

SHADE Protocol

Specification

Date: 04.08.2025

Protocol version: 1.3

Document revision: 1

Introduction

Simple Header Arbitrary Data Exchange protocol (“shade” from now on) is a data exchange protocol created to exchange data and multiplex data packets between multiple sessions and recipients over character devices such as UART, while keeping minimal overhead. The shade packets have highly configurable structure, and may include various advanced features. The protocol provides diverse communication capabilities, such as discovery, numeric code communication, payload data communication, session and session stage management and packet checksum transmission, while maintaining low overhead for high-throughput or low bandwidth systems. The unused features of the header can be removed from the packet, and features with limited use can utilize minimal space. Packet header section sizes are configurable. The smallest valid data packet is 2 bytes and is a packet header without payload data (contains 1 byte of potentially useful data). The smallest packet header overhead of a packet with payload data is 3 bytes (1-256 bytes of payload data), 4 bytes (1-65536 bytes of payload data) or 6 bytes (1-4294967296 bytes of payload data). The maximum packet header size with full feature set is 16 bytes.

Data Packet Header Structure

The data packet header can have the following fields, in the order of appearance:

PFC – Packet Flow Control. 1 byte. Always the first byte of the message. The only field with fixed size and location in the packet. Describes the byte structure of the rest of the header, also called “Mode”. See detailed description of PFC below.

PSID – Packet Session ID. 0 bytes or 1 byte. Optional field intended to help multiplexing the message between multiple potential consumers (sessions). Available only in Long Header Mode. Not present in Short Header Mode.

PTIDL – Packet Type ID Low Byte. 1 byte. Mandatory field intended to contain packet type to signal an event or describe the data in the packet. Can be used as 1-byte payload data field.

PTIDH – Packet Type ID High Byte. 0 bytes or 1 Byte. Optional high byte counterpart of PTIDL. Available only in Long Header Mode. Not present in Short Header Mode.

PISC – Packet In-Session Counter. 0 bytes, 1 byte, 2 bytes or 4 bytes. An optional field intended to describe in-session communication stage, so the sender can specify for what specific stage of communication session the packet is intended.

PDS – Payload Data Size. 0 bytes, 1 byte, 2 bytes, 4 bytes. An optional field that describes the payload data size. Must be absent from the packet, if there is no payload. If present in the packet, payload must be present. For payload size N, the raw value of PDS contains N-1.

PCS – Packet Checksum. 0 bytes, 1 byte, 2 bytes, 4 bytes. An optional field intended to contain the checksum of the entire packet, including header and payload data. The field should be treated as all zeroes for the purpose of packet checksum calculation. Checksum algorithm is not specified.

The header is optionally followed by the payload data.

All multi-byte header values have little-endian byte order.

Packet Flow Control

Packet Flow Control (PFC) is the only mandatory packet field with fixed position and size. It's always the first byte of any message, and contains the description of header structure.

Bit Fields of PFC:

Bits[0..1]: PISC. Packet In-Session Counter (PISC) size.

- Value 0x00: 0 bytes
- Value 0x01: 1 byte
- Value 0x02: 2 bytes
- Value 0x03: 4 bytes

Bits[2..3]: PDS. Payload Data Size (PDS) size.

- Value 0x00: 0 bytes – No payload data in the packet
- Value 0x01: 1 byte – Payload data present
- Value 0x02: 2 bytes – Payload data present
- Value 0x03: 4 bytes – Payload data present

Bits[4..5]: PCS. Packet Checksum (PCS) size.

- Value 0x00: 0 bytes
- Value 0x01: 1 byte
- Value 0x02: 2 bytes
- Value 0x03: 4 bytes

Bit[6]: SHMEN. Short Header Mode Enable. Removes Session ID from the packet header, reduces Packet Type ID from 2 bytes to 1 byte.

- Value 0x00: Short Header Mode disable, Long Header Mode
- Value 0x01: Short Header Mode enable

Bit[7]:

- Reserved. Keep it 0x00.

Optional fields can be used as dummies to force payload data alignment to allow more efficient data access and advanced in-place data processing.

The combination of PFC flags is called a “Mode”. A mode name looks like MODE_ABCD, where:

- A: Short Header Mode Enable (On = 1, Off = 0)
- B: A value from the set {0, 1, 2, 4}, which describes PISC size
- C: A value from the set {0, 1, 2, 4}, which describes PDS size
- D: A value from the set {0, 1, 2, 4}, which describes PCS size

For example, MODE_1224 means Short Header Mode, 2 bytes Packet In-Session Counter, 2 bytes Payload Data Size (payload present), 4 bytes Packet Checksum. MODE_0100 means Long Header Mode, 1 byte Packet In-Session Counter, no Payload Data Size (and no payload), no checksum.

Reserved Packet Type ID

Shade reserves 16 highest Packet Type ID values for Short Header Mode and Long Header Mode for service needs. The reserved Packet Type ID are:

SHADE_PTID_ACK	0xFFFF0	SHADE_PTID_SHM_ACK	0xF0
SHADE_PTID_NACK	0xFFFF1	SHADE_PTID_SHM_NACK	0xF1
SHADE_PTID_ SHADE_VERSION_REQUEST	0xFFFF2	SHADE_PTID_SHM_ SHADE_VERSION_REQUEST	0xF2
SHADE_PTID_ SHADE_VERSION_REPLY	0xFFFF3	SHADE_PTID_SHM_ SHADE_VERSION_REPLY	0xF3
SHADE_PTID_ PACKET_REPEAT_REQUEST	0xFFFF4	SHADE_PTID_SHM_ PACKET_REPEAT_REQUEST	0xF4
SHADE_PTID_ PACKET_MODE_DENIED	0xFFFF5	SHADE_PTID_SHM_ PACKET_MODE_DENIED	0xF5
SHADE_PTID_ SUPPORTED_MODE_REQUEST	0xFFFF6	SHADE_PTID_SHM_ SUPPORTED_MODE_REQUEST	0xF6
SHADE_PTID_ SUPPORTED_MODE_REPLY	0xFFFF7	SHADE_PTID_SHM_ SUPPORTED_MODE_REPLY	0xF7
SHADE_PTID_ MAX_MSG_LENGTH_REQUEST	0xFFFF8	SHADE_PTID_SHM_ MAX_MSG_LENGTH_REQUEST	0xF8

SHADE_PTID_ MAX_MSG_LENGTH_REPLY	0xFFF9	SHADE_PTID_SHM_ MAX_MSG_LENGTH_REPLY	0xF9
-------------------------------------	--------	---	------

SHADE_PTID_SHADE_VERSION_REPLY (and its SHM counterpart) has 2 bytes of payload data. Byte[0] contains the major version of SHADE, Byte[1] contains the minor version of SHADE.

SHADE_PTID_SUPPORTED_MODE_REPLY (and its SHM counterpart) has 1 or more bytes of payload data, that contain supported modes in order of preference. Maximum total size of the packet is 16 bytes. If all modes are supported, the mode value is 0xFF. If there are preferred modes, but all other modes are also supported, first come the preferred modes, then 0xFF. If only a subset of modes is supported, but not all modes, only these specific modes are sent without 0xFF.

SHADE_PTID_MAX_MSG_LENGTH_REPLY (and its SHM counterpart) has 4 bytes of payload data, which form a little-endian 32-bit value with the maximum number of bytes the party can receive at a time (RX buffer size of the party, which has to fit both header and payload).

Shade Compatibility Requirements

Every shade-compatible party is guaranteed to be able to receive at least 16 bytes as a single message (total message size, including header and payload).

Every shade-compatible device is guaranteed to be able to reply to the following MODE_0000 requests in MODE_0010 format:

Incoming request (MODE_0000)	Outgoing reply (MODE_0010)
SHADE_PTID_ SHADE_VERSION_REQUEST	SHADE_PTID_ SHADE_VERSION_REPLY
SHADE_PTID_ SUPPORTED_MODE_REQUEST	SHADE_PTID_ SUPPORTED_MODE_REPLY
SHADE_PTID_ MAX_MSG_LENGTH_REQUEST	SHADE_PTID_ MAX_MSG_LENGTH_REPLY

Beyond the MODE_0010 replies to the listed requests, the device doesn't have to support full MODE_0000 and MODE_0010 communication. It must only be able to reply to these specific requests.

If the device receives a message of unsupported mode, it must be able to reply with the same Long or Short header mode, as the incoming message:

- Unsupported Long Header Mode message:
SHADE_PTID_PACKET_MODE_DENIED using MODE_0000
- Unsupported Short Header Mode message:
SHADE_PTID_SHM_PACKET_MODE_DENIED using MODE_1000

The exchange of shade compatibility requests and replies comes with PSID (packet session id) set to 0.

Discovery And Synchronization

A typical handshake procedure is not fixed in sequence and is not obligatory (especially if the capabilities of the parties are known in advance). In case the handshake is desired, it consists of one or two devices querying each other about capabilities using MODE_0000 to request information and MODE_0010 to respond to these requests. Typically, the devices would request shade version, maximum RX buffer length of each party and the supported modes.

Appendix

Revision history

Revision	Changes
1	<ul style="list-style-type: none">Published version 1 of the document for the shade protocol version 1.3