

Original

IfGoto 1, L0, L1

b := Add a, 1

Analyse

Transformation

Optimiert

Goto L0

b := Add a, 1

```
graph LR; subgraph Original; A[IfGoto 1, L0, L1]; B[b := Add a, 1]; end; subgraph Optimiert; D[Goto L0]; E[b := Add a, 1]; end; Analyse[Analyse]; Transformation[Transformation]; A --> Analyse; B --> Analyse; A --> Transformation; B --> Transformation; Analyse --> Transformation; Transformation --> E;
```

The diagram illustrates a code optimization process. It starts with an 'Original' code block containing two instructions: 'IfGoto 1, L0, L1' (in a red box) and 'b := Add a, 1' (in a yellow box). These instructions are processed by two intermediate blocks: 'Analyse' (orange box) and 'Transformation' (blue box). Arrows show that both original instructions feed into both 'Analyse' and 'Transformation'. Additionally, 'Analyse' feeds into 'Transformation'. Finally, the 'Transformation' block produces the 'Optimiert' (Optimized) code block, which contains 'Goto L0' (in a green box) and 'b := Add a, 1' (in a yellow box). The 'b := Add a, 1' instruction remains unchanged, while the 'IfGoto' instruction is simplified to a 'Goto'.