

Contents

Introduction.....	2
Learning Outcome.....	3
Nomenclature.....	4
C47/R47 ELEC Menu content.....	5
Introducing Triple Registers.....	7
Triple Register Mapping.....	7
Phasor Representation of Three-Phase Quantities.....	8
Introducing Star Delta Transforms in Registers.....	9
Example 1: Star Delta Transform, with 3-phase Ohm's law, using triple registers	10
Example 2: Star Delta Transforms using triple registers	11
Introducing sequence component conversions without matrices.....	12
Example 3: Symmetrical Components without matrices	12
Symmetrical-Component Transformation Matrix Definition	13
Symmetrical-Component Matrix Calculations.....	13
Example 4: Symmetrical Components with matrices	14
Example 5: Symmetrical Components without matrices:.....	14
Version history:.....	15

Introduction

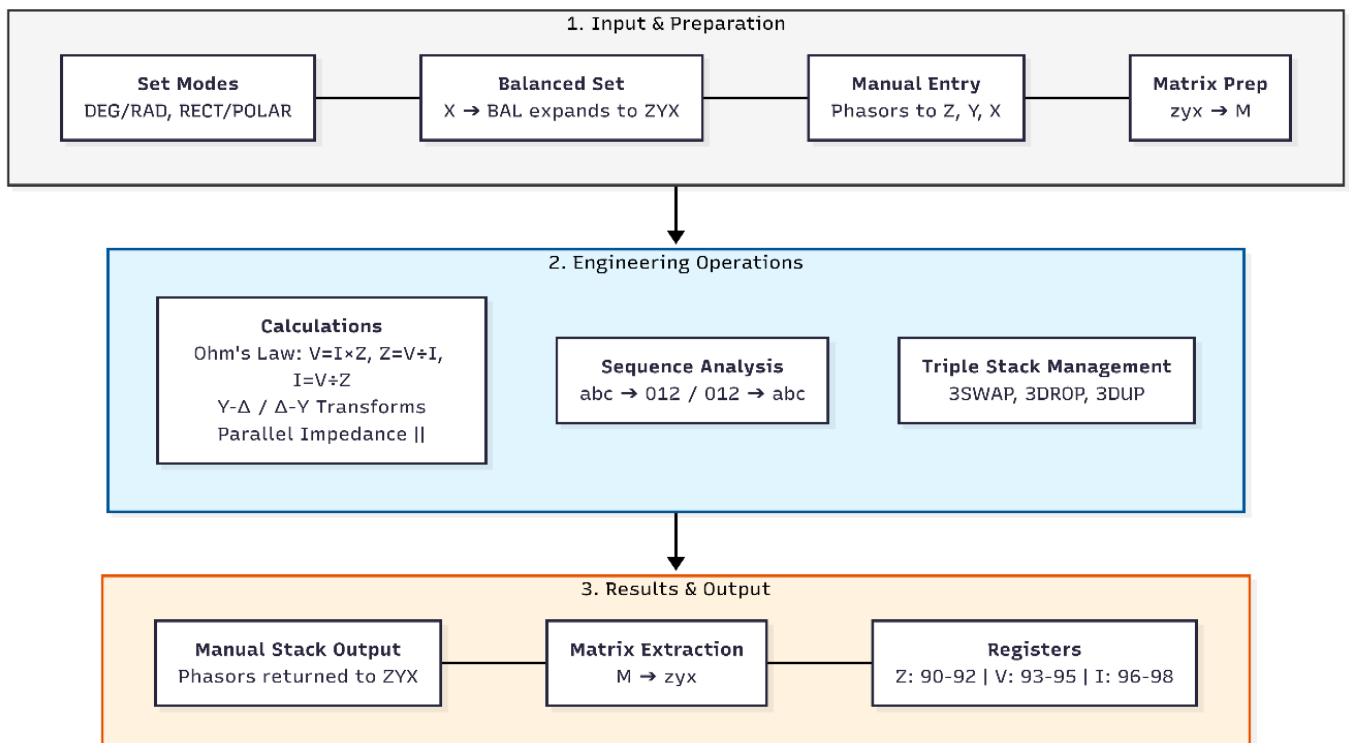
This application note illustrates three-phase power engineering calculations. The tutorial is presented both without and with matrices. Calculations are performed directly, without use of the customary per-unit system.

Working with base units (V, Z, I) in individual registers allows one to reason through problems using actual values, rather than relying on procedural abstractions like matrix methods or per-unit notation, which can obscure the real magnitudes and their relative sizes.

The ELEC menu is a dedicated application module for electrical engineering calculations. It extends standard complex arithmetic by introducing triple registers that represent three-phase quantities as a single grouped entity, allowing direct manipulation of both unbalanced and balanced systems.

Engineering workflows for impedance, voltage, and current calculations are emphasised without requiring explicit matrix manipulation, while symmetrical-component matrix operations are also demonstrated. Complex Ohm's law, star-delta and delta-star transformations, sequence component conversions, and parallel impedance calculations are all supported.

The following block diagram provides an overview of the ELEC menu and its functionality, giving the reader a visual map of the workflow and how the various calculation tools are organized, before diving into individual calculations.



Learning Outcome

After completing this application note, the reader will be able to use the ELEC menu as an RPN-based application module for electrical engineering calculations involving three-phase systems.

The reader will also understand how triple complex registers are used to group and manipulate impedance, voltage, and current quantities with or without reliance on matrix operations. The reader will be able to perform:

- Complex Ohm's law calculations
- Complex star-delta and delta-star conversions
- Transformations between three-phase quantities and symmetrical components
- Calculations done with rectangular and polar input and display
- Analysis of balanced and unbalanced systems using direct register-based workflows aligned with classical electrical engineering formulations

The examples in this application note and the ELEC power-engineering tools require use of basic complex-number input and output, complex arithmetic, and matrix manipulation on the C47/R47. These fundamental skills emerge as a useful side effect of this application note, highlighting the practical value of the exercises.

Nomenclature

Star (Y) (Wye)	Three-phase connection in which all three phase windings have either their start terminals or their end terminals connected to a common point, forming a star shape. This point is the neutral, which may or may not be accessible to the user. 'Star' is also called a 'wye' configuration in various parts of the world, which is why we use Y as a symbol for the star configuration.
Delta (Δ)	Three-phase connection in which the terminals of the phase windings are connected start-to-end, forming a closed triangle.
Sequence components	This is the method originally proposed by Fortescue (1918) ¹ and summarized in 100 Years of Symmetrical Components ² .
V_1 Positive sequence V_2 Negative sequence V_0 Zero sequence	Three-phase <u>phasor</u> , transformed into <u>sequence</u> components.
V, Z, I	Voltage, impedance, current (used throughout).
a, a^2	Phase rotation operators for <u>sequence</u> components.
A, A^{-1}	Symmetrical component transformation matrices
R90–R98	Triple register range.
ZYX	Register ordering.

¹ Charles L. Fortescue, *Method of Symmetrical Co-ordinates Applied to the Solution of Polyphase Networks*, presented at the Annual Convention of the American Institute of Electrical Engineers (AIEE) in 1918

² Gianfranco Chicco and Andrea Mazza, 100 Years of Symmetrical Components

C47/R47 ELEC Menu content

C47 ELEC Menu p1:

DEG $\odot$	RAD $\circ$	zyx→M	M→zyx	RECT $\circ$	POLAR $\odot$
⇒DEG	⇒RAD	a	a ²	[A]	[A ⁻¹]
$\hat{i}$	$\hat{i} \odot$	x ²	[M] ⁻¹		CLSTK

R47 ELEC Menu p1:

DEG $\odot$	RAD $\circ$	zyx→M	M→zyx	RECT $\circ$	POLAR $\odot$
⇒DEG	⇒RAD	a	a ²	[A]	[A ⁻¹]
$\hat{i}$	$\hat{i} \odot$	COMPLEX	[M] ⁻¹		CLSTK

C47 ELEC Menu	Detail	Notes / Explanation
Angle shortcuts DEG(), RAD(), RECT(), POLAR()	Setting modes: Angular Display (ADM) or complex display	Choose operating complex display mode
Matrix shortcut [M]-1	Matrix inversion	Also [1/x] on keyboard
Other shortcuts $\hat{i}$, $\hat{i} \odot$, x ² , CLSTK, COMPLEX	Complex entry, square, clear stack	Shortcuts to other menu items
Electrical operators a, a ² , [A], [A-1]	$a = e^{(2\pi\hat{i}/3)}$, $a^2 = e^{(4\pi\hat{i}/3)}$, [A], [A-1]	Standard symmetrical-component transformation matrices
X.FN menu shortcut MULTSTK	Menu shortcut	More triple and other commands
Matrix conversions zyx→M, M→zyx	Convert complex registers z, y, x ↔ matrix [z y x] ^T	complex ↔ matrix translations

Angle shortcuts from menus: DEG(), ⇒DEG, RAD(), ⇒RAD, RECT(), POLAR()

Setting the Angular Display Mode (ADM) of DEG and RAD

Directly creating tagged degrees or radians in X

Setting the operating complex display mode RECT or POLAR

Matrix shortcut from menu: [M]⁻¹

Matrix inversion (also available as [1/x] on the keyboard)

Various shortcuts from menus: $\hat{i}$, $\hat{i} \odot$, x², CLSTK

Complex entry $\hat{i}$, $\hat{i} \odot$, square and clear stack

Electrical constants a, a², [A], [A⁻¹]

Operators $a = e^{2\pi\hat{i}/3}$, $a^2 = e^{4\pi\hat{i}/3}$ matrices [A], [A⁻¹] being the standard symmetrical-component transformation matrices

Shortcut from X.FN menu: ||

Impedances or resistances in parallel, XY/(X+Y), for real or complex quantities.

Matrix conversions: zyx→M, M→zyx

Convert complex stack registers z, y, x to and from matrix [z y x]^T

3STOV ₉₀	3STOI ₉₃	3STOZ ₉₆	abc→012	012→abc	3R↔3P
3RCLV ₉₀	3RCLI ₉₃	3RCLZ ₉₆	Δ → Y	Y → Δ	MULTSTK
V=I×Z	I=V÷Z	Z=V÷I	X→BAL	3SWAP	3DROP

C47 ELEC Menu	Detail	Notes / Explanation
Triple storage: 3STOV ₉₀ , 3STOI ₉₃ , 3STOZ ₉₆ , 3RCLV ₉₀ , 3RCLI ₉₃ , 3RCLZ ₉₆	Register triplets are used to store three-phase quantities. Registers 90-92, 93-95, 96-98 are accessed by each shortcut. 3STO and 3RCL in MULTISTK can handle any triplet in the range 00-97	Allows storing and recalling three-phase register triplets efficiently
Triple stack operations: 3SWAP, 3DROP, 3R↔3P	Triple swap exchanges ZYX with BAT; triple drop removes ZYX from stack. 3DUP in MULTISTK; triple ZYX Rect ↔ Polar	Operations on triple stack registers for manipulation
Ohm's law operations: Z=V÷I, V=I×Z, Z=V÷Z	These equations evaluate and store results into Z (R90-92), V (R93-R95), I (R96-R98)	Calculates and stores three-phase quantities
Sequence components: abc→012, 012→abc	Triple register in ZYX is used as input; results returned to ZYX	Converts between phase and sequence components
Star – Delta: Y → Δ, Δ → Y	Triple register in ZYX is used as input; results returned to ZYX.	Converts between star and delta configurations
Balanced input: X → BAL	Single register X expanded into ZYX: Z=X, Y=a·Z, X=a ² ·Z, forming a balanced set. ZYX order corresponds to Z = phase A, Y = phase B, X = phase C.	Prepares a balanced set for sequence component or star-delta operations

- Triple storage** 3STOV₉₀, 3STOI₉₃, 3STOZ₉₆, 3RCLV₉₀, 3RCLI₉₃, 3RCLZ₉₆
 Register triplets store three-phase quantities, with registers 90–92, 93–95, and 96–98 accessed by 3STO.../3RCL... shortcuts, while the 3STO and 3RCL commands in the **MULTISTK** menu can generically store or recall any triplet from 00–97.
- Triple stack operations** 3SWAP, 3DROP
 Register triple operations include 3SWAP, exchanging registers ZYX with BAT, and 3DROP, removing ZYX from the stack. 3DUP in the **MULTISTK** menu duplicates ZYX to BAT, completing the suite of triple-register operations. 3R↔3P changes ZYX to rect/polar.
- Ohm's law operations** Z=V÷I, V=I×Z, Z=V÷Z
 These equation commands evaluate the right-hand side and store the results into the V (R90-92), I (R93-R95) & R (R96-R98) register triplets.
- Sequence components** abc→012, 012→abc
 The triple register in ZYX is used as input, with results returned to ZYX. Details and formulas are given below.
- Star – Delta** Y - Δ, Δ - Y
 The triple register in ZYX is used as input, with results returned to ZYX. Details and formulas are given below.
- Balanced input** X→BAL
 A single register X is used as input and expanded into ZYX. The X input is copied to Z; a·Z to Y and a²·Z to X, forming a balanced set suitable for sequence component or star-delta operations.

Introducing Triple Registers

The ELEC menu provides triple complex registers (R90–R98) to hold complete three-phase quantities for voltage (V), current (I) and impedance (Z).

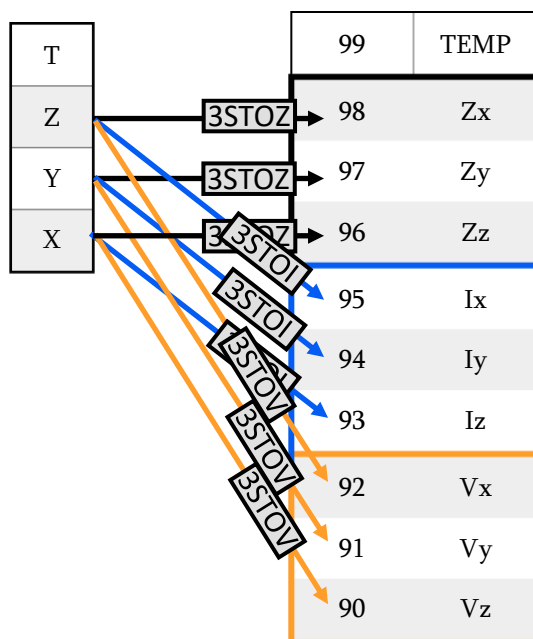
Each triple register stores all three phase quantities together, enabling direct manipulation without the need for matrix operations. This design simplifies standard calculations such as star-delta conversions, parallel impedance calculations, and sequence component transformations while keeping the workflow intuitive and aligned with classical electrical engineering practice.

Triple Register Mapping

The registers are organized as follows:

- V represents three voltages stored in R90–R92
- I represents three currents stored in R93–R95
- Z represents three impedances stored in R96–R98

These triple registers can be considered grouped registers, each representing a complete three-phase set in a single entity.



3STOV ₉₀	3STOI ₉₃	3STOZ ₉₆	abc→012	012→abc	3R↔3P
3RCLV ₉₀	3RCLI ₉₃	3RCLZ ₉₆	Δ → Y	Y → Δ	MULTSTK
V=I×Z	I=V÷Z	Z=V÷I	X → BAL	3SWAP	3DROP

ELEC p2 menu, showing the 3-phase storage commands in the g-line:

3STOV₉₀ means it stores the ZYX voltages into physical registers 90-92, where treated as the 3-phase voltage V.

2STO	2RCL	nDROP	nDUP	nSWAP	
3STO	3RCL	3DROP	3DUP	3SWAP	4SWAP

The STK, MULTSTK menu, featuring generic triple storage commands:

3STO for example, asks for the bottom destination register nn and stores $X \rightarrow R_{nn}$, $Y \rightarrow R_{nn+1}$, $Z \rightarrow R_{nn+2}$

Phasor Representation of Three-Phase Quantities

An a.c. system may be represented using phasors, which can be interpreted as vectors rotating on the complex plane at a fixed angular frequency imposed by the rotating generator. The power system comprises three voltage vectors and three current vectors rotating in unison. Each quantity is offset in time, and therefore each vector is separated by a fixed angle of 120° in a balanced system.

Voltages and currents are naturally treated as rotating vectors, while impedances are fixed complex vectors that relate voltage and current through Ohm's law. This representation allows all three phases to be analysed simultaneously in a single plane. By treating the vectors as snapshots with magnitudes and relative phase angles, we eliminate the continuous rotational time dependence. For those with a mechanical background, comparison with top dead centre and the advance angle could be drawn, but further details are out of scope here.

In this tutorial, voltages and currents are treated as phasors and impedances as vectors, both being complex numbers representing magnitude and phase angle. Phasors are generally expressed in complex polar form (magnitude and angle) while impedances (and in some cases associated power vectors) are often used as inputs and outputs in rectangular form (real plus imaginary, eg. resistance + reactance). These representations map directly to the triple-register system discussed below.

The triple register system we develop in this app note lets us express all three values as a bundle, each of which expresses a phasor or vector on the complex plane, so that all three together describe the operation of a power generator or a motor driven by that power source.

Phasors stored in triple registers or 3×1 matrices can represent generator or motor phase currents or voltages. For example, generator output may be stored as phase A voltage $240 \text{ V} \angle 0^\circ$, phase B $240 \text{ V} \angle 120^\circ$, and phase C $240 \text{ V} \angle 240^\circ$. Note the fixed 120° phase offsets, which illustrate the balanced system.

In this example, the current drawn may be stored as phase A $2 \text{ A} \angle 10^\circ$, phase B $2 \text{ A} \angle 130^\circ$, and phase C $2 \text{ A} \angle 250^\circ$. Note the relative 10° offsets from the voltages which reflect the phase shift caused by reactive power in the load.

Introducing Star Delta Transforms in Registers

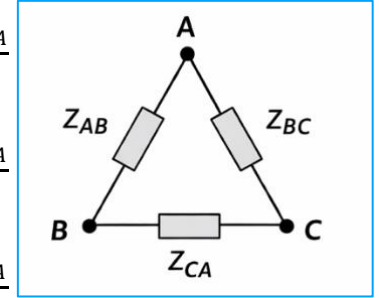
In the ELEC menu, star to delta and delta to star conversions are performed directly on triple complex registers, enabling complete three-phase calculations for unbalanced systems. Each of the three triple registers (R90–R98) holds a full set of three-phase voltages or currents in phasor form, or impedances in complex form, such that all three phases can be transformed simultaneously.

- **Star to Delta:**

$$Z_{\Delta AB} = \frac{Z_{YA} \times Z_{YB} + Z_{YB} \times Z_{YC} + Z_{YC} \times Z_{YA}}{Z_{YC}}$$

$$Z_{\Delta BC} = \frac{Z_{YA} \times Z_{YB} + Z_{YB} \times Z_{YC} + Z_{YC} \times Z_{YA}}{Z_{YA}}$$

$$Z_{\Delta CA} = \frac{Z_{YA} \times Z_{YB} + Z_{YB} \times Z_{YC} + Z_{YC} \times Z_{YA}}{Z_{YB}}$$

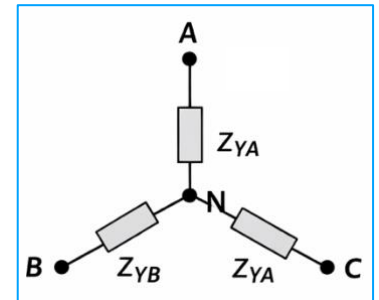


- **Delta to Star:**

$$Z_{YA} = \frac{Z_{\Delta AB} \times Z_{\Delta CA}}{Z_{\Delta AB} + Z_{\Delta BC} + Z_{\Delta CA}},$$

$$Z_{YB} = \frac{Z_{\Delta AB} \times Z_{\Delta BC}}{Z_{\Delta AB} + Z_{\Delta BC} + Z_{\Delta CA}}$$

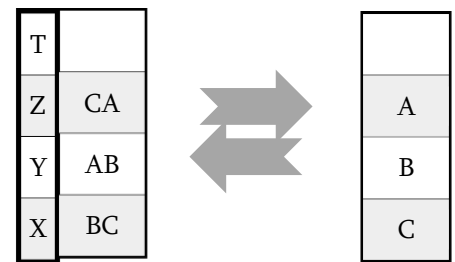
$$Z_{YC} = \frac{Z_{\Delta BC} \times Z_{\Delta CA}}{Z_{\Delta AB} + Z_{\Delta BC} + Z_{\Delta CA}}$$



Each formula^{3 4} is applied phase by phase using triple registers for unbalanced impedances.

The expected stack allocation is as follows: Input is taken from the ZYX triple registers and results are returned to the same ZYX triple registers.

This arrangement allows all three phases to be transformed simultaneously without any matrix operations, while preserving the correct phase relationships.



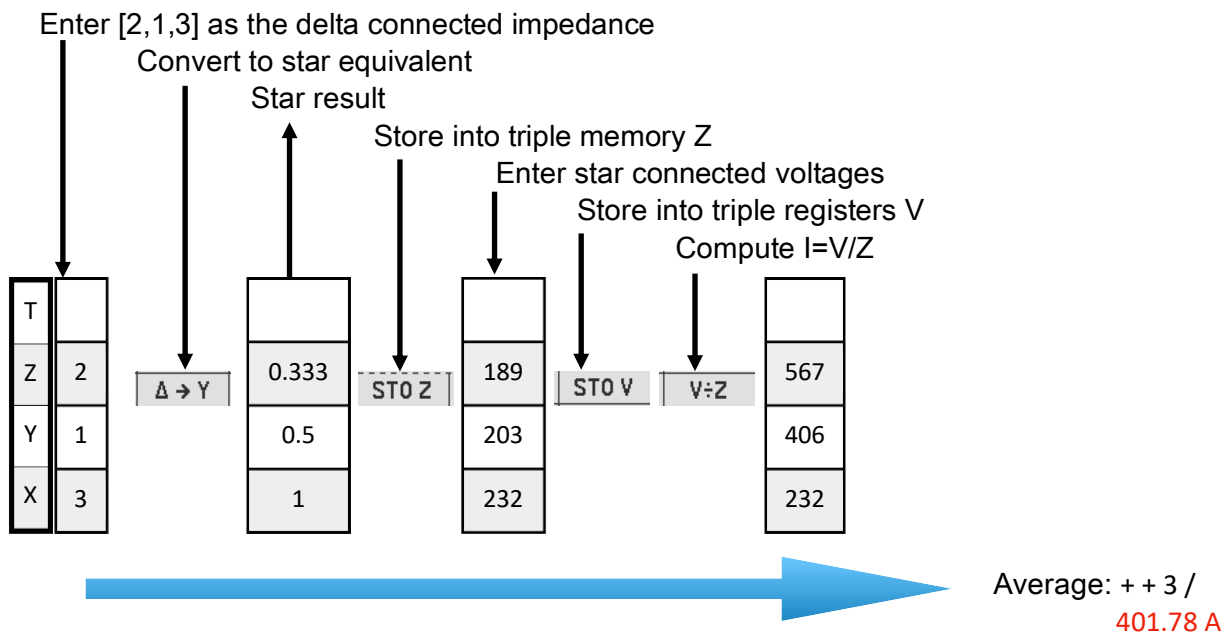
³ [Wikipedia](#), Y-Δ transform

⁴ [Tutorials Point](#), Network Theory - Star to Delta Conversion

Example 1: Star Delta Transform, with 3-phase Ohm's law, using triple registers

Calculate the average current of an unbalanced three phase system with star voltages of [189, 203, 232] V and delta motor terminal impedances of [2, 1, 3] Ohm.

Below the calculation workflow to calculate the line currents from an *unbalanced* system fed from an *unbalanced* voltage using no matrix arithmetic:



Workflow: Start at the top left by entering 2 ENTER 1 ENTER 3, and proceed to the bottom right to press ++ 3 / to get the average current of 401.8 A.

Example 2: Star Delta Transforms using triple registers

Find the delta equivalent of components of a star connected impedance measured to be $6\angle 0^\circ \Omega$, $12\angle -90^\circ \Omega$, $11\angle -90^\circ \Omega$.

Enter impedances as complex numbers. Once the calculator recognises a value as a complex data type, it internally converts between rectangular and polar coordinates as required and displays the result in either rectangular or polar form, according to the user's selection.

- Entry method 1:

Set calculator to Polar mode, for default input and output to POLAR form.

Set modes to: POLAR [DEG()] SIG 3

6 [ENTER] 12 [ENTER] 90 [CHS][COMPLEX] 11 [ENTER] 90 [CHS][COMPLEX] [Y - Δ]

- Entry method 2:

The calculator can be in either POLAR or RECT mode as the method of using $i\odot$ as complex number entry already tells the calculator to expect a complex number in polar form.

Set modes to: [DEG()] SIG 3

6 [ENTER] 12 [$i\odot$] 90 [CHS] [ENTER] 11 [$i\odot$] 90 [CHS] [Y - Δ]

- Result: [Y - Δ]:

ca: $15.91 \angle -43.73^\circ \Omega$ ab: $17.36 \angle -43.73^\circ \Omega$, bc: $31.83 \angle -133.7^\circ \Omega$

- Reverse: [Δ - Y]:

a: 6Ω , b: $12.00 \angle -90.00 \Omega$, c: $11.00 \angle -90.00 \Omega$

Introducing sequence component conversions without matrices

In the ELEC menu, conversions between phase and sequence components are performed directly on triple registers (R90–R98), each storing a complete set of three-phase voltages, currents, or impedances as complex quantities.

For **phase-to-sequence conversions**, sequence components are calculated directly as:

$$\begin{aligned} V_{A0} = V_0 &= (V_A + V_B + V_C)/3 \\ V_{A1} = V_1 &= (V_A + a \cdot V_B + a^2 \cdot V_C)/3 \\ V_{A2} = V_2 &= (V_A + a^2 \cdot V_B + a \cdot V_C)/3 \end{aligned}$$

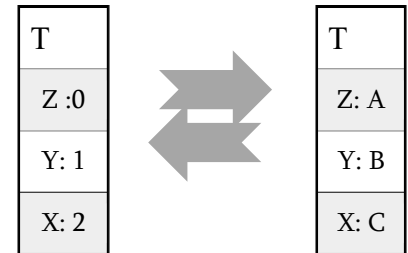
For **sequence-to-phase conversions**, each phase is reconstructed explicitly:

$$\begin{aligned} V_A &= V_{A0} + V_{A1} + V_{A2} \\ V_B &= V_{B0} + V_{B1} + V_{B2} = V_0 + a^2 \cdot V_1 + a \cdot V_2 \\ V_C &= V_{C0} + V_{C1} + V_{C2} = V_0 + a \cdot V_1 + a^2 \cdot V_2 \\ V_{B0} &= V_0; \quad V_{B1} = a^2 \cdot V_1; \quad V_{B2} = a \cdot V_2 \\ V_{C0} &= V_0; \quad V_{C1} = a \cdot V_1; \quad V_{C2} = a^2 \cdot V_2 \end{aligned}$$

By performing these conversions directly in triple registers, users can operate on each phase individually, avoiding matrices. This approach allows intuitive reasoning about system imbalance, fault conditions, and sequence component behaviour while remaining fully consistent with classical three-phase and power-system theory.

The triple register stack allocation for V_{012} stored in ZYX is (Z = V_0 , Y = V_1 , X = V_2) and for V_{ABC} stored in ZYX is (Z = phase A, Y = phase B, X = phase C).

Performing $[abc \rightarrow 012]$ and $[012 \rightarrow abc]$ in succession returns the original phase values.



Example 3: Symmetrical Components without matrices^{5 6}

$$\begin{bmatrix} V_0 \\ V_1 \\ V_2 \end{bmatrix} = \frac{1}{3} \begin{bmatrix} 1 & 1 & 1 \\ 1 & a & a^2 \\ 1 & a^2 & a \end{bmatrix} \begin{bmatrix} 1.00 \angle 0^\circ \\ 1.20 \angle -110^\circ \\ 0.90 \angle 120^\circ \end{bmatrix} = \begin{bmatrix} 0.047 + 0.116i \\ 1.027 + 0.070i \\ -0.074 + 0.047i \end{bmatrix} = \begin{bmatrix} 0.125 \angle -68.16^\circ \\ 1.030 \angle -3.87^\circ \\ 0.087 \angle -147.7^\circ \end{bmatrix}$$

Use $[abc \rightarrow 012]$ function to determine symmetrical components. Note that both $[abc \rightarrow 012]$ and $[012 \rightarrow abc]$ functions require that the calculator be in SSIZE8 mode. Inputs should be provided in phase order ZYX order where Z corresponds to phase 1.

```
SIG 3 DEG POLAR
1 ENTER 1.2 ENTER 110 CHS COMPLEX 0.9 ENTER 120 COMPLEX
[abc->012]
```

```
sym0 : 0.125 0 -68.16°
sym1 : 1.030 -3.868°
sym2 : 0.087 27 -147.7°
```

Use $[012 \rightarrow abc]$ to determine the phase components:

```
a : 1.000 0.000°
b : 1.200 -110.000°
c : 0.900 120.000°
```

Note $[abc \rightarrow 012]$ followed by $[012 \rightarrow abc]$ returns the original.

⁵ Academia, *Symmetrical components, Chapter 7 Ex. 7.2*, [link](#), copies: [NPTEL](#), [Studylib](#)

⁶ KDK College of Engineering, *Symmetrical component Transformation, Chapter 1 Ex. 1.2 on p4*, [link](#)

Symmetrical-Component Transformation Matrix Definition

The transformation matrix A is defined for positive sequence with the middle cell 2,2 = a, and the transformations used are below. Forward and inverse transforms are inverse pairs:

Rotation	Forward	Inverse	Notes
ABC	$V_{012} = \frac{1}{3}AV_{ABC}$	$V_{ABC} = A^{-1}V_{012}$	Standard positive sequence
ACB	$V_{012} = \frac{1}{3}A^{-1}V_{ABC}$	$V_{ABC} = AV_{012}$	For ACB, A and A^{-1} are swapped.

In prior versions, the ELEC menu used a historical A-matrix with middle cell 2,2 = a². In this revision, [A] in the calculator is updated to the standard ABC convention (middle cell 2,2 = a).

Symmetrical-Component Matrix Calculations

While non-matrix methods are convenient, matrices remain extremely useful and are fully supported. The non-matrix formulas in the previous section are derived from the forward and inverse ABC sequence standard symmetrical component transformations:

Phase to Sequence:

$$\begin{bmatrix} V_0 \\ V_1 \\ V_2 \end{bmatrix} = \frac{1}{3} \cdot \begin{bmatrix} 1 & 1 & 1 \\ 1 & a & a^2 \\ 1 & a^2 & a \end{bmatrix} \cdot \begin{bmatrix} V_A \\ V_B \\ V_C \end{bmatrix}, \quad A = \begin{bmatrix} 1 & 1 & 1 \\ 1 & a & a^2 \\ 1 & a^2 & a \end{bmatrix} = \begin{bmatrix} 1. \angle 0.^\circ & 1. \angle 0.^\circ & 1. \angle 0.^\circ \\ 1. \angle 0.^\circ & 1. \angle 120.^\circ & 1. \angle -120.^\circ \\ 1. \angle 0.^\circ & 1. \angle -120.^\circ & 1. \angle 120.^\circ \end{bmatrix}$$

Sequence to Phase:

$$\begin{bmatrix} V_A \\ V_B \\ V_C \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & a^2 & a \\ 1 & a & a^2 \end{bmatrix} \cdot \begin{bmatrix} V_0 \\ V_1 \\ V_2 \end{bmatrix}, \quad A^{-1} = \frac{1}{3} \begin{bmatrix} 1 & 1 & 1 \\ 1 & a^2 & a \\ 1 & a & a^2 \end{bmatrix} = \begin{bmatrix} 1. \angle 0.^\circ & 1. \angle 0.^\circ & 1. \angle 0.^\circ \\ 1. \angle 0.^\circ & 1. \angle -120.^\circ & 1. \angle 120.^\circ \\ 1. \angle 0.^\circ & 1. \angle 120.^\circ & 1. \angle -120.^\circ \end{bmatrix}$$

For ABC rotation:

$$V_{012} = \frac{1}{3}AV_{ABC}, \quad V_{ABC} = A^{-1}V_{012}$$

For ACB rotation:

$$V_{012} = \frac{1}{3}A^{-1}V_{ABC}, \quad V_{ABC} = AV_{012}$$

$a = e^{j2\pi/3}$ is the phase rotation operator. The same formulas apply to currents I as to voltages V . See the lecture notes⁷ supporting the above.

The C47 [A] matrix:

$$[A] = \begin{bmatrix} 1. \angle 0.^\circ & 1. \angle 0.^\circ & 1. \angle 0.^\circ \\ 1. \angle 0.^\circ & 1. \angle 120.^\circ & 1. \angle -120.^\circ \\ 1. \angle 0.^\circ & 1. \angle -120.^\circ & 1. \angle 120.^\circ \end{bmatrix}$$

To change the rotation use [A⁻¹]:

$$[A^{-1}] = \begin{bmatrix} 1. \angle 0.^\circ & 1. \angle 0.^\circ & 1. \angle 0.^\circ \\ 1. \angle 0.^\circ & 1. \angle -120.^\circ & 1. \angle 120.^\circ \\ 1. \angle 0.^\circ & 1. \angle 120.^\circ & 1. \angle -120.^\circ \end{bmatrix}$$

⁷ University of Idaho, [lecture notes](#), pp.3 and 4

Example 4: Symmetrical Components with matrices ^{8 9}

$$\begin{bmatrix} V_0 \\ V_1 \\ V_2 \end{bmatrix} = \frac{1}{3} \begin{bmatrix} 1 & 1 & 1 \\ 1 & a & a^2 \\ 1 & a^2 & a \end{bmatrix} \begin{bmatrix} 1.00 \angle 0^\circ \\ 1.20 \angle -110^\circ \\ 0.90 \angle 120^\circ \end{bmatrix} = \begin{bmatrix} 0.047 + 0.116i \\ 1.027 + 0.070i \\ -0.074 + 0.047i \end{bmatrix} = \begin{bmatrix} 0.125 \angle -68.16^\circ \\ 1.030 \angle -3.87^\circ \\ 0.087 \angle -147.7^\circ \end{bmatrix}$$

For comparison, let's do the same problem with matrices: $V_{012} = \frac{1}{3} A V_{ABC}$

```
ALL 3 DEG POLAR
1 ENTER 1.2 ENTER 110 CHS COMPLEX 0.9 ENTER 120
[zyx->M] [A] [X<>Y] x 3 / [->P]
[M->zyx] Bug : Only Real parts output. Already corrected in
2025 in WIP vector branch. Workaround: values can be
extracted via RCLVEL:
[RCL][...VEL1][X<>Y]
[RCL][...VEL2][X<>Y]
[RCL][...VEL3][X<>Y]
```

```
[ 1.00 ∠ 0.00° 1.20 ∠ -110.00° 0.90 ∠ 120.00° ]T
[ 0.125 ∠ -68.2° 1.04 ∠ 3.87° 0.087 3 ∠ 148.° ]T
```

```
0.125 0 ∠ -68.16°
1.030 ∠ 3.868°
0.087 27 ∠ 147.7°
```

Continuing to use above results, the phase components: $V_{ABC} = A^{-1} V_{012}$

```
[zyx->M] [A-1] [X<>Y] x [->P]
[M->zyx] Bug : Only Real parts output. Already corrected in
2025 in WIP vector branch. Workaround: values can be
extracted via RCLVEL:
[RCL][...VEL1][X<>Y]
[RCL][...VEL2][X<>Y]
[RCL][...VEL3][X<>Y]
```

```
[ 1.00 ∠ 0.00° 1.20 ∠ -110.00° 0.90 ∠ 120.00° ]T
```

```
1.000 ∠ 2.292 × 10-33°
1.200 ∠ -110.0°
0.900 0 ∠ 120.0°
```

Example 5: Symmetrical Components without matrices:

Find the sequence components of a voltage measured to be $120 \angle 0$, $21 \angle 120$, $31 \angle 240$.

- Entry method 1: Polar mode: POLAR [DEG()] SIG 3

120 [ENTER] 21 [ENTER] 120 [COMPLEX] 31 [ENTER] 240 [COMPLEX] [>012]

```
2020-02-08 19:53 ☉ /max 64:2 ↑
0.000
sym0 = 31.47 ∠ -5.264°
sym1 = 31.47 ∠ 5.264°
sym2 = 57.33 ∠ -3.331 × 10-34°
```

- Entry method 2: Any mode: [DEG()] SIG 3

120 [ENTER] 21 [∠] 120 [ENTER] 31 [∠] 240 [>012]

- Remove the -3E-34 angle in X by hiding numbers smaller than 1E-30 using HIDE 30

```
2026-01-18 ☉ /64x 64:2 ↑
31.83 ∠ -133.7°
sym0 : 31.47 ∠ -5.264°
sym1 : 31.47 ∠ 5.264°
sym2 : ≈ 57.33 ∠ 0°
```

⁸ Academia, Symmetrical components, Chapter 7 Ex. 7.2, [link](#), copies: [NPTEL](#), [Studylib](#)

⁹ KDK College of Engineering, Symmetrical component Transformation, Chapter 1 Ex. 1.2 on p4, [link](#)

Version history:

2020-02 Original

Written for the original name WP43C

2023-07-19 Major Revision

Updated for name change C43

2024-09-09 Major Revision

Updated for name change C47

2026-01-19 Major Revision and ELEC function changes

Updated for name change to C47/R47

Previous software and documentation versions used the matrix A to implement an ACB sequence by redefining A in the forward transform rather than applying A^{-1} . This behaviour is now corrected to align with standard ABC conventions.

This update changes the content of ELEC $[A]$ to align with textbook conventions. The matrix is updated by swapping the instances of a and a^2 from the previous version (prior to 00.109.3.01b). Any existing user programs that reference A will experience a phase rotation reversal.

Author: Jaco Mostert; Copyright (C) 2026 The C47 Development Team. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license can be downloaded from gnu.org.