RetargetCircuit

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

Deprecation

Renamed to **GetCircuitRetargeted** as of v0.18

?RetargetCircuit

> Symbol
>
> This function is deprecated. Please instead use GetCircuitRetargeted.
>
> ⌄

RetargetCircuit[X_2, 2 → 1]

⋯ GetCircuitRetargeted : The function RetargetCircuit[] is deprecated, though has still been attemptedly performed. In future, please use GetCircuitRetargeted[], or temporarily hide this message using Quiet[].

{X_1}

?GetCircuitRetargeted

> Symbol
>
> GetCircuitRetargeted[circuit, rules] returns the given circuit but with its target and control qubits modified as per the given rules. The rules can be anything accepted by ReplaceAll.
>
> For instance GetCircuitRetargeted[..., {0->1, 1->0}] swaps the first and second qubits, and GetCircuitRetargeted[..., q_ -> q + 10] shifts every qubit up by 10.
>
> This function modifies only the qubits in the circuit, carefully avoiding modifying gate arguments and other data, so it is a safe alternative to simply evaluating (circuit /. rules).
>
> Custom user gates are supported provided they adhere to the standard QuESTlink subscript format.
>
> ⌄

Doc

? RetargetCircuit

> Symbol
>
> RetargetCircuit[circuit, rules] returns the given circuit but with its target and control qubits
>   modified as per the given rules. The rules can be anything accepted by ReplaceAll.
> For instance RetargetCircuit[..., {0–>1, 1–>0}] swaps the first and
>   second qubits, and RetargetCircuit[..., q_ –> q + 10] shifts every qubit up by 10.
> This function modifies only the qubits in the circuit, carefully avoiding modifying gate
>   arguments and other data, so it is a safe alternative to simply evaluating (circuit /. rules).
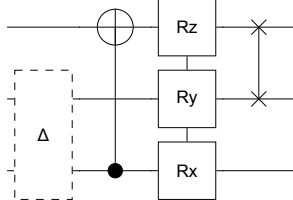> Custom user gates are supported provided they adhere to the standard QuESTlink subscript format.
>
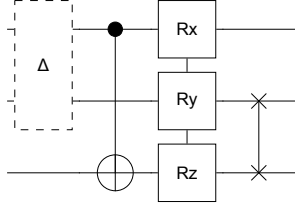> ⌄

Tests

? QuEST`Gate`*

> ⌄ QuEST`Gate`
>
> | | | | | |
> |------|-----|------|-----|------------|
> | **Damp** | **H** | **Matr** | **Ry** | **U** |
> | **Deph** | **Id** | **P** | **Rz** | **UNonNorm** |
> | **Depol** | **Kraus** | **Ph** | **S** | **X** |
> | **Fac** | **KrausNonTP** | **R** | **SWAP** | **Y** |
> | **G** | **M** | **Rx** | **T** | **Z** |
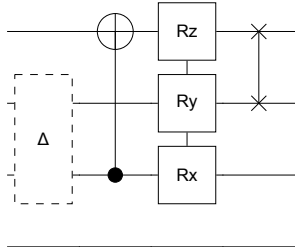
Arbitrary rules

```
in = Circuit[Depol_{0,1}[x] C_0[X_2] R[x, X_0 Y_1 Z_2] SWAP_{1,2}];
DrawCircuit[in]
```

```
out = RetargetCircuit[in, {0 → 2, 2 → 0}]
DrawCircuit[out]
```

$\{\text{Depol}_{2,1}[x], C_2[X_0], R[x, X_2 Y_1 Z_0], \text{SWAP}_{1,0}\}$


```
out = RetargetCircuit[in, q_ → q + 1]
DrawCircuit[out]
```

$\{\text{Depol}_{1,2}[x], C_1[X_3], R[x, X_1 Y_2 Z_3], \text{SWAP}_{2,3}\}$

Gate support, and no modification of arguments

```
RetargetCircuit[ Circuit[
        Damp₀[0] Deph₀,₁[0] Depol₀,₁[0] Fac[0] × G[0] H₀ Id₀,₁ ×
        Kraus₀,₁[0, 0] KrausNonTP₀,₁[0, 1] M₀,₁ Matr₀,₁[0] ×
        P₀,₁[0, 1] Ph₀,₁[0] R[0, X₀] × R[0, X₀ Y₀ Z₁] Rx₀,₁[0] ×
        Ry₀,₁[0] Rz₀,₁[0] S₀ SWAP₀,₁ T₀ U₀,₁[0, 1] ×
        UNonNorm₀,₁[0, 1] X₀ Y₀ Z₀],
    0 → a]
```

$\{\text{Damp}_a[0], \text{Deph}_{a,1}[0], \text{Depol}_{a,1}[0], \text{Fac}[0], G[0], H_a, \text{Id}_{a,1}, \text{Kraus}_{a,1}[0, 0],$
$\text{KrausNonTP}_{a,1}[0, 1], M_{a,1}, \text{Matr}_{a,1}[0], P_{a,1}[0, 1], \text{Ph}_{a,1}[0], R[0, X_a], R[0, X_a Y_a Z_1],$
$\text{Rx}_{a,1}[0], \text{Ry}_{a,1}[0], \text{Rz}_{a,1}[0], S_a, \text{SWAP}_{a,1}, T_a, U_{a,1}[0, 1], \text{UNonNorm}_{a,1}[0, 1], X_a, Y_a, Z_a\}$

Controls

```
RetargetCircuit[C₀,₁,₂[X₃], {0 → a, 3 → b}]
```

$\{C_{a,1,2}[X_b]\}$

Qubit configurations (sequence vs list)

```
RetargetCircuit[{X₀, X₀,₁, X₍₀₎, X₍₀,₁₎}, 0 → a]
```

$\{X_a, X_{a,1}, X_{\{a\}}, X_{\{a,1\}}\}$

$\text{RetargetCircuit}\big[\{\text{Ph}_0[0], \text{Ph}_{0,1}[0], \text{Ph}_{\{0\}}[0], \text{Ph}_{\{0,1\}}[0]\}, \ 0 \to a\big]$

$\big\{\text{Ph}_a[0], \text{Ph}_{a,1}[0], \text{Ph}_{\{a\}}[0], \text{Ph}_{\{a,1\}}[0]\big\}$

$\text{RetargetCircuit}[\{R[0, X_0], R[0, X_0\ Y_0\ Z_1]\}, \ 0 \to a]$

$\{R[0, X_a], R[0, X_a\ Y_a\ Z_1]\}$

$\text{RetargetCircuit}\big[\{C_0@R[0, X_0], C_{\{0,1\}}@R[0, X_0\ Y_0\ Z_1]\}, \ 0 \to a\big]$

$\big\{C_a[R[0, X_a]], C_{\{a,1\}}[R[0, X_a\ Y_a\ Z_1]]\big\}$

Non-integer qubits

$\text{RetargetCircuit}[\{\text{Rx}_a[a]\}, \ a \to b]$

$\{\text{Rx}_b[a]\}$

Errors

$\text{RetargetCircuit}[\{X_0, Y_1\}, \text{invalid}]$

⋯ ReplaceAll : {invalid} is neither a list of replacement rules nor a valid dispatch table, and so cannot be used for replacing.

$Failed

$\text{RetargetCircuit}[\{\text{Anon}\}, 0 \to 1]$

RetargetCircuit : Could not identify qubits in unrecognised gate: Anon

$Failed

$\text{RetargetCircuit}[\text{hi}]$

RetargetCircuit : Invalid arguments. See ?RetargetCircuit

$Failed

$\text{RetargetCircuit}[\text{hello, there, friend}]$

RetargetCircuit : Invalid arguments. See ?RetargetCircuit

$Failed