**2023年中国研究生创芯大赛华为赛题**

# 赛题一：24~29.5G毫米波PA设计

**描述及要求（基础）：**

1. 频率范围：24GHz~30GHz;
2. 增益(S21)：带内15-18dB
3. S21 Gain ripple: <1.5dB(@任意3GHz BW，越小越好);
4. NF：<6dB;
5. 输入差分100欧姆，输出单端50欧姆；
6. OP1: >17dBm;
7. OIP3:>20dBm(双音频率间隔50M~800MHz，单音输出功率范围：0~10dBm);
8. Peak PAE:>22%;
9. PAE@6dB\_PBO:>11%(相对于OP1回退)；
10. AMAM:<1dB；AMPM:<10o ；
11. S11<-10dB,S22<-7dB;
12. 电源电压：可根据工艺自己选择合适电压域，变化范围+/-5%；
13. 结温范围：−20℃至+85℃
14. 建议使用标准CMOS工艺；

**评审得分点：**

1. 有完整的电路原理图、版图及前后仿结果。
2. 电路稳定性必须保证，要有稳定性仿真结果。
3. OP1、Peak\_PAE、PAE@6dB\_PBO等性能越好，得分越高。
4. 关键版图寄生必须采用电磁仿真抽取，整版电磁仿真是加分项。
5. 需要提供PVT仿真结果；有电路可靠性分析和考虑等内容是加分项；
6. 需要有设计文档，文档中要体现具体设计思路(如电路指标分析分解、架构选取、关键指标的设计分析、core管类型及尺寸的选取依据、匹配网络的设计考虑、版图寄生的影响等)。
7. 版图布局合理，面积紧凑；
8. 查询业界典型产品或paper的指标，分析差距存在的原因，和可能的改进方向；

**输出要求：**

1. 详细的设计说明文档。
2. 电路版图。
3. 电路原理图及仿真TB设置说明。

# 赛题二：高精度PhaseShifter芯片设计

**描述及要求：**

1. 频率范围：76G~81G
2. 移相步进: 5.6度（更小步进更好）
3. 移相RMS Error:2.8度（RMS Error越小越好）
4. 移相Max Error(INL)<5.6度（指相邻状态之间的相位差，值越小越好）
5. 移相增益变化<0.5dB（指360度变化条件下的增益波动，波动越小越好）
6. 增益：>0dB
7. 噪声系数：<10dB
8. 线性度：IP1dB>-1 dBm
9. 功耗: <50mW （功耗越低越好）
10. 电源电压：可根据工艺自己选择合适电压域，变化范围+/-5%；
11. 结温范围：−20℃至+85℃
12. 建议使用标准CMOS工艺；

**评审得分点：**

1. 思路正确，没有大的Bug；可以采用有源PS，也可以采用无源PS+AMP组合；
2. RMS Error、Max Error、噪声等关键性能指标越高，得分越高；
3. 需要有文档，说明各个子电路性能指标的分解依据，子电路结构的选择依据等；
4. 查询业界典型产品或paper的指标，分析差距存在的原因，和可能的改进方向；
5. 面积、功耗有合理分析；
6. 提供PVTF（工艺、电压、温度、频率）Corner仿真结果；
7. 各个子模块的功耗、移相Error和噪声贡献（用饼状图给出占比分析）；
8. 如果需要校准，提供校准方法及校准开销分析（推荐单点（55度、78.5GHz）校准，全温、全频率范围用同一套码字）。

**输出要求：**

1. 详细的设计说明文档。
2. 电路版图。
3. 电路原理图及仿真TB设置说明。

# 赛题三：低功耗高精度ADC设计

**描述及要求：**

1.输入信号带宽：>100K

2.有效分辨率（ENOB）: >16bit

3.功耗:<2mW

5.输入信号幅度:1Vpp

6.架构：不限

7.工艺：建议使用标准CMOS工艺

8.温度范围：−20℃至+85℃

9.供电电压：随选定工艺而定

**评审得分点：**

1.思路正确，根据性能、功耗的要求要有合理的架构选型分析；

2.在满足指标要求的情况下FOM越高，得分越高。FOM超过185dB可获得额外加分；

3.需要有文档，说明各个子电路性能指标的分解依据，子电路结构的选择依据等；

4.各个子模块的功耗、噪声和非线性等用饼状图给出占比分析；

5.校准算法选择及有效性分析是加分项；

6.查询业界典型产品和paper的指标，分析差距存在的原因，和可能的改进方向；

**输出要求：**

1.系统模型或电路模型；

2.详细分析设计文档；

3.电路原理图仿真验证数据；

# 赛题四：LDPC编解码模块设计

**描述及要求：**

1. 基于题目中提供的编解码矩阵码字设计LDPC编解码模块LDPC\_ENC/LDPC\_DEC；
2. 通过编码IP完成编码后，参赛者使用标准BPSK调制、加噪、解调生成5bit LLR译码数据，通过解码IP进行解码；
3. 分析信噪比SNR与误帧率PER之间的关系，以1000帧随机码字为标准，PER从10%下降到0%时，SNR上升不超过3dB；
4. 设计、优化定点算法，使用verilog实现该LDPC\_ENC/LDPC\_DEC模块；
5. 在100MHz时钟主频下，吞吐率不低于200Mbps；
6. 相同SNR性能下，追求面积功耗优化；
7. ****

**评审得分点：**

1. 实现算法功能正确，满足题目要求；
2. 设计方案文档描述清晰，模块功能划分合理；
3. 算法文档明确说明模块内部量化定标，及对应得性能分析；
4. 编码IP分数占30%，解码IP分数占70%；
5. PER 10%与0%时的SNR越低，得分越高；
6. 文档包含对模块面积和功耗优化的措施说明，优化措施越有效，模块面积越小，功耗越低，得分越高；
7. 上述SNR/PPA指标均以所有参赛团队在各个专项的归一化分数统计，以各专项第一名的指标为10分，最后一名的指标为1分，其他名次指标在中间做线性量化分数。
8. 要求有完备的验证方案和验证用例；

**输出要求：**

1. 算法设计与优化分析报告；（含方案分析与性能仿真结果）
2. LDPC\_ENC/LDPC\_DEC详细设计文档和RTL代码；
3. LDPC\_ENC/LDPC\_DEC验证环境、验证用例、验证数据和波形截图；
4. 提供IP的功耗、性能、面积评估数据，使用工艺库评估的需标明工艺库；使用FPGA工具评估的，需写明工具版本、device型号、资源占用、时序信息等。

# 赛题五：基于非线性PA(power amplifier)的预失真补偿模块设计

**描述及要求：**

1. 根据给定的PA失真模型，设计一套预失真校准与补偿模块，模块名DPD\_CALI与DPD\_COMP；
2. 挂载赛题中提供的“PA失真参数模型paDistProc”，使用“训练数据源trainData.dat”循环发数进行DPD补偿系数训练；
3. 训练完成后，使用赛题中提供的3档输入增益paInputGain对3组“测试数据源verifyData1/2/3.dat”进行补偿结果验证；
4. 针对3档输入增益，给出训练完成的DPD+“PA失真参数模型”在不同输出功率的EVM曲线（EVM的计算方式参见附件说明）、及对应的MASK裕量曲线（MASK计算方式参见附件说明）；
5. 在赛题中给定的采样率和输入、输出数据信号定标下，完成算法方案设计和RTL模块实现；
6. RTL设计需要综合考虑性能和PPA指标；
7. 

**评审得分点：**

1. 功能正确，符合题目要求；
2. 算法分析报告要求全面，综合考虑各功率档位性能；
3. 中低功率段，重点优化EVM；高功率段，重点优化mask裕量
4. RTL详设文档重点描述定点化策略、模块划分方法和功耗面积优化手段；
5. 同等性能水平下，模块实现面积、功耗越小，分数越高；

**输出要求：**

1. 算法分析报告（包含方案选择分析、性能结果数据）
2. 算法定点化实现文档，仿真代码和仿真结果数据；
3. DPD模块详细设计文档和RTL代码，详设文档需说明针对功耗、面积所采取的优化措施和取得的结果；
4. DPD模块验证环境、验证用例、验证报告；
5. 提供IP的功耗、性能、面积评估数据，使用工艺库评估的需标明工艺库；使用FPGA工具评估的，需写明工具版本、device型号、资源占用、时序信息等。

# 赛题六：基于行波模型对激光器的瞬态响应进行数值建模设计

**描述及要求：**

1. 用于仿真DFB激光器的一组耦合行波方程（coupled traveling wave equations）为：

其中 和 分别为前向和后向传播的波，变量为z和t。 是一个随z变化的增益项，随着注入电流而改变。为了简化问题，可以假设 ， A是一个常数。 和 为自发辐射，它是一个给定的随z变化的函数。 和 是常数， . 边界条件为:

其中 和 是常数. 一般采用时域或频域方法进行数值求解。

1. 目标是采用这两种方法或其他方法基于C++或Python等语言进行数值建模，计算激光器瞬态功率随注入电流的频率响应，腔内载流子和光子浓度分布。从精度、收敛性、适用场景、计算速度等方面对其中两种方法进行定量比较。

**评审得分点：**

1. 方案设计描述清晰，包含详细的数值模型推导过程、求解方程的核心算法说明和框架设计；
2. 理论分析越全面；
3. 算法速度快，收敛性好；
4. 同样精度下计算量越小，得分越高；
5. 考虑纵向空间烧孔效应；
6. 两种方法的比较全面。

**输出要求：**

1. 详细的设计文档和代码；
2. 给出算法的收敛阶数和精度，并予以证明；
3. 列出所有的假设，以及模型中使用参数的有效范围；
4. 输出测试用例、验证数据和测试分析报告；
5. 给出两种算法的计算量和计算时间等，多维度分析比较。

# 赛题七：300GB+高带宽InP调制器设计

**描述及要求（基础）：**

1. 设计一个满足性能要求的300 GB+高带宽InP基MZ调制器；
2. 6-dB EO带宽 > 150 GHz；
3. 半波电压(Vπ) < 3 V；
4. 给出InP基MZ调制器的外延材料、波导结构和RF电极的设计方案和仿真结果；
5. 完成InP调制器的制作工艺流程方案和工艺容差分析；

**描述及要求（加分）：**

1. 6dB EO带宽 > 180 GHz ，越大越好；
2. 半波电压(Vπ) < 2.5 V，越小越好；
3. 调制器插入损耗 < 10 dB, 越小越好；
4. 调制器总功耗越低越好；
5. 制作简单，工艺步骤越少越好，工艺容差越大越好；

**评审得分点：**

1. InP基MZ调制器设计方案具有可行性，能满足300GB+波特率的性能要求；
2. 6dB EO带宽指标越大，得分越高；
3. 半波电压(Vpi)越小，得分越高；
4. 调制光学损耗越小越好；
5. 制作工艺简单可靠，工艺步骤越少，得分越高；工艺容差越大，得分越高；

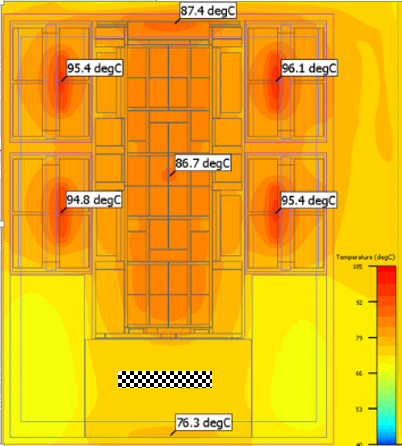
**输出要求：**

1. 300 GB+高带宽InP基MZ调制器系统设计思路。
2. InP基MZ调制器系统制作工艺流程方案和工艺容差分析报告。
3. 仿真结果报告(EO response)，需提供仿真中采用的材料参数表。
4. 总结：方案创新点、优势、不足，改进建议等。

# 赛题八：PMU TDT Tsensor

**背景介绍：**

Tsensor（温度传感器）可集成在芯片内部，用于实现高精度Die内节温检测，在DIE上合理分布放置Tsensor可以得到DIE上的热力云图。Tsensor不断输出带有温度信息的二进制码值给SOC，芯片系统利用码值进行OTP（过热保护）/DVFS（动态电压频率调整）等控制。



**题目描述及要求：**

设计一个片上全集成Tsensor，输入信号仅提供电源（VDD），输出信号为可表征温度信息的数字信号（TEMP<X：0>），如下图所示。



主要指标的基本要求如下表所示，对于温度检测精度/吞吐率/功耗三项的性能得分详见评分标准表格。

|  |  |  |
| --- | --- | --- |
| 指标名 | 基础要求 | 备注 |
| 温感单元 | 不使用BJT | - |
| 温度检测范围 | -40℃~125℃ | - |
| 温度检测精度 | ≤±1℃ | 输出码值计算温度值与温度设定值间的偏差 |
| 吞吐率 | ≥10K sps | 输出数字码值刷新率 |
| 功耗 | ≤0.6mW | 包含所有电路（温感单元及检测电路） |

**系统设计及仿真要求：**

1. 顶层电路统一命名为tsensor\_top；
2. 输出数字信号为二进制码值，位宽不做限定，可根据系统指标分解自行决定；
3. 仿真需包含P（工艺corner）V（电源电压）T（-40℃~125℃）三种仿真条件遍历，其中电源电压遍历典型值/0.95\*典型值/1.05\*典型值三个档位，温度检测精度仿真需覆盖-40°C~125°C范围中一系列温度点，相邻点的间隔至多5°C（即5℃/step），例如，-40°C，-35°C，-30°C，......，120°C，125°C”。
4. 校正条件限定为不超过两个温度点校正，如设计中包含校准，需在详细设计报告中说明校正方案，并在前仿真报告中体现校准前和校准后的温度检测精度仿真结果。例如：-40℃对应的十进制码值为426，125℃对应的十进制码值为721，为得到更精准的输出温度值，采取校正公式，公式中B2D（TEMP<X：0>）表示二进制到十进制的转换，上述校正公式仅使用上文提及的两个温度点校正，通过校正公式即可得到每个码值对应的温度值。

**交付件要求：**

1. 详细设计报告：内容包含但不限于系统框图、系统&子模块工作原理分析、子模块指标分解和电路原理图；
2. 前仿真报告：内容需包含子模块及整系统的仿真条件、仿真电路、仿真波形及仿真结果分析，仿真波形包括但不限于DC/AC/TRAN/MC；
3. 电路原理图及仿真电路数据库；

**评分标准**

1. 温感单元不使用BJT，温度检测范围-40℃~125℃这两项要求为强约束，不满足则判定为0分；
2. 电路中出现理想器件判定为0分；
3. 其余评分标准及最高分值如下表所示，针对详细设计仿真报告和前仿真报告，评委根据实际完成情况在最高分范围内进行打分；
4. 针对指标达成度，请提供证明达成情况的仿真环境和仿真用例，例如，可提供温度间隔为0.5°C的两个温度点的仿真用例展示达成情况，同时也方便阅卷专家修改用例进行相关的达成度测试。

|  |  |  |
| --- | --- | --- |
| 类别 | 评分标准 | 最高分值 |
| 详细设计报告 | 1、系统框图清晰； 2、系统&子模块工作原理分析正确； 3、系统指标分解合理、完备； | 15 |
| 前仿真报告 | 1、仿真用例对系统&模块指标覆盖率； 2、指标选取仿真器（DC/ AC/ TRAN/MC）合理； 3、仿真结果正确； 4、仿真结果分析合理； | 15 |
| 指标达成度 （全PVT条件，每一项指标项按照达成的最高值获取相应分数） | 温度检测精度  Accuracy≤±0.5℃：50分 ±0.5℃＜Accuracy≤±1℃：40分 ±1℃＜Accuracy≤±3℃ ：30分 | 50 |
| 功耗 W≤0.1mW ：10分 0.1mW＜W≤0.6mW ：8分 0.6mW＜W≤6mW ：5分 | 10 |
| 吞吐率 100Ksps≤fs：10分 10Ksps≤fs＜100Ksps ：8分  1Ksps≤fs＜10Ksps ：5分 | 10 |

# 赛题九：SerDes TDT可配置SAR ADC

**题目描述：**

1. 基于深亚微米CMOS（28nm/65nm/90nm/180nm…）工艺，参考异步逐次逼近型模数转换器结构（SAR ADC），设计一款单通道分辨率可配置（6位和8位）的奈奎斯特高速模数转换器SAR ADC，该SAR ADC可以根据需要（模式配置信号），分别工作在6位和8位模式下：

模式一：6位模式的采样率不低于200MS/s。输入信号差分幅度Vppd=2\*LVCC，LVCC为core device的标准电源电压。输入信号频率为奈奎斯特频率正弦波下，信噪失真比（SNDR）不低于35dB。输入信号源为理想信号源+100ohm输出阻抗。

模式二：8位模式采样速率为6位的1/4。输入信号差分幅度Vppd=2\*LVCC，LVCC为core device的标准电源电压。输入信号频率为奈奎斯特频率正弦波下，信噪失真比（SNDR）不低于46dB。输入信号源为理想信号源。输入信号源为理想信号源+100ohm输出阻抗。

1. 请根据分辨率可配置的要求设计该SAR ADC的结构，并完成电路设计及仿真，要求：

前仿真，TT Corner，温度27度，标准电压

SAR ADC中不能使用理想器件，如理想电阻、电容、理想开关、VerilogA模块等。

按照分辨率可配置的要求，对所设计的可配置方案进行阐述和分析。

输出整体可配置SAR ADC的设计和仿真报告，包括但不限于：分辨率可配置方案原理及相关分析、整体SAR ADC结构及时序、核心子模块电路结构（比如比较器、采样保持开关等）及仿真结果（比如时域波形、频谱、时序图等）、整体前仿真结果、testbench说明等。

所设计的可配置SAR ADC要充分考虑整体功耗、转换速度、量化精度、结构复杂度等，进行均衡设计。并根据仿真结果，计算出上述模式一/模式二条件下的SNR、SFDR、THD和SNDR，并优值FoM（FoM=Power/(ERBW×2ENOB），Power为ADC总功耗、ERBW为ADC带宽（相比于低频SNDR，带宽处SNDR衰减3dB）、ENOB为ADC奈奎斯特频率采样下的有效位数（此处ENOB定义为，当ADC输入信号频率为对应采样率的奈奎斯特频点时的有效位数）。

**输出要求：**

1. 可配置SAR ADC的设计报告和仿真报告，并针对仿真结果进行分析，计算出FoM。
2. 可配置SAR ADC的schematic数据库，以及上述指标对应的仿真testbench数据库。Testbench中需要包含能够展示功耗、SNDR、速率、Enob、FOM的仿真用例

**评分标准：**

1. 上述题干中模式一和模式二的要求为必备条件，不满足则判定为0分
2. ADC电路中不能使用理想器件，不满足判定为0分
3. 其余评分标准及最高分值如下表所示，针对详细设计和仿真报告，评委根据实际完成情况在最高分范围内进行打分。

|  |  |  |
| --- | --- | --- |
| 类别 | 评分标准 | 分值 |
| 详细设计报告 | 1、系统框图清晰； 2、对所设计的可配置方案有详细的阐述和分析，特别是6/8bit模式切换的设计 | 15 |
| 前仿真报告 | 1、仿真用例对系统&模块指标覆盖率； 2、仿真结果正确； 3、仿真结果分析合理； | 15 |
| Schematic和tb数据库 | 1. schematic层次完整，在报告中说明清晰 2. tb库完整，在报告中说明清晰 | 10 |
| 指标达成度 | 6位模式下的速度fs  fs < 200M: 0  200M ≤ fs < 300M: 10+10\*(fs-200M)/100M  fs > 300M: 20 | 20 |
| 6位模式下精度为SNDR  SNDR < 35dB: 0  35dB≤SNDR < 37.85dB: 5+1.75\*(SNDR - 35）  SNDR ≥ 37.85 : 10 | 10 |
| 8位模式下精度SNDR  SNDR < 46dB: 0  46dB≤SNDR < 49.9dB: 5+1.28\*(SNDR – 46）  SNDR≥ 49.9dB : 10 | 10 |
| 6位模式FoM  FoM > 156fJ/conv.-step : 0  15.6fJ/conv.-step <FoM < 156fJ/conv.-step : (156 – FoM)/140.4\*10  FoM < 15.6fJ/conv.-step: 10 | 10 |
| 8位模式FoM  FoM > 156fJ/conv.-step : 0  15.6fJ/conv.-step <FoM < 156fJ/conv.-step : (156 – FoM)/140.4\*10  FoM < 15.6fJ/conv.-step: 10 | 10 |

# 赛题十：高效定点乘法器设计

**背景介绍：**

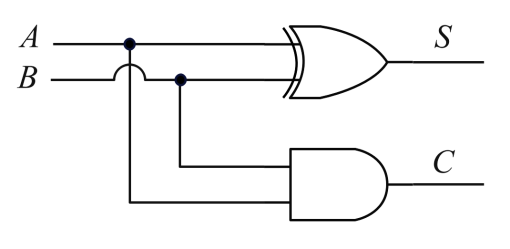
1. 定点乘法器是现代信号处理常用的运算单元之一，其整体性能直接决定了系统的竞争力。通常乘法器包含三个基本运算：部分积的产生，部分积的压缩以及对压缩结果进行求和。在进行最终求和之前，可以采用不同的压缩方法将所有部分积压缩成两个部分积，最后采用进位延迟加法器CPA (Carry Propagate Adder) 求出最终结果。
2. 以4\*4阵列定点乘法器的实现过程为例：被乘数和乘数单bit相与后产生16个部分积 ，每一列的部分积对应的权重一致。其中，X3Y3对应为最高权重，记MSB；X0Y0对应为最低权重，记LSB。LSB到MSB权重按2的幂次递增。图1显示了所述乘法器的具体实现过程。

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 被乘数 |  |  | *X*3 | *X*2 | *X*1 | *X*0 |
| 乘数 |  | \* | *Y*3 | *Y*2 | *Y*1 | *Y*0 |
|  |  |  | *X*3*Y*0 | *X*2*Y*0 | *X*1*Y*0 | *X*0*Y*0 |
|  |  | *X*3*Y*1 | *X*2*Y*1 | *X*1*Y*1 | *X*0*Y*1 |  |
|  | *X*3*Y*2 | *X*2*Y*2 | *X*1*Y*2 | *X*0*Y*2 |  |  |
| *X*3*Y*3 | *X*2*Y*3 | *X*1*Y*3 | *X*0*Y*3 |  |  |  |
| *X*3*Y*3 | … | | | | *X*1*Y*0+*X*0*Y*1 | *X*0*Y*0 |

4\*4的乘法器实现过程

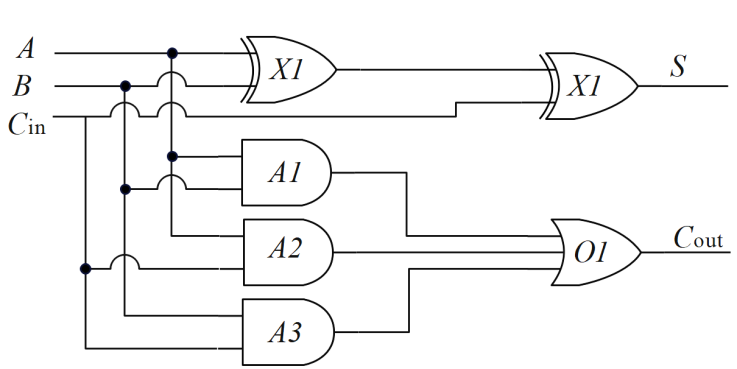
1. 对于一般的阵列乘法器，X, Y两数相乘，同一权重（同一行上的部分积）需要进行压缩操作，将多个数据压缩到一个。即每一列的加法进位输入依赖于前一列的进位输出，这需要用到半加器或者全加器 ，其电路实现和真值表如图2和图3所示。

|  |  |  |  |
| --- | --- | --- | --- |
| *A* | *B* | *Carry* | *Sum* |
| 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 |
| 0 | 1 | 0 | 1 |
| 1 | 1 | 1 | 0 |



半加器逻辑实现和真值表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *A* | *B* | *C*in | *C*out | *S* |
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 |
| 0 | 1 | 0 | 0 | 1 |
| 1 | 1 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 1 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 1 | 1 | 1 | 1 |



全加器逻辑实现和真值表

1. 评判一个定点乘法器实现的优劣，除了运算结果必须正确，还需要评估以下指标：

硬件资源消耗：硬件资源越少，乘法器的成本越低。本题目中使用资源代价分来考察硬件资源消耗；

最大延时：从乘法器的输入数据到输出数据中，最长的计算路径定义为关键路径，其长度定义为最大延时。关键路径的长度，决定了一次乘法操作的所需时间。关键路径越短，乘法器的工作速率越快，性能竞争力越强。本题目中使用性能代价分来考察最大延时。

1. 前人对乘法器做了很多优化，其中较为常见的是乘数编码和加法树压缩。乘数编码的目的是减少部分积的个数并降低加法树的深度，减少硬件资源使用。加法树压缩的目的是加快加法树的化简过程，减少硬件资源使用。两种方法的目的都是为了优化乘法器的实现过程，缩短关键路径，降低资源消耗。下面分别举例两种优化算法。

乘数编码：

乘数编码有多种方案 ，其中booth编码最为常见，这里以基4booth算法举例，其原理如下：

对于N比特数B来说：

基4 Booth乘法器的基系数为：

其中

将A与B相乘，则：

以下是基4 Booth编码表，其中A为被乘数，B为乘数。其真值表为

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Bi+*1 | *Bi* | *Bi*-1 | -2*Bi* +1+ *Bi* + *Bi* -1 | 部分积操作 |
| 0 | 0 | 0 | +0 | 0 |
| 0 | 0 | 1 | +1 | *A* |
| 0 | 1 | 0 | +1 | *A* |
| 0 | 1 | 1 | +2 | 2*A* |
| 1 | 0 | 0 | -2 | -2*A* |
| 1 | 0 | 1 | -1 | -*A* |
| 1 | 1 | 0 | -1 | -*A* |
| 1 | 1 | 1 | -0 | 0 |

基4 Booth真值表

以7\*9为例说明基4 Booth乘法器实现过程：



基4 Booth乘法器实现过程

加法树的压缩：部分积的压缩则采用进位保留加法器的原理，使用进位保留加法器对部分积进行逐级压缩，直到最后压缩成只剩下两个部分积，即全加器的进位输出和模2和输出。以三个部分积压缩过程为例：首先将3个部分积压缩成两个部分积，然后再与下一个部分积组成新的一组，进行3:2压缩，直至最后压缩成2个部分积，具体过程如图 5所示。需要说明的是，进位保留的压缩方式是串行进行的。即对于N个部分积，需要N-2次压缩才能完成，即需要N-2级全加器。

加法树的压缩有多种方式，其中Wallace压缩是最为常见，以图一所示4\*4的乘法过程举例Wallace压缩原过程其中红色数据用全加器处理，绿色数据采用半加器处理。

第一次压缩：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  | *X*3 | *X*2 | *X*1 | *X*0 |
|  |  | \* | *Y*3 | *Y*2 | *Y*1 | *Y*0 |
|  |  |  | *X*3*Y*0 | *X*2*Y*0 | *X*1*Y*0 | *X*0*Y*0 |
|  |  | *X*3*Y*1 | *X*2*Y*1 | *X*1*Y*1 | *X*0*Y*1 |  |
|  | *X*3*Y*2 | *X*2*Y*2 | *X*1*Y*2 | *X*0*Y*2 |  |  |
| *X*3*Y*3 | *X*2*Y*3 | *X*1*Y*3 | *X*0*Y*3 |  |  |  |

第一次用两个半加器分别对X1Y2 和X0Y3 ，X2Y2+X1Y3进行处理，得到如下的部分积结果。

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | *X*3*Y*2 | *X*3*Y*1 | *X*3*Y*0 | *X*2*Y*0 | *X*1*Y*0 | *X*0*Y*0 |
|  | *X*2*Y*3 | *b*1[0] | *X*2*Y*1 | *X*1*Y*1 | *X*0*Y*1 |  |
| *X*3*Y*3 | *b*1[1] | *b*0[1] | *b*0[0] | *X*0*Y*2 |  |  |

其中b0[0]为X1Y2 + X0Y3 的求和值，b1[0]为X1Y2 + X0Y3的进位值;b0[1]和b1[1]分别为对应X2Y2+X1Y3的求和值与进位值。

第二次压缩：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| *X*3*Y*3 | *c*3[0] | *c*2[0] | *c*1[0] | *X*2*Y*0 | *X*1*Y*0 | *X*0*Y*0 |
| *c*3[1] | *c*2[1] | *c*1[1] | *c*0[1] | *c*0[0] | *X*0*Y*1 |  |

Wallace压缩流程

其中c0[0]为X1Y1 + X0Y2的求和值，c1[0]为该半加器进位值；c0[1]为X3Y0 + X2Y1+ b0[0]的求和值，c1[1]为该全加器进位值，以此类推c2，c3值；两次压缩后 加法树的深度降低为2，最后通过加法器得到乘法运算的最终结果。

**题目介绍：**

设计一个高效的16bit有符号数定点乘法器乘数和被乘数均为16bit有符号随机数，输出为32bit有符号数。不限定乘法器的设计方案和形式，要求乘法器运算结果正确，且最大可能的降低资源消耗，提高性能。最大性能代价分满足设计要求 (资源代价得分，性能代价得分，将在第五部分详细介绍)。请根据要求规划该高效乘法器的结构并设计实现方案。

**设计要求：**

1. 使用附录1中的8种逻辑门电路及其组合搭建16\*16bit有符号数乘法器。
2. 代码顶层统一命名为HIS\_MULT\_TOP，乘数使用A\_NUM标记，被乘数使用B\_NUM标记，输出积使用C\_NUM标记。
3. 算法实现使用硬件描述语言，如VHDL/Verilog搭建。

**验收说明：**

1. 详细设计报告：包括但不限于算法方案，详细设计文档，设计代码，乘数编码操作和加法树的压缩过程；算法方案要求给出详细的乘法器工作原理分析。详细设计文档要求给出详细设计框图、子模块间的连接关系、乘数编码操作和加法树的压缩过程，可以参考第一部分给出的乘数编码示例给出编码公式和操作，加法树压缩示例给出每一次的压缩过程。
2. 设计方案需要合理考虑资源消耗和延时代价，尽可能的优化或者折中，并且准确统计出该乘法器的资源代价分和性能代价分，添加在详细设计报告的附录中。

**评分标准**

1. 必要条件：

乘法器运算结果正确，通过随机数测试；

性能代价分不大于1000；

资源代价分不大于8000

准确统计出乘法器的资源代价分和性能代价分；

满足以上条件者方可进行有效评分；

1. 资源代价评估得分 50

根据所有参考人员完成该乘法器消耗的资源代价打分。资源代价分为完成该乘法器设计使用附录1中逻辑门电路数量和其对应的资源代价分之积。如完成该乘法器使用了5个与门（与门的资源代价分为6），10个非门（非门资源代价分为2），则资源代价分为30+20分。资源代价分采用相对绝对模式，分数越低，得分越高：

≤5000 ： 得分50分，

≤6000 ： 得分30分，

≤7000 ： 得分10分。

1. 性能代价评估得分 35

根据所有参考人员完成该乘法器的性能代价打分。性能代价分考察对象为该乘法器设计的最长延时路径。统计最长延时路径使用附录1中逻辑门电路数量和其对应的性能代价分之积。如最长路径上使用了5个与门（与门的性能代价分为7），10个非门（非门性能代价分3），则设计者的性能代价分为35+30分。性能代价分采用绝对打分模式，分数越低，则排名越靠前：

≤500 ： 得分35分，

≤600 ： 得分20分，

≤800 ： 得分10分。

1. 设计文档质量打分 15

根据所有参考人员的设计文档质量打分。文档要求：算法方案原理分析完备正确，设计文档描述详细，逻辑缜密；实现流程图，子模块的链接清晰关系，并能和交付代码完全对应；实现代码注释详细，通俗易懂；文档质量分采用绝对打分模式

文档交付完完备，工作原理分析正确，描述清晰易懂： 得分15分，

1. 参考人员最终得分为资源代价得分、性能代价的分和设计文档质量打得分之和。

**附录1：**

该乘法器可能用到的八种常用逻辑门（逻辑表达式、逻辑符号、真值表、逻辑运算规则）

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 名称 | 逻辑表达式 | 逻辑符号 | 真值表 | | | | | 逻辑运算规则 | 资源代价分 | 性能代价分 |
| 与门 |  |  | *A* | 0 | 0 | 1 | 1 | 有0得0  全1得1 | 6 | 7 |
| *B* | 0 | 1 | 0 | 1 |
| *F* | 0 | 0 | 0 | 1 |
| 或门 |  |  | *A* | 0 | 0 | 1 | 1 | 有1得1  全0得0 | 6 | 7 |
| *B* | 0 | 1 | 0 | 1 |
| *F* | 0 | 1 | 1 | 1 |
| 非门 |  |  | *A* | 0 | 1 |  |  | 有0得1  有1得0 | 2 | 3 |
| *F* | 1 | 0 |  |  |
| 与非门 |  |  | *A* | 0 | 0 | 1 | 1 | 有0得1  全1得0 | 4 | 5 |
| *B* | 0 | 1 | 0 | 1 |
| *F* | 1 | 1 | 1 | 0 |
| 或非门 |  |  | *A* | 0 | 0 | 1 | 1 | 有1得0  全0得1 | 4 | 5 |
| *B* | 0 | 1 | 0 | 1 |
| *F* | 1 | 0 | 0 | 0 |
| 与或非门 |  |  | *A* | 0 | 0 | … | 1 | *AB*或*CD*有一组或两组全是1结果得0  其余输出全得1 | 8 | 9 |
| *B* | 0 | 0 | … | 1 |
| *C* | 0 | 0 | … | 1 |
| *D* | 0 | 1 | … | 1 |
| *F* | 1 | 1 | … | 0 |
| 异或门 |  |  | *A* | 0 | 0 | 1 | 1 | 不同得1  相同得0 | 12 | 15 |
| *B* | 0 | 1 | 0 | 1 |
| *F* | 0 | 1 | 1 | 0 |
| 同或门 |  |  | *A* | 0 | 0 | 1 | 1 | 不同得0  相同得1 | 12 | 15 |
| *B* | 0 | 1 | 0 | 1 |
| *F* | 1 | 0 | 0 | 1 |

# 赛题十一：ASIC芯片物理设计中的Path Group规划算法

**赛题十一、赛题十二测试集下载：**<https://cpipc.acge.org.cn/sysFile/downFile.do?fileId=2435b87d92f74a748bc33211605eb59a>

**描述及要求：**

Netlist是ASIC芯片物理设计的输入件，其中定义了数以百万计的standard cell之间的连接情况。物理设计EDA工具会基于netlist完成standard cell的placement、routing以及PPA优化等任务。其中，定义path group是一种常见的PPA优化方法，它通过识别设计中的关键时序路径，定义关键时序路径在时序优化过程中的权重分级，引导工具的优化。具体来说，为了定义path group，需要给定一条或一组时序路径的起点寄存器和终点寄存器（多个寄存器可借助通配符\*表达）以及该path group的优化权重。通常path group需要根据placement阶段后的时序情况来定义，将时序较差的若干路径设置为高权重的path group，具体操作中需要多次迭代与试错，效率较低。因此本课题希望能够寻找一种方法，基于原始的netlist文件，通过图论领域中的路径搜索、partition、连接复杂度分析等算法，在物理设计前发现设计中的潜在时序路径瓶颈，并形成path group约束传递到EDA工具中，以期引导工具达成更好的QoR结果。

要求1：在给定的测试集上，完成netlist网表到图数据（由节点和边以及相关属性组成）的转换。

要求2：在由上一步建模得到的图上，搜索寄存器到寄存器的时序路径、用适当的方式表征路径时序风险、按照时序风险对时序路径进行排序并完成时序关键路径的识别（以上可以只精确到路径的起点和终点）。可利用路径经过的节点数、每个节点的入度、出度或者其他合理方式表征路径的时序情况。

要求3：能够基于图上的时序情况预估结果，形成EDA工具可读的path group约束，并根据时序的瓶颈程度进行分级。即最终需要给出各path group的起点、终点和优化权重/优先级。

说明1：除精确算法，也鼓励使用启发式算法，做好预测精度与算法资源开销的平衡。

说明2：鼓励同学结合自身独特的背景和知识结构，跳出题目给出的提示，理解算法的最终目标，创造性地解决问题。

说明3：group path的权重分级有10个分档，权重分别为1到10.

**评审得分点：**

1. 算法效率。给出以大O符号表示的各阶段算法时间复杂度，以及各阶段算法在测试集上的实际运行时间。各阶段包括但不限于netlist到图的转化，以及在图上搜索寄存器到寄存器间路径长度的算法等。
2. 硬件资源开销，内存资源开销越小越好。尽量给出以大O符号表示的各阶段算法空间复杂度，以及各阶段算法在测试集上的实际平均内存占用和峰值内存占用。如果算法对特殊内存空间有显著开销，例如递归算法与栈空间的开销，还应给出这部分的空间复杂度和执行过程中的实际开销。
3. 路径瓶颈预测精度，预测到的瓶颈点与真实瓶颈点的相符性越高越好

**输出要求：**

1. 算法设计文档、代码与编译脚本，编程语言不限，可以调用开源组件。
2. 在测试用例上的运行结果，要求至少以文本形式输出设计中按时序风险严重程度的起点寄存器与终点寄存器对，以及对应的group path约束文件。要求覆盖设计中的所有寄存器。

# 赛题十二：ASIC芯片物理设计中的high fanout寄存器识别算法

**赛题十一、赛题十二测试集下载：**<https://cpipc.acge.org.cn/sysFile/downFile.do?fileId=2435b87d92f74a748bc33211605eb59a>

**描述及要求：**

Netlist是ASIC芯片物理设计的输入件，其中定义了数以百万计的standard cell之间的连接情况，物理设计EDA工具会基于netlist完成standard cell的placement、routing等任务。其中，寄存器输出端的信号通过组合逻辑网络会传播到其他寄存器的输入端，通常从某一个寄存器出发，信号会传播到很多个寄存器（这些寄存器的数量常称为fanout数量）。当某寄存器的fanout数量很多时，它有可能会成为整个设计的时序与功耗瓶颈。因此，能够高效的在设计早期发现high fanout寄存器对设计优化十分有帮助。

要求1：在给定的测试集上，完成netlist网表到图数据（由节点和边以及相关属性组成）的转换。

要求2：在转换得到的图上，完成寄存器fanout搜索算法，搜索某个寄存器通过组合逻辑网络能传播到多少寄存器的输入端。

要求3：能够基于搜索算法的结果统计所有寄存器的fanout数量，并输出报告。

说明1：除精确算法，也鼓励使用启发式算法，做好预测精度与算法资源开销的平衡。

说明2：鼓励同学结合自身独特的背景和知识结构，跳出题目给出的提示，理解算法的最终目标，创造性地解决问题。

**评审得分点：**

1. 算法效率。给出以大O符号表示的各阶段算法时间复杂度，以及各阶段算法在测试集上的实际运行时间。各阶段包括但不限于netlist到图的转化，以及在图上搜索寄存器的fanout数量的算法等。
2. 硬件资源开销，内存资源开销越小越好。给出以大O符号表示的各阶段算法空间复杂度，以及各阶段算法在测试集上的实际平均内存占用和峰值内存占用。如果算法对特殊内存空间有显著开销，例如递归算法与栈空间的开销，还应给出这部分的空间复杂度和执行过程中的实际开销。
3. 寄存器fanout搜索的准确度，准确度越高越好。

**输出要求：**

1. 算法设计文档、代码与编译脚本，编程语言不限，可以调用开源组件。
2. 在测试用例上的运行结果，要求至少以文本形式输出设计中所有寄存器的fanout数量。

华为赛题专项奖设置：

华为赛题专项奖专门用于奖励选择华为赛题的获奖赛队。华为专项奖是重复奖，参赛赛队可同时参加大赛执行委员会组织的其他大赛奖项的评审和获奖。

华为专项奖设：

特等奖 2队，每队奖金2万元，合计4万元。

一等奖 5队，每队奖金1万元，合计5万元。

二等奖 12队，每队奖金0.5万元，合计6万元。

华为赛题咨询答疑：华为公司王老师，电子邮箱：wangbo24@hisilicon.com