BC7215

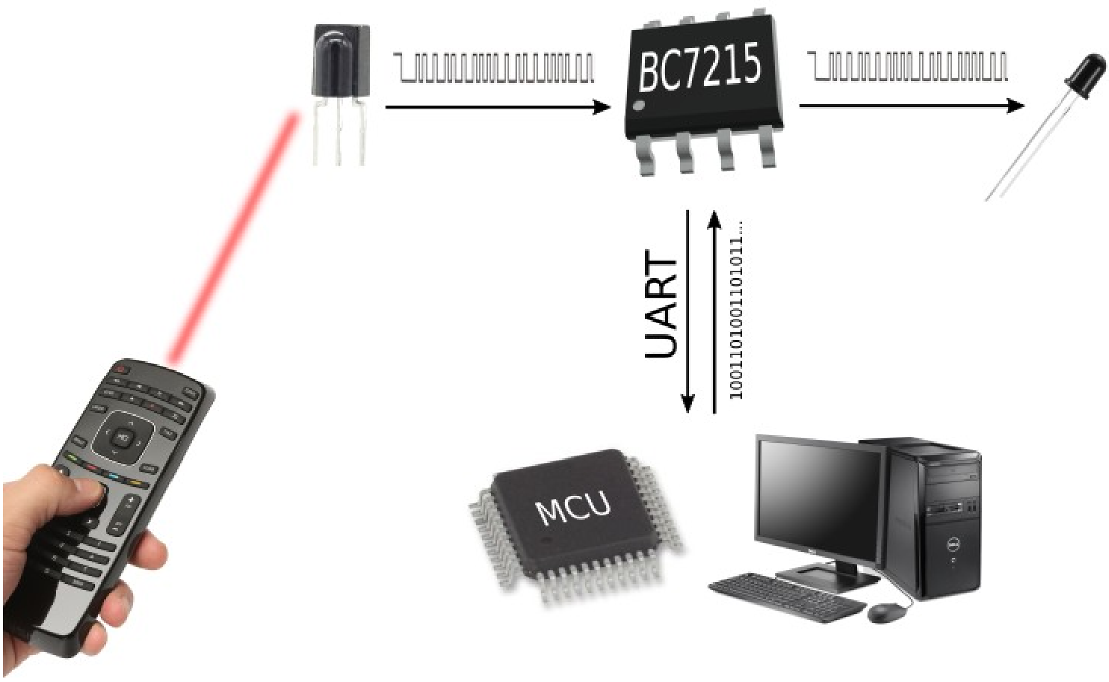
IR Communication And Universal Remote Control Decoder/Encoder

The BC7215 chip serves as a comprehensive solution for encoding (transmitting) and decoding (receiving) infrared remote control signals, supporting an extensive range of common infrared remote formats. It boasts an automatic recognition system for infrared signal encoding formats, ensuring compatibility with over 99.5% of air conditioner remotes and more than 98% of audiovisual device remotes. By integrating both decoding and encoding functionalities within a single unit, the BC7215 streamlines the process of both interpreting and generating infrared signals.

Distinctly, the chip segregates the modulation format information from the encoded raw data. In its decoding operation, it directly outputs raw data, whereas for transmission, it encodes the data in real-time based on the input provided by the user. This efficient processing ensures minimal data exchange with the main system, making the BC7215 an ideal choice for compact systems with constrained resources. Its capability to handle raw data input and output directly also positions it as a viable option for low-speed infrared data communication tasks.

Equipped with a UART interface, the BC7215 facilitates straightforward integration with a variety of microcontrollers, embedded systems, as well as Android, Linux, and Windows platforms, eliminating the necessity for intricate low-level driver implementation or I/O port manipulation.

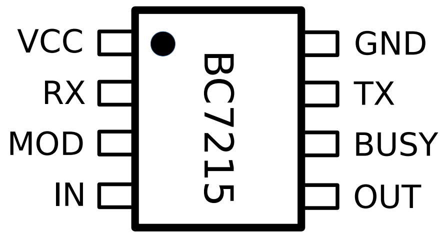
Given its versatile functionalities, the BC7215 finds application across a broad spectrum of products and sectors, including IoT smart homes, universal learning remote controls, generic infrared controllers, data collection mechanisms, and platforms requiring low-speed infrared data transmission.



Features

- Automatically recognises infrared remote control signal formats
- Single chip receiving (decoding) and transmitting (encoding).
- Suitable for bidirectional low-speed data communication
- Replicating remote control signals by simply sending back the decoded raw data
- UART interface
- Built-in carrier generator, supports 37.5k and 56k carriers
- 7uA@3.3Vstandby mode

Pin Diagram



Index

Pin Descriptions

Pin No.	Pin Name	Description
1	VCC	Power in, voltage range 2.7-5.5V
2	RX	UART receiving
3	MOD	Mode selection: When MOD is high, it is in receiving/decoding mode; when MOD is low, it is in encoding/sending mode.
4	IN	IR signal in, usually connected to the output of IR receiver module
5	OUT	Infrared driving signal output, low effective. Depending on the settings, the output can be or not be modulated with the carrier frequency. Max driving capability 90mA.
6	BUSY	Busy indicator, set high during the decoding and when the UART buffer is full during sending
7	TX	UART transmitting
8	GND	Ground Pin

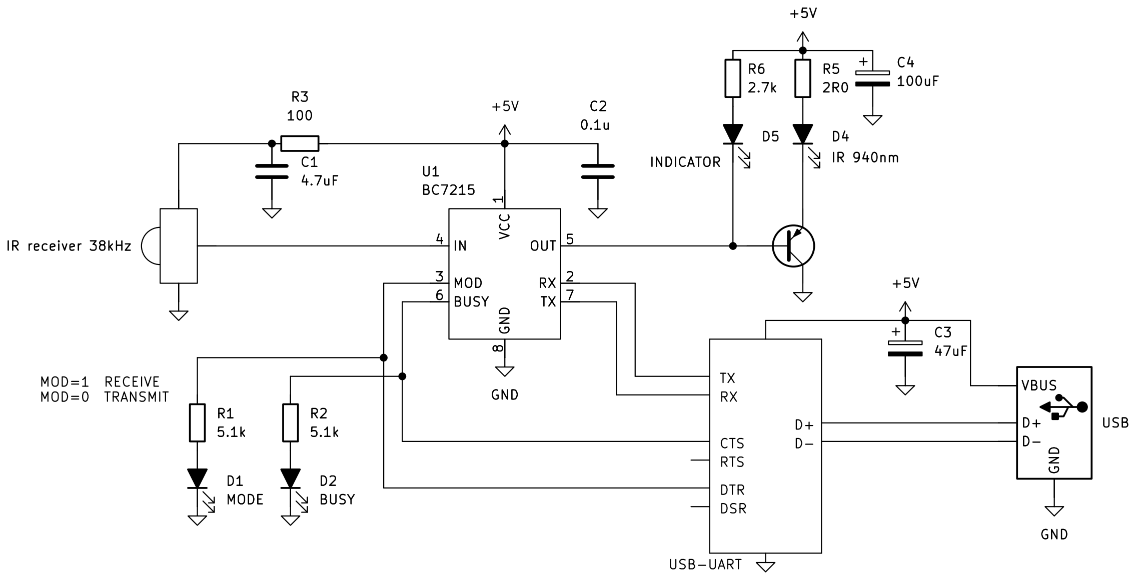
Operating Modes

The BC7215 features two primary modes of operation: receiving (decode) mode and transmission (encode) mode, dictated by the state of the MOD pin. Herein, these modes will be consistently referred to as "receiving mode" and "transmission mode." The chip enters receiving mode when MOD is high and switches to transmission mode when MOD is low. Additionally, in transmission mode, commands can be issued to place the BC7215 into a standby state.

Users have the flexibility to change modes at any juncture. It's crucial to note, however, that altering modes during an ongoing receive or transmit process will halt the current activity. The mode transition period requires consideration: switching from receiving to transmission mode can be as quick as 1ms, whereas moving from transmission back to receiving mode may extend up to 18ms. This delay should be accounted for in programming since attempting to send UART commands to the BC7215 before completing the transition could lead to data loss or incorrect command execution. Consequently, it's advisable to refrain from sending data to the BC7215 immediately after a MOD pin level change until the transition fully concludes.

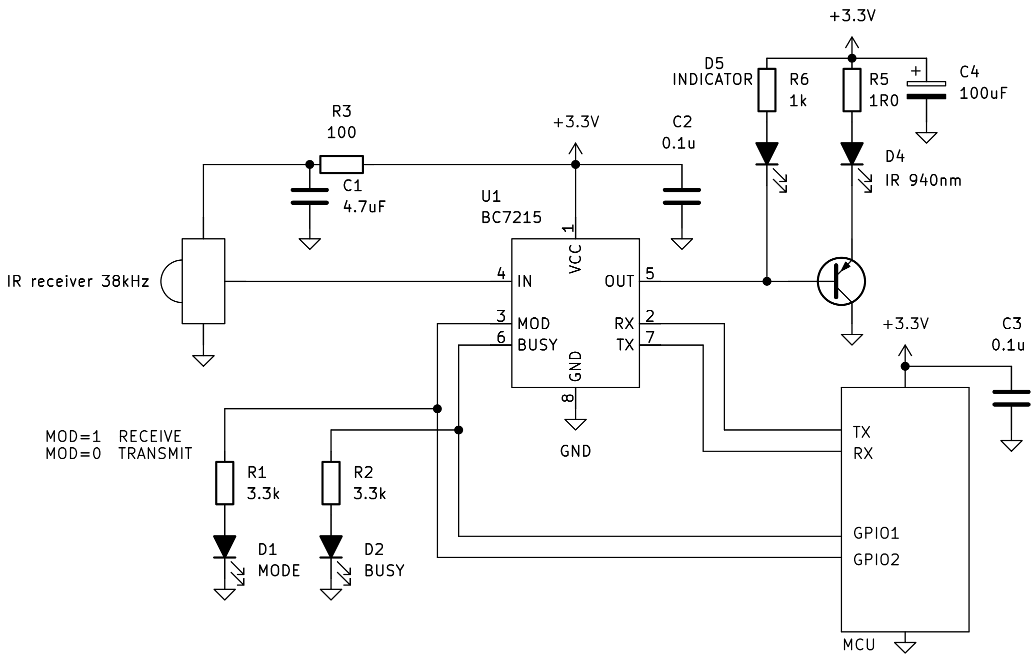
Typical Applications

Typical Application 1 (PC)



(When connected to a computer, please ensure that your device driver supports the hardware flow control of the USB-UART chip. Otherwise, the BC7215's UART buffer might overflow, causing the failure of long frame sending.)

Typical Application 2 (MCU)



Infrared LEDs generally require a relatively high peak current to achieve longer transmission range. Many infrared LEDs can withstand peak currents of up to 1A. During transmission, the average working current of an infrared LED (with carrier) can exceed 200mA. This significant current demand should be carefully accommodated in the design of the power supply.

UART

The BC7215's UART interface operates with non-configurable settings: it uses a baud rate of 19200, 8 data bits, 2 stop bits, and no parity check. Users must verify the status of the BUSY pin before transmitting data to the BC7215. Attempting to transmit data while BUSY is high will lead to data loss. When interfacing the BC7215 with a computer, it's feasible to leverage the computer's hardware flow control capabilities by utilizing the BUSY signal as the CTS (Clear To Send) signal. This arrangement allows the BC7215 to manage data transmission seamlessly.

The BC7215 dispatches data via UART under two specific conditions:

1. **In Receiving Mode:** Upon detecting a valid infrared (IR) signal, the BC7215 outputs the decoded data through UART, emitting 8 bits at a time. From the onset of the first byte's output, the BUSY signal is activated (set high) to prevent any incoming data transmissions to the BC7215. The BUSY signal reverts to low after the decoding process concludes and the terminator is dispatched. In this mode, the BUSY signal acts as an indicator for the controller to identify the reception of decoded IR data.
2. **In Transmission Mode:** Following the completion of data transmission for a sending command, the BC7215 issues a terminating 0x7A byte (a single byte) to signal the end of transmission to the controller. Additionally, upon receiving a shutdown command, the BC7215 confirms receipt by sending back a terminator 0x7A byte.

The UART data output functionality of the BC7215 is managed internally, based on the chip's operational status, without external intervention.

The BC7215 features a UART reception section equipped with a 16-byte buffer for incoming data. In scenarios where the chip's data processing rate is outpaced by the rate of UART data reception—such as during infrared (IR) transmission activities characterized by inherently slow data transfer rates—a prolonged data frame may lead to a fully occupied UART buffer. Under these circumstances, the BUSY signal is asserted (set to high), indicating that the buffer capacity has been reached. As the chip proceeds to transmit data via the OUT port, thereby freeing up buffer space, the BUSY signal is deasserted (reset to low), signaling readiness to accommodate additional incoming data.

Byte Stuffing Encoding

The BC7215 utilizes byte stuffing encoding for processing both received and transmitted UART data. In the UART data stream of the BC7215, the byte 0x7A (representing the ASCII character 'z') is designated as a special marker, signaling either the conclusion of data transmission or a reset command. In receive mode, this character marks the end of a data packet, serving as a clear endpoint indicator for users. Conversely, in transmit mode, the BC7215 issues a 0x7A byte as a termination signal and confirmation before shifting into shutdown mode. When the host system sends a 0x7A to the BC7215, it prompts the chip to halt its current activity and reset to a state ready for new instructions.

The 0x7A byte is excluded from appearing within actual data or format packets. Should a 0x7A value be present in the packet's data content, it necessitates substitution. The substitution rule is straightforward: a 0x7A byte in the original data is replaced with the sequence 0x7B 0xFA, and a 0x7B byte is replaced with 0x7B 0xFB.

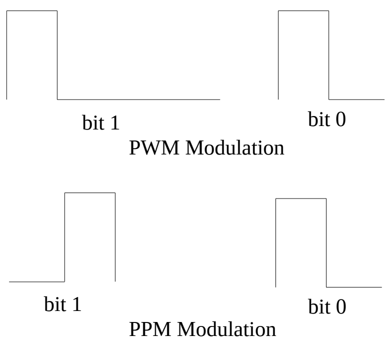
Reversing the byte stuffing process is equally uncomplicated: encountering a 0x7B within the data prompts the removal of the 0x7B byte and modification of the next byte's highest bit to 0.

For applications involving straightforward replication of remote control signals, given that both the output from and input to the BC7215 are subject to byte stuffing encoding, users can bypass the decoding step and directly relay the BC7215's output back into the device. However, for data communication or analysis applications, it's essential to revert to the original data format by undoing the byte stuffing encoding. Similarly, data being sent to the BC7215 must be prepared using byte stuffing encoding.

Note that byte stuffing encoding has the potential to expand the data volume, with each instance of a 0x7A byte in the original text incrementing the total data count by one byte.

Infrared Encoding Techniques

Remote controls typically utilize unique encoding formats that broadly fall into two modulation categories: Pulse Width Modulation (PWM) and Pulse Phase Modulation (PPM, also known as Bi-Phase Modulation), with NEC and RC5 being the respective hallmark formats for these methods.



Data Transmission Sequence and Modulation

The method of modulation directly influences the sequence in which data is transmitted. Infrared (IR) signals are transmitted serially, with variations in the order of bit significance—either transmitting the Most Significant Bit (MSB) first or the Least Significant Bit (LSB) first. The BC7215 approaches the decoding of Pulse Width Modulation (PWM)-encoded IR signals by starting with the LSB, in accordance with the NEC code standard prevalent in remote controls. In contrast, Phase Pulse Modulation (PPM)-encoded signals are decoded starting with the MSB, adhering to the protocol of the widely utilized RC5 encoding standard. For additional information, please refer to the "Data Bit Alignment" section.

Pulse Duration Interpretation and Encoding Formats

Encoding formats in PWM mode may vary in their interpretation of pulse lengths, with longer pulses signifying either '0' or '1'. The BC7215 standardly designates shorter pulses to represent '0' and longer pulses for '1', as illustrated previously.

Significance of Bits and Bytes in Remote Control Encoding

Various remote control encoding formats allocate specific meanings to particular bits or bytes, such as address codes and control codes. For instance, bytes in air conditioner remotes may indicate the set temperature. The BC7215 processes all bits and bytes without differentiation, leaving the task of interpreting any specialized data significances to the user. This requires users to acquaint themselves with the specific encoding format of their remote.

Signal Termination Detection

The BC7215 identifies the end of an IR signal through time intervals. If the level on the IN pin remains unchanged for more than 18ms, it indicates the signal's end, prompting the initiation of a new decoding cycle for subsequently received signals.

Receiving Mode

During reception, the BC7215 outputs two distinct types of data packets: decoded raw data and infrared encoding format information. Each packet type concludes with a terminator; raw data packets terminate with a single 0x7A byte, while format information packets close with two successive 0x7A bytes. The BC7215 offers two UART output modes in reception: Simple and Composite. The Simple Mode provides only raw data alongside minimal supplementary information. In contrast, Composite Mode delivers a format information packet immediately following the raw data packet.

Simple Mode

Upon power-up, the BC7215 is set to operate in Simple Mode by default. In this mode, when a valid infrared (IR) signal is detected, the chip outputs raw data at a rate of one byte for every eight bits of the received signal. After the completion of the signal reception, the BC7215 outputs an additional three bytes of data, forming a complete raw data packet, which is then terminated with a 0x7A byte.

Format of Raw Data Packets

The length of raw data packets varies depending on the amount of received data, with potential payload sizes ranging from 1 to 512 bytes. Due to the relatively low transmission speeds characteristic of infrared remote encoding formats—capable of transmitting merely tens of bytes per second—transmissions involving larger volumes of data are subject to prolonged durations. This increased transmission time can elevate the susceptibility to external interferences, thereby affecting transmission reliability. Typically, the length of raw data packets from remote controls is kept within a few bytes to several tens of bytes to mitigate these risks. Each raw data packet is finalized with a 0x7A terminator byte, marking the packet's end. The format of these raw data packets is as follows:

Raw data packet				Terminator
Raw Data(1-512 bytes)	status-signature(1 byte)	Lower byte of bit length(1byte)	Higher byte of bit length(1 byte)	0x7A(1 byte)

Status-signature Byte Overview

The status-signature byte, located immediately before the final two bytes preceding the terminator, encapsulates key information about the infrared signal's decoding status and its fundamental characteristics. The most critical data is held within the upper four bits, designated as follows:

b7: ERR	b6: REV	b5: TP_1	b4: TP_0	b3	b2	b1	b0

b7: ERR - An error flag set to 1 signifies a failure to decode the received signal correctly.

b6: REV - An inversion flag that, when set to 1, indicates the data within the raw data segment is presented in inverted form, with 0xFF symbolizing an actual original data value of 0x00.

b5:b4: TP1:TP0 - These bits specify the encoding type. A value of 11 for TP1:TP0 denotes a PWM (Pulse Width Modulation) format with LSB-first decoding. Other values indicate a PPM (Pulse Phase Modulation) format with MSB-first decoding.

Raw Data Bit Length

The raw data bit length in the BC7215 is represented by a 16-bit integer spanning two bytes, with the least significant byte positioned first. This value can extend up to 0xFFF. It is important to note that the actual bit count might not always fall on octet (8-bit) boundaries. This aspect is particularly relevant for various remote control protocols that do not adhere to octet-aligned bit lengths. For example, the RC5 standard utilizes a 14-bit structure for its signals. The BC7215 requires that the raw data for any decoded remote control signal comprises a minimum of 8 bits.

Data Bit Alignment

Pulse Width Modulation (PWM) Signals

In the context of Pulse Width Modulation (PWM) modulated infrared signals, where the encoding type (indicated by TP1:TP0 bits in the signature byte) is set to 11, the BC7215 employs a Least Significant Bit (LSB) first strategy for aligning the data. This alignment principle ensures that bits received earlier are placed towards the lower end of the output byte sequence. A prime example of this alignment method in action is observed in the transmission protocol of the uPD6121 chip, which adheres to the NEC format, dictating a specific sequence for the raw data.

Transmitting sequence:

C_0	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_0'	C_1'	C_2'	C_3'	C_4'	C_5'	C_6'	C_7'	D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	$\overline{D}_0$	$\overline{D}_1$	$\overline{D}_2$	$\overline{D}_3$	$\overline{D}_4$	$\overline{D}_5$	$\overline{D}_6$	$\overline{D}_7$

Data transmission starts with C_0, with C_0-C_7 representing the lower 8 bits of the user code, C0'-C7' representing the upper 8 bits of the user code, D_0-D_7 being the key code, and $\overline{D}_0$-$\overline{D}_7$ also serving as the complement of the key code. The first four bytes of the raw data packet output by the BC7215 will be:

1st byte								2nd byte								3rd byte								4th byte							
b7	b6	b5	b4	b3	b2	b1	b0	b7	b6	b5	b4	b3	b2	b1	b0	b7	b6	b5	b4	b3	b2	b1	b0	b7	b6	b5	b4	b3	b2	b1	b0
C_7	C_6	C_5	C_4	C_3	C_2	C_1	C_0	C_7'	C_6'	C_5'	C_4'	C_3'	C_2'	C_1'	C_0'	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	$\overline{D}_7$	$\overline{D}_6$	$\overline{D}_5$	$\overline{D}_4$	$\overline{D}_3$	$\overline{D}_2$	$\overline{D}_1$	$\overline{D}_0$

The total bit count is 32, therefore the last 3 bytes of the packet will be 0x34 0x20 0x00.

When the total bit count of raw data deviates from an exact multiple of eight (an octet), the final byte within this data will contain a mix of valid and potentially invalid bits. The valid bits commence from the least significant bit (LSB) of the last byte, populating upwards towards the more significant bits. A notable example of this scenario is observed with the TC9148 chip, which produces a total of 12 bits transmitted in a specific sequence:

C1	C2	C3	H	S1	S2	K1	K2	K3	K4	K5	K6

Commencing with C1, the data transmission results in the initial duo of bytes in the BC7215's raw data output embodying the sequence:

1st byte								2nd byte							
b7	b6	b5	b4	b3	b2	b1	b0	b7	b6	b5	b4	b3	b2	b1	b0
K2	K1	S2	S1	H	C3	C2	C1	-	-	-	-	K6	K5	K4	K3

Given the 12-bit total, the final representation within the raw data packet manifests as three bytes: 0x3D, 0x0C, and 0x00, with dashes indicating placeholders for non-significant, arbitrary values. This approach underscores the method for aligning data bits that do not neatly fit into byte-wide segments, ensuring clarity in distinguishing between meaningful and filler data within the packet structure.

Pulse Phase Modulation (PPM) Signals

For Phase Pulse Modulation (PPM) signals, identified by any TP1:TP0 combination in the signature byte other than 11, the BC7215 adopts a Most Significant Bit (MSB) first strategy for decoding. This approach places bits received earlier in the higher-order positions of the output byte.

Consequently, if the last byte is not completely filled with 8 bits, the valid bits will be aligned towards the leftmost side of the byte. A case in point is the use of the RC5 format by the SAA3010 chip, which entails a configuration of 14 bits as follows:

- Two leading '1' bits
- A control (toggle) bit
- A system code of 5 bits
- A command code of 6 bits

1	1	CTL	SYS4	SYS3	SYS2	SYS1	SYS0	CMD5	CMD4	CMD3	CMD2	CMD1	CMD0

Given this structure, RC5 encoding begins with two leading '1' bits, succeeded by a series of specifically ordered control, system, and command bits. Adhering to an MSB-first decoding logic, the BC7215's output two bytes:

1st byte								2nd byte							
b7	b6	b5	b4	b3	b2	b1	b0	b7	b6	b5	b4	b3	b2	b1	b0
1	1	CTL	SYS4	SYS3	SYS2	SYS1	SYS0	CMD5	CMD4	CMD3	CMD2	CMD1	CMD0	-	-

Accumulating to a 14-bit total, the concluding segment of the raw data packet is hence captured as three bytes: 0x05, 0x0E, 0x00.

Composite Mode

In Composite Mode, after the BC7215 outputs a complete raw data packet, it immediately follows up with a format information packet. This packet contains comprehensive details about the encoding format of the currently decoded infrared signal.

Format Information Packet

The BC7215 issues format information packets, uniformly sized at 33 bytes, which are terminated with a pair of 0x7A bytes to denote the conclusion of the packet. The format packet contains all the modulation information needed to replicate the received infrared signal.

Format packet		Terminator
signature (1 byte)	format information (32 bytes)	0x7A 0x7A (2 bytes)

The signature byte closely aligns with the signature byte found within the raw data packet but omits the two most significant bits that indicate decoding status. Combined with the encoding format information, this provides all essential details necessary for the reconstruction of the received infrared remote control signal.

With each new reception of a remote signal, the format information is updated. When the BC7215 detects more than four infrared pulses, it initiates a new decoding process and overwrites any previously stored format information. Given the variability in factors such as the receiver's physical gain, sensitivity, infrared intensity, and external disturbances, signals from the same remote control may exhibit slight variations upon each reception by the BC7215. Consequently, the format information generated may vary from one reception to another.

Therefore, the format information packet should not be used to determine if two remote control signal formats are identical. Its primary function is to facilitate the restoration of the received infrared signal's format, enabling users to create infrared remote control signals with custom data.

It's important to note that during the transmission of the format information packet, the BUSY signal is activated, preventing the reception of UART data. Additionally, to preserve the integrity of the format information, the reception of infrared signals is also suspended during this phase. Continuous transmissions with intervals less than 36ms (comprising the BC7215's designated infrared signal gap and the format packet transmission duration) may lead to the loss of partial data frames and decoding inaccuracies.

Receiving Mode Control

In receive mode, when the BUSY signal is not high, users can control the decoding operation mode of the BC7215 by sending a control word via UART. The control word consists of a single byte. Upon receiving a control word, the BC7215 does not issue any acknowledgment signal. Any new control word received will overwrite the previous settings, with the most recently received control word taking precedence and being the one that influences the chip's behavior.

The upper 6 bits of the control word are reserved and can be set to any value without affecting the operation. The significant operational settings are determined by the lower 2 bits of the control word, dictating the specific decoding behavior of the BC7215.

b7	b6	b5	b4	b3	b2	b1	b0
-	-	-	-	-	-	INV	FMT

FMT Output Control

The FMT bit switches between simple and composite output modes. In its default state (bit 0), the chip is in simple mode and outputs only raw data for each received signal. Setting this bit to 1 activates composite mode, adding a format information packet after the raw data. This setting remains until changed by a new control command.

INV Decoding Priority

The INV bit adjusts decoding attempts between PWM and PPM to manage signals with similar features that might confuse the decoder. The BC7215 initially leans towards PWM decoding. If PWM results are incorrect, indicated by an error in the output (ERR bit in the status byte is set), toggling INV to 1 on the next signal reception makes the chip try PPM decoding first, potentially fixing decoding issues for signals that were problematic in PWM mode.

This INV setting is temporary, affecting only the next received signal before defaulting back to PWM preference, ensuring a balance between flexibility in handling various signal types and maintaining decoder simplicity.

Transmission Mode

Transmission mode refers to the infrared sending functionality of the BC7215. In this mode, the BC7215 encodes and transmits data received via UART according to the infrared encoding format specified in the format information packet. The format of the transmitted infrared signal is wholly determined by this format information; by reloading a complete format information packet into the BC7215, it can transmit infrared signals in the corresponding format. To replicate a recently received infrared signal, one can switch to transmission mode immediately after reception, bypassing the format information download step since the format data already exists within the chip. Inputting the raw data to be transmitted allows for sending the infrared signal in its original format.

In the infrared transmission mode, the data length for transmission is flexible and need not match the original signal length, making the BC7215 suitable for data transmission applications where variable data lengths are required.

While in transmission mode, the BC7215 does not receive infrared signals, eliminating the concern of incoming signals overwriting format information stored in memory. The stored format information remains intact until a new format information packet is received or the chip switches to receiving mode.

During transmission, the BC7215 recognizes four types of commands: format information download, transmission, reset, and shutdown commands. The format information download and transmission commands are 16-bit (two-byte) commands, whereas the reset command is a single 0x7A character. Both the format information download and transmission commands must be byte-stuffed to avoid the inclusion of the 0x7A character. Any data received by the BC7215 in standby mode that does not match these four commands will be ignored.

Format Information Download Command F6 01

By sequentially sending 0xF6 0x01, the BC7215 enters a state to receive a format information packet. The following 33 bytes received are written into the chip's format information memory.

Download command		Format packet	
0xF6(1 byte)	0x01(1 byte)	signature(1 byte)	format(32 bytes)

The format information you download is the format packet that was sent during the composite output mode in receiving mode, minus the 0x7A end marker. You're allowed to change the top two bits of the signature byte as needed. In format packets output during receiving mode, these top two bits of the signature byte are always set to 0. However, in the format packets downloaded during transmission mode, these two bits serve specific control functions. The signature byte in the format information download command includes:

b7: NOCA	b6: C56K	b5: TP_1	b4: TP_0	b3	b2	b1	b0

- **b7 (NOCA):** When set to 1, the signal output on OUT to the infrared LED does not include a carrier, just alternating high and low levels, which is useful for driving a custom carrier circuit or for other specific applications.
- **b6 (C56K):** Setting this bit to 1 switches the output carrier frequency from 37.5KHz to 56KHz, catering to different transmission requirements.

Infrared Send Command F5 02

When the BC7215 is waiting and receives the sequence 0xF5 0x02, it switches to infrared sending mode. The structure of this send command is outlined below:

Send command		Bit length of data to be sent		Data
0xF5 (1 byte)	0x02 (1 byte)	lower byte of bit length (1 byte)	higher byte of bit length (1 byte)	data to be sent

Upon receiving the first byte of the data set to be transmitted, the BC7215 immediately begins transmitting an infrared signal from the OUT port, based on the encoding format stored internally. The BC7215 is equipped with a 16-byte UART buffer. Given the generally low transmission rates of infrared signals, approximately several tens of bytes per second, subsequent data is buffered awaiting transmission once infrared sending starts. If the data to be sent exceeds 16 bytes (128 bits), the buffer may fill up, triggering the BUSY signal to go high. It's crucial for users to check the BUSY signal before sending each byte. For computer systems, it's recommended to use BUSY as a hardware flow control signal to prevent UART buffer overflow.

If the reception buffer is emptied and more data is expected, the chip will continue to wait in its current state for additional data. If this occurs during infrared transmission, the transmission will also pause, with the carrier output either being turned off or remaining active depending on the status of the last bit that could be transmitted. Such interruptions can disrupt the timing of the transmitted infrared signal, potentially leading to reception errors.

When executing a transmission command, ensure the total number of bits sent initially matches the length of the subsequent raw data and that all data is timely fed into the chip. The format information packet does not include data length information; the length of the transmitted data depends entirely on the length specified in the send command. This flexibility allows for sending data of any length using any encoding format the BC7215 can decode correctly, facilitating data communication. When replicating infrared remote control signals, note that if the data length varies, the receiving device may refuse the signal.

After all data transmission is completed, the BC7215 emits a 0x7A terminator. This terminator can be used to determine if the infrared data has been fully sent. The BC7215 does not manage the interval between individual infrared data frames. If two transmission commands are sent to the BC7215 consecutively, the two data frames will be emitted back-to-back without a gap. Since infrared remote control signals typically rely on the interval between signals to distinguish the start and end of data, transmitting two frames consecutively could corrupt the format of both frames, making them unrecognizable to the receiver. To transmit multiple data frames consecutively, users must manage the interval between frames, waiting for the 0x7A terminator from the BC7215 after the first command, then waiting the necessary interval before sending the next transmission command. A recommended interval between infrared data frames is over 40ms, though the exact minimum value depends on the data format specifications and receiver requirements.

For replicating remote control signals, since the output data has already undergone byte stuffing encoding, the output received in receiving mode can be directly transmitted back to the BC7215. However, when sending custom data, users must ensure to apply byte stuffing encoding, avoiding the inclusion of the 0x7A character in the data packet.

Infrared transmission commands can be sent continuously without a gap following a format information download command or can be sent separately.

Reset Command 0x7A

In transmission mode, encountering a 0x7A is unusual due to byte stuffing. If a 0x7A is received, the BC7215 automatically halts its operation, clears its UART buffer, and resets. During infrared signal transmission, it disregards unsent buffer data, finishing only the current bit before stopping. Given the BC7215's maximum 18ms pulse duration, allow at least 36ms after sending a reset command before transmitting new data to ensure buffer integrity.

Shutdown Command F7 00

This command, without associated data, sends the BC7215 into shutdown mode upon receipt of 0xF7 0x00 during a command wait state. Shutdown mode ceases all chip activities and UART command reception. Reactivation requires either a power reset or toggling the MOD pin to re-enable receive mode, effectively restarting chip operations.

Upon receiving a shutdown signal, the BC7215 signals entry into shutdown mode by sending a 0x7A terminator back to the host.

Handling Communication and Data Integrity with BC7215

For effective infrared transmission with the BC7215, accurate data input and avoiding communication mishaps are key. Watch out for these pitfalls:

1. **Avoid Data Transfer when BUSY is Active**: When the BC7215's buffer is maxed out, it signals BUSY high and can't process additional data. Always verify the BUSY status isn't high before sending more bytes.
2. **Mind the Data Byte Interval**: With BUSY low, spacing data bytes more than 15ms apart might reset the BC7215, thinking the communication was cut off. Be cautious if system interrupts could extend beyond 15ms during command transmission.

Data issues also pose a threat to signal accuracy, affecting both format information and specific IR command data. Ensure format details is exactly the same as the BC7215's outputs, and when issuing IR commands, align the declared bit length with the data being transmitted to avoid misinterpretation or loss of excess data.

Software Workflow

UART Transmission

For hosts like computers that typically have UART with hardware flow control capabilities, setting up the UART transmission doesn't require special handling. Just set it to use CTS/DSR hardware flow control, and link the BC7215's BUSY output as the CTS signal. Users simply need to write the data they want to send into the transmission buffer, and the hardware takes care of the rest.

For microcontroller-based setups lacking hardware flow control, users must manually check the state of the BUSY pin before sending each byte. If BUSY is high, you need to wait until it goes low before proceeding with transmission.

Data being sent must undergo byte stuffing encoding, which involves replacing any occurrences of 0x7A or 0x7B in the original data with substitute byte sequences, as follows:

- 0x7A becomes two bytes: 0x7B 0xFA
- 0x7B becomes two bytes: 0x7B 0xFB

It's important to note that if you plan to perform multiple infrared transmissions back-to-back, you should manually insert at least a 40ms delay between them. Without this pause, the BC7215 or any receiving hardware like TVs may not correctly process the incoming signals.

UART Reception

The data output from the BC7215 via UART can be categorized into two modes: transmission mode and reception mode.

Transmission Mode

The UART output in transmission mode is straightforward, outputting a single byte 0x7A upon completion of either of these two commands to indicate the end of command execution:

- Infrared send command F5 02

- Shutdown command F7 00

Users can monitor for the 0x7A to determine if a command has finished executing or simply ignore this signal. When executing consecutive infrared send commands, this byte can help users identify when the previous command has ended, or alternatively, waiting for over 100ms can ensure the prior command has completed along with a 40ms infrared sending interval.

Receiving Mode

In receiving mode, the BC7215 behaves differently based on the output setting: in simple mode, it outputs just the raw data packet; in composite mode, it sequentially outputs both the raw data packet and the format information packet.

The length of the raw data packet is variable, dictated by the received signal. Typical audio-visual device remotes send data packets of up to 4 bytes, whereas air conditioner remotes may send packets up to 32 bytes. For data transmission purposes, the theoretical maximum data size is 512 bytes, but given the slow transmission rate of infrared signals, larger packets can significantly increase transmission time and error probability, making packets no larger than 16 bytes recommended for data communication.

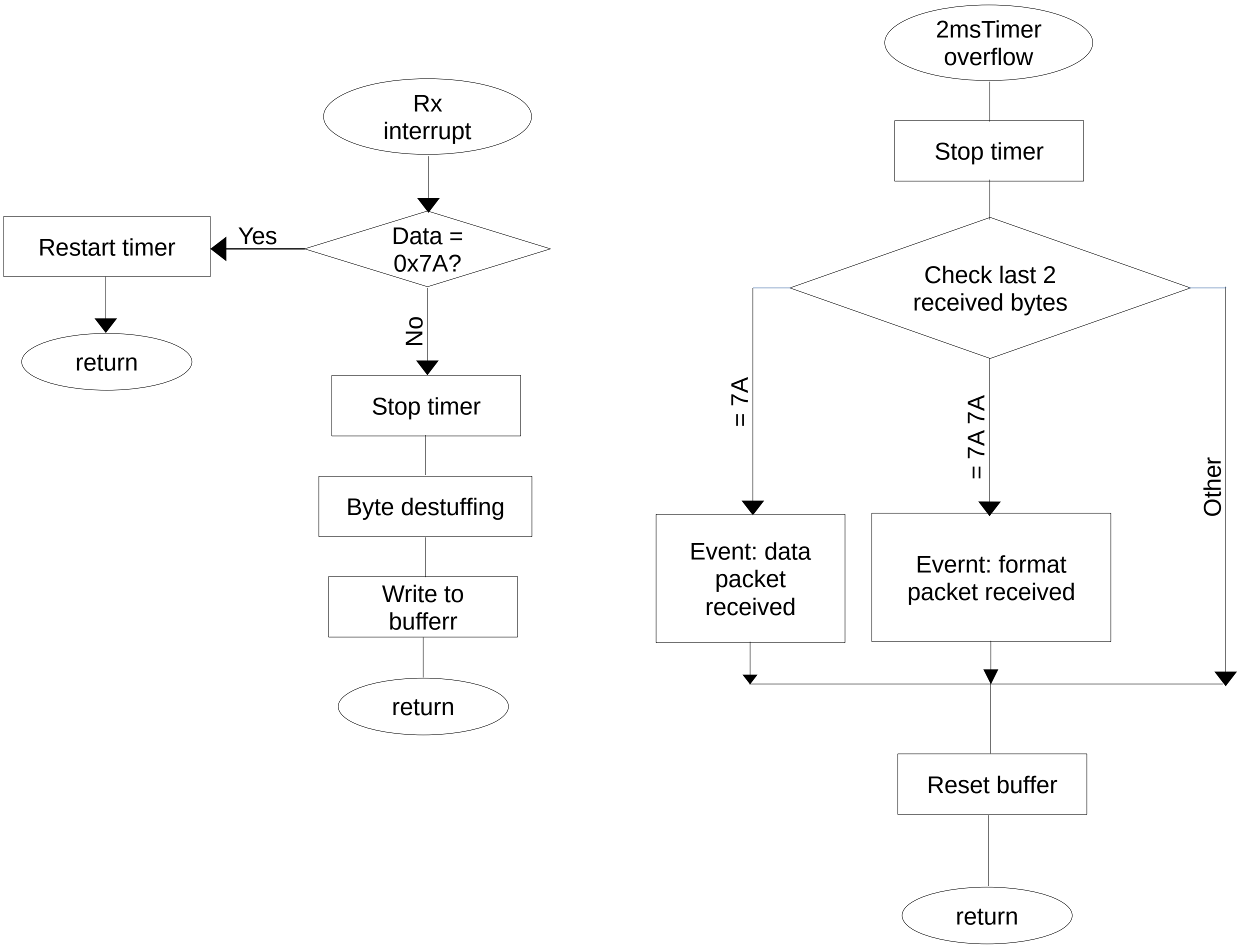
The host side should prepare a UART buffer to store data output from the BC7215, as it starts outputting UART data immediately upon infrared signal reception without internal buffering. The buffer size should at least accommodate the sum of the raw and format data packet lengths. Given the possibility of receiving packets exceeding the expected size, overflow management for the host buffer is crucial.

There are two approaches to processing received UART data:

- If a timer is available, leveraging the uninterrupted sequence of the format and raw data packets allows for relatively straightforward packet completion detection by monitoring the gap following a 0x7A character.
- Without a timer, the process becomes more complex, though both methods utilize the terminating 0x7A byte as a cue for data packet completion.

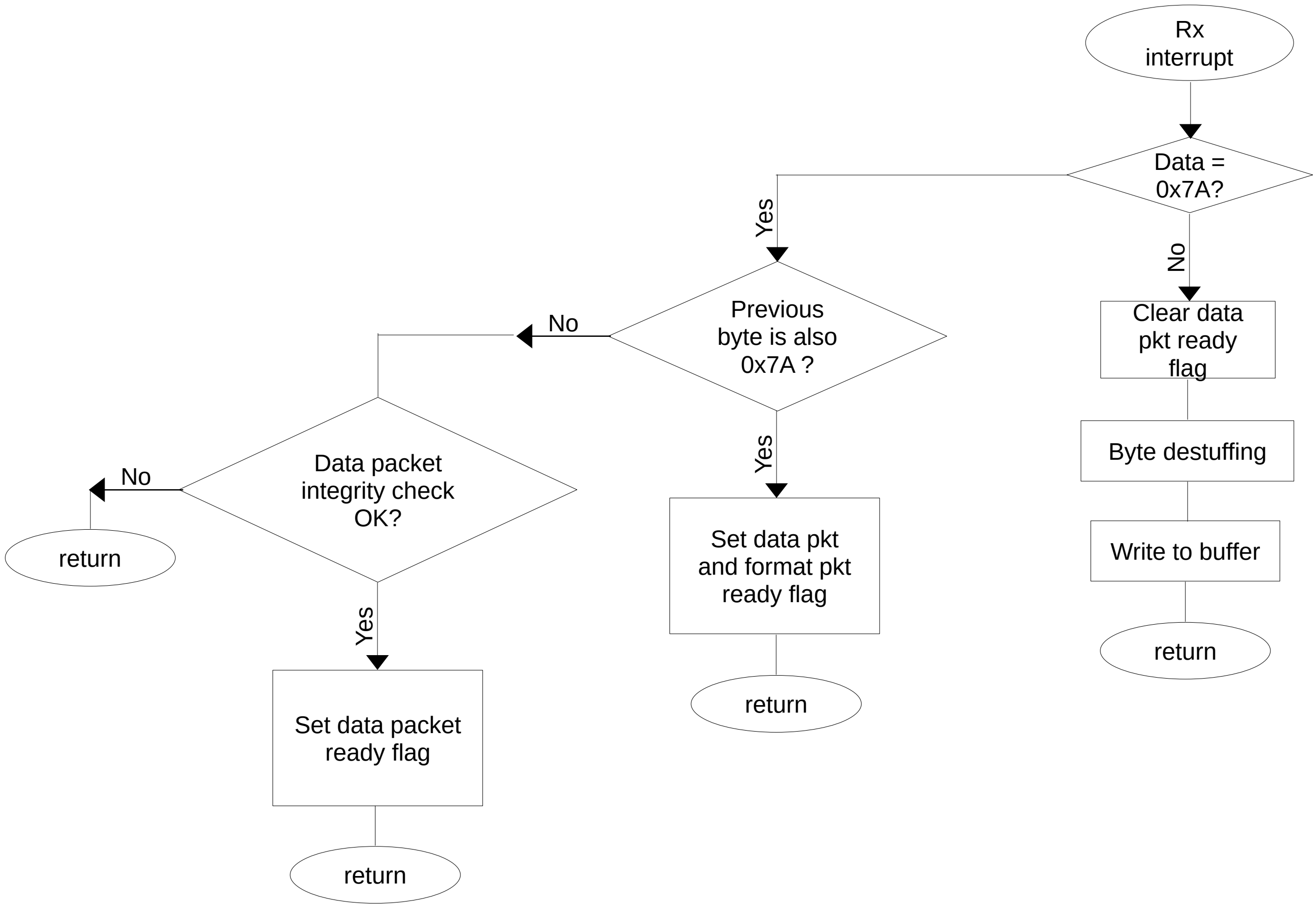
UART Data Processing with Timer Interrupt

Processing Logic: In composite mode, because the raw data packet and format data packet are output sequentially without any gaps, and there's always at least about a 20ms interval (the gap between infrared signals plus the time for 8 bits) between each decoding output by the BC7215, the duration of silence following a 0x7A character can be used to determine whether the data packet output has concluded.



UART Data Processing Without a Timer Interrupt

Processing Logic: The character 0x7A is used as an indicator to signal the end of a data packet. However, upon receiving the first 0x7A at the end of a format information packet, it's essential to perform some data checks to determine whether a new raw data packet has been received.



Typical Application Scenarios

Using for Infrared Data Transmission

Infrared remote control encoding is a time-tested, reliable method for infrared data transmission. Although its rate is relatively low, it's suitable for scenarios without real-time requirements or the need to transfer large amounts of data, such as sensor data gathering, remote data collection, and remote control. The BC7215 offers a simple, reliable, and cost-effective solution for these needs. With both receive and transmit capabilities in a single chip, only one BC7215, an infrared receiver, and an infrared LED are needed to create a complete infrared data communication circuit.

The BC7215's format information packet carries complete details about the modulation and encoding format but does not include data length, which is determined entirely by the length data within the raw data packet. Thus, the BC7215 can send and receive data of any length in any format, making it particularly useful for bidirectional communication of small data quantities. However, when choosing an infrared encoding format, be aware that some formats have specific data content requirements, such as the RC5 format, which requires the first two bits of the data to be "1", making it less suitable for data transmission. The infrared format with a signature byte of 0x34 is generally recommended for data transmission.

Users can collect format information using the BC7215's receiving mode. The following format information packet data will modulate the infrared signal in NEC code format with a carrier frequency of 37.5kHz. A standard NEC code consists of 4 bytes: the first two are address codes, and the next two are key codes and their complements. However, for data transmission purposes, both the data length and content are entirely up to the user. Users simply need to input the required data length and original data into the infrared transmission command. Here's an example of a format information packet following the NEC code encoding format:

```
34 14 5D 0D 5D 14 3D 3D 1D 1C 9C 62 A0 29 B2 99 44 00 00 C2 36 9F
F7 FA B8 E2 9A A3 26 EA 90 87 30
```

Please note, as previously mentioned, the format information packet is not unique; the NEC format information packet you collect yourself may differ from the example given above. Since the BC7215 cannot selectively receive signals of a specific format, there's a possibility that the receiver might pick up data not intended for it when used for infrared communication. Therefore, in addition to using the signature byte and data length for identification, it's advisable to include unique identifiers within the data packet, such as address codes, and to verify the integrity of the data with checksums or CRC checks to mitigate interference from other infrared remote signals.

Moreover, longer data packets are more susceptible to external interference. For data transmission purposes, it's recommended to keep the packet size to no more than 16 bytes. If there's a need to transmit a larger volume of data, consider splitting it into multiple packets, each no larger than 16 bytes.

Be aware that in receiving mode, the BC7215 updates its internal infrared format information with every received signal. If switching from receiving to transmitting during communication, it's best to re-download the format information to ensure data is consistently sent in the same format.

For data communication, remember that all data output by the BC7215 is byte-stuffed and needs to be restored by the user. Similarly, data input into the BC7215 must also undergo byte stuffing encoding.

Using for Device Remote Control

Since the BC7215 can automatically recognize and decode infrared remote signals, outputting the raw data, it can be used as a remote control receiver for user devices. However, because the BC7215 can receive signals from any remote control, when used as a device's remote receiver, users must determine whether the received data comes from the intended remote control. To do this, users should first compare the bit count and the signature byte of the output data. If these two values match, it's likely that the remote control data is from the same model of remote control (using the same control chip). Moreover, many remote controls have customizable user codes (address codes) for further identification. If the user code also matches, the signal can be identified as coming from the specified remote control. To execute specific actions based on specific remote control buttons, the entire raw data must be compared; only if the raw data is identical can a specific remote control button be confirmed.

This application method requires users to have an understanding of the data format used by their remote control.

Accepting Infrared Control from Any Remote

Because the BC7215 can receive signals from any remote control, you can use this feature to create devices that can be controlled by any remote control. For instance, you could make a remote-controlled power switch that can be operated using spare buttons on other device remotes around your home. Your user program only needs to record the raw data packet (the raw data, signature byte, and data length) of a specific button on a remote beforehand. Then, upon receiving the same data, it can determine that it has received a signal from that specific button on the remote.

Copying Infrared Remote Signals

With the BC7215's infrared transmission capability, copying any received infrared remote signal is straightforward. To replicate a just-received infrared remote signal, simply switch the BC7215 to transmission mode in time (before receiving 4 new infrared pulses) since the format information is already stored in the chip. At this point, you don't need to download the format information packet again; just use the send command with the decoded raw data to replicate the received remote control signal.

Universal Learning Remote Control

By recording the format information from various remote controls, capturing the raw data for each button, or generating the data to be sent based on analyzed patterns, you can create a universal learning remote control. This remote can control a wide range of devices in your home using a single BC7215.

Infrared Control Hub

In Internet of Things (IoT) applications, the BC7215 can serve as a key component of an infrared control hub. It can be configured remotely by a server, pre-loaded with data at the factory, or locally learn infrared signal formats to control all infrared remote devices within a specific area. Alternatively, it can be part of an extensive infrared control network comprising several emission nodes distributed across different locations. Each node receives commands from the control center to centrally manage infrared devices scattered across various sites.

Absolute Maximum Ratings

Specifications	Symbol	Range
Storage Temperature	Ts	-55˚C ~ +125˚C
Operating Temperature	Ta	-20˚C ~ +85˚C
Supply Voltage	Vcc	-0.3V ~ 5.5V
Voltage on Any Pin Relative to Ground	Vpg	-0.3V ~ Vcc+0.3

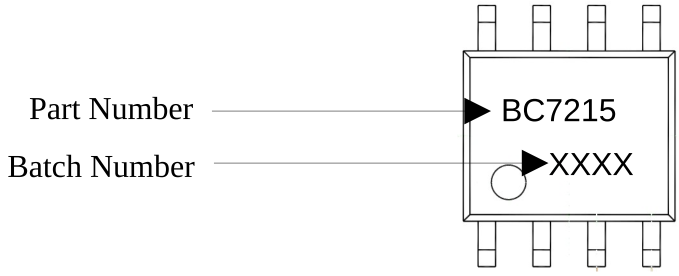
Electrical Characteristics

Note: Ta = 25˚C

Specifications	Symbol	Min	Typical	Max	Unit	Comment
Supply Voltage	Vcc	2.7		5.5	V	
Operating Current	Icc		1.7		mA	Vcc=3.3V
			2.5		mA	Vcc=5V
			7		uA	Shutdown mode, Vcc=3.3V
Input Low Voltage	V_{IL}			0.3Vcc	V	
Input High Voltage	V_{IH}	0.7Vcc			V	
Sink Current for Output Pins	I_{OL}			-20	mA	TX, BUSY
				-60	mA	OUT, Vcc=3.3V
				-90	mA	OUT, Vcc=5V
Source Current for Output Pins	I_{OH}			20	mA	TX, BUSY

Packaging Information

Identification Explanation

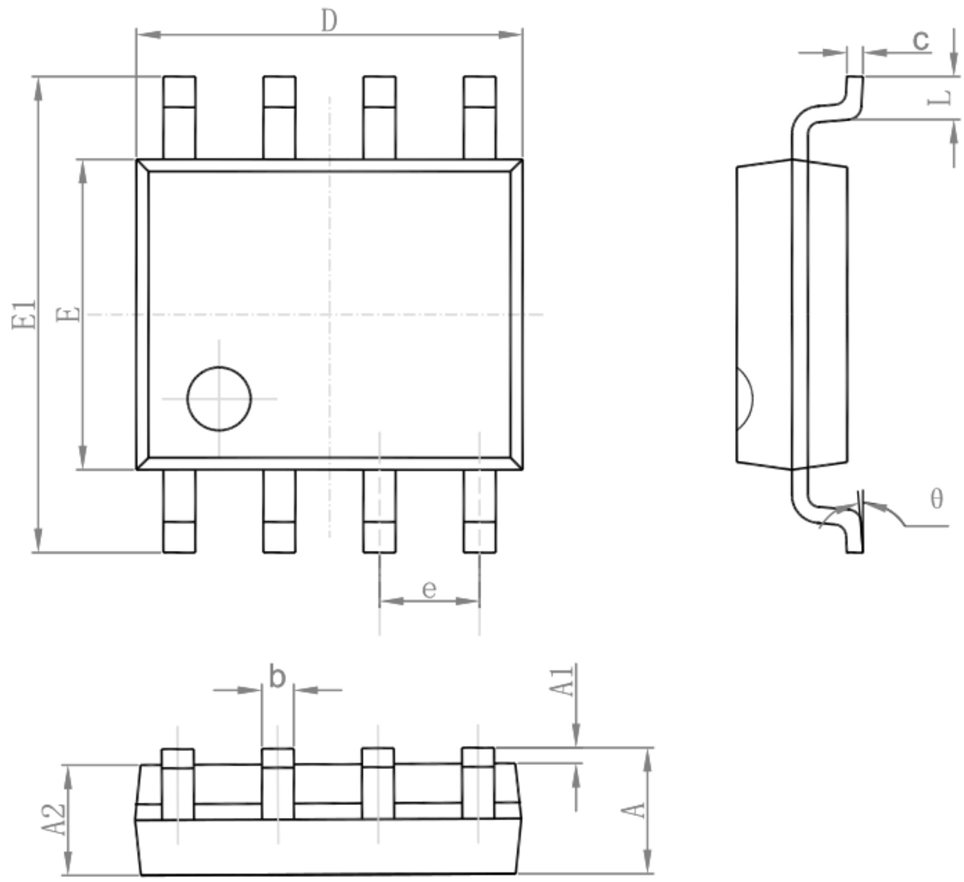


Packing Information

Ordering Model	Package Type	Qty per package

BC7215-T	Tube	10000
BC7215-RS	Tape and Reel	1800

Dimensions



Symbol	Dimensions In Millimeters		Dimensions In Inches	
	Min	Max	Min	Max
A	1.350	1.750	0.053	0.069
A1	0.100	0.250	0.004	0.010
A2	1.350	1.550	0.053	0.061
b	0.330	0.510	0.013	0.020
c	0.170	0.250	0.006	0.010
D	4.700	5.100	0.185	0.200
E	3.800	4.000	0.150	0.157
E1	5.800	6.200	0.228	0.244
e	1.270 (BSC)		0.050 (BSC)	
L	0.400	1.270	0.016	0.050
θ	0°	8°	0°	8°