



UNIVERSIDAD CARLOS III DE MADRID

Dpto. de Tecnología Electrónica

PROYECTO FIN DE CARRERA
INGENIERÍA DE TELECOMUNICACIONES

Sistema de Caracterización Automática de láseres de semiconductor y aplicación a láseres en anillo.

AUTOR: MIRIAM ORGANISTA AYALA

TUTOR: GUILLERMO CARPINTERO DEL BARRIO

OCTUBRE, 2010

Título: Sistema de Caracterización Automática de láseres de semiconductor y aplicación a láseres en anillo.

Autor: Miriam Organista Ayala

Director: Guillermo Carpintero del Barrio

EL TRIBUNAL

Presidente: Pablo Acedo Gallardo

Vocal: Daniel Segovia Vargas

Secretario: Marta Ruiz Llata

Realizado el acto de defensa y lectura del Proyecto Fin de Carrera el día 29 de Octubre de 2010 en Leganés, en la Escuela Politécnica Superior de la Universidad Carlos III de Madrid, acuerda otorgarle la CALIFICACIÓN de

VOCAL

SECRETARIO

PRESIDENTE

Agradecimientos

En primer lugar quiero dar las gracias a mi tutor en este proyecto, Guillermo Carpintero. Gracias por darme la oportunidad de realizar este proyecto sobre el estudio de láseres, por la confianza y por siempre estar dispuesto a ayudar.

Gracias también a mis compañeros del departamento por las horas de despacho y laboratorio juntos, a Álvaro Jiménez Galindo por su colaboración en este proyecto y a Ángel Rubén Criado por ayudarme cuando le he necesitado.

A todos mis compañeros de universidad y de piso durante estos seis años, por haber sido verdaderamente más que compañeros y por demostrarme lo grata que es la convivencia.

Y por último gracias a mi familia. A mis padres y a mi hermano por su apoyo y aguante cada periodo de exámenes durante estos años. También a mi tía, Rosa Ayala, por haber sido una guía y referente para mí durante esta carrera.

Muchas gracias a todos.

Resumen

El trabajo realizado durante el presente proyecto fin de carrera ha consistido en la automatización de la adquisición de datos de los diferentes instrumentos utilizados en un laboratorio de investigación optoelectrónico. Se han desarrollado diversas interfases software, para obtener los parámetros característicos de un láser de semiconductor, como son la curva de potencia óptica frente a corriente (curva P-I), la caracterización del ruido de fase.

Estos interfases han sido utilizados para la caracterización de los láseres en anillo diseñados en la Universidad Carlos III de Madrid y fabricados en la plataforma europea de InP JePPIX. Las medidas realizadas sobre estos dispositivos son la curva P-I, el espectro óptico y el espectro eléctrico.

Los resultados obtenidos, durante la medida de la curva P-I, han permitido verificar que los dispositivos operan como láser, obteniendo sus corrientes umbrales. Adicionalmente, la medida característica en diferentes tipos de estructuras (asimétricas y simétricas), ha permitido identificar que en los anillos asimétricos con múltiples absorbentes saturables se puede influir en la competición de modos que se contrapropagan en un anillo. Y mediante la polarización de cada uno de los absorbentes saturables independientemente, se ha demostrado que se puede compensar o aumentar el efecto que producen las asimetrías de la cavidad. Por otra parte, las medidas realizadas sobre el anillo simétrico han permitido observar como al aumentar la tensión inversa del absorbente saturable ambas direcciones pasan a presentar potencias semejantes, sugiriendo que el dispositivo entra en un modo de operación de colisión de pulsos entre los modos que se contrapropagan. Por el contrario, si en el anillo asimétrico se polarizaban los dos absorbentes saturables siempre se ha obtenido un modo dominante según la corriente aplicada.

Con la medida del espectro óptico se ha ampliado la información de los dispositivos, obteniéndose información del número de modos de emisión y de la separación entre modos. Este último dato está relacionado con la longitud del resonador óptico. En los láseres en anillo, la tasa de repetición es inversa a la longitud del resonador L , según c/nL . Según las medidas realizadas en el anillo asimétrico se obtiene una tasa de 12.73 GHz y en el láser en anillo simétrico 15.52GHz.

Finalmente, las medidas realizadas del espectro eléctrico han requerido amplificar la señal óptica. Los amplificadores ópticos disponibles eran dos EDFAs y un SOA disponible en el laboratorio. Sin embargo, el SOA era un dispositivo optimizado para generar efectos no lineales, y por tanto los resultados obtenidos para esta medida no son los esperados. En un futuro estudio será necesaria la adquisición de un SOA lineal para realizar un correcto análisis del ruido de fase.

Abstract

This end of degree project has been the automation the data acquisition from different instruments, which are used in optoelectronic research laboratory. Two interfaces are developed in this project. These interfaces are a useful tool to obtain the characteristic parameters of a semiconductor laser. The first of these interfaces allows measurement of optical power curve versus current (P-I curve) and the second interface is used for the characterization of phase noise.

These interfaces have been used for the characterization of ring lasers. These ring lasers were designed at the University Carlos III of Madrid and manufactured in the European platform InP JePPIX. The measurements made in these lasers are the P-I curve, the optical spectrum and the electrical spectrum.

The results obtained with the P-I curve, allowed to verify that the devices operate as lasers. In addition, the P-I curve was performed in different types of structures (asymmetric and symmetric). These measures has identified that the competition between modes which propagate in opposite directions can be modified in asymmetric rings with multiple saturable absorber. In the asymmetric ring laser always appears a dominant mode when both saturable absorbers are biased. Moreover, the polarization of each saturable absorber independently has proved that the effect of the asymmetries of the cavity can be reduced or increased. On the other hand, if the inverse tension of symmetric ring laser was increased, the modes of opposite directions acquired the same power. This suggests that the device enters a mode of operation of pulse collision between modes which propagate in opposite directions.

The measurement of the optical spectrum provides more information about the device, as he number of emission modes and the separation between modes. This last fact is related to the length of the optical resonator. In a ring laser, the repetition rate is the inverse of a laser resonator length L , as c/nL . In the results, the rate for the asymmetric ring laser was 12.73 GHz and the rate in the symmetric ring laser was 15.52 GHz.

Finally, the electric spectrum measurements are requires amplifying the optical signal. The optical amplifiers available in the laboratory were two EDFAs and a SOA. For measuring the electric spectrum was used a SOA which was available in the laboratory. However, the SOA was a device optimized to generate nonlinear effects. Because of this, the results for this measure weren't expected. In the future study, a linear SOA will be requires to perform an accurate analysis of the phase noise.

Índice general

1. MOTIVACIÓN Y OBJETIVOS	1
1.1 Entorno de desarrollo	3
1.2 Metodología de trabajo	8
2. INTRODUCCIÓN A LOS LÁSERES DE SEMICONDUCTOR.....	13
2.1 Introducción.....	13
2.2 Principio de funcionamiento	14
2.3 Modelado de los dispositivos.....	17
2.3.1 Ecuaciones de tasa.....	17
2.3.2 Modelado de dispositivos mediante software comercial.....	28
2.4 Láseres <i>Mode Locking</i>	32
2.5 Láseres <i>Mode Locking</i> en anillo	36
3. INSTRUMENTALIZACIÓN Y DESARROLLO DE INTERFASES.....	41
3.1 Caracterización de la curva P-I.....	41
3.1.1 <i>Interfaz Fuente de Corriente – Lector de Potencia.</i>	41
3.1.2 <i>Post-Procesado de los datos</i>	49
3.2 Caracterización del ruido de fase	51
3.2.1 <i>Tipos de ruido.</i>	52
3.2.2 <i>Interfaz del analizador de espectros eléctrico (RFSA)</i>	53
4. RESULTADOS OBTENIDOS	61
4.1 Curva P-I	61
4.1.1 <i>Láser en anillo 1</i>	63
4.1.2 <i>Láser en anillo 8</i>	68
4.2 Medidas de espectro óptico.....	71
4.2.1 <i>Láseres en anillo 1 y 8.</i>	72
4.2.2 <i>Láser en anillo 6</i>	76
4.3 Medidas de espectro eléctrico.....	78
5. CONCLUSIONES Y TRABAJOS FUTUROS	83
6. PRESUPUESTO	87
7. ANEXOS – EXTRACTOS DE CÓDIGO.....	89

ÍNDICE GENERAL

8.1 Anexo A – Análisis del régimen estático.....	90
8.2 Anexo B – Respuesta en pequeña señal.....	92
8.3 Anexo C – Respuesta transitoria “Forward Euler”	94
8.4 Anexo D – Respuesta transitoria “Forward Euler normalizado”	95
8.5 Anexo E – Inicialización del interfaz	96
8.6 Anexo F – Conexión/Desconexión.....	97
8.7 Anexo G – Control de Temperatura	99
8.8 Anexo H – Panel de corriente.....	101
8.9 Anexo I – Lectura de potencia.....	104
8.10 Anexo J – Representación de la Curva P-I	105
8.11 Anexo K – Cierre ordenado.....	109
8.12 Anexo L – Control de parámetros (RFSA)	110
8.13 Anexo M – Mostrar Traza	116
8.14 Anexo N – Caracterización del generador de RF.....	118
8.15 Anexo O – Funciones de configuración	120
8.16 Anexo P – Pintar Ruido de Fase	125
8.17 Anexo Q – Guardar trazas	127
8.18 Anexo R – SOA-XN-OEC-1550	129
8.19 Anexo S – SOA-L-OEC-1550.....	132
8. GLOSARIO	135
9. REFERENCIAS	137

Índice de figuras

Figura 1.1. Interfaz para la caracterización de la Curva P-I.....	2
Figura 1.2. Interfaz para la caracterización del ruido de fase.....	2
Figura 1.3. Entorno de desarrollo	3
Figura 1.4. Estación de Puntas Wentworth Laboratories PML 8000	4
Figura 1.5. ILX Lightwave LDC 3724	4
Figura 1.6. Newport 842-PE.....	5
Figura 1.7. Analizador de espectros eléctrico Anritsu MS2668C (arriba) y Generador de RF Anritsu MG3695A (abajo).....	5
Figura 1.8. Analizador de espectros óptico Yogokawa AQ6370B.....	6
Figura 1.9. Interfaz del OSA en el PC	6
Figura 1.10. Agilent 82357B USB/GPIB Interface (izquierda), bus de comunicaciones GPIB (derecha).....	7
Figura 1.11. Mapa de Conexiones (Agilent IO).....	8
Figura 1.12. Láseres en anillo.....	9
Figura 1.13. Láseres en anillo asimétricos	10
Figura 1.14. Láseres en anillo simétricos	10
Figura 2.1. Estructura de bandas de un láser semiconductor y sus transiciones radiativas (3)	14
Figura 2.2. Esquema de un láser monolítico con dos caras clivadas (4)	15
Figura 2.3. Esquema de un láser en anillo simétrico (6)	16
Figura 2.4. Curva P-I de un láser clivado (izquierda), Curva P-I de un láser en anillo (derecha)..	17
Figura 2.5. Densidad de fotones vs. Corriente (arriba), Densidad de portadores vs. Corriente (abajo)	21
Figura 2.6. Respuesta en pequeña señal del dispositivo Ridge Waveguide Laser.....	23
Figura 2.7. Respuesta en pequeña señal del dispositivo EMBH Laser.....	23
Figura 2.8. Respuesta transitoria de la densidad de fotones de un dispositivo Ridge Waveguide Laser, I = 60 mA	25
Figura 2.9. Respuesta transitoria de la densidad de fotones y de portadores en un dispositivo EMBH Laser con I = 20 mA	25
Figura 2.10. Respuesta de un sistema subamortiguado	26
Figura 2.11. Respuesta transitoria con ecuaciones normalizadas de un dispositivo Ridge Waveguide Laser, I = 60 mA	28
Figura 2.12. Circuito para medir la curva P-I en VPI Photonics	28
Figura 2.13. Curva P-I inicial	29
Figura 2.14. Curva P-I parámetro α modificado	29
Figura 2.15. Curva P-I parámetro N_o modificado	30

Figura 2.16. Curva P-I parámetro g_o modificado	30
Figura 2.17. Curva P-I parámetro τ_p modificado (<i>InternalLoss</i>)	31
Figura 2.18. Curva P-I parámetro τ_p modificado (<i>FacetReflectivity</i>)	31
Figura 2.19. Curva P-I parámetro τ_n modificado (<i>BimolecularRecombination</i>)	32
Figura 2.20. Regímenes de funcionamiento Mode-locking frente a corriente y tensión inversa (6)	33
Figura 2.21. Láser <i>Passive Mode Locking</i> (4)	34
Figura 2.22. Corriente de salida vs corriente de entrada en un láser en anillo modo CCW (a) y modo CW (b). Tres regímenes de operación (I) bidireccional, (II) biestable y (III) inestable (18)..	37
Figura 2.23. Espectro obtenido para el modo CW de un láser en anillo para la corriente de 160mA(a) y 175mA (b) (18)	37
Figura 2.24. Salida de la potencia óptica de un láser en anillo por ambas guías (corta y larga)(6)	37
Figura 2.25. Espectro de un láser en anillo en régimen de operación bidireccional(6)	38
Figura 2.26. Láseres en anillo simétricos	39
Figura 2.27. Láseres en anillo asimétricos	39
Figura 3.1. Panel de Conexión (Interfaz 1)	42
Figura 3.2. Mensaje de información	43
Figura 3.3. Panel de control de Temperatura (Interfaz 1)	44
Figura 3.4. Panel de control de la Corriente (Interfaz 1)	45
Figura 3.5. Panel para la lectura de potencia (Interfaz 1)	45
Figura 3.6. Diagrama de flujo del botón Pintar	46
Figura 3.7. Panel para la caracterización de la curva P-I (Interfaz 1)	47
Figura 3.8. Mensaje de error por exceder el límite de corriente (Interfaz 1)	47
Figura 3.9. Representación grafica de la curva P-I (Interfaz 1)	47
Figura 3.10. Ventana de dialogo para guardar la curva P-I (Interfaz 1)	48
Figura 3.11. Ventana de dialogo para cerrar la interfaz (Interfaz 1)	48
Figura 3.12. Cálculo de la corriente umbral mediante ajuste lineal (20)	49
Figura 3.13. Cálculo de la corriente umbral mediante ajuste por dos segmentos (20)	50
Figura 3.14. Cálculo de la corriente umbral mediante la primera derivada (20)	50
Figura 3.15. Cálculo de la corriente umbral mediante la segunda derivada (20)	51
Figura 3.16. Ruido de amplitud (rojo) y ruido de fase (verde) en un pulso(12)	52
Figura 3.17. Análisis del Ruido de fase de SSB y comparación con las especificaciones del fabricante (derecha) (12)	53
Figura 3.18. Panel de conexión (Interfaz 2).....	54
Figura 3.19. Panel de parámetros (Interfaz 2).....	54
Figura 3.20. Diagrama de flujo de los botones fijar frecuencia central, fijar RBW y fijar SPAN..	55
Figura 3.21. Panel de traza (Interfaz 2)	56
Figura 3.22. Panel del Ruido de Fase (Interfaz 2)	57
Figura 3.23. Ventana dialogo para cargar configuración (Interfaz 2)	59
Figura 4.1. Localización de las puntas y soporte de la fibra óptica.....	62
Figura 4.2. Alineamiento de la fibra	62
Figura 4.3. Láser en Anillo 1	63
Figura 4.4. Láser en anillo 1 (2-SA a -1V)	64
Figura 4.5. Láser en anillo 1 (2-SA a -2V)	64
Figura 4.6. Láser en anillo 1 (SA inferior en OC)	66
Figura 4.7. Láser en anillo 1 (SA inferior a -2V)	66
Figura 4.8. Láser en anillo 1 (SA superior en OC)	67
Figura 4.9. Láser en anillo 1 (SA superior a -2V).....	68
Figura 4.10. Láser en Anillo 8.....	68

ÍNDICE DE FIGURAS

Figura 4.11. Láser en anillo 8 (SA en OC)	69
Figura 4.12. Láser en anillo 8 (SA a 0V)	69
Figura 4.13. Láser en anillo 8 (SA a -1V).....	70
Figura 4.14. Láser en anillo 8 (SA a -2V).....	70
Figura 4.15. EDFAs (NORTEL).....	73
Figura 4.16. Espectro Óptico del anillo 1 (Salida del acoplador 10%)	73
Figura 4.17. Espectro Óptico del anillo 1 (con 2 EDFAs).....	74
Figura 4.18. Espectro Óptico del anillo 8 (Salida del acoplador 10%).....	74
Figura 4.19. Espectro Óptico del anillo 8 (Salida del acoplador 10%, detallada)	75
Figura 4.20. Espectro Óptico del anillo 8 (con 2 EDFAs)..	75
Figura 4.21. Láser en anillo 6	76
Figura 4.22. Espectro Óptico del anillo 6	76
Figura 4.23. Espectro Óptico del anillo 6 (con SOA)	77
Figura 4.24. SOA-XN-OEC-1550	77
Figura 4.25. Conexiones SOA-XN-OEC-1550..	78
Figura 4.26. Set-Up del experimento	78
Figura 4.27. u2t XPDV2020R	79
Figura 4.28. Ruido de Fase del Anillo 6 (escala logarítmica)..	79

Índice de tablas

Tabla 2.1. Parámetros de Ridge Waveguide Laser y EMBH Laser (7)	20
Tabla 2.2. Frecuencia propia del sistema	26
Tabla 3.1. Configuración para la caracterización del generador	58
Tabla 4.1. Resultados Anillo 1 - 2SA	64
Tabla 4.2. Resultados Anillo 1 – SA_inferior	66
Tabla 4.3. Resultados Anillo 1 – SA_superior	67
Tabla 4.4. Resultados Anillo 8.....	71
Tabla 4.5. Longitud total de la cavidad/ Frecuencia (GHz)	72
Tabla 4.6. Trazas para el análisis del ruido de fase	80
Tabla 6.1. Material del proyecto	88
Tabla 6.2. Tareas del proyecto	88
Tabla 6.3. Coste del proyecto	88

Capítulo 1

Motivación y objetivos

En el entorno de un laboratorio de investigación es importante automatizar la adquisición de datos de los diversos instrumentos de medida, facilitando la toma masiva de estos, permitiendo analizar la repetitividad de las medidas y facilitando el posterior análisis de los datos.

El entorno en el que se desarrolla este Proyecto de Fin de Carrera es en el laboratorio de investigación de láseres de semiconductor del Grupo de Optoelectrónica y Tecnología Láser. En él es esencial disponer de una herramienta que permita automatizar las medidas que caracterizan un láser de semiconductor.

En el presente proyecto, se desarrollan diversas interfases software que permiten al investigador realizar diversas medidas de caracterización. La primera medida que se ha automatizado es la obtención de la curva potencia óptica frente a corriente inyectada (curva P-I), medida básica en todo dispositivo láser de semiconductor (Fig. 1.1). La segunda medida automatizada resulta fundamental en el campo de los diodos láser mode-locked, como es la medida del ruido de fase (Fig. 1.2). Ambas medidas requieren controlar instrumentos del laboratorio. Un lector de potencia óptica y fuente de corriente para la curva P-I y un analizador de espectros eléctricos (RFSA) la segunda. De los tres, el lector de potencia es conectado mediante USB y los otros disponen de un bus de comunicación GPIB.

En el campo de la automatización de medidas, existen dos entornos disponibles actualmente. Por un lado, la herramienta Labview de National Instrument, y por otro, el entorno de programación científica MATLAB. En este proyecto ha sido escogido este último por ser el entorno de trabajo habitual para el análisis de datos, y resultar más familiar al inicio.

A continuación se pasa a describir el entorno de trabajo para el desarrollo de las medidas, describiendo tanto la ubicación de los dispositivos a medir como los instrumentos de caracterización disponibles.

Capítulo 1: Motivación y objetivos

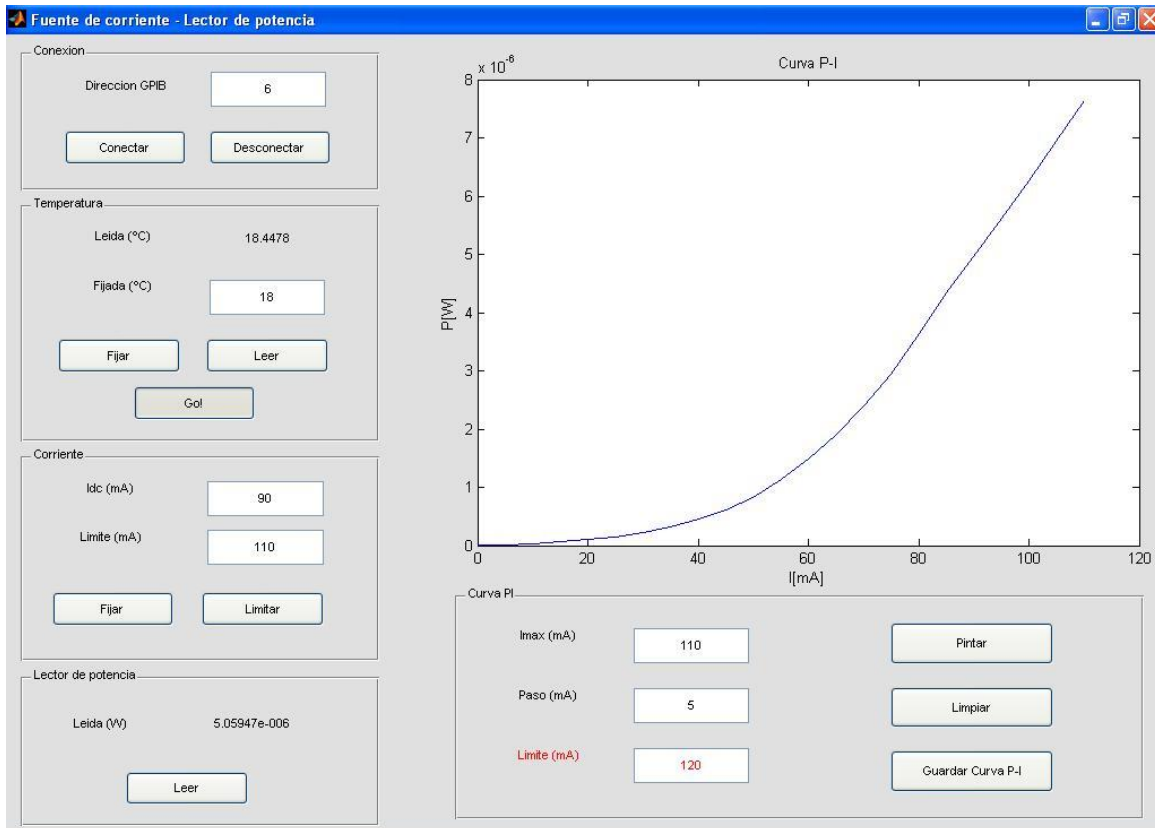


Figura 1.1. Interfaz para la caracterización de la Curva P-I

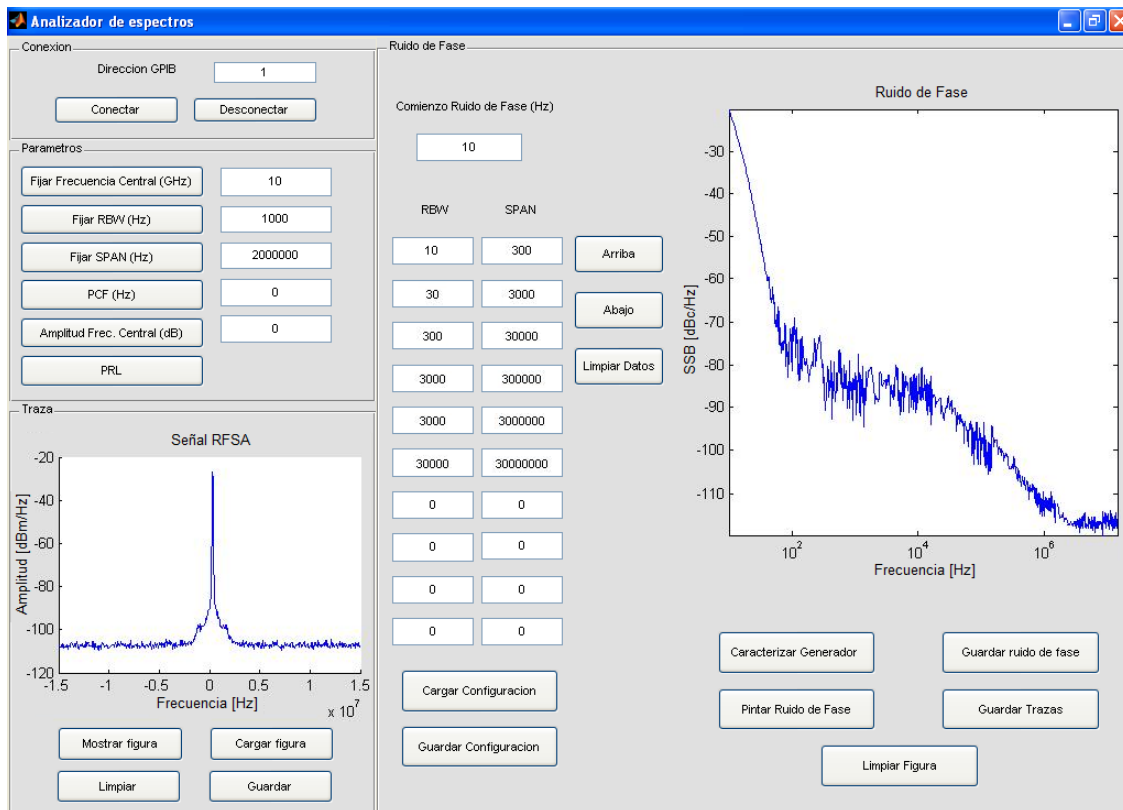


Figura 1.2. Interfaz para la caracterización del ruido de fase.

1.1 Entorno de desarrollo

En este apartado se realiza una descripción de los elementos hardware y software que permiten desarrollar los interfases para la automatización de las medidas y obtener la caracterización de los láseres anteriormente mencionados.

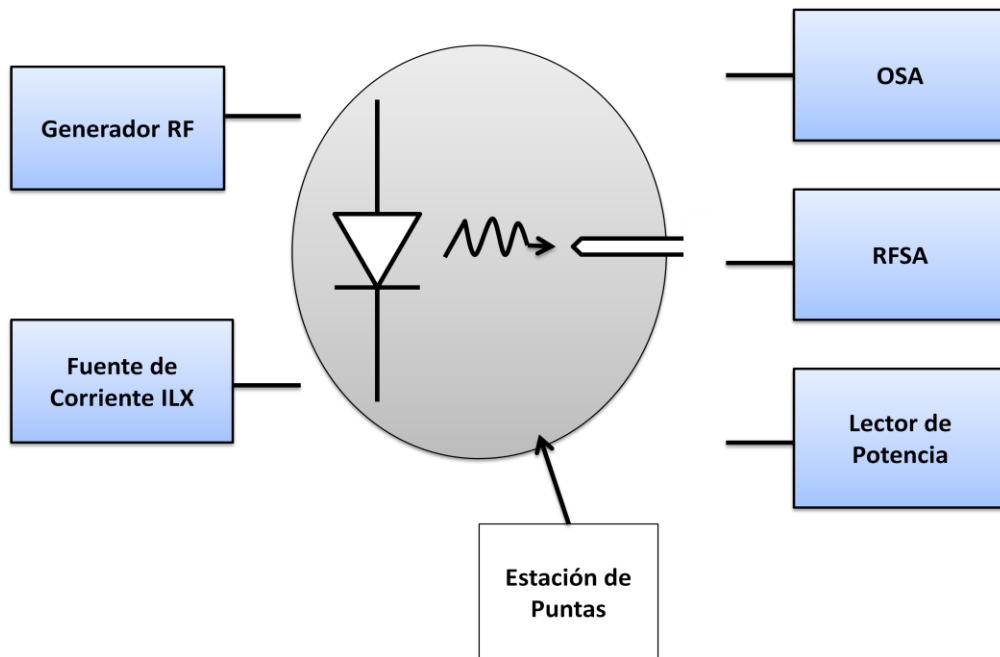


Figura 1.3. Entorno de desarrollo.

En la figura superior en el lado izquierdo se encuentran los dos instrumentos que actúan como entrada del sistema y al lado derecho los tres instrumentos de salida. En el entorno de desarrollo presentado son controlados mediante software, la fuente de corriente (ILX LDC3724B), el medidor de Potencia Óptica y el RFSA (Análisis de espectros de RF).

- **Estación de Puntas Wentworth Laboratories PML 8000**

Para el desarrollo de las medidas en un laboratorio de investigación de láseres de semiconductor es precisa una estación de puntas (Fig. 1.4), donde se localiza el dispositivo. Permite precisión y estabilidad en la inyección de corriente en los dispositivos correspondientes, así como el alineamiento de la fibra con el objetivo de conseguir una máxima potencia para una corriente fijada.

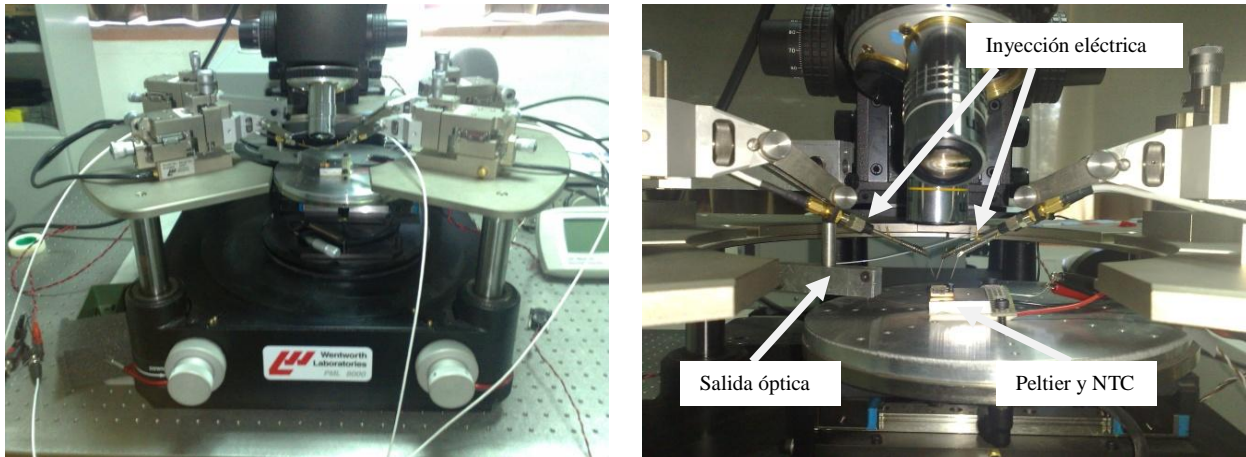


Figura 1.4. Estación de Puntas Wentworth Laboratories PML 8000.

- **Fuente de Corriente ILX (ILX Lightwave LDC 3724)**

Los láseres de semiconductor operan suministrándoles una corriente de polarización, generando una potencia óptica proporcional a la corriente inyectada. La potencia óptica también depende de la temperatura a la que se encuentre el dispositivo. Por este motivo, un controlador de un diodo láser es un equipo que combina una fuente de corriente (que suministra una corriente constante sin transitorios abruptos que dañan el dispositivo láser), con un controlador PID para regular la temperatura (permitiendo utilizar cualquier tipo de sensor de temperatura NTC, PTC, LM335) y suministra la corriente a un elemento peltier. El control de la corriente de salida permite un encendido y apagado seguro del láser, ya que cuando la salida es activada la fuente cuenta con un circuito de comienzo lento permitiendo que la corriente se desvíe al láser lentamente. Además de proporcionar la posibilidad de fijar un límite en la corriente nominal de salida.

La comunicación entre esta fuente de corriente y el PC se realizan a través del bus de comunicaciones GPIB definido por el estándar IEEE-488.

A través de los comandos especificados por el fabricante se establece comunicación con la fuente y se puede determinar el valor de temperatura para estabilizar la medida, indicando también las corrientes que se corresponderán a los puntos de la curva P-I.



Figura 1.5. ILX Lightwave LDC 3724.

- **Lector de Potencia Óptica Newport 842-PE**

El rango de potencia del lector de potencia se encuentra entre 5pW y 10kW, y la conexión con el PC se realiza físicamente a través de una conexión USB.



Figura 1.6. Newport 842-PE.

- **Analizador de espectros eléctrico Anritsu MS2668C**

El analizador de espectros eléctrico (Fig. 1.7) se conecta al PC a través de el bus de comunicaciones GPIB, permitiendo a través de los comandos especificados por el fabricante del instrumento obtener y modificar los parámetros influyentes en la medida de un espectro eléctrico (RBW, SPAN, frecuencia central).

Con la modificación de estos parámetros es posible la caracterización del ruido de fase de un dispositivo láser mediante varios barridos de la señal.

El generador de RF utilizado se encuentra en la Fig. 1.7 (abajo). Como se muestra en dicha figura el generador se utiliza para realizar una previa calibración del interfaz del ruido de fase. Para realizar esta calibración se genera un tono a una frecuencia dada y se introduce al analizador de espectros eléctrico, permitiendo el análisis del ruido de fase del generador.



Figura 1.7. Analizador de espectros Anritsu MS2668C (arriba) y Generador de RF Anritsu MG3695A (abajo).

- **Analizador de espectros óptico Yokogawa AQ6370B**

El analizador de espectros óptico es utilizado previamente a la introducción de la señal óptica en el RFSA y posterior análisis del ruido de fase. Con este instrumento se pueden observar los modos enganchados en fase de los láseres de semiconductor mode-locked.

El OSA proporciona un interfaz para el usuario (Fig. 1.9) con acceso a través del PC. La conexión entre el PC y el OSA se realiza a través de Ethernet esta conexión permite una transferencia de datos más rápida que la comunicación a través de GPIB.



Figura 1.8. Analizador de espectros óptico Yogokawa AQ6370B.

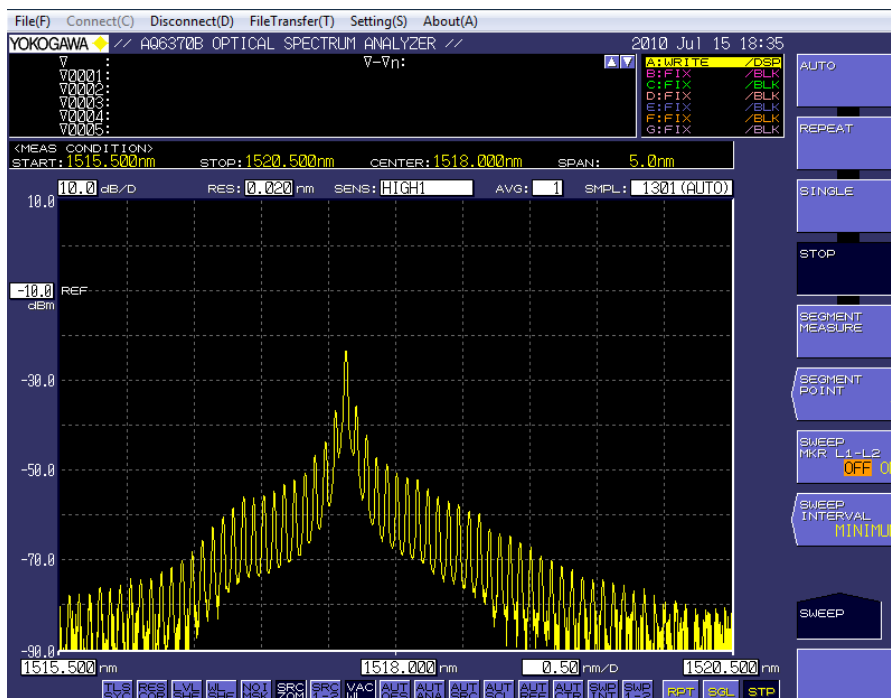


Figura 1.9. Interfaz del OSA en el PC.

- **Bus de comunicaciones GPIB**

GPIB es un estándar de conexión que permite la comunicación de un ordenador con instrumentos electrónicos de medida. Entre los instrumentos especificados anteriormente presentes en el laboratorio de investigación pueden ser conectados por GPIB la fuente de corriente, el RFSA, el OSA y el generador de RF. Para el desarrollo del presente proyecto como ha sido aclarado el OSA es conectado mediante TCP/IP y el generador de funciones únicamente es utilizado para calibrar el interfaz del ruido de fase.

El bus GPIB permite conectar los dos dispositivos (fuente de corriente y RFSA) mediante una conexión en cadena, siendo controlados desde el PC. La conexión con el PC es realizada a través del interfaz facilitado por el dispositivo *Agilent 82357B USB/GPIB Interface* (Fig. 1.10) el cual proporciona una conexión directa desde un puerto USB de el ordenador a los instrumentos de medida.

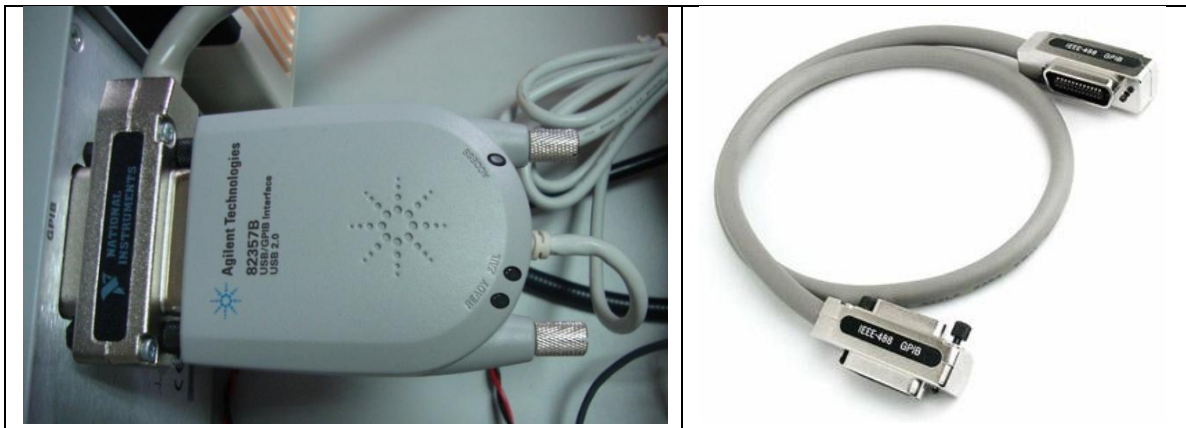


Figura 1.10. Agilent 82357B USB/GPIB Interface (izquierda), bus de comunicaciones GPIB (derecha).

A través de la aplicación *Agilent IO Control* (aplicación gratuita de Agilent) se controlan los instrumentos conectados por puertos GPIB, serie o USB. Una de las facilidades que es proporcionada por las librerías de Agilent es la posibilidad de verificar la conexión de los instrumentos permitiendo a la vez la comunicación (*Agilent Connection Expert*). Esta última aplicación muestra un mapa de las conexiones del PC con los instrumentos de E/S. Sobre este mapa de conexiones se puede acceder a la información de cada instrumento, como es la dirección GPIB.

En el panel frontal correspondiente a las conexiones de los instrumentos utilizados (Fig. 1.11) se puede observar que la conexión del lector de potencia óptica se realiza por medio de un puerto serie (ASRL3). Esto determina como son los comandos que le son enviados al lector de potencia y como se debe abrir la conexión con el objeto mediante la herramienta software. Aunque la conexión física del lector de potencia al PC se realice mediante un puerto USB.

En la Fig. 1.11 se observan las especificaciones que permite visualizar la aplicación de Agilent del instrumento seleccionado. En este caso el elemento seleccionado es el RFSA.

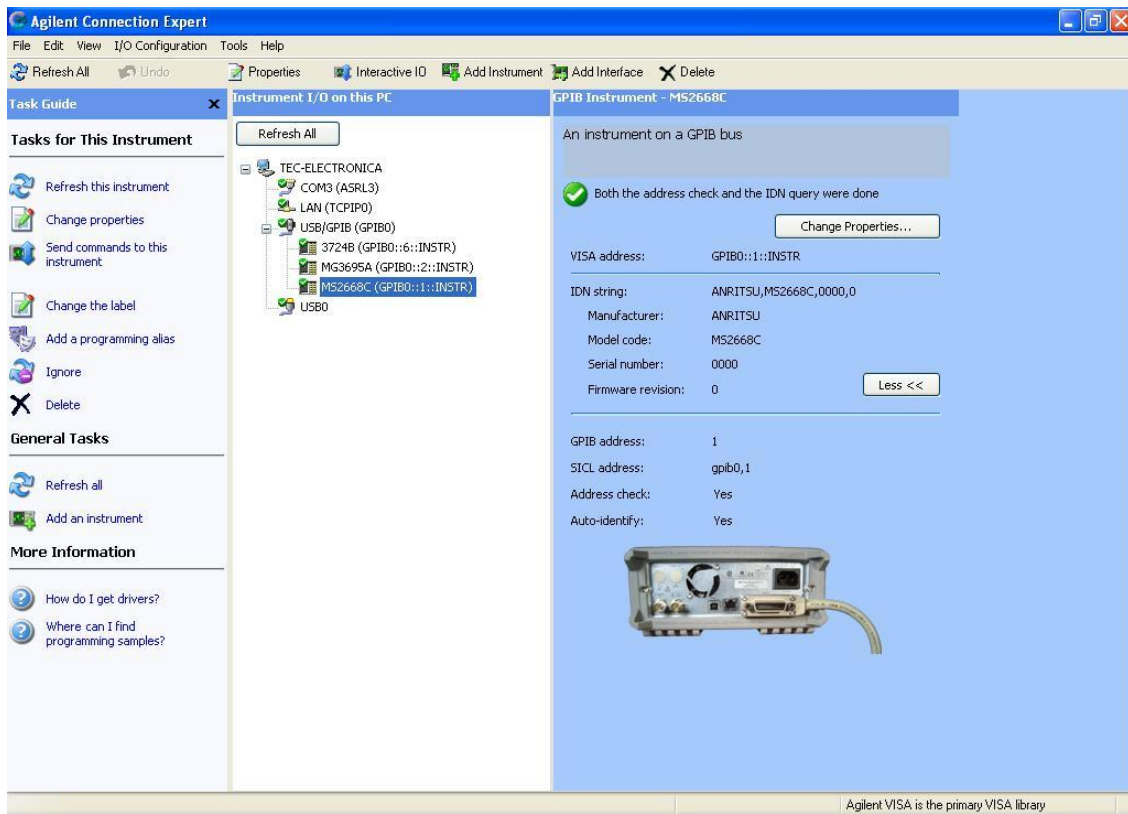


Figura 1.11. Mapa de Conexiones (Agilent IO).

1.2 Metodología de trabajo

Antes de comenzar a desarrollar los interfases descritos, en este proyecto ha sido necesario realizar un estudio previo, consistente en un estudio mediante modelado de un láser de semiconductor. Mediante este estudio se permite conocer los parámetros que afectan a una de las características fundamentales de un láser como es la corriente umbral. La cual posteriormente es determinada por los dispositivos en el laboratorio. Además se adquiere conocimiento sobre el funcionamiento del dispositivo a través de su respuesta transitoria y su respuesta en frecuencia. Este estudio es realizado mediante dos herramientas software, MATLAB de Mathworks y VPIPhotonics.

Este estudio previo, permite la familiarización con MATLAB influyendo en la elección de la herramienta para el desarrollo de los interfases. Como hemos comentado anteriormente, las herramientas software que permiten la adquisición de datos a través del bus digital GPIB utilizado son MATLAB (Mathworks) y LabVIEW (National Instruments). Como ambas herramientas ofrecen una funcionalidad semejante a los interfases demandados por un investigador y desarrollados en este proyecto, ambos han sido realizados únicamente con MATLAB.

Finalmente y como objetivo fundamental del proyecto, estos interfases han sido aplicados en el estudio de láseres de semiconductor mode-locking. Los dispositivos analizados presentan

estructura de anillo, cuya principal ventaja reside en que pueden ser implementados mediante óptica integrada. Los láseres de óptica integrada son especialmente importantes ya que permiten reemplazar un conjunto de elementos fabricados e interconectados individualmente en único chip. Los láseres de óptica integrada utilizados en este estudio, diseñados en la Universidad Carlos III de Madrid y fabricados en la plataforma europea de InP JePPIX, se presentan en la Fig. 1.12 de la que podemos destacar que contienen estructuras asimétricas y simétricas. Las estructuras asimétricas y simétricas se distinguen en las fotografías presentes en la Fig. 1.13 y Fig. 1.14 respectivamente.

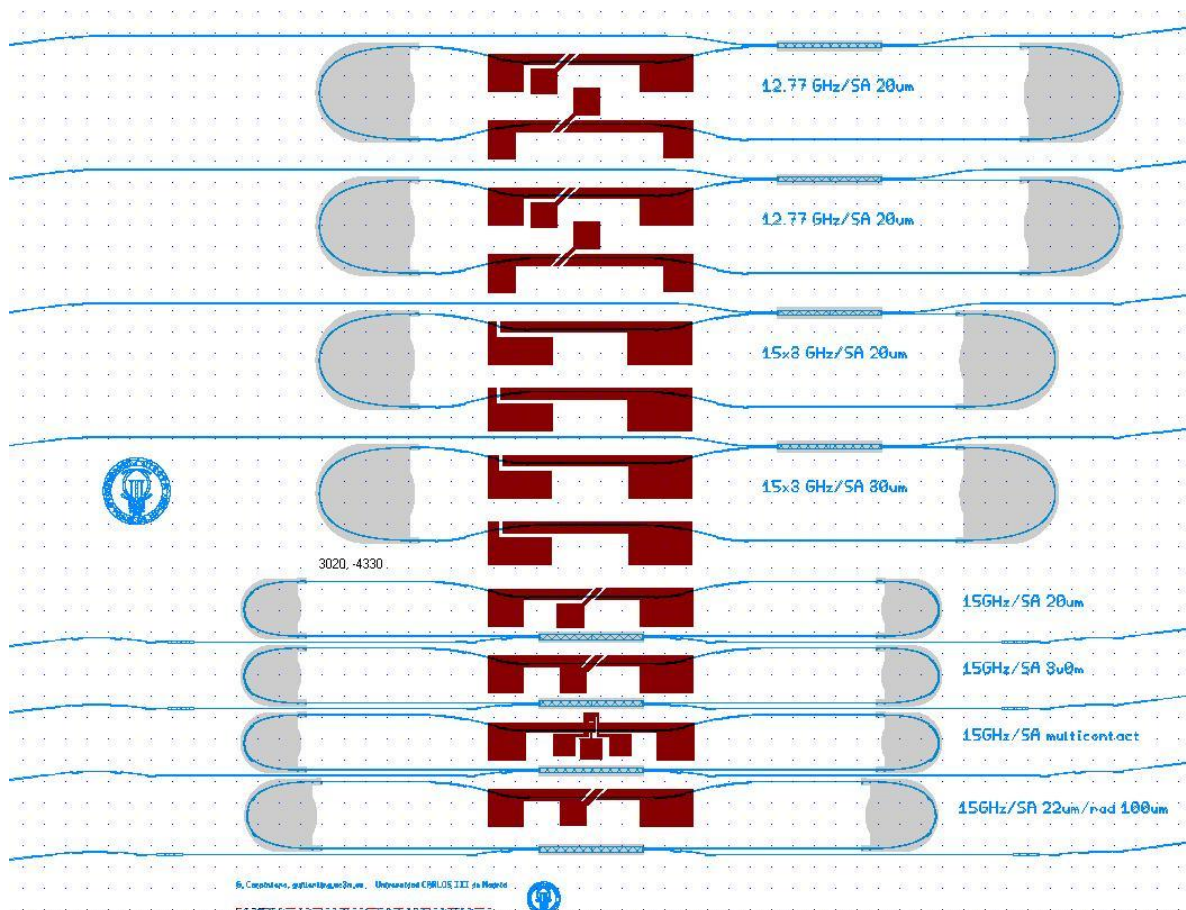


Figura 1.12. Láseres en anillo

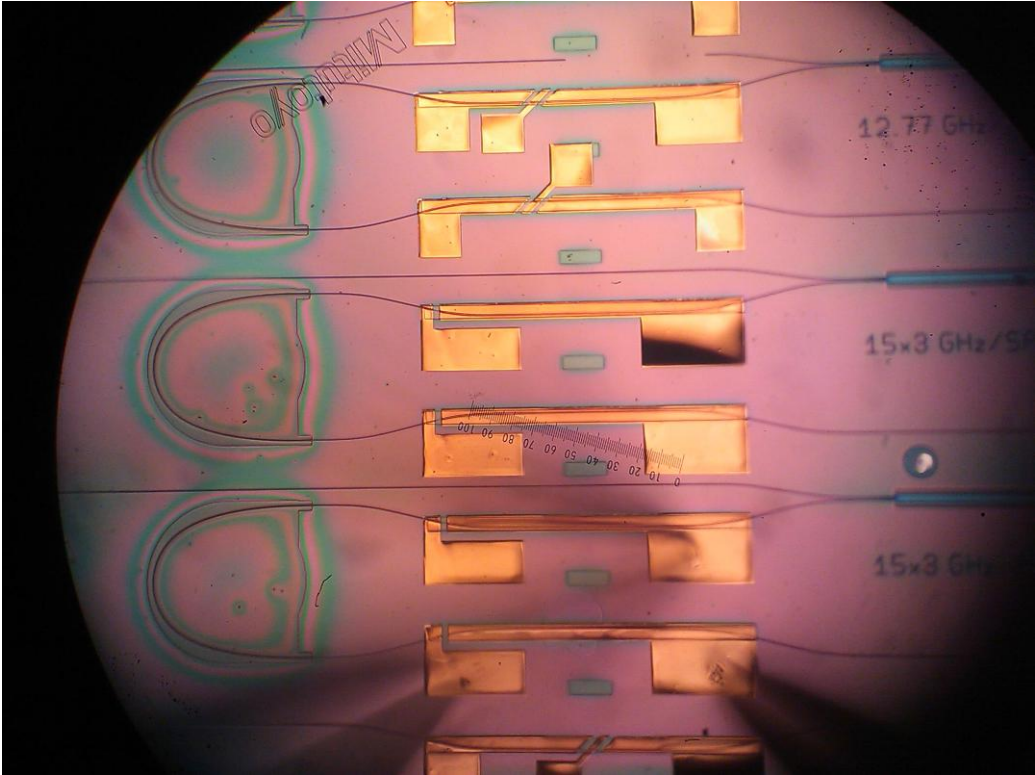


Figura 1.13. Láseres en anillo asimétricos.

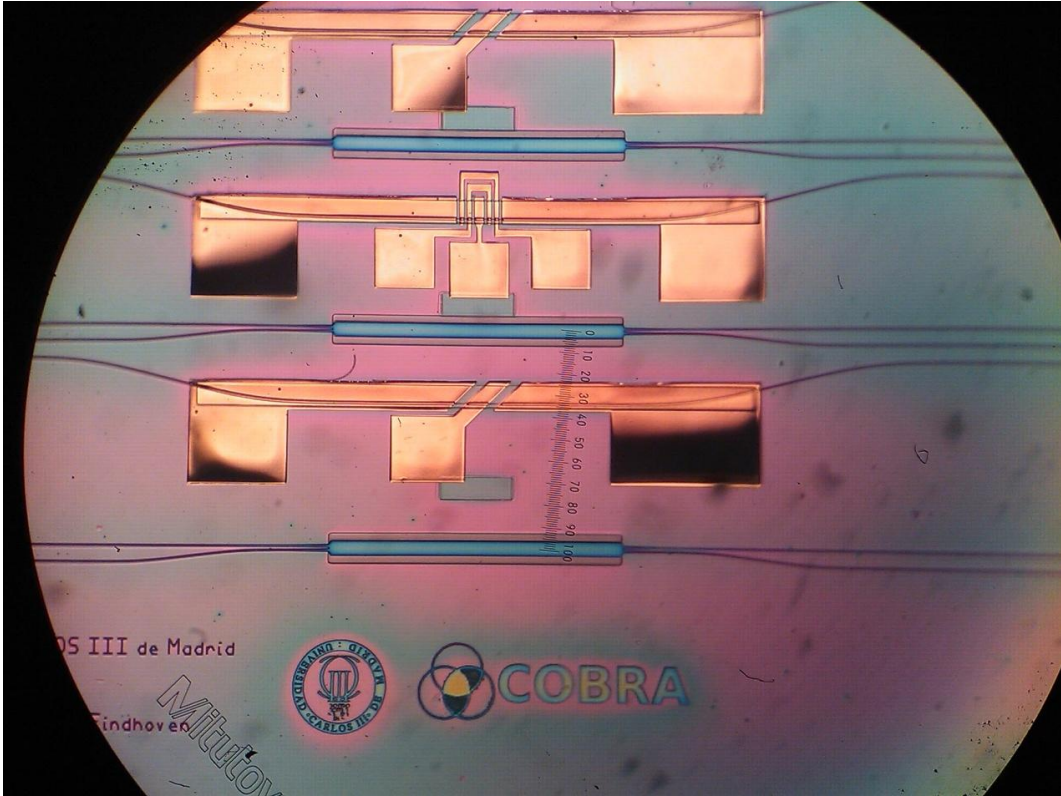


Figura 1.14. Láseres en anillo simétricos

La estructura del proyecto es la siguiente: En el **capítulo 1** son descritos los objetivos del presente proyecto y el entorno de desarrollo en el que se realiza. En el **capítulo 2** se presentan los principios básicos de funcionamiento de los láseres de semiconductor, así como las técnicas de modelado empleadas. En el **capítulo 3** son descritos los dos interfases desarrollados para la caracterización de láseres de semiconductor y también se presentan las diferentes formas de realizar el post-procesado de los datos obtenidos en una curva P-I. Los resultados obtenidos son presentados en el **capítulo 4** dividiéndose según la medida realizada en el análisis de la curva P-I, espectro óptico y espectro eléctrico. Finalmente, en el **capítulo 5** se exponen las conclusiones obtenidas de la caracterización de los láseres en anillo presentados anteriormente.

Capítulo 2

Introducción a los láseres de semiconductor

En este capítulo se presenta una introducción y descripción general del funcionamiento de los láseres de semiconductor. Se expone el principio de funcionamiento de estos dispositivos y se realiza un análisis de las ecuaciones de tasa para distintos ejemplos de estos láseres. El objetivo final de este documento es identificar los parámetros característicos de este tipo de láseres, como son la corriente umbral y la eficiencia cuántica externa, así como presentar las características propias de los láseres del tipo Mode Locking.

2.1 Introducción

Los láseres de semiconductor son considerados las mejores fuentes de luz para comunicaciones por fibra óptica a alta tasa de bit y sistemas de procesamiento de señal debido a que sus principales cualidades son: su pequeño tamaño, su alta eficacia, el bajo coste y la capacidad de modularlos a alta frecuencia (tasas de decenas de gigahercios), introduciendo las variaciones directamente en la corriente que se inyecta en el dispositivo (1).

En las comunicaciones por fibra, se emplean dos técnicas de multiplexación para aumentar la capacidad de transmisión: multiplexación por división de la longitud de onda (WDM) y multiplexación por división del tiempo (TDM). Actualmente se utiliza una combinación de ambas en la que cada longitud de onda WDM establece un canal de transmisión en el que se utiliza TDM. Los láseres de semiconductor actuales son capaces de sintonizar su longitud de onda para adaptarse a los estándares WDM, y permiten además generar pulsos ultracortos a tasas de repetición ultrarrápidas para generar la señal TDM, además de tener la capacidad de modular y codificar. Los láseres de semiconductor son los candidatos para esta función por las ventajas indicadas previamente(2).

2.2 Principio de funcionamiento

Los láseres de semiconductor son básicamente un oscilador a frecuencias ópticas (193 THz para 1550nm). Como todo oscilador requieren de dos elementos fundamentales para operar, un amplificador y una red de realimentación.

Amplificación Óptica

Los fenómenos de amplificación a frecuencias ópticas están íntimamente relacionados con los procesos de transición de los portadores de carga (electrones y huecos) entre bandas de valencia y conducción de los materiales, y su iteración generando fotones cuya energía coincide con la energía del gap. Los láseres de semiconductor son dispositivos en los que estos fenómenos se producen cuando una unión p-n se polariza en directa con un voltaje externo, haciendo posible la creación de una inversión de población en el material (electrones en la banda de valencia y huecos en la banda de conducción). De esta manera simplificada, la estructura de bandas y los procesos de iteración con la luz en un láser de semiconductor se muestran, en la Fig. 2.1. Cabe destacar la condición necesaria para la transición entre la banda de conducción y la banda de valencia se produzca es que la energía del fotón verifique (2.1), donde h es la constante de Planck y las energías se corresponden a cada una de las bandas (valencia y conducción).

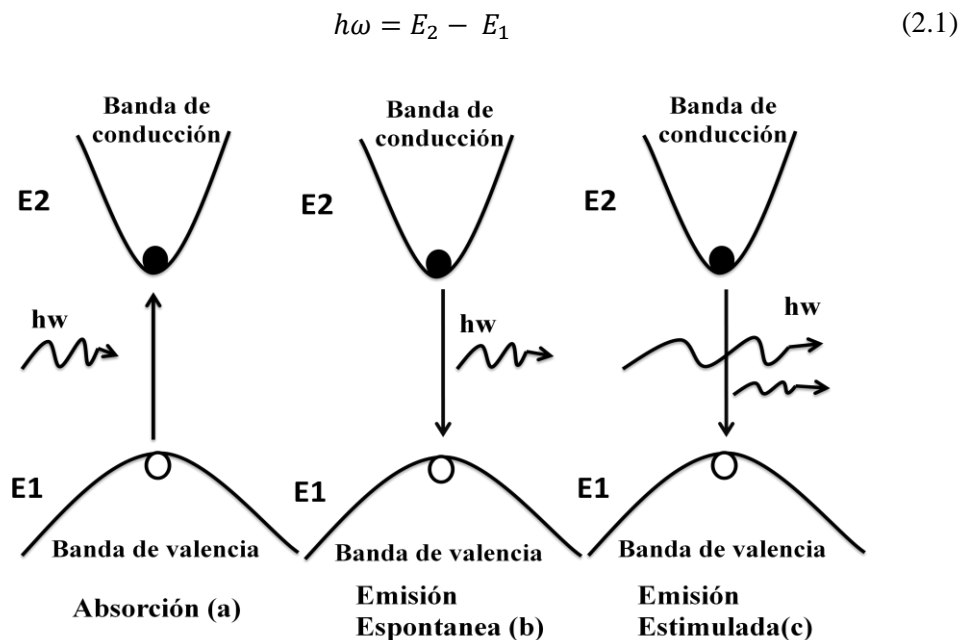


Figura 2.1. Estructura de bandas de un láser de semiconductor y sus transiciones radiativas (3).

La Fig. 2.1 (a) muestra un proceso de absorción de fotones, que produce pérdidas ópticas en el material. Este proceso ocurre cuando un fotón se propaga por un material, siendo absorbido por este produciendo una transición de un electrón de la banda de valencia a la de conducción.

Los procesos de liberación de fotones son, por un lado la emisión espontánea –presentado en la Fig. 2.1 (b) – y por otro, la emisión estimulada - Fig. 2.1(c). En la emisión espontánea no

existe ningún factor externo que induzca al electrón a pasar de la capa de conducción a la de valencia. En la emisión estimulada, también conocido como efecto láser, un fotón induce al electrón a pasar a la capa de valencia liberando otro fotón, si en este caso existe retroalimentación y selectividad se obtendría emisión coherente (3). Este último es propiamente un efecto de amplificación de luz. Debido a que compite con la absorción, solo cuando la corriente a través de la unión excede un determinado valor, se alcanza una inversión de población para la cual la tasa de emisión de fotones debida a la recombinación estimulada supera a la tasa de absorción producida por la generación de pares electrón – hueco. El nivel de inversión necesaria se denomina transparencia, pues el material se convierte en transparente al paso de luz.

Esto permite establecer la condición necesaria para conseguir emisión láser, debiendo la tasa de emisión estimulada superar las distintas fuentes de pérdidas. La ganancia óptica está directamente relacionada con la tasa de emisión estimulada, es linealmente dependiente con la concentración de portadores y varía parabólicamente con la longitud de onda (2.2).

$$g = g_o(n - n_t) + b(\lambda - \lambda_p)^2 \quad (2.2)$$

La ganancia óptica no es la única condición para que el láser opere, ya que como hemos indicado anteriormente, es necesario proporcionar una red de realimentación óptica.

Red de realimentación

La realimentación óptica consiste en crear una cavidad o resonador óptico, dentro del cual deberá localizarse el medio amplificador. Existen diferentes formas de hacerlo.

La más sencilla es mediante el clivado (Fig. 2.2) de dos caras del dispositivo, que en realidad son parcialmente espejos reflectores. De esta forma se obtiene selectividad en una dirección y únicamente los fotones que viajan en ese eje son reflejados adquiriendo máxima ganancia y contribuyendo al proceso de emisión estimulada. Aunque esta forma presenta problemas de integración.

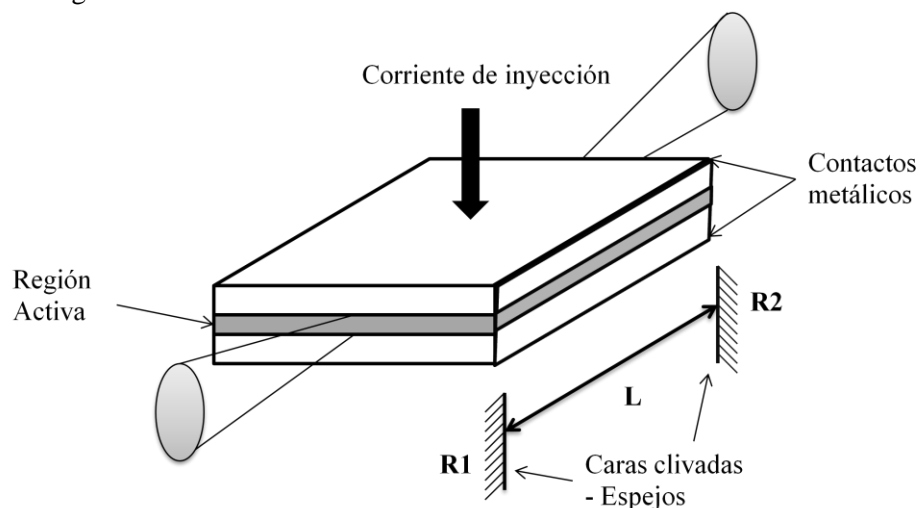


Figura 2.2. Esquema de un láser monolítico con dos caras clivadas (4).

La realimentación óptica puede también conseguirse mediante resonadores ópticos en anillo, creando estructuras circulares. Estos últimos han tomado elevada importancia tras sus primeras demostraciones debido a sus aplicaciones en óptica integrada. No requieren clivado de sus caras para producir realimentación óptica, y por tanto se puede establecer de forma precisa la longitud del resonador. Además, estas estructuras son especialmente adecuadas para ser integradas en circuitos ópticos. Son prometedores en aplicaciones de filtrado de longitud de onda, permitiendo régimen unidireccional determinando la dirección de propagación de los fotones, y en aplicaciones de multiplicación y demultiplexación (5).

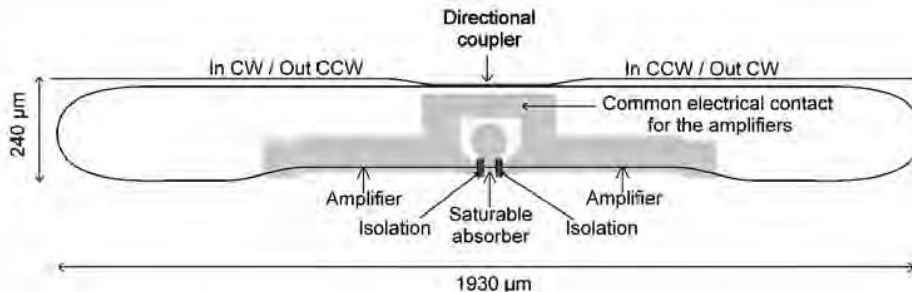


Figura 2.3. Esquema de un láser en anillo simétrico (6).

Aunque la emisión espontánea y estimulada de fotones suceden mediante la aplicación de una determinada corriente al dispositivo, el láser no genera emisión coherente de fotones hasta que la corriente no excede un valor denominado corriente umbral. Para obtener la condición de corriente umbral se necesita que el campo óptico en la cavidad se reproduzca por sí mismo (4).

Curva característica Potencia frente Corriente

El valor fundamental que se observa en la curva característica Potencia frente Corriente es la corriente umbral. A partir de este valor el dispositivo se encuentra en zona láser. Para superar la densidad umbral que determina la corriente umbral, primeramente se debe obtener la densidad de transparencia y posteriormente la densidad umbral. La densidad de transparencia la determina el material y su valor se consigue cuando la ganancia g en el dispositivo, es igual a las pérdidas α . Aunque el láser cumpla esta condición el dispositivo no se encuentra en zona láser, ya que para ello debe superar la densidad umbral, esto implica superar las pérdidas debidas a la estructura. En un láser clivado las pérdidas debidas a la estructura se corresponden con las de los espejos, mientras que en una cavidad circular se deben superar las pérdidas debidas a un acoplador interferencia multimodo (MMI) que permite obtener las dos salidas correspondientes a las dos direcciones de propagación dentro del anillo.

La corriente umbral varía de un láser clivado a un láser en anillo al igual que la forma de su curva P-I (Fig. 2.4). En la curva P-I correspondiente a un anillo se pueden observar los cambios bruscos de potencia debidos al modo de funcionamiento del anillo para cada valor de la corriente. Por otra parte en un dispositivo monolítico la potencia crece monótonamente hasta un valor de saturación.

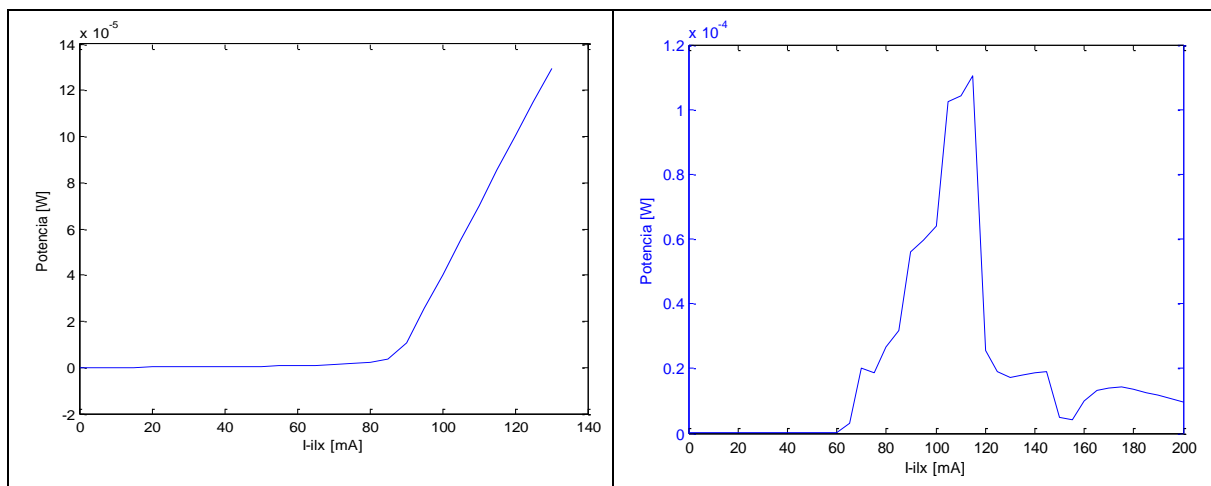


Figura 2.4. Curva P-I de un láser clivado (izquierda), Curva P-I de un láser en anillo (derecha).

2.3 Modelado de los dispositivos

En este apartado se presentan las ecuaciones de tasa las cuales describen el funcionamiento de un láser de semiconductor mediante una aproximación fenomenológica. A partir de ellas se puede determinar la corriente umbral y obtener la respuesta transitoria y la respuesta en pequeña señal de un diodo láser. Por otra parte, los parámetros de diseño de la cavidad de un diodo láser varían el valor de la corriente umbral. Estos también serán analizados a continuación.

2.3.1 Ecuaciones de tasa

Entre los primeros métodos que se suelen utilizar para modelar un láser de semiconductor se encuentran las ecuaciones de tasa (2.3) (2.4). Este modelo representa una descripción fenomenológica de los procesos fundamentales que intervienen en el dispositivo y permite obtener algunas de las características más relevantes del mismo. Las variables en estas ecuaciones son la inyección de corriente ($I [A]$), la densidad de portadores ($N [m^{-3}]$) y la densidad de fotones ($S [m^{-3}]$) (3).

$$\frac{dS}{dt} = \Gamma S G_{in}(N) - S/\tau_p + \Gamma \beta \frac{N}{\tau_n} - A_m S \quad (2.3)$$

$$\frac{dN}{dt} = -G_{in}(N)S - \frac{N}{\tau_n} + \frac{I}{qV} \quad (2.4)$$

En la tasa de variación del número de fotones dentro de una cavidad láser (2.3) el primer término hace referencia a los fotones generados por emisión estimulada, el cual está determinado por el coeficiente de ganancia $G_{in}(N)$ por unidad de tiempo. El tiempo de vida de un fotón es

τ_p por lo tanto el segundo término determina la tasa de pérdidas de fotones de la cavidad. El tercer término esta dado por la emisión espontanea causada por la recombinación espontanea de electrones con huecos. Y el último término es la absorción por unidad de tiempo (3).

El factor Γ representa el factor de confinamiento óptico, y se trata de la fracción de la energía de un modo que realmente contiene la región activa.

En la ecuación de la tasa de variación del número de portadores (2.4) el primer término se corresponde con la perdida de electrones producida por la emisión estimulada, igualmente asociado con el coeficiente de ganancia $G_{in}(N)$. El segundo término representa la tasa de emisión espontanea. La ecuación correspondiente a la tasa de recombinación espontanea de electrones – huecos es (2.5) (3).

$$\frac{N}{\tau_n} = An + Bn^2 + Cn^3 \quad (2.5)$$

Y el último término está determinado por la corriente de conducción, que especifica la tasa I/q que empuja a los electrones en la banda de conducción.

Una aproximación útil de la ganancia óptica $G_{in}(N)$ puede ser representada como:

$$G_{in}(N) \sim \frac{g(N - N_t)}{1 + \epsilon S} \sim g(N - N_t) * (1 - \epsilon S) \quad (2.6)$$

Donde N_t es la densidad de transparencia (3).

Agrupando términos se puede obtener ecuaciones de tasa de gran utilidad partiendo de (2.3) y (2.4). La tasa de escape por las caras y la tasa de absorción pueden ser agrupadas dando lugar a una tasa de pérdidas de fotones (2.7) en único término en la ecuación de tasa de la variación de fotones (2.8) (3).

$$\frac{S}{\tau'_p} = \frac{S}{\tau_p} + A_m S \quad (2.7)$$

Las ecuaciones de tasa quedan finalmente con esta forma, que es habitual en numerosos textos y referencias:

$$\frac{dS}{dt} = \{\Gamma G_{in}(N) - 1/\tau'_p\}S + \Gamma\beta \frac{N}{\tau_n} \quad (2.8)$$

$$\frac{dN}{dt} = \frac{I}{\alpha} - \frac{N}{\tau_n} - G_{in}(N)S \quad (2.9)$$

Donde se ha aplicado:

$$\alpha = qV \quad (2.10)$$

Las ecuaciones de tasa permiten el análisis de un dispositivo láser de dos formas. La primera es analizando sus características estáticas, considerando que la corriente que circulara por el dispositivo es constante. La otra forma es considerando un régimen dinámico, donde se analiza la respuesta del dispositivo a variaciones temporales en la corriente.

2.3.1.1 Régimen estático

La respuesta estática de un diodo láser depende de las densidades de fotones y portadores en equilibrio cuando la inyección de la corriente es constante. Por ello según (2.8) y (2.9) si son igualadas las derivadas a cero, se pueden calcular las ecuaciones de N_0 y S_0 en el punto de equilibrio.

$$0 = \{\Gamma G_{in}(N_o) - 1/\tau_p'\}S_0 + \Gamma\beta \frac{N_0}{\tau_n} \quad (2.11)$$

$$0 = \frac{I}{\alpha} - \frac{N_0}{\tau_n} - G_{in}(N_o)S_0 \quad (2.12)$$

Combinando las expresiones (2.11) y (2.12) se obtiene (2.13).

$$0 = \frac{I}{\alpha} - \frac{N_0}{\tau_n} - \frac{S_0}{\Gamma\tau_p'} + \Gamma\beta \frac{N_0}{\tau_n} \quad (2.13)$$

En (2.13) el último término es del orden de 10^{-5} esta es la razón por la cual se desprecia obteniendo la expresión definitiva de S_0 .

$$S_0 = \frac{\Gamma\tau_p'}{\alpha} \left(I - \frac{\alpha N_0}{\tau_n} \right) \quad (2.14)$$

Identificando sobre (2.14) se deduce la expresión de la **corriente umbral**.

$$I_{th} = \frac{\alpha N_0}{\tau_n} \quad (2.15)$$

Considerando el valor de $G_{in}(N_o)$ que se corresponde con:

$$G_{in}(N_o) = g_o(N - N_t)(1 - \varepsilon S) \quad (2.16)$$

Se obtiene la densidad de portadores estática.

$$N_0 = N_t + \frac{1}{g_o(1 - \varepsilon S_o)\Gamma\tau_p'} \quad (2.17)$$

Capítulo 2: Introducción a los láseres de semiconductor

El valor de N_t viene definido por la corriente de transparencia, que es la obtenida sin tener en cuenta los espejos del interior del láser.

Los cálculos presentados anteriormente se desarrollan manteniendo la suposición de que el dispositivo está por encima del umbral. El significado de esto se expresa como:

- Encima del umbral $\rightarrow g(N - N_t) \neq 0$
- Debajo del umbral $\rightarrow g(N - N_t) = 0$

Esto se debe a que por debajo del umbral la amplificación del material es menor que las pérdidas, por eso en este caso es predominante el término debido a la emisión espontánea en las ecuaciones de tasa. Eliminando el término de emisión estimulada en las ecuaciones de tasa se puede observar la dependencia de N_0 con respecto a la I (2.18) y la relación entre la densidad de fotones S_0 con la densidad de portadores N_0 .

$$0 = \frac{I}{\alpha} - \frac{N_0}{\tau_n} \quad (2.18)$$

$$0 = -\frac{S_0}{\tau_p'} + \Gamma\beta \frac{N_0}{\tau_n} \quad (2.19)$$

A continuación se presenta la tabla con los parámetros específicos de dos dispositivos láseres diferentes (7). Para estos posteriormente son mostradas las curvas correspondientes a la densidad de fotones y a la densidad de portadores con respecto a la corriente, permitiendo observar el valor de I_{th} .

Parámetro	Ridge Waveguide Laser	EMBH Laser
α	$6.6 * 10^{-35}$	$2.2 * 10^{-35}$
Γ	0.3	0.4
ϵ	$6.7 * 10^{-23}$	$4.5 * 10^{-23}$
τ_p'	10^{-12}	10^{-12}
τ_n	$3 * 10^{-9}$	$3 * 10^{-9}$
N_t	10^{24}	10^{24}
g_0	$3.2 * 10^{-12}$	$2.4 * 10^{-12}$

Tabla 2.1. Parámetros de Ridge Waveguide Laser y EMBH Laser (7)

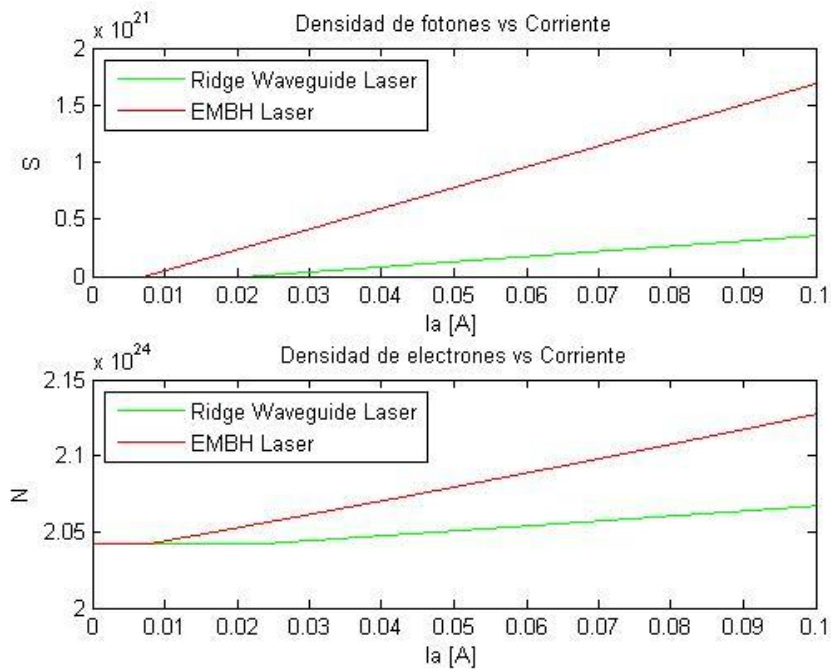


Figura 2.5. Densidad de fotones vs. Corriente (arriba), Densidad de portadores vs. Corriente (abajo)

El código correspondiente a la grafica se encuentra en el Anexo A.

Se debe tener en cuenta que en estas imágenes no ha sido mostrada la pendiente que presenta la densidad de portadores previamente al valor de corriente umbral.

2.3.1.2 Régimen dinámico

En el régimen dinámico se tienen en cuenta las variaciones temporales de la corriente inyectada, siguiendo la evolución debida a estas variaciones en la densidad de portadores y de fotones. Entre las características dinámicas se estudian los parámetros de frecuencia propia y factor de amortiguamiento realizando un análisis de la respuesta en pequeña señal. Además se describe la respuesta transitoria del láser observando la frecuencia de relajación anteriormente obtenida.

2.3.1.2.1 Respuesta en pequeña señal

Para obtener la respuesta en pequeña señal de un dispositivo láser, se debe considerar una variación de la corriente en pequeña señal lo que se supone una variación senoidal de las densidades de fotones y portadores en torno a un punto de equilibrio. Por ello, se debe escoger un punto de trabajo que se encuentre por encima de la corriente umbral. Este valor de corriente determina un valor de densidad de fotones y densidad de portadores a los que se debe sumar el término de variación senoidal correspondiente (2.20) (2.21) (2.22).

$$I = I_0 + \tilde{i}e^{j\omega t} \quad (2.20)$$

$$S = S_0 + \tilde{s}e^{j\omega t} \quad (2.21)$$

$$N = N_0 + \tilde{n}e^{j\omega t} \quad (2.22)$$

Estas expresiones se sustituyen en las ecuaciones de tasa, despreciando aquellos valores que presenten dos o más productos de pequeña señal. Finalmente se igualan los términos que dependen de $e^{j\omega t}$ y se obtienen (2.23) y (2.24).

$$j\omega\tilde{n} = \frac{\tilde{i}}{\alpha} - \frac{\tilde{n}}{\tau_n} - g_o[\tilde{s}(N_o - N_t) + \tilde{n}S_o] \quad (2.23)$$

$$\tilde{s}\left(j\omega + \frac{1}{\tau'_p}\right) = \Gamma g_o [\tilde{s}(N_o - N_t) + \tilde{n}S_o] - \frac{\tilde{s}}{\tau'_p} + \frac{\Gamma\beta\tilde{n}}{\tau_n} \quad (2.24)$$

De esta forma las ecuaciones en pequeña señal quedan definidas con \tilde{s} , \tilde{n} e \tilde{i} . Con estas variaciones se obtiene la función de transferencia de la modulación en amplitud del láser, $H_m(\omega) = \frac{\tilde{s}}{\tilde{i}}$. Esta función de transferencia es normalizada a su valor estacionario obteniendo (2.25).

$$\frac{H_m(\omega)}{H_m(0)} = \frac{\omega_0^2}{-\omega^2 + j\omega(2\zeta\omega_0) + \omega_0^2} \quad (2.25)$$

Identificando términos según la función transferencia (2.25), con las ecuaciones de variación de la densidad de fotones y densidad de electrones (2.23) (2.24), se obtiene:

$$2\zeta\omega_0 = g_oS_o + \frac{1}{\tau'_p} + \frac{1}{\tau_n} - \Gamma g_o(N_o - N_t) \quad (2.26)$$

$$\omega_0^2 = \frac{g_oS_o}{\tau'_p} + \frac{1}{\tau_n\tau'_p} - \frac{\Gamma g_o(N_o - N_t)(1 - \beta)}{\tau_n} \quad (2.27)$$

En estas ecuaciones ω_0 es la frecuencia propia del sistema y ζ el factor de amortiguamiento. En su obtención el valor de ε ha sido considerado despreciable.

A continuación se presentan las figuras para distintos tipos de láser, exactamente los mismos que fueron utilizados en el régimen estático. El código correspondiente en MATLAB a esta simulación se encuentra en el Anexo B. Este código se ejecuta posteriormente a un script denominado start que contiene los parámetros de uno u otro láser. En ambas figuras 2.6 y 2.7 se observa la respuesta en pequeña señal para diferentes corrientes, tomando valores por encima de la corriente umbral.

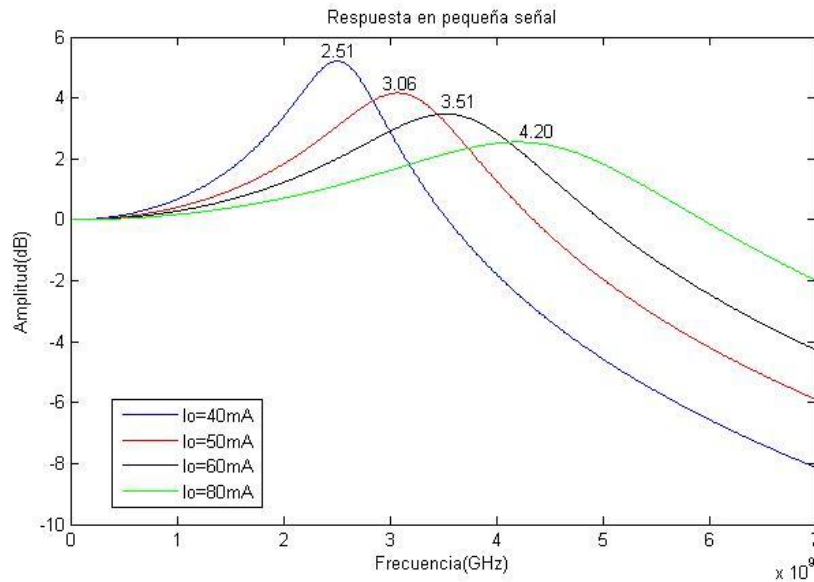


Figura 2.6. Respuesta en pequeña señal del dispositivo Ridge Waveguide Laser

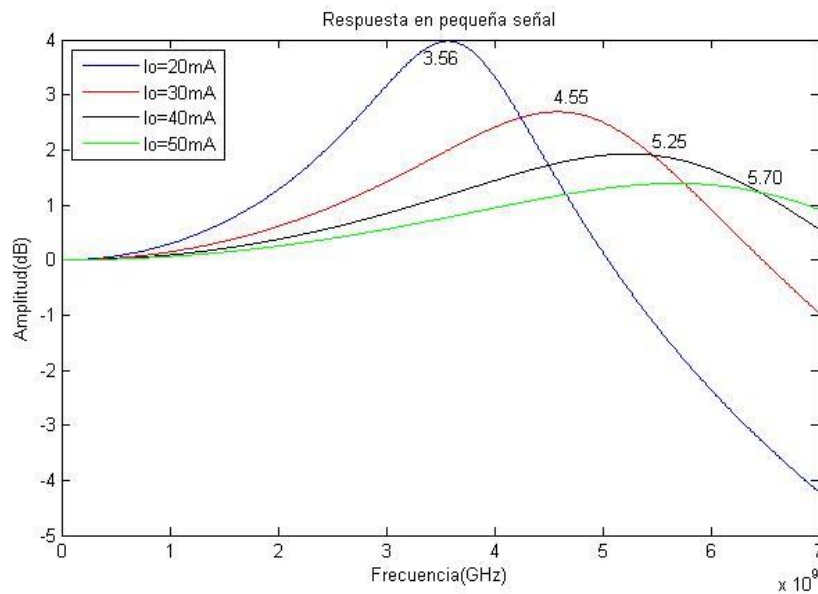


Figura 2.7. Respuesta en pequeña señal del dispositivo EMBH Laser

La ecuación 2.27 determina la frecuencia propia del sistema esta expresión puede ser aproximada por el primer término ya que los dos últimos son de un orden bastante inferior (2.28).

$$\omega_0^2 = \frac{g_0 S_0}{\tau_p'} \tag{2.28}$$

De esta forma se puede concluir que la frecuencia del sistema depende directamente del coeficiente lineal de la ganancia y de la densidad de fotones mientras que es inversamente proporcional del tiempo de vida de los fotones. Tal y como es observado en las figuras anteriores al aumentar la corriente de inyección la frecuencia propia también aumenta.

El valor que toma el factor de amortiguamiento como se puede observar en las figuras anteriores da lugar a sistemas subamortiguados. Por encima del valor de la frecuencia de resonancia el sistema cae con una pendiente aproximada de -40 dB/década. Además el factor de amortiguamiento actúa de forma diferente a la frecuencia de resonancia con respecto a la densidad de fotones, ya que a medida que la corriente aumenta el factor de amortiguamiento disminuye.

2.3.1.2.2 Respuesta transitoria

El análisis de la respuesta transitoria se determina como es la respuesta de un diodo láser al aplicar un nivel de corriente continua por encima del umbral hasta que los valores de las densidades de fotones y portadores alcanzan el nivel determinado para la corriente fijada. La respuesta transitoria de un dispositivo láser se puede aproximar por diferentes métodos numéricos de resolución de las ecuaciones de tasa. En este apartado se observa el método numérico Forward Euler y su posterior normalización.

- **Ecuaciones diferenciales ordinarias “Forward Euler”**

Dada una ecuación diferencial ordinaria:

$$\frac{dF}{dz} = gF \quad (2.29)$$

Para una función continua $F(z)$, que puede ser aproximada por $F(Zs)$, donde Z es un entero y s es el paso o avance en la función. El método de *forward Euler* determina la solución como (3):

$$F(Zs + s) \approx F(Zs) + s \frac{dF}{dz} \quad (2.30)$$

En el caso de las ecuaciones de tasa, la solución queda entonces formulada según (2.31) y (2.32).

$$N_0(t_0 + \Delta t) = N_0(t_0) + \frac{dN}{dt} \Delta t \quad (2.31)$$

$$S_0(t_0 + \Delta t) = S_0(t_0) + \frac{dS}{dt} \Delta t \quad (2.32)$$

La respuesta transitoria para los dos dispositivos láser anteriormente estudiados aparece en las figuras 2.8 y 2.9. Esta simulación ha sido obtenida a partir del código presente en el Anexo C. En ella se parte de un valor inicial de N_0 y S_0 igual a 0 en ambos y el valor de corriente final que se desea obtener será diferente en cada láser. Esto último se debe a que el umbral en cada láser toma distinto valor. Para cada láser se obtienen una grafica con respecto al tiempo, en la que se representa simultáneamente la densidad de fotones y la densidad de portadores.

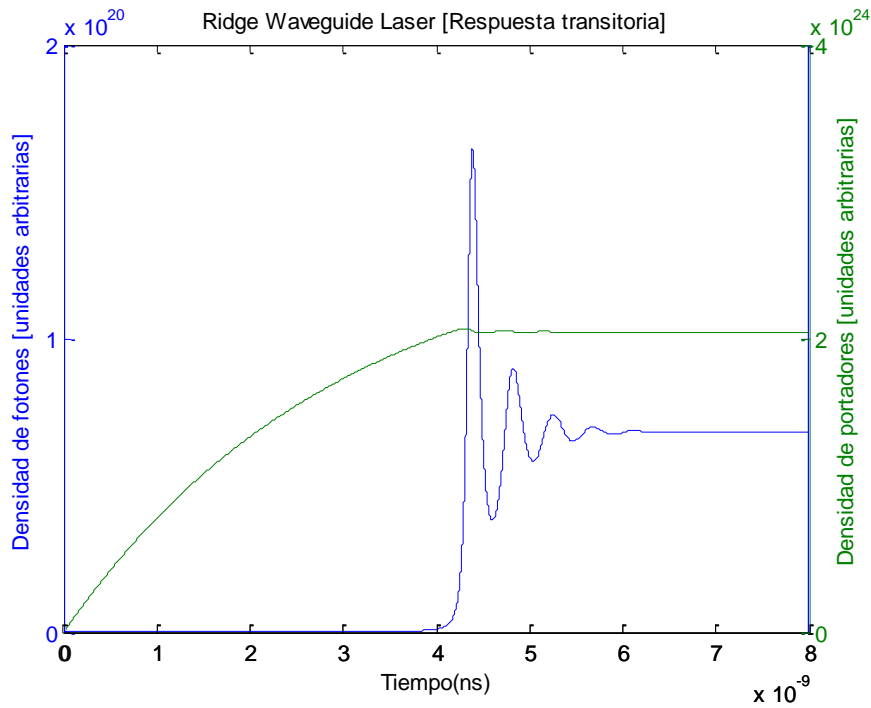


Figura 2.8. Respuesta transitoria de la densidad de fotones y densidad de portadores de un dispositivo Ridge Waveguide Laser, $I = 60 \text{ mA}$

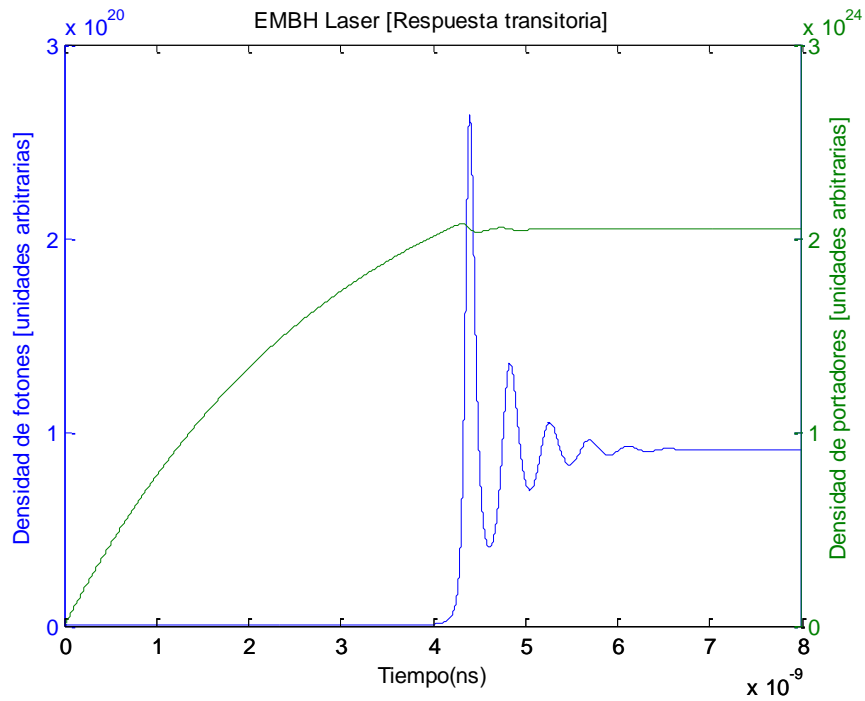


Figura 2.9. Respuesta transitoria de la densidad de fotones y de portadores en un dispositivo EMBH Laser con $I = 20 \text{ mA}$

En la respuesta transitoria se puede observar el tiempo de retardo, intervalo de tiempo mientras el cual la densidad de fotones se mantiene nula y la densidad de portadores aumenta. La razón de que hasta ese instante de tiempo no se observe densidad de fotones, se debe a que la

densidad de portadores no ha alcanzado el nivel umbral por lo tanto no se produce emisión estimulada.

Tras el tiempo de retardo aparecen las oscilaciones amortiguadas en las densidades de portadores y fotones hasta que se llega a régimen permanente.

Con la respuesta transitoria también se puede obtener la frecuencia propia del sistema y el factor de amortiguamiento según la Fig. 2.10 y las ecuaciones (2.33), (2.34), (2.35) y (2.36).

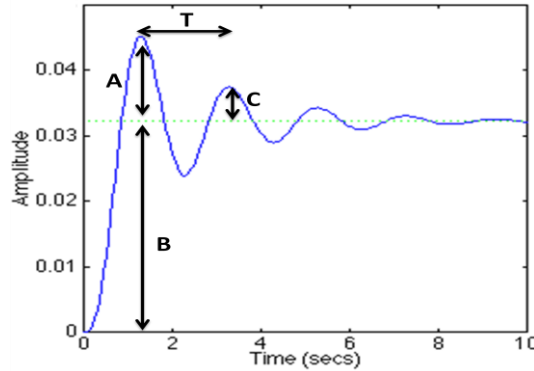


Figura 2.10. Respuesta de un sistema subamortiguado

$$\frac{A}{B} = e^{-\frac{\pi\zeta}{\sqrt{1-\zeta^2}}} \quad (2.33)$$

$$\frac{C}{A} = e^{-\frac{2\pi\zeta}{\sqrt{1-\zeta^2}}} \quad (2.34)$$

$$T = \frac{2\pi\tau}{\sqrt{1-\zeta^2}} \quad (2.35)$$

$$w_n = \frac{2\pi}{\tau}; w_n = 2\pi f_n \quad (2.36)$$

El valor obtenido de la frecuencia propia del sistema en el caso de la simulación en régimen transitorio es comparado con el obtenido en el caso de la respuesta en pequeña señal.

	Frecuencia [GHz] Respuesta transitoria	Frecuencia [GHz] Respuesta en pequeña señal
Ridge Waveguide Láser	2.36	3.51
EMBH	2.40	3.56

Tabla 2.2 Frecuencia propia del sistema

Se observa que las frecuencias obtenidas mediante el análisis del régimen transitorio son menores que las obtenidas en la respuesta en pequeña señal. Una razón que permite explicar esta diferencia es la inexactitud del método 'forward Euler', esto se debe a que únicamente tiene en cuenta el paso temporal siguiente para el cálculo de la aproximación de $F(Zs)$. Por tanto para encontrar los parámetros del láser que determinan la frecuencia propia y el factor de

amortiguamiento es apropiado utilizar la respuesta en pequeña señal. En cuanto a los métodos numéricos, estos pueden ser mejorados mediante diferentes técnicas. Por una parte, la normalización del sistema de ecuaciones, y por otra, cambiando el método numérico utilizado para su resolución.

- **Normalización de las ecuaciones de tasa**

A continuación se realiza una normalización de las variables del sistema, vista la diferencia entre las magnitudes del problema – tiempo en nanosegundos, densidad de portadores en 10^{24} , densidad de fotones en 10^{20} . El objetivo de la normalización es disminuir la carga computacional en el programa. Las variables que se definen para la normalización se presentan en las siguientes ecuaciones.

$$n = \frac{N}{N_t} \quad (2.37)$$

$$I_t = \frac{qVN_t}{\tau_n} \quad (2.38)$$

$$\tau = \frac{t}{\tau_n} \quad (2.39)$$

$$s = \tau_n g_o S \quad (2.40)$$

$$e = \frac{\epsilon}{\tau_n g_o} \quad (2.41)$$

$$k = N_t a \tau_p \quad (2.42)$$

$$t' = \frac{\tau_p}{\tau_n} \quad (2.43)$$

Con las variables definidas, se obtienen las siguientes ecuaciones de tasa normalizadas:

$$\frac{ds}{d\tau} = \frac{\Gamma k(n-1)(1-es)s - s + \Gamma \beta kn}{t'} \quad (2.44)$$

$$\frac{dn}{d\tau} = \frac{I}{I_t} - n - (n-1)(1-es)s \quad (2.45)$$

Esta normalización ya permite identificar un hecho característico de este sistema de ecuaciones, fundamentalmente que ambas ecuaciones operan en diferentes escalas de tiempo, siendo la evolución de la ecuación de portadores mucho más lenta que la de los fotones. Esta es la razón de que aparezca el parámetro t' . Con este parámetro ambas ecuaciones diferenciales se mantienen dependientes del valor $d\tau$. El código Matlab correspondiente a ese método se encuentra en el Anexo D.

En la simulación observada de un dispositivo Ridge Waveguide Laser para ecuaciones normalizadas (Fig. 2.11) se puede observar que la forma de la respuesta se mantiene, pero los ejes se encuentran normalizados.

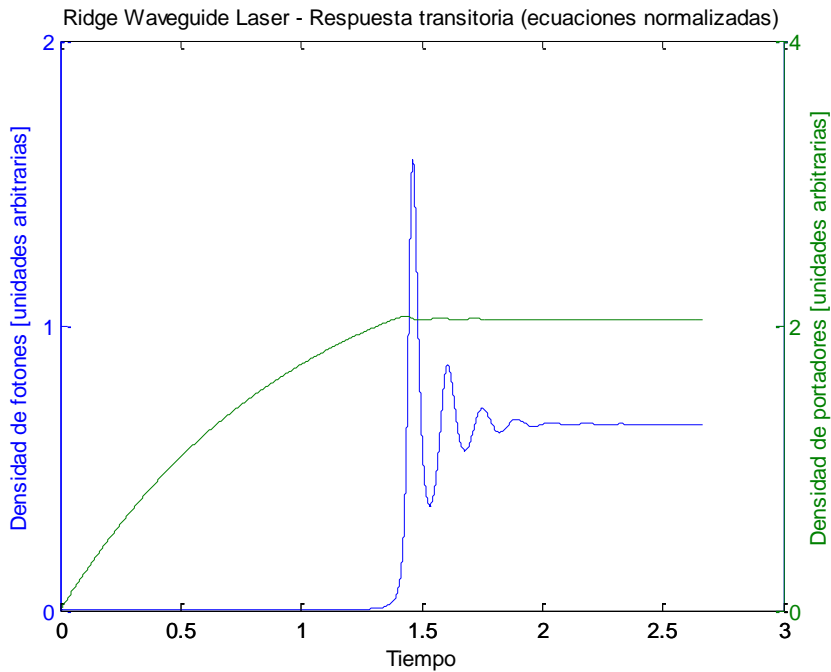


Figura 2.11. Respuesta transitoria con ecuaciones normalizadas de un dispositivo Ridge Waveguide Laser, $I = 60 \text{ mA}$

2.3.2 Modelado de dispositivos mediante software comercial.

En los apartados anteriores se han presentado los resultados que ofrecen diferentes tipos de algoritmos numéricos para analizar las ecuaciones características de un láser de semiconductor. Sin embargo, existen programas de simulación comerciales que facilitan esta tarea. En este apartado se presentan los resultados obtenidos utilizando el programa, TransmissionMaker de VPI Photonics. Este software permite realizar la simulación de circuitos ópticos. Por ello, para determinar la corriente umbral se utilizara el circuito presente en la figura 2.12 (8).

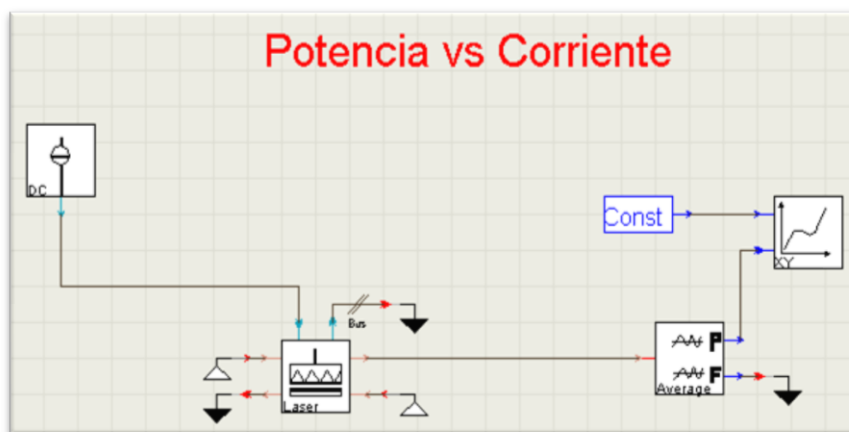


Figura 2.12. Circuito para medir la curva P-I en VPIPhotonics

Para obtener la curva P-I es desarrollado un circuito sencillo mediante el software indicado mediante una sección láser y un visualizador XY. A la sección láser se la introduce un barrido en corriente observando la figura 2.13. Esta simulación se realiza con dos iteraciones, apareciendo dos curvas P-I con una corriente umbral en torno a 50-60 mA.

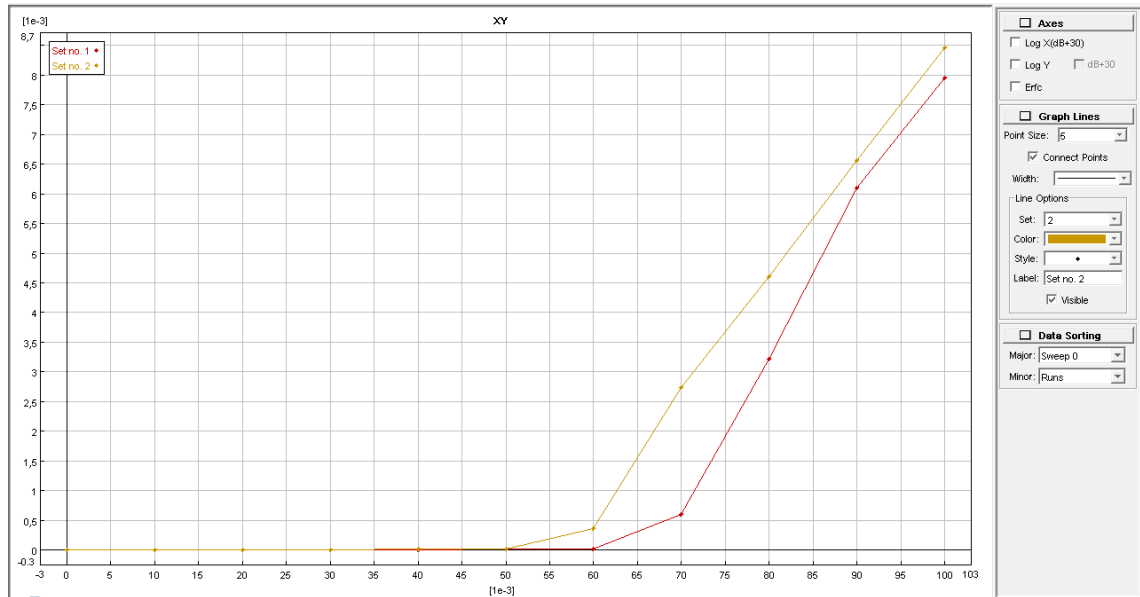


Figura 2.13. Curva P-I inicial

La corriente umbral (2.15) depende de tres parámetros α , N_o y τ_n . Estos parámetros a su vez dependen de otros que pueden ser controlados mediante VPI, con el fin de analizar en qué grado afectan a la corriente umbral.

Si se analiza el parámetro α , $\alpha = Vq$ donde q es la carga del electrón y V volumen. El parámetro modificable en este caso es el volumen, para este se tienen que obtener los términos correspondientes en VPI. Los términos son *DeviceSectionLength* y *ActiveRegionWidth*. Si el valor del segundo es reducido a la mitad, el valor de la corriente umbral también se ve disminuido hasta 10-20 mA (Fig. 2.14).

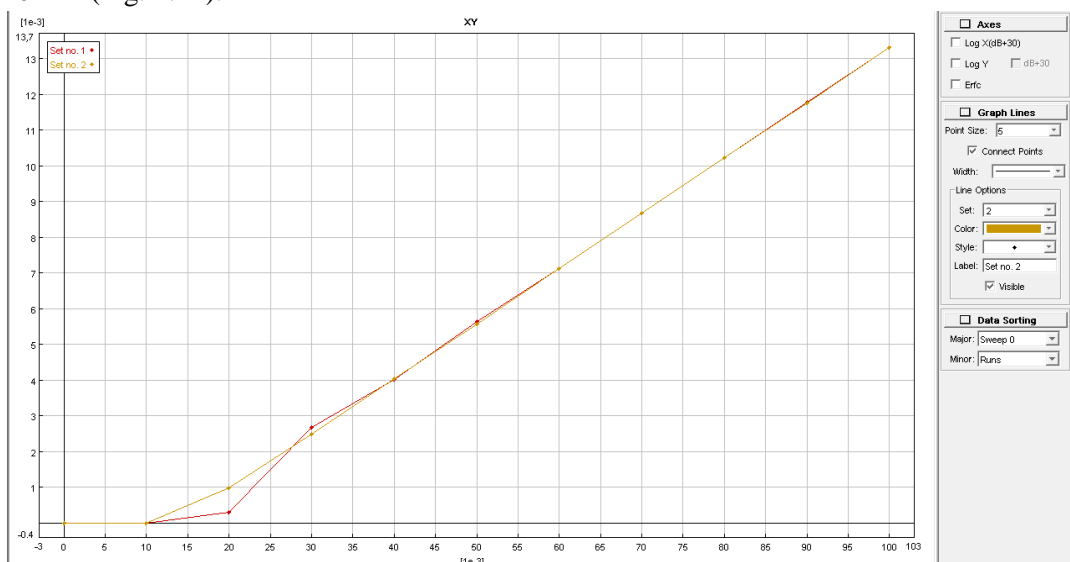


Figura 2.14. Curva P-I parámetro α modificado.

El valor de N_0 se puede descomponer en $N_t + \frac{1}{g_0 \Gamma \tau_{p'}}$, de estos parámetros mediante el programa VPI se varía la densidad de portadores de transparencia *CarrierDensityTransparency*. El valor de la densidad de portadores de transparencia en la curva P-I inicial corresponde a $1.5 \cdot 10^{-24}$ el cual se reduce a 10^{-24} valor que presentaban los láseres simulados en Matlab. La figura obtenida 2.15 presenta una corriente umbral de 20 mA tras la modificación de la densidad de portadores de transparencia, este valor es semejante al que era obtenido en Matlab.

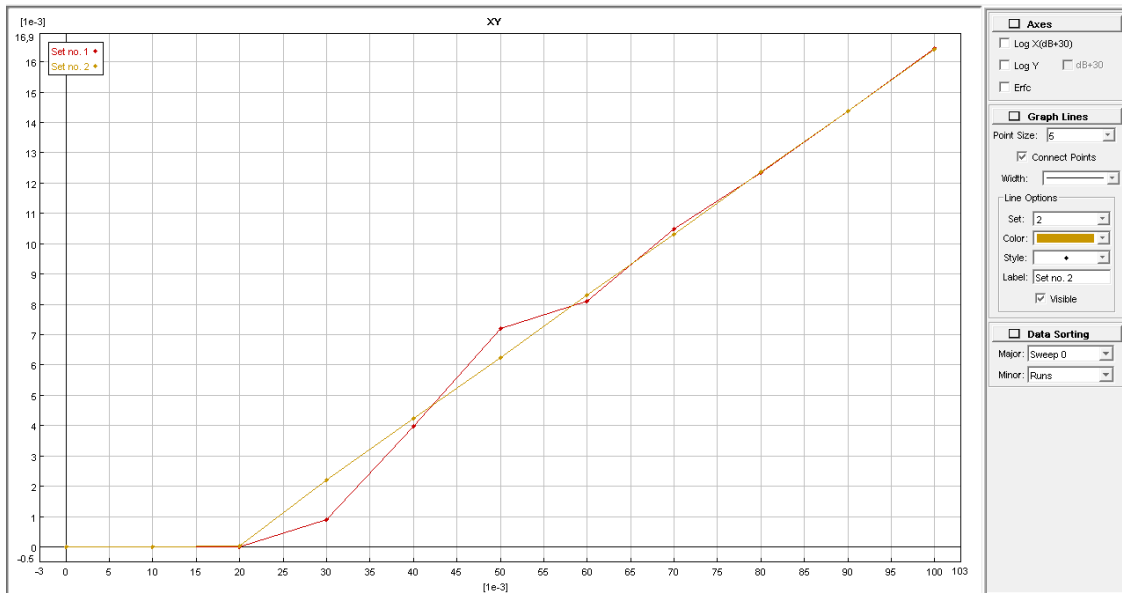


Figura 2.15. Curva P-I parámetro N_0 modificado.

Para variar el valor de N_0 también es posible en VPI variar el parámetro g_0 correspondiente a *GainCoefficientLinear* * c donde $c = 3 \cdot 10^8$ [m/s]. Para realizar esta prueba correctamente se restablece el valor de N_t y posteriormente se disminuye el valor de g_0 comprobando en la figura 2.16 que la corriente umbral aumenta.

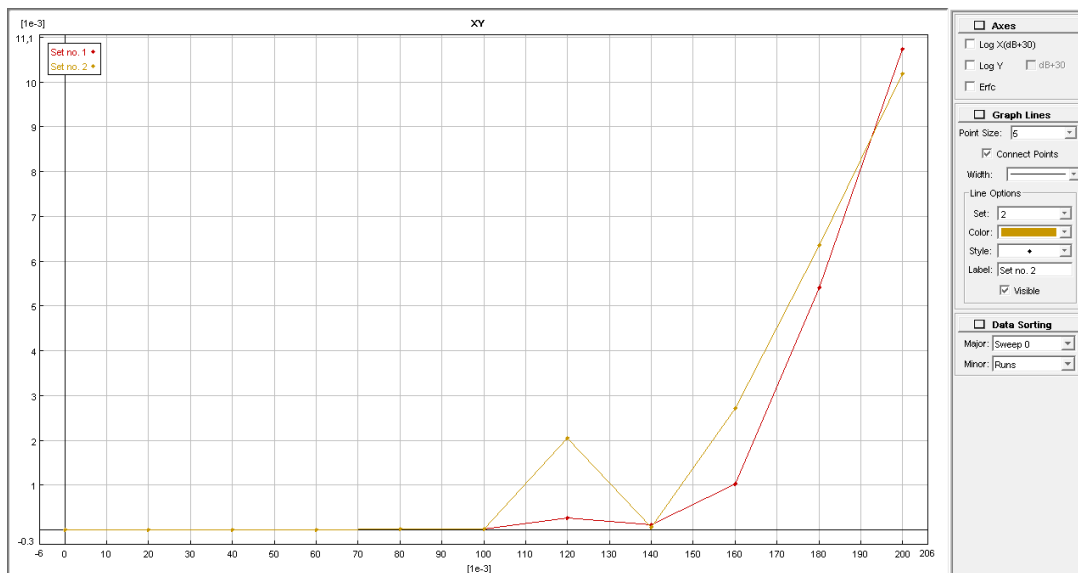


Figura 2.16. Curva P-I parámetro g_0 modificado.

τ_p' es otro factor del que depende la densidad de portadores, y τ_p' depende de las pérdidas internas por absorción de la cavidad. Este parámetro es denominado *InternalLoss*, y para observar su efecto sobre la corriente umbral es duplicado, obteniendo un aumento considerable de la corriente umbral.

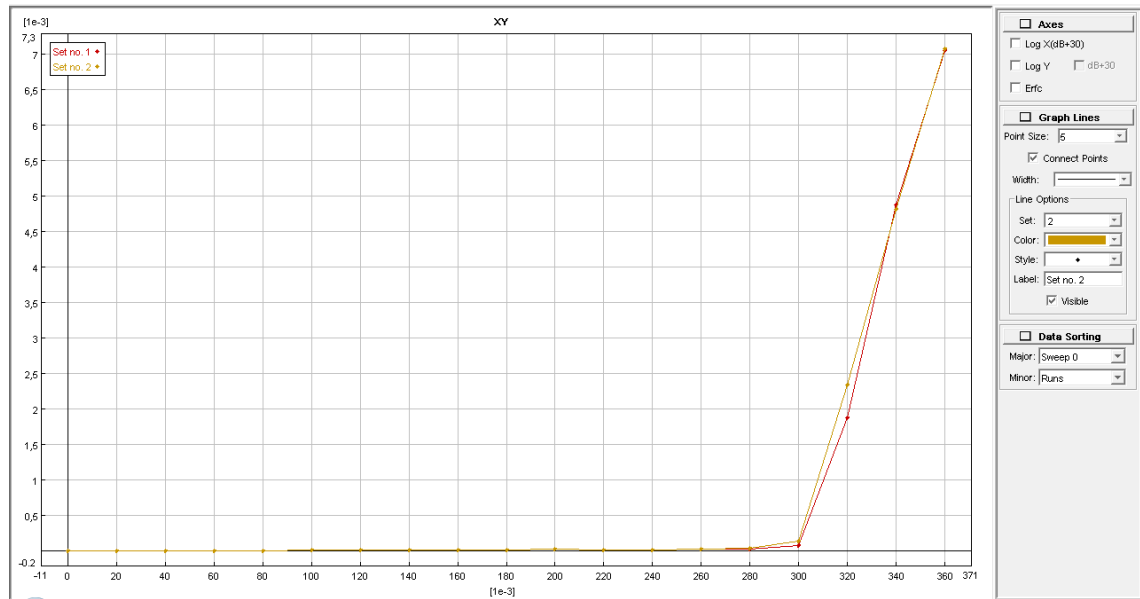


Figura 2.17. Curva P-I parámetro τ_p' modificado (*InternalLoss*).

El término τ_p' también puede ser modificado por otro factor, la reflectividad de los espejos del láser. El tiempo de vida de los fotones se ve aumentado al aumentar la reflectividad, y por tanto la corriente umbral disminuye. En la siguiente simulación, el parámetro en VPI *FacetReflectivity* se modifica de 0.32 a 0.5 obteniendo la figura 2.18.

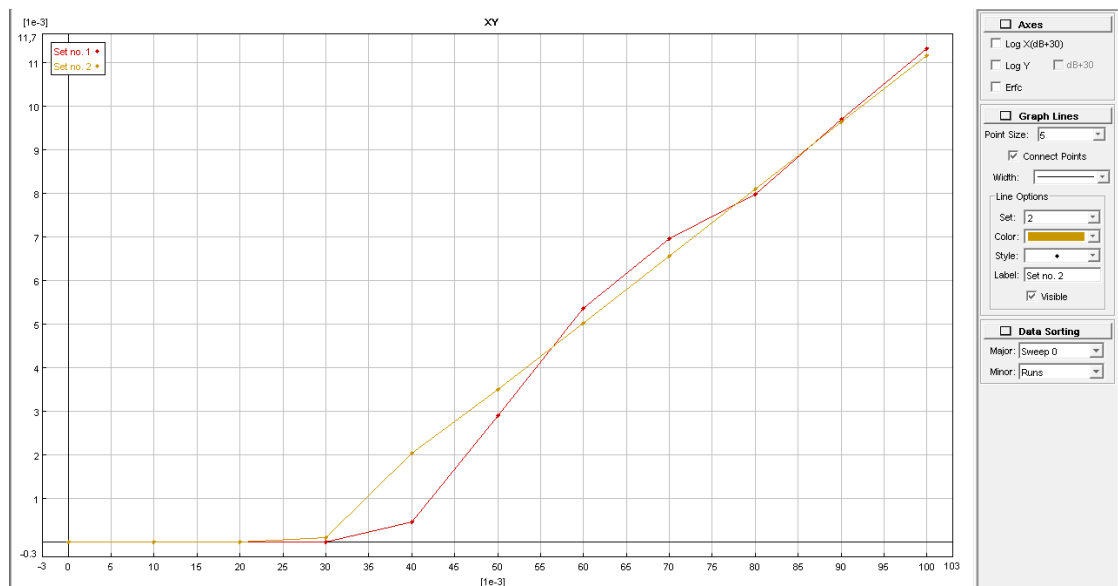


Figura 2.18. Curva P-I parámetro τ_p' modificado (*FacetReflectivity*).

El término τ_n es el último parámetro del que depende la corriente umbral, este parámetro en VPI es controlado por tres parámetros *LinearRecombination*, *BimolecularRecombination* y *AugerRecombination* (2.46).

$$\frac{1}{\tau_n} = A + B \cdot N + C \cdot N^2 \quad (2.46)$$

Aumentando el valor de cualquiera de estos parámetros, el valor de τ_n se ve disminuido y por tanto la corriente umbral aumenta.

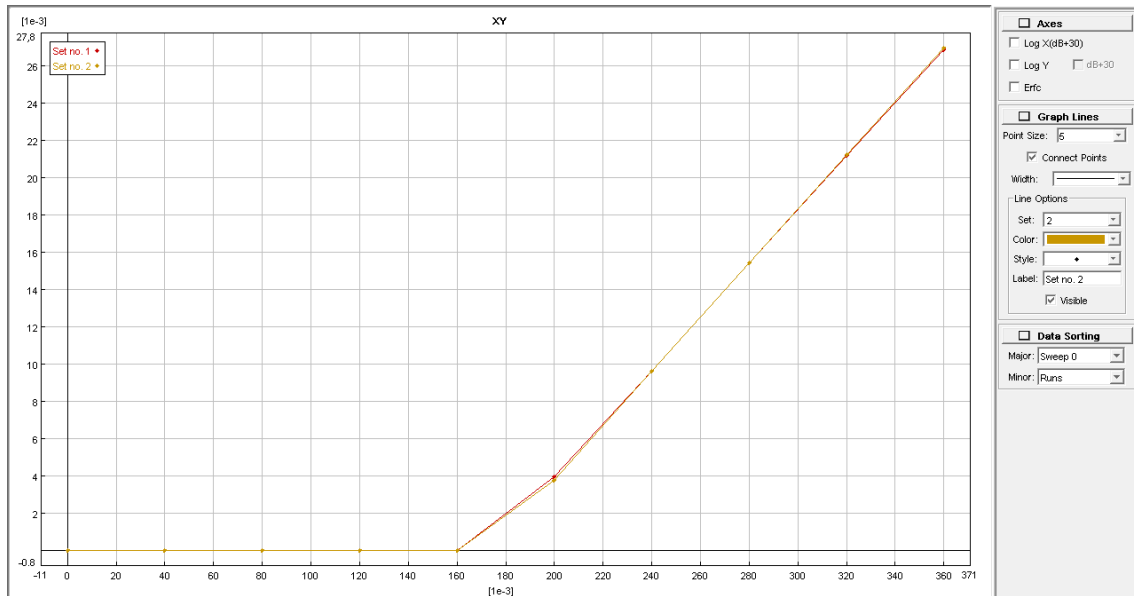


Figura 2.19. Curva P-I parámetro τ_n modificado (*BimolecularRecombination*)

2.4 Láseres *Mode Locking*

Los pulsos de alta tasa de repetición son necesarios para un gran número de aplicaciones que han ido apareciendo a lo largo de años. De entre las diferentes técnicas que han sido desarrolladas para pulsar la luz láser, la técnica de enganchar modos en fase (a la que en adelante nos referimos mediante su denominación en inglés, *Mode-Locking*) es la base de estudio en el presente proyecto, técnica con la que se consiguen los pulsos más cortos y el nivel de ruido más bajo.

Cuando las fuentes para la generación de pulsos son láseres de semiconductor mode-locked, a las ventajas anteriormente citadas podemos añadir su pequeño tamaño la posibilidad de integrarlos con otros elementos de semiconductor (9).

El régimen mode-locking se caracteriza por presentar una relación de fase constante entre los diferentes modos de la emisión láser (9). Estos dispositivos han demostrado ser capaces de alcanzar tasas de repetición de hasta 860 GHz, siendo habitual encontrar tasas de repetición de

hasta 40 GHz, ya que el coste y el consumo de energía aumentan con la frecuencia (10). Estas tasas de repetición moderadas permiten además sincronizar el tren de pulsos a una señal eléctrica, siendo habitual encontrar canales operando a tasas de repetición de 10 GHz (11).

La principal aplicación que han encontrado estos dispositivos son las redes de comunicación ópticas basadas en multiplexación óptica por división en el tiempo (OTDM). Frente a la técnica de multiplexación por división en longitud de onda (WDM), esta permite aumentar el número de canales sin multiplicar los costes en la tecnología ni el número de componentes, y por tanto su correspondiente coste de gestión (1). Las aplicaciones OTDM requieren sin embargo disponer de una fuente de luz estable, que genere pulsos ópticos con bajo jitter (9; 12).

Recientemente han ido apareciendo otros campos de aplicación de los láseres mode-locked como son por ejemplo el procesado óptico de señal y la conversión analógica digital (1).

Actualmente existen diferentes técnicas para enganchar los modos de un láser. En concreto la técnica de interés para el presente proyecto es el mode-locking pasivo (*Passive Mode-Locking* - PML). Esta técnica presenta como principal ventaja que no requiere de una señal de RF para operar, siendo suficiente alimentar el dispositivo con una corriente y una tensión constantes (4), (13). Sin embargo, dependiendo del nivel de corriente y tensión aplicados al dispositivo, este presenta diferentes regímenes de funcionamiento. Además del régimen de interés para este proyecto (mode-locking), es habitual encontrar otros dos regímenes: *Q-switching* y *Q-switched Mode Locking*(6).

A continuación se presenta una breve descripción de los procesos implicados en el régimen de mode-locking que permite enganchar los modos en fase mediante la técnica de mode locking pasivo. Posteriormente se describen las características de los otros dos regímenes que suelen también aparecer. En la figura 2.20 se puede observar un mapa ejemplo de los regímenes de funcionamiento de estos dispositivos.

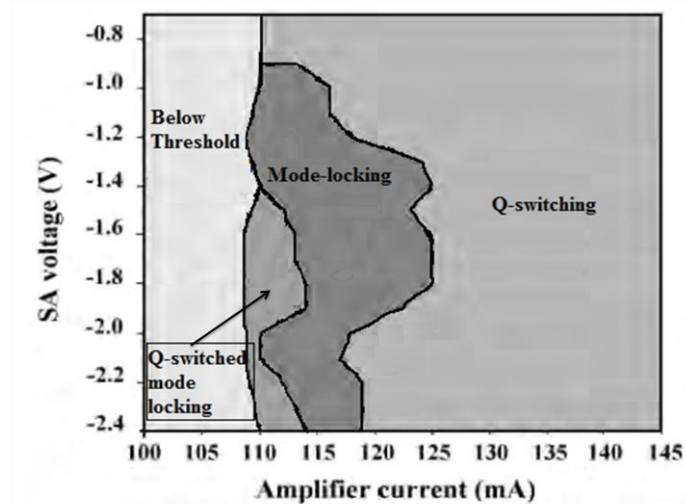


Figura 2.20. Regímenes de funcionamiento Mode-locking frente a corriente y tensión inversa(6).

Mode –locking Pasivo (PML)

El elemento clave para enganchar los modos en fase mediante mode-locking pasivo es el absorbente saturable. Por tanto, el dispositivo consta de una zona de ganancia, a la que se añade el absorbente saturable, el cual es básicamente una unión p-n como la región de ganancia del láser de semiconductor. Mientras que la región de ganancia se polariza con una corriente continua, el absorbente saturable se polariza con una tensión inversa también constante.

El desarrollo del mode-locking pasivo entre estas dos secciones de dispositivo se encuentra ilustrado en la figura 2.21. Es un requisito fundamental en el dispositivo que las pérdidas se saturen más rápido que la ganancia, de forma que exista un breve instante de tiempo en el cual la ganancia supere a las pérdidas, dando lugar a ganancia óptica neta en el sistema. Esto se expresa habitualmente de forma que la energía de saturación del absorbente debe ser menor que la de la sección de ganancia. Si además el tiempo de recuperación del absorbente es más rápido que el de la ganancia, entonces las pérdidas superan a la ganancia en todos los instantes de tiempo excepto en el que se localiza el máximo de potencia óptica del pulso (4).

En cada recorrido de ida y vuelta en la cavidad, la acción combinada de la saturación de ganancia y pérdidas influye en la forma del pulso óptico, amplificando el pico máximo del pulso y atenuando los extremos anterior y posterior. Ello produce un efecto neto de reducción de la anchura temporal del pulso que en régimen estacionario es compensada, por los mecanismos de ensanchamiento como la dispersión (2).

El mínimo ancho del pulso se encuentra limitado por el ancho de banda de la ganancia y los efectos no lineales (5).

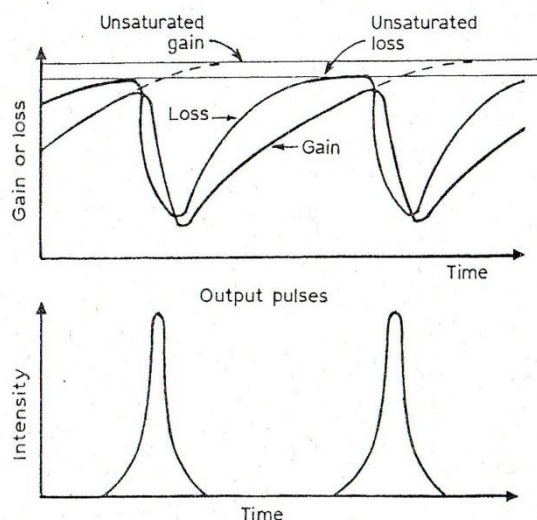


Figura 2.21. Láser Passive Mode Locking (4)

Como hemos comentado anteriormente y se observa en la figura 2.21, las oscilaciones mode-locked crean variaciones sustanciales en la ganancia y las pérdidas durante el instante de paso del pulso, por lo que es deseable que la tasa de repetición de los pulsos sea del orden del tiempo de recuperación de la ganancia o de las pérdidas. Además, y dado que los pulsos deben

auto-arrancarse, el valor máximo de ganancia debe ser superior al valor máximo de las pérdidas (4).

La diferencia principal del régimen PML frente a los siguientes es que la frecuencia de repetición de los pulsos está determinada por la longitud de la cavidad, que es un parámetro físico que podemos controlar.

Q-switching

Históricamente fue la primera técnica para generar pulsos ultracortos de luz, y se basa en poder variar el factor Q o factor de calidad del resonador mediante la introducción de un atenuador variable. Mientras el factor Q es bajo, las pérdidas son altas e impiden el fenómeno de amplificación estimulada de la luz y por tanto la emisión láser. Ello produce que se eleve la energía almacenada en la cavidad. En un instante dado, se conmuta el factor Q, resultando que la energía almacenada excede con mucho las pérdidas y, se libera en forma de un intenso pulso óptico (4; 14).

Si la conmutación se produce de forma externa a esta técnica se le denomina *active Q-switching*, y si es utilizado un absorbente saturable, *self-Q-switching*. Uno de los principales requisitos del absorbente saturable es tener una absorción lineal a la longitud de onda del láser. Por ello, son destacables los láseres de semiconductor ya que entre sus características cuentan con la capacidad de incorporar absorbentes saturables en el mismo material que se encuentra la zona activa del láser en contraste con otros tipos de láser que necesita de un absorbente saturable externo (4; 14).

Este régimen se evita porque dado que su origen es la emisión espontánea, y mediante procesos propios del material, no es posible controlar la tasa de repetición a la que se generan los pulsos.

Q-switched Mode Locking (QML)

Se trata de un régimen de funcionamiento que combina los dos anteriores, en el que la frecuencia de mode-locking, dada por el inverso del tiempo de propagación a lo largo de la cavidad, se modula con la frecuencia de oscilación Q-switching, debida a inestabilidades dinámicas (15).

Las desventajas de los láseres mode locking pasivos son su elevado jitter y su difícil sincronización con fuentes externas de reloj. Estos problemas han llevado al desarrollo de la técnica denominada *Hybrid Mode Locking* - HML, en la que se introduce una señal eléctrica de reloj en el absorbente saturable (13). De esta forma, la tasa de repetición que resulta, se encuentra sincronizada a la señal de RF inyectada. Esta técnica no debe ser confundida con la técnica *Gain-switching*, que en contraste con otras técnicas de generación de pulsos ultracortos no necesita ninguna sección especial en el dispositivo. Los pulsos son generados por la inyección de un gran número de portadores (electrones) en la región activa del dispositivo, permitiendo que la densidad de portadores sobrepase el umbral láser. La inyección de portadores se realiza por medio de un generador de RF. Cuando el umbral es superado se produce la emisión estimulada. Sin embargo

como la emisión estimulada se produce más rápido que la inyección de los portadores, la densidad de estos vuelve de nuevo a caer por debajo del umbral terminando la emisión óptica. Si la inyección de electrones no cesa el proceso se repite (4) (16).

Aunque ambas técnicas (Gain-switching y HML) introducen una señal de RF en el dispositivo su función es diferente. En el caso de Gain-switching la señal de RF determina los instantes en los que se inyecta un número de portadores pero cada ráfaga de estos no emite luz en el mismo periodo de tiempo. Por tanto, la señal de RF en este caso no determina la tasa de los pulsos ópticos a la salida del dispositivo. En cambio, en el caso la técnica HML los modos quedan enganchados en fase y la señal de RF introducida determina la tasa de los pulsos de salida.

2.5 Láseres *Mode Locking* en anillo

Existen distintas configuraciones posibles para un dispositivo mode-locking pasivo. Cabe destacar entre ellas la configuración en anillo, ya que presenta ventajas significativas respecto a los láseres Fabry-Perot. Mientras que en estos últimos la cavidad queda definida por los espejos que forman las caras clivadas del material semiconductor, en los láseres en anillo la cavidad es una guía de onda circular. Esto permite que un parámetro fundamental, la dimensión del resonador óptico, pueda ser controlado por fotolitografía, y con este la tasa de repetición. Dado que la tasa de repetición es inversa al tiempo de ida y de vuelta de un pulso en el láser, en el caso de la cavidad Fabry-Perot el tiempo sigue la ecuación $2 * L$ y en el anillo L , siendo L la longitud del resonador. Los láseres en anillo presentan un control muy superior en la tasa de repetición con respecto a los láseres lineales (6).

Otra importante ventaja de los láseres en anillo es que pueden ser integrados junto con otros componentes ópticos, formando un circuito integrado fotónico (*Photonic Integrated Circuit* – PIC). Esto se debe principalmente a que la estructura en anillo no encuentra las restricciones que impone la necesidad de crear los espejos mediante clivado del cristal. Debido a estas limitaciones de los láseres clivados surgió la necesidad de presentar nuevos métodos de realimentación óptica. Entre ellos está presente la utilización de un resonador circular, un láser en anillo (17).

Los láseres en anillo presentan una característica distintiva. En la cavidad, se propagan modos en dos direcciones opuestas. Por un lado, encontramos modos que viajan en sentido de las agujas del reloj (*Clock Wise- CW*) y modos que viajan en el sentido contrario a las agujas del reloj (*Counter Clock Wise- CCW*). Además, es habitual observar como ambos tipos de modos compiten entre sí. En el caso de un láser en anillo asimétrico dicha competición puede provocar en muchos casos situaciones de inestabilidad (6), dando lugar a biestabilidad unidireccional. En anillos en los que el resonador es una cavidad activa, se ha observado que puede controlarse la dirección de propagación de la luz con la inyección de corriente en los contactos. Este efecto lleva a la conmutación espacial, que puede ser de interés para la modulación óptica, tanto eléctrica como óptica pulsada. En la figura 2.22 se puede observar la salida obtenida de ambas direcciones CW y CCW y los modos que presentan (18).

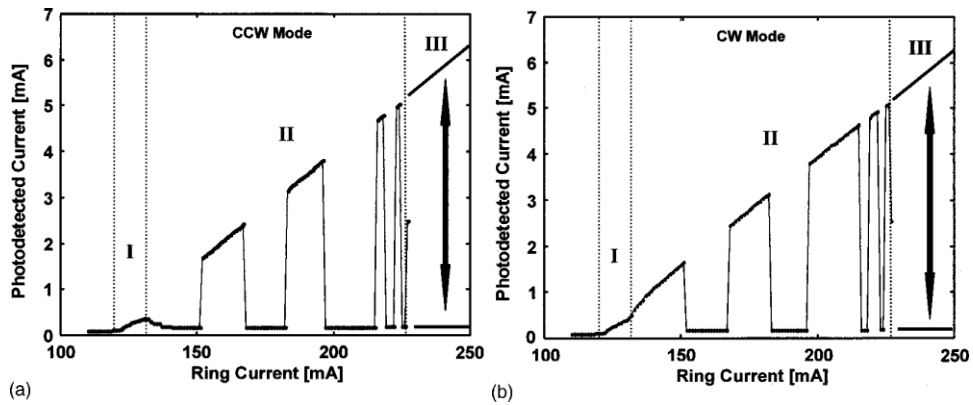


Figura 2.22. Corriente de salida vs corriente de entrada en un láser en anillo modo CCW (a) y modo CW (b). Tres regímenes de operación (I) bidireccional, (II) biestable e (III) inestable (18).

Como se observa en la figura 2.22. durante el régimen biestable no siempre domina un modo longitudinal sino que estos conmutan dependiendo de la corriente inyectada. Si se observa el espectro del modo CW para dos corrientes 160mA y 175mA, según Fig. 2.23 se debe tener el modo dominante para la segunda corriente. En el caso de 160mA el espectro observado será producto de la emisión espontánea Fig. 2.23 (18).

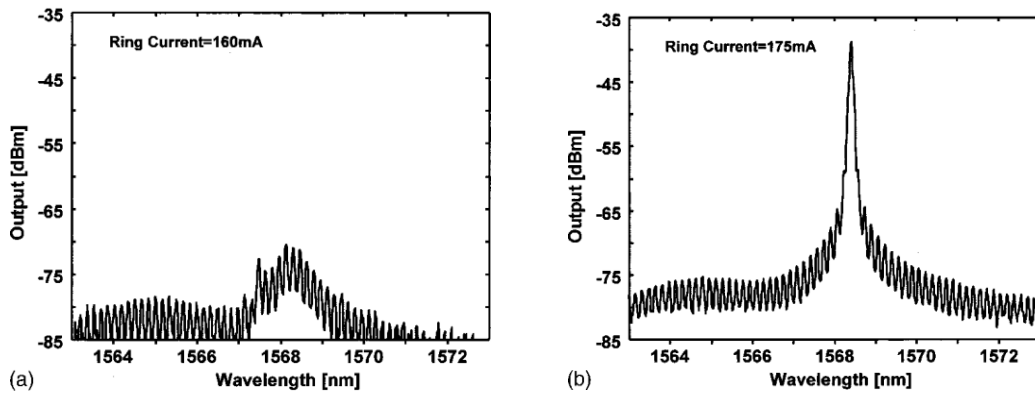


Figura 2.23. Espectro obtenido para el modo CW de un láser en anillo para la corriente de 160mA(a) y 175mA (b) (18).

Por el contrario, si el láser presenta operación bidireccional, la curva P-I se asemeja a la Fig. 2.24.

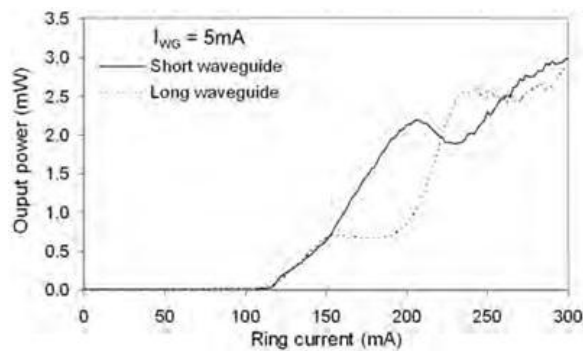


Figura 2.24. Salida de la potencia óptica de un láser en anillo por ambas guías (corta y larga) (6).

Aunque el láser en la Fig. 2.24 está funcionando como láser bidireccional, a partir de la inyección de 150mA se observa que la potencia se transfiere entre ambos modos (CW y CCW). En este caso el espectro es más uniforme y contiene un mayor número de modos con semejante amplitud y enganchados en fase, semejante al de un láser Fabry-Perot Fig. 2.25(6).

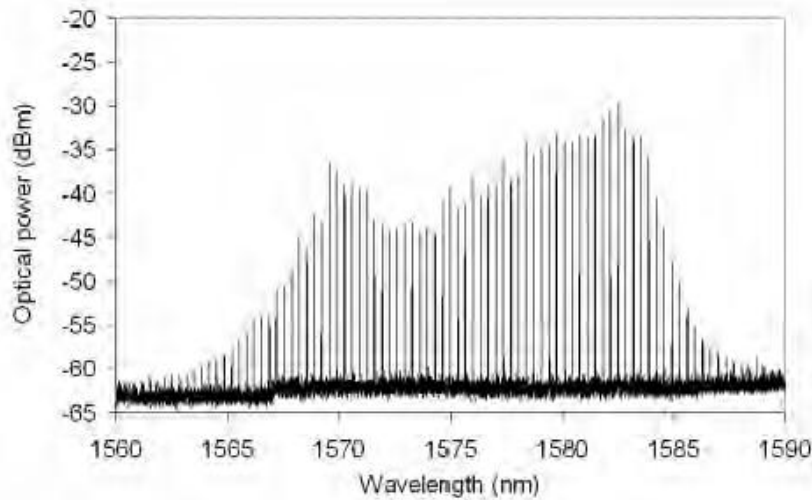


Figura 2.25. Espectro de un láser en anillo en régimen de operación bidireccional(6).

Sin embargo, se ha encontrado que las estructuras mode-locking en anillo operan en régimen *Colliding Pulse Mode – Locked (CPM)*, siendo los dos modos que se propagan en direcciones opuestas en la cavidad los que se encuentran en el absorbente saturable. De esta forma, la energía de cada pulso para saturar el absorbente se reduce a la mitad, produciendo además una reducción del ancho temporal del pulso, habiéndose llegado a la generación de pulsos estables inferiores a picosegundos en un láser (4) (13).

Como se ha indicado, en los láseres en anillo la cavidad es una guía óptica circular, y por tanto no hay pérdidas de luz asociadas a los espejos. Sin embargo, para obtener luz del dispositivo es necesario incluir en el resonador una estructura que permita extraer luz al exterior. Este elemento, así como la posición respecto el resto de elementos en la cavidad influyen profundamente en las características de operación del láser (19) (5).

Cuando la disposición de los diferentes elementos (acoplador, ganancia, absorbente saturable) en la cavidad del láser mode –locking en anillo es tal que los dos pulsos que se contra-propagan encuentran el mismo camino óptico (aunque en dirección opuesta), la estructura se denomina simétrica. Si no es así, la cavidad se denomina asimétrica.

En la figura 2.26 se puede observar un conjunto de láseres en anillo simétricos, con diferentes secciones: dos secciones amplificadoras, un absorbente saturable y dos secciones aislantes. En el anillo hay dos direcciones de propagación de los pulsos los cuales se encuentran en el absorbente saturable(6).

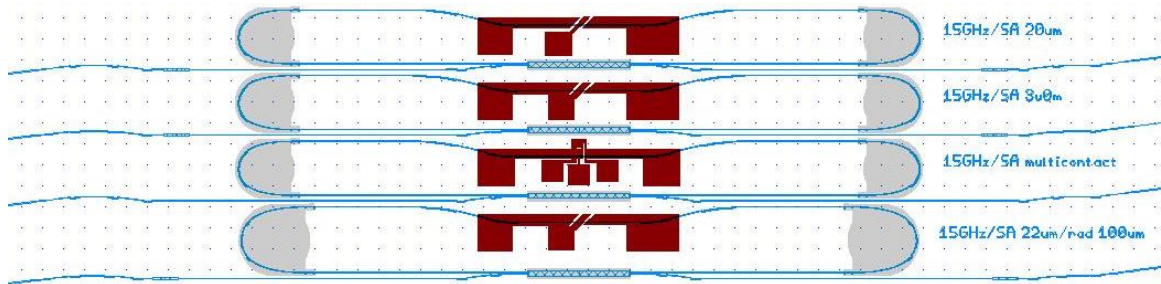


Figura 2.26 Láseres en anillo simétricos.

Si observamos el diseño de un láser en anillo asimétrico (Fig. 2.27), tanto el acoplador interferencial para extraer luz, como los absorbentes saturables no se encuentran en posiciones simétricas. Además, no disponemos de uno sino de dos absorbentes saturables en la cavidad. Idealmente, la disposición asimétrica puede favorecer una de las direcciones de propagación en la cavidad, haciendo que esta opere en régimen unidireccional, manteniéndose una diferencia en la ganancia de los dos pulsos(6).

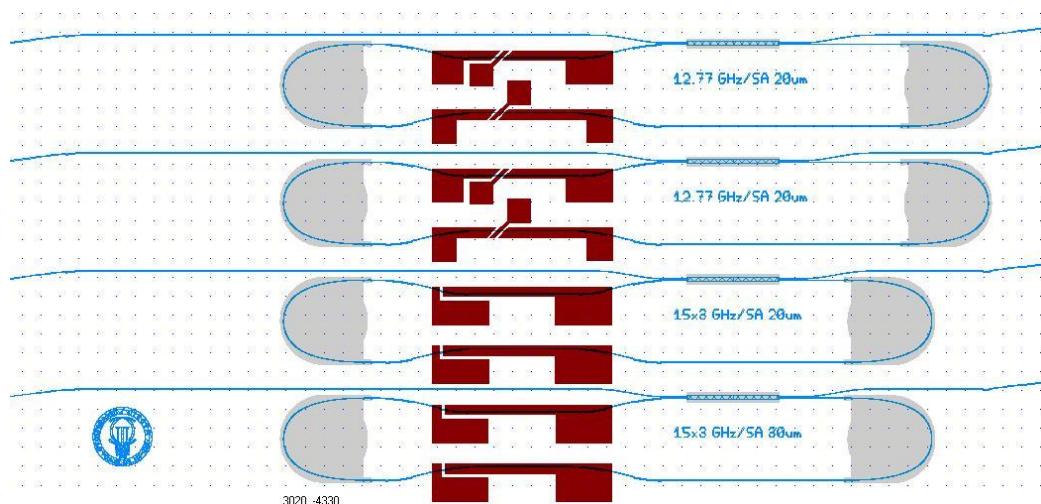


Figura 2.27. Láseres en anillo asimétricos.

Los láseres en anillo pueden estar en diferentes regímenes de funcionamiento, dependiendo de la corriente inyectada y del tiempo de vida de los portadores del absorbente saturable. En la región estable de mode-locking el ancho del pulso incrementa con la corriente y de la misma forma el ancho del pulso disminuye con el tiempo de vida de los portadores del absorbente saturable. Todo ello se cumple si no está cerca del régimen inestable (6).

En un láser simétrico existen diferentes modos de funcionamiento definidos. Y los pulsos incrementan su ancho al aumentar la corriente, y se ve disminuido al descender el tiempo de vida de los portadores del absorbente saturable. Dependiendo de este tiempo se define el modo de

Capítulo 2: Introducción a los láseres de semiconductor

operación del láser, manteniéndose en régimen unidireccional o con dos pulsos con diferentes intensidades y anchos(6).

Capítulo 3

Instrumentalización y desarrollo de interfaces

En este capítulo se presentan dos medidas de caracterización de un láser de semiconductor, la curva potencia óptica frente a corriente inyectada (curva P-I) y el análisis del ruido de fase. Para ambas medidas se describe el interfaz realizado que permite la automatización de la adquisición de datos en el laboratorio.

De la primera medida se obtiene un parámetro fundamental de estos dispositivos, la corriente umbral. Este valor puede ser determinado a partir de diferentes métodos de post-procesado de los datos, los cuales son también descritos en el presente capítulo.

3.1 Caracterización de la curva P-I

La curva P-I proporciona unos de los parámetros más importantes de los diodos láser. El primero es la corriente umbral, el cual representa la corriente a partir de la cual comienza la emisión estimulada. En segundo lugar, se encuentra la eficiencia cuántica, que se trata de la tasa de conversión de corriente de entrada en potencia de salida.

3.1.1 Interfaz Fuente de Corriente – Lector de Potencia.

En el entorno de desarrollo para este interfaz se necesita controlar dos instrumentos que intervienen en la medida, la fuente de corriente ILX Lightwave LDC 3724 y el lector de potencia Newport 842-PE. El interfaz obtenido se encuentra en la Fig.1.1.

Para el desarrollo del interfaz se utiliza la herramienta software de MATLAB. MATLAB ofrece la posibilidad de la creación de interfaces Gráficas de Usuario (GUI). Una vez desarrollado el interfaz, para su correcto funcionamiento se han de mantener en la misma carpeta dos archivos (.fig y .m). El archivo .fig define la forma de la interfaz y el archivo .m su funcionalidad. La

ejecución del interfaz se realiza nombrando en la línea de comandos de MATLAB el archivo .m como si se llamara a una función sin parámetros.

El espacio mostrado al usuario en la Fig. 1.1. se organiza en cinco regiones: conexión, temperatura, corriente, lector de potencia y curva P-I.

Previamente a la ejecución de cualquier función disponible en el interfaz se ejecuta la inicialización de ella, donde se genera el interfaz, se declaran los manejadores correspondientes y se dan valores iniciales a los campos dinámicos del interfaz. El extracto de código correspondiente a esta inicialización se encuentra en el Anexo E.

3.1.1.1 Panel de Conexión

El control de la fuente se realiza a través del bus de comunicaciones GPIB, a través del cual se le comunica a la fuente las operaciones a realizar mediante los comandos proporcionados por el fabricante. Para realizar la correcta conexión con la fuente se ha de conocer la dirección GPIB que esta tiene asignada. Esta dirección también es modificable en el propio instrumento. Si el sistema está formado por varios instrumentos interconectados por el bus GPIB, se han de respetar las direcciones ya asignadas. En el caso de la fuente ILX Lightwave LDC 3724 en este proyecto la fue asignada la dirección número 6. Este número aparece por defecto al inicializar el interfaz en el campo correspondiente a la dirección (Fig. 3.1). El usuario debe prestar atención y comprobar si este valor ha sido modificado para el correcto funcionamiento del interfaz, previamente a la inicialización.

Para realizar la comunicación con los instrumento a través de MATLAB se necesita la creación de objetos. Estos representan el enlace a través del que se pasa la información entre MATLAB y el driver correspondiente. Cada objeto debe estar asociado a un único instrumento.



Figura 3.1. Panel de Conexión (Interfaz 1).

Los objetos de la fuente de corriente y el lector de potencia, son diferentes. El PC mantiene la comunicación con la fuente de corriente a través de un objeto VISA. VISA es un estándar para la configuración de programación y comprobación de instrumentos. Entre los instrumentos para los que existe soporte VISA se encuentra GPIB. VISA en este caso proporciona el interfaz de comunicación entre el hardware y el entorno de desarrollo MATLAB.

En la creación de un objeto VISA se debe especificar el nombre del vendedor del driver ('agilent'), y el nombre del recurso VISA que controla el instrumento. El nombre del recurso se

compone del nombre del driver con interfaz VISA 'GPIB0' y la dirección del instrumento. Las sentencias específicas son:

```
stringdireccion = strcat('GPIB0::', get(handles.editedireccion, 'String'), '::INSTR');
fuente = visa('agilent', stringdireccion);
```

En el caso del lector de potencia la comunicación se realiza por puerto serie, aunque físicamente se encuentra conectado mediante USB. De igual forma, se precisa de un objeto para la comunicación con el instrumento. El interfaz de Agilent permite visualizar el puerto serie al que corresponde la conexión con el lector de potencia, especificando su nombre, la tasa de bits por segundo y el tamaño de los datos en bits.

```
lector = serial('COM3','BaudRate',115200,'DataBits',8);
```

Al pulsar el botón conectar se establece la comunicación con ambos instrumentos, comprobando previamente que el proceso no haya sido realizado antes.

Si se intenta realizar una operación en el interfaz, sin previamente haber realizado la conexión. Se muestra un mensaje de información (Fig. 3.2.) sobre la necesidad de realizar la conexión. De igual forma si se pretende conectar el dispositivo estando ya conectado, se presenta otro mensaje de información.

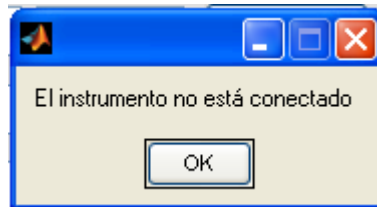


Figura 3.2. Mensaje de información.

El botón desconectar se encarga de cerrar los dos objetos (fuente y lector) al ser presionado.

El extracto de código correspondiente a este subpanel se encuentra en el Anexo F.

3.1.1.2 Panel de Temperatura

La fuente de corriente ILX Lightwave LDC 3724 permite la estabilización de la temperatura. Con este control se pueden realizar distintas medidas manteniendo la misma temperatura del peltier o por el contrario variar la temperatura observando su influencia en la potencia óptica obtenida a la salida de un dispositivo láser.

En este subpanel (Fig. 3.3.) se puede leer la temperatura actual en el peltier. También se le comunica a la fuente que fije una temperatura. Cuando se presiona el botón fijar únicamente se envía a través del bus GPIB la información de la temperatura que se desea, pero hasta que el usuario no presiona el botón Go! el control de temperatura no es activado.

El botón Go! es un botón biestado (*ToggleButton*) que se mantiene pulsado. Mientras este botón este pulsado el control de temperatura de la fuente estará activado. Permitiendo realizar una medida estable a una temperatura fijada.

El código correspondiente a este subpanel se encuentra en el Anexo G.



Figura 3.3. Panel de control de Temperatura (Interfaz 1).

3.1.1.3 Panel de Corriente

El siguiente subpanel (Fig. 3.4.) es de interés porque es importante mantener una corriente constante mientras se realiza el alineamiento de la fibra. Previamente a la elección de esa corriente, se realiza una curva P-I con la fibra acercada a la salida óptica del dispositivo. El objetivo de la realización de esta curva es conocer el valor de corriente para el cual se obtiene una salida de potencia óptica máxima. Posteriormente en este subpanel es fijado el valor de la corriente.

En el interfaz se comprueba que la salida de corriente de la fuente esta activada, antes de fijar el valor de corriente que se desea. Además, se debe tener en cuenta que si la corriente especificada es mayor que el límite de corriente fijado en la fuente no se ejecutara dicha función.

Como se indico en el capítulo 1, entre las características que presenta la fuente de corriente, esta cuenta con la posibilidad de limitar la corriente nominal de salida. Este valor debe ser fijado por el usuario. Para el valor del límite y por tanto de la corriente de salida existen dos límites impuestos por la fuente como máximo 200mA o 500mA. En el panel frontal de la fuente es visible mediante un LED saber cuál de ellos está activo.

El extracto de código correspondiente a este subpanel se encuentra en el Anexo H.



Figura 3.4. Panel de control de la Corriente (Interfaz 1).

3.1.1.4 Panel del Lector de Potencia

Este panel (Fig. 3.5.) permite leer la potencia para un valor fijado de corriente. En el únicamente se establece comunicación con el lector de potencia para solicitarle el valor leído. El valor devuelto por el lector de potencia Newport 842-PE debe ser tratado ya que es un String con valor de potencia leído y otros caracteres. El código de este panel se encuentra en el Anexo I.



Figura 3.5. Panel para la lectura de potencia (Interfaz 1).

3.1.1.5 Panel Curva P-I

El objetivo final de este interfaz es permitir la caracterización automatizada de la curva P-I característica de un diodo láser. Esta función es realizada a través del siguiente panel, aunque no son menos importantes los paneles explicados anteriormente ya que son necesarios para la comunicación con los instrumentos, estabilización de la temperatura y el proceso de alineamiento de la fibra.

En el panel de caracterización de la curva P-I (Fig. 3.7.), se encuentran tres campos dinámicos a través de los que se especifica la corriente máxima a la que debe llegar la curva P-I, el paso o salto en mA que marca el número de puntos de la gráfica, y el límite de corriente que se establece en la fuente de corriente. Si el límite de corriente no se especifica o es un valor inferior al asignado a la corriente máxima, el interfaz informa del error al usuario (Fig. 3.8.) al presionar

el botón pintar. De esta forma se evita la aplicación de valores elevados de corriente sobre el dispositivo láser.

Los tres botones que se muestran en este panel realizan las siguientes funciones:

Pintar: Al presionar este botón se desarrolla el diagrama de flujo presentado en la figura 3.6. Donde se comprueban una serie de especificaciones primero, antes de iniciar la toma de datos correspondientes a la curva P-I. Este último proceso requiere de un tiempo que depende del número de saltos de corriente especificado, además entre la asignación de una corriente y la lectura de potencia se especifica un tiempo de espera. El tiempo de espera es suficiente como para que el diodo láser pueda volver a estar en régimen permanente antes de la lectura de la potencia para la corriente fijada. El resultado final se representa gráficamente según la Fig. 3.9.

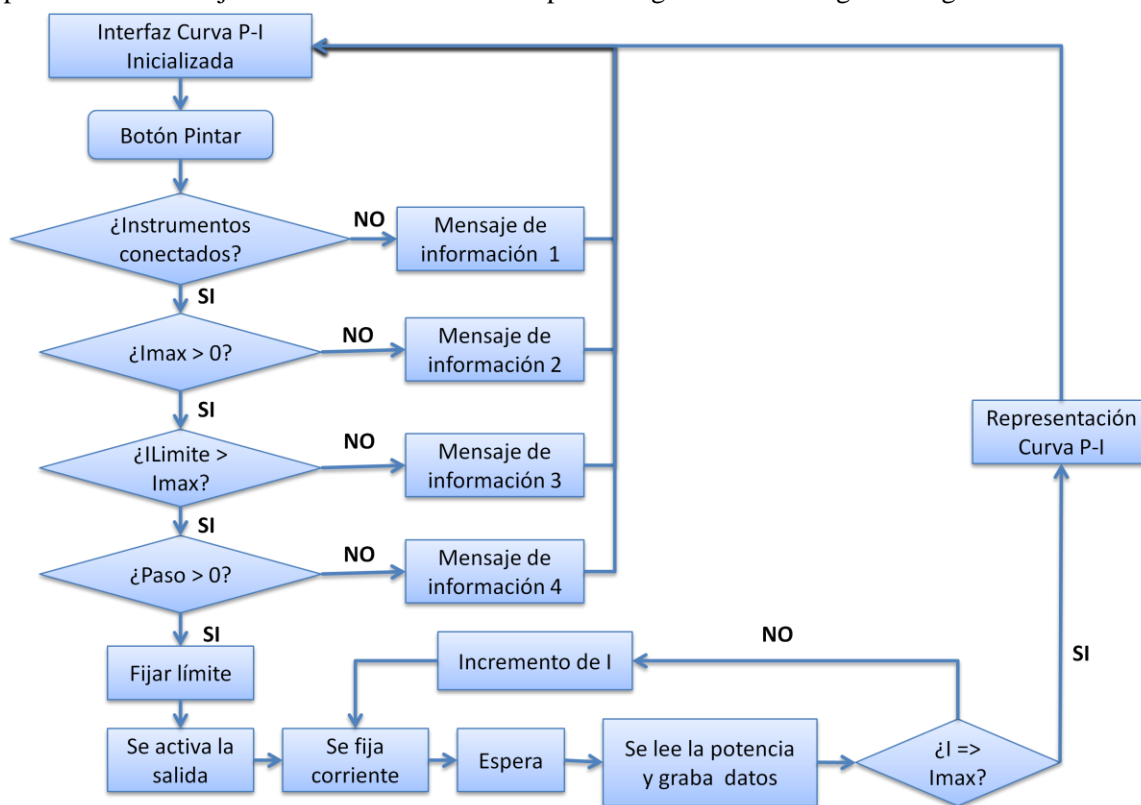


Figura 3.6. Diagrama de flujo del botón Pintar.

Limpiar: Elimina las trazas que hayan sido pintadas en la gráfica si no se desea que dos curvas se solapen.

Guardar Curva P-I: Al presionar este botón se abre una ventana de dialogo estándar (Fig. 3.10) que permite guardar los datos de corrientes y potencias adquiridos al pintar la curva P-I, con el nombre que el usuario especifique. Este fichero se guarda en .mat para posteriormente poder ser utilizado mediante MATLAB.

El código para este panel se encuentra en el anexo J.



Figura 3.7. Panel para la caracterización de la curva P-I (Interfaz 1).

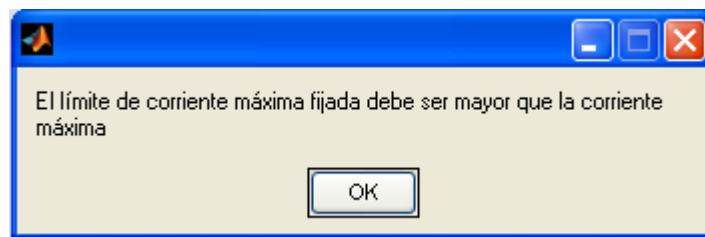


Figura 3.8. Mensaje de error por exceder el límite de corriente (Interfaz 1).

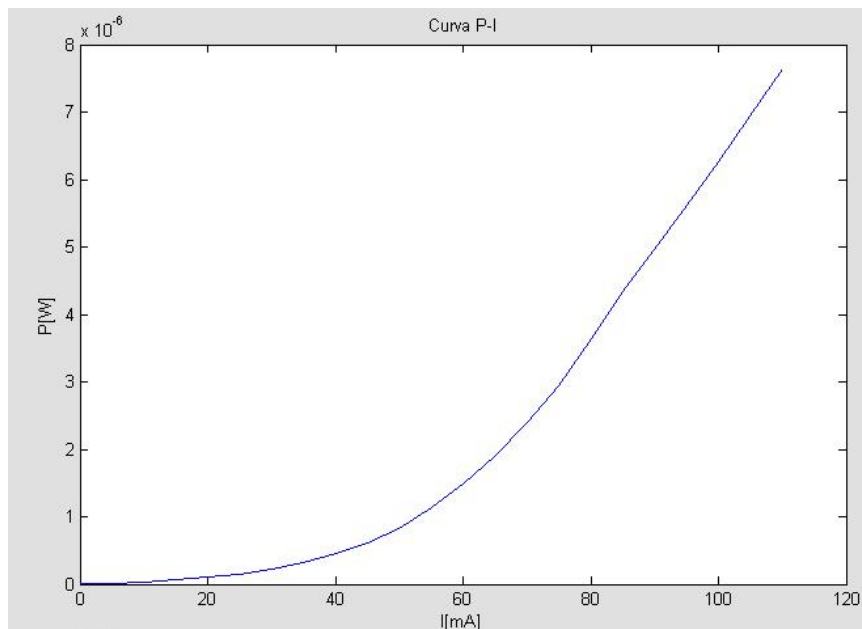


Figura 3.9. Representación grafica de la curva P-I (Interfaz 1).

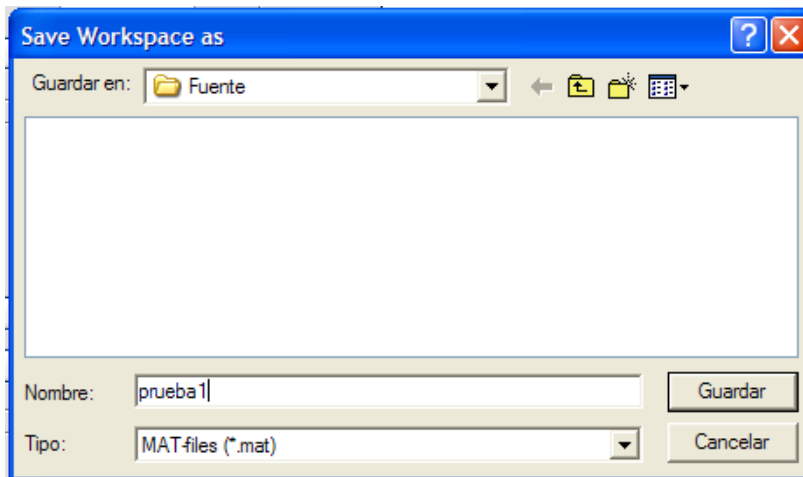


Figura 3.10. Ventana de dialogo para guardar la curva P-I (Interfaz 1).

3.1.1.6 Cierre ordenado

Cuando el usuario presiona el botón de cierre del interfaz, se crea un mensaje de dialogo (Fig 3.11.) permitiendo decidir al usuario si desea o no cerrar el interfaz. Si la elección es cerrarlo se realiza un cierre ordenado que consiste en desconectar los dos instrumentos utilizados en el interfaz y posteriormente eliminar los datos de ambos objetos. La eliminación de los datos es importante, porque si no se produce, no es posible volver a ejecutar el interfaz.

El código para el cierre ordenado de la conexión se encuentra en el Anexo K.

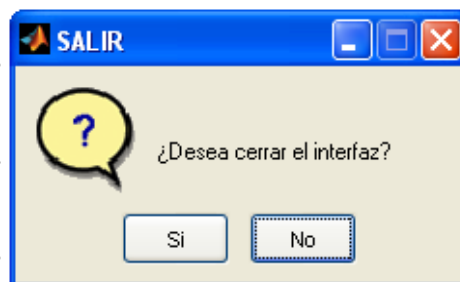


Figura 3.11. Ventana de dialogo para cerrar el interfaz (Interfaz 1).

3.1.2 Post-Procesado de los datos

Existen diferentes métodos matemáticos para obtener el valor de la corriente umbral: ajuste lineal, ajuste por dos segmentos, primera derivada y segunda derivada (20).

Ajuste Lineal

El ajuste lineal es el método más sencillo, simplemente se trata de extender una línea recta correspondiente a la región en la que el dispositivo produce emisión láser hasta cortar el eje horizontal. El punto de intersección con este eje es el umbral. El valor calculado para la corriente umbral en este caso presenta serias desventajas. En primer lugar este valor es altamente dependiente de la pendiente o eficiencia cuántica del láser. Esto puede ser causa de error dado que la eficiencia y por tanto la pendiente puede cambiar debido a las propiedades internas del láser, o incluso por el acoplamiento inadecuado de la fibra a la salida (20).

Por otra parte si esta línea es calculada como una regresión lineal, se debe tener en cuenta que los puntos lejanos al codo que se forma en la curva P-I y por tanto distantes de la corriente umbral, pueden influir fuertemente por las no linealidades. Si la línea es calculada mediante dos puntos la elección del primer punto es crítica. Si el primer punto está cerca del codo, la corriente umbral será desplazada y dará un valor inferior. Si por el contrario se considera un valor elevado, las no linealidades pueden afectar como se indico en el anterior caso (20).

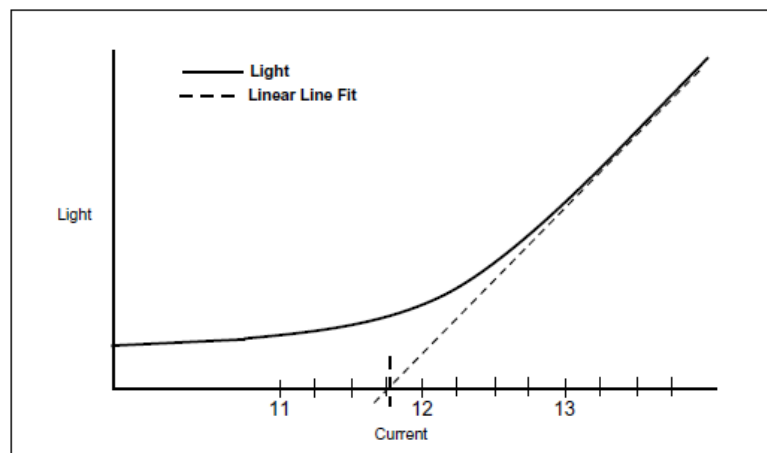


Figura 3.12. Cálculo de la corriente umbral mediante ajuste lineal (20).

Ajuste por dos segmentos

Divide la curva P-I en dos secciones, situando al codo de la corriente umbral en medio. Cada sección es aproximada por una línea recta (una es la parte pre-umbral de la curva P-I y otra es zona en la que el dispositivo produce emisión láser). El punto de intersección de ambas líneas se proyecta sobre el eje horizontal y se determina la corriente umbral (20).

Con respecto al anterior, este método es menos sensible a la variación de la eficiencia. Pero mantiene los errores debidos a las no linealidades a la hora de aproximar ambos segmentos (20).

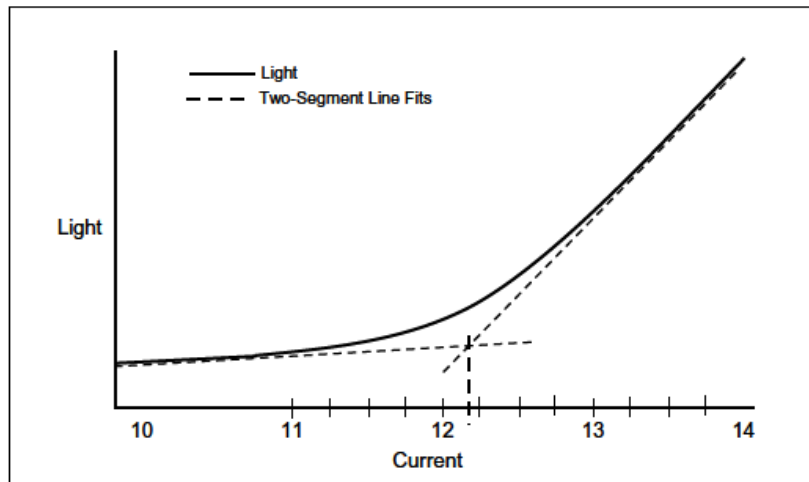


Figura 3.13. Cálculo de la corriente umbral mediante ajuste por dos segmentos (20).

Primera derivada dP/dI

Se realiza la primera derivada de la curva P-I. Y el límite umbral se considera la corriente para la que la primera derivada llega a la mitad de su valor máximo. El problema surge cuando el máximo de la primera derivada no es fácil de definir, este problema se ve agravado por el ruido de medición (20).

En este caso la repetitividad se ve dificultada, dado que si el máximo obtenido no se consigue tras otro barrido, el valor de la corriente umbral tampoco será el mismo (20).

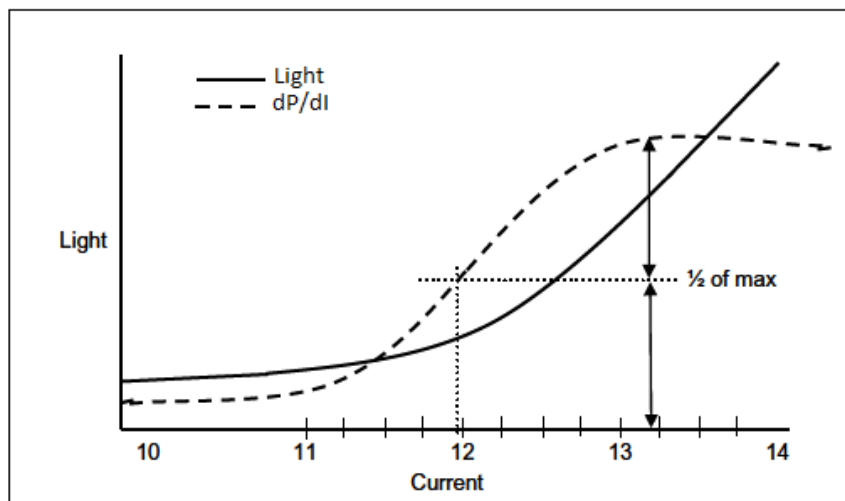


Figura 3.14. Cálculo de la corriente umbral mediante la primera derivada (20).

Segunda derivada dP^2/dI^2

El método de la segunda derivada localiza el punto de la tasa máxima de cambio de la curva P-I, que se trata del punto de inflexión de la primera derivada. Este punto no es necesariamente el obtenido con la primera derivada y es insensible a las no-linealidades anteriores y posteriores al codo umbral(18).

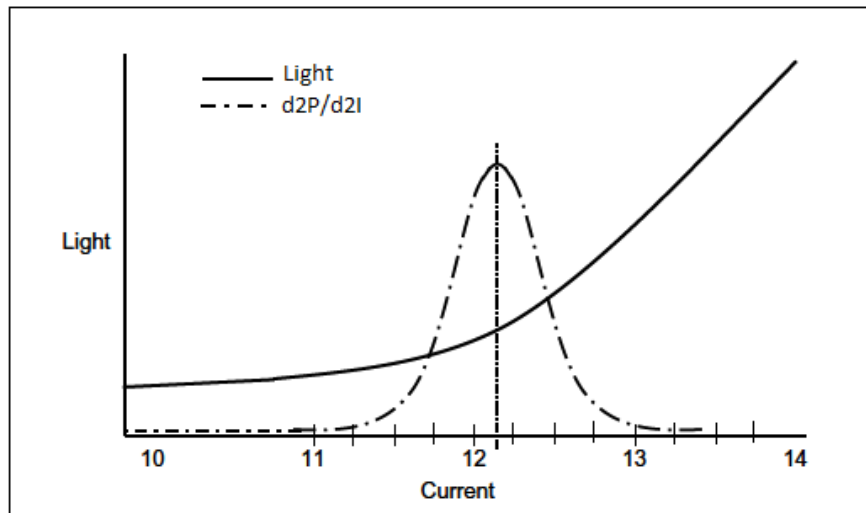


Figura 3.15. Cálculo de la corriente umbral mediante la segunda derivada (20).

3.2 Caracterización del ruido de fase

El análisis del ruido de fase se realiza mediante el analizador de espectros eléctrico (RFSA). Por ello, la señal óptica debe convertirse en una señal de RF mediante un fotodetector.

En el presente proyecto como fue mencionado en el capítulo 2 son de interés los láseres mode-locking. Para saber que el diodo láser presenta régimen mode-locking existen varias formas de comprobación. Una de ellas, es la observación de un tono frecuencial en el analizador de espectros eléctrico. Este tono aparece a la frecuencia correspondiente a la separación frecuencial de los modos que se pueden observar en un analizador de espectros óptico. La otra forma de identificar el régimen mode-locking es a través del autocorrelador, con presencia de un tren de pulsos a su salida. Este último aspecto no es desarrollado en este proyecto.

En el caso de mantener un diodo láser en régimen mode-locking en el analizador de espectros eléctrico se puede obtener información de la tasa de repetición según la frecuencia en la que quede centrado el tono. También se pueden medir las componentes de ruido de la portadora, que se componen del ruido en amplitud y del ruido de fase (12).

3.2.1 Tipos de ruido.

Como es indicado previamente el ruido se compone del ruido de amplitud y del ruido de fase (Fig. 3.16.). La variación de un nivel máximo del pulso desde el valor medio es considerado como el ruido en amplitud. La variación de la posición del pulso en el tiempo de su posición predeterminada es causada por el ruido de fase (12).

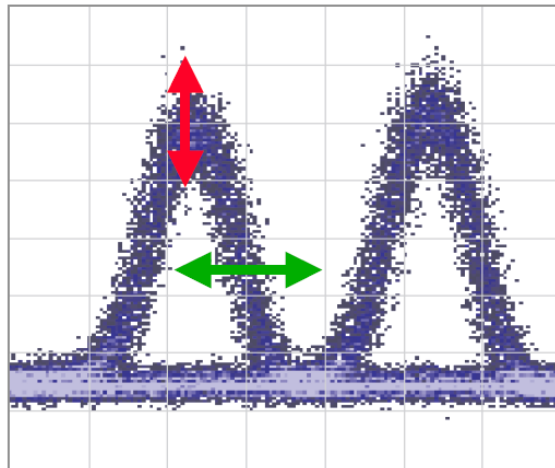


Figura 3.16. Ruido de amplitud (rojo) y ruido de fase (verde) en un pulso (12).

El ruido total se compone de fluctuaciones aleatorias y el ruido determinista. Las fluctuaciones aleatorias son causadas por la naturaleza cuántica de la emisión y por los procesos de emisión espontánea. El ruido determinista puede tener diferentes orígenes, entre ellos está el Q-switching residual (12).

Para resolver las contribuciones del ruido de fase se debe previamente evaluar su importancia según los sistemas de transmisión. En frecuencias bajas de hasta 1 MHz puede ser tolerado por el uso de circuitos de retroalimentación y la recuperación del reloj. Las frecuencias más altas son el ruido de fase más crítico para la operación de libres errores de los sistemas OTDM, porque los circuitos de recuperación de reloj en este caso no tienen ancho de banda para generar señales de sincronización de referencia. Un reajuste rápido de impulsos de control se vuelve difícil (12).

Medida del ruido de amplitud

El cálculo del ruido de amplitud partiendo del espectro eléctrico, es obtenido mediante la ecuación (3.1). Donde se realiza la raíz cuadrada de la integral del *RIN* (*Relative intensity noise*, la inestabilidad en el nivel de potencia de un láser). Los límites de la integral son f_1 y f_2 donde el límite inferior es la frecuencia más baja que permite el RFSa y el límite superior la mitad de la tasa de repetición del láser mode-locking (21).

$$N_{AM} = \sqrt{\int_{f_1}^{f_2} RIN df} \quad (3.1)$$

Medida del ruido de fase

Existen distintos métodos para el cálculo del ruido de fase. La siguiente explicación se centra en una de ellas, la medida de la banda lateral (single sideband, SSB) de la portadora observada en el espectro eléctrico. Esta es la utilizada en el interfaz de caracterización del ruido de fase desarrollada en este proyecto (12).

El nivel de ruido de la SSB se define como $L(f_m)$ y es el ratio de la potencia en la banda lateral P_{SSB} a una frecuencia de offset f_m entre la potencia P_o a la frecuencia de la portadora f_o . De esta forma, $L(f_m)$ resulta (3.2) y queda relacionado a 1Hz de ancho de banda (12).

$$L(f_m) = \frac{P_{SSB}(f_m)|_{1\text{HZ}}}{P_o} \tag{3.2}$$

La precisión de la medida del nivel de ruido de fase en la SSB, $L(f_m)$ se ve limitado por el ruido de fase intrínseco del analizador de espectros eléctrico (RFSa). El fabricante del RFSa debe proporcionar por ello en las especificaciones del dispositivo los valores de $L(f_m)$, que representan el nivel de ruido mínimo que debe medir el dispositivo. En la figura 3.17 (derecha) la línea negra punteada se corresponde con la especificación del fabricante para un determinado analizador de espectros eléctrico (12).

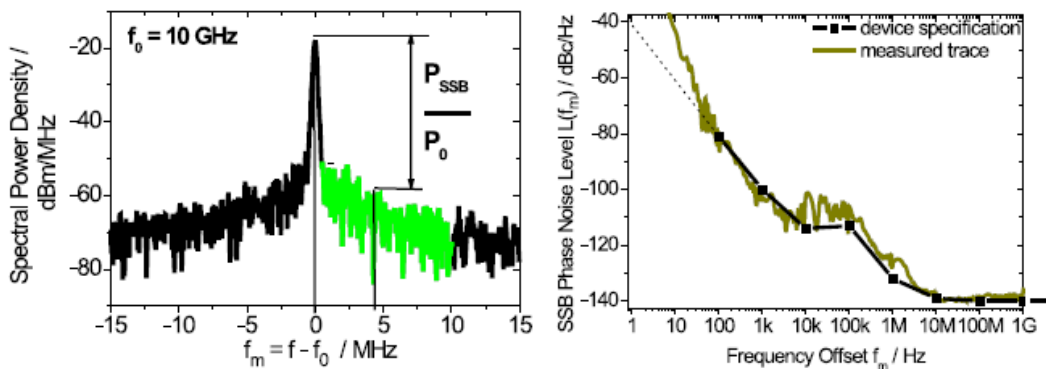


Figura 3.17. Análisis del Ruido de fase de SSB y comparación con las especificaciones del fabricante (derecha) (12).

3.2.2 Interfaz del analizador de espectros eléctrico (RFSa)

En el desarrollo de este interfaz el instrumento que se ha de controlar mediante software es el Analizador de espectros eléctrico Anritsu MS2668C. Obteniéndose el interfaz presente en la Fig.1.2.

Para el desarrollo del interfaz se utiliza la herramienta software de MATLAB, igualmente que en el caso de la caracterización de la curva P-I. Posteriormente a la realización del interfaz esta es calibrada mediante la utilización del Generador de RF Anritsu MG3695A.

El espacio mostrado al usuario en la Fig. 1.2. está dividido en cuatro regiones: conexión con el instrumento, especificación de parámetros, visualización de la traza del RFSa y medida

del ruido de fase. La inicialización del interfaz de caracterización del ruido de fase es semejante a la correspondiente a la caracterización de la Curva P-I presente en el Anexo E, en ella también se declaran los valores por defecto de los campos dinámicos, se actualizan los manejadores y se declaran las etiquetas de las graficas utilizadas.

3.2.2.1 Panel de Conexión

Este panel es utilizado para establecer la conexión con el analizador de espectros. Al igual que con la fuente de corriente en el interfaz de caracterización de la Curva P-I, la conexión se realiza mediante un objeto VISA. La dirección asignada en el laboratorio para el analizador de espectros es el número 1. Este valor aparece en el campo de dirección por defecto. Es fundamental comprobar que este valor es el presente en el dispositivo antes de realizar la comunicación.

Si se intenta realizar una operación sin haber establecido la conexión se presenta un mensaje de información al usuario y esto ocurre también en el caso de intentar establecer repetidamente la comunicación una vez ya realizada. Ambos mensajes aparecían en el interfaz de la Curva P-I.

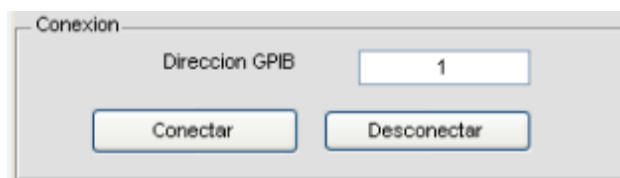


Figura 3.18. Panel de conexión (Interfaz 2).

3.2.2.2 Panel de Parámetros

En este panel se permiten establecer los parámetros determinantes en la realización de un barrido de la señal en el RFSA, como son la frecuencia central, el RBW y el SPAN. Se debe tener en cuenta que los dos últimos determinan el tiempo de barrido de la señal. Y durante ese tiempo se debe realizar una espera debido a que el RFSA no puede recibir otra instrucción a la vez que está realizando el barrido de la señal.



Figura 3.19. Panel de parámetros (Interfaz 2).

El RBW (ancho de banda de resolución) establece la capacidad de un analizador para distinguir dos componentes frecuenciales muy próximas entre sí. Para ello, el analizador de espectros permite seleccionar el ancho de banda del filtro de frecuencia intermedia (*IF-Intermediate Frequency*) de frecuencia más baja. La elección de este parámetro no solo modifica el valor de la resolución, sino que también el valor de la sensibilidad del instrumento. A medida que se reduce el ancho de banda de medida se reduce también el ruido térmico, y consecuentemente la señal a la salida es menos ruidosa mejorando el valor de la sensibilidad. El SPAN o ancho de banda de medida es la anchura del margen de frecuencias en el cual se desea medir el espectro (22).

El tiempo de barrido es el tiempo que tarda el analizador en barrer todo el margen de frecuencias en el que se hace la medida. Su valor depende principalmente al retardo producido por el filtro de resolución. Este retardo hace que las señales necesiten un tiempo no nulo para pasar a través del filtro, dando lugar a errores si se elige un tiempo de barrido demasiado corto. Si se desea realizar la medida correctamente, es necesario que la señal sea detectada en el mismo instante en el que aparece en pantalla (22).

En el caso de los tres primeros botones se sigue el siguiente diagrama de flujo:

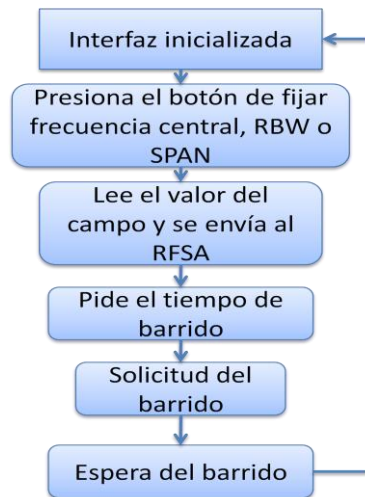


Figura 3.20. Diagrama de flujo de los botones fijar frecuencia central, fijar RBW y fijar SPAN.

El cuarto botón denominado PCF (Peak Center Frequency), ordena al instrumento centrar la frecuencia en el valor de máxima amplitud de la señal. El diagrama de flujo para este botón es semejante al mostrado para los tres botones anteriores, salvo que al final de su ejecución solicita el valor de la frecuencia central fijada y es mostrado en el campo dinámico a la derecha del botón en color rojo.

El siguiente botón solicita el valor de la amplitud que tiene la frecuencia central, el valor es mostrado en el campo dinámico a la derecha del botón en color rojo y en dB's. La solicitud de la amplitud se realiza mediante un marker que se sitúa en la frecuencia central y del cual se puede pedir la amplitud al RFSA.

El botón PRL (Peak Reference Level) sigue un esquema similar al del diagrama de flujo de la Fig. 3.20. Cuando es pulsado asigna el nivel del mayor valor de amplitud del espectro al nivel de referencia. Y posteriormente realiza un nuevo barrido de la señal. El código correspondiente a este panel se encuentra en el Anexo L.

3.2.2.3 Panel de Traza

En este subpanel se puede visualizar la traza que es representada en el RFSA, según los ejes que aparecen en la Fig. 3.21. Esta función se realiza cuando es pulsado el botón ‘Mostrar Figura’. Para el correcto desarrollo de esta función se ha de tener en cuenta el número de puntos en los que se divide la traza presente en el RFSA, el valor es 501. Cada uno de estos valores es enviado consecutivamente tras realizar su solicitud. Antes de solicitar los datos se debe especificar el formato, para este interfaz el formato seleccionado es ASCII. Los datos recibidos también requieren de una transformación ya que son recibidos en una escala logarítmica de 0.01dBm.

Para que las imágenes no se solapen si se desea mostrar una nueva figura, se utiliza el botón Limpiar. Además, se dispone del botón Guardar, el cual abre una ventana de diálogo al ser pulsado para salvar los datos como la presente en la fig. 3.10. Los datos quedan guardados en un archivo .mat donde se guardan los dos ejes de la figura. Esto permite cargar de nuevo esta imagen a través del botón ‘Cargar Figura’. El extracto de código se encuentra en el Anexo M.

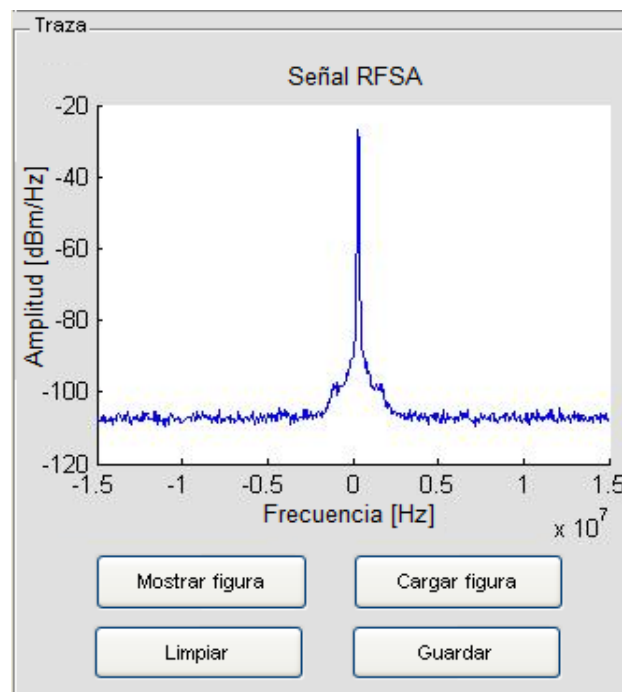


Figura 3.21. Panel de traza (Interfaz 2).

3.2.2.4 Panel del Ruido de Fase

El panel del ruido de fase es mostrado en la figura siguiente:

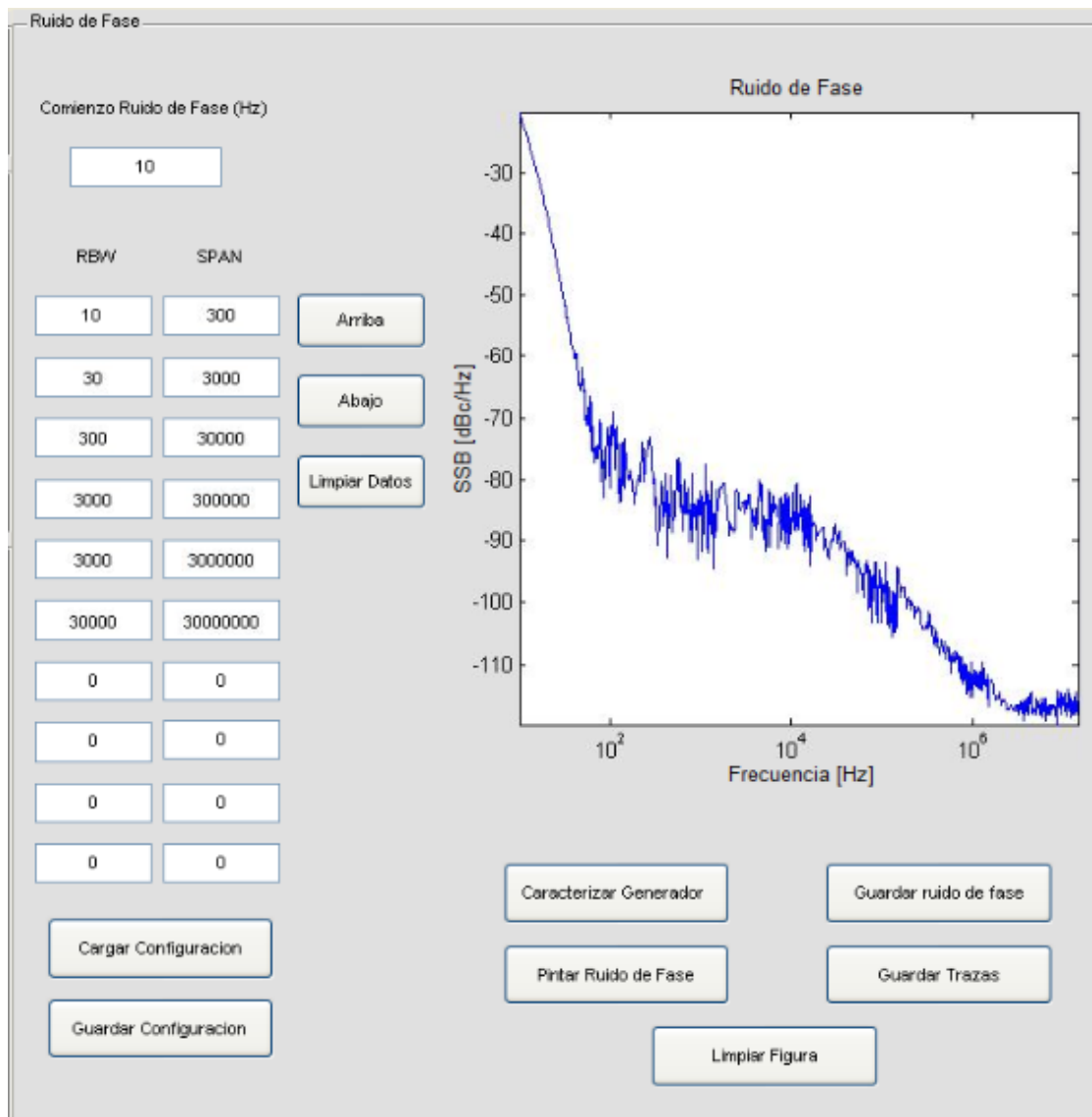


Figura 3.22. Panel del Ruido de Fase (Interfaz 2).

El ruido de fase se compone del solapamiento de distintas trazas en torno a una portadora con diferentes valores de RBW y SPAN. Permitiendo observar la falda producida por el ruido de fase en los laterales de la portadora.

La metodología para obtener el ruido de fase consiste en tomar una traza para una pareja especificada de RBW y SPAN. Si es la primera traza el inicio tomado para obtener el ruido de fase es un valor especificado. Este valor dado en frecuencia debe identificarse con uno de los puntos obtenidos de la traza. El final de esa parte del ruido de fase se corresponde con el final de la traza. Si no es la primera traza, el final de la traza analizada anteriormente es el principio de la siguiente.

El valor en amplitud de la traza ha de ser normalizado al valor máximo de la portadora, además de ser relacionado a un RBW de 1Hz.

Según la fórmula 3.2 se determina la siguiente sentencia de código. Donde para que el valor de amplitud que representa la variable leído quede referenciada a 1Hz se debe restar el término correspondiente al RBW utilizado en esa traza. De la operación se obtienen dBc/Hz.

$$\text{leido} = \text{leido} - \max(\text{leido}) - 10 \cdot \log_{10}(\text{RBW});$$

El ruido de fase presente en la fig. 3.22 se obtiene al pulsar el botón ‘Caracterizar generador’ que permite calibrar el interfaz. Para realizar esta simulación se introduce una portadora en la frecuencia de 10 GHz desde el Generador de RF Anritsu MG3695A al RFSA. El extracto de código correspondiente a esta función se encuentra en el Anexo N. El extracto de código únicamente presenta las sentencias correspondientes al cálculo realizado con una traza, ya que el resto de la función son sentencias repetitivas. El ruido de fase del generador se caracteriza a partir de 6 parejas de valores de RBW y SPAN. La diferencia con respecto a la caracterización del ruido de fase de un láser mode locking aparece en la primera iteración donde el comienzo de ruido es 10 Hz cuando se caracteriza el generador mientras que en la caracterización de un dispositivo este valor se debe especificar. Los valores de RBW y SPAN según las iteraciones realizadas para la caracterización del generador aparecen en la tabla 3.1.

Iteración	RBW [Hz]	SPAN [Hz]
1	10	300
2	30	3000
3	300	30000
4	3000	300000
5	3000	3000000
6	30000	30000000

Tabla 3.1. Configuración para la caracterización del generador

En la caracterización de un dispositivo láser las parejas de valores de RBW y SPAN y el valor del comienzo de ruido deben poder ser modificables, buscando la configuración adecuada para el dispositivo en cuestión. Por ello, en el panel del ruido de fase a la izquierda aparecen dos columnas de diez campos dinámicos que permiten configurar estos valores. Así como el campo superior que se corresponde con el comienzo del ruido de fase y toma como valor por defecto 10 Hz.

A la derecha de estas columnas y en la parte inferior aparecen un total de 5 botones que permiten manipular la configuración para poder pintar el ruido de fase de un láser de semiconductor. Los botones a la derecha permiten limpiar los datos introducidos y desplazarlos hacia arriba y hacia abajo, facilitando su eliminación y la introducción de otros valores. Por otra parte, los botones correspondientes a la parte inferior permiten guardar la configuración con todos los valores de RBW, SPAN y el comienzo del ruido de fase, o cargar una configuración ya utilizada a través de una ventana de dialogo fig. 3.23. El código correspondiente a estos botones se encuentra en el Anexo O.

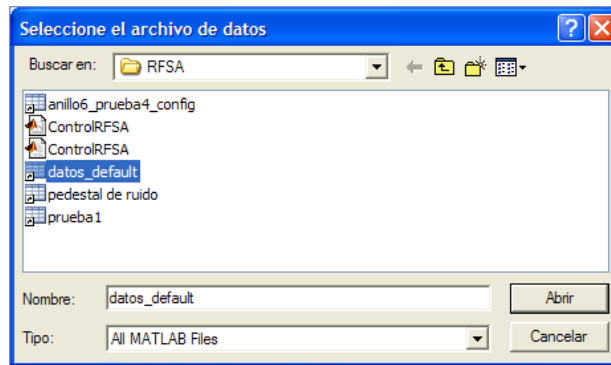


Figura 3.23. Ventana dialogo para cargar configuración (Interfaz 2).

Tras realizar la configuración si es pulsado el botón Pintar se caracteriza el ruido de fase del láser de semiconductor, con los valores RBW, SPAN e inicio del ruido de fase introducidos. Los valores correspondientes al RBW pueden no ser permitidos por el RFSA ya que deben ser valores correspondientes al ancho de banda de los filtros de resolución de los que dispone el RFSA. Cuando estos no se correspondan a los filtros utilizados por el RFSA, el instrumento aproxima ese RBW al filtro inmediatamente superior. Si esto ocurre el valor de RBW aproximado se introduce en el campo asignado en color rojo, permitiendo su visualización al usuario. El código correspondiente a parte de esta función se encuentra en el Anexo P.

Además de guardar la configuración utilizada se pueden guardar el ruido de fase obtenido y las trazas. En las tres funciones se abre una ventana de dialogo como la presente en la fig. 3.10. Para guardar estos valores son utilizadas variables globales. Al guardar el ruido de fase únicamente son guardados los dos ejes en un fichero .mat. Mientras que al pulsar el botón “Guardar Trazas”, se puede seleccionar almacenar los datos en un fichero .mat o .xls. Si se elige un fichero .mat son guardados el RBW, el SPAN, la frecuencia central, la potencia de pico, el eje y las correspondientes amplitudes de cada una de las trazas en distintos ficheros .mat para cada una de las trazas. Las distintas trazas son guardadas en ficheros con el nombre seleccionado seguido del RBW y SPAN de cada traza en cada uno de los nombres que se da a los ficheros (ej. nombre_RBW300_SPAN1000000.mat). Si se selecciona que las trazas sean guardadas en Excel (fichero .xls) los datos de cada traza se guardan en una hoja independiente por columnas. Por ejemplo, en la primera hoja en la columna A y B aparecen correspondientemente el eje de la traza 1 y sus amplitudes. En el campo C1, D1 y E1 se guardan el valor de frecuencia central, RBW y SPAN utilizados para esa traza. Y en el campo F1 se almacena la potencia de pico obtenida. El código correspondiente a la función guardar trazas se encuentra en el Anexo Q.

Al igual que en todas las representaciones en la parte inferior de este panel aparece un botón que permite limpiar la figura.

3.2.2.5 Cierre ordenado

Este interfaz también realiza un cierre ordenado, permitiendo volver a ser llamada. Cuando el usuario presiona el botón de cierre del interfaz, aparece una ventana de dialogo (Fig. 3.11.) permitiendo decidir al usuario si desea o no cerrar el interfaz. Si el interfaz es cerrada primeramente se desconectan los instrumentos y después son eliminados todos los datos de la creación del interfaz.

Capítulo 4

Resultados obtenidos

Este capítulo recoge los resultados obtenidos con los interfases desarrollados, de las medidas que se han realizado sobre el chip que contiene hasta ocho láseres en anillo y que se presento en el capítulo 1, a los que se hará referencia numerándolos de 1 al 8 de arriba a abajo según su disposición en la figura 1.12. Los diferentes resultados que se presentan han permitido identificar sus regímenes de funcionamiento de forma experimental.

Como primera medida, se ha obtenido la curva P-I medida básica para la caracterización de todo dispositivo. Esta permite medir la corriente umbral del dispositivo. Adicionalmente, durante la realización de esta medida, se midió el espectro óptico del dispositivo. Los datos de este instrumento han sido adquiridos mediante un interfase suministrado por el fabricante. Esta medida resulta de interés ya que permite analizar la separación de los modos, la cual revela las dimensiones de la cavidad. Si se observa que la distribución de amplitudes de los modos indica, que estos pueden estar enganchados en fase y la potencia óptica es adecuada para que esta sea introducida en el fotodetector, esta señal es analizada con el RFSA. Posteriormente, si en el espectro eléctrico aparece un tono que determine el régimen mode-locking es caracterizado el ruido de fase.

4.1 Curva P-I

La caracterización de la curva P-I se realiza a través del interfaz desarrollado en el presente proyecto (fig. 1.1) y descrito en el capítulo 3. Como se ha indicado, esta es una medida importante para caracterizar los dispositivos, pues permite verificar que este opera como láser tras los diferentes procesos de fabricación.

La estructura particular que tienen los láseres en anillo da lugar a una serie de consideraciones a la hora de realizar las medidas de su curva P-I. Una primera, y fundamental, es que dada la cavidad circular, la potencia óptica se extrae mediante un acoplador interferencial multimodo (MMI), con dos salidas correspondientes a las dos direcciones de propagación (horaria – CW- y antihoraria – CCW-). Las graficas de potencia que se presentan a continuación muestran

Capítulo 4: Resultados obtenidos

la potencia óptica medida en ambas salidas, superpuestas sobre la misma grafica. Esto permite visualizar la potencia óptica en cada dirección de propagación, permitiendo comparar ambas salidas. A la hora de realizar estas medidas se presento un problema, dado que una vez medida la potencia óptica en una dirección, el dispositivo se debe cambiar de posición para realizar la medida de la dirección contraria. Esto se debe a que únicamente es posible la situación del soporte de la fibra óptica en el lado derecho de la base de puntas (Fig. 4.1).

Previamente a la realización de la curva P-I definitiva, se debe proceder al alineamiento de la fibra (Fig. 4.2). Por ello, se realiza una curva inicial localizando la corriente que permite obtener la potencia máxima.

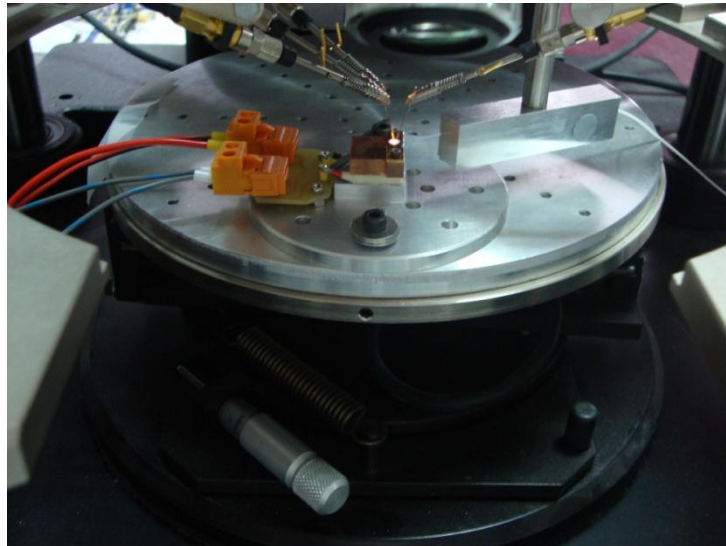


Figura 4.1. Localización de las puntas y soporte de la fibra óptica.

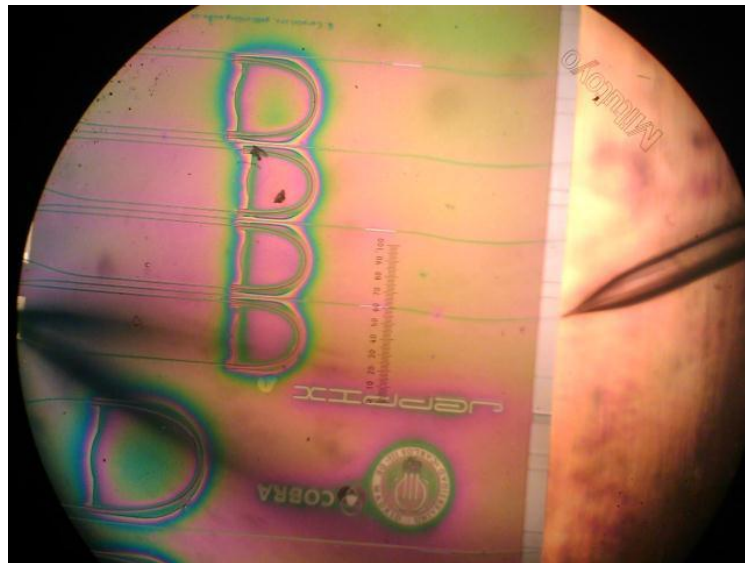


Figura 4.2. Alineamiento de la fibra.

A continuación, se presentan las curvas obtenidas en los diferentes tipos de estructuras en el chip. Las estructuras se diferencian en la ubicación y número de secciones de absorbente saturable, dispuestas para enganchar los modos del dispositivo. Dependiendo de la disposición de la cavidad circular respecto a estas secciones, podemos definir estructuras simétricas (si trazando

una línea vertical respecto a la posición del SA la cavidad presenta simetría especular respecto a este eje) y estructuras asimétricas. En este capítulo se presentan los resultados obtenidos aplicando diferentes tensiones inversas sobre los absorbentes saturables.

4.1.1 Láser en anillo 1

El láser en anillo número 1 es una estructura asimétrica, en la que se presentan cuatro zonas de ganancia. Estas han sido unidas mediante un hilo de oro por “*wire bonding*”. Ello permite inyectar la corriente de polarización en estas secciones utilizando una única punta de acceso (Fig. 4.3). Esto es importante por el limitado número de puntas del que se dispone, y el limitado espacio para acceder al dispositivo.

De esta forma, sólo son precisas dos puntas mas para alimentar los dos absorbentes saturables con los que cuenta esta estructura (Fig. 4.3). Las pruebas realizadas contemplan diferentes situaciones respecto de la alimentación de los absorbentes saturables, pudiendo estar:

- Sin conexión (NC- *not connected*), cuando en el pad no se apoya punta alguna.
- En abierto (OC – *open connection*), cuando en el pad se apoya una punta conectada a la fuente de tensión, pero la salida de esta no se encuentra habilitada.
- Conectado, cuando en el pad se apoya una punta conectada a la fuente de tensión, y esta se encuentra suministrando una tensión en inversa.

En este dispositivo pueden darse diversas combinaciones según se conecten ambos absorbentes o solo uno de ellos.

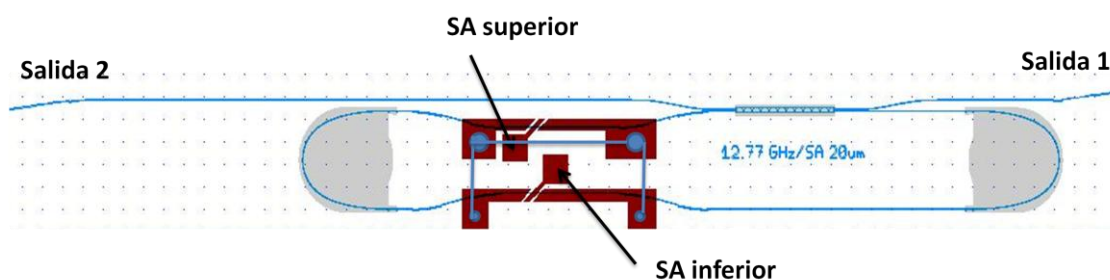


Figura 4.3. Láser en Anillo 1

Polarización simultánea de ambos Absorbentes Saturables

Los dos absorbentes saturables son alimentados desde la misma fuente de tensión, aplicando la misma tensión inversa. Se han realizado medidas de la curva P-I para los siguientes valores de tensión aplicados: -1V (Fig. 4.4) y -2V (Fig. 4.5).

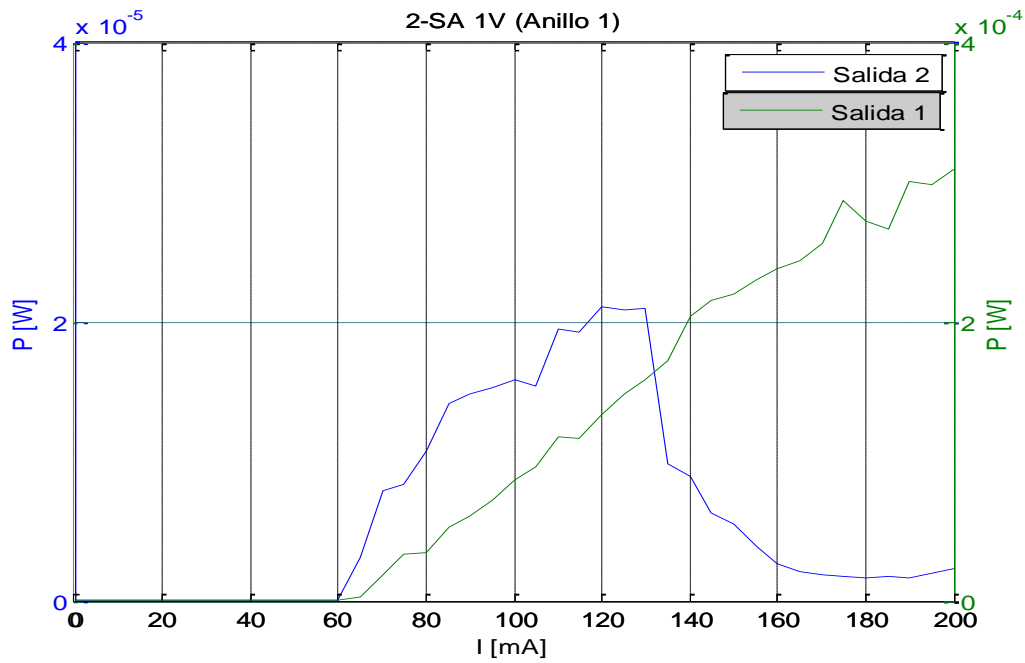


Figura 4.4. Láser en anillo 1 (2-SA a -1V)

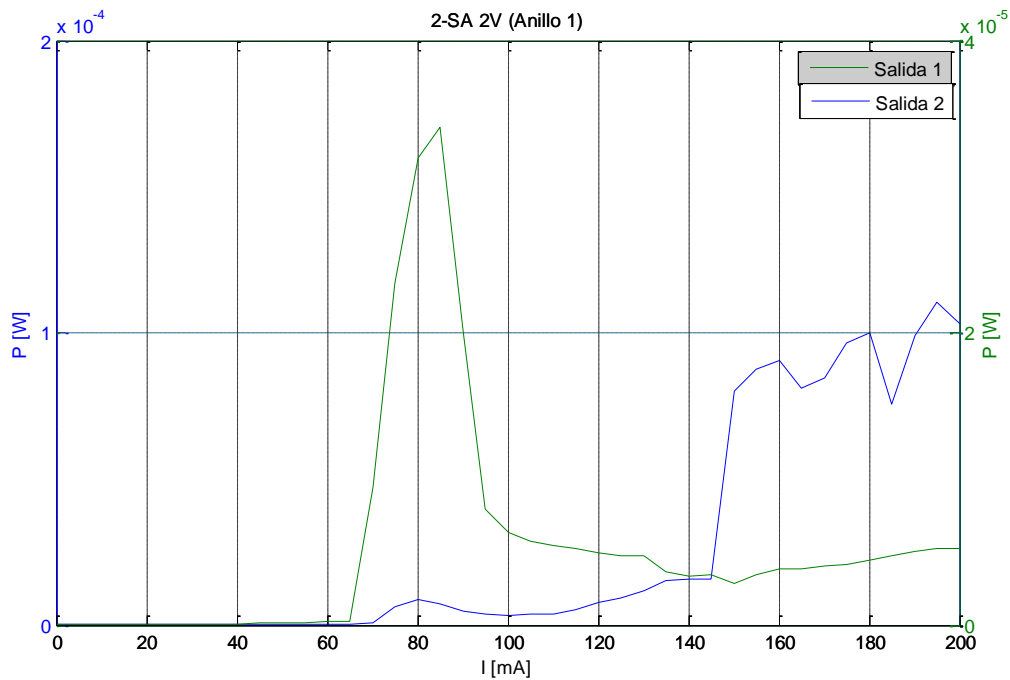


Figura 4.5. Láser en anillo 1 (2-SA a -2V)

Tensión	Salida	P [μW]	I[mA]	Ith[mA]
2 SA -1V	Salida 1	309.2	200	65
	Salida 2	20.85	125	62
2 SA -2V	Salida 1	34.03	85	67
	Salida 2	110.2	195	70

Tabla 4.1. Resultados Anillo 1 - 2SA

Con el fin de analizar los resultados obtenidos, en el siguiente cuadro se presentan los valores más significativos obtenidos de las graficas anteriores para cada valor de tensión aplicada en los absorbentes saturables en cada una de las dos salidas. Se especifican los valores de máxima potencia y la corriente a la que se consigue esa potencia. Además se incluye el valor de la corriente umbral que se obtiene en cada caso. La corriente umbral ha sido determinada utilizando la técnica de post-procesado de la primera derivada de la función tras representarla en ejes semilogarítmicos (eje Y).

En vista de las curvas P-I que se han obtenido, en las que se puede observar el incremento abrupto de la emisión de luz llegados a un determinado nivel de corriente, se puede afirmar que el dispositivo funciona.

En las figuras obtenidas cuando son polarizados por igual los dos SA se observa que dependiendo de la corriente el modo dominante es CW o CCW. En la figura 4.4 el CCW es el modo dominante para el rango de corrientes próximos a la corriente umbral, mientras que cuando se aplica una tensión inversa de 2V, los papeles se invierten. Entonces el modo dominante a valores de corriente menores es el CW y a mayores el CCW. Esta primera observación ha permitido identificar el hecho de que la disposición de múltiples absorbentes saturables influye notablemente en la competición de modos que se contrapropagan en el anillo.

También se puede observar como el modo que alcanza su potencia máxima para una corriente menor, obtiene menor valor de potencia que el otro. Esto es coherente con la inyección de portadores que se produce a dicha corriente.

Polarización individual de los Absorbentes Saturables

A continuación se ha medido la curva P-I cuando se polariza únicamente un solo absorbente saturable. En primer lugar cuando se conecta el absorbente saturable inferior mientras que el superior permanece en estado NC. Las medidas se han realizado bajo dos situaciones para el absorbente saturable inferior, en OC (Fig. 4.6) y bajo una tensión inversa de 2V (Fig. 4.7).

En las trazas presentadas se observa que al aplicar únicamente tensión inversa sobre el absorbente saturable inferior, la potencia máxima de la salida 1 (correspondiente al modo que se propaga en dirección horaria – CW-) del anillo se desplaza en el eje de corrientes hasta coincidir con el valor en el que se produce la máxima potencia en la salida 2. En la tabla 4.2 se presentan las potencias máximas obtenidas con la corriente aplicada, y las corrientes umbrales, realizando el mismo análisis que en el anterior apartado.

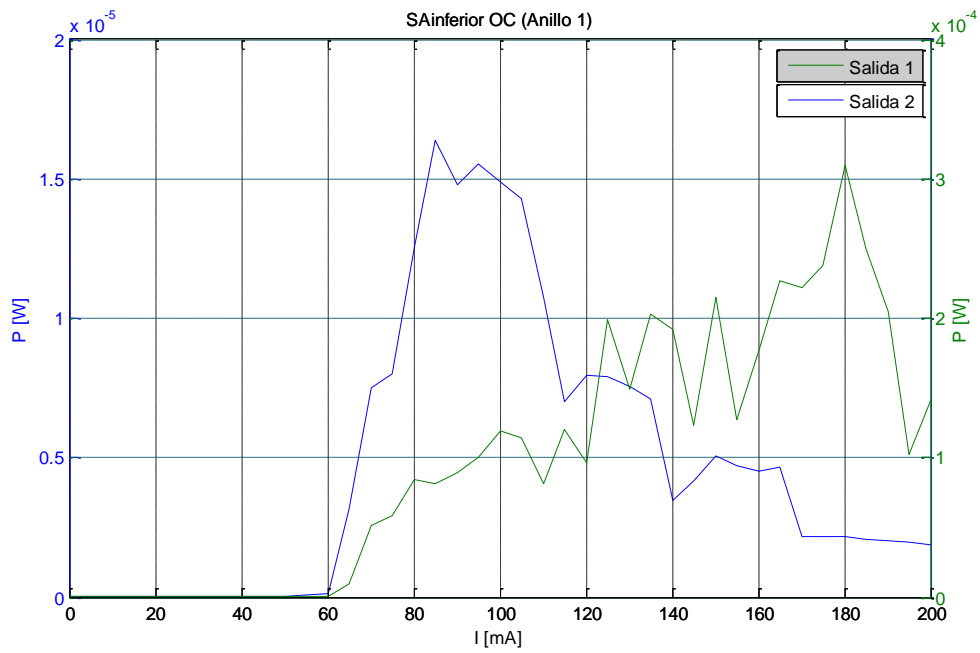


Figura 4.6. Láser en anillo 1 (SA inferior en OC)

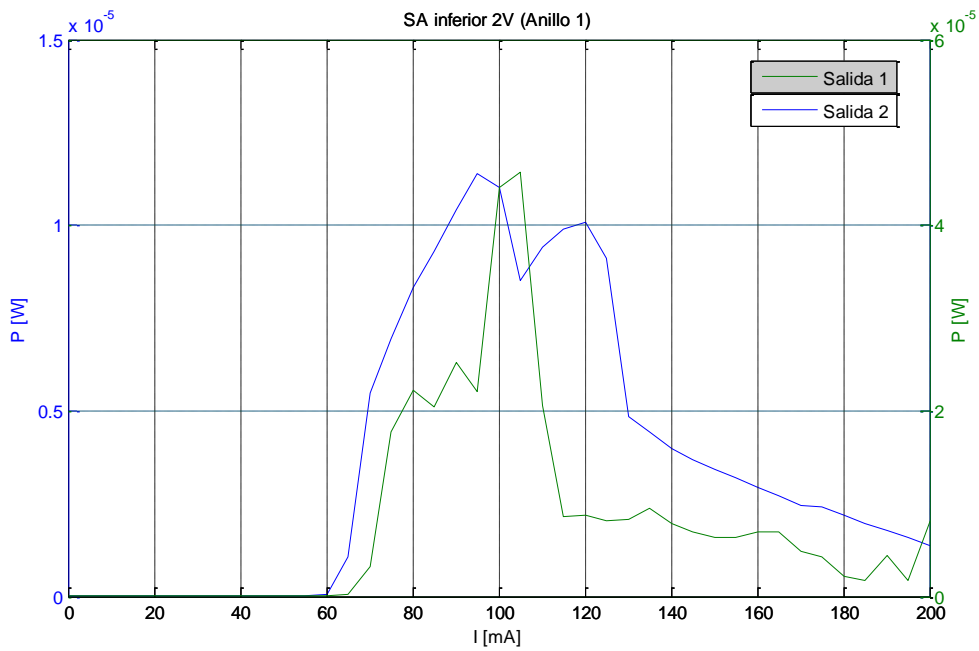


Figura 4.7. Láser en anillo 1 (SA inferior a -2V)

Tensión	Salida	P [μ W]	I[mA]	Ith[mA]
SA inferior OC	Salida 1	309.8	180	64
	Salida 2	16.36	85	64
SA inferior -2V	Salida 1	45.7	105	69
	Salida 2	11.39	95	64

Tabla 4.2. Resultados Anillo 1 – SA_inferior

Cuando se pasa a polarizar únicamente el absorbente saturable superior, se observa un cambio significativo en los resultados, según se presenta en las figuras 4.8 y 4.9. Cabe destacar los siguientes efectos observados.

En este caso, cuando el anillo superior este en OC, es el modo CW el que domina la cavidad en el entorno de corrientes próximo al umbral, al contrario de lo que ocurriría cuando se aplicaba la polarización inversa a los absorbentes. En cambio, cuando el absorbente es polarizado con una tensión inversa de 2V, la situación es que el modo CW resulta favorecido, pues deja de experimentar la caída en potencia que presentaba cuando el absorbente saturable estaba en OC. Al polarizar el absorbente superior se crea mayor asimetría en la cavidad, favoreciendo el modo CW.

Ello parece indicar, que efectivamente se logra compensar o aumentar el efecto que producen las asimetrías de la cavidad polarizando diferentemente los absorbentes en la cavidad.

En la tabla 4.3 se presentan los datos significativos de las graficas representadas.

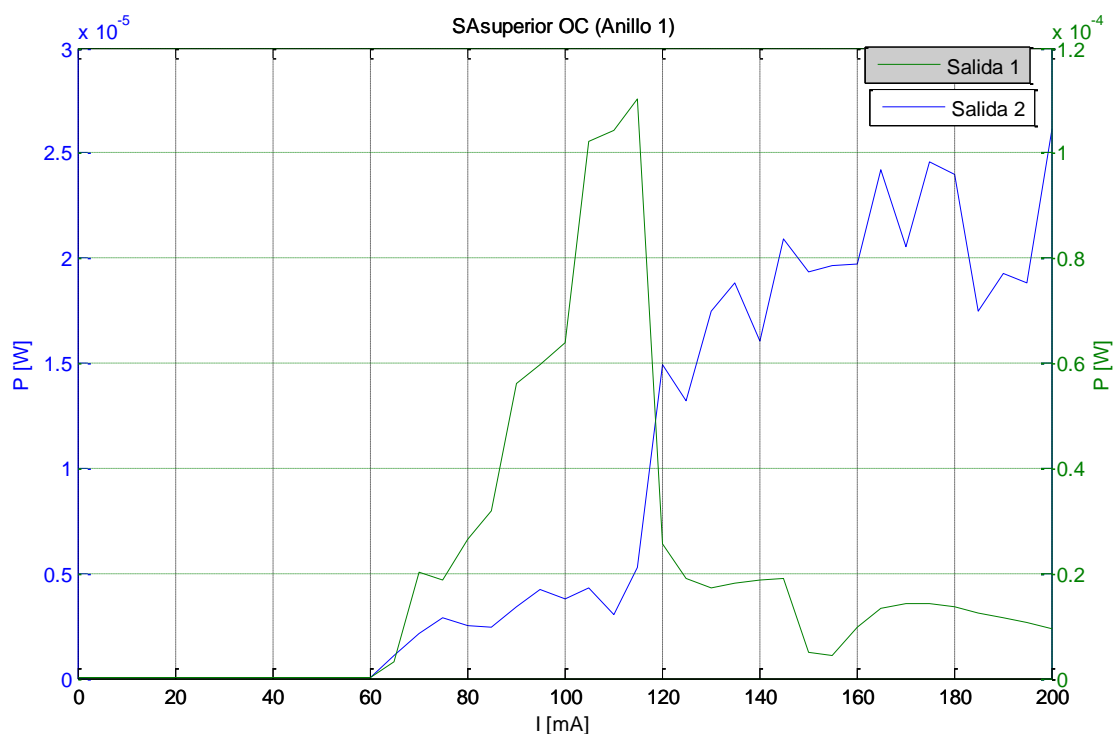


Figura 4.8. Láser en anillo 1 (SA superior en OC)

Tensión	Salida	P [μ W]	I[mA]	Ith[mA]
SA superior OC	Salida 1	110.4	115	65
	Salida 2	26.18	200	63
SA superior -2V	Salida 1	65.14	200	80
	Salida 2	11.37	185	67

Tabla 4.3. Resultados Anillo 1 – SA_superior

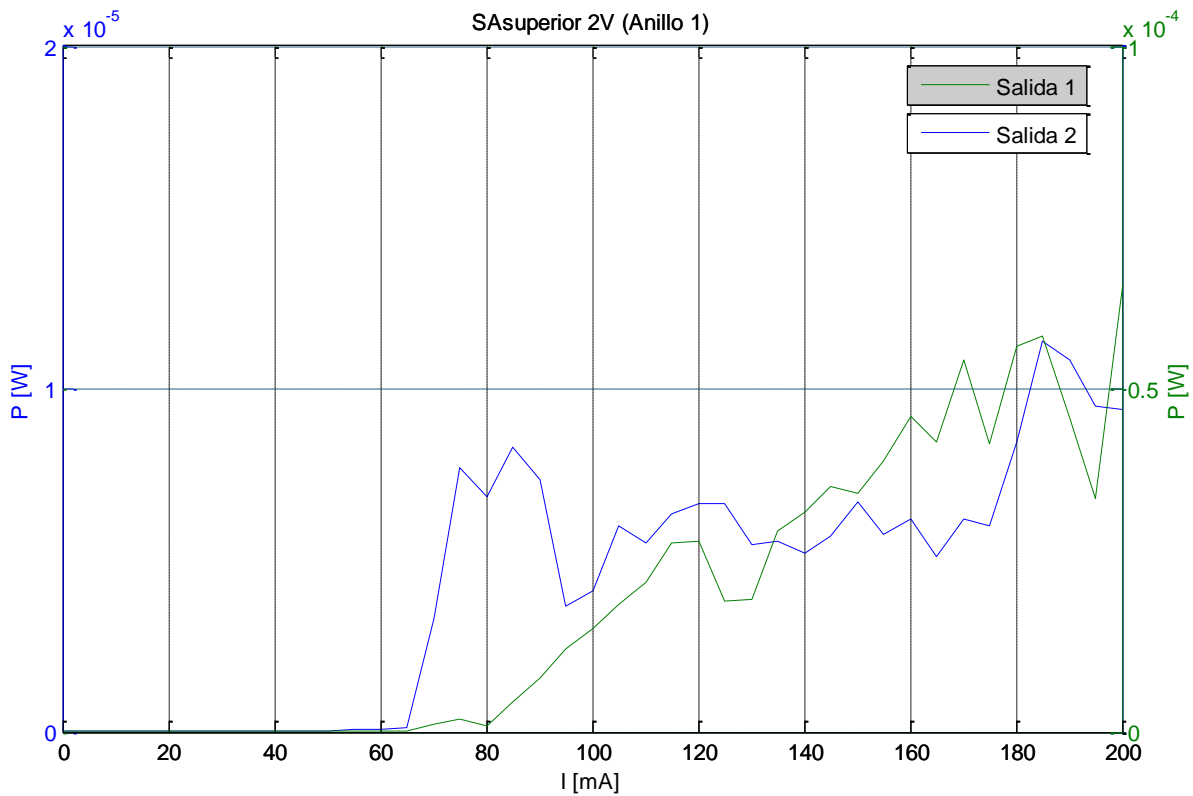


Figura 4.9. Láser en anillo 1 (SA superior a -2V)

4.1.2 Láser en anillo 8

En este apartado se pasa a analizar un láser en anillo simétrico, el anillo número 8, cuya estructura se presenta en la figura 1.12. Este se muestra independientemente en la figura 4.10 según es polarizado.

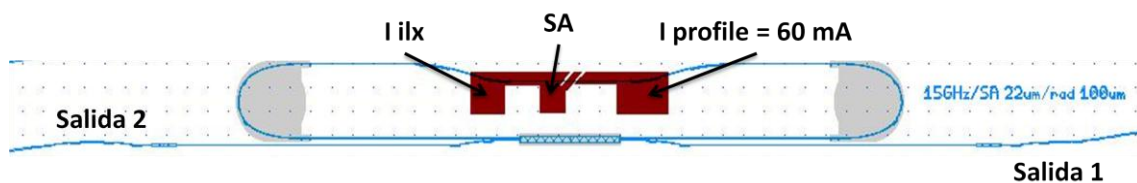


Figura 4.10. Láser en Anillo 8

En este caso, la zona activa queda dividida en dos secciones de ganancia de igual longitud, separadas por el absorbente saturable en el medio. El contacto presenta una inclinación de 45° con el fin de reducir reflexiones en la cavidad.

El barrido de corriente para obtener la curva P-I se realiza inyectando corriente únicamente en una de las zonas de ganancia, mientras que en la otra se inyecta un valor de corriente constante, a 60 mA. Las graficas realizadas para este láser también son para distintos valores de la polarización del absorbente saturable (OC y de 0-2V).

Si las graficas son observadas secuencialmente, se puede distinguir como la corriente a la que se obtiene la potencia máxima se desplaza a medida que se introduce más tensión inversa en el absorbente saturable. Llegando a tomar los mismos valores en ambas salidas cuando se polariza el absorbente saturable con 2V en inversa.

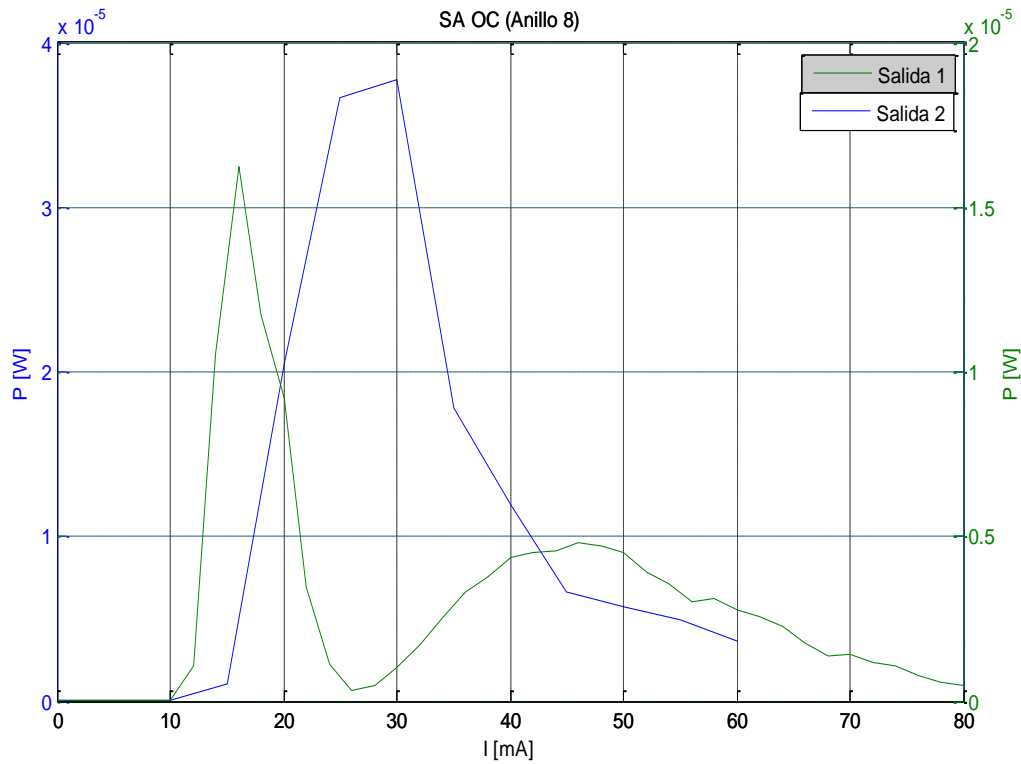


Figura 4.11. Láser en anillo 8 (SA en OC)

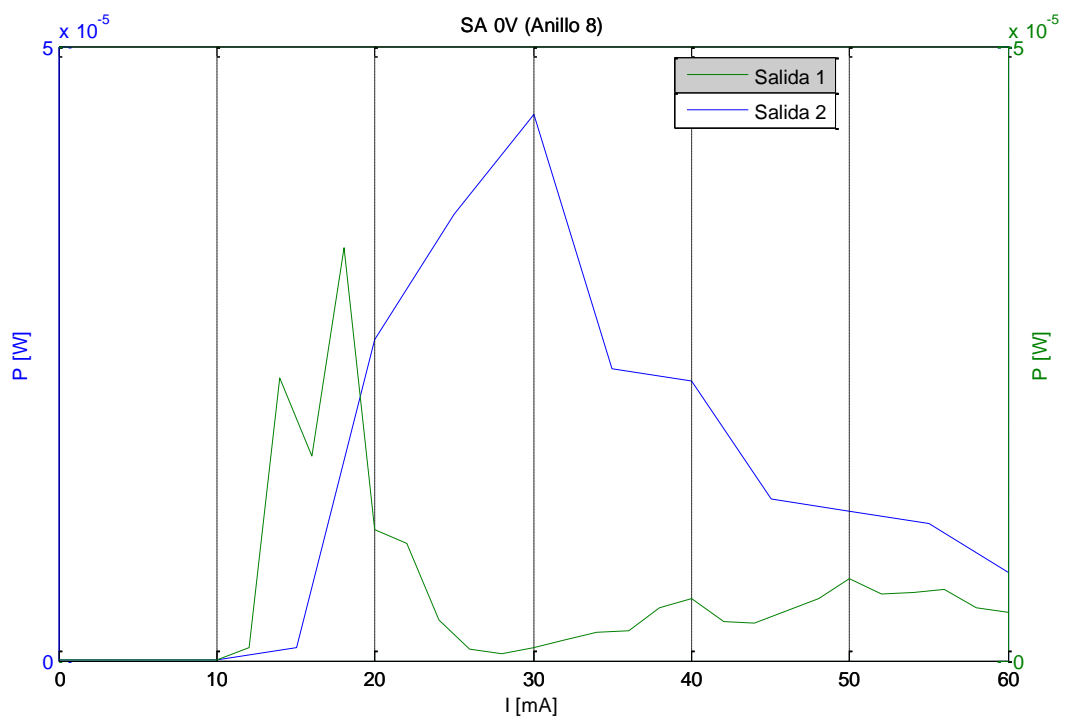


Figura 4.12. Láser en anillo 8 (SA a 0V)

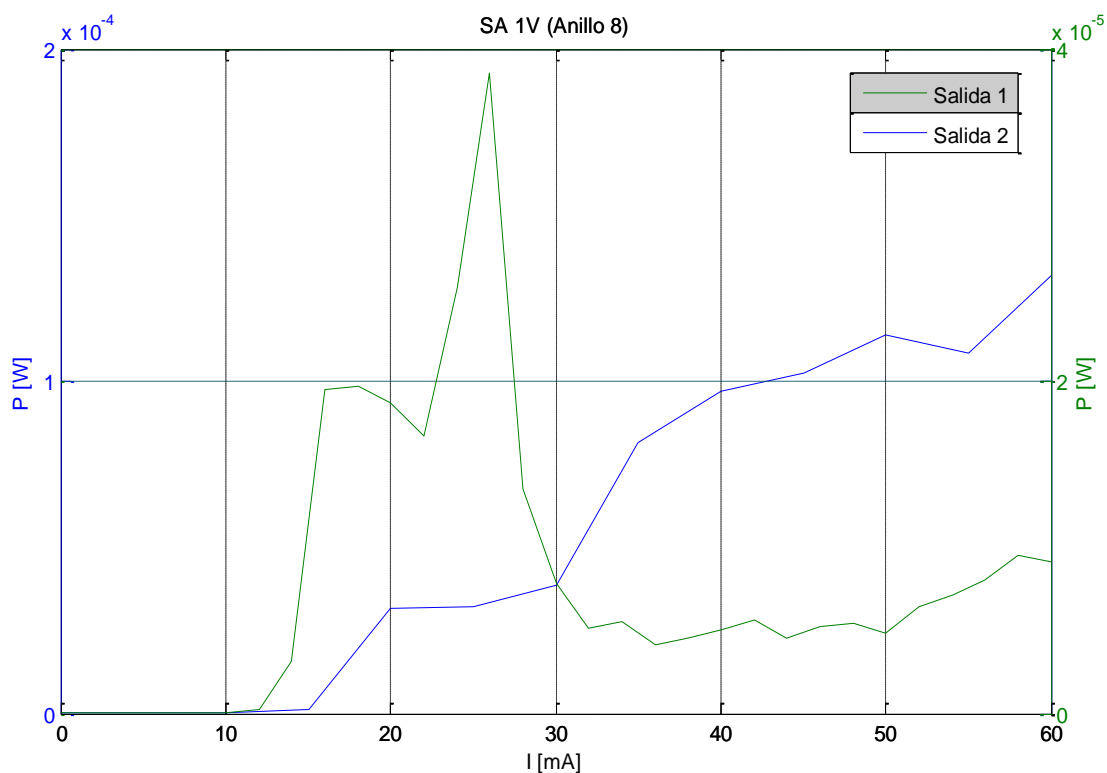


Figura 4.13. Láser en anillo 8 (SA a -1V)

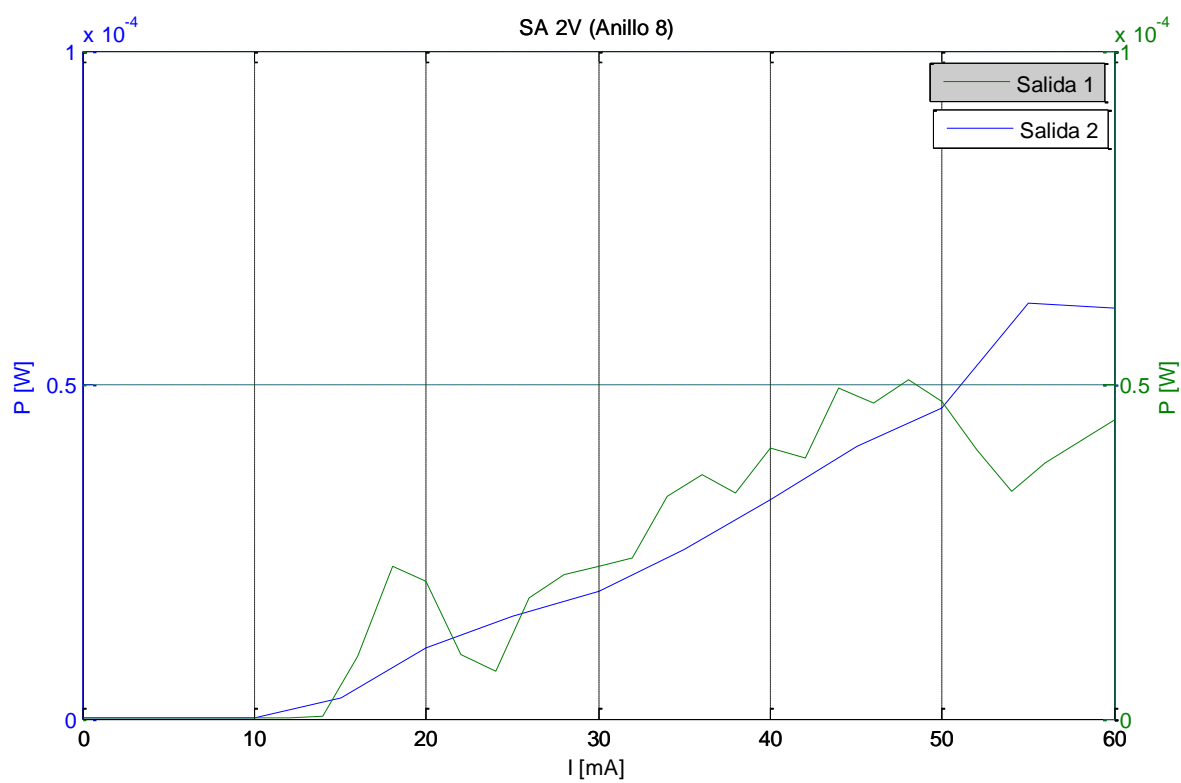


Figura 4.14. Láser en anillo 8 (SA a -2V)

Tensión	Salida	P [μ W]	I[mA]	I _{th} [mA]
SA OC	Salida 1	16.22	16	12
	Salida 2	37.75	30	15
SA 0V	Salida 1	33.6	18	12
	Salida 2	44.4	30	15
SA -1V	Salida 1	38.6	26	13
	Salida 2	132	60	15
SA -2V	Salida 1	50.66	48	15
	Salida 2	61.49	60	12

Tabla 4.4. Resultados Anillo 8

Según las figuras presentadas en este apartado, podemos afirmar que ambos láseres en anillo (1 y 8) funcionan ya que en ellos se observa el umbral de emisión láser. Una de las características principales que se puede observar es que una vez superada la corriente umbral en el dispositivo ninguna de las salidas se anula. Esta es una propiedad que parece ser característica de los láseres en anillo procesados combinando secciones activas/pasivas. Ello contrasta con las observaciones en dispositivos láser en anillo completamente activos, como se presentó en el capítulo 2, en los que se observa la alternancia entre rangos de corriente para los que el dispositivo opera de forma unidireccional, bien CW o CCW, anulándose completamente la potencia óptica en la dirección opuesta.

La diferencia entre el anillo 1(asimétrico) y anillo 8 (simétrico) se analiza a partir de las curvas del anillo 1 con los 2 absorbente saturables. En este láser en anillo se observa siempre un modo dominante según la corriente aplicada, y el cual puede favorecerse mediante la aplicación de alimentaciones asimétricas en los absorbentes saturables del dispositivo. Por el contrario, en el anillo 8 al aumentar la tensión inversa del absorbente saturable, ambas direcciones presentan semejante potencia de salida, sugiriendo que el dispositivo entra en un modo de operación de colisión de pulsos entre los modos que se contrapropagan.

4.2 Medidas de espectro óptico

En este apartado se analiza el espectro óptico de los láseres en anillo para los cuales se han realizado las curvas P-I del apartado anterior (láseres en anillo 1 y 8). Con el análisis del espectro óptico se obtiene mayor información extremadamente importante del dispositivo como es el número de modos de emisión así como la separación entre estos. Este último dato es importante pues está relacionado con la longitud del resonador óptico y determina, en el caso de régimen mode - locking, la tasa de repetición a la que opera el dispositivo.

Incluimos además de los resultados en los láseres 1 y 8 las medidas obtenidas del espectro óptico en el láser en anillo 6 (Fig. 1.12.), dado que este es el dispositivo en el cual se ha podido analizar su respuesta en el analizador de espectros eléctrico.

4.2.1 Láseres en anillo 1 y 8

Para observar el espectro óptico simultáneamente con la potencia óptica a la salida del dispositivo, se ha dividido la potencia de la salida mediante un acoplador direccional con ratio de división 90/10 en sus salidas. Al lector de potencia se le ha conectado la salida con el 90% de la potencia entrante en el acoplador, permitiendo realizar el correcto alineamiento de la fibra. La salida con el 10% de la potencia óptica entrante se ha conectado al analizador de espectros ópticos (OSA).

Los resultados obtenidos para el láser en anillo 1 se presentan en la figura 4.16. Para realizar la medida del láser en anillo 1 la corriente inyectada es 190 mA y la tensión inversa de los dos absorbentes saturables es 2V. En el lector de potencia se obtienen 80 μ W.

Las figuras 4.18 y 4.19 presentan el espectro obtenido del láser en anillo 8. Dado que en este dispositivo las zonas de ganancia se encuentran separadas, se utilizan dos fuentes de corriente para inyectar con ambas 60 mA en cada sección, mientras que la tensión inversa para esta medida es de 1V, obteniéndose 100 μ W en el lector de potencia.

En las medidas realizadas se obtiene en primer lugar que la separación entre modos en el láser en anillo 1 es de 12.73 GHz, mientras que en el láser en anillo 8, esta es de 15.52 GHz. Esta frecuencia se corresponde con la proyectada en el diseño inicial y serigrafiada junto a los dispositivos (ver Fig. 1.12), determinada en cada dispositivo por las longitudes relevantes de la cavidad. En los láseres en anillo la tasa de repetición es inversa a la longitud del resonador L, según c/nL , siendo los parámetros de diseño de los láseres analizados los presentados en la siguiente tabla.

Ref. Anillo	Longitud total de la Cavidad / μ m	$f_{\text{repetición}}$ / GHz	Radio Curvatura / μ m
1	6643	12.2	500
2	6643	12.2	500
3	5844	15.27	500
4	5844	15.27	500
5	5405	~ 15	85
6	5405	~ 15	85
7	5405	~ 15	85
8	5625	~ 15	100

Tabla 4.5. Longitud total de la cavidad/ Frecuencia (GHz)

Para aumentar la potencia en ambos espectros son utilizados dos EDFAs (*Erbium Doped Fiber Amplifier*) (Fig. 4.15) con el propósito de conseguir suficiente potencia como para posteriormente analizar el ruido de fase en el RFSA. La ganancia máxima de cada EDFA es 20dBs, aunque lo que el instrumento permite configurar es la potencia de salida en dBm. El hecho de configurar la potencia de salida, determina que la ganancia depende de la entrada en cada

instante. Si la ganancia fijada era superior a la máxima permitida el instrumento informaba de un error.

De las medidas realizadas se puede apreciar la diferencia en el espectro óptico de un láser en anillo asimétrico de uno simétrico. En el láser en anillo 1 (asimétrico) se observa que existen modos dominantes que tienen una diferencia de 40 dBm con la mayor parte de los modos presentes en el espectro. Mientras en el láser en anillo 8 se tiene un espectro más uniforme. Esta es la razón por la cual a la salida de los EDFAs en el láser en anillo 8 se conservan más modos que en el láser en anillo 1.

Los espectros ópticos obtenidos a la salida de los dos EDFAs con el anillo 1 y el 8 son las figuras 4.17 y 4.20 respectivamente. En esta medida la entrada de los EDFAs es la salida del 90 % del acoplador que previamente se encontraba conectada al lector de potencia. La introducción de los EDFAs aumenta la potencia pero introduce un nivel de ruido que elimina un gran parte de los modos que quedan por debajo de su potencia de ruido.



Figura 4.15. EDFAs (NORTEL)

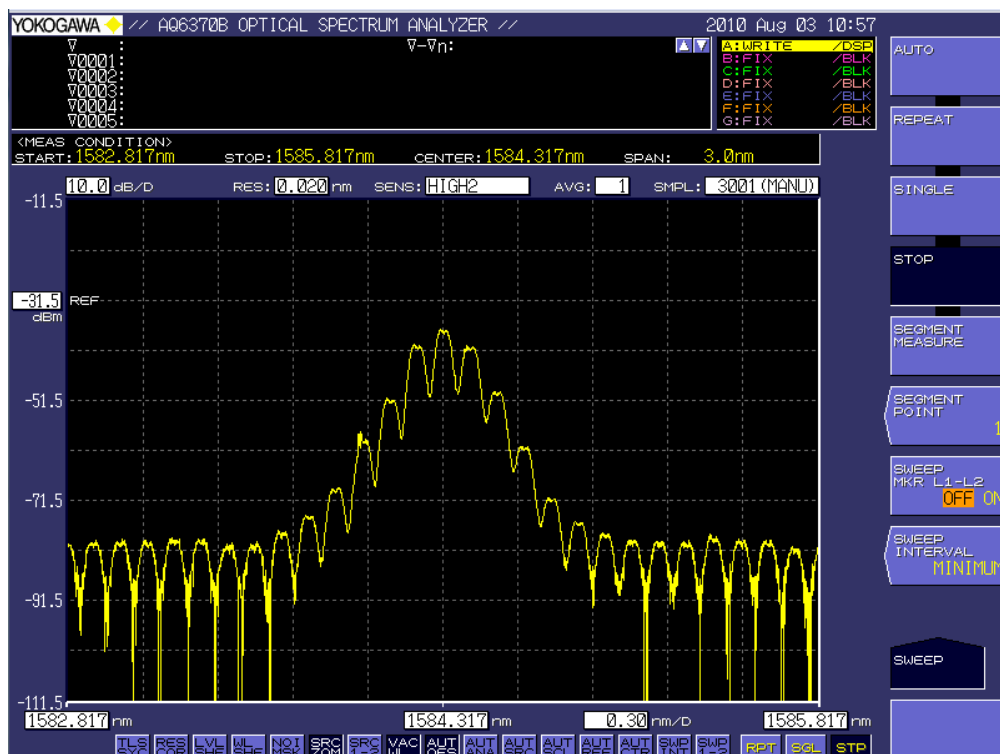


Figura 4.16. Espectro Óptico del anillo 1 (Salida del acoplador 10%)

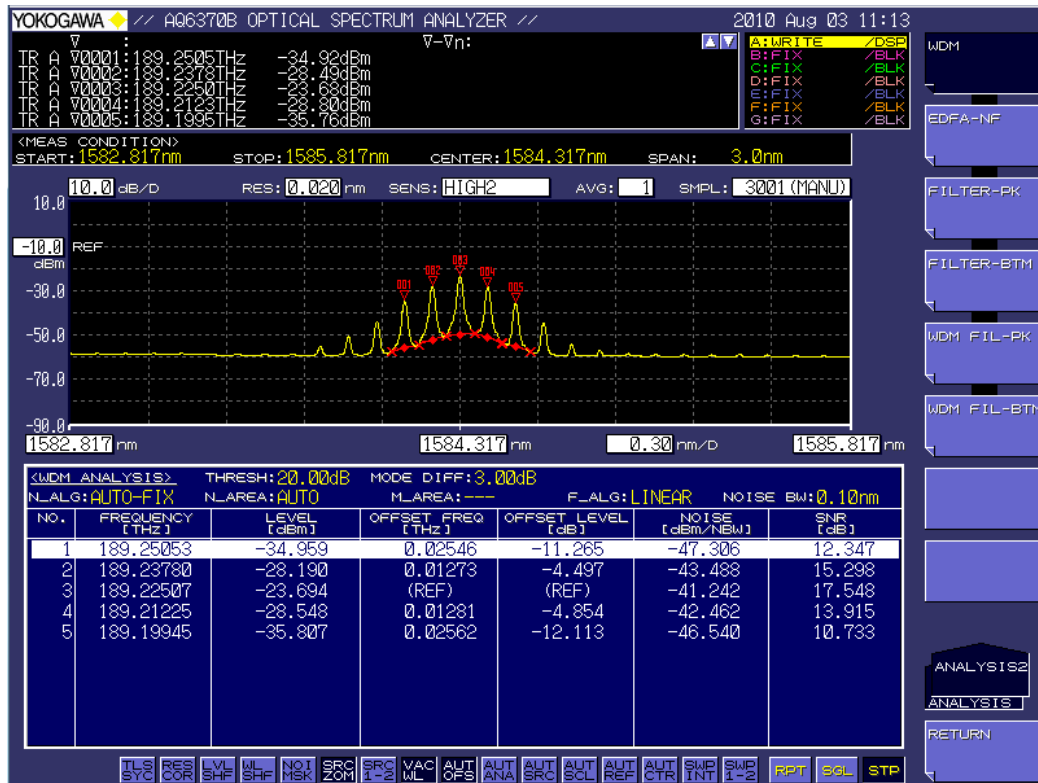


Figura 4.17. Espectro Óptico del anillo 1 (con 2 EDFAS)



Figura 4.18. Espectro Óptico del anillo 8 (Salida del acoplador 10%)

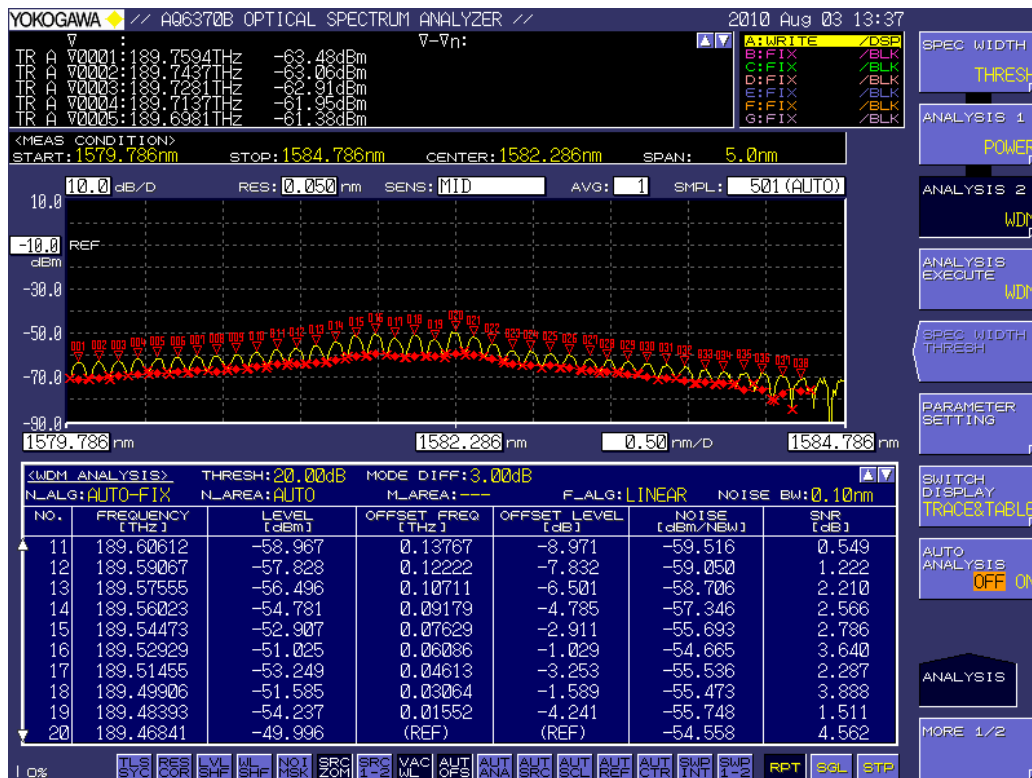


Figura 4.19. Espectro Óptico del anillo 8 (Salida del acoplador 10%, detallada)

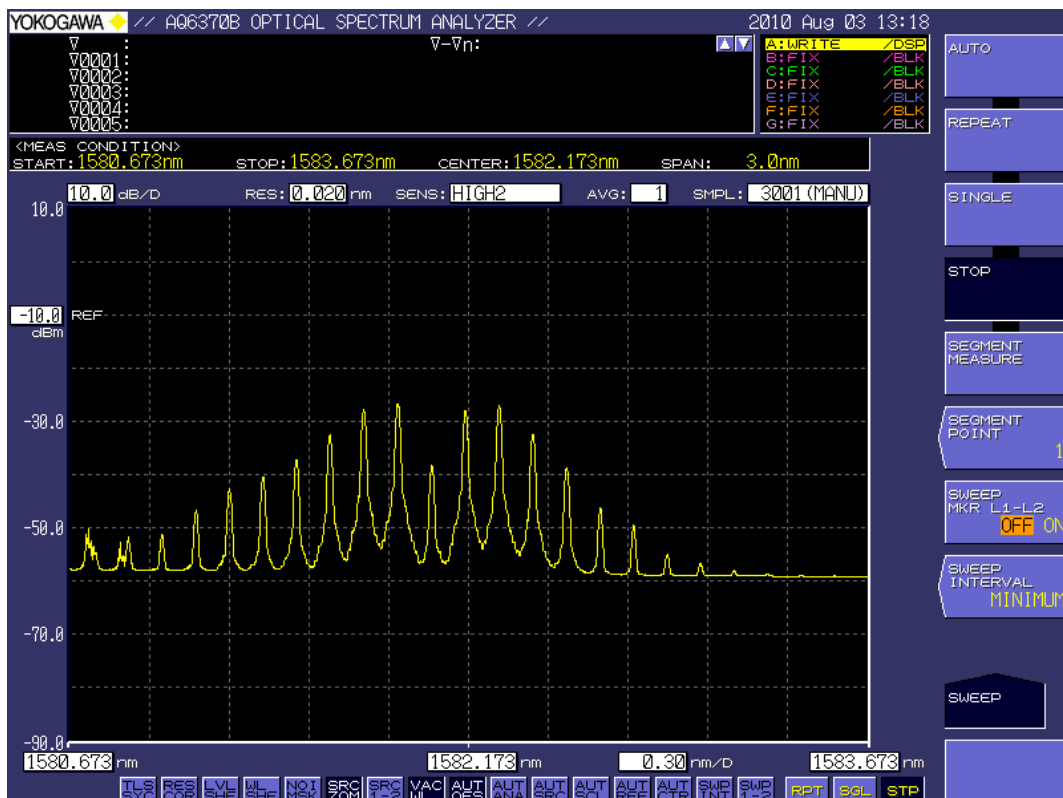


Figura 4.20. Espectro Óptico del anillo 8 (con 2 EDFAs)

4.2.2 Láser en anillo 6

Los EDFAs no proporcionan la suficiente potencia como para observar una señal en el RFSA. Por ello, en la medida realizada en el anillo 6 (Fig. 4.21) se utilizó el SOA-XN-OEC-1550 disponible en el laboratorio. El SOA (*Semiconductor Optical Amplifier*) se polarizó con 100 mA (Fig. 2.24 y Fig. 2.25). La medida inicial del espectro óptico sin el SOA (Fig. 2.22) permite observar un espectro más uniforme que en los mostrados en los láseres en anillo anteriores

La salida tomada en los tres anillos para realizar la medida del espectro óptico es la salida 2 tal como muestra la Figura 2.21.

Antes de conectar el SOA se conecta un aislador a su entrada para evitar las reflexiones de este instrumento. El SOA amplifica más ya que no elimina los modos, en comparación con los EDFAs pero introduce mucho ruido.

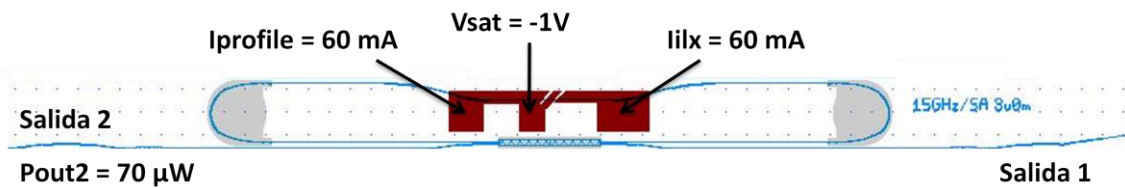


Figura 4.21. Láser en anillo 6

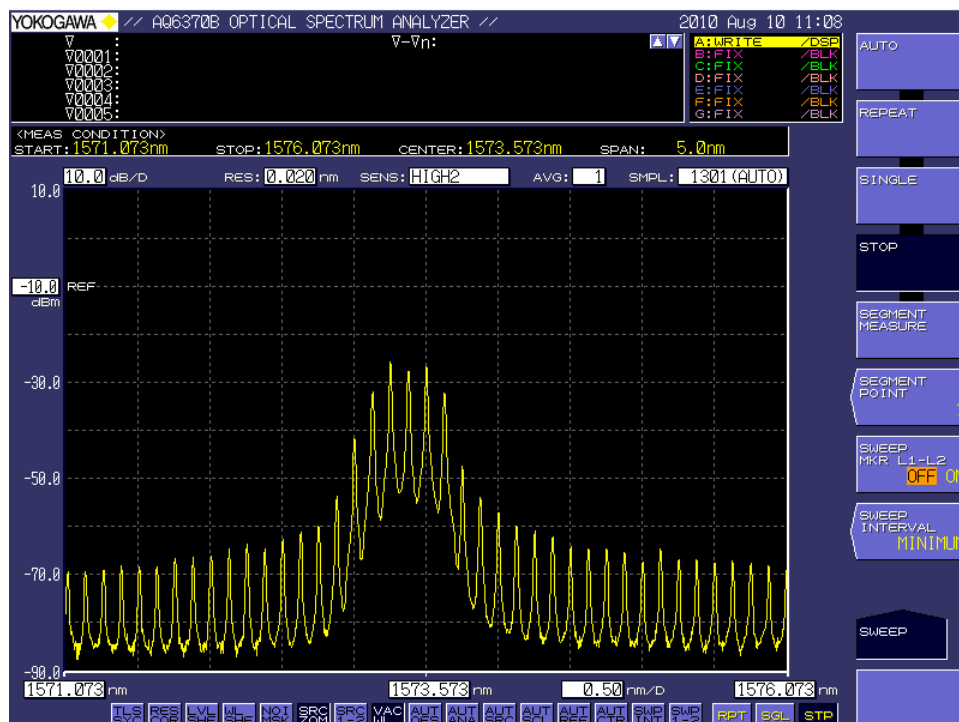


Figura 4.22. Espectro Óptico del anillo 6

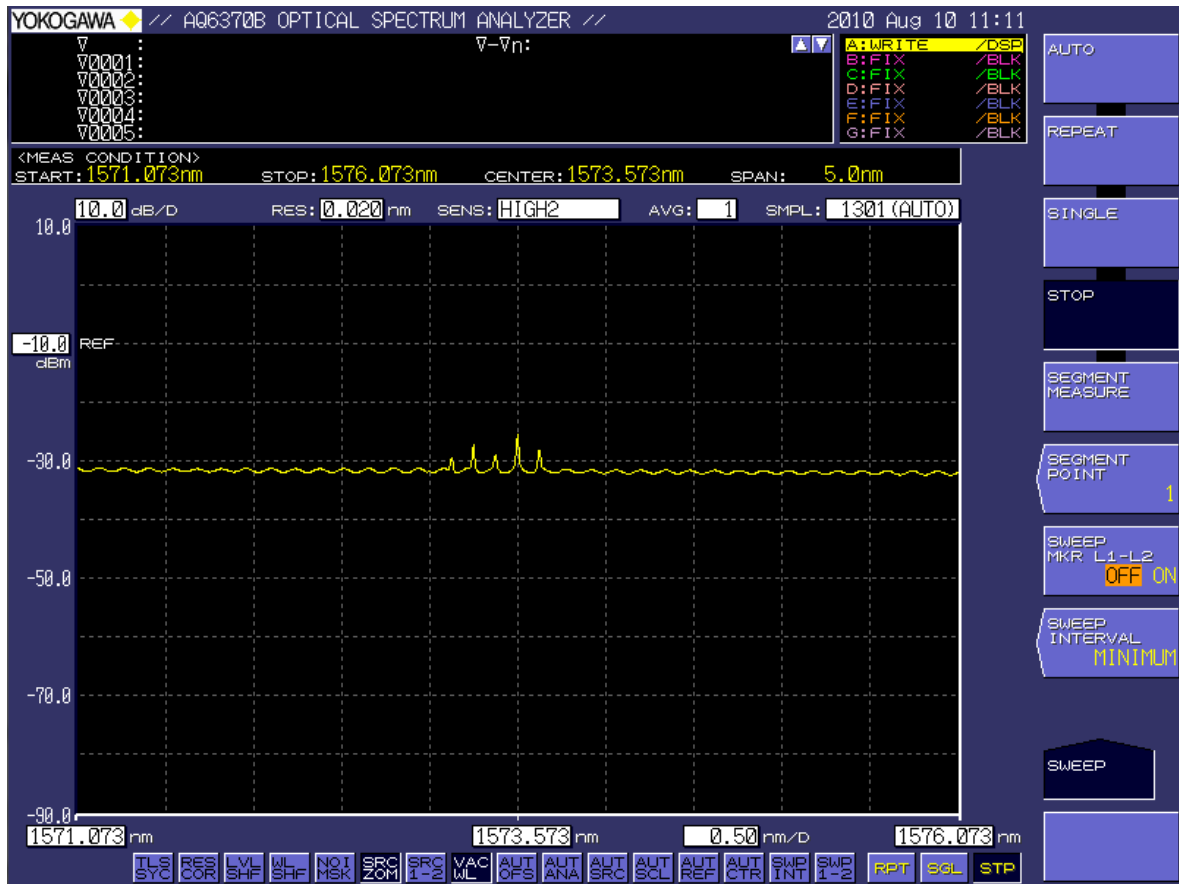


Figura 4.23. Espectro Óptico del anillo 6 (con SOA)

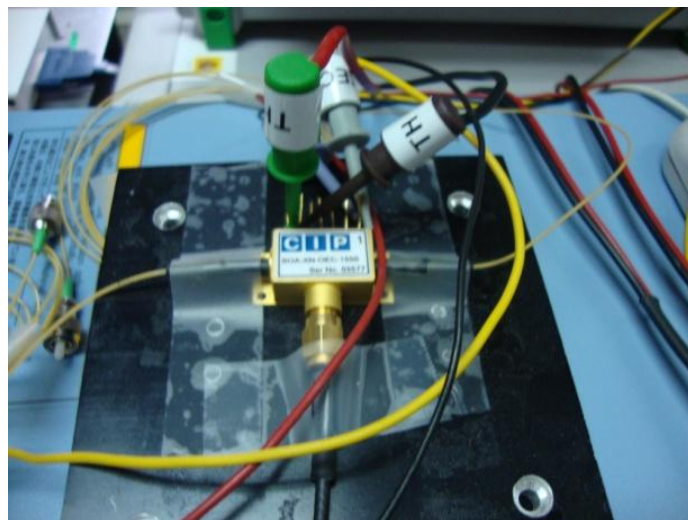


Figura 4.24. SOA-XN-OEC-1550

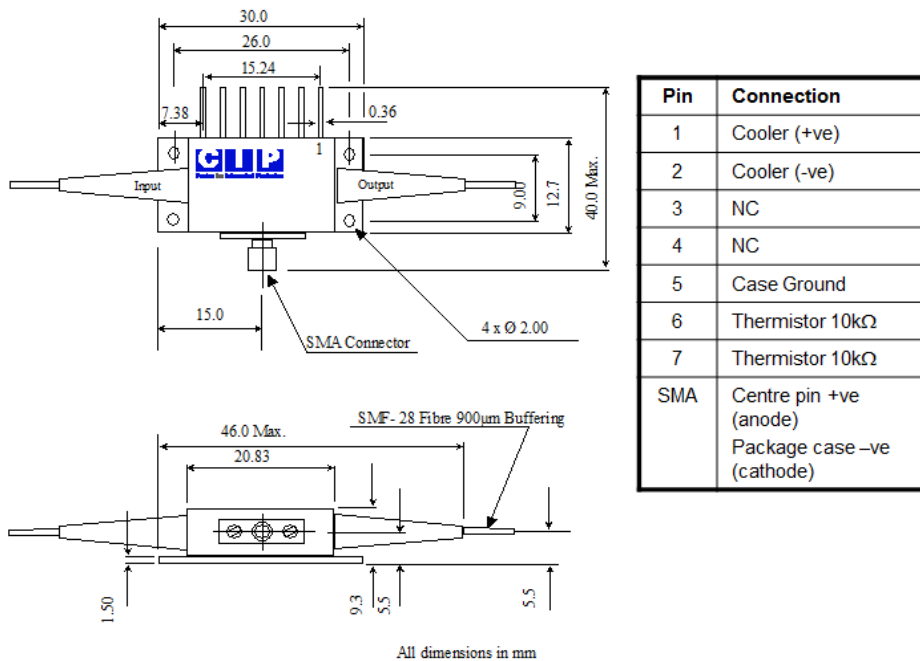


Fig. 4.25 Conexiones SOA-XN-OEC-1550

4.3 Medidas de espectro eléctrico.

En la Figura 4.26 se presenta el set – up para la realización de la medida en el espectro eléctrico. En este diseño se parte de la configuración dispuesta para la medida del espectro óptico del anillo 6 en el apartado anterior. A la salida del SOA de dicha configuración se conecta el fotodetector u2t XPDV2020R (fig.4.27) y un amplificador de RF, de ancho de banda de 40 GHz CERNEX CBL02403050, previo a la entrada del RFSA.

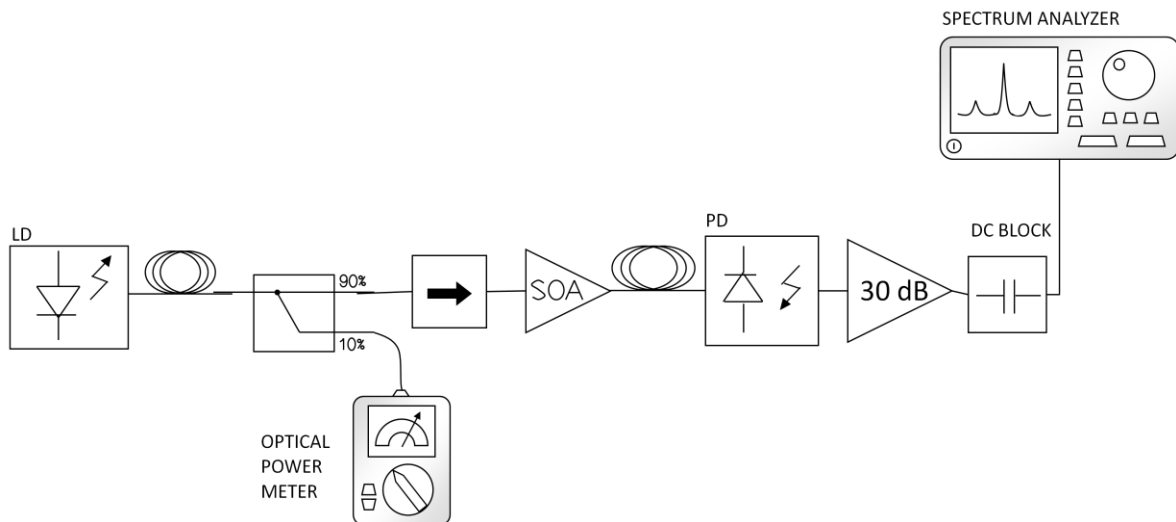


Figura 4.26. Set-Up del experimento.



Figura 4.27. u2t XPDV2020R

En el espectro eléctrico obtenido se observa un tono frecuencial a la frecuencia de 15.25 GHz, frecuencia que coincide con la serigrafiada en este dispositivo en la figura 1.12. Y que es determinada por la longitud del resonador según la tabla 4.4. A partir de este tono frecuencial y con la utilización del interfaz para la caracterización del ruido de fase se obtiene la figura 4.28.

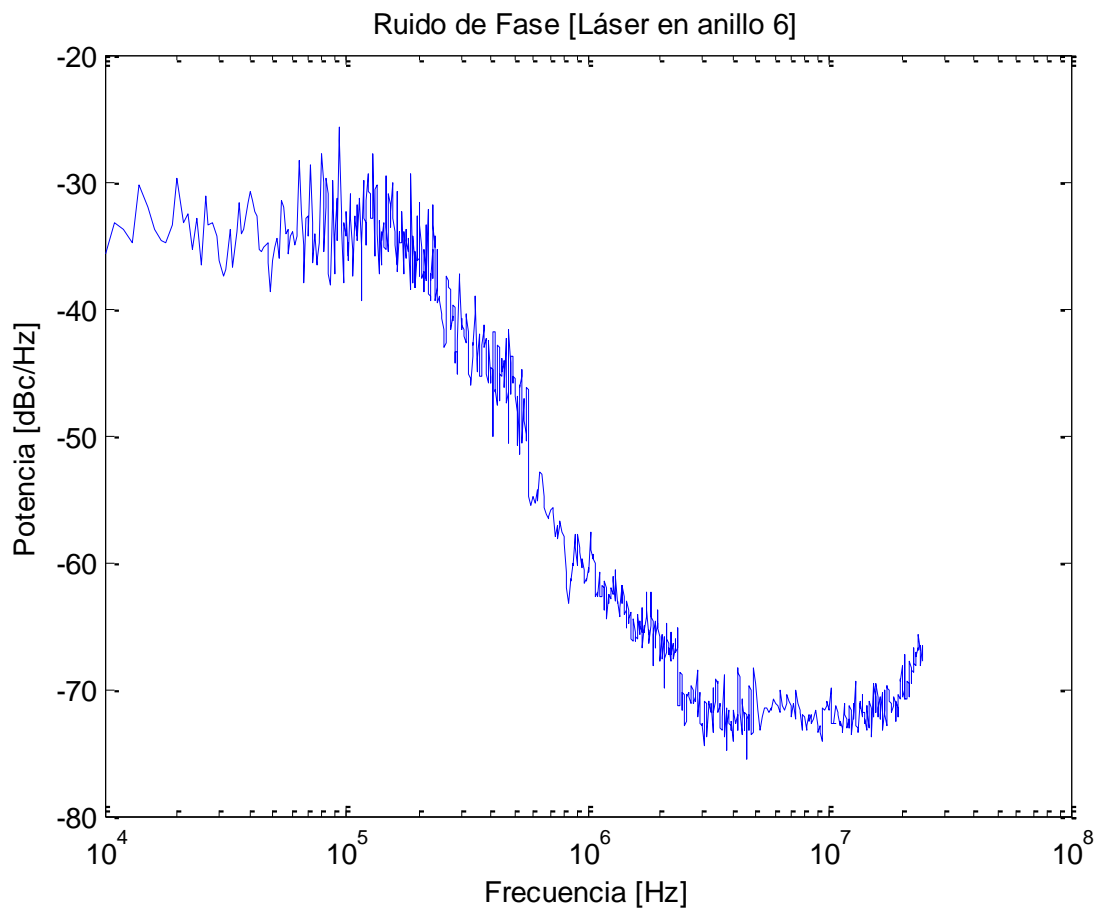


Figura 4.28. Ruido de Fase del Anillo 6 (escala logarítmica)

Para obtener el ruido de fase se realizaron varios barridos de la señal con diferentes SPAN y RBW. Las trazas que realizadas para caracterizar el ruido de fase del láser en anillo 6 se presentan en la tabla 4.3

Capítulo 4: Resultados obtenidos

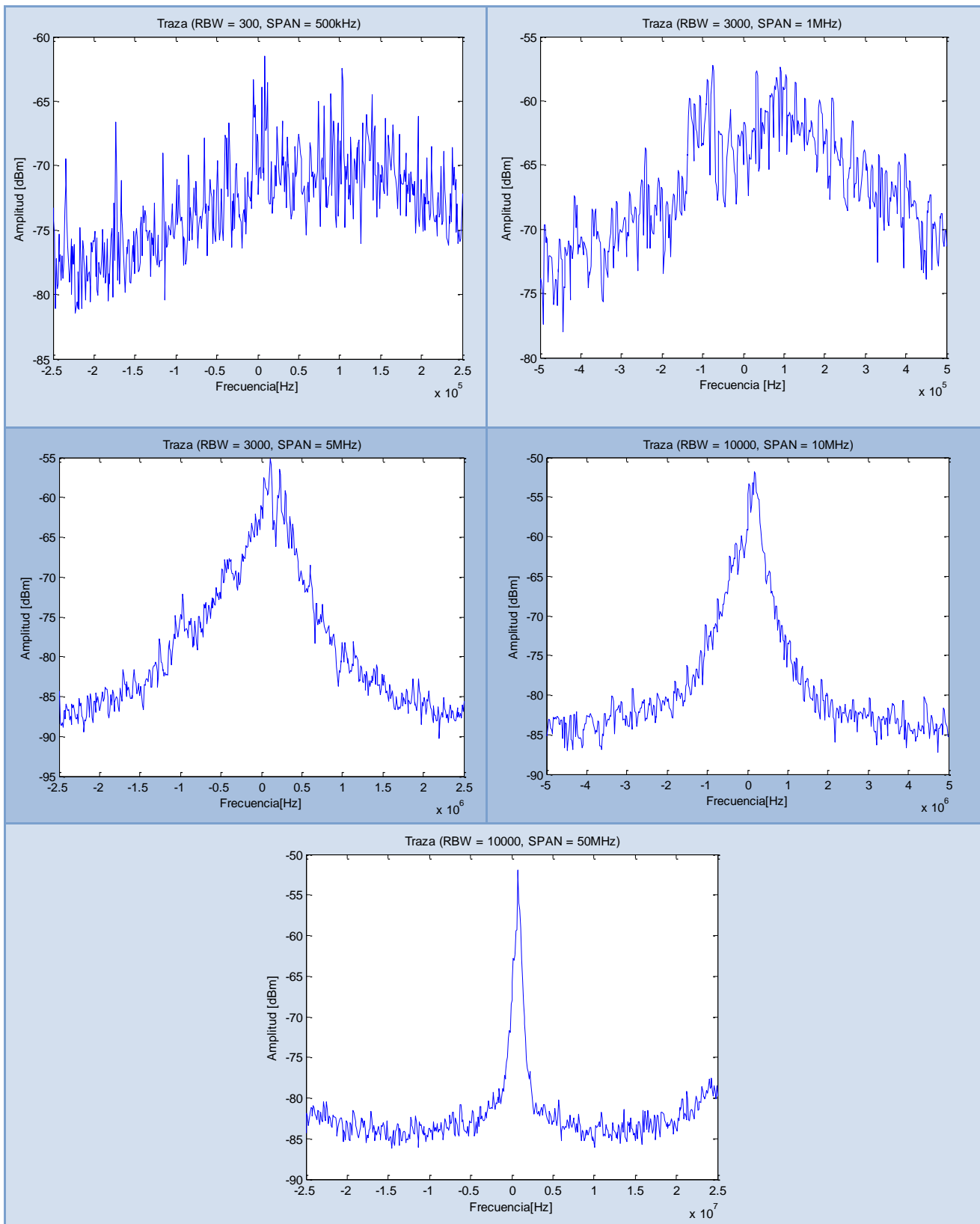


Tabla 4.6. Trazas para el análisis del ruido de fase

La medida obtenida del ruido de fase no es la esperada, ya que el ruido de fase únicamente desciende hasta -70 dBm/Hz, cuando lo normal es observar un valor en el rango de -120 dBm/Hz. Esta diferencia de valores se debe al SOA utilizado, el único disponible en el laboratorio (SOA-XN-OEC-1550). Este SOA es no lineal y sus hojas de características se encuentran disponibles en el Anexo R. Este dispositivo está optimizado para generar efectos no lineales y es adecuado para otras aplicaciones pero no para este estudio. Aunque las medidas se tuvieron que realizar con él.

Otra desventaja que presenta el SOA-XN-OEC-1550 es el ancho de banda de ganancia. Según el espectro óptico del láser en anillo 6, este se encuentra centrado en 1573.5 nm y la ganancia entorno al pico del SOA es desde 1550 nm a 1570 nm según el Anexo R.

Por ello, para el análisis del ruido de fase de este láser en anillo en un futuro se recomienda la adquisición del SOA-L-OEC-1550 por distintas razones. La primera de ellas es que este dispositivo es lineal. En segundo lugar, en el datasheet de este dispositivo (Anexo S) se pueden observar las gráficas NF(dB) vs $\text{wavl}(\text{nm})$ y $\text{Psat}(\text{dBm})$ vs $\text{wavl}(\text{nm})$ que determinan que para los valores próximos a 1573 nm se mantiene una potencia de saturación alta y una figura de ruido baja.

Capítulo 5

Conclusiones y Trabajos Futuros

El presente Proyecto Fin de Carrera ha consistido en desarrollar dos interfases con instrumentos de laboratorio bajo MATLAB, realizando con ellos el estudio las características de funcionamiento de láseres en anillo enganchados en fase, diseñados en la Universidad Carlos III de Madrid, y fabricados en la plataforma europea de InP JePPIX. El interés de estos dispositivos reside en su potencial aplicación en la que se desea concentrar la luz en instantes muy cortos de tiempo, aumentando proporcionalmente la energía y en la multiplexación óptica por división en el tiempo (OTDM).

Como paso previo a dicho trabajo, la primera parte de este proyecto presenta los resultados obtenidos utilizando diferentes técnicas de modelado, analizando de forma teórica los parámetros básicos de operación de los láseres de semiconductor. En especial, cabe destacar uno de los parámetros más importantes, como es la corriente umbral de los diodos láser. Las técnicas de modelado que se han desarrollado están basadas en la descripción del dispositivo mediante las ecuaciones de tasa, obteniéndose los parámetros que determinan la corriente umbral y observando la influencia que sobre ella tienen. A partir de las ecuaciones de tasa también se ha desarrollado la respuesta en pequeña señal y la respuesta transitoria, determinando la frecuencia de resonancia y el factor de amortiguamiento de la respuesta de un láser. En la resolución de las ecuaciones se han aplicado diferentes métodos, así como un software comercial.

De este estudio se ha determinado que los parámetros influyentes en la corriente umbral son las pérdidas α cuya variable es el volumen de la cavidad, la densidad de portadores umbral que a su vez depende de la densidad de portadores de transparencia, el factor de confinamiento óptico, el tiempo de vida de los fotones y la ganancia óptica, y por último, el tiempo de vida de los portadores.

La frecuencia de resonancia y el factor de amortiguamiento se han obtenido con la respuesta en pequeña señal y la respuesta transitoria utilizando el método numérico “forward Euler”. Mediante el análisis en pequeña señal se obtiene como resultados una expresión analítica, de la frecuencia de resonancia y el factor de amortiguamiento, observando que la frecuencia de resonancia del sistema depende directamente del coeficiente lineal de la ganancia y de la densidad de fotones mientras que es inversamente proporcional al tiempo de vida de los fotones. En los cálculos realizados se observa que al aumentar la corriente de inyección la frecuencia propia

también se ve aumentada. Los sistemas observados son subamortiguados debido al valor que toma el factor de amortiguamiento. Y por encima de la frecuencia de resonancia el sistema cae con una pendiente de -40 dB/década. Además el factor de amortiguamiento actúa de forma diferente a la frecuencia de resonancia con respecto a la densidad de fotones, disminuyendo a medida que la corriente aumenta.

En el análisis por régimen transitorio, las frecuencias obtenidas son menores que las obtenidas mediante la respuesta en pequeña señal. La razón de esta diferencia se estima que procede del error del propio método de integración *'forward Euler'*. En cuanto a los métodos numéricos, estos pueden ser mejorados mediante diferentes técnicas como son, la normalización del sistema de ecuaciones o la modificación del método numérico utilizado para su resolución.

El principal objetivo de este proyecto ha sido el desarrollo de dos interfases software para la automatización de las medidas características de estos dispositivos. El primer interfaz permite obtener la Curva P-I, medida básica en la caracterización de un láser. Y el segundo permite la representación del ruido de fase, medida importante en los dispositivos Mode-Locking. Los interfases fueron realizados en el entorno de desarrollo descrito en el capítulo 1, y con la herramienta MATLAB de Mathworks facilitando los cálculos de post procesado y presentación de resultados que se observan en el capítulo 5.

Concluidos los interfases se procedió a realizar la caracterización de los láseres en anillo en el laboratorio de investigación. La caracterización de los dispositivos se dividió en tres partes. La obtención de las curvas P-I en ambas salidas de los dispositivos en anillo, comprobando su correcto funcionamiento. La obtención del espectro óptico mediante el OSA, a través de la cual se puede obtener valiosa información de la emisión del dispositivo. Y en último lugar, el análisis del ruido de fase mediante el analizador de espectros eléctrico cuando se obtiene un tono frecuencial en el espectro eléctrico que permite identificar el régimen Mode- Locking.

Las medidas de las curva P-I fueron realizadas sobre los dos tipos de estructuras disponibles en el chip que constaba de ocho láseres: en anillos asimétricos y anillos simétricos. Para todas estas estructuras, hemos determinado mediante el interfase la corriente umbral mediante la técnica de post-procesado de la primera derivada de la función tras representarla en ejes semilogarítmicos (eje Y).

En la estructura asimétrica se han probado diversas configuraciones con la polarización de los absorbentes para determinar el funcionamiento del dispositivo. Cuando estos son polarizados por igual, observamos que el modo dominante (CW o CCW) depende del nivel de la corriente inyectada. Esta observación ha permitido identificar el hecho de que la disposición de múltiples absorbentes saturables influye notablemente en la competición de modos que se contrapropagan en el anillo. Estas medidas también permiten observar que el modo que alcanza mayor potencia es el modo dominante para corrientes mayores, esto es coherente dado el incremento de inyección de portadores.

Cuando por otro lado, aplicamos polarización a uno solo de los absorbentes obtenemos un comportamiento diferente. Polarizando el absorbente inferior, se observa cómo se favorece el modo CW del anillo desplazándose hacia las corrientes en las que era dominante el modo CCW.

Por otra parte, en las medidas en las que se polariza el absorbente superior, es el modo CW el que domina la cavidad en el entorno de corrientes próximo al umbral. Al contrario de lo que ocurriría cuando se aplicaba la polarización inversa a los absorbentes. Aumentando la tensión inversa en esta polarización el modo CW deja de experimentar la caída en potencia que presentaba para corrientes mayores. Todo ello parece indicar, que se logra compensar o aumentar el efecto que producen las asimetrías de la cavidad polarizando diferentemente los absorbentes en la cavidad.

En la estructura simétrica se obtuvo la curva P-I en ambas salidas polarizando el único absorbente saturable en OC y de 0 a 2V de tensión inversa. Las medidas realizadas muestran la evolución de los modos secuencialmente, observándose como la corriente a la que se obtiene la potencia máxima se desplaza a medida que se introduce más tensión inversa en el absorbente saturable. Llegando a tomar los mismos valores en ambas salidas a la máxima tensión inversa aplicada.

Finalmente, de la medida de la curva P-I se puede concluir que ambos láseres funcionan ya que en ellos se observa el umbral de emisión láser. También es importante destacar, que una vez superada la corriente umbral en el dispositivo ninguna de las salidas se anula. Esta propiedad parece ser característica de los láseres en anillo procesados combinando secciones activas/pasivas y contrasta con los dispositivos completamente activos presentados en el capítulo 2 donde los dispositivos funcionan de forma unidireccional.

La diferencia entre el láser en anillo asimétrico y el láser en anillo simétrico según las curvas P-I se presenta al observar siempre un modo dominante en la estructura asimétrica, pudiendo favorecerse el modo dominante mediante la aplicación de alimentaciones asimétricas de los absorbentes saturables. Por el contrario, en el láser en anillo simétrico al aumentar la tensión inversa del absorbente saturable, ambas direcciones presentan semejante potencia de salida, sugiriendo que el dispositivo entra en un modo de operación de colisión de pulsos entre los modos que se contrapropagan.

La segunda medida de caracterización de los dispositivos láser fue la obtención del espectro óptico mediante el interfaz proporcionado por el fabricante del OSA. A través de esta medida, se obtuvo mayor información del dispositivo como es el número de modos de emisión y la separación entre estos. El último dato viene determinado por la longitud del resonador óptico y proporciona la tasa de repetición del dispositivo en régimen mode - locking.

Las medidas del espectro óptico fueron realizadas sobre la misma estructura asimétrica y simétrica que las curvas P-I. Y para estos dispositivos se obtuvo la tasa de repetición de 12.73 GHz (láser en anillo asimétrico) y 15.52GHz (láser en anillo simétrico). Estas frecuencias se corresponden con las seriografiadas junto a los dispositivos (Fig. 1.12), y están determinadas en cada dispositivo por las longitudes relevantes de la cavidad. En los láseres en anillo la tasa de repetición es inversa a la longitud del resonador L, según c/nL .

De las medidas realizadas se puede apreciar la diferencia en el espectro óptico de un láser en anillo asimétrico de uno simétrico. En el láser en anillo asimétrico se observa que existen modos dominantes que tienen una diferencia de 40 dBm con la mayor parte de los modos presentes en el espectro. Mientras en el láser en anillo 8 se tiene un espectro más uniforme.

Para aumentar la potencia en ambos espectros son utilizados dos EDFAs (Erbium Doped Fiber Amplifier), con el objetivo de obtener con suficiente potencia para el RFSA. La introducción de los EDFAs aumenta la potencia pero introduce un nivel de ruido que elimina un gran parte de los modos que quedan por debajo de su potencia de ruido. Esta es la razón de que posteriormente se sustituyeran los EDFAs por el SOA disponible en laboratorio.

Finalmente, las medidas del espectro eléctrico se realizaron con el láser simétrico que mejor espectro ofrecía. Y con la utilización de el SOA disponible en laboratorio para la amplificación de la señal óptica. Sin embargo, el SOA era un dispositivo optimizado para generar efectos no lineales, y por tanto los resultados para la medida del ruido de fase no fueron los esperados, obteniéndose un elevado nivel del ruido de fase. En un futuro estudio será necesaria la adquisición de un SOA lineal para realizar un correcto análisis del ruido de fase.

El trabajo futuro de este proyecto se centra en la comprobación del régimen de funcionamiento de los láseres en anillo utilizados por medio de otros instrumentos. Primeramente debería ser comprobado tal y como se ha realizado en el presente proyecto, mediante la visualización en el espectro eléctrico de un tono a la correspondiente separación entre frecuencias visualizada en el espectro óptico.

No obstante la obtención del tono en el espectro eléctrico no determina con exactitud que el régimen de operación del dispositivo sea mode-locking. Para aumentar la información del régimen de operación de los láseres en anillo, debería obtenerse la salida de un autocorrelador. En la salida de este instrumento se podrían comprobar correctamente el tren de pulsos obtenidos y su separación temporal.

Para un correcto análisis de estos pulsos es interesante desarrollar un interfaz, que permita el post-procesado de los datos obtenidos. Permitiendo determinar la forma y anchura de los pulsos. Estos últimos parámetros característicos de un láser mode-locking también pueden ser controlados por medio de la corriente inyectada y la tensión inversa fijada sobre los absorbentes saturables.

Capítulo 6

Presupuesto

En el presupuesto se presenta un desglose de las tareas y subtareas realizadas en el presente proyecto (Tabla 6.2). Acompañando a cada tarea se hace referencia a las horas empleadas para el desarrollo de la misma.

En el presupuesto se contabiliza el coste del personal en el cual se presentan las horas dedicadas del ingeniero al cargo y del ingeniero senior o tutor responsable del desarrollo del mismo. También se presenta un listado de las herramientas hardware y software aunque no se incluyen en el coste final ya que este material se encontraba disponible previamente al inicio de este proyecto.

LISTADO DEL MATERIAL
Licencia MATLAB
Licencia VPITransmissionMaker Optical Systems
Estación de puntas Wentworth Laboratories PML 8000
Fuente de corriente ILX Lightwave LDC 3724
Lector de Potencia Óptica NewPort 842-PE
Analizador de espectros eléctrico Anritsu MS2668C
Analizador de espectros óptico Yokogawa AQ6370B
Generador de RF Anritsu MG3695A
Agilent 82357 B USB/GPIB Interface
Bus de comunicaciones GPIB
Chip de 8 láseres en anillo (fabricado en InP JePPIX)
Fuente de tensión Protek
EDFA NORTEL
Lensed Fibers
u2t XPDV2020R
SOA – XN – OEC – 1550

CERNEX CBL02403050
DC-BLOCK K261
Aisladores y acopladores
Portátil Packard Bell EasyNote

Tabla 6.1. Material del proyecto

Tareas	Definición	Horas
Tarea 1	Estudio del modelado de los láser de semiconductor	122
<i>Tarea 1.1</i>	<i>Documentación de los láseres de semiconductor</i>	20
<i>Tarea 1.2</i>	<i>Documentación de los láseres Mode-Locking y láseres en anillo</i>	30
<i>Tarea 1.3</i>	<i>Ecuaciones de tasa</i>	24
<i>Tarea 1.4</i>	<i>Modelado de los dispositivos mediante software comercial</i>	48
Tarea 2	Familiarización con el entorno de desarrollo	36
Tarea 3	Desarrollo de los interfases en Matlab	136
<i>Tarea 3.1</i>	<i>Interfaz Curva P-I</i>	64
<i>Tarea 3.2</i>	<i>Interfaz Ruido de Fase</i>	72
Tarea 4	Realización de medidas	116
<i>Tarea 4.1</i>	<i>Medida Curva P-I</i>	40
<i>Tarea 4.2</i>	<i>Medida espectro óptico</i>	48
<i>Tarea 4.3</i>	<i>Medida espectro eléctrico</i>	28
Tarea 5	Realización de la documentación de este proyecto	100
	Total	510

Tabla 6.2. Tareas del proyecto

Ingeniero (25.11 €/hora)	12806.1
Ingeniero Senior o Tutor (10% horas) (31.85 €/hora)	1624.35
IVA (18 %)	2597.5
Coste total	17027.95

Tabla 6.3. Coste del proyecto

Según la tabla 6.2 y 6.3, el presupuesto total de este proyecto asciende a la cantidad de 17027.95 euros.

Leganés a 29 de Octubre de 2010

La ingeniera proyectista

Fdo. Miriam Organista Ayala

Capítulo 7

Anexos – Extractos de código

En este capítulo se encuentran los anexos correspondientes a:

- Extractos de código del modelado de láseres de semiconductor (Anexo A - Anexo D)
- Extractos de código del desarrollo del interfase para la caracterización de la curva P-I (Anexo E - Anexo K)
- Extractos de código del desarrollo del interfase para la caracterización del ruido de fase (Anexo L - Anexo Q)
- Datasheet del SOA-XN-OEC-1550 (Anexo R)
- Datasheet del SOA-L-OEC-1550 (Anexo S)

8.1 Anexo A – Análisis del régimen estático

```

% Ridge Waveguide Laser
GammaR = 0.3;
TauPR = 1*10^(-12);
alphaR = 6.6*10^(-35);
NtR = 10^24;
TauNR = 3*10^(-9);
IaR = 0:10^(-4):100*10^(-3);
epsilonR = 6.7*10^(-23);
g0R = 3.2*10^(-12); % Corresponde al valor de 'a' de las ecuaciones
% Cálculo y dibujo del umbral
S0R = GammaR*TauPR/alphaR*(IaR - alphaR*NtR/TauNR);
for count=1:length(IaR)
if (S0R(count) < 0)
S0R(count) = 0;
end
end
subplot(2,1,1)
plot(IaR,S0R,'g')
title('Densidad de fotones vs Corriente')
ylabel('S')
xlabel('Ia [A]')
hold on

% EMBH Laser
GammaE = 0.4;
TauPE = 1*10^(-12);
alphaE = 2.2*10^(-35);
NtE = 10^24;
TauNE = 3*10^(-9);
IaE = 0:10^(-4):100*10^(-3);
epsilonE = 4.5*10^(-23);
g0E = 2.4*10^(-12); % Corresponde al valor de 'a' de las ecuaciones
% Cálculo y dibujo del umbral
S0E = GammaE*TauPE/alphaE*(IaE - alphaE*NtE/TauNE);
for count=1:length(IaE)
if (S0E(count) < 0)
S0E(count) = 0;
end
end
subplot(2,1,1)
plot(IaE,S0E,'r')
legend('Ridge Waveguide Laser', 'EMBH Laser', 'Location', 'NorthWest')
hold off

% Ridge Waveguide Laser
% Cálculo del número de fotones
N0R = NtR + 1./(g0R*GammaR*TauPR*(1-epsilonR*S0R));
subplot(2,1,2)
plot(IaR,N0R,'g')
title('Densidad de electrones vs Corriente')

```

```
ylabel('N')
xlabel('Ia [A]')
hold on

% EMBH Laser
% Cálculo del número de fotones
N0E = NtE + 1./(g0E*GammaE*TauPE*(1-epsilonE*S0E));
plot(IaR,N0E,'r')
legend('Ridge Waveguide Laser', 'EMBH Laser', 'Location', 'NorthWest')
hold off
```

8.2 Anexo B – Respuesta en pequeña señal

Los scripts de start:

StartR.m

```
% Ridge Waveguide Laser
laser='Ridge';
Gamma = 0.3;
TauP = 1*10^(-12);
alpha = 6.6*10^(-35);
Nt = 10^24;
TauN = 3*10^(-9);
Ith= alpha*Nt/TauN;
Ia = [40,50,60,80]*10^(-3);
epsilon = 6.7*10^(-23);
a = 3.2*10^(-12);
Betta =2*10^(-4);
S = Gamma*TauP/alpha*(Ia - Ith);
N = Nt + 1./(a*Gamma*TauP*(1-epsilon*S));
```

StartE.m

```
% EMBH Laser
laser='EMBH';
Gamma = 0.4;
TauP = 1*10^(-12);
alpha = 2.2*10^(-35);
Nt = 10^24;
TauN = 3*10^(-9);
Ith= alpha*Nt/TauN;
Ia = [20,30,40,50]*10^(-3);
epsilon = 4.5*10^(-23);
a= 2.4*10^(-12);
Betta =1*10^(-4);
S = Gamma*TauP/alpha*(Ia - Ith);
N = Nt + 1./(a*Gamma*TauP*(1-epsilon*S));
```

Código del cálculo en pequeña señal:

```
% Cálculo de pequeña señal
% Realizar un start previo
f=0:10^7:7*10^9;
w=2*pi*f;

S0=S(1);
N0=N(1);
C=a*S0/TauP + 1/(TauP*TauN) - (Gamma*a*(N0-Nt)*(1-Betta)/TauN);
B=a*S0+1/TauP+1/TauN-Gamma*a*(N0-Nt);
H=C./(-(w.^2)+1i*w*B+C);
```

```

H=abs(H);
H=10*log10(H);
figure(1);plot(f,H,'b')
title('Respuesta en pequeña señal')
ylabel('Amplitud(dB)')
xlabel('Frecuencia(GHz)')
hold on

S0=S(2);
N0=N(2);
C=a*S0/TauP +1/(TauP*TauN) - (Gamma*a*(N0-Nt)*(1-Betta)/TauN);
B=a*S0+1/TauP+1/TauN-Gamma*a*(N0-Nt);
H=C./(-(w.^2)+1i*w*B+C);
H=abs(H);
H=10*log10(H);
figure(1); plot(f,H,'r')

S0=S(3);
N0=N(3);
C=a*S0/TauP +1/(TauP*TauN) - (Gamma*a*(N0-Nt)*(1-Betta)/TauN);
B=a*S0+1/TauP+1/TauN-Gamma*a*(N0-Nt);
H=C./(-(w.^2)+1i*w*B+C);
H=abs(H);
H=10*log10(H);
figure(1); plot(f,H,'k')

S0=S(4);
N0=N(4);
C=a*S0/TauP +1/(TauP*TauN) - (Gamma*a*(N0-Nt)*(1-Betta)/TauN);
B=a*S0+1/TauP+1/TauN-Gamma*a*(N0-Nt);
H=C./(-(w.^2)+1i*w*B+C);
H=abs(H);
H=10*log10(H);
figure(1); plot(f,H,'g')

if strcmp(laser,'Ridge')
legend('Io=40mA','Io=50mA','Io=60mA','Io=80mA','Location','NorthWest')
else
legend('Io=20mA','Io=30mA','Io=40mA','Io=50mA','Location','NorthWest')
end
hold off

```

8.3 Anexo C – Respuesta transitoria “Forward Euler”

```
% Representación de gran señal
% Método 'forward Euler'
%  $No(t_0+t) = No(t_0) + (dN/dt)t$ 
% Se parte de  $No=0, So=0, paso= TauP/2$ 
% Realizar un start previo
No = [];
So = [];
N=0;
S=0;
avance= TauP/2;
%  $I = 20 \cdot 10^{-3}$ ; % para el EMBH
I= 60*10-3;
for i=2:16000
derN = I/alpha-N/TauN-a*(N-Nt)*(1-epsilon*S)*S;
derS = (Gamma*a*(N-Nt)*(1-epsilon*S)-1/TauP)*S + Gamma*Betta*N/TauN;
N=N+derN*avance;
S=S+derS*avance;
No = [No; N];
So = [So; S];
end

t1=avance*(1:size(No)-1);
[AX,H1,H2]=plotyy(t1,So,t1,No);
set(get(AX(1),'Ylabel'),'String','Densidad de fotones [unidades arbitrarias]')
set(get(AX(2),'Ylabel'),'String','Densidad de portadores [unidades arbitrarias]')
xlabel('Tiempo(ns)')
if strcmp(laser,'Ridge')
title('Ridge Waveguide Laser [Respuesta transitoria]')
else
title('EMBH Laser [Respuesta transitoria]')
end
hold off
```


8.4 Anexo D – Respuesta transitoria “Forward Euler normalizado”

```

% Respuesta transitoria
% Con formulas normalizadas
% Método 'forward Euler'
%  $No(t_0+t) = No(t_0) + (dN/dt)t$ 
% Partimos de  $No=0, S_0=0, \text{paso} = \tau_P / (\tau_N^2)$ 

% Normalizar%
k = Nt*a* $\tau_P$ ;
tprima =  $\tau_P / \tau_N$ ;
It =  $\alpha * Nt / \tau_N$ ;
s=0; %  $s = \tau_N * a * S$ ;  $S=0$ 
%  $\tau = 0$ ; %  $\tau = t / \tau_N$ ;  $t=0$ 
n=0; %  $n = N / Nt$ ;  $N=0$ 
e=epsilon/(a* $\tau_N$ );

% Integración Numérica%
No = [];
So = [];
avance=( $\tau_P / (\tau_N^2)$ );
% I =  $20 * 10^{-3}$ ; % para el EMBH
I=  $60 * 10^{-3}$ ;
for i=1:16000
dern = I/It-n-(n-1)*s*(1-e*s);
ders = (Gamma*k*(n-1)*(1-e*s)*s -s + Gamma*Betta*k*n)/tprima;
n=n+dern*avance;
s=s+ders*avance;
No = [No; n];
So = [So; s];
end

t1=avance*([1:size(No)]-1);
[AX,H1,H2]=plotyy(t1,So,t1,No);
set(get(AX(1),'Ylabel'),'String','Densidad de fotones [unidades arbitrarias]')
set(get(AX(2),'Ylabel'),'String','Densidad de portadores [unidades arbitrarias]')
xlabel('Tiempo')
if strcmp(laser,'Ridge')
title('Ridge Waveguide Laser - Respuesta transitoria (ecuaciones normalizadas)')
else
title('EMBH Laser - Respuesta transitoria (ecuaciones normalizadas)')
end
end
hold off

```

8.5 Anexo E – Inicialización del interfaz

```
% --- Inicialización del interfaz
function initialize_gui(fig_handle, handles)
% Valores por defecto para los datos utilizados:
handles.tempdata.fijada = '21';
handles.corrientedata.idc = '0';
handles.corrientedata.limite = '0';
handles.curvaPI.pasovalor = '0';
handles.curvaPI.limitecurvavalor = '0';
handles.curvaPI.corrientemaximavalor = '0';
% Actualización de los valores en sus manejadores correspondientes
set(handles.leida, 'String', 0);
set(handles.fijada, 'String', handles.tempdata.fijada);
set(handles.idc, 'String', handles.corrientedata.idc);
set(handles.limite, 'String', handles.corrientedata.limite);
set(handles.limitecurvavalor, 'String', handles.curvaPI.limitecurvavalor);
set(handles.pasovalor, 'String', handles.curvaPI.pasovalor);
set(handles.corrientemaximavalor, 'String', handles.curvaPI.corrientemaximavalor);
% Se etiqueta el eje de la figura que aparece en el interfaz
xlabel(handles.curvasPI, 'I[mA]');
ylabel(handles.curvasPI, 'P[W]');
title(handles.curvasPI, 'Curva P-I');
% Actualización de los manejadores del interfaz
guidata(handles.figure1, handles);
```

8.6 Anexo F – Conexión/Desconexión

```

% --- Si es presionado el botón conectar se establece la comunicación con
% --- la fuente de corriente y el lector de potencia que son los dos
% --- instrumentos con los cuales se opera en el interfaz
function pushconectar_Callback(hObject, eventdata, handles)
% hObject   manejador para el botón pushconectar
% eventdata reservado para futuras implementaciones
% handles   estructura con manejadores y datos de usuario
% Variables globales de los objetos que permiten abrir la comunicación
% En este interfaz se utilizan dos instrumentos diferentes
global fuente
global lector
% Se comprueba que el instrumento no haya sido ya creado previamente
if(isempty(fuente) == 0)
    % En caso de que el instrumento ya se encuentre conectado, se muestra un
    % mensaje de error
    if (strcmp(fuente.status, 'open') == 1)
        msgbox('Los instrumentos ya están conectados');
    else
        % En caso de que los instrumentos estén creados, solamente deberemos
        % conectarlos
        fopen(fuente);
        fopen(lector);
    end
else
    % Creación del String para la petición de conexión a la fuente
    stringdireccion = strcat('GPIB0:', get(handles.editdireccion, 'String'), '::

```

Capítulo 7: Anexos – Extractos de código

```
else
    % Cierre de la conexión
    fclose(fuente);
    fclose(lector);
end
```

8.7 Anexo G – Control de Temperatura

```

% --- Se ejecuta en la creación del objeto, después de fijar sus
% --- propiedades
function fijada_CreateFcn(hObject, eventdata, handles)
% hObject   manejador de fijada
% eventdata reservado - para futuras implementaciones
% handles   vacío
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Se ejecuta al pulsar el botón leer.
% --- Lee la temperatura del dispositivo ofrecida por la fuente
function leer_Callback(hObject, eventdata, handles)
% hObject   manejador de leer
% eventdata reservado - para futuras implementaciones
% handles   estructura con manejadores y datos de usuario
% Variable global para el manejo del instrumento
global fuente
% Se comprueba que el instrumento haya sido ya creado previamente
if isempty(fuente) == 0)
    % Se comprueba que el instrumento esté conectado
    if (strcmp(fuente.status, 'open') == 1)
        % Se pregunta por la temperatura al instrumento
        fprintf(fuente,'TEC:T?');
        % Se realiza la lectura del valor
        td = fscanf(fuente);
        % El manejador es actualizado para que sea visualizado por el
        % usuario
        set(handles.leida, 'String', td);
    % Si no está conectado se da un mensaje de error
    else
        msgbox('Los instrumentos no están conectados');
    end
% El mismo mensaje de error aparece si el instrumento no ha sido creado
else
    msgbox('Los instrumentos no están conectados');
end

% --- Se ejecuta al pulsar el botón fijar
% --- Fija el valor de la temperatura introducido
function fijar_Callback(hObject, eventdata, handles)
% hObject   manejador de fijar
% eventdata reservado - para futuras implementaciones
% handles   estructura con manejadores y datos de usuario
% Variable global para el manejo del instrumento
global fuente
% Se comprueba que el instrumento haya sido ya creado previamente
if isempty(fuente) == 0)
    % Se comprueba que el instrumento esté conectado
    if (strcmp(fuente.status, 'open') == 1)

```

Capítulo 7: Anexos – Extractos de código

```
% Se crea un string a partir del valor de temperatura especificado
% por el usuario
datos = ['TEC:T ' handles.tempdata.fijada];
% Se envía la operación al instrumento
fprintf(fuente, datos);
% Si no está conectado se da un mensaje de error
else
    msgbox('Los instrumentos no están conectados');
end
% El mismo mensaje de error aparece si el instrumento no ha sido creado
else
    msgbox('Los instrumentos no están conectados');
end

% --- Se ejecuta al pulsar el botón go
% --- Cambia el estado del control de temperatura entre encendido/apagado
function go_Callback(hObject, eventdata, handles)
% hObject   manejador de go
% eventdata reservado - para futuras implementaciones
% handles   estructura con manejadores y datos de usuario
% Variable global para el manejo del instrumento
global fuente
% Se comprueba que el instrumento haya sido ya creado previamente
if(isempty(fuente) == 0)
    % Se comprueba que el instrumento esté conectado
    if(strcmp(fuente.status, 'open') == 1)
        % Lectura de el botón de estado(ON/OFF)
        val = get(hObject, 'Value');
        switch val
            % Si el botón no está presionado, el control de la temperatura
            % se encuentra deshabilitado
            case 0,
                datos = ['TEC:OUT ' '0'];
            % Si el botón está presionado, el control de la temperatura
            % se encuentra habilitado
            case 1,
                datos = ['TEC:OUT ' '1'];
        end
        % Se envía el comando al instrumento
        fprintf(fuente, datos);
        % Si no está conectado se da un mensaje de error
    else
        msgbox('Los instrumentos no están conectados');
    end
    % El mismo mensaje de error aparece si el instrumento no ha sido creado
else
    msgbox('Los instrumentos no están conectados');
end
```

8.8 Anexo H – Panel de corriente

```

% --- Actualización del valor introducido en el campo idc
function idc_Callback(hObject, eventdata, handles)
% hObject   manejador de idc
% eventdata reservado - para futuras implementaciones
% handles   estructura con manejadores y datos de usuario
current = str2double(get(hObject, 'String'));
% Save the new current value
handles.corrientedata.idc = current;
% Actualización de los manejadores de la figura
guidata(hObject,handles)

% --- Se ejecuta en la creación del objeto, después de fijar todas sus
% --- propiedades
function idc_CreateFcn(hObject, eventdata, handles)
% hObject   manejador de idc
% eventdata reservado - para futuras implementaciones
% handles   vacio
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Se ejecuta al pulsar el botón fijar_idc
% --- Fija el valor introducido de corriente
function fijar_idc_Callback(hObject, eventdata, handles)
% hObject   manejador de fijar_idc
% eventdata reservado - para futuras implementaciones
% handles   estructura con manejadores y datos de usuario
% Variable global para el manejo del instrumento
global fuente
% Se comprueba que el instrumento haya sido ya creado previamente
if(isempty(fuente) == 0)
    % Se comprueba que el instrumento esté conectado
    if (strcmp(fuente.status, 'open') == 1)
        % Consulta si está encendida la salida de la corriente en la fuente
        datos = 'LAS:OUT?';
        fprintf(fuente, datos);
        % Realiza la lectura de la información proporcionada por la fuente
        encendido = str2double(fscanf(fuente));
        % Se almacena el valor que el usuario desea fijar
        final = handles.corrientedata.idc;
        % Si está cortada la salida de corriente se abre
        if (encendido == 0)
            datos = 'LAS:OUT 1';
            fprintf(fuente, datos);
        end
        % Se envía el valor de corriente
        datos = ['LAS:LDI ' num2str(final)];
        fprintf(fuente, datos);
        % Si se ha fijado la corriente a un valor final de cero se cierra la

```

```
% salida
if (final == 0)
    datos = 'LAS:OUT 0';
    fprintf(fuente, datos);
end
% Si no está conectado se da un mensaje de error
else
    msgbox('Los instrumentos no están conectados');
end
% El mismo mensaje de error aparece si el instrumento no ha sido creado
else
    msgbox('Los instrumentos no están conectados');
end

% --- Se ejecuta al pulsar el botón limitar
% --- Limita la corriente al valor introducido por el usuario.
function limitar_Callback(hObject, eventdata, handles)
% hObject   manejador de limitar
% eventdata reservado - para futuras implementaciones
% handles   estructura con manejadores y datos de usuario
% Variable global para el manejo del instrumento
global fuente
% Se comprueba que el instrumento haya sido ya creado previamente
if(isempty(fuente) == 0)
    % Se comprueba que el instrumento esté conectado
    if (strcmp(fuente.status, 'open') == 1)
        % Se fija un valor límite para la corriente en la fuente
        datos = ['LAS:LIM:I2 ' handles.corrientedata.limite];
        fprintf(fuente, datos);
    % Si no está conectado se da un mensaje de error
    else
        msgbox('Los instrumentos no están conectados');
    end
% El mismo mensaje de error aparece si el instrumento no ha sido creado
else
    msgbox('Los instrumentos no están conectados');
end

% --- Se ejecuta cuando se introduce un nuevo valor de corriente limite en
% --- el campo correspondiente
function limite_Callback(hObject, eventdata, handles)
% hObject   manejador de la figura
% eventdata reservado - para futuras implementaciones
% handles   estructura con manejadores y datos de usuario
% Se obtiene el valor marcado como límite de corriente
limit_idc = get(hObject, 'String');
% Se guarda el nuevo valor de corriente en su manejador correspondiente
handles.corrientedata.limite = limit_idc;
% Se actualizan los manejadores
guidata(hObject,handles)

% --- Se ejecuta para la creación del campo que permite introducir el
% --- limite de corriente
```



```
function limite_CreateFcn(hObject, eventdata, handles)
% hObject  manejador de la figura
% eventdata reservado - para futuras implementaciones
% handles  vacio
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end
```

8.9 Anexo I – Lectura de potencia

```
% --- Se ejecuta al pulsar el botón lectorPotencia
% --- Lee la potencia captada por el lector y la muestra
function lectorPotencia_Callback(hObject, eventdata, handles)
% hObject   manejador de lectorPotencia
% eventdata reservado - para futuras implementaciones
% handles   estructura con manejadores y datos de usuario
% Variable global para el manejo del instrumento
global lector
% Se comprueba que el instrumento haya sido ya creado previamente
if isempty(lector) == 0
    % Se comprueba que el instrumento esté conectado
    if (strcmp(lector.status, 'open') == 1)
        % Se pregunta al lector de potencia por el valor leído
        fprintf(lector, '*CVU');
        td = fscanf(lector);
        % Se convierte el valor leído para mostrarlo correctamente
        potencia=td(17:end-2);
        potencia_numero= str2double(potencia);
        % Se hace visible este valor para el usuario
        set(handles.potenciaLeida, 'String', potencia);
    % Si no está conectado se da un mensaje de error
    else
        msgbox('Los instrumentos no están conectados');
    end
% El mismo mensaje de error aparece si el instrumento no ha sido creado
else
    msgbox('Los instrumentos no están conectados');
end
```

8.10 Anexo J – Representación de la Curva P-I

```
% --- Se ejecuta cuando se introduce un valor de corriente máxima para una
% --- la realización de una posterior curva PI
function corrientemaximavalor_Callback(hObject, eventdata, handles)
% hObject   manejador de la figura
% eventdata reservado - para futuras implementaciones
% handles   estructura con manejadores y datos de usuario
% Se toma el valor introducido por el usuario
corriente = str2double(get(hObject, 'String'));
% Se guarda el nuevo valor de corriente
handles.curvaPI.corrientemaximavalor = corriente;
% Se actualizan los manejadores
guidata(hObject,handles)
```

```
% --- Se ejecuta durante la creación del campo que permite introducir el
% --- valor de la corriente máxima a la que se llegara cuando se pinte la
% --- curva PI
function corrientemaximavalor_CreateFcn(hObject, eventdata, handles)
% hObject   manejador de la figura
% eventdata reservado - para futuras implementaciones
% handles   vacio
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end
```

```
% --- Se ejecuta al asignar el valor de paso entre corrientes sucesivas, el
% --- cual determinara los puntos correspondientes a las curvas PI
function pasovalor_Callback(hObject, eventdata, handles)
% hObject   manejador de la figura
% eventdata reservado - para futuras implementaciones
% handles   estructura con manejadores y datos de usuario
% Lectura del nuevo valor que indica el salto entre corriente para
% realizar una curva PI
paso = str2double(get(hObject, 'String'));
% Se guarda el valor obtenido en su manejador
handles.curvaPI.pasovalor = paso;
% Se actualizan los manejadores correspondientes
guidata(hObject,handles)
```

```
% --- Es ejecutado en la creación del campo paso que identifica la
% --- distancia de una corriente a su sucesiva, definiendo los puntos que
% --- permiten obtener la curva PI
function pasovalor_CreateFcn(hObject, eventdata, handles)
% hObject   manejador de la figura
% eventdata reservado - para futuras implementaciones
% handles   vacio
if ispc
    set(hObject,'BackgroundColor','white');
else
```

```

    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end

% --- Actualiza el límite de corriente de la fuente para posteriormente
% --- realizar una curva P-I
function limitecurvavalor_Callback(hObject, eventdata, handles)
% hObject   manejador de la figura
% eventdata reservado - para futuras implementaciones
% handles   estructura con manejadores y datos de usuario
limitecurva = get(hObject, 'String');
% Almacena el valor leído de la corriente en su correspondiente manejador
handles.curvaPI.limitecurvavalor = limitecurva;
% Actualiza los manejadores
guidata(hObject,handles)

% --- Se ejecuta en la creación del campo limitecurvavalor
function limitecurvavalor_CreateFcn(hObject, eventdata, handles)
% hObject   manejador de la figura
% eventdata reservado - para futuras implementaciones
% handles   vacio
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end

% --- Se ejecuta al presionar el botón Pintar permitiendo la realización
% --- de la curva PI correspondiente
function pintar_Callback(hObject, eventdata, handles)
% hObject   manejador de la figura
% eventdata reservado - para futuras implementaciones
% handles   estructura con manejadores y datos de usuario
% Variables globales para el manejo de los instrumentos
global lector
global fuente
% Variables globales para posteriormente guardar la curva P-I
global potencia_curvaPI
global corriente_curvaPI

% Se comprueba que el instrumento haya sido ya creado previamente
if isempty(lector) == 0
    % Se comprueba que el instrumento esté conectado
    if (strcmp(lector.status, 'open') == 1)
        % Comprobación de los parámetros introducidos
        % Se comprueba que el valor máximo de corriente sea mayor que 0
        if (str2double(get(handles.corrientemaximavalor, 'String')) <= 0)
            msgbox('La corriente máxima fijada debe ser mayor que 0');
        % Se comprueba que el valor del límite máximo de corriente sea
        % mayor que la corriente maxima
        elseif(str2double(get(handles.limitecurvavalor,'String'))<
str2double(get(handles.corrientemaximavalor, 'String')))
            msgbox('El límite de corriente máxima fijada debe ser mayor que la corriente máxima');
        % Se comprueba que el paso de las iteraciones sea mayor que 0

```

```

elseif (str2double(get(handles.pasovalor, 'String')) <= 0)
    msgbox('El paso fijado debe ser mayor que 0');
else

    %Se declaran los arrays para guardar los valores correspondientes a
    %la curva P-I
    corrientes_valores= [];
    potencias_valores = [];
    % Se fija el valor limite de corriente en la fuente antes de
    % comenzar a asignar valores
    datos = ['LAS:LIM:I2 ' handles.curvaPI.limitecurvavalor];
    fprintf(fuente, datos);
    % Se abre la salida de la corriente
    datos = ['LAS:OUT ' '1'];
    fprintf(fuente, datos);
    % Se lee el valor fijado por el usuario para realizar los incrementos
    % de corriente de forma que se caracterice la curva de
    % Potencia-Corriente y el máximo valor al que se debe llegar en la
    % curva P-I
    paso_bucle = handles.curvaPI.pasovalor;
    máximo= handles.curvaPI.corrientemaximavalor;
    % Recorremos el valor de las corrientes
    for corriente = 0:paso_bucle:máximo
        % Se asigna cada valor de corriente
        datos = ['LAS:LDI ' num2str(corriente)];
        fprintf(fuente, datos);
        % Se espera un tiempo para que la corriente se estabilice y
        % poder realizar la lectura de la potencia correctamente
        pause(1.5);
        % Se solicita el valor de potencia al lector
        fprintf(lector, '*CVU');
        td = fscanf(lector);
        % Se manipula el string obtenido porque se devuelve en el
        % siguiente formato 'Current Value: -7.4198e-011' del cual a
        % se quiere obtener el valor numérico
        potencia=td(17:end-2);
        potencia_numero= str2double(potencia);
        % Los valores obtenidos se almacenan para la posterior
        % representación de la curva
        corrientes_valores = [corrientes_valores corriente];
        potencias_valores = [potencias_valores potencia_numero];
    end
    % Se representa la curva
    axes(handles.curvasPI);
    plot(corrientes_valores,potencias_valores);
    % Se etiquetan los ejes de la figura que aparece en el interfaz
    % y se determina el titulo de la figura
    xlabel(handles.curvasPI, 'I[mA]');
    ylabel(handles.curvasPI, 'P[W]');
    title(handles.curvasPI,'Curva P-I');
    % Se guardan los valores en las variables globales para su
    % posterior utilización
    potencia_curvaPI = potencias_valores;

```

Capítulo 7: Anexos – Extractos de código

```
    corriente_curvaPI = corrientes_valores;
    % Se fija la corriente a 0
    datos = ['LAS:LDI ' num2str(0)];
    fprintf(fuente, datos);
    pause(3);
    % Se corta la salida de la señal
    datos = ['LAS:OUT ' '0'];
    fprintf(fuente, datos);
end
% Si no está conectado se da un mensaje de error
else
    msgbox('Los instrumentos no están conectados');
end
% El mismo mensaje de error aparece si el instrumento no ha sido creado
else
    msgbox('Los instrumentos no están conectados');
end

% --- Se ejecuta al presionar el botón limpiar, y borra la figura
% --- presentada
function limpiar_Callback(hObject, eventdata, handles)
% hObject   manejador de la figura
% eventdata reservado - para futuras implementaciones
% handles   estructura con manejadores y datos de usuario
axes(handles.curvasPI);
cla;

% --- Se ejecuta cuando se presiona el botón guardar curva, permitiendo
% --- guardar las dos variables correspondientes a los ejes de la figura en
% --- un archivo .mat
function guardarcurva_Callback(hObject, eventdata, handles)
% hObject   manejador de la figura
% eventdata reservado - para futuras implementaciones
% handles   estructura con manejadores y datos de usuario
% Variables globales usadas en la función pushpintar para la
% caracterización del ruido de fase
global potencia_curvaPI
global corriente_curvaPI
% Llamada a una ventana para la elección de la ubicación y el nombre del
% fichero donde se guardarán los datos correspondientes
[filename, pathname] = uiputfile('*.mat', 'Save Workspace as');
% Se controla que el usuario haya seleccionado un archivo
if filename ~= 0
    % Se guardan los datos de la figura
    save(filename, 'potencia_curvaPI', 'corriente_curvaPI');
end
```

8.11 Anexo K – Cierre ordenado

```

% --- Se ejecuta cuando el usuario presiona el botón correspondiente al
% --- cierre del interfaz
function figure1_CloseRequestFcn(hObject, eventdata, handles)
% hObject   manejador de la figura
% eventdata reservado - para futuras implementaciones
% handles   estructura con manejadores y datos de usuario
% Se crea un mensaje de dialogo para el usuario, permitiéndole decidir si
% desea o no cerrar el interfaz
opc = questdlg('¿Desea cerrar el interfaz?','SALIR','Si','No','No');
% Se comprueba la opción elegida por el usuario
if strcmp(opc,'No')
    return;
end
global lector
global fuente
% Se comprueba que los instrumentos hayan sido creados previamente
if(isempty(fuente) == 0)
    % Cierre de la conexión
    fclose(fuente);
    fclose(lector);
    clear fuente;
    clear lector;
end
% Cierra el interfaz
delete(hObject);

```

8.12 Anexo L – Control de parámetros (RFSA)

```
% --- Lee el valor introducido en el campo de frecuencia central
function editfc_Callback(hObject, eventdata, handles)
% hObject   manejador de editfc
% eventdata reservado - para futuras implementaciones
% handles   estructura con manejadores y datos de usuario
fc = get(hObject, 'String');
% Guarda el valor de la nueva frecuencia central
handles.uipanelparametros.editfc = fc;
% Actualiza el manejador
guidata(hObject,handles)

% --- Se ejecuta en la creación del campo dinámico de la frecuencia
% --- central, definiendo sus propiedades
function editfc_CreateFcn(hObject, eventdata, handles)
% hObject   manejador de editfc
% eventdata reservado - para futuras implementaciones
% handles   vacio
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end

% --- Lee el valor introducido en el campo de RBW
function editrbw_Callback(hObject, eventdata, handles)
% hObject   manejador de editrbw
% eventdata reservado - para futuras implementaciones
% handles   estructura con manejadores y datos de usuario
rbw = get(hObject, 'String');
% Guarda el valor del nuevo RBW
handles.uipanelparametros.editrbw = rbw;
% Actualiza el valor del manejador
guidata(hObject,handles)

% --- Se ejecuta en la creación del campo dinámico del RBW
% --- definiendo sus propiedades
function editrbw_CreateFcn(hObject, eventdata, handles)
% hObject   manejador de editrbw
% eventdata reservado - para futuras implementaciones
% handles   vacio
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end

% --- Lee el valor introducido en el campo de Span
function editspan_Callback(hObject, eventdata, handles)
% hObject   manejador de editspan
% eventdata reservado - para futuras implementaciones
```



```

% handles estructura con manejadores y datos de usuario
span = get(hObject, 'String');
% Guarda el nuevo valor de Span
handles.uipanelparametros.editspan = span;
% Actualiza el valor del manejador
guidata(hObject,handles)

% --- Se ejecuta en la creación del campo dinámico del Span
% --- definiendo sus propiedades
function editspan_CreateFcn(hObject, eventdata, handles)
% hObject manejador de editspan
% eventdata reservado - para futuras implementaciones
% handles vacio
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end

% --- Se ejecuta cuando el botón pushset es pulsado
% --- El valor de frecuencia central es enviado
function pushset_Callback(hObject, eventdata, handles)
% hObject manejador de pushset
% eventdata reservado - para futuras implementaciones
% handles estructura con manejadores y datos de usuario
% Variable global que permite manejar el RFSA
% La comunicación será mediante VISA y se puede crear el instrumento,
% conectar o desconectar
global vi
% Se comprueba que el instrumento haya sido ya creado previamente
if(isempty(vi) == 0)
    % Se comprueba que el instrumento esté conectado
    if (strcmp(vi.status, 'open') == 1)
        % Se envía el comando correspondiente para asignar la frecuencia central
        stringfc = strcat('CF ', handles.uipanelparametros.editfc, 'GHZ');
        fprintf(vi,stringfc);
        % Lectura del tiempo en el que se realizara el barrido
        fprintf(vi, 'ST?');
        tiempo= str2double(fscanf(vi));
        %Solicitud para la realización del barrido en el RFSA
        fprintf(vi, 'TS');
        %Espera para la toma de la traza
        pause(tiempo/10^6);
    % Si no está conectado se da un mensaje de error
    else
        msgbox('El instrumento no está conectado');
    end
% El mismo mensaje de error aparece si el instrumento no ha sido creado
else
    msgbox('El instrumento no está conectado');
end

% --- Se ejecuta cuando el botón pushsetRBW es pulsado

```

Capítulo 7: Anexos – Extractos de código

```
% % --- El valor de RBW es enviado
function pushsetRBW_Callback(hObject, eventdata, handles)
% hObject   manejador de pushset
% eventdata reservado - para futuras implementaciones
% handles   estructura con manejadores y datos de usuario
% Variable global que permite manejar el RFSA
% La comunicación será mediante VISA y se puede crear el instrumento,
% conectar o desconectar
global vi
% Se comprueba que el instrumento haya sido ya creado previamente
if(isempty(vi) == 0)
    % Se comprueba que el instrumento esté conectado
    if (strcmp(vi.status, 'open') == 1)
        % Se envía el comando correspondiente para asignar el RBW
        stringrbw = strcat('RB ', handles.uipanelparametros.editrbw);
        fprintf(vi,stringrbw);
        %Lectura del tiempo en el que se realizara el barrido
        fprintf(vi, 'ST?');
        tiempo= str2double(fscanf(vi));
        %Solicitud para la realización del barrido en el RFSA
        fprintf(vi, 'TS');
        %Espera para la toma de la traza
        pause(tiempo/10^6);
    % Si no está conectado se da un mensaje de error
    else
        msgbox('El instrumento no está conectado');
    end
% El mismo mensaje de error aparece si el instrumento no ha sido creado
else
    msgbox('El instrumento no está conectado');
end

% --- Se ejecuta cuando el botón pushsetSPAN es pulsado
% --- El valor de SPAN es enviado
function pushsetSPAN_Callback(hObject, eventdata, handles)
% hObject   manejador de pushset
% eventdata reservado - para futuras implementaciones
% handles   estructura con manejadores y datos de usuario
% Variable global que permite manejar el RFSA
% La comunicación será mediante VISA y se puede crear el instrumento,
% conectar o desconectar
global vi
% Se comprueba que el instrumento haya sido ya creado previamente
if(isempty(vi) == 0)
    % Se comprueba que el instrumento esté conectado
    if (strcmp(vi.status, 'open') == 1)
        % Se envía el comando correspondiente para asignar el span
        stringspan = strcat('SP ', handles.uipanelparametros.editspan);
        fprintf(vi,stringspan);
        %Lectura del tiempo en el que se realizara el barrido
        fprintf(vi, 'ST?');
        tiempo= str2double(fscanf(vi));
        %Solicitud para la realización del barrido en el RFSA
```

```

    fprintf(vi, 'TS');
    %Espera para la toma de la traza
    pause(tiempo/10^6);
    % Si no está conectado se da un mensaje de error
    else
        msgbox('El instrumento no está conectado');
    end
    % El mismo mensaje de error aparece si el instrumento no ha sido creado
    else
        msgbox('El instrumento no está conectado');
    end
end

```

```

% Se ejecuta al presionar el botón PFC
function CentrarFrecuencia_Callback(hObject, eventdata, handles)
% hObject   manejador de centrar_frecuencia
% eventdata reservado - para futuras implementaciones
% handles   estructura con manejadores y datos de usuario
% Variable global que permite manejar el RFSA
% La comunicación será mediante VISA y se puede crear el instrumento,
% conectar o desconectar
global vi
% Se comprueba que el instrumento haya sido ya creado previamente
if isempty(vi) == 0)
    % Se comprueba que el instrumento esté conectado
    if (strcmp(vi.status, 'open') == 1)
        % Se envía el comando para fijar la frecuencia central
        stringpcf = strcat('PCF');
        fprintf(vi,stringpcf);
        %Lectura del tiempo en el que se realizara el barrido
        fprintf(vi, 'ST?');
        tiempo= str2double(fscanf(vi));
        %Solicitud para la realización del barrido en el RFSA
        fprintf(vi, 'TS');
        %Espera para la toma de la traza
        pause(tiempo/10^6);
        %Se solicita al RFSA la frecuencia central asignada
        fprintf(vi, 'CF?');
        frecuencia_central= str2double(fscanf(vi));
        set(handles.pcfdevuelto,'ForegroundColor','r');
        set(handles.pcfdevuelto, 'String', frecuencia_central);
    % Si no está conectado se da un mensaje de error
    else
        msgbox('El instrumento no está conectado');
    end
    % El mismo mensaje de error aparece si el instrumento no ha sido creado
    else
        msgbox('El instrumento no está conectado');
    end
end

```

```

function pcfdevuelto_Callback(hObject, eventdata, handles)
% hObject   manejador de pcfdevuelto
% eventdata reservado - para futuras implementaciones
% handles   estructura con manejadores y datos de usuario

```

Capítulo 7: Anexos – Extractos de código

```
pcf = get(hObject, 'String');
% Guarda el nuevo valor de PCF
handles.uipanelparametros.pcfdevuelto = pfc;
% Actualiza el valor del manejador
guidata(hObject,handles)

% --- Se ejecuta durante la creación y asigna las propiedades
function pcfdevuelto_CreateFcn(hObject, eventdata, handles)
% hObject   manejador de pcfdevuelto
% eventdata reservado - para futuras implementaciones
% handles   vacio
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end

% --- Se ejecuta cuando se presiona el botón de amplitud central.
function amplitudcentral_Callback(hObject, eventdata, handles)
% hObject   manejador de amplitudcentral
% eventdata reservado - para futuras implementaciones
% handles   estructura con manejadores y datos de usuario
% Variable global que permite manejar el RFSA
% La comunicación será mediante VISA y se puede crear el instrumento,
% conectar o desconectar
global vi
% Se comprueba que el instrumento haya sido ya creado previamente
if(isempty(vi) == 0)
    % Se comprueba que el instrumento esté conectado
    if (strcmp(vi.status, 'open') == 1)
        % Se envía el comando para el marker en la frecuencia central
        stringpcf = strcat('MKCF');
        fprintf(vi,stringpcf);
        %Se solicita al RFSA el valor en amplitud del marker
        fprintf(vi, 'MKA?');
        amplitud= str2double(fscanf(vi));
        set(handles.valoramplitud,'ForegroundColor','r');
        set(handles.valoramplitud, 'String', amplitud);
        % Si no está conectado se da un mensaje de error
    else
        msgbox('El instrumento no está conectado');
    end
% El mismo mensaje de error aparece si el instrumento no ha sido creado
else
    msgbox('El instrumento no está conectado');
end

function valoramplitud_Callback(hObject, eventdata, handles)
% hObject   manejador de valoramplitud
% eventdata reservado - para futuras implementaciones
% handles   estructura con manejadores y datos de usuario
amplitud = get(hObject, 'String');
```

```

% Guarda el nuevo valor de amplitud
handles.uipanelparametros.valoramplitud = amplitud;
% Actualiza el valor del manejador
guidata(hObject,handles)

% --- Se ejecuta durante la creación, asignando propiedades al campo
% --- valor amplitud
function valoramplitud_CreateFcn(hObject, eventdata, handles)
% hObject   manejador de valoramplitud
% eventdata reservado - para futuras implementaciones
% handles   vacio
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end

% --- Se ejecuta al pulsar PRL
function peakreferencelevel_Callback(hObject, eventdata, handles)
% hObject   manejador de peakreferencelevel
% eventdata reservado - para futuras implementaciones
% handles   estructura con manejadores y datos de usuario
% Variable global que permite manejar el RFSA
% La comunicación será mediante VISA y se puede crear el instrumento,
% conectar o desconectar
global vi
% Se comprueba que el instrumento haya sido ya creado previamente
if isempty(vi) == 0
    % Se comprueba que el instrumento esté conectado
    if (strcmp(vi.status, 'open') == 1)
        % Se envía el comando para fijar el nivel de referencia
        stringpcf = strcat('PRL');
        fprintf(vi,stringpcf);
        % Lectura del tiempo en el que se realizara el barrido
        fprintf(vi, 'ST?');
        tiempo= str2double(fscanf(vi));
        %Solicitud para la realización del barrido en el RFSA
        fprintf(vi, 'TS');
        %Espera para la toma de la traza
        pause(tiempo/10^6);
        % Si no está conectado se da un mensaje de error
    else
        msgbox('El instrumento no está conectado');
    end
end
% El mismo mensaje de error aparece si el instrumento no ha sido creado
else
    msgbox('El instrumento no está conectado');
end
end

```

8.13 Anexo M – Mostrar Traza

```

% --- Se ejecuta cuando es pulsado el botón pushmostrar
% --- La traza que se encuentre pintada en el instrumento
% --- es representada en el interfaz
function pushmostrar_Callback(hObject, eventdata, handles)
% hObject   manejador de pushmostrar
% eventdata reservado - para futuras implementaciones
% handles   estructura con manejadores y datos de usuario
% Variable para el manejo del instrumento
global vi
% Variable para guardar la traza
global figura
global ejefigura
% Se comprueba que el instrumento haya sido ya creado previamente
if(isempty(vi) == 0)
    % Se comprueba que el instrumento esté conectado
    if (strcmp(vi.status, 'open') == 1)
        % Número de puntos disponibles en la traza del instrumento
        num_puntos = 501;
        % Variable en la que guardamos los valores de potencia leídos
        leido = [];
        % Configuramos el instrumento para que los datos sean devueltos en ASCII
        fprintf(vi,'BIN 0');
        % Se pide cada uno de los puntos de la traza
        for i=0:(num_puntos-1)
            peticion = strcat('XMA? ', num2str(i), ',1');
            fprintf(vi,peticion);
            leido = [leido str2double(fscanf(vi))];
        end
        % Se escalan los valores leídos a dBm
        traza = leido*0.01;
        % Indicación del cuadro que se utilizaran para representar la traza
        axes(handles.axestraza);
        % Solicitud del span al instrumento
        fprintf(vi,'SP?');
        span = str2double(fscanf(vi));
        % Escalado del eje de representación
        eje = linspace(-span/2,span/2,501);
        figura = traza;
        ejefigura = eje;
        % Representación de la traza
        plot(eje, traza);
    % Si no está conectado se da un mensaje de error
    else
        msgbox('El instrumento no está conectado');
    end
% El mismo mensaje de error aparece si el instrumento no ha sido creado
else
    msgbox('El instrumento no está conectado');
end

```

```

% --- Se ejecuta al pulsar el botón pushlimpiar
% --- Limpieza del cuadro donde se muestran las trazas individuales
function pushlimpiar_Callback(hObject, eventdata, handles)
% hObject   manejador de pushlimpiar
% eventdata reservado - para futuras implementaciones
% handles   estructura con manejadores y datos de usuario
% Toma del control del manejador adecuado
axes(handles.axestraza);
% Borrado
cla;

% --- Se ejecuta al pulsar el botón pushcargarfigura
% --- Representación de una traza guardada anteriormente y
% --- almacenada en un fichero mat
function pushcargarfigura_Callback(hObject, eventdata, handles)
% hObject   manejador de pushcargarfigura
% eventdata reservado - para futuras implementaciones
% handles   estructura con manejadores y datos de usuario
% Selección del fichero donde se encuentra almacenada la traza
[archivo,path,FilterIndex] = uigetfile('mat','Seleccione el archivo de datos');
% Se controla que el usuario haya elegido un fichero
if archivo ~= 0
    % Carga del fichero, que debe contener las variables eje y traza
    load(archivo)
    % Toma del control del manejador adecuado
    axes(handles.axestraza);
    % Representación de la traza, la cual saldrá centrada en 0, punto
    % correspondiente a la frecuencia central. El eje se encuentra escalado en Hz
    plot(eje,traza)
end

% --- Se ejecuta al pulsar el botón Guardar correspondiente a una traza
% mostrada
% --- Podrá guardarse la figura mostrada
function pushbutton23_Callback(hObject, eventdata, handles)
% hObject   manejador de Guardar
% eventdata reservado - para futuras implementaciones
% handles   estructura con manejadores y datos de usuario
% Variables globales usadas en la función pushpintar para la
% caracterización del ruido de fase
global figura
global ejefigura
traza = figura;
eje =ejefigura;
% Llamada a una ventana para la elección de la ubicación y el nombre del
% fichero donde se guardaran los datos correspondientes
[filename, pathname] = uiputfile('*.*mat', 'Save Workspace as');
% Se controla que el usuario haya seleccionado un archivo
if filename ~= 0
    % Se guardan los datos de la figura
    save(filename, ' traza ', 'eje');
end

```

8.14 Anexo N – Caracterización del generador de RF

```

% Quinta iteración de la toma de medidas para la caracterización del ruido
% de fase
% Se fijan los valores de SPAN y RBW para está iteración, se envían las
% instrucciones correspondientes al RFSA y se realiza un barrido de la traza
% teniendo en cuenta el tiempo estimado para esos valores de SPAN y
% RBW
SPAN = 3000000;
RBW = 3000;
% Se guardan los manejadores correspondientes para que
% posteriormente puedan ser utilizados para guardar las trazas
% utilizadas en la caracterización del ruido de fase
set(handles.editspan5, 'String', num2str(SPAN));
set(handles.editrbw5, 'String', num2str(RBW));

stringrbw = strcat('RB ', num2str(RBW));
fprintf(vi,stringrbw);

stringspan = strcat('SP ', num2str(SPAN));
fprintf(vi,stringspan);

fprintf(vi, 'ST?');
tiempo= str2double(fscanf(vi));

fprintf(vi, 'TS');

pause(tiempo/10^6);

% El número de puntos de una traza es el definido por el RFSA, se solicita
% en cada iteración un punto de la traza mediante la instrucción definida
% en el RFSA
leido = [];

for i=0:(num_puntos-1)
    peticion = strcat('XMA? ', num2str(i), ',1');
    fprintf(vi,peticion);
    leido = [leido str2double(fscanf(vi))];
end

% Conversión de los datos leídos a dBm
leido = leido*0.01;
% Localización del máximo de amplitud dentro de la traza
inicio = find(leido == max(leido),1);
%Se debe tener en cuenta que procesamos exactamente 500 puntos y calcular de
%está forma el comienzo de las muestras a partir de el final de las
%muestras que se tomaron en la anterior iteración
offset = floor(comienzo_ruido/SPAN*500);
% Frecuencia en la cual se termina de leer el ruido, que siempre
% coincidirá con el final de la traza
final_ruido = floor((501 - inicio)*SPAN/501);
% Para cada traza utilizada se guarda el valor de la potencia de pico

```



```

ppicos(5) = max(leido);
% Almacenamiento de la traza en su variable global
trazas(:,5) = leido;
% Se normaliza el valor obtenido teniendo en cuenta la corrección del RBW y
% la normalización del máximo valor
leido = leido - max(leido) - 10*log10(RBW);
% Actualización de la figura resultante de ruido de fase con los nuevos
% valores de la iteración
resultado = [resultado leido((inicio+ offset + 1):end)];
% Actualización del eje del ruido de fase para su correcta representación
eje =[eje linspace(comienzo_ruido,final_ruido,501-offset-inicio)];
% Almacenamiento del eje de la traza individual en su variable global
ejes(:,5) = linspace(-SPAN/2,SPAN/2,501);
% Actualización del punto donde se continuaran las medidas en la
% siguiente iteración
comienzo_ruido = final_ruido;

```

8.15 Anexo O – Funciones de configuración

```

% --- Se ejecuta al pulsar el botón pushbutton_up cuando el usuario desea
% --- el primer eliminar un RBW y SPAN
% --- Los campos de RBW y SPAN son desplazados una posición hacia arriba,
% --- eliminando la primera posición
function pushbutton_up_Callback(hObject, eventdata, handles)
% hObject   manejador de pushbutton_up
% eventdata reservado - para futuras implementaciones
% handles   estructura con manejadores y datos de usuario
% Lectura de los campos de RBW
RBW_lista = [str2double(get(handles.editrbw1,'String')),
             str2double(get(handles.editrbw2,'String')),
             str2double(get(handles.editrbw3,'String')),
             str2double(get(handles.editrbw4,'String')),
             str2double(get(handles.editrbw5,'String')),
             str2double(get(handles.editrbw6,'String')),
             str2double(get(handles.editrbw7,'String')),
             str2double(get(handles.editrbw8,'String')),
             str2double(get(handles.editrbw9,'String')),
             str2double(get(handles.editrbw10,'String'))];
% Los valores leídos son asignados a la posición inmediatamente anterior
set(handles.editrbw1, 'String', num2str(RBW_lista(2)));
set(handles.editrbw2, 'String', num2str(RBW_lista(3)));
set(handles.editrbw3, 'String', num2str(RBW_lista(4)));
set(handles.editrbw4, 'String', num2str(RBW_lista(5)));
set(handles.editrbw5, 'String', num2str(RBW_lista(6)));
set(handles.editrbw6, 'String', num2str(RBW_lista(7)));
set(handles.editrbw7, 'String', num2str(RBW_lista(8)));
set(handles.editrbw8, 'String', num2str(RBW_lista(9)));
set(handles.editrbw9, 'String', num2str(RBW_lista(10)));
set(handles.editrbw10, 'String', '0'); %el último valor se le asigna
                                     %un valor de 0
% Lectura de los campos de span
SPAN_lista = [str2double(get(handles.editspan1,'String')),
             str2double(get(handles.editspan2,'String')),
             str2double(get(handles.editspan3,'String')),
             str2double(get(handles.editspan4,'String')),
             str2double(get(handles.editspan5,'String')),
             str2double(get(handles.editspan6,'String')),
             str2double(get(handles.editspan7,'String')),
             str2double(get(handles.editspan8,'String')),
             str2double(get(handles.editspan9,'String')),
             str2double(get(handles.editspan10,'String'))];
% Los valores leídos son asignados a la posición inmediatamente anterior
set(handles.editspan1, 'String', num2str(SPAN_lista(2)));
set(handles.editspan2, 'String', num2str(SPAN_lista(3)));
set(handles.editspan3, 'String', num2str(SPAN_lista(4)));
set(handles.editspan4, 'String', num2str(SPAN_lista(5)));
set(handles.editspan5, 'String', num2str(SPAN_lista(6)));
set(handles.editspan6, 'String', num2str(SPAN_lista(7)));
set(handles.editspan7, 'String', num2str(SPAN_lista(8)));

```

```

set(handles.editspan8, 'String', num2str(SPAN_lista(9)));
set(handles.editspan9, 'String', num2str(SPAN_lista(10)));
set(handles.editspan10, 'String', '0');
% Actualiza el valor del manejador
guidata(hObject,handles)

% --- Se ejecuta al pulsar el botón pushbutton_down
% --- Los campos de RBW y SPAN son desplazados una posición hacia abajo,
% --- dejando un nuevo hueco en la primera posición para introducir nuevos
% --- valores
function pushbutton_down_Callback(hObject, eventdata, handles)
% hObject   manejador de pushbutton_down
% eventdata reservado - para futuras implementaciones
% handles   estructura con manejadores y datos de usuario
% Lectura de los campos de RBW
RBW_lista = [str2double(get(handles.editrbw1,'String')),
             str2double(get(handles.editrbw2,'String')),
             str2double(get(handles.editrbw3,'String')),
             str2double(get(handles.editrbw4,'String')),
             str2double(get(handles.editrbw5,'String')),
             str2double(get(handles.editrbw6,'String')),
             str2double(get(handles.editrbw7,'String')),
             str2double(get(handles.editrbw8,'String')),
             str2double(get(handles.editrbw9,'String')),
             str2double(get(handles.editrbw10,'String'))];
% Los valores leídos son asignados a la posición inmediatamente siguiente
set(handles.editrbw10, 'String', num2str(RBW_lista(9)));
set(handles.editrbw9, 'String', num2str(RBW_lista(8)));
set(handles.editrbw8, 'String', num2str(RBW_lista(7)));
set(handles.editrbw7, 'String', num2str(RBW_lista(6)));
set(handles.editrbw6, 'String', num2str(RBW_lista(5)));
set(handles.editrbw5, 'String', num2str(RBW_lista(4)));
set(handles.editrbw4, 'String', num2str(RBW_lista(3)));
set(handles.editrbw3, 'String', num2str(RBW_lista(2)));
set(handles.editrbw2, 'String', num2str(RBW_lista(1)));
set(handles.editrbw1, 'String', '0');
% Lectura de los campos de span
SPAN_lista = [str2double(get(handles.editspan1,'String')),
             str2double(get(handles.editspan2,'String')),
             str2double(get(handles.editspan3,'String')),
             str2double(get(handles.editspan4,'String')),
             str2double(get(handles.editspan5,'String')),
             str2double(get(handles.editspan6,'String')),
             str2double(get(handles.editspan7,'String')),
             str2double(get(handles.editspan8,'String')),
             str2double(get(handles.editspan9,'String')),
             str2double(get(handles.editspan10,'String'))];
% Los valores leídos son asignados a la posición inmediatamente siguiente
set(handles.editspan10, 'String', num2str(SPAN_lista(9)));
set(handles.editspan9, 'String', num2str(SPAN_lista(8)));
set(handles.editspan8, 'String', num2str(SPAN_lista(7)));
set(handles.editspan7, 'String', num2str(SPAN_lista(6)));
set(handles.editspan6, 'String', num2str(SPAN_lista(5)));

```

```

set(handles.editspan5, 'String', num2str(SPAN_lista(4)));
set(handles.editspan4, 'String', num2str(SPAN_lista(3)));
set(handles.editspan3, 'String', num2str(SPAN_lista(2)));
set(handles.editspan2, 'String', num2str(SPAN_lista(1)));
set(handles.editspan1, 'String', '0');
% Actualiza el valor del manejador
guidata(hObject,handles)

% --- Se ejecuta al pulsar el botón pushbuttonlimpiardatos
% --- Pone a 0 todos los campos de RBW y SPAN
function pushbuttonlimpiardatos_Callback(hObject, eventdata, handles)
% hObject   manejador de pushbuttonlimpiardatos
% eventdata reservado - para futuras implementaciones
% handles   estructura con manejadores y datos de usuario
% Resetea los valores de RBW
set(handles.editrbw1, 'String', '0');
set(handles.editrbw2, 'String', '0');
set(handles.editrbw3, 'String', '0');
set(handles.editrbw4, 'String', '0');
set(handles.editrbw5, 'String', '0');
set(handles.editrbw6, 'String', '0');
set(handles.editrbw7, 'String', '0');
set(handles.editrbw8, 'String', '0');
set(handles.editrbw9, 'String', '0');
set(handles.editrbw10, 'String', '0');
% Resetea los valores de SPAN
set(handles.editspan1, 'String', '0');
set(handles.editspan2, 'String', '0');
set(handles.editspan3, 'String', '0');
set(handles.editspan4, 'String', '0');
set(handles.editspan5, 'String', '0');
set(handles.editspan6, 'String', '0');
set(handles.editspan7, 'String', '0');
set(handles.editspan8, 'String', '0');
set(handles.editspan9, 'String', '0');
set(handles.editspan10, 'String', '0');
% Actualiza el valor del manejador
guidata(hObject,handles)

% --- Se ejecuta al pulsar el botón pushbuttoncargardatos
% --- Si el usuario ha guardado datos de RBW y SPAN de una simulación
% --- anterior, podrá cargarlos desde el fichero correspondiente
function pushbuttoncargardatos_Callback(hObject, eventdata, handles)
% hObject   manejador de pushbuttoncargardatos
% eventdata reservado - para futuras implementaciones
% handles   estructura con manejadores y datos de usuario
% Llamada a ventana para la selección del archivo de datos
[fichero,path,FilterIndex] = uigetfile('mat','Seleccione el archivo de datos');
% Se controla que el usuario haya elegido un fichero
if fichero ~= 0
    % Carga del fichero de datos (extensión '.mat')
    load(fichero)
    % Asignación de los valores de RBW almacenados a los campos de datos de la

```

```

% interfaz
set(handles.editrbw1, 'String', num2str(RBW(1)));
set(handles.editrbw2, 'String', num2str(RBW(2)));
set(handles.editrbw3, 'String', num2str(RBW(3)));
set(handles.editrbw4, 'String', num2str(RBW(4)));
set(handles.editrbw5, 'String', num2str(RBW(5)));
set(handles.editrbw6, 'String', num2str(RBW(6)));
set(handles.editrbw7, 'String', num2str(RBW(7)));
set(handles.editrbw8, 'String', num2str(RBW(8)));
set(handles.editrbw9, 'String', num2str(RBW(9)));
set(handles.editrbw10, 'String', num2str(RBW(10)));
% Asignación de los valores de SPAN almacenados a los campos de datos de la
% interfaz
set(handles.editspan1, 'String', num2str(SPAN(1)));
set(handles.editspan2, 'String', num2str(SPAN(2)));
set(handles.editspan3, 'String', num2str(SPAN(3)));
set(handles.editspan4, 'String', num2str(SPAN(4)));
set(handles.editspan5, 'String', num2str(SPAN(5)));
set(handles.editspan6, 'String', num2str(SPAN(6)));
set(handles.editspan7, 'String', num2str(SPAN(7)));
set(handles.editspan8, 'String', num2str(SPAN(8)));
set(handles.editspan9, 'String', num2str(SPAN(9)));
set(handles.editspan10, 'String', num2str(SPAN(10)));
set(handles.comienzoRuido, 'String', num2str(principio_ruido_frecuencia));
% Actualiza el valor del manejador
guidata(hObject,handles)
end

% --- Se ejecuta al pulsar el botón pushbuttonguardardatos
% --- El usuario podrá guardar datos de RBW y SPAN de una simulación
% --- en un fichero .mat para su posterior utilización
function pushbuttonguardardatos_Callback(hObject, eventdata, handles)
% hObject   manejador de pushbuttonguardardatos
% eventdata reservado - para futuras implementaciones
% handles   estructura con manejadores y datos de usuario
% Llamada a ventana para la elección de la ubicación y el nombre del
% fichero de datos
[filename, pathname] = uiputfile('*.mat', 'Save Workspace as');
% Se controla que el usuario haya seleccionado un fichero
if filename ~= 0
    % Se guardan los valores de RBW de el interfaz
    RBW = [str2double(get(handles.editrbw1,'String')),
           str2double(get(handles.editrbw2,'String')),
           str2double(get(handles.editrbw3,'String')),
           str2double(get(handles.editrbw4,'String')),
           str2double(get(handles.editrbw5,'String')),
           str2double(get(handles.editrbw6,'String')),
           str2double(get(handles.editrbw7,'String')),
           str2double(get(handles.editrbw8,'String')),
           str2double(get(handles.editrbw9,'String')),
           str2double(get(handles.editrbw10,'String'))];
    % Se guardan los valores de SPAN de el interfaz
    SPAN = [str2double(get(handles.editspan1,'String')),

```

```
    str2double(get(handles.editspan2,'String')),
    str2double(get(handles.editspan3,'String')),
    str2double(get(handles.editspan4,'String')),
    str2double(get(handles.editspan5,'String')),
    str2double(get(handles.editspan6,'String')),
    str2double(get(handles.editspan7,'String')),
    str2double(get(handles.editspan8,'String')),
    str2double(get(handles.editspan9,'String')),
    str2double(get(handles.editspan10,'String'))];
%Se guarda la frecuencia desde donde se comienza a medir el ruido de
%fase
principio_ruido_frecuencia = str2double(get(handles.comienzoRuido,'String'));
% Se guardan las variables en el fichero seleccionado
save(filename, 'RBW', 'SPAN','principio_ruido_frecuencia')
end
```

8.16 Anexo P – Pintar Ruido de Fase

Código correspondiente a la adquisición de trazas y caracterización del ruido de fase:

```
% Se comprueba que los primeros valores de la lista no sean cero
if (RBW_lista(1)== 0 | SPAN_lista(1)== 0)
    msgbox('Los datos deben estar ordenados desde el primer campo en adelante (siendo
distintos de cero)');
else
    % Teniendo en cuenta los datos introducidos por el usuario, estos se envían
    % al RFSA mediante los comandos correspondientes de forma secuencial y
    % se obtienen cada una de las señales necesarias para la caracterización
    % del ruido de fase.

for i=1:10
    if (RBW_lista(i)== 0 | SPAN_lista(i)== 0)
        break;
    end
    % Se toman los valores indicados por el usuario, obteniéndose de las
    % variables en las que inicialmente son guardados
    SPAN = SPAN_lista(i);
    RBW = RBW_lista(i);

    % Se indica al RFSA que actualice el SPAN y el RBW
    stringspan = strcat('SP ', num2str(SPAN));
    fprintf(vi,stringspan);
    stringrbw = strcat('RB ', num2str(RBW));
    fprintf(vi,stringrbw);

    % Al enviar el RBW el RFSA lo aproxima a su filtro de resolución
    % inmediatamente mayor, este valor es leído del RFSA para posteriormente
    % ser indicado al usuario
    fprintf(vi, 'RB?');
    RBW_actualizado(i) = str2double(fscanf(vi));

    % Lectura del tiempo en el que se realizara el barrido
    fprintf(vi, 'ST?');
    tiempo= str2double(fscanf(vi));
    % Solicitud para la realización del barrido en el RFSA
    fprintf(vi, 'TS');
    % Espera para la toma de la traza
    pause(tiempo/10^6);

    % variable para obtener la información de la traza obtenida en el RFSA
    leído = [];
    % Se envía el comando correspondiente para solicitar al RFSA cada una de
    % las muestras tomadas para la realización de la traza. El número de
    % puntos es 501 que son las muestras obtenidas por el RFSA
    for j=0:(num_puntos-1)
        peticion = strcat('XMA? ', num2str(j), ', 1');
        fprintf(vi,peticion);
        leído = [leído str2double(fscanf(vi))];
    end
end
```

```

end

%Se normaliza el valor para ponerlo en dBm's
leido = leido*0.01;
%Para cada traza utilizada se guarda el valor de la potencia de pico
ppicos(i) = max(leido);
% Almacenamiento de la traza en su variable global
trazas(:,i) = leido;
%Se corrige el valor de la lectura teniendo en cuenta el valor de
%amplitud máximo, así como el RBW utilizado.
leido = leido - max(leido) - 10*log10(RBW_actualizado(i));

% Localización del máximo de amplitud dentro de la traza
inicio = find(leido == max(leido),1);
% Punto de la traza desde el cual se comienzan a guardar los datos de
% ruido en cada iteración
offset = floor(comienzo_ruido/SPAN*500);
% Frecuencia en la cual se termina de leer el ruido, que siempre
% coincidirá con el final de la traza
final_ruido = floor((501 - inicio)*SPAN/501);

% Actualización de la figura resultante de ruido de fase con los nuevos
% valores de la iteración
resultado = [resultado leido((inicio+ offset + 1):end)];
% Actualización del eje del ruido de fase para su correcta representación
eje = [eje linspace(comienzo_ruido,final_ruido,501-offset-inicio)];

% Almacenamiento del eje de la traza individual en su variable global
ejes(:,i) = linspace(-SPAN/2,SPAN/2,501);

% Actualización del punto donde se continuaran las medidas en la
% siguiente iteración
comienzo_ruido = final_ruido;
end

```

Código correspondiente a la representación del ruido de fase:

```

% Si se han tomado medidas, se presentan los resultados en el eje
% correspondiente
if (~isempty(resultado))

    % Almacenamiento del resultado y su eje correspondiente en sus
    % variables globales
    ruidofase = resultado;
    ejeruidofase = eje;

    axes(handles.axesruidofase);
    semilogx(ejeruidofase,ruidofase);
    axis tight
end

```


8.17 Anexo Q – Guardar trazas

```

function pushbuttonguardartrazas_Callback(hObject, eventdata, handles)
% hObject   manejador de pushbuttonguardartrazas
% eventdata reservado - para futuras implementaciones
% handles   estructura con manejadores y datos de usuario
% Variables globales donde se encuentran los resultados de la última
% simulación de caracterización de ruido
global trazas
global ejes
global fc
global ppicos

% Llamada a una ventana para la elección de la ubicación y el nombre del
% fichero donde se guardaran los datos correspondientes a cada traza
[filename, pathname] = uiputfile({'*.mat','MAT-files (*.mat)'; '*.xls','Excel-files (*.xls)';}, 'Save
Workspace as');
% Cuando el usuario haya seleccionado un nombre de fichero se procede a
% guardar los datos
if filename ~= 0
    % Se obtienen los valores de RBW del interfaz
    RBW_lista = [str2double(get(handles.editrbw1,'String')),
        str2double(get(handles.editrbw2,'String')),
        str2double(get(handles.editrbw3,'String')),
        str2double(get(handles.editrbw4,'String')),
        str2double(get(handles.editrbw5,'String')),
        str2double(get(handles.editrbw6,'String')),
        str2double(get(handles.editrbw7,'String')),
        str2double(get(handles.editrbw8,'String')),
        str2double(get(handles.editrbw9,'String')),
        str2double(get(handles.editrbw10,'String'))];

    % Se obtienen los valores de SPAN de la interfaz
    SPAN_lista = [str2double(get(handles.editspan1,'String')),
        str2double(get(handles.editspan2,'String')),
        str2double(get(handles.editspan3,'String')),
        str2double(get(handles.editspan4,'String')),
        str2double(get(handles.editspan5,'String')),
        str2double(get(handles.editspan6,'String')),
        str2double(get(handles.editspan7,'String')),
        str2double(get(handles.editspan8,'String')),
        str2double(get(handles.editspan9,'String')),
        str2double(get(handles.editspan10,'String'))];

    i = 1;
    % Si el fichero es '.mat'
    if strcmp(filename(end-3:end),'.mat')
        % Para cada RBW se crea un fichero
        while(RBW_lista(i) ~= 0)
            % El nombre de fichero se crea con la concatenación del nombre
            % elegido por el usuario, el RBW y el SPAN correspondientes

```

```

    nombrefichero = strcat(filename(1:end-
4),'_RBW',num2str(RBW_lista(i)),'_SPAN',num2str(SPAN_lista(i)),'.mat');
    % Se guardan los datos correspondientes a cada traza
    RBW = RBW_lista(i);
    SPAN = SPAN_lista(i);
    eje = ejes(:,i);
    traza = trazas(:,i); % La traza se guarda en dBc
    ppico = ppicos(i);
    % Se guardan los datos en el fichero
    save(nombrefichero, 'RBW', 'SPAN', 'eje', 'traza', 'fc', 'ppico')
    i = i + 1;
end
else
    % Se desactivan los warnings ya que, con cada hoja Excel creada
    % dentro del fichero, se muestra un mensaje por pantalla indeseado
    warning('off', 'all')
    % Para cada RBW se crea una hoja nueva
    while(RBW_lista(i) ~= 0)
        % Se guardan los datos correspondientes a cada traza
        RBW = RBW_lista(i);
        SPAN = SPAN_lista(i);
        eje = ejes(:,i);
        traza = trazas(:,i); % La traza se guarda en dBc
        ppico = ppicos(i);
        % Se guardan los datos en la hoja, ordenados por columnas
        % A --> eje de la traza
        % B --> valores de potencia
        % C1 --> frecuencia central utilizada
        % D1 --> RBW utilizado
        % E1 --> SPAN utilizado
        % F1 --> Potencia de pico obtenida
        [estado]= xlswrite(filename,eje,i,'A1');
        [estado]= xlswrite(filename,traza,i,'B1');
        [estado]= xlswrite(filename,fc,i,'C1');
        [estado]= xlswrite(filename,RBW,i,'D1');
        [estado]= xlswrite(filename,SPAN,i,'E1');
        [estado]= xlswrite(filename,ppico,i,'F1');
        i = i + 1;
    end
    % Se restablecen los warning una vez finalizada la creación de las
    % hojas
    warning('on', 'all')
end
end
end

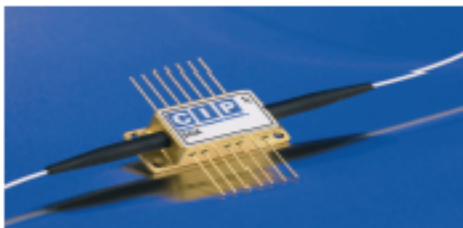
```

8.18 Anexo R – SOA-XN-OEC-1550

PRELIMINARY DATASHEET | SOA-XN-OEC-1550



SOA-XN-OEC-1550 - 1.55µm Non-Linear Semiconductor Optical Amplifier (SOA)



Features

- 1.55µm operation
- High gain
- Low saturation output power
- Very fast gain recovery times
- Low saturated PDG
- InP Buried Heterostructure design
- TEC cooled
- Available as packaged device
- APC connectors provided

Application Examples

- All optical processing
- Wavelength conversion
- Cross-phase modulation
- Cross-gain modulation
- Four-wave mixing

Description

The ultra non-linear SOA-XN is a high confinement factor device optimised for high gain and very fast gain recovery providing very fast optical switching speeds. It utilises CIP's proprietary InP buried heterostructure design and is available in either a 14-pin butterfly package with a thermistor, thermoelectric cooler and single mode fibre pig-tails

The product is appropriate in wavelength converter or four wave mixing applications and can also be used in CIP's hybrid integrated regenerator/wavelength converter products. Custom specification variants may be defined as appropriate for other non-linear applications. Arrays of non-linear SOAs are also available either packaged or as chip-on-carrier.

Optical and electrical specifications All measurements are at chip temperature of 20°C @ 1535-1560nm unless stated otherwise.

Item	Test condition	Min.	Typ.	Max.	Unit
Small signal Gain	I = 500mA		25		dB
Polarisation Dependent Saturated Gain (PDG)	I = 500mA, P _{in} >0dBm		1		dB
Saturated Output Power (SOP)	I = 500mA		12		dBm
Gain Peak	I = 500mA	1550		1570	nm
Saturated Gain Recovery Time (1/e)	I = 500mA, P _{in} >0dBm, 1555nm		10		ps
3dB Optical bandwidth	I = 500mA		35		nm

www.ciphotonics.com

.....photonic solutions

CIP Technologies, Adastral Park, Martlesham Heath, Ipswich, IPS 3RE, UK Tel: +44 (0) 1473 663210 info@ciphotonics.com REV: F



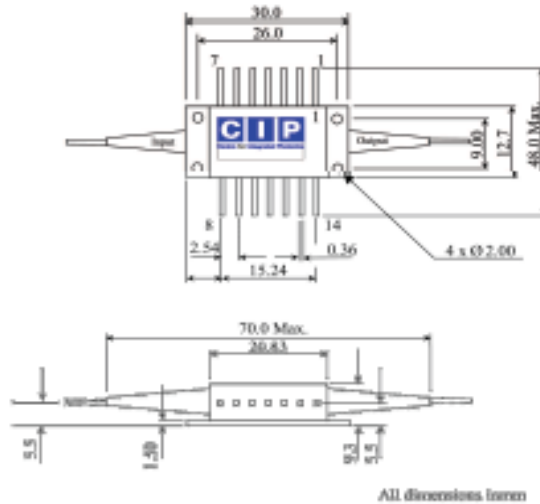
Absolute maximum ratings

Item	Symbol	Rating	Unit
Maximum DC current forward bias	I_{Tmax}	600	mA
Maximum DC voltage forward bias	V_{Tmax}	5.0	V
Maximum DC voltage reverse bias	V_{Rmax}	2.0	V
Maximum optical input power	P_{Max}	+13	dBm
Maximum TEC current	TEC_{IK}	2.0	A
Maximum TEC voltage	TEC_{Vik}	3.6	V
Case operating temperature	T	15-40	°C
Storage temperature	T_s	10-70	°C
Fibre type	SMF-28 900µm tight buffer, >1m		

Ordering Information— Part Number SOA-XN-OEC-1550

For custom products please contact CIP Sales on +44 1473 663210 or e-mail sales@ciphotonics.com.
For details of your local agent, visit www.ciphotonics.com

Wiring Diagram and Dimensions



Pin Number	Connection
1	Cooler (-)
2	Ther resistor
3	Case Ground
4	NC
5	Ther resistor (Dk2)
6	Case Ground
7	NC
8	NC
9	NC
10	SOA Chip (+)
11	SOA Chip (-)
12	Case Ground
13	Case Ground
14	Coder (-)



CIP reserves the right to make changes in the design, specifications and other information at any time, and without prior notice. The information contained within the Data Sheet is believed to be accurate. However, no responsibility is assumed for possible inaccuracy or omission. Any information contained herein shall legally bind CIP only if it is specifically incorporated in the terms and conditions of a sales agreement.

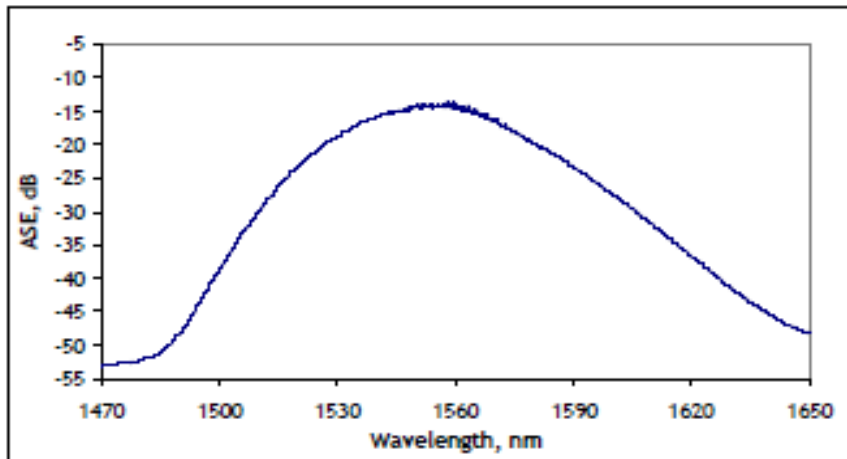
Parts of this product are manufactured under one or more of the following patents licensed from British Telecommunications PLC :

European: 14000-384764; 416879-531377; 200129-156566; 227783-218344; 279680-261943; 390614-174729; 228435-228435; 242084-245025; 746887-767923; 830721-1181591; 93527-3-1183581; 170457-225085; 247834-292288; 320305-537207; 624257-642027; 94905188-2691044; 772924-782713; 822428; 822426; 813378-979003753; 97908417-5; 97908417-5; 85124-15-4826295; 5426312-548197; 5202897-2008926; 473487-4728628; 485936; 475445; 4964134; 524257-53329542; 4736164; 481207; 4981814; 5015964; 844929-585246; 6188511; 6625371; 657107-959329; 08-4744619; 4792690; 4879761-4869704; 497302-4995100; 5286237-5371800; 5656507; 6075625; 6329945; 6097512; 5719874; 5832011-5917636; 5841928; 5978400; 610452-6052213; 5974073; 6170880. Canada: 297211-128460; 2182591-218095; 2221693; 2372581-2379401-2373546; 235485; 1246119; 1281802-1298887-1293996; 285596; 2117602-2280472; 2153798; 2155528; 2185132-2199933; 2367133-2282736; 2240519-2248042; 126848; 2047196; 2065247; 202939-2243279; 1236554; 1228936; 1261483; 265604; 1295722-1332541; 2449356. Japan: 2837265; 2968335; 100942; 2134710; 2547001; 2925415; 2140794; 2708165; 298465; 2018663; 868104; 2670519; 2128900; 2764141; 1957418; 2664457; 2081567; 3117708; 3404040; 356665; 3346570; 95-525402; 97-525789; 97-54136



Typical parameter curves

ASE curve for packaged device at chip temperature of 20°C and drive current of 500mA

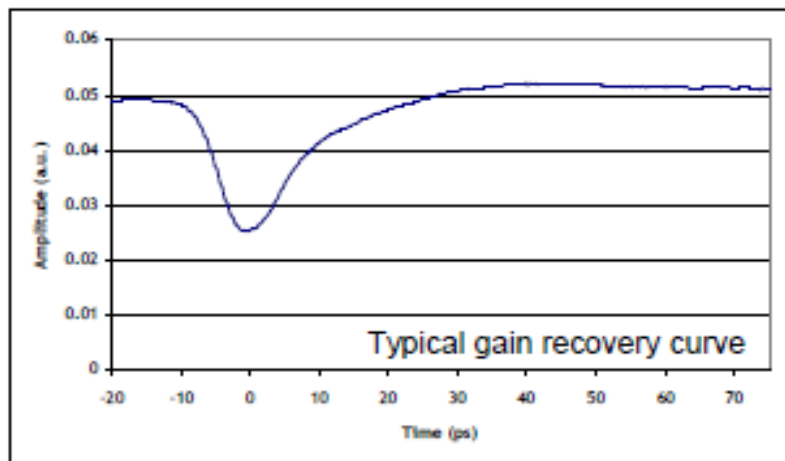


Typical Non-Linear Results

For devices tested at 20°C at 500mA drive current. Pump/probe sampling measurements taken with:

- Input 1550nm CW signal of 0.0dBm*
- Pulses (3ps) @ 2.5GHz and a wavelength of 1539nm*

The typical 1/e recovery time is ~10ps.

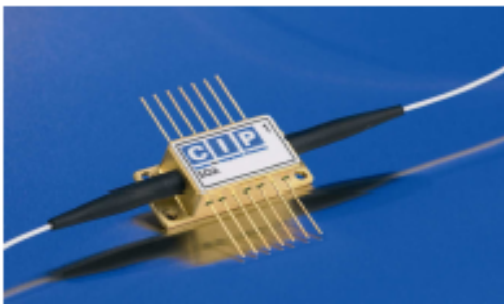


8.19 Anexo S – SOA-L-OEC-1550

PRELIMINARY DATASHEET | SOA-L-OEC-1550



SOA-L-OEC-1550 - 1.55µm Linear Semiconductor Optical Amplifier (SOA)



Features

- 1.55µm operation
- Bit rate independent– Suitable for all data rates
- Very high saturated output power (~16dBm)
- Low noise figure (<7.5dB)
- Low ASE ripple (<0.2dB)
- Low polarisation dependence (1dB)
- InP buried heterostructure design
- TEC cooled
- APC connectors provided

Application Examples

- Power booster (DWDM & CWDM networks)
- Transmission amplifier
- OADM gain block
- Access PON amplifier

Package Qualified to Telcordia GR-468 Issue 2

Description

The Linear SOA-L is a low confinement factor device optimised for moderate gain, low noise and high saturated output power. It utilises CIP's proprietary InP buried heterostructure design and is available in a 14-pin butterfly package with a thermistor, thermo-electric cooler and single mode fibre pigtailed.

Custom specification variants may be defined as appropriate in other custom wavebands. More detailed operation characteristics outside specification below are available on request.

Optical and electrical specifications All measurements are at chip temperature of 20°C @ 1535-1560nm unless stated otherwise. I= bias current

Item	Test condition	Min.	Typ.	Max.	Unit
Gain	I = 500mA	10	14		dB
Polarisation Dependent Gain (PDG)	I = 500mA		0.5	1.2	dB
Noise Figure (NF)	I = 500mA		6.5	7.5	dB
Saturated Output Power (SOP)	I = 500mA		16		dBm
ASE Ripple	I = 500mA			0.2	dB
3dB Optical Bandwidth	I = 500mA		90		nm
Wavelength of Peak Gain	I = 500mA		1495		nm
Extinction Ratio at 1550nm	I = 0mA to I = 500mA		55		dB
Electrical switching speed	10-90% rise and fall times		1		ns

www.ciphotonics.com

.....photonic solutions

CIP Technologies, Adastral Park, Martlesham Heath, Ipswich, IP5 3RE, UK Tel: +44 (0) 1473 663210 Info@ciphotonics.com REV: M

PRELIMINARY DATASHEET | SOA-L-OEC-1550



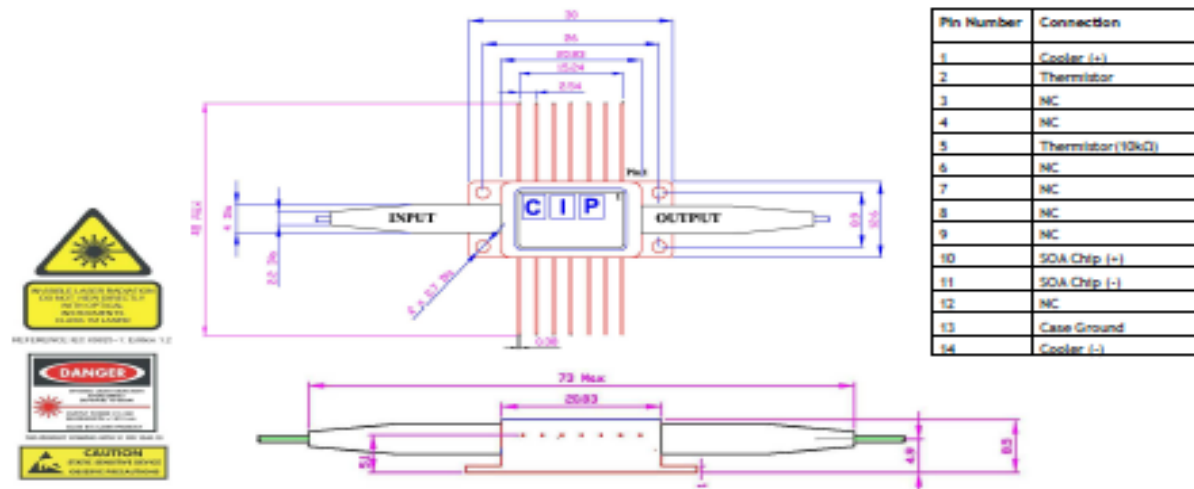
Absolute maximum ratings

Item	Symbol	Rating	Unit
Maximum DC current forward bias	I_{Fmax}	600	mA
Maximum DC voltage forward bias	V_{Fmax}	3.0	V
Maximum DC voltage reverse bias	V_{Rmax}	2.0	V
Maximum optical input power	P_{Max}	+13	dBm
Maximum TEC current	TEC _{IM}	2.0	A
Maximum TEC voltage	TEC _{VM}	3.6	V
Case operating temperature	T	-5 to 70	°C
Storage temperature	T _S	-40 to 85	°C
Fibre type	SMF-28 900µm tight buffer, >1m		

Ordering Information— Part Number SOA-L-OEC-1550

For custom products please contact CIP Sales on +44 1473 663210 or e-mail sales@ciphotonics.com. For details of your local agent, visit www.ciphotonics.com

Wiring Diagram and Dimensions



CIP reserves the right to make changes in the design, specifications and other information at any time, and without prior notice. The information contained within the Data Sheet is believed to be accurate. However, no responsibility is assumed for possible inaccuracy or omission. Any information contained herein shall legally bind CIP only if it is specifically incorporated in the terms and conditions of a sales agreement.

Parts of this product are manufactured under one or more of the following patents licensed from British Telecommunications PLC :

European 143000;384764;416879;531377;890129;156566;227783;218344;279680;261943;390614;174729;228435;228435;242084;245085;746887;767921;830721;1181591;93527_3;1183561;170457;225015;247834;292328;320305;537237;624257;647327;94905188_2;891044;772904;782713;822425;822425;813761;87900375_3;87906417_5;87908417_5;865124 US 4826295;5426310;5481397;5202897;6008926;4734387;4728628;4935936;4754459;4964134;5243857;53329542;4736164;4817207;4981814;5015964;844929;5852896;6188511;6625371;6571037;959329109;4744619;4793690;4879761;4969704;4973122;4995100;5216237;5371820;5656507;6075625;6229945;6097512;5719974;5832011;5917636;5841928;5975400;6104852;6052213;5974073;6178280; Canada 297211;1284683;2182591;2193095;2221693;2372581;2372401;2373546;1255485;1244519;1281802;1296887;1293996;2085596;2117682;2280473;2153796;2153528;2185132;2199513;2236713;2212736;2240319;2248042;1268848;2047196;2065247;2082939;2243279;1236554;1238136;1261482;285804;1295722;1322341;2048356; Japan 2837285;2968325;1000942;2134710;2547001;2925415;2140794;2708165;2494365;2018662;1866104;2670519;2128400;2764141;1957418;2664457;2081567;3117706;3404040;3556665;3346570;95-525482;97-525789;97-524136

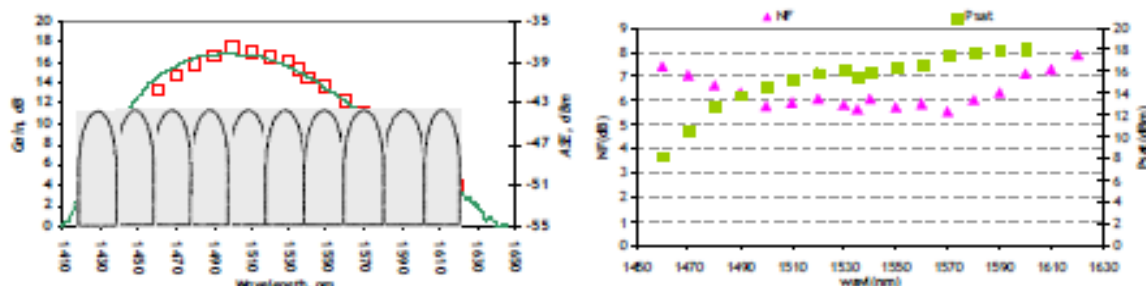
Parts of this product are manufactured under the following patent licensed from Corning Incorporated : US 10/376466

www.ciphotonics.com**photonic solutions**
 CIP Technologies, Adastral Park, Martlesham Heath, Ipswich, IP5 3RE, UK Tel: +44 (0) 1473 663210 info@ciphotonics.com REV: M

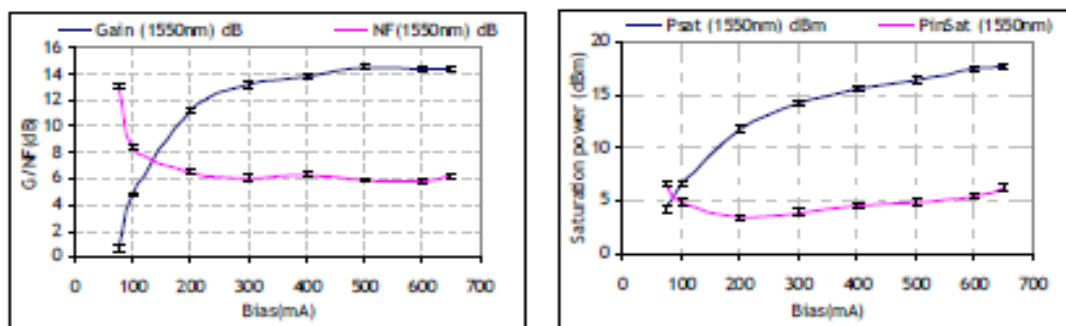
Typical parameter curves

For packaged device at chip temperature of 20°C

Performance at 500mA drive vs wavelength



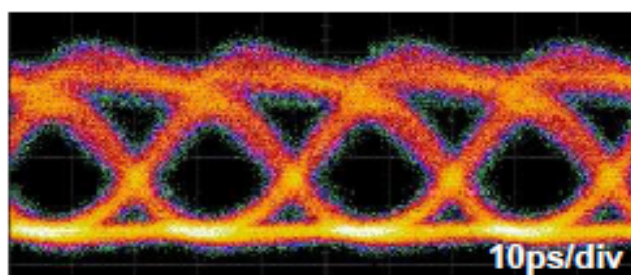
Performance at 1550nm vs drive current



Data modulation at 40 Gbit/s.

Below the saturated output power SOA performance is bit-rate independent

The eye diagram below shows the SOA operating as in-line amplifier at 40Gb/s (Pin=0dBm, bias 500mA)



Glosario

ASCII – American Standard Code for Information Interchange

ASRL – Asynchronous Serial

CCW – Counter-ClockWise

CPM – Colliding Pulse Mode - Locked

CW – ClockWise

EDFA –Erbium Doped Fiber Amplifier

EMBH – Etched-Mesa Buried Heterostructure

IF – Intermediate Frequency

GPIB – General Purpose Interface Bus

GUI – Graphic User Interface

HML – Hybrid Mode Locking

InP JePPIX – Joint European Platform for InP-based Photonic Integrated Components and Circuits

LED – Light Emitting Diode

MMI – Multi-Mode Interference

NC – Not Connected

NF – Noise Figure

OC – Open Connection

OSA – Optical Spectrum Analyzer

OTDM – Optical Time Division Multiplexing

Glosario

PC – Personal Computer

PCF – Peak Center Frequency

PIC – Photonic Integrated Circuit

PML – Passive Mode Locking

PRL – Peak Reference Level

QML – Q-switched Mode Locking

RBW – Resolution BandWidth

RFSA – Radio-Frequency Spectrum Analyzer

RIN – Relative Intensity Noise

SA – Saturable absorber

SOA – Semiconductor Optical Amplifier

SSB – Single Side Band

TDM – Time Division Multiplexing

USB – Universal Serial Bus

VISA – Virtual Instrument Software Architecture

WDM – Wavelength Division Multiplexing

Referencias

1. **Kaiser, Ronald y Hüttl, Bernd.** Monolithic 40-GHz Mode-Locked MQW DBR Lasers for High-Speed Optical Communication Systems. *IEEE JOURNAL OF SELECTED TOPICS IN QUANTUM ELECTRONICS*. Enero/Febrero 2007, Vol. 13, 1.
2. **Bente, Erwin.** Unravelling high-speed optical dataflows. *OTDM Freeband*. 2006.
3. **Carroll, J., Whiteaway, J. y Plumb, D.** *Distributed feedback semiconductor lasers*. s.l. : SPIE PRESS, 1998. ISBN: 0-8194-2660-1.
4. **Vasil'ev, Peter.** *Ultrafast diode lasers: fundamentats and applications*. s.l. : ARTECH HOUSE, 1995. ISBN: 0-89006-736-8.
5. **Sorel, M., y otros.** Operating Regimes of GaAs–AlGaAs Semiconductor. *IEEE JOURNAL OF QUANTUM ELECTRONICS*. Octubre 2003, Vol. 39, 10.
6. **Barbarin, Yohan.** *1.55 μm Integrated Modelocked Semiconductor Lasers*. Eindhoven : CIP-DATA LIBRARY TECHNISCHE UNIVERSITEIT EINDHOVEN, Abril 2007. ISBN: 978-90-386-2013-8. Disponible [Internet]: < <http://alexandria.tue.nl/extra2/200710406.pdf> > [15 de septiembre de 2010]
7. **Tucker, Rodney S. y Kaminow, Ivan P.** High- Frecuency Characteristics od Directly Modulates InGaAsP Ridge Waveguide and Buried Heterostructure Lasers. *IEEE JOURNAL OF LIGHTWAVE TECNOLOGY*. Agosto 1984, Vols. LT-2, 4
8. **VPIPhotonics.** *Active Photonics User's Manual*. Mayo 2007. CMA - UM01 - 1307131.
9. **Avrutin, E.A., Mash, J.H. y Portnoi, E.L.** Monolithic and multi-GigaHertz mode-locked semiconductor lasers: Constructions, experiments, models and applications. *IEE Proc.- Optoelectron*. Agosto 2000, Vol. 147, 4.
10. **Shimizu, T., Ogura, I. y Yokoyama, H.** 860 GHz rate asymmetric colliding pulse modelocked diode lasers. *Electronics Letters*. 1997, Vol. 33, 22.
11. **Yvind, K., y otros.** High-performance 10GHz all-active monolithic modelocked semiconductor lasers. *ELECTRONICS LETTERS*. Junio 2004, Vol. 40, 12.
12. **Kroh, Marcel.** *Semiconductor Mode-Locked Laser for High-Speed OTDM Transmission*. Berlín : s.n., Enero 2006. Disponible [Internet]: < http://opus.kobv.de/tuberlin/volltexte/2006/1268/pdf/kroh_marcel.pdf > [26 de septiembre de 2010]

Referencias

13. **Williams, K. A., Thompson, M. G. y White, I. H.** Long-wavelength monolithic mode-locked diode lasers. *New Journal of Physics*. Noviembre 2004, 6. Disponible [Internet]: <http://iopscience.iop.org/1367-2630/6/1/179/pdf/1367-2630_6_1_179.pdf> [10 de julio de 2010]
14. Encyclopedia of Laser Physics and Technology. [En línea] 16 de Mayo de 2010. [Citado el: 2 de Octubre de 2010.] http://www.rp-photonics.com/q_switching.html.
15. Encyclopedia of Laser Physics and Technology. [En línea] 13 de Julio de 2008. [Citado el: 2 de Octubre de 2010.] http://www.rp-photonics.com/q_switched_mode_locking.html.
16. Encyclopedia of Laser Physics and Technology. [En línea] 16 de Mayo de 2010. [Citado el: 2 de Octubre de 2010.] http://www.rp-photonics.com/gain_switching.html.
17. **Krauss, T., Laybourn, P.J.R. y Roberts, J.** CW operation of semiconductor ring lasers. *ELECTRONICS LETTERS*. Octubre 1990, Vol. 26, 25.
18. **Sorel, M., y otros.** Unidirectional bistability in semiconductor waveguide ring lasers. *APPLIED PHYSICS LETTERS*. Abril 2002, Vol. 80, 17.
19. **Krauss, T.F., y otros.** Efficient Semiconductor Ring Lasers Made by a Simple Self-Aligned Fabrication Process. *IEEE JOURNAL OF SELECTED TOPICS IN QUANTUM ELECTRONICS*. Junio 1995, Vol. 1, 2.
20. **ILX Lightwave Corporation.** *The Differences Between Threshold Current Calculations Methods*. 2003. Disponible [Internet]: <<http://www.ilxlightwave.com/appnotes/AN%2012%20REV01%20Differences%20Between%20Threshold%20Current%20Calculations.pdf>> [23 de septiembre de 2010]
21. **Bao, H. C y Liu, H. F.** Amplitude Noise of Subharmonically Hybrid Mode-Locked Pulses Generated From a Monolithic Semiconductor Laser. *IEEE PHOTONICS TECHNOLOGY LETTERS*. Enero 2002, Vol. 14, 1.
22. **Miranda, J. M., y otros.** *Ingeniería de Microondas*. Madrid : Prentice Hall, 2002. ISBN: 84-205-3099-9.
23. **Anritsu.** *Technical Datasheet. MG3690A R/F Microwave Signal Generators*.
24. **ILX Lightwave** (Laser Diode Instrumentation & Test Systems). *User's Guide. Laser Diode Controller. LCD-3700B Series*.
25. **Anritsu.** *MS2668C- Spectrum Analyzer 9kHz to 40 GHz. For Microwave and Milimeter wave Measurements*.