

HStack

KNX registered TP-1 System B protocol stack, delivered as ANSI C source code

HTKNXSTK001

DOCUMENT	STACK VERSION	REVISION	ISSUE DATE	WEB
HT-DS-STK001-EN	1.0	r1.0 ENGLISH	08/2026	horizentis.com

Description

HStack is a complete KNX TP-1 protocol stack for device manufacturers. It covers every layer between the twisted pair and the application: bit level transmission and reception on the KNX bus, frame handling, addressing, transport connections, and the full System B application layer with group objects, interface objects and KNX Data Secure. A device built on HStack is commissioned from ETS like any other KNX product, with no external gateway, companion chip or licensed runtime in between.

The stack was written from scratch by Horizentis and is registered with KNX Association under registration number **867/23232/26**. Manufacturers licence the complete C source code, so the KNX layer of the product can be read, debugged and maintained by the same team that writes the rest of the firmware, for the whole life of the product.

KNX REGISTRATION	MANUFACTURER ID	DEVICE DESCRIPTOR	MEDIUM
867/23232/26	867	07B0h, System B	TP-1, TP1-256

Features

- **Complete stack** from the physical layer up to the application interface layer. No vendor library underneath and no black box object code.
- **KNX Data Secure** with tool key, group keys and point to point keys, a security interface object, and security failure reporting.
- **System B profile** (device descriptor type 0 = 07B0h), the profile ETS expects from a modern end device.
- **Extended frames** with an APDU length of 69 octets, so long property and memory transfers complete in a single telegram.
- **System mode commissioning** with ETS 5 and ETS 6: individual addressing, verify mode, load and run state machines, and four authorisation levels.
- **Extended property, memory and function property services**, which large modern application programs depend on.
- **Runs on a Cortex-M0+**. The reference target is an STM32G0B0CET6 with an onsemi NCN5110 transceiver, which keeps the bill of materials low.
- **Full C source code** under a per product licence, including the STM32G0 porting layer and the Keil project used for certification.

Architecture



Specifications

Item	Value
Product	HTKNXSTK001 HStack KNX TP-1 protocol stack, source code licence
KNX registration number	867/23232/26
KNX manufacturer ID	867 (Horizentis)
Device descriptor type 0	07B0h (System B)
KNX medium	TP-1 twisted pair, TP1-256
Device type	Server, that is a KNX end device. Router, bridge and client profiles are not implemented
Layers implemented	Physical, data link, network, transport, application and application interface
Configuration mode	System mode (S-mode), commissioned with ETS 5 and ETS 6
Frame formats	Standard frames and extended frames
Maximum APDU length	69 octets (PID_MAX_APDULENGTH)
KNX Data Secure	Tool key, group keys and point to point keys, security interface object, security failure log with report on the bus
Authorisation	4 access levels, 0 highest to 3 lowest, keys held in non-volatile memory, factory default is free access
Address table	Realisation type 7. 32 group addresses by default, set at compile time
Association table	Realisation type 6. 32 associations by default, set at compile time
Group object table	Realisation type 7
Group object sizes	1 bit to 14 octets: UINT1 to UINT8, UINT16, OCTET3, OCTET4, OCTET6, OCTET8, OCTET10, OCTET14
Group object flags	Communication, read, write, transmit, update, read on initialisation
Restart	Connectionless and connection oriented. Master reset erase codes 01h, 02h and 07h
Target microcontroller	STMicroelectronics STM32G0, Arm Cortex-M0+. Reference target STM32G0B0CET6
Transceiver	onsemi NCN5110. TP-1 bit timing and encoding run in MCU firmware
Operating system	Arm RTX (CMSIS-RTOS2)
Language and toolchain	ANSI C, built with Keil MDK uVision 5 and Arm Compiler 5
Approximate footprint	53 kB flash and 14 kB RAM on the reference target, debug build, with KNX Data Secure enabled. See the note below

NOTE: The footprint figures are measured from a debug build of the reference target and include KNX Data Secure. They are a planning guide, not a guarantee. A release build is smaller, and the figure your project reaches depends on the compiler and optimisation settings you choose and on the size of the address, association and group object tables your application declares.

Supported application layer services

Group	Services
Group communication	A_GroupValue_Read , A_GroupValue_Write , A_GroupValue_Response , sent and received, with 1:n associations
Device identification	A_DeviceDescriptor_Read type 0, A_UserManufacturerInfo_Read
Individualisation	A_IndividualAddress_Read and _Write , A_IndividualAddressSerialNumber_Read and _Write , A_SystemNetworkParameter_Read for serial number discovery
Memory	A_Memory_Read and _Write , A_MemoryBit_Write , A_MemoryExtended_Read and _Write with 24 bit addressing
Properties	A_PropertyValue_Read and _Write , A_PropertyDescription_Read , and the extended forms A_PropertyExtValue_Read , _WriteCon , _WriteUnCon , _InfoReport and A_PropertyExtDescription_Read
Function properties	A_FunctionPropertyCommand , A_FunctionPropertyExtCommand , A_FunctionPropertyExtStateRead
Device control	A_Restart , A_Authorize_Request , A_Key_Write , A_ADC_Read
Security	A_SecureService for secured group and point to point communication, and A_NetworkParameter_InfoReport for security failure reporting

Connection oriented services are also available connectionless where the KNX specification allows it. The complete service by service statement is published in the *HStack KNX Conformance Statement* (HT-CS-STK001-EN).

Hardware requirements

HStack is a software product, so the hardware around it is yours to design. The reference design that carried the stack through KNX certification is deliberately small, and any board that meets the requirements below will run the stack without changes to the KNX layers.

Item	Requirement
Microcontroller	STM32G0 series, Arm Cortex-M0+. Certified on STM32G0B0CET6
Flash and RAM	At least 64 kB flash and 24 kB RAM available to the stack and the application together, on the reference build
Timers	Two general purpose timers, one for pulse capture and one for pulse generation, both clocked at 1 MHz
Non-volatile storage	Internal flash pages reserved for the KNX tables, the security keys and the application parameters
Transceiver	onsemi NCN5110, which also supplies the board from the bus
Indicators	One programming button and one programming LED, as required by the KNX specification

NOTE: Porting to another Cortex-M microcontroller or to another KNX transceiver is possible, because the hardware dependent code sits in one clearly separated porting layer. Horizentis can quote this work as an engineering service. Note that a change of microcontroller family or transceiver requires a new KNX registration for the resulting stack.

Scope of delivery

- Complete C source code of every KNX layer, from the physical layer to the application interface layer, with the header files that form the application API.
- The STM32G0 porting layer, covering timers, flash, reset and the programming button and LED.
- The Keil MDK project used for the certification build, with the debug and release targets.
- An example application and an example ETS product database project showing how group objects and parameters are declared on both sides.
- The *HStack Integration Guide* and the *HStack KNX Conformance Statement*.

How a project runs

1. **Hardware.** Build your board around the STM32G0 and the NCN5110, or start from the Horizentis reference design.
2. **Port.** Adapt the target layer to your board. If you keep the reference pin assignment there is nothing to change.
3. **Declare.** Describe your group objects, parameters and interface objects once, and build the matching ETS product database.
4. **Apply.** Write your application against the stack API. The stack calls you when a group object is updated, and you call the stack when a value changes.
5. **Certify.** Because the stack is already registered, your device certification covers your application and your hardware, not the KNX protocol implementation.

Licensing

HStack is licensed per product to the manufacturer, and the licence covers the source code, the right to build it into that product, and the right to maintain it. Volume terms, multi product terms and the support and update options are quoted per project. The KNX registration stays with Horizentis, and the licence grants your device the benefit of it, so your own KNX certification does not have to repeat the protocol conformance tests.

IMPORTANT: A KNX manufacturer ID and a device certification of your own are still required to place a KNX product on the market. HStack removes the protocol stack from that effort, not the device certification itself.

Further documentation

For the porting layer, the application API and the ETS product database workflow, refer to the *HStack Integration Guide* (HT-IG-STK001-EN). For the service by service conformance statement, refer to the *HStack KNX Conformance Statement* (HT-CS-STK001-EN). Both are available at horizentis.com.

NOTE: Horizentis reserves the right to change the specifications in this document without notice, as the stack is maintained against the current KNX specification. The revision that applies to a licence is the one named in the licence agreement.