

FIG. 1

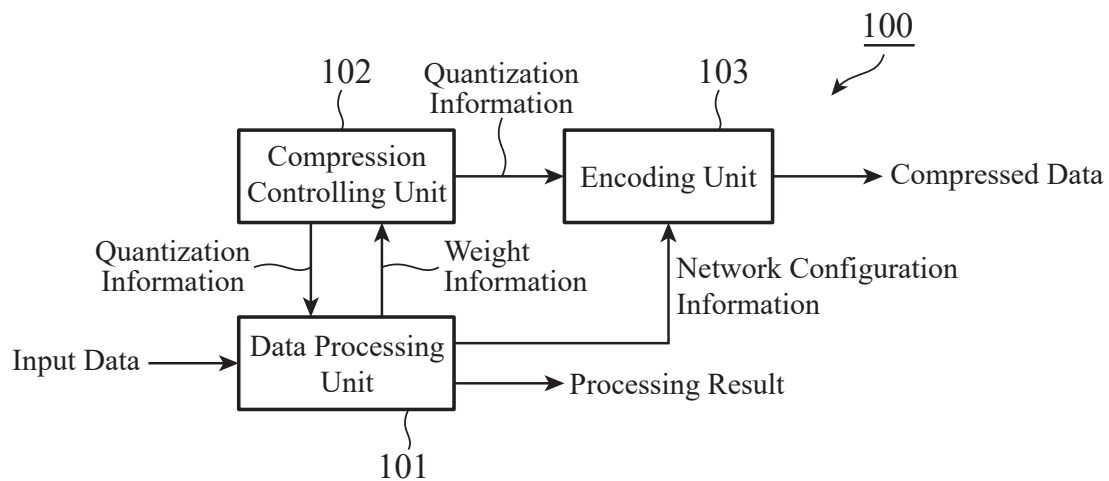


FIG. 2

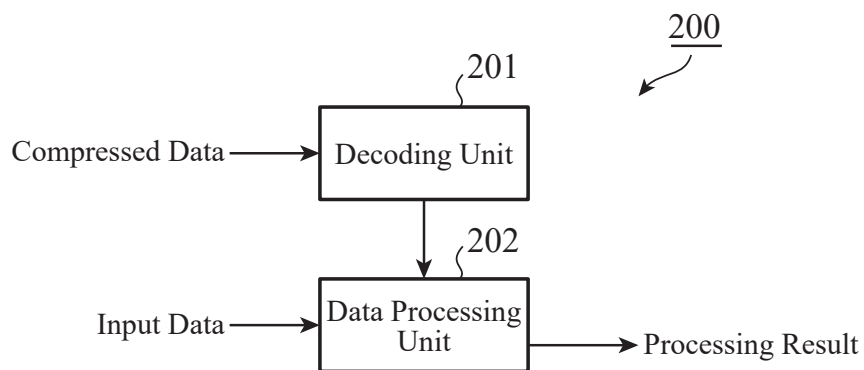


FIG. 3A

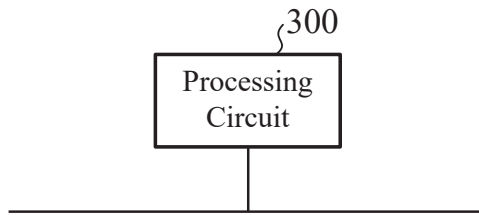


FIG. 3B

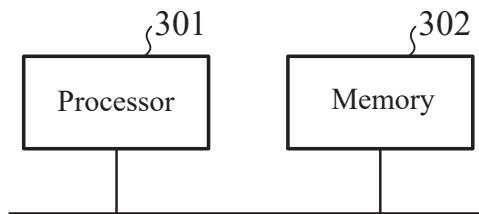


FIG. 4

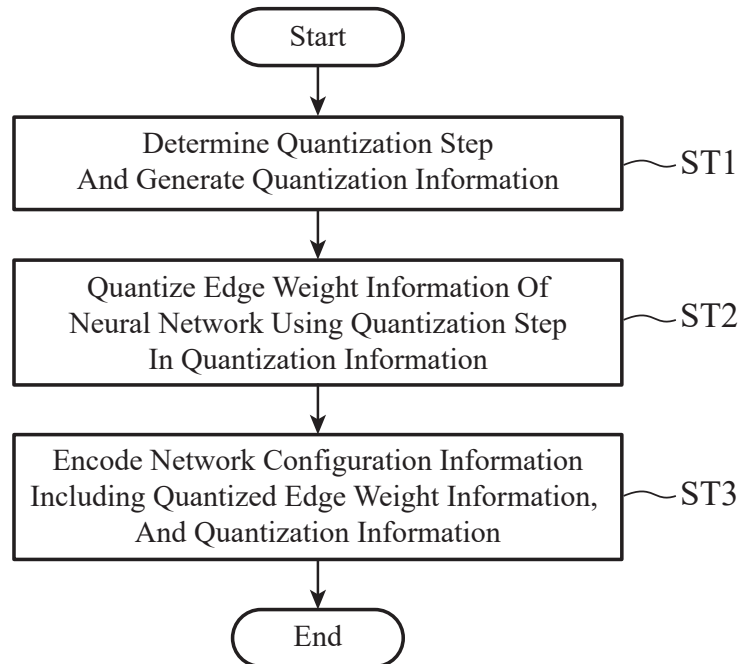


FIG. 5

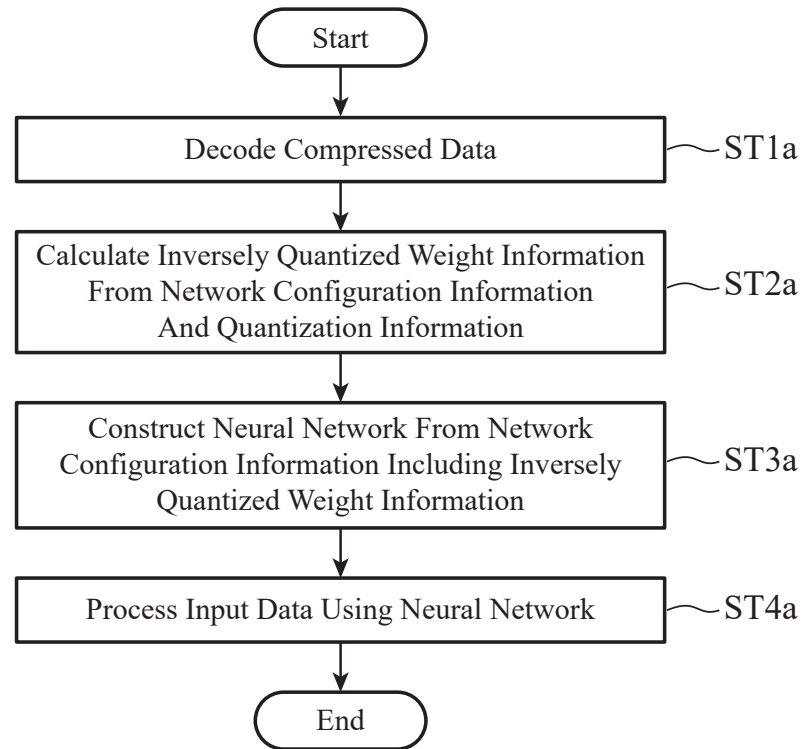


FIG. 6

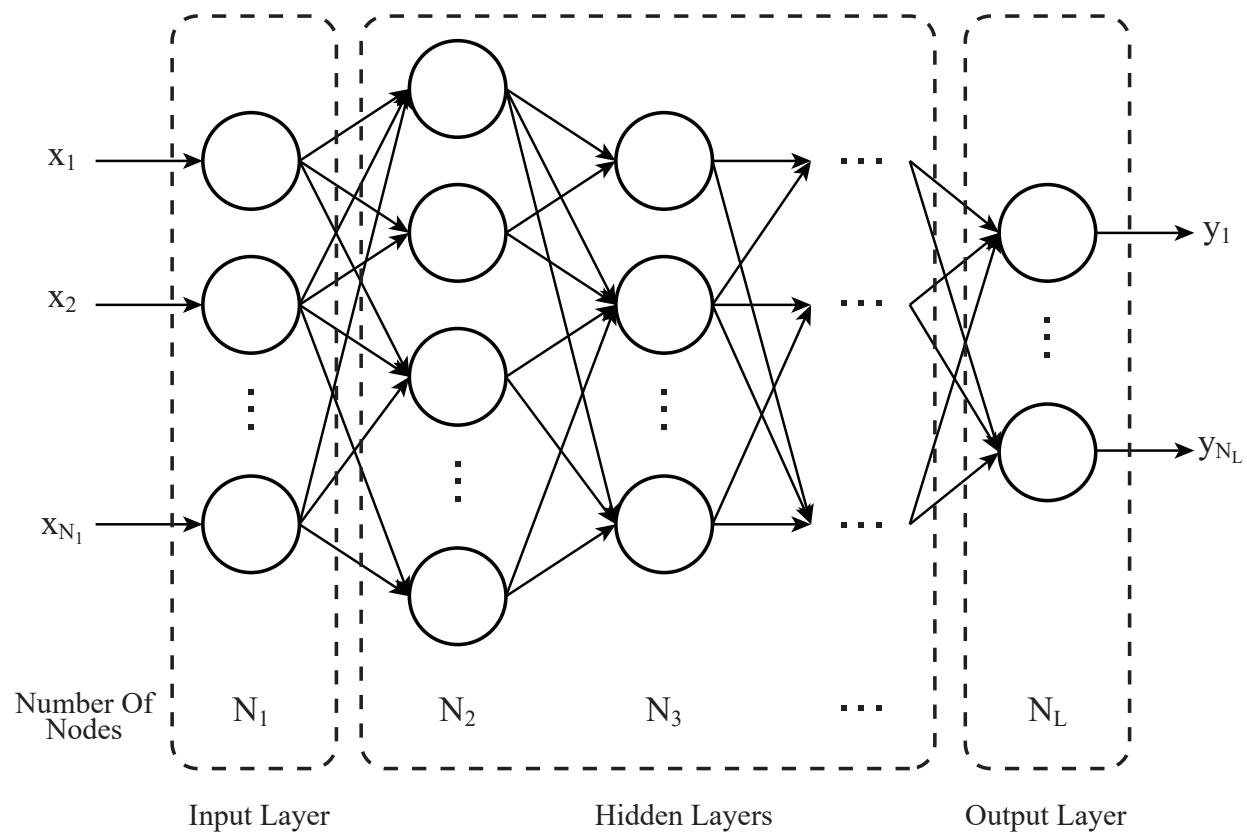


FIG. 7

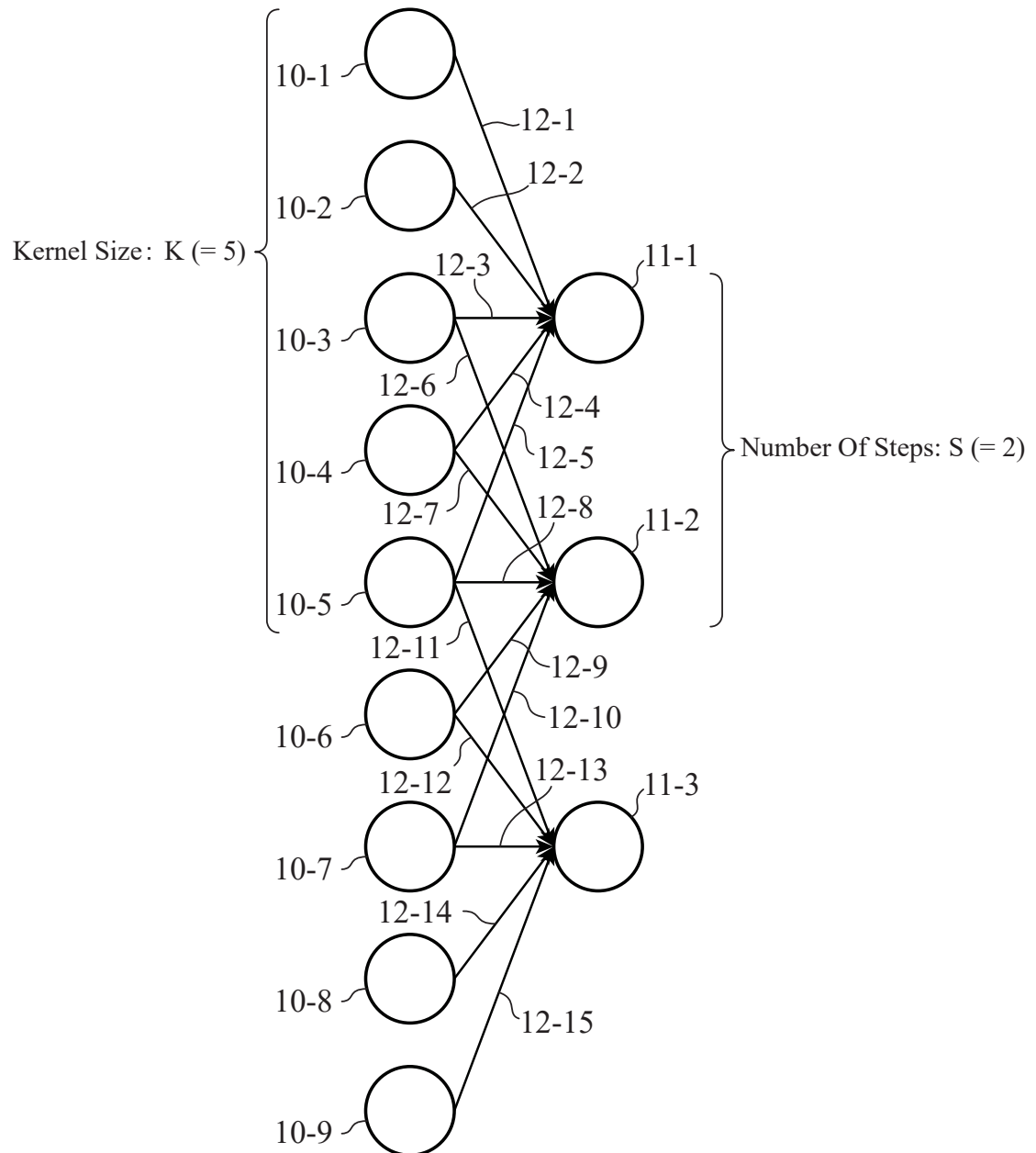


FIG. 8

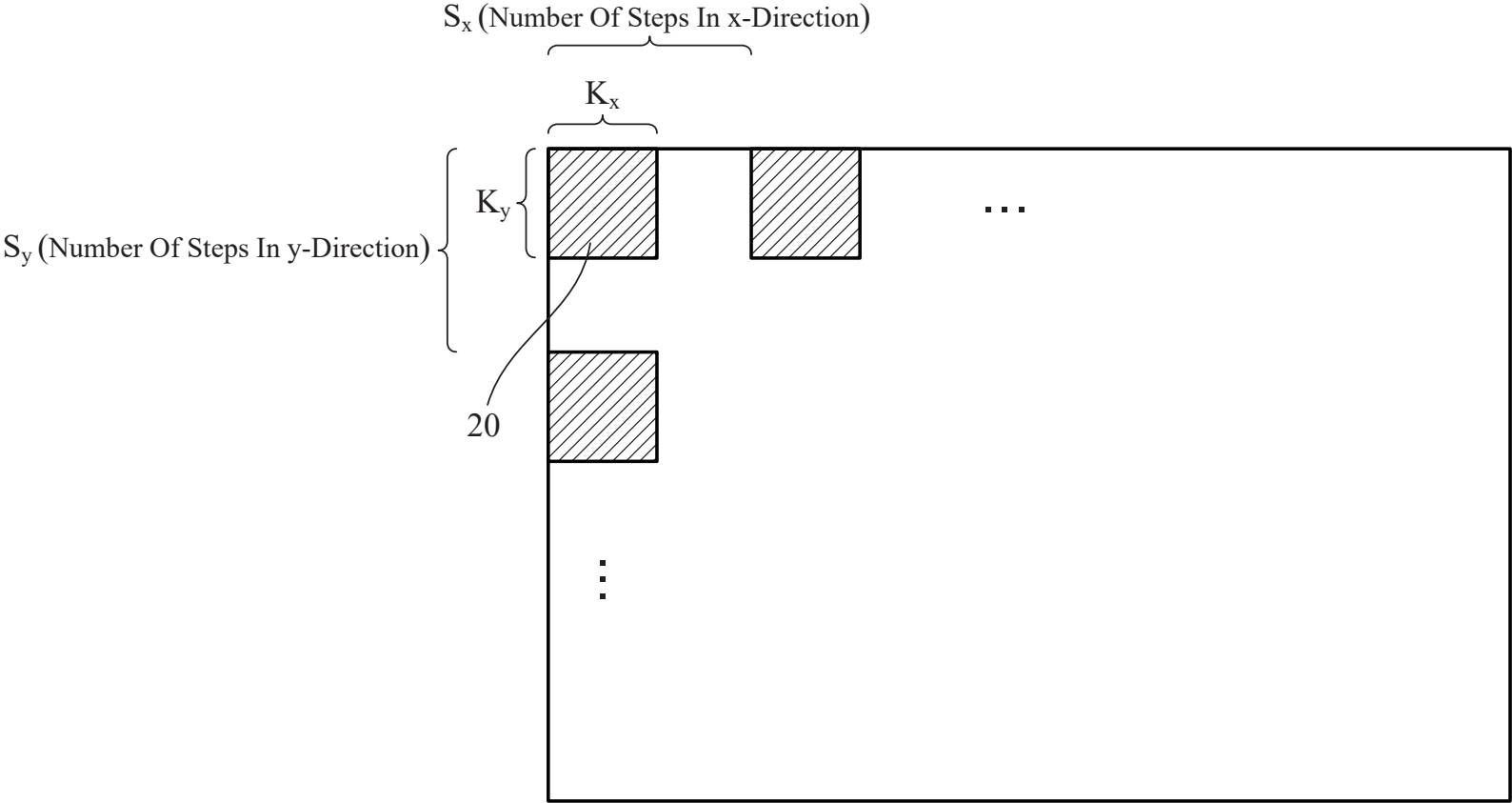


FIG. 9

		Node Index				
		1	2	3	...	N_l
Edge Index	1	w_{11}	w_{21}	w_{31}	...	w_{N_l1}
	2	w_{12}	w_{22}	w_{32}	...	w_{N_l2}
	3	w_{13}	w_{23}	w_{33}	...	w_{N_l3}
	$\vdots$	$\vdots$	$\vdots$	$\vdots$	...	$\vdots$
	N_{l-1}	$w_{1N_{l-1}}$	$w_{2N_{l-1}}$	$w_{3N_{l-1}}$	...	$w_{N_lN_{l-1}}$

FIG. 10

		Node Index				
		1	2	3	...	N_l
Edge Index	1	q_{11}	q_{21}	q_{31}	...	q_{N_l1}
	2	q_{12}	q_{22}	q_{32}	...	q_{N_l2}
	3	q_{13}	q_{23}	q_{33}	...	q_{N_l3}
	$\vdots$	$\vdots$	$\vdots$	$\vdots$	...	$\vdots$
	N_{l-1}	$q_{1N_{l-1}}$	$q_{2N_{l-1}}$	$q_{3N_{l-1}}$	...	$q_{N_lN_{l-1}}$

FIG. 11

		Kernel Index				
		1	2	3	...	M_l
Edge Index	1	w_{11}	w_{21}	w_{31}	...	$w_{M_l 1}$
	2	w_{12}	w_{22}	w_{32}	...	$w_{M_l 2}$
	3	w_{13}	w_{23}	w_{33}	...	$w_{M_l 3}$
	$\vdots$	$\vdots$	$\vdots$	$\vdots$	...	$\vdots$
	K_l	w_{1K_l}	w_{2K_l}	w_{3K_l}	...	$w_{M_l K_l}$

FIG. 12

		Kernel Index				
		1	2	3	...	M_l
Edge Index	1	q_{11}	q_{21}	q_{31}	...	$q_{M_l 1}$
	2	q_{12}	q_{22}	q_{32}	...	$q_{M_l 2}$
	3	q_{13}	q_{23}	q_{33}	...	$q_{M_l 3}$
	$\vdots$	$\vdots$	$\vdots$	$\vdots$	...	$\vdots$
	K_l	q_{1K_l}	q_{2K_l}	q_{3K_l}	...	$q_{M_l K_l}$

FIG. 13

```

quant_coding() {
    quant_enable_flag
    if (quant_enable_flag) {
        layer_adaptive_quant_flag
        if (layer_adaptive_quant_flag) {
            matrix_adaptive_quant_flag
            if (matrix_adaptive_quant_flag) {
                for (i = 1; i < L; i++)
                    matrix_quant_coding(i)
            } else {
                layer_quant_coding()
            }
        } else {
            fixed_quant_step
        }
    }
}

```

FIG. 14

```

matrix_quant_coding(layer_id) {
    for (j = 0; j < E; j++)
        base_quant_step[j]
    for (i = 1; i < C; i++) {
        prev_quant_copy_flag[i-1]
        if (!prev_quant_copy_flag[i-1]) {
            for (j = 0; j < E; j++)
                diff_quant_value[i-1][j]
        }
    }
}

```

FIG. 15

```

layer_quant_coding() {
    base_layer_quant_step
    for (i = 2; i < L; i++) {
        layer_quant_copy_flag[i-2]
        if (!layer_quant_copy_flag[i-2])
            layer_quant_step[i-2]
    }
}

```