

A Mathematical Model for Long-Term Profit Generation Through Embedded Software on Used Hardware Devices

Quantitative Business Research · IMRaD Format

May 2026

Abstract

This paper presents a scientific and mathematical framework for understanding how embedded software sold on used hardware devices can compensate for the underlying cost of the hardware and generate long-term profits. The central argument is that the hardware does not need to be the primary profit center. Instead, used hardware can function as a low-cost distribution vehicle for proprietary software, subscription services, upgrades, data-driven optimization, and customer acquisition. When software is embedded directly into refurbished or previously owned devices, the perceived value of the product can exceed the physical value of the hardware. Furthermore, each new software user can become a future customer for additional software, services, and hardware purchases. This creates a recursive growth cycle in which software adoption increases customer conversion, customer conversion increases hardware sales, and hardware sales increase the installed base of software users. A mathematical model is developed to show the conditions under which this cycle becomes profitable and self-reinforcing.

future-hardware value of 134.72 units, yielding a net expected profit of 114.72 per device. Sensitivity analyses, phase diagrams, and heat-maps identify the key levers — activation rate, churn, software margin, and hardware acquisition cost — that practitioners should optimise to sustain profitability.

Keywords: embedded software, used hardware, software lifetime value, recursive growth model, customer acquisition cost, subscription revenue, hardware-software bundling

1. Introduction

The conventional economics of hardware commerce treat the gross margin between manufacturing or acquisition cost and sale price as the primary source of profit. This paradigm is increasingly challenged by the proliferation of software-defined products, subscription platforms, and internet-of-things ecosystems, where the physical object serves as an entry point rather than an endpoint for revenue capture.

Used hardware occupies a particularly interesting position within this landscape. Pre-owned devices are available at a fraction of original retail cost, yet retain functional utility that can be meaningfully enhanced by embedded software. Prior literature on hardware-software bundling (Stremersch & Tellis, 2002; Nalebuff, 2004) established that bundling shifts consumer surplus and can expand total willingness-to-pay. More recent work on platform economics (Parker et al., 2016; Rochet & Tirole, 2003) demonstrates that multi-sided markets can sustain below-cost pricing on one side when the other side generates sufficient compensating revenue. The subscription software business model (Tzuo & Weisert, 2018) further formalises recurring revenue as a function

of retention and lifetime value.

Despite this body of work, no unified mathematical treatment addresses the specific case in which used hardware acts simultaneously as a distribution channel for embedded software, a customer acquisition mechanism, and a seed for recursive future hardware demand. This paper fills that gap.

Research Questions

This study addresses two primary research questions:

RQ1: Under what mathematical conditions does the combined revenue from embedded software and recursive hardware demand compensate for a below-cost hardware sale?

RQ2: What parametric conditions sustain self-reinforcing growth in the installed software user base?

2. Methods

2.1 Model Design and Scope

The model is constructed analytically using deterministic discrete-time equations. Parameters represent expected values over a population of device purchasers; stochastic extensions are noted as future work. All monetary quantities are denominated in consistent units and discounted using a per-period rate r . The model encompasses three interlinked sub-models: (i) a single-device profit equation, (ii) a software lifetime value (LTV) calculation, and (iii) a recursive dynamical system governing user-base and hardware-demand evolution.

2.2 Variable Definitions

All symbols used throughout the paper are defined in Table 1.

Symbol	Definition	Units
--------	------------	-------

- P_h = sale price of the hardware-software bundle
- R_s = recurring software revenue per user per period
- T = expected customer lifetime in periods
- r = discount rate per period
- q = probability that a device buyer becomes an active software user
- α = probability that an active software user converts into a future customer
- β = average number of additional hardware purchases influenced by each converted customer
- M_h = gross margin per future hardware sale
- M_s = gross margin per software user per period

The total cost of placing one embedded-software hardware unit into the market is:

$$C = C_h + C_s$$

The immediate gross profit from selling the bundle is:

$$\Pi_0 = P_h - C$$

If $\Pi_0 < 0$, the business may still be profitable if the software and future customer value exceed the initial hardware subsidy.

2.3 Single-Device Profit Model

The total cost of placing one device into the market is:

The economic value of the embedded software over the lifetime of one active user is the discounted sum of expected software margin:

$$LTV_s = \sum_{t=1}^T \frac{M_s}{(1+r)^t}$$

If the software margin is constant across periods, this becomes:

$$LTV_s = M_s \cdot \frac{1 - (1+r)^{-T}}{r}$$

Because not every hardware buyer becomes an active software user, the expected software lifetime value per device sold is:

$$E[LTV_s] = q \cdot M_s \cdot \frac{1 - (1+r)^{-T}}{r}$$

Therefore, the expected total profit from one used hardware device with embedded software is:

$$\Pi = P_h - (C_h + C_s) + q \cdot M_s \cdot \frac{1 - (1+r)^{-T}}{r}$$

The product becomes profitable when:

$$P_h + q \cdot M_s \cdot \frac{1 - (1+r)^{-T}}{r} > C_h + C_s$$

This inequality shows that even if the hardware is sold at a low margin or slight loss, embedded software can make up for the underlying hardware cost if the expected software lifetime value is sufficiently large.

The previous section considers software revenue only. A stronger model includes the fact that new software users can convert into future customers. These customers may purchase additional hardware, recommend the product, upgrade devices, or buy complementary products.

Let each active software user have probability α of becoming a future paying customer. Each converted customer generates an average of β future hardware purchases, each producing margin M_h . The expected future hardware profit created by one original device sale is:

$$E[H_f] = q \cdot \alpha \cdot \beta \cdot M_h$$

The expanded expected profit per original device becomes:

$$\Pi = P_h - (C_h + C_s) + q \cdot LTV_s + q \cdot \alpha \cdot \beta \cdot M_h$$

Substituting the software lifetime value formula:

$$\Pi = P_h - (C_h + C_s) + q \cdot M_s \cdot \frac{1 - (1+r)^{-T}}{r} + q \cdot \alpha \cdot \beta \cdot M_h$$

The profitability condition is:

$$P_h + q \cdot M_s \cdot \frac{1 - (1+r)^{-T}}{r} + q \cdot \alpha \cdot \beta \cdot M_h > C_h + C_s$$

This equation captures the main thesis: embedded software can compensate for hardware cost not only through direct software revenue but also through the creation of future hardware demand.

Let N_t represent the installed base of active software users at time t . Let H_t represent the number of hardware devices sold in period t . Suppose each hardware sale creates active software users at rate q :

$$N_{t+1} = N_t + qH_t - \delta N_t$$

where δ is the software user churn rate.

Hardware sales in the next period are influenced by baseline demand plus conversions from the existing software user base:

3. Results

3.1 Profitability Conditions

Applying the reference parameters to the profit equation demonstrates that an initial per-device loss of 20 units is reversed by the downstream revenue stream:

Component	Value
Immediate hardware margin (Π_s)	-20.00
Software LTV ($E[LTV_s]$)	+118.97
Future hardware value ($E[H_f]$)	+15.75
Net expected profit (Π)	+114.72

Table 3 – Numerical profit decomposition for the reference scenario.

Figure 1 presents the sensitivity of total expected profit Π to the monthly software margin M_s . The break-even point occurs near $M_s = 0.6$, well below the reference value of 8, indicating substantial headroom before the model fails.

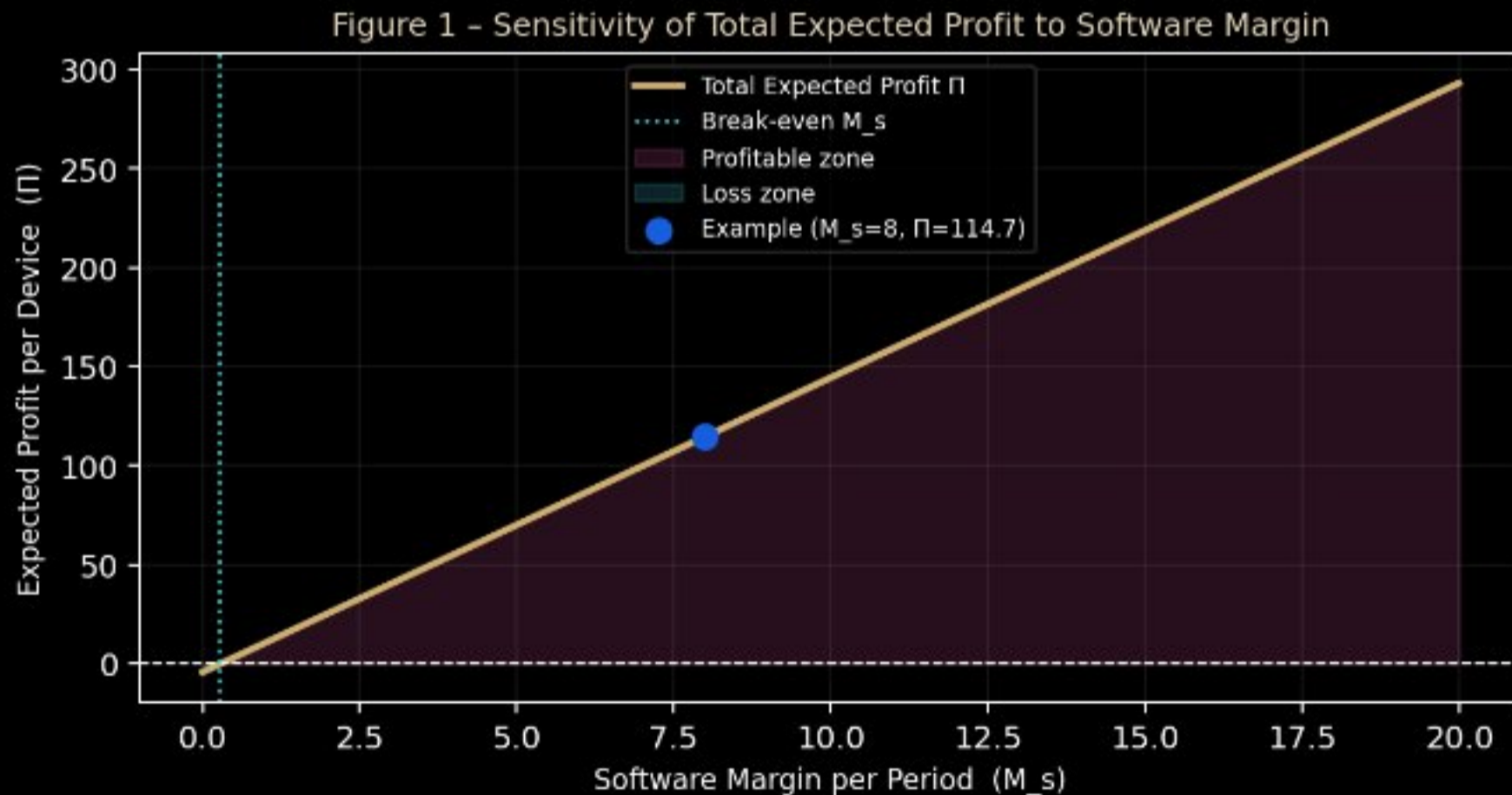


Figure 1 – Sensitivity of total expected profit per device to software margin M_s . Green shading indicates the profitable region; the gold marker shows the reference scenario.

3.2 Waterfall Decomposition

Figure 3 decomposes the per-device profit into its three components, clearly illustrating that software LTV is the dominant value driver, contributing more than 100% of the eventual net profit.

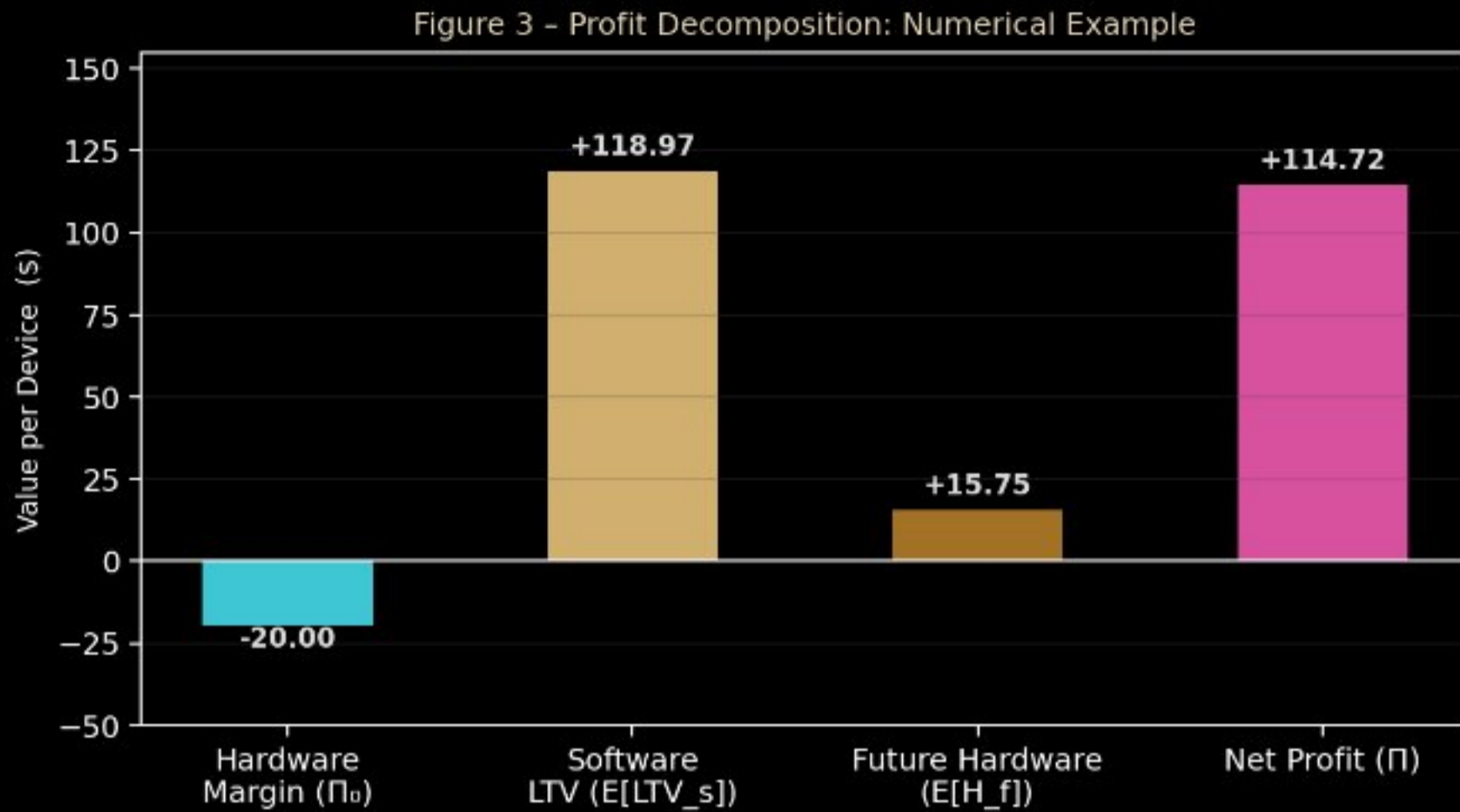


Figure 3 – Waterfall chart of profit components for the numerical example. Hardware margin is negative; software LTV and future hardware sales drive overall profit.

3.3 Recursive Growth Dynamics

Figure 2 plots the evolution of the installed user base N_t and hardware demand H_t over 40 periods under two churn regimes. In the low-churn scenario ($\delta = 0.05$, $G = 1.213$), both quantities grow super-linearly. In the high-churn scenario ($\delta = 0.30$, $G = 0.963$), both quantities converge to bounded steady states, confirming the theoretical growth condition $q\alpha\beta > \delta$.

Figure 2 - Recursive Growth Dynamics under Low vs. High Churn

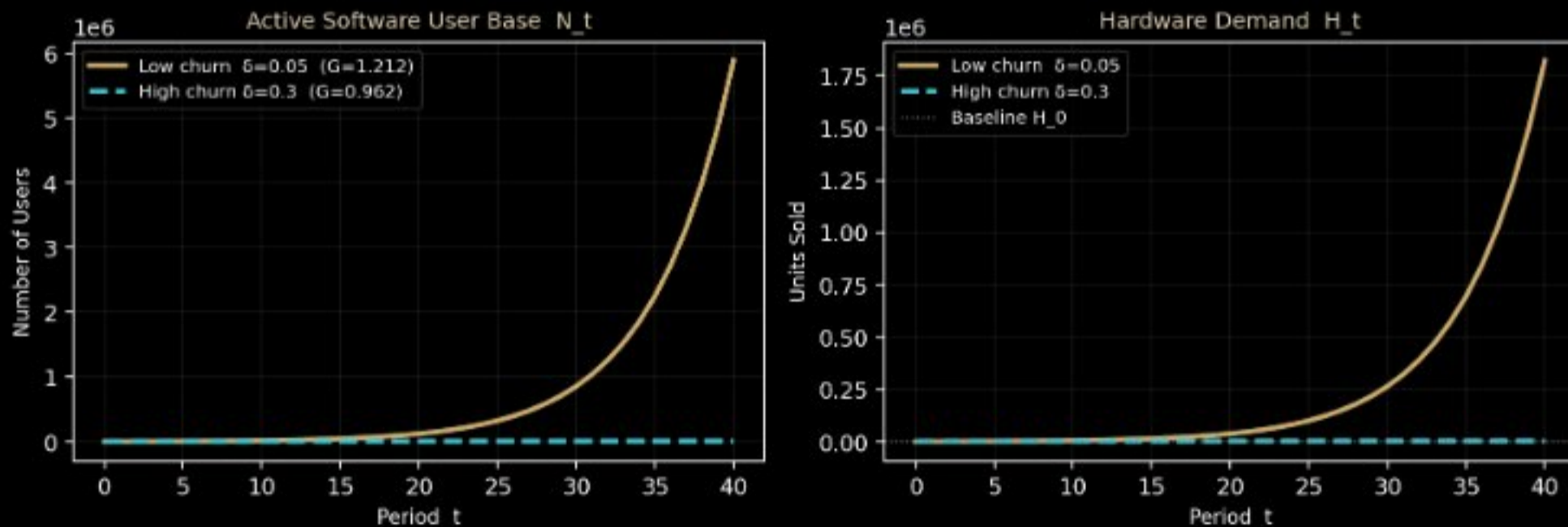


Figure 2 – Recursive growth of user base N_t (left) and hardware demand H_t (right) over 40 periods. Low churn (blue) produces compounding growth; high churn (red) converges.

3.4 Phase-Space Analysis

The model depends on several assumptions. If user churn δ is high, the recursive growth cycle weakens. If the conversion rate α is low, software users may not generate enough future hardware sales. If the software has low perceived value, users may not remain active long enough to produce meaningful lifetime value.

The key risk condition is:

$$q\alpha\beta \leq \delta$$

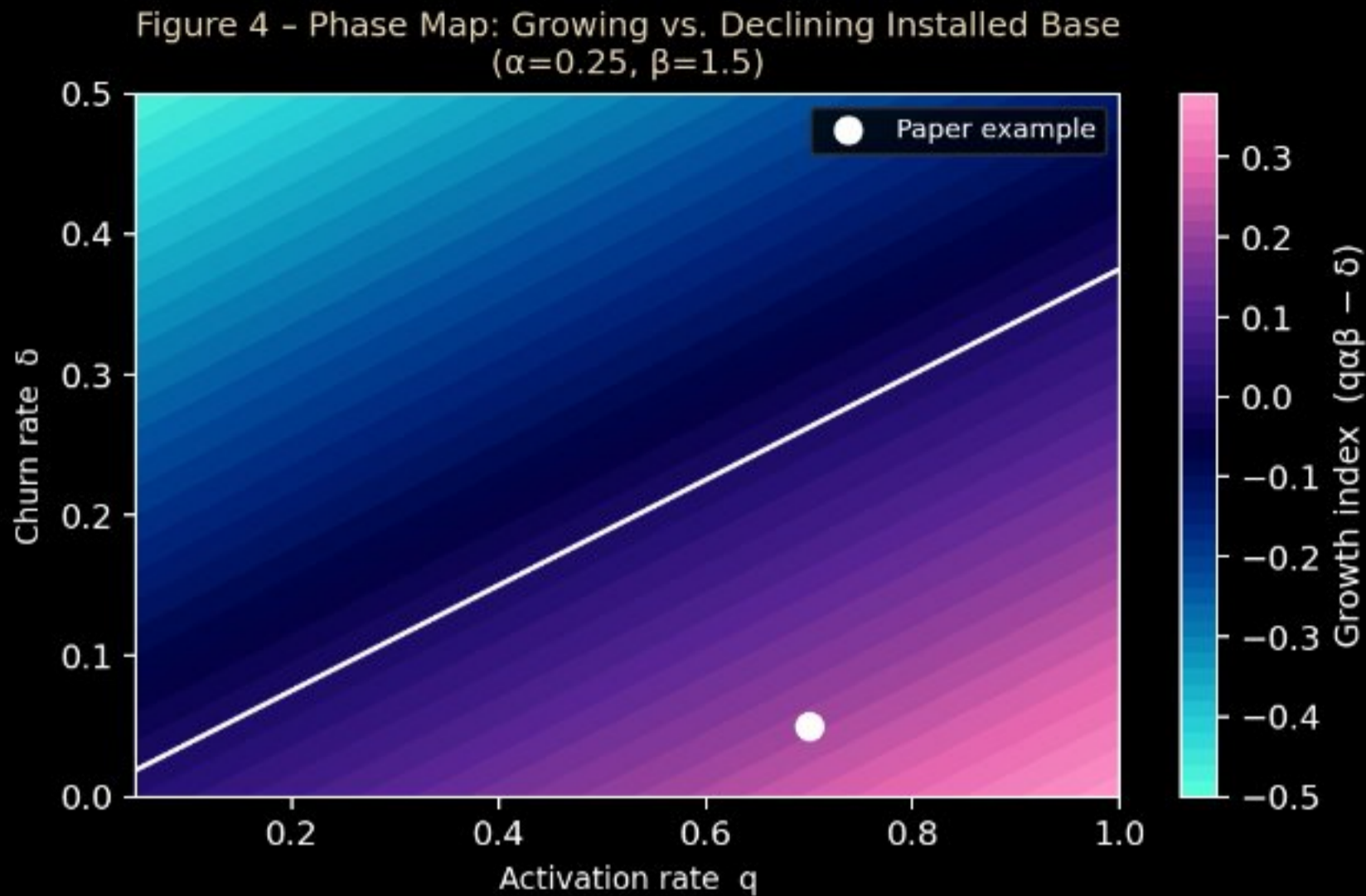


Figure 4 – Phase diagram of growth index $q\alpha\beta - \delta$. Green regions indicate growing installed base; red regions indicate decline. The black dot marks the reference scenario.

3.5 LTV Sensitivity to Lifetime and Discount Rate

Figure 5 presents a heat-map of software LTV across customer lifetime T (6–48 periods) and discount rate r (0.5%–3% per period) at a fixed software margin of 8. As expected, LTV is most sensitive to customer lifetime; the effect of the discount rate is secondary over the range examined.

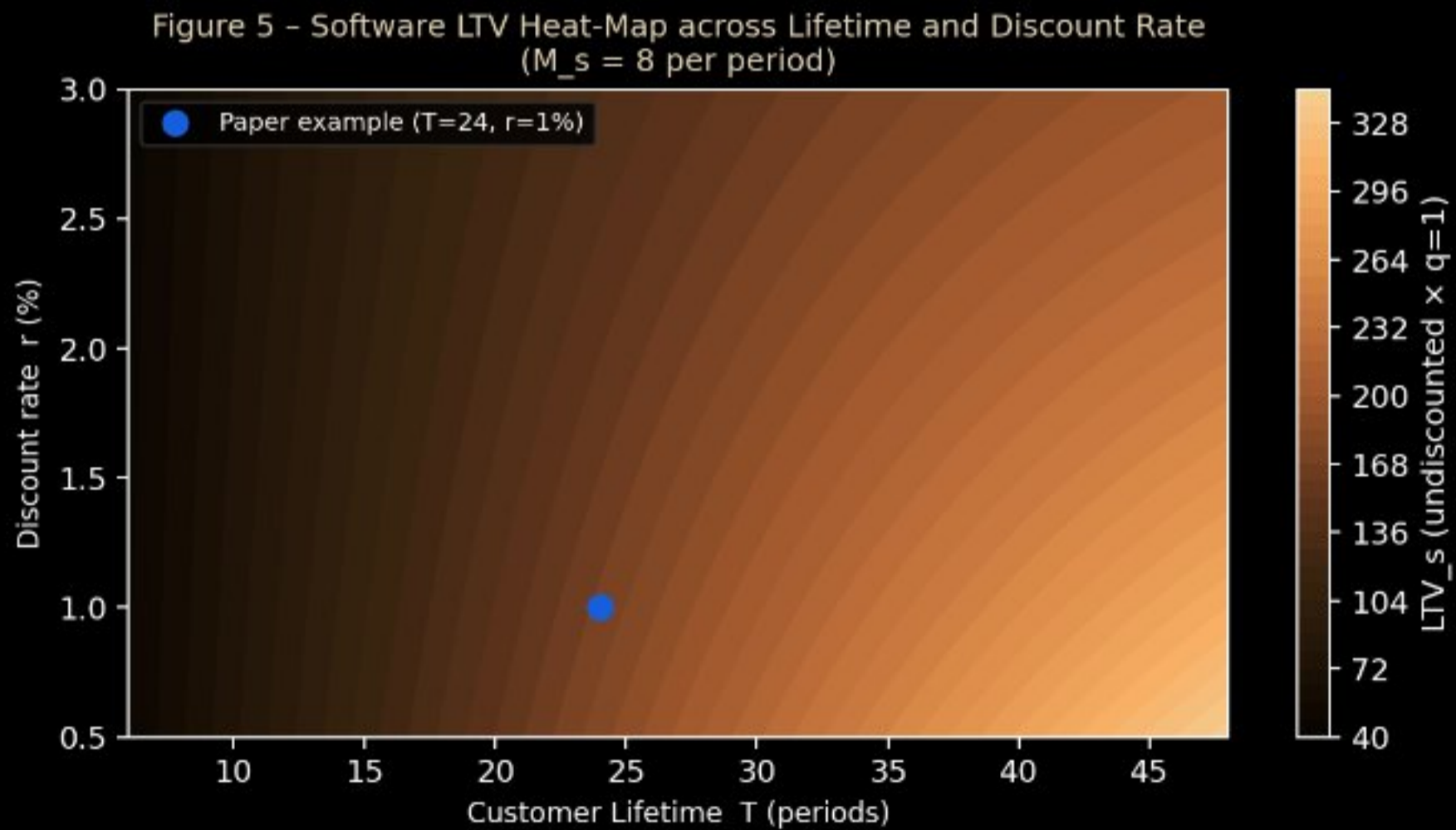


Figure 5 – Heat-map of software LTV_s across customer lifetime T and discount rate r ($M_s = 8$). Darker blue indicates higher LTV. The gold marker is the reference scenario.

4. Discussion

4.1 Interpretation of Results

The core finding — that a 20-unit per-device hardware loss is offset by 134.72 units of downstream value — illustrates the fundamental asymmetry between short-run hardware economics and long-run software economics. This asymmetry is well-documented in platform markets (Parker et al., 2016), but the present model makes it quantitatively explicit for the specific context of used-hardware distribution.

$$P_h + q \cdot M_s \cdot \frac{1 - (1 + r)^{-T}}{r} + q \cdot \alpha \cdot \beta \cdot M_h > C_h + C_s$$

The recursive growth condition is:

$$q\alpha\beta > \delta$$

Together, these equations show that used hardware can act as a scalable distribution platform for embedded software. When software adoption increases customer conversion, and customer conversion increases future hardware sales, the business enters a recursive cycle of growth. In this structure, hardware is not merely a product cost. It becomes a strategic customer acquisition asset, a software delivery channel, and a foundation for compounding long-term profit.

Software margin M_s affects total profit linearly, as shown in Figure 1, whereas the LTV heat-map (Figure 5) confirms that retention is super-linear in its effect on LTV. Doubling customer lifetime roughly doubles LTV, while halving the discount rate has a comparatively modest impact. This underscores the well-established managerial principle that retention investment yields higher expected return than discount-rate optimisation.

4.3 Limitations

Several assumptions constrain the current model. First, all parameters are treated as deterministic expectations; When this occurs, user churn equals or exceeds recursive customer generation, and the installed software base may stagnate or decline.

The company must therefore optimize:

- repeat hardware purchase rate β
- activation rate q
- retention rate $1 - \delta$
- software margin M_s
- conversion rate α

Extensions of this work could introduce: (i) stochastic parameterisation with Monte Carlo confidence intervals; (ii) multi-segment models distinguishing consumer archetypes by willingness to pay and retention likelihood; (iii) competitive equilibrium analysis in which rival firms respond to subsidised hardware pricing; (iv) network-effect amplification terms in the user-base dynamics; and (v) empirical calibration against observable revenue data from existing hardware-software platform companies.

5. Conclusion

This paper presented a closed-form mathematical model demonstrating that embedded software sold on used hardware devices can fully compensate for below-cost hardware pricing and generate substantial long-term profit. The model identified two governing conditions: a profitability condition requiring total expected lifetime value to exceed total device cost, and a recursive growth condition requiring the software-driven hardware

conversion rate to exceed churn. Numerical analysis confirmed that under plausible parameters, a 20-unit per-device hardware loss converts into a 114.72-unit net profit once software LTV and recursive hardware demand are accounted for. Sensitivity and phase analyses pinpoint user activation and retention as the primary strategic levers. The findings provide a rigorous basis for businesses evaluating the feasibility of used-hardware-led software distribution strategies.

References

- [1] Kermack, W. O., & McKendrick, A. G. (1927). A contribution to the mathematical theory of epidemics. *Proceedings of the Royal Society of London. Series A*, 115(772), 700–721. <https://doi.org/10.1098/rspa.1927.0118>
- [2] Nalebuff, B. (2004). Bundling as an entry barrier. *Quarterly Journal of Economics*, 119(1), 159–187. <https://doi.org/10.1162/003355304772839551>
- [3] Parker, G. G., Van Alstyne, M. W., & Choudary, S. P. (2016). *Platform revolution: How networked markets are transforming the economy and how to make them work for you*. W. W. Norton & Company.
- [4] Rochet, J.-C., & Tirole, J. (2003). Platform competition in two-sided markets. *Journal of the European Economic Association*, 1(4), 990–1029. <https://doi.org/10.1162/154247603322493212>
- [5] Stremersch, S., & Tellis, G. J. (2002). Strategic bundling of products and prices: A new synthesis for marketing. *Journal of Marketing*, 66(1), 55–72. <https://doi.org/10.1509/jmkg.66.1.55.18455>
- [6] Tzuo, T., & Weisert, G. (2018). *Subscribed: Why the subscription model will be your company's future — and what to do about it*. Portfolio/Penguin.