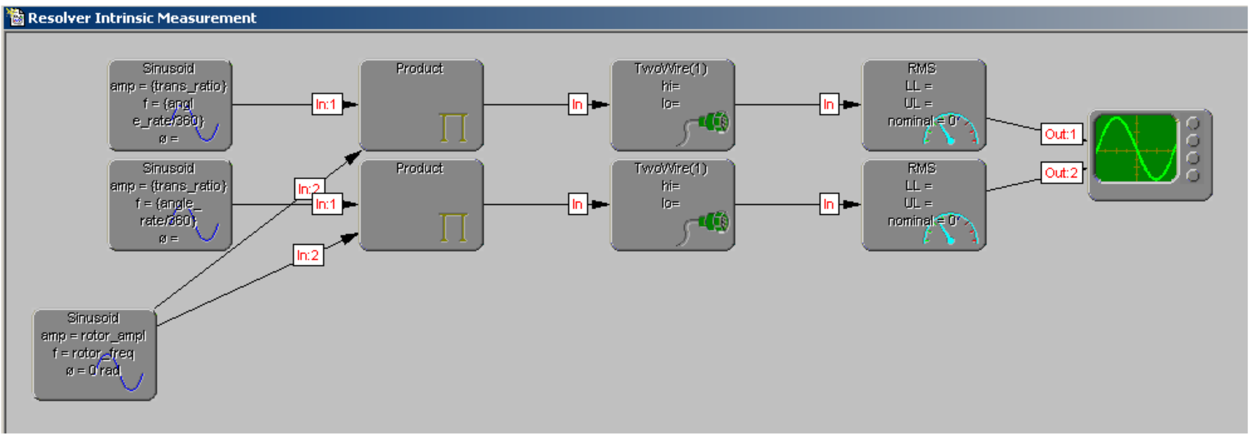
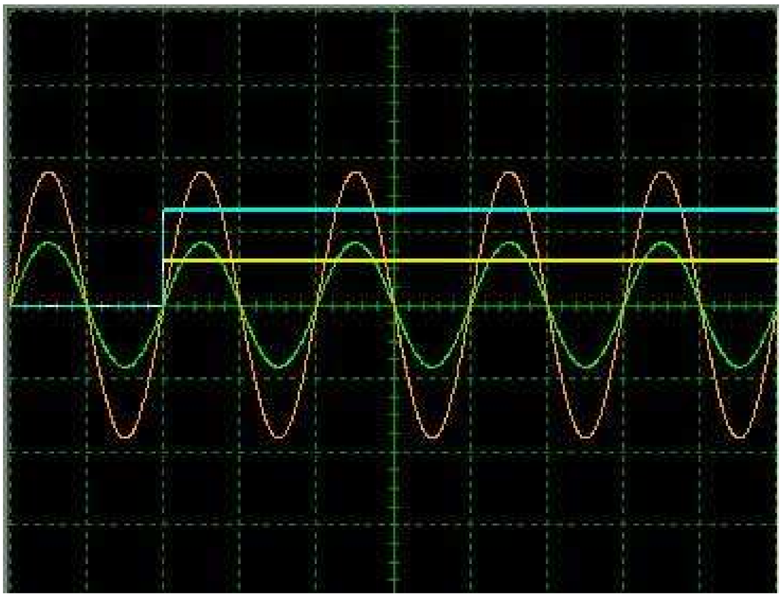


Figure 28—Representation of an intrinsic measurement

Figure 29 provides an illustration of the two RMS measurements. From the parameters used for this example (also shown in Figure 29) and the output simulation scope trace, it is possible to do an approximate estimate of the shaft angle measurement. The scope trace represents 2 V per division, and this gives approximately 1.2 V for the S2 (S4) voltage and 2.6 V for the S1 (S3) voltage. The voltage ratio ($V_{S2(S4)}$ / $V_{S2(S4)}$) gives a value of 0.46. Calculating $\tan^{-1}$ (0.46) gives a value of 24.7 deg, which is close to the 25 deg value used.



Attribute values used

```
shaft_angle =  25 deg
zero_index  =  0 deg
angle_rate  =  0 deg/s
trans_ratio =  1.0
rotor_ampl  =  10 V
rotor_freq  =  25 Hz
```

Trace information

```
Green    =  S2 (S4) signal
Orange   =  S1 (S3) output
Yellow   =  S2 (S4) rms value
Blue     =  S1 (S3) rms value
```

Figure 29—Intrinsic rms measurements

The intrinsic measurement method is equally valid, and may also be represented in IEEE 1641. Ultimately, the programmer should use the method most appropriate to his application.

Conclusions

It is not important which methods are used to express the signals and measurements required; source or transformation TSFs, generic or intrinsic measurements, or how the connections are identified. The test engineer may choose an approach to suit his application, but with IEEE 1641 information essential to the successful testing of the UUT will not be disguised or omitted, and it will not rely on the instrumentation or architecture of the original target test system. The signal *requirements* are clearly defined and these are always portable to another system as circumstances require.