

# Chapter 3

## Design and Implementation of the FPGA Module

### 3.1 FM10 Board Overview

The FM10 is a compact-sized FPGA processing module in the ioNeurons<sup>TM</sup> series. It is capable of performing Hard-Real-Time digital logic operations in parallel; It also contains huge resources of Logic Gates(LGs), an 8-channel Analog-to-Digital Converter (ADC), and a 4MB Static Random Access Memory (SRAM) to implement complex digital computations. The primary function of the ADC is to translate analog quantities to digital data for further data processing. The SRAM provides users high-speed access memory with fully static operations, no clock or refresh required. FM10 can take a wide range of voltage input from 3.3V to 16.0V benefiting from an efficient battery management chip; Meanwhile it can provide a 3.3V DC power source at maximum 1A current.

The physical dimension of FM10 is 40.0 mm x 55.0 mm. It has one 24-pin Flat Flex Cable (FFC) connector and two 30-pin P5K series connectors, which permit 40 GPIOs

in total and outputs of 3.3V or 5.0V voltage individually. FM10 can easily transfer data to computers because it integrates a Future Technology Devices International (FTDI) USB-to-Serial port chip and a USB type-AB connector.

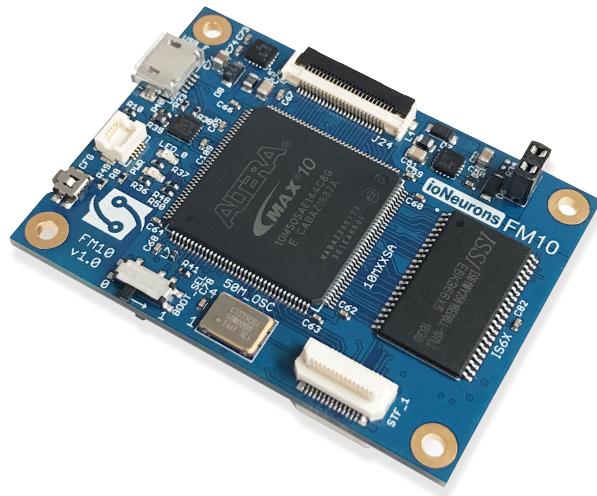


Figure 3.1: ioNeurons™ FM10

The FM10 can work standalone; however, with the design of the stackable connector, it can also collaborate with multiple FM10 or EM4 modules, which can enhance the computing capacity and increase the number of Input/Output interfaces. FM10 is equipped with user programmable LEDs, push buttons and slide buttons, which allows developers to apply different configurations to meet specific requirements.

The FM10 uses an Intel Max 10 series FPGA chip 10M25SAE114I7G (144-pin EQFP); a single-chip, low-cost, programmable logic device with 25000 Logic Elements(LEs). It also integrates a non-volatile user flash memory and ADCs in a hard IP core block. The ADCs allow the FPGA to process analog signals from real-world sensors. It provides 8-channel 12-bit digital representation of the analog signal being observed with a cumulative sampling rate of up to 1 million samples per seconds (MSPS).

### 3.2 Layout and Components

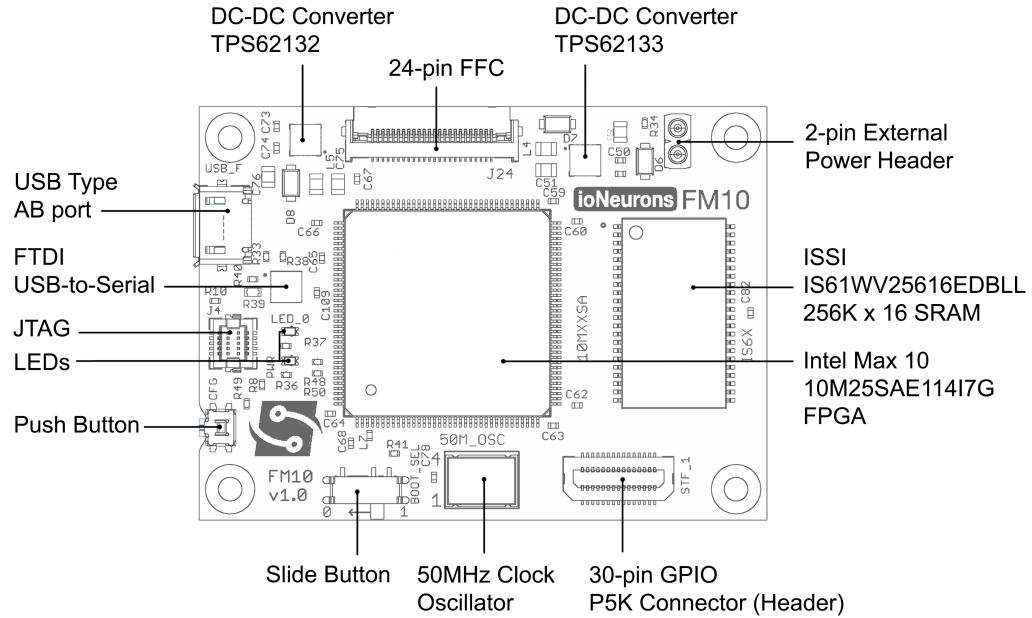


Figure 3.2: FM10 module PCB and components (top view)

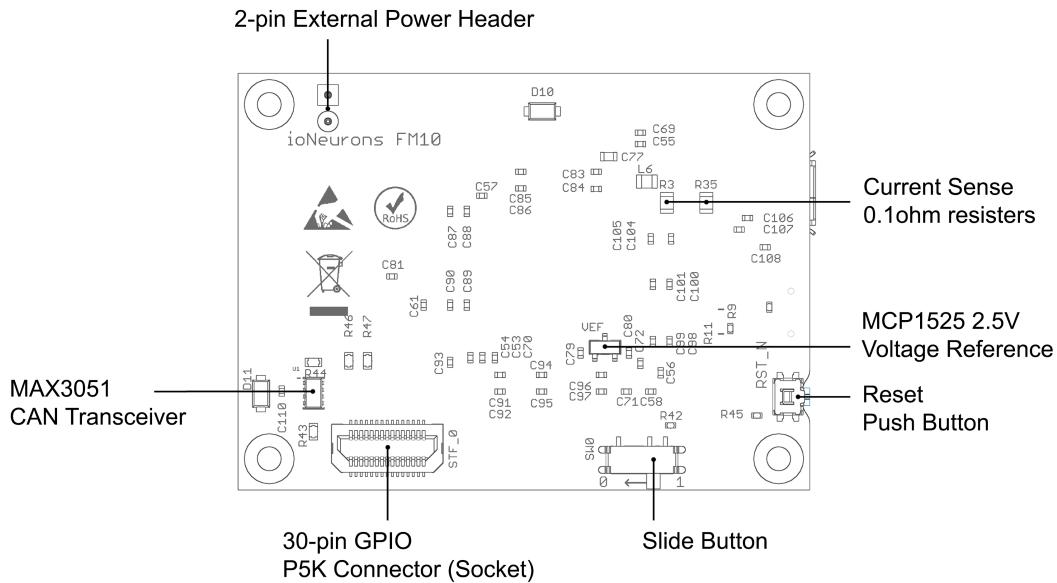


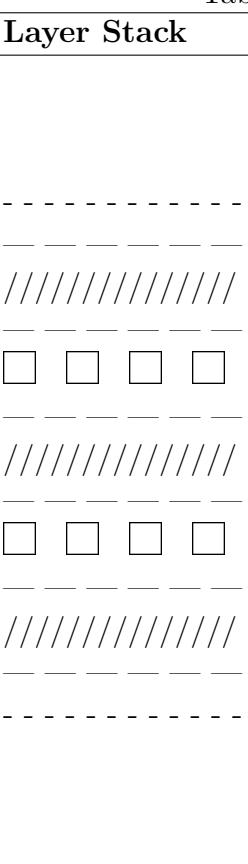
Figure 3.3: FM10 module PCB and components (bottom view)

It is vitally important that every component is placed in a suitable position on a

printed circuit board since it determines the overall dimension, number of layers, and complexity of tracing. Taking production cost and manufacturing techniques into account while keeping the overall size as small as possible, the FM10 employs a 6 layer PCB structure and plating through holes only; no blind via holes or buried via holes were used since they significantly increase the cost of production. The minimum diameter of through holes is 0.15 mm. The solder resist layer covers any through holes which are under 0.35 mm in diameter to protect them and avoid a short circuit, and the stop layer masks will not be applied to these hole locations. The minimum width of a trace is 0.1524 mm, and the minimum clearance between different signal wires is 0.1778 mm. The minimum distance between objects on signal layers and the board dimension is 0.2032 mm, and the thermal isolation is 0.1778 mm. The FR4 insulation material is applied between any two conductor layers. Table 3.1 shows the PCB layer stack. Electroless Nickel Immersion Gold (ENIG) surface treatment is applied on the top and bottom layer during PCB manufacture to prevent the nickel from oxidizing with a thin layer of immersion gold.

When determining the relative position of components, the one having the most pins takes priority because its position has the most significant impact on tracing. The FPGA chip (10M23SAE114I7G) has 144 pins; it is the chip with the largest number of pins on this board. If one side of this chip, where pin number 1 is located, is defined as the chip's left side, most of the high-speed pin and GPIO of the FPGA are located on the bottom side (pin37 to pin72) and right side (pin73 to pin108). Relatively, the SRAM and the FFC connector are the two components that have the next most pins; to reduce the distance between them and the FPGA, the SRAM is placed below the FPGA chip while the FFC connector is placed on the right of the FPGA. In addition, the 50 MHz crystal is placed as close to the FPGA as possible because of its high-frequency clock signal, which is a critical factor to maintain stable

Table 3.1: FM10 PCB Layer Stack

<b>Layer</b>	<b>Layer Stack</b>	<b>Layer Name</b>	<b>Material</b>	<b>Thickness</b>	<b>Plane</b>
1		Top paster	Sn63/Pb37	0.12mm	
2		Top overlay			
3		Top solder	Solder resist	0.01mm	
4		Top ENIG	Gold	0.05µm	
5		Top	Copper	0.018mm	Signal
6		Dielectric 1	FR-4 TG140	0.0913mm	
7		Layer 2	Copper	0.03mm	D_GND
8		Dielectric 2	FR-4 TG140	0.53mm	
9		Layer 3	Copper	0.03mm	Signal
10		Dielectric 3	FR-4 TG140	0.1056mm	
11		Layer 4	Copper	0.03mm	5V_SYS
12		Dielectric 4	FR-4 TG140	0.53mm	
13		Layer 5	Copper	0.03mm	A_GND
14		Dielectric 6	FR-4 TG140	0.0913mm	
15		Bottom	Copper	0.018mm	Signal
16		Bottom ENIG	Gold	0.05µm	
17		Bottom solder	Solder resist	0.01mm	
18		Bottom overlay			
19		Bottom paster	Sn63/Pb37	0.12mm	

operation of the whole system.

In order to reduce electromagnetic interference (EMI), the FM10 6-layer structure adopts the top layer, the third layer, and the bottom layer as signal layers, in which copper is paved in the area around signal routes and connects to the ground plane. The second and the fifth layer are the ground planes, and the fourth layer is the power plane. The solid ground plane will minimize inductance, and any signal plane close to the ground plane will have continuous return current.

Decoupling capacitors are placed near the power pins of each chip. When the chip's payload suddenly increases, the circuit tries to increase its current, however the inductance in the power supply line hinders the current rising and causes the power line voltage to sag. At this time the decoupling capacitors provide short bursts of current to maintain the voltage at a stable level.

### **3.2.1 Power management and distribution**

The FM10 can be powered by a DC power supply, USB, and Li-Ion or other batteries because of its high-performance power management module design which has a wide operating input voltage range from 3.3 V to 17 V. It allows FM10 to easily work in various application environments with flexible power requirements. At the same time, FM10 can provide 5 V and 3.3 V power sources, which allows small sensor modules that connect to FM10 to be powered, thus avoiding redundant power circuit design for external sensors.

The power management module of FM10 has two synchronous Step-Down DC to DC Converters from Texas Instruments; their part numbers are TPS62133 and TPS62132. Both of them adopt 16-pin 3x3 mm QFN packages and can provide 3 A of continuous output current at 3.3 V and 5 V respectively with a maximum up to 95% converting efficiency. These two chips are made for mobile application platforms;

They use advanced regulation topology (DCS-Control) that combines advantages of multiple control modes, which can monitor output voltage changes and feed it directly to a fast comparator to set the switching frequency. This technology provides immediate response for dynamic load changes to achieve fast and stable voltage operation. These DC to DC converters also prevent the circuit from shorting and overloading. When a load or a short circuit is detected and the output voltage drops below 0.5V, the current limit will be reduced to 1.6A immediately. If the output voltage picks up above 0.5V again, the device recovers to the normal operating mode. Figure 3.4 shows the schematic of the voltage converter connectivity in FM10. A 1.0uH inductor and a 22.0uF capacitor are connected to its output as an external LC output filter.

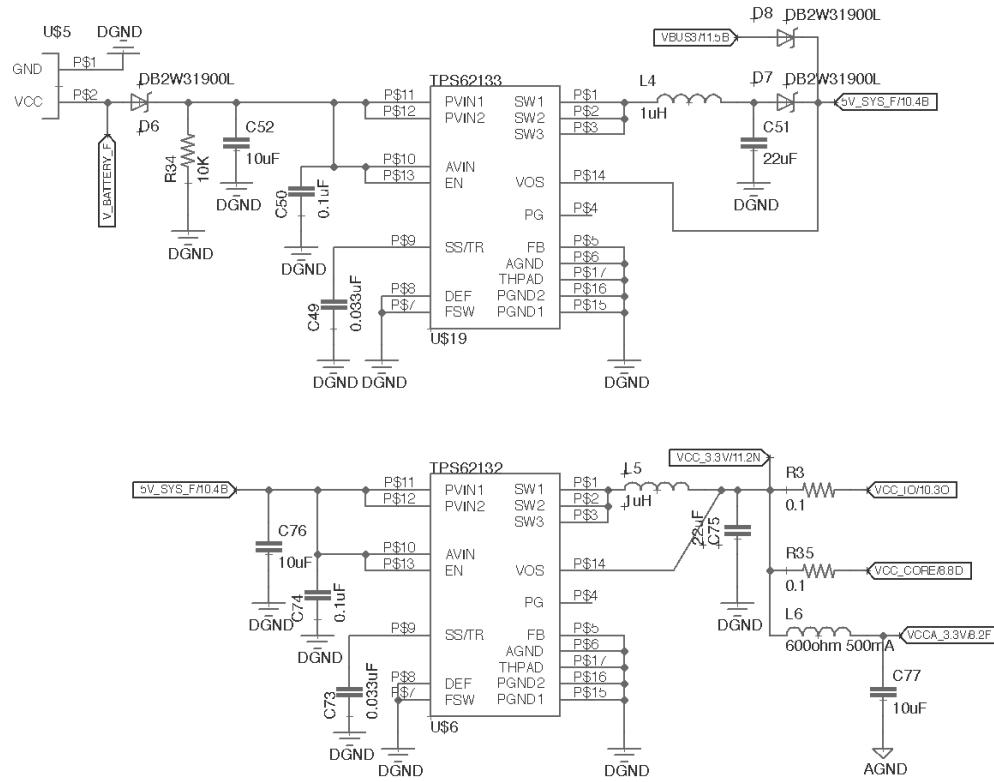


Figure 3.4: FM10 module power schematic

The chip, TPS62133, accepts a maximum of 17.0 V input power, and its output voltage is fixed 5.0 V, which is the input voltage for the next level DC-DC converter

chip TPS62132. A DB2W900L diode from Panasonic Electronic Components is placed between the two chips to prevent reverse current. Moreover, the USB connector also has a 5.0V power line connecting to the input of TPS62132 via a DB2W900L diode, and this allows the system to also be powered by a USB connector directly. TPS62132 provides 3.3 V output voltage to the FPGA and other chips whose rated voltages are 3.3 V. The two diodes (D8, D7) feature 30.0 V reverse breakdown voltage which prevents the USB from being burnt by a high reverse voltage, and allows FM10 to be powered by battery and USB at the same time.

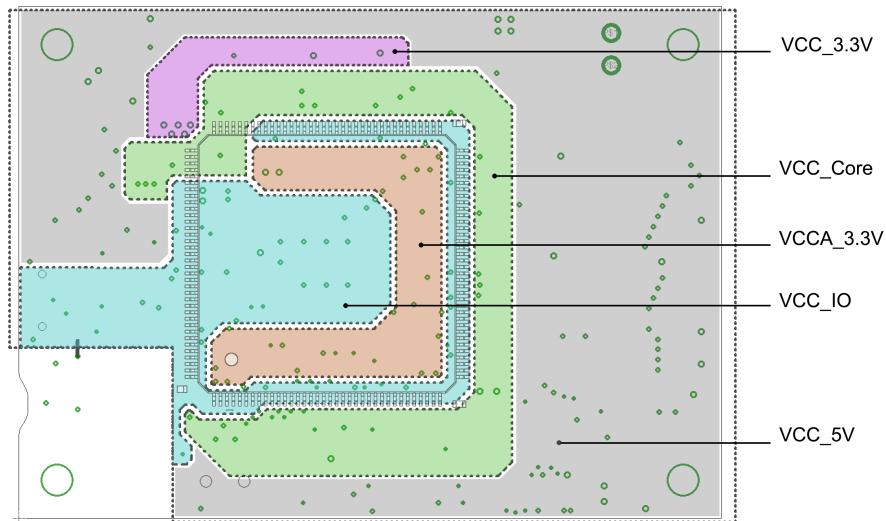


Figure 3.5: Power layer design of FM10’s PCB

In order to provide the FPGA chip with a clean, low-noise power supply, a power tree was designed to separate the 3.3V external power supply into different groups serving as multiple inputs to the MAX 10 device via VCC\_ONE, VCC\_IO and VCCA. Figure 3.5 shows the power layer design in the PCB. The VCC\_3.3V power supply is divided into VCC\_Core, VCC\_3.3V and VCC\_IO to power different portions varying in function on the FPGA chip. This reduces the interference between different functional groups which can arise due to voltage fluctuations caused by the

different current demands of each functional group.

### **3.2.2 FPGA: 10M25SAE144I7G**

The chip, 10M255AE144I7G, belongs to Intel® MAX10 series; it is single-supply FPGA, which requires only 1 external power supply of 3.3 V [14]. It is built on 55nm embedded NOR flash technology, providing 25000 logic elements, integrated Analog-to-Digital Converters (ADC) and 2 MB of user flash memory to store configuration files and user data. The ADC provides 8 channels with 12-bit digital representation of the analog signals being observed, with a cumulative sampling rate of up to 1million samples per second (MSPS). The user flash memory provides ideal storage space to store non-volatile information. It uses an EQFP package, offering 144 pins, in which 101 pins have general-purpose input/output (GPIO) functionality. In addition, the speed grade -7I is another important parameter in design. There is no consistent definition of a speed grade for all FPGA/CPLD devices. The -7I for this chip means it is industrial class; the pin-to-pin delay is 7 nanoseconds. In other words, the smaller the number, the faster the speed, which has an impact on the selection of SRAM.

The maximum power consumption of the 10M25S chip can be 2.943 W, which comes from the VCC\_ONE pins. All VCC\_ONE pins are powered by VCC\_Core. If the user's design is beyond the maximum power consumption, it will cause that part of the function to perform abnormally. The minimum system design includes the power supply, crystal oscillator, voltage reference and adapter interface. The type of crystal oscillator is CB3LV-3C-50M0000 from CTS Electronics Components providing a 50 MHz clock signal. The voltage reference chip, MCP1525 from Microchip, provides a 2.5 V precision voltage reference that has initial maximum tolerance of  $\pm 1\%$ , and temperature stability of  $\pm 50 \text{ ppm}/\text{c}^\circ$ .

### 3.2.3 SRAM: IS61WV25616EDBLL

Static Random Access Memory (SRAM) is a kind of random access memory. The 'static' means the memory can store the data permanently as long as it is powered. Data does not need to be refreshed periodically and would not be lost. By contrast, the data in SDRAM needs to be refreshed periodically to ensure the effectiveness of the data. Both of them are called volatile memory since the stored data would be lost if the power is cut off. Compared to SDRAM, the SRAM has advantages of faster speed, lower power consumption, especially during idle, and easier control. This is also the reason that SRAM is selected for FM10. The disadvantages of SRAM are more complicated internal circuit and high production cost, therefore, it is not suitable to be used in the applications requiring large storage.

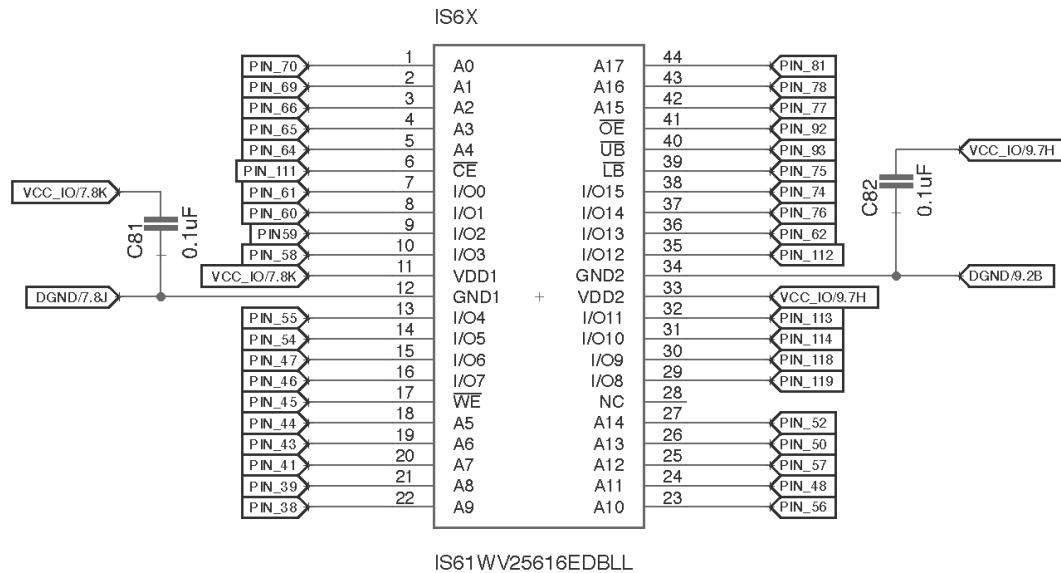


Figure 3.6: The SRAM schematic of FM10 module

The ISSI, IS61WV25616EDBLL, on FM10 is a high-speed asynchronous CMOS SRAM chip, which has 4194304 bits of memory organized as 262144 words by 16 bits. This chip can satisfy most of the application requirements that need high computing speed. The asynchronous means that the read and write operation is triggered by

rising or falling edge of the control signal or the address signal. The read and write can occur at any time. However, synchronous SRAM has one clock signal, thus all the operations are driven by the clock signal. In other words, the read and write operation only happens at expected times. The access time of IS61Wv25616EDBLL is  $8 \sim 10$  ns, which is faster than the FPGA's clock signal. In another words, the SRAM can be accessed and operated by the FPGA on every clock cycle meaning there are no waiting clock cycles. Figure 3.6 shows how the SRAM is powered in FM10 and its pin connection to FPGA.

### 3.2.4 CAN Transceiver

A CAN bus standing for Controller Area Network bus is a robust data communication protocol that was initially used on vehicles enabling devices to communicate with each other, but now it is not limited to use in the automobile field only. CAN's high performance and reliability have been recognized, and are widely used in industrial automation, ships, medical equipment, and so on. The CAN transceiver makes the FM10 interfaces more friendly to access CAN network, and this greatly increases the applications of FM10. The CAN bus provides powerful support for distributed control platform which is using ioNeurons<sup>TM</sup> processing modules, to realize real-time and reliable data communication between various nodes.

Unlike other data transfer protocols, a pair of CAN bus twisted wires allows more than two devices to connect in parallel and exchange data; in other words, it is not a fixed point-to-point communication protocol; it can create a communication network and offer flexible point-to-point or point-to-multipoint communications. Those devices connecting to a CAN bus are Electronic Control Units (ECUs) also known as nodes. A node can be a simple I/O device or a complicated embedded computer. Any of the devices can be a host as long as it has the corresponding functionality. Once a device

broadcasts a message on the bus, every other device will listen and identify if they are interested in this message, and then the message will be received by the one that is interested.

To transfer data, CAN bus signalling uses differential signals on the two wires: one is called CAN\_high, the other one is called CAN\_low. When CAN\_high is driven to 5 V and CAN\_low is driven to 0V, the transmission is recognized as dominant state representing logic level 0. When neither of them is driven, they are levelled to the reference voltage which is 2.5 V, and at this time, the transmission is recognized as recessive state representing logic level 1. In the physical layer, these two wires are twisted together in order to reduce the influence of electromagnetic interference. Two  $120 \Omega$  termination resistors are placed on each end of the CAN bus to suppress signal reflections.

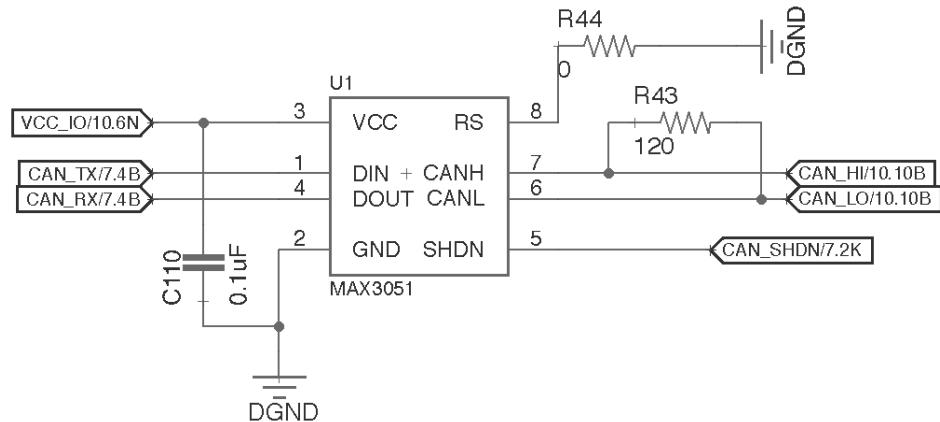


Figure 3.7: The CAN transistor schematic of FM10 module

In order to generate the differential signals, a transceiver chip that is based on CAN bus protocol is required. In FM10, the transceiver chip is MAX3051 from Maxim Integrated. This chip can translate the logic level signal 0 or 1 from the host controller to differential signals that are transmitted on the CAN bus. It also features four different modes of operation that enables data transfer speed control.

The schematic in Figure 3.7 shows a typical operating circuit.

### 3.2.5 FTDI

As the rapid development of laptops continues toward lighter and thinner, many traditional connectors are replaced by those having lower volume and faster transfer speeds. Universal Serial Bus (USB) facilitates the communication between computer and other peripherals to a large extent. USB is standardized as an industrial standard that prescribes the communication protocol, cable, and connector type. It substitutes the traditional serial communication ports and the connectors that were previously used such as the D-subminiature connector. However, in the development of FPGA and embedded systems, traditional serial communication protocols take less resources, and are relatively easy to implement, so they still play an important role in data transfer and debugging of embedded developments. Universal asynchronous receiver-transmitter (UART) is a very common communication protocol. In order to bridge the connection between USB on the computer and serial output on the FPGA via the UART communication protocol, a UART to USB converter chip from FTDI is needed. A corresponding driver must be installed in the computer operating system, to allow the USB port to be recognized as a virtual serial communication port.

FT234XD USB to basic UART IC provides USB and an asynchronous serial data transfer interface, which is fully compliant with the USB 2.0 specification. Its UART interface has support for 7 and 8 data bits with 1 or 2 stop bits. The data transfer rate can be configured from 300 baud to 3M baud rate. Figure 3.8 illustrates the FT234XD typical configuration with a USB bus powered design. The UART\_TXD and UART\_RXD are connected to PIN\_124 and PIN\_120 on the FPGA through  $0\ \Omega$  resistors or a suitable ferrite bead according to the current; These can reduce the EMI noise that the FT234XD radiates to the USB host over the USB cable.

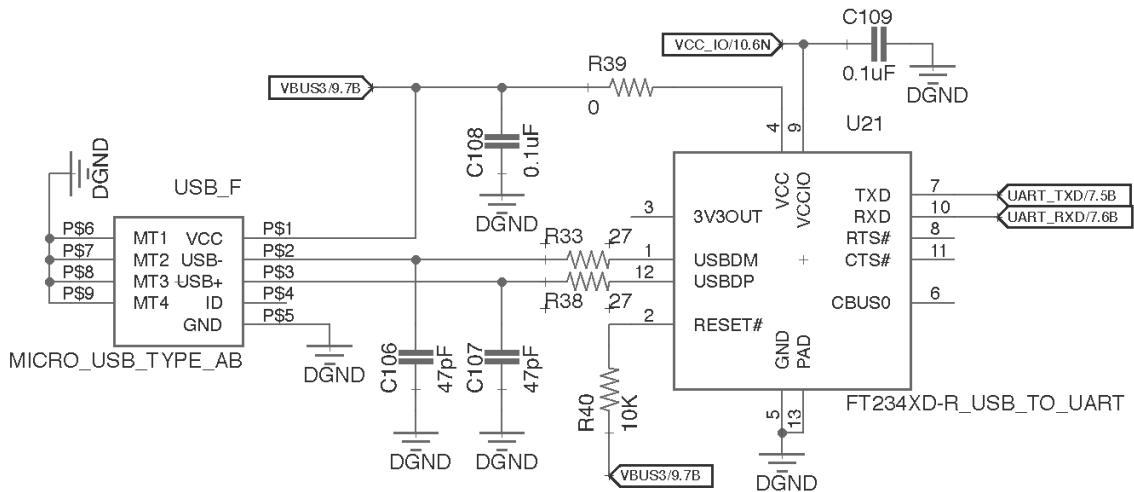


Figure 3.8: The FTDI schematic of FM10 module

### 3.3 Interfaces and pin mapping

As a processing module, the selection of suitable connectors for FM10 directly impacts the user experience and the way its used. In order to facilitate adoption of FM10 in as many projects as possible and considering the cost-benefit factor, FM10 is designed to use two different types of connector: Flat Flex Cable (FFC) connectors and P5KS series board to board narrow pitch connectors; Both of them are classified as surface mount type and high-speed connectors. The primary function of FFC is to provide and expand the connectivity to other peripherals. P5KS connectors are utilized for implementing expandable and stackable features between ioNeurons<sup>TM</sup> modules.

The FFC connector mounted on FM10 module is 24-pin with a 0.4mm pitch, fitting a 12mm wide flat flex cable, which can connect FM10 with another module. In the 24-pin FFC connector, 20 out of 24 pins are defined as data transfer pins, and other 4 pins are set as power pins: 2 for VCC and 2 for GND. This reinforced dual power pin design effectively avoids the possibility of burning the connector due to overloading current on a single pin, thereby increasing the capacity to take and supply large

current flow. In addition, the FFC connector has a double-bottom-contact terminal design which can remove dust and contaminants better than single-bottom-contact terminals. Two contact points on each pin also provides stable redundant contact forces for secure electrical reliability. The FFC is placed at upper side of the FPGA chip, and it is very close to FPGA Pin\_73 ~ Pin\_108. Most of these pins are able to carry high-speed differential signals. Not only does this layout design shorten the trace length between FPGA and FCC connector, but it also reduces the EMI noise. FM10 chooses to use the parallel-to-board type FFC connector to achieve more space between two adjacent boards thus making it easier for users to install cables.

FM10 uses Panasonic P5KS series narrow pitch connectors for communicating between ioNeurons<sup>TM</sup> processing modules. A pair of 30-pin connectors are soldered on the top and bottom side of FM10. The header of P5KS connector is on the top, while the socket of the connector is on the bottom. The socket and header are protected from reverse mating by two chamfered corners.

Each connector has a total of 30 pins, which are distributed in two rows with a pin-to-pin pitch of 0.5 mm. It was chosen to be used on FM10 due to the diverse mating height choices for this type of connector. In the design, an 8.0 mm mating height is used to avoid collision between the Intel<sup>®</sup> Edison module and FM10 when the modules are assembled. Both the header and socket have 4 pins that were defined as power pins: 2 for VCC and 2 for GND; each pin can take up to 0.5A current. This reinforced design effectively avoids the possibility of burning the connector due to large current flow, thereby increasing the capacity to take and supply large current flow. A DB2W900L diode from Panasonic Electronic Components is placed between the power source and power pins to prevent current from going backward. In other words, users cannot power FM10 via power pins from the P5KS connector, but FM10 can provide a 5.0 V power output. Table 3.3 and Table 3.4 show the connection

Table 3.2: FM10 24-pin FFC pin-out and signal list

<b>FFC Pin Number</b>	<b>FFC Pin Name</b>	<b>FPGA Pin Name</b>	<b>Function</b>
1	FFC_0	PIN_79	GPIO
2	FFC_1	PIN_80	GPIO
3	FFC_2	PIN_84	GPIO
4	FFC_3	PIN_85	GPIO
5	FFC_4	PIN_86	GPIO
6	FFC_5	PIN_87	GPIO
7	FFC_6	PIN_88	GPIO
8	FFC_7	PIN_89	GPIO
9	FFC_8	PIN_90	GPIO
10	FFC_9	PIN_91	GPIO
11	FFC_VCC	N/A	VCC 3.3V
12	FFC_VCC	N/A	VCC 3.3V
13	FFC_GND	N/A	GND
14	FFC_GND	N/A	GND
15	FFC_10	PIN_96	GPIO
16	FFC_11	PIN_97	GPIO
17	FFC_12	PIN_98	GPIO
18	FFC_13	PIN_99	GPIO
19	FFC_14	PIN_100	GPIO
20	FFC_15	PIN_101	GPIO
21	FFC_16	PIN_102	GPIO
22	FFC_17	PIN_105	GPIO
23	FFC_18	PIN_106	GPIO
24	FFC_19	PIN_110	GPIO

between the 30-pin connectors and the FPGA with pin's functionalities.

In this thesis, it is defined that any pins on the top header and the bottom socket that have the corresponding position to each other and are connected through PCB via, are Via Pins. A pair of Via Pins carry the same signal. The FM10's P5KS connectors contain 16 pairs of Via Pins: some of them are connected to the FPGA's specific pins to build up the I2C or CAN bus communication circuit; while some of them are just pierced forming a bridge without connecting to FPGA. Via Pins can solve issues such as device expansion, module perpendicular connectivity and cross-module communications.

Table 3.3: FM10 30-pin P5KS Header pin-out and signal list

P5K Pin Number	P5K Pin Name	FPGA Pin Name	Via Pin	Function
1	HS_1	N/A	Yes	VCC 5V
2	HS_2	N/A	Yes	CAN_LO
3	HS_3	N/A	Yes	VCC 5V
4	HS_4	N/A	Yes	CAN_HI
5	HS_5	N/A	Yes	GND
6	HS_6	PIN_87	Yes	RESET_N
7	HS_7	N/A	Yes	GND
8	HS_8	PIN_122	Yes	DEV_OE
9	HS_9	PIN_135	Yes	GPIO
10	reserved	N/A	N/A	N/A
11	HS_11	PIN_140	Yes	GPIO
12	reserved	N/A	N/A	N/A
13	HS_13	N/A	Yes	N/A
14	H_14	PIN_130	No	GPIO
15	HS_15	N/A	Yes	N/A
16	H_16	PIN_134	No	GPIO
17	HS_17	PIN_25	Yes	GPIO
18	H_18	PIN_26	No	GPIO
19	HS_19	PIN_24	Yes	GPIO
20	H_20	PIN_22	No	GPIO
21	reserved	N/A	N/A	N/A
22	reserved	N/A	N/A	N/A
23	HS_23	N/A	Yes	N/A
24	H_24	PIN_33	No	GPIO
25	HS_25	N/A	Yes	N/A
26	H_26	PIN_32	No	GPIO
27	H_27	PIN_21	No	GPIO
28	H_28	PIN_30	No	GPIO
29	H_29	PIN_17	No	GPIO
30	H_30	PIN_29	No	GPIO

Table 3.4: FM10 30-pin P5KS Socket pin-out and signal list

P5K Pin Number	P5K Pin Name	FPGA Pin Name	Via Pin	Function
1	HS_1	N/A	Yes	VCC 5V
2	HS_2	N/A	Yes	CAN_LO
3	HS_3	N/A	Yes	VCC 5V
4	HS_4	N/A	Yes	CAN_HI
5	HS_5	N/A	Yes	GND
6	HS_6	PIN_87	Yes	RESET_N
7	HS_7	N/A	Yes	GND
8	HS_8	PIN_122	Yes	DEV_OE
9	HS_9	PIN_135	Yes	GPIO
10	reserved	N/A	N/A	N/A
11	HS_11	PIN_140	Yes	GPIO
12	reserved	N/A	N/A	N/A
13	HS_13	N/A	Yes	N/A
14	S_14	PIN_123	No	GPIO
15	HS_15	N/A	Yes	N/A
16	S_16	PIN_141	No	GPIO
17	HS_17	PIN_25	Yes	GPIO
18	S_18	PIN_8	No	GPIO
19	HS_19	PIN_24	Yes	GPIO
20	S_20	PIN_10	No	GPIO
21	reserved	N/A	N/A	N/A
22	reserved	N/A	N/A	N/A
23	HS_23	N/A	Yes	N/A
24	S_24	PIN_14	No	GPIO
25	HS_25	N/A	Yes	N/A
26	S_26	PIN_11	No	GPIO
27	S_27	PIN_6	No	GPIO
28	S_28	PIN_13	No	GPIO
29	S_29	PIN_7	No	GPIO
30	S_30	PIN_12	No	GPIO

## 3.4 FM10 Firmware

This chapter introduces how the FPGA controls and cooperates with the other on-board chips. The SRAM controller and UART transceiver were implemented by the FPGA in Very High-Speed Integrated Circuit Hardware Description Language (VHDL).

### 3.4.1 SRAM Controller

The SRAM controller can facilitate user access to the data stored in SRAM, control the mode of SRAM read and SRAM write, and change the operating status of the SRAM. In this thesis, the SRAM controller is designed to accomplish the write and read operation of random addresses for 8-bit data through controlling a IS61WV16EDBLL SRAM chip.

In typical FPGA image processing applications, SRAM is used to store image pixel data whose value has a range of  $0 \sim 255$ , each pixel data only needs 8-bits of storage space. ISSI's IS61WV6EDBLL SRAM has 262144 addresses, and each of them can store a 16-bit value. If each 16-bit location is used to store 8-bit data, the maximum occupancy ratio would be 50% meaning a large waste of SRAM storage space. To improve the efficiency of occupancy, the SRAM controller implemented in this thesis can access the upper byte and lower byte of a 16-bit data field separately via configuring the signals on the SRAM's Upper Byte (UB) pin and Lower Byte(LB) pin. When the UB pin is asserted HIGH, and the LB pin is asserted LOW, the write and read operation is only valid for the LB of the data. When the UB pin is asserted LOW, and the LB pin is asserted HIGH, the write and read operation is only valid for the UB of the data. This method gives users 524288 addresses, which is double the number of 16-bit data addresses, with individual 8-bit data storage space. Figure

3.9 illustrates the SRAM controller block diagram with its interfaces.

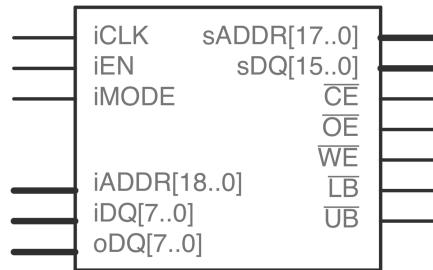


Figure 3.9: SRAM Controller block diagram

The data interface of the SRAM controller is 8-bit wide, and the Address Input of the SRAM controller is 19-bits wide; it consists of the IS61WV6EDBLL SRAM's 18-bit Address Input and 1-bit Byte Control. The Address Inputs of the SRAM from A17 to A0 correspond to the Address Input of the SRAM Controller from A18 to A1. The least significant bit (A0) of the SRAM Controller Address Input is a 1-bit Byte Control which controls the signals on the UB pin and the LB pin. This 1-bit Byte Control allows the separation of a 16-bit storage location into two 8-bit storage locations assigned with individual addresses. Table 3.5 shows the truth table of the SRAM Controller Address Input A0 and UB, LB signals.

Table 3.5: Truth table of the SRAM controller address input A0 and UB, LB signals

<b>Mode</b>	<b>A0</b>	<b>iMode</b>	<b>iEN</b>	<b>oDQ0 - oDQ7</b>	<b>iDQ0 - iDQ7</b>
SRAM disabled	X	X	L	High-Z	High-Z
Read	L	H	H	Dout0 - Dout7	High-Z
Read	H	H	H	Dout8 - Dout15	High-Z
Write	L	L	H	High-Z	Din0 - Din7
Write	H	L	H	High-Z	Din8 - Din15

The SRAM controller has a clock input accepting not greater than a 100MHz clock signal, which is for synchronization of the control signal and data signal; in this design, all signals are triggered by the rising edge of the clock signal except iEN. When the input signal iEN is asserted LOW, the SRAM controller sets the SRAM Chip Enable (CE) pin to HIGH immediately, and the SRAM enters standby mode with minimum power consumption; meanwhile, the data terminals become high impedance. When iEN is asserted HIGH, the controller sets the  $\overline{\text{CE}}$  pin to LOW, and the SRAM chip is enabled to use. The input signal, iMode, is used to determine the write or read action: when iMode is asserted HIGH, the controller reads data from the SRAM; when iMode is asserted LOW, the controller writes data to the SRAM.

In the physical connections, the SRAM controller's sADDR17 to sADDR0 pins are connected to pin A17 to A0 of the SRAM sequentially. The SRAM controller's sDQ15 to sDQ0 pins are connected to pin D15 to D0 of the SRAM. Other control signals, such as CE, OE, WE, LB, UB, are also connected to the corresponding pins on the SRAM.

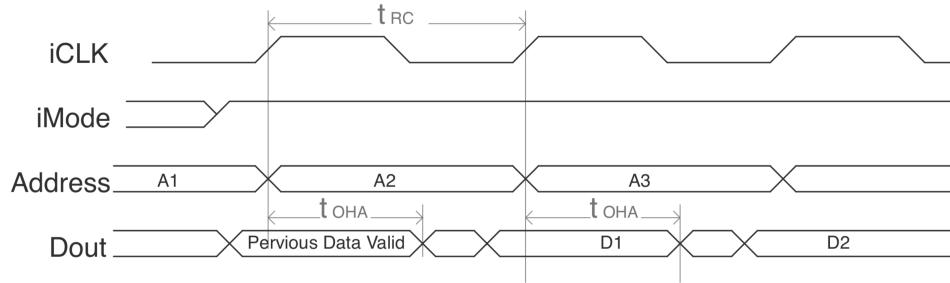


Figure 3.10: SRAM Read switch

Read access is operated when iMode is set to be HIGH. The SRAM controller maps data from the SRAM's Dout0 ~ Dout7 to oDQ0 ~ oDQ7 when the least significant bit of the controller's Address Input (A0) is 0. Conversely, when A0 is 1, the SRAM controller maps data from SRAM's Dout8 ~ Dout15 to oDQ0 ~ oDQ7. At this time,

the input interfaces  $iDQ0 \sim iDQ7$  are high impedance. The minimum value of the Read Cycle Time ( $t_{RC}$ ) is 10ns. The minimum value of Output Hold Time ( $t_{OHA}$ ) is 2ns; These values are part of the SRAM switching characteristics, which may be different depending on the SRAM chip used.

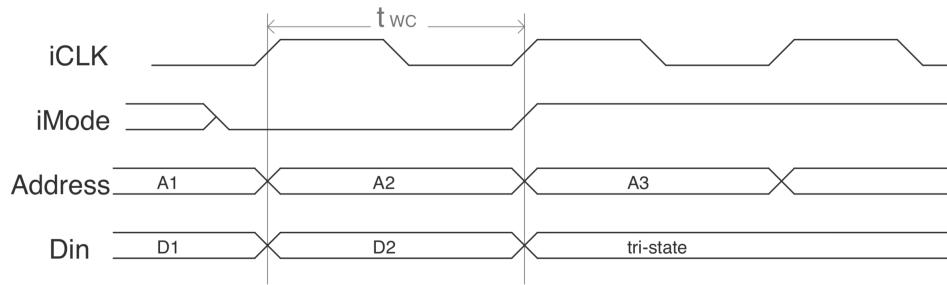


Figure 3.11: SRAM Write switch

Write access is operated when  $iMode$  is set to be LOW. The SRAM controller maps data from  $iDQ0 \sim iDQ7$  to SRAM's  $Din0 \sim Din7$  when the least significant bit of the controller's Address Input ( $A_0$ ) is 0. Conversely, when  $A_0$  is 1, the SRAM controller maps data from  $iDQ0 \sim iDQ7$  to SRAM's  $Din8 \sim Din15$ . At this time, the output interfaces  $oDQ0 \sim oDQ7$  are high impedance. Table xx shows the control signal and data signal in the different operating modes. The minimum value of the Write Cycle Time ( $t_{RC}$ ) is 10ns which is part of the SRAM switching characteristics, and may be different depending on the SRAM chip used.

### 3.4.2 UART

Serial Communication is widely used in embedded system development, and one of the most commonly used protocols is Universal Asynchronous Receiver-Transmitter (UART). It has no synchronized clock signal during data transfer which makes it an asynchronous protocol. Asynchronous transmission allows data to be transmitted with timing parameters agreed by both the sender and receiver. There are three wires

required to transfer data: one wire for data transmitting, one wire for data receiving, and the last one for ground reference. When transmitting data, the transmitter converts incoming parallel data to serial data and sends byte data on the communication line bit by bit. When receiving data, the receiver obtains data from the communication line bit by bit, and temporarily stores them in the memory until a whole byte data is collected, then it sends the byte length data to the processor in parallel. Using an FPGA to implement UART avoids using extra chips, thus saving space and enhancing system reliability.

Because UART is an asynchronous transmission protocol, keeping the same transmission speed between transmitter and receiver is a prerequisite for ensuring correct data transmission and reception. The baud rate is used to specify the transmission speed of both parties. Baud rate is the unit for symbol rate represented by bit per second in the UART. Data is transmitted in bytes; each byte contains 8 bits. It takes bytes of data and sequentially sends the individual bits. There is a start bit in front of the 8-bit data, and a stop bit at the end of the 8-bit data. Figure 3.12 illustrates the single byte data formatting.



Figure 3.12: UART Frame

The FPGA UART design is modularized; according to the functions, the whole system is separated into baud rate generator, transmitter, and receiver. Figure 3.13 shows its block diagram.

When the communication line is idle, the line stays in a high-voltage state, which is considered as logic HIGH. Once the transmitter starts to send data, it will generate a start bit signal, which is represented by a low-voltage (logic LOW), on the communication line before each data byte. The baud rate determines the time required to

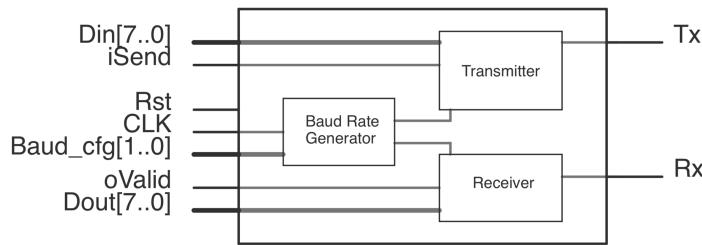


Figure 3.13: UART block diagram

transfer each bit. If the baud rate is 1, the start bit logic LOW persists for 1 second and then the logic level will change according to the value of bit 0. If the value of bit 0 is 1, the communication line becomes high-voltage. If the value of bit 0 is 0, the communication line becomes low-voltage and remains for 1 second. The transmission follows the same rules, until the last bit of the data byte is transferred, then a stop bit is generated, which is a high-voltage (logic HIGH) on the communication line. The stop bit means the receiver is informed that this is the end of the current byte transfer. After the stop bit, the communication line returns to the idle state. The stop bit can be 1-bit, 2-bit or 3-bit based on the requirement of the application environment. When the stop bit is 1-bit, a one-byte data frame consists of 10 bits; it needs 10 seconds to complete the transfer if the baud rate is 1. It requires 86.8  $\mu$ s to complete the transfer if the baud rate is 115200.

When receiving data, the receiver samples the incoming data on the communication line. If the initial state is assumed to be idle on the communication line, once the receiver detects a logic LOW signal, it changes to receiving mode immediately. It samples the incoming data based on the pre-set baud rate and stores each bit sequentially. After collecting all the bits of the data ending up with a stop bit, the data transfer is completed. The receiver continues monitoring for a change of logic level on the communication line.

### 3.5 Configuring FM10 with USB-Blaster

Intel MAX10 series FPGAs can be configured using a JTAG connector. Intel provides users with a USB-Blaster that can send configuration data from the host computer to the FPGA. It has a USB connector and a standard 10-pin female connector, in which the USB connector connects to a host computer and the other end connects to the FPGA's JTAG interface located on the PCB. Developers can use Intel Quartus Prime software to compile the VHDL code, and then the compiled file can be uploaded to FPGA through a USB-Blaster to finish the FPGA configuration process.

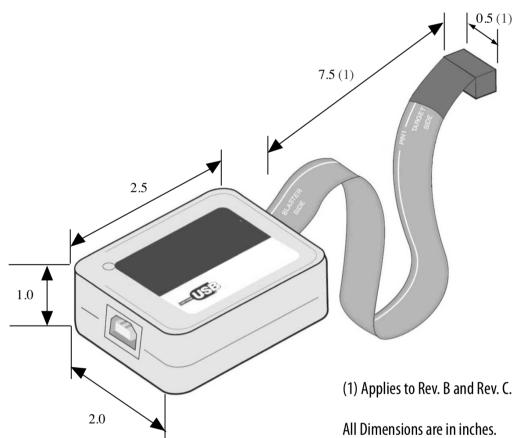
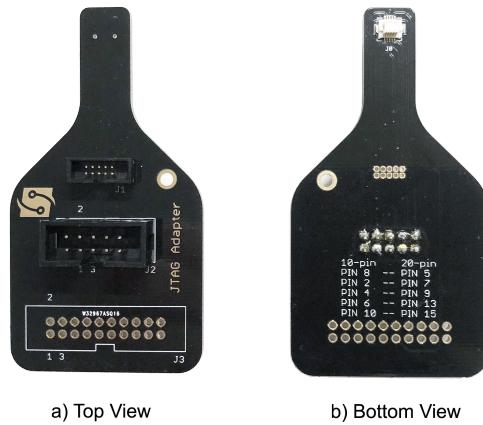


Figure 3.14: USB Blaster

However, the size of the 10-pin female connector on the USB Blaster is relatively big; it is 12.7 mm in length, which is against the compact design intention of FM10, meaning there is not enough space for the mating connector. FM10 substitutes it with a DF12(3.0)-10DP-0.5V(86) female connector from Hirose Electric Co Ltd, which has the same number of pins but is smaller in size (5.5mm x 3.8mm). In order to establish the connection between the 10-pin connection header on the USB-Blaster and the Hirose connector on FM10, a JTAG adapter was designed. Figure 3.15 shows the JTAG Adapter.



a) Top View                          b) Bottom View

Figure 3.15: JTAG Adapter

The JTAG Adapter has a DF12 (3.0)-10DS-0.5V(86) male connector that is the mating connector to the female connector on FM10. Meanwhile, it also has a standard 10-pin male connector that the USB-Blaster can connect to. The Hirose male connector and the standard 10-pin header connector are interconnected correspondingly via traces in the PCB. The JTAG Adapter acts as a bridge to connect the USB-Blaster and the JTAG of FM10. Figure 3.16 shows the signals on the USB-Blaster 10-pin connector and the design of the FM10 JTAG.

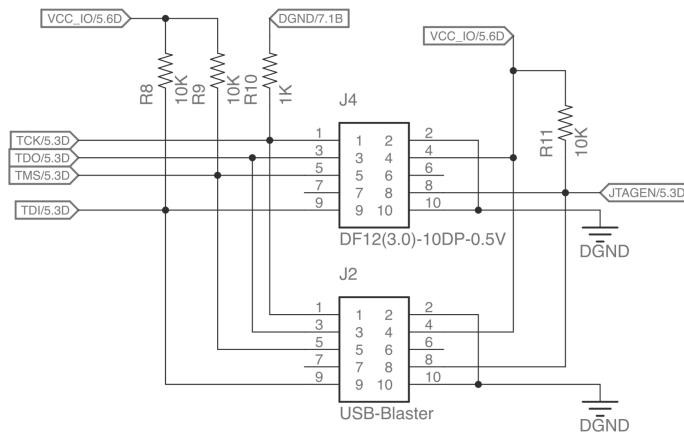


Figure 3.16: The JTAG adapter schematic