GetPauliStringReformatted

```
SetDirectory @ NotebookDirectory[];
Import["../Link/QuESTlink.m"];
```

Doc

```
? GetPauliStringReformatted
```

Symbol

Reformats symbolic Pauli strings into a variety of other formats convenient for processing.
GetPauliStringReformatted[product, "Index"] returns the integer index of the given Pauli product
in the ordered basis of Pauli products. The zero target is treated as least significant.
GetPauliStringReformatted[string, "Index"] returns a list of
{index, coefficient} pairs which describe all Pauli products in the given string.
GetPauliStringReformatted[..., "Digits"] returns the individual digits of the basis Pauli
string's index (or indices), in base 4, where the rightmost digit is the least significant.
GetPauliStringReformatted[..., "Kronecker"] expands the Pauli string into an explicit Kronecker form.
The zero target in the given product corresponds to the rightmost Pauli in the Kronecker form.
GetPauliStringReformatted[..., "String"] returns a compact, string–form of the "Kronecker" format.
GetPauliStringReformatted[..., numQubits] expands the "Digits", "Kronecker" and "String"
formats to the specified number of qubits, by padding with '0' digits or 'Id' operators.

Correctness

Index

Product

```
GetPauliStringReformatted[Id4, "Index"]
```

```
0
```

```
GetPauliStringReformatted[X_0, "Index"]
GetPauliStringReformatted[Y_0, "Index"]
GetPauliStringReformatted[Z_0, "Index"]
```

```
1
```

```
2
```

```
3
```

```
GetPauliStringReformatted[X_1, "Index"]
GetPauliStringReformatted[Id_5 X_1 Id_0, "Index"]
```

```
4
```

```
4
```

```
GetPauliStringReformatted[Z_34, "Index"] === 3 × 4^34
```

```
True
```

```
p = Product[Z_t, {t, 0, 34}];
GetPauliStringReformatted[p, "Index"] === 4^(34+1) - 1
```

```
True
```

```
ind = 1 245 194;
str = GetPauliString[ind]
ind === GetPauliStringReformatted[str, "Index"]
```

```
X_10 Y_0 Y_1 Z_8
```

```
True
```

String

```
GetPauliStringReformatted[X_0 + a Y_0 + Z_3 X_0 + a X_0 b Y_2 c Z_4, "Index"]
```

```
{{1, 1}, {2, a}, {193, 1}, {801, a b c}}
```

```
GetPauliStringReformatted[X_0 + a z X_0, "Index"]
```

```
{{1, 1}, {1, a z}}
```

```
GetPauliStringReformatted[ Z_3 X_0 , "Index"]
GetPauliStringReformatted[ X_0 Y_2 Z_4, "Index"]
GetPauliStringReformatted[X_0 + a Y_0 + Z_3 X_0 + a X_0 b Y_2 c Z_4, "Index"]
```

```
193
```

```
801
```

```
{{1, 1}, {2, a}, {193, 1}, {801, a b c}}
```

```
GetPauliStringReformatted[ X_0 , "Index"]
GetPauliStringReformatted[ 1. X_0 , "Index"]
```

```
1
```

```
{{1, 1.}}
```

Digits

Product

```
GetPauliStringReformatted[X_0, "Digits"]
GetPauliStringReformatted[X_0, 5, "Digits"]
GetPauliStringReformatted[X_0, "Digits", 10]
```

```
{1}
```

```
{0, 0, 0, 0, 1}
```

```
{0, 0, 0, 0, 0, 0, 0, 0, 0, 1}
```

```
GetPauliStringReformatted[Y_4, "Digits"]
```

```
{2, 0, 0, 0, 0}
```

```
GetPauliStringReformatted[X_0 Y_1 Z_2, "Digits"]
```

```
{3, 2, 1}
```

String

```
GetPauliStringReformatted[a X_0 Y_1 Z_2, "Digits"]
```

```
{{{3, 2, 1}, a}}
```

```
GetPauliStringReformatted[a Z_5 + b Z_5, "Digits"]
```

```
{{{3, 0, 0, 0, 0, 0}, a}, {{3, 0, 0, 0, 0, 0}, b}}
```

Kronecker

Product

```
GetPauliStringReformatted[X_0, "Kronecker"]
GetPauliStringReformatted[Z_0, "Kronecker"]
```

```
⊗X
```

```
⊗Z
```

```
GetPauliStringReformatted[X_1, "Kronecker"]
```

```
X ⊗ Id
```

```
GetPauliStringReformatted[X_1, 3, "Kronecker"]
GetPauliStringReformatted[X_1, "Kronecker", 4]
```

```
X ⊗ Id
```

```
X ⊗ Id
```

```
GetPauliStringReformatted[X0 Y1 Z2 Id3, "Kronecker"]
```

```
Id⊗Z⊗Y⊗X
```

```
GetPauliStringReformatted[Z9, "Kronecker"]
```

```
Z⊗Id⊗Id⊗Id⊗Id⊗Id⊗Id⊗Id⊗Id⊗Id
```

```
GetPauliStringReformatted[X0 Y1 Z2 Id3, 10, "Kronecker"]
```

```
Id⊗Id⊗Id⊗Id⊗Id⊗Id⊗Id⊗Z⊗Y⊗X
```

```
str = X0 Y1 Z2 X3 Y4;
```

```
matr = KroneckerProduct @@ (PauliMatrix[# /. {Id → 0, X → 1, Y → 2, Z → 3}] & /@
      GetPauliStringReformatted[str, "Kronecker"]);
```

```
matr === Normal @ CalcPauliExpressionMatrix[str]
```

```
True
```

String

```
GetPauliStringReformatted[   X0 Y1 Z2 Id3, "Kronecker"]
GetPauliStringReformatted[1. X0 Y1 Z2 Id3, "Kronecker"]
GetPauliStringReformatted[a  X0 Y1 Z2 Id3, "Kronecker"]
```

```
Id⊗Z⊗Y⊗X
```

```
{{Id⊗Z⊗Y⊗X, 1.}}
```

```
{{Id⊗Z⊗Y⊗X, a}}
```

```
str = a X0 Y1 Z2 Id3 + b c X4 + d e f Y0 + Z0 + X1 Z4;
GetPauliStringReformatted[str, "Kronecker"]
```

```
{{X⊗Id⊗Id⊗Id⊗Id, b c}, {Id⊗Id⊗Id⊗Id⊗Y, d e f},
 {Id⊗Id⊗Id⊗Id⊗Z, 1}, {Id⊗Id⊗Z⊗Y⊗X, a}, {Z⊗Id⊗Id⊗X⊗Id, 1}}
```

String

Product

```
GetPauliStringReformatted[X0 Y1 Z2, "String"]
```

```
ZYX
```

```
GetPauliStringReformatted[X0 Y1 Z2, 10, "String"]
```

```
IIIIIIIZYX
```

```
GetPauliStringReformatted[Z40, "String"]
```

```
ZIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII
```

String

```
str = GetRandomPauliString[4, 5]
GetPauliStringReformatted[str, "String"]
GetPauliStringReformatted[str, "String", 10]
```

$$-0.128532\, X_3\, Y_2\, Z_1 - 0.241491\, X_0\, Y_2\, Y_3\, Z_1 + 0.635716\, X_0\, X_2\, Z_3 - 0.795895\, X_1\, X_2\, Z_3 + 0.144195\, Y_0\, Z_2\, Z_3$$

```
{{XYZI, -0.128532}, {YYZX, -0.241491},
 {ZXIX, 0.635716}, {ZXXI, -0.795895}, {ZZIY, 0.144195}}
```

```
{{IIIIIIXYZI, -0.128532}, {IIIIIIYYZX, -0.241491},
 {IIIIIIZXIX, 0.635716}, {IIIIIIZXXI, -0.795895}, {IIIIIIZZIY, 0.144195}}
```

Errors

```
GetPauliStringReformatted[Z10, 10, "String"]
```

GetPauliStringReformatted: The given Pauli string targeted a larger index qubit than the number of qubits specified.

```
$Failed
```

```
GetPauliStringReformatted[X-1 Y0 + X2, "Index"]
```

GetPauliStringReformatted: Invalid arguments. See ?GetPauliStringReformatted

```
$Failed
```

```
GetPauliStringReformatted[X-1]
```

GetPauliStringReformatted: Invalid arguments. See ?GetPauliStringReformatted

```
$Failed
```

```
GetPauliStringReformatted[X2, "BadMethod"]
```

GetPauliStringReformatted: Invalid arguments. See ?GetPauliStringReformatted

```
$Failed
```

```
GetPauliStringReformatted[X2, 1, "Kronecker"]
```

GetPauliStringReformatted: The given Pauli string targeted a larger index qubit than the number of qubits specified.

```
$Failed
```

```
GetPauliStringReformatted[X0 X0, "Index"]
GetPauliStringReformatted[X0 Y0, "Index"]
```

GetPauliStringReformatted: Invalid arguments. See ?GetPauliStringReformatted

```
$Failed
```

GetPauliStringReformatted: Invalid arguments. See ?GetPauliStringReformatted

```
$Failed
```

```
GetPauliStringReformatted[1]
```

GetPauliStringReformatted: Invalid arguments. See ?GetPauliStringReformatted

```
$Failed
```

```
GetPauliStringReformatted[]
```

GetPauliStringReformatted: Invalid arguments. See ?GetPauliStringReformatted

```
$Failed
```