

UNIVERSIDAD AUTONOMA DE MADRID

ESCUELA POLITECNICA SUPERIOR



PROYECTO FIN DE CARRERA

**TÉCNICAS DE LOW-POWER DESIGN EN
FPGAs**

Godoy Garcés Mencía

SEPTIEMBRE 2013

TÉCNICAS DE LOW-POWER DESIGN EN FPGAs

AUTOR: Godoy Garcés Mencía

TUTOR: Eduardo Boemo Scalvinoni

Digital System Lab

Dpto. Tecnología Electrónica y de Comunicaciones

Escuela Politécnica Superior

Universidad Autónoma de Madrid

Septiembre de 2013

Resumen

En el presente proyecto se abordan distintas técnicas de diseño con el fin de conseguir reducir el consumo de potencia en un determinado tipo de circuito electrónico, las FPGAs. Para ello se han elegido diferentes *benchmarks* con el fin de estudiar el funcionamiento de las diferentes técnicas bajo distintas circunstancias.

Se han utilizado distintas técnicas de medida del consumo de potencia para poder determinar el error de las estimaciones de la herramienta del fabricante y analizar en que tipo de circunstancias este tipo de medidas tiene menores errores y por tanto es más fiable.

Finalmente se ha analizado la influencia del diseñador en el proceso de *floorplanning* para poder saber la eficiencia de este proceso si se usa junto con los *settings* de la herramienta del fabricante.

Palabras clave

FPGA, floorplanning, low-power, AES, FFT, consumo, potencia, pipeline, benchmark, propagación de errores, potencia dinámica, potencia estática, potencia de cortocircuito, LUT, DSP, temperatura de unión, condensador de carga, paralelización, puerta lógica, corriente de fuga.

Abstract

Different designs techniques are addresses in this project in order to reduce the power consumption of a particular type of electronic circuit: the FPGA. Different benchmarks have been chosen in order to study the behavior of the different techniques under specific situations.

Several techniques for measuring power consumption have been used to obtain EDA tool estimation errors and analyze their circumstances in order to determine which one implies the lowest error and, therefore, which one is the most reliable.

Finally, the designer influence has been analyzed at the floorplanning process to know the efficiency of this process when it is used with the settings provided by the software.

Key words

FPGA, floorplanning, low-power, AES, FFT, consumption, power, pipeline, benchmark, propagation of error, dynamic power, static power, short-circuit power, LUT, DSP, junction temperature, load condenser, parallelization, logic gates, leakage current.

Agradecimientos

En primer lugar, agradecer a mis padres, Francisco y Emma, su apoyo incondicional durante toda mi vida y ayudarme a llegar hasta aquí, facilitándome todo lo necesario para ello. Gracias por la educación y el cariño recibido. A mi hermano, Héctor, por poder contar con él incondicionalmente, por sus consejos y por ser el mejor ejemplo a seguir.

Gracias a mi tutor, Eduardo Boemo, por haberme dado la oportunidad, las ideas y las herramientas necesarias para realizar este proyecto. También agradecer a Juan Pablo Oliver su ayuda, sus explicaciones y comentarios en los momentos más importantes del proyecto.

Gracias a mis amigos del colegio, por ser una familia después de tantos años y sentirles siempre cerca, aunque algunos estén lejos. Agradecer a mis compañeros de Erasmus, por hacer de aquel año el más importante de mi vida, compartiendo buenos momentos y experiencias.

Gracias a los compañeros de la escuela, por los momentos de prácticas y estudio en los que sin su apoyo todo hubiera sido más difícil. También agradecer a mis compañeros de Deimos, por todo lo que he aprendido este año, por el buen ambiente de trabajo y por siempre estar disponibles para ayudarme.

Y gracias especialmente a Henar, por aprender de ella cada día, por todo lo vivido y compartido. Teniendo cerca una persona tan brillante, todo es más fácil.

Godoy Garcés Mencía

Septiembre 2013

Índice de contenido

Índice de figuras	13
Índice de tablas	21
1. Introducción	23
1.1. Motivación y objetivos	23
1.2. Organización de la Memoria	25
2. Estado del Arte y Marco Tecnológico	27
2.1. FPGAs	27
2.2. Principales fabricantes	28
3. Consumo en circuitos integrados	29
3.1. Potencia y energía	29
3.1.1. Glitches	31
3.1.2. Corrientes de cortocircuito	32
3.1.3. Método tradicional vs método actual en el diseño	34
3.1.4. Relación entre potencia y retardo	35
3.1.5. El espacio Energía-Retardo.....	35
3.2. Diseño temporal. Técnicas a nivel de circuito	36
3.2.1. Sensibilidad Energía-Retardo	37
3.2.2. Reducción de energía activa (dinámica)	38
3.2.3. Múltiples tensiones de alimentación.....	38
3.2.4. Distribuyendo múltiples fuentes de alimentación	39
3.2.4.1. Convencional	39
3.2.4.2. Shared n-well.....	39
3.2.5. Tecnología de mapeo	40
3.2.6. Lecciones del circuito de optimización	42
3.2.7. Reduciendo las fugas (leakage). Longitud del canal	42
3.2.8. Resumen	43
3.3. Diseño temporal. Arquitectura, algoritmos y sistemas	44
3.3.1. Reduciendo la tensión de alimentación.....	45
3.3.2. Implementación de pipeline	47
3.3.4. Mejorando la eficiencia computacional	48
4. Benchmarks	50

4.1. Core Multiplicador Xilinx	50
4.2. Multiplicador segmentado	54
4.3. AES (Advanced Encryption Standard)	56
4.4. FFT (Fast Fourier Transform)	59
5. Opciones de ISE y PlanAhead relacionadas con consumo	61
5.1. Opciones de Xilinx ISE 13.1	61
5.2. Opciones de Xilinx PlanAhead 13.1	67
6. Medidas	70
6.1. Multiplicadores Xilinx	75
6.1.1. ISE	76
6.1.1.1. Balanceado	76
6.1.1.2. Área	79
6.1.1.3. Optimización de potencia	83
6.1.1.4. Tiempo de ejecución	86
6.1.1.5. Rendimiento temporal	90
6.1.2. PlanAhead	93
6.1.2.1. Balanceado	93
6.1.2.2. Optimización de potencia	97
6.1.2.3. Rendimiento temporal con IOB (Input/Output Blocks) Packing	100
6.1.2.4. Rendimiento temporal sin IOB (Input/Output Blocks) Packing	104
6.2. Multiplicador segmentado	108
6.2.1. ISE	109
6.2.2. PlanAhead	111
6.3. AES (Advanced Encryption Standard)	113
6.4. FFT (Fast Fourier Transform)	115
6.5. Floorplanning	117
6.5.1. Multiplicadores Xilinx	117
6.5.1.1. Balanceado	117
6.5.1.2. Optimización de potencia	136
6.5.2. Multiplicador segmentado	149
6.5.2.1. Balanceado	149
6.5.2.2. Optimización de potencia	155
6.6. Número de elementos	160

6.6.1. Multiplicadores embebidos optimizados en área	160
6.6.2. Multiplicadores embebidos optimizados en velocidad	162
6.6.3. Multiplicadores usando LUTs optimizados en área.....	163
6.6.4. Multiplicadores usando LUTs optimizados en velocidad.....	165
6.6.5. Multiplicador segmentado	167
6.6.6. AES.....	168
6.6.7. FFT	169
6.7. Frecuencia	171
6.7.1. Multiplicadores Xilinx	171
6.7.2. Multiplicador segmentado	173
6.7.3. AES.....	174
6.7.4. FFT	174
6.8. Temperatura.....	175
6.9. Condensador de carga.....	177
7. Análisis de los resultados.....	178
7.1. Multiplicadores Xilinx	178
7.1.1. ISE.....	178
7.1.2. Floorplanning con PlanAhead	179
7.2. Multiplicador segmentado	181
7.2.1. ISE.....	181
7.2.2. Floorplanning con PlanAhead	181
7.3. AES (Advanced Encryption Standard).....	182
7.4. FFT (Fast Fourier Transform)	183
7.5. Temperatura.....	183
7.6. Condensador de carga C_L	184
7.7. Potencia vs Frecuencia	185
7.7.1. Multiplicadores Xilinx	185
7.7.2. Multiplicador segmentado	185
7.7.3. AES.....	185
7.7.4. FFT	186
7.8. Comparación de diferentes técnicas de medición	186
8. Conclusiones.....	188
Referencias	190

<i>Anexo A: Tarjeta de desarrollo</i>	192
<i>A.1. Descripción</i>	192
<i>A.2. Descripción funcional</i>	193
<i>A.2.1. Xilinx Spartan-6 LX16 FPGA</i>	194
<i>A.2.2. Cypress PSoC 3 Mixed-Signal Array</i>	195
<i>A.2.3. Relojes</i>	195
<i>A.2.4. Memoria</i>	195
<i>A.2.5. Comunicaciones</i>	196
<i>A.2.6. Puertos de expansión</i>	197
<i>A.2.7. Interfaces de usuario</i>	197
<i>A.2.8 Potencia</i>	197
<i>A.2.8.1. Fuentes de alimentación de la tarjeta</i>	197
<i>A.2.8.2. Gestión de batería</i>	198
<i>A.2.8.3. Reguladores de voltaje</i>	198
<i>A.2.8.4 Sistema de gestión de bajo consumo</i>	198
<i>A.2.8.5 Medida de potencia usando PSoC</i>	199
<i>A.2.8.6 Desacoplamiento FPGA</i>	199
<i>A.2.8.7 Resultados de potencia</i>	199
<i>A.2.9 Configuración</i>	200
<i>Anexo B: Propagación de errores</i>	201
<i>Presupuesto</i>	207
<i>Glosario</i>	209
<i>Pliego de condiciones</i>	210

Índice de figuras

Figura 3.1. Potencia dinámica. Extraído de [11]	29
Figura 3.2. Corriente interna de transición. Extraído de [11]	30
Figura 3.3. Ejemplo de glitches. Extraído de [10].....	32
Figura 3.4. Corriente de cortocircuito. Extraído de [10]	32
Figura 3.5. Reducción de glitches. Extraído de [10]	32
Figura 3.6. Corriente de cortocircuito vs tensión de entrada. Extraído de [10]	33
Figura 3.7. Corriente de cortocircuito para C_L grande. Extraído de [10].....	33
Figura 3.8. Corriente de cortocircuito para C_L pequeño. Extraído de [10].....	33
Figura 3.9. Corrientes de cortocircuito para diferentes C_L . Extraído de [10].	34
Figura 3.10. Relación entre V_{DD} , V_{TH} , potencia y entre V_{DD} , V_{TH} y retardo. Extraído de [12]	35
Figura 3.11. Espacio Energía-Retardo. Extraído de [10].....	35
Figura 3.12. Puntos de optimización Energía-Retardo. Extraído de [8]	36
Figura 3.13. Pila de abstracción. Extraído de [10].....	36
Figura 3.14. Top-down vs bottom-up. Extraído de [10]	37
Figura 3.15. Sensibilidad Energía-Retardo. Extraído de [8].....	37
Figura 3.16. Energía activa y potencia activa. Extraído de [10]	38
Figura 3.17. Diseño para la distribución <i>conventional</i> . Extraído de [10]	39
Figura 3.18. Diseño para la distribución <i>shared n-well</i> . Extraído de [10]	39
Figura 3.19. Ejemplo de reestructuración lógica. Extraído de [10].....	41
Figura 3.20. Ejemplo de reestructuración lógica usando inversores. Extraído de [10]	41
Figura 3.21. Resultados para distintas longitudes del transistor, partiendo de un CMOS de 90 nm. Extraído de [10].....	42
Figura 3.22. Pila de abstracción. Extraído de [10].....	44
Figura 3.23. Gráfica Energía-Retardo. Extraído de [10]	44
Figura 3.24. Gráfica Energía-Retardo al eliminar ineficiencias. Extraído de [10].....	45
Figura 3.25. Gráfica Energía-Retardo. Extraído de [10]	45
Figura 3.26. Circuito a estudio para aplicar paralelismo. Extraído de [4]	45
Figura 3.27. Implementación de paralelismo. Extraído de [4]	46
Figura 3.28. Implementación de pipeline. Extraído de [4].....	47
Figura 3.29. Mejoras usando pipeline. Extraído de [4]	47
Figura 3.30. Aumento del rendimiento en los microprocesadores debido al paralelismo. Extraído de [5].....	47
Figura 3.31. Programación espacial vs programación no espacial. Extraído de [10].....	48
Figura 3.32. Ejemplo de jerarquía de memoria. Extraído de [10]	49
Figura 4.1. Pines del multiplicador. Extraído de [19]	50
Figura 4.2. Pines del multiplicador. Extraído de [19]	51
Figura 4.3. Menú de configuración del multiplicador	51
Figura 4.4. Menú de configuración del multiplicador	52
Figura 4.5. Menú de configuración del multiplicador	53
Figura 4.6. Esquema de segmentación. Extraído de [9].....	54
Figura 4.7. Esquema multiplicador segmentado. Extraído de [9].....	55
Figura 4.8. Proceso del algoritmo AES. Extraído de [20].....	57

Figura 4.9. Esquema de interfaces y módulos. Extraído de [13].....	58
Figura 4.10. Esquema de FFT. Extraído de [14].....	60
Figura 6.1. Pads para la medida del consumo de la FPGA. Extraído de [21].....	71
Figura 6.2. Esquema del multiplicador Xilinx bajo estudio	71
Figura 6.3. Esquema sin multiplicador	72
Figura 6.4. Consumo del sistema sin multiplicador	73
Figura 6.5. Esquema de trabajo en PlanAhead	74
Figura 6.6. Esquema de trabajo para los multiplicadores Xilinx	75
Figura 6.7. Medidas usando ISE y estrategia de diseño balanceada.....	76
Figura 6.8. Medidas usando ISE y estrategia de diseño balanceada.....	76
Figura 6.9. Medidas usando ISE y estrategia de diseño balanceada.....	77
Figura 6.10. Medidas usando ISE y estrategia de diseño balanceada.....	77
Figura 6.11. Comparativa de las medidas obtenidas con XPower Analyzer	78
Figura 6.12. Comparativa de las medidas obtenidas con Avnet	78
Figura 6.13. Comparativa de las medidas obtenidas midiendo sobre el circuito	79
Figura 6.14. Medidas usando ISE y estrategia de diseño de optimización de área	79
Figura 6.15. Medidas usando ISE y estrategia de diseño de optimización de área	80
Figura 6.16. Medidas usando ISE y estrategia de diseño de optimización de área	80
Figura 6.17. Medidas usando ISE y estrategia de diseño de optimización de área	81
Figura 6.18. Comparativa de las medidas obtenidas con XPower Analyzer	81
Figura 6.19. Comparativa de las medidas obtenidas con Avnet	82
Figura 6.20. Comparativa de las medidas obtenidas midiendo sobre el circuito	82
Figura 6.21. Medidas usando ISE y estrategia de diseño de optimización de potencia	83
Figura 6.22. Medidas usando ISE y estrategia de diseño de optimización de potencia	83
Figura 6.23. Medidas usando ISE y estrategia de diseño de optimización de potencia	84
Figura 6.24. Medidas usando ISE y estrategia de diseño de optimización de potencia	84
Figura 6.25. Comparativa de las medidas obtenidas con XPower Analyzer	85
Figura 6.26. Comparativa de las medidas obtenidas con Avnet	85
Figura 6.27. Comparativa de las medidas obtenidas midiendo sobre el circuito	86
Figura 6.28. Medidas usando ISE y estrategia de diseño de optimización del tiempo de ejecución	86
Figura 6.29. Medidas usando ISE y estrategia de diseño de optimización del tiempo de ejecución	87
Figura 6.30. Medidas usando ISE y estrategia de diseño de optimización del tiempo de ejecución	87
Figura 6.31. Medidas usando ISE y estrategia de diseño de optimización del tiempo de ejecución	88
Figura 6.32. Comparativa de las medidas obtenidas con XPower Analyzer	88
Figura 6.33. Comparativa de las medidas obtenidas con Avnet	89
Figura 6.34. Comparativa de las medidas obtenidas midiendo sobre el circuito	89
Figura 6.35. Medidas usando ISE y estrategia de diseño de optimizar el rendimiento temporal	90
Figura 6.36. Medidas usando ISE y estrategia de diseño de optimizar el rendimiento temporal	90

Figura 6.37. Medidas usando ISE y estrategia de diseño de optimizar el rendimiento temporal	91
Figura 6.38. Medidas usando ISE y estrategia de diseño de optimizar el rendimiento temporal	91
Figura 6.39. Comparativa de las medidas obtenidas con XPower Analyzer	92
Figura 6.40. Comparativa de las medidas obtenidas con Avnet	92
Figura 6.41. Comparativa de las medidas obtenidas midiendo sobre el circuito	93
Figura 6.42. Medidas usando PlanAhead y estrategia de diseño balanceada	93
Figura 6.43. Medidas usando PlanAhead y estrategia de diseño balanceada	94
Figura 6.44. Medidas usando PlanAhead y estrategia de diseño balanceada	94
Figura 6.45. Medidas usando PlanAhead y estrategia de diseño balanceada	95
Figura 6.46. Comparativa de las medidas obtenidas con XPower Analyzer	95
Figura 6.47. Comparativa de las medidas obtenidas con Avnet	96
Figura 6.48. Comparativa de las medidas obtenidas midiendo sobre el circuito	96
Figura 6.49. Medidas usando PlanAhead y estrategia de diseño de optimización de potencia. 97	
Figura 6.50. Medidas usando PlanAhead y estrategia de diseño de optimización de potencia. 97	
Figura 6.51. Medidas usando PlanAhead y estrategia de diseño de optimización de potencia. 98	
Figura 6.52. Medidas usando PlanAhead y estrategia de diseño de optimización de potencia. 98	
Figura 6.53. Comparativa de las medidas obtenidas con XPower Analyzer	99
Figura 6.54. Comparativa de las medidas obtenidas con Avnet	99
Figura 6.55. Comparativa de las medidas obtenidas midiendo sobre el circuito	100
Figura 6.56. Medidas usando PlanAhead y estrategia de diseño de optimización del rendimiento temporal con IOB	100
Figura 6.57. Medidas usando PlanAhead y estrategia de diseño de optimización del rendimiento temporal con IOB	101
Figura 6.58. Medidas usando PlanAhead y estrategia de diseño de optimización del rendimiento temporal con IOB	101
Figura 6.59. Medidas usando PlanAhead y estrategia de diseño de optimización del rendimiento temporal con IOB	102
Figura 6.60. Comparativa de las medidas obtenidas con XPower Analyzer	102
Figura 6.61. Comparativa de las medidas obtenidas con Avnet	103
Figura 6.62. Comparativa de las medidas obtenidas midiendo sobre el circuito	103
Figura 6.63. Medidas usando PlanAhead y estrategia de diseño de optimización del rendimiento temporal sin IOB.....	104
Figura 6.64. Medidas usando PlanAhead y estrategia de diseño de optimización del rendimiento temporal sin IOB.....	104
Figura 6.65. Medidas usando PlanAhead y estrategia de diseño de optimización del rendimiento temporal sin IOB.....	105
Figura 6.66. Medidas usando PlanAhead y estrategia de diseño de optimización del rendimiento temporal sin IOB.....	105
Figura 6.67. Comparativa de las medidas obtenidas con XPower Analyzer	106
Figura 6.68. Comparativa de las medidas obtenidas con Avnet	106
Figura 6.69. Comparativa de las medidas obtenidas midiendo sobre el circuito	107
Figura 6.70. Esquema de trabajo para el multiplicador segmentado	108
Figura 6.71. Medidas usando ISE y estrategia de diseño balanceada.....	109

Figura 6.72. Medidas usando ISE y estrategia de diseño de optimización de área	109
Figura 6.73. Medidas usando ISE y estrategia de diseño de optimización de potencia	110
Figura 6.74. Medidas usando ISE y estrategia de diseño de optimización de tiempo de ejecución	110
Figura 6.75. Medidas usando ISE y estrategia de diseño de optimizar el rendimiento temporal	111
Figura 6.76. Medidas usando PlanAhead y estrategia de diseño balanceada	111
Figura 6.77. Medidas usando PlanAhead y estrategia de diseño de optimización de potencia	112
Figura 6.78. Esquema de trabajo para AES	113
Figura 6.79. Medidas obtenidas con Avnet.....	114
Figura 6.80. Medidas obtenidas con XPower Analyzer	114
Figura 6.81. Esquema de trabajo para FFT.....	115
Figura 6.82. Medidas obtenidas con Avnet.....	116
Figura 6.83. Medidas obtenidas con XPower Analyzer	116
Figura 6.84. Ejemplo de las modificaciones realizadas mediante floorplanning	117
Figura 6.85. Medidas usando PlanAhead y sin aplicar modificaciones.....	118
Figura 6.86. Esquema del circuito usando PlanAhead y sin aplicar modificaciones	118
Figura 6.87. Medidas usando PlanAhead y aplicando modificaciones	119
Figura 6.88. Esquema de circuito usando PlanAhead y aplicando modificaciones	119
Figura 6.89. Medidas usando PlanAhead y aplicando modificaciones	120
Figura 6.90. Esquema del circuito usando PlanAhead y aplicando modificaciones.....	120
Figura 6.91. Medidas usando PlanAhead y sin aplicar modificaciones.....	121
Figura 6.92. Esquema de circuito usando PlanAhead y sin aplicar modificaciones.....	121
Figura 6.93. Medidas usando PlanAhead y aplicando modificaciones	122
Figura 6.94. Esquema del circuito usando PlanAhead y aplicando modificaciones.....	122
Figura 6.95. Medidas usando PlanAhead y aplicando modificaciones	123
Figura 6.96. Esquema del circuito usando PlanAhead y aplicando modificaciones.....	123
Figura 6.97. Medidas usando PlanAhead y aplicando modificaciones	124
Figura 6.98. Esquema del circuito usando PlanAhead y aplicando modificaciones.....	124
Figura 6.99. Medidas usando PlanAhead y aplicando modificaciones	125
Figura 6.100. Esquema del circuito usando PlanAhead y aplicando modificaciones	125
Figura 6.101. Medidas usando PlanAhead y aplicando modificaciones	126
Figura 6.102. Esquema del circuito usando PlanAhead y aplicando modificaciones	126
Figura 6.103. Medidas usando PlanAhead y aplicando modificaciones	127
Figura 6.104. Esquema del circuito usando PlanAhead y aplicando modificaciones	127
Figura 6.105. Medidas usando PlanAhead y aplicando modificaciones	128
Figura 6.106. Esquema del circuito usando PlanAhead y aplicando modificaciones	128
Figura 6.107. Medidas usando PlanAhead y sin aplicar modificaciones.....	129
Figura 6.108. Esquema del circuito usando PlanAhead y sin aplicar modificaciones.....	129
Figura 6.109. Medidas usando PlanAhead y aplicando modificaciones	130
Figura 6.110. Esquema del circuito usando PlanAhead y aplicando modificaciones	130
Figura 6.111. Medidas usando PlanAhead y aplicando modificaciones	131
Figura 6.112. Esquema del circuito usando PlanAhead y aplicando modificaciones	131
Figura 6.113. Medidas usando PlanAhead y aplicando modificaciones	132
Figura 6.114. Esquema del circuito usando PlanAhead y aplicando modificaciones	132

Figura 6.115. Medidas usando PlanAhead y sin aplicar modificaciones.....	133
Figura 6.116. Esquema del circuito usando PlanAhead y sin aplicar modificaciones.....	133
Figura 6.117. Medidas usando PlanAhead y aplicando modificaciones.....	134
Figura 6.118. Esquema del circuito usando PlanAhead y aplicando modificaciones.....	134
Figura 6.119. Medidas usando PlanAhead y aplicando modificaciones.....	135
Figura 6.120. Esquema del circuito usando PlanAhead y aplicando modificaciones.....	135
Figura 6.121. Medidas usando PlanAhead y sin aplicar modificaciones.....	136
Figura 6.122. Esquema del circuito usando PlanAhead y sin aplicar modificaciones.....	136
Figura 6.123. Medidas usando PlanAhead y aplicando modificaciones.....	137
Figura 6.124. Esquema del circuito usando PlanAhead y aplicando modificaciones.....	137
Figura 6.125. Medidas usando PlanAhead y aplicando modificaciones.....	138
Figura 6.126. Esquema del circuito usando PlanAhead y aplicando modificaciones.....	138
Figura 6.127. Medidas usando PlanAhead y sin aplicar modificaciones.....	139
Figura 6.128. Esquema del circuito usando PlanAhead y sin aplicar modificaciones.....	139
Figura 6.129. Medidas usando PlanAhead y aplicando modificaciones.....	140
Figura 6.130. Esquema del circuito usando PlanAhead y aplicando modificaciones.....	140
Figura 6.131. Medidas usando PlanAhead y aplicando modificaciones.....	141
Figura 6.132. Esquema del circuito usando PlanAhead y aplicando modificaciones.....	141
Figura 6.133. Medidas usando PlanAhead y aplicando modificaciones.....	142
Figura 6.134. Esquema del circuito usando PlanAhead y aplicando modificaciones.....	142
Figura 6.135. Medidas usando PlanAhead y sin aplicar modificaciones.....	143
Figura 6.136. Esquema del circuito usando PlanAhead y sin aplicar modificaciones.....	143
Figura 6.137. Medidas usando PlanAhead y aplicando modificaciones.....	144
Figura 6.138. Esquema del circuito usando PlanAhead y aplicando modificaciones.....	144
Figura 6.139. Medidas usando PlanAhead y aplicando modificaciones.....	145
Figura 6.140. Esquema del circuito usando PlanAhead y aplicando modificaciones.....	145
Figura 6.141. Medidas usando PlanAhead y aplicando modificaciones.....	146
Figura 6.142. Esquema del circuito usando PlanAhead y aplicando modificaciones.....	146
Figura 6.143. Medidas usando PlanAhead y sin aplicar modificaciones.....	147
Figura 6.144. Esquema del circuito usando PlanAhead y sin aplicar modificaciones.....	147
Figura 6.145. Medidas usando PlanAhead y aplicando modificaciones.....	148
Figura 6.146. Esquema del circuito usando PlanAhead y aplicando modificaciones.....	148
Figura 6.147. Medidas usando PlanAhead y sin aplicar modificaciones.....	149
Figura 6.148. Esquema del circuito usando PlanAhead y sin aplicar modificaciones.....	149
Figura 6.149. Medidas usando PlanAhead y aplicando modificaciones.....	150
Figura 6.150. Esquema del circuito usando PlanAhead y aplicando modificaciones.....	150
Figura 6.151. Medidas usando PlanAhead y aplicando modificaciones.....	151
Figura 6.152. Esquema del circuito usando PlanAhead y aplicando modificaciones.....	151
Figura 6.153. Medidas usando PlanAhead y aplicando modificaciones.....	152
Figura 6.154. Esquema del circuito usando PlanAhead y aplicando modificaciones.....	152
Figura 6.155. Medidas usando PlanAhead y aplicando modificaciones.....	153
Figura 6.156. Esquema del circuito usando PlanAhead y aplicando modificaciones.....	153
Figura 6.157. Medidas usando PlanAhead y aplicando modificaciones.....	154
Figura 6.158. Esquema del circuito usando PlanAhead y aplicando modificaciones.....	154
Figura 6.159. Medidas usando PlanAhead y sin aplicar modificaciones.....	155

Figura 6.160. Esquema del circuito usando PlanAhead y sin aplicar modificaciones	155
Figura 6.161. Medidas usando PlanAhead y aplicando modificaciones	156
Figura 6.162. Esquema del circuito usando PlanAhead y aplicando modificaciones	156
Figura 6.163. Medidas usando PlanAhead y aplicando modificaciones	157
Figura 6.164. Esquema del circuito usando PlanAhead y aplicando modificaciones	157
Figura 6.165. Medidas usando PlanAhead y aplicando modificaciones	158
Figura 6.166. Esquema del circuito usando PlanAhead y aplicando modificaciones	158
Figura 6.167. Medidas usando PlanAhead y aplicando modificaciones	159
Figura 6.168. Esquema del circuito usando PlanAhead y aplicando modificaciones	159
Figura 6.169. Porcentaje de utilización de los elementos disponibles para multiplicadores embebidos optimizados en área	160
Figura 6.170. Porcentaje de utilización de los elementos disponibles para multiplicadores embebidos optimizados en área	160
Figura 6.171. Porcentaje de utilización de los elementos disponibles para multiplicadores embebidos optimizados en área	161
Figura 6.172. Porcentaje de utilización de los elementos disponibles para multiplicadores embebidos optimizados en área	161
Figura 6.173. Porcentaje de utilización de los elementos disponibles para multiplicadores embebidos optimizados en área	161
Figura 6.174. Porcentaje de utilización de los elementos disponibles para multiplicadores embebidos optimizados en velocidad.....	162
Figura 6.175. Porcentaje de utilización de los elementos disponibles para multiplicadores embebidos optimizados en velocidad.....	162
Figura 6.176. Porcentaje de utilización de los elementos disponibles para multiplicadores embebidos optimizados en velocidad.....	162
Figura 6.177. Porcentaje de utilización de los elementos disponibles para multiplicadores embebidos optimizados en velocidad.....	163
Figura 6.178. Porcentaje de utilización de los elementos disponibles para multiplicadores embebidos optimizados en velocidad.....	163
Figura 6.179. Porcentaje de utilización de los elementos disponibles para multiplicadores usando LUTs optimizados en área	163
Figura 6.180. Porcentaje de utilización de los elementos disponibles para multiplicadores usando LUTs optimizados en área	164
Figura 6.181. Porcentaje de utilización de los elementos disponibles para multiplicadores usando LUTs optimizados en área	164
Figura 6.182. Porcentaje de utilización de los elementos disponibles para multiplicadores usando LUTs optimizados en área	164
Figura 6.183. Porcentaje de utilización de los elementos disponibles para multiplicadores usando LUTs optimizados en área	165
Figura 6.184. Porcentaje de utilización de los elementos disponibles para multiplicadores usando LUTs optimizados en velocidad	165
Figura 6.185. Porcentaje de utilización de los elementos disponibles para multiplicadores usando LUTs optimizados en velocidad	165
Figura 6.186. Porcentaje de utilización de los elementos disponibles para multiplicadores usando LUTs optimizados en velocidad	166

Figura 6.187. Porcentaje de utilización de los elementos disponibles para multiplicadores usando LUTs optimizados en velocidad	166
Figura 6.188. Porcentaje de utilización de los elementos disponibles para multiplicadores usando LUTs optimizados en velocidad	166
Figura 6.189. Porcentaje de utilización de los elementos disponibles para el multiplicador segmentado.....	167
Figura 6.190. Porcentaje de utilización de los elementos disponibles para el multiplicador segmentado.....	167
Figura 6.191. Porcentaje de utilización de los elementos disponibles para el multiplicador segmentado.....	167
Figura 6.192. Porcentaje de utilización de los elementos disponibles para el multiplicador segmentado.....	167
Figura 6.193. Porcentaje de utilización de los elementos disponibles para el multiplicador segmentado.....	168
Figura 6.194. Porcentaje de utilización de los elementos disponibles para el AES	168
Figura 6.195. Porcentaje de utilización de los elementos disponibles para el AES	168
Figura 6.196. Porcentaje de utilización de los elementos disponibles para el AES	168
Figura 6.197. Porcentaje de utilización de los elementos disponibles para el AES	169
Figura 6.198. Porcentaje de utilización de los elementos disponibles para el FFT.....	169
Figura 6.199. Porcentaje de utilización de los elementos disponibles para el FFT.....	169
Figura 6.200. Porcentaje de utilización de los elementos disponibles para el FFT.....	170
Figura 6.201. Porcentaje de utilización de los elementos disponibles para el FFT.....	170
Figura 6.202. Comparación de la frecuencia de síntesis.....	171
Figura 6.203. Comparación de la frecuencia de síntesis.....	171
Figura 6.204. Comparación de la frecuencia de síntesis.....	172
Figura 6.205. Comparación de la frecuencia de síntesis.....	172
Figura 6.206. Comparación de la frecuencia de síntesis.....	173
Figura 6.207. Comparación de la frecuencia de síntesis.....	173
Figura 6.208. Comparación de la frecuencia de síntesis.....	174
Figura 6.209. Comparación de la frecuencia de síntesis.....	174
Figura 6.210. Aumento de consumo variando la temperatura ambiente	175
Figura 6.211. Aumento de la temperatura de unión variando la temperatura ambiente.....	175
Figura 6.212. Reducción de la temperatura de unión aumentando LFM.....	176
Figura 6.213. Aumento de la potencia IOs variando el condensador de carga (C_L).....	177
Figura 6.214. Aumento de la potencia dinámica variando el condensador de carga (C_L)	177
Figura A.1. Spartan-6 LX16 Evaluation Kit. Extraído de [16]	193
Figura A.2. Diagrama de bloques para la Spartan-6 LX16 Evaluation Board. Extraído de [16].	194
Figura A.3. Interfaces de memoria para la Spartan-6 LX16 Evaluation Board. Extraído de [16]	196
Figura A.4. Interfaz LX16 Power Measurement	199
Figura B.1. Indicación de error. Extraído de [17]	201

Índice de tablas

Tabla 1.1. Medidas realizadas	24
Tabla 3.1. Comparativa de resultados para distintas puertas lógicas usando distintas librerías. Extraído de [10]	40
Tabla 3.2. Comparativa de implementación de un misma puerta lógica usando distintos elementos. Extraído de [10]	40
Tabla 4.1. Tabla comparativa de las distintas opciones para el multiplicador de Xilinx	52
Tabla 4.2. Rendimiento para las distintas claves. Extraído de [13]	58
Tabla 5.1. Estrategia de reducción del área	61
Tabla 5.2. Estrategia de mínimo tiempo de ejecución	62
Tabla 5.3. Estrategia de optimización de potencia	62
Tabla 5.4. Estrategia de mejora de rendimiento temporal	63
Tabla 5.5. Todas las opciones editables	66
Tabla 5.7. Estrategia por defecto de PlanAhead	67
Tabla 5.8. Estrategia de mínimo tiempo de ejecución usando IOB	67
Tabla 5.9. Estrategia de mínimo tiempo de ejecución sin usar IOB	68
Tabla 5.10. Estrategia de reducción de área	68
Tabla 5.11. Estrategia de optimización del consumo de potencia	69
Tabla 6.1. Estrategias de diseño aplicadas con ISE	73
Tabla 6.2. Estrategias de diseño aplicadas con PlanAhead	74
Tabla 6.3. Medidas realizadas	74
Tabla 7.1. Comparativa de consumos en ISE	178
Tabla 7.2. Comparativa de consumos en PlanAhead aplicando <i>floorplanning</i>	179
Tabla 7.3. Comparativa de consumos en PlanAhead aplicando <i>floorplanning</i>	179
Tabla 7.4. Comparativa de consumos en PlanAhead aplicando <i>floorplanning</i>	179
Tabla 7.5. Comparativa de consumos en PlanAhead aplicando <i>floorplanning</i>	179
Tabla 7.6. Comparativa de consumos en PlanAhead aplicando <i>floorplanning</i>	180
Tabla 7.7. Comparativa de consumos en PlanAhead aplicando <i>floorplanning</i>	180
Tabla 7.8. Comparativa de consumos en PlanAhead aplicando <i>floorplanning</i>	180
Tabla 7.9. Comparativa de consumos en PlanAhead aplicando <i>floorplanning</i>	180
Tabla 7.10. Comparativa de consumos en ISE	181
Tabla 7.11. Comparativa de consumos en PlanAhead aplicando <i>floorplanning</i>	181
Tabla 7.12. Comparativa de consumos en PlanAhead aplicando <i>floorplanning</i>	182
Tabla 7.13. Comparativa de consumos en ISE	182
Tabla 7.14. Comparativa de consumos en ISE	183
Tabla 7.15. Comparativa de consumos en Xpower Analyzer	183
Tabla 7.16. Comparativa de consumos en Xpower Analyzer	184
Tabla 7.17. Comparativa de frecuencias de síntesis	185
Tabla 7.18. Comparativa de frecuencias de síntesis	185
Tabla 7.19. Comparativa de frecuencias de síntesis	186
Tabla 7.20. Relación de las medidas obtenidas con XPower Analyzer y medidas sobre el circuito	186

Tabla 7.21. Relación de las medidas obtenidas con XPower Analyzer y medidas sobre el circuito	187
Tabla 7.22. Relación de las medidas de obtenidas XPower Analyzer y de las medidas sobre el circuito AES.....	187
Tabla 7.23. Relación de las medidas de obtenidas XPower Analyzer y de las medidas sobre el circuito FFT	187
Tabla 8.1. Medidas realizadas	188

1. Introducción

1.1. Motivación y objetivos

La frecuencia de operación y densidad de puertas de FPGAs actuales convierte la reducción del consumo en una técnica obligada para operaciones a alta velocidad. El consumo en los dispositivos programables actuales puede situarse fácilmente en el orden de vatios, con una tendencia a incrementarse en los próximos años.

La aplicación de transformaciones circuitales para reducir el consumo de potencia tiene los siguientes efectos prácticos:

- Permite reemplazar encapsulados caros, con mayor capacidad de transmisión de calor, por encapsulados más económicos.
- Aumenta la fiabilidad y vida útil del circuito.
- Simplifica o hace innecesaria la refrigeración.
- Aumenta la autonomía de los sistemas portátiles.
- Reduce la complejidad de PCB, la distribución de tensiones y simplifica la fuente de alimentación.

El objetivo del proyecto es analizar la influencia del diseñador cuando asiste a la herramienta en el proceso de *floorplanning*, estudiar el efecto de algunos *settings* de la herramienta, estudiar la importancia de la topología de diversos bloques aritméticos y finalmente evaluar los errores de diferentes técnicas de medición.

Para alcanzar los objetivos planteados, el proyecto se desarrolla de acuerdo a las siguientes tareas:

- **Elección y codificación VHDL de un conjunto de circuitos *benchmark*:** Estos circuitos serán elegidos con el objeto de poder probar diferentes ideas y pueden o no ser circuitos utilizables en una aplicación real.
- **Puesta a punto del banco de medida, basado en FPGA *Spartan 6*:** Se utilizará una tarjeta de desarrollo de Avnet *Xilinx Spartan 6 LX16 Evaluation Kit* y se analizarán las fuentes de error de los diversos métodos de medición.

- **Análisis de la influencia de la herramienta:** Utilizando *Xilinx ISE 13.1* se realizará una comparación sobre un mismo circuito en particular. Éste debe realizar operaciones idénticas sobre los mismos datos. Se variarán algunos *settings* de la herramienta y se analizará su influencia sobre los resultados de consumo.
- **Influencia del diseñador:** Utilizando la herramientas de Xilinx (*PlanAhead 13.1*) se remplazará parte del procesamiento automático por un diseño asistido y se comprobará el efecto sobre el consumo. En particular se analizará la influencia de un *floorplaning* asistido.

Se han realizado 181 medidas de potencia divididas de la siguiente manera:

Circuito	Número de medidas
Multiplicadores Xilinx	139
Multiplicador segmentado	26
AES	8
FFT	8
Total	181

Tabla 1.1. Medidas realizadas

A estas medidas hay que sumar las 20 estimaciones realizadas variando la temperatura de ambiente y las 4 estimaciones variando el condensador de carga, lo que hace un **total de 205 medidas**.

1.2. Organización de la Memoria

La memoria está dividida en los siguientes capítulos:

- Estado del arte: Se repasarán las técnicas relacionadas con este proyecto y sobre las que se sustenta.
- Consumo en circuitos integrados: Se analizará y definirán los aspectos más importantes en el diseño de circuitos de bajo consumo
- *Benchmarks*: Se describirán los distintos circuitos usados para el desarrollo del proyecto.
- Opciones de *ISE/PlanAhead* relacionadas con consumo: Se analizarán las principales estrategias de diseño que presenta el *software* utilizado.
- Medidas: Se describirá el proceso para tomar las medidas, tanto simuladas como sobre el circuito, y se mostrarán los resultados obtenidos.
- Análisis de los resultados: Se evaluarán los resultados obtenidos y se analizarán que opciones son más eficientes.
- Conclusiones: Se expondrán las conclusiones del proyecto.

2. Estado del Arte y Marco Tecnológico

En este capítulo se van repasar las técnicas más destacables relacionadas con este proyecto, como por ejemplo, dispositivos programables y fabricantes.

2.1. FPGAs

Las primeras FPGAs nacen como una evolución de los CPLDs (*Complex Programmable Logic Device*) en el año 1984. Ambos circuitos tienen un gran número de elementos lógicos programables, teniendo los CPLDs una cantidad de puertas lógicas del orden de decenas de miles de puertas lógicas y en un FPGA del orden de cientos de miles e incluso llegando al orden de millones de puertas lógicas.

La principal diferencia no es la cantidad de elementos, sino su arquitectura. La arquitectura de los CPLDs consiste en sumas de productos programables cuyos resultados se mandan a flip-flops. La arquitectura de las FPGAs tiene un gran número de bloques utilizados para realizar operaciones lógicas sencillas. La posibilidad de conexiones entre estos bloques proporciona a las FPGAs una gran flexibilidad. Además, en la mayoría de las FPGAs se pueden encontrar funciones de alto nivel embebidas en la propia FPGA, como por ejemplo multiplicadores.

Gracias a la posibilidad de interconexión de los bloques lógicos de la FPGA permite que desempeñe cualquier función lógica deseada. En los últimos años la tendencia está siendo la de interconectar los bloques lógicos de la FPGA con microprocesadores, generando arquitecturas híbridas.

Las aplicaciones donde más se utilizan las FPGAs son para DSP (*Digital Signal Processing*), sistemas aeroespaciales y de defensa, sistemas de imágenes para medicina, reconocimiento de voz y emulación de hardware.

En el caso de las FPGAs no se habla de programación, sino de diseño hardware. Esto es, definir la función lógica que se realizará en la FPGA. Para ello se usan entornos de desarrollo para diseño de sistemas y su implementación en una FPGA. Los lenguajes usados para los diseños se conocen como HDL (*Hardware Description Language*), entre los que destacan, VHDL y Verilog.

2.2. Principales fabricantes

A continuación se enumeran los principales fabricantes de FPGAs así como algunos de los productos que ofrecen.

Xilinx: Es la principal empresa de investigación y desarrollo de FPGAs y provee de dispositivos electrónicos como soluciones para múltiples aplicaciones como aeroespacial y defensa, automovilismo o medicina.

Entre los principales productos se puede destacar:

- Componentes electrónicos divididos en tres familias: una familia de glue logic (*CoolRunner*), una familia de FPGAs de bajo coste (*Spartan*) y otra de alto rendimiento (*Virtex*)
- Herramientas de diseño: diseño lógico (ISE), diseño de sistemas embebidos (EDK) y para procesamiento digital de señales (*System Generators for DSP*)
- IP cores y diseños de referencia: entre los que cabe destacar el microprocesador *Microblaze*

Altera: Principal competidor de Xilinx. Sus principales productos son:

- Componentes electrónicos divididos en tres familias: una familia de CPLDs (Max), otra familia de FPGAs de bajo coste (Cyclone y Arria) y otra de alto rendimiento (Stratix)
- Herramientas de diseño: para FPGAs tiene las herramientas MAX+PLUS II y Quantus II
- IPs de referencia: procesadores soft-core Nios II y otros interfaces de comunicación

Lattice Semiconductor: lanzó al mercado dispositivos FPGA con tecnología de 90nm. Es proveedor líder en tecnología no volátil, FPGAs basadas en tecnología Flash, con productos de 90nm y 130nm.

Actel: tiene FPGAs basadas en tecnología Flash reprogramable. También ofrece FPGAs con mezcladores de señales basados en Flash.

Atmel: sus productos son reconfigurables. Se enfocaron en proveer microcontroladores AVR con FPGAs en el mismo encapsulado.

3. Consumo en circuitos integrados

3.1. Potencia y energía

El objetivo principal de este capítulo es analizar y definir los aspectos más importantes en el diseño de circuitos de bajo consumo.

La potencia total para el diseño de un componente electrónico consiste en **potencia dinámica** (o activa) y **potencia estática**.

La potencia dinámica es la potencia consumida cuando el dispositivo está activo, esto es, cuando las señales están cambiando sus valores.

La potencia estática es la potencia consumida cuando el dispositivo está encendido pero las señales no cambian de valor. En los dispositivos CMOS la potencia estática es debido a las fugas.

La primera y principal fuente de consumo de **potencia dinámica** es la potencia de transición, relacionada con la carga y descarga de los condensadores.

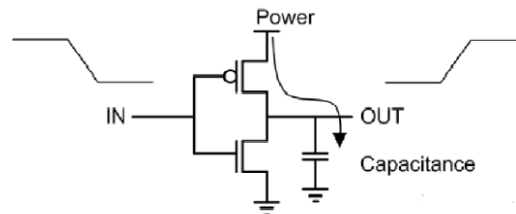


Figura 3.1. Potencia dinámica. Extraído de [11]

Si C es el condensador de carga y V es la tensión de alimentación, se puede deducir que la potencia dinámica es igual a:

$$P_{dyn} = \frac{Energy}{transition} \cdot f = C \cdot V^2 \cdot P_{trans} \cdot f_{clock}$$

Donde f es la frecuencia de transiciones, P_{trans} es la probabilidad de una transición en la salida y f_{clock} es la frecuencia del reloj del sistema.

Si se define:

$$C_{eff} = P_{trans} \cdot C$$

Se puede describir la potencia dinámica para un nodo, con una expresión más familiar:

$$P_{dyn} = C_{eff} \cdot V^2 \cdot f_{clock}$$

Observar que la potencia de transición no es función del tamaño del transistor, pero sí que depende de la actividad de transiciones y del condensador de carga, por lo tanto, depende de los datos.

En adición a la potencia de transición, la potencia interna (o de cortocircuito) también contribuye a la potencia dinámica. En la siguiente figura se muestran las corrientes internas de transición. La potencia interna consiste en las corrientes de cortocircuito que suceden cuando los transistores NMOS y los PMOS están en ON al mismo tiempo, así como la corriente requerida para cargar la capacitancia interna de la celda.

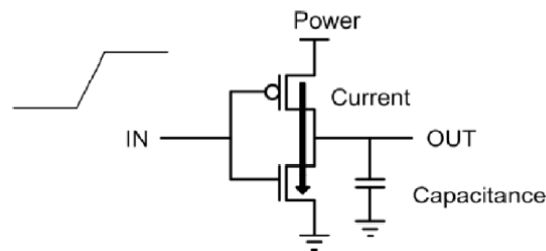


Figura 3.2. Corriente interna de transición. Extraído de [11]

Si se añade la expresión de la potencia interna a la ecuación anterior, se puede describir la potencia dinámica como:

$$P_{dyn} = \underbrace{(C_{eff} \cdot V^2 \cdot f_{clock})}_{\text{Potencia de transición}} + \underbrace{(t_{sc} \cdot V \cdot I_{peak} \cdot f_{clock})}_{\text{Potencia de cortocircuito}}$$

Donde t_{sc} es la duración temporal de la corriente de corto circuito y I_{peak} es la corriente interna de transición total (corriente de cortocircuito más la corriente requerida para cargar la capacitancia interna).

En tanto en cuanto se pueda mantener la corriente de corto circuito a cero, se mantendrá t_{sc} a cero y la potencia dinámica será dominada por la potencia de transición. Por esta razón se puede simplificar usando la fórmula de la potencia de transición:

$$P_{dyn} = C_{eff} \cdot V^2 \cdot f_{clock}$$

Existen numerosas técnicas de arquitectura, de diseño lógico y de diseño de circuito que permiten reducir la potencia para una función implementada en particular. Estas técnicas se enfocan en los componentes de tensión y frecuencia de la ecuación, así como reducir la dependencia entre los datos y la actividad de transición.

Debido a que la tensión afecta de manera cuadrática a la potencia, decrementar la potencia de alimentación es una opción altamente recomendada para reducir la potencia dinámica. Pero hay que tener en cuenta que la velocidad de las puertas decrece con la tensión de alimentación. El diseñador tendrá que usar diversas técnicas para minimizar este efecto, como por ejemplo:

- Para bloques que no necesitan una velocidad en particular, como los periféricos, se pueden alimentar con un tensión menor que otros bloques que necesitan una velocidad de funcionamiento en concreto. Esto es conocido como multi-tensión.
- Para procesadores se puede proporcionar un tensión de alimentación variable: durante las tareas que requieren picos de rendimiento, se le suministra una alimentación más alta podrá tener frecuencias de reloj mayores. Para tareas que requieren menor rendimiento, se le administra un tensión menor por tanto tendrá menor velocidad de reloj. Esta aproximación es conocida como escala de tensión.

A continuación se desarrollaran diversos conceptos que serán de utilidad para el estudio del consumo de circuitos electrónicos.

3.1.1. Glitches

Este efecto se da cuando una entrada cambia, causando múltiples transiciones en la salida de la puerta. Estos eventos, conocido como *glitches*, son costosos cuando la carga o descarga de un condensador no tiene impacto en el resultado final.

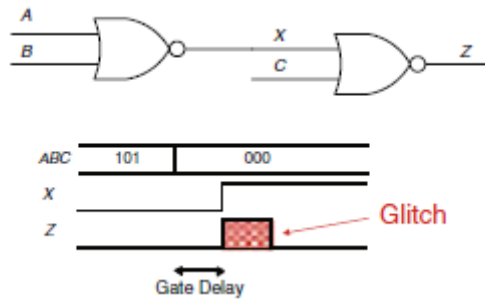


Figura 3.3. Ejemplo de glitches. Extraído de [10]

El resultado es correcto, pero hay un gasto extra de potencia disipada.

En este otro ejemplo también se observan *glitches* así como una solución para resolver el problema de los *glitches*.

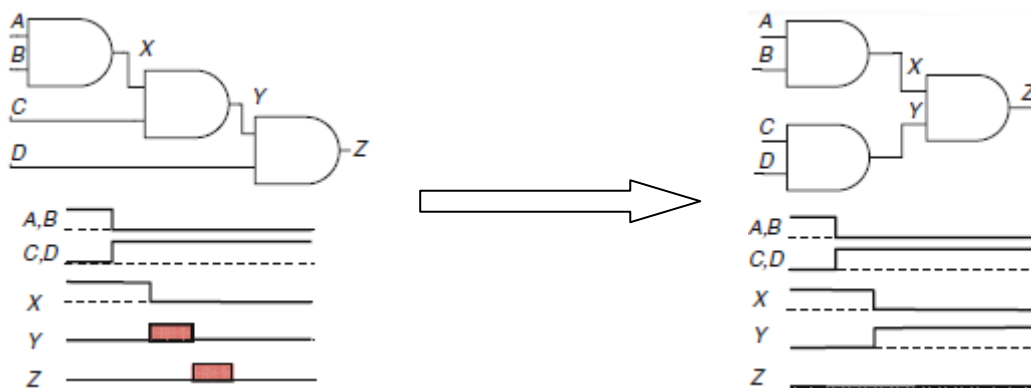


Figura 3.4. Reducción de glitches. Extraído de [10]

Son circuitos equivalentes, pero topológicamente diferentes. Rediseñando la red de modo que todos los tiempos de llegada sean iguales, se reduciría altamente el número de transiciones superfluas.

3.1.2. Corrientes de cortocircuito

Dado el siguiente circuito:

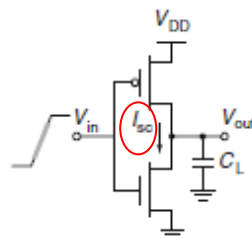


Figura 3.5. Corriente de cortocircuito. Extraído de [10]

En la rampa de subida de la señal de entrada (V_{in}), durante el cambio se genera una corriente entre V_{DD} y GND durante un corto periodo.

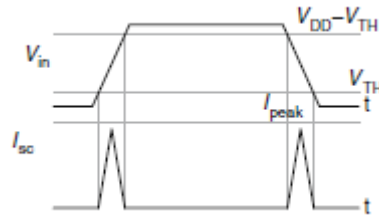


Figura 3.6. Corriente de cortocircuito vs tensión de entrada. Extraído de [10]

La potencia extra disipada (P_{sc} , short circuit) es proporcional a la cantidad de cambios de estado que existan:

$$P_{sc} \approx f$$

La corriente de corto circuito depende del valor de C_L , dependiendo de si su valor es grande o pequeño:

- C_L grande:

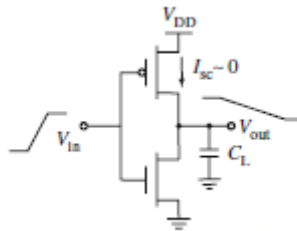


Figura 3.7. Corriente de cortocircuito para C_L grande. Extraído de [10]

La duración de la bajada en V_{out} es significativamente mayor que V_{in} . De esta manera la entrada pasa al estado de transición antes de que la salida comience a cambiar. Así la tensión del PMOS (transistor superior) es aproximadamente 0 durante este periodo. Por tanto, aumentado C_L se disminuye I_{sc} .

- C_L pequeño:

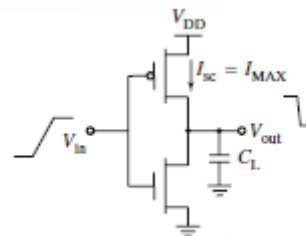


Figura 3.8. Corriente de cortocircuito para C_L pequeño. Extraído de [10]

En este caso, con un condensador pequeño el tiempo de bajada de la señal de salida es más pequeño que el tiempo de subida de la entrada. La tensión en el PMOS es igual a V_{DD} en la

mayor parte del tiempo de transición, garantizando una corriente de cortocircuito máxima. Este es el peor de los casos, por tanto, disminuyendo C_L aumentará I_{sc} .

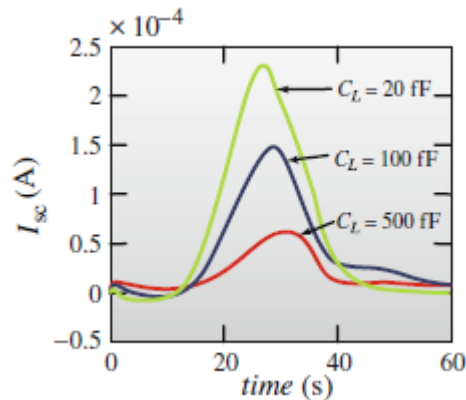


Figura 3.9. Corrientes de cortocircuito para diferentes C_L . Extraído de [10].

Este análisis dependiente del valor de C_L puede llevar a confusión. La potencia disipada debido a la corriente de cortocircuito se minimiza haciendo que el tiempo de bajada de la señal de salida sea muy grande, pero esto ralentiza el circuito y causa grandes corrientes de cortocircuito en la conexión de las puertas. Así pues, la potencia disipada debido a las corrientes de cortocircuito se minimiza haciendo coincidir los tiempos de subida/bajada de la señal de entrada y de la de salida.

3.1.3. Método tradicional vs método actual en el diseño

Tradicionalmente el objetivo principal era el maximizar el rendimiento. Las tensiones de alimentación y de umbral eran fijadas para aumentar el rendimiento y se pensaba que un aumento del ciclo de reloj estaba directamente relacionado con un aumento del rendimiento.

Las nuevas filosofías de diseño se basan en que un aumento del rendimiento requiere de un muy alto consumo de potencia y no siempre es realizable. Se abandona la idea de que un aumento en el ciclo de reloj equivale a un aumento de rendimiento. El diseño se convierte en un ejercicio de equilibrio entre velocidad y energía (o potencia).

3.1.4. Relación entre potencia y retardo

El compromiso entre potencia y retardo es un aspecto importante en cualquier diseño. Dado un nivel de actividad, la potencia se reduce mientras el retardo se mantiene constante cuando V_{DD} y V_{TH} se reducen simultáneamente, como ocurriría al moverse de A a B en la figura 3.10:

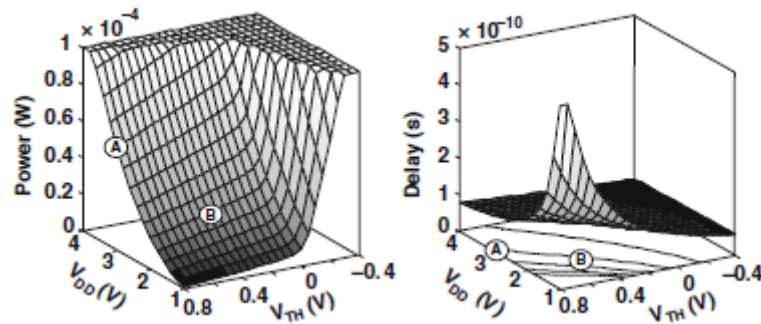


Figura 3.10. Relación entre V_{DD} , V_{TH} , potencia y entre V_{DD} , V_{TH} y retardo. Extraído de [12]

3.1.5. El espacio Energía-Retardo

Los contornos de idénticas energías son más evidentes en una representación de dos dimensiones. Es particularmente interesante el punto donde se encuentra la mínima energía. Si se baja la tensión por debajo de este punto las fugas energéticas se hacen muy importantes y el rendimiento se deteriora rápidamente.

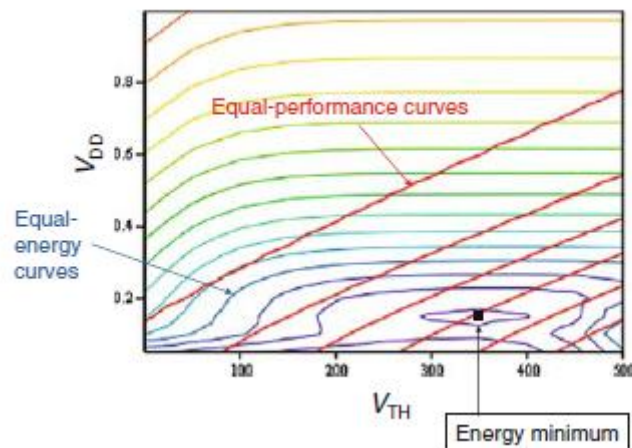


Figura 3.11. Espacio Energía-Retardo. Extraído de [10]

3.2. Diseño temporal. Técnicas a nivel de circuito

Uno de los objetivos de las técnicas de diseño actuales es el compromiso energía-retardo. Para conseguir este compromiso también será importante la optimización de potencia dinámica y de potencia estática.

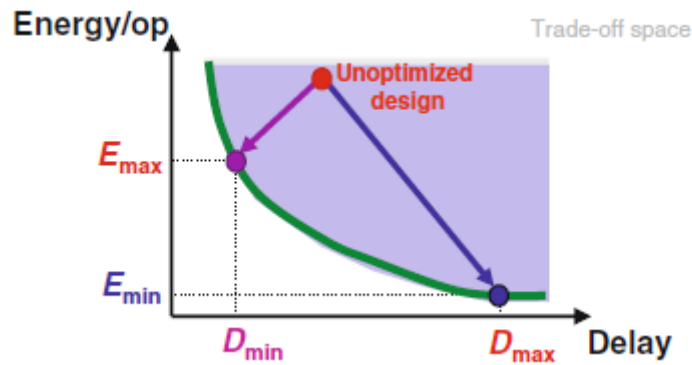


Figura 3.12. Puntos de optimización Energía-Retardo. Extraído de [8]

El objetivo es maximizar el rendimiento para una energía dada o minimizar la energía para un rendimiento dado. En esta gráfica es sencillo encontrar la curva óptima, sin embargo, en la realidad es bastante más complejo. El problema es que existen muchos parámetros para ajustar, unos son continuos (tamaño de los transistores, umbrales de tensión...) y otros son discretos (estilos lógicos, topología y microarquitectura). En teoría es posible considerar todos los parámetros a la vez, pero en la práctica no. Para resolver este problema existen metodologías de diseño para gestionar la complejidad. Son dos las metodologías usadas, abstracción (ocultando los detalles) y jerárquico (construir grandes entidades a partir de otras más pequeñas). A menudo estas dos metodologías van de la mano.

La pila de abstracción para el diseño típico de un circuito integrado es la siguiente:

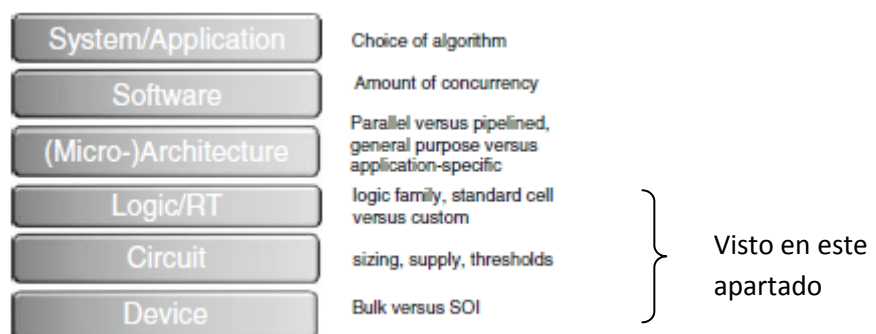


Figura 3.13. Pila de abstracción. Extraído de [10]

Estos bloques no son independientes y la optimización hace que se reorganicen:

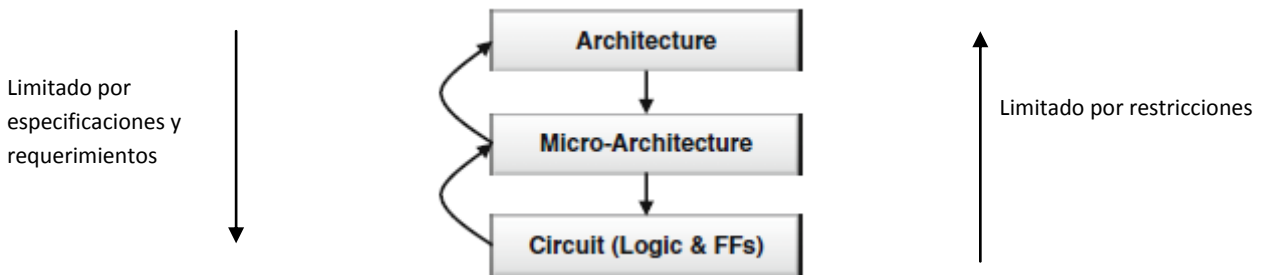


Figura 3.14. Top-down vs bottom-up. Extraído de [10]

Para el diseño optimizado existirán dos posibilidades dependiendo de las limitaciones: top-down, si las limitaciones vienen dadas por los requerimientos y especificaciones desde las capas superiores y bottom-up, si las limitaciones vienen dadas por restricciones y se propagan desde las capas más bajas.

3.2.1. Sensibilidad Energía-Retardo

La sensibilidad energía-retardo es una herramienta para evaluar la efectividad de los cambios en varias variables de diseño. Para este estudio se basa en la figura 3.12 de energía-retardo.

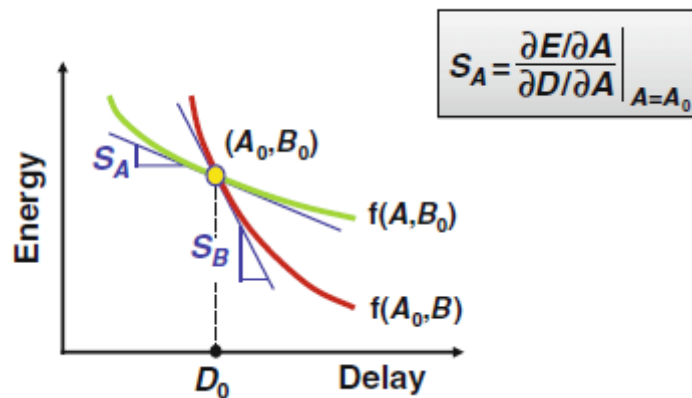


Figura 3.15. Sensibilidad Energía-Retardo. Extraído de [8]

Considerando el punto de operación (A_0, B_0) , donde A y B son las variables de diseño bajo estudio, la sensibilidad de cada una de estas variables será la curva obtenida por un pequeño cambio en esa variable.

3.2.2. Reducción de energía activa (dinámica)

$$E_{\text{active}} \sim \alpha \cdot C_L \cdot V_{\text{swing}} \cdot V_{DD}$$

$$P_{\text{active}} \sim \alpha \cdot C_L \cdot V_{\text{swing}} \cdot V_{DD} \cdot f$$

Figura 3.16. Energía activa y potencia activa. Extraído de [10]

A partir de estas ecuaciones de la figura 3.16 se desprende como reducir la energía activa:

- Reducir actividad (α): Reduciendo los cambios mediante transformaciones (dependen de la topología del circuito) y reduciendo el *glitching*.
- Reducir el tamaño de los transistores (C_L). Condensadores más pequeños tienen una menor capacitancia.
- Reduciendo las tensiones: Reducir tensiones de alimentación (V_{DD}) a expensas de la velocidad de reloj.

El aspecto más importante es el de reducir las tensiones, ya que tiene un efecto cuadrático, el resto son lineales. Para una topología de circuito dada, el compromiso más interesante es entre la tensión de alimentación y el tamaño de las puertas lógicas, ya que estas afectan a la energía y al rendimiento. Sin embargo, la tensión juega un papel secundario si en el impacto del rendimiento no hay influencia de la energía dinámica.

Hay que tener en cuenta que reduciendo la alimentación en un camino crítico se reduce la frecuencia de operación. La reducción de alimentación en un camino no crítico se puede reducir más libremente.

3.2.3. Múltiples tensiones de alimentación

Existen dos formas de asignación de la alimentación, por nivel de bloques y suministro múltiple dentro de un bloque.

La asignación de alimentación por nivel de bloques se basa en que los bloques con mayor nivel de rendimiento o menor latencia son implementados con tensiones (V_{DD}) altas. Las funciones lentas están implementadas con tensiones (V_{DD}) bajas. Por todo esto existirán redes de alimentación.

En el caso de suministro de múltiples alimentaciones dentro de un bloque, los caminos no críticos están alimentados a bajo tensión. En este caso el reto es el diseño físico.

3.2.4. Distribuyendo múltiples fuentes de alimentación

Existen dos maneras, *conventional* y *shared n-well*:

3.2.4.1. Conventional

Consiste en agrupar puertas con la misma alimentación. Esto funciona bien para un modelo de "isla de tensión" donde una alimentación es elegida para un módulo completo. Se requiere cable extra, el cual reduce la potencia, por tanto es importante mantener la conectividad espacial de las conexiones.

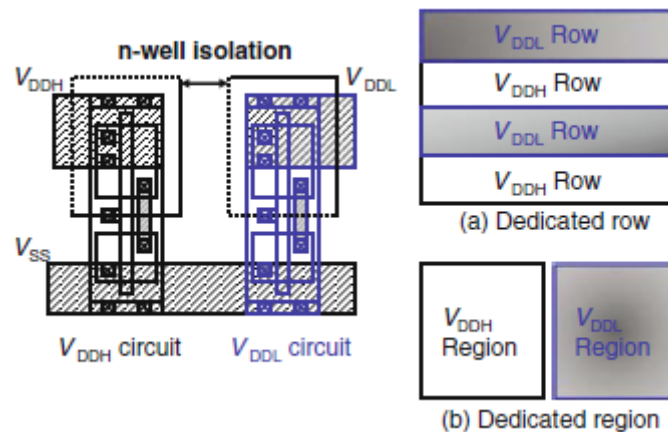


Figura 3.17. Diseño para la distribución *conventional*. Extraído de [10]

3.2.4.2. Shared n-well

La aproximación más versátil es rediseñar las celdas y tener vías V_{DDL} y V_{DDH} en cada celda. En este diseño no importa la partición del área. V_{DDL} y V_{DDH} pueden estar colindando una con la otra.

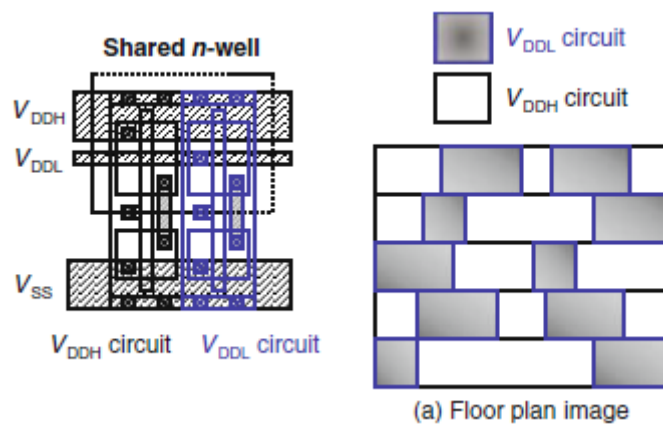


Figura 3.18. Diseño para la distribución *shared n-well*. Extraído de [10]

3.2.5. Tecnología de mapeo

En los diseños actuales la fase de "tecnología de mapeo" es la selección de las librerías que se usarán para la implementación de una función lógica dada. En la última década ha sido común entender que las puertas simples son buenas desde un punto de vista del rendimiento. Desde el punto de vista de la energía, la puertas complejas son más atractivas, ya que las capacitancias intrínsecas son menores que la capacitancia de las rutas que unen puertas simples. Es por ello que las puertas complejas son usadas preferentemente en caminos no críticos.

Lo explicado anteriormente se ve con un ejemplo en el que se comparan dos librerías, una enfocada al bajo consumo y otra a un alto rendimiento:

Gate type	Area (cell unit)	Input cap. (fF)	Library 1:	Library 2:
			High-Speed	Low-Power
INV	3	1.8	Average delay (ps)	Average delay (ps)
			$7.0 + 3.8C_L$	$12.0 + 6.0C_L$
NAND2	4	2.0	$10.3 + 5.3C_L$	$16.3 + 8.8C_L$
NAND4	5	2.0	$13.6 + 5.8C_L$	$22.7 + 10.2C_L$
NOR2	3	2.2	$10.7 + 5.4C_L$	$16.7 + 8.9C_L$

(delay formula: C_L in fF)

Tabla 3.1. Comparativa de resultados para distintas puertas lógicas usando distintas librerías. Extraído de [10]

Se observa que dependiendo de la librería usada para el diseño los resultados de los retardos son distintos.

En este otro ejemplo se usan dos librerías para implementar una puerta AND de cuatro entradas (AND4). Por un lado usaremos una NAND4 + INV y por otro lado NAND2 + NOR2:

four-input AND	(a) NAND4 + INV	(b) NAND2 + NOR2
Area	8	11
HS: Delay (ps)	$31.0 + 3.8C_L$	$32.7 + 5.4C_L$
LP: Delay (ps)	$53.1 + 6.0C_L$	$52.4 + 8.9C_L$
Sw Energy (fF)	$0.1 + 0.06C_L$	$0.83 + 0.06C_L$

Tabla 3.2. Comparativa de implementación de un misma puerta lógica usando distintos elementos. Extraído de [10]

- Área: La puerta de cuatro entradas es más compacta que usar dos puertas de dos entradas.
- Tiempo: El uso de puertas complejas en este ejemplo ligeramente más rápido. Los dos caso tienen dos estados, pero la puerta INV tiene mejor rendimiento que la puerta NOR2. Para un mayor número de bloques complejos, las puertas simples tendrán mejor rendimiento.
- Energía: Cuantas más conmutaciones existen (caso de dos puertas de dos entradas) se aumenta el consumo de energía. Las librerías de bajo consumo tienen peores retardos pero menores fugas.

La tecnología de mapeo también se encarga de la asignación de pines y su colocación espacial. Habrá que optimizar y manipular la topología de la red para minimizar el retardo, la potencia o el área. Para ello existirá en equilibrio, no solo entre potencia y retardo sino también en ocasiones entre la potencia de diferentes componentes y su actividad y capacidad.

A continuación se muestran dos ejemplos de reestructuración lógica dentro de la tecnología de mapeo:

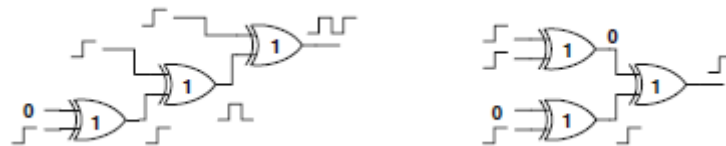


Figura 3.19. Ejemplo de reestructuración lógica. Extraído de [10].

Se reestructura la red y se obtiene un equilibrio de los caminos que repercutirá en un menor retardo.

En este ejemplo se muestra el uso de buffers no inversores:

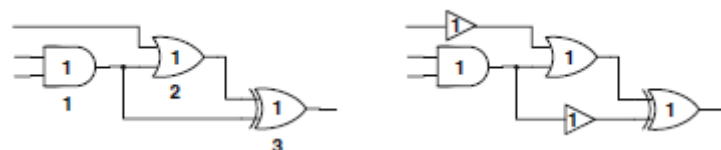


Figura 3.20. Ejemplo de reestructuración lógica usando inversores. Extraído de [10].

Mediante la introducción de buffers no inversores en los caminos más rápidos se minimiza el *glitching*. Sin embargo, la introducción de buffers es un proceso que hay que hacerlo con cuidado porque añade capacitancia extra.

3.2.6. Lecciones del circuito de optimización

Basándose en lo visto anteriormente, ahora se puede trazar algunas directrices sobre la optimización de energía-retardo en un circuito a nivel lógico:

- La optimización sobre múltiples parámetros de diseño es posible basándose en la sensibilidad vista en el circuito de optimización.
- El máximo rendimiento es muy ineficiente desde el punto de vista de la potencia. Alrededor del 70% de reducción de energía tiene una penalización del 20% de retardo.
- La elección entre los parámetros de dimensionado y la tensión de alimentación depende de la topología del circuito.

Sin embargo, las fugas no han sido consideradas hasta ahora. Las fugas tienen una importante contribución en el consumo de potencia de los circuitos actuales.

3.2.7. Reduciendo las fugas (leakage). Longitud del canal

Cuando se intenta manipular la corriente de fuga, el diseñador tiene múltiples opciones a su disposición, las cuales son muy similares a las usadas en la optimización de la potencia dinámica: tamaño de los transistores, tensiones de umbral y de alimentación. La forma en la que afectan estos factores en la corriente de fuga es distinto que en el caso de la optimización de potencia dinámica.

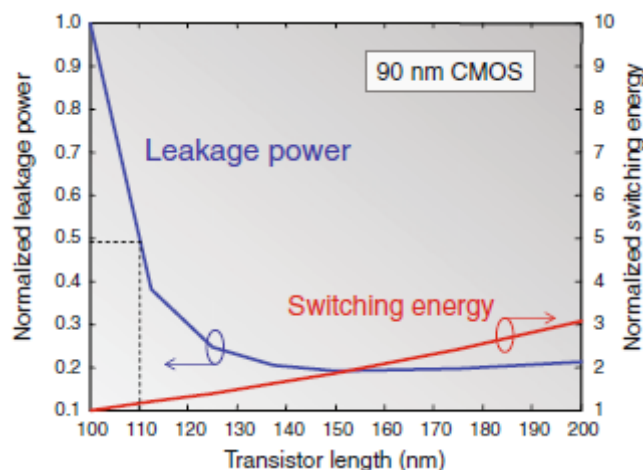


Figura 3.21. Resultados para distintas longitudes del transistor, partiendo de un CMOS de 90 nm. Extraído de [10]

Partiendo de CMOS con longitud de canal de 90 nm, si se incrementa la longitud de canal en un 10%, se reduce la potencia de fuga en un 50% y se aumenta la potencia dinámica un 18%.

Si se dobla la longitud del canal del transistor, la potencia de fuga se reduce por 5. Esto es interesante cuando no se tiene requerimientos de espacio.

3.2.8. Resumen

- El compromiso energía-retardo es un reto que se puede plantear como un problema de optimización. Existen múltiples opciones disponibles para el diseñador: tamaño de los transistores, múltiples tensiones de alimentación, tensiones de umbral y topología de los circuitos.
- La gráfica energía-retardo es el mecanismo perfecto para analizar el compromiso energía-retardo.

3.3. Diseño temporal. Arquitectura, algoritmos y sistemas

La optimización en la arquitectura puede permitir una minimización de la potencia más efectiva manteniendo el rendimiento. Ejemplos de estas optimizaciones son, la posibilidad de reducir la tensión de alimentación, reducir las tasas de conmutación o reducir las fugas. Las optimizaciones a altos niveles de abstracción suelen tener un gran impacto en la potencia.

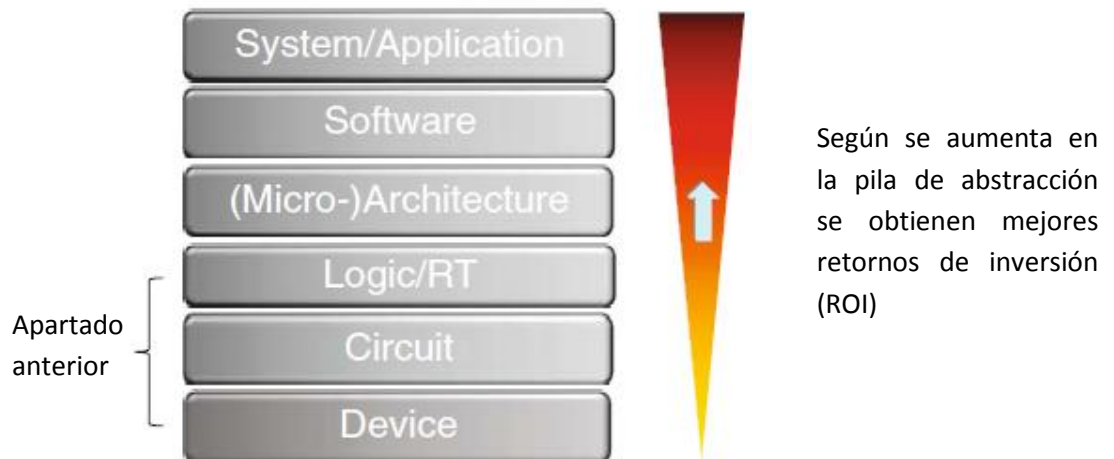


Figura 3.22. Pila de abstracción. Extraído de [10]

En el nivel de circuito los parámetros usados son tamaño de los transistores y tensión de alimentación y estos parámetros son continuos. En los niveles más altos de abstracción las opciones son discretas: qué topología de sumador usar o cuanto *pipeline* usar. Estas opciones discretas ayudan a ampliar el espacio energía-retardo.

Por ejemplo, cuando dos topologías de sumador están disponibles cada una de ellas tiene su propia curva óptima de energía-retardo. El espacio de diseño es ahora la combinación de las dos y la nueva curva energía-retardo será de esta manera:

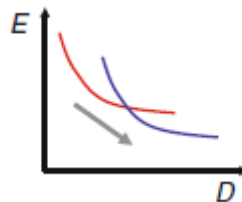


Figura 3.23. Gráfica Energía-Retardo. Extraído de [10]

En algunos casos, una función es siempre superior a la otra (por ejemplo al eliminar ineficiencias), lo que hace que la elección sea muy simple:

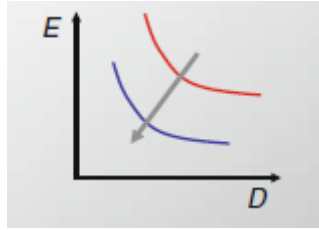


Figura 3.24. Gráfica Energía-Retardo al eliminar ineficiencias. Extraído de [10]

Otro caso es en el que la exploración consiste en muchas opciones discretas. En este caso se deriva la curva óptima de energía-retardo (E-D) seleccionando la mejor opción disponible por cada nivel de rendimiento:

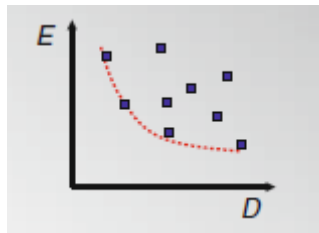


Figura 3.25. Gráfica Energía-Retardo. Extraído de [10]

Existen técnicas de concurrencia que se usan para reducir la potencia. Estas son, paralelismo y pipeline.

3.3.1. Reduciendo la tensión de alimentación

Un típico ejemplo del equilibrio energía-retardo es el que se va a desarrollar a continuación. Consiste en permitir aumentos grandes de tensión, o equivalentemente, proporcionar una mejora del rendimiento con una energía por operación (EOP) fija.

Para demostrar este concepto se parte de un diseño de referencia básico, el cual opera a una tensión de alimentación V_{DDref} y a una frecuencia f_{ref} . El promedio de la capacitancia del diseño es C_{ref} .

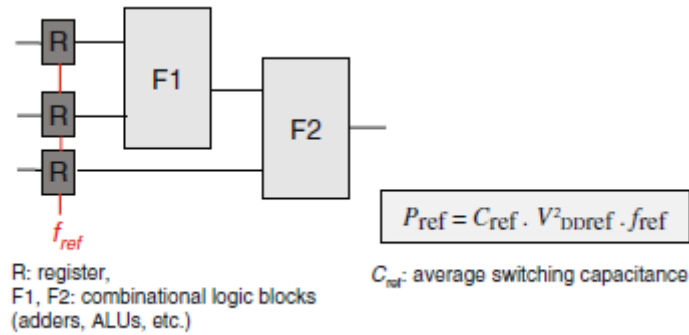


Figura 3.26. Circuito a estudio para aplicar paralelismo. Extraído de [4]

Una implementación en paralelo del mismo diseño anterior consiste básicamente en una replicación de este. Las entradas del segundo bloque paralelo es una réplica del primer bloque. Es necesario un demultiplexor a la salida, para recombinar las salidas y producir un único flujo de salida.

Debido al paralelismo las ramas pueden operar ahora a la mitad de velocidad, por tanto $f_{par} = f_{ref}/2$. Esto reduce el requerimiento de retardo y permite una reducción de la tensión de alimentación por un factor de ϵ_{par} . Esto es un efecto cuadrático que hace esta técnica muy efectiva. La multiplexación en la salida es típicamente muy pequeña, especialmente cuando el paralelismo se aplica en bloques grandes.

Hay que observar que el incremento de gasto ($ov = overhead$) debido a la capacitancia de conmutación es mínimo, lo que no ocurre con el área.

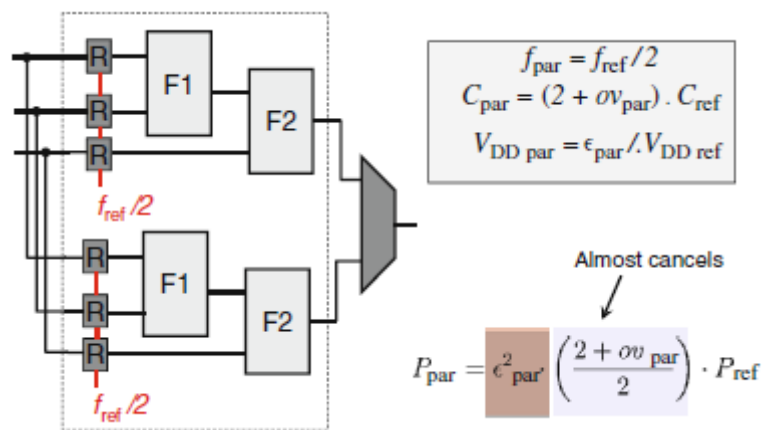


Figura 3.27. Implementación de paralelismo. Extraído de [4]

- Funcionando más despacio, se reduce el requerimiento de tensión de alimentación.
- Mejora cuadrática en la reducción de la potencia.

3.3.2. Implementación de pipeline

Otras formas de introducir concurrencia igual de efectiva que la reducción de tensión de alimentación es el *pipeline*, en el cual cada bloque trabaja concurrentemente, insertando registros entre los bloques. El área del *pipeline* es menor que en caso del paralelismo. Sin embargo, debido a los nuevos registros que se introducen, aumenta la capacitancia en la ruta de datos y el número de nodos. Asumiendo un incremento del gasto (*overhead*) del 10% debido al pipeline, se consigue un ahorro de potencia similar al obtenido con el paralelismo y con un gasto de espacio sustancialmente menor:

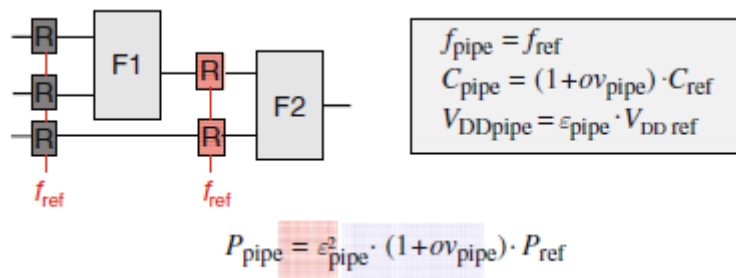


Figura 3.28. Implementación de pipeline. Extraído de [4]

Assuming $P_{pipe} = 0.66^2 \cdot 1.1 \cdot P_{ref} = 0.48 P_{ref}$
 $ov_{pipe} = 10\%$ $P_{pipe4} = 0.52^2 \cdot 1.1 \cdot P_{ref} = 0.29 P_{ref}$

Figura 3.29. Mejoras usando pipeline. Extraído de [4]

3.3.3. Incremento de la concurrencia

La idea de usar la concurrencia como forma de aumentar el rendimiento manteniendo la potencia es algo que a partir del 2004 se empezó a usar por todos los fabricantes de microprocesadores, lo que condujo a una multitud de arquitecturas de multi-núcleo.

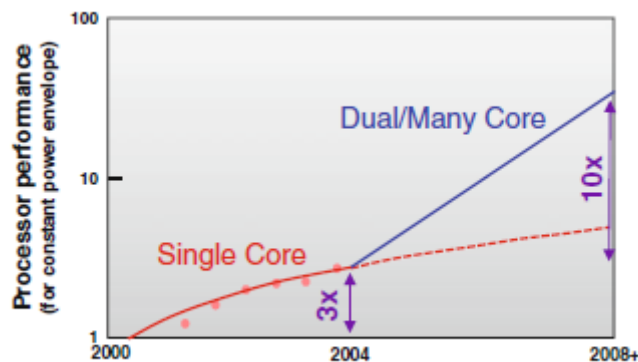


Figura 3.30. Aumento del rendimiento en los microprocesadores debido al paralelismo. Extraído de [5]

La idea del multi-núcleo (multi-core) empezó con la arquitectura de doble núcleo y rápidamente se expandió a los cuatro y más núcleos.

3.3.4. Mejorando la eficiencia computacional

Distintas implementaciones para una función dada pueden ser ineficientes y pueden ser reemplazadas por versiones más eficientes con menores penalizaciones en energía o retardo.

Una mejora de eficiencia computacional es hacer coincidir computación y arquitectura, se ilustra con un ejemplo de cómo calcular un polinomio de segundo grado. En el estilo de procesamiento tradicional de Von-Neumann el cálculo se realiza en un conjunto de instrucciones secuenciales, multiplexadas al final en una ALU genérica. La arquitectura del cálculo y la topología del algoritmo están completamente desasociadas y tienen poco en común.

Otra opción es tener el algoritmo y la arquitectura coincidentes uno con el otro directamente. La ventaja de esto es que cada operación y cada comunicación se realizan sin sobrecoste (*overhead*). Este estilo de arquitectura programable se llama "programación espacial" y es el más adecuado para plataformas de hardware reconfigurables (dibujo de la izquierda).

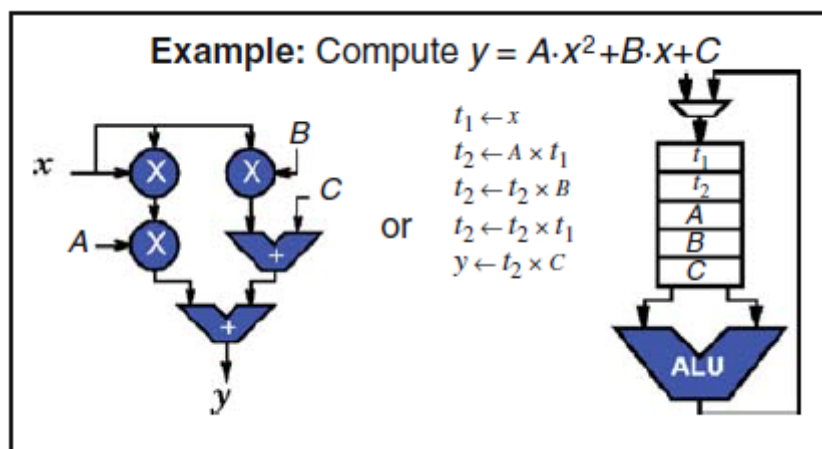


Figura 3.31. Programación espacial vs programación no espacial. Extraído de [10]

También merece mencionar el concepto de localidad. Mantener la *localidad de referencia* es otro mecanismo para incrementar la eficiencia de una arquitectura. Esto no es directamente útil para la potencia, pero ayuda al rendimiento y al área. Cada vez que una instrucción tiene que recorrer una larga distancia, conlleva un coste de energía y retardo. Mantener las instrucciones más usadas o más relevantes cerca del lugar donde son procesadas es una buena

idea. Esto es la principal motivación que está detrás de la construcción de la jerarquía de memoria, así como los multiniveles de caches.

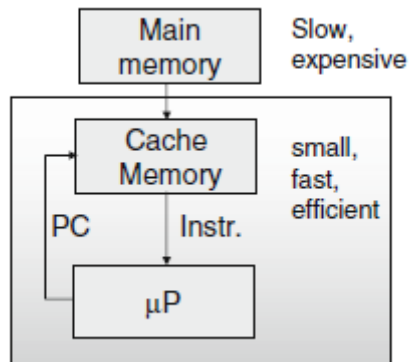


Figura 3.32 Ejemplo de jerarquía de memoria. Extraído de [10]

4. Benchmarks

En este capítulo se va a describir los distintos circuitos usados para el desarrollo del proyecto. Estos son multiplicadores de Xilinx de 32 bits (sintetizados en LUTs y multiplicadores embebidos de la FPGA), multiplicador segmentado de tantos niveles como bits tenga cada una de las entradas, AES (Advanced Encryption Standard) y FFT (Fast Fourier Transform).

4.1. Core Multiplicador Xilinx

Existen un conjunto de multiplicadores que vienen en el *Xilinx ISE 13.1* y han sido desarrollados por Xilinx. Uno de estos es el *LogiCORE IP Multiplier v11.2*. Se puede acceder a este multiplicador desde el ISE así como a sus propiedades.

El multiplicador puede ser configurado para alguna de las siguientes arquitecturas:

- Paralelo: El multiplicador acepta entradas en los buses A y B y generar el producto de estos dos valores. Varias implementaciones son ofrecidas para permitir un balance entre partes lógicas, dedicando recursos de multiplicadores embebidos y maximizando la frecuencia de reloj realizable.
- Coeficientes constantes: El multiplicador acepta datos por el bus A y lo multiplica por una constante definida. El multiplicador puede ser construido desde memoria distribuida, bloques de memoria junto con partes lógicas o desde multiplicadores embebidos.

La distribución de los pines es la siguiente:

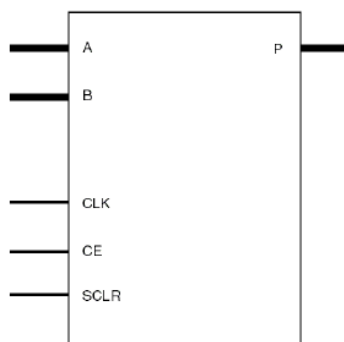


Figura 4.1. Pines del multiplicador. Extraído de [19]

Signal	Direction	Description
A[N-1:0]	Input	A operand input bus, N bits wide
B[M-1:0]	Input	B operand input bus, M bits wide (parallel multipliers only)
CLK	Input	Rising-edge clock input
CE	Input	Active high Clock Enable
SCLR	Input	Active high Synchronous Clear (SCLR/CE priority is configurable)
P[X:Y]	Output	Product Output – bit X down to bit Y

Figura 4.2. Pines del multiplicador. Extraído de [19]

Dispone de un menú gráfico para configurar las opciones del multiplicador:

Figura 4.3. Menú de configuración del multiplicador

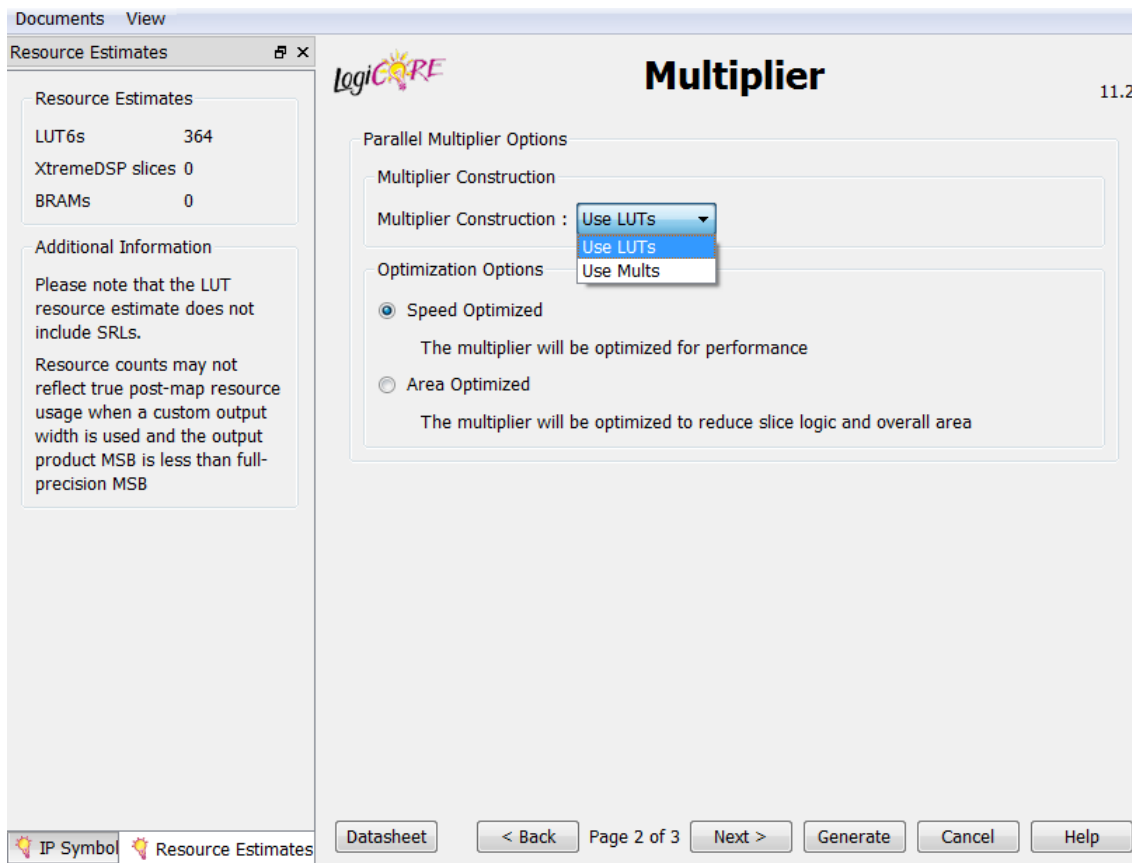


Figura 4.4. Menú de configuración del multiplicador

Usando LUTs o multiplicadores embebidos, se puede elegir entre las opciones de optimizar en velocidad o en área. En el caso de usar LUTs, la optimización de velocidad consistirá en una optimización del rendimiento y la optimización de área consistirá en una reducción de los slices usados y del área global. En el caso de usar multiplicadores embebidos, la optimización de velocidad consistirá en usar tantos multiplicadores embebidos como sean necesarios. La optimización de área se centrará en hacer un uso compartido entre los multiplicadores embebidos y la lógica combinacional.

Optimización	LUTs	Multiplicadores embebidos
Área	Reducir slices usados y área	Usar multiplicadores embebidos y lógica combinacional
Velocidad	Optimizar el rendimiento	Usar multiplicadores embebidos

Tabla 4.1. Tabla comparativa de las distintas opciones para el multiplicador de Xilinx

En el último paso se puede definir el número de bits que tenga la salida y el número de estados de la segmentación en el caso de que se desee usarla.

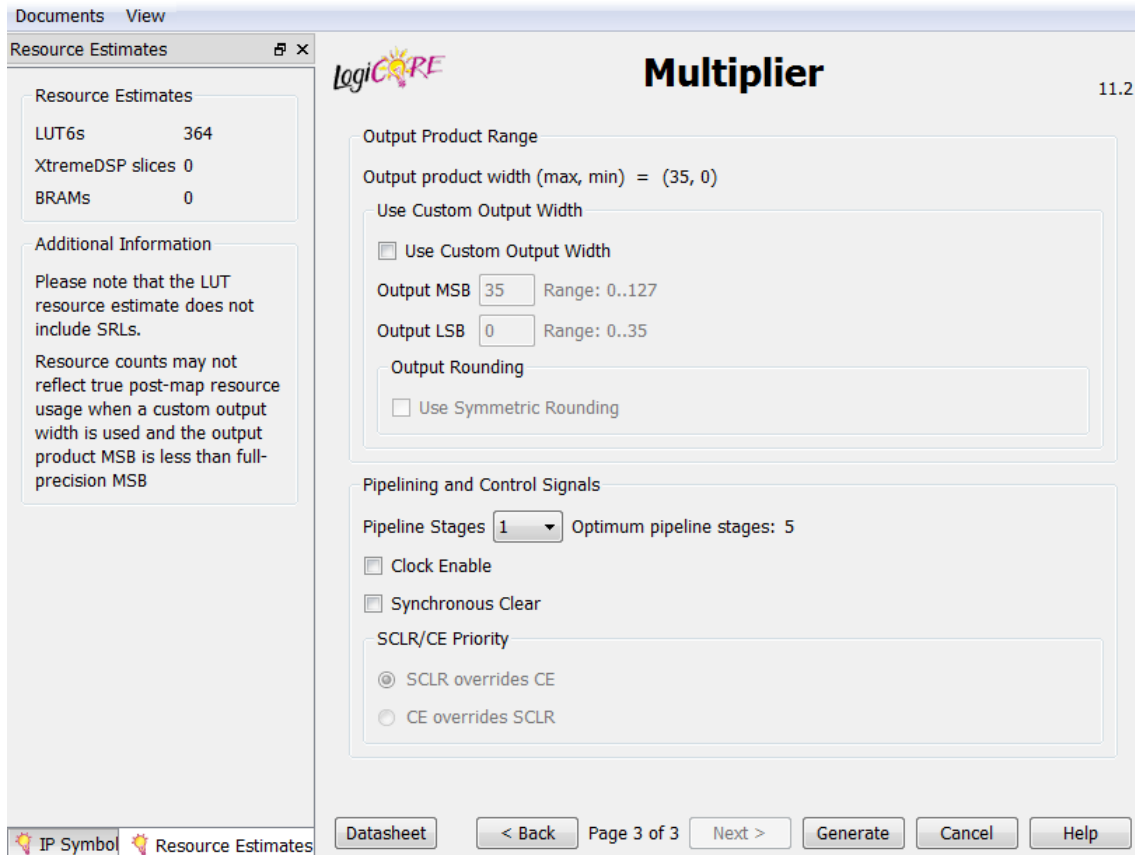


Figura 4.5. Menú de configuración del multiplicador

4.2. Multiplicador segmentado

La segmentación consiste en dividir un procesamiento en tareas más sencillas y separarlas por elementos de memoria. La siguiente figura muestra un esquema de esta técnica.

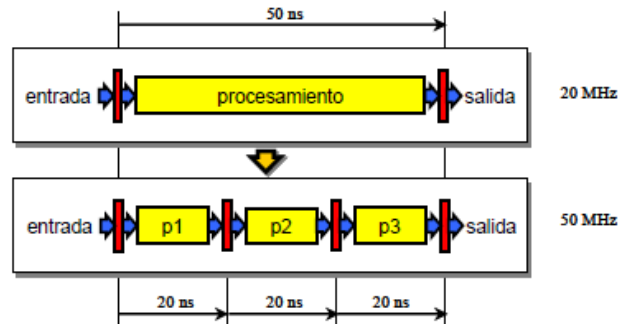


Figura 4.6. Esquema de segmentación. Extraído de [9]

Con la segmentación (o pipeline) se consigue aumentar la frecuencia de funcionamiento, aumentando la cantidad de elementos procesados por unidad de tiempo.

En la figura se puede ver que debido a la longitud del proceso sin segmentar, la frecuencia máxima del circuito está limitada al tiempo de procesamiento de este módulo. Podría ocurrir que el resto del circuito tuviera que ir más lento debido a la lentitud de este módulo, que impone la frecuencia a 20 MHz. Esto implicaría un detrimento en las prestaciones del resto del circuito. En cambio, si segmentamos el proceso, cada una de las tres divisiones podrá ir más rápidamente, con lo que la frecuencia máxima del circuito puede ser más alta (50 MHz en el ejemplo).

La segmentación permite aumentar la cantidad de elementos procesados por unidad de tiempo ya que el procesamiento se realiza concurrentemente gracias a la separación que establecen los elementos de memoria. Es decir, al mismo tiempo que p3 está terminando un proceso, p2 está procesando el siguiente elemento y p1 está procesando un elemento nuevo. De esta manera, tendremos un proceso terminado cada 20 ns (50 MHz).

Pero la segmentación no es ideal, ya que al dividir entre tres procesos (ejemplo de la figura) no se obtiene el triple de frecuencia ya que es difícil tener procesos iguales, de tal manera que cada uno tarde un tercera parte del original. También hay que tener en cuenta el llenado de la cadena de segmentación y que si el proceso solo tiene un elemento se consumen los tres ciclos de reloj. Finalmente, los retardos de pista y de los flip-flops también reducen la máxima velocidad alcanzable.

La segmentación conlleva un aumento del área debido a los elementos de memoria añadidos y la lógica de control extra.

Para segmentar un multiplicador tenemos que poner registros entre cada una de las etapas, teniendo cuidado de ir trasladando correctamente los datos de cada etapa. Teniendo en cuenta el esquema del multiplicador combinacional, se puede dividir en tantas etapas como bits de los multiplicandos. El siguiente esquema es para un multiplicador de 4 bits, por tanto, de 4 etapas. El multiplicador que se usará en las medidas será de 32 bits, por tanto, de 32 etapas. El esquema será conceptualmente similar, pero repetido 8 veces y los registros serán de 32 bits.

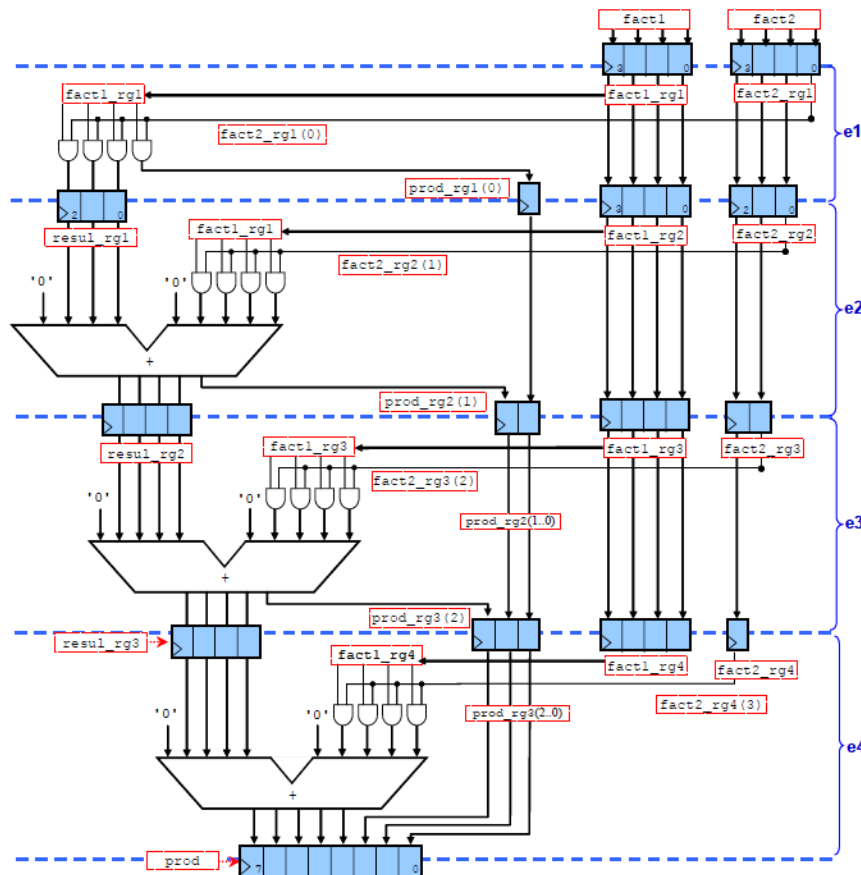


Figura 4.7. Esquema multiplicador segmentado. Extraído de [9]

Este diseño se ha hecho genérico y por ello hay elementos de señales vectoriales que no se usan, como por ejemplo algunos elementos del vector $prod_rg$ y $fact2_rg$, hecho así para simplificar la comprensión código. Debido a esto, durante la sintetización del código aparecen avisos de señales no usadas.

4.3. AES (*Advanced Encryption Standard*)

En este proyecto se han realizado medidas usando un diseño que implementa el algoritmo de encriptación y desencriptación Rijndael usado para el estándar AES. En 1997, el Instituto Nacional de Estándares y Tecnología de EEUU (NIST), empieza un proceso abierto para la selección de un nuevo algoritmo de cifrado, que sustituya al existente hasta entonces, el algoritmo DES (*Data Encryption Standard*) que empezaba a mostrar vulnerabilidades. Este estándar sería usado inicialmente para proteger los datos confidenciales del gobierno de EEUU y rápidamente se adoptaría en el sector privado y los países europeos.

Los requisitos mínimos que debían cumplir todos los algoritmos que se presentaran al concurso eran:

- El algoritmo debe ser público.
- La longitud de la clave debe ser como mínimo de 128 bits.
- Deber ser un algoritmo de cifrado en bloque simétrico.
- El diseño debe permitir un aumento de la longitud de la clave si se requiere.
- Debe ser implementable en hardware y en software.

Este último punto es fundamental para haber sido usado en este proyecto.

NIST quiso que cualquier institución, empresa u organismo pudiera presentar algoritmos al concurso o enviar informes de cualquier tipo para poner en evidencia cualquier algoritmo presentado. La intención es que el algoritmo sea lo más robusto posible hasta mitad del siglo XXI. Finalmente, el algoritmo Rijndael ganó el concurso al ser la mejor combinación de seguridad, velocidad y eficiencia.

El proceso del algoritmo consiste en aplicar 4 funciones matemáticas sobre la información. La información generada por cada función es un resultado intermedio, que se conoce como estado. Este estado se representa como una matriz rectangular. Gráficamente la descripción del proceso de cifrado se puede ver como:

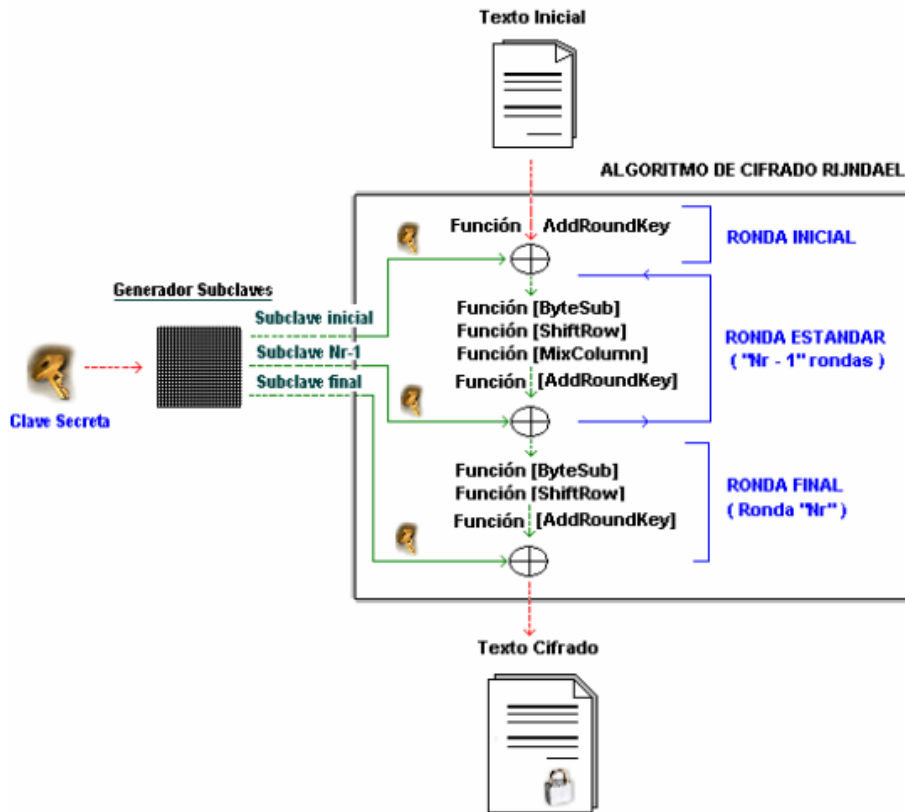


Figura 4.8. Proceso del algoritmo AES. Extraído de [20]

El número de iteraciones sobre la información depende de la versión del algoritmo que se utilice. El diseño permite encriptaciones y desencriptaciones de tres niveles de longitud de la clave: 128, 192, 256 bits. Se ha implementado con el tamaño de bus de datos máximo para minimizar el tiempo requerido de carga de información, esto implica que el diseño requiere buses de datos muy grandes y que podría no ser sintetizable en alguna FPGA. Para soportar las altas tasas de datos el diseño tiene habilitada una segmentación de cuatro niveles.

La siguiente tabla resume el máximo rendimiento para cada tamaño de clave. La tabla también proporciona el máximo ratio de encriptación y desencriptación, en bits por segundo, que se puede conseguir a una determinada frecuencia:

Key Size	Operation	Clock Cycles	Maximum Data Rate	Maximum Data Rate with 100MHz clock
128 bits	Key Expansion	45		
	Encryption / Decryption	42	12.19 bps/Hz	1.2Gbps
192 bits	Key Expansion	37		
	Encryption / Decryption	50	10.24 bps/Hz	1024Gbps
256 bits	Key Expansion	57		
	Encryption / Decryption	58	8.83 bps/Hz	882Mbps

Tabla 4.9. Rendimiento para las distintas claves. Extraído de [13]

El diseño usa una memoria RAM interna para almacenar las claves expandidas. Esta memoria está sintetizada como una memoria distribuida y tiene dos posibles configuraciones con pequeñas diferencias en los resultados de tiempo y área. La selección de cada una de estas opciones se hace comentando o des-comentando parte del código.

El siguiente diagrama representa un diagrama resumido del diseño, interfaces y módulos usados:

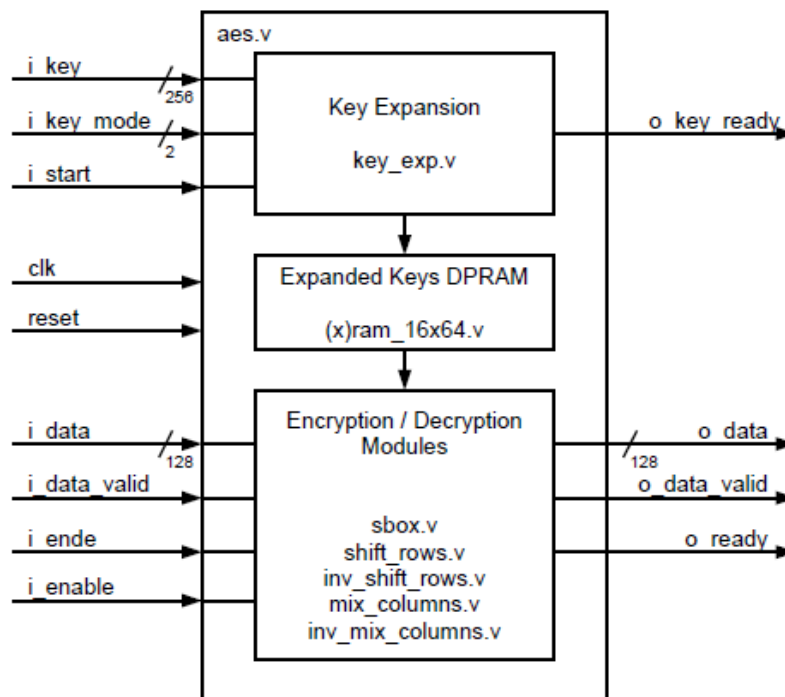


Figura 4.10. Esquema de interfaces y módulos. Extraído de [13]

4.4. FFT (Fast Fourier Transform)

El último diseño usado para este proyecto es la FFT. La FFT es un eficiente algoritmo para calcular la DFT (*Discrete Fourier Transform*), usado en muchas aplicaciones, entre las que destaca el análisis de frecuencia y el procesamiento en el dominio frecuencia. La DFT es la descomposición de una muestra de una señal en términos de componentes sinusoidales. Las propiedades de simetría y periodicidad de la DFT han sido aprovechadas debido a los bajos requerimientos computacionales. Los algoritmos resultantes son conocidos como *Fast Fourier Transforms* (FFTs). Una DFT de 256 puntos calcula a partir de una secuencia, $x[n]$, de 256 valores complejos, otra secuencia, $X(k)$, con longitud de 256, según la fórmula:

$$X(k) = \sum_{n=0}^{255} x(n)e^{-j2\pi nk/256}; k = 0 \text{ to } 255$$

Para simplificar la notación, el valor complejo $e^{-j2\pi nk/256}$ es definido como W_{256}^n , donde $W_{256}^n = \cos(2\pi n/256) - j\sin(2\pi n/256)$. La FFT aprovecha las propiedades de simetría y periodicidad de W_{256}^n , lo cual reduce notablemente el número de cálculos que la DFT requiere. El fundamento de la FFT es que la DFT se puede reducir en DFTs más pequeñas.

$$X(k) = X(16r + s) = \sum_{m=0}^{15} W_{16}^{mr} W_{256}^{ms} \sum_{l=0}^{15} x(16l + m) W_{16}^{sl}, r = 0 \text{ to } 15, s = 0 \text{ to } 15$$

Por tanto, una DFT de 256 puntos se puede dividir en dos DFTs de 16 puntos. Este algoritmo se ilustra en el gráfico siguiente. La entrada $x[n]$ es representado por un array de dos dimensiones del dato $x(16l + m)$. Las columnas de este array son calculadas por DFTs de 16 puntos. Este resultado es multiplicado por W_{16}^{sl} , y el valor resultante $X(16r + s)$, es pasado por DFTs de 16 puntos colocados en filas, generando el resultado intermedio. La DFT de 16 puntos es la base del algoritmo FFT.

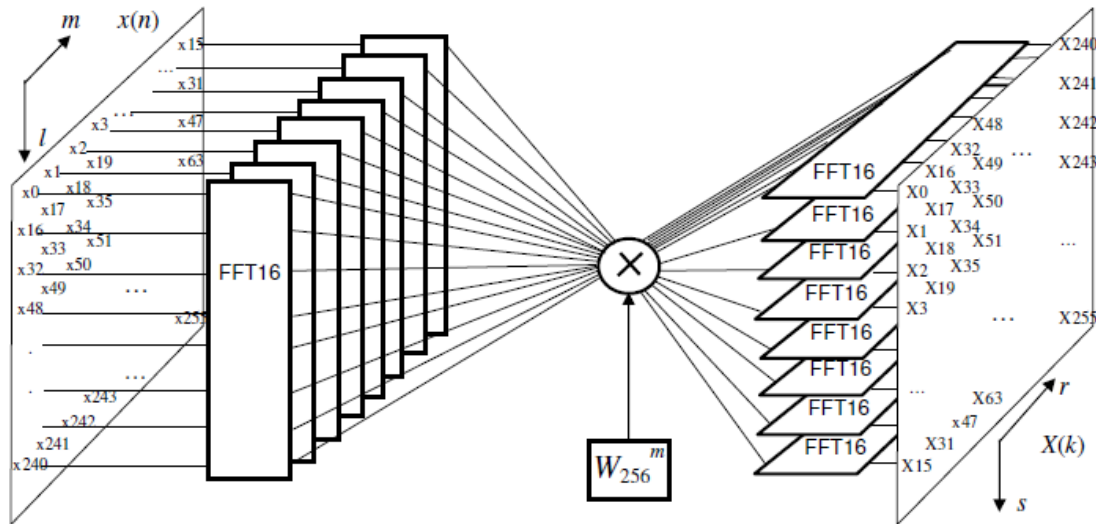


Figura 4.11. Esquema de FFT. Extraído de [14]

Cada operación de la FFT es calculado mediante la FFT16 y esta calcula la DFT de 16 puntos con una alta segmentación. El retardo de la FFT16 desde que entre el primer dato hasta su salida es igual a 30 ciclos de reloj.

El número de multiplicadores necesarios para hacer la FFT de 256 puntos es igual a 4. Si se configura en una FPGA de Xilinx estos multiplicadores son implementados en 4 DSP48

5. Opciones de ISE y PlanAhead relacionadas con consumo

5.1. Opciones de Xilinx ISE 13.1

El programa *Xilinx ISE 13.1* permite seleccionar y editar entre varios tipos de optimización y según que optimización se elija se modificaran unos parámetros u otros. Muchas de las configuraciones de los parámetros son editables manualmente para un rendimiento más cercano al deseado por el diseñador.

De entre todas las opciones que permite *editar Xilinx ISE 13.1* las más importantes son las que se editarán según los objetivos de diseño. Aspecto importante es la elección de las librerías para las puertas lógicas (puertas complejas o puertas simples). Cada una de las opciones que se explican a continuación tendrán distintos resultados de consumo.

- **Balanceado:** Es la estrategia por defecto y proporciona una optimización balanceada entre rendimiento y tiempo de ejecución. Esta estrategia mantiene todas las propiedades desbloqueadas por lo que pueden ser modificadas como se desee. En los siguientes puntos se usa este parámetro como comparación (*Current Value*).
- **Reducción del área:** Esta estrategia permitirá minimizar la cantidad de LUTs usadas a la vez que se intenta mantener el rendimiento temporal. Las diferencias con la opción balanceada son:

Property Name	Strategy Value	Current Value
[-] Synthesize - XST		
Optimization Goal	Area	Speed
Optimization Effort	Normal	Normal
Register Balancing	No	No
Max Fanout	100000	100000
Pack I/O Registers into IOBs	Yes	Auto
Automatic BRAM Packing	true	false
LUT Combining	Auto	Auto
[-] Map		
Placer Effort Level	High	High
LUT Combining	Auto	Off
Pack I/O Registers/Latches into IOBs	For Inputs and Outputs	Off
[-] Place & Route		
Place And Route Mode	Route Only	Route Only
Place & Route Effort Level (Overall)	High	High

Tabla 5.1. Estrategia de reducción del área

- **Mínimo tiempo de ejecución:** Esta estrategia minimiza el tiempo de ejecución. Esta estrategia no optimizará el rendimiento temporal. Sus diferencias con respecto a la opción balanceada son:

Property Name	Strategy Value	Current Value
Synthesize - XST		
Read Cores	false	true
Map		
Placer Effort Level	Standard	High
Enable Multi-Threading	2	Off
Place & Route		
Place & Route Effort Level (Overall)	Standard	High
Enable Multi-Threading	2	Off

Tabla 5.2. Estrategia de mínimo tiempo de ejecución

- **Optimización de potencia:** Optimiza el consumo de potencia de los circuitos a estudio. Las diferencias con la opción balanceada son:

Property Name	Strategy Value	Current Value
Synthesize - XST		
Optimization Goal	Area	Speed
Optimization Effort	Normal	Normal
Register Balancing	No	No
Max Fanout	100000	100000
Pack I/O Registers into IOBs	No	Auto
Automatic BRAM Packing	true	false
Power Reduction	true	false
LUT Combining	Auto	Auto
FSM Encoding Algorithm	Compact	Auto
Map		
Placer Effort Level	High	High
LUT Combining	Auto	Off
Pack I/O Registers/Latches into IOBs	Off	Off
Power Reduction	true	Off
Place & Route		
Power Reduction	true	false
Place & Route Effort Level (Overall)	High	High

Tabla 5.3. Estrategia de optimización de potencia

- **Rendimiento temporal:** Esta estrategia está orientada a conseguir un tiempo de ejecución reducido mientras se mantienen los bloques de entrada/salida. Es recomendable usar esta estrategia si el diseño tiene unos requerimientos de entrada/salida muy estrictos. Las diferencias con la opción balanceada son:

Property Name	Strategy Value	Current Value
[-] Synthesize - XST		
Optimization Goal	Speed	Speed
Optimization Effort	High	Normal
Register Balancing	Yes	No
Pack I/O Registers into IOBs	Yes	Auto
[-] Map		
Placer Effort Level	High	High
Placer Extra Effort	Normal	None
Pack I/O Registers/Latches into IOBs	For Inputs and Outputs	Off
[-] Place & Route		
Place And Route Mode	Route Only	Route Only
Place & Route Effort Level (Overall)	High	High
Extra Effort (Highest PAR level only)	Normal	None

Tabla 5.4. Estrategia de mejora de rendimiento temporal

Todas las opciones editables se muestran a continuación:

Property Name	Current Value
Synthesize - XST	
Optimization Goal	Speed
Optimization Effort	Normal
Power Reduction	<input type="checkbox"/>
Use Synthesis Constraints File	<input checked="" type="checkbox"/>
Synthesis Constraints File	
Keep Hierarchy	No
Netlist Hierarchy	As Optimized
Global Optimization Goal	AllClockNets
Generate RTL Schematic	Yes
Read Cores	<input checked="" type="checkbox"/>
Cores Search Directories	
Write Timing Constraints	<input type="checkbox"/>
Cross Clock Analysis	<input type="checkbox"/>
Hierarchy Separator	/
Bus Delimiter	<>
LUT-FF Pairs Utilization Ratio	100
BRAM Utilization Ratio	100
DSP Utilization Ratio	100
Case	Maintain
Work Directory	C:/Users/root1/Desktop/proyect...
HDL INI File	
Library for Verilog Sources	
Library Search Order	
Verilog Include Directories	
Generics, Parameters	
Verilog Macros	
Other XST Command Line Options	
FSM Encoding Algorithm	Auto
Safe Implementation	No
Case Implementation Style	None
FSM Style	LUT
RAM Extraction	<input checked="" type="checkbox"/>
RAM Style	Auto
ROM Extraction	<input checked="" type="checkbox"/>
ROM Style	Auto
Automatic BRAM Packing	<input type="checkbox"/>
Shift Register Extraction	<input checked="" type="checkbox"/>
Shift Register Minimum Size	2
Resource Sharing	<input checked="" type="checkbox"/>
Use DSP Block	Auto
Asynchronous To Synchronous	<input type="checkbox"/>
Add I/O Buffers	<input checked="" type="checkbox"/>
Max Fanout	100000
Number of Clock Buffers	16
Register Duplication	<input checked="" type="checkbox"/>
Equivalent Register Removal	<input checked="" type="checkbox"/>
Register Balancing	No
Move First Flip-Flop Stage	<input checked="" type="checkbox"/>
Move Last Flip-Flop Stage	<input checked="" type="checkbox"/>
Pack I/O Registers into IOBs	Auto
LUT Combining	Auto
Reduce Control Sets	Auto
Use Clock Enable	Auto
Use Synchronous Set	Auto
Use Synchronous Reset	Auto
Optimize Instantiated Primitives	<input type="checkbox"/>
Translate	
Use LOC Constraints	<input checked="" type="checkbox"/>
Netlist Translation Type	Timestamp
Macro Search Path	
Create I/O Pads from Ports	<input type="checkbox"/>
Allow Unexpanded Blocks	<input type="checkbox"/>

Property Name	Current Value
... User Rules File for Netlist Launcher	
... Allow Unmatched LOC Constraints	<input type="checkbox"/>
... Allow Unmatched Timing Group Constraints	<input type="checkbox"/>
... Other Ngdbuild Command Line Options	
Map	
... Placer Effort Level	High
... Placer Extra Effort	None
... Starting Placer Cost Table (1-100)	1
... Extra Cost Tables	0
... Combinatorial Logic Optimization	<input type="checkbox"/>
... Register Duplication	Off
... Register Ordering	4
... Global Optimization	Off
... Retiming	<input type="checkbox"/>
... Equivalent Register Removal	<input checked="" type="checkbox"/>
... Ignore User Timing Constraints	<input type="checkbox"/>
... Timing Mode	Performance Evaluation
... Trim Unconnected Signals	<input checked="" type="checkbox"/>
... Allow Logic Optimization Across Hierarchy	<input type="checkbox"/>
... Generate Detailed MAP Report	<input type="checkbox"/>
... Use RLOC Constraints	Yes
... Pack I/O Registers/Latches into IOBs	Off
... Maximum Compression	<input type="checkbox"/>
... LUT Combining	Off
... Map Slice Logic into Unused Block RAMs	<input type="checkbox"/>
... Power Reduction	Off
... Power Activity File	
... Enable Multi-Threading	Off
... Other Map Command Line Options	
Place & Route	
... Place And Route Mode	Route Only
... Place & Route Effort Level (Overall)	High
... Extra Effort (Highest PAR level only)	None
... Ignore User Timing Constraints	<input type="checkbox"/>
... Timing Mode	Performance Evaluation
... Generate Asynchronous Delay Report	<input type="checkbox"/>
... Generate Clock Region Report	<input type="checkbox"/>
... Generate Post-Place & Route Simulation Model	<input type="checkbox"/>
... Generate Post-Place & Route Power Report	<input type="checkbox"/>
... Power Reduction	<input type="checkbox"/>
... Power Activity File	
... Enable Multi-Threading	Off
... Other Place & Route Command Line Options	
Generate Programming File	
... Run Design Rules Checker (DRC)	<input checked="" type="checkbox"/>
... Create Bit File	<input checked="" type="checkbox"/>
... Create Binary Configuration File	<input type="checkbox"/>
... Create ASCII Configuration File	<input type="checkbox"/>
... Create IEEE 1532 Configuration File	<input type="checkbox"/>
... Enable BitStream Compression	<input type="checkbox"/>
... Enable Debugging of Serial Mode BitStream	<input type="checkbox"/>
... Enable Cyclic Redundancy Checking (CRC)	<input checked="" type="checkbox"/>
... Retry Configuration if CRC Error Occurs	<input type="checkbox"/>
... Other Bitgen Command Line Options	
... Configuration Rate	2
... Configuration Pin Program	Pull Up
... Configuration Pin Done	Pull Up
... JTAG Pin TCK	Pull Up
... JTAG Pin TDI	Pull Up
... JTAG Pin TDO	Pull Up
... JTAG Pin TMS	Pull Up
... Unused IOB Pins	Pull Down
... UserID Code (8 Digit Hexadecimal)	0xFFFFFFFF
... Enable External Master Clock	<input type="checkbox"/>
... Setup External Master Clock Division	1

Property Name	Current Value
Set SPI Configuration Bus Width	1
Watchdog Timer Value	0xFFFF
Place MultiBoot Settings into Bitstream	<input type="checkbox"/>
MultiBoot: Starting Address for Next Configuration	0x00000000
MultiBoot: Use New Mode for Next Configuration	<input checked="" type="checkbox"/>
MultiBoot: Next Configuration Mode	001
MultiBoot: Starting Address for Golden Configuration	0x00000000
MultiBoot: User-Defined Register for Failsafe Scheme	0x0000
FPGA Start-Up Clock	CCLK
Enable Internal Done Pipe	<input type="checkbox"/>
Done (Output Events)	Default (4)
Enable Outputs (Output Events)	Default (5)
Release Write Enable (Output Events)	Default (6)
Wait for DCM and PLL Lock (Output Events)	Default (NoWait)
Drive Done Pin High	<input type="checkbox"/>
Security	Enable Readback and Reconfigu...
Create ReadBack Data Files	<input type="checkbox"/>
Allow SelectMAP Pins to Persist	<input type="checkbox"/>
Create Logic Allocation File	<input type="checkbox"/>
Create Mask File	<input type="checkbox"/>
Encrypt Bitstream	<input type="checkbox"/>
Encrypt Key Select	BGRAM
AES Key (Hex String)	
Input Encryption Key File	
AES Initial Vector	
Enable Suspend/Wake Global Set/Reset	<input type="checkbox"/>
Drive Awake Pin During Suspend/Wake Sequence	<input type="checkbox"/>
Wakeup Clock	Startup Clock
GWE Cycle During Suspend/Wakeup Sequence	5
GTS Cycle During Suspend/Wakeup Sequence	4
Enable Multi-Pin Wake-Up Suspend Mode	<input type="checkbox"/>
Mask Pins for Multi-Pin Wake-Up Suspend Mode	0x00
Generate Post-Place & Route Static Timing	
Report Type	Verbose Report
Number of Paths in Error/Verbose Report	3
Perform Advanced Analysis	<input type="checkbox"/>
Change Device Speed To	-3
Report Unconstrained Paths	
Report Paths by Endpoint	3
Report Fastest Path(s) in Each Constraint	<input checked="" type="checkbox"/>
Generate Datasheet Section	<input checked="" type="checkbox"/>
Generate Timegroups Section	<input type="checkbox"/>
Stamp Timing Model Filename	
Generate Constraints Interaction Report	<input type="checkbox"/>

Tabla 5.5. Todas las opciones editables

5.2. Opciones de Xilinx PlanAhead 13.1

Al igual que Xilinx ISE, Xilinx PlanAhead 13.1 también dispone de múltiples opciones según la estrategia que se desee. Todas las opciones de optimización se encuentran divididas en dos grupos, estrategias de síntesis y estrategias de implementación. En este proyecto solo se han usado las estrategias de síntesis.

Estrategias de síntesis:

- *PlanAhead* defaults: Modo por defecto de PlanAhead. Similar a un modo balanceado de Xilinx ISE.

-opt_mode	speed
-opt_level	1
-register_balancing	no
-fsm_encoding	auto
-lc	off
-auto_bram_packing	no
-use_dsp48	auto
-resource_sharing	yes
-iob	auto
-netlist_hierarchy*	as_optimized
-power	no
-ram_style	auto
-bufg	
-equivalent_register_removal	yes
-mux_extract	yes

Tabla 5.6. Estrategia por defecto de PlanAhead

- Timing Performance with IOB Packing: Optimiza el rendimiento temporal usando bloques de entrada/salida.

-opt_mode	speed
-opt_level*	2
-register_balancing*	yes
-fsm_encoding	auto
-lc	off
-auto_bram_packing	no
-use_dsp48	auto
-resource_sharing	yes
-iob*	true
-netlist_hierarchy	rebuilt
-power	no
-ram_style	auto
-bufg	
-equivalent_register_removal	yes
-mux_extract	yes

Tabla 5.7. Estrategia de mínimo tiempo de ejecución usando IOB

- Timinig Performance without IOB Packing: Optimiza el rendimiento temporal sin usar bloques de entrada/salida.

-opt_mode	speed
-opt_level*	2
-register_balancing*	yes
-fsm_encoding	auto
-lc	off
-auto_bram_packing	no
-use_dsp48	auto
-resource_sharing	yes
-iob*	false
-netlist_hierarchy	rebuilt
-power	no
-ram_style	auto
-bufg	
-equivalent_register_removal	yes
-mux_extract	yes

Tabla 5.8. Estrategia de mínimo tiempo de ejecución sin usar IOB

- Area Reduction: Reduce el área usado en la síntesis.

-opt_mode*	area
-opt_level	1
-register_balancing	no
-fsm_encoding	auto
-lc*	auto
-auto_bram_packing*	yes
-use_dsp48	auto
-resource_sharing	yes
-iob*	true
-netlist_hierarchy	rebuilt
-power	no
-ram_style	auto
-bufg	
-equivalent_register_removal	yes
-mux_extract	yes

Tabla 5.9. Estrategia de reducción de área

- Power Optimization: Optimiza el consumo de potencia.

-opt_mode*	area
-opt_level	1
-register_balancing	no
-fsm_encoding*	compact
-lc*	auto
-auto_bram_packing*	yes
-use_dsp48	auto
-resource_sharing	yes
-iob*	false
-netlist_hierarchy	rebuilt
-power*	yes
-ram_style	auto
-bufg	
-equivalent_register_removal	yes
-mux_extract	yes

Tabla 5.10. Estrategia de optimización del consumo de potencia

El uso principal que se le ha dado al *PlanAhead* en este proyecto ha sido la reubicación de los componentes en la FPGA (*floorplanning*) para analizar la influencia del diseñador así como estudiar la importancia de la topología.

También se han realizado las medidas de potencia con *Xilinx PlanAhead*, pero se ha usado el *Xilinx ISE* para una evaluación más detallada de las técnicas de medición (simulación y medida sobre el circuito) y el estudio de los efectos de los *settings* de la herramienta.

6. Medidas

En este capítulo se muestran los resultados obtenidos y describe el proceso que se ha realizado para tomar las medidas, tanto simuladas como sobre el circuito (FPGA). Primero se describirá el proceso de simulación y a continuación las medidas realizadas sobre el circuito.

Para lograr una correcta estimación de potencia se genera el fichero SAIF (*Switching Activity Interchange Format*), que muestra las transiciones en todos los nodos, por tanto, es una medida de con qué frecuencia una red, un pin o un puerto tiene una transición de 0 -> 1 o de 1 ->0, es decir, un cambio de estado. El fichero SAIF se genera después de realizar la simulación *post-route* con el simulador de *Xilinx ISE (ISim)*.

Antes de iniciar la simulación se deberá seleccionar adecuadamente la duración de esta. Una simulación muy corta no permitiría un correcto cálculo de la actividad de los cambios de estado para todos los elementos del circuito. En este proyecto las simulaciones han sido de 100.000.000 ps con el fin de asegurar que haya habido suficientes cambios de estado en el circuito.

La estimación de consumo se realizará mediante la herramienta *Xilinx XPower Analyzer 13.1* y se le adjuntará el fichero SAIF generado después de la simulación. Si no se generara este fichero, a la hora de calcular el consumo de potencia, esta tomará un valor por defecto para la actividad interna de los nodos del circuito, alejándose de la realidad y desvirtuando la medida. La herramienta *XPower Analyzer* dará la estimación de la corriente que entra al núcleo de la FPGA y sabiendo que la FPGA está alimentada por la tarjeta de desarrollo a 1,2 V se obtiene la potencia mediante:

$$P = V \cdot I$$

Paralelamente a la simulación y estimación de consumo, se han realizado las medidas del diseño implementado, de dos formas diferentes. La primera sobre la FPGA de la tarjeta de desarrollo a temperatura ambiente de entre 22 °C y 25 °C y la segunda mediante el software *Avnet Board Programming Utility 4.05*, incluido por el fabricante de la tarjeta de desarrollo.

La primera medida se realiza sobre una resistencia con un valor muy bajo (0.05 Ω) que está a la entrada del núcleo de la FPGA:

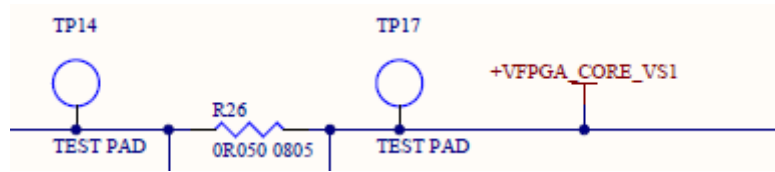


Figura 6.1. Pads para la medida del consumo de la FPGA. Extraído de [21]

Se usará un multímetro de precisión (*Agilent 34410A Digital Multimeter*) para obtener la tensión que cae sobre la resistencia para a continuación calcular la corriente que atraviesa dicha resistencia.

$$I = \frac{V}{R}$$

Con esta corriente y sabiendo que la tarjeta de desarrollo alimenta a la FPGA a 1,2 V, se calcula la potencia que consume la FPGA.

$$P = V \cdot I$$

La medida obtenida con el multímetro de precisión se representará en las gráficas de los resultados como "Medida".

La medida obtenida con *Avnet 4.05* es equivalente a la realizada con el multímetro de precisión. La tensión que cae sobre la resistencia R26 se pasa a un amplificador operacional y después de este hay un convertor analógico-digital y esta medida se devuelve al *Avnet*. La medida obtenida de esta manera se representará en las gráficas como "Avnet".

El esquema del sistema de medida para el multiplicador es el siguiente:

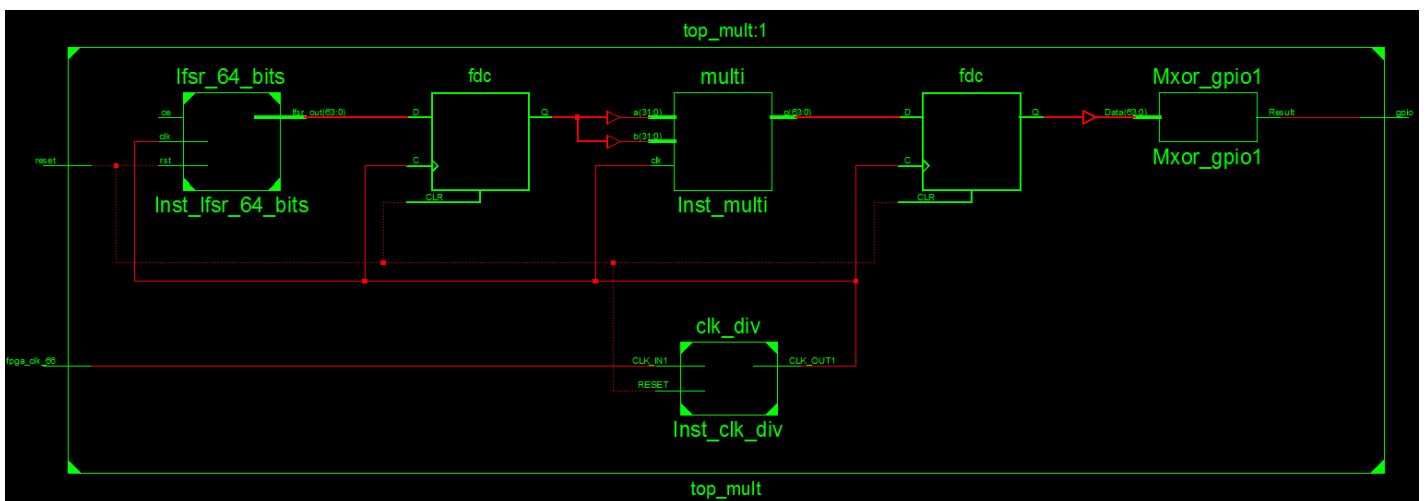


Figura 6.2. Esquema del multiplicador Xilinx bajo estudio

A la entrada tiene un LFSR (*Linear Feedback Shift Register*) que genera señales pseudo-aleatorias de 64 bits. El LFSR se conecta a su vez a un flip-flop y este a su vez está conectado a las dos entradas del multiplicador. La salida del multiplicador se conecta a un segundo flip-flop y la salida de este está conectado a una puerta XOR tras el cual se obtiene el valor resultante. Todo ello está controlado por un reloj a 33 MHz.

En el proyecto se han realizado medidas con diferentes multiplicadores, lo que quiere decir que lo que cambia es la instancia *multi* manteniéndose el resto constante. El esquema anterior completo se implementa y se carga en la FPGA para realizar las medidas.

En el caso de los multiplicadores, para el análisis de los resultados solo se ha tenido en cuenta el consumo de potencia relativo al multiplicador, no al resto del circuito, como por ejemplo, LFSR, flip-flops o el reloj. Para hacer esto, se parte del consumo del sistema total (calculado como se ha explicado anteriormente) y se le resta el consumo de todo lo que no sea el multiplicador. Esto se consigue eliminando el multiplicador del sistema y reconectando las señales.

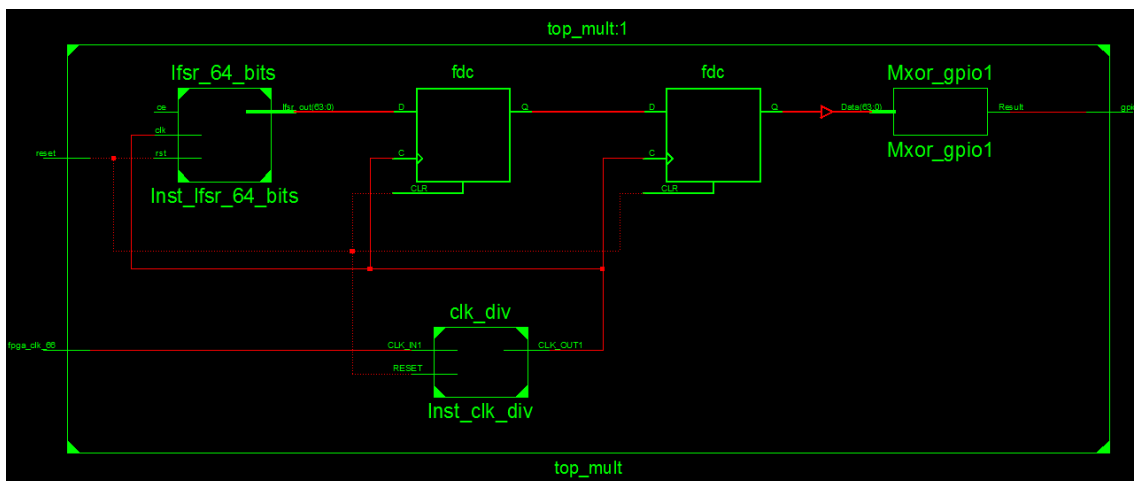


Figura 6.3. Esquema sin multiplicador

El circuito sin multiplicador presenta los siguientes consumos:

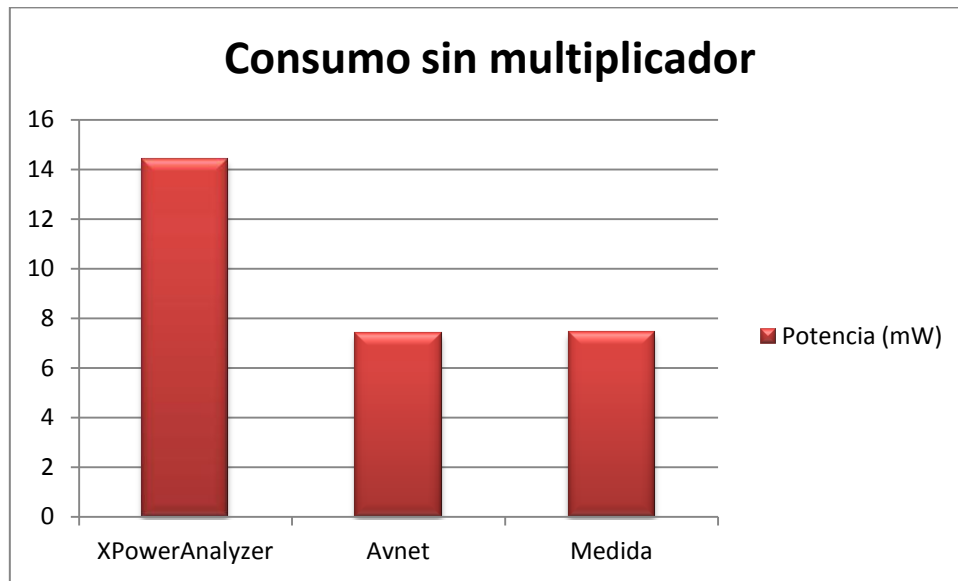


Figura 6.4. Consumo del sistema sin multiplicador

De esta manera se consigue que el análisis de los resultados obtenidos sea únicamente sobre el multiplicador y las estrategias de diseño aplicadas a él.

En el caso de los circuitos AES y FFT, debido a su mayor complejidad y consumo se ha tenido en cuenta el consumo del sistema completo para el análisis de los resultados.

Con el programa ISE se ha usado los multiplicadores, AES y FFT, con el *PlanAhead* se ha usado solo los multiplicadores.

	<i>Multiplicadores</i>	<i>FFT</i>	<i>AES</i>
Balanceado	✓	✓	✓
Optimización de potencia	✓	✓	✓
Rendimiento temporal	✓	✓	✓
Área	✓	✓	✓
Reducción del tiempo de ejecución	✓		

Tabla 6.1. Estrategias de diseño aplicadas con ISE

<i>Multiplicadores</i>	
Balanceado	✓
Optimización de potencia	✓
Rendimiento temporal	✓

Tabla 6.2. Estrategias de diseño aplicadas con PlanAhead

Adicionalmente se ha estudiado la influencia del diseñador mediante *floorplanning*. Para ello se ha usado la herramienta *PlanAhead*, después de aplicar las estrategias de diseño de optimización de potencia y balanceado.

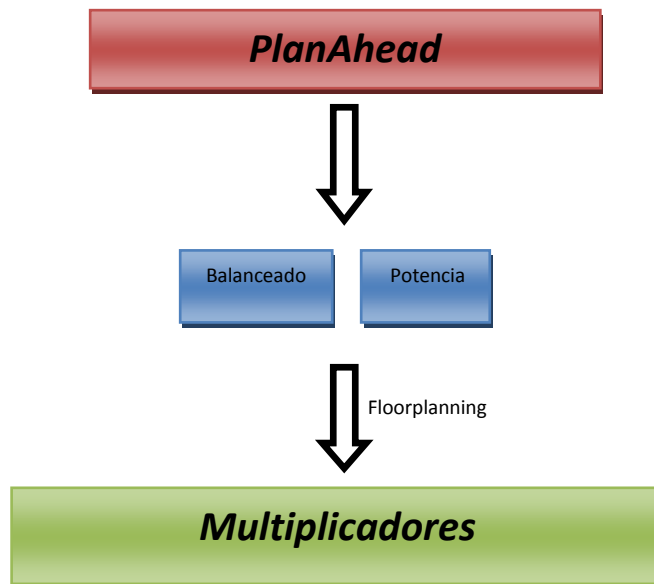


Figura 6.5. Esquema de trabajo en PlanAhead

Se han realizado 181 medidas de potencia divididas de la siguiente manera:

Circuito	Número de medidas
Multiplicadores Xilinx	139
Multiplicador segmentado	26
AES	8
FFT	8
Total	181

Tabla 6.3. Medidas realizadas

A estas medidas hay que sumar las 20 estimaciones realizadas variando la temperatura de ambiente y las 4 estimaciones variando el condensador de carga, lo que hace un **total de 205 medidas**.

A continuación se desglosa los resultados obtenidos. Primero se mostrarán los resultados de los distintos multiplicadores y las estrategias de diseño y después los circuitos AES y FFT con sus respectivas estrategias de diseño.

6.1. Multiplicadores Xilinx

Las medidas para los multiplicadores se han dividido de la siguiente manera

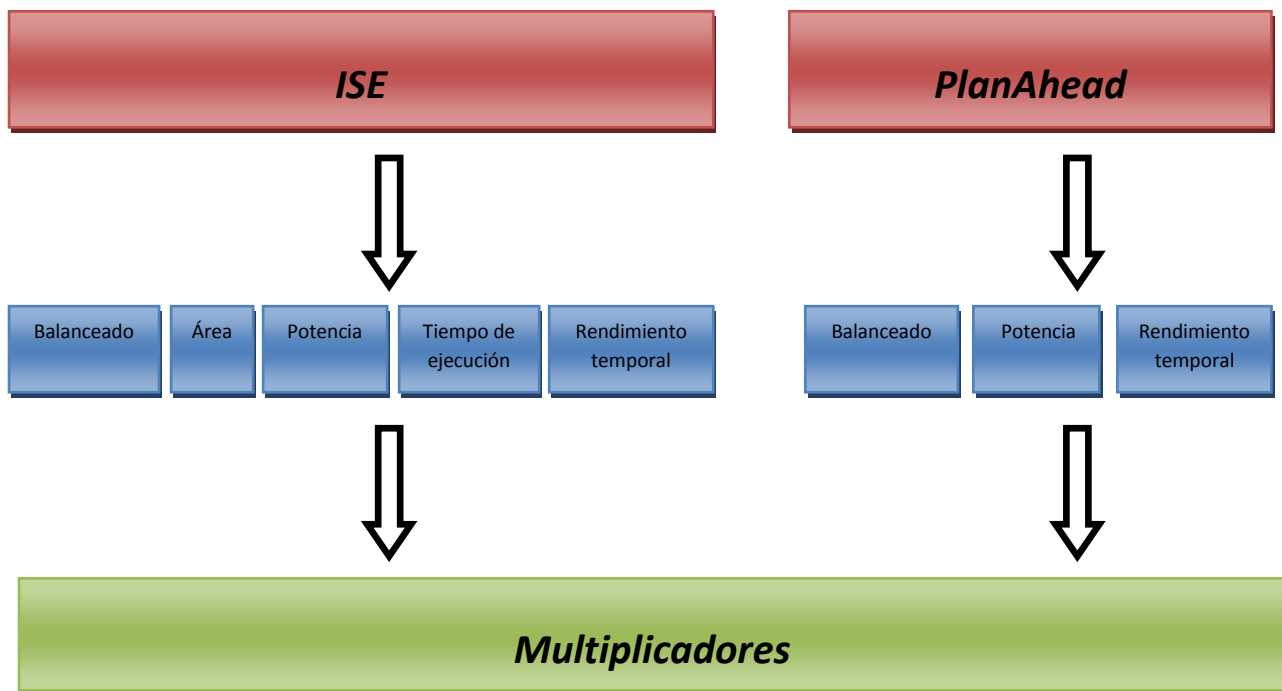


Figura 6.6. Esquema de trabajo para los multiplicadores Xilinx

Los multiplicadores están divididos en cuatro grupos:

- Multiplicador embebidos y área: multiplicador construido a partir de DSPs embebidos y optimizado para compartir recursos entre los DSPs embebidos y la lógica.
- Multiplicador embebidos y velocidad: multiplicador construido a partir de DSPs embebidos con el fin de optimizar el rendimiento y usar tantos circuitos embebidos como sea necesario.

- Multiplicadores usando LUTs y área: el multiplicador será optimizado para reducir la lógica y el área global.
- Multiplicador usando LUTs y velocidad: el multiplicador será optimizado para obtener el mejor rendimiento posible.

6.1.1. ISE

6.1.1.1. Balanceado

Las primeras gráficas muestran los resultados obtenidos para los diferentes multiplicadores.

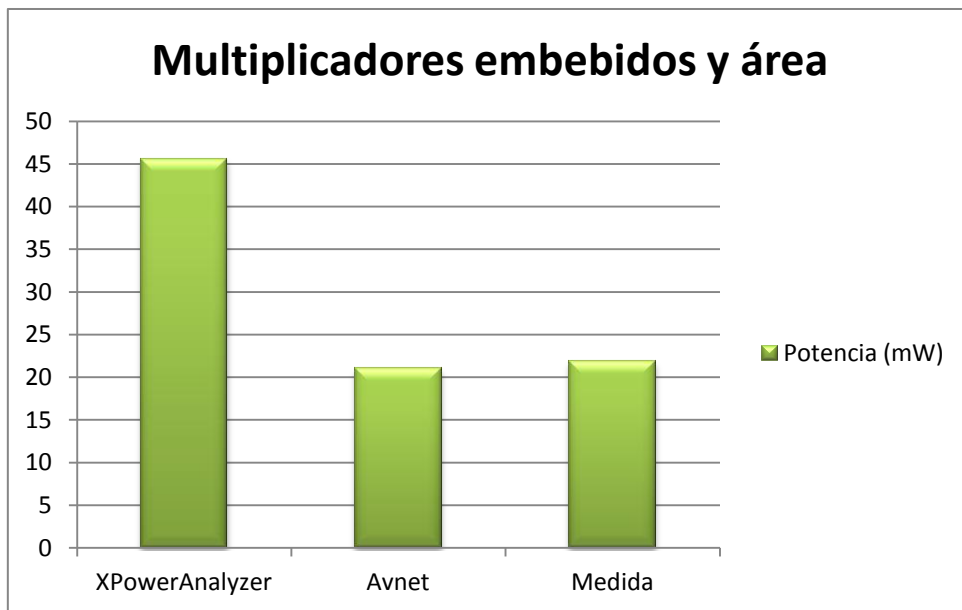


Figura 6.7. Medidas usando ISE y estrategia de diseño balanceada

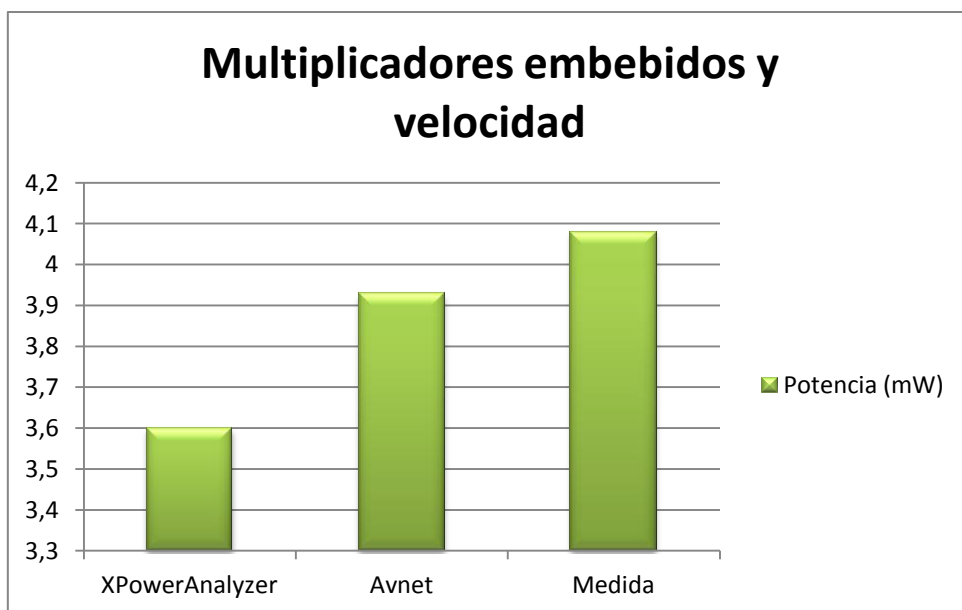


Figura 6.8. Medidas usando ISE y estrategia de diseño balanceada

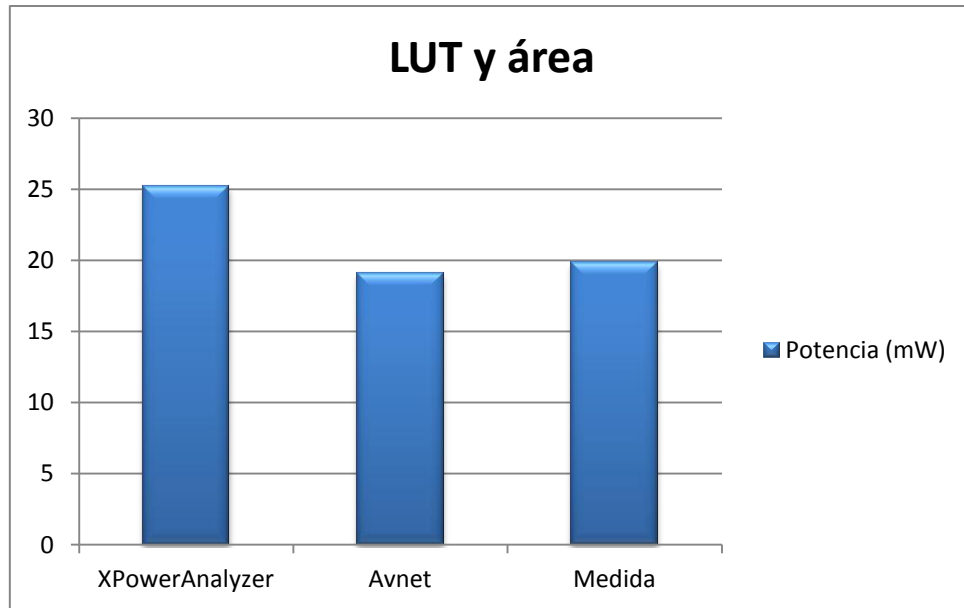


Figura 6.9. Medidas usando ISE y estrategia de diseño balanceada

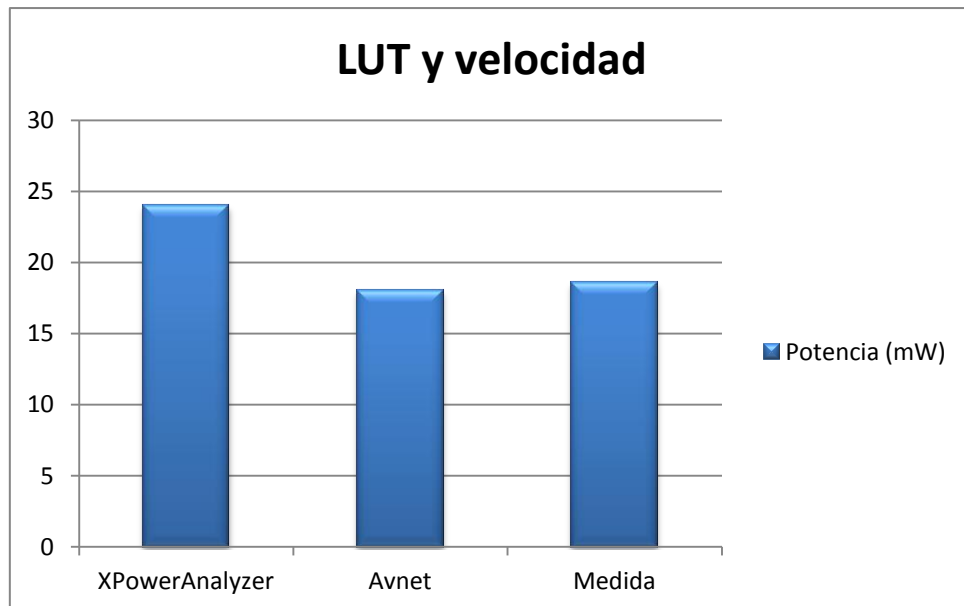


Figura 6.10. Medidas usando ISE y estrategia de diseño balanceada

El segundo grupo de gráficas muestra los resultados en función de las distintas técnicas de medición.

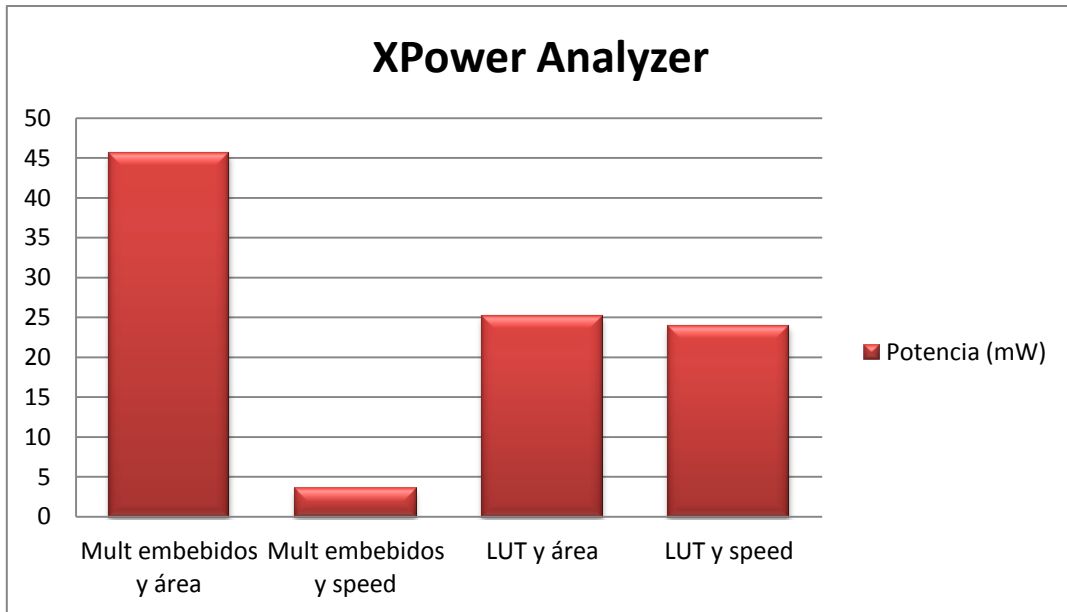


Figura 6.11. Comparativa de las medidas obtenidas con XPower Analyzer

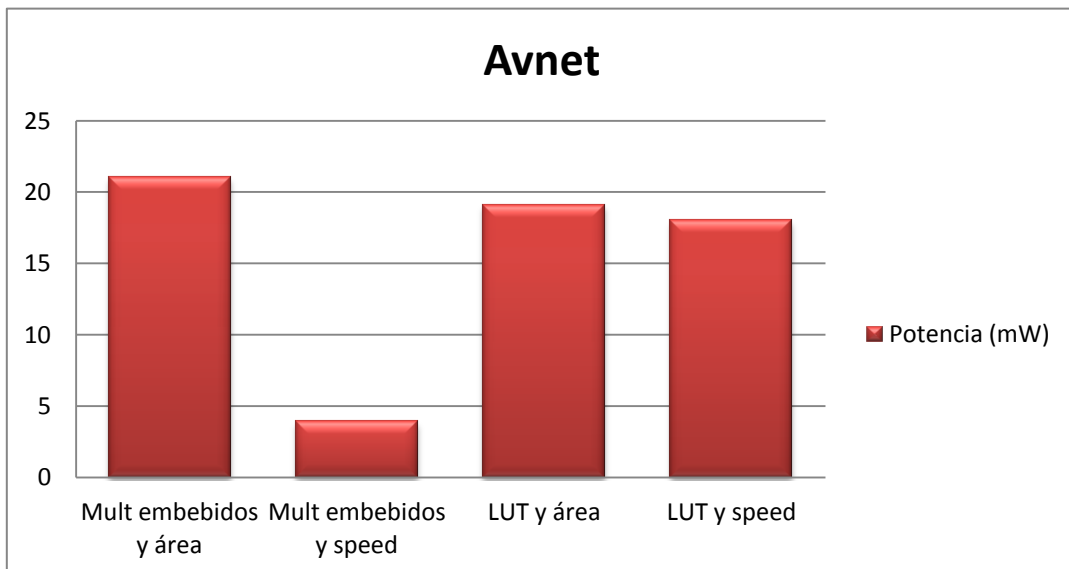


Figura 6.12. Comparativa de las medidas obtenidas con Avnet

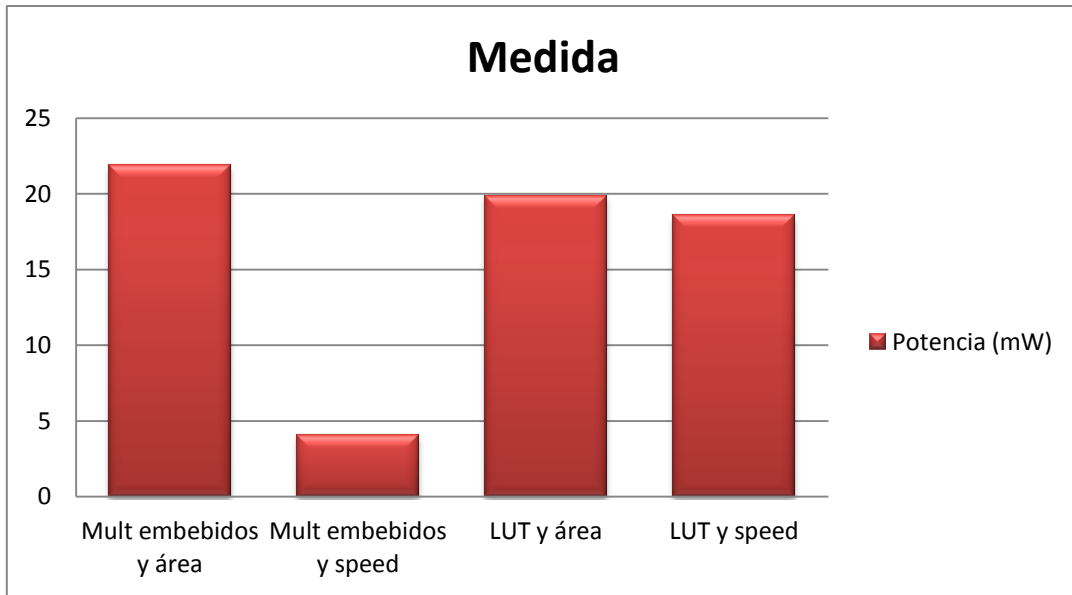


Figura 6.13. Comparativa de las medidas obtenidas midiendo sobre el circuito

6.1.1.2. Área

Las primeras gráficas muestran los resultados obtenidos para los diferentes multiplicadores.

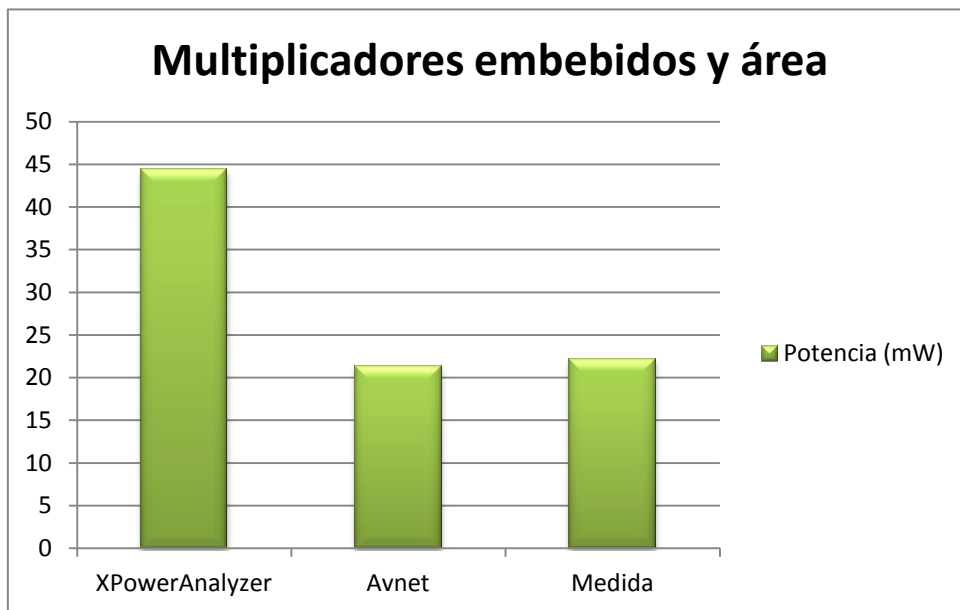


Figura 6.14. Medidas usando ISE y estrategia de diseño de optimización de área

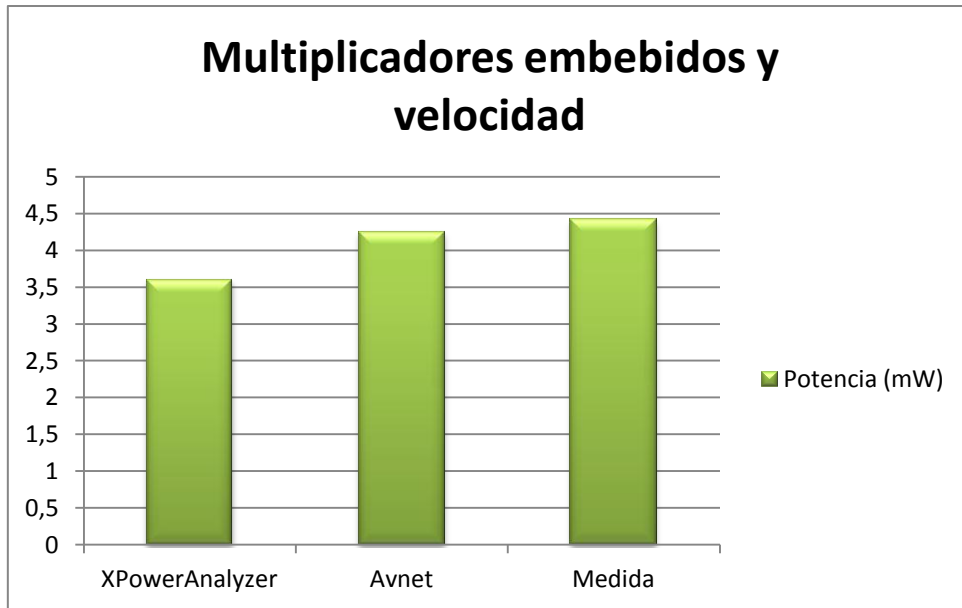


Figura 6.15. Medidas usando ISE y estrategia de diseño de optimización de área

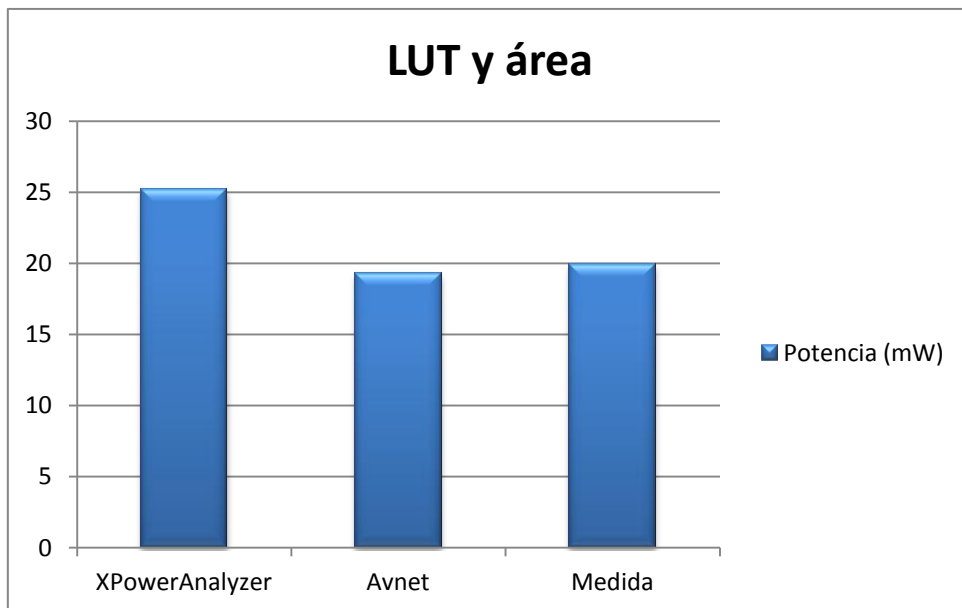


Figura 6.16. Medidas usando ISE y estrategia de diseño de optimización de área

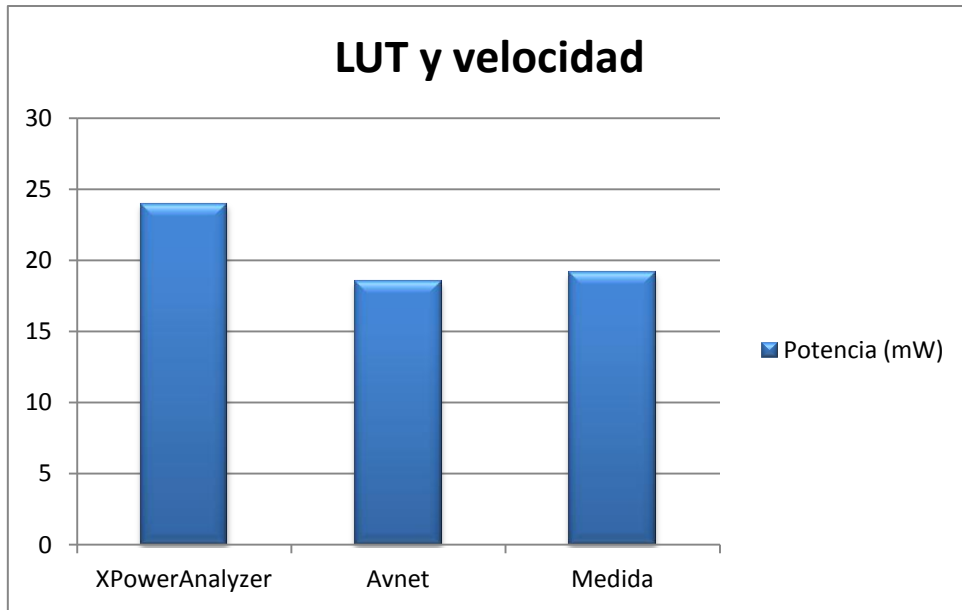


Figura 6.17. Medidas usando ISE y estrategia de diseño de optimización de área

El segundo grupo de gráficas muestra los resultados en función de las distintas técnicas de medición.

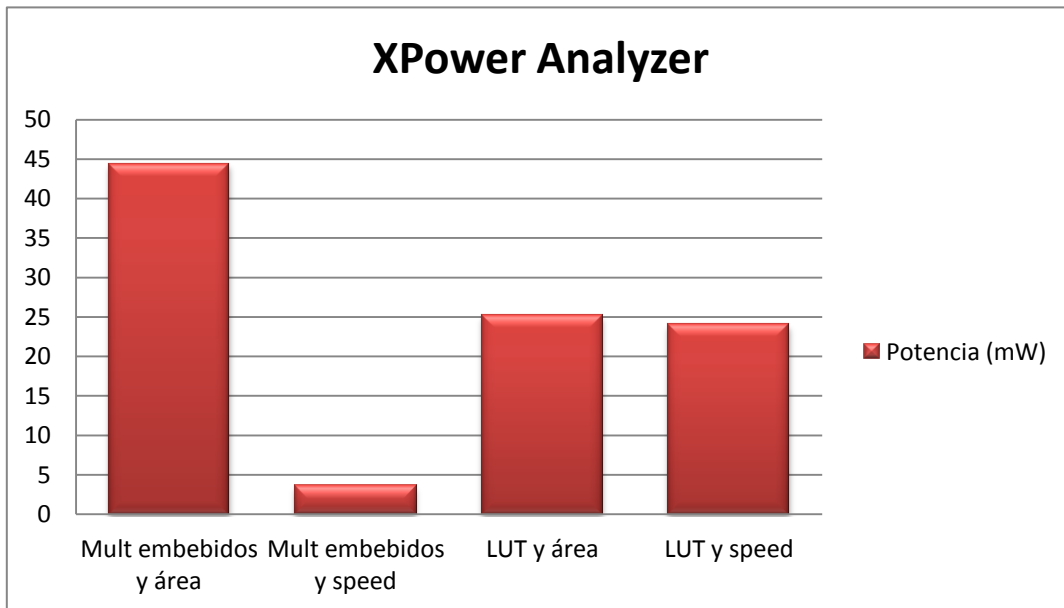


Figura 6.18. Comparativa de las medidas obtenidas con XPower Analyzer

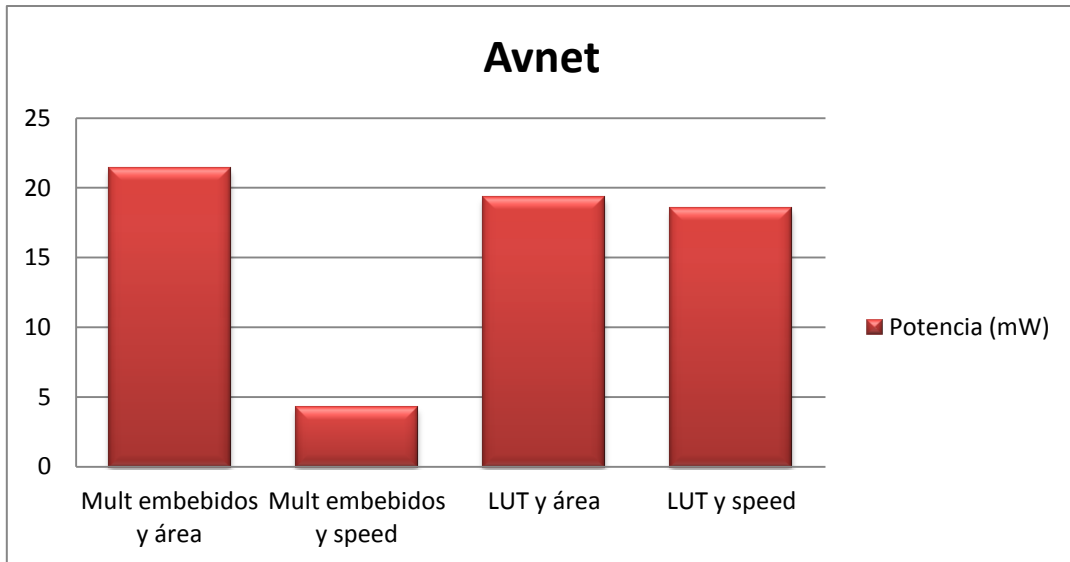


Figura 6.19. Comparativa de las medidas obtenidas con Avnet

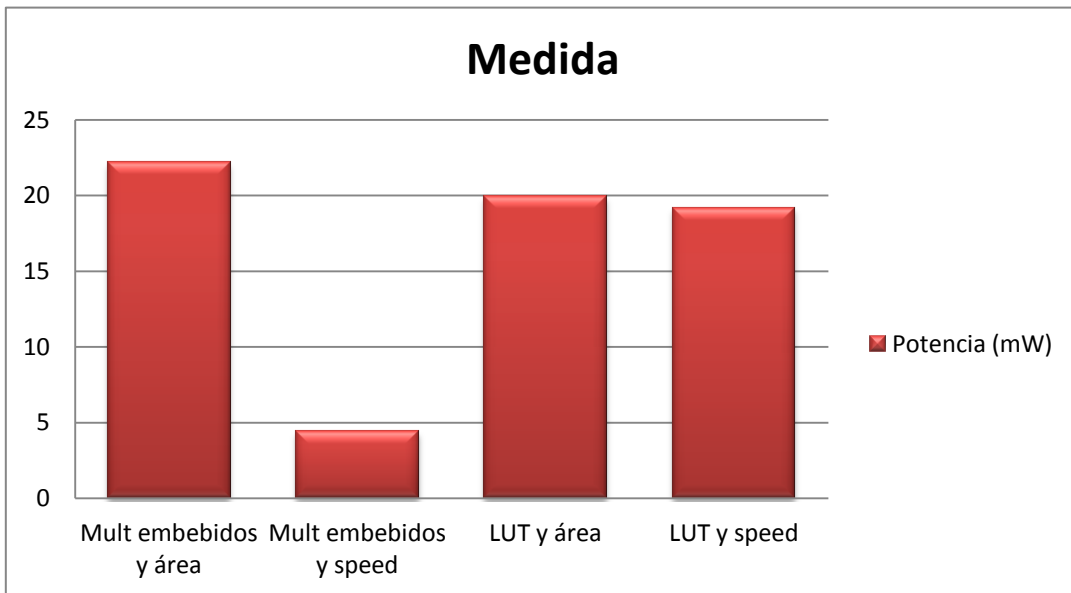


Figura 6.20. Comparativa de las medidas obtenidas midiendo sobre el circuito

6.1.1.3. Optimización de potencia

Las primeras gráficas muestran los resultados obtenidos para los diferentes multiplicadores.

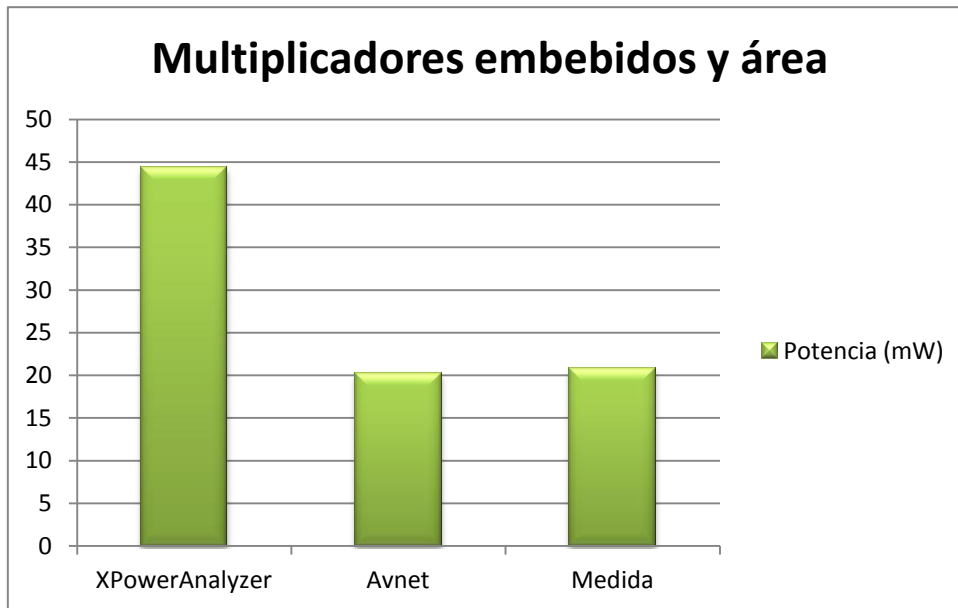


Figura 6.21. Medidas usando ISE y estrategia de diseño de optimización de potencia

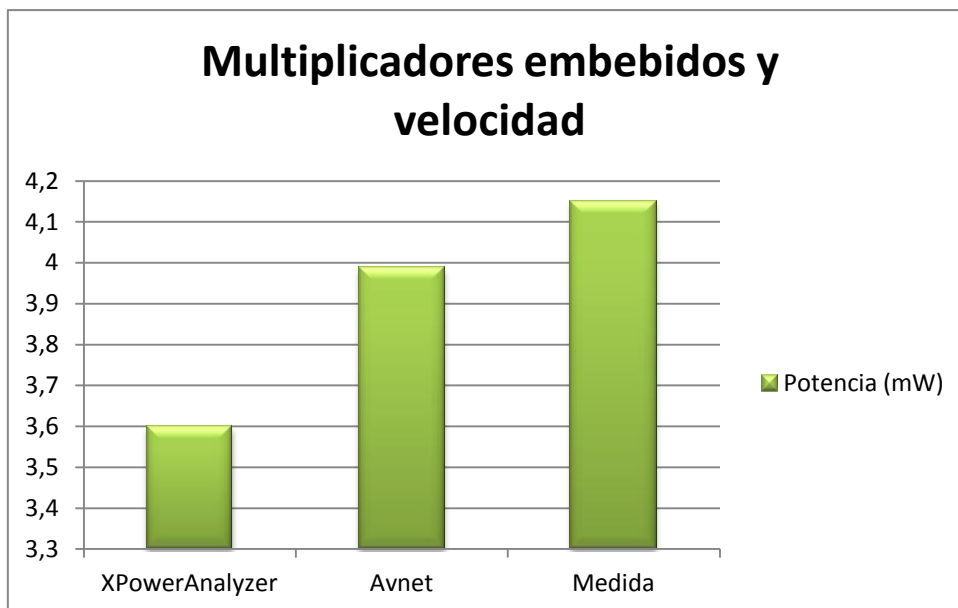


Figura 6.22. Medidas usando ISE y estrategia de diseño de optimización de potencia

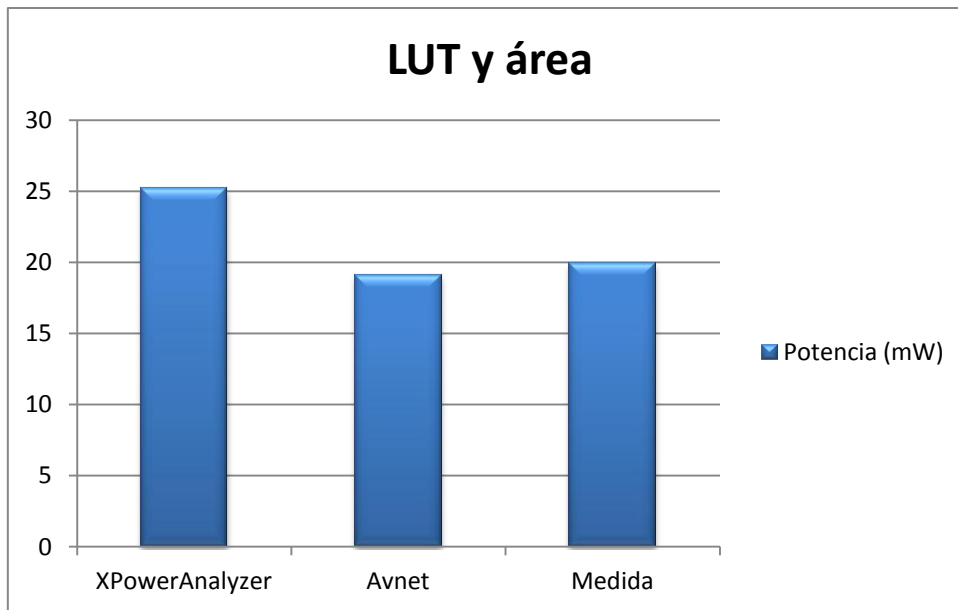


Figura 6.23. Medidas usando ISE y estrategia de diseño de optimización de potencia

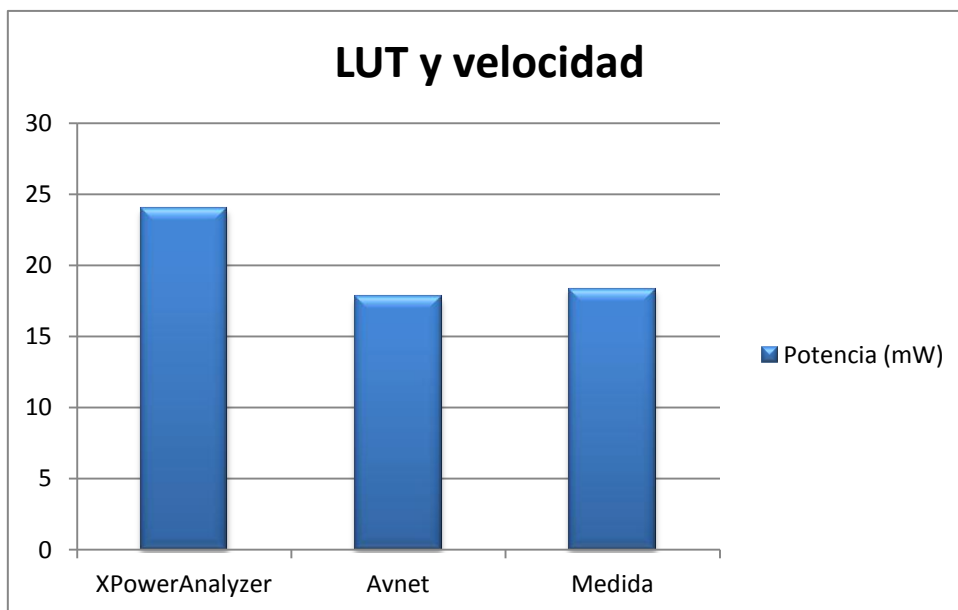


Figura 6.24. Medidas usando ISE y estrategia de diseño de optimización de potencia

El segundo grupo de gráficas muestra los resultados en función de las distintas técnicas de medición.

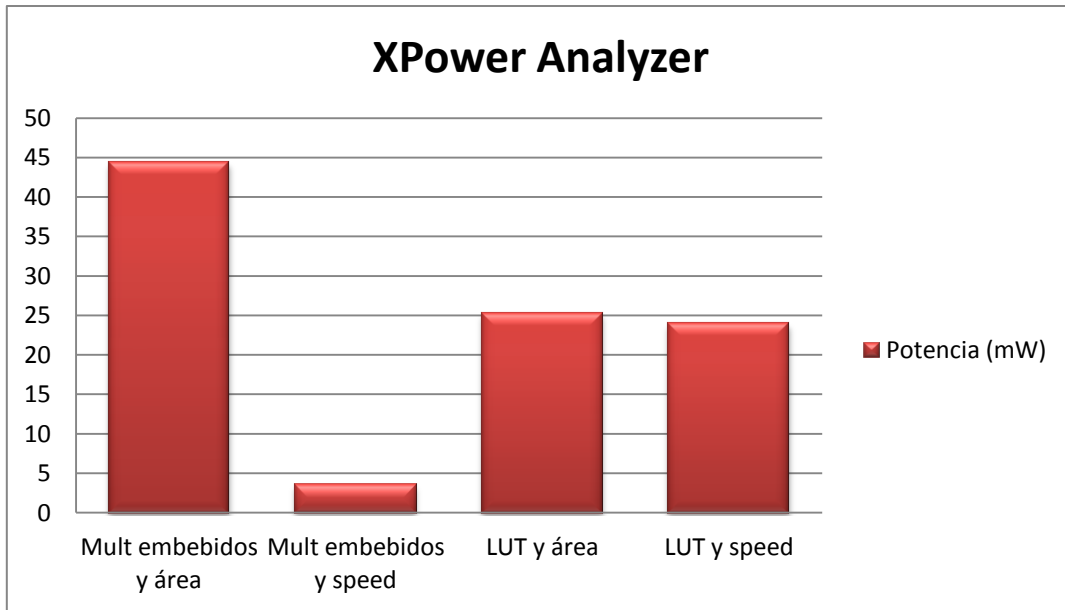


Figura 6.25. Comparativa de las medidas obtenidas con XPower Analyzer

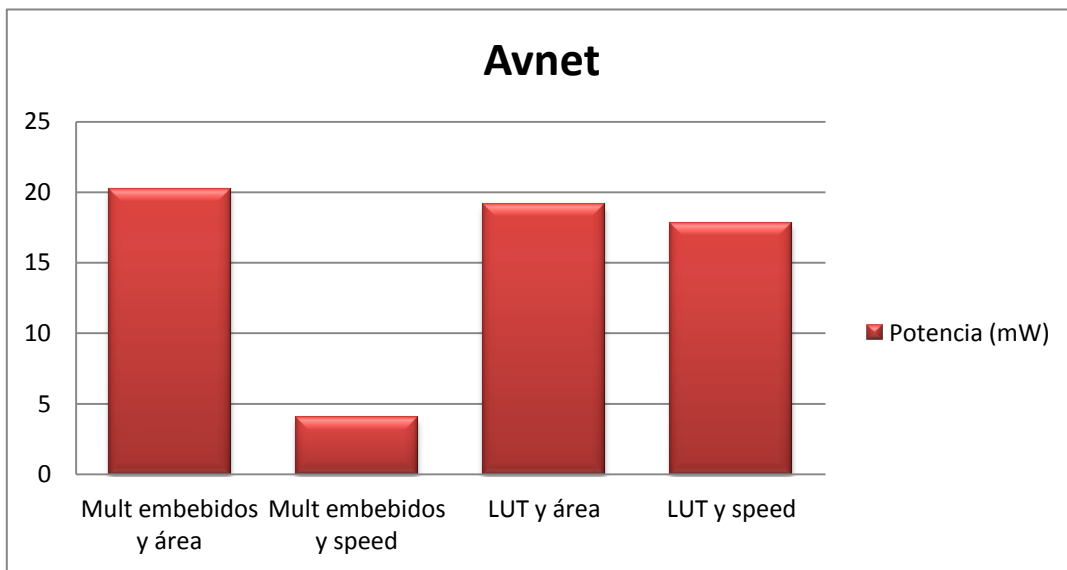


Figura 6.26. Comparativa de las medidas obtenidas con Avnet

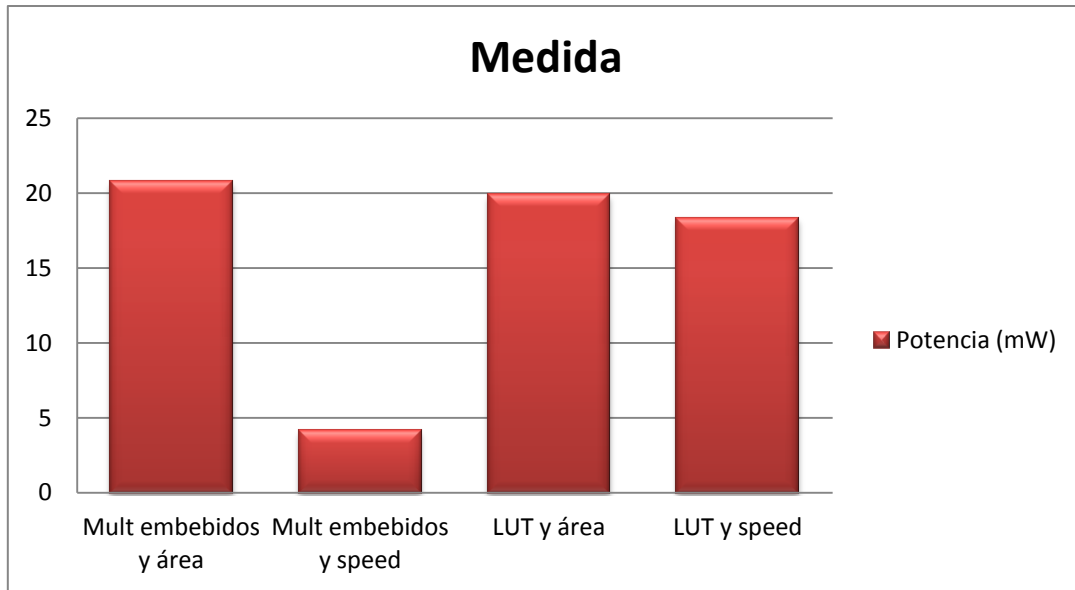


Figura 6.27. Comparativa de las medidas obtenidas midiendo sobre el circuito

6.1.1.4. Tiempo de ejecución

Las primeras gráficas muestran los resultados obtenidos para los diferentes multiplicadores.

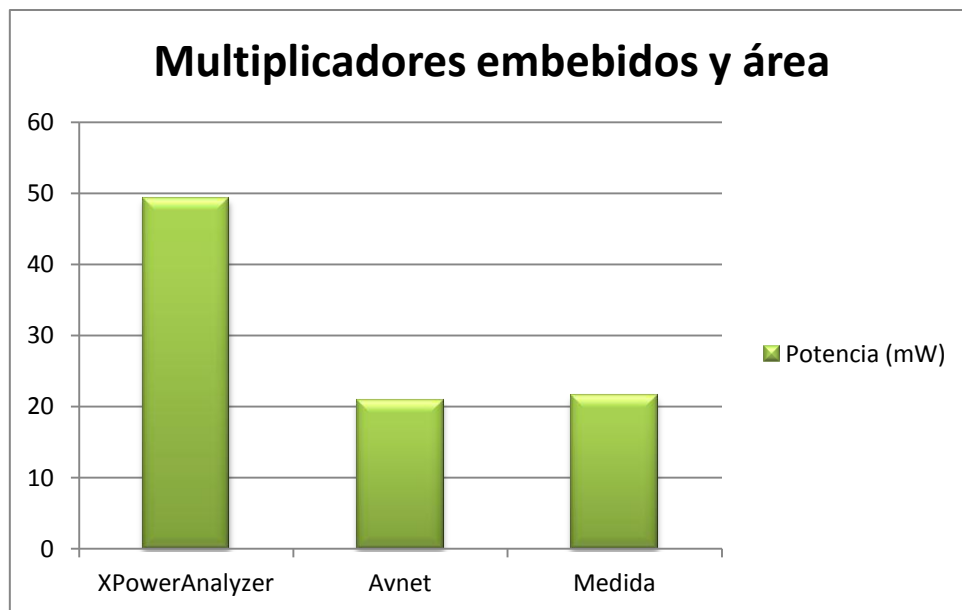


Figura 6.28. Medidas usando ISE y estrategia de diseño de optimización del tiempo de ejecución

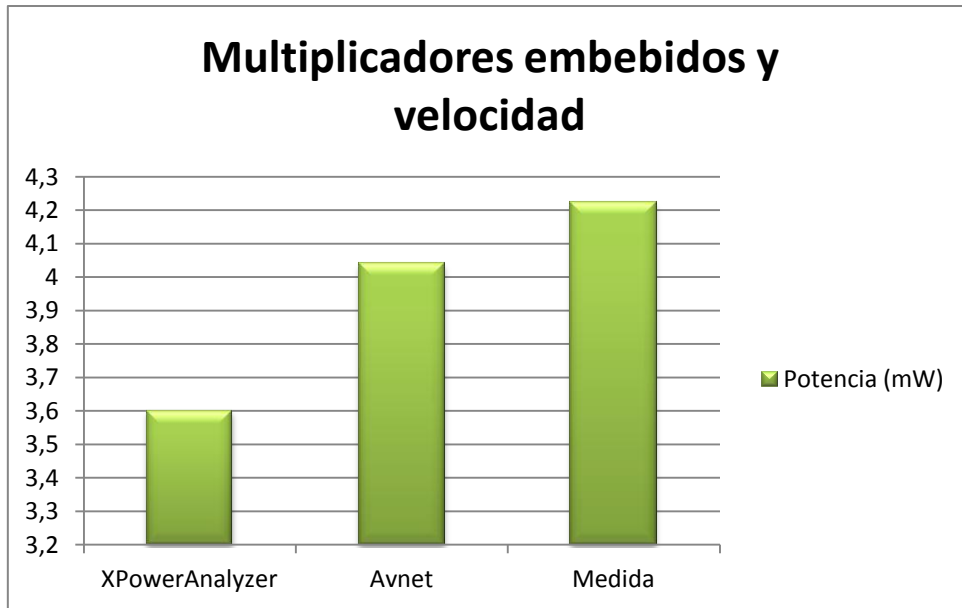


Figura 6.29. Medidas usando ISE y estrategia de diseño de optimización del tiempo de ejecución

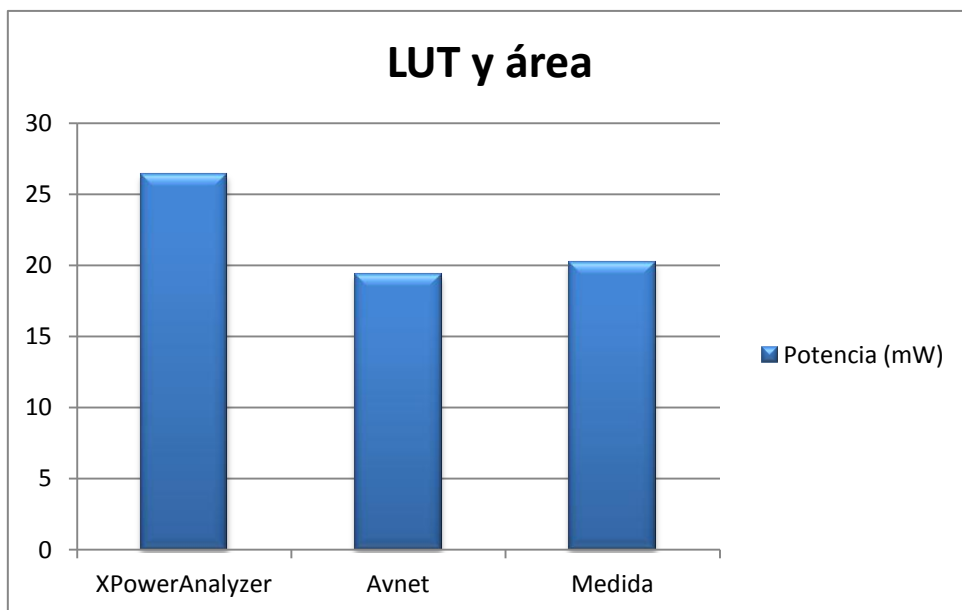


Figura 6.30. Medidas usando ISE y estrategia de diseño de optimización del tiempo de ejecución

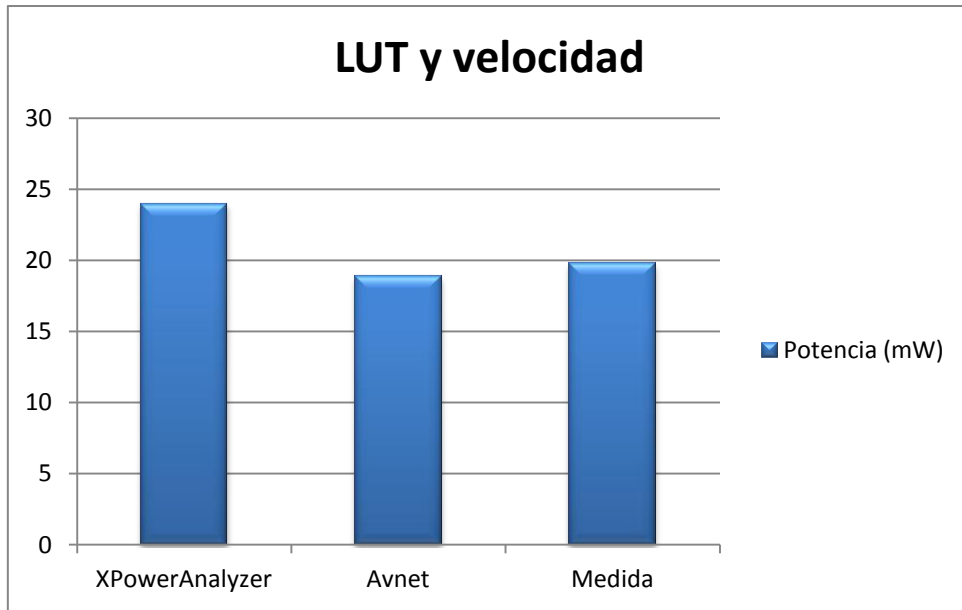


Figura 6.31. Medidas usando ISE y estrategia de diseño de optimización del tiempo de ejecución

El segundo grupo de gráficas muestra los resultados en función de las distintas técnicas de medición.

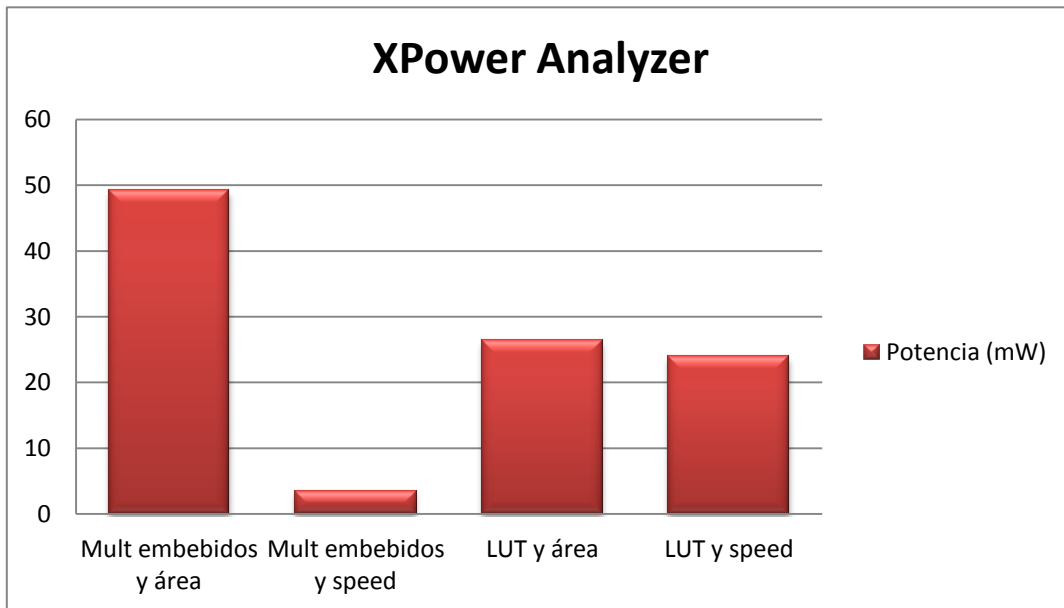


Figura 6.32. Comparativa de las medidas obtenidas con XPower Analyzer

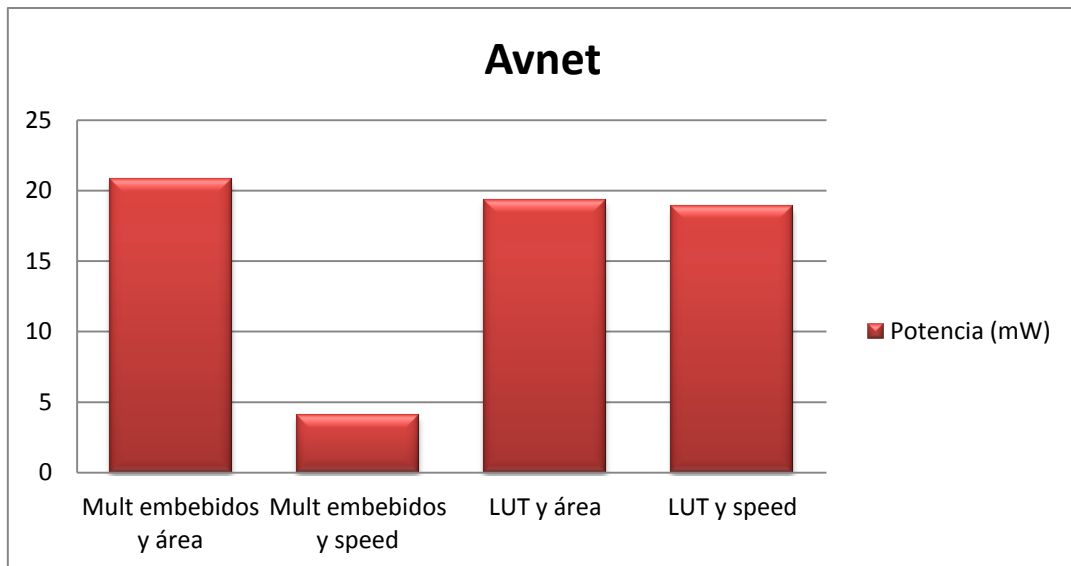


Figura 6.33. Comparativa de las medidas obtenidas con Avnet

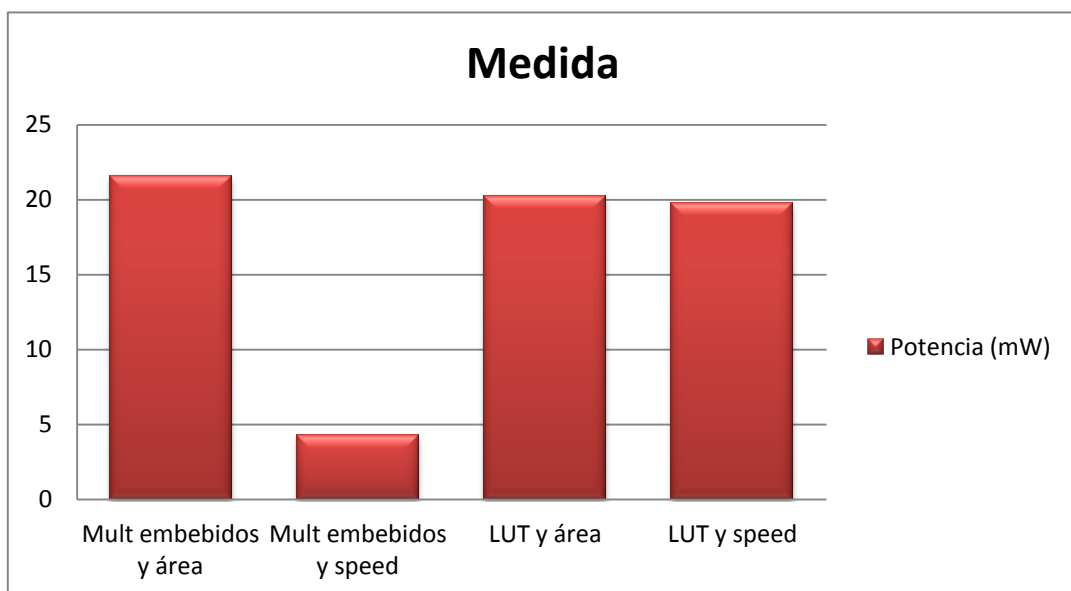


Figura 6.34. Comparativa de las medidas obtenidas midiendo sobre el circuito

6.1.1.5. Rendimiento temporal

Las primeras gráficas muestran los resultados obtenidos para los diferentes multiplicadores.

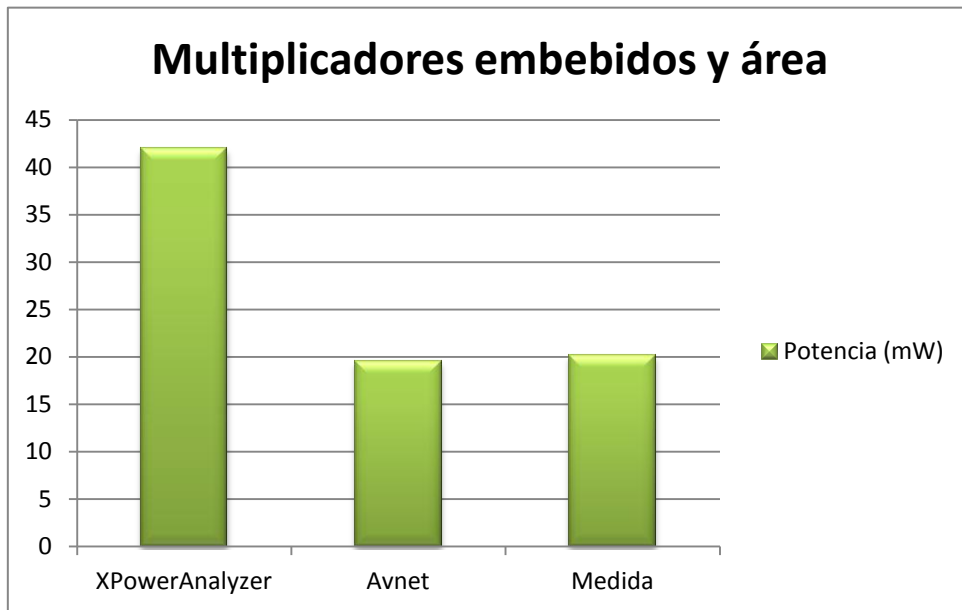


Figura 6.35. Medidas usando ISE y estrategia de diseño de optimizar el rendimiento temporal

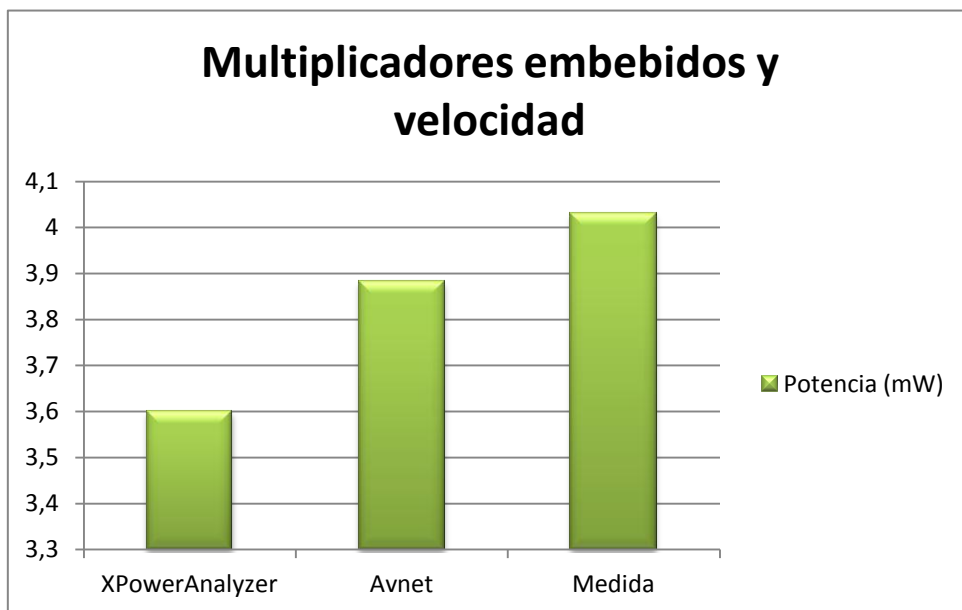


Figura 6.36. Medidas usando ISE y estrategia de diseño de optimizar el rendimiento temporal

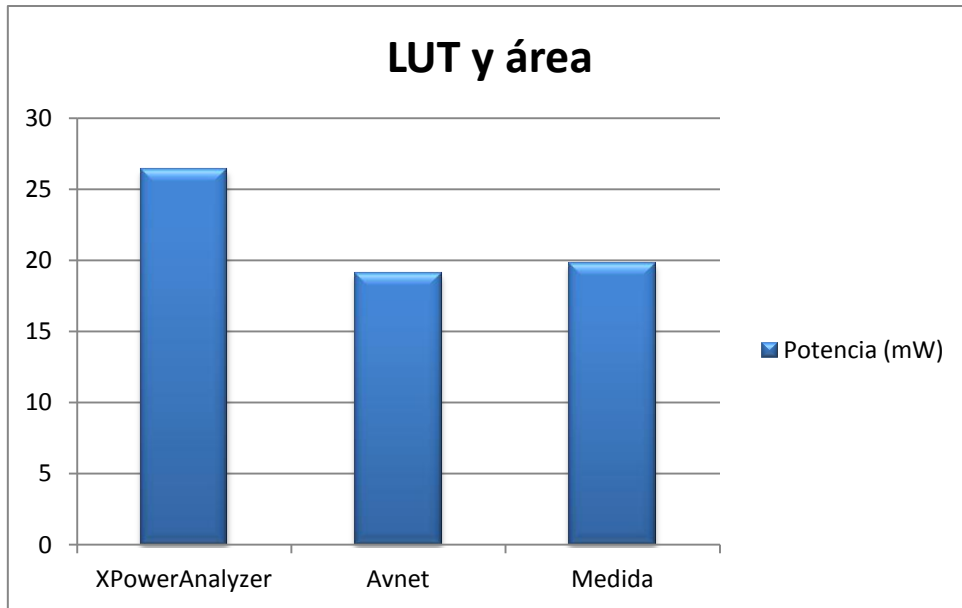


Figura 6.37. Medidas usando ISE y estrategia de diseño de optimizar el rendimiento temporal

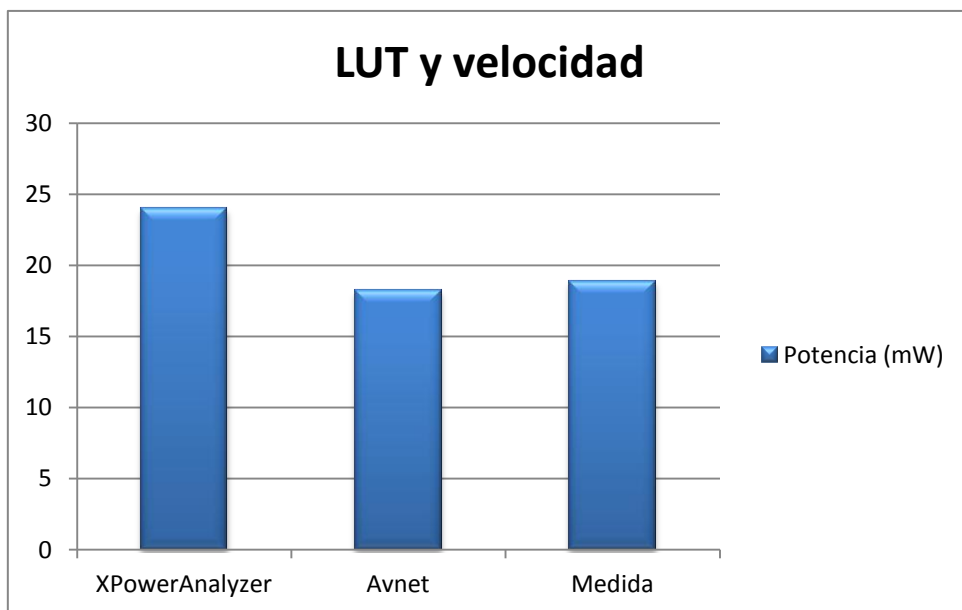


Figura 6.38. Medidas usando ISE y estrategia de diseño de optimizar el rendimiento temporal

El segundo grupo de gráficas muestra los resultados en función de las distintas técnicas de medición.

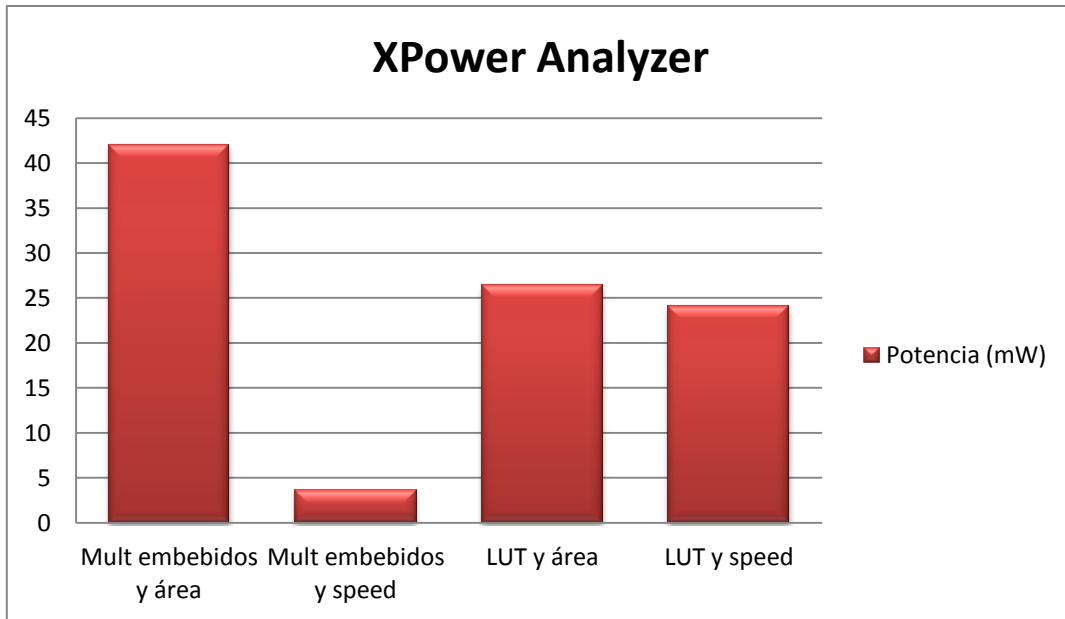


Figura 6.39. Comparativa de las medidas obtenidas con XPower Analyzer

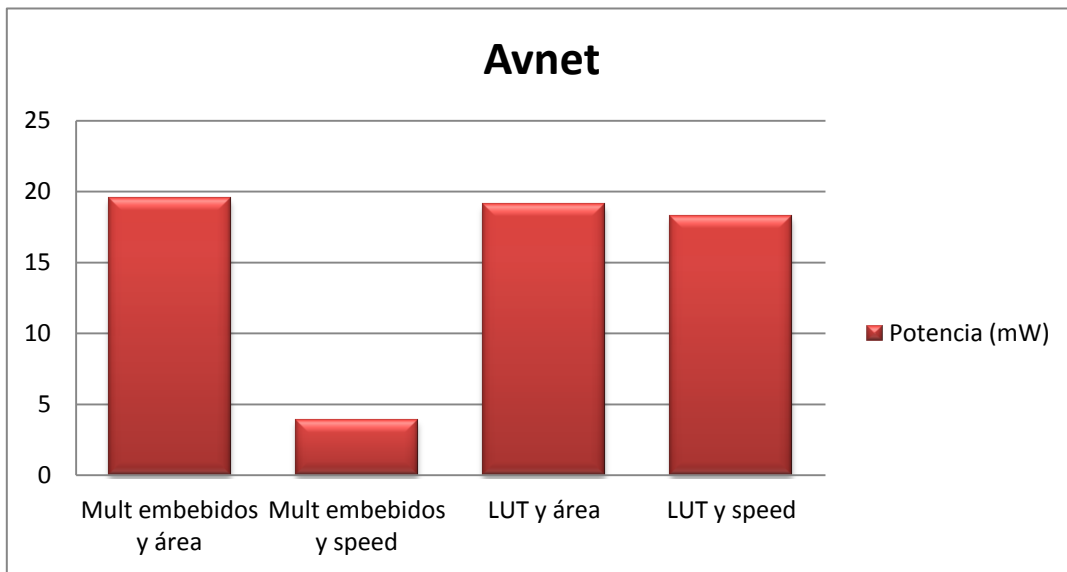


Figura 6.40. Comparativa de las medidas obtenidas con Avnet

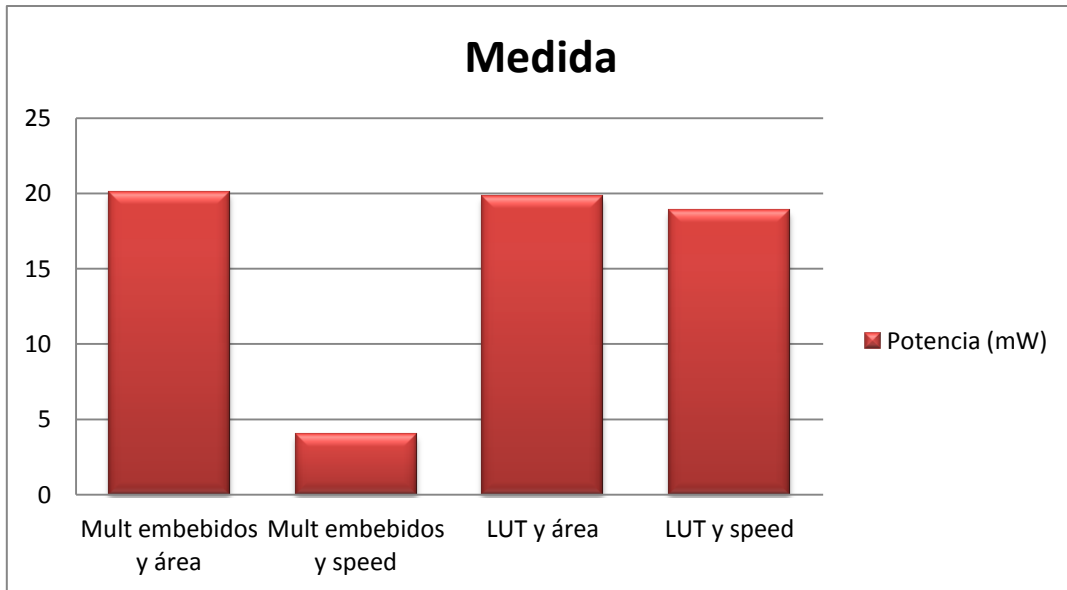


Figura 6.41. Comparativa de las medidas obtenidas midiendo sobre el circuito

6.1.2. PlanAhead

6.1.2.1. Balanceado

Las primeras gráficas muestran los resultados obtenidos para los diferentes multiplicadores.

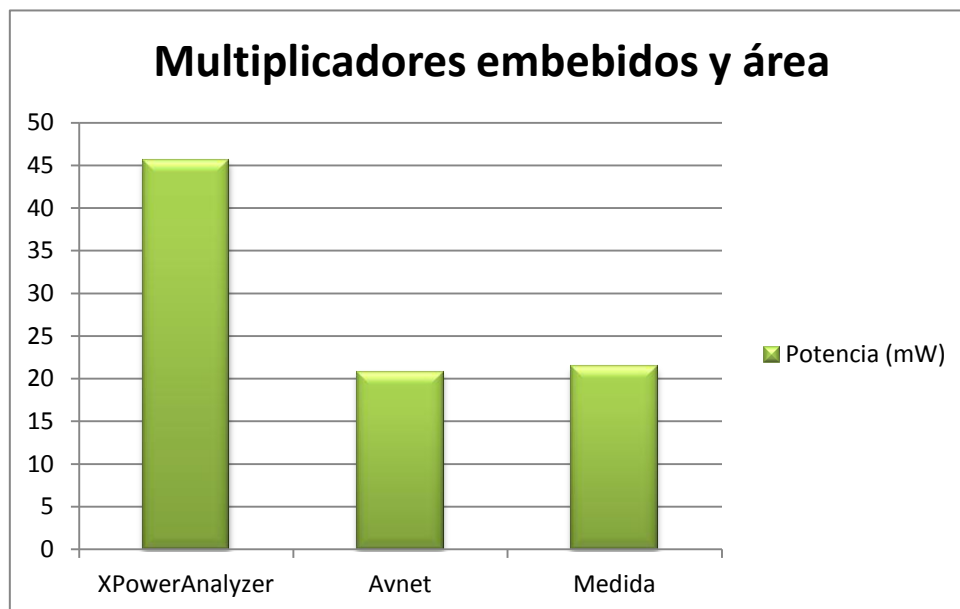


Figura 6.42. Medidas usando PlanAhead y estrategia de diseño balanceada

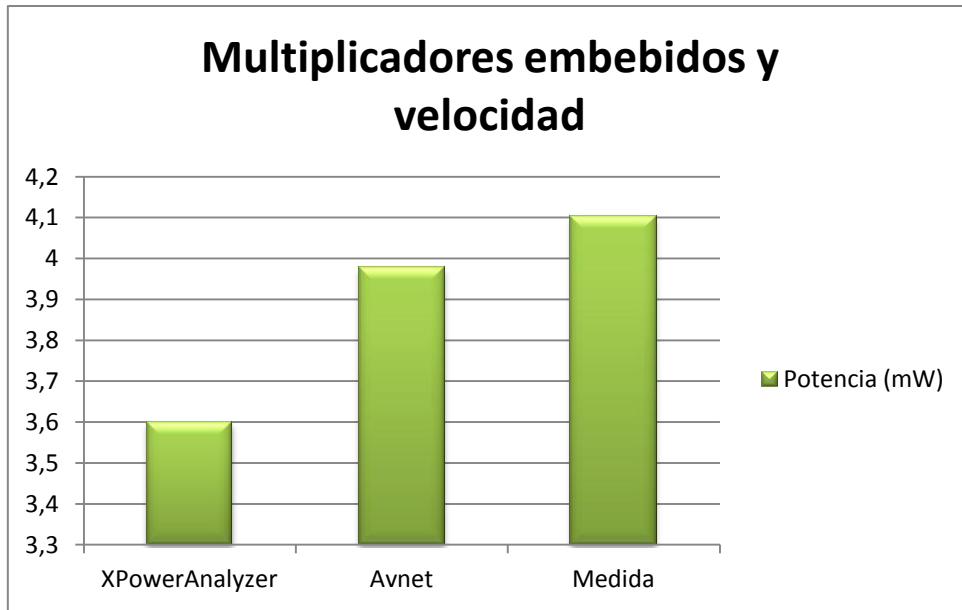


Figura 6.43. Medidas usando PlanAhead y estrategia de diseño balanceada

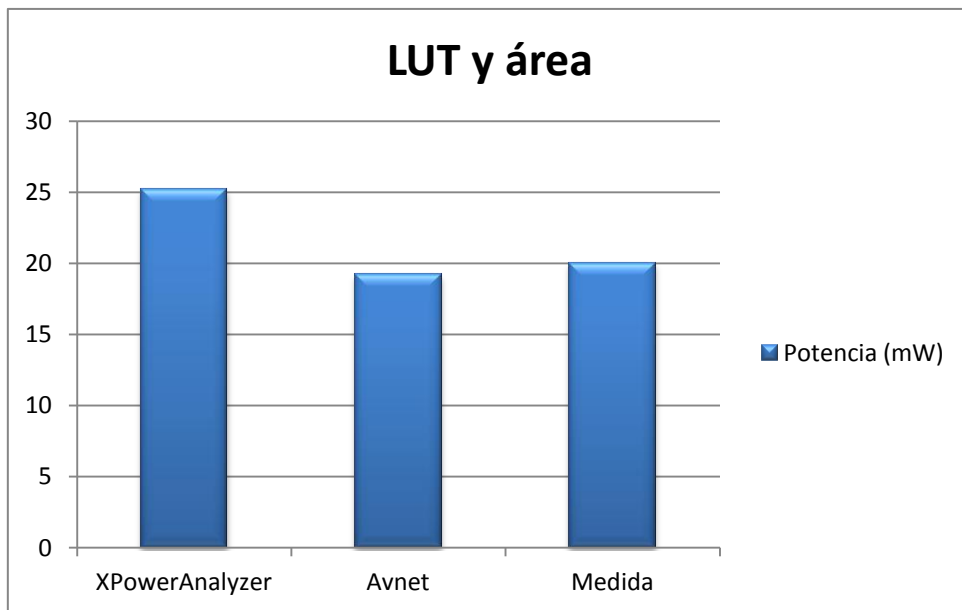


Figura 6.44. Medidas usando PlanAhead y estrategia de diseño balanceada

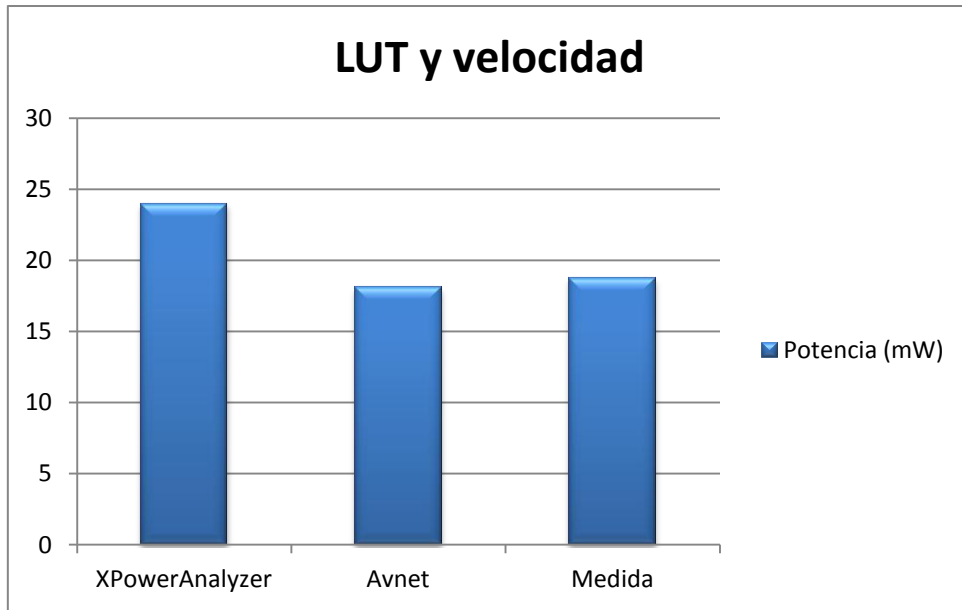


Figura 6.45. Medidas usando PlanAhead y estrategia de diseño balanceada

El segundo grupo de gráficas muestra los resultados en función de las distintas técnicas de medición.

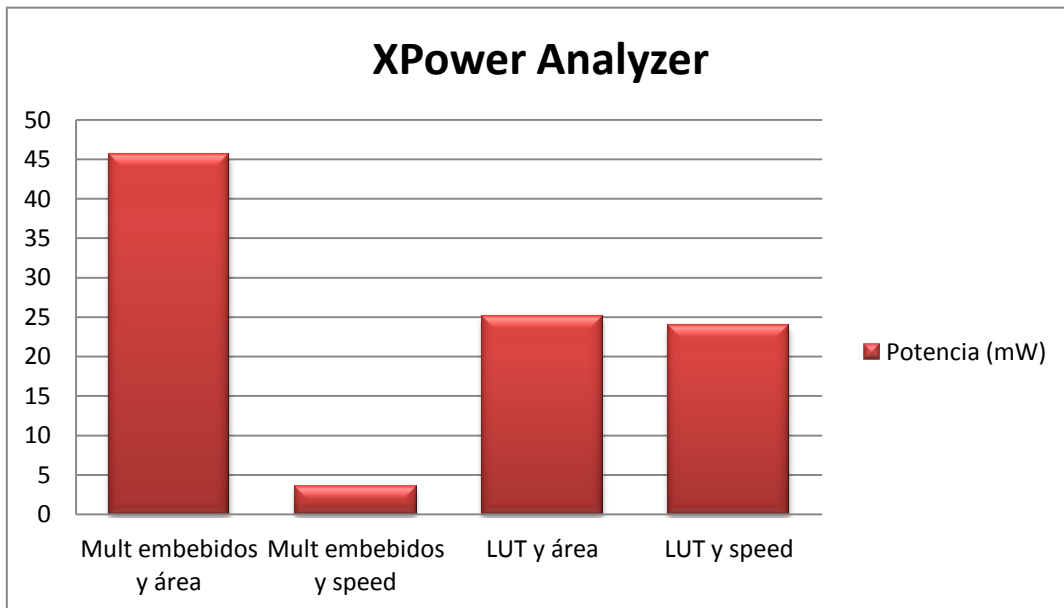


Figura 6.46. Comparativa de las medidas obtenidas con XPower Analyzer

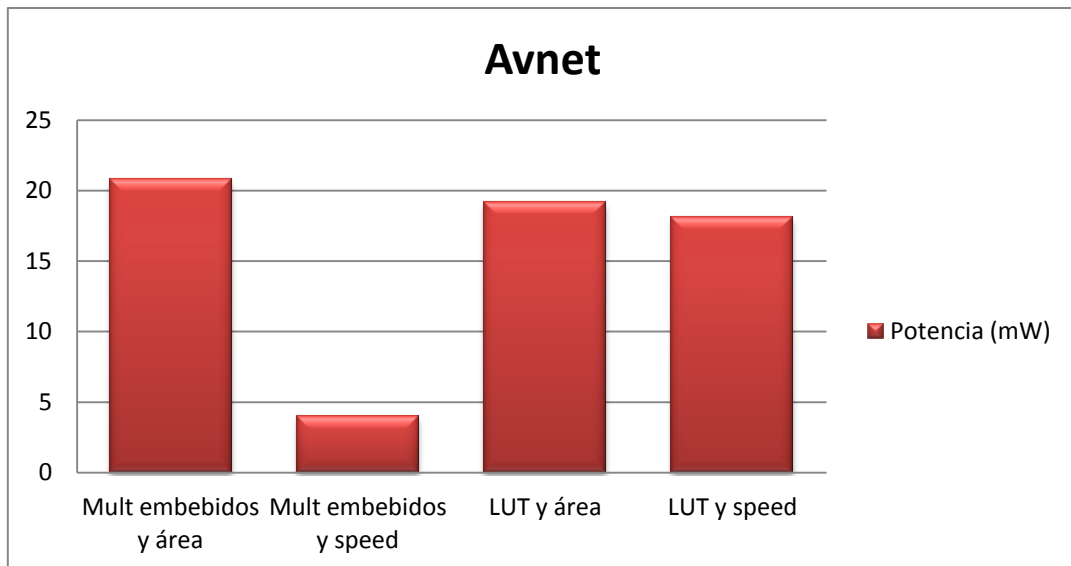


Figura 6.47. Comparativa de las medidas obtenidas con Avnet

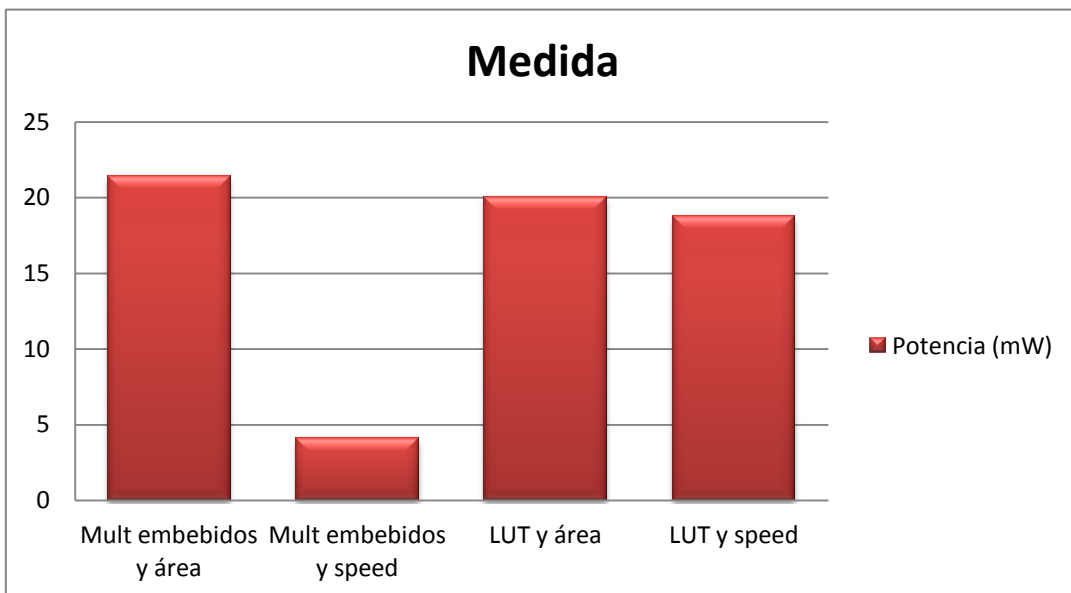


Figura 6.48. Comparativa de las medidas obtenidas midiendo sobre el circuito

6.1.2.2. Optimización de potencia

Las primeras gráficas muestran los resultados obtenidos para los diferentes multiplicadores.

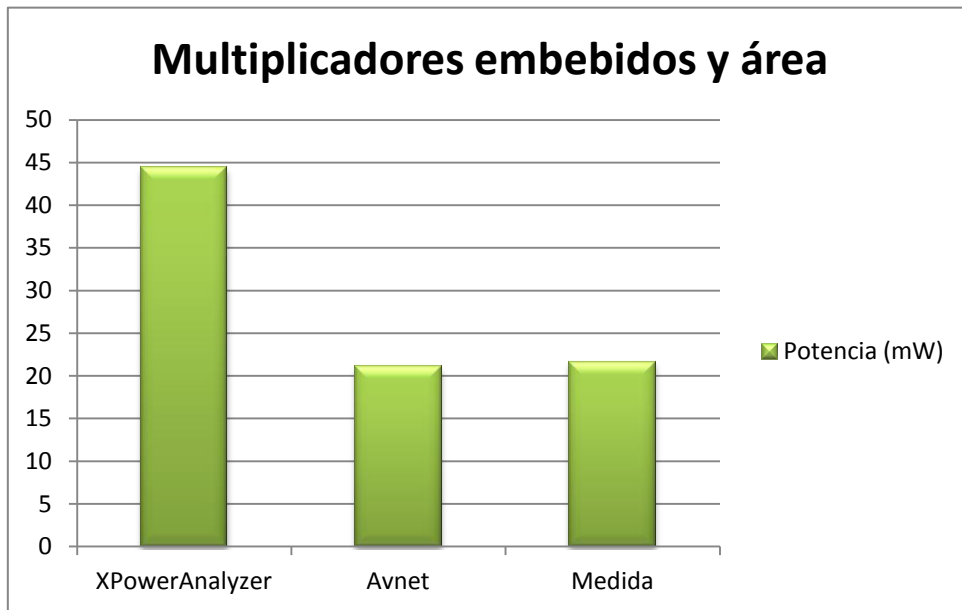


Figura 6.49. Medidas usando PlanAhead y estrategia de diseño de optimización de potencia

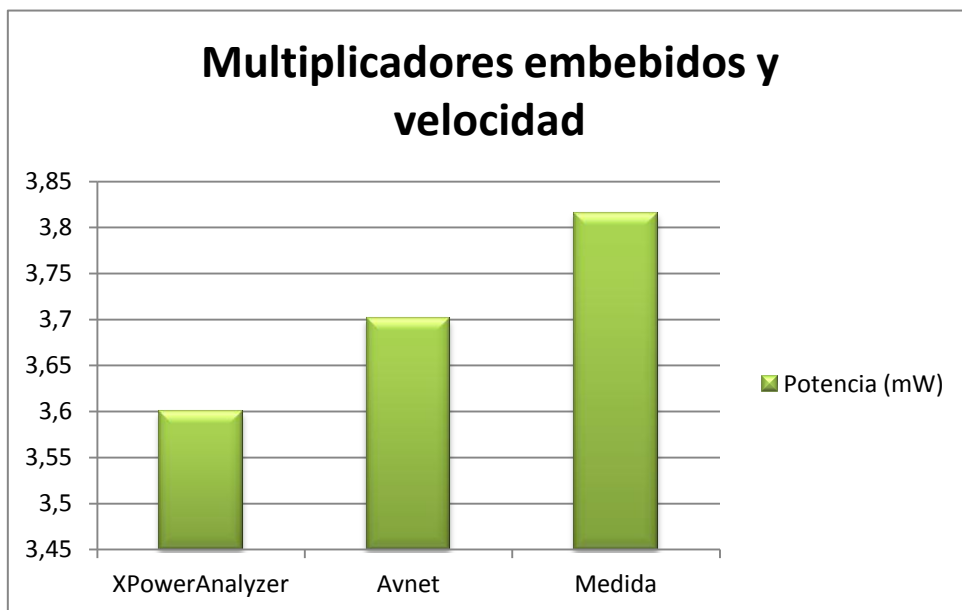


Figura 6.50. Medidas usando PlanAhead y estrategia de diseño de optimización de potencia

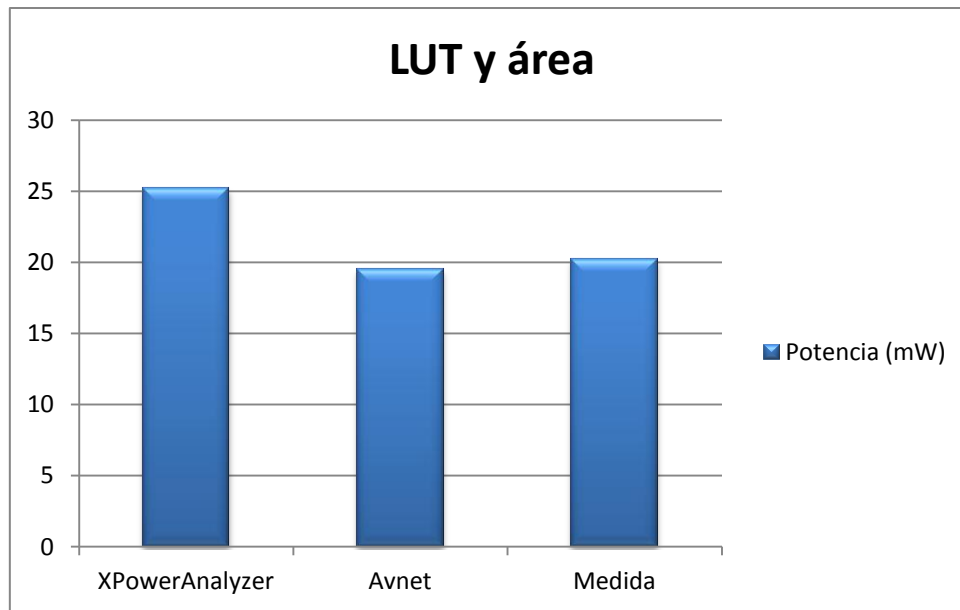


Figura 6.51. Medidas usando PlanAhead y estrategia de diseño de optimización de potencia

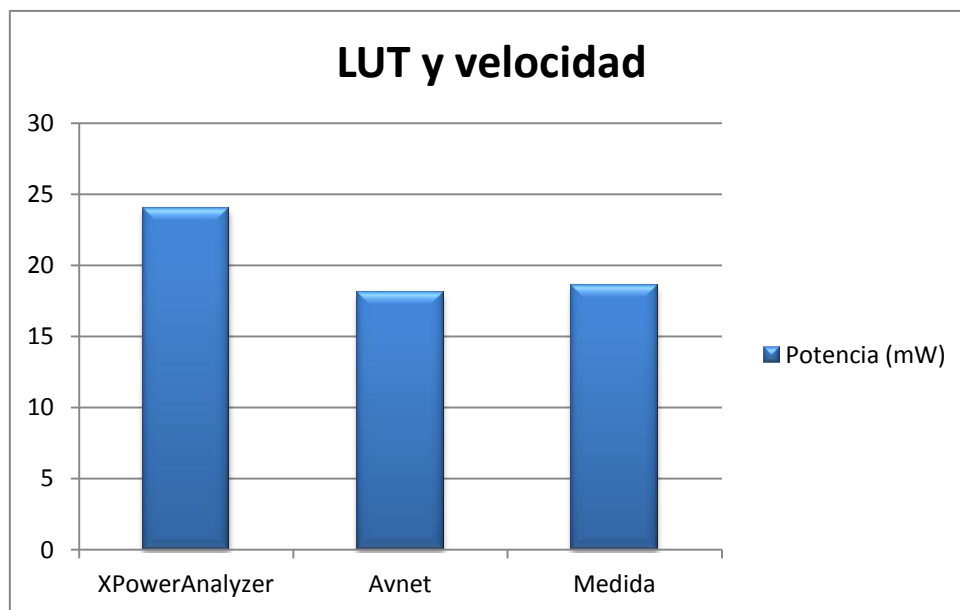


Figura 6.52. Medidas usando PlanAhead y estrategia de diseño de optimización de potencia

El segundo grupo de gráficas muestra los resultados en función de las distintas técnicas de medición.

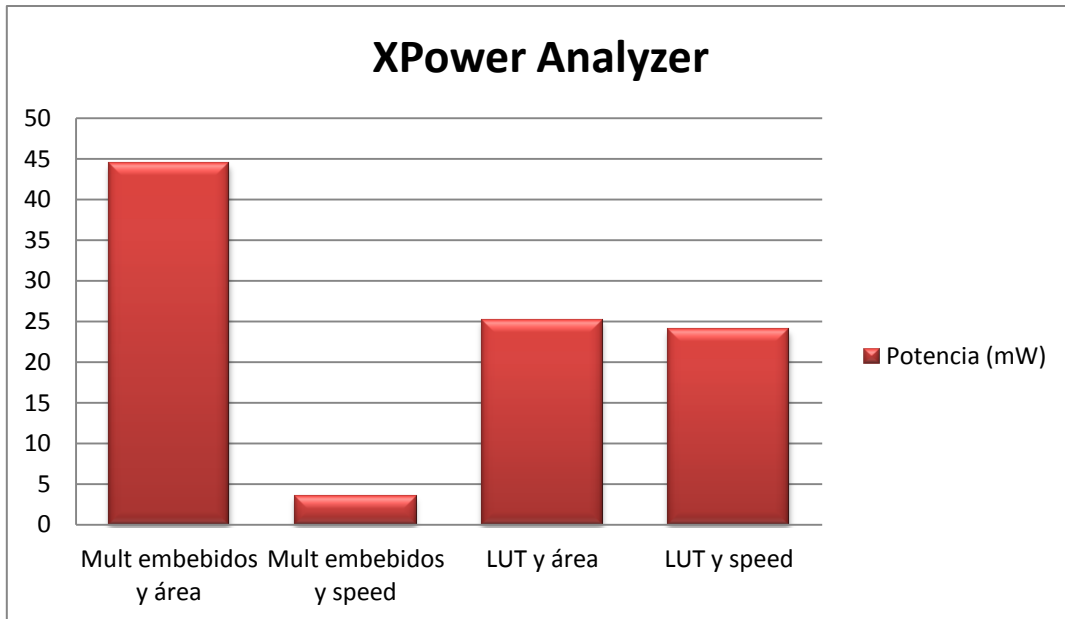


Figura 6.53. Comparativa de las medidas obtenidas con XPower Analyzer

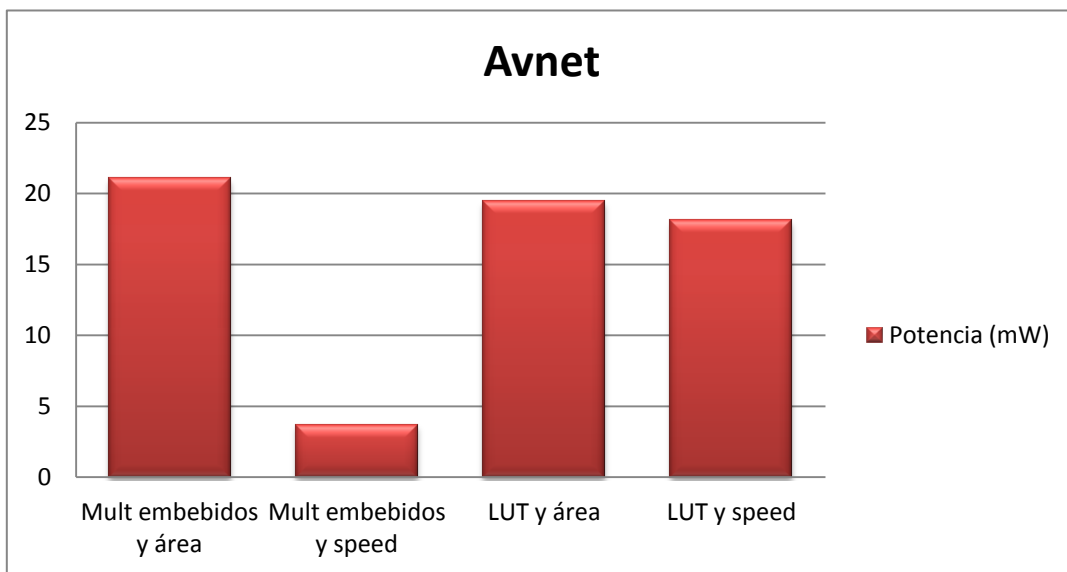


Figura 6.54. Comparativa de las medidas obtenidas con Avnet

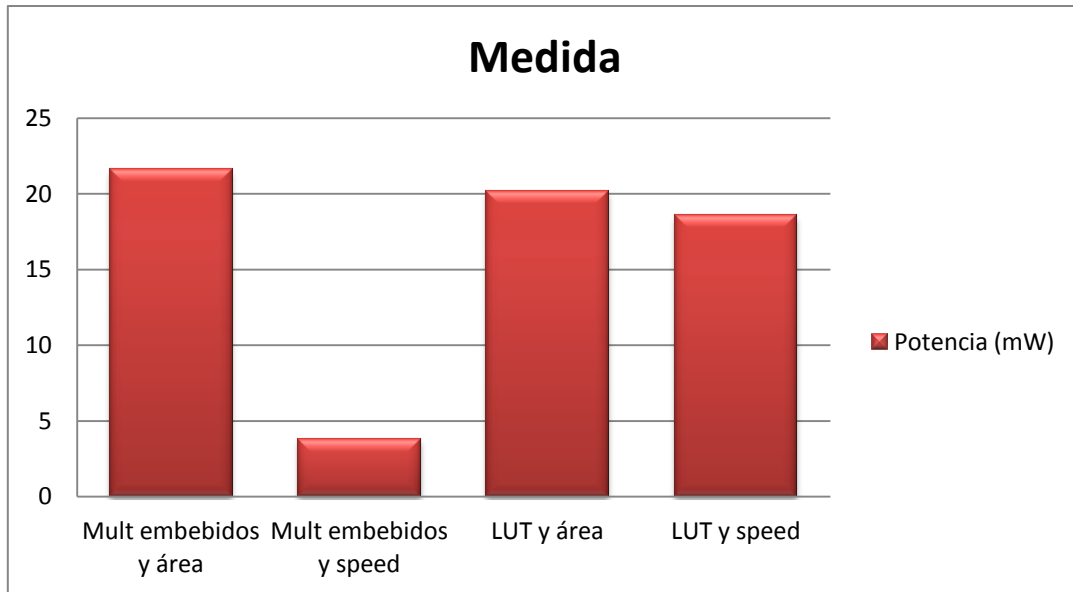


Figura 6.55. Comparativa de las medidas obtenidas midiendo sobre el circuito

6.1.2.3. Rendimiento temporal con IOB (Input/Output Blocks) Packing

Las primeras gráficas muestran los resultados obtenidos para los diferentes multiplicadores.

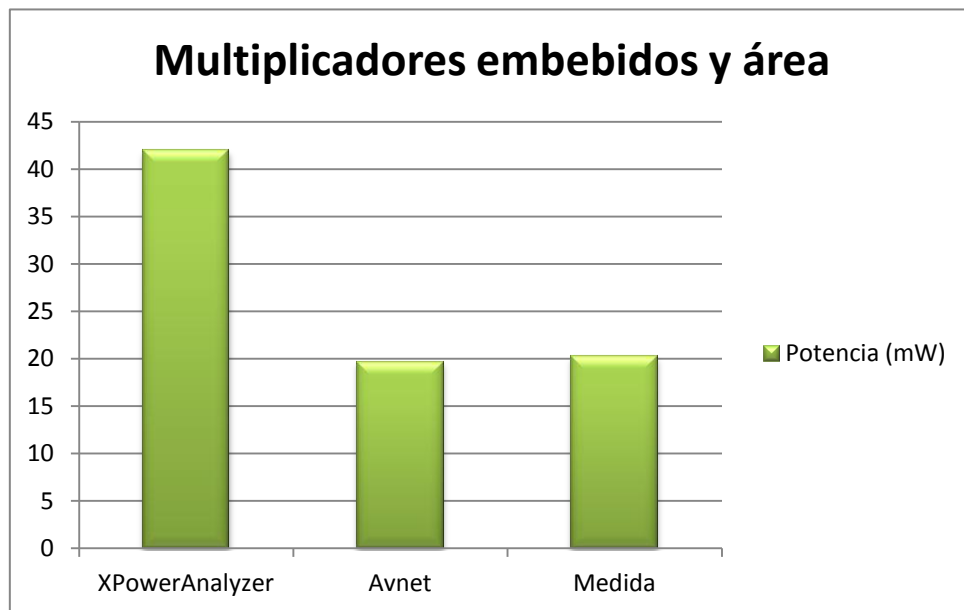


Figura 6.56. Medidas usando PlanAhead y estrategia de diseño de optimización del rendimiento temporal con IOB

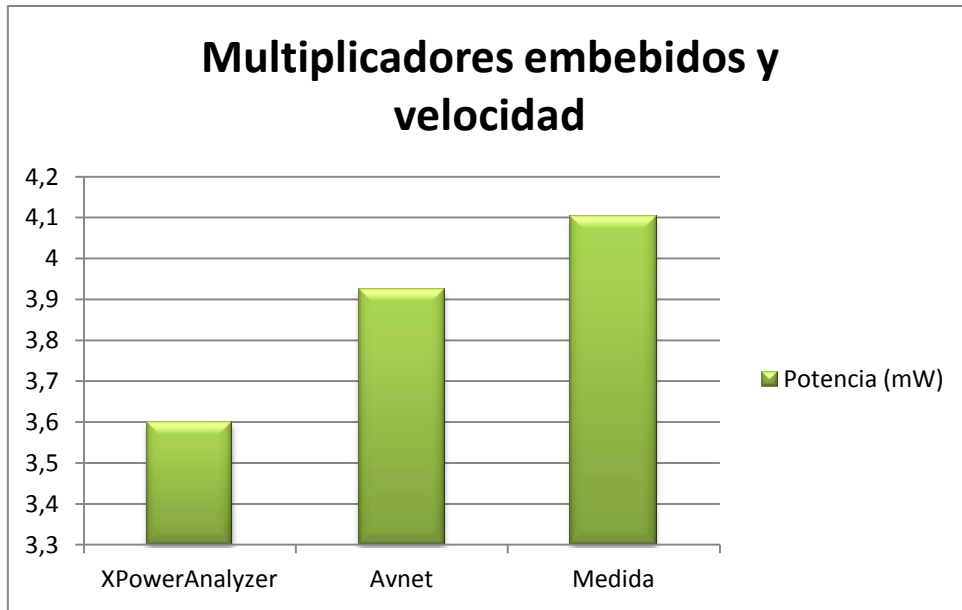


Figura 6.57. Medidas usando PlanAhead y estrategia de diseño de optimización del rendimiento temporal con IOB

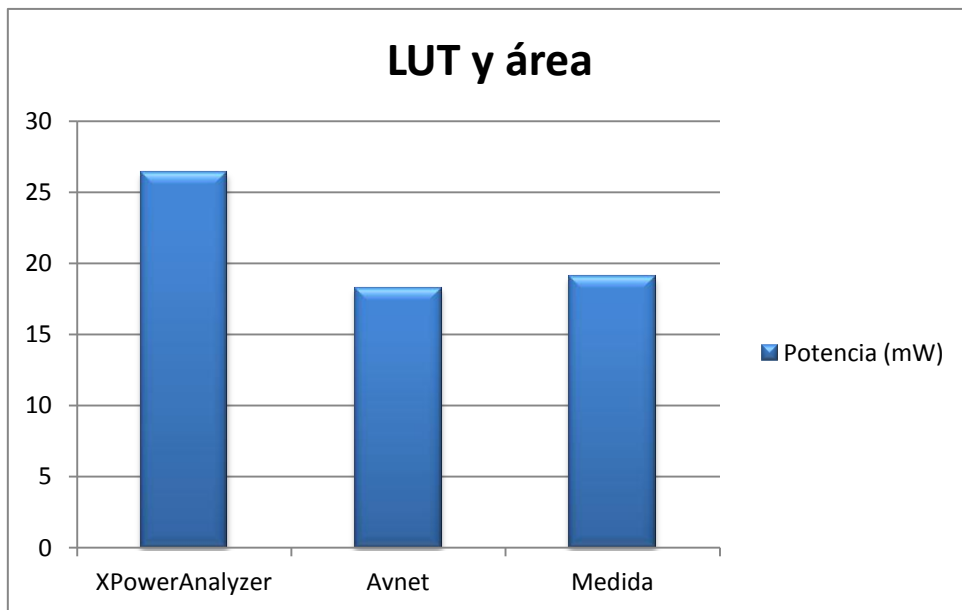


Figura 6.58. Medidas usando PlanAhead y estrategia de diseño de optimización del rendimiento temporal con IOB

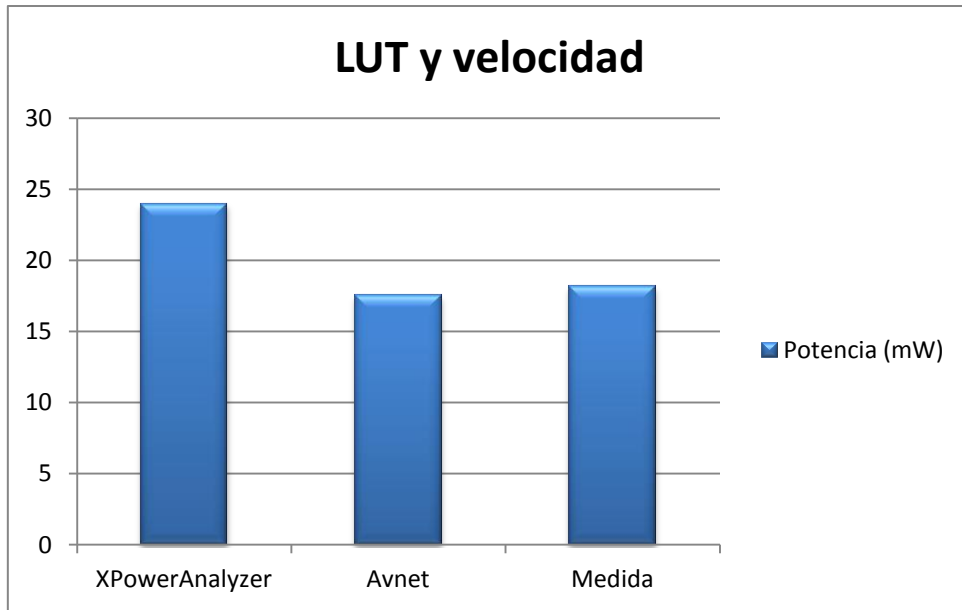


Figura 6.59. Medidas usando PlanAhead y estrategia de diseño de optimización del rendimiento temporal con IOB

El segundo grupo de gráficas muestra los resultados en función de las distintas técnicas de medición.

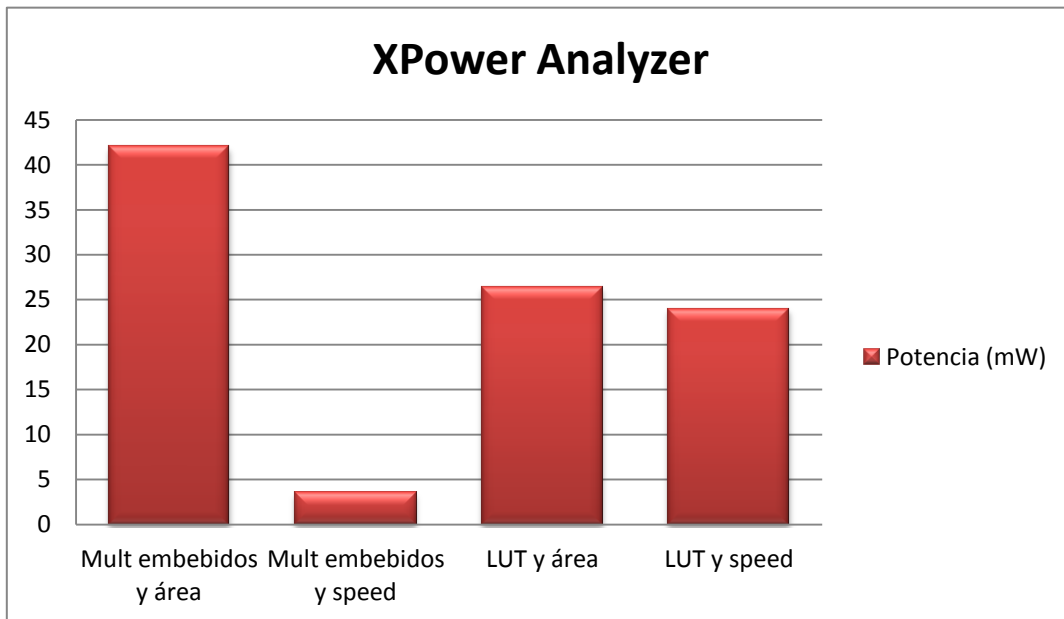


Figura 6.60. Comparativa de las medidas obtenidas con XPower Analyzer

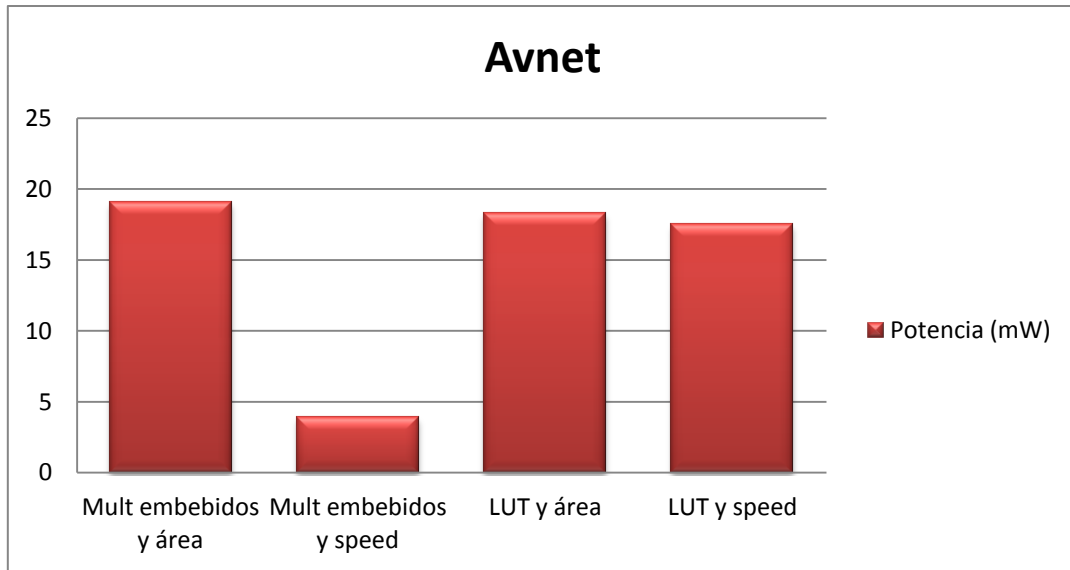


Figura 6.61. Comparativa de las medidas obtenidas con Avnet

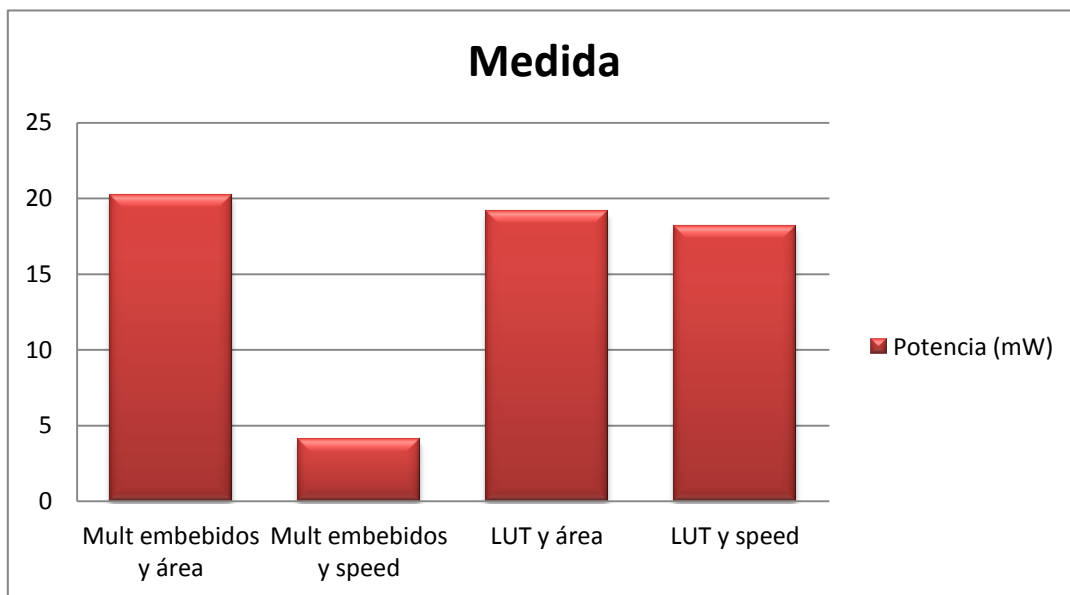


Figura 6.62. Comparativa de las medidas obtenidas midiendo sobre el circuito

6.1.2.4. Rendimiento temporal sin IOB (Input/Output Blocks) Packing

Las primeras gráficas muestran los resultados obtenidos para los diferentes multiplicadores.

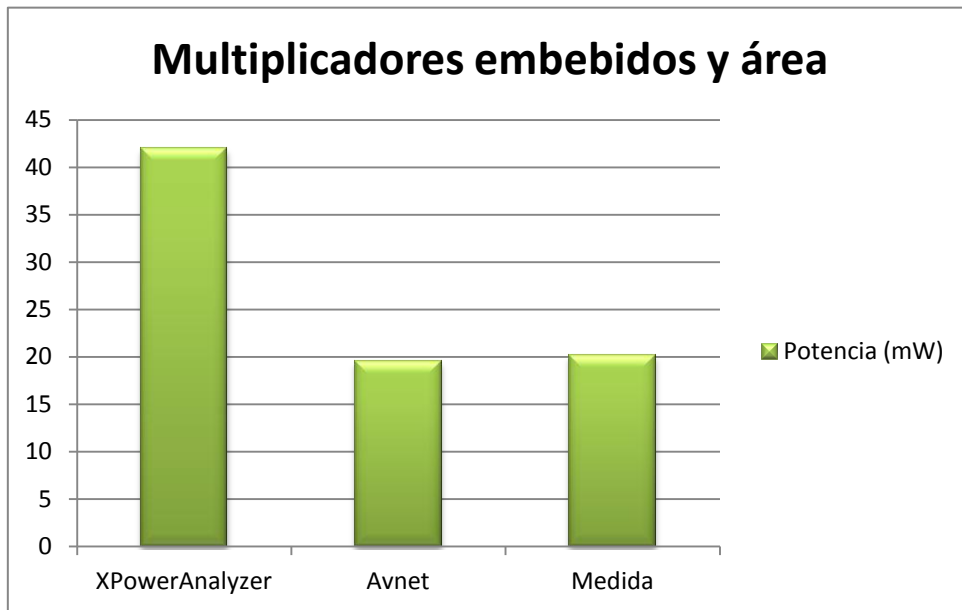


Figura 6.63. Medidas usando PlanAhead y estrategia de diseño de optimización del rendimiento temporal sin IOB

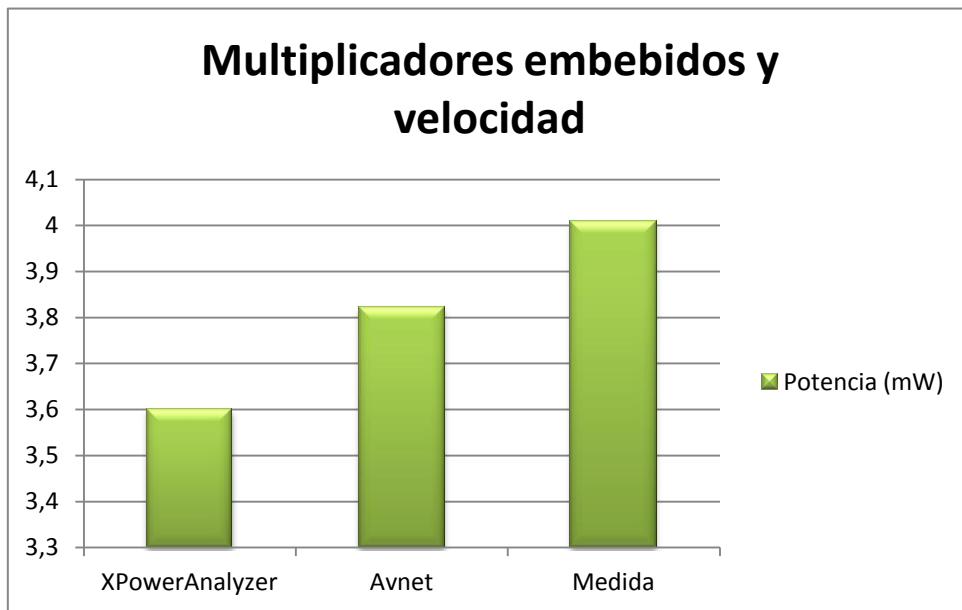


Figura 6.64. Medidas usando PlanAhead y estrategia de diseño de optimización del rendimiento temporal sin IOB

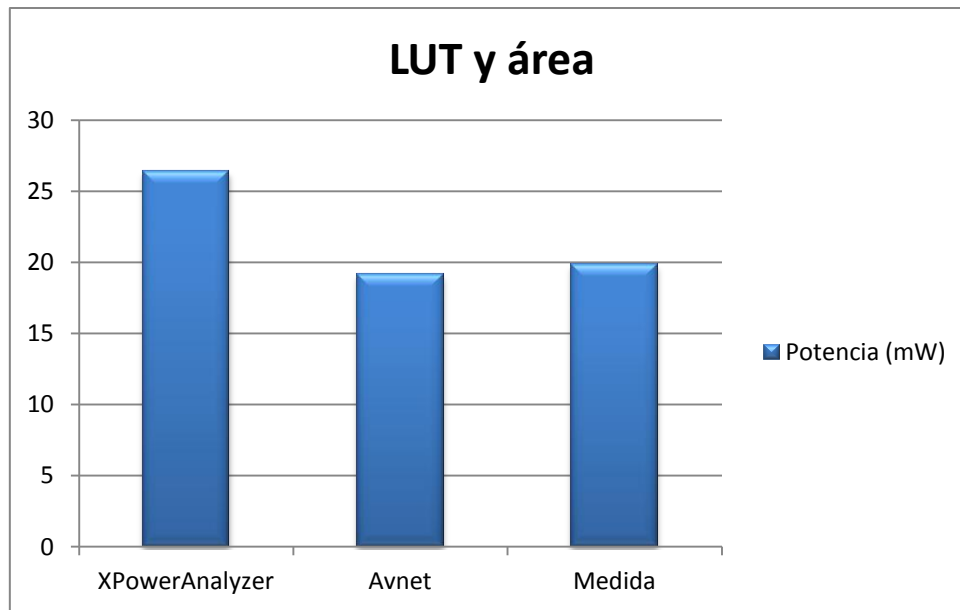


Figura 6.65. Medidas usando PlanAhead y estrategia de diseño de optimización del rendimiento temporal sin IOB

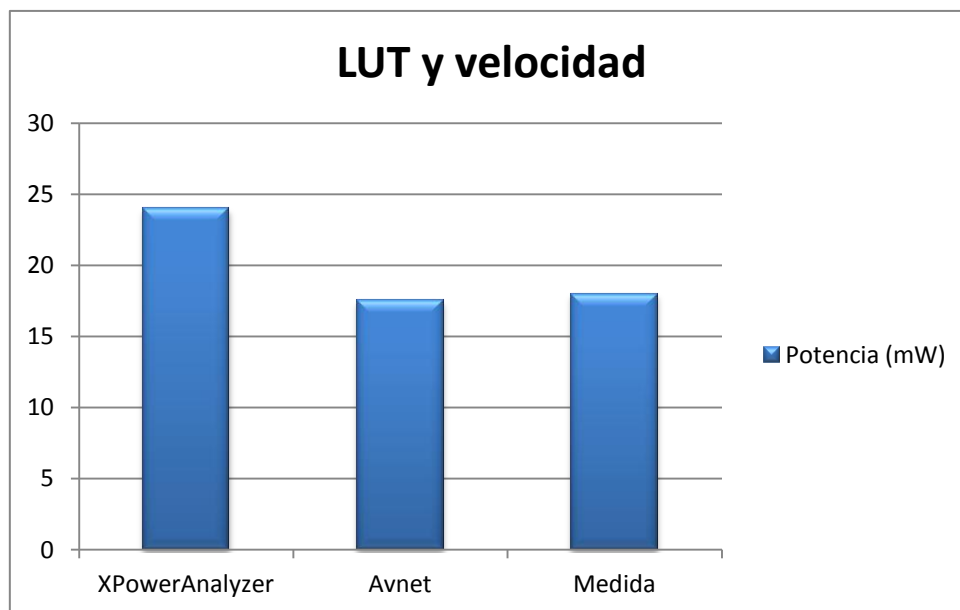


Figura 6.66. Medidas usando PlanAhead y estrategia de diseño de optimización del rendimiento temporal sin IOB

El segundo grupo de gráficas muestra los resultados en función de las distintas técnicas de medición.

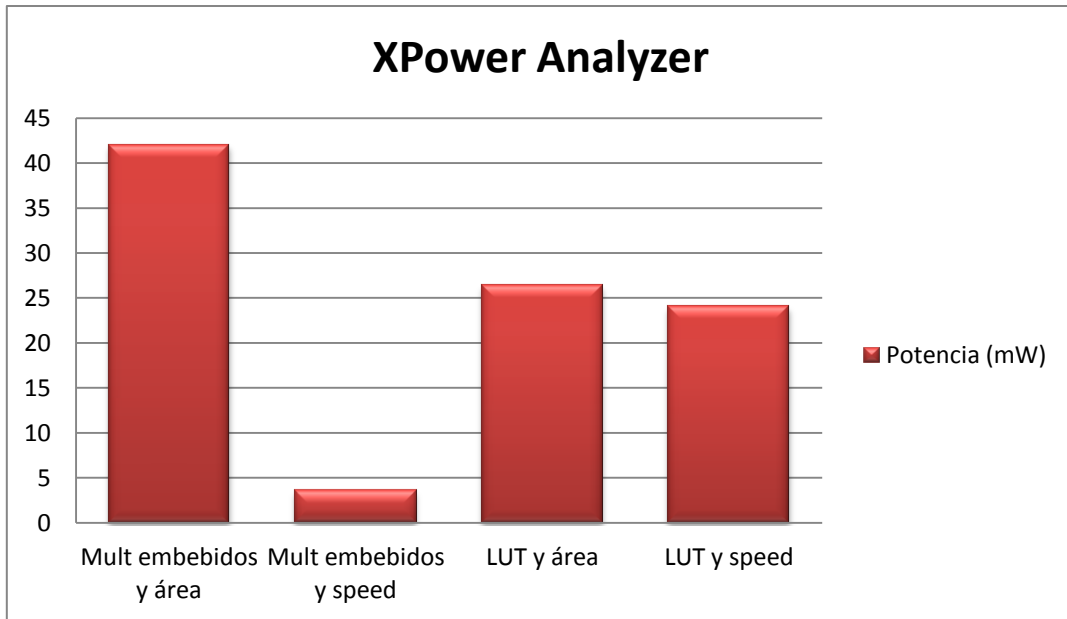


Figura 6.67. Comparativa de las medidas obtenidas con XPower Analyzer

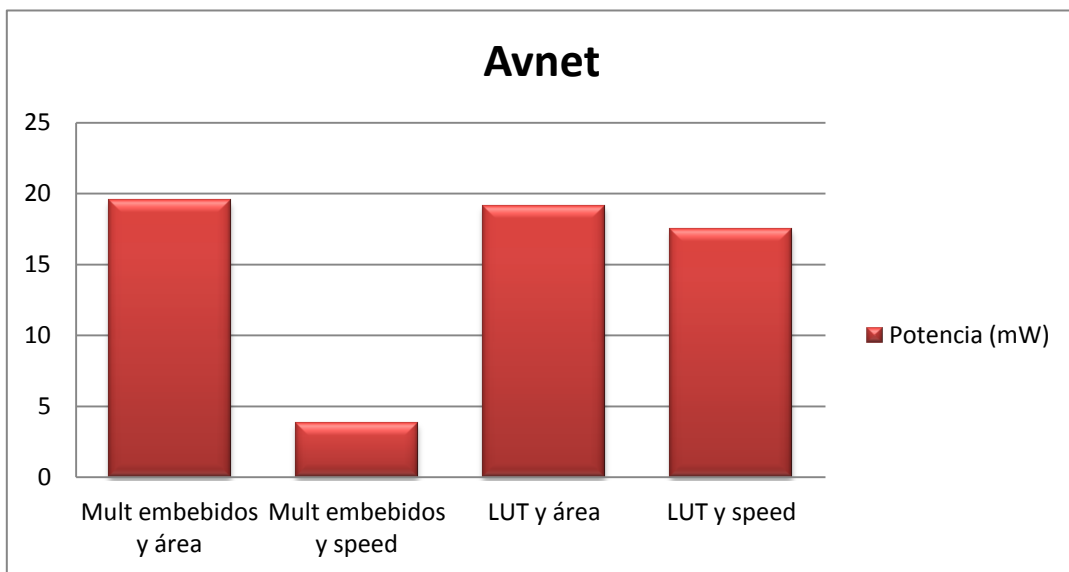


Figura 6.68. Comparativa de las medidas obtenidas con Avnet

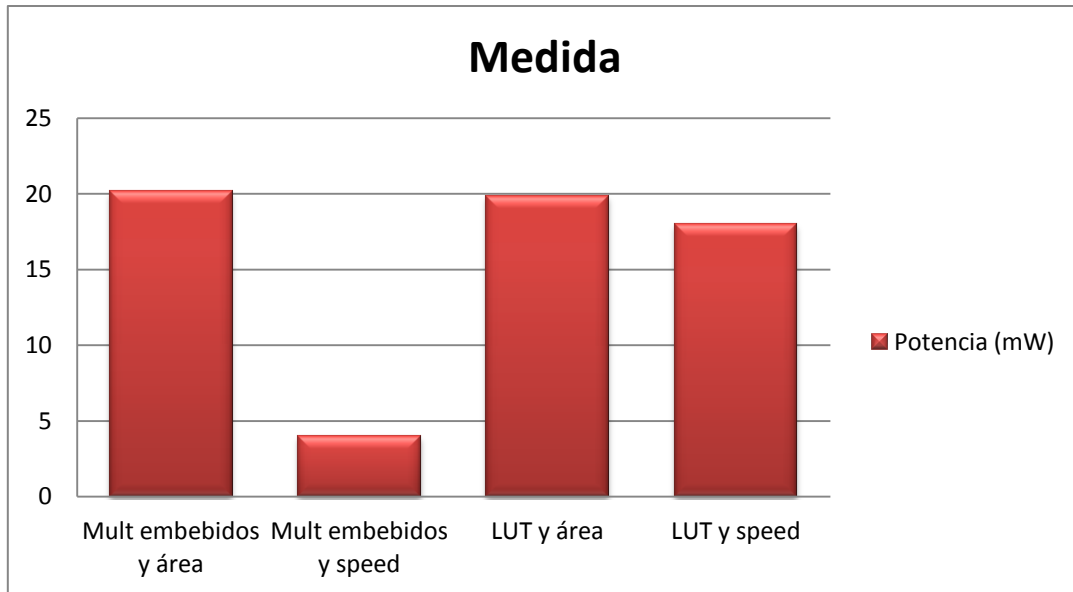


Figura 6.69. Comparativa de las medidas obtenidas midiendo sobre el circuito

6.2. Multiplicador segmentado

Las medidas para el multiplicador segmentado se han dividido de la siguiente manera:

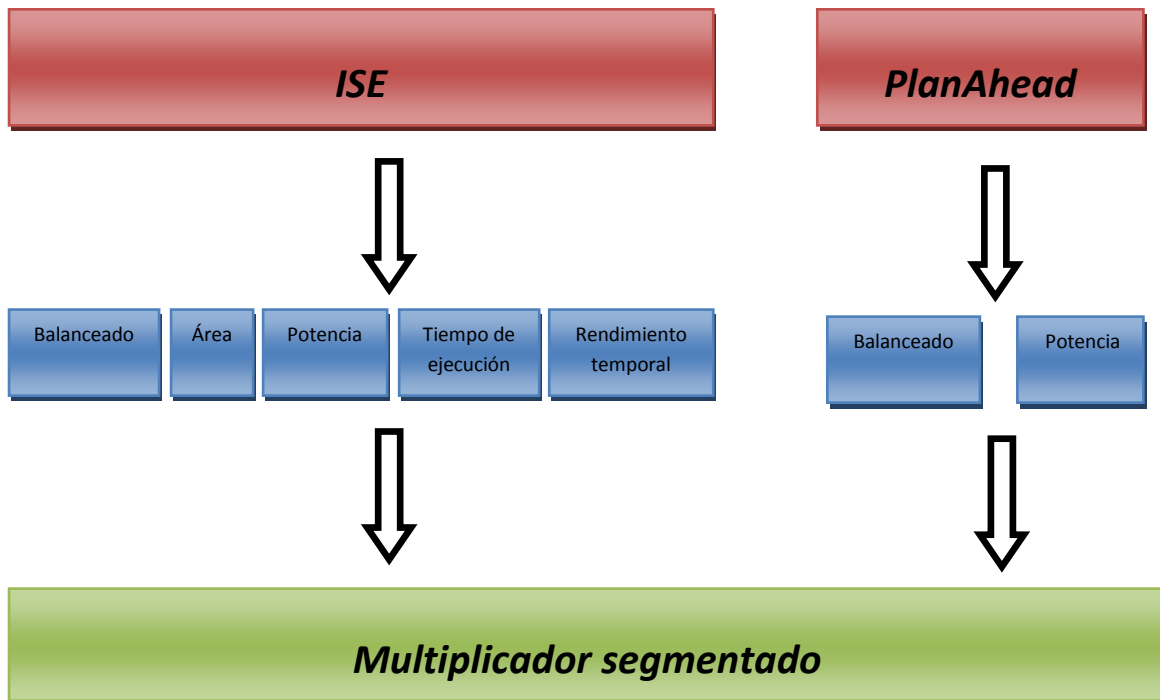


Figura 6.70. Esquema de trabajo para el multiplicador segmentado

El diseño del multiplicador se ha hecho genérico y por ello hay elementos de señales vectoriales que no se usan, como por ejemplo algunos elementos del vector *prod_rg* y *fact2_rg*, hecho así para simplificar la comprensión del código. Debido a esto, durante la sintetización del código aparecen avisos de señales no usadas. Este hecho hace que el fichero de actividad SAIF en este caso no sea tan fiable como en los otros circuitos. Para el multiplicador segmentado, el porcentaje de emparejamientos entre las redes con actividad, según el fichero SAIF y de estas, cuales se encontraban en el diseño inicial es del 81%, mientras que en el resto de circuitos este porcentaje se sitúa entre el 91 y 98%. Esto dará lugar a que las estimaciones del *XPower Analyzer* no tengan la misma relación, con respecto a las medidas sobre la FPGA que se dan en el resto de benchmarks.

6.2.1. ISE

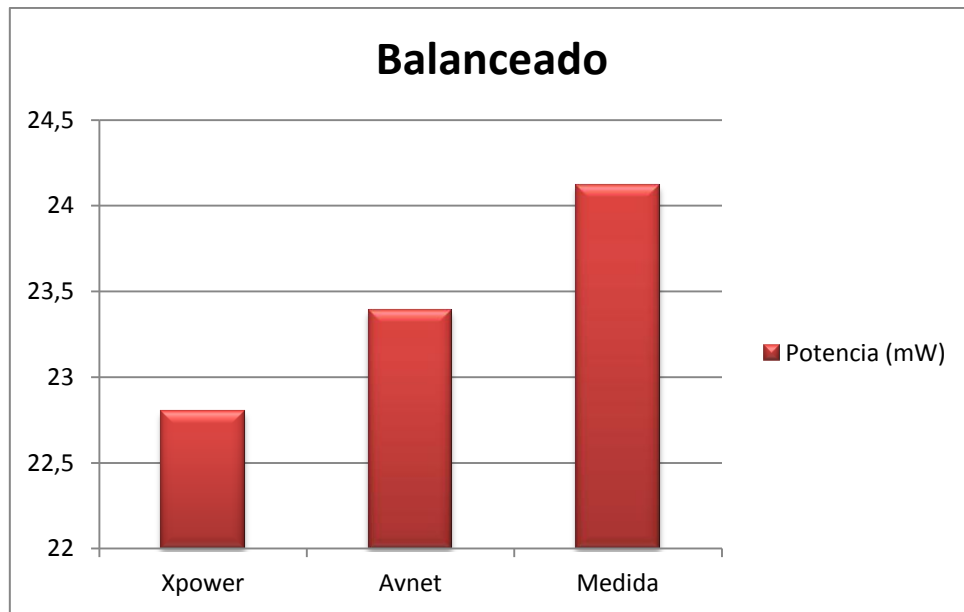


Figura 6.71. Medidas usando ISE y estrategia de diseño balanceada

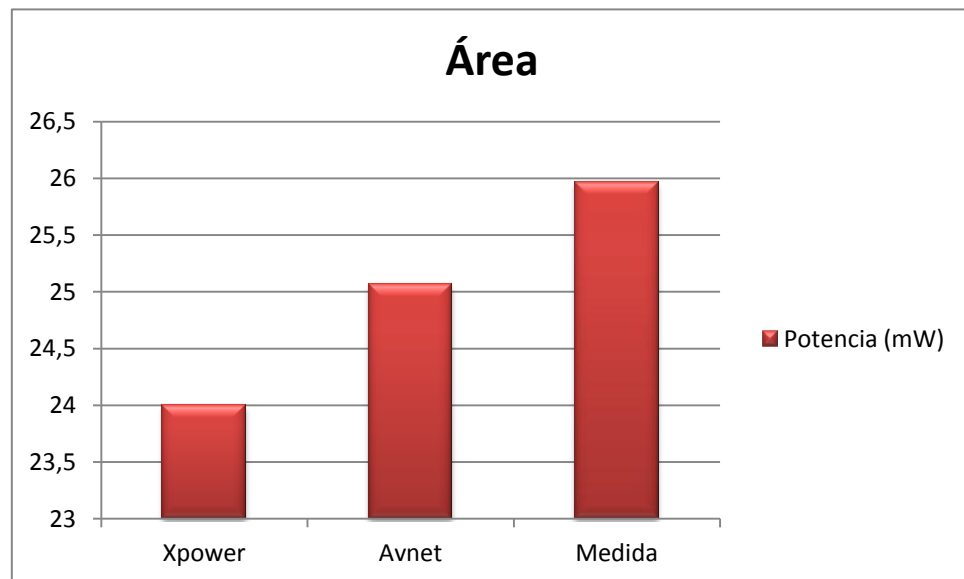


Figura 6.72. Medidas usando ISE y estrategia de diseño de optimización de área

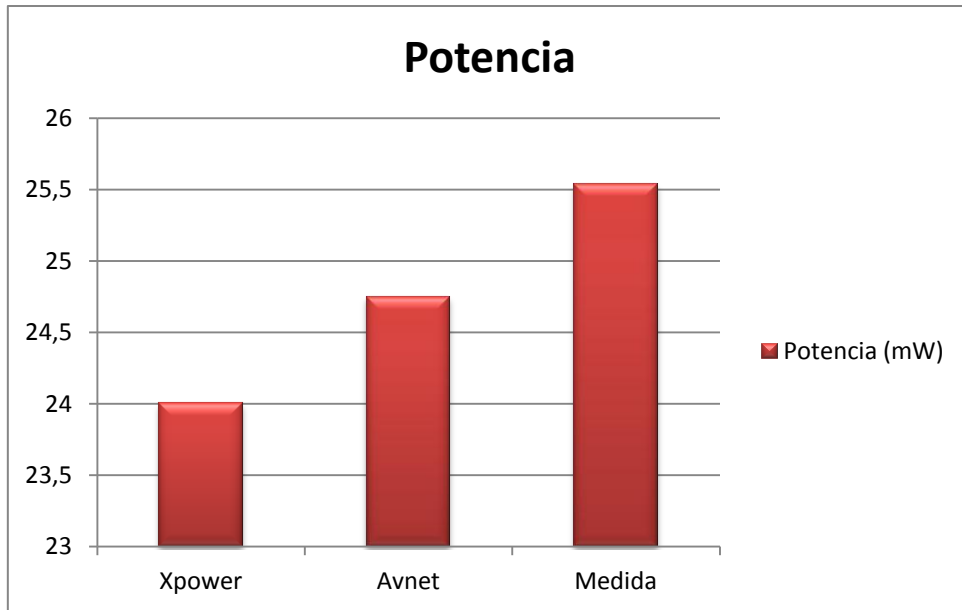


Figura 6.73. Medidas usando ISE y estrategia de diseño de optimización de potencia

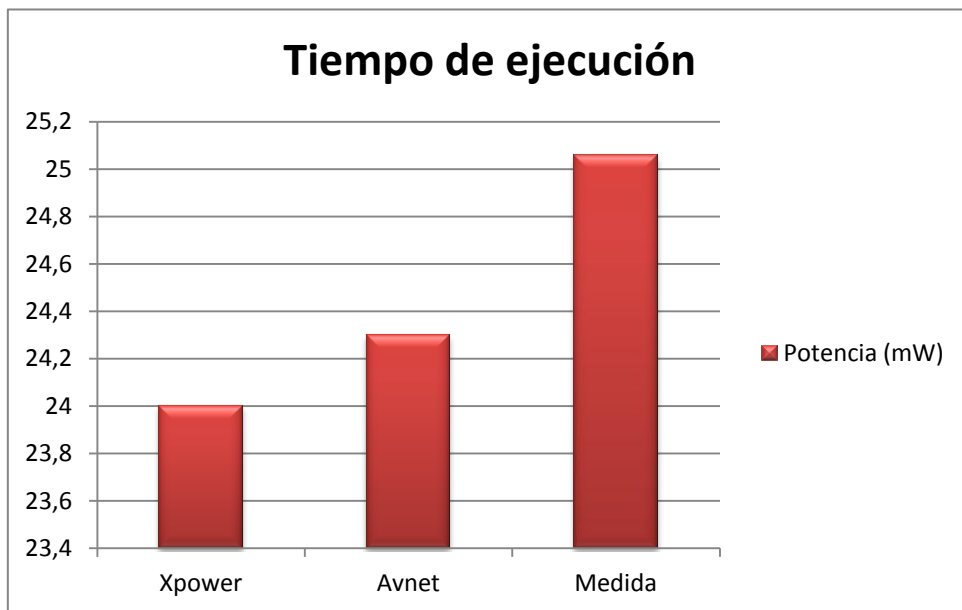


Figura 6.74. Medidas usando ISE y estrategia de diseño de optimización de tiempo de ejecución

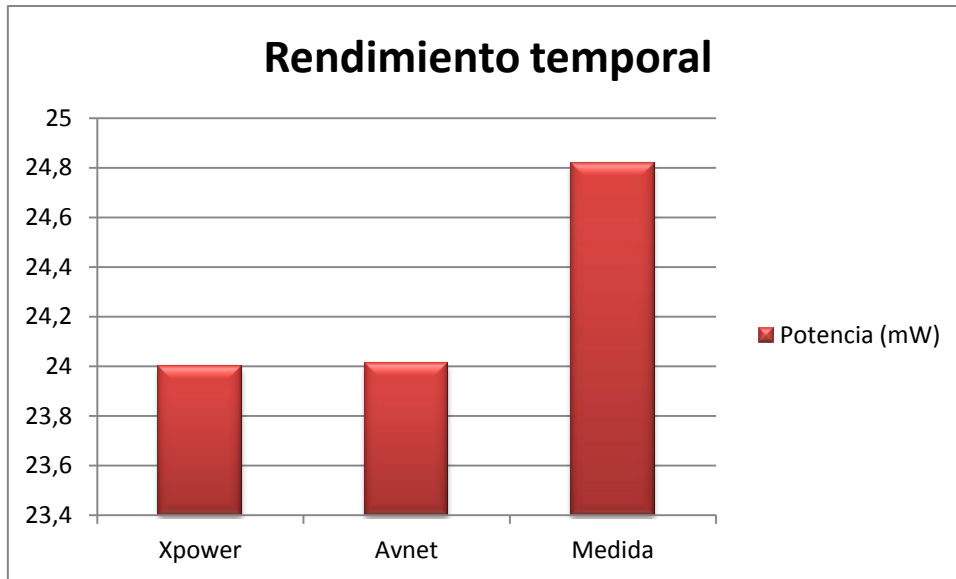


Figura 6.75. Medidas usando ISE y estrategia de diseño de optimizar el rendimiento temporal

6.2.2. PlanAhead

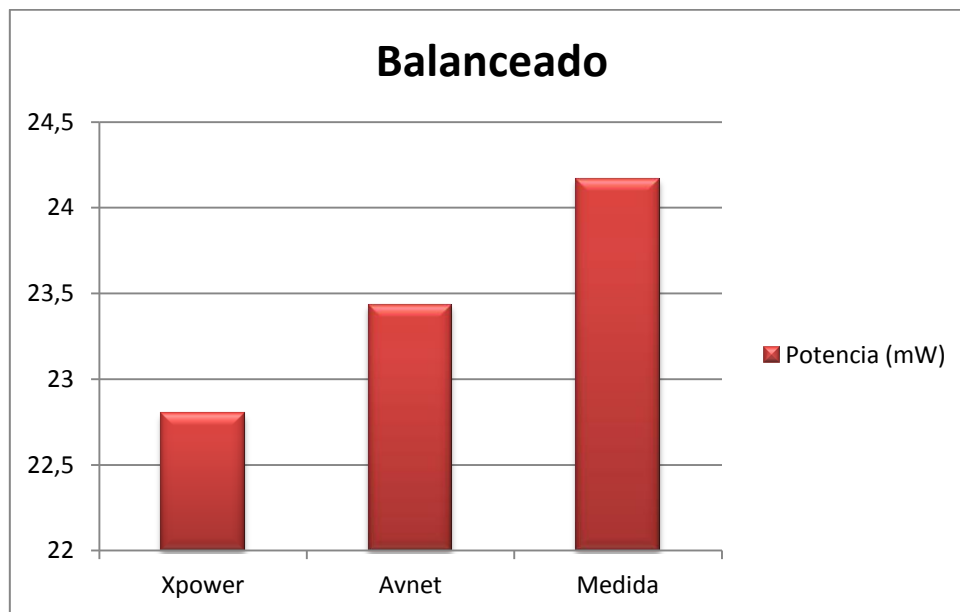


Figura 6.76. Medida usando PlanAhead y estrategia de diseño balanceada

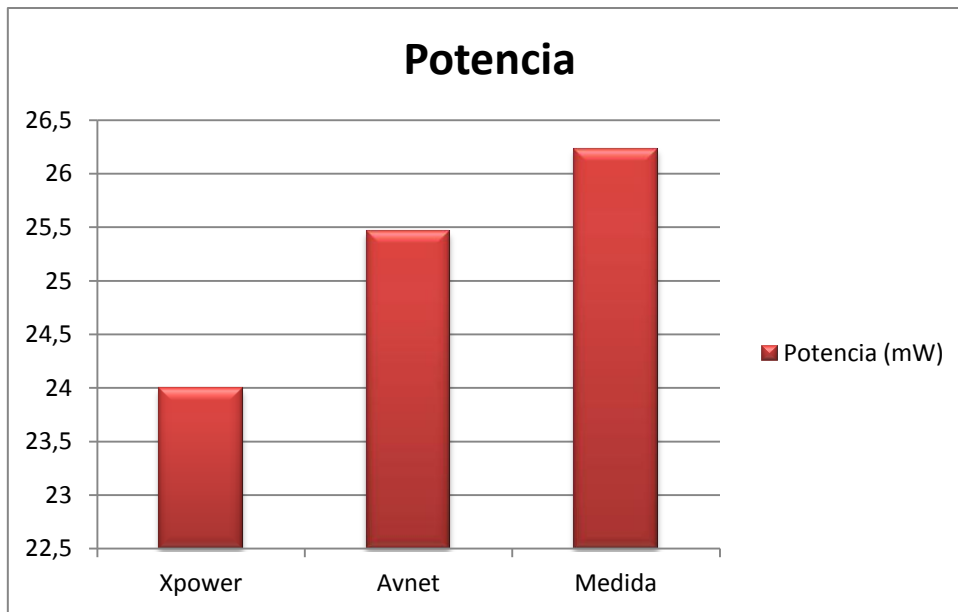


Figura 6.77. Medidas usando PlanAhead y estrategia de diseño de optimización de potencia

6.3. AES (Advanced Encryption Standard)

Las medidas para el circuito de AES se han dividido de la siguiente manera:

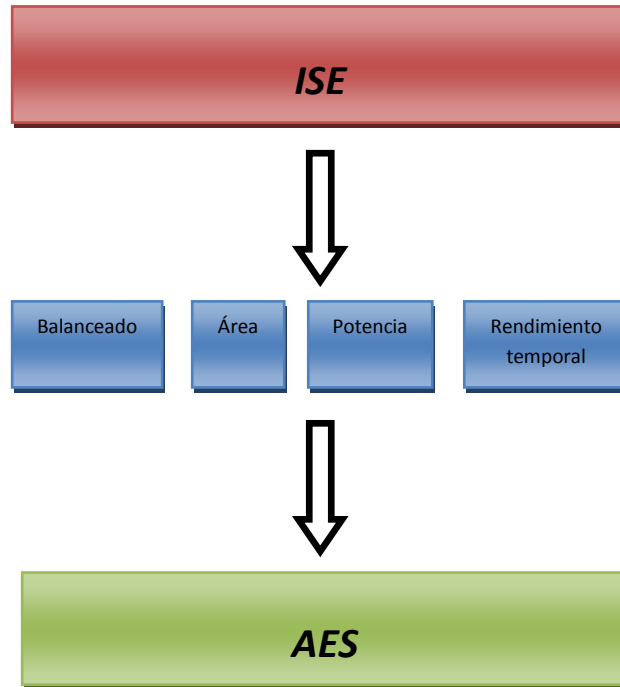


Figura 6.78. Esquema de trabajo para AES

En este apartado solo se han realizado medidas mediante Avnet y estimaciones con *XPower Analyzer*. Debido a la similitud en la forma de obtener los datos entre Avnet y la medida directa sobre la FPGA con un multímetro de precisión, no se ha realizado este último tipo de medida para el circuito AES.

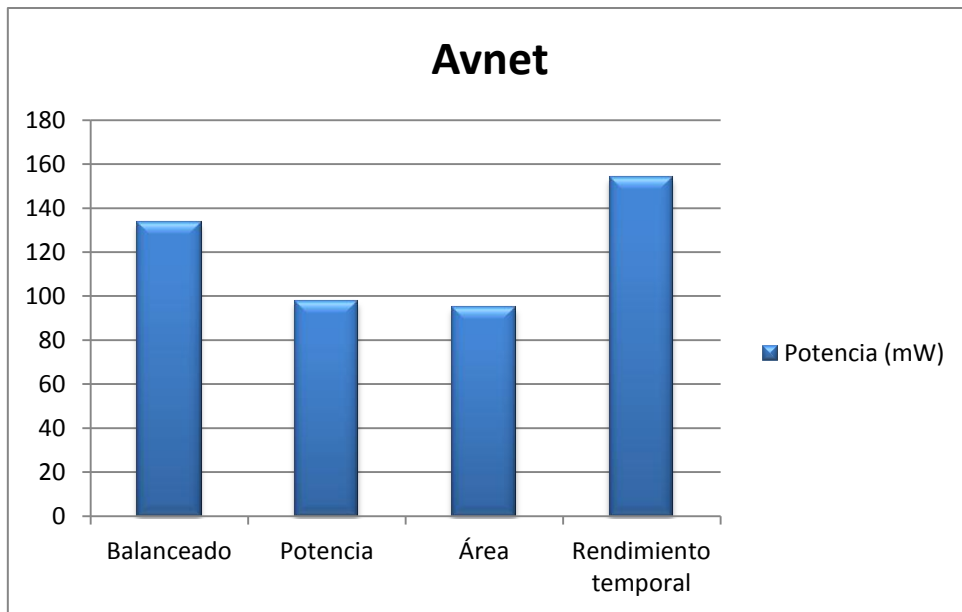


Figura 6.79. Medidas obtenidas con Avnet

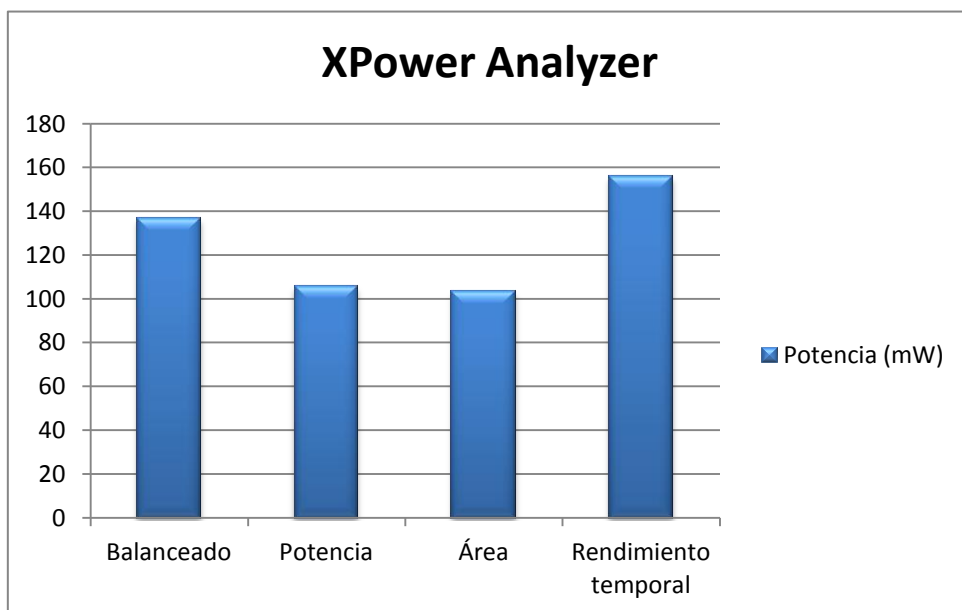


Figura 6.80. Medidas obtenidas con XPower Analyzer

6.4. FFT (Fast Fourier Transform)

Las medidas para el circuito de FFT se han dividido de la siguiente manera:

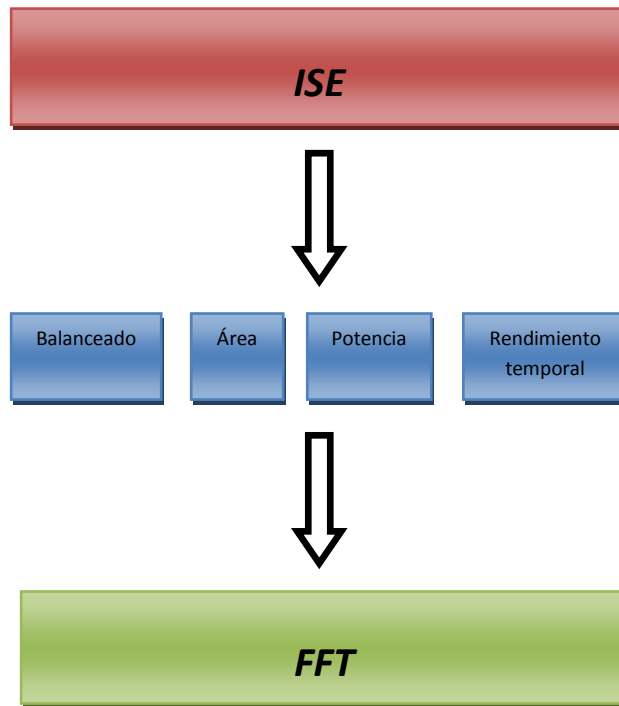


Figura 6.81. Esquema de trabajo para FFT

Al igual que para el circuito AES solo se han realizado medidas mediante *Avnet* y estimaciones con *XPower Analyzer*. Debido a la similitud en la forma de obtener los datos entre *Avnet* y la medida directa sobre la FPGA con un multímetro de precisión, no se ha realizado este último tipo de medida para el circuito FFT.

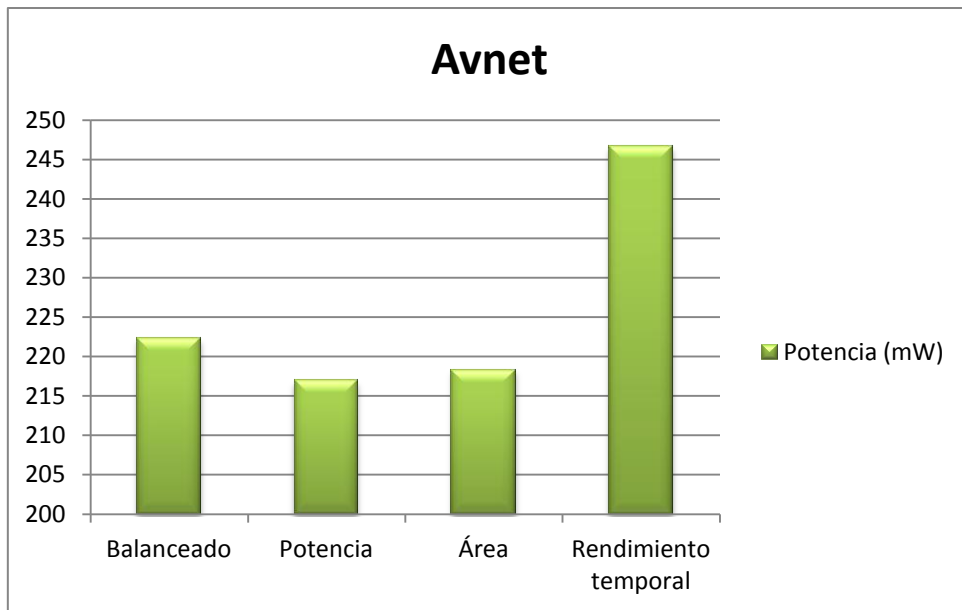


Figura 6.82. Medidas obtenidas con Avnet

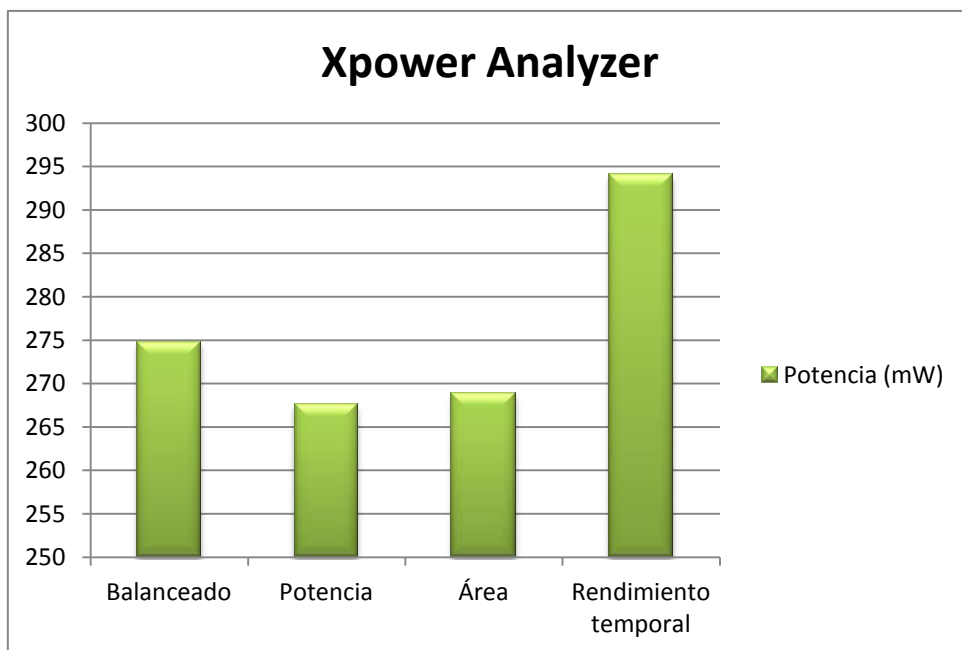


Figura 6.83. Medidas obtenidas con XPower Analyzer

6.5. Floorplanning

Las modificaciones de *floorplanning* se han realizado sobre los multiplicadores Xilinx y el multiplicador segmentado y sobre dos estrategias de diseños, balanceado y optimización de potencia. Para cada uno de estas se ha realizado cambios en el circuito, se ha medido los consumos de potencia. Las modificaciones de *floorplanning* se han hecho usando *PlanAhead*. No se han realizado simulaciones de consumo para los multiplicadores de Xilinx.

La principal técnica aplicada durante las modificaciones de *floorplanning* ha sido la de localidad espacial entre los elementos que se usan con más frecuencia. Es decir, organizar los elementos dependiendo de cómo se comunican unos con otros, analizando el camino crítico e intentando que todos los elementos de este camino estén lo más cerca posible unos de otros. Para estos elementos existe una alta probabilidad de que para otros caminos distintos vayan a ser referenciados entre sí.

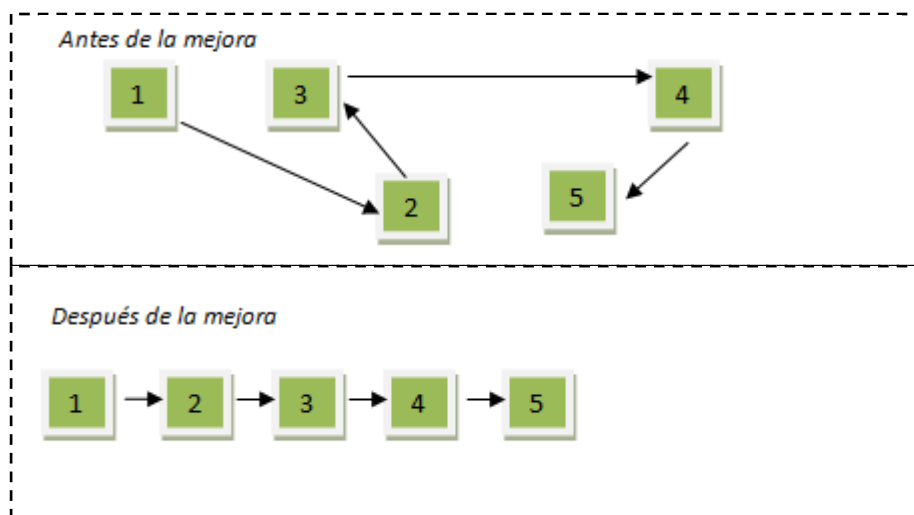


Figura 6.84. Ejemplo de las modificaciones realizadas mediante floorplanning

6.5.1 Multiplicadores Xilinx

6.5.1.1 Balanceado

Se mostrarán los resultados para los distintos multiplicadores, acompañados de su *floorplanning*. En cada *floorplanning* modificado se muestran los elementos cambiados en color naranja.

En el figura 6.84 se muestra un ejemplo para representar la idea de cómo se han realizado las modificaciones de *floorplanning*. En los circuitos usados en este proyecto este proceso es costoso debido a la gran cantidad de elementos que tienen.

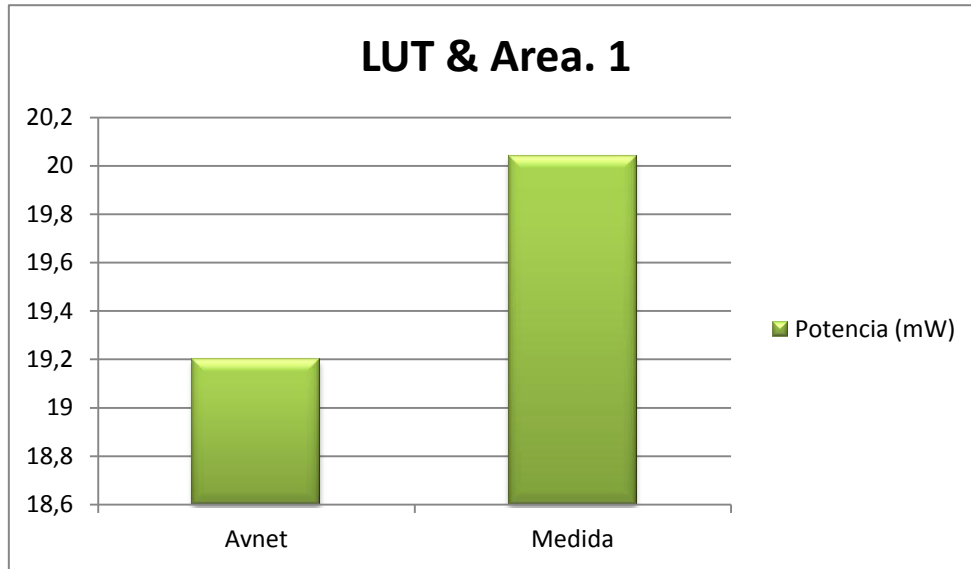


Figura 6.85. Medidas usando PlanAhead y sin aplicar modificaciones

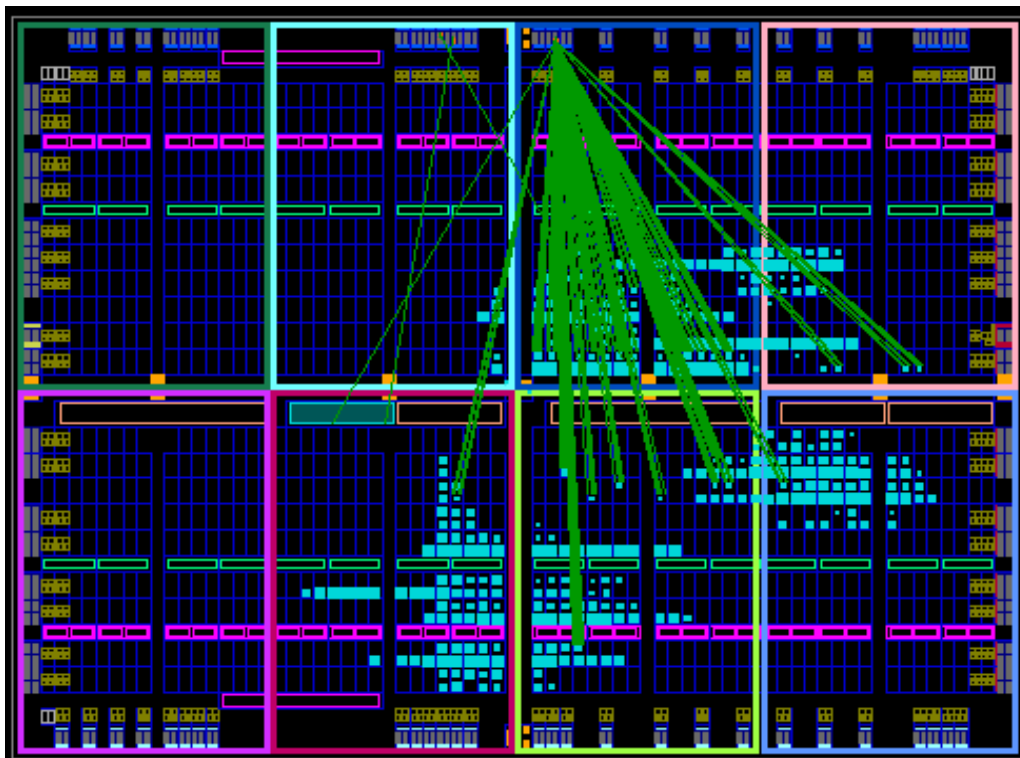


Figura 6.86. Esquema del circuito usando PlanAhead y sin aplicar modificaciones

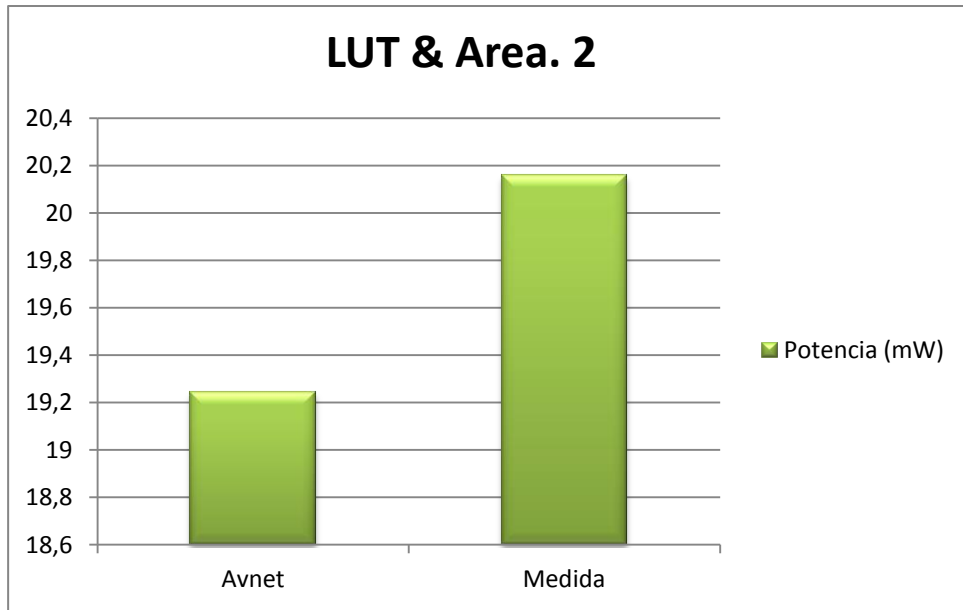


Figura 6.87. Medidas usando PlanAhead y aplicando modificaciones

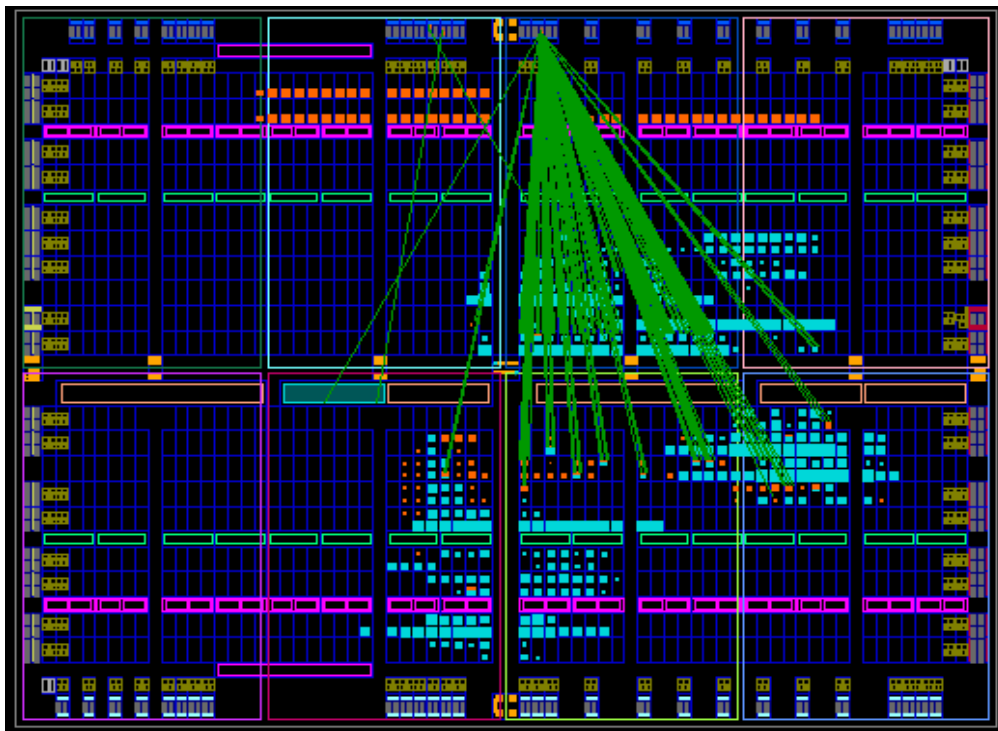


Figura 6.88. Esquema de circuito usando PlanAhead y aplicando modificaciones

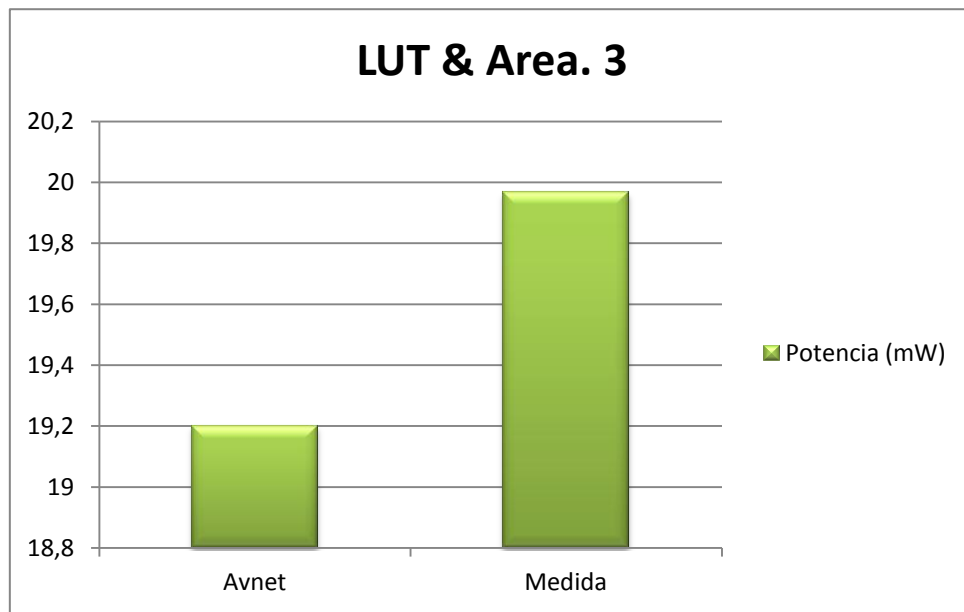


Figura 6.89. Medidas usando PlanAhead y aplicando modificaciones

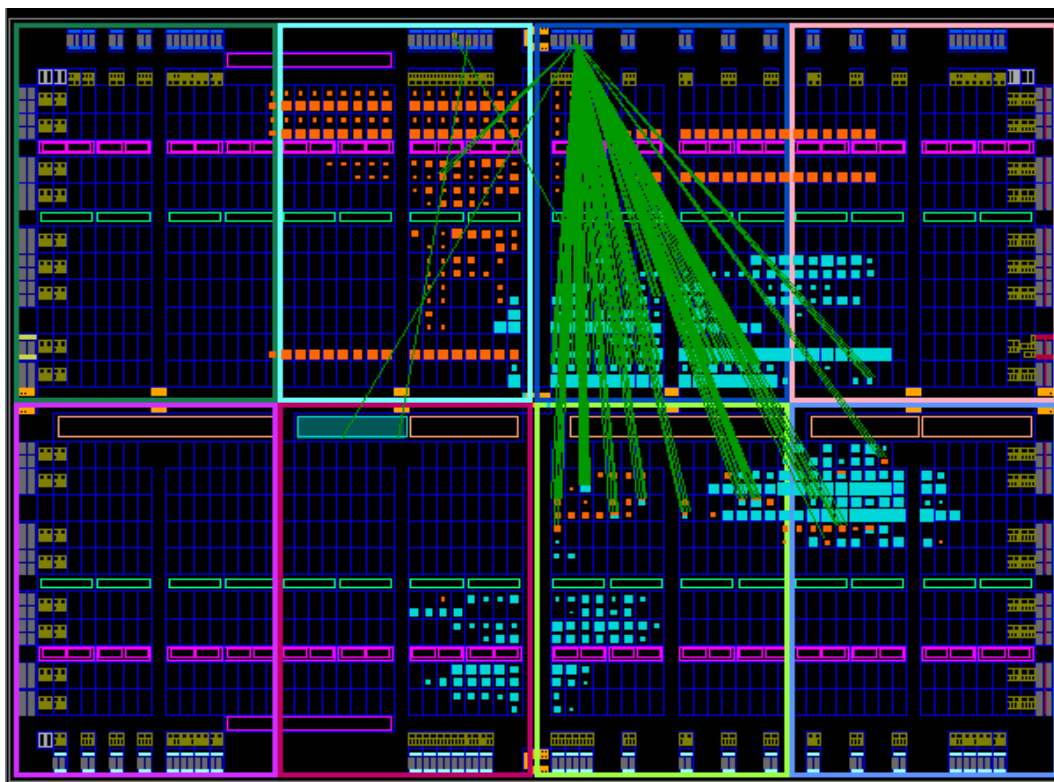


Figura 6.90. Esquema del circuito usando PlanAhead y aplicando modificaciones

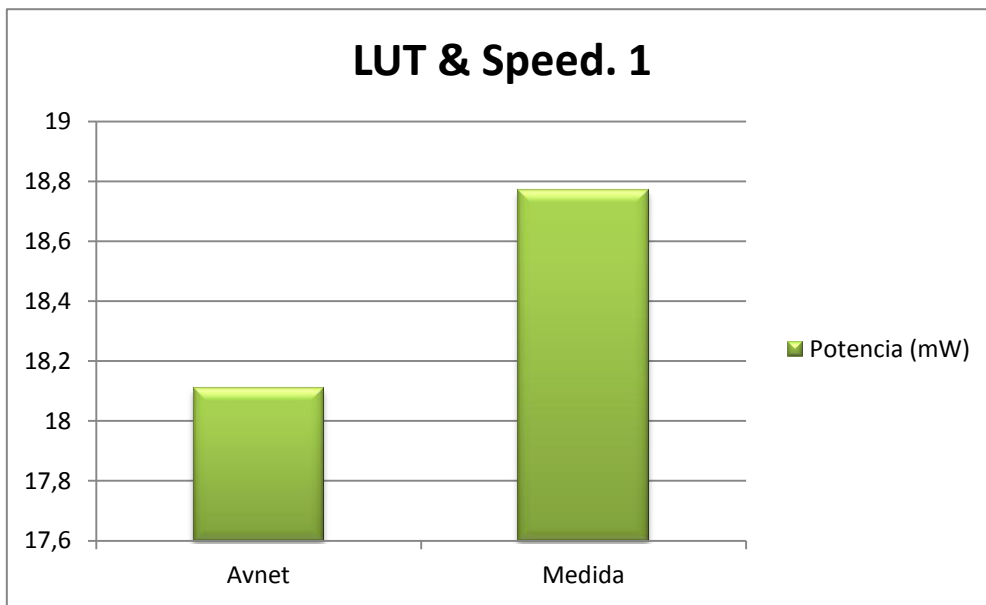


Figura 6.91. Medidas usando PlanAhead y sin aplicar modificaciones

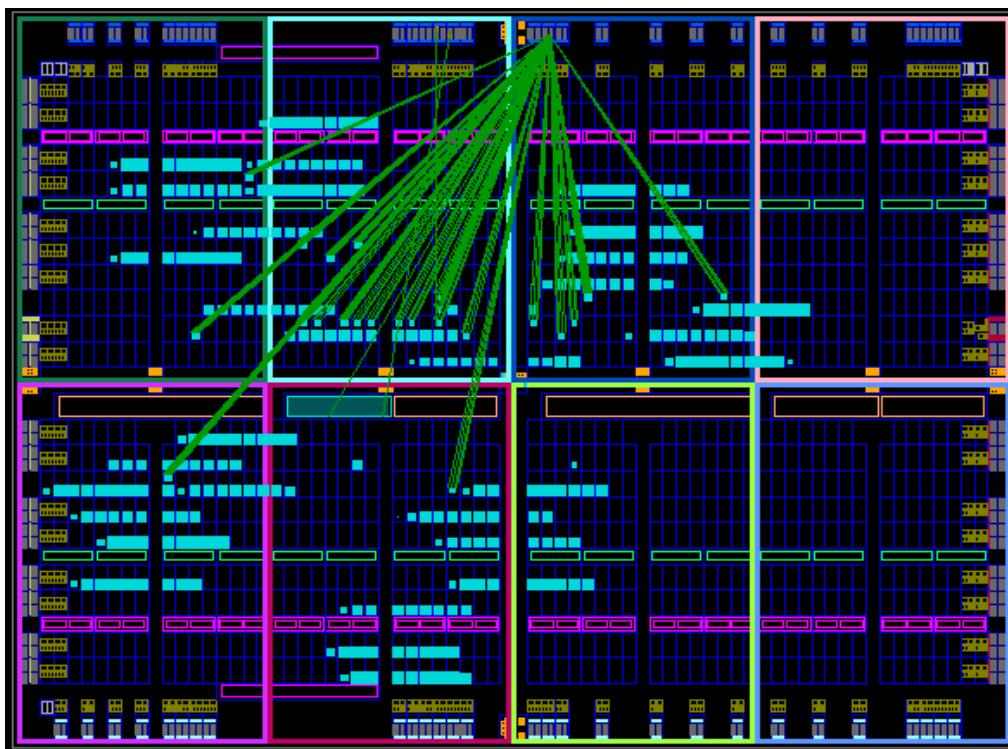


Figura 6.92. Esquema de circuito usando PlanAhead y sin aplicar modificaciones

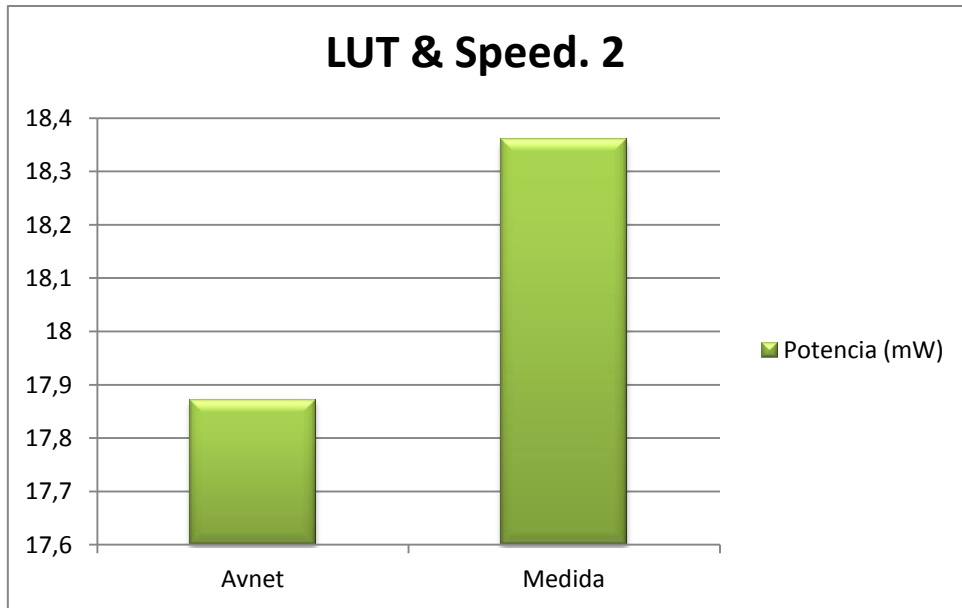


Figura 6.93. Medidas usando PlanAhead y aplicando modificaciones

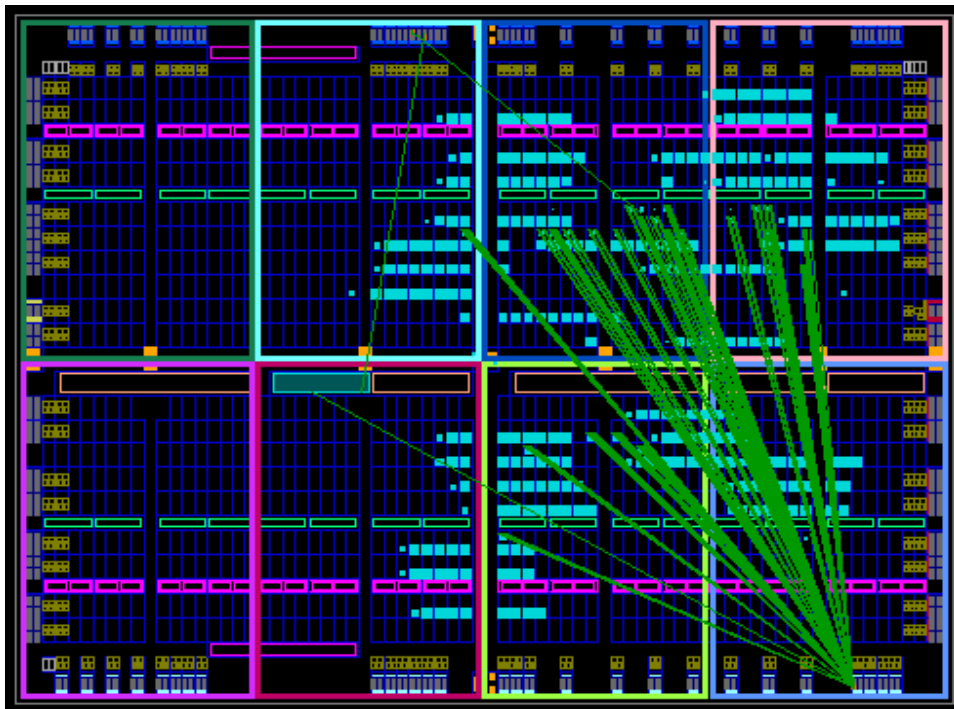


Figura 6.94. Esquema del circuito usando PlanAhead y aplicando modificaciones

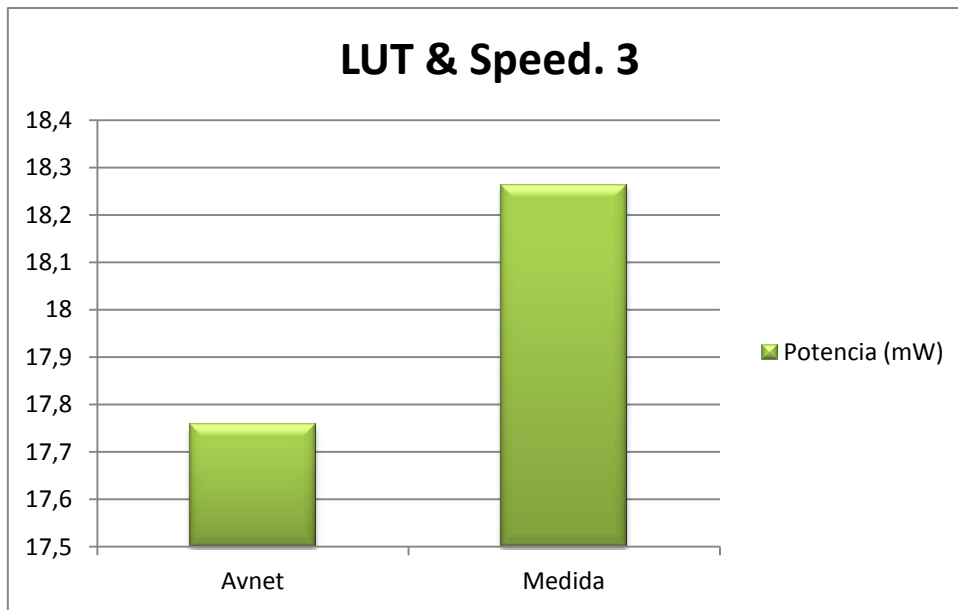


Figura 6.95. Medidas usando PlanAhead y aplicando modificaciones

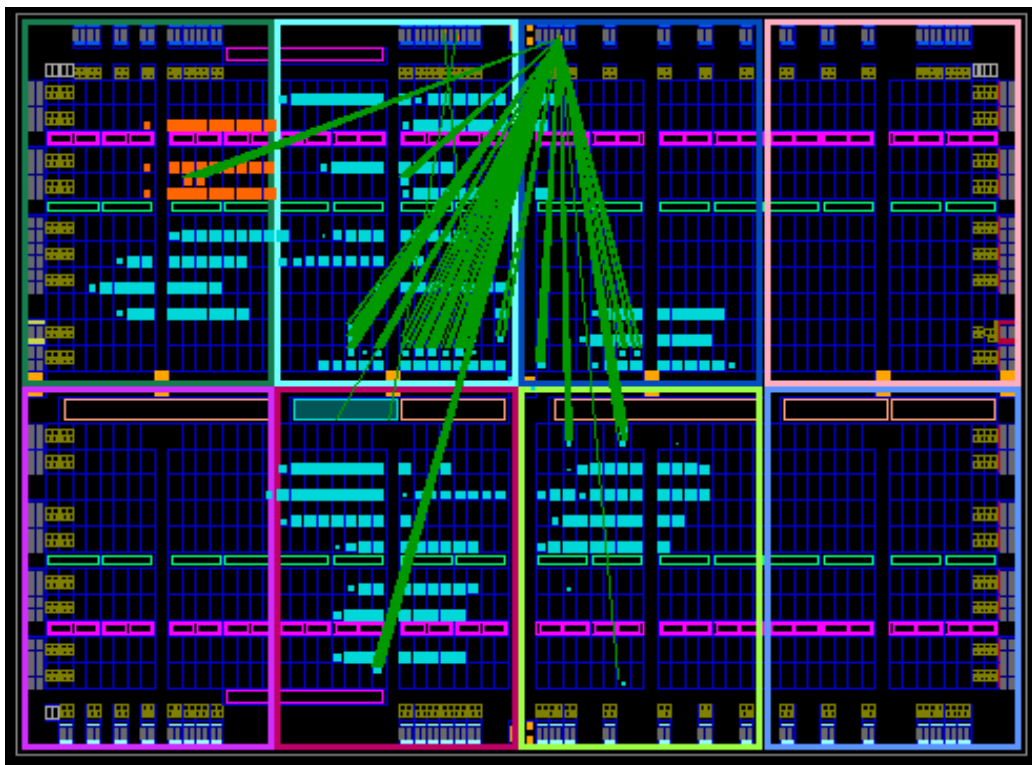


Figura 6.96. Esquema del circuito usando PlanAhead y aplicando modificaciones

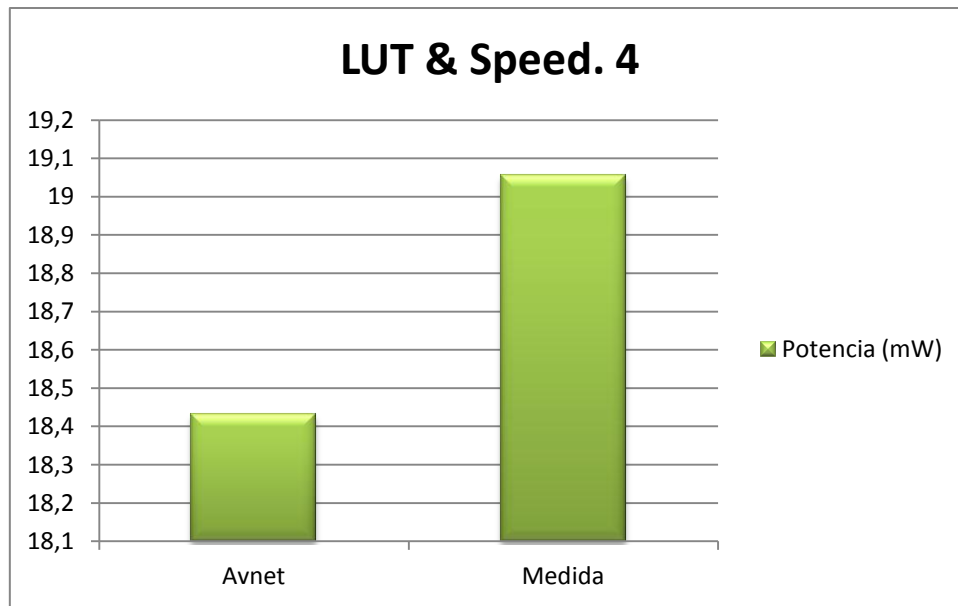


Figura 6.97. Medidas usando PlanAhead y aplicando modificaciones

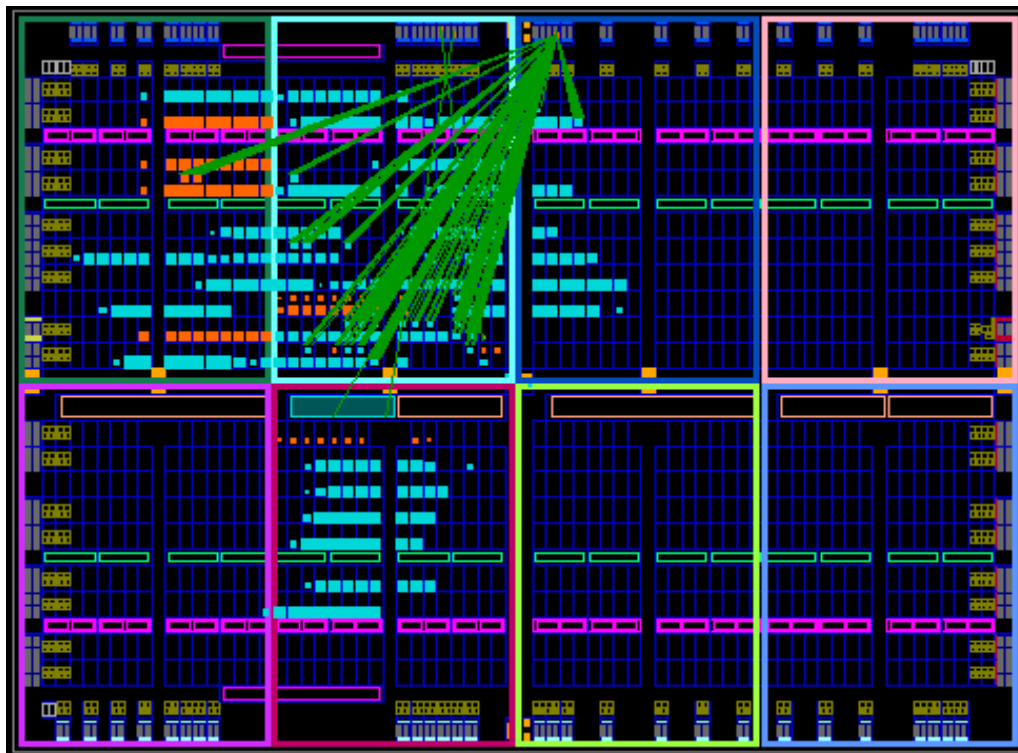


Figura 6.98. Esquema del circuito usando PlanAhead y aplicando modificaciones

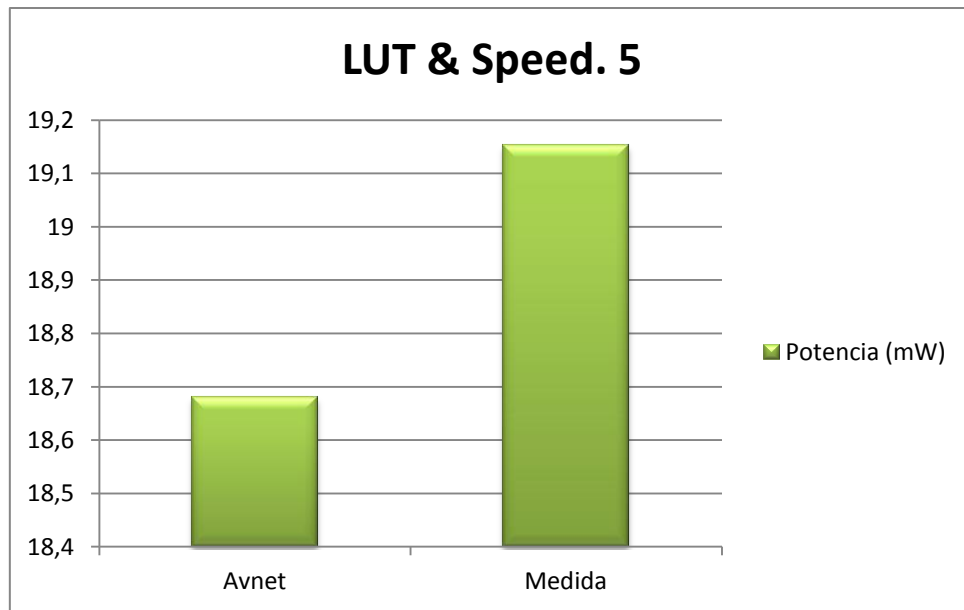


Figura 6.99. Medidas usando PlanAhead y aplicando modificaciones

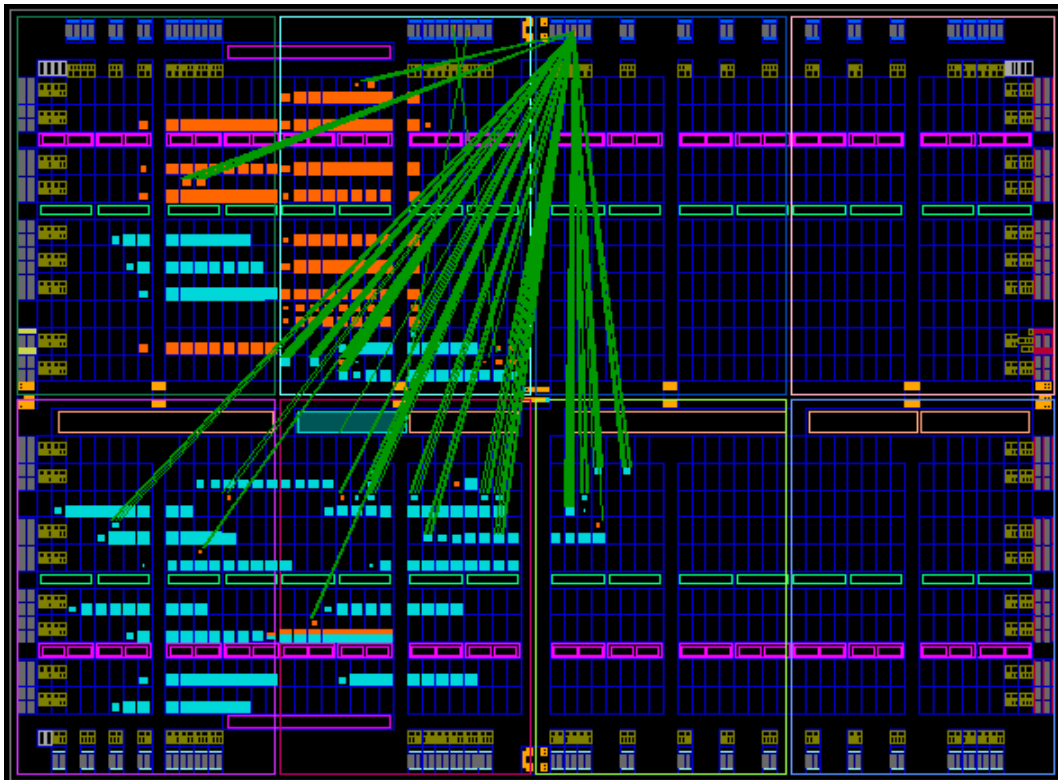


Figura 6.100. Esquema del circuito usando PlanAhead y aplicando modificaciones

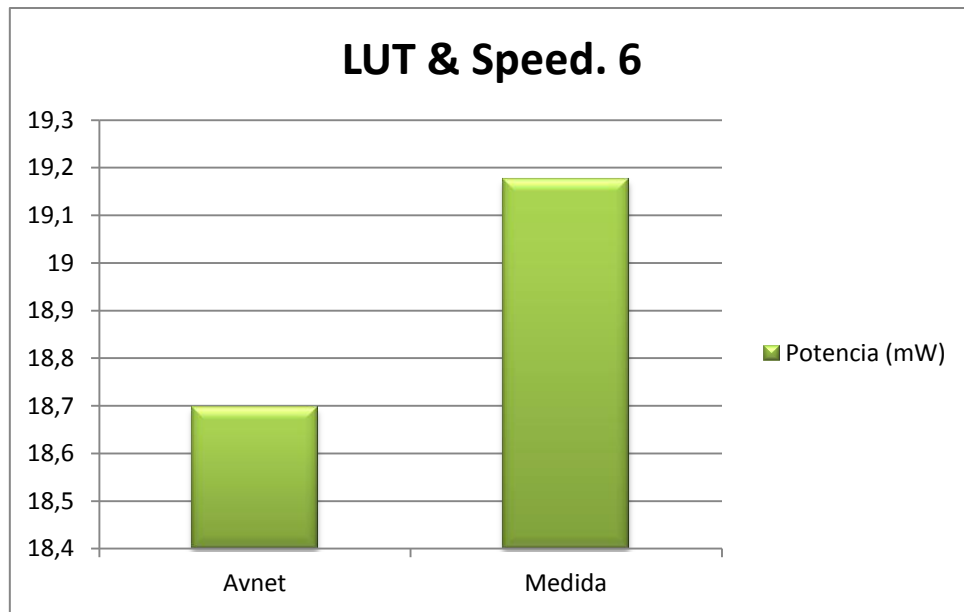


Figura 6.101. Medidas usando PlanAhead y aplicando modificaciones

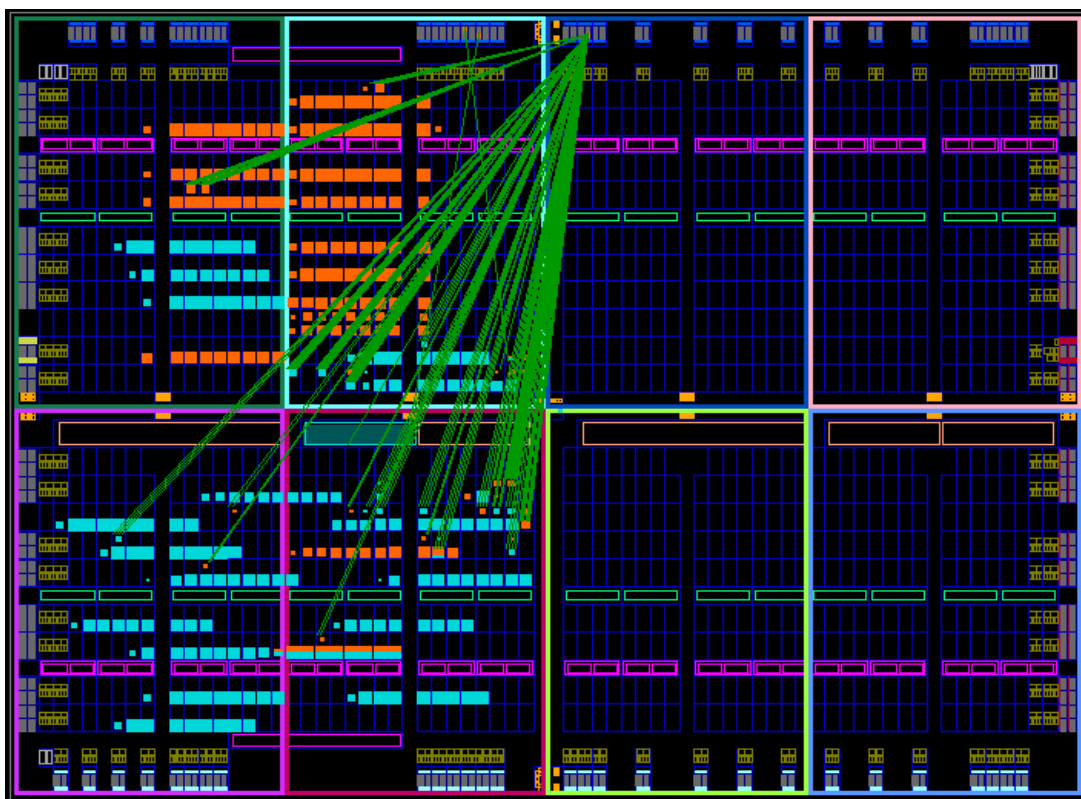


Figura 6.102. Esquema del circuito usando PlanAhead y aplicando modificaciones

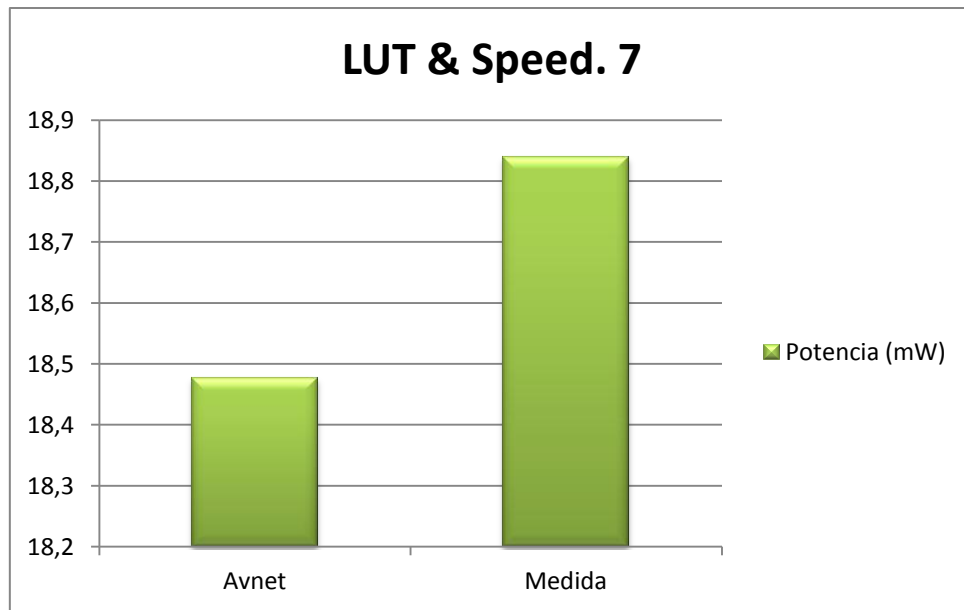


Figura 6.103. Medidas usando PlanAhead y aplicando modificaciones

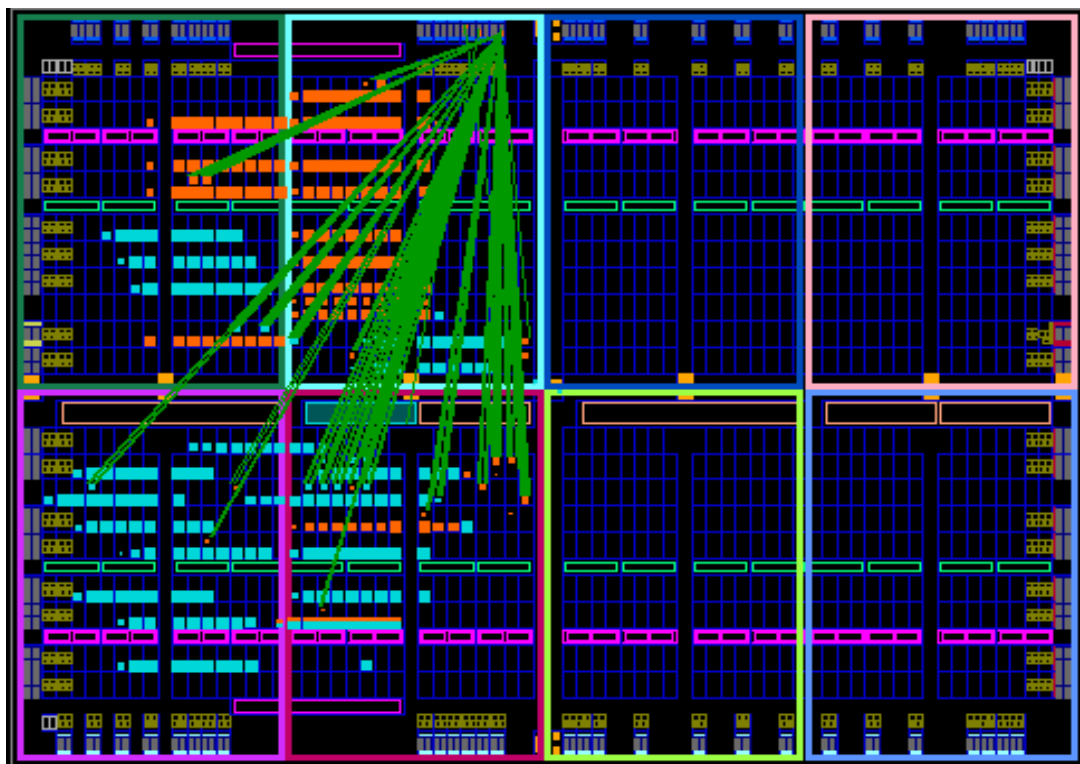


Figura 6.104 Esquema del circuito usando PlanAhead y aplicando modificaciones

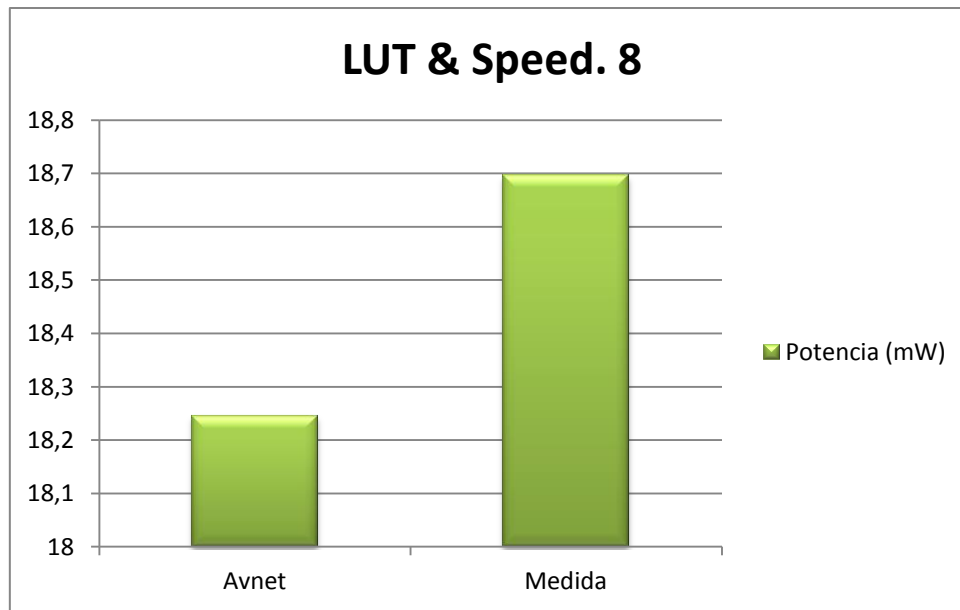


Figura 6.105. Medidas usando PlanAhead y aplicando modificaciones

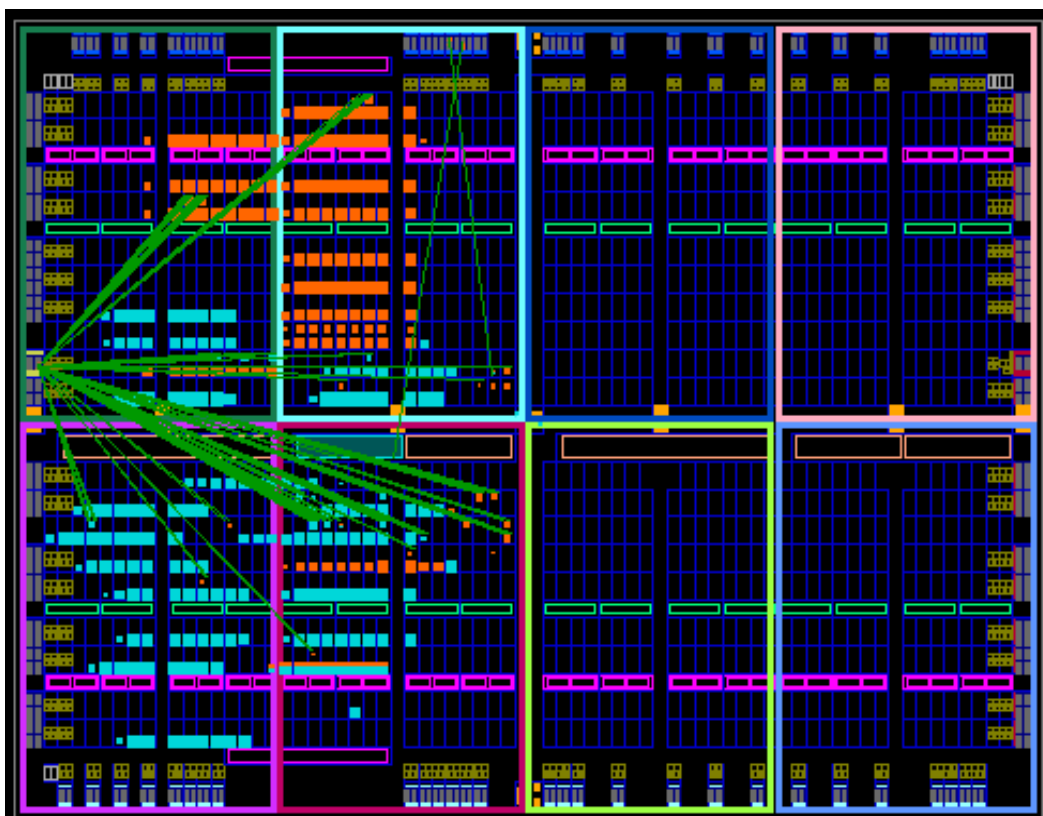


Figura 6.106. Esquema del circuito usando PlanAhead y aplicando modificaciones

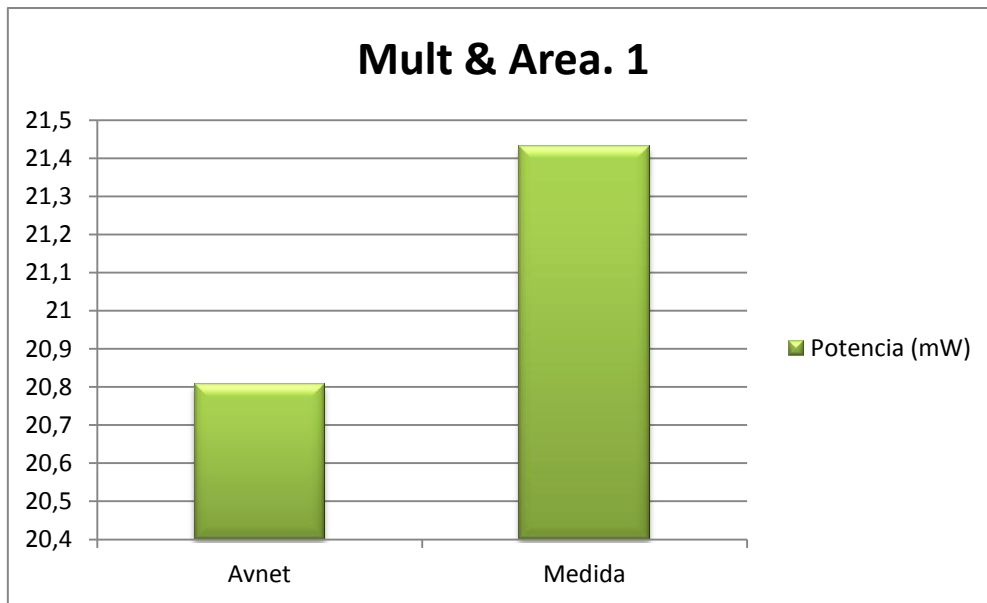


Figura 6.107. Medidas usando PlanAhead y sin aplicar modificaciones

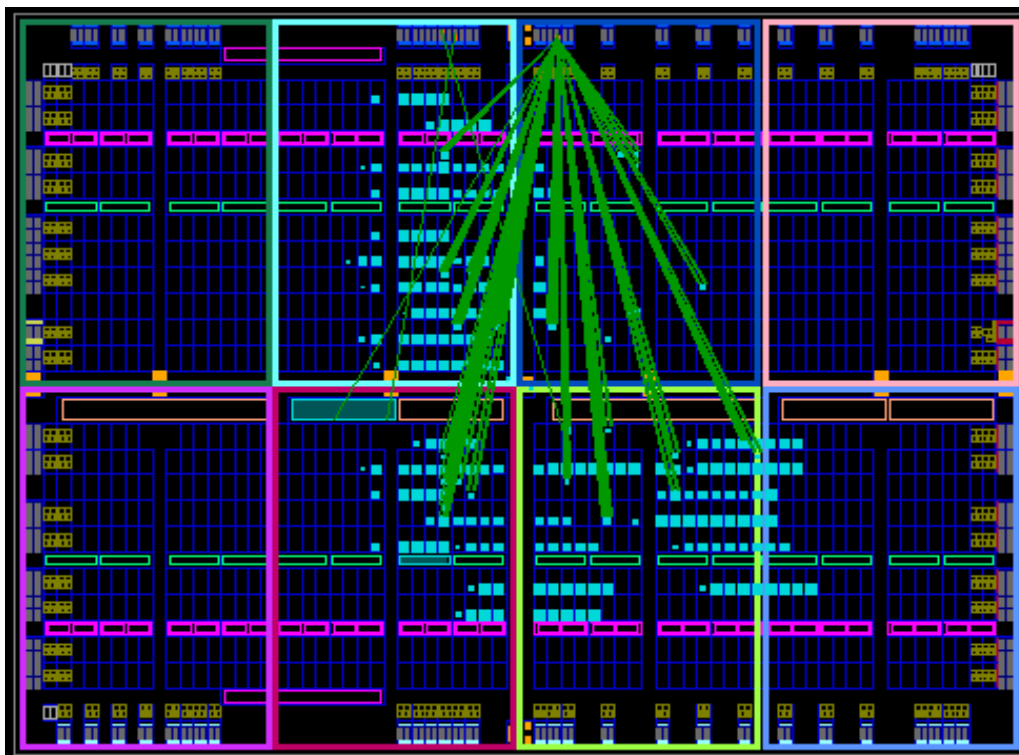


Figura 6.108. Esquema del circuito usando PlanAhead y sin aplicar modificaciones

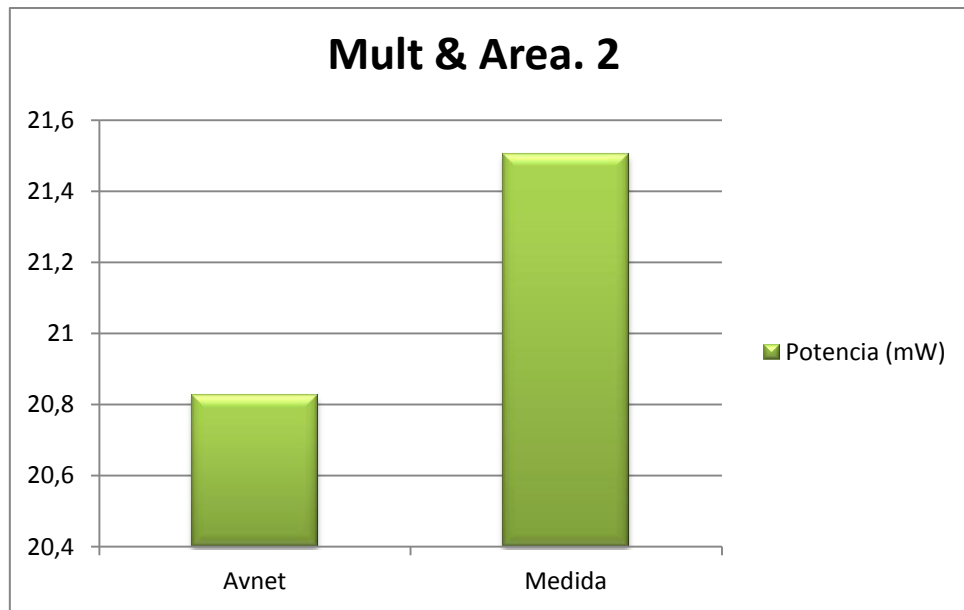


Figura 6.109. Medidas usando PlanAhead y aplicando modificaciones

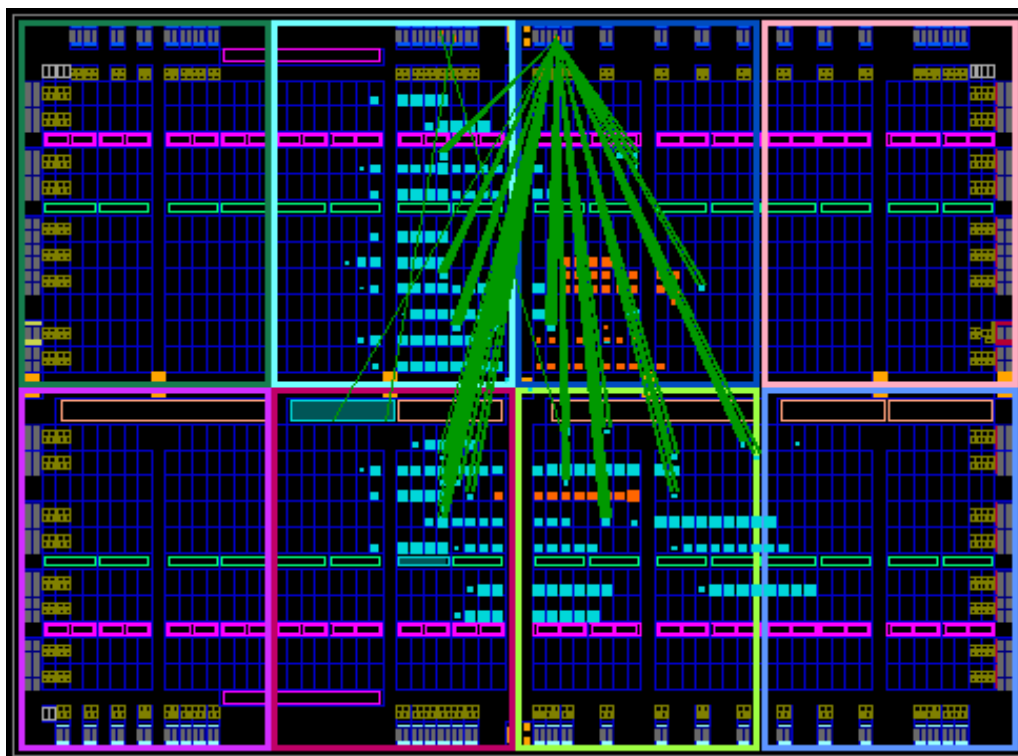


Figura 6.110. Esquema del circuito usando PlanAhead y aplicando modificaciones

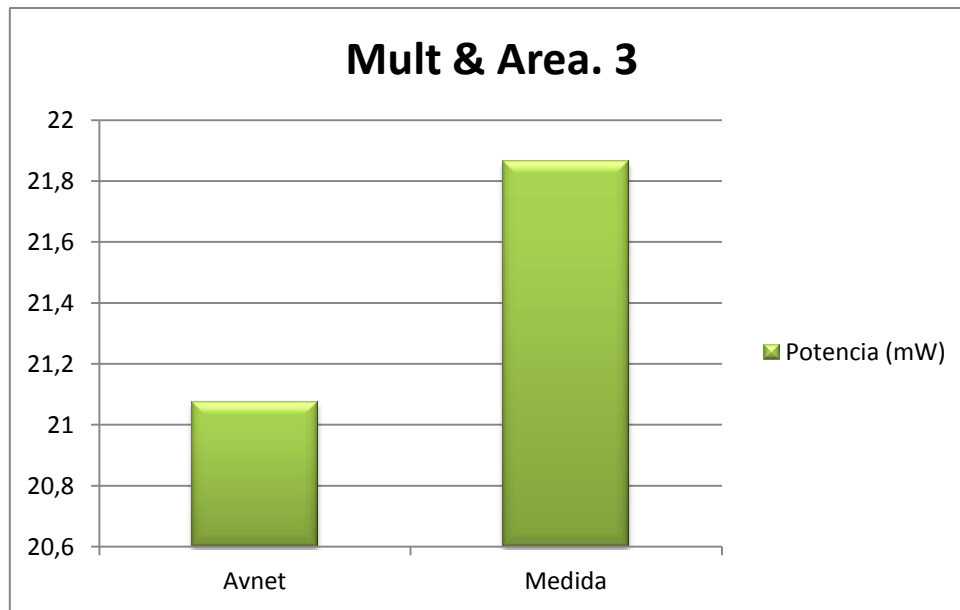


Figura 6.111. Medidas usando PlanAhead y aplicando modificaciones

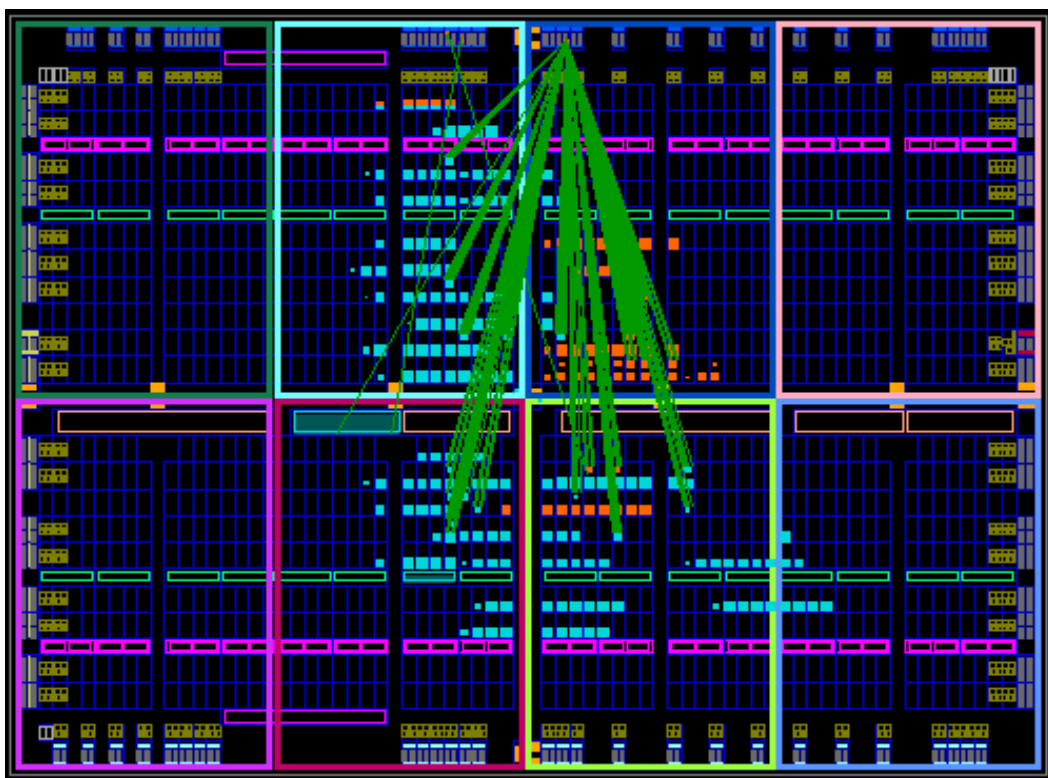


Figura 6.112. Esquema del circuito usando PlanAhead y aplicando modificaciones

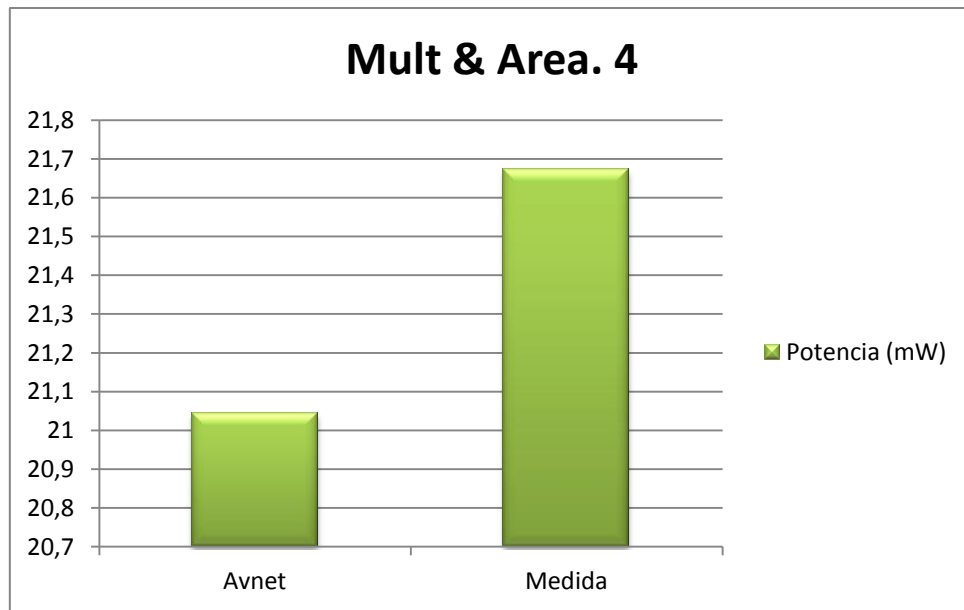


Figura 6.113. Medidas usando PlanAhead y aplicando modificaciones

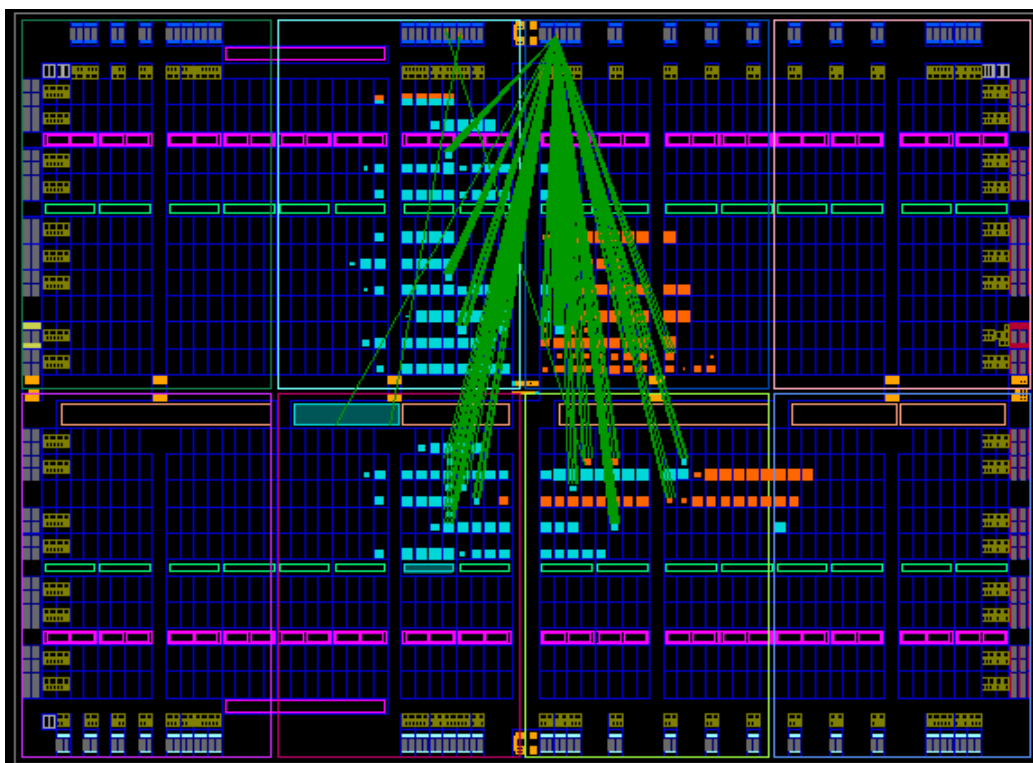


Figura 6.114. Esquema del circuito usando PlanAhead y aplicando modificaciones

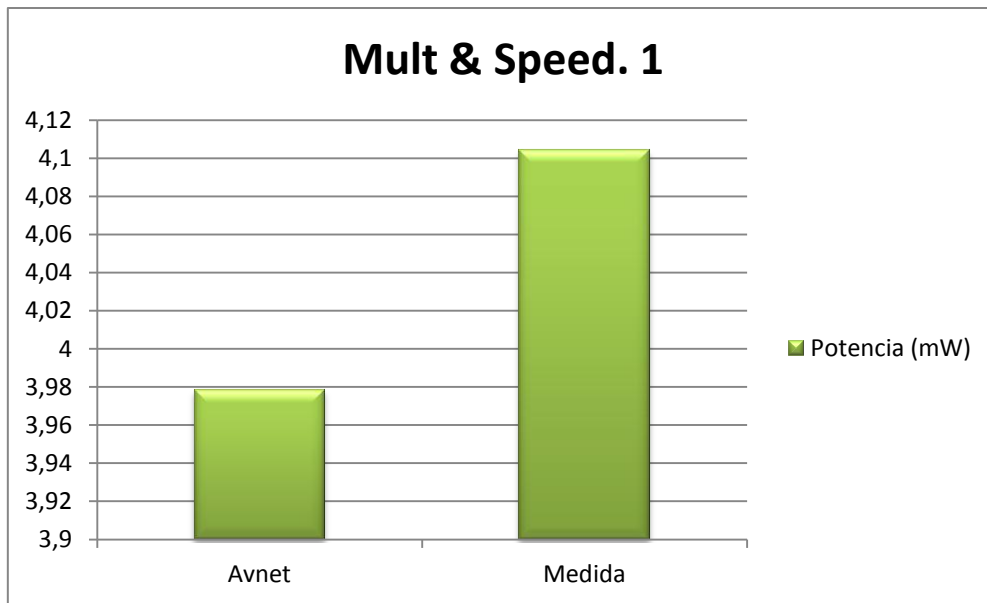


Figura 6.115. Medidas usando PlanAhead y sin aplicar modificaciones

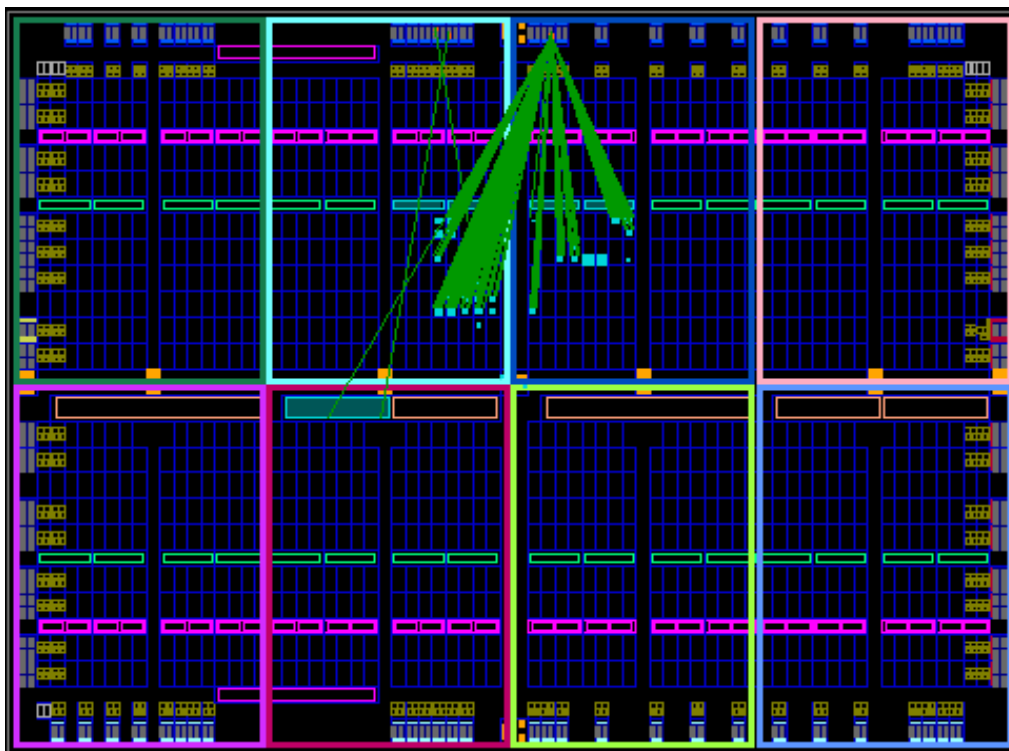


Figura 6.116. Esquema del circuito usando PlanAhead y sin aplicar modificaciones

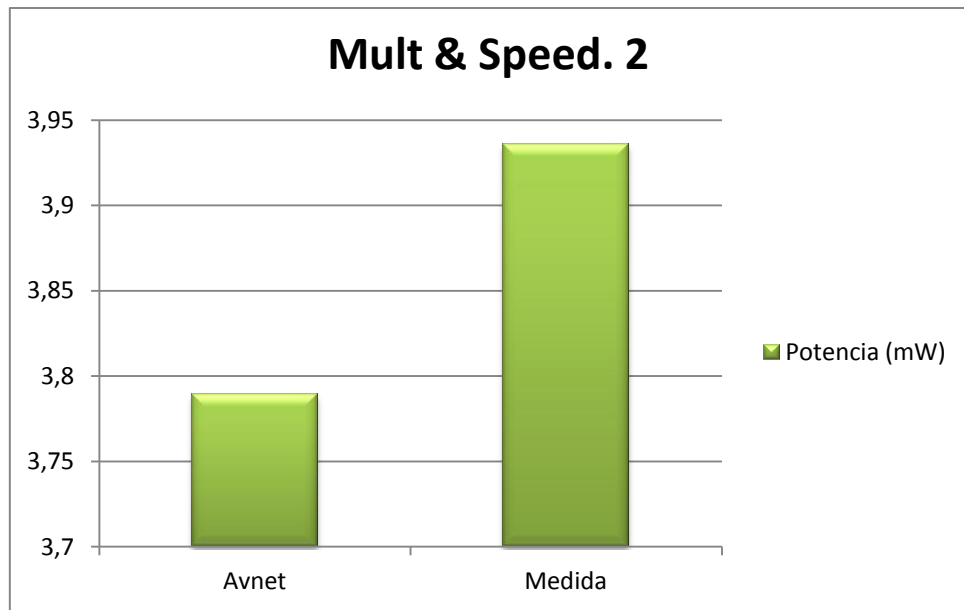


Figura 6.117. Medidas usando PlanAhead y aplicando modificaciones

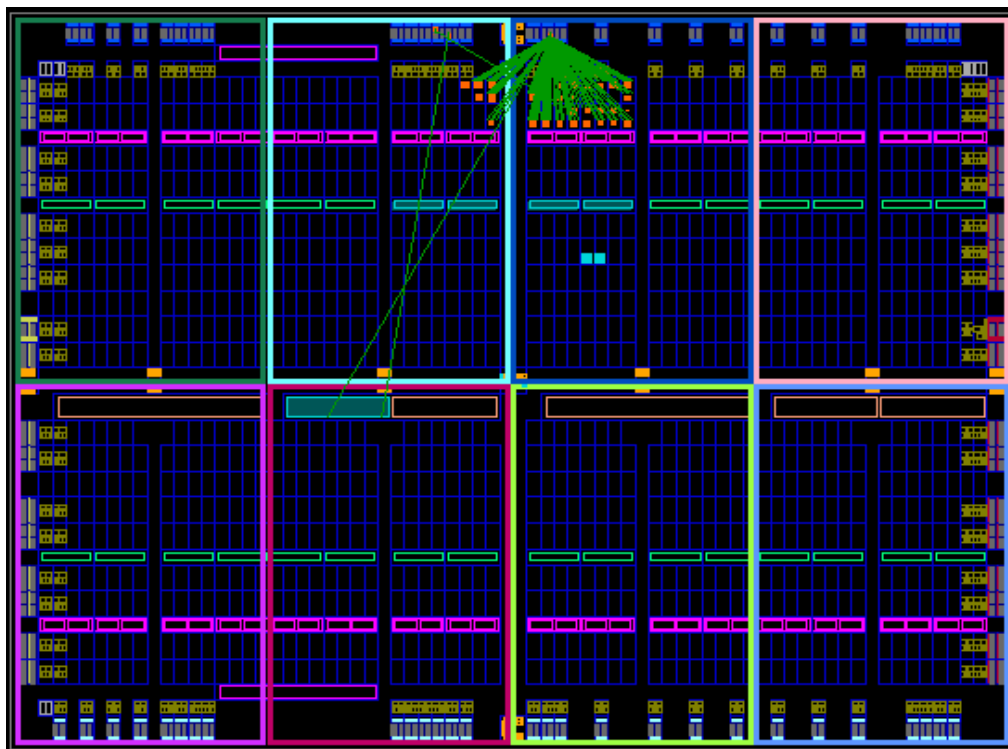


Figura 6.118. Esquema del circuito usando PlanAhead y aplicando modificaciones

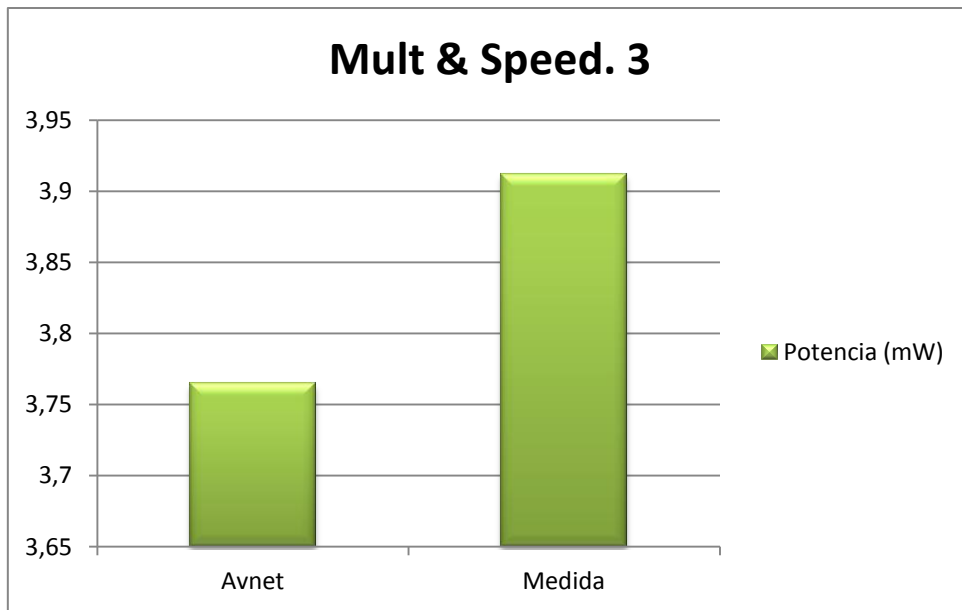


Figura 6.119. Medidas usando PlanAhead y aplicando modificaciones

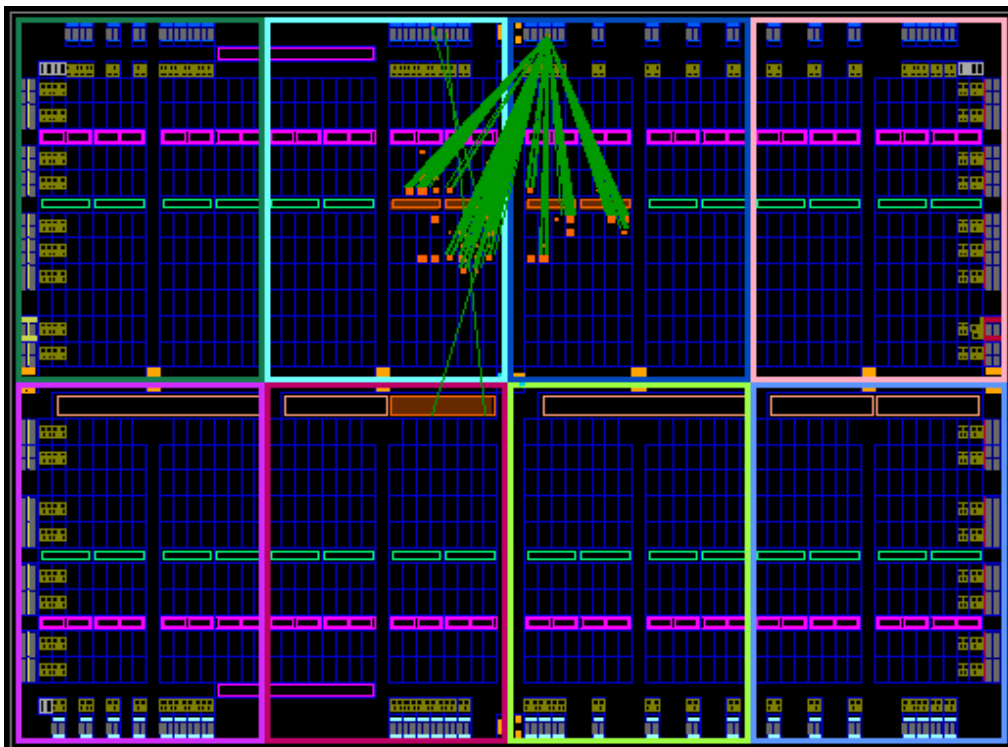


Figura 6.120. Esquema del circuito usando PlanAhead y aplicando modificaciones

6.5.1.2 Optimización de potencia

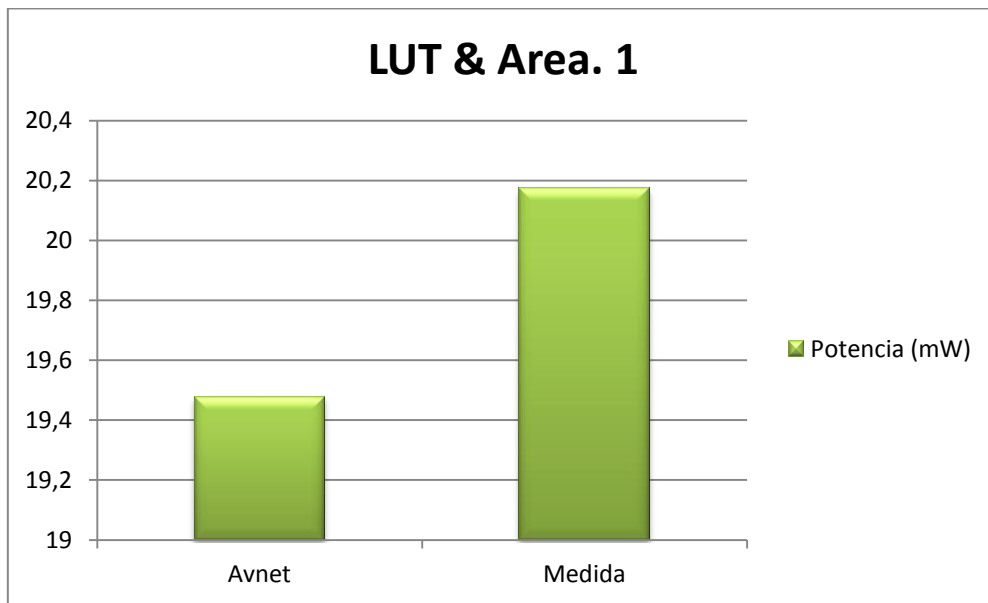


Figura 6.121. Medidas usando PlanAhead y sin aplicar modificaciones

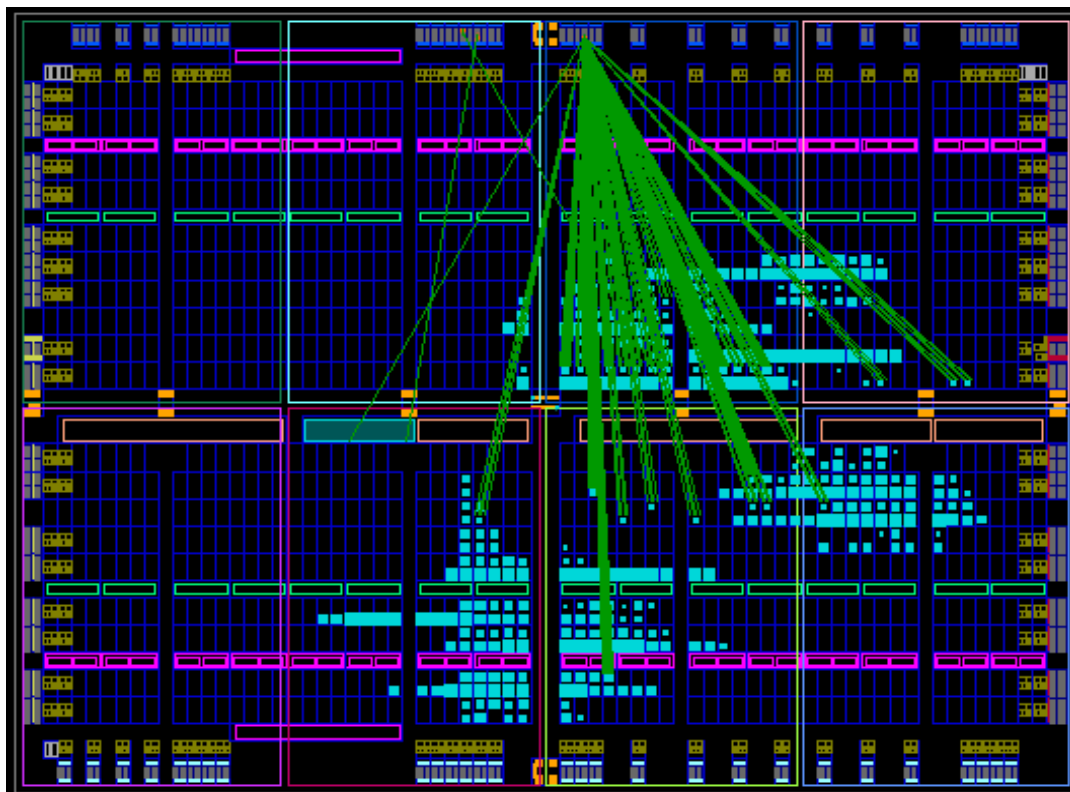


Figura 6.122. Esquema del circuito usando PlanAhead y sin aplicar modificaciones

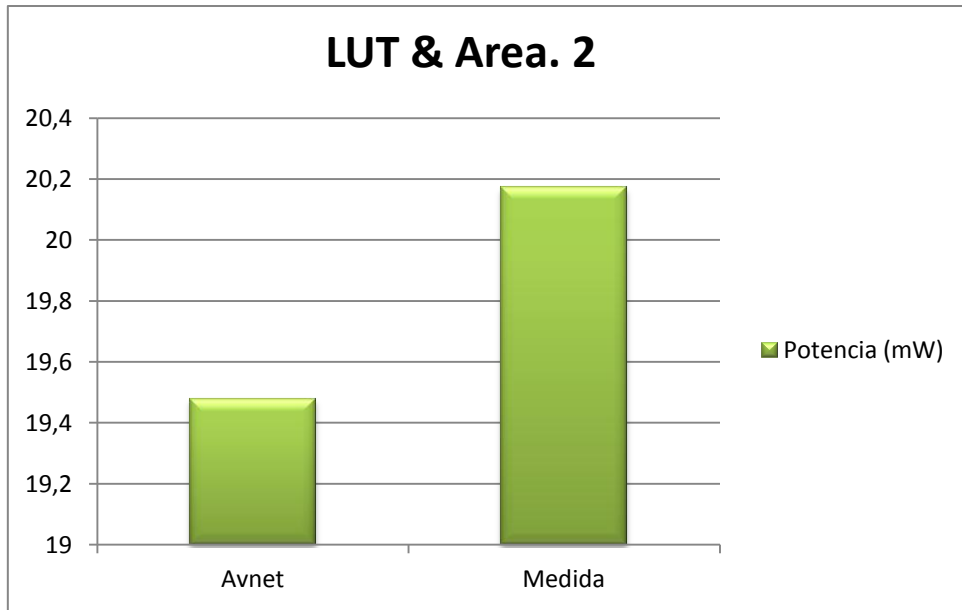


Figura 6.123. Medidas usando PlanAhead y aplicando modificaciones

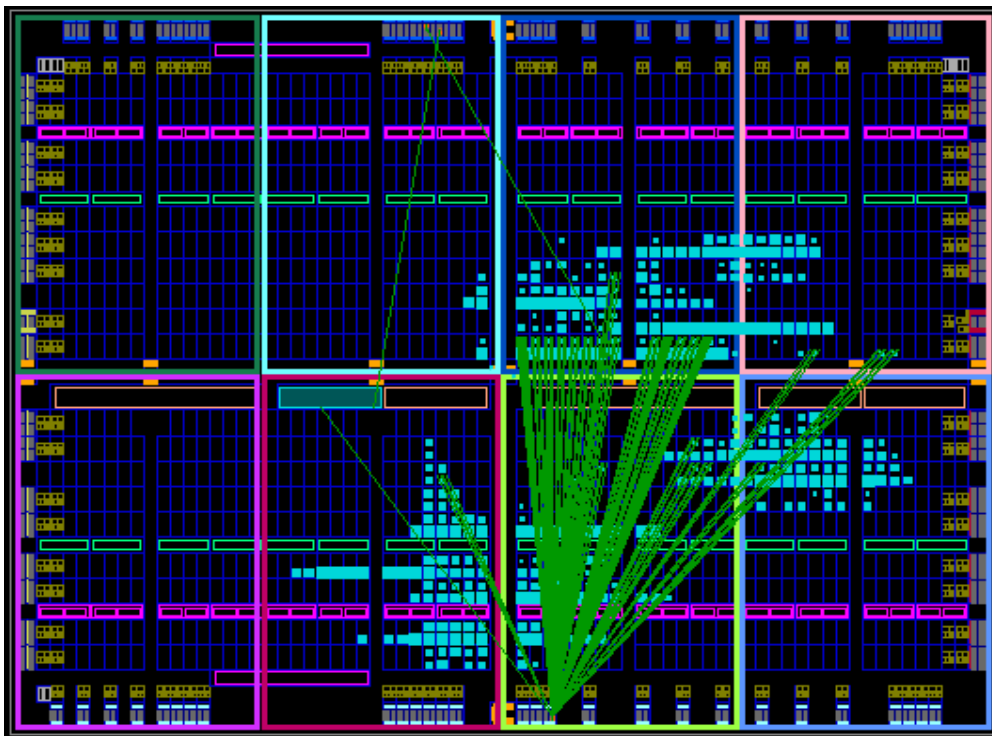


Figura 6.124. Esquema del circuito usando PlanAhead y aplicando modificaciones

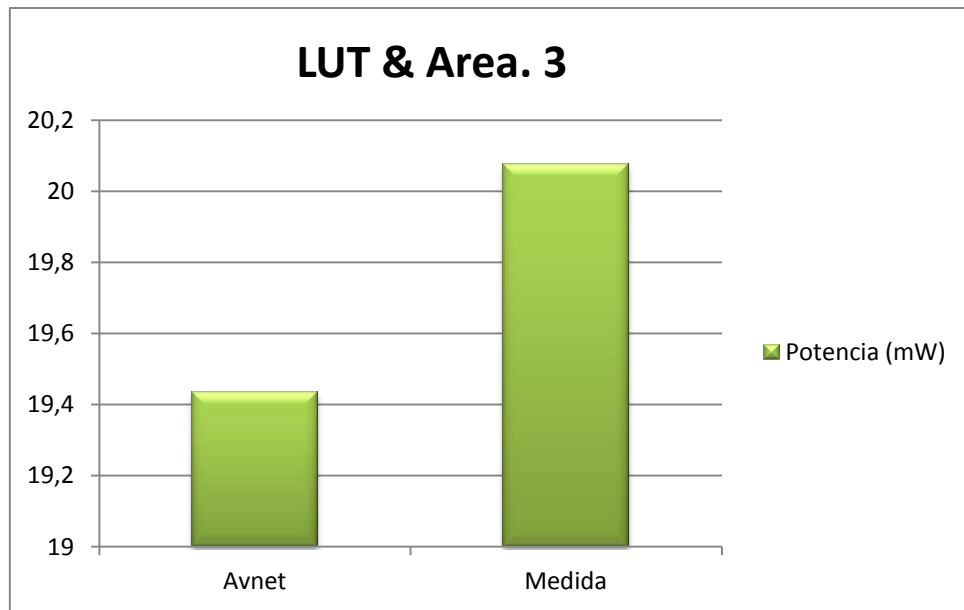


Figura 6.125. Medidas usando PlanAhead y aplicando modificaciones

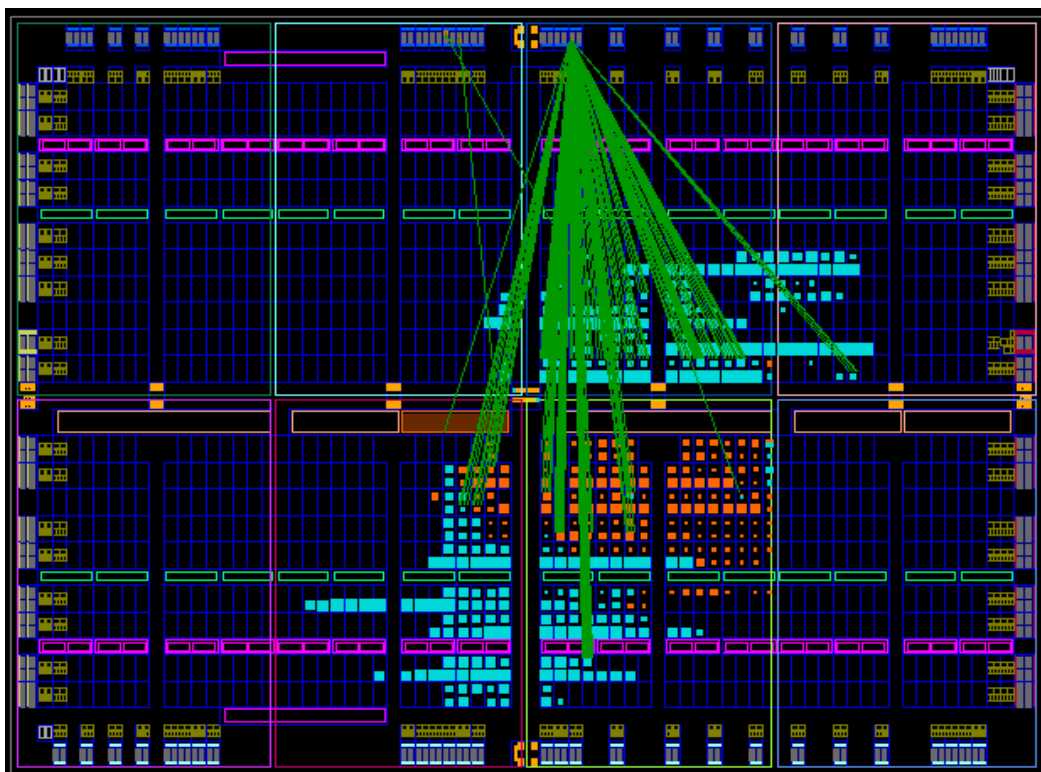


Figura 6.126. Esquema del circuito usando PlanAhead y aplicando modificaciones

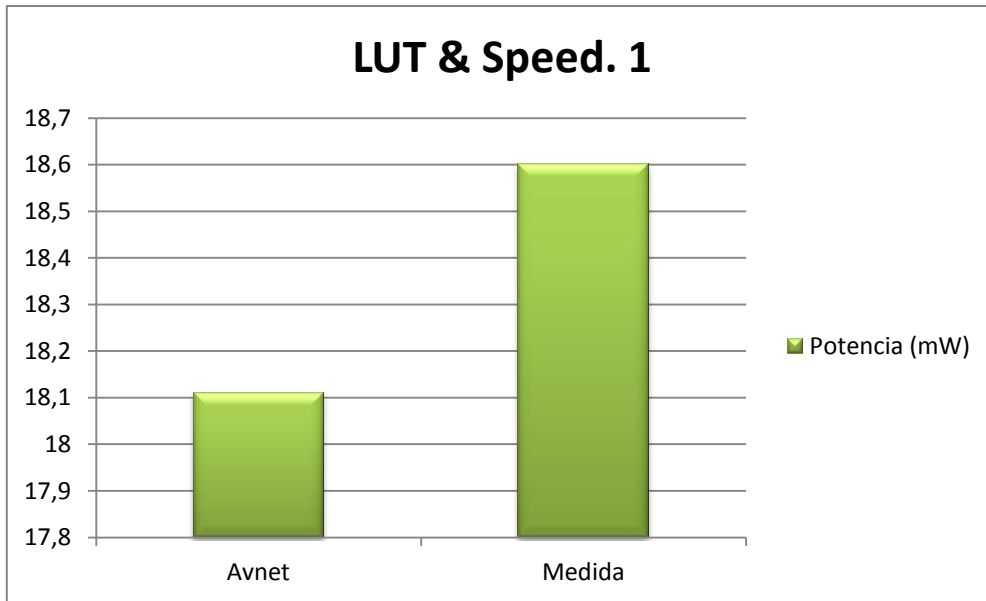


Figura 6.127. Medidas usando PlanAhead y sin aplicar modificaciones

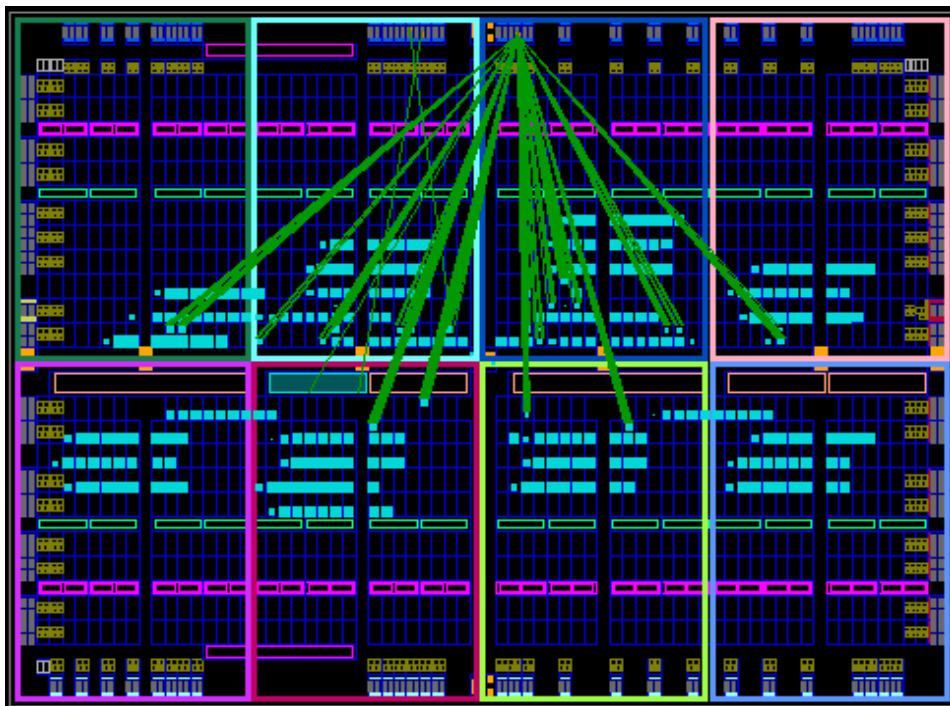


Figura 6.128. Esquema del circuito usando PlanAhead y sin aplicar modificaciones

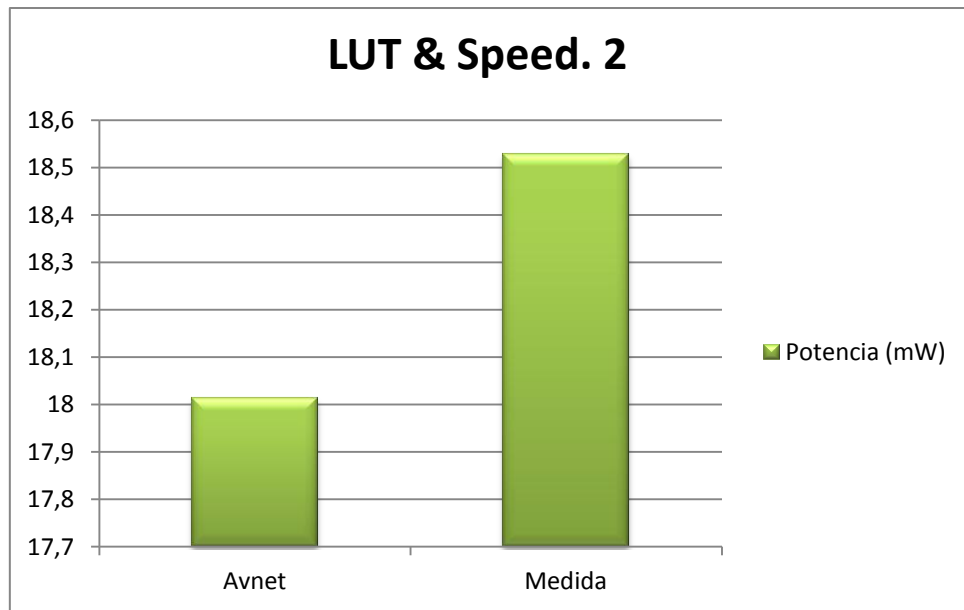


Figura 6.129. Medidas usando PlanAhead y aplicando modificaciones

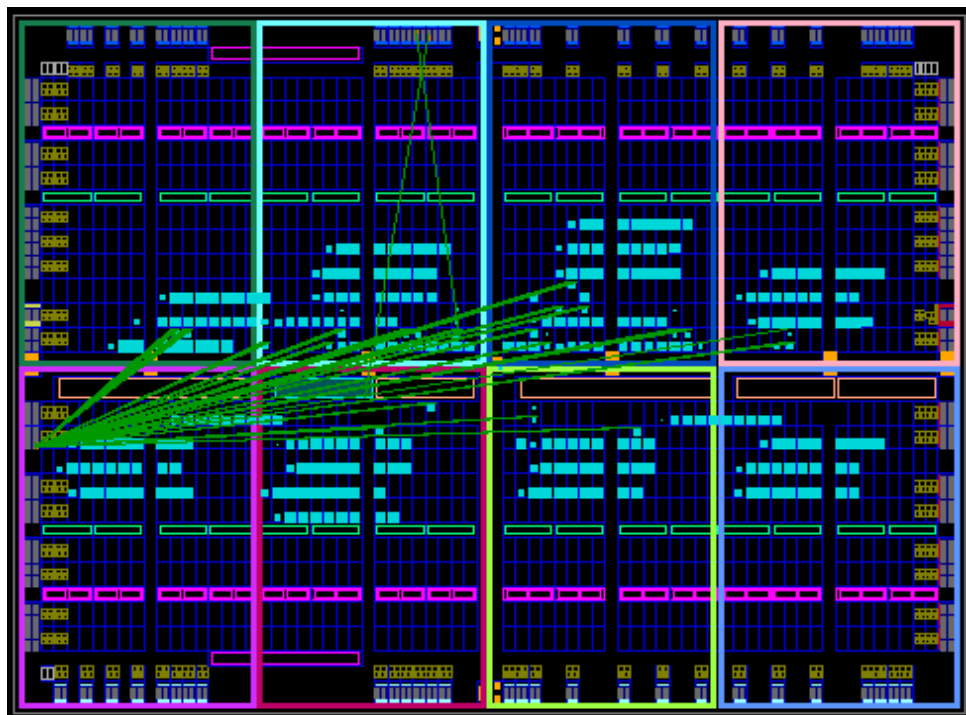


Figura 6.130. Esquema del circuito usando PlanAhead y aplicando modificaciones

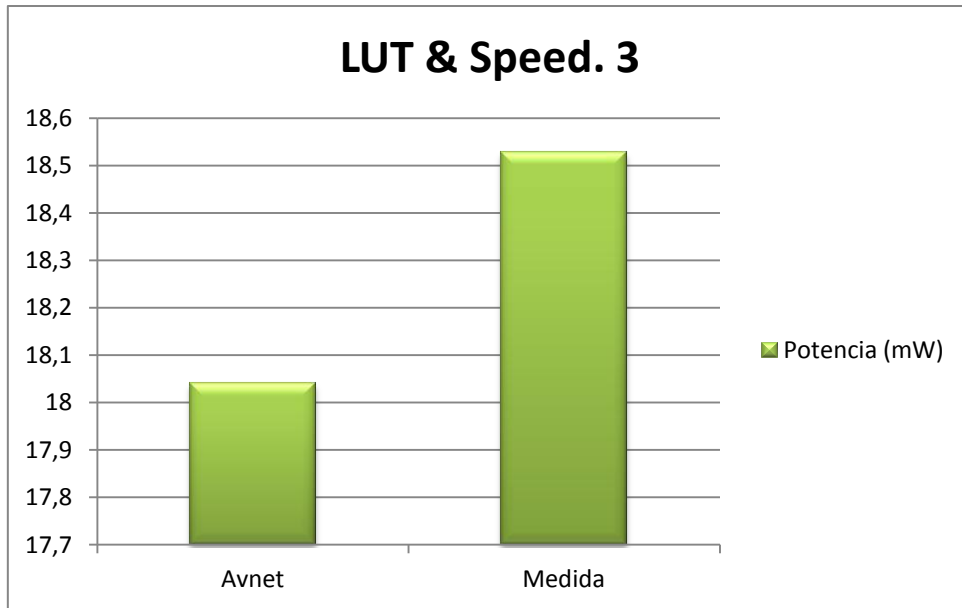


Figura 6.131. Medidas usando PlanAhead y aplicando modificaciones

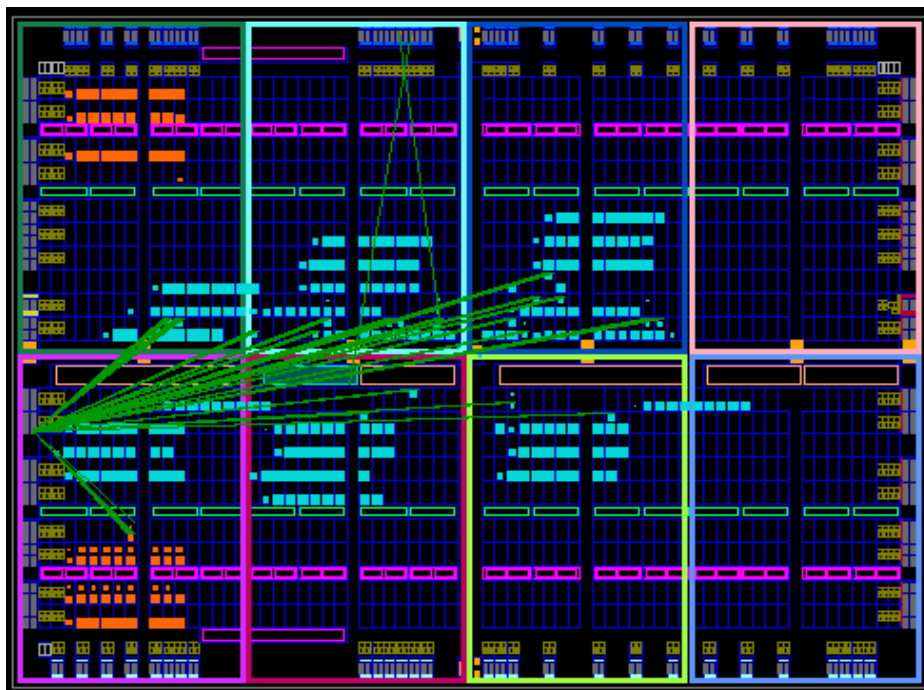


Figura 6.132. Esquema del circuito usando PlanAhead y aplicando modificaciones

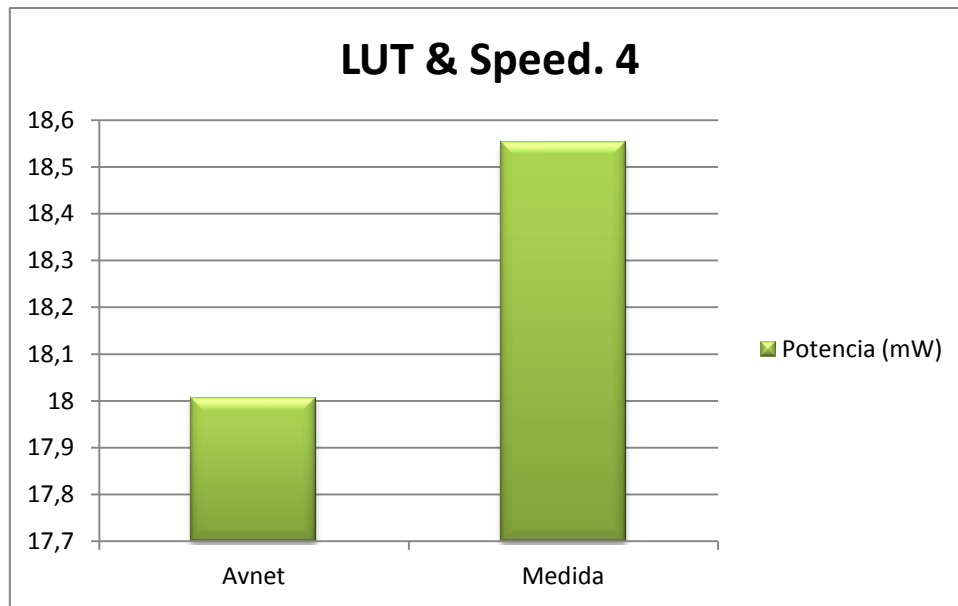


Figura 6.133. Medidas usando PlanAhead y aplicando modificaciones

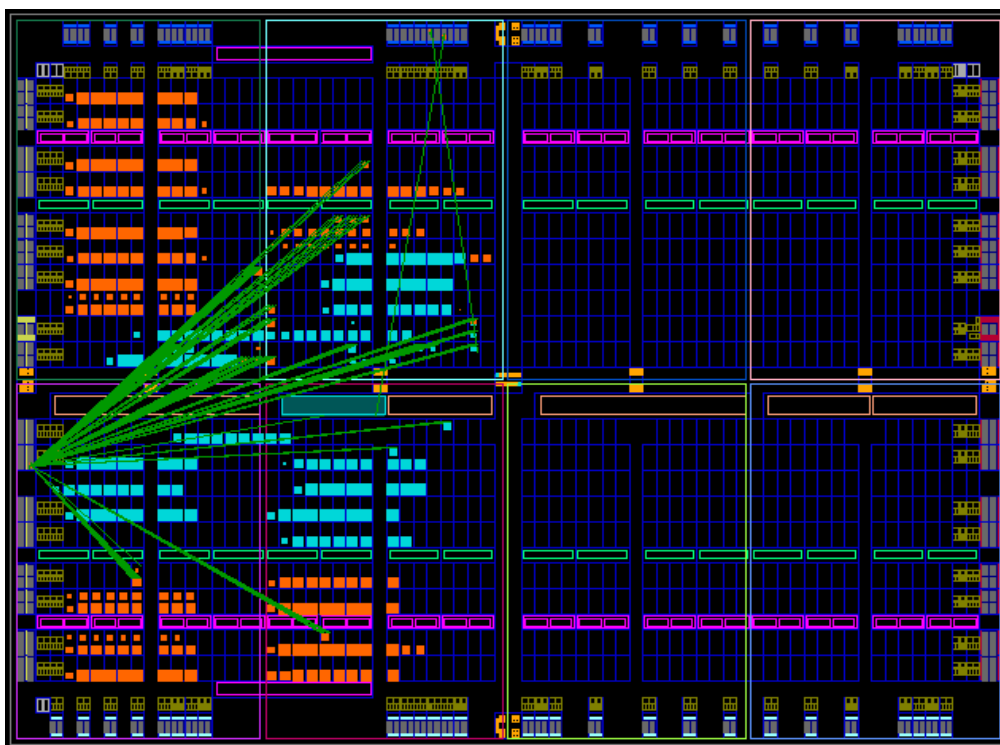


Figura 6.134. Esquema del circuito usando PlanAhead y aplicando modificaciones

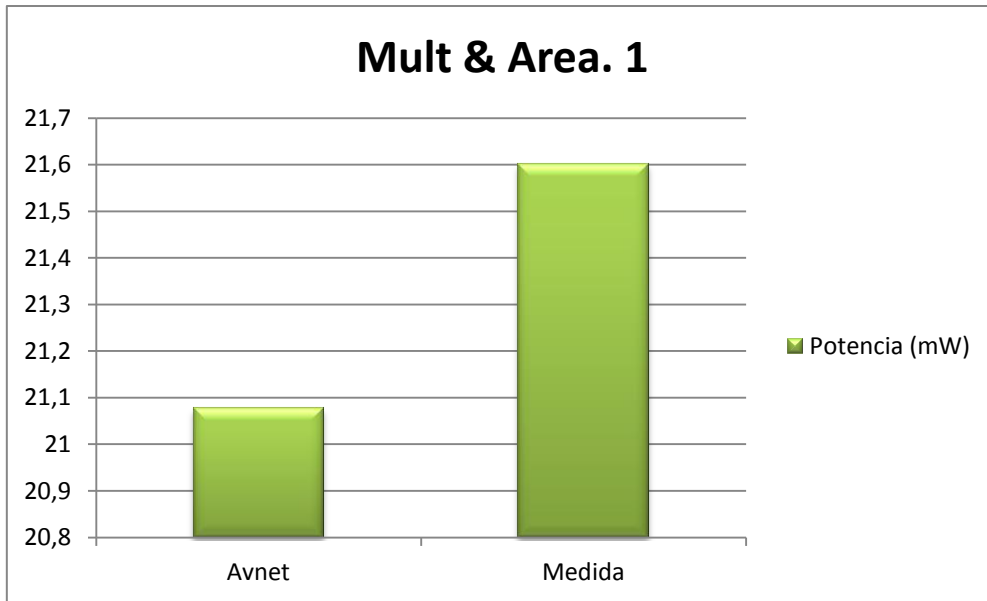


Figura 6.135. Medidas usando PlanAhead y sin aplicar modificaciones

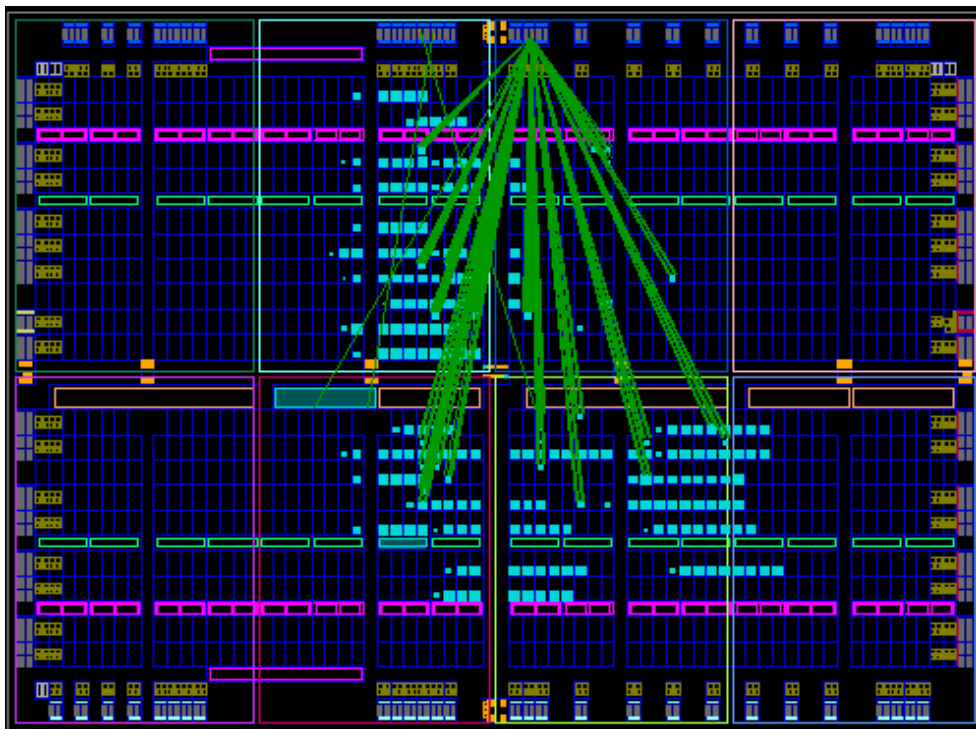


Figura 6.136. Esquema del circuito usando PlanAhead y sin aplicar modificaciones

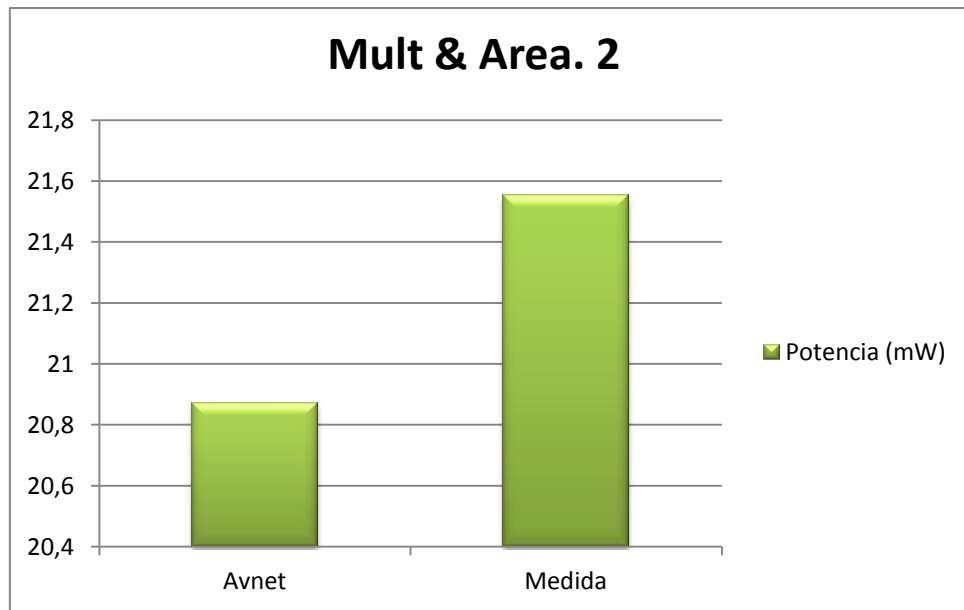


Figura 6.137. Medidas usando PlanAhead y aplicando modificaciones

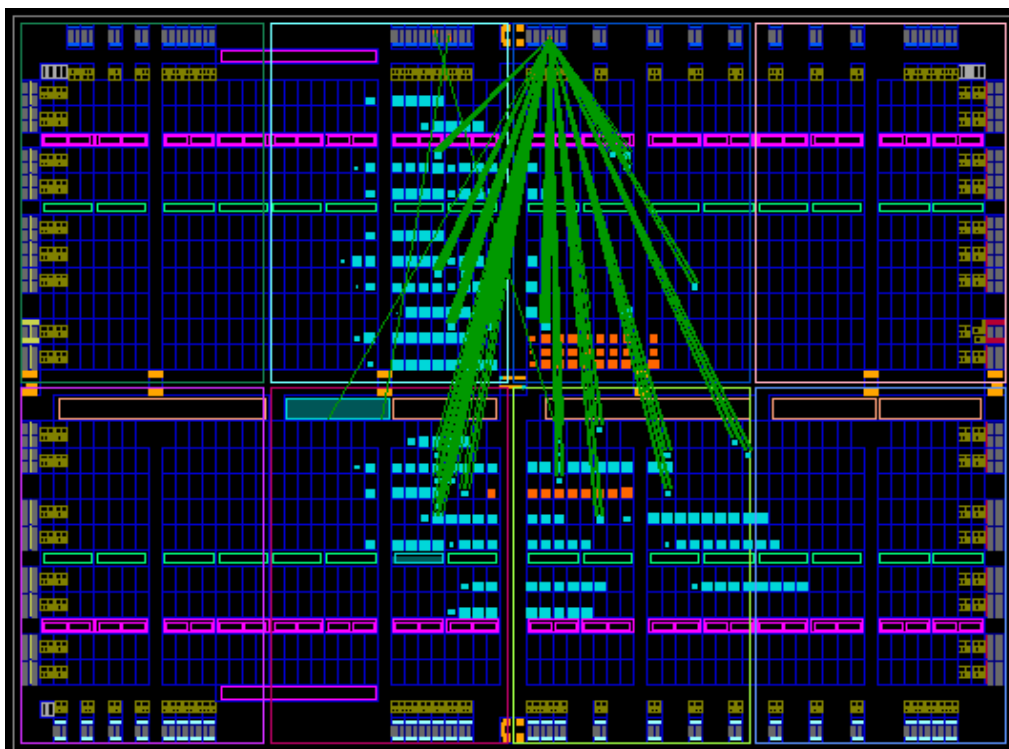


Figura 6.138. Esquema del circuito usando PlanAhead y aplicando modificaciones

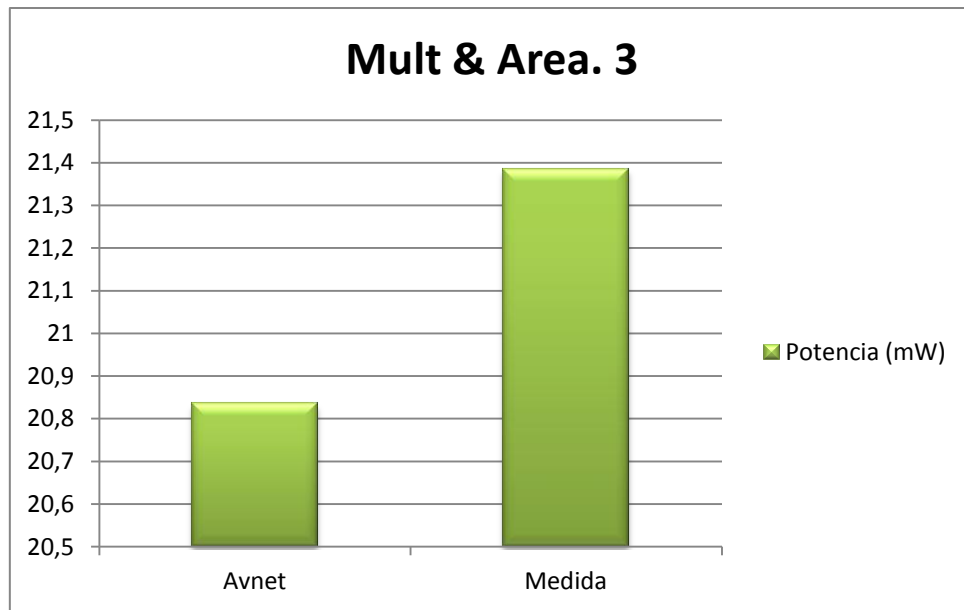


Figura 6.139. Medidas usando PlanAhead y aplicando modificaciones

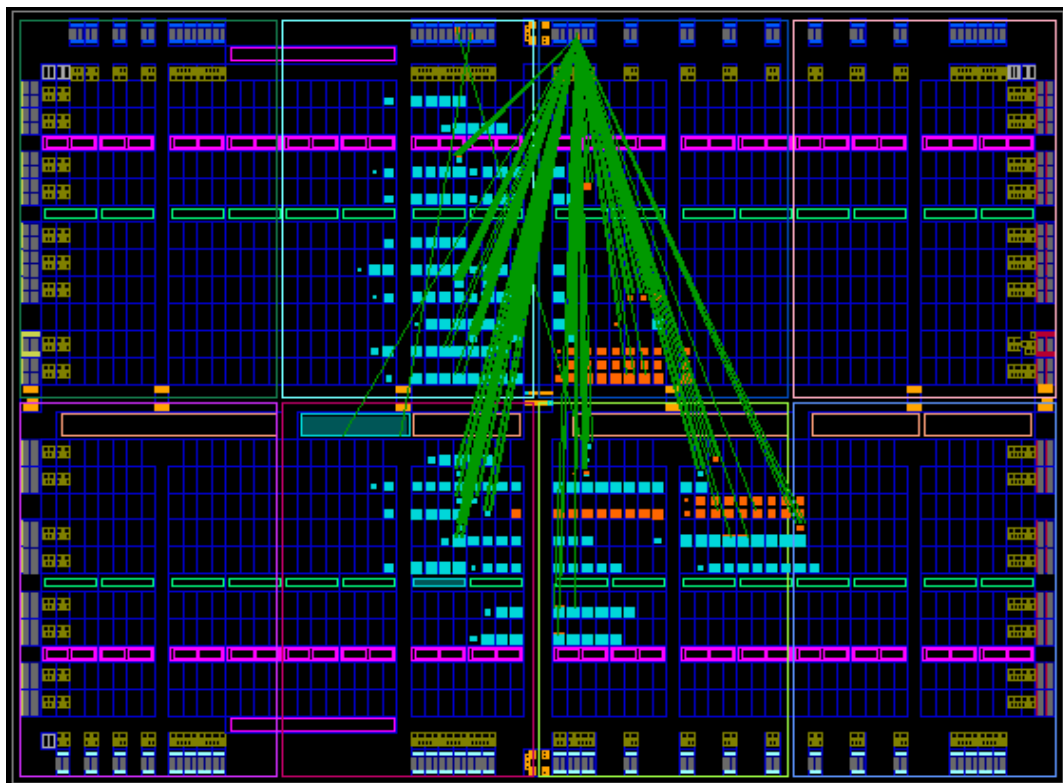


Figura 6.140. Esquema del circuito usando PlanAhead y aplicando modificaciones

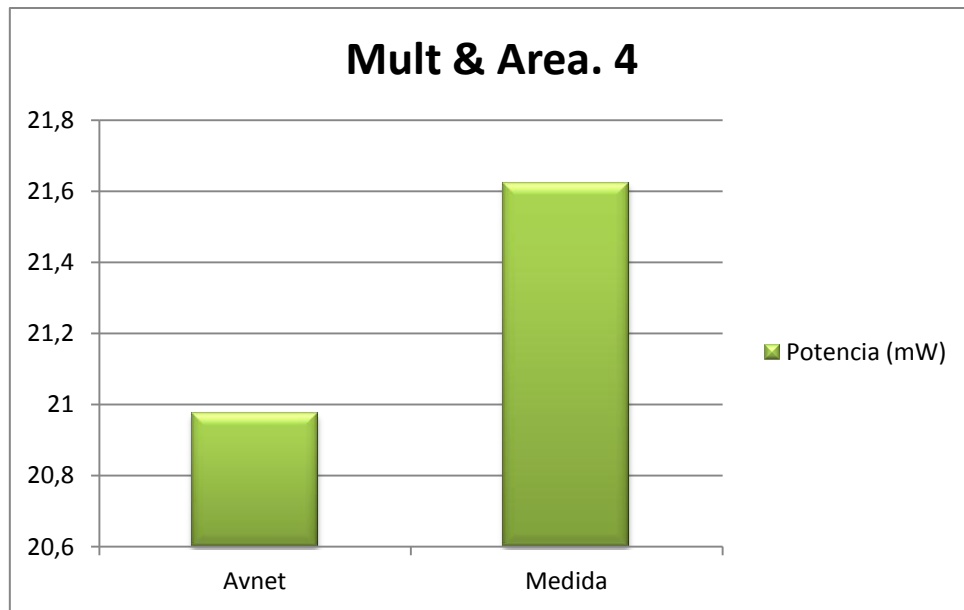


Figura 6.141. Medidas usando PlanAhead y aplicando modificaciones

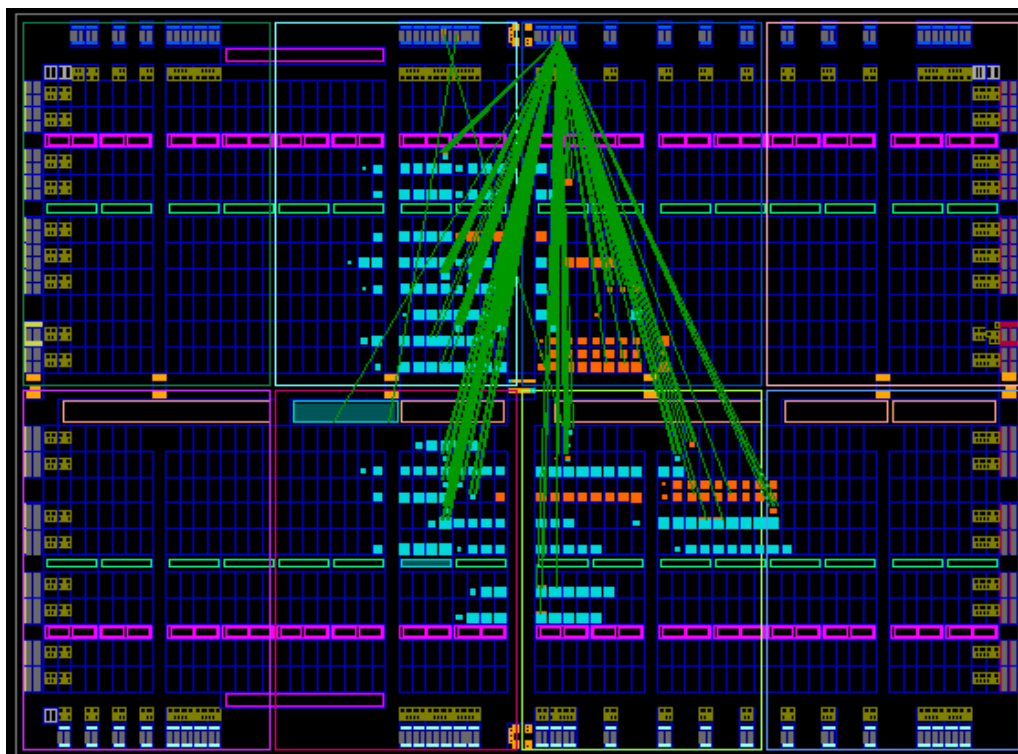


Figura 6.142. Esquema del circuito usando PlanAhead y aplicando modificaciones

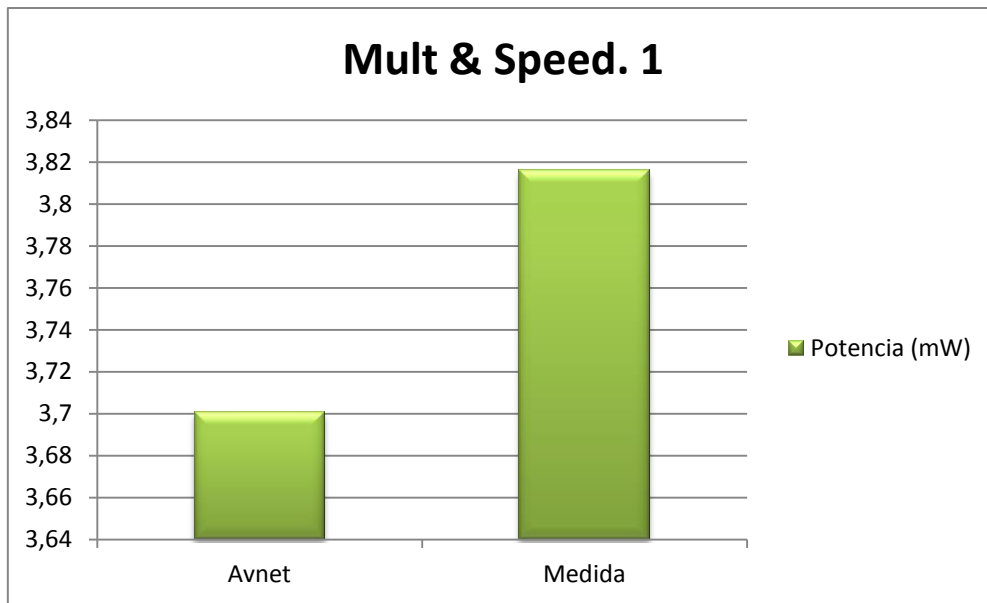


Figura 6.143. Medidas usando PlanAhead y sin aplicar modificaciones

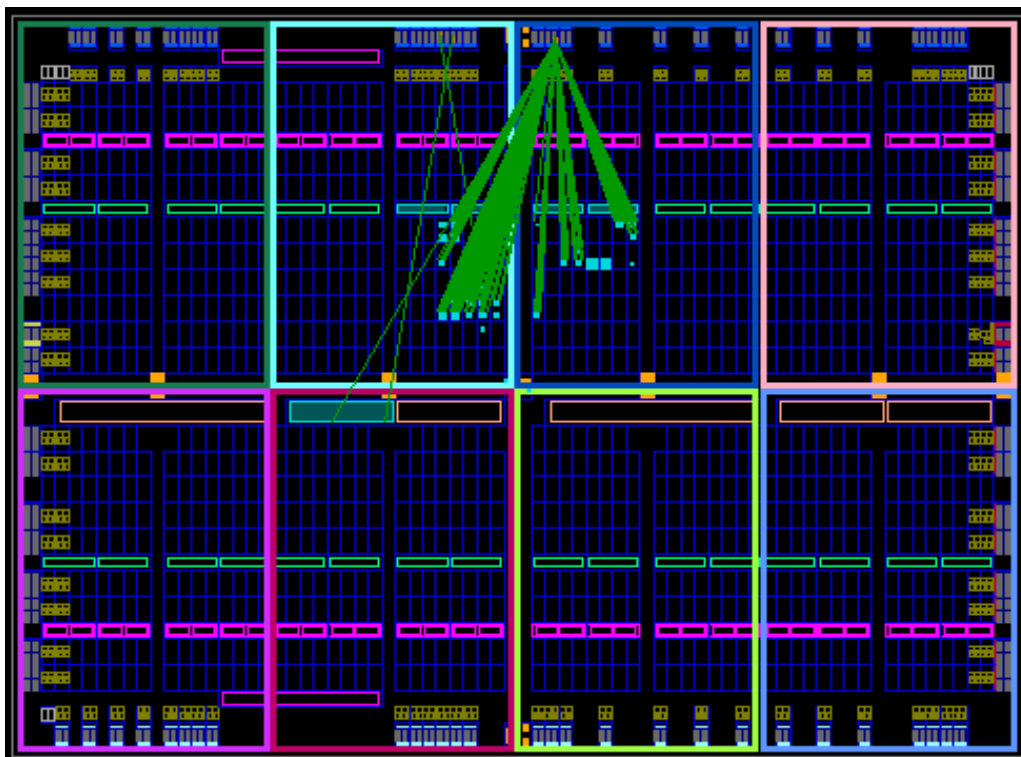


Figura 6.144. Esquema del circuito usando PlanAhead y sin aplicar modificaciones

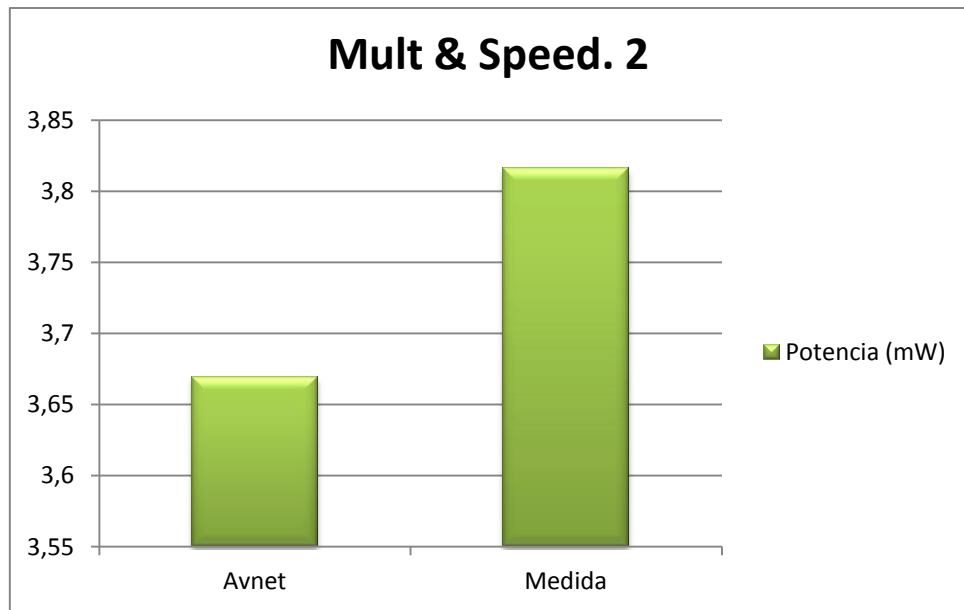


Figura 6.145. Medidas usando PlanAhead y aplicando modificaciones

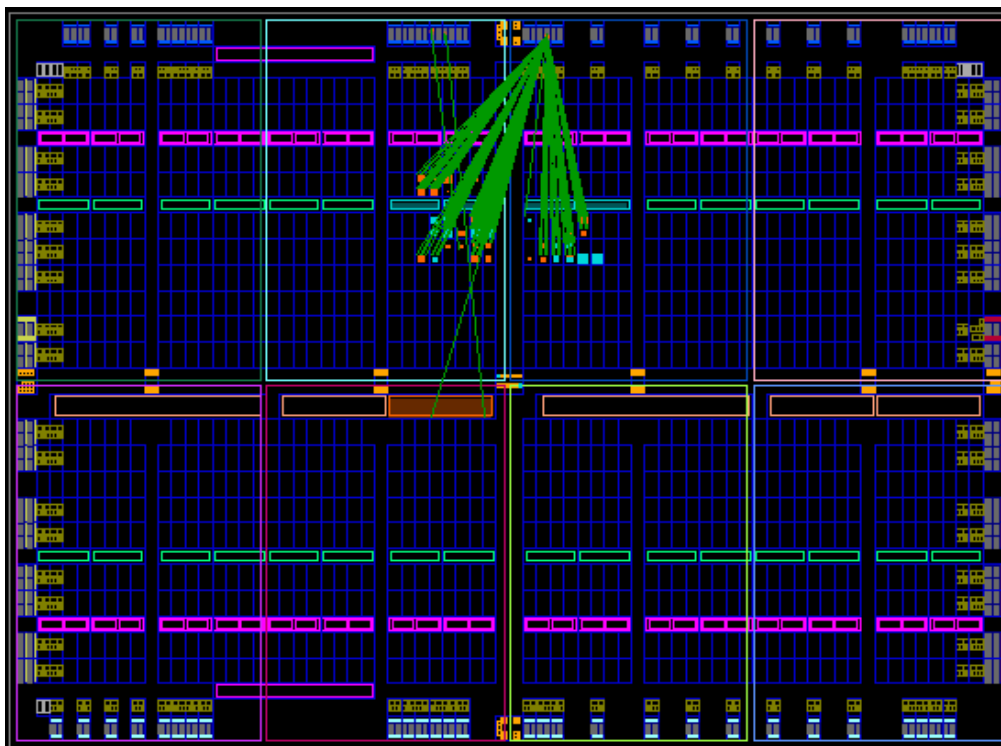


Figura 6.146. Esquema del circuito usando PlanAhead y aplicando modificaciones

6.5.2 Multiplicador segmentado

6.5.2.1 Balanceado

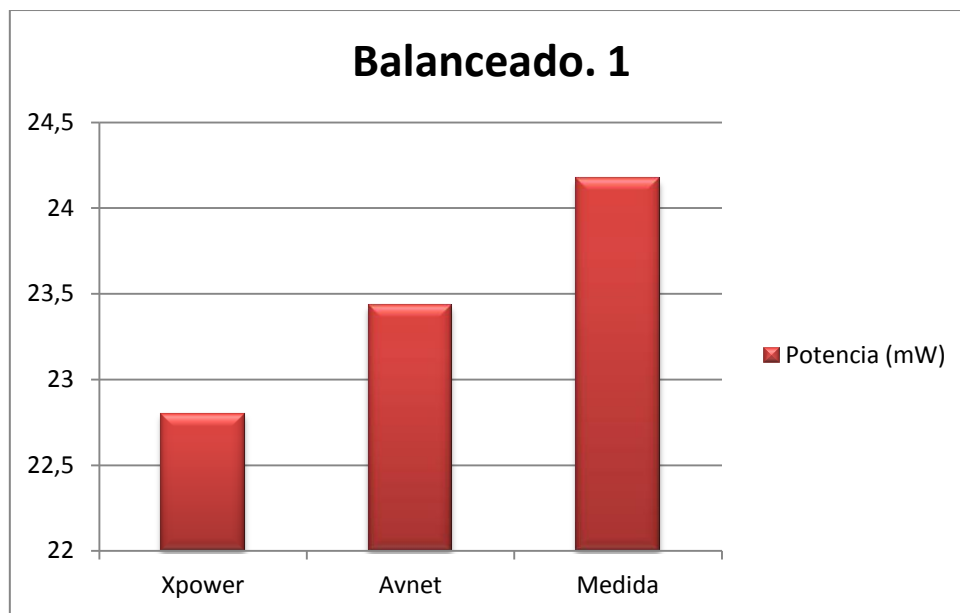


Figura 6.147. Medidas usando PlanAhead y sin aplicar modificaciones

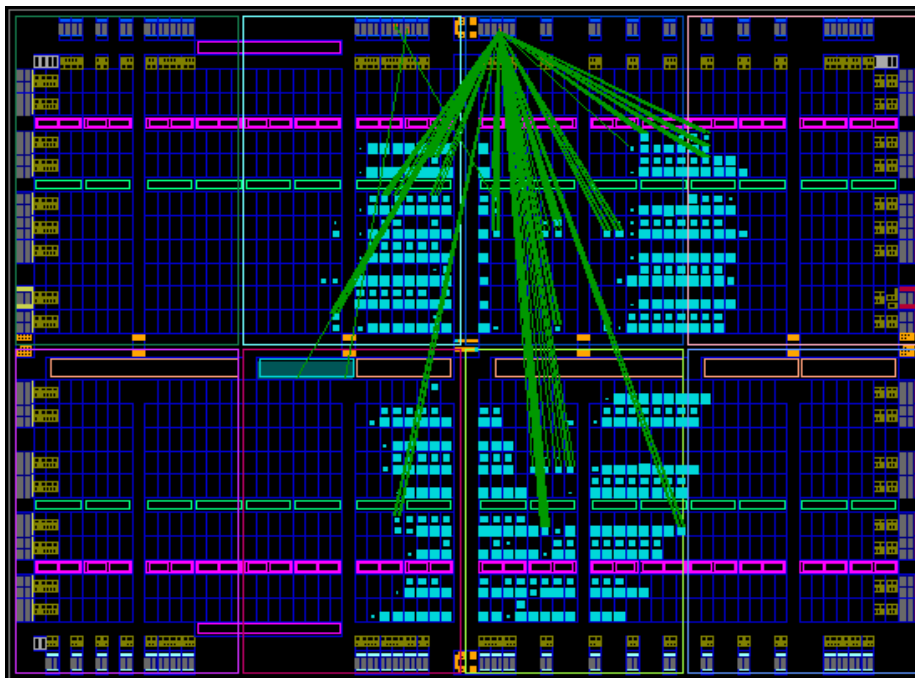


Figura 6.148. Esquema del circuito usando PlanAhead y sin aplicar modificaciones

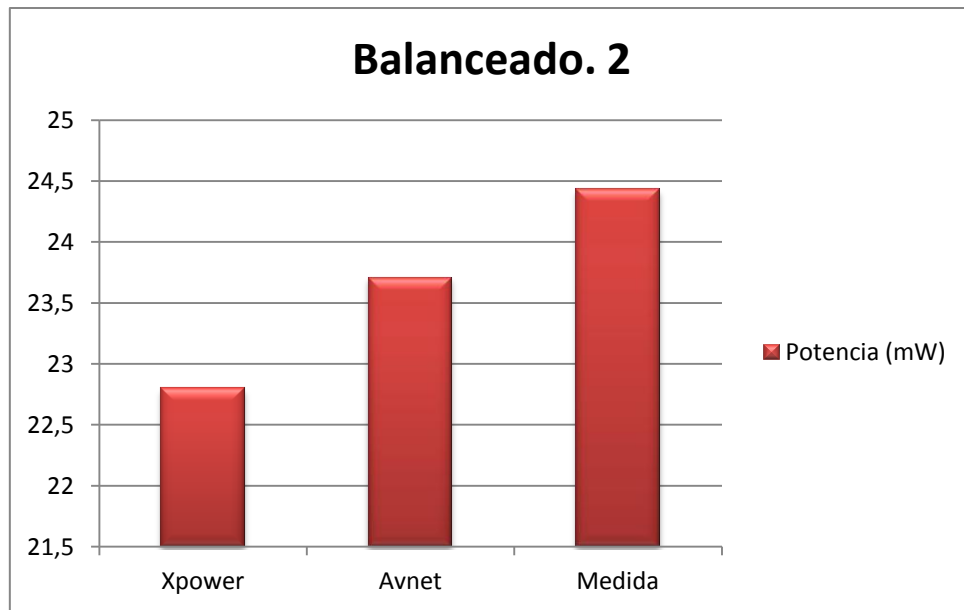


Figura 6.149. Medidas usando PlanAhead y aplicando modificaciones

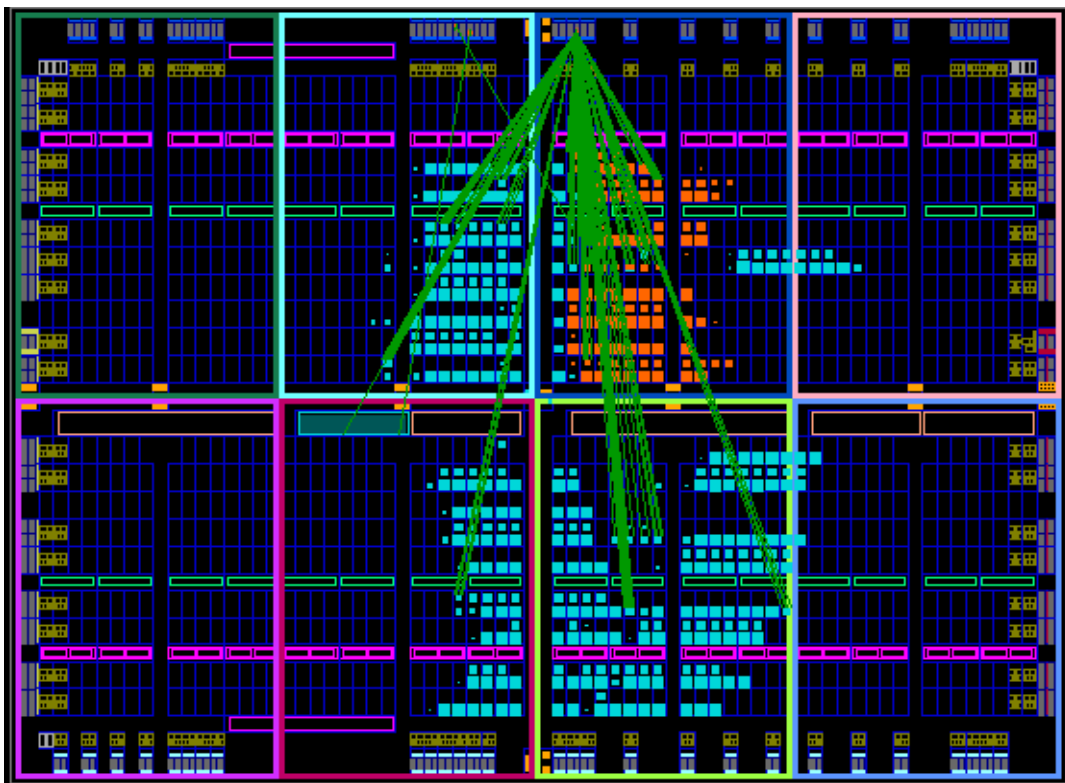


Figura 6.150. Esquema del circuito usando PlanAhead y aplicando modificaciones

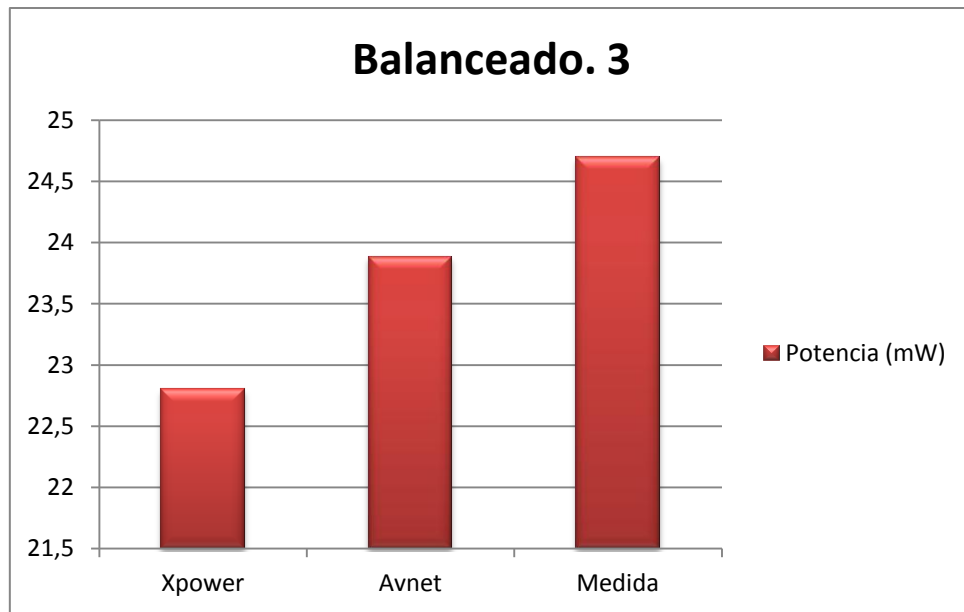


Figura 6.151. Medidas usando PlanAhead y aplicando modificaciones

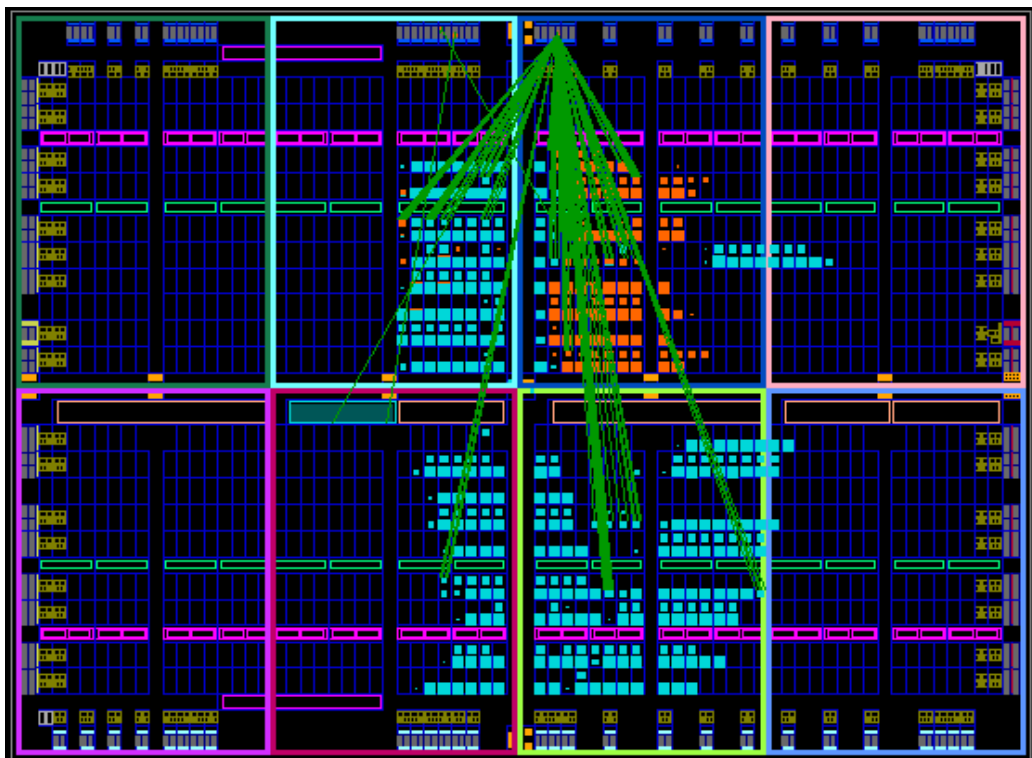


Figura 6.152. Esquema del circuito usando PlanAhead y aplicando modificaciones

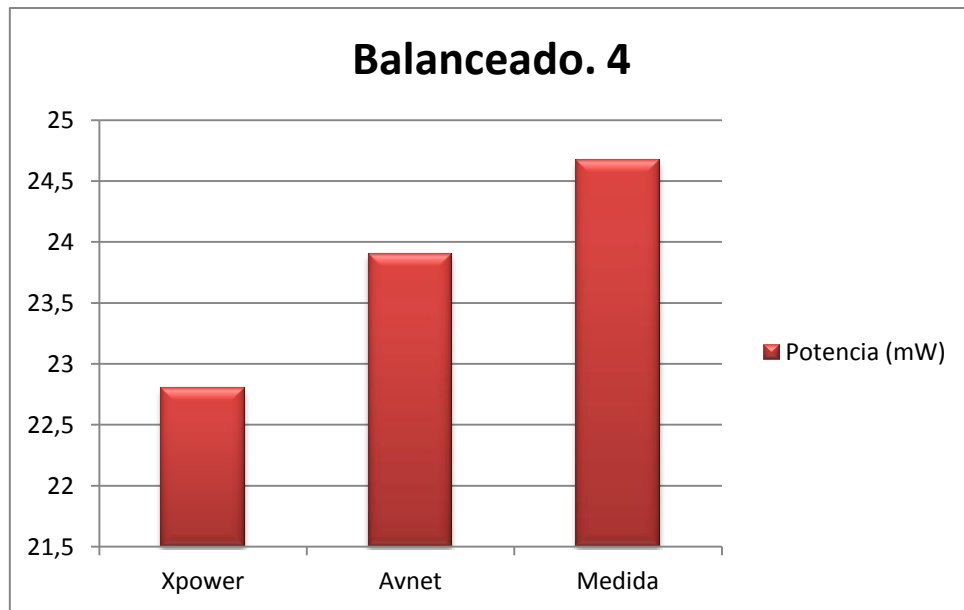


Figura 6.153. Medidas usando PlanAhead y aplicando modificaciones

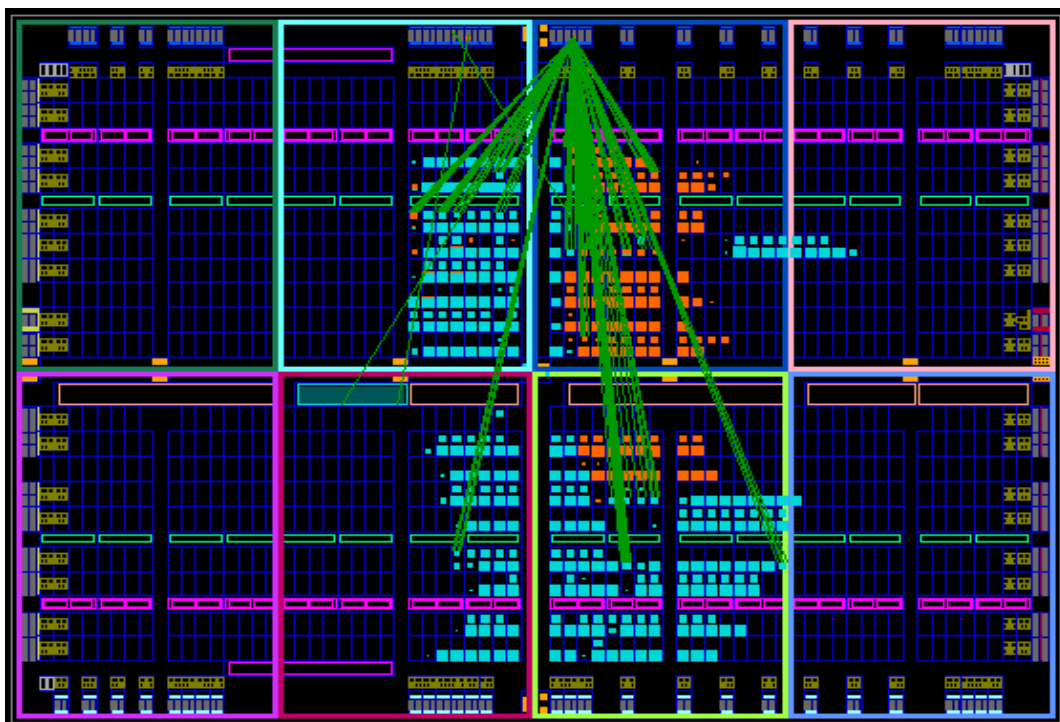


Figura 6.154. Esquema del circuito usando PlanAhead y aplicando modificaciones

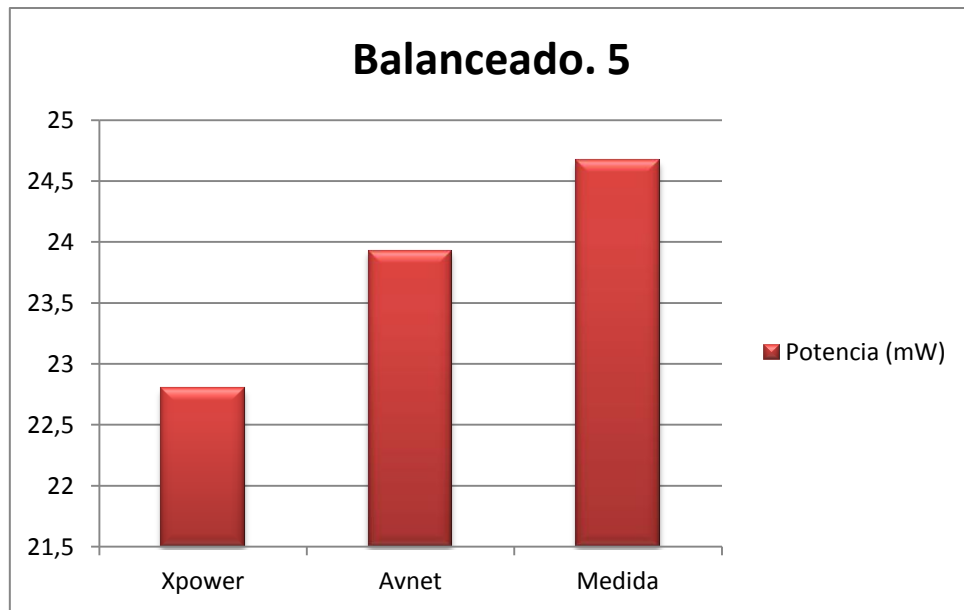


Figura 6.155. Medidas usando PlanAhead y aplicando modificaciones

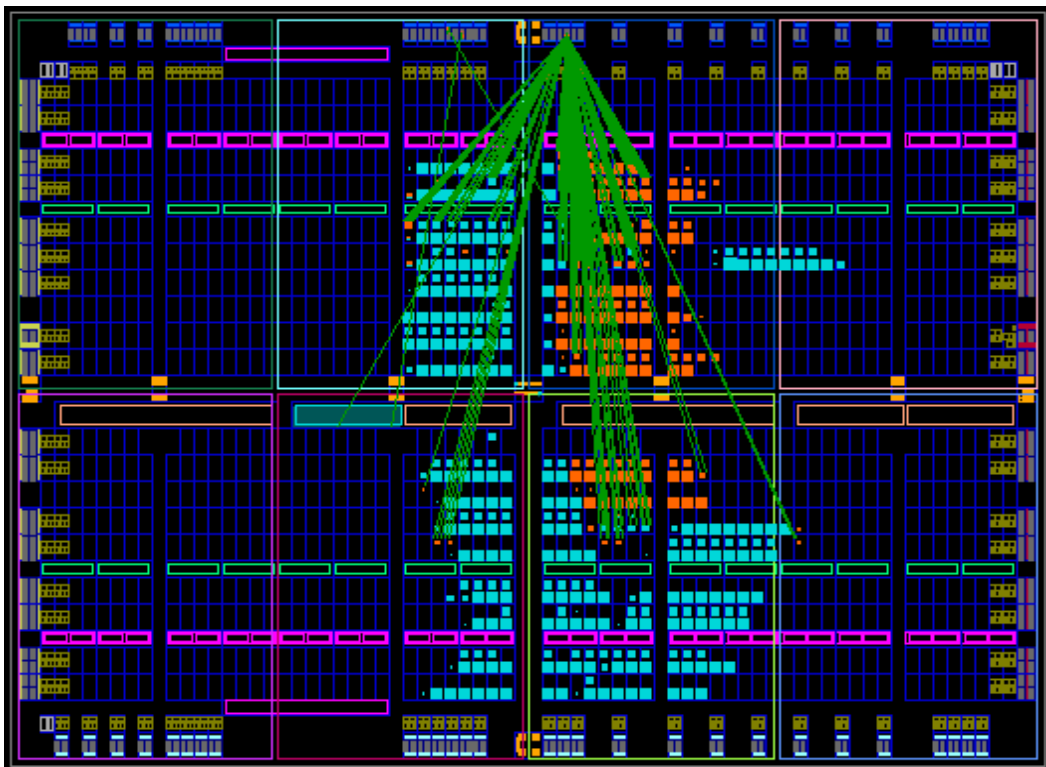


Figura 6.156. Esquema del circuito usando PlanAhead y aplicando modificaciones

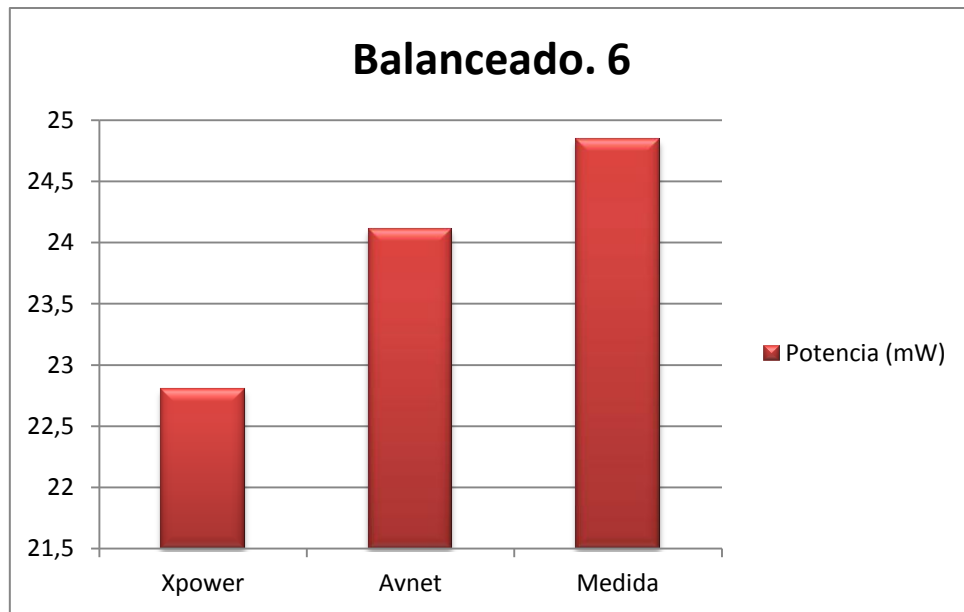


Figura 6.157. Medidas usando PlanAhead y aplicando modificaciones

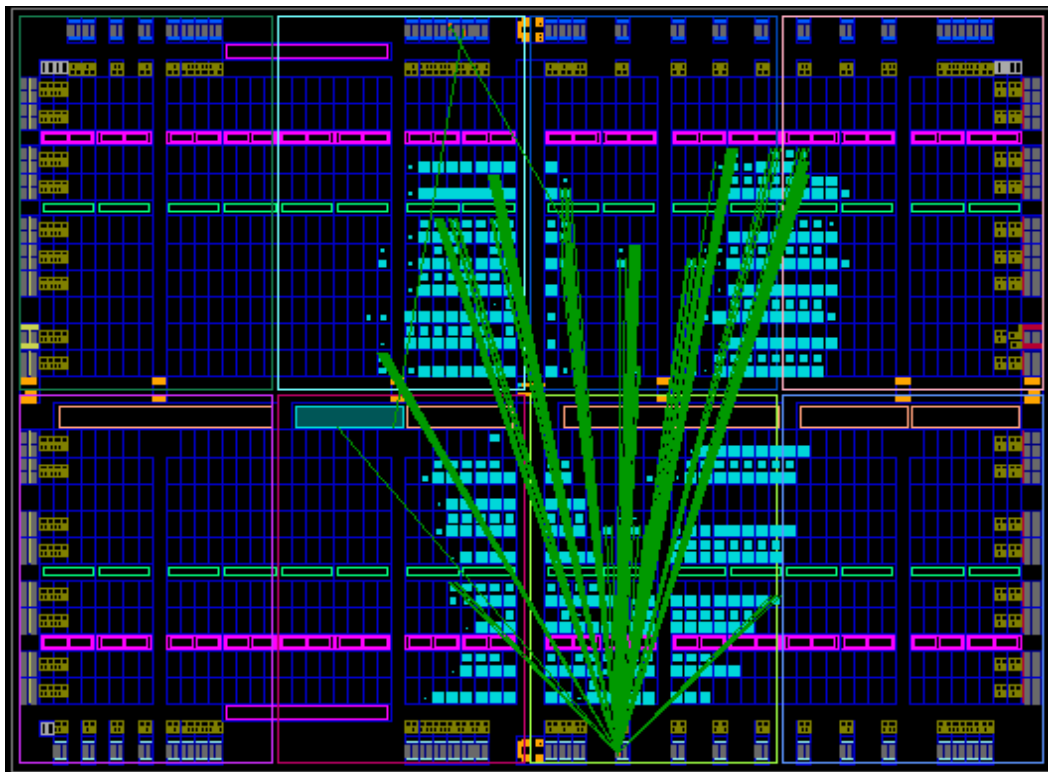


Figura 6.158. Esquema del circuito usando PlanAhead y aplicando modificaciones

6.5.2.2 Optimización de potencia

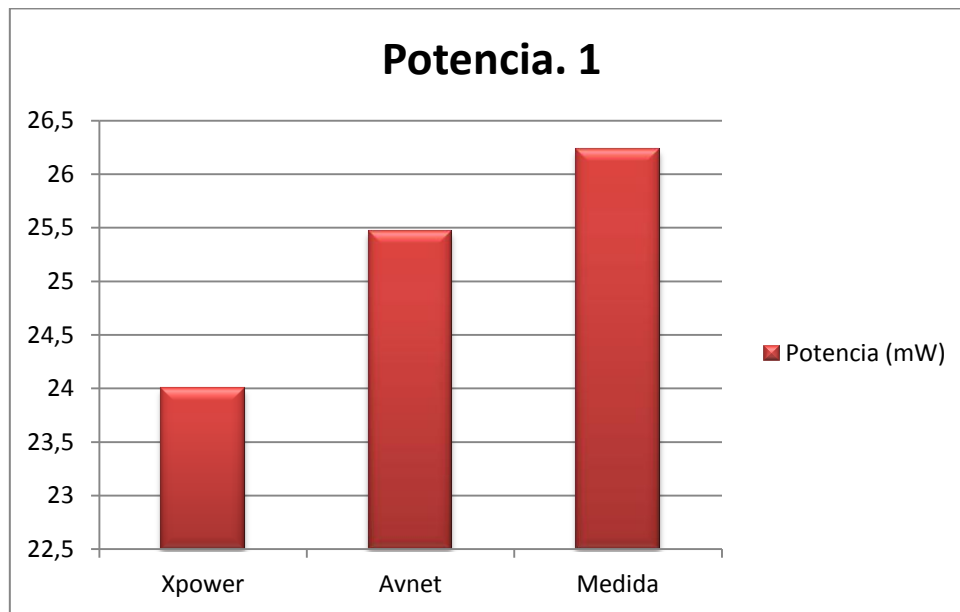


Figura 6.159. Medidas usando PlanAhead y sin aplicar modificaciones

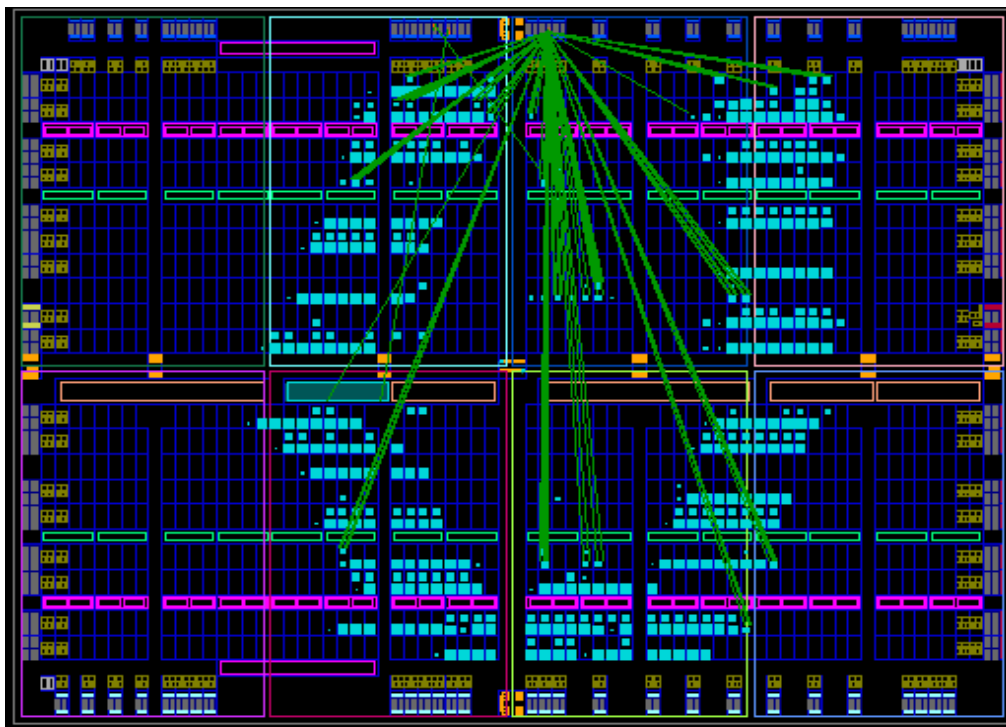


Figura 6.160. Esquema del circuito usando PlanAhead y sin aplicar modificaciones

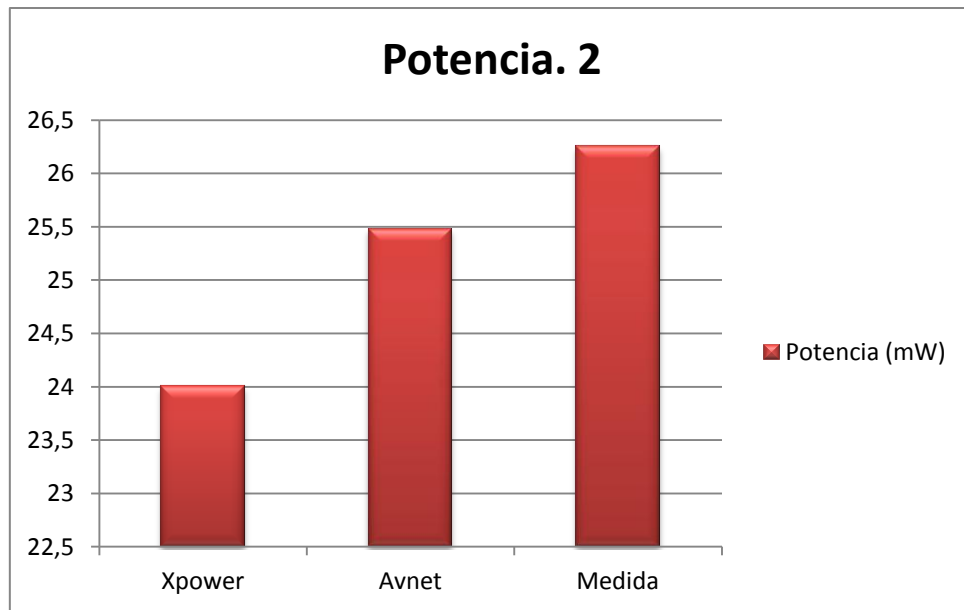


Figura 6.161. Medidas usando PlanAhead y aplicando modificaciones

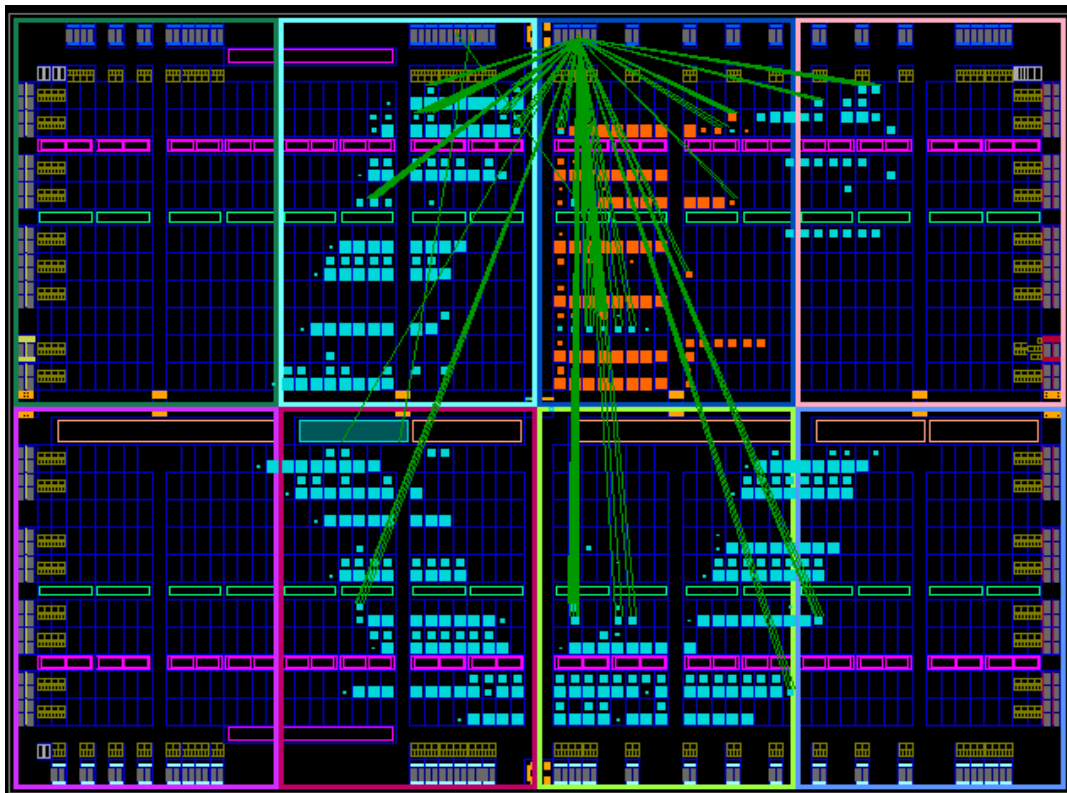


Figura 6.162. Esquema del circuito usando PlanAhead y aplicando modificaciones

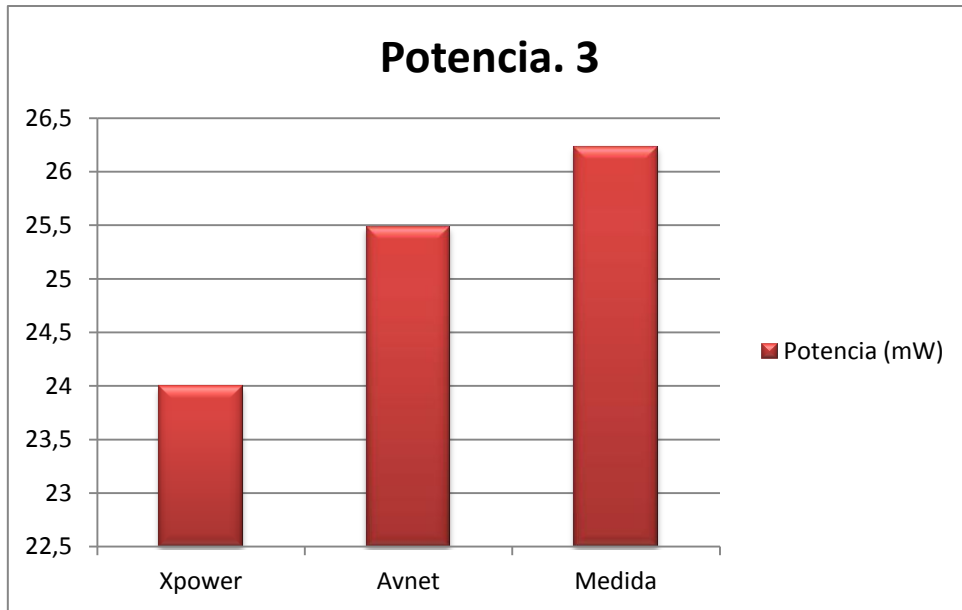


Figura 6.163. Medidas usando PlanAhead y aplicando modificaciones

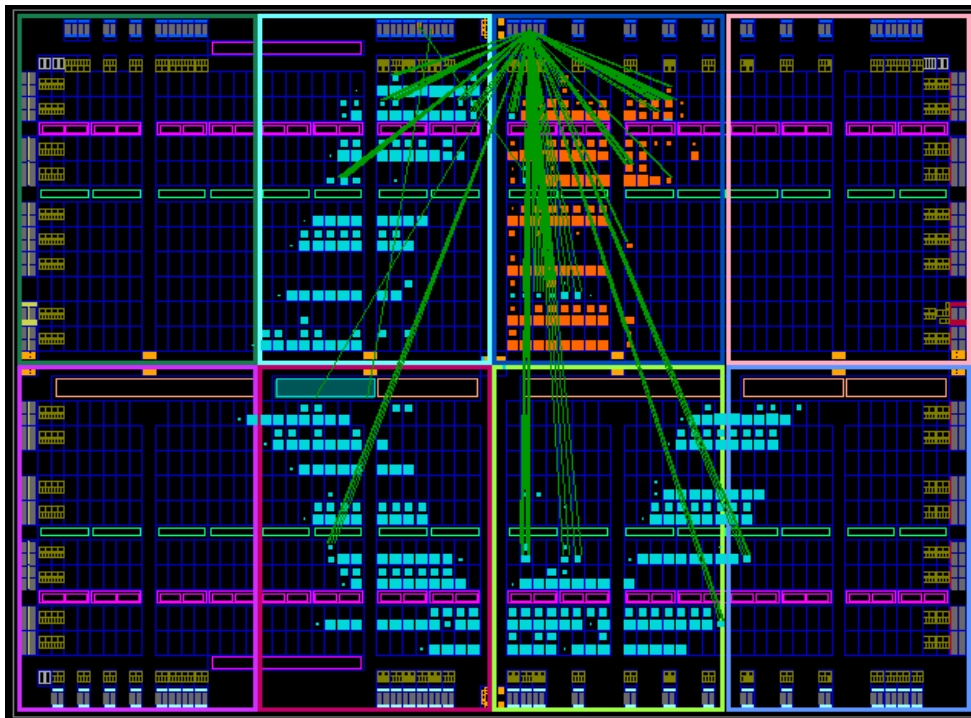


Figura 6.164. Esquema del circuito usando PlanAhead y aplicando modificaciones

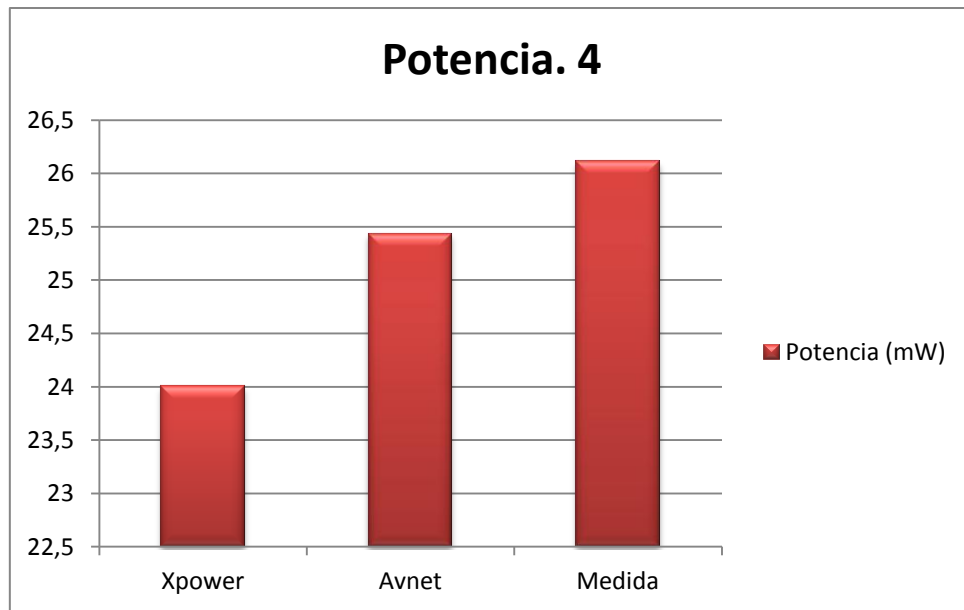


Figura 6.165. Medidas usando PlanAhead y aplicando modificaciones

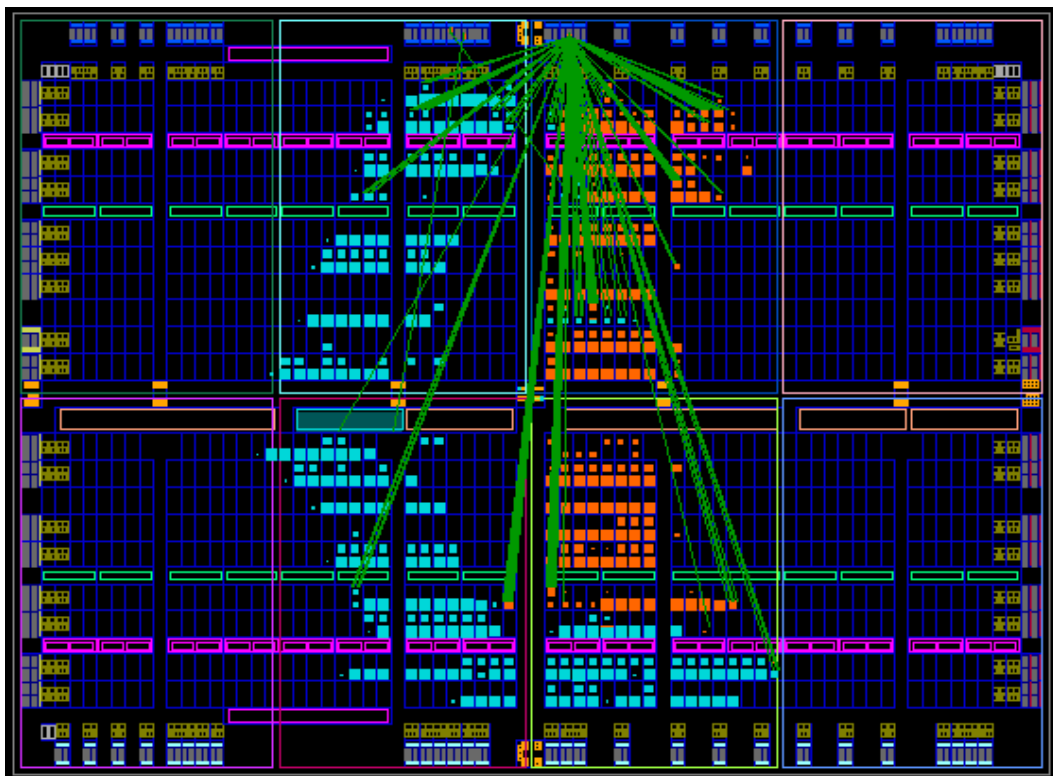


Figura 6.166. Esquema del circuito usando PlanAhead y aplicando modificaciones

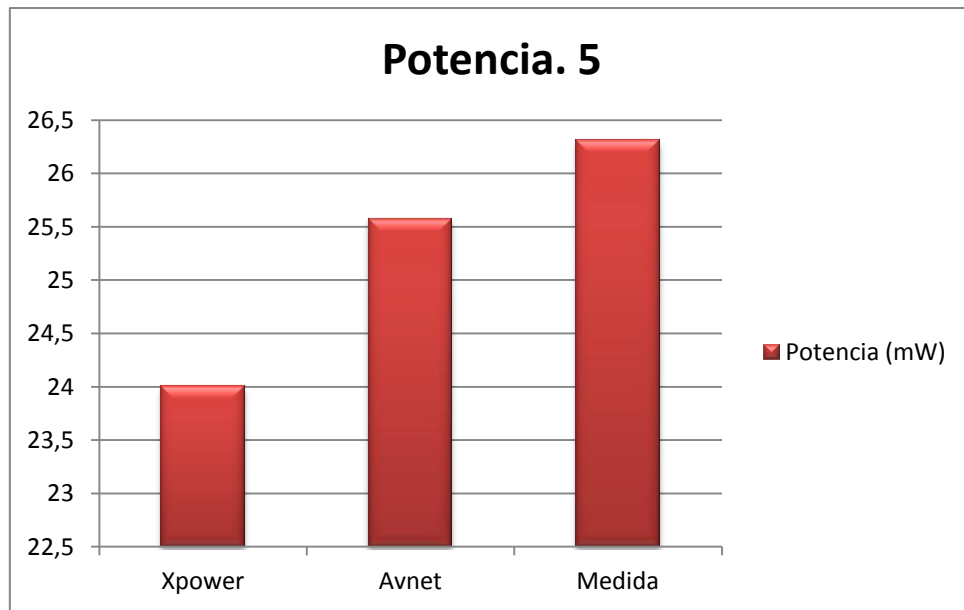


Figura 6.167. Medidas usando PlanAhead y aplicando modificaciones

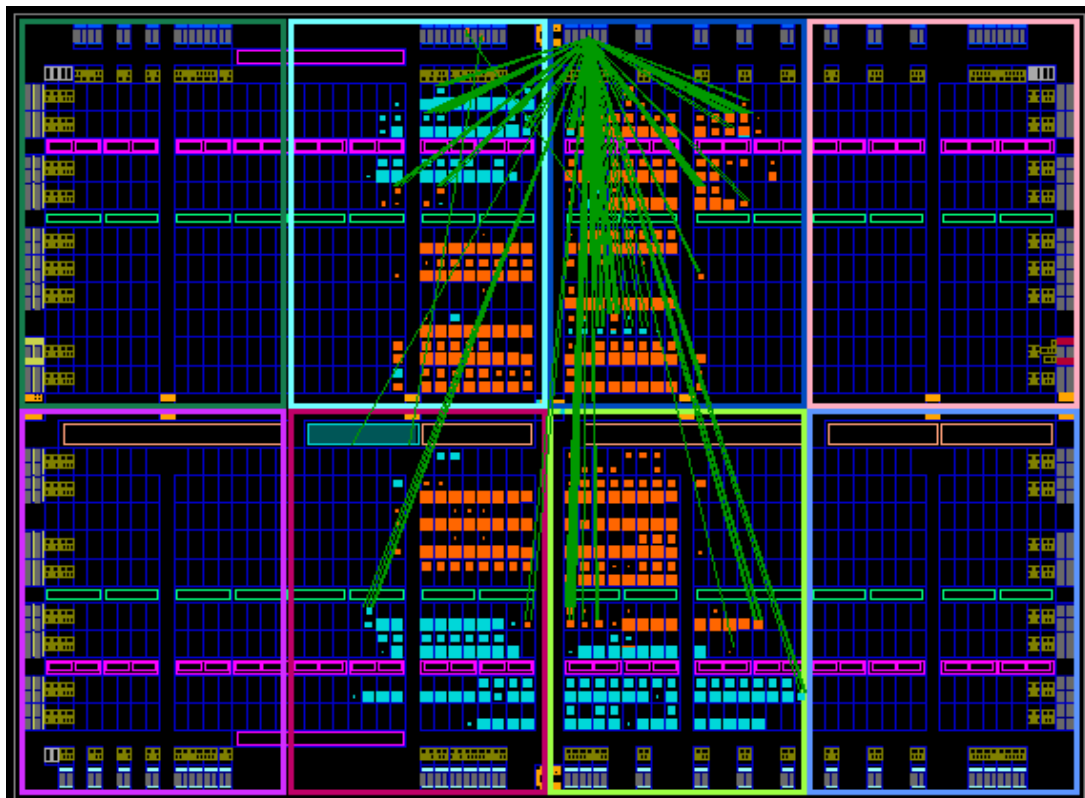


Figura 6.168. Esquema del circuito usando PlanAhead y aplicando modificaciones

6.6. Número de elementos

En este apartado se muestra el número de elementos utilizados para cada tipo de benchmark y estrategia de diseño, como por ejemplo, slice, que son las unidades elementales de las FPGAs y contienen los elementos básicos para implementar la lógica, LUTs, memorias, multiplexores y lógica aritmética.

6.6.1. Multiplicadores embebidos optimizados en área

Balanceado

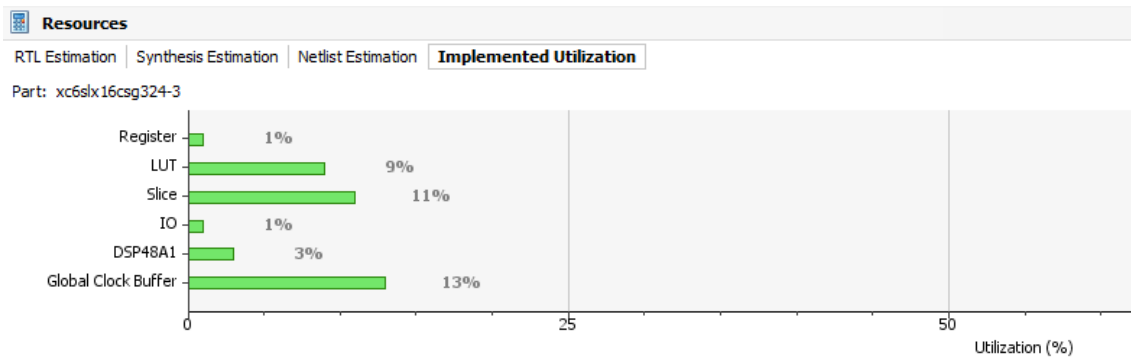


Figura 6.169. Porcentaje de utilización de los elementos disponibles para multiplicadores embebidos optimizados en área

Optimización de potencia

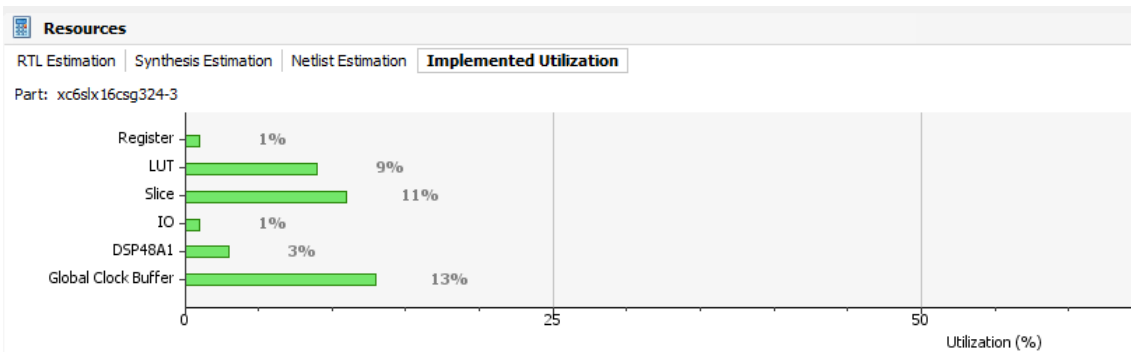


Figura 6.170. Porcentaje de utilización de los elementos disponibles para multiplicadores embebidos optimizados en área

Rendimiento temporal

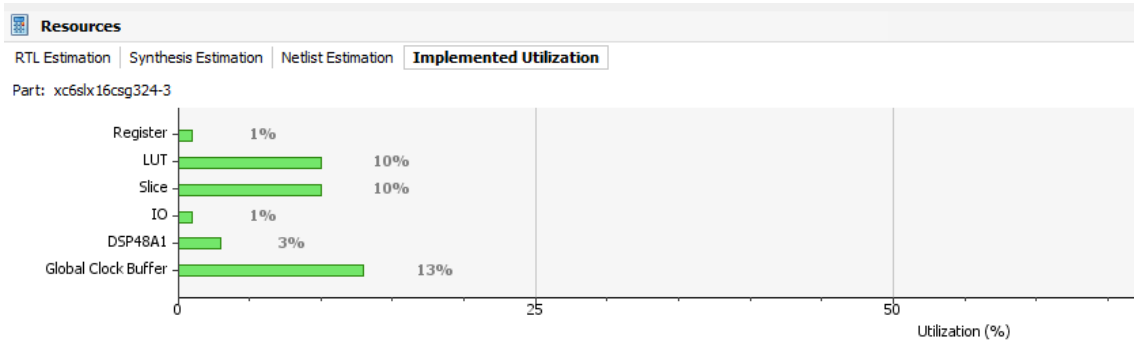


Figura 6.171. Porcentaje de utilización de los elementos disponibles para multiplicadores embebidos optimizados en área

Área

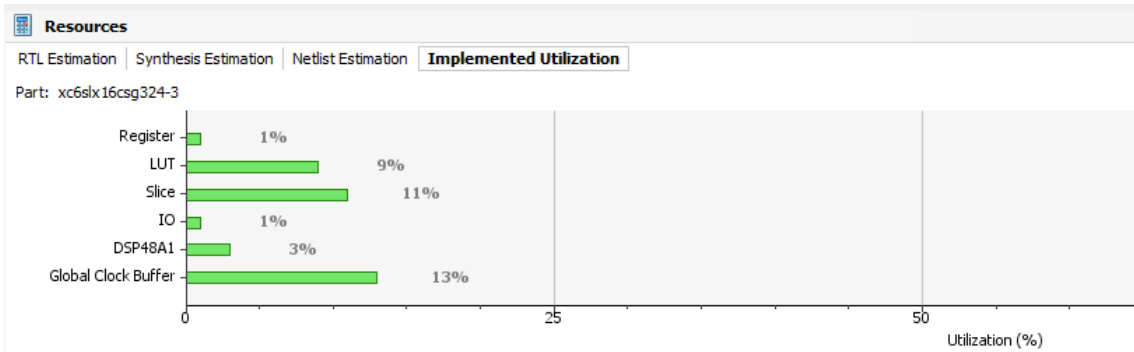


Figura 6.172. Porcentaje de utilización de los elementos disponibles para multiplicadores embebidos optimizados en área

Tiempo de ejecución

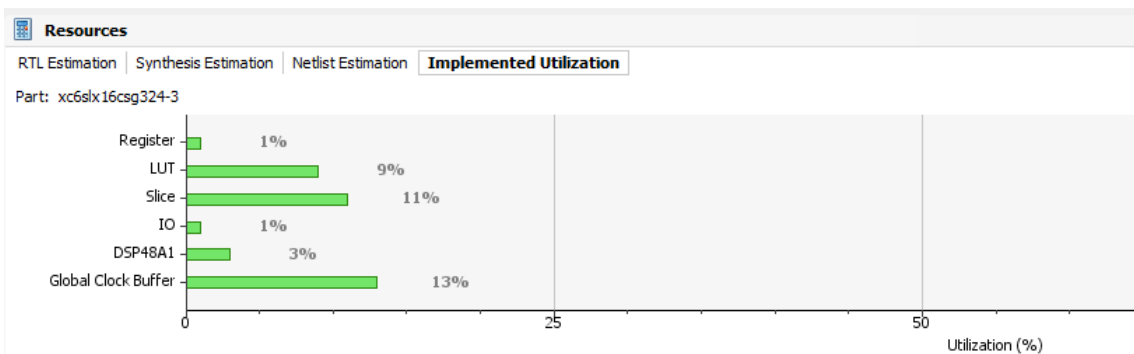


Figura 6.173. Porcentaje de utilización de los elementos disponibles para multiplicadores embebidos optimizados en área

6.6.2. Multiplicadores embebidos optimizados en velocidad

Balanceado

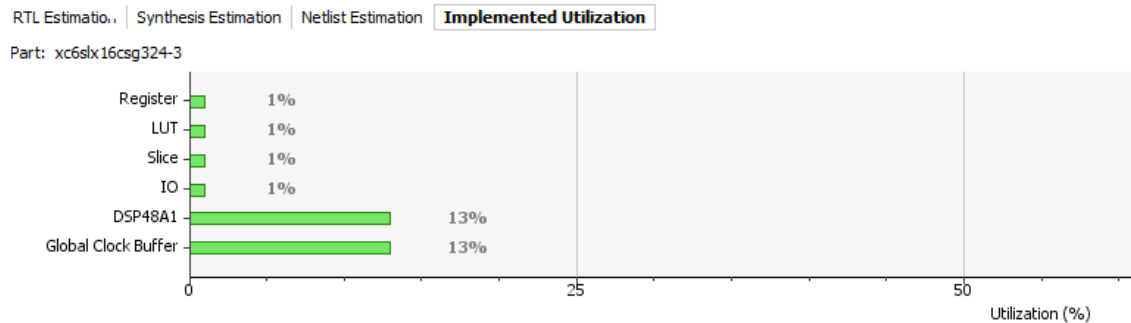


Figura 6.174. Porcentaje de utilización de los elementos disponibles para multiplicadores embebidos optimizados en velocidad

Optimización de potencia

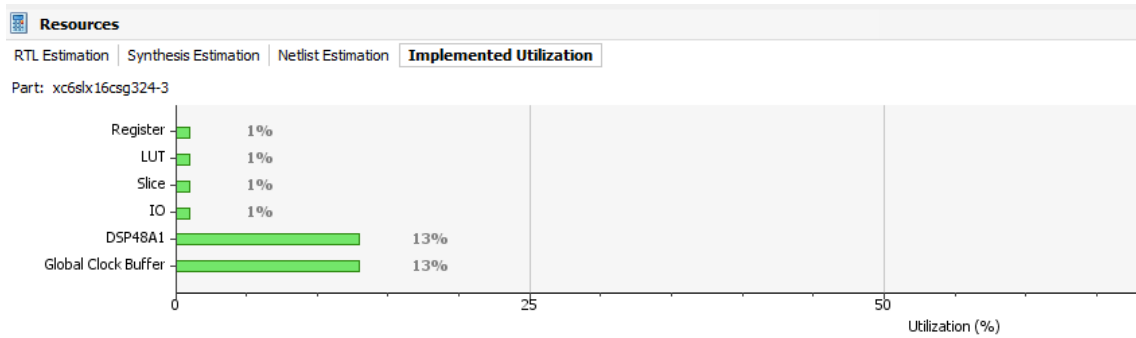


Figura 6.175. Porcentaje de utilización de los elementos disponibles para multiplicadores embebidos optimizados en velocidad

Rendimiento temporal

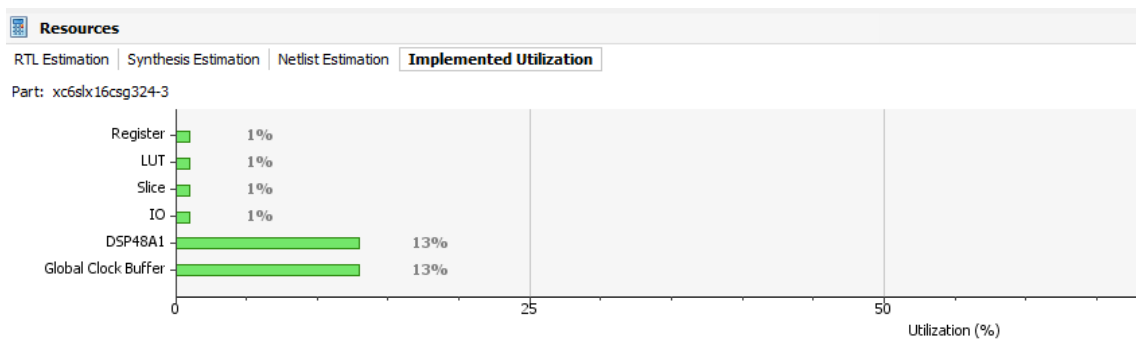


Figura 6.176. Porcentaje de utilización de los elementos disponibles para multiplicadores embebidos optimizados en velocidad

Área

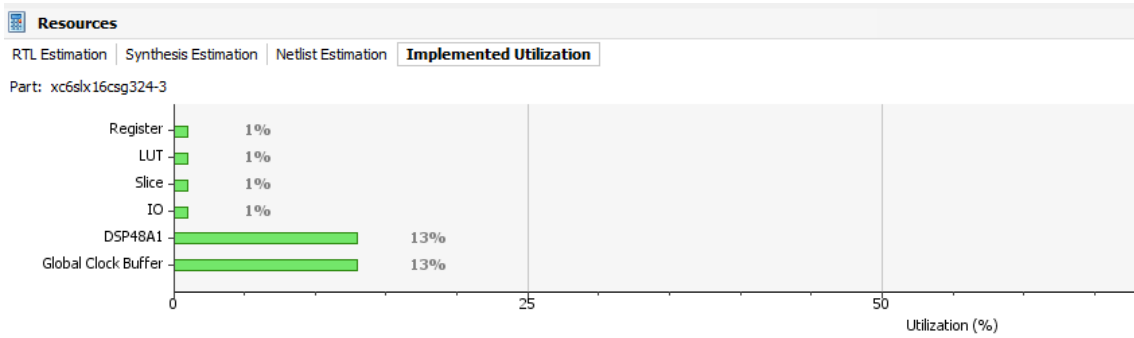


Figura 6.177. Porcentaje de utilización de los elementos disponibles para multiplicadores embebidos optimizados en velocidad

Tiempo de ejecución

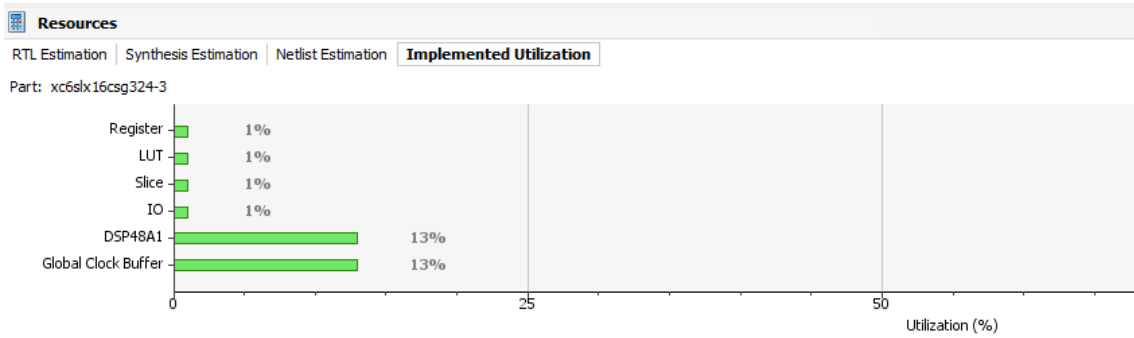


Figura 6.178. Porcentaje de utilización de los elementos disponibles para multiplicadores embebidos optimizados en velocidad

6.6.3. Multiplicadores usando LUTs optimizados en área

Balanceado

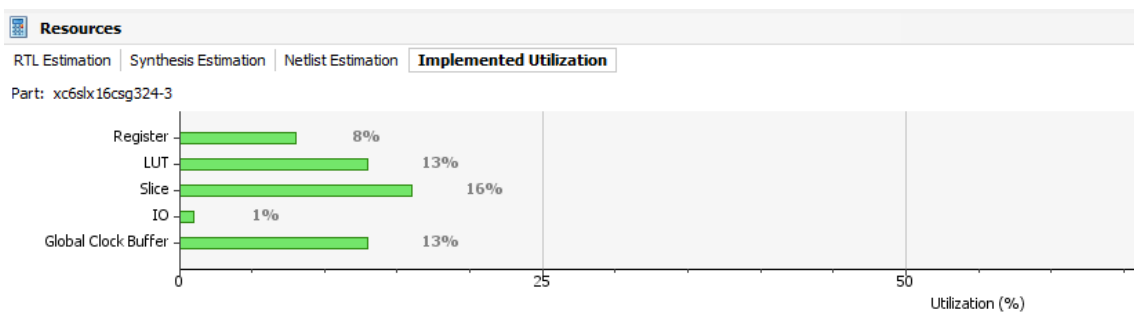


Figura 6.179. Porcentaje de utilización de los elementos disponibles para multiplicadores usando LUTs optimizados en área

Optimización de potencia

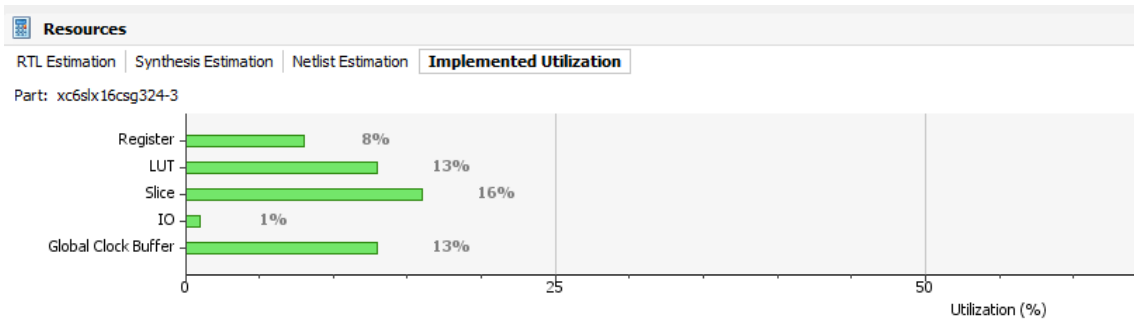


Figura 6.180. Porcentaje de utilización de los elementos disponibles para multiplicadores usando LUTs optimizados en área

Rendimiento temporal

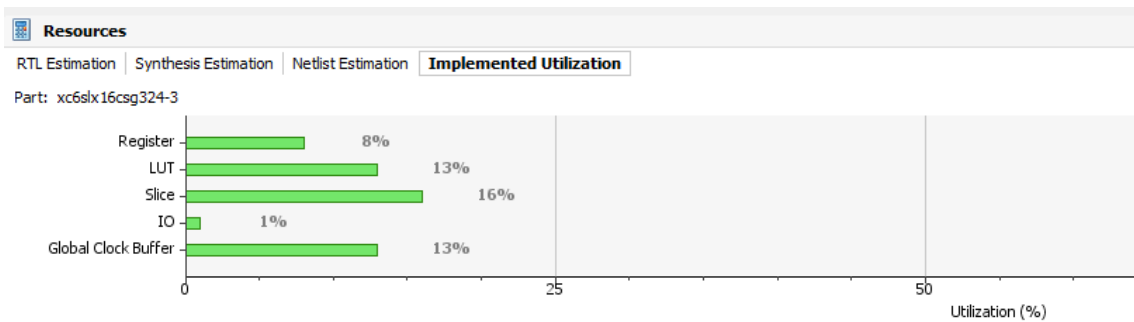


Figura 6.181. Porcentaje de utilización de los elementos disponibles para multiplicadores usando LUTs optimizados en área

Área

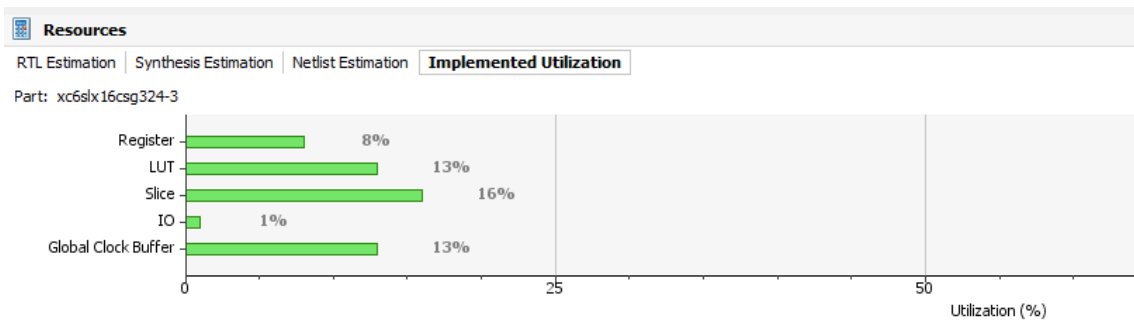


Figura 6.182. Porcentaje de utilización de los elementos disponibles para multiplicadores usando LUTs optimizados en área

Tiempo de ejecución

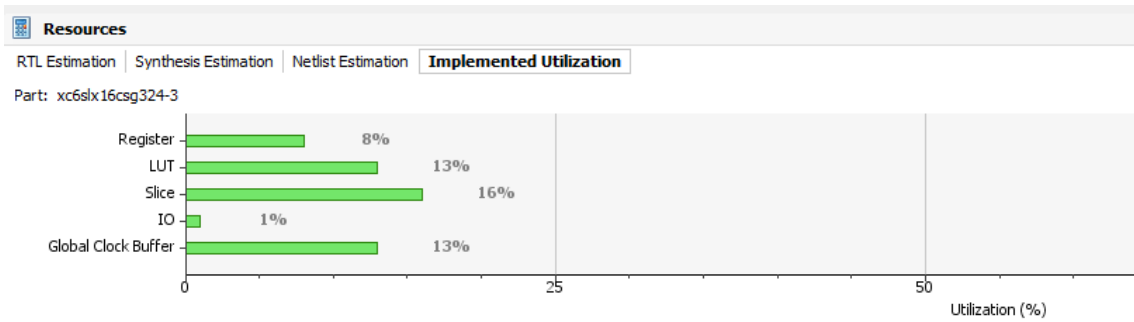


Figura 6.183. Porcentaje de utilización de los elementos disponibles para multiplicadores usando LUTs optimizados en área

6.6.4. Multiplicadores usando LUTs optimizados en velocidad

Balanceado

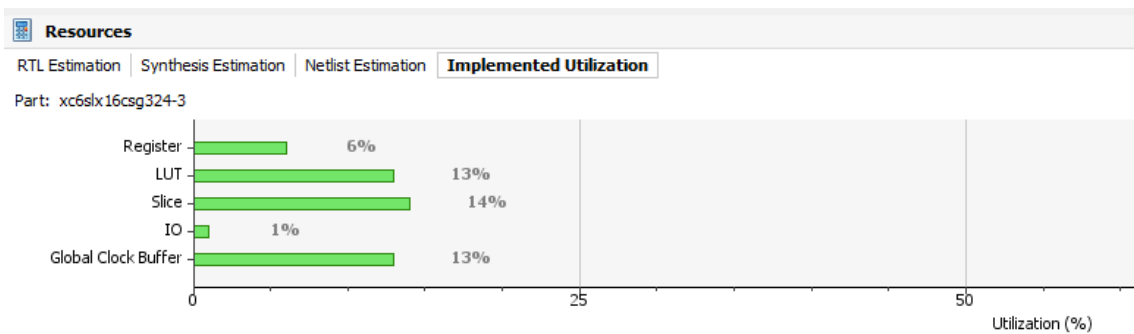


Figura 6.184. Porcentaje de utilización de los elementos disponibles para multiplicadores usando LUTs optimizados en velocidad

Optimización de potencia

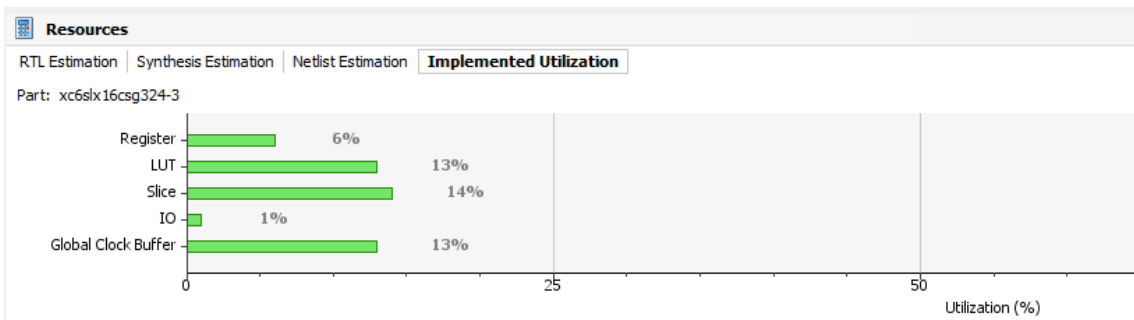


Figura 6.185. Porcentaje de utilización de los elementos disponibles para multiplicadores usando LUTs optimizados en velocidad

Rendimiento temporal

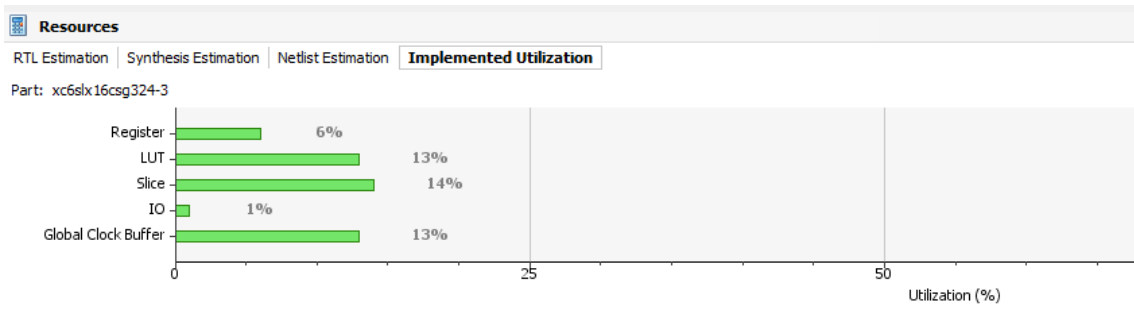


Figura 6.186. Porcentaje de utilización de los elementos disponibles para multiplicadores usando LUTs optimizados en velocidad

Área

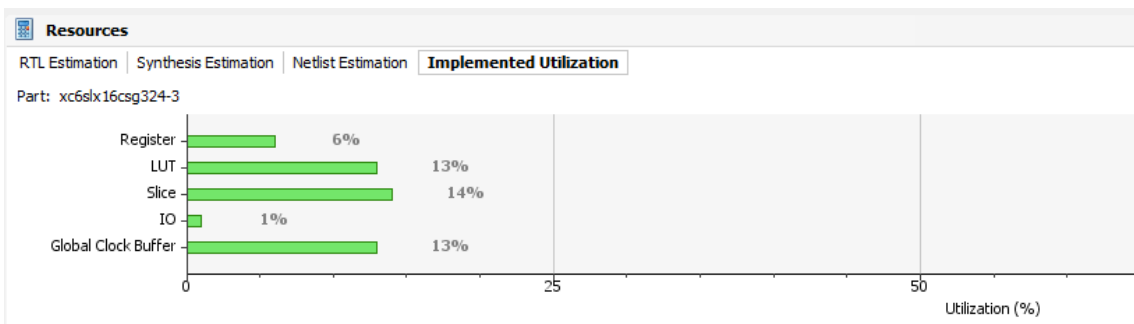


Figura 6.187. Porcentaje de utilización de los elementos disponibles para multiplicadores usando LUTs optimizados en velocidad

Tiempo de ejecución

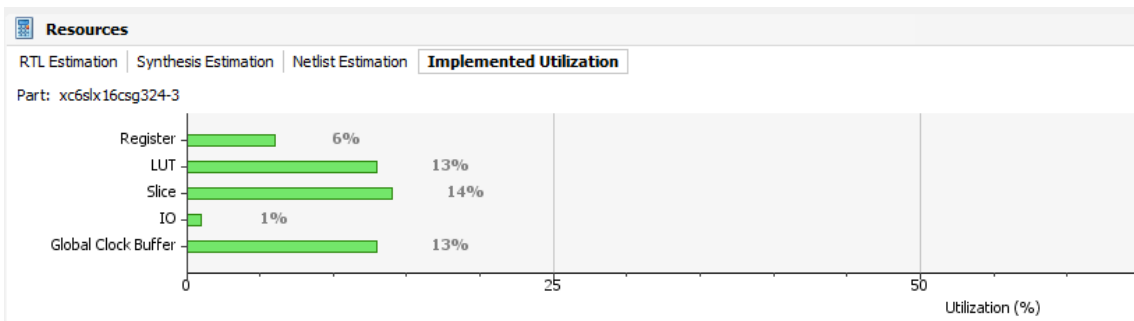


Figura 6.188. Porcentaje de utilización de los elementos disponibles para multiplicadores usando LUTs optimizados en velocidad

6.6.5. Multiplicador segmentado

Balanceado

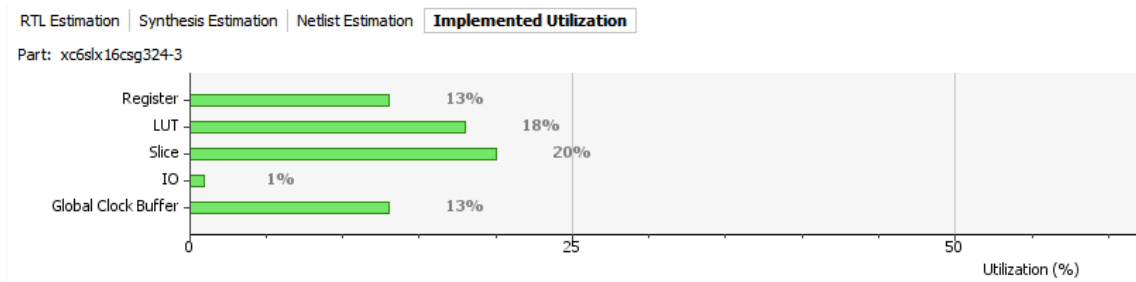


Figura 6.189. Porcentaje de utilización de los elementos disponibles para el multiplicador segmentado

Optimización de potencia

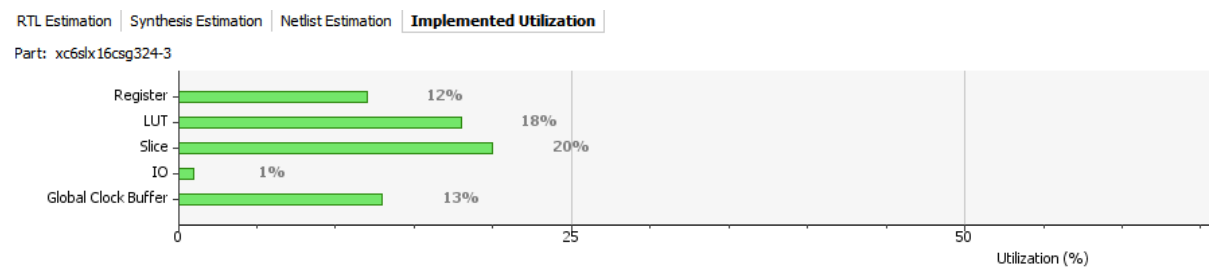


Figura 6.190. Porcentaje de utilización de los elementos disponibles para el multiplicador segmentado

Rendimiento temporal

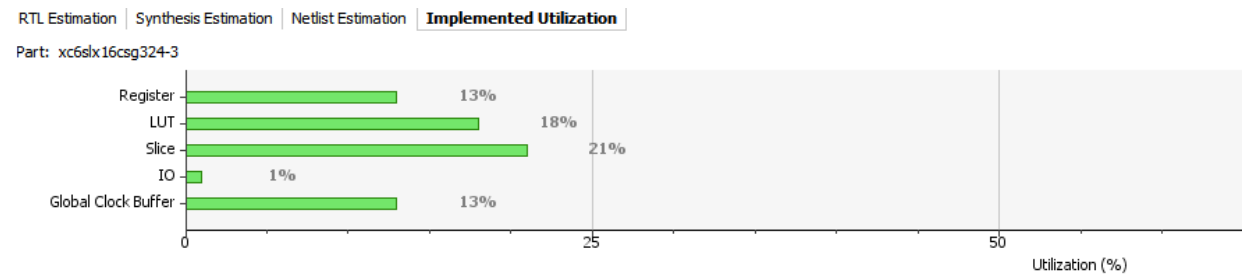


Figura 6.191. Porcentaje de utilización de los elementos disponibles para el multiplicador segmentado

Área

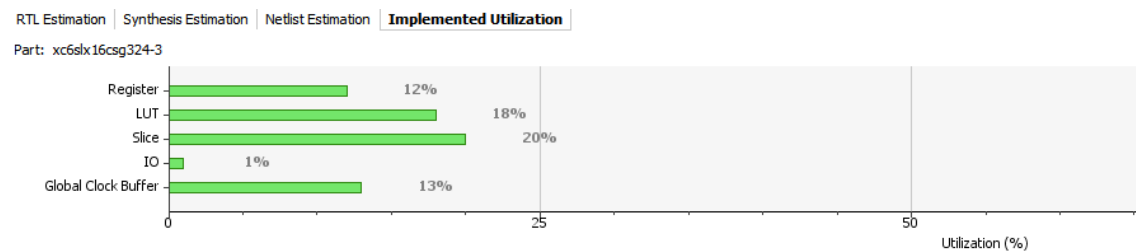


Figura 6.192. Porcentaje de utilización de los elementos disponibles para el multiplicador segmentado

Tiempo de ejecución

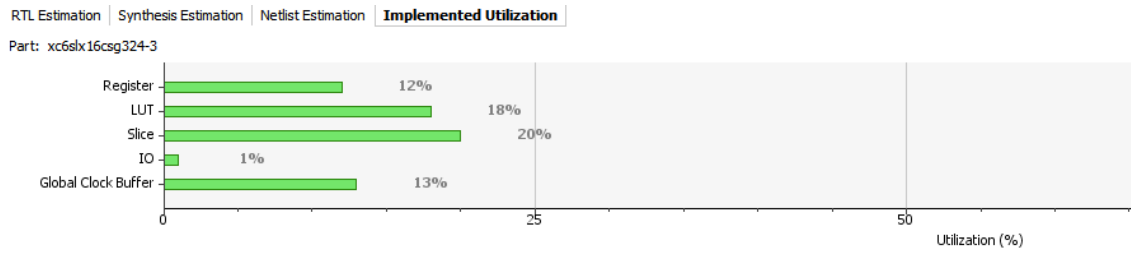


Figura 6.193. Porcentaje de utilización de los elementos disponibles para el multiplicador segmentado

6.6.6. AES

Balanceado

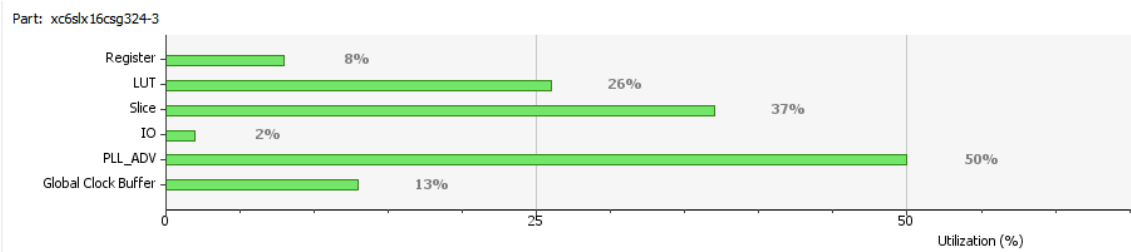


Figura 6.194. Porcentaje de utilización de los elementos disponibles para el AES

Área

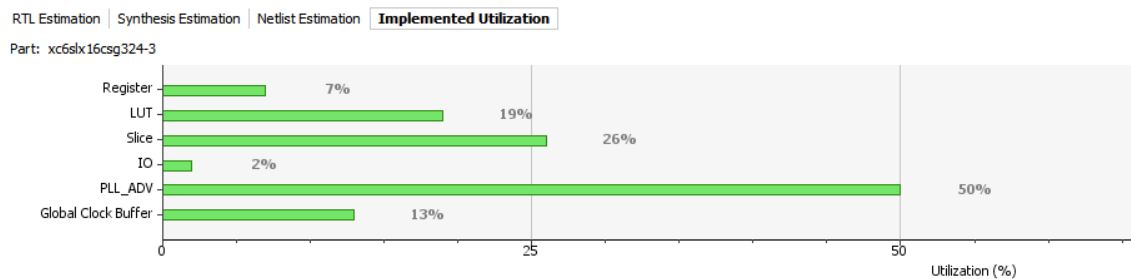


Figura 6.195. Porcentaje de utilización de los elementos disponibles para el AES

Optimización de potencia

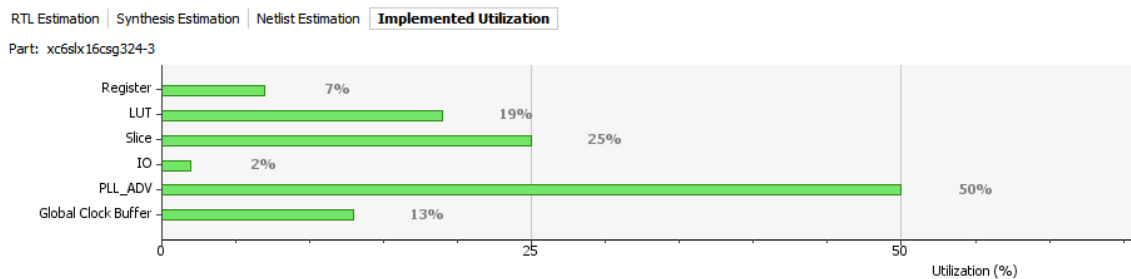


Figura 6.196. Porcentaje de utilización de los elementos disponibles para el AES

Rendimiento temporal

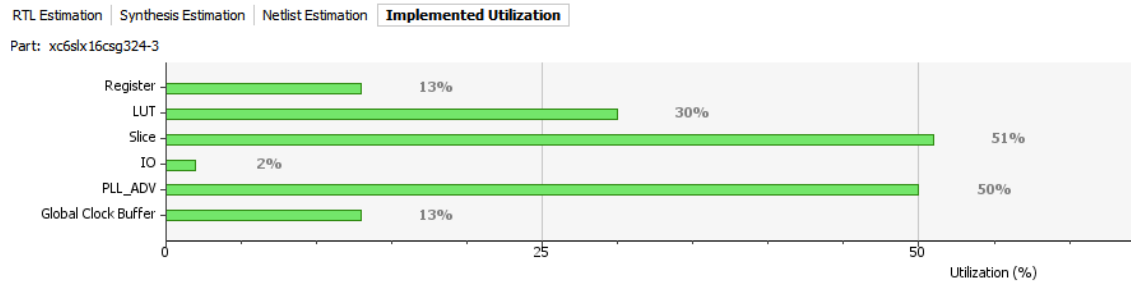


Figura 6.197. Porcentaje de utilización de los elementos disponibles para el AES

6.6.7. FFT

Balanceado

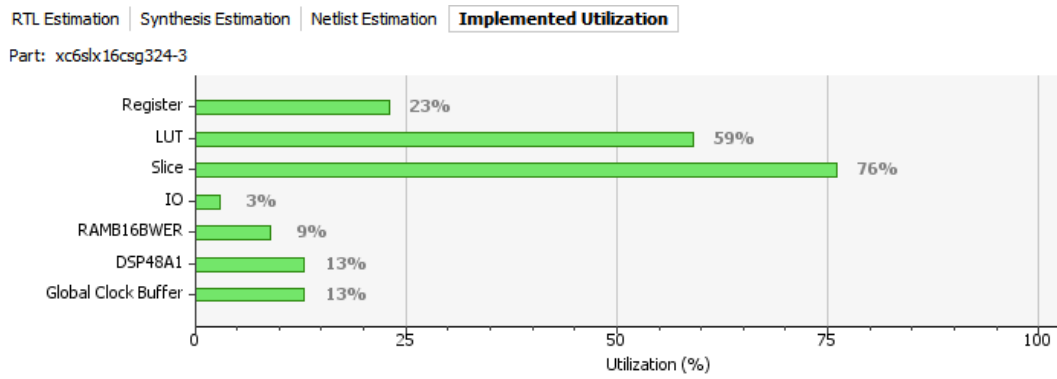


Figura 6.198. Porcentaje de utilización de los elementos disponibles para el FFT

Área

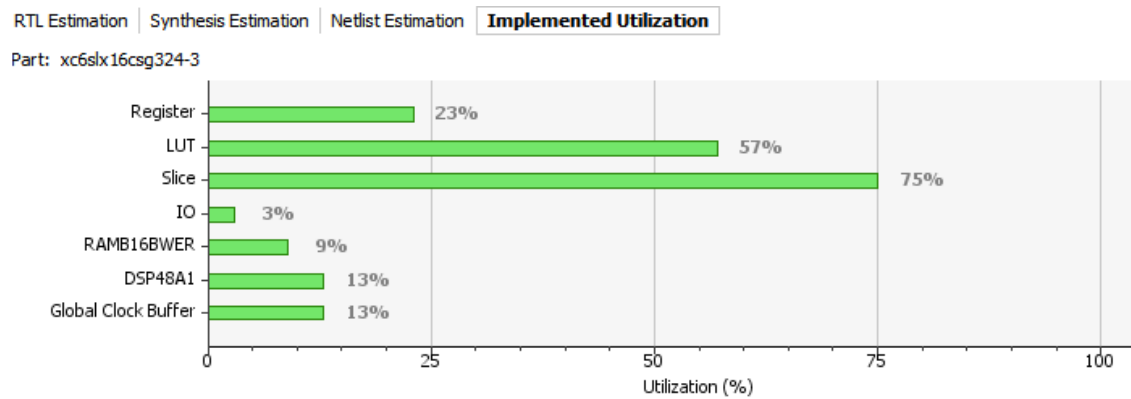


Figura 6.199. Porcentaje de utilización de los elementos disponibles para el FFT

Optimización de potencia

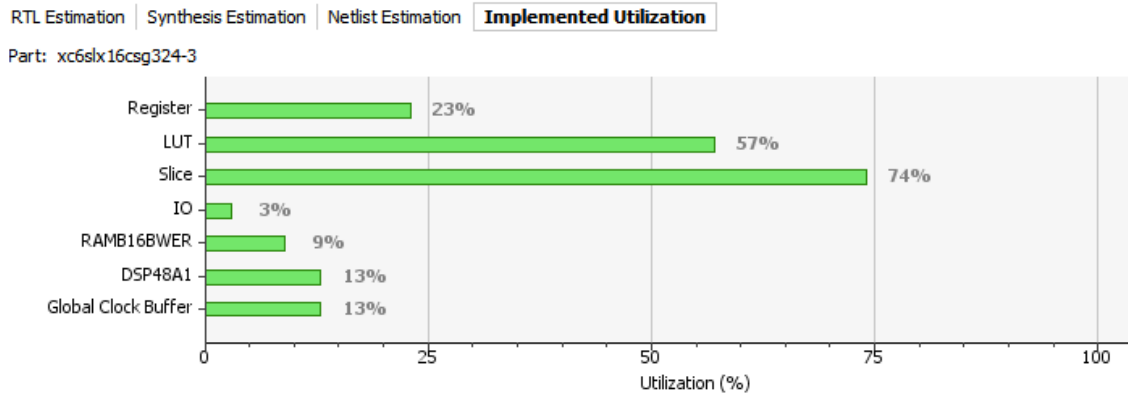


Figura 6.200. Porcentaje de utilización de los elementos disponibles para el FFT

Rendimiento temporal

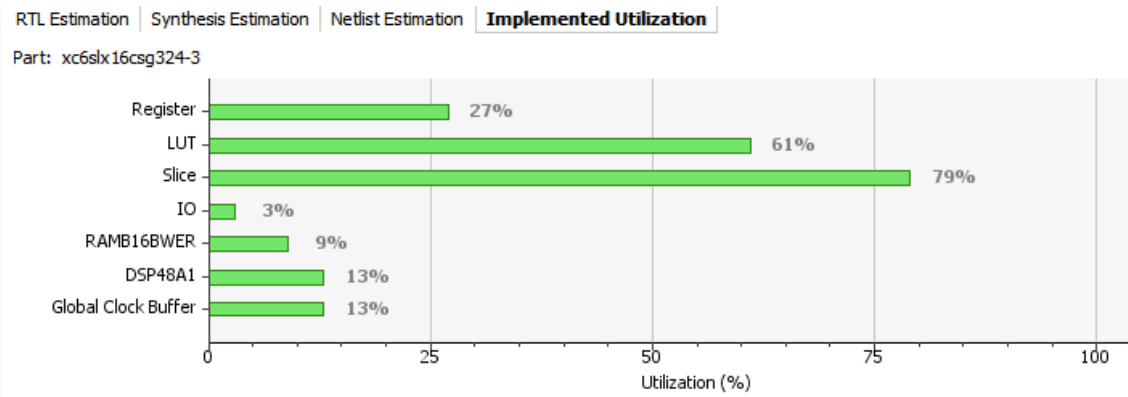


Figura 6.201. Porcentaje de utilización de los elementos disponibles para el FFT

6.7. Frecuencia

6.7.1. Multiplicadores Xilinx

Para estos multiplicadores se muestran los resultados de la frecuencia de síntesis obtenidos en función de las distintas estrategias de diseños.

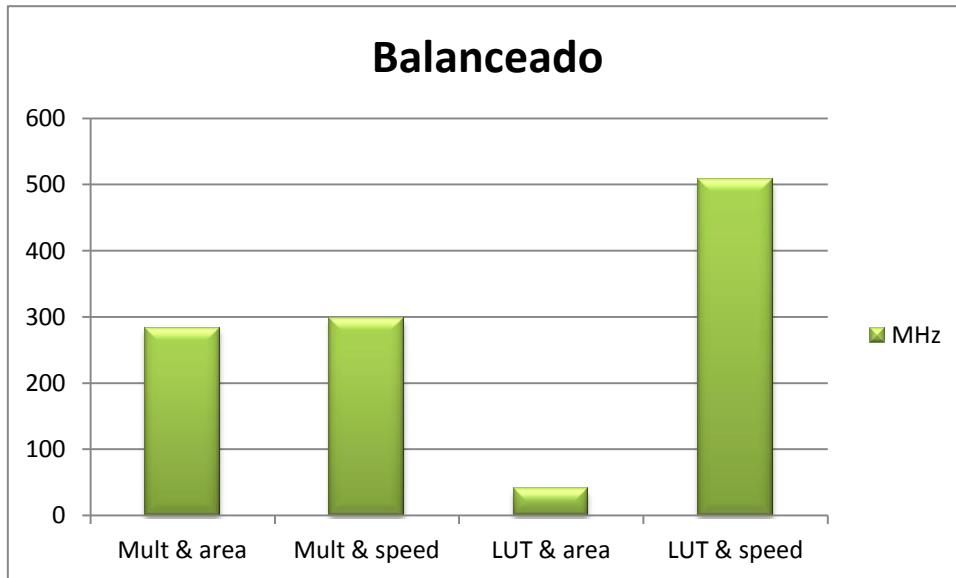


Figura 6.202. Comparación de la frecuencia de síntesis

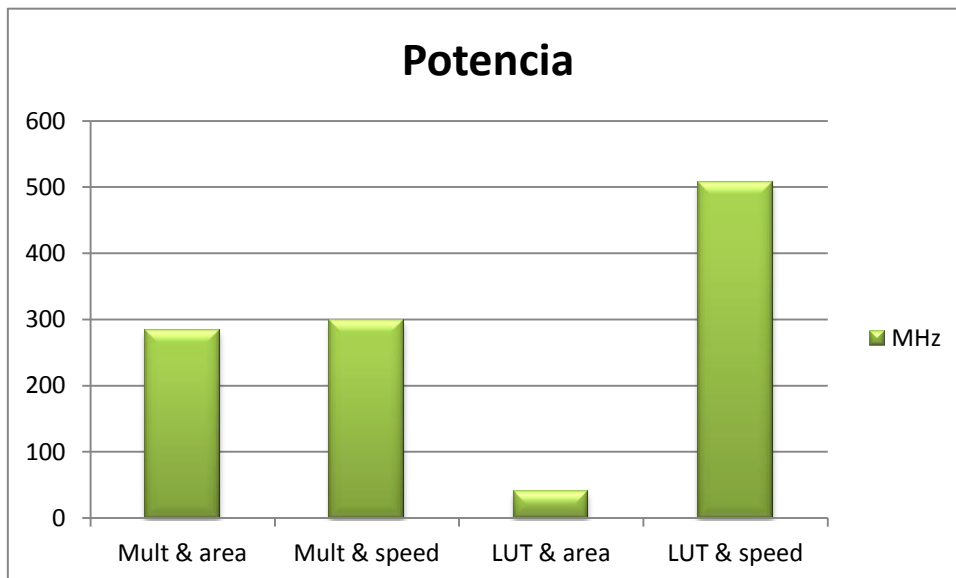


Figura 6.203. Comparación de la frecuencia de síntesis

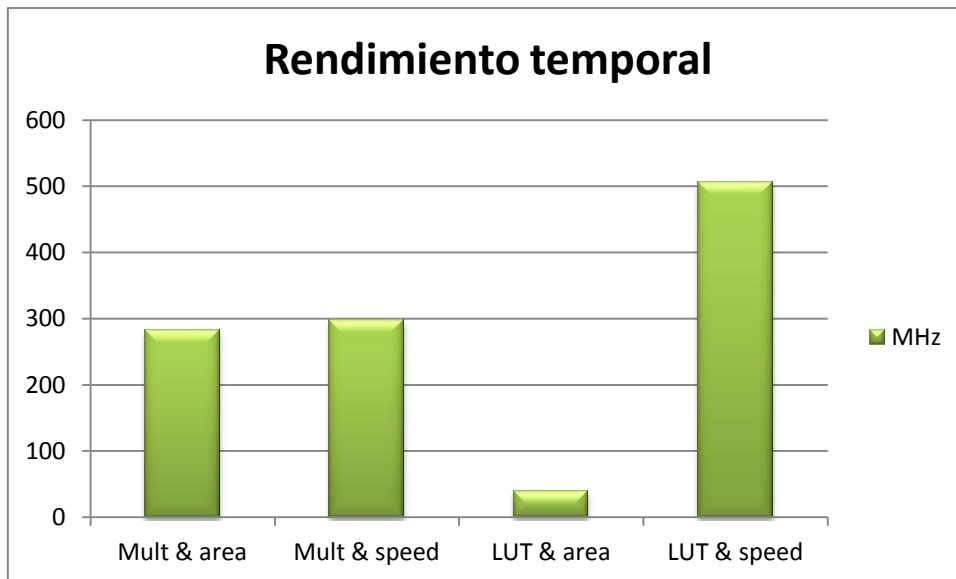


Figura 6.204. Comparación de la frecuencia de síntesis

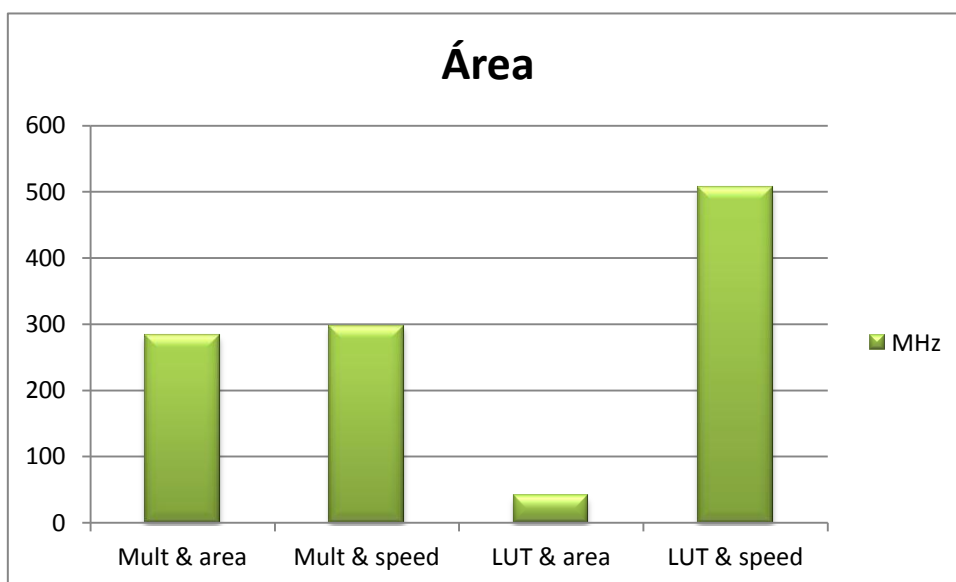


Figura 6.205. Comparación de la frecuencia de síntesis

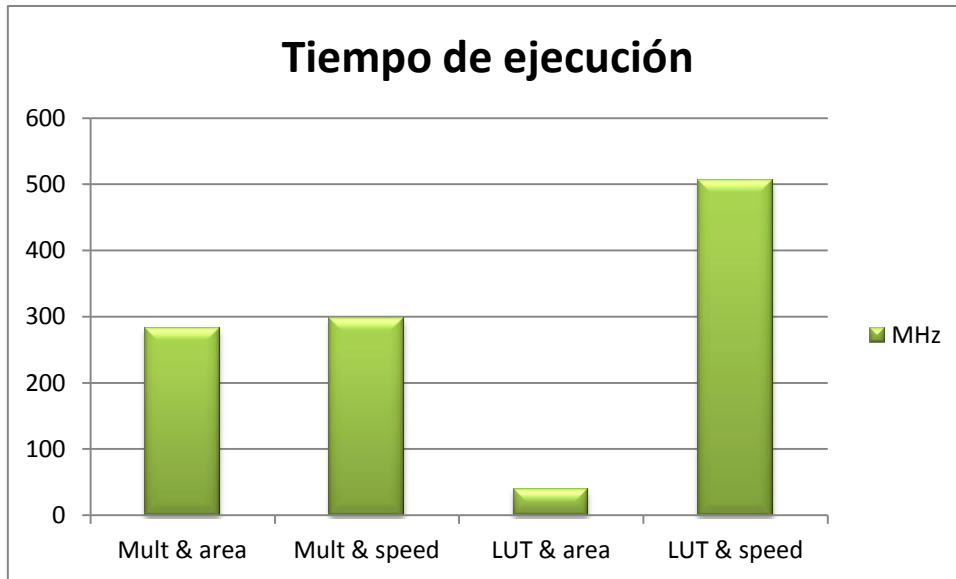


Figura 6.206. Comparación de la frecuencia de síntesis

6.7.2. Multiplicador segmentado

En la siguiente figura se muestran los resultados comparativos entre las cinco estrategias de diseño.

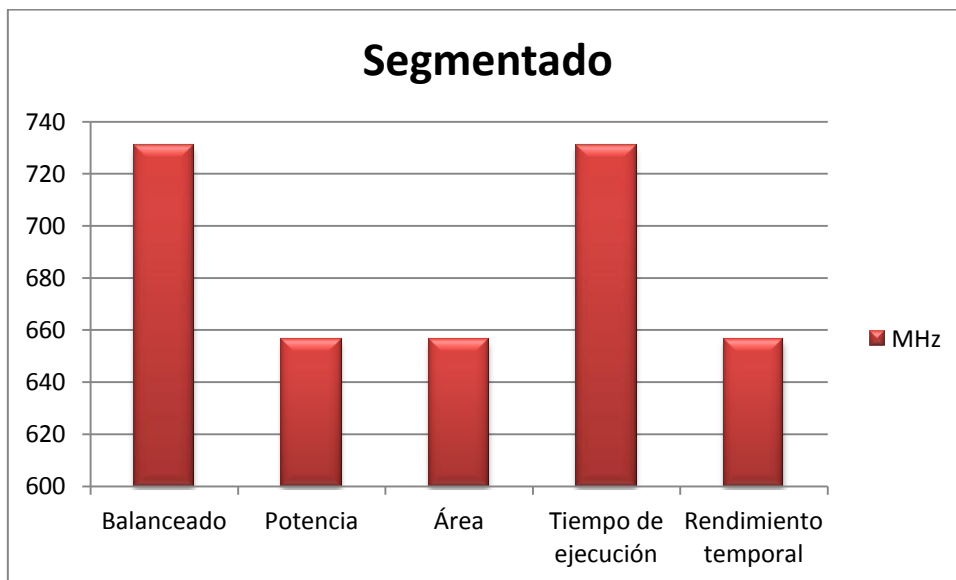


Figura 6.207. Comparación de la frecuencia de síntesis

6.7.3. AES

En la siguiente figura se muestran los resultados comparativos entre las cuatro estrategias de diseño.

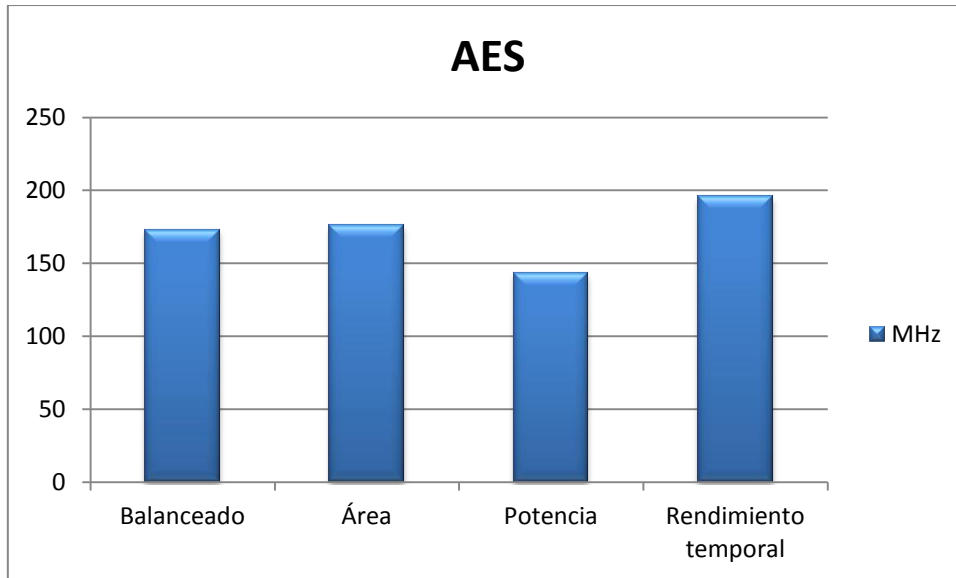


Figura 6.208. Comparación de la frecuencia de síntesis

6.7.4. FFT

En la siguiente figura se muestran los resultados comparativos entre las cuatro estrategias de diseño.

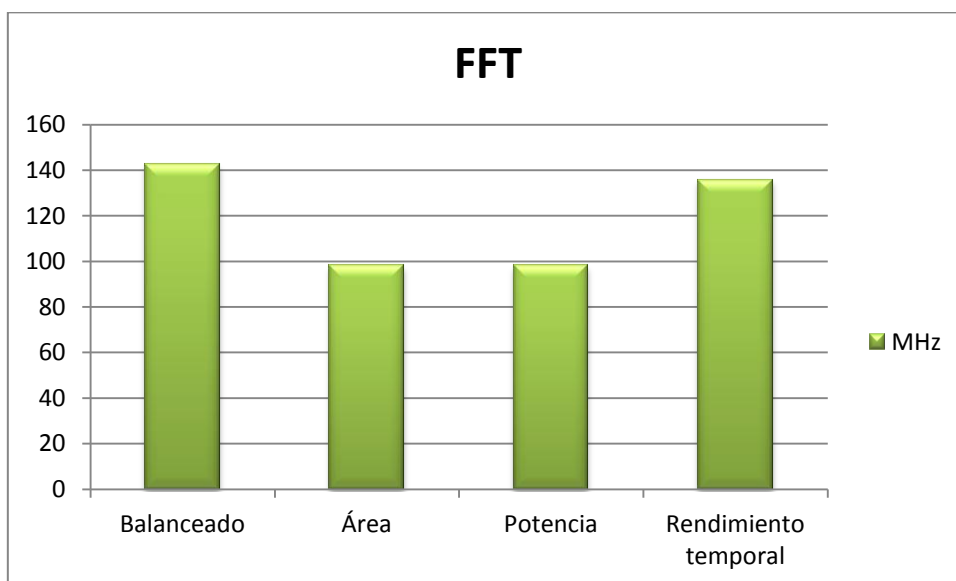


Figura 6.209. Comparación de la frecuencia de síntesis

6.8. Temperatura

En este apartado se incluye un importante tipo de estimación realizada con *XPower Analyzer*, variar la temperatura de ambiente. Con este tipo de modificación se puede obtener importantes reducciones de consumo. Solo se ha realizado las estimaciones con *Xpower Analyzer* debido a la dificultad de realizar este tipo de medidas sobre el circuito real ya que se ha simulado variaciones de temperatura de entre 0°C y 85 °C. El circuito elegido ha sido el multiplicador de Xilinx construido a partir de LUTs y optimizando la velocidad y usando la estrategia de diseño balanceada.

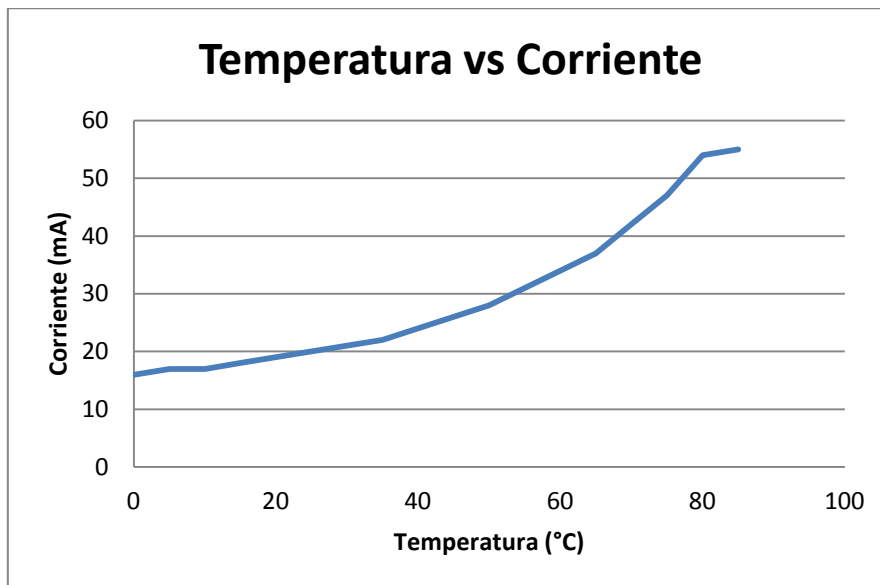


Figura 6.210. Aumento de consumo variando la temperatura ambiente

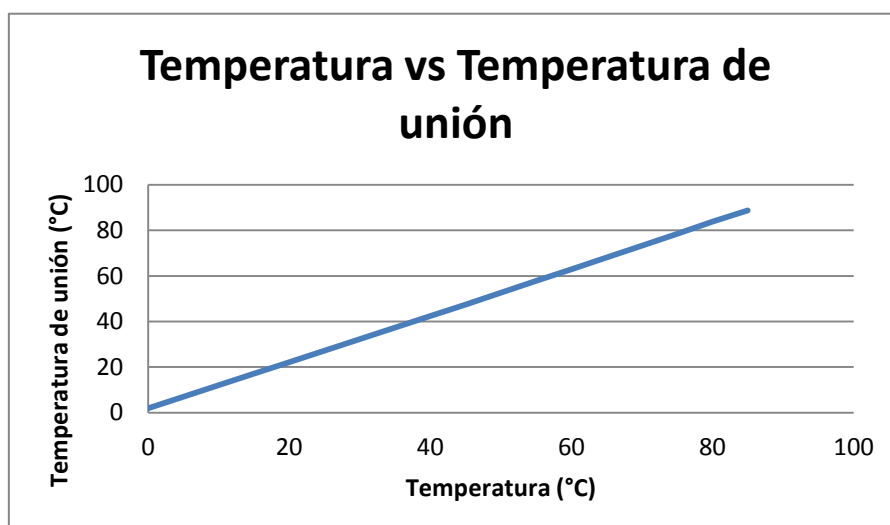


Figura 6.211. Aumento de la temperatura de unión variando la temperatura ambiente

Para el caso de 85°C de temperatura ambiente, la temperatura de unión es de 88°C lo cual supera el máximo permitido de 85°C de temperatura de unión. En este caso la fiabilidad y otras características no pueden ser garantizada que sus medidas sean correctas. Con el fin de reducir la temperatura de unión se pueden aplicar técnicas para enfriar el sistema y/o técnicas de diseño de bajo consumo. En este apartado se ha usado un flujo de aire para reducir la temperatura del sistema y poder mantener la temperatura de unión dentro de los límites permitidos.

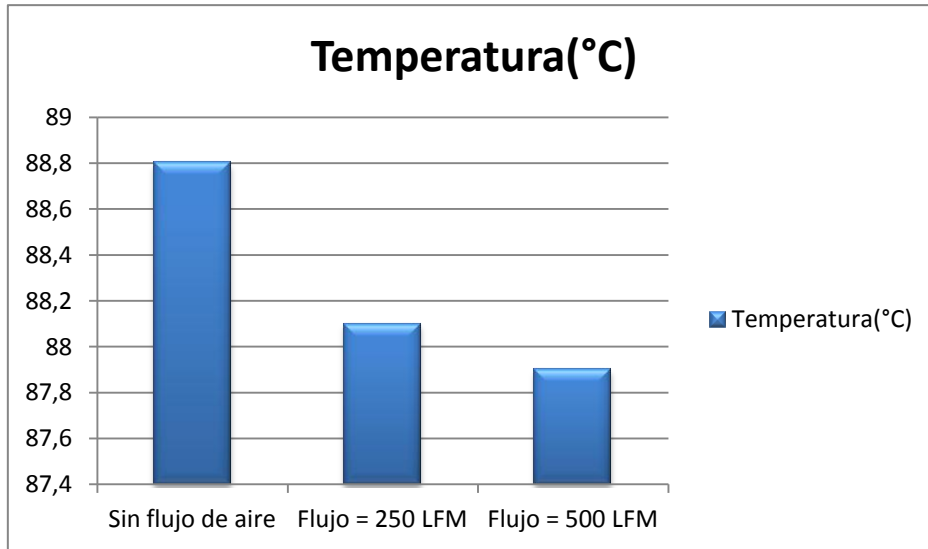


Figura 6.212. Reducción de la temperatura de unión aumentando LFM

LFM= Linear Feet per Minute. El *XPower Analyzer* no muestra mejoras en el consumo del circuito entre el caso no usar flujo de aire y los casos en los que si usa.

6.9. Condensador de carga

En este apartado se trata el caso de variar el condensador de carga (C_L). Las estimaciones se han realizado con *XPower Analyzer*, para valores del condensador de 0.1, 5, 10, 15 y 20 pF. El valor por defecto es de 5 pF.

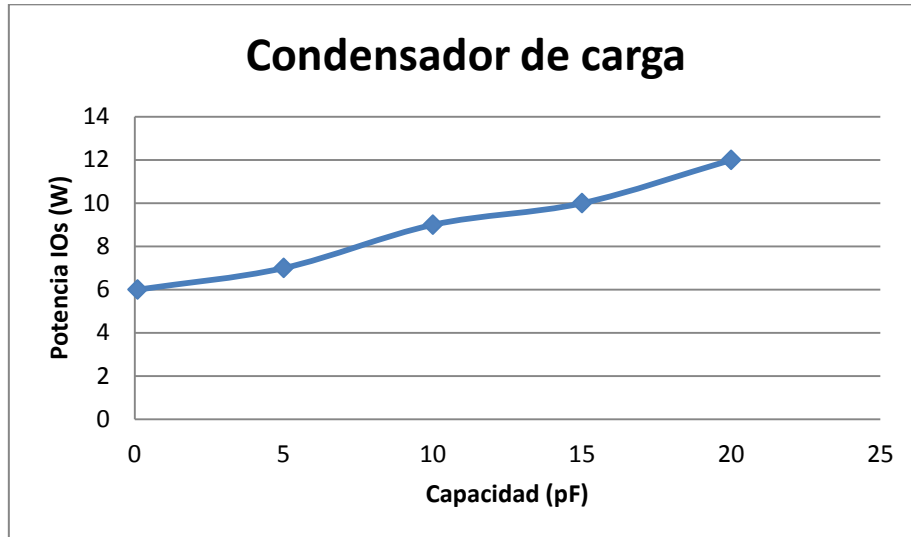


Figura 6.213. Aumento de la potencia IOs variando el condensador de carga (C_L)

Se observa que para mayores valores de C_L aumenta el consumo de los dispositivos de IOs del sistema, los cuales están relacionados con la potencia dinámica del sistema completo (no solo la FPGA), por tanto, se produce un aumento de esta.

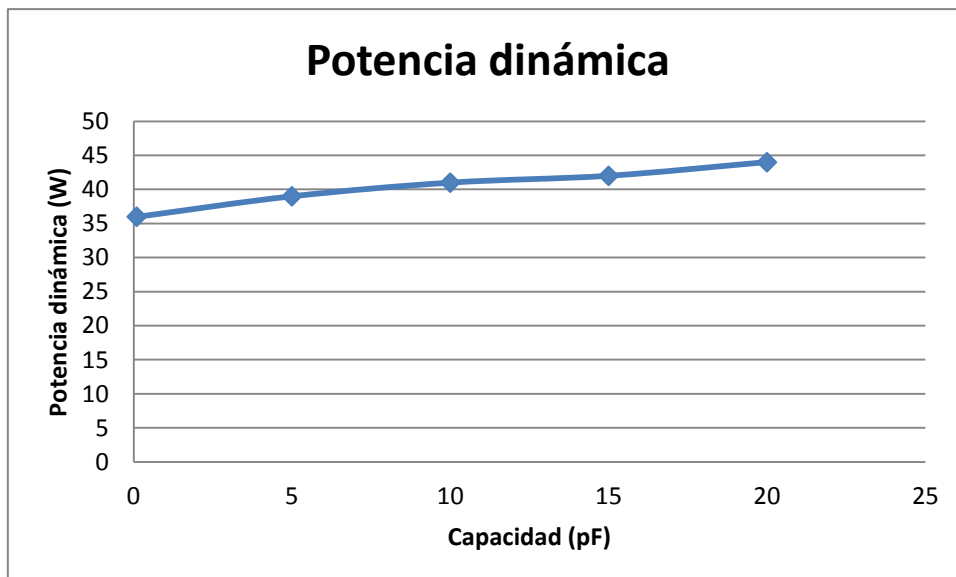


Figura 6.214. Aumento de la potencia dinámica variando el condensador de carga (C_L)

7. Análisis de los resultados

En este capítulo se va a realizar la evaluación de los resultados obtenidos así como interpretar y analizar cuáles de entre todas las optimizaciones aplicadas son más eficientes potencialmente aplicables.

El primer paso será ver cuáles han sido los casos de éxito para cada circuito y en qué porcentaje se ha producido un ahorro en el consumo de potencia. La estrategia de diseño de referencia es balanceado

7.1. Multiplicadores Xilinx

7.1.1. ISE

Comparación de los resultados obtenidos usando *ISE* para las diferentes estrategias de diseño y midiendo directamente sobre la FPGA. Los resultados son en mW. El fondo verde significa que se ha obtenido una mejora con respecto a la estrategia balanceada.

Estrategia de diseño	Mult & área	Mult & velocidad	LUT & área	LUT & velocidad
Balanceado	21,874	4,08	19,8696	18,6
Área	22,176	4,416	19,944	19,176
Potencia	20,784 (-4,98%)	4,152	19,908	18,336 (-1,42%)
Rendimiento temporal	20,088 (-8,16%)	4,032 (-1,18%)	19,8 (-0,35%)	18,84
Tiempo de ejecución	21,528 (-1,58%)	4,224	20,196	19,776

Tabla 7.1. Comparativa de consumos en ISE

A partir de esto la opción con más posibilidad de éxito es usar la estrategia de diseño de rendimiento temporal, seguido de la de optimización de potencia.

7.1.2. Floorplanning con PlanAhead

A continuación se muestran las mejoras producidas mediante *floorplanning* usando *PlanAhead*, primero para el caso de aplicar la estrategia balanceada (resultados del apartado 6.5.1.1). Resultados en mW.

LUT & área	
Inicial	20,04
1ª modificación	20,16
2ª modificación	19,968 (-0.36%)

Tabla 7.2. Comparativa de consumos en PlanAhead aplicando *floorplanning*

LUT & velocidad	
Inicial	18,768
1ª modificación	18,36 (-2.17%)
2ª modificación	18,264 (-2.68%)
3ª modificación	19,056
4ª modificación	19,152
5ª modificación	19,176
6ª modificación	18,84
7ª modificación	18,696 (-0.38%)

Tabla 7.3. Comparativa de consumos en PlanAhead aplicando *floorplanning*

Mult & área	
Inicial	21,432
1ª modificación	21,504
2ª modificación	21,864
3ª modificación	21,672

Tabla 7.4. Comparativa de consumos en PlanAhead aplicando *floorplanning*

Mult & velocidad	
Inicial	4,104
1ª modificación	3,936 (-4.09%)
2ª modificación	3,912 (-4.68%)

Tabla 7.5. Comparativa de consumos en PlanAhead aplicando *floorplanning*

Para el segundo caso se aplica la estrategia de diseño de optimización de potencia (resultados del apartado 6.5.1.2). Resultados en mW.

LUT & área	
Inicial	20,172
1ª modificación	20,172
2ª modificación	20,076 (-0.47%)

Tabla 7.6. Comparativa de consumos en PlanAhead aplicando *floorplanning*

LUT & velocidad	
Inicial	18,6
1ª modificación	18,528 (-0.39%)
2ª modificación	18,528 (-0.39%)
3ª modificación	18,552 (-0.26%)

Tabla 7.7. Comparativa de consumos en PlanAhead aplicando *floorplanning*

Mult & área	
Inicial	21,6
1ª modificación	21,552 (-0.22%)
2ª modificación	21,384 (-1%)
3ª modificación	21,624

Tabla 7.8. Comparativa de consumos en PlanAhead aplicando *floorplanning*

Mult & velocidad	
Inicial	3,816
1ª modificación	3,816

Tabla 7.9. Comparativa de consumos en PlanAhead aplicando *floorplanning*

7.2. Multiplicador segmentado

7.2.1. ISE

Comparación de las distintas medidas obtenidas para el multiplicador segmentado usando *ISE* y midiendo directamente sobre la FPGA. No se ha conseguido mejoras para ninguna estrategia de diseño. Los resultados son en mW.

Estrategia de diseño	Multiplicador segmentado
Balanceado	24,12
Área	25,968
Potencia	25,536
Rendimiento temporal	24,816
Tiempo de ejecución	25,056

Tabla 7.10. Comparativa de consumos en ISE

7.2.2. Floorplanning con PlanAhead

A continuación se muestran los resultados producidos mediante *floorplanning* usando *PlanAhead*. Primero para el caso de aplicar la estrategia balanceada (resultados del apartado 6.5.2.1). En este caso no se obtienen mejoras en el consumo de potencia. Resultados en mW.

Segmentado	
Inicial	24,168
1ª modificación	24,432
2ª modificación	24,696
3ª modificación	24,672
4ª modificación	24,672
5ª modificación	24,84

Tabla 7.11. Comparativa de consumos en PlanAhead aplicando *floorplanning*

Para el segundo caso se aplica la estrategia de diseño de optimización de potencia (resultados del apartado 6.5.2.2). Resultados en mW.

Segmentado	
Inicial	26,232
1ª modificación	26,256
2ª modificación	26,232
3ª modificación	26,112 (-0.46%)
4ª modificación	26,304

Tabla 7.12. Comparativa de consumos en PlanAhead aplicando *floorplanning*

7.3. AES (Advanced Encryption Standard)

Comparación de las distintas medidas obtenidas para el circuito de AES y midiendo mediante Avnet. El fondo verde significa que se ha obtenido una mejora con respecto a la estrategia balanceada.

Estrategia de diseño	Potencia (mW)
Balanceado	133,4652
Área	94,908 (-28,89%)
Potencia	97,734 (-26,77%)
Rendimiento temporal	153,9072

Tabla 7.13. Comparativa de consumos en ISE

Para este circuito, con las estrategias de diseño de optimización de área y optimización de potencia se obtienen notables resultados.

7.4. FFT (Fast Fourier Transform)

Comparación de las distintas medidas obtenidas para el circuito de FFT y midiendo mediante Avnet. El fondo verde significa que se ha obtenido una mejora con respecto a la estrategia balanceada.

Estrategia de diseño	Potencia (mW)
Balanceado	222,288
Área	218,3376 (-1,78%)
Potencia	216,9348 (-2.41%)
Rendimiento temporal	246,6516

Tabla 7.14. Comparativa de consumos en ISE

Para este circuito, con las estrategias de diseño de optimización de área y optimización de potencia se obtienen reducciones de consumo.

7.5. Temperatura

Con la técnica de simular una variación en la temperatura ambiental se consiguen importantes resultados en cuanto al ahorro de corriente, y por tanto de potencia. Por otra parte estos experimentos demuestran la importancia de controlar aumentos de temperatura ambiental que generan importantes aumentos de temperatura.

Temperatura	Potencia (mW)
25°C (ambiental)	24
0°C	19.2 (-20%)
15°C	21.6 (-10%)
20°C	22.8 (-5%)
40°C	28.8 (+20%)
60°C	40.8 (+70%)
85°C	66 (+175%)

Tabla 7.15. Comparativa de consumos en Xpower Analyzer

Se puede conseguir un ahorro del 20% de potencia, pero manteniendo una temperatura ambiental de 0°C lo cual es costoso. Para temperaturas ambientales menos costosas de crear artificialmente como 15°C y 20°C se consiguen ahorros del 10% y 5% respectivamente, lo cual es un ahorro significativo.

Por otro lado está el caso contrario. Para temperaturas ambientales de 40°C hay un aumento del 20% del consumo de potencia. Esta temperatura no es difícil de conseguir en ciertas épocas del año si el lugar de funcionamiento no está debidamente refrigerado. También hay que tener en cuenta el caso de que haya otros equipos funcionando en un mismo lugar simultáneamente, lo cual puede ser crítico sin un sistema de refrigeración, alcanzando temperaturas superiores a 40°C y aumentos del consumo de potencia de 70% para temperaturas ambientales de 60°C.

7.6. Condensador de carga C_L

El primer paso es recordar lo estudiado en el capítulo 3, un aumento del C_L genera una disminución de la corriente de cortocircuito (I_{sc}) y una disminución del C_L causa un aumento de la I_{sc} .

También es importante recordar la relación entre C_L y la potencia dinámica.

$$P_{dyn} = \frac{Energy}{transition} \cdot f = C_L \cdot V^2 \cdot P_{trans} \cdot f_{clock}$$

Por tanto, un aumento de la C_L genera una disminución de la I_{sc} pero causa un aumento de la potencia dinámica.

C_L (pF)	Potencia dinámica (mW)
5 (por defecto)	39
0.1	36 (-7.69%)
10	41
15	42
20	44

Tabla 7.16. Comparativa de consumos en Xpower Analyzer

En la reducción del C_L se debe tener un equilibrio, porque genera un aumento de la I_{sc} y puede ocurrir que en el balance global no salga rentable realizar la reducción de C_L .

La potencia disipada debido a las corrientes de cortocircuito se minimiza haciendo coincidir los tiempos de subida/bajada de la señal de entrada y de la de salida.

7.7. Potencia vs Frecuencia

En este apartado se analizará como en los casos en los que existe ahorro en el consumo de potencia (usando las medidas del *ISE*), habrá mejora en la frecuencia de síntesis del circuito o no.

7.7.1. Multiplicadores Xilinx

Multiplicadores	Optimización balanceada	Optimización de potencia	Optimización de rendimiento temporal	Optimización de tiempo de ejecución	Área
Mult y área	283,288 MHz	283,288 MHz	283,288 MHz	283,288 MHz	283,288 MHz
Mult y velocidad	297,845 MHz	297,845 MHz	297,845 MHz	297,845 MHz	297,845 MHz
LUT y área	40,19 MHz	40,19 MHz	40,19 MHz	40,19 MHz	40,19 MHz
LUT y velocidad	506,881 MHz	506,881 MHz	506,881 MHz	506,881 MHz	506,881 MHz

Tabla 7.17. Comparativa de frecuencias de síntesis

El fondo verde son los casos en los que existe ahorro de potencia. No se observa variación de la frecuencia de síntesis para ningún caso.

7.7.2. Multiplicador segmentado

Para este multiplicador no se realizan comparaciones de frecuencia, ya que no se consiguen mejoras en el consumo de potencia usando las estrategias de diseño del *ISE*.

7.7.3. AES

Estrategia de diseño	Frecuencia (MHz)
Balanceado	173,096
Área	176,01 (+1.68%)
Potencia	142,848

Tabla 7.18. Comparativa de frecuencias de síntesis

Se consigue un aumento de la frecuencia de síntesis del 1,68%, con un ahorro de potencia del 28,89% en el caso de la estrategia de diseño orientado a optimizar el área.

7.7.4. FFT

Estrategia de diseño	Frecuencia (MHz)
Balanceado	142,28
Área	98,006
Potencia	98,006

Tabla 7.19. Comparativa de frecuencias de síntesis

No se consigue una mejora en la frecuencia de síntesis para ninguno de los casos.

7.8. Comparación de diferentes técnicas de medición

En primer lugar se estudiará la relación entre la medida obtenida usando el multímetro de precisión y la obtenida mediante Avnet, la cual sigue la siguiente relación:

$$\text{Medida} = \text{Avnet} \cdot k + B$$

$$k \approx 1$$

$$B \approx 0$$

Con los datos experimentales obtenidos se llega a la siguiente conclusión:

$$k \approx 1.03$$

$$B = 0$$

Para el caso de la estrategia de diseño balanceada, la relación entre la medida obtenida usando el multímetro de precisión y los datos obtenidos mediante el *XPower Analyzer* depende del circuito sobre el que se aplica.

Multiplicadores Xilinx	
Circuito	Relación XPower/Medida
Mult & área	2.08
Mult & velocidad	0.88
LUT & área	1.27
LUT & velocidad	1.29

Tabla 7.20. Relación de las medidas obtenidas con XPower Analyzer y medidas sobre el circuito

Para el resto de estrategias de diseño esta relación es muy semejante. Teniendo en cuenta el redondeo que sufren los datos obtenidos mediante *XPower Analyzer* se puede considerar que las relaciones de la tabla anterior es común para todos los casos.

A continuación se muestran las relaciones para el resto de circuitos. Primero se muestra la relación para el multiplicador segmentado para la optimización de diseño optimizada. Esta relación se cumple para todas las estrategias de diseño.

Circuito	Relación XPower/Medida
Multiplicador segmentado	0.94

Tabla 7.21. Relación de las medidas obtenidas con *XPower Analyzer* y medidas sobre el circuito

Para los circuito AES y FFT solo se han tomado valores usando *XPower Analyzer* y *Avnet*, por lo que se ha aplicado la relación explicada:

$$Medida = Avnet \cdot k + B$$

Circuito	Relación XPower/Medida
AES & balanceado	0.99
AES & potencia	1.05
AES & área	1.05
AES & rendimiento temporal	0.98

Tabla 7.22. Relación de las medidas de obtenidas *XPower Analyzer* y de las medidas sobre el circuito AES

Circuito	Relación XPower/Medida
FFT & balanceado	1.08
FFT & potencia	1.2
FFT & área	1.18
FFT & rendimiento temporal	1.16

Tabla 7.23. Relación de las medidas de obtenidas *XPower Analyzer* y de las medidas sobre el circuito FFT

8. Conclusiones

El objetivo de este proyecto era estudiar los efectos de algunos *settings* de la herramienta sobre el consumo de potencia de ciertos circuitos, analizar la influencia del diseñador en el proceso de *floorplanning* y evaluar los errores de diferentes técnicas de medición.

Se han realizado 181 medidas de potencia divididas de la siguiente manera:

Circuito	Número de medidas
Multiplicadores Xilinx	139
Multiplicador segmentado	26
AES	8
FFT	8
Total	181

Tabla 8.1. Medidas realizadas

A estas medidas hay que sumar las 20 estimaciones realizadas variando la temperatura de ambiente y las 4 estimaciones variando el condensador de carga, lo que hace un **total de 205 medidas**.

- Mediante los *settings* de las herramienta se ha conseguido:
 - ✓ Multiplicadores Xilinx:
 - **Reducción** del consumo de potencia para **37,5% de los casos**.
 - Hasta un **8.16% de reducción** del consumo de potencia.
 - ✓ Multiplicador segmentado:
 - No se han conseguido mejoras para ninguna estrategia de diseño.
 - ✓ AES:
 - **Reducción** del consumo de potencia para el **66% de los casos**.
 - Hasta un **28.89% de reducción** del consumo de potencia.
 - ✓ FFT:
 - **Reducción** del consumo de potencia para el **66% de los casos**.
 - Hasta un **2.41% de reducción** del consumo de potencia.

- Mediante el estudio de la influencia del diseñador en el proceso de *floorplanning* (usando *Xilinx PlanAhead*, solo para los multiplicadores *Xilinx* y el multiplicador segmentado) se ha conseguido:
 - ✓ Multiplicadores Xilinx:
 - Reducción del consumo de potencia para **52.17% de los casos**.
 - Hasta un **4.68% de reducción** del consumo de potencia.
 - ✓ Multiplicador segmentado:
 - Reducción del consumo de potencia para **11.11% de los casos**.
 - Hasta un **0.46% de reducción** del consumo de potencia.
- Adicionalmente, se han realizado estimaciones variando la temperatura ambiente y el condensador de carga.
 - Temperatura ambiente:
 - **Reducción** de hasta un **20% del consumo**.
 - Condensador de carga:
 - **Reducción** de hasta un **7.69% de la potencia dinámica** del sistema completo
- Haciendo uso de **multiplicadores embebidos** en lugar de multiplicadores construidos a partir de LUTs, se consigue un **ahorro del 79,47%**
- El multiplicador de Xilinx que está construido con LUTs y orientado a optimizar la velocidad, obtiene mejores resultados para la frecuencia de síntesis.
- El multiplicador segmentado, presenta mejores resultados para la frecuencia de síntesis que los multiplicadores de Xilinx, pero conlleva un mayor consumo de potencia.

Referencias

1. E. Boemo, G. Gonzalez de Rivera, S.Lopez-Buedo and J. Meneses, "Some Notes on Power Management on FPGAs", Lecture Notes in Computer Science, No.975, pp.149-157. Berlin: Springer-Verlag 1995.
2. J.P. Oliver and E. Boemo: "Power Estimations ss. Power Measurements in Cyclone III Devices". In VII Southern Conference on Programmable Logic, pp.87-90, IEEE Press, 2011.
3. J. Rabaey and M. Pedram (editors), Low Power Design Methodologies, Kluwer Academic Publishers, Boston, 1995.
4. Anantha P. Chandrakasan, "Low-Power Digital CMOS Design".
5. S. Chou, "Innovation and Integration in the Nanoelectronics Era".
6. M. Pedram, "Design technologies for low power VLSI," In Encyclopedia of Computer Science and Technology, Vol. 36, Marcel Dekker, Inc., 1997, pp. 73-96.
7. Stephen Brown and Jonathan Rose, "Architecture of FPGAs and CPLDs: A Tutorial," IEEE Design and Test of Computers, Vol. 13, No. 2, pp. 42-57, 1996.
8. V.Stojanovic, D.Markovic, B.Nikolic, M.Horowitz, R.Brodersen, "Energy-Delay Tradeoffs in Combinational Logic Using Gate Sizing and Supply Voltage Optimization".
9. F. Machado, S. Borromeo, C. Rodriguez , "Diseño de sistemas digitales con VHDL".
10. Jan Rabaey , "Low Power Design Essentials" .
11. Michael Keating, David Flynn, Robert Aitken, Alan Gibbons, Kaijian Shi, "Low Power Methodology Manual".
12. T.Kuroda, T. Sakurai, "Overview of low-power ULSI circuit techniques".
13. High Throughput & Low. Area AES. Core Specifications.
14. Pipelined FFT/IFFT 256 points (Fast Fourier Transform). IP Core User Manual.
15. Digital ASIC Group, Lund Institute of Technology, " Digital ASIC Design. A Tutorial on the Design Flow".
16. Avnet Programming Utility. User Manual. Xilinx® Spartan®-6 LX16. Evaluation Kit. User Guide.
17. Departamento Ingeniería Ambiental e Ingeniería Electrónica de la Universidad Nacional de Río Negro, "Introducción al Proceso de Medición".
18. Juan Zúñiga Román , "Técnicas experimentales de Física General".
19. LogiCORE IP Multiplier v11.2. Especificaciones del producto.

20. Alfonso Muñoz Muñoz , "Seguridad Europea para EEUU. Algoritmo Criptográfico Rijndael".
21. Schematics Xilinx® Spartan®-6 LX16.

Anexo A: Tarjeta de desarrollo

El propósito de este anexo es describir la funcionalidad y el contenido de la *Avnet Spartan-6 LX16 Evaluation Kit* de *Avnet Electronics Marketing*. En la documentación de el *Spartan-6 LX16 Evaluation Kit* se incluyen instrucciones para el funcionamiento de la tarjeta, descripciones de las características del hardware, y las explicaciones de los códigos de prueba que se encuentran en la memoria programable de la tarjeta.

A.1. Descripción

El *Spartan-6 LX16 Evaluation Kit* proporciona un completo entorno de hardware para que los diseñadores puedan acelerar el tiempo de comercialización.

El kit ofrece una plataforma estable para desarrollar y probar los diseños dirigidos a tener un bajo coste y bajo consumo de potencia de la familia de FPGAs *Xilinx Spartan-6* y los dispositivos *Cypress PSoC 3 (Programmable System-on-Chip) Mixed Signal Array*.

La *Spartan-6 LX16* ofrece un entorno de prototipado para demostrar la eficacia y aumentar los beneficios de las soluciones de bajo coste de Xilinx. Existen diseños de referencia que se incluyen en el kit para una ejercitación con los periféricos de la tarjeta de evaluación y tener una rápida familiarización con el dispositivo.

El *Spartan-6 LX16 Evaluation Kit* contiene los siguientes elementos:

- Placa de evaluación *Avnet Spartan-6 LX16*
- Avnet LCD PSoC módulo de expansión
- Módulo de expansión Multi I/O plug-in Flash
- Cable USB de tipo A a tipo mini-B
- ISE DVD
- Carta de Bienvenida
- Guía de inicio rápido

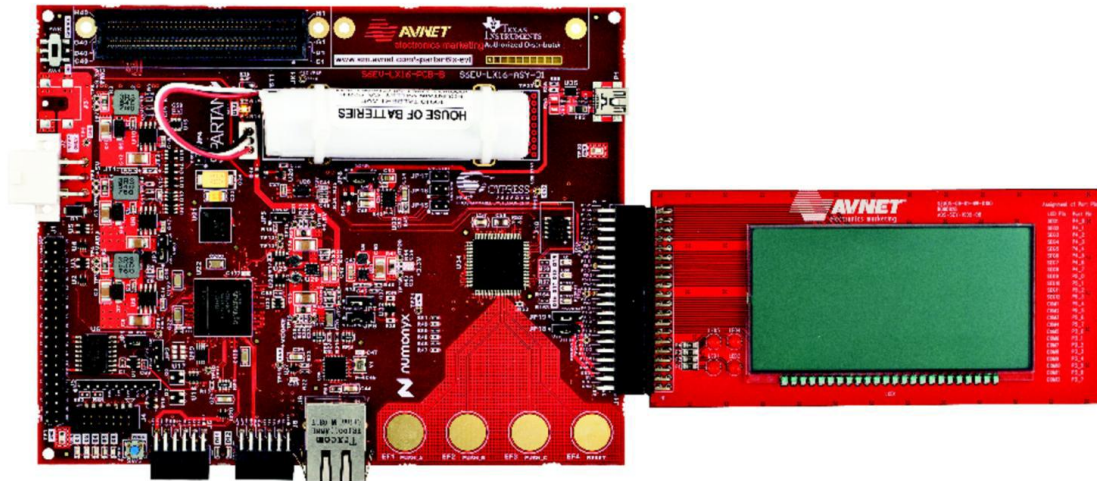


Figura A.1. Spartan-6 LX16 Evaluation Kit. Extraído de [16]

En el kit no se incluye el adaptador a la fuente de alimentación.

A.2. Descripción funcional

Una FPGA Xilinx Spartan-6LX16 y un Cypress CY8C3866AXI-40 PSoC3 Mixed-Signal Array son los componentes principales de la Avnet Spartan-6 LX16 Evaluation Board. Además de las funciones de procesamiento interno, los dispositivos PSoC (*Programmable System-on-Chip*) ofrecen comunicaciones externas vía un USB 2.0 de alta velocidad. Las comunicaciones entre el PSoC y la FPGA es facilitada mediante una variedad de mecanismos. Un puerto Ethernet 10/100 ofrece otra posibilidad de comunicaciones externas.

La memoria interna consiste en 256 Mbit x 16 LPDDR (*Low Power Double Data Rate*) SDRAM y una memoria Flash de 128 Mbit para los puertos I/O que también puede ser usada por la FPGA para configuración.

El diagrama de alto nivel de la *Spartan-6 LX16 Evaluation Board* se muestra a continuación junto con una breve descripción de cada subapartado:

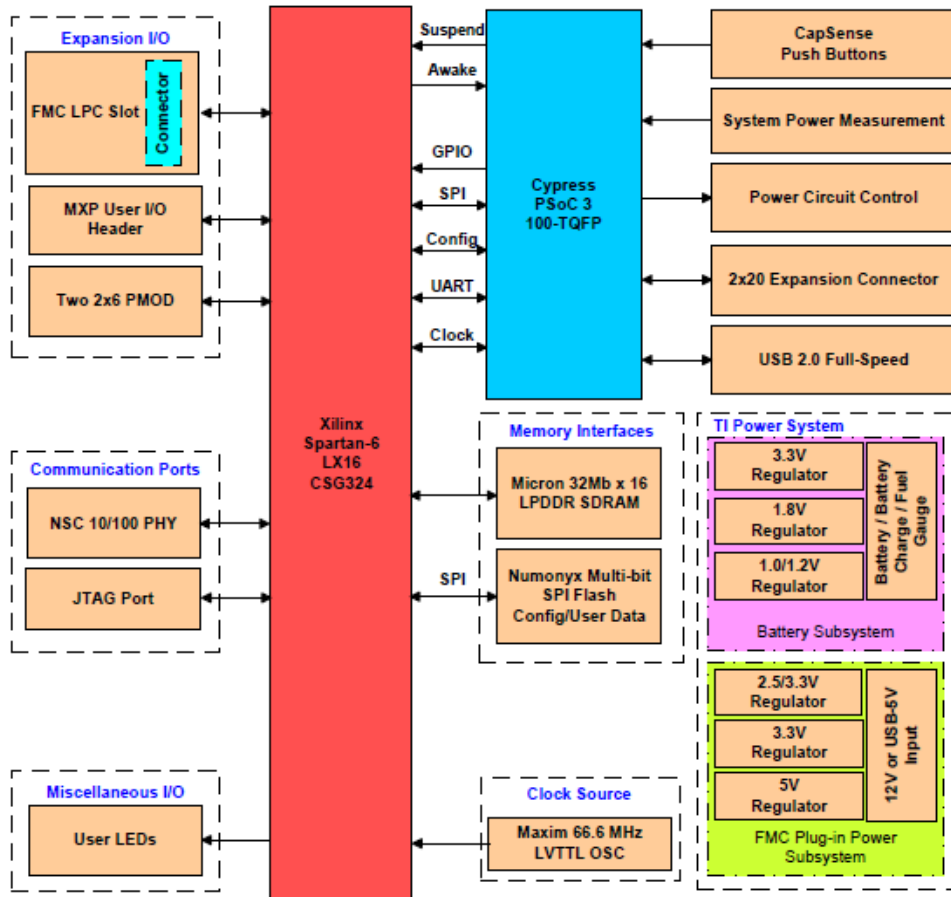


Figura A.2. Diagrama de bloques para la Spartan-6 LX16 Evaluation Board. Extraído de [16]

A.2.1. Xilinx Spartan-6 LX16 FPGA

El dispositivo *XC6SLX16-2CSG324* diseñado sobre la *Spartan-6LX16 Evaluation Board* es un miembro de la familia Xilinx *Spartan-6 LX* FPGA. Esta familia de FPGAs ofrece una nueva y más eficiente LUT (look-up table) de 6 entradas y una rica selección de sistemas incorporados a nivel de bloques. Estos incluyen una RAM de 18 Kb (2 x 9 Kb), controladores de memoria SDRAM, un sistema mejorado de control de reloj, tecnología *SelectIO™*, modos de gestión avanzada de potencia, opciones de configuración autodetectables y un aumento de seguridad IP mediante un dispositivo de protección DNA. Estas características ofrecen una alternativa de programación de bajo coste personalizable frente a los productos ASIC (*Application-Specific Integrated Circuit*) con una facilidad de uso sin precedentes. Las FPGAs Spartan-6 ofrecen la mejor solución para diseños con alto volumen lógico, orientado a los diseñadores de DSP (*Digital Signal Processor*) y a las aplicaciones embebidas sensibles a los costes.

En la *Avnet Spartan -6 LX16 Evaluation Board*, la FPGA tiene cuatro bancos de I/O. El banco 0 y el banco 2 están vinculados con la alimentación V_{CCAUX} que es seleccionable entre 2,5 V y 3,3 V.

El interfaz del banco 1 está conectado con los periféricos y está a 3,3 V. El interfaz del banco 3 conectado a la memoria LPDDR está a 1,8 V debido al diseño de bajo consumo de la memoria. A pesar de que los múltiples voltajes de alimentación aumentan la complejidad del diseño de la tarjeta, una parte importante del ahorro se puede conseguir basándose en las características del diseño de la FPGA.

A.2.2. Cypress PSoC 3 Mixed-Signal Array

El Cypress CY8C3866 es un dispositivo configurable que contiene bloques analógicos y digitales y dispositivos periféricos que permiten al usuario crear diferentes configuraciones para el soporte de las aplicaciones. Tal y como se ha configurado en la *Spartan-6 LX16 Evaluation Board*, el PSoC ofrece un interfaz USB 2.0 a 12 Mbps, un UART (*Universal Asynchronous Receiver-Transmitter*), un bus SPI (*Serial Peripheral Interface*), interfaces I2C (*Inter-Integrated Circuit*), cuatro pads táctiles (la condición de estos se envía a la FPGA), monitorización de la gestión de la batería y 28 líneas de I/O de propósito general.

Adicionalmente el programa *MiniProg3* se puede usar para depurar los códigos de las aplicaciones usando el entorno de desarrollo Cypress PSoC Creator.

El Cypress CY8C3866 tiene cuatro bancos de I/O configurables todos conectados a 3,3 V

A.2.3. Relojes

Se dispone de un reloj a 66,6 MHz de bajo coste y un reloj programable para el PSoC. Este último es programable gracias a un oscilador interno. El firmware *S6LX16* fija la frecuencia por defecto a 24 MHz. Este firmware puede ser controlado por el AvProg para ajustar la frecuencia desde 8 a 33 MHz.

A.2.4. Memoria

La *Spartan-6 LX15 Evaluation Board* es popular por tener una memoria LPDDR y una SPI flash. Esta última puede ser usada para la configuración de la FPGA. La memoria LPDDR tiene una arquitectura que permite el doble de tasa de datos para conseguir altas velocidades de operación. El siguiente diagrama muestra las conexiones a alto nivel en la tarjeta.

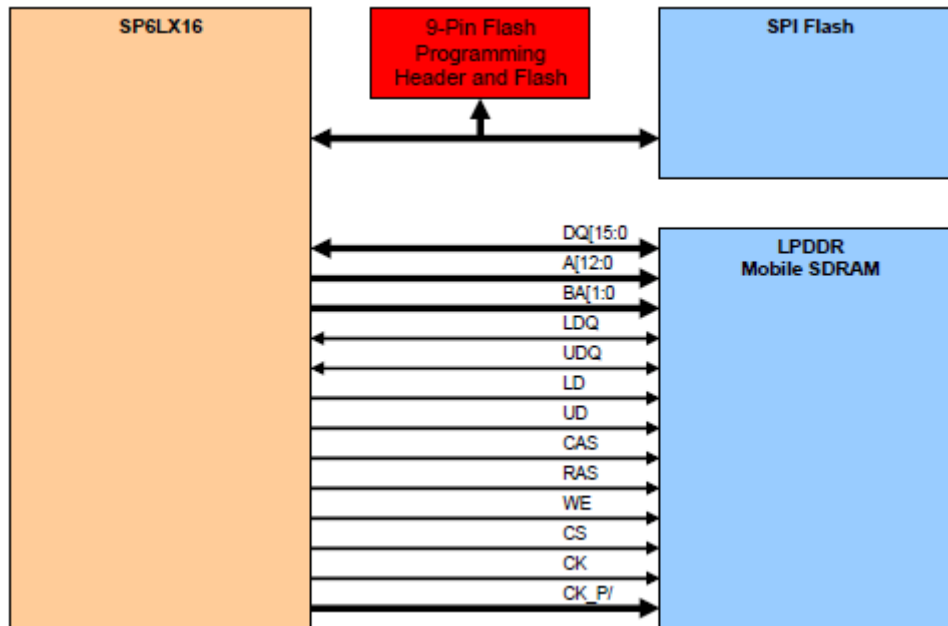


Figura A.3. Interfaces de memoria para la Spartan-6 LX16 Evaluation Board. Extraído de [16]

La memoria SPI puede ser programada de las siguientes maneras:

- Usando el software *AvProg* para programar la memoria flash indirectamente
- Usando el cable USB con *Flyleads* y la utilidad *XSPI* de Xilinx
- Usando el cable USB vía JTAG para programar la memoria flash indirectamente usando *iMPACT 12.1* o sucesivo

A.2.5. Comunicaciones

Se dispone de interfaces USB-UART usados para reemplazar los RS232 como medio de comunicaciones entre el PSoC y la FPGA. Esta interfaz opera a 3,3 V y soporta tasas de hasta 1 Mb/s.

El PSoC puede enlazar los datos del USB a un puerto de la SPI que se conecta con el puerto de configuración en la *Spartan-6*. Con el correcto software se consigue que el usuario transfiera vía USB la configuración a la FPGA con una tasa máxima de 3 Mb/s.

También existe la posibilidad de usar una conexión Ethernet, la cual requiere una corriente de entre 81 y 92 mA según el modo de uso, por lo que dependiendo de otros circuitos usados simultáneamente puede suceder que la batería o la alimentación USB no sean suficientes y sea necesaria una alimentación externa de 12 V.

A.2.6. Puertos de expansión

Los puertos de expansión que existen en la *Spartan-6 LX16* son el *VITA 57.1 FMC-LPC Connector* (necesita alimentación externa de 12 V, estos conectores permiten altas velocidades de transmisión con bajas pérdidas de inserción), *Avnet Mini Expansion Port (MXP)* (necesita alimentación externa de 12 V) y *PSoC Expansion* (usado para el modulo LCD) y el *Peripheral Module (PMOD)*.

A.2.7. Interfaces de usuario

-LEDs: Hay instalados cuatro LEDs en la tarjeta que pueden ser usados para mostrar el estado de la lógica interna.

-Botones táctiles capacitivos *PSoC CapSense*: Son cuatro botones utilizados por el usuario y conectados al dispositivo PSOC y transmitido a la FPGA.

-Cuatro pulsadores configurables: Transmitido desde los botones táctiles *PSoC CapSense*.

-Interruptor de potencia: Tiene dos posiciones: ON y CARGA. Cuando está en la posición de carga la batería continua cargado desde la alimentación del USB o desde la alimentación de 12 V. Cuando el interruptor está en la posición ON se habilitan todos los reguladores de descarga.

-Potencia: Habilita el sistema de potencia de la *Spartan-6 LX16 Evaluation Board*.

A.2.8. Potencia

A.2.8.1. Fuentes de alimentación de la tarjeta

- Batería recargable de Li-Ion de Panasonic: La *Spartan-6 LX16* usa el modelo de batería recargable *Panasonic CGR18650CF*. Es una batería de 3,6 V y 2250 mAh. El tiempo de carga de la batería es de 2 horas.

- USB 5V: Conectando el cable USB de tipo Mini-B se obtiene una alimentación de 5 V, esta es la principal fuente de energía para la carga de la batería.

- Alimentación de 12 V: Existen dos formas de alimentación a 12 V para la *Spartan-6 LX16 Evaluation Board*. Los componentes necesarios no están incluidos en el kit.

A.2.8.2. Gestión de batería

- Cargador de Texas Instruments: La *Spartan-6 LX16 Evaluation Board* utiliza el cargador TI BQ24032A Li-ion para la carga y gestión de la batería. Este dispositivo está integrado en un puerto USB y en un adaptador AC, con un selector autónomo de potencia para la gestión de esta.
- Medidor de carga de la batería: Ofrece información acerca de la capacidad que tiene la batería (mAh), estado de carga (%), autonomía de la batería (min), voltaje de la batería (mV) y temperatura (°C).

A.2.8.3. Reguladores de voltaje

- 3,3 V: Existen dos fuentes para 3,3 V en la *Spartan-6 LX16 Evaluation Board*. Una fuente para el circuito de la tarjeta principal y otro para la fuente de FMC (*FPGA Mezzanine Card*) con 3,3 V -*FMC Vadj*: El *FMC Vadj* está alimentado por el TI TPS54332 a 3,5 A y 28 V (máxima entrada).
- 1,8 V: Está derivado desde el TI TPS62400 que acepta un voltaje de entrada entre 2,5 y 6 V con una corriente de 400-600 mA.
- 1,2 V: Está derivado desde el TI TPS62400 que acepta voltajes de 2,5 a 6 V y corrientes de 400-600 mA. Este regulador alimenta el núcleo de la FPGA.
- 5 V: Este regulador está alimentado a 12 V. Utiliza el cargador de batería como alimentación opcional.

A.2.8.4. Sistema de gestión de bajo consumo

Existen 4 interruptores de carga de Texas Instruments, uno controla las conexiones PMOD (*Peripheral Modules*), otro controla la alimentación del Ethernet, otro los bancos 0 y 2 de I/O y otro el banco 1 y la memoria flash SPI.

Modo de suspensión: La *Spartan-6* dispone de un pin para la suspensión y otro pin para la activación después de la suspensión. Es controlado por el PSoC.

Hibernación: La hibernación en la *Spartan-6* es una potente opción de ahorro de potencia. La hibernación es habilitada por el PSoC.

A.2.8.5. Medida de potencia usando PSoC

Las vías primarias de alimentación para 1,2 V, 1,8 V y 3,3 V son medibles dinámicamente por el PSoC mediante una resistencia de precisión enlazada a la vía. Esto permite al PSoC calcular la corriente y la potencia consumida para cada una de las vías. La información puede ser vista en el *AvProg*:

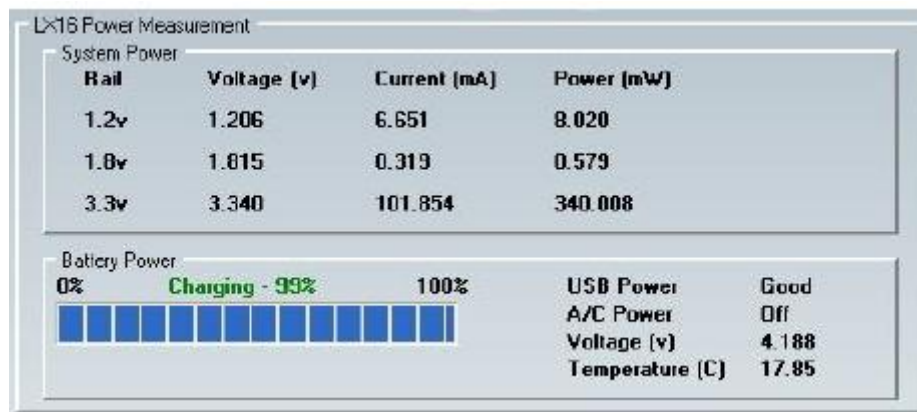


Figura A.4. Interfaz LX16 Power Measurement

A.2.8.6. Desacoplamiento FPGA

Los requerimientos de desacoplamiento para la *Spartan-6* están especificados por la guía del fabricante. Para la *S6LX16* se continua con esas directrices pero con algunas variaciones. Se usaran distintos condensadores (0.22, 0.47, 100 μ F) dependiendo de la capacidad que se necesite.

A.2.8.7. Resultados de potencia

La circuitería fue probada durante la fase de prototipado para probar la compatibilidad en temas de requerimientos de potencia de la *Spartan-6*, tales como:

-Tolerancia:

- 1,2 V V_{CCINT} -> de 1,14 a 1,26 V
- V_{CCAUX} -> de 2,375 a 2,625 V
- V_{CCO} -> +/-5%
- Tiempo de subida: de 0,2 a 85 ms (hay que comprobar que el circuito de arranque no esté sobrecargado al iniciar)
- Monotonicidad: No deben haber caídas en V_{CCINT} o V_{CCAUX}
- Secuenciación: La secuenciación debe responder según lo diseñado.

A.2.9. Configuración

La *Spartan-6 LX16 Evaluation Board* soporta muchos modos de configuración de la FPGA. Las posibles configuraciones incluye *Boundary-scan* (cable JTAG), *Serial Peripheral Interface* (SPI Flash) o *Slave Serial* mediante el Cypress PSoC USB. El LED azul de la tarjeta indica cuando la FPGA ha sido configurada satisfactoriamente.

Anexo B: Propagación de errores

En el proceso de medición el valor medido se determina mediante el uso de distintos instrumentos de medición y mediante un método seleccionado para tal fin. Para este proceso existen limitaciones dadas por los instrumentos utilizados, el método empleado e incluso por el sujeto que realiza la medición.

Con la indicación del error de medición expresamos, en forma cuantitativa las limitaciones que nuestro proceso de medición introduce en la determinación de la magnitud medida. De manera gráfica:

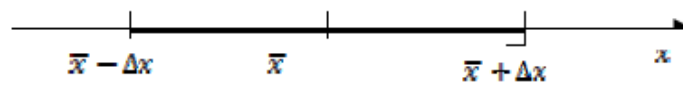


Figura B.1. Indicación de error. Extraído de [17]

Se busca establecer un intervalo $\bar{x} - \Delta x \leq x \leq \bar{x} + \Delta x$ (Fig. 1), donde con cierta probabilidad, podamos decir que se encuentra el mejor valor de la magnitud x . Este mejor valor \bar{x} es el más representativo de nuestra medición y denominamos Δx a la incertidumbre o error absoluto de la medición. La figura anterior representa el resultado de una medición, en lugar de dar un único número como resultado, se define un intervalo.

Se define como medidas indirectas, aquellas magnitudes que se calculan a partir de los valores encontrados en las medidas de otras magnitudes. Conociendo $\bar{x} \pm \Delta x$, $\bar{y} \pm \Delta y$, calculamos $z = f(x, y, \dots)$. La cuestión será saber cuál es el error de z .

La propagación de errores es un conjunto de reglas que permiten asignar un error a z , conocidas las incertidumbres de x e y . Permiten asignar un error al resultado final e indica la importancia relativa de las diferentes medidas directas.

• **Propagación de errores en sumas y diferencias**

Datos iniciales: $\bar{x} \pm \Delta x, \bar{y} \pm \Delta y$

Sea su suma $q = \bar{x} + \bar{y}$ y su diferencia $q = \bar{x} - \bar{y}$, ¿cuál será la incertidumbre Δq ?

	Suma	Diferencia
Valor máximo de q	$q = \bar{x} + \Delta x + \bar{y} + \Delta y$ $= \bar{x} + \bar{y} + (\Delta x + \Delta y)$	$q = \bar{x} + \Delta x - \bar{y} - \Delta y$ $= \bar{x} - \bar{y} + (\Delta x + \Delta y)$
Valor mínimo de q	$q = \bar{x} - \Delta x + \bar{y} - \Delta y$ $= \bar{x} + \bar{y} - (\Delta x + \Delta y)$	$q = \bar{x} - \Delta x - \bar{y} + \Delta y$ $= \bar{x} - \bar{y} - (\Delta x + \Delta y)$

El error absoluto de la suma y de la diferencia de dos o más magnitudes es la suma de los errores absolutos de dichas magnitudes:

$$q = \bar{x} + \bar{y} \rightarrow \Delta q \approx \Delta x + \Delta y$$

• **Propagación de errores en productos**

Datos iniciales: $\bar{x} \pm \Delta x = \bar{x} \left(1 \pm \frac{\Delta x}{|\bar{x}|}\right), \bar{y} \pm \Delta y = \bar{y} \left(1 \pm \frac{\Delta y}{|\bar{y}|}\right)$

Sea su producto $q = \bar{x} \cdot \bar{y}$, ¿cuál será la incertidumbre Δq ?

	Producto
Valor máximo de q	$q_{max} = \bar{x} \left(1 + \frac{\Delta x}{ \bar{x} }\right) \bar{y} \left(1 + \frac{\Delta y}{ \bar{y} }\right) \cong \bar{x} \bar{y} \left(1 + \frac{\Delta x}{ \bar{x} } + \frac{\Delta y}{ \bar{y} }\right)$ \uparrow $\frac{\Delta x \Delta y}{ \bar{x} \bar{y} } \ll$
Valor mínimo de q	$q_{min} = \bar{x} \left(1 - \frac{\Delta x}{ \bar{x} }\right) \bar{y} \left(1 - \frac{\Delta y}{ \bar{y} }\right) \cong \bar{x} \bar{y} \left(1 - \frac{\Delta x}{ \bar{x} } - \frac{\Delta y}{ \bar{y} }\right)$ \uparrow $\frac{\Delta x \Delta y}{ \bar{x} \bar{y} } \ll$

El error relativo del producto es igual a la suma de los errores relativos:

$$q = \bar{x} \cdot \bar{y} \rightarrow \frac{\Delta q}{|q|} \approx \frac{\Delta x}{|\bar{x}|} + \frac{\Delta y}{|\bar{y}|}$$

• **Propagación de errores en cocientes**

Datos iniciales: $\bar{x} \pm \Delta x = \bar{x} \left(1 \pm \frac{\Delta x}{|\bar{x}|}\right), \bar{y} \pm \Delta y = \bar{y} \left(1 \pm \frac{\Delta y}{|\bar{y}|}\right)$

Sea su producto $q = \frac{\bar{x}}{\bar{y}}$, ¿cuál será la incertidumbre Δq ?

Cocientes

Valor máximo de q

$$q_{max} = \frac{\bar{x} \left(1 + \frac{\Delta x}{|\bar{x}|}\right)}{\bar{y} \left(1 - \frac{\Delta y}{|\bar{y}|}\right)} \cong \frac{\bar{x}}{\bar{y}} \left(1 + \frac{\Delta x}{|\bar{x}|}\right) \left(1 + \frac{\Delta y}{|\bar{y}|}\right) \cong \frac{\bar{x}}{\bar{y}} \left(1 + \frac{\Delta x}{|\bar{x}|} + \frac{\Delta y}{|\bar{y}|}\right)$$

\uparrow $\frac{1}{1-\varepsilon} = 1 + \varepsilon$ \quad \uparrow $\frac{\Delta x \Delta y}{|\bar{x}| |\bar{y}|} \ll 1$

Valor mínimo de q

$$q_{min} = \frac{\bar{x} \left(1 - \frac{\Delta x}{|\bar{x}|}\right)}{\bar{y} \left(1 + \frac{\Delta y}{|\bar{y}|}\right)} \cong \frac{\bar{x}}{\bar{y}} \left(1 - \frac{\Delta x}{|\bar{x}|}\right) \left(1 - \frac{\Delta y}{|\bar{y}|}\right) \cong \frac{\bar{x}}{\bar{y}} \left(1 - \left[\frac{\Delta x}{|\bar{x}|} + \frac{\Delta y}{|\bar{y}|}\right]\right)$$

\uparrow $\frac{1}{1+\varepsilon} = 1 - \varepsilon$ \quad \uparrow $\frac{\Delta x \Delta y}{|\bar{x}| |\bar{y}|} \ll 1$

El error relativo del cociente es igual a la suma de los errores relativos:

$$q = \frac{\bar{x}}{\bar{y}} \rightarrow \frac{\Delta q}{|q|} \approx \frac{\Delta x}{|\bar{x}|} + \frac{\Delta y}{|\bar{y}|}$$

- **Error del producto por una constante**

Datos iniciales: $\bar{x} \pm \Delta x$

Sea $q = Ax$, ¿cuál será la incertidumbre Δq ?

Aplicando la regla del producto:

$$\frac{\Delta q}{|q|} \approx \frac{\Delta A}{|A|} + \frac{\Delta x}{|\bar{x}|} = \frac{\Delta x}{|\bar{x}|} \rightarrow \Delta q = |A| \Delta x$$

El error absoluto del producto de una constante por una magnitud es igual al producto de la constante por el error absoluto de la magnitud:

$$\Delta q = |A| \Delta x$$

- **Error de una potencia**

Datos iniciales: $\bar{x} \pm \Delta x$

Sea $q = x^n = x \cdot x \cdots x$

Aplicando la regla del producto:

$$\frac{\Delta q}{|q|} \approx \frac{\Delta x}{|\bar{x}|} + \frac{\Delta x}{|\bar{x}|} + \dots + \frac{\Delta x}{|\bar{x}|} = |n| \frac{\Delta x}{|\bar{x}|}$$

El error relativo de una potencia es el producto de la potencia por el error relativo de la magnitud:

$$\frac{\Delta q}{|q|} = |n| \frac{\Delta x}{|\bar{x}|}$$

- **Error en el caso de estudio**

En el caso de estudio en este proyecto en el error en el resultado final será de la siguiente manera:

$$P + \Delta P = V_{cc} \cdot (I + \Delta I) = 1.2 \cdot (I + \Delta I)$$

$$I + \Delta I = \frac{V + \Delta V}{R + \Delta R}$$

Donde V es la tensión medida en la FPGA, R el valor de la resistencia sobre la que se mide V, ΔV es el 0.003% (obtenido del *datasheet* del multímetro) y ΔR es el 1% (obtenido de la información de la placa de pruebas).

El error relativo de un cociente es igual a la suma de los errores relativos

$$\frac{\Delta I}{|I|} \approx \frac{\Delta V}{|V|} + \frac{\Delta R}{|R|}$$

Finalmente para calcular el error de potencia habrá que aplicar el error del producto por una constante

$$\Delta P = 1.2 \cdot \Delta I$$

Para los multiplicadores (Xilinx y segmentado) se harán unas estimaciones del error para cada uno de ellos, a partir de la estrategia de diseño balanceada.

- *Multiplicador embebido optimizado en área*

$$I = 18.2283 \text{ mA} \rightarrow (x 0.05) \quad V = 0.9114 \text{ mV}$$

$$\frac{\Delta I}{|I|} \approx \frac{\Delta V}{|V|} + \frac{\Delta R}{|R|} = \frac{0.00003}{0.9114 \cdot 10^{-3}} + \frac{0.01}{0.05} = 0.03291 + 0.2$$

$$\Delta I = (0.03291 + 0.2) \cdot I = 0.03291 \cdot I + 0.2 \cdot I = 5.999 \cdot 10^{-4} + 3.64566 \cdot 10^{-3} = 4.2455 \cdot 10^{-3} = 0.42455\%$$

$$\Delta P = 1.2 \cdot \Delta I = \mathbf{0.5095 \%}$$

- *Multiplicador embebido optimizado en velocidad*

$$I = 3.4 \text{ mA} \rightarrow (x 0.05) V = 0.17 \text{ mV}$$

$$\frac{\Delta I}{|I|} \approx \frac{\Delta V}{|V|} + \frac{\Delta R}{|R|} = \frac{0.00003}{0.17 \cdot 10^{-3}} + \frac{0.01}{0.05} = 0.1765 + 0.2$$

$$\Delta I = (0.1765 + 0.2) \cdot I = 0.1765 \cdot I + 0.2 \cdot I = 6.001 \cdot 10^{-4} + 6.8 \cdot 10^{-4} = 1.2801 \cdot 10^{-3} = 0.12801\%$$

$$\Delta P = 1.2 \cdot \Delta I = \mathbf{0.1536 \%}$$

- *Multiplicador con LUTs optimizado en área*

$$I = 16.558 \text{ mA} \rightarrow (x 0.05) V = 0.8279 \text{ mV}$$

$$\frac{\Delta I}{|I|} \approx \frac{\Delta V}{|V|} + \frac{\Delta R}{|R|} = \frac{0.00003}{0.8279 \cdot 10^{-3}} + \frac{0.01}{0.05} = 0.0362 + 0.2$$

$$\Delta I = (0.0362 + 0.2) \cdot I = 0.0362 \cdot I + 0.2 \cdot I = 6 \cdot 10^{-4} + 3.3116 \cdot 10^{-3} = 3.9116 \cdot 10^{-3} = 0.3911\%$$

$$\Delta P = 1.2 \cdot \Delta I = \mathbf{0.4694 \%}$$

- *Multiplicador con LUTs optimizado en velocidad*

$$I = 15.5 \text{ mA} \rightarrow (x 0.05) V = 0.775 \text{ mV}$$

$$\frac{\Delta I}{|I|} \approx \frac{\Delta V}{|V|} + \frac{\Delta R}{|R|} = \frac{0.00003}{0.775 \cdot 10^{-3}} + \frac{0.01}{0.05} = 0.0387 + 0.2$$

$$\Delta I = (0.0387 + 0.2) \cdot I = 0.0387 \cdot I + 0.2 \cdot I = 5.999 \cdot 10^{-4} + 3.3116 \cdot 10^{-3} = 3.9115 \cdot 10^{-3} = 0.39115\%$$

$$\Delta P = 1.2 \cdot \Delta I = \mathbf{0.4694 \%}$$

○ *Multiplicador segmentado*

$$I = 20.1 \text{ mA} \rightarrow (x 0.05) V = 1.005 \text{ mV}$$

$$\frac{\Delta I}{|I|} \approx \frac{\Delta V}{|V|} + \frac{\Delta R}{|R|} = \frac{0.00003}{1.005 \cdot 10^{-3}} + \frac{0.01}{0.05} = 0.02985 + 0.2$$

$$\Delta I = (0.02985 + 0.2) \cdot I = 0.02985 \cdot I + 0.2 \cdot I = 5.999 \cdot 10^{-4} + 4.02 \cdot 10^{-3} = 4.62 \cdot 10^{-3} = 0.462\%$$

$$\Delta P = 1.2 \cdot \Delta I = \mathbf{0.5544 \%}$$

Para los diseños AES y FFT se harán las estimaciones del error a partir de la estrategia de diseño balanceada. Las medidas para AES y FFT se tomaron con Avnet, no se realizaron medidas mediante el multímetro de precisión, por lo que hay que aplicar una corrección a las medidas de Avnet.

○ *AES*

$$I_{Avnet} = 111.221 \text{ mA} \rightarrow I_{medida} = 114.557 \text{ mA} \rightarrow (x 0.05) V = 5.728 \text{ mV}$$

$$\frac{\Delta I}{|I|} \approx \frac{\Delta V}{|V|} + \frac{\Delta R}{|R|} = \frac{0.00003}{5.728 \cdot 10^{-3}} + \frac{0.01}{0.05} = 0.0052 + 0.2$$

$$\Delta I = (0.0052 + 0.2) \cdot I = 0.0052 \cdot I + 0.2 \cdot I = 5.957 \cdot 10^{-4} + 0.0229 = 0.0235 = 2.35\%$$

$$\Delta P = 1.2 \cdot \Delta I = \mathbf{2.82 \%}$$

○ *FFT*

$$I_{Avnet} = 185.24 \text{ mA} \rightarrow I_{medida} = 190.797 \text{ mA} \rightarrow (x 0.05) V = 9.539 \text{ mV}$$

$$\frac{\Delta I}{|I|} \approx \frac{\Delta V}{|V|} + \frac{\Delta R}{|R|} = \frac{0.00003}{9.539 \cdot 10^{-3}} + \frac{0.01}{0.05} = 0.003145 + 0.2$$

$$\Delta I = (0.003145 + 0.2) \cdot I = 0.003145 \cdot I + 0.2 \cdot I = 6 \cdot 10^{-4} + 0.0382 = 0.0388 = 3.88\%$$

$$\Delta P = 1.2 \cdot \Delta I = \mathbf{4.656 \%}$$

Presupuesto

1) Ejecución Material

- Compra de ordenador personal (Software incluido) 1.000 €
- Spartan-6 LX16 Evaluation Board 170 €
- Multímetro Agilent 34410A Digital 1.025 €
- Material de oficina..... 50 €
- Total de ejecución material 2.245 €

2) Gastos generales

- 16 % sobre Ejecución Material..... 359,2 €

3) Beneficio Industrial

- 6 % sobre Ejecución Material..... 134,7 €

4) Honorarios Proyecto

- 836 horas a 15 € / hora 12.540 €

5) Material fungible

- Gastos de impresión 60 €
- Encuadernación 200 €

6) Subtotal del presupuesto

- Subtotal Presupuesto..... 15.538,9 €

7) I.V.A. aplicable

- 21% Subtotal Presupuesto..... 3.263,2 €

8) Total presupuesto

- Total Presupuesto 18.802,1 €

Madrid, septiembre de 2013

El Ingeniero Jefe de Proyecto

Fdo.: Godoy Garcés Mencía
Ingeniero Superior de Telecomunicación

Glosario

FPGA: Field Programmable Gate Array

HDL: Hardware Description Language

AES: Advanced Encryption Standard

FFT: Fast Fourier Transform

DFT: Discrete Fourier transform

CPLD: Complex Programmable Logic Device

DSP: Digital Signal Processing

LUT: Lookup Table

PMOS: P-type metal-oxide-semiconductor

NMOS: N-type metal-oxide-semiconductor

CMOS: Complementary metal–oxide–semiconductor

SAIF: Switching Activity Interchange Format

LFSR: Linear Feedback Shift Register

IO: Input/Output

PSoC: Programmable System-on-Chip

Pliego de condiciones

Este documento contiene las condiciones legales que guiarán la realización, en este proyecto, de un análisis de *Técnicas de Low-Power Design en FPGAs*. En lo que sigue, se supondrá que el proyecto ha sido encargado por una empresa cliente a una empresa consultora con la finalidad de realizar dicho sistema. Dicha empresa ha debido desarrollar una línea de investigación con objeto de elaborar el proyecto. Esta línea de investigación, junto con el posterior desarrollo de los programas está amparada por las condiciones particulares del siguiente pliego.

Supuesto que la utilización industrial de los métodos recogidos en el presente proyecto ha sido decidida por parte de la empresa cliente o de otras, la obra a realizar se regulará por las siguientes:

Condiciones generales

1. La modalidad de contratación será el concurso. La adjudicación se hará, por tanto, a la proposición más favorable sin atender exclusivamente al valor económico, dependiendo de las mayores garantías ofrecidas. La empresa que somete el proyecto a concurso se reserva el derecho a declararlo desierto.

2. El montaje y mecanización completa de los equipos que intervengan será realizado totalmente por la empresa licitadora.

3. En la oferta, se hará constar el precio total por el que se compromete a realizar la obra y el tanto por ciento de baja que supone este precio en relación con un importe límite si este se hubiera fijado.

4. La obra se realizará bajo la dirección técnica de un Ingeniero Superior de Telecomunicación, auxiliado por el número de Ingenieros Técnicos y Programadores que se estime preciso para el desarrollo de la misma.

5. Aparte del Ingeniero Director, el contratista tendrá derecho a contratar al resto del personal, pudiendo ceder esta prerrogativa a favor del Ingeniero Director, quien no estará obligado a aceptarla.

6. El contratista tiene derecho a sacar copias a su costa de los planos, pliego de condiciones y presupuestos. El Ingeniero autor del proyecto autorizará con su firma las copias solicitadas por el contratista después de confrontarlas.

7. Se abonará al contratista la obra que realmente ejecute con sujeción al proyecto que sirvió de base para la contratación, a las modificaciones autorizadas por la superioridad o a las órdenes que con arreglo a sus facultades le hayan comunicado por escrito al Ingeniero Director de obras siempre que dicha obra se haya ajustado a los preceptos de los pliegos de condiciones, con arreglo a los cuales, se harán las modificaciones y la valoración de las diversas unidades sin que el importe total pueda exceder de los presupuestos aprobados. Por consiguiente, el número de unidades que se consignan en el proyecto o en el presupuesto, no podrá servirle de fundamento para entablar reclamaciones de ninguna clase, salvo en los casos de rescisión.

8. Tanto en las certificaciones de obras como en la liquidación final, se abonarán los trabajos realizados por el contratista a los precios de ejecución material que figuran en el presupuesto para cada unidad de la obra.

9. Si excepcionalmente se hubiera ejecutado algún trabajo que no se ajustase a las condiciones de la contrata pero que sin embargo es admisible a juicio del Ingeniero Director de obras, se dará conocimiento a la Dirección, proponiendo a la vez la rebaja de precios que el Ingeniero estime justa y si la Dirección resolviera aceptar la obra, quedará el contratista obligado a conformarse con la rebaja acordada.

10. Cuando se juzgue necesario emplear materiales o ejecutar obras que no figuren en el presupuesto de la contrata, se evaluará su importe a los precios asignados a otras obras o materiales análogos si los hubiere y cuando no, se discutirán entre el Ingeniero Director y el contratista, sometiéndolos a la aprobación de la Dirección. Los nuevos precios convenidos por uno u otro procedimiento, se sujetarán siempre al establecido en el punto anterior.

11. Cuando el contratista, con autorización del Ingeniero Director de obras, emplee materiales de calidad más elevada o de mayores dimensiones de lo estipulado en el proyecto, o sustituya una clase de fabricación por otra que tenga asignado mayor precio o ejecute con mayores dimensiones cualquier otra parte de las obras, o en general, introduzca en ellas cualquier modificación que sea beneficiosa a juicio del Ingeniero Director de obras, no tendrá derecho sin embargo, sino a lo que le correspondería si hubiera realizado la obra con estricta sujeción a lo proyectado y contratado.

12. Las cantidades calculadas para obras accesorias, aunque figuren por partidaalzada en el presupuesto final (general), no serán abonadas sino a los precios de la contrata, según las condiciones de la misma y los proyectos particulares que para ellas se formen, o en su defecto, por lo que resulte de su medición final.

13. El contratista queda obligado a abonar al Ingeniero autor del proyecto y director de obras así como a los Ingenieros Técnicos, el importe de sus respectivos honorarios facultativos por formación del proyecto, dirección técnica y administración en su caso, con arreglo a las tarifas y honorarios vigentes.

14. Concluida la ejecución de la obra, será reconocida por el Ingeniero Director que a tal efecto designe la empresa.

15. La garantía definitiva será del 4% del presupuesto y la provisional del 2%.

16. La forma de pago será por certificaciones mensuales de la obra ejecutada, de acuerdo con los precios del presupuesto, deducida la baja si la hubiera.

17. La fecha de comienzo de las obras será a partir de los 15 días naturales del replanteo oficial de las mismas y la definitiva, al año de haber ejecutado la provisional, procediéndose si no existe reclamación alguna, a la reclamación de la fianza.

18. Si el contratista al efectuar el replanteo, observase algún error en el proyecto, deberá comunicarlo en el plazo de quince días al Ingeniero Director de obras, pues transcurrido ese plazo será responsable de la exactitud del proyecto.

19. El contratista está obligado a designar una persona responsable que se entenderá con el Ingeniero Director de obras, o con el delegado que éste designe, para todo relacionado con ella. Al ser el Ingeniero Director de obras el que interpreta el proyecto, el contratista deberá consultarle cualquier duda que surja en su realización.

20. Durante la realización de la obra, se girarán visitas de inspección por personal facultativo de la empresa cliente, para hacer las comprobaciones que se crean oportunas. Es obligación del contratista, la conservación de la obra ya ejecutada hasta la recepción de la misma, por lo que el deterioro parcial o total de ella, aunque sea por agentes atmosféricos u otras causas, deberá ser reparado o reconstruido por su cuenta.

21. El contratista, deberá realizar la obra en el plazo mencionado a partir de la fecha del contrato, incurriendo en multa, por retraso de la ejecución siempre que éste no sea debido a causas de fuerza mayor. A la terminación de la obra, se hará una recepción provisional previo reconocimiento y examen por la dirección técnica, el depositario de efectos, el interventor y el jefe de servicio o un representante, estampando su conformidad el contratista.

22. Hecha la recepción provisional, se certificará al contratista el resto de la obra, reservándose la administración el importe de los gastos de conservación de la misma hasta su recepción definitiva y la fianza durante el tiempo señalado como plazo de garantía. La recepción definitiva se hará en las mismas condiciones que la provisional, extendiéndose el acta correspondiente. El Director Técnico propondrá a la Junta Económica la devolución de la fianza al contratista de acuerdo con las condiciones económicas legales establecidas.

23. Las tarifas para la determinación de honorarios, reguladas por orden de la Presidencia del Gobierno el 19 de Octubre de 1961, se aplicarán sobre el denominado en la actualidad "Presupuesto de Ejecución de Contrata" y anteriormente llamado "Presupuesto de Ejecución Material" que hoy designa otro concepto.

Condiciones particulares

La empresa consultora, que ha desarrollado el presente proyecto, lo entregará a la empresa cliente bajo las condiciones generales ya formuladas, debiendo añadirse las siguientes condiciones particulares:

1. La propiedad intelectual de los procesos descritos y analizados en el presente trabajo, pertenece por entero a la empresa consultora representada por el Ingeniero Director del Proyecto.

2. La empresa consultora se reserva el derecho a la utilización total o parcial de los resultados de la investigación realizada para desarrollar el siguiente proyecto, bien para su publicación o bien para su uso en trabajos o proyectos posteriores, para la misma empresa cliente o para otra.

3. Cualquier tipo de reproducción aparte de las reseñadas en las condiciones generales, bien sea para uso particular de la empresa cliente, o para cualquier otra aplicación, contará con autorización expresa y por escrito del Ingeniero Director del Proyecto, que actuará en representación de la empresa consultora.

4. En la autorización se ha de hacer constar la aplicación a que se destinan sus reproducciones así como su cantidad.

5. En todas las reproducciones se indicará su procedencia, explicitando el nombre del proyecto, nombre del Ingeniero Director y de la empresa consultora.

6. Si el proyecto pasa la etapa de desarrollo, cualquier modificación que se realice sobre él, deberá ser notificada al Ingeniero Director del Proyecto y a criterio de éste, la empresa consultora decidirá aceptar o no la modificación propuesta.

7. Si la modificación se acepta, la empresa consultora se hará responsable al mismo nivel que el proyecto inicial del que resulta el añadirla.

8. Si la modificación no es aceptada, por el contrario, la empresa consultora declinará toda responsabilidad que se derive de la aplicación o influencia de la misma.

9. Si la empresa cliente decide desarrollar industrialmente uno o varios productos en los que resulte parcial o totalmente aplicable el estudio de este proyecto, deberá comunicarlo a la empresa consultora.

10. La empresa consultora no se responsabiliza de los efectos laterales que se puedan producir en el momento en que se utilice la herramienta objeto del presente proyecto para la realización de otras aplicaciones.

11. La empresa consultora tendrá prioridad respecto a otras en la elaboración de los proyectos auxiliares que fuese necesario desarrollar para dicha aplicación industrial, siempre que no haga explícita renuncia a este hecho. En este caso, deberá autorizar expresamente los proyectos presentados por otros.

12. El Ingeniero Director del presente proyecto, será el responsable de la dirección de la aplicación industrial siempre que la empresa consultora lo estime oportuno. En caso contrario, la persona designada deberá contar con la autorización del mismo, quien delegará en él las responsabilidades que ostente.