m10

Subscriber Communicator

User Manual

MobiApps, Inc.
610 Herndon Parkway
Suite 500
Herndon, VA 20170

Date: 21 October 2008

Version: 1.20

Copyright Notice

Trademarks

Document Revision History

Version	Description	Date
1.00	Initial release of the user manual	23 Sep 2008
1.10	FAQ section added and minor changes made	30 Sep 2008
1.20	Technical review changes and minor additions	21 Oct 2008

Table of Contents

List of Tables

List of Figures

About This Manual

This manual defines and specifies the functions of the m10 and its usage. It also contains guidelines for using the DK10, the development kit for the m10.

Target Audience

The target audience for this user manual comprises engineers and business managers who want to augment their company's existing asset-tracking systems with satellite networks. A typical user would be an engineer who is well versed with wireless data communications.

Organization of this Manual

This content in this manual is divided into seven chapters (excluding this one) and four appendices.

Chapter 1	Introduction and overview of the m10
Chapter 2	The different interfaces available on the m10
Chapter 3	The m10's environmental, electrical, and mechanical specifications
Chapter 4	Approvals and certifications obtained by the m10
Chapter 5	How to use the m10
Chapter 6	Troubleshooting the m10
Chapter 7	Frequently asked questions regarding the m10
Appendix A	m10 mechanical drawings
Appendix B	DK10 schematic
Appendix C	Fletcher checksum
Appendix D	Glossary

Symbols Used in This Manual

Symbol	Description
	Provides additional information corresponding to the topic that is being discussed.
	Warning! If ignored, may cause permanent damage to the development kit or injury to a person. Caution! Failure to comply with a caution may result in failure or damage to the device. Danger! Failure to comply with a danger symbol may result in serious injury.

Reference Documents

The following documents have been used in the preparation of this user manual.

- Department of Defense Test Method Standard for Environmental Engineering Considerations and Laboratory Tests, MIL STD 810E
- European Telecommunications Standards Institute, EN 300 832 (A1 version 1.1.1), Electromagnetic Compatibility for Mobile Earth Stations.
- International Electrotechnical Commission, IEC 1000-4-4, Testing and Measurement Techniques -- Immunity to Fast Transient / Burst Immunity Test.
- International Electrotechnical Commission, IEC 61000-4-6, Testing and Measurement Techniques -- Immunity to Conducted Disturbances, Induced by Radio-Frequency Fields.
- ORBCOMM Serial Interface Specification, E80050015.
- ORBCOMM Subscriber Communicator (SC) Standards and Specifications, E25050102.
- ORBCOMM Subscriber Communicator Type Approval Plan, A25TP0017.
- ORBCOMM System Overview, A80TD0008.
- RoHS FAQ -- http://www.rohs.gov.uk/FAQs.aspx#10
- Society of Automotive Engineers, SAE J1455, Joint SAE / TMC Recommended Environmental Practices for Electronic Equipment Design (Heavy-Duty Trucks).
- Technical Requirements for Satellite Earth Stations and Systems (SES); Mobile Earth Stations (MES) Providing Low Bit Rate Data Communications (LBRDC) using Low Earth Orbiting (LEO) satellites operating below 1 GHz, EN 3200 721.

Where to Get Help

For additional information or clarifications regarding this user manual, please get in touch with MobiApps using the contact information provided below.

Address	610 Herndon Parkway, Suite 500, Herndon, VA 20170
Phone	(+1) 240 395 1900
Email	support.m10@mobiapps.com

1. General Description

This chapter provides an overview of the m10 ORBCOMM Subscriber Communicator (SC) and its features.

1.1 Background

The ORBCOMM system is a two-way system that supports communication to and from mobile or fixed Subscriber Communicators (SCs). In most applications, a message or other data is first generated by an SC. From that source, the message is transmitted to the nearest ORBCOMM satellite. The satellite downlinks the message to the Gateway Earth Station (GES), which then transmits the message to the desired Gateway Control Center (GCC). Within the GCC, the message is processed and forwarded to its ultimate destination by the Gateway Message Switching System (GMSS). The destination may be another SC, a pager, a corporate resource management system or any personal or business e-mail or Internet address.

The ORBCOMM system is available to transfer information for an end user whenever a satellite is in view of the SC. If the satellite is connected to a GES, the satellite is considered in real-time messaging mode and the message is passed immediately through to the GCC for routing to its final destination. If the satellite is not connected to a GES, the satellite switches to a store-and-forward mode to accept GlobalGrams. GlobalGrams are short messages (up to approximately 200 bytes) stored on-board the satellite until it connects to the SC's host GCC through an affiliated GES.

For detailed information on the ORBCOMM system, refer to the *ORBCOMM System Overview*.

1.2 Overview of the m10

The m10 (see Figure 1) is an ORBCOMM Subscriber Communicator, which enables modem-like access to the ORBCOMM satellite network. It is ideal for machine-to-machine (M2M) applications (e.g. asset tracking, utility meters, pipeline monitoring, etc.) that monitor or control remote assets. The m10 can be added to an existing tracking system or run as a stand-alone modem.

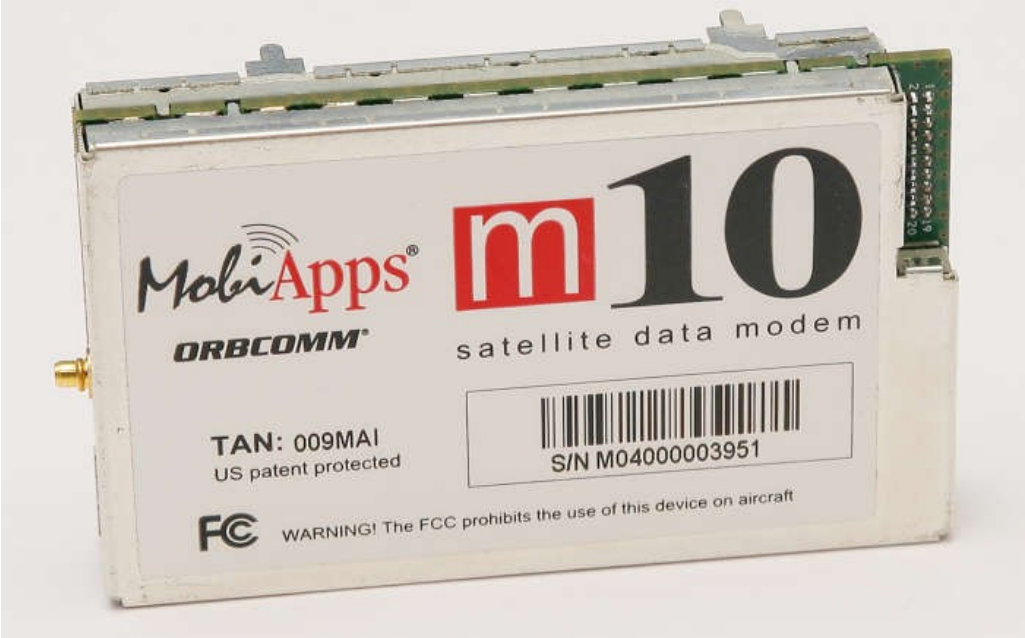

Figure 1: The m10 Subscriber Communicator

1.3 Inside the m10

The main system components used in the m10 are shown in Table 1.

Table 1: Main System Components in the m10

Analog Devices Blackfin DSP	ADSP-21531
SDRAM	32 MB
EEPROM	2 MB (Used by the m10 boot loader and firmware. Not available to the host controller.)
Watchdog	Software and hardware
Operating System (OS)	Fusion RTOS

1.3.1 DSP

The m10 uses the Analog Devices Blackfin processor. The Blackfin processor provides both a microcontroller (MCU) and digital signal processor (DSP) functionality in a unified architecture. Some highlights of this processor include:

- Dual 16-bit multiply accumulate (MAC) digital signal processing
- Full 16 / 32-bit RISC MCU programming model, optimized for C / C++
- Power efficient, designed for use in portable appliances

1.3.2 ASIC

The programmable Application-Specific Integrated Circuit (ASIC) from MobiApps performs the Radio Frequency (RF) signal mixing, down-conversion, and up-conversion. This ASIC enables a significant reduction in size and power consumption when compared to the previous generations of satellite modems. It achieves this reduction by integrating the transmit and receive chain mixers, and the filter and control functions, which were previously discrete.

The ASIC converts analog samples to baseband digital samples and feeds them to the DSP. Similarly, the ASIC receives baseband digital samples from the DSP and converts them to analog samples.

1.3.3 Operating System

The m10 software is based on the Fusion RTOS (Real-Time Operating System) from Unicoi Systems. Fusion is an embedded RTOS designed and optimized for 16-bit and 32-bit DSPs, and for next-generation media processors. It is priority-based, preemptible, deterministic, and protects against priority inversion. Fusion provides an extensive set of real-time services including task control, task communications, task synchronization, packet management, and memory and time management.

1.4 Features

The m10 is designed to tightly couple with a host controller, in a manner similar to the cellular transceivers available today. The following are the main features of the m10.

Direct board mounting	Direct mechanical and electrical mounting to the host controller board using a dual-row header
TTL-level serial	The m10 and host controller board communicate at 3.3 V levels
Low power consumption	The m10's power consumption is among the lowest in the industry. (Searching and receiving: 60 mA, transmitting: 2 A)
Power-down mode	The host controller can shut off power when the m10 is not being used. The m10 accepts serial commands after two seconds following power-up.
Data available	A 3.3 V output that indicates messages from satellite
Industrial grade	High temperature rating of -40 °F to 185 °F (-40 °C to 85 °C) SAE J1455 shock and vibration rating.
RoHS	The m10 is compliant with the EU's RoHS directive.
Satellite available	A 3.3 V output that indicates satellite availability.
RF connector	Industry-standard micro-coax RF jack-type MMCX.
Pick and place	The m10's design is compatible with SMT processes. This reduces time taken and complexity required for installation on a host sub-system.

The term 'host controller' refers to the embedded processor on the host and its PCB circuitry, which interface with and control the m10.

1.5 Block Diagram

Figure 2 shows the connections between the different components in the m10.

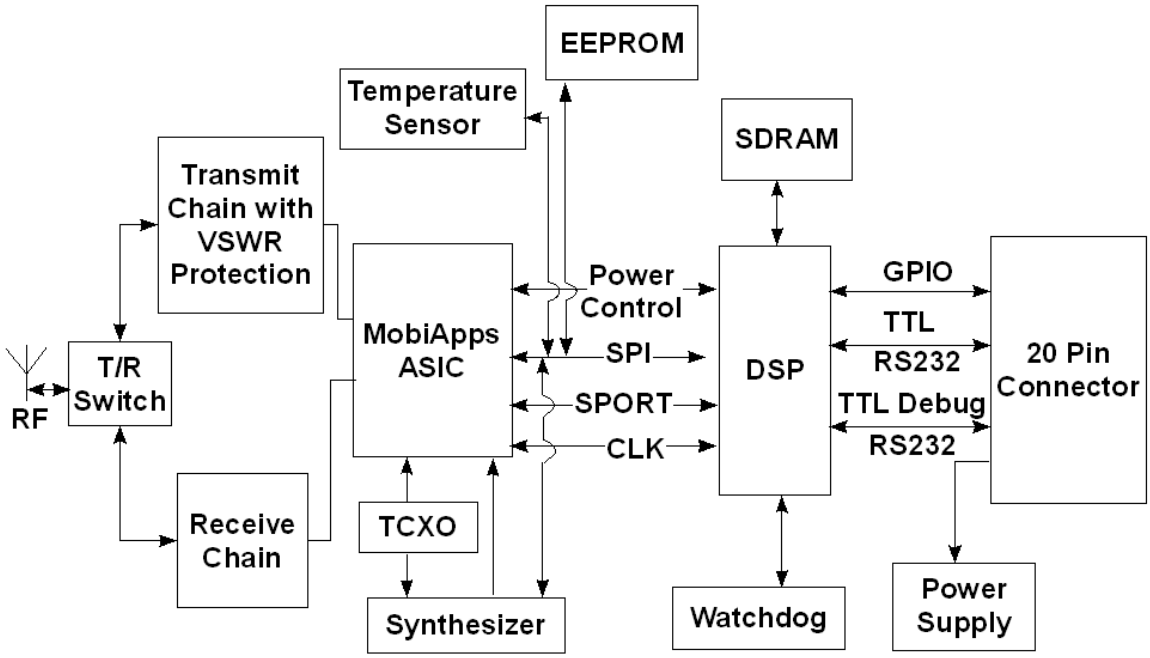

Figure 2: Block Diagram of the m10

2. Interfaces

This chapter explains the different interfaces available on the m10.

2.1 Connector Pins

The host controller provides power and ground, and communicates serially with the m10. The connector that goes on the host controller board is an easily available, reliable, dual-row SAMTEC connector, with 10 pins, enabling direct board mounting.

The m10 is outfitted with a factory-mounted, 20-pin (10 pins across 2 rows) SAMTEC FTS-110-02-F-D male connector (see Figure 3 and Figure 4). The details of the pin designations are provided below.

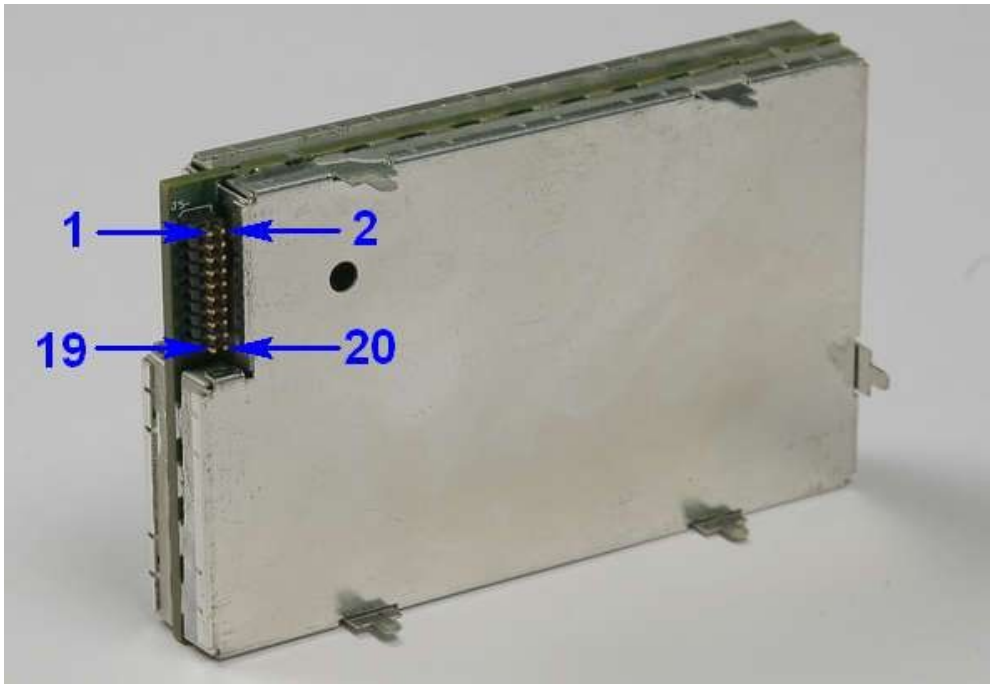

Figure 3: m10 Pin Numbering

The first half of the connector (pins 1 through 10) provides the interface necessary to communicate over a 3.3 V TTL level serial link and to power the m10. The second half of the connector (pins 11 through 20) has been reserved for MobiApps. Refer to Table 2 for details. The m10 can be connected to the host PCB using a 10-pin SAMTEC female FLE-105-01-G-DV-K-TR.

Pins 11 through 20 must be left unconnected.

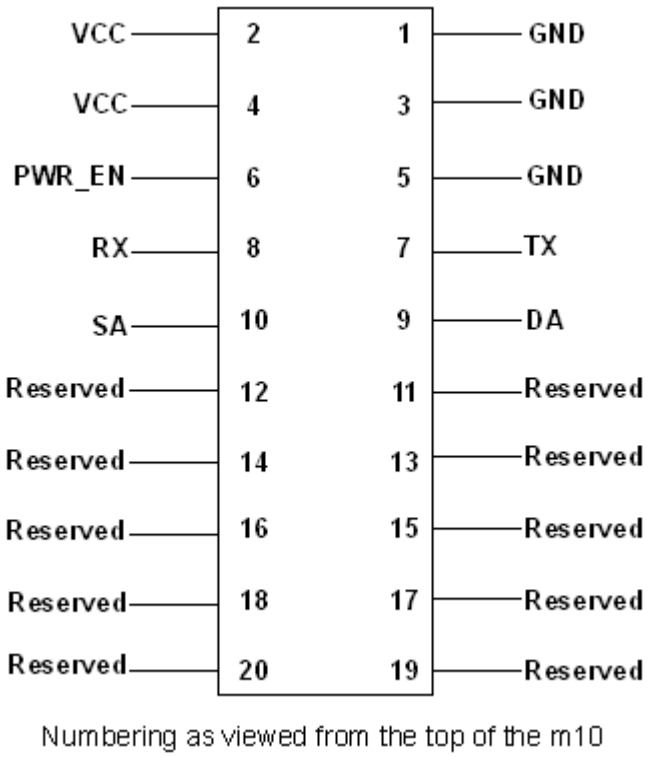

Figure 4: Pin Assignment

Table 2: Connector Pins on the m10

Pin	Name	Description	Digital I/O Direction	Comments
1	GND	Ground		
2	VCC	Supply		
3	GND	Ground		
4	VCC	Supply		
5	GND	Ground		
6	PWR_EN	m10 Power Enable	Input	Pulled low using 47KΩ
7	TX	3.3 V RS232 Interface. m10 to the host controller	Output	
8	RX	3.3 V RS232 Interface. Host controller to the m10	Input	
9	DA	Data Available	Output	
10	SA	Satellite Available	Output	
11-20	Reserved for MobiApps			

2.2 Serial Interface

The main serial interface provides device management and monitoring using 3.3 V levels. The serial data is carried on the Transmit (TX) and Receive (RX) lines. The details of the Transmit and Receive pins are shown in Table 3. An RS232 level shifter is required to connect the m10 to a PC. This is already available on the DK10, the m10 development kit.

The serial specifications, which are software-configurable, are shown in Table 4. The specifications can be configured using the ORBCOMM *Set Parameter* packet (see Table 13). (Refer to *Table 3.19* in the *ORBCOMM Serial Interface Specifications* document for details.)

Using this packet, the *parameter_num* (byte 5 in Table 13) and *parameter value byte* (byte 7 to byte 7+n in Table 13) should be set according to Table 5. (Refer to *Appendix A SC Parameters* in the *ORBCOMM Serial Interface Specifications* document for details.)

Table 3: Transmit and Receive Pins

Pin	Name	Description
7	TX	3.3 V RS232 interface; m10 to the host controller
8	RX	3.3 V RS232 interface; host controller to the m10

Table 4: Serial Interface Specifications

Baud rate	300 bps to 19200 bps
Parity Bits	0, 1, 2 (none, odd, even)
Stop Bits	0, 1, 2

Table 5: Parameters Numbers and Associated Values

Number (*parameter_num*)	Name	Value (*parameter value byte*)	Comments
0x29	baud_rate	300 bps to 19200 bps	DTE baud rate
0x2A	parity_bits	0, 1, 2	DTE parity (0: none, 1: odd, 2: even)
0x2B	stop_bits	0, 1, 2	DTE stop bits

2.3 Power and Ground

The m10 is powered using five pins of its 20-pin connector (see Table 6). The five mounting pins on the shield of the m10 are also connected to the ground plane of the m10.

The input supply for the m10 should be in the range of 9 V to 18 V and the input supply should be able to source up to 2 A.

The m10 does not need a constant 2 A current. Instead, 2 A will be consumed as pulsed current when the RF transmit is in progress. The power supply must be able to source 2 A instantaneously (<100 µs), otherwise communications will fail.

The transmitter ON time can vary from 3.3 milliseconds to 4.8 seconds respectively.

Table 6: Power Pins

Pin	Name	Description
1	GND	Ground
2	VCC	Supply
3	GND	Ground
4	VCC	Supply
5	GND	Ground

2.4 Power Enable

The Power Enable signal is used to turn the m10 ON or OFF. It is a 3.3 V active high logic-level signal. This signal has to be maintained at an active high level in order to keep the m10 ON.

Table 7: Power Enable Pin

Pin	Name	Description
6	PWR_EN	m10 Power ON / OFF; 3.3 V Logic Level, High=ON

2.4.1 Powering Up

To power on the m10, the host controller must provide an input voltage between 9 V and 18 V DC and then set the PWR_EN pin to active high. Typically, the m10 will take two seconds to be in a ready state. After powering on, the m10 will toggle the DA pin once to let the host controller know that it is ready to receive serial commands.

After the DA pin's initial toggle, its subsequent active high state indicates messages from the satellite that are available in the m10's inbound (IB) queue. The m10's timing diagram for powering up is shown in Figure 5.

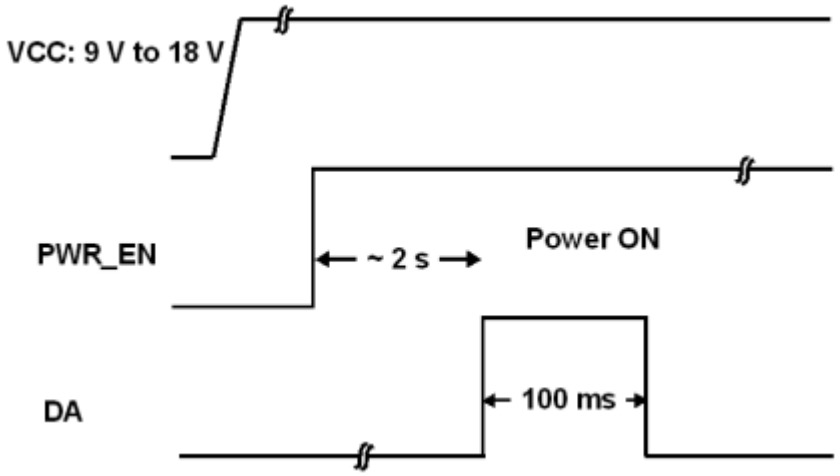

Figure 5: Timing Diagram for Powering Up

The delay between PWR_EN going high with respect to VCC going high is not critical. MobiApps recommends a 10 milliseconds delay.

2.4.2 Powering Down

The host controller initiates the m10's power down sequence by sending a Power Down packet (see Section 5.5.1). When the m10 receives the command, it closes all open sessions, saves critical parameters to its non-volatile memory, and then sends an Acknowledgment packet (see Section 5.5.2) to the host controller indicating that it is ready to be powered down.

The host controller then sets the PWR_EN pin LOW. The host controller can leave the supply voltage for m10 ON, or remove it at its discretion. If the supply is left ON with PWR_EN set to low, the m10 will be powered down. However, it will consume a leakage current of less than 10 µA. The timing diagram of the m10's power down sequence is shown in Figure 6.

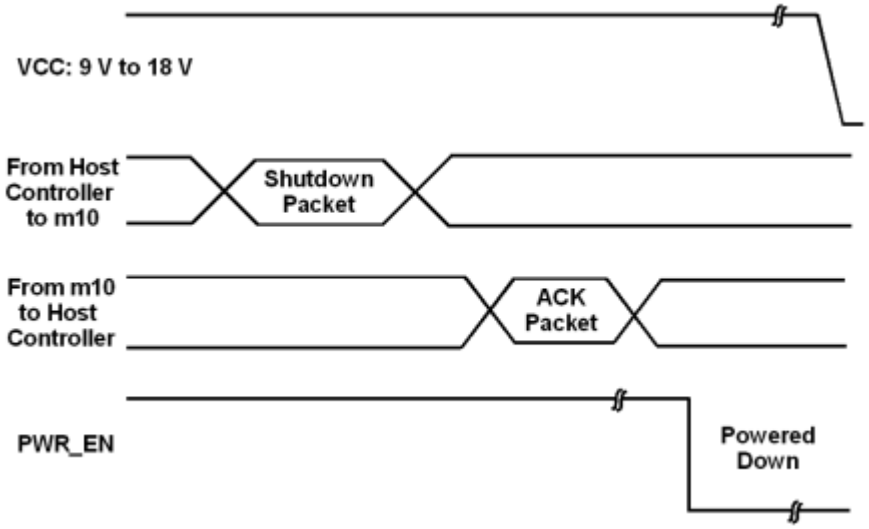

Figure 6: Timing Diagram for Powering Down

It is the responsibility of the host controller to back up all messages, reports, GlobalGrams, etc. before powering down the m10.

2.5 DA (Data Available)

The Data Available signal is driven high when the m10 receives messages, user commands, or GlobalGrams from the satellite. The signal goes low only after the host controller has cleared the outbound (OB) queue. The m10's timing diagram for DA is shown in Figure 7.

Table 8: DA Pin

Pin	Name	Description
9	DA	Data Available. Normally low, driven high when the m10 receives messages, user commands, or GlobalGrams from the satellite.

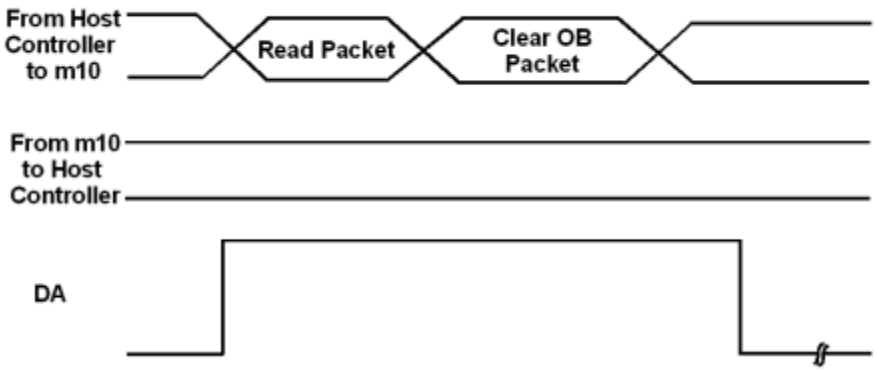

Figure 7: Timing Diagram for Data Available

2.6 SA (Satellite Available)

The Satellite Available signal is set to high when the m10 is receiving valid downlink signals. The m10's timing diagram for SA is shown in Figure 8.

Table 9: SA Pin

Pin	Name	Description
10	SA	Satellite Available. Normally low, driven high when the m10 is receiving valid downlink signals.

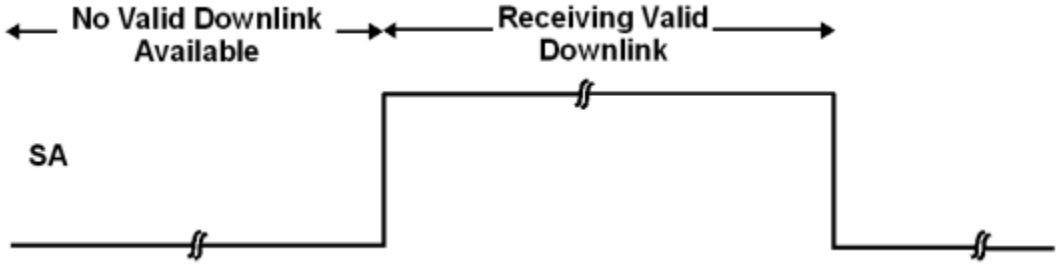

Figure 8: Timing Diagram for Satellite Available

2.7 RF Interface

The RF connector for the m10 (see Figure 9) is an MMCX plug, with an impedance of 50 Ω nominal. Refer to Table 10 for details.

Figure 9: RF Connector on the m10

Table 10: RF Interface Details

Parameter	Value
Transmit Power	5 W (37 + / -1 dBm)
TX Frequency	148 MHz to 150.05 MHz
RX Frequency	137 MHz to 138 MHz
VSWR	< 2:1 preferred
Sensitivity	≥ -119 dBm
Dynamic Range	40 dB
Antenna Gain	≥ 0 dBi, in at least one direction
Antenna Impedance	50 Ω nominal

The m10 is protected from VSWR failure. However, for best performance, MobiApps recommends using an antenna with good VSWR readings (see Table 10) at both the RX and TX frequencies.

3. Technical Specifications

This chapter describes the environmental, electrical, and mechanical specifications of the m10.

3.1 Environmental

This content is provisional and may be modified later.

The m10 conforms to the relevant sections of the following documents:

1. ORBCOMM Subscriber Communicator (SC) Standards and Specifications
2. SAE J1455 Joint SAE / TMC Recommended Environmental Practices for Electronic Equipment Design (Heavy-Duty Trucks)
3. FCC Part 2 and Part 25
4. CE mark (EC directive- 95 / 5 / EC)

3.1.1 Electromagnetic Compatibility

3.1.1.1 Radiated Emissions

The m10 conforms to *Section 4.1 Radiated Emissions* of the *ORBCOMM Subscriber Communicator (SC) Standards and Specifications* document for emission within and outside the 148.000 to 150.050 MHz band. Refer to *EN 3200 721* for details.

3.1.1.2 Conducted Emissions

The RF connector on the m10 complies with the test methodology and test levels for conducted emissions defined in *EN 300 832*. Refer to *Section 4.3 Conducted Emissions* of the *ORBCOMM Subscriber Communicator (SC) Standards and Specifications* document.

3.1.1.3 Radiated Immunity

The m10 complies with the test methodology and test levels for radiated immunity, defined in *EN 300 832*. Refer to *Section 4.2 Radiated Immunity* of the *ORBCOMM Subscriber Communicator (SC) Standards and Specifications* document.

The m10 also complies with the test methodology defined in *IEC 1000-4-6* for radiated immunity at the following test levels:

- 1 V for 10 - 150 KHz
- 10 V for 0.15 - 2.0 MHz
- 30 V for 2 - 80 MHz

3.1.1.4 Conducted Immunity

The RF connector on the m10 complies with the test methodology and test levels for conducted immunity defined in *EN 300 832*. (Refer to *Section 4.4 Conducted Immunity* of the *ORBCOMM Subscriber Communicator (SC) Standards and Specifications* document.) The RF connector also complies with the test methodology and test levels for conducted immunity defined in *IEC 1000-4-4 (Level 3)*.

3.1.1.5 Electrostatic Discharge

The RF connector on the m10 has protection against electrostatic discharge (ESD) events of 8 kV contact discharge and 15 kV air discharge with the *Human Body Model*. The RF connector also complies with the test methodology and test levels at the RF pin defined in *EN 300 832*.

3.1.2 RoHS

The m10 complies with the European Union's directive *2002 / 95 / EC, Restriction of Hazardous Substances in Electrical and Electronic Equipments (RoHS)*, and similar regulations that may be adopted by other countries. The RoHS directive restricts lead, mercury, cadmium, hexavalent chromium, polybrominated biphenyis (PBBs), and polybrominated biphenyl ethers (PBDEs) in electrical and electronics equipment, except for certain maximum concentration levels in homogeneous materials.

This specification applies to all materials, parts, components, and / or products used in the m10.

3.1.3 Operating Temperature

The m10 operates from -40 °F to +185 °F (-40 °C to +85 °C). The lower temperature limit has been verified in accordance with *MIL-STD-810E, Method 502.3*, *Procedure II*, 4-hour test duration. The upper temperature limit has been verified in accordance with *MIL-STD-810E, Method 501.3, Procedure II*, 3-diurnal-cycle test duration. Refer to *Sections 3.2 High Temperature and 3.3 Low Temperature* of the *ORBCOMM Subscriber Communicator Standards and Specifications* for the temperature limits and verification methodology.

The m10 has been designed and tested to operate under the conditions shown in Table 11.

Table 11: Operating Conditions

Operating Temperature	Humidity Cycle	Comments
-40 °F to +185 °F (-40 °C to +85 °C)	70% relative humidity (RH) – 40% moisture by volume	For 24 hours with a 12 V supply

3.1.4 Storage Temperature

The m10 can be stored for extended periods from -40 °F to +185 °F (-40 °C to +85 °C). It has been subjected to the upper and lower temperature limits for 24 hours each and then checked for stability.

3.1.5 Operational Shock

The m10 remains structurally intact and operational after the drop test (as defined in *SAEJ1455*) from 1.22 m onto each of its corners and its face.

The m10 also meets the *MIL-STD-810E, Method 516.4, Procedure I* as defined in the *ORBCOMM Subscriber Communicator (SC) Standards and Specifications*.

3.1.6 Operational Vibration

The m10 meets the *MIL-STD-810E, Method 514.4* and *ETSI 300 721* as defined in the *ORBCOMM Subscriber Communicator (SC) Standards and Specifications*.

In addition, the m10 has also been tested according to *SAEJ1455* (as per the practices for heavy duty trucks), which specifies that the vibration profile for CAB-mounted vibration PSD for Vertical Axis (*Figure 6*, 1.04 g rms) is to be tested in three mutually perpendicular planes and that a sweep rate of ½ octave per minute is to be used.

3.2 Electrical

This section describes the electrical specification of the m10. Many of the requirements listed below refer to the *ORBCOMM Subscriber Communicator (SC) Standards and Specifications* document.

3.2.1 Power Design

3.2.1.1 Power Supply

The m10 operates on a single unregulated power source of 9 V to 18 V DC (nominal 12 V) supplied through the pins of the host connector. The Power ON / OFF control circuit is outside the m10 and is controlled by host controller.

- Low: OFF state
- High: ON state

The m10 does not have reverse polarity protection. Therefore, ensure that only positive voltage is supplied to avoid damaging the m10.

3.2.1.2 Power Consumption

The maximum current drawn is as follows:

- **Transmitting**: 2 A at 12 V nominal input power, when the RF Module is transmitting at 5 W output power.
- **Receiving**: Maximum < 60 mA at 12 V nominal input power when searching for or continuously receiving the satellite signal.
- **Powering-down**: <10 µA at 12 V when the m10 is in the OFF state.

3.2.2 Digital Input / Output Specifications

The output pins can source / sink up to 10mA. For higher currents, the outputs must be buffered.

Table 12: Digital I/O Electrical Specifications

Parameter	Min (V)	Max (V)	Maximum Recommended Load
$V_{Input\ Low}$	0	0.8	Not applicable (NA)
$V_{Input\ High}$	2.0	3.3	NA
$V_{Output\ Low}$	0	0.4	2 mA
$V_{Output\ High}$	2.4	3.3	2 mA

3.2.3 Lifespan

3.2.3.1 Design

Each major component has a product roadmap of pin-compatible replacements for a minimum of 3 years.

3.2.3.2 Failure Rate

The m10 has a Mean Time between Failure (MTBF) of at least 43,800 hours with a transmitter duty cycle of 0.05% and receiver duty cycle of 99.95% over the entire period. Refer to *Section 2.2.9 Reliability* of the *ORBCOMM Subscriber Communicator (SC) Standards and Specifications*.

3.3 Mechanical

This section describes the mechanical specification of the m10.

3.3.1 Module Size

The m10 (see Figure 10 and Figure 11) has a maximum footprint of 2.95" x 1.81" x 0.52" (74.93 mm x 45.97 mm x 13.21 mm) with all the RFI covers in place.

Figure 10: m10 Dimensions: Length

Figure 11: m10 Dimensions: Width

3.3.2 Mounting

The m10 can be mounted directly to the host PCB using a board-to-board connector. The m10 has five solder legs, which must be soldered to the host controller's common ground reference plane. The host controller's PCB needs to have five mounting holes to accommodate the m10.

The ground reference plane for the m10's connector and the solder legs should be the same.

Refer to Appendix A for the m10 mechanical drawings.

4. Approvals and Certifications

This content is provisional and may be modified later.

MobiApps is in the process of obtaining the following regulatory approvals and certifications for the m10.

- FCC
- CE Mark
- Industry Canada
- ORBCOMM Type Approval
- SAE

5. Using the m10

This chapter describes the details regarding the functioning of the m10.

5.1 Interfacing with the m10

Figure 12 illustrates how the host controller can be interfaced with the m10.

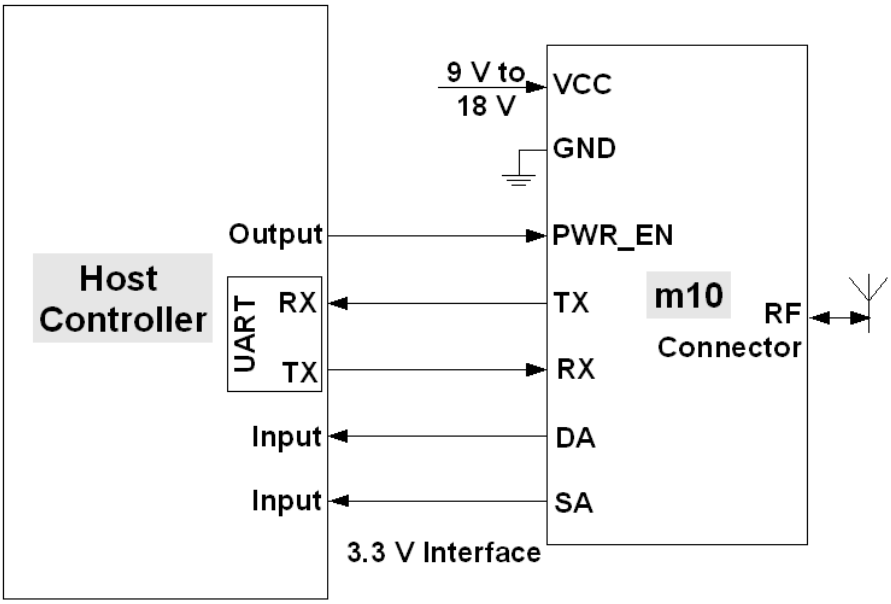

Figure 12: Interface Between the Host Controller and the m10

5.2 Powering Up

Refer to Section 2.4.1 for details of how to power up the m10.

5.3 Powering Down

Refer to Section 2.4.2 for details of how to power down the m10.

5.4 How the m10 Functions

The host controller operates the m10 by sending it serial commands that conform to the ORBCOMM Serial Interface Specifications.

For a detailed list of all the commands, refer to the *ORBCOMM Serial Interface Specification*.

As an example, the commands that a host controller needs to send to the m10 to set it up for transmission are provided here. To set up the m10 for transmission, the host controller first sends *Set Parameter* packets, provided by ORBCOMM (see Table 13). (Refer to *Table 3.19* in the *ORBCOMM Serial Interface Specifications* document for details.)

The host controller needs to send the Set Parameter packets only once every power up.

Using the *Set Parameter* packet, the *parameter_num* (byte 5 in Table 13) and *parameter value byte* (byte 7 to byte 7+n in Table 13) should be set according to Table 14. (Refer to *Appendix A SC Parameters* in the *ORBCOMM Serial Interface Specifications* document for details.)

The Little Endian byte order is used for all the MobiApps Extended Packets.

Table 13: ORBCOMM Set Parameter Packet

	7 6 5 4 3 2 1 0	
0	0x85 / 0x86	Packet header byte. (0x85 is used when byte 4 is to be interpreted as *retry_count*. 0x86 is used when byte 4 is to be interpreted as *pkt_seq_num*.)
1	0x13	Packet type
2	length0	Packet length, starting with byte 0, includes checksum
3	length1	
4	retry_count / pkt_seq_num	Number of times this packet has been re-sent / packet sequence number
5	*parameter_num*	Number of parameter whose value should be returned, see Appendix A
6	*parameter value byte count*	Number of parameter value bytes specified (n)
7	*parameter value byte 0*	One or more hexadecimal bytes indicating the desired value of the specified parameter; parameter value byte 0 is LSB
	parameter value byte 1	
7+n	*parameter value byte n-1*	
8+n	checksum byte 0	Fletcher checksum (Refer to Appendix C for details)
9+n	checksum byte 1	

Table 14: Parameter Numbers and Associated Values

Number (*parameter_num*)	Name	Value (*parameter value byte*)	Comments
0x01	desired_gwy_id	1	Gateway ID for direct mode messaging
0x03	def_ack_level	2	Default acknowledgment level for messages.
0x08	def_serv_type	2	Default service type for reports
0x09	gwy_search_mode	2	'2' indicates that the m10 should maintain a lock with the first-discovered downlink

If the desired gateway is set incorrectly and the search mode is set to its default value of 0 (search for desired gateway), then the modem will have difficulty locking onto a satellite. This could happen if the modem is located outside of North or South America, but the desired gateway is still set to its default (gateway = 1).

The m10 queues up an SC-Originated (SCO) report by sending the following ORBCOMM Serial Interface Specification packet (see Table 15). (Refer to *Table 3.8* in the *ORBCOMM Serial Interface Specifications* document for details.)

Table 15: ORBCOMM SC-Originated Report

	7 6 5 4 3 2 1 0	
0	0x85 / 0x86	Packet header byte. (0x85 is used when byte 4 is to be interpreted as *retry_count* and 0x86 is used when byte 4 is to be interpreted as *pkt_seq_num*.)
1	0x08	Packet type
2	0x12	Packet length, starting with byte 0, includes checksum
3	0x00	
4	retry_count / pkt_seq_num	Number of times this packet has been re-sent / packet sequence number
5	*gwy_id*	Destination ORBCOMM Gateway ID
6	*polled*	SC-Originated: polled by ORBCOMM Gateway or initiated by SC (see *Section 3.2, note 1*)
7	*serv_type*	Service type (see *Section 3.2, note 2*)
8	*or_ind*	Originator / Recipient indicator, only values 0 - 3 (see *Section 3.2, note 4*)
9	*mha_ref_num*	DTE assigned, used to uniquely identify messages
10	*user data byte 0*	User data
11	*user data byte 1*	
12	*user data byte 2*	
13	*user data byte 3*	
14	*user data byte 4*	
15	*user data byte 5*	
16	checksum byte 0	Fletcher checksum (Refer to Appendix C for details)
17	checksum byte 1	

Figure 13 illustrates the flow of a serial packet through the m10's host packet processing module.

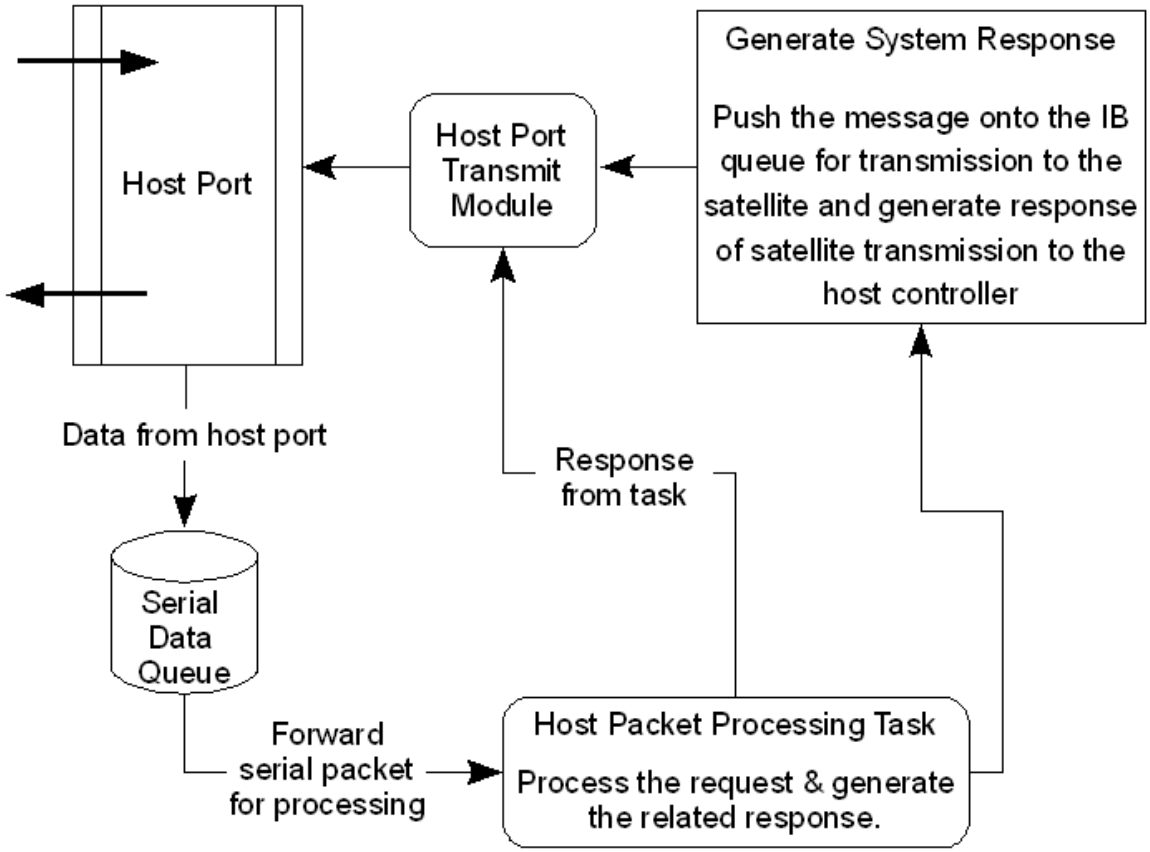

Figure 13: Serial Interface Through Host Port

Figure 14 illustrates the typical transmission cycle for an SCO report after a host controller queues up a packet, as described in Figure 13.

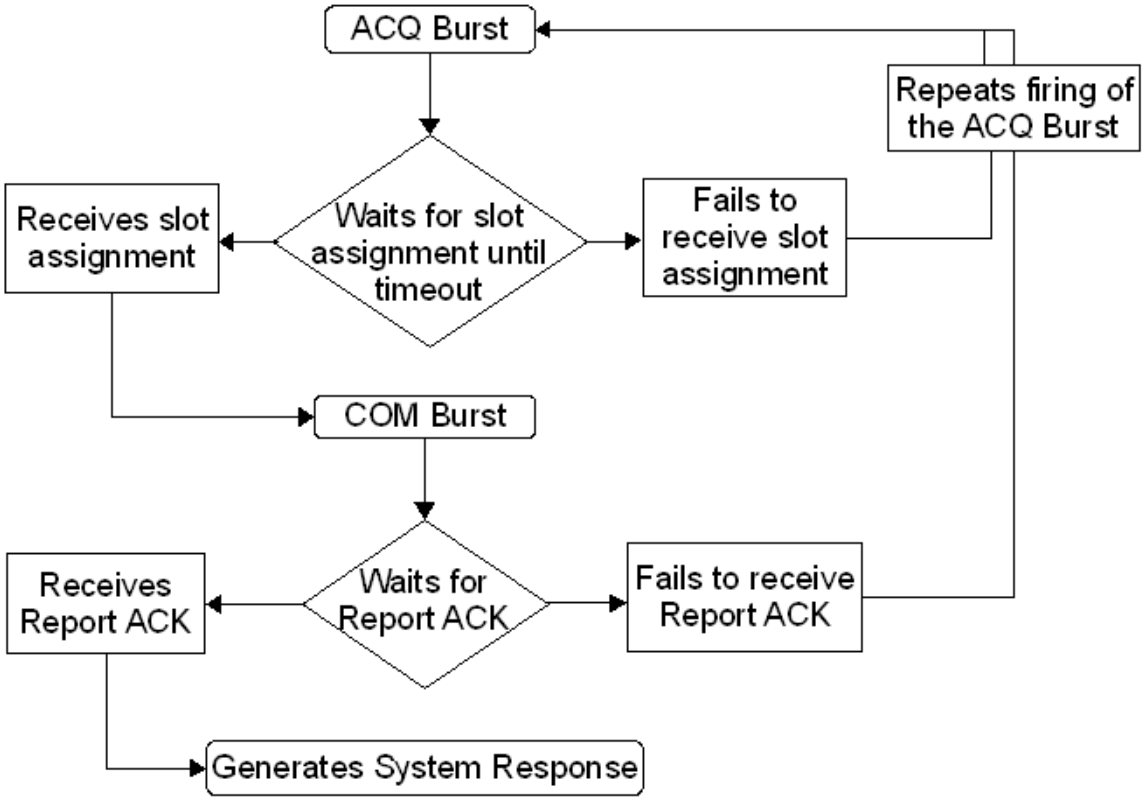

Figure 14: SCO Report Transmission Cycle

5.4.1 Inbound Message Management

The host controller sends messages via ORBCOMM by sending the appropriate serial specification commands to the m10. The m10 then inserts the messages into its inbound (IB) message queue.

Non-GlobalGram (direct mode) messages are sent when:

1. The RF module is on,
2. The RF module is locked onto a satellite, and
3. The satellite is connected to the ORBCOMM gateway specified in a message.

GlobalGram messages are sent when

1. The RF module is on,
2. The RF module is locked onto a satellite, and
3. The satellite is not connected to any ORBCOMM gateway.

Messages are sent in the descending order of priority, i.e. from the highest to the lowest priority. The message priorities are *special delivery*, *urgent*, *normal*, and *low*. Messages with the same priority are sent in the order in which they are added, unless the host controller inserts a message at the front of its priority queue. The m10's default priority for messages is set to *normal*. Each message must have a unique MHA (Message Handler) number that can take any value from 0 to 254.

It is the responsibility of the host controller to maintain the correct MHA list and to avoid having two pending messages with the same MHA.

Messages remain in the inbound message queue until ORBCOMM has successfully transferred them to a gateway, or to a satellite in the case of GlobalGrams. After a successful transfer, ORBCOMM will send an acknowledgment message to the m10. The m10 will then delete the message from its inbound message queue.

The m10 will not store any inbound messages in its non-volatile memory. It is the responsibility of host controller to back up these messages.

5.4.2 Outbound Message Management

Messages are delivered to the m10 either directly, or because of a poll from the modem to a gateway (direct mode) or to a satellite (GlobalGram mode). Outbound messages are maintained by the m10 in its outbound message queue. The host controller can check the outbound message queue either periodically or on notification of an outbound message from the m10.

The m10 will not store any outbound messages in its non-volatile memory. It is the responsibility of host controller to back up these messages.

The host controller can poll the m10 for SC-terminated messages, commands, or Global Grams by sending a Communications Command packet (see Table 16). (Refer to *Table 3.2* in the *ORBCOMM Serial Interface Specifications* document for details.)

Table 16: Communications Command Packet

	7 6 5 4 3 2 1 0	
0	0x85 / 0x86	Packet header byte. (0x85 is used when byte 4 is to be interpreted as *retry_count*. 0x86 is used when byte 4 is to be interpreted as *pkt_seq_num*.)
1	0x13	Packet type
2	length0	Packet length, starting with byte 0, includes checksum
3	length1	
4	retry_count / pkt_seq_num	Number of times this packet has been re-sent / packet sequence number
5	*type_code*	type of action requested (0 = request all SC-Terminated messages/commands queued in ORBCOMM Gateway)
6	*value byte 0*	generic value pertinent to type of action requested (used with *type_code*: 4, 5, 7, 8, 10-15, 18, & 25)
7	*value byte 1*	
8	*value byte 2*	
9	*value byte 3*	
10	*gwy_id*	destination ORBCOMM Gateway (used with *type_code*: 0, 1, 3, 5-15, 23)
11	checksum byte 0	Fletcher checksum (Refer to Appendix C for details)
12	checksum byte 1	

5.5 MobiApps Extended Packets

This content is provisional and may be modified later.

In addition to the ORBCOMM serial commands, the following MobiApps Extended Packets are available to the host controller.

1. Power Down
2. Acknowledgment
3. Self Power Down
4. Temperature Query
5. Temperature Query Response

The details of these packets are given in the sections below.

5.5.1 Power Down

This packet (see Table 17) is sent by the host controller to the m10 to indicate that it wants to power down the m10. The host controller must wait for an acknowledgment from the m10 before making PWR_EN low.

Table 17: Power Down Packet

	7 6 5 4 3 2 1 0	
0	0x85	Packet header (0x85 indicates that byte 4 is to be interpreted as *retry_count*.)
1	0x18	Packet Type
2	length0	*length0* and *length1* corresponds to entire length of the packet including the checksum, where *length0* is the Least Significant Byte (LSB) and *length1* is the Most Significant Byte (MSB)
3	length1	
4	retry_count / pkt_seq_num	Number of times this packet has been re-sent / packet sequence number
5	fletch0	*fletch0* and *fletch1* are Fletcher checksums, where *fletch0* is the LSB and *fletch1* is the MSB (Refer to Appendix C for details)
6	fletch1	

5.5.2 Acknowledgment

This packet (see Table 18) is sent from the m10 to the host controller to acknowledge a MobiApps Extended Packet with an appropriate type code.

Table 18: Acknowledgment Packet

	7 6 5 4 3 2 1 0	
0	0x6	Packet Header (0x6 indicates that byte 4 is to be interpreted as *pkt_seq_num*.)
1	0x19	Packet Type
2	length0	*length0* and *length1* corresponds to entire length of the packet including the checksum, where *length0* is LSB and *length1* is the MSB
3	length1	
4	retry_count / pkt_seq_num	Number of times this packet has been re-sent / packet sequence number
5	typecode	To Be Determined (TBD)
6	fletch0	*fletch0* and *fletch1* are Fletcher checksums, where *fletch0* is the LSB and *fletch1* is the MSB (Refer to Appendix C for details)
7	fletch1	

5.5.3 Self Power Down

This packet (see Table 19) is sent from the m10 to the host controller when it detects a critical condition and needs to power down. After the m10 sends this packet, it will power down. For example, if the m10's temperature is not in the acceptable temperature range (-40°F to 185°F or -40°C to 85°C), then the m10 will initiate a power down.

Table 19: Self Power Down Packet

	7 6 5 4 3 2 1 0	
0	0x6	Packet Header (0x6 indicates that byte 4 is to be interpreted as *pkt_seq_num*.)
1	0x20	Packet Type
2	length0	*length0* and *length1* corresponds to entire length of the packet including the checksum, where *length0* is LSB and *length1* is the MSB
3	length1	
4	retry_count / pkt_seq_num	Number of times this packet has been re-sent / packet sequence number
5	typecode	TBD
6	fletch0	*fletch0* and *fletch1* are Fletcher checksums, where *fletch0* is the LSB and *fletch1* is the MSB (Refer to Appendix C for details)
7	fletch1	

5.5.4 Temperature Query

This packet (see Table 20) is sent from the host controller to query the m10's temperature.

Table 20: Temperature Query Packet

	7 6 5 4 3 2 1	
0	0x85	Packet Header (0x85 indicates that byte 4 is to be interpreted as *retry_count*.)
1	0x21	Packet Type
2	length0	*length0* and *length1* corresponds to entire length of the packet including the checksum, where *length0* is LSB and *length1* is the MSB
3	length1	
4	retry_count / pkt_seq_num	Number of times this packet has been re-sent / packet sequence number
5	fletch0	*fletch0* and *fletch1* are Fletcher checksums, where *fletch0* is the LSB and *fletch1* is the MSB (Refer to Appendix C for details)
6	fletch1	

5.5.5 Temperature Query Response

This packet (see Table 21) is sent from the m10 to the host controller, indicating the m10's temperature in degree Celsius.

Table 21: Temperature Query Response Packet

	7 6 5 4 3 2 1 0	
0	0x6	Packet Header (0x6 indicates that byte 4 is to be interpreted as *pkt_seq_num*.)
1	0x19	Packet Type
2	length0	*length0* and *length1* corresponds to entire length of the packet including the checksum, where *length0* is LSB and *length1* is the MSB
3	length1	
4	retry_count / pkt_seq_num	Number of times this packet has been re-sent / packet sequence number
5	status	Status = 0 indicates success in obtaining the temperature. Status = -1 indicates an error
6	temp0	32-bit floating-point number, where *temp0* is the LSB and *temp3* is the MSB (TBD)
7	temp1	
8	temp2	
9	temp3	
10	fletch0	*fletch0* and *fletch1* are Fletcher checksums, where *fletch0* is the LSB and *fletch1* is the MSB (Refer to Appendix C for details)
11	fletch1	

5.6 Upgrading the m10's Firmware

MobiApps will provide the upgrade protocol between the host controller and the m10 and the source files to the customer as part of the integration process. Please contact MobiApps (see Where to Get Help) for further details.

5.7 The m10 Development Kit (DK10)

The DK10 board (see Figure 15 and Figure 16) provides users with an easy way to evaluate or start using the m10.

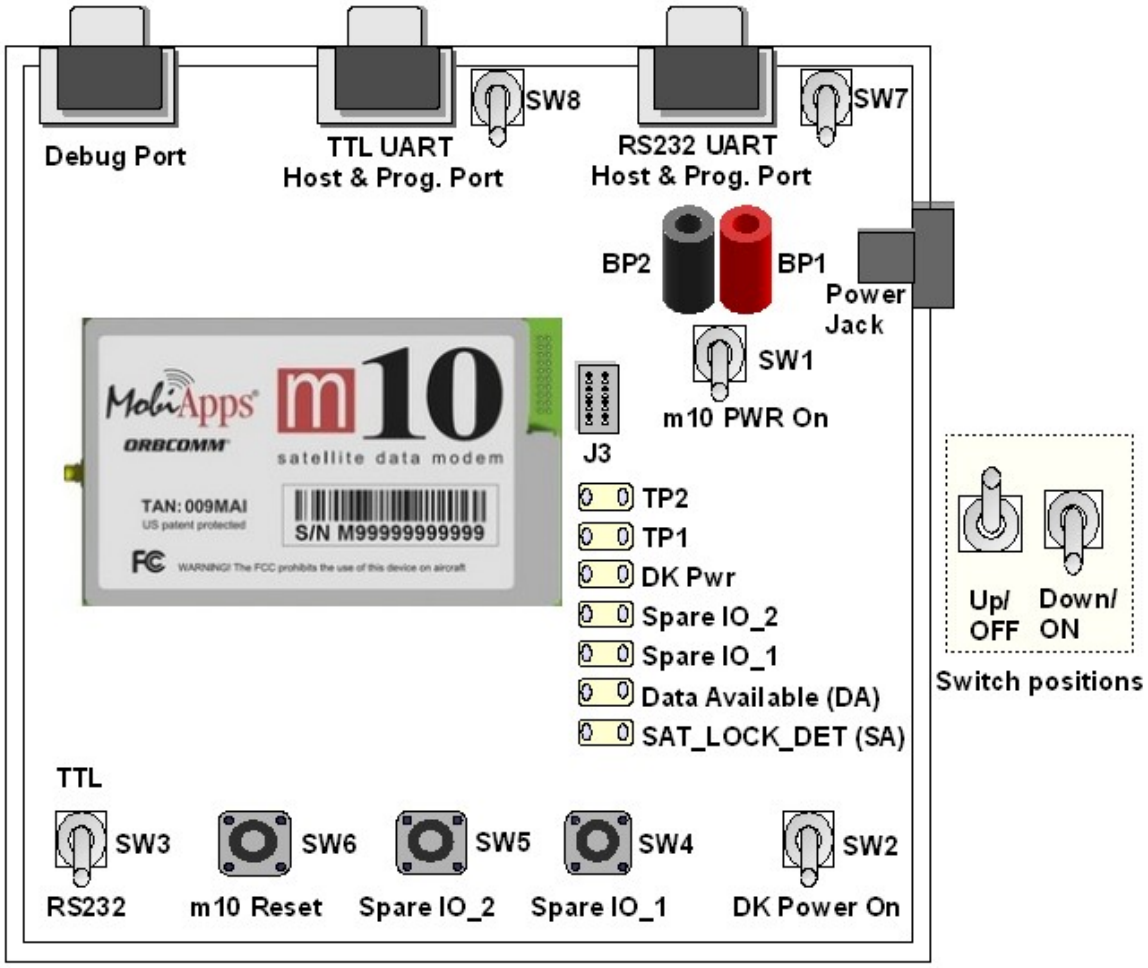

Figure 15: Layout of the DK10

The different components of the DK10 are as follows:

- **SW1**: This switch is used to enable power (PWR_EN) for the m10. However, it is only an enable signal. The actual power is fed through the banana plugs (red and black), which are to be *shorted together* in order to provide power for the m10.
- **SW2**: This switch is used to power on the DK10. The power ON status is indicated by the illumination of the DK Pwr LED.
- **SW3**: This switch is for selecting RS232 level or TTL level UART interface for the host port. If the connection is through the RS232 UART Host & Prog. Port, then this switch (SW3) should be set towards the RS232 position (see Figure 15) and vice-versa.

- **SW4 and SW5**: These are spare I/Os that are reserved for MobiApps' future use. These I/Os are set low by default.
- **SW6**: This switch is used to reset the m10.
- **SW7**: This switch is used to short (down position, as shown in Figure 15) the RTS / CTS for the RS232 UART & Prog. Port.
- **SW8**: This switch is used to short (down position, as shown in Figure 15) the RTS / CTS for the TTL UART & Prog. Port.

If the DK10 is connected using a PC, then the toggle switches on the DK10 should be in the down position (see Figure 15) by default.

- **Power Jack**: A 9 - 18 V supply can be fed through this jack.
- **Debug Port**: The m10's diagnostics can be accessed through this port using a terminal emulation program.

The m10's debug port can be accessed only through the DK10.

- **RS232 UART Host & Prog. Port**: Users can send ORBCOMM serial commands to the m10 at RS232 levels using this port.
- **TTL UART Host & Prog. Port**: Users can send ORBCOMM serial specification commands to the m10 at TTL levels using this port.
- **Banana Plugs**: The external power provided to the DK10 is routed separately to the m10 through two banana plugs, so that users can measure the current drawn only by the m10.
- **TP1 and TP2**: LEDs reserved for MobiApps' future use.
- **DK Pwr**: This LED illuminates when the DK10 is powered on.
- **Spare IO_1 and IO_2**: LEDs reserved for MobiApps' future use.
- **Data Available (DA)**: This LED illuminates when the m10 receives messages, user commands, or GlobalGrams from the satellite. Refer to Section 2.5 for further details.
- **SAT_LOCK_DET (SA)**: This LED illuminates when the m10 is receiving valid downlink signals. Refer to Section 2.6 for further details.
- **J3 (EEPROM)**: This port is used for programming the boot code of the m10. Please contact MobiApps (see Where to Get Help) for further details.

To test the m10 on the DK10, insert the m10 on to the DK10, ensuring that all 20 pins of the m10 connector are inserted properly into the DK10 connector. To ensure proper connectivity, MobiApps recommends that you solder the m10 to the DK10 using the 5 mounting solder legs of the m10.

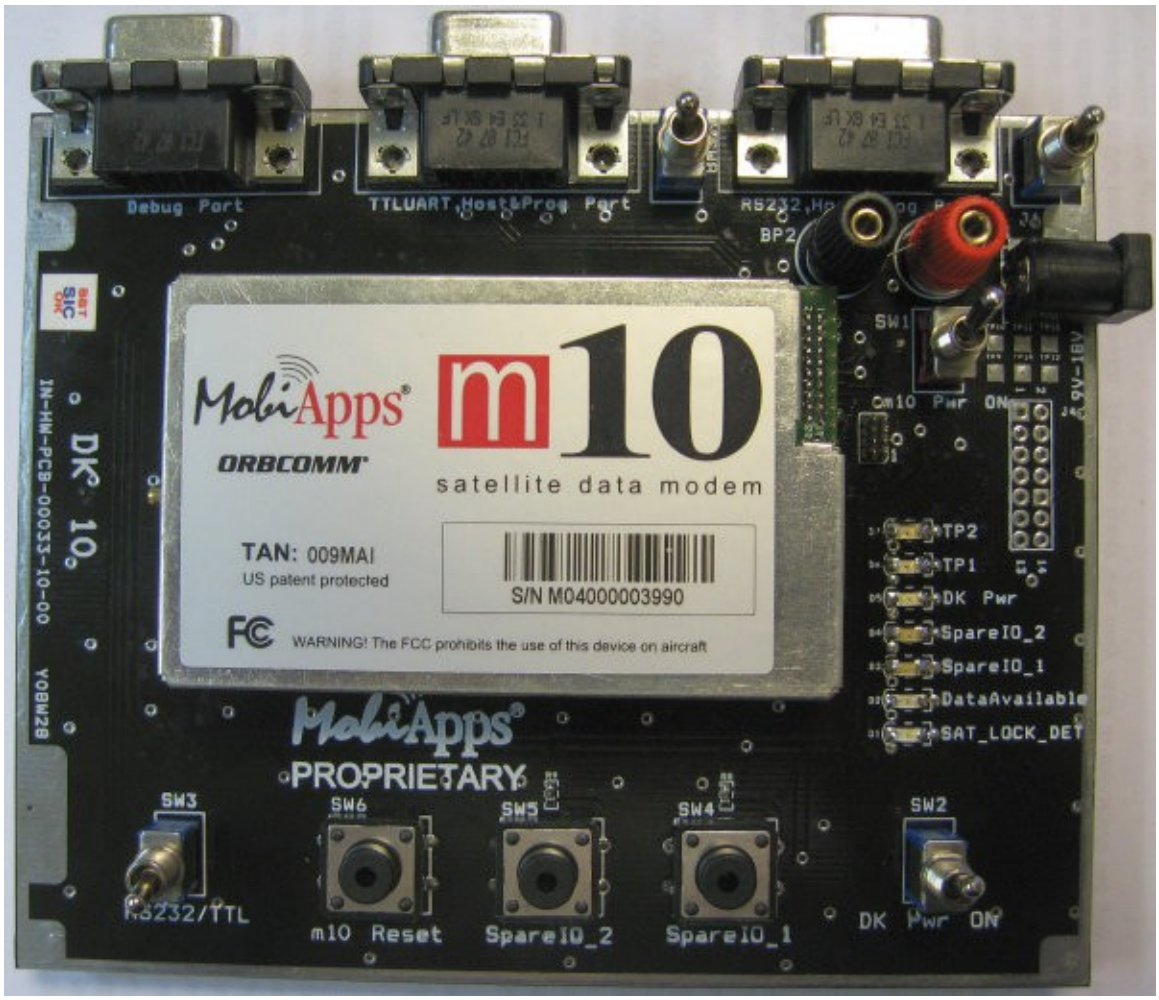

Figure 16: DK10 with the m10

5.7.1 Firmware Upgrade using the DK10

To load the firmware upgrade file from a PC (running Windows XP or later) onto the m10, do the following:

Do not interrupt power supply during the upgrade. If the power supply is interrupted, ensure that the upgrade process is repeated.

1. Place the *m10UpgradeUtil.exe* application and the loader file in the same folder on the PC.
2. With a serial cable (not a null modem), connect the PC's serial port (DB9 connector) with the **RS232 UART Host & Prog. Port** of the DK10 (see Figure 17).
3. If the HyperTerminal is open, disconnect it by clicking the **Disconnect** icon in the HyperTerminal **Tool Bar**.

4. Ensure that the power to the m10 has been switched off by flipping the m10's power switch (SW1 on the DK10) to OFF.
5. Navigate to the folder containing the *m10UpgradeUtil.exe* application. Double-click or open *m10UpgradeUtil.exe* to begin the upgrade.

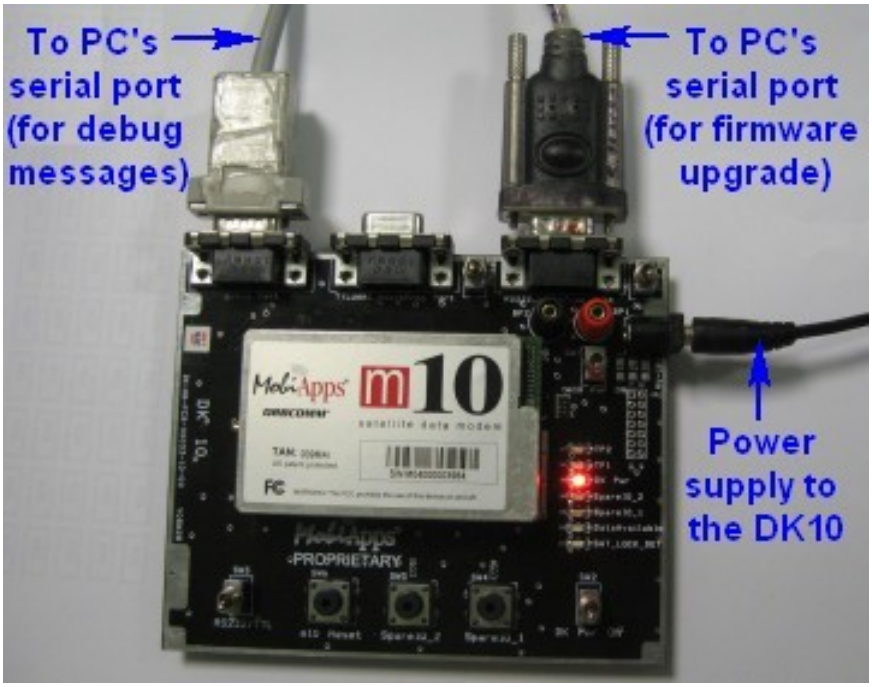

Figure 17: Connecting the PC to the DK10

6. In the command prompt window that appears (see Figure 18), type in the serial port number (1 for COM1, 2 for COM2, etc.) and then press **Enter**. A '*Waiting for ACK*' message is displayed.

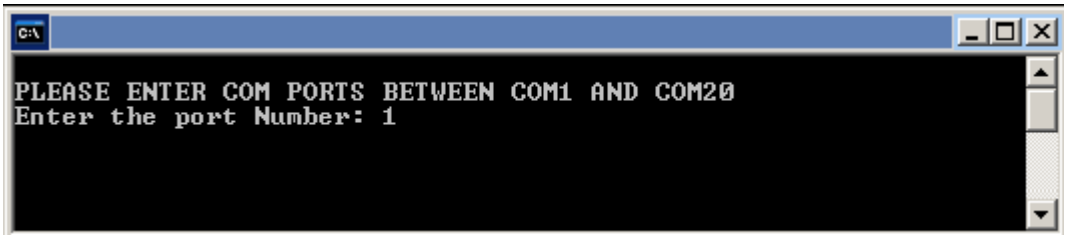

Figure 18: Entering the Port Number

7. Power on the m10 by flipping the m10's power switch (SW1) on the DK10.
8. The m10 is initialized and menu options are displayed in the command window (see Figure 19). Choose option '4' (*Upgrade the Code*) and press **Enter**.

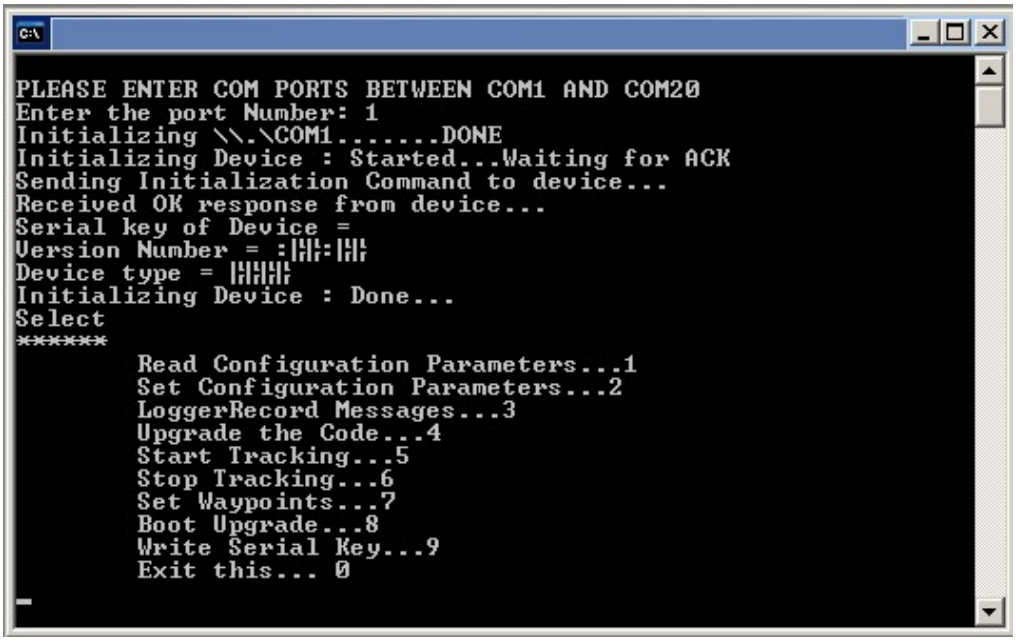

Figure 19: Selecting the Menu Option

9. A message asking for the file name is displayed. Type the complete file name of the loader file, including its extension (e.g. *m10_v1_0_0_0_OC.ldr*) and press **Enter**.
10. The message *'Received OK response from device*' is displayed and the m10 upgrade process commences. The upgrade process takes around 1 minute to complete.
11. When the reprogramming has completed, the message '*Press any key to continue*' is displayed (see Figure 20).

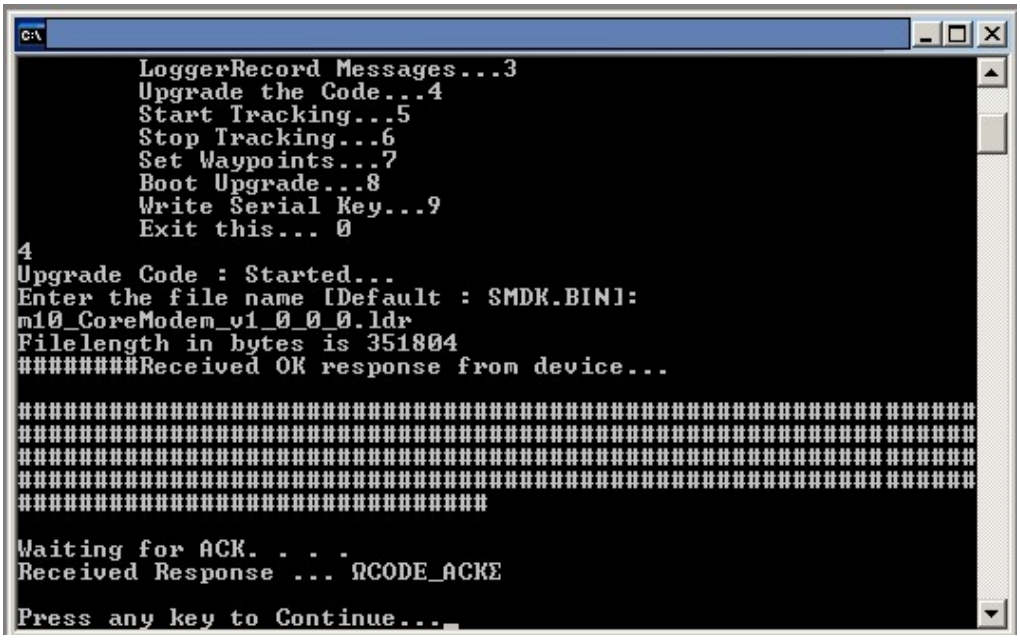

Figure 20: Upgrade Completed

12. Press **Enter** and then type '0' to exit and press **Enter** again.
13. Start HyperTerminal and configure if it has not been configured already. (Refer to Section 5.7.3 for instructions on how to configure HyperTerminal.)
14. Restart the m10 by switching off its power and then switching the power back on.
15. The m10 will dump messages into HyperTerminal (see Figure 21) through the DK10's debug port (see Figure 17). This indicates that the firmware upgrade was successful and that the m10 is working properly.

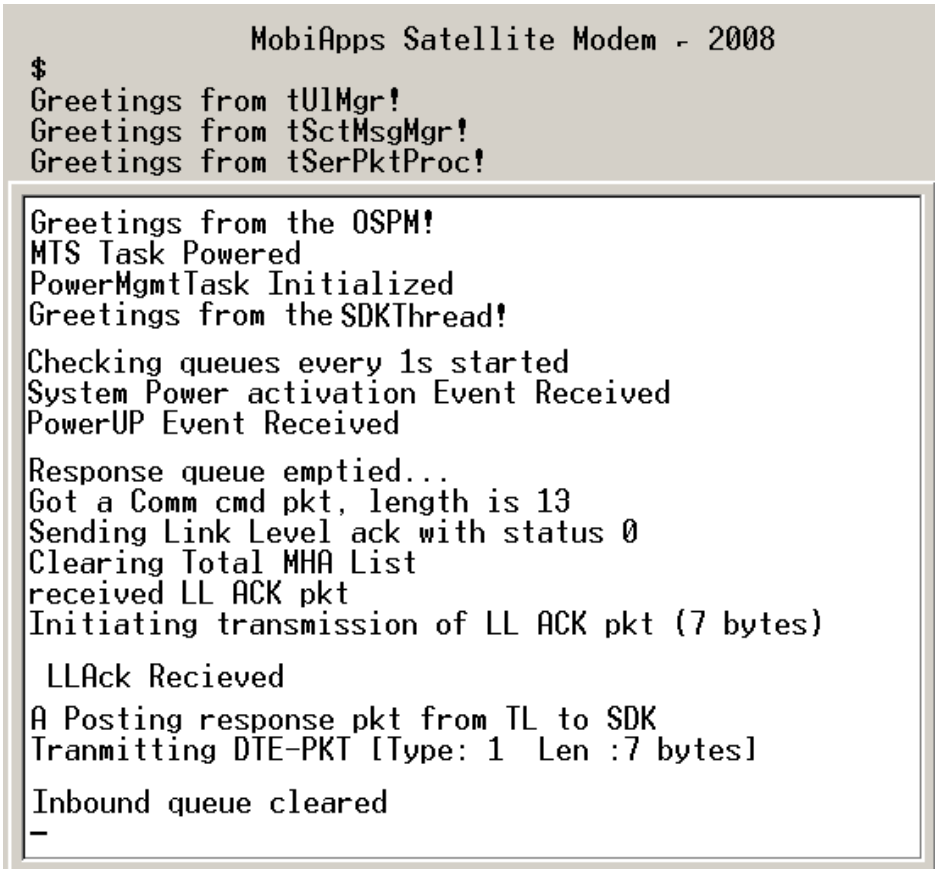

Figure 21: HyperTerminal with m10 Messages

1. **This procedure depends on connecting the m10 to the PC's DB9 connector on the PC's serial port. If the PC does not have a serial port, then a high-quality serial adapter card must be used. For notebook computers with PCMCIA slots, MobiApps recommends the Quatech serial adapter.**
2. **USB-to-serial adapters cannot be used because of timing incompatibilities.**

5.7.2 DK10 Schematic

Please refer to Appendix B for the schematic of the DK10.

5.7.3 Setting up HyperTerminal

To set up HyperTerminal, do the following:

1. Open HyperTerminal from the **Start** menu (in Windows XP) by doing the following. Click **Start**, point to **All Programs**, point to **Accessories**, point to **Communications**, and click **HyperTerminal**.
2. If you are using HyperTerminal for the first time, the *Default Telnet Program* dialog box (see Figure 22) appears. Select the check box and then click **Yes**.

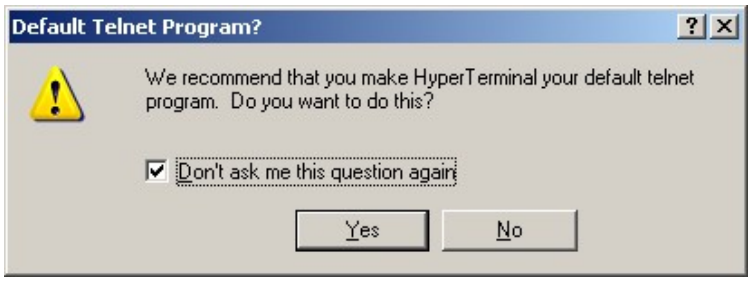

Figure 22: Default Telnet Program Dialog Box

3. In the subsequent *Connection Description* dialog box (see Figure 23), type in a name and select an icon for the connection, and then click **OK**.

Figure 23: Connection Description Dialog Box

4. In the subsequent *Connect To* dialog box (see Figure 24), select the port (e.g. COM1) from the **Connect using** drop-down box and click **OK**.

Figure 24: Connect To Dialog Box

5. In the subsequent dialog box (see Figure 25), select the values in the drop-down boxes as shown in the table below. Click **OK**.

Parameter	Value
Bits per second	115,200
Data bits	8
Parity	None
Stop bits	1
Flow control	None

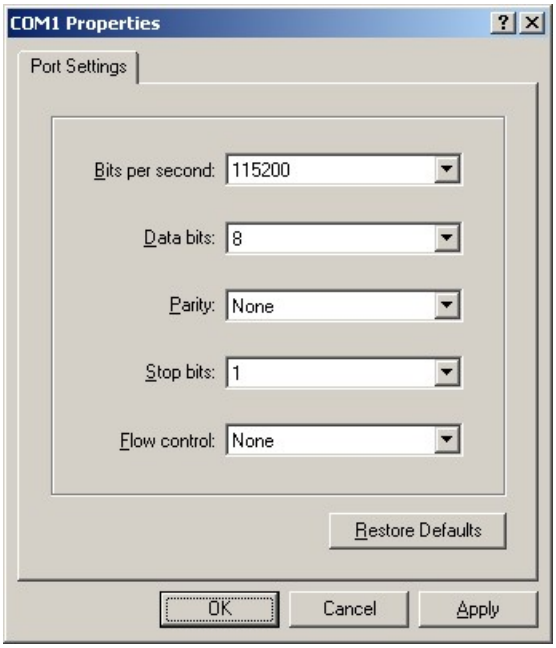

Figure 25: Properties Dialog Box

6. In the subsequent HyperTerminal window, click the **Properties** icon (see Figure 26).

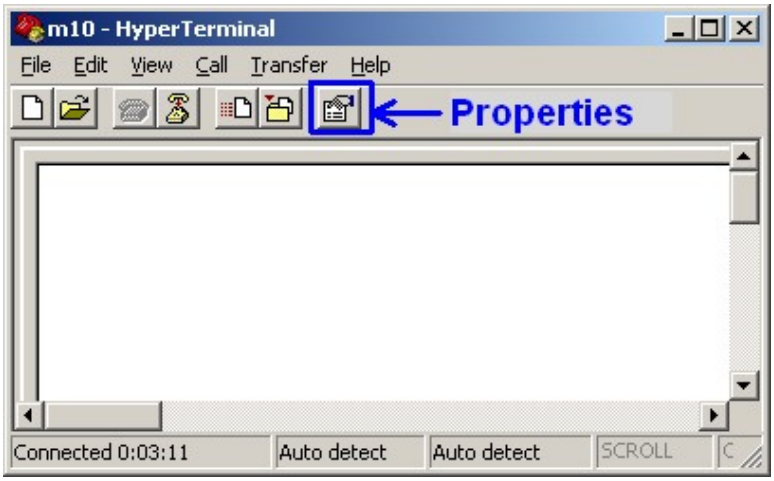

Figure 26: Blank HyperTerminal Window

7. In the subsequent dialog box (see Figure 27), select the **Settings** tab and click the **ASCII Setup** button.

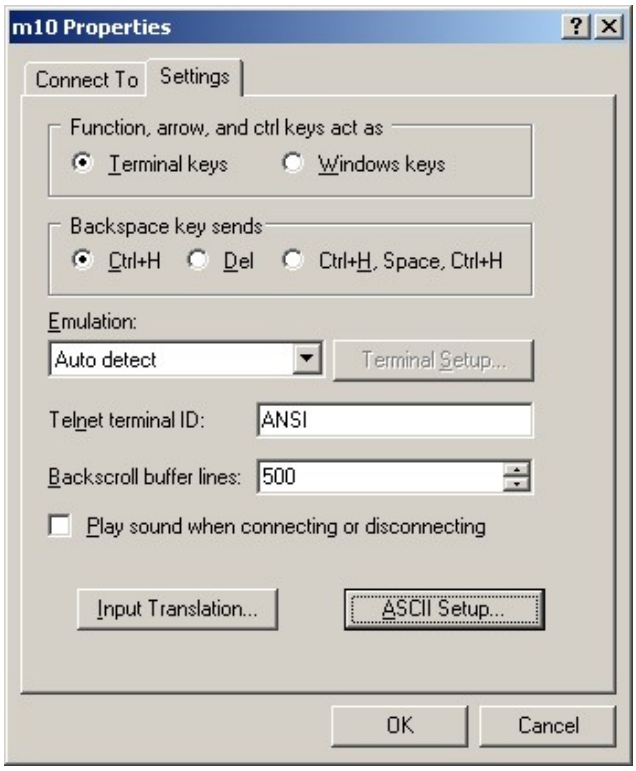

Figure 27: Properties Dialog Box

8. In the *ASCII Setup* dialog box (see Figure 28), clear the *Send line ends with line feeds* check box. Enter `40 milliseconds` in the **Line delay** text box and `100 milliseconds` in the **Character delay** text box. Click **OK** twice, once in each dialog box.

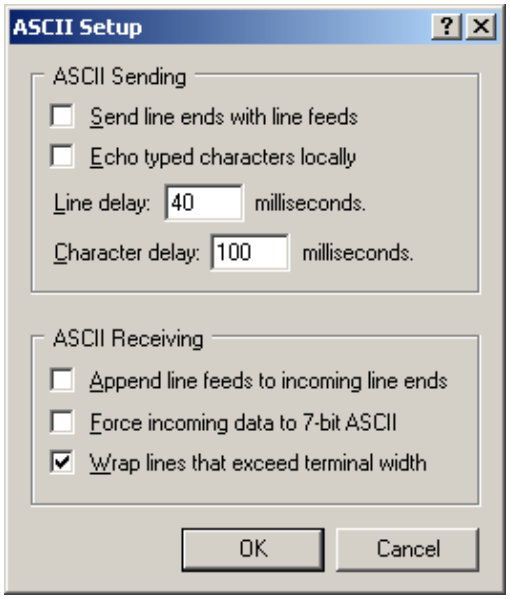

Figure 28: ASCII Setup Dialog Box

The HyperTerminal set up is now complete and the HyperTerminal is ready for use.

Instead of the default Windows HyperTerminal application, MobiApps recommends the use of rigorous terminal emulation applications such as PuTTY and Tera Term.

6. Troubleshooting

This chapter helps in troubleshooting common problems that may be faced when using the m10.

6.1 Connection / Communication

Type	Cause	Remedy
Antenna	ORBCOMM antenna is not connected	Connect the ORBCOMM antenna
	TNC connector is not securely attached to antenna cable	Crimp / solder new connector to antenna cable
	Antenna cable is wound too tightly or has kinks	Unwind and replace cable, if necessary
	Antenna is too close to second VHF antenna	Separate antennas by at least 2 meters
	Antenna cable is too lossy	Check cable data sheet and verify that total cable loss at ORBCOMM frequencies is less than 3 dB.
	Antenna SWR not optimized for ORBCOMM transmit / receive frequencies	Check antenna with SWR meter, and verify SWR is < 2 for ORBCOMM transmit (148 - 150 MHz) and receive (137 - 138 MHz) frequencies
	Antenna line-of-sight obstructed	Clear any obstacles or move antenna
	m10 may not be receiving satellite signals.	Check the SA pin and check that the antenna has been properly connected.
Satellites	Satellite is not in view	Wait until satellite is in view
Provisioning	m10 is not provisioned with ORBCOMM	Contact ORBCOMM to provision unit
	m10 is not provisioned for destination gateway	Contact ORBCOMM to add destination gateway to provisioning
	Speed dial or destination e-mail address is invalid	Correct speed dial or destination e-mail address
Messaging	No message is in inbound queue	Create message
	Communicator's GCC ID does not match local GCC	Check and change message's destination gateway to correct GCC
	Satellite is in direct mode and pending message is a GlobalGram	Create direct mode message or wait for satellite to switch to GlobalGram mode (location dependent)
	Satellite is in GlobalGram mode and pending message is a GlobalGram	Create GlobalGram mode message or wait for satellite to switch to direct mode (location dependent)

6.2 Upgrade

Type	Cause	Remedy
m10UpgradeUtil.exe closes immediately when the port number is selected.	Terminal emulation application is still connected to the PC's serial port	Disconnect the terminal emulation application and then restart the upgrade process.
	The m10's power is still ON.	Turn the power OFF and restart the upgrade process.
After upgrading the m10, there is no output on the terminal emulation application.	The serial cable has not been connected to the debug port of the DK10.	Connect it and check if there is an output on the terminal emulation application.
	The incorrect loader file was uploaded to the m10.	Restart the upgrade using the correct loader file. Please contact MobiApps (see Where to Get Help) for the latest firmware.
	m10 has not been reset after upgrading.	Manually reset the m10 (using the DK10) after the upgrade process.
The firmware upload was interrupted and the m10 does not respond anymore.	The firmware upgrade was not completed successfully.	Repeat the upgrade process from the beginning and the m10 should respond when the upgrade has completed successfully.

6.3 Power

Type	Cause	Remedy
Power	Power is below 9 V or above 18 V.	Ensure power in valid range for 9 V or 18 V systems
	Power and ground reversed	The m10 will be damaged beyond repair and will need to be replaced.
	Power supply is current limiting	Verify that power supply is capable of sourcing 2 A current. Verify that current limiter is set above 2 A.
	Power supply is unable to source current quickly. (Many regulated power supplies cannot respond within microseconds to 2 A current burst required for ORBCOMM transmissions.)	Use high quality power supply or good quality power adapter. (Phihong PSA60W-150 is recommended.)

7. FAQs

This chapter addresses common questions regarding the usage of the m10.

7.1 Connection / Communication

1. **The host controller is unable to communicate with the m10 through the serial port.**

   The host port of the m10 is at TTL level. Therefore, please ensure that the host controller is not communicating at RS232 levels. If it is, then it is necessary to level-translate to TTL.

2. **Can I change parameters and control the m10 over the air?**

   No, this cannot be done.

3. **How do I send a message from the host controller to the m10?**

   The m10 conforms to the ORBCOMM serial specifications protocol. Please refer to the *ORBCOMM Serial Interface Specifications* document

4. **Why does the m10 receive the satellite signal but cannot transmit any messages or reports?**

   Check that the gateway ID is set correctly and that messages have been queued in the m10. Sometimes, the ORBCOMM satellites may not send the required uplink channel information to the m10. This can happen due to congestion in the ORBCOMM network.

5. **Can the m10 receive messages if it is powered off?**

   No, it cannot.

6. **How do I check for messages that have been received by the m10?**

   The host controller can poll the m10 for SC-terminated messages, commands, or Global Grams by sending a Communications Command packet (see Table 16). (Refer to *Table 3.2* in the *ORBCOMM Serial Interface Specifications* document for details.)

7. **My host controller only has an RS232 interface. Is it possible to interface with the m10?**

   Yes, you need an RS232 level shifter to convert the RS232 level to TTL level. This level shifter will have to be introduced on the host controller PCB.

7.2 Antenna

8. **Which antenna should be used with the m10?**

   Please refer to *ORBCOMM Application Note #12*, which can be downloaded from http://www.orbcomm.com/equipment/pdf/App_Note_12__ORBCOMM_Antennas.pdf. It provides details and comparisons of antennas from different vendors.

9. **If the antenna cable is not long enough to reach my installation, how do I pick a suitable RF cable to each it?**

   You can use any 50 Ω impedance, RF coax cable, provided the overall loss from the antenna to the m10 is less than 3 dB at 150 MHz. The lower the cable loss, the costlier the cable, so there is a trade off between price and performance. LMR-400 would be an ideal choice for a 100 feet long cable. LMR-195 is an excellent choice for shorter lengths.

10. **What should I consider when I mount my antenna?**

    The antenna should be kept away from metal objects and the objects should not be parallel to the antenna, as this will degrade the performance of the antenna. If a VSWR meter is available, measure the VSWR of the antenna at 137.5 MHz and 149 MHz. A VSWR reading below 2 is required for good performance from the antenna. (Refer to Section 2.7 for details.)

7.3 Miscellaneous

11. **Who can help me troubleshoot my hardware problem?**

    Please go through this chapter to check if your problem has been addressed. In case you are unable to solve the problem, please contact MobiApps (see Where to Get Help).

12. **How do I provision the m10?**

    Please contact ORBCOMM (http://www.orbcomm.com/) to verify that the m10 has been provisioned.

13. **One of the m10's connector pins was damaged during transit. How can I get a replacement?**

    Please contact MobiApps (see Where to Get Help).

14. **Is the m10 waterproof?**

    No, it is not.

15. **Can the m10 interface with any sensors?**

    Please contact MobiApps (see Where to Get Help) for details.

A. Appendix: m10 Mechanical Drawings

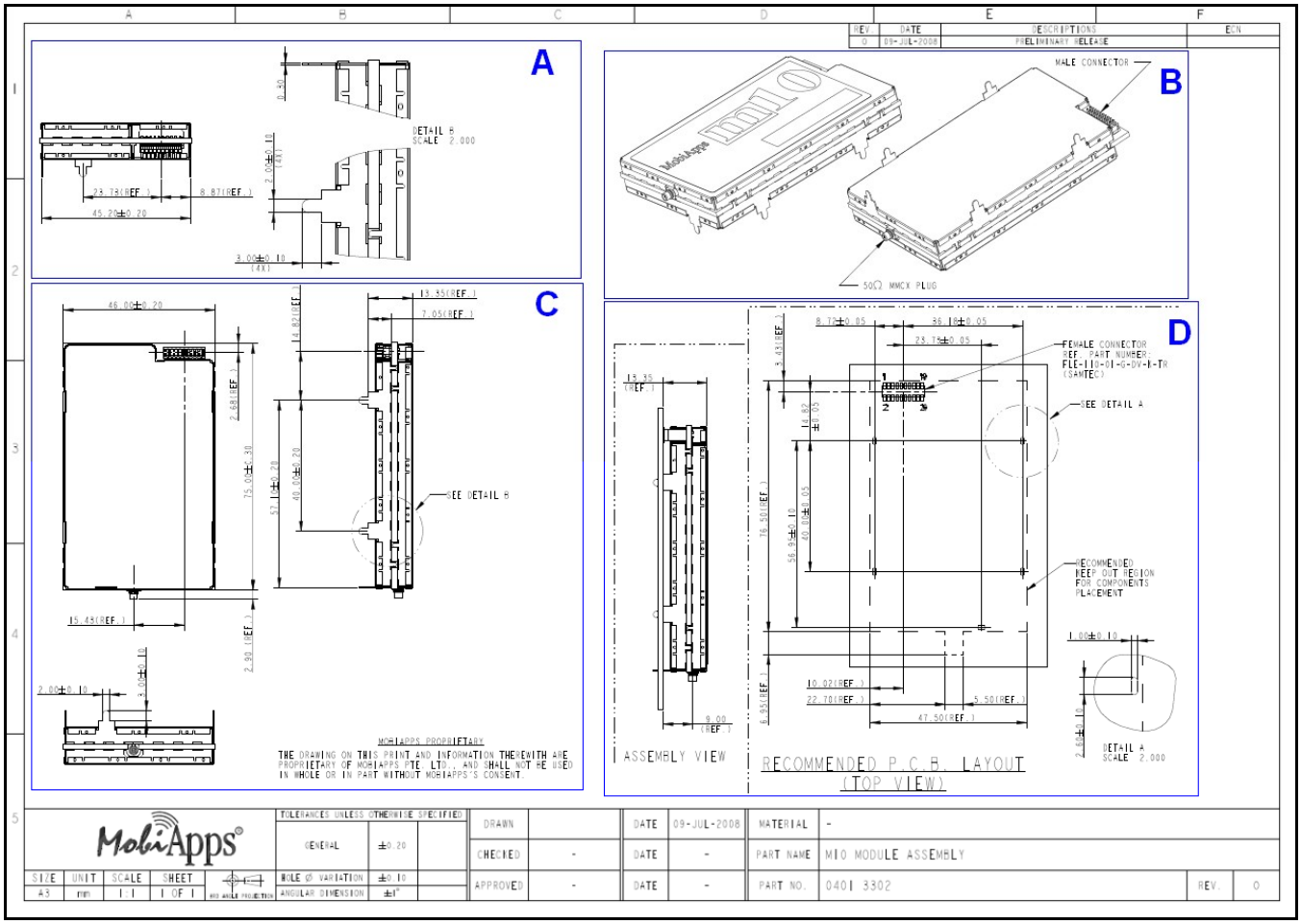

Figure 29: m10 Mechanical Drawing

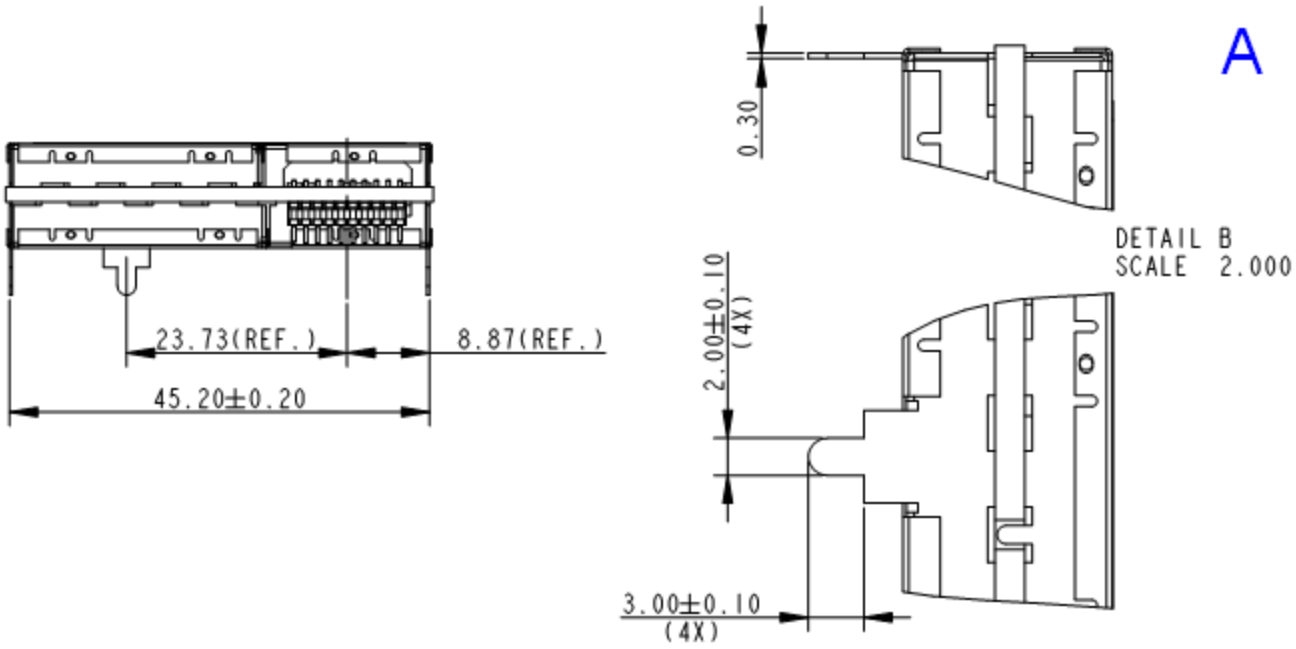

Figure 30: m10 Mechanical Drawing: Part A

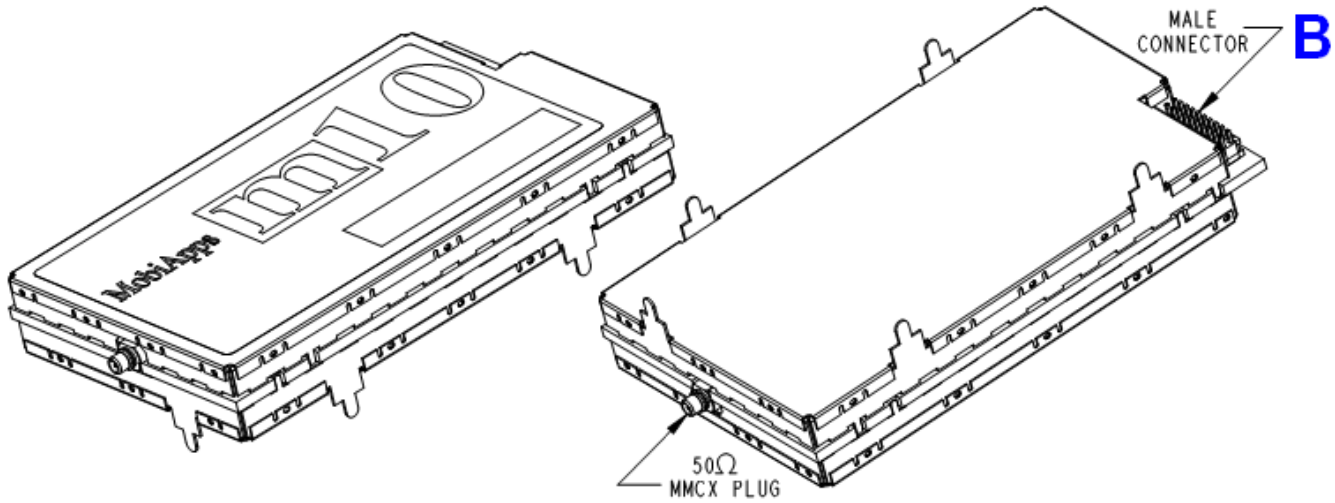

Figure 31: m10 Mechanical Drawing: Part B

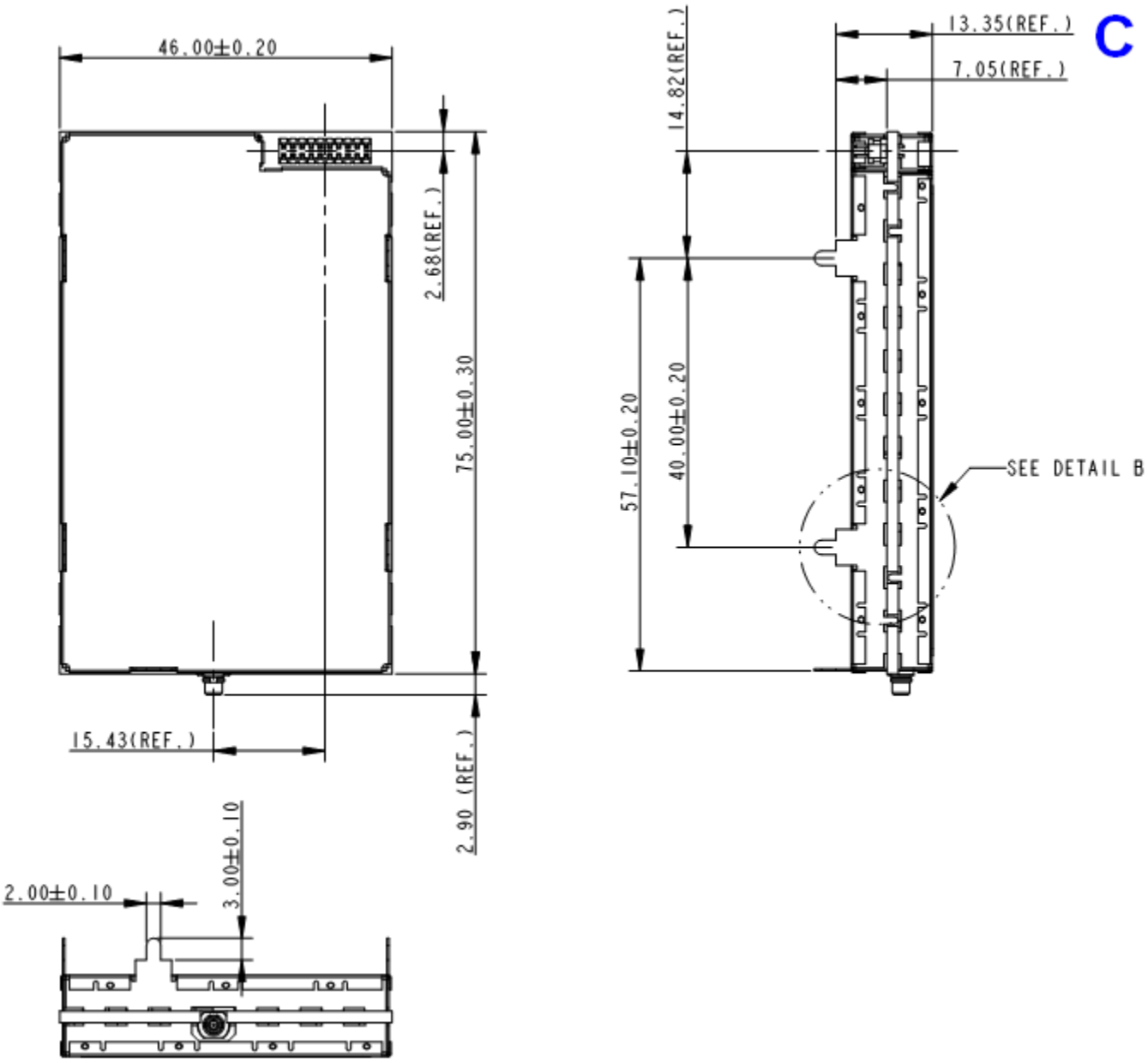

Figure 32: m10 Mechanical Drawing: Part C

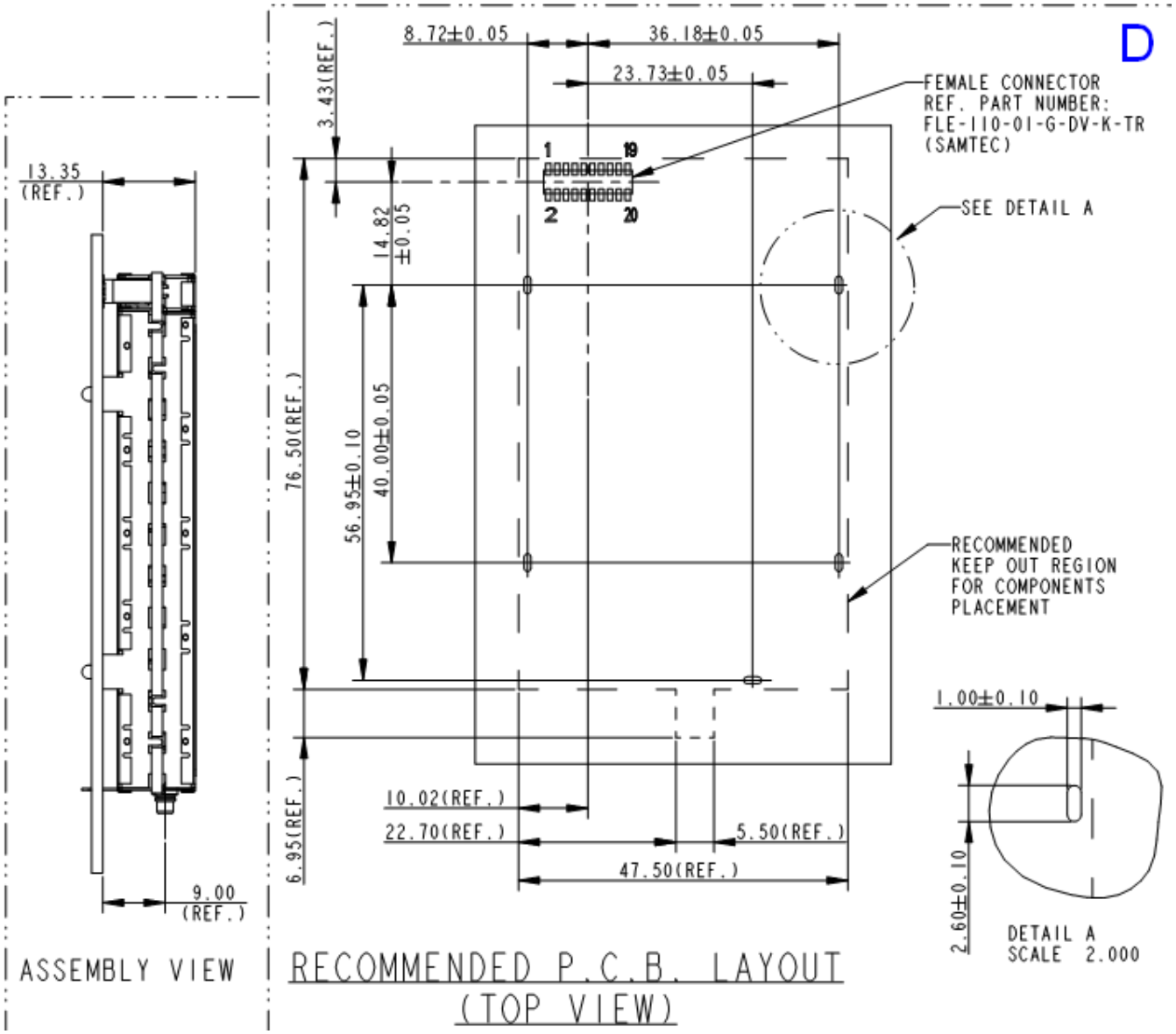

Figure 33: m10 Mechanical Drawing: Part D

B. Appendix: DK10 Schematic

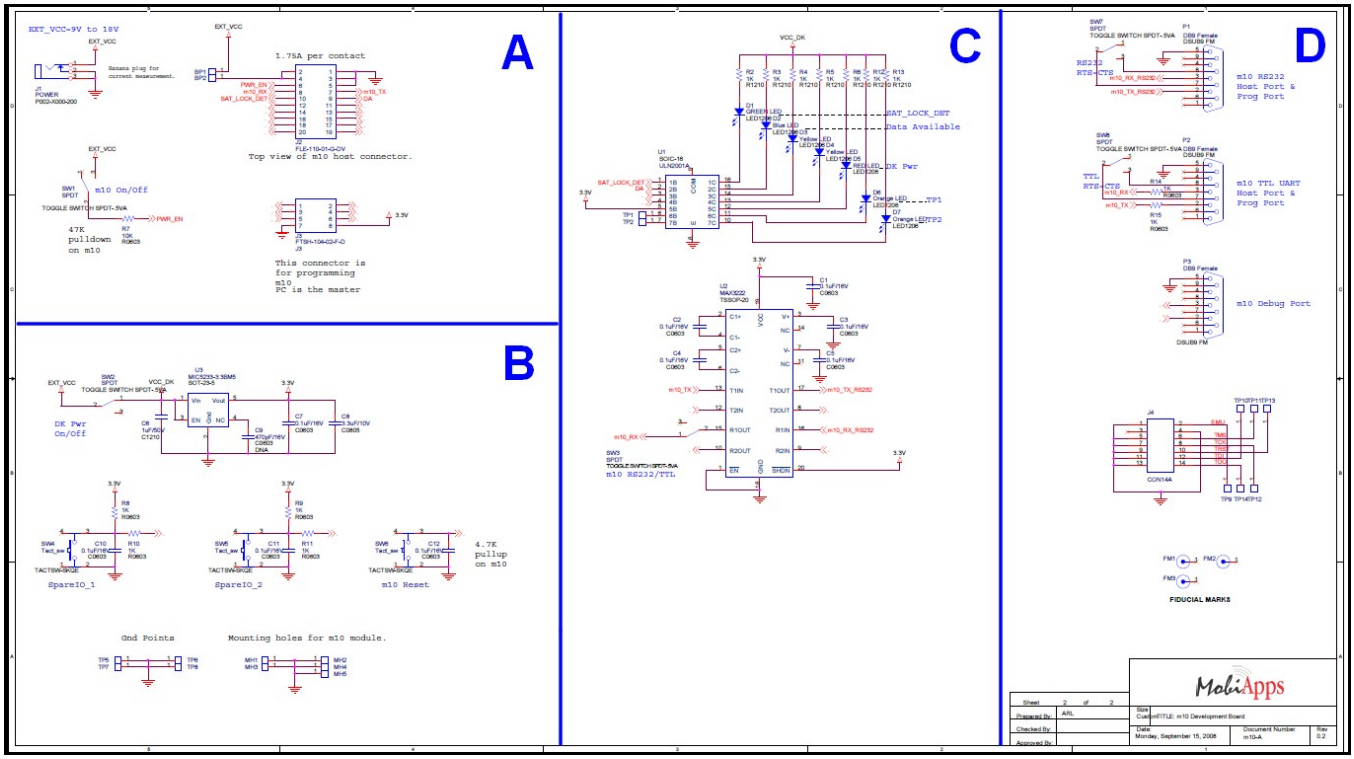

Figure 34: DK10 Schematic

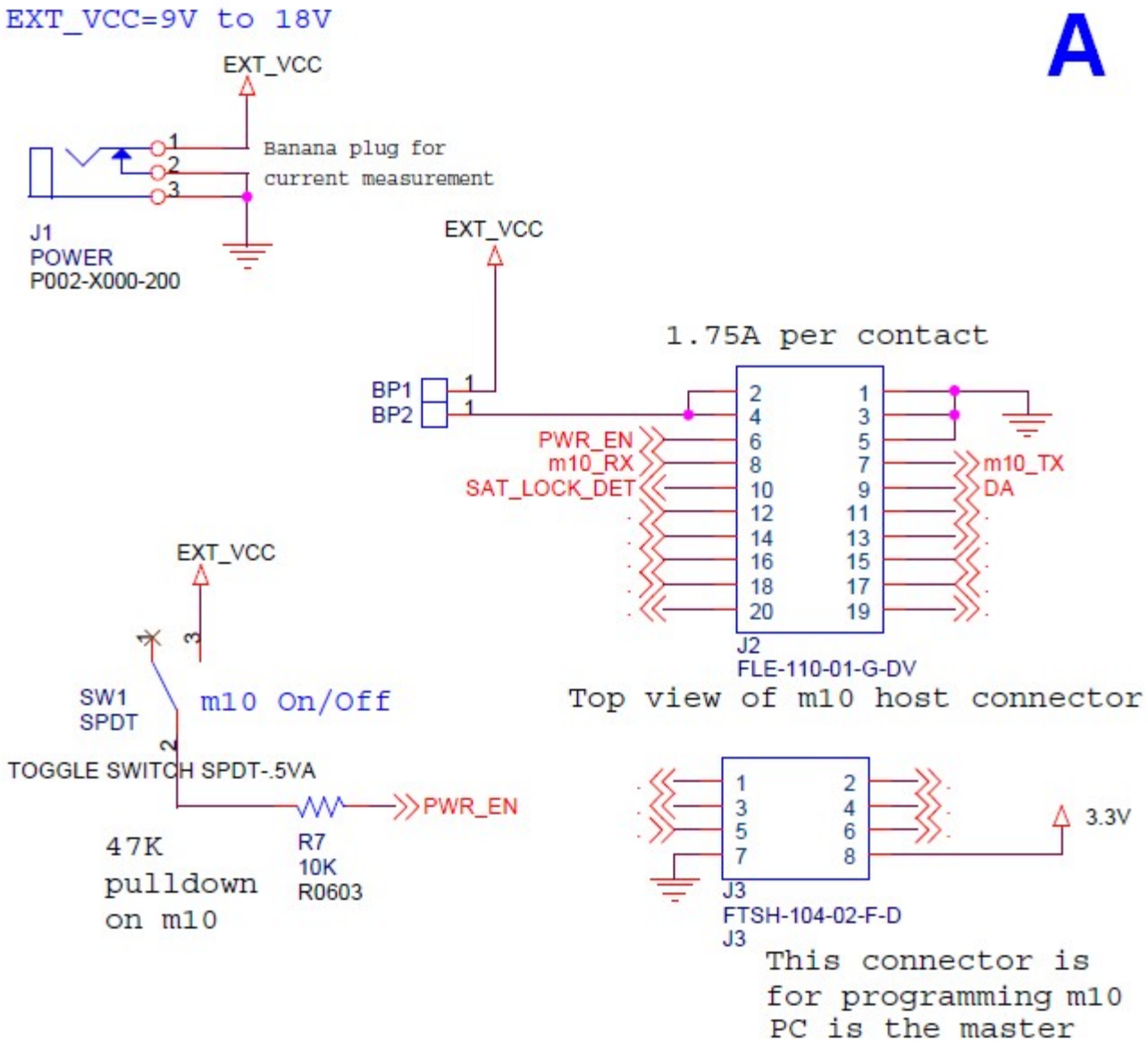

Figure 35: DK10 Schematic (Part A)

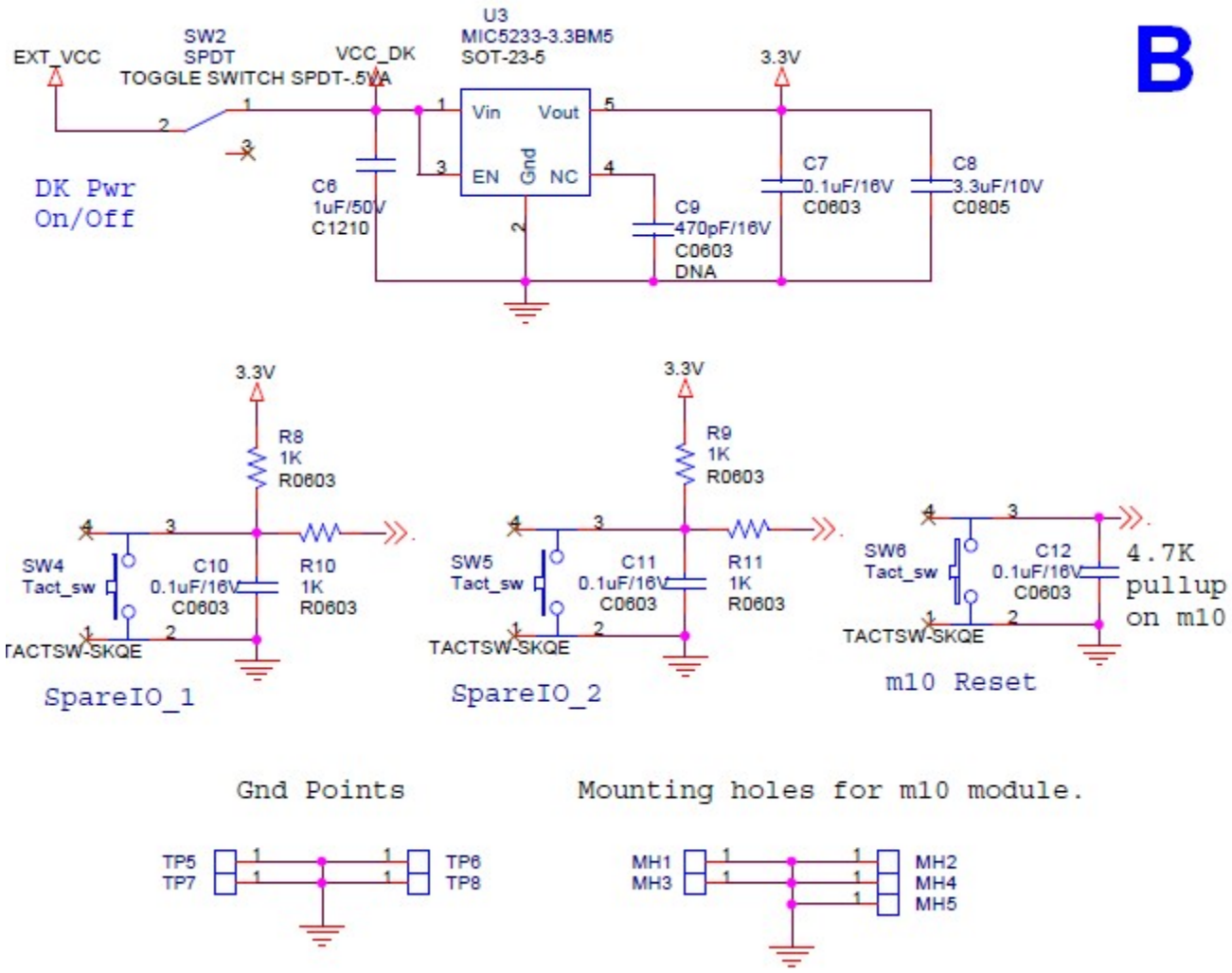

Figure 36: DK10 Schematic (Part B)

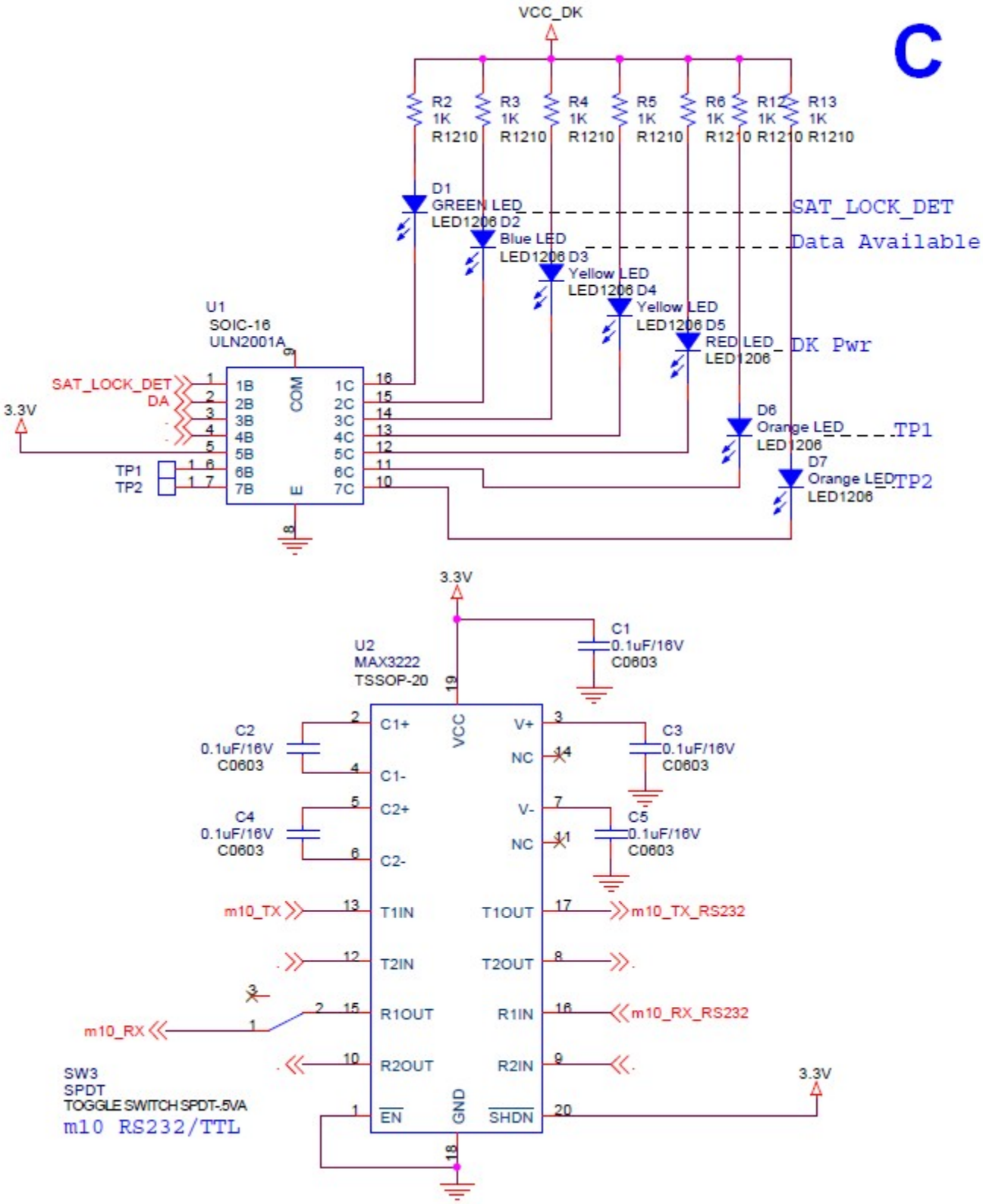

Figure 37: DK10 Schematic (Part C)

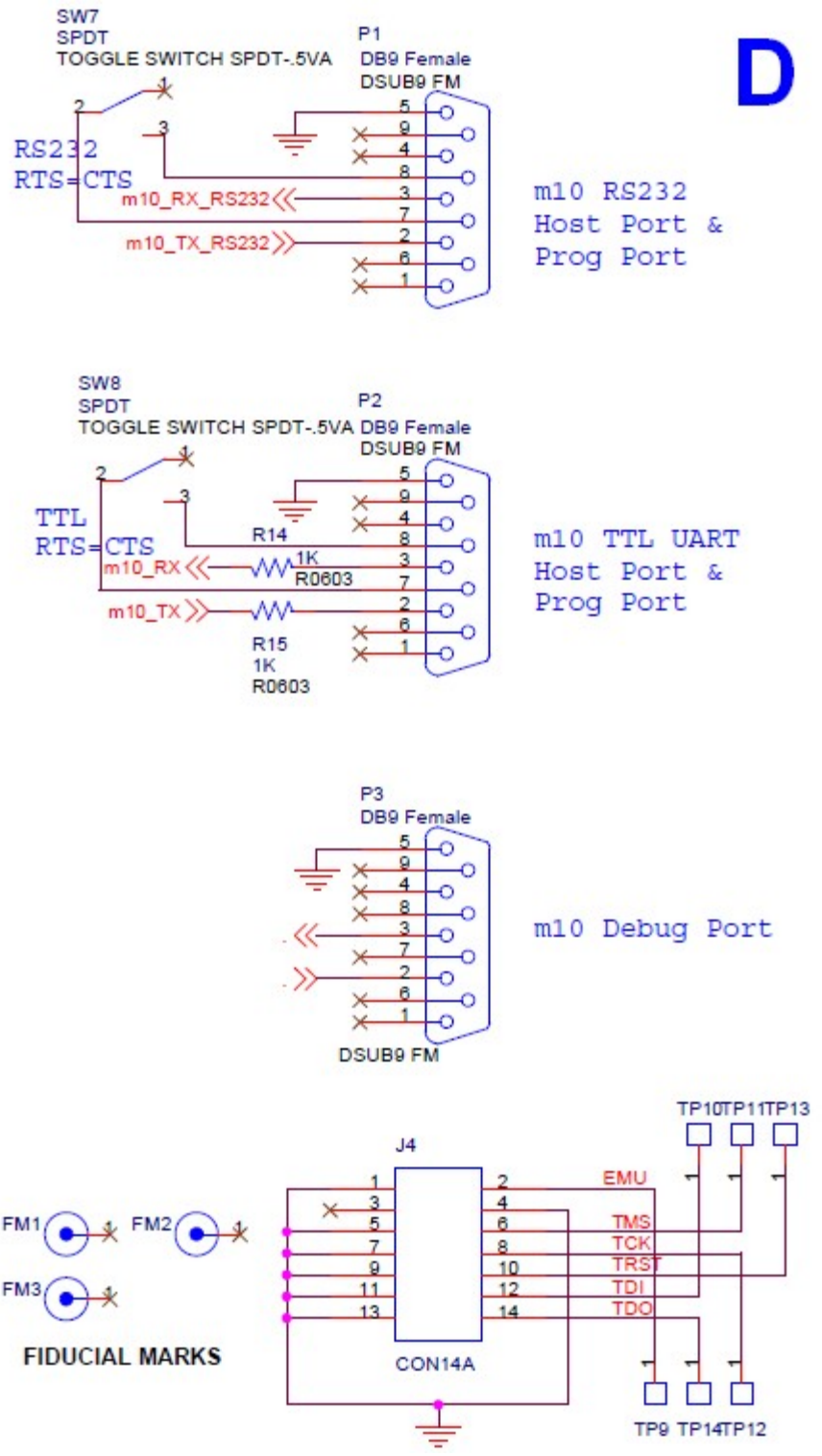

Figure 38: DK10 Schematic (Part D)

C. Appendix: Fletcher Checksum

```
/*
* operator fletcher_encode
*/
fletcher_encode( unsigned char* buffer, long count )
{
   int i;
   unsigned char c0 = 0;
   unsigned char c1 = 0;
   *( buffer + count - 1 ) = 0;
   *( buffer + count - 2 ) = 0;
   for( i = 0; i < count; i++)
   {
        c0 = c0 + *( buffer + i );
        c1 = c1 + c0;
   }
   *( buffer + count - 2 ) = c0 - c1;
   *( buffer + count - 1 ) = c1 - 2*c0;
}
/*
* operator fletcher_decode
*/
long fletcher_decode( unsigned char* buffer, long count )
{
   long result = 0;
   int i;
   unsigned char c0 = 0;
   unsigned char c1 = 0;
   for( i = 0; i < count; i++)
   {
        c0 = c0 + *( buffer + i );
        c1 = c1 + c0;
   }
return( (long)(c0 + c1) );
}
```

D. Appendix: Glossary

The following abbreviations and / or acronyms have been used in this user manual.

Short Form	Expanded Form
ACK	Acknowledgment
ACQ	Acquisition
ASIC	Application-Specific Integrated Circuit
DB9	A 9-pin physical connector that is common on PCs and a secondary connector specified in RS232.
DSP	Digital Signal Processor
DTE	Data Terminal Equipment. Any digital device that transmits and receives data.
EEPROM	Electrically-Erasable Programmable Read Only Memory
ESD	Electrostatic Discharge
GND	Common Ground
GPIO	General Purpose Input / Output
I/O	Input / Output
IEC	International Electrotechnical Commission
IB	Inbound
Inbound	Direction of message traffic from the m10 to the ORBCOMM gateway
LEO	Low Earth Orbit
LSB	Least Significant Byte
M2M	Machine-to-Machine
MAC	Multiply Accumulate
MCU	Microcontroller Unit
MHA	Message Handler
MTBF	Mean Time Between Failure
MSB	Most Significant Byte
NVM	Non-Volatile Memory

Short Form	Expanded Form
OB	Outbound
OS	Operating System
Outbound	Direction of message traffic from ORBCOMM gateway to the m10
PCB	Printed Circuit Board
RF	Radio Frequency
RISC	Reduced Instruction Set Computer
RMS	Root Mean Square
RoHS	Restriction of Hazardous Substances in Electrical and Electronic Equipments
RTOS	Real-Time Operating System
RX	Receive
SAE	Society of Automotive Engineers
SC	Subscriber Communicator (ORBCOMM's term for its data modems)
SCO	SC-Originated
SCT	SC-Terminated
SDRAM	Synchronous Dynamic Random Access Memory
SPI	Serial Peripheral Interface
SPORT	Synchronous Serial Port
SWR	Standing Wave Ratio
TBD	To Be Determined
TCXO	Temperature-Compensated Crystal Oscillator
T/R	Transmit / Receive
TTL	Transistor-Transistor Logic
TX	Transmit
UART	Universal Asynchronous Receiver-Transmitter
VSWR	Voltage Standing Wave Ratio