CELESTE

IACTEC Space

SOFTWARE TECHNOLOGY · SPACE COMMUNICATIONS

MAKE PUS INTEGRATION SIMPLE

libPUS, a C library for ECSS Packet Utilization Standard (PUS-C)

The PUS standard lets satellite payloads speak a common language with ground control, but implementing it from scratch can take years. **libPUS** gives you all the essential logic already built: not just packet serialization, but the full behavior of each PUS service. One codebase, any mission, zero external dependencies.



ECSS-compliant

GUARANTEES INTEROPERABILITY ACROSS THE ESA ECOSYSTEM.

Flexible

ADAPTS TO ANY MISSION VIA RUNTIME CONFIGURATION.

Easy to integrate

INCLUDES READY-TO-USE EXAMPLES, WITH DIRECT SUPPORT FROM THE DEVELOPERS.

Reliable

FOLLOWS MISRA-C GUIDELINES FOR CRITICAL SYSTEMS.

iactec.space@iac.es
www.iac.es

LIBPUS · PUS C LIBRARY · IACTEC-SPACE · IAC

FROM YOUR PAYLOAD TO MISSION CONTROL

ADVANTAGES

Compatible with ESA ecosystem
PUS-C compliance unlocks interoperability with ESA missions and the broader European NewSpace ecosystem.

Fast onboarding, templates included
Working example applications (onboard_app, ground_app) serve as templates. Get integrated in hours, not weeks.

Reusable. One binary, any mission
A single compiled binary adapts to any mission at runtime via configuration. No code fork, no recompilation per project.

Full PUS logic, not just serialization
Implements the actual behavior of each service (e.g. periodic housekeeping reports), not just read/write packet functions.

Self-contained. Zero dependencies
Just a C compiler (GCC) and optionally CMake. No third-party libraries or frameworks. Not tied to any specific OS.

GETTING STARTED WITH LIBPUS

1. **Configure at runtime**
Pass your mission config structure: Services, packet formats, parameters. No recompilation needed.

2. **Receive telecommands**
Call pus_accept() and pus_exec(), **libPUS** handles the full PUS service logic automatically.

3. **Send telemetry**
libPUS calls your pus_send() callback whenever a packet is ready. Fully decoupled from the transport layer.

4. **Plug into any interface**
UART, SpaceWire, Ethernet, UNIX sockets... libPUS is interface-agnostic. Works with Yamcs and other 3rd-party software.

Supported PUS services

ST[01]	Request verification
ST[02]	Device access
ST[03]	Housekeeping
ST[05]	Event reporting
ST[06]	Memory management
ST[09]	Time management
ST[12]	On-board monitoring
ST[17]	Test
ST[18]	On-board control procedures *Linux targets only*
ST[19]	Event-action
ST[20]	Parameter management
ST[23]	File management

Covers services required by most satellite payloads.
Custom services can be added via the public API.

EUROPEAN UNION
EUROPEAN REGIONAL DEVELOPMENT FUND
A WAY TO MAKE EUROPE
GOBIERNO DE ESPAÑA · MINISTERIO DE CIENCIA, INNOVACIÓN Y UNIVERSIDADES
tenerife innova
Plan de Recuperación, Transformación y Resiliencia
CABILDO DE TENERIFE
Canarias avanza con Europa
ECAN
MEDI Marco Estratégico de Desarrollo Insular