Improved file sizes and cycle times through optimization of GDSII Stream

Chin Le, Micron Technologies, Inc.; David Gariepy, Saratoga Data Systems, Inc.

ABSTRACT

Filling for the prevention of CMP dishing and resolution enhancement technologies (OPC, PSM) can cause the size of IC designs represented in the popular GDSII Stream format to balloon by a factor of ten or more, resulting file sizes of tens of gigabytes and longer throughput times for the tools that must subsequently process the files.

We describe the effects of optimizing GDSII Stream files on the tape-out flow. GDSII Stream file sizes can be reduced by as much as 95% (20X reduction) and subsequent tool throughput improved by factors of up to five (5X runtime improvement).

Keywords: GDSII Stream, optimization, compression, tape-out

1. INTRODUCTION

Having been designed nearly thirty years ago, the GDSII Stream format serves as a ubiquitous, defacto industry standard file format for the interchange of IC layout data. GDSII Stream files representing modern IC designs can typically become tens of gigabytes in size. Such large GDSII Stream files are unwieldy to store, time consuming to transfer, and adversely affect the throughput time of the downstream tools that must process them.

GDSII Stream users typically combat the file size by using common compression utilities (e.g., gzip). Many common GDSII Stream processing tools read and write compressed GDSII Stream files. Nonetheless, it is widely accepted that a more efficient file format is needed to replace GDSII Stream.

1.1. Optimization

Saratoga Data Systems, Inc. has developed a program, Bantam, for optimizing GDSII Stream files to be more space efficient. Bantam finds repetition and, while still using valid GDSII Stream format constructs, represents these repetitive elements in a more space efficient manner. Bantam reads a GDSII Stream file, optimizes its contents, and writes a significantly smaller, but 100% point-for-point functionally equivalent, GDSII Stream file.

1.2. Not compression

Bantam is optimization, not compression. No compression of any kind is used in a Bantam output GDSII Stream file. Bantam optimized GDSII Stream files are readable by any tool that reads the GDSII Stream file format. In fact, standard compression may be applied to Bantam optimized files for a multiplicative effect on reducing total file size.

1.3. Advantages of optimization over compression

A compressed GDSII Stream file requires decompression prior to use. The tool processing the file still reads and processes the original large volume of GDSII Stream data.

Additionally, decompression requires processing time, adversely affecting tool throughput.

Tools processing optimized GDSII Stream files simply read less, more efficiently organized, data, with no preprocessing penalty for decompression. And, these tools may reap additional benefit from the efficient organization of an optimized GDSII Stream file: they may actually perform their operations faster than with non-optimized data.

2. OPTIMIZATION EFFECT ON FILE SIZE

We used Saratoga Data Systems' Bantam GDSII Stream Optimization software on GDSII Stream files over a wide range of sizes in an attempt to characterize its effect on file size.

2.1. Optimization test results

Table 1 lists the results of optimization applied to 25 GDSII Stream files ranging in size from 28 MB to 68 GB. These files were largely chosen at random, based on their size and availability. They represent a variety of design styles. Many are a subset, or a single layer, of an entire design.

Original GDSII Stream File Size (bytes)	Optimized GDSII Stream File Size (bytes)	Percent Effective
28,868,608	20,736,220	28.2%
127,000,000	300,000	99.8%
261,091,328	138,330,100	47.0%
290,138,112	134,414,114	53.7%
318,441,472	157,533,350	50.5%
436,822,016	145,102,348	66.8%
461,236,224	186,944,846	59.5%
462,247,936	162,559,822	64.8%
791,000,000	12,000,000	98.5%
1,110,000,000	33,000,000	97.0%
1,194,686,464	278,857,472	76.7%
1,819,920,384	315,818,754	82.6%
1,819,944,960	316,158,982	82.6%
2,122,000,000	153,000,000	92.8%
2,974,000,000	328,000,000	89.0%
3,183,000,000	871,000,000	72.6%
3,365,892,096	625,960,058	81.4%
3,907,616,768	659,945,996	83.1%
4,877,000,000	26,000,000	99.5%
5,008,000,000	27,000,000	99.5%
7,046,074,368	2,285,012,992	67.6%
7,397,636,096	101,149,722	98.6%
14,039,000,000	99,000,000	99.3%
16,999,444,480	875,823,476	94.8%
68,368,273,408	607,598,592	99.1%

Table 1: Effect of Optimization on GDSII File Size For a Range of File Sizes

The calculation of effectiveness is the percentage of the original file size that has been obviated, or "optimized away". It is calculated as: (original file size – optimized file size)*100/original file size. For example, if the effectiveness for a particular file is 90%, then optimization obviated 90% of the original file and the optimized file is 10% the size of the original file.

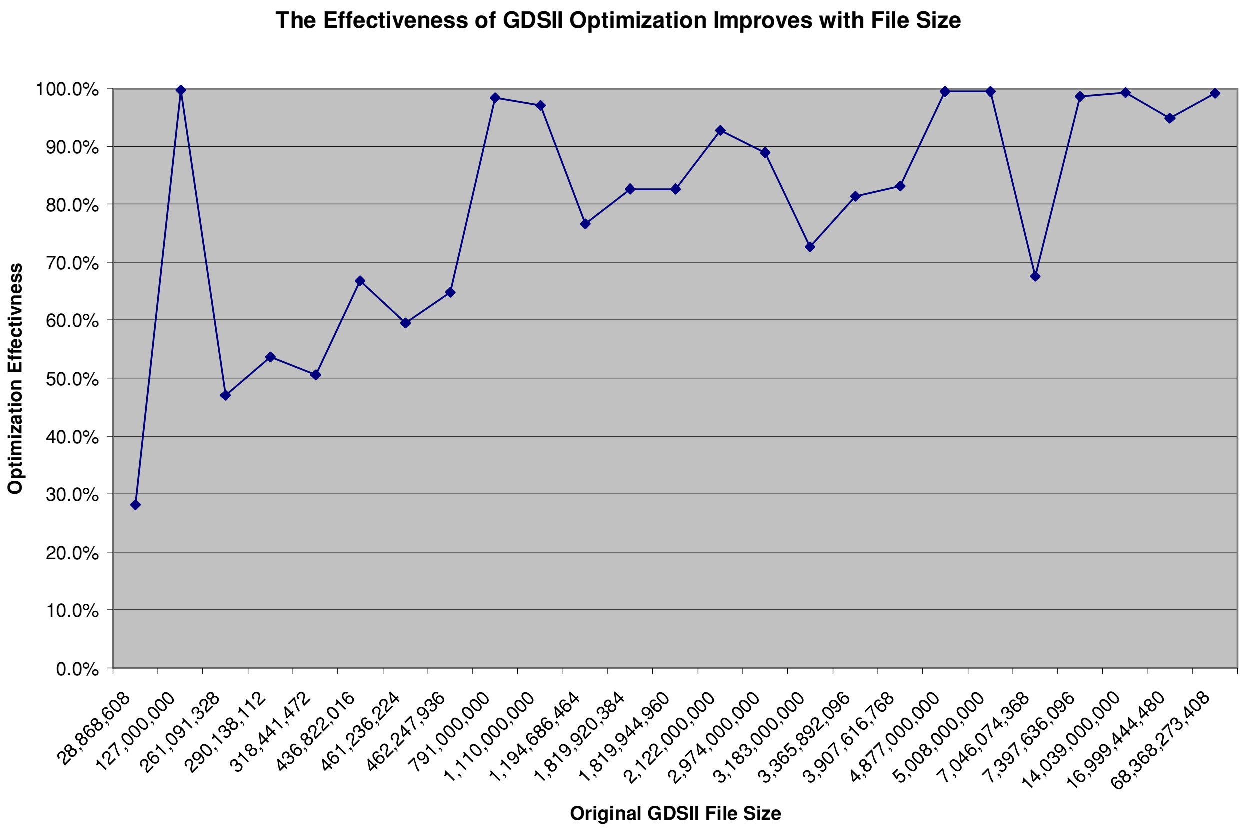

Figure 1: Optimization Effectiveness on Reducing GDSII Stream File Size. Data is from **Table 1**.

2.2. Observations

From **Figure 1** we observe that the larger the GDSII Stream file, the more effective optimization becomes in reducing file size. Optimization reaches 90% effectiveness in reducing the size of GDSII Stream files at approximately 2GB and above. Optimization becomes very effective in reducing GDSII Stream file size when the files are very large, tens of gigabytes.

GDSII Stream files that represent simple design intent are rarely larger than ten gigabytes, most being smaller than two gigabytes. GDSII Stream files larger than two gigabytes are typically produced during the tape-out process just prior to manufacturing. Modern fabrication processes require a host of embellishments to the design layout data, including layer filling to prevent dishing during Chemical Mechanical Polishing (CMP) and Optical Proximity Correction (OPC) to enhance resolution. Each of these manufacturing embellishments adds a significant volume of data to a design, and to its GDSII Stream file(s) representation. In fact, all the GDSII Stream files greater than two gigabytes in size listed in **Table 1** have been embellished with filling and OPC.

Also note, the file size reduction effectiveness of GDSII Stream optimization seems to be largely independent of design style. ASIC's, SOC's, analog designs, memories, and "what-have-you" design styles all seem to have much less influence on file size reduction effectiveness than simply the size of the original GDSII Stream file. And, again, a GDSII Stream file's size is most strongly influenced by how much it has been embellished with the necessities of manufacturing like filling and OPC.

Figure 2 illustrates a modern generic tape-out flow. Notice the relative size of the GDSII Stream file balloons after each tape-out embellishment operation.

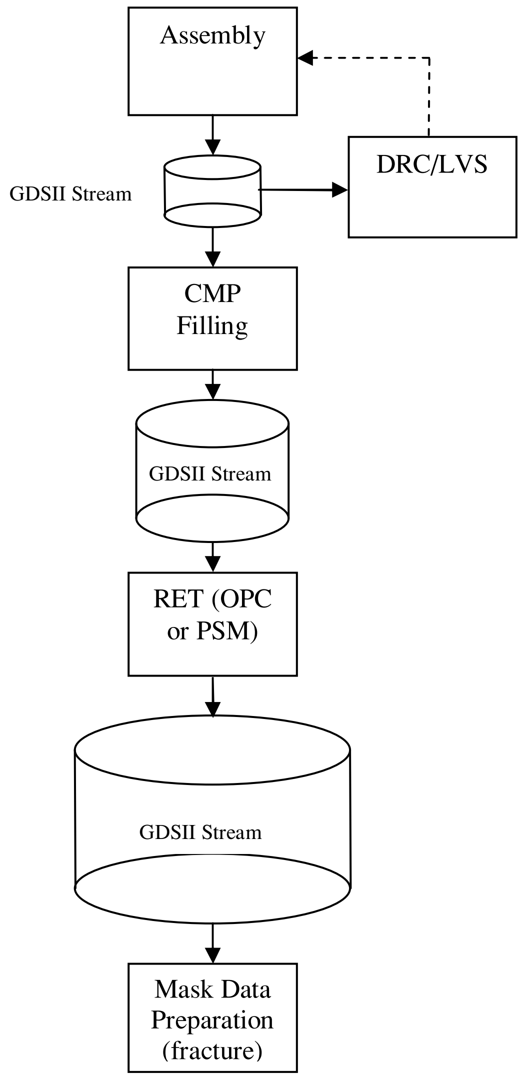

Figure 2: Generic GDSII Stream-based tape-out flow

2.3. Accuracy

All test cases cited here were verified accurate with a variety of commercially available XOR operations. In all cases the XOR operation verified that the Bantam optimized GDSII Stream file was exactly equivalent to the input GDSII Stream file.

2.4. Compressing optimized GDSII Stream files

Gzip was applied to several optimized GDSII Stream files to observe if optimization had any significant affect on the effectiveness of gzip compression. **Table 2** summarizes the results of applying both optimization and gzip compression to GDSII Stream files over a range of sizes.

Original GDSII File Size (bytes)	gzip'd Original GDSII File Size (bytes)	gzip Effective-ness	Optimized GDSII File Size (bytes)	Optimizat-ion Effective-ness	gzip'd Optimized GDSII File Size (bytes)	Optimized + gzip Effective-ness	gzip Effective-ness on Optimized Files
1,194,686,464	309,137,493	74.1%	278,857,472	76.7%	64,196,089	94.6%	77%
1,819,920,384	475,154,775	73.9%	315,818,754	82.6%	70,690,053	96.1%	78%
1,819,944,960	471,415,607	74.1%	316,158,982	82.6%	70,967,424	96.1%	78%
2,114,351,104	427,745,860	79.8%	902,801,556	57.3%	139,414,792	93.4%	85%
2,671,089,664	491,924,480	81.6%	609,905,986	77.2%	187,470,267	93.0%	69%
2,850,000,000	961,000,000	66.3%	201,000,000	92.9%	85,500,000	97.0%	57%
3,365,892,096	742,387,712	77.9%	625,960,058	81.4%	170,959,622	94.9%	73%
3,907,616,768	1,025,503,232	73.8%	659,945,996	83.1%	190,750,431	95.1%	71%
5,008,232,108	1,253,650,827	75.0%	27,817,676	99.4%	6,693,320	99.9%	74%
24,751M	6,158M	**75.1%**	3,938M	**84.1%**	987M	**96.0%**	**73%**

Table 2: Compression may be effectively applied to optimized GDSII Stream files

From **Table 2**, above, observe that gzip is, on average, approximately 75% effective at reducing the size of GDSII Stream files whether or not the file has been optimized. GDSII Stream optimization has no significant effect on the effectiveness of gzip compression.

Observe also, that GDSII Stream optimization and gzip compression, when used together, reduce GDSII Stream files by an average of 96%, or by 25X, regardless of file size.

3. OPTIMIZATION RUNTIME

Table 3 lists Bantam GDSII Stream Optimization CPU runtime for a large range of GDSII file sizes. The runtimes are normalized to a 3.8 GHz Pentium 4 class processor. Throughput, expressed as minutes per gigabyte is calculated for each GDSII Stream file. Overall throughput, again in minutes per gigabyte, is calculated at the bottom of **Table 3.** We conclude that Bantam requires, on average, 2.3 minutes to optimize one gigabyte of GDSII Stream data on a 3.8 GHz Pentium 4 class computer.

	Original GDSII Stream File Size (bytes)	Optimization Runtime (CPU seconds)	Optimization Throughput (minutes/GB)
	458,029,056	116	4.2
	677,967,872	185	4.6
	1,110,000,000	106	1.6
	1,317,781,504	347	4.4
	2,122,000,000	297	2.3
	2,974,000,000	410	2.3
	3,183,000,000	598	3.1
	4,877,000,000	957	3.3
	7,046,074,368	765	1.8
	14,039,000,000	2835	3.4
	16,049,420,288	1956	2.0
	16,999,444,480	3600	3.5
	68,368,273,408	6989	1.7
Sum	**139,221,990,976**	**19163**	---
Overall Average Throughput (minutes/GB)	**2.3**		

Table 3: Bantam requires, on average, 2.3 minutes to optimize 1GB of GDSII Stream data

4. OPTIMIZATION EFFECT ON TAPE-OUT FLOW

Given that optimization becomes more effective the larger the GDSII Stream file and that the largest GDSII Stream files are produced as they are embellished as part of a modern tape-out flow, we attempted to characterize the overall effect of GDSII Stream optimization on a GDSII Steam tape-out flow.

Figure 3 below illustrates where GDSII Stream optimization may be best applied in a generic tape-out process.

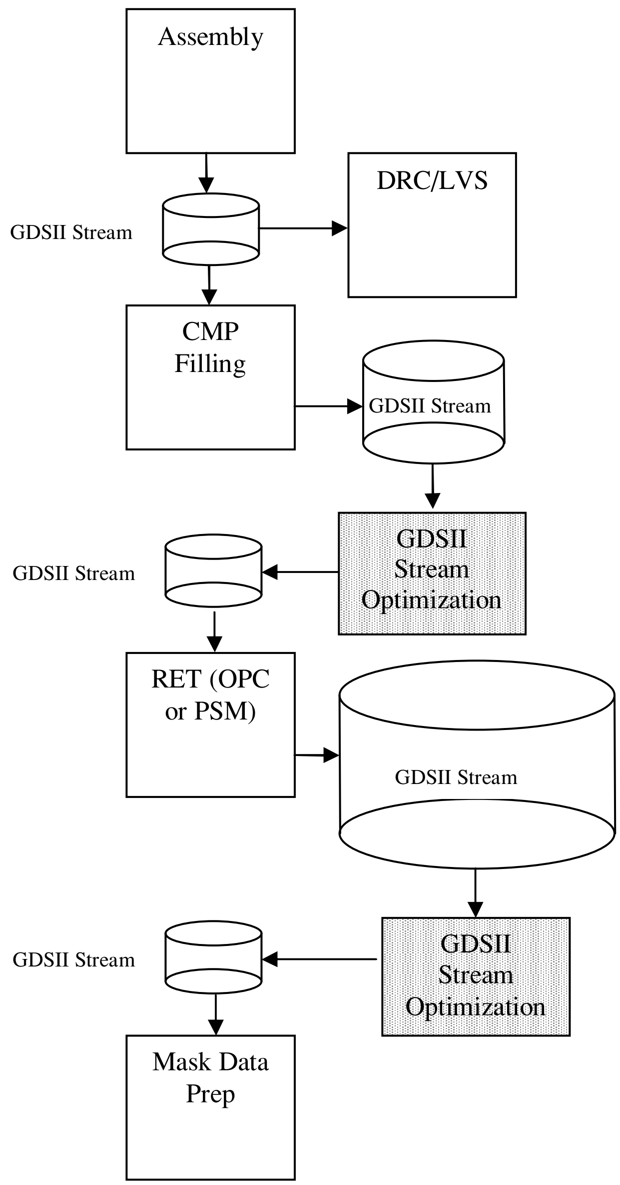

Figure 3: GDSII Stream optimization in a generic GDSII Stream-based tape-Out flow

Table 4 summarizes GDSII Stream optimization as applied following OPC and its effect on Mask Data Preparation (MDP), or “fracturing”. A variety of file sizes were tested, from less than a gigabyte to 68 gigabytes.

Observe that GDSII Stream optimization improved average MDP throughput by 40%. Care has been taken to include the optimization runtime in the throughput calculation.

	Original File Size (bytes)	Original File MDP Runtime (CPU seconds)	Optimized File MDP Runtime (CPU seconds)	MDP Runtime Improvement Factor	Optimization Runtime (seconds)	Throughput Improvement Factor
	458,029,056	114	86	1.3	116.3	0.6
	677,967,872	121	94	1.3	185.3	0.4
	1,317,781,504	274	186	1.5	347.4	0.5
	7,046,074,368	1,834	1548	1.2	765.3	0.8
	16,049,420,288	3,556	2343	1.5	1956.0	0.8
	16,999,444,480	8,363	1729	4.8	3600.0	1.6
	68,368,273,408	15,975	917	17.4	6989.0	2.0
Sum		**30,237**	**6,903**		**13,959**	
Overall Throughput Improvement Factor	**1.4**					

Table 4: Optimization of GDSII Stream files improved average MDP throughput by 40%

4.1. Other tape-out flow effects

Other throughput improvements have been reported by users of GDSII Stream optimization.

4.1.1. Pre-OPC

Applying GDSII Stream optimization to a 60MB prior to running outrigger insertion vastly improved its runtime. Without optimization, outrigger insertion ran in 18 hours, 20 minute. After optimization, outrigger insertion ran in 11 minutes, a runtime improvement factor of 97. The optimized GDSII Stream file was 463KB.

4.1.2. Post-OPC

Applying optimization to an optically corrected GDSII Stream file prior to an OPC verification operation that included a logical XOR improved its runtime by 11.5 hours: the runtime was reduced from 26.2 hours to 14.7 hours.

5. CONCLUSION

GDSII Stream optimization has demonstrated vast reduction in file sizes and significant overall improvement in tape-out throughput, possibly obviating a near term change of file format.