2024 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)
July 15-19, 2024. Boston, MA, USA

Blockchain-Enabled Automated Recipe Management System for Semiconductor Production

Rick Lin[1], Verna Fu[2], Iuon-Chang Lin[3] , and Pai Ching Tseng[4]
Morrison Academy, Taichung, Taiwan[1], Taipei European School (TES),Taipei, Taiwan [2], National Chung Hsing University, Taichung Taiwan (iclin@nchu.edu.tw)[3], and Feng Chia University, Taichung, Taiwan (tpcp630@gmail.com)[4]

***Abstract*—The main purpose of this paper is to focus on the semiconductor production process. Using blockchain technology, a secure and tamper-proof recipe transmission process is designed to ensure that the recipe consistently maintains the experimental results after being uploaded by process engineers, reducing the occurrence of production errors caused by personnel mistakenly modifying recipes. This paper proposes incorporating secure hashing algorithms combined with blockchain into the recipe management system, updating blockchain data simultaneously when uploading recipes, and performing computations during the download process. By comparing the current recipe with the records in the blockchain and using binary tree algorithms to obtain a list of differential files, differential downloading is performed to reduce file transmission time and accelerate the production process.**

***Index Terms*—Automated transport systems, Semiconductor manufacturing, Recipe management system, Blockchain**

I. INTRODUCTION

Through the design proposed in this paper, utilizing pre-order binary tree traversal, the machine and RMS (Recipe Management System) are scanned to identify recipe files with differing SHA hash values, enabling a process of differentially downloading these files. Additionally, the concept of blockchain is integrated to ensure data integrity. The anticipated benefits vary depending on whether a single recipe or a hierarchical structure of recipes is implemented with this design architecture. For a single recipe, the efficiency gain is contingent upon the file size of the main recipe. In hierarchical scenarios, significant savings in download time are achieved since not all recipe files need to be downloaded.

2. SOLUTIONS

This paper effectively addresses the following issues:
1. When some machines use a recipe, the last modification time of the recipe changes, resulting in different MD5 hash results between the machine and server ends.
Solution: By systematically traversing each recipe file, commonly altered main recipes are identified. Through differentially downloading, only the main recipe undergoes re-downloading, thus saving download time for the remaining sub-recipes.
2. Recipe management is internal to the factory, requiring remote downloading for external or partner factories. If the recipe is large, the download process is time-consuming.
Solution: By comparing and identifying altered files through systematic file traversal, differentially downloading involves re-downloading only the recipe files listed in the difference list, thus saving download time for other recipes.

According to the design of this study, as shown in Fig. 1, the comparison method is changed from RMS system comparing MD5 hashes to comparing SHA differences in the blockchain. If differences are found, they are identified using a binary tree to download the differential files, aiming to reduce the time required for complete download coverage.

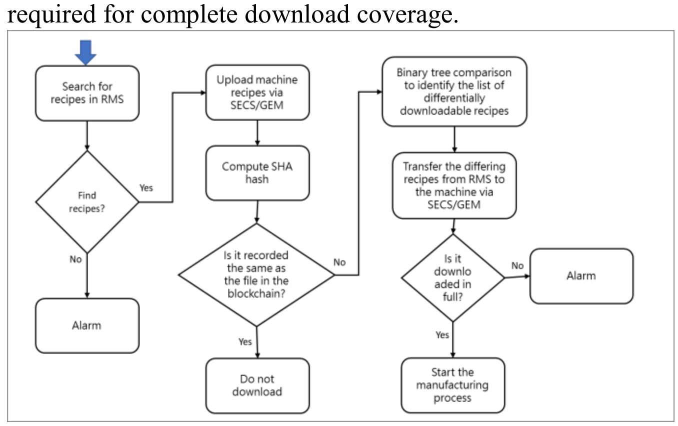

Fig. 1. Diagram of the recipe download process integrated with blockchain.

3.RESULTS

Tab. 1 presents the time spent for a single download of hierarchical recipes. In network environment A, the total time saved for downloading files on over 200 machines is 65,616.34 seconds, equivalent to about 18.23 hours. In network environment B, the total time saved is 6,553.92 seconds, equivalent to about 1.82 hours. This significant time reduction contributes to the shortened process time after updating the recipe.

Tab 1. A comparison between our proposed approach and the traditional method

Hierarchical recipe		
Network environment	traditional method	our proposed approach
	Download time(Sec)	Download time(Sec)
100Mbps/100Mbps	344.064	Main Recipe- 0.0039 Sub-Recipe 1 - 0.0117 Sub-Recipe 2 - 327.68
1000Mbps/600Mbps	34.4064	Main Recipe- 0.0004 Sub-Recipe 1 - 0.0012 Sub-Recipe 2 - 32.768