Note on Chapter 2 of [Manber]: Proving a Loop Invariant

Below is the program given in Chapter 2 (Page 27) of Manber's book that converts a decimal number into a binary one:

Algorithm Convert_to_Binary (n);
begin
    $t := n$;
    $k := 0$;
    while $t > 0$ **do**
        $k := k + 1$;
        $b[k] := t \bmod 2$;
        $t := t \text{ div } 2$;
end

Let $Inv(n, t, k, b)$ denote the following assertion:

$n = t \times 2^k + m$ and $t \geq 0$,
where m is the binary number represented by b, i.e.,

$$m = \begin{cases} 0 & \text{if } k = 0 \\ b[k] \times 2^{k-1} + b[k-1] \times 2^{k-2} + \cdots + b[1] \times 2^0 & \text{if } k \geq 1 \end{cases}$$

Claim: $Inv(n, t, k, b)$ is a loop invariant of the while loop, assuming that the decimal number passed via variable n is non-negative. (The invariant is sufficient to deduce that, when the program terminates, b stores the binary representation of n.)

Proof: The proof is by induction on the number of times the loop body is executed. More specifically, we show that (1) the assertion is true when the flow of control reaches the loop for the first time and (2) given that the assertion is true and the loop condition holds, the assertion will remain true after the next iteration (i.e., after the loop body is executed once more).

(1) When the flow of control reaches the loop for the first time, $t = n$ (≥ 0) and $k = 0$. With m denoting the binary number represented by b, $t \times 2^k + m = n \times 2^0 + 0 = n$ and $t \geq 0$. Therefore, the assertion $Inv(n, t, k, b)$ holds.

(2) Assume that $Inv(n, t, k, b)$ is true at the start of the next iteration and the loop condition $(t > 0)$ holds. Let n', t', k', and b' denote respectively the values of n, t, k, and b after the next iteration. We need to show that $Inv(n', t', k', b')$ also holds.

From the loop body, we deduce the following relationship:

$k' = k + 1$
$b'[k'] = t \bmod 2$
$b'[i] = b[i]$ for all $i \neq k'$
$t' = t$ div 2
$n' = n$ (the value of n never changes)

There are two cases to consider: when t is odd and when t is even. We prove the first case; the second case can be proven similarly. If t is odd, $t \bmod 2$ contributes 1 to $b'[k']$. The binary number m' represented by b' equals $b'[k'] \times 2^{k'-1} + b'[k'-1] \times 2^{k'-2} + \cdots + b'[1] \times 2^0 = 2^{k'-1} + b[k] \times 2^{k-1} + b[k-1] \times 2^{k-2} + \cdots + b[1] \times 2^0 = 2^k + m$. We then have $t' \times 2^{k'} + m' = \frac{t-1}{2} \times 2^{k+1} + 2^k + m = (t-1) \times 2^k + 2^k + m = t \times 2^k + m = n = n'$. In addition, since $t > 0$ (given that the loop condition holds), $t' = t$ div $2 \geq 0$. Therefore, $Inv(n', t', k', b')$ holds after the next iteration in the case of odd t.