

MANUAL DE INSTALACIÓN

Sistema CODEOAM

Codificación y Decodificación a partir del
Momento Angular Orbital de la Luz y GNU Radio

Deiby Fernando Ariza Cadena

Código: 2195590

Director:

Dr. Omar Javier Tijero Rojas

Universidad Industrial de Santander

Escuela de Ingenierías Eléctrica, Electrónica y de Telecomunicaciones (E3T)

Bucaramanga

2025

Índice

1. Información Importante	3
1.1. Sobre el Sistema	3
1.2. Requisitos Previos	3
2. Paso 1: Preparar el Sistema Operativo	4
2.1. Opción A: Ya tiene Ubuntu instalado	4
2.2. Opción B: Usar WSL2 en Windows	4
3. Paso 2: Abrir la Terminal	4
4. Paso 3: Instalar Software Necesario	5
4.1. Actualizar el Sistema	5
4.2. Instalar GNU Radio y Dependencias	5
4.3. Verificar Instalación	5
5. Paso 4: Descargar el Código del Repositorio	6
5.1. Clonar el Repositorio	6
5.2. Instalar Dependencias Python Adicionales	6
6. Paso 5: Instalar Bloques OAM en GNU Radio	6
6.1. Ejecutar el Script de Instalación	6
6.2. Verificar Instalación de Bloques	7
7. Paso 6: Ejecutar el Sistema	7
7.1. Método 1: Ejecución Python Directa (Recomendado)	7
7.2. Método 2: Ejecución vía GNU Radio Companion	8
7.3. Método 3: Modo Sin Interfaz (Headless)	8

8. Paso 7: Cambiar Parámetros del Sistema	9
8.1. Archivo de Configuración Central	9
8.2. Parámetros Principales	9
9. Paso 8: Solución de Problemas	10
9.1. Los Bloques No Aparecen en GNU Radio	10
9.2. Error: ModuleNotFoundError	10
9.3. Datos Corruptos o Caché Inválido	11
9.4. GNU Radio Version Incorrecta	11
10. Resultados Esperados	11
10.1. Salida por Consola	11
10.2. Archivos Generados	11
11. Información Adicional	12
11.1. Repositorio del Código	12
11.2. Documentación	12
11.3. Soporte y Contacto	12

1 Información Importante

Este manual describe el proceso de instalación del sistema de comunicación OAM (Orbital Angular Momentum) desarrollado como parte del trabajo de grado “*Codificación y Decodificación a partir del Momento Angular Orbital de la Luz y GNU Radio*”.

1.1 Sobre el Sistema

El sistema OAM es una plataforma de simulación de comunicaciones ópticas de espacio libre (FSO) que utiliza haces Laguerre-Gauss con momento angular orbital para transmisión de datos.

Permite modelar:

- Generación de haces LG con diferentes cargas topológicas ($\ell = \pm 1, \pm 2, \pm 3, \pm 4$)
- Propagación atmosférica con turbulencia (modelo de Kolmogorov)
- Detección modal mediante proyección y correlación normalizada (NCC)
- Análisis de desempeño bajo diferentes condiciones atmosféricas

1.2 Requisitos Previos

Sistema Operativo:

- Ubuntu 22.04 LTS o posterior (recomendado)
- Otras distribuciones Linux con soporte para GNU Radio 3.10+

Acceso:

- Permisos de administrador (sudo) para instalar dependencias
- Conexión a internet para descargar repositorio y paquetes

2 Paso 1: Preparar el Sistema Operativo

2.1 Opción A: Ya tiene Ubuntu instalado

Si ya tiene Ubuntu 22.04 o posterior instalado, puede proceder directamente al **Paso 2**.

2.2 Opción B: Usar WSL2 en Windows

Si utiliza Windows, puede instalar Ubuntu mediante WSL2 (Windows Subsystem for Linux 2):

1. Abrir PowerShell como administrador

2. Ejecutar:

```
wsl --install -d Ubuntu-22.04
```

3. Reiniciar el equipo si es necesario

4. Al iniciar Ubuntu por primera vez, crear usuario y contraseña

Nota: WSL2 proporciona un entorno Linux completo dentro de Windows con rendimiento casi nativo.

3 Paso 2: Abrir la Terminal

Una vez que tenga Ubuntu disponible:

- En Ubuntu nativo: Presionar Ctrl + Alt + T
- En WSL2: Buscar “Ubuntu” en el menú de inicio de Windows

Todas las instrucciones siguientes se ejecutan en esta terminal.

4 Paso 3: Instalar Software Necesario

4.1 Actualizar el Sistema

Primero, actualizar la lista de paquetes disponibles:

```
1 sudo apt update
```

4.2 Instalar GNU Radio y Dependencias

Instalar GNU Radio y las bibliotecas Python necesarias:

```
1 sudo apt install gnuradio python3-gnuradio python3-pip \  
2                 python3-numpy python3-scipy python3-matplotlib \  
3                 python3-pyqt5 git
```

Explicación de componentes:

- gnuradio: Plataforma de procesamiento de señales (versión 3.10+)
- python3-gnuradio: Bindings Python para GNU Radio
- python3-pip: Gestor de paquetes Python
- python3-numpy: Biblioteca de arrays y cálculo numérico
- python3-scipy: Funciones científicas (Laguerre, Bessel)
- python3-matplotlib: Visualización de gráficos
- python3-pyqt5: Framework de interfaz gráfica
- git: Control de versiones para clonar el repositorio

4.3 Verificar Instalación

Comprobar que GNU Radio se instaló correctamente:

```
1 gnuradio-companion --version
```

Debe mostrar: GNU Radio Companion 3.10.x.x o superior.

5 Paso 4: Descargar el Código del Repositorio

5.1 Clonar el Repositorio

El código fuente está disponible públicamente en GitHub:

```
1 git clone https://github.com/DeibyArizac/OAM.git
2 cd OAM
```

Contenido del repositorio:

- `oam_complete_system.py`: Sistema completo autónomo
- `oam_complete_flowgraph.grc`: Diagrama de flujo GNU Radio
- `oam_system_config.py`: Configuración centralizada
- `install.sh`: Script de instalación de bloques
- `requirements.txt`: Dependencias Python adicionales
- `grc/`: Definiciones de bloques personalizados
- Módulos principales: `oam_source.py`, `oam_encoder.py`, `oam_channel.py`, `oam_decoder.py`, `oam_visualizer.py`

5.2 Instalar Dependencias Python Adicionales

Si hay dependencias adicionales en el archivo `requirements.txt`:

```
1 pip3 install -r requirements.txt
```

6 Paso 5: Instalar Bloques OAM en GNU Radio

6.1 Ejecutar el Script de Instalación

Para usar la interfaz gráfica de GNU Radio Companion, instalar los bloques personalizados:

```
1 chmod +x install.sh
2 ./install.sh
```

Acciones del script:

1. Verifica que GNU Radio esté instalado
2. Crea el directorio ~/.grc_gnuradio/blocks/ si no existe
3. Copia los archivos .block.yml a dicho directorio
4. Configura PYTHONPATH para que GNU Radio encuentre los módulos

6.2 Verificar Instalación de Bloques

Comprobar que los bloques se instalaron correctamente:

```
1 ls ~/.grc_gnuradio/blocks/oam_*.block.yml
```

Debe mostrar 5 archivos:

- oam_source.block.yml
- oam_encoder.block.yml
- oam_channel.block.yml
- oam_decoder.block.yml
- oam_visualizer.block.yml

7 Paso 6: Ejecutar el Sistema

7.1 Método 1: Ejecución Python Directa (Recomendado)

La forma más rápida de ejecutar el sistema es directamente desde Python:

```
1 cd /path/to/OAM
2 python3 oam_complete_system.py
```


Ventajas:

- Ejecución rápida sin overhead de GUI
- Ideal para simulaciones por lotes
- Automatización y scripting facilitados

7.2 Método 2: Ejecución vía GNU Radio Companion

Para usar la interfaz gráfica (útil para presentaciones y ajustes visuales):

```
gnuradio-companion
```

Luego:

1. Ir a File >Open
2. Navegar a /path/to/OAM/oam_complete_flowgraph.grc
3. Hacer clic en el botón *Execute* (triángulo verde) o presionar F6

Nota: Si los bloques OAM no aparecen, cerrar completamente GNU Radio Companion y volver a abrirlo.

7.3 Método 3: Modo Sin Interfaz (Headless)

Para ejecutar sin ventanas gráficas (ideal para servidores):

```
python3 oam_complete_system.py --headless
```

Los resultados se guardan automáticamente en el directorio `current_run/`.

8 Paso 7: Cambiar Parámetros del Sistema

8.1 Archivo de Configuración Central

Todos los parámetros del sistema están centralizados en `oam_system_config.py`. Editar este archivo:

```
nano oam_system_config.py
```

8.2 Parámetros Principales

Modos OAM:

```
'num_oam_modes': 6, # Opciones: 2, 4, 6, 8
```

Determina cuántos modos OAM se usan. Con 6 modos: $\ell = \pm 1, \pm 2, \pm 3$.

Longitud de Onda:

```
'wavelength': 1550e-9, # metros (1550 nm telecom)
```

Valores típicos: 633 nm (HeNe), 1550 nm (telecomunicaciones).

Distancia de Propagación:

```
'propagation_distance': 50, # metros
```

Distancia entre transmisor y receptor (rango típico: 10-500 m).

Turbulencia Atmosférica:

```
'cn2': 1e-15, # m-2/3
```

Constante de estructura del índice de refracción. Valores:

- 1×10^{-17} : Débil
- 1×10^{-15} : Moderada
- 1×10^{-14} : Fuerte
- 5×10^{-14} : Severa

SNR Objetivo:

```
1 'snr_target': 30, # dB
```

Relación señal-ruido deseada (rango típico: 20-40 dB).

Resolución de Grilla:

```
1 'grid_size': 512, # p x e l e s
```

Mayor valor = mayor precisión pero más lento. Opciones: 256, 512, 1024.

9 Paso 8: Solución de Problemas

9.1 Los Bloques No Aparecen en GNU Radio

Si los bloques OAM no son visibles en GNU Radio Companion:

1. Cerrar completamente GNU Radio Companion
2. Verificar instalación:

```
1 ls ~/.grc_gnuradio/blocks/oam*.block.yml
```

3. Si no están, volver a ejecutar:

```
1 ./install.sh
```

4. Reiniciar GNU Radio Companion

9.2 Error: ModuleNotFoundError

Si aparece ModuleNotFoundError: No module named 'oam*_wr':

```
1 cd /path/to/OAM
2 export PYTHONPATH="${PYTHONPATH}:${pwd}"
3 python3 oam_complete_system.py
```

9.3 Datos Corruptos o Caché Inválido

Si hay problemas con datos corruptos, limpiar el caché:

```
1 rm -rf current_run/  
2 rm -rf /tmp/oam/
```

9.4 GNU Radio Version Incorrecta

Si la versión de GNU Radio es <3.10, actualizar:

```
1 sudo apt update  
2 sudo apt upgrade gnuradio  
3 gnuradio-companion --version
```

10 Resultados Esperados

10.1 Salida por Consola

Después de ejecutar el sistema, debe ver:

- Confirmación de mensaje transmitido
- Contador de símbolos procesados
- Mensaje decodificado recuperado
- Mediciones de SNR por símbolo
- Ubicación de datos guardados

10.2 Archivos Generados

El sistema genera datos en dos ubicaciones:

Simulación actual: current_run/

- `fields_*.npz`: Campos ópticos complejos
- `meta.json`: Metadata de la simulación
- `metrics.jsonl`: Métricas por símbolo (SNR, NCC)
- `config_hash.txt`: Hash de configuración para caché

Simulaciones archivadas: `runs/<timestamp>/`

- Misma estructura que `current_run/`
- Con marca de tiempo para comparación histórica

11 Información Adicional

11.1 Repositorio del Código

El código fuente completo está disponible públicamente en:

<https://github.com/DeibyArizac/OAM>

11.2 Documentación

La documentación completa del sistema, incluyendo fundamentos teóricos, arquitectura de software y resultados de validación, se encuentra en el trabajo de grado:

“Codificación y Decodificación a partir del Momento Angular Orbital de la Luz y GNU Radio”

11.3 Soporte y Contacto

Autor: Deiby Fernando Ariza Cadena

Email: 2195590@correo.uis.edu.co / deibyarizac@gmail.com

Institución: Universidad Industrial de Santander

Escuela: Ingenierías Eléctrica, Electrónica y de Telecomunicaciones (E3T)

*Este manual describe la instalación del sistema OAM desarrollado
como parte del trabajo de grado presentado a la
Universidad Industrial de Santander en 2025.*