Summary

Pixie provides a complete ZigBee solution for OEMs, integrating IEEE 802.15.4 2.4GHz PHY / MAC layers with Microchip Technology’s NWK / APL / ZDO layers. It provides a path for extremely rapid migration from Microchip’s PICDEM Z ZigBee development environment to market-ready product.

Pixie is capable of full function device (FFD) operation and is suitable for router and coordinator nodes. The lower cost Pixie Lite version is used for reduced function devices, e.g. endpoints, with 2 fewer I/O lines and being 3mm more compact.

- Free space range approx 120m
- FCC / CE / IC compliant
- MAC address allocation
- Signature G antenna, low “hand effect” design
- Output power 1mW / 0dBm
- Consumes less than 25mA when active, 2μA in sleep mode
- 15 general I/O lines, featuring 6-channel 10-bit A/D, serial UART, 4 interrupts, counter input
- Master SPI
- Surface mount and DIL through-hole versions
- Wide temperature range -40ºC to +85ºC
- 2.1V - 3.6V supply, ideal for Li rechargeables

Firmware Features

Pixie is electrically virtually identical to the PICDEM Z ZigBee development environment and is fully compatible with Microchip Technology’s free ZigBee stack, featuring:

- Compatibility with ZigBee 1.0 specification
- FlexiPanel Ltd profiles in development for switching, serial data communications, UI servers, displays and sensors
- Only configuration bits and I/O pin definitions are changed during migration from PICDEM Z

Ordering Information

<table>
<thead>
<tr>
<th>Part No</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PIXIE-SO-XXX</td>
<td>Pixie 26-contact surface mount module</td>
</tr>
<tr>
<td>PIXIE-DIL-XXX</td>
<td>Pixie 26-pin 0.1” Dual-in-Line module</td>
</tr>
<tr>
<td>PIXILITE-SO-XXX</td>
<td>Pixie Lite 24-contact surface mount module</td>
</tr>
<tr>
<td>PIXILITE-DIL-XXX</td>
<td>Pixie Lite 24-pin 0.1” Dual-in-Line module</td>
</tr>
</tbody>
</table>

XXXX refer to preloaded firmware options, refer to firmware data sheets

Refer to separate data sheets for all-weather enclosures
Contents

Summary ................................................................. 1
Electronic .................................................................. 2
Firmware ................................................................. 4
Regulatory ............................................................... 7
Radio Frequency ....................................................... 8
Mechanical ............................................................. 8
Pixie Lite .................................................................... 9
Future releases of Pixie & Pixie Lite .......................... 9
Bibliography ............................................................. 9
Drawing ..................................................................... 11
Appendix I .................................................................. 12
Reference ............................................................... 13

Electronic

Pin Connections

<table>
<thead>
<tr>
<th>Pin Number</th>
<th>Pin Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1,2,3,20</td>
<td>Gnd</td>
<td>Power supply ground reference and ground plane connection (note 3)</td>
</tr>
<tr>
<td>4</td>
<td>SCK</td>
<td>Serial Peripheral Interface (SPI) clock output (note 1)</td>
</tr>
<tr>
<td>5</td>
<td>SDI</td>
<td>Serial Peripheral Interface (SPI) data input (note 1)</td>
</tr>
<tr>
<td>6</td>
<td>RA0</td>
<td>PIC digital I/O RA0 / analog input AN0</td>
</tr>
<tr>
<td>7</td>
<td>RB4</td>
<td>PIC digital I/O RB4 / interrupt input KB10</td>
</tr>
<tr>
<td>8</td>
<td>RB5</td>
<td>PIC digital I/O RB5 / interrupt input KB11</td>
</tr>
<tr>
<td>9</td>
<td>RA4</td>
<td>PIC digital I/O RA4 / counter input T0CKI (note 3)</td>
</tr>
<tr>
<td>10</td>
<td>RA1</td>
<td>PIC digital I/O RA1 / analog input AN1</td>
</tr>
<tr>
<td>11</td>
<td>RA2</td>
<td>PIC digital I/O RA2 / analog input AN2 / negative voltage reference</td>
</tr>
<tr>
<td>12</td>
<td>RA3</td>
<td>PIC digital I/O RA3 / analog input AN3 / positive voltage reference</td>
</tr>
<tr>
<td>13</td>
<td>RE2</td>
<td>PIC digital I/O RE2 / analog input AN7</td>
</tr>
<tr>
<td>14</td>
<td>RE1</td>
<td>PIC digital I/O RE1 / analog input AN6</td>
</tr>
<tr>
<td>13,14</td>
<td>n.c.</td>
<td>No connection</td>
</tr>
<tr>
<td>15</td>
<td>Vunreg</td>
<td>Unregulated voltage input (note 2)</td>
</tr>
<tr>
<td>16</td>
<td>RB6</td>
<td>PIC digital I/O RB6 / interrupt input KB12 / ICD &amp; ICSP programming pin PGC</td>
</tr>
<tr>
<td>17</td>
<td>RE3</td>
<td>PIC digital input RE3 / reset input (active low) / ICSP programming pin Vpp</td>
</tr>
<tr>
<td>18</td>
<td>RB7</td>
<td>PIC digital I/O RB7 / interrupt input KB13 / ICD &amp; ICSP programming pin PGD</td>
</tr>
<tr>
<td>19</td>
<td>RC6</td>
<td>PIC digital I/O RC6 / serial UART TxD output</td>
</tr>
<tr>
<td>20</td>
<td>RE0</td>
<td>PIC digital I/O RE0 / analog input AN5</td>
</tr>
<tr>
<td>21</td>
<td>RC7</td>
<td>PIC digital I/O RC7 / serial UART Rx output</td>
</tr>
<tr>
<td>22</td>
<td>SDO</td>
<td>Serial Peripheral Interface (SPI) data output (note 1)</td>
</tr>
</tbody>
</table>
| 23         | Vdd      | Regulated power supply input
|            |          | Regulated power supply output (note 2) |
| 24,25,26   | Gnd      | Power supply ground reference and ground plane connection |

Table 1. Pin descriptions for Pixie & Pixie Lite

1. Operates in master SPI mode. Slave devices need a chip select line as the SPI is also used for internal communications.
2. Requires optional voltage regulator option to be fitted on onboard regulation to be functional.
3. This pin may no longer function in future devices and will become a Vddcore power supply pin. Refer to the Future releases of Pixie & Pixie Lite section of this data sheet and the documentation for 18F46J10 and 18F25J10 series devices from Microchip Technology (www.microchip.com).
Circuit Description

The schematic diagram in figure 3 details the electrical configuration of the Pixie and Pixie Lite modules. Items shown in blue are options and are not fitted as standard. For stock management reasons, units may occasionally be supplied with unrequested options fitted.

The EasyBee radio provides PHY and MAC layers conforming to the IEEE 802.15.4 standard. It is electrically identical to the EasyBee radio marketed by FlexiPanel Ltd and for which detailed documentation is separately available.

The EasyBee radio in turn matches both the PICDEM Z radio board and the reference circuit supplied by Chipcon, suppliers of the circuit's main component, a CC2420 MAC/PHY transceiver. The main difference is the smaller size.

The remainder of the circuit, centered on the PIC18LF4620, is intended to provide the higher stack layers. In the case of ZigBee, these would be the NWK, APS, APL and ZDO layers. In addition, it provides the application layer or, if the application resides off-chip, an interface to it.

The OEM may choose to develop firmware in-house, make use of free Microchip’s ZigBee Stack, and/or make use of the free firmware solutions provided by FlexiPanel Ltd as needed.

The circuit is electrically very similar to the PICDEM Z Microchip ZigBee Stack evaluation kit, with the following exceptions:
A 16MHz oscillator is normally fitted.

- No voltage regulator is normally fitted. If the PIC is clocked at 4MHz, DC power between 2.1V and 3.6V should be supplied to Vdd. If the PIC is clocked at 16MHz, Vdd should be between 2.7V and 3.6V.
- A few I/O lines are not available for external connection.
- Some peripheral components, such as the RS232 converter, LEDs and pushbuttons, have been omitted.
- Pixie considerably smaller and lighter.
- A unique MAC address may be obtained from FlexiPanel for use with the module.

When used with the ZigBee stack as supplied from Microchip Technology, the following resources are used by the stack for internal communications:

- RB0-RB3
- RC0-RC2
- SPI (RC3-RC5) operating in master mode, although this may be shared with external components as detailed in the SPI section below.
- Watchdog timer (software enabled)
- Timer 0
- CCP2
- High priority interrupts on CCP2 (FIFOP), INT2 (SFD)

**Hardware Options**

The following options may be requested for volume orders:

- **Crystal:** If requested, the crystal oscillator X1 and oscillator capacitors C3, C4 may be omitted.

Where fitted, X1 is a 16MHz crystal rather than the 4MHz crystal used in the PICDEM Z evaluation kit. This is in order to reduce the overall size of the module. To compensate for this difference, disable the oscillator PLL mode, which the 4MHz oscillator uses to mimic a 16MHz oscillator.

**Voltage regulator:** The power supply to Vdd should be 2.1V to 3.3V. If requested, a voltage regulator may be fitted to regulate this from a DC supply to Vunreg within the limits 2.4V to 20V, 100mA maximum current.

If the device being designed is to be battery powered, aim to use lithium batteries rather than alkaline since they generally have much better ageing characteristics and reduced voltage decline over their lifetime.

**SPI Port**

The internal circuitry of the module uses the PIC MSSI peripheral in SPI Master mode. With care, other SPI slaves may also be connected.

- Chip select lines must be provided to avoid collision with internal communications.
- Microchip Stack firmware should be carefully understood to ensure communication does not occur while internal chip select lines are activated. Specifically, no external slaves should be selected between calls to PHYSelect() and PHYDeselect().

**MRF24J40 migration**

As Microchip roll out their MRF24J40 IEEE 802.15.4 radio chip, their ZigBee stack support will be increasingly focused on it rather than the CC2420. We are working with Microchip to provide a seamless, pin compatible migration path for the Pixie range. Our intention is to develop MRF24J40-based Pixie devices as soon as possible. CC2420-based Pixies will continue to be manufactured.

**Firmware**

Any communications system based on the IEEE 802.15.4 standard could use the Pixie module. However, it is intended for use as a ZigBee transceiver using the Microchip ZigBee stack.

**ZigBee overview**

This section provides a brief overview of the ZigBee communications protocol. For further details, refer to the bibliography.

ZigBee’s primary aim is to make low data rate, battery power communications feasible. Primarily it does this by using mesh networks of adjacent devices rather than point-to-point communications over long distances. Additionally, devices are divided into those which are battery powered end devices which can enter sleep modes, and routers, which can pass multi-hop messages through the network and store messages for end devices while they sleep. One router, called the coordinator, has special roles relating to overall network management.
Each end device can have up to 240 endpoints, which are separate applications sharing the same radio. For example, a three-gang light switch would have two distinct endpoints sharing the same radio electronics and battery.

The coordinator also manages the **Binding Table**. This saves an endpoint such as a light switch from knowing which other endpoints (i.e. light) it is connected to. The coordinator, and possibly the routers, store this information for the end devices in a lookup table. Endpoints are associated by adding entries to this lookup table in a process known as **binding** (Figure 5).

FlexiPanel Ltd has a continuing program of firmware development for Pixie devices. In general, firmware is provided free for use with Pixie devices.

The ZigBee stack is loosely based on the Open Systems Interconnection model. Referring to figure 6, the basic tasks of each layer in the stack are as follows:

**Physical Layer (PHY):** The PHY layer consists of a half duplex IEEE 802.15.4 radio transceiver. Pixie uses the 16-channel 2.4GHz ISM band, which is the most widely accepted worldwide.

**Medium Access Control (MAC):** The MAC layer provides reliable, error-free communications between a node and its immediate neighbors.

**Network Layer (NWK):** The NWK layer provides the routing and multi-hop capability required to turn MAC-level communications into a star, tree or mesh network.

**Application Support Layer (APS):** The APS layer routes messages on the network to the different application endpoints on the node.

**Applications Objects:** An application object is the software at an endpoint which achieves what the device is designed to do.

**ZigBee Device Object (ZDO):** The ZDO is a special application object responsible for overall device management, security keys and security policies. Its ZigBee Device Profile (ZDP) is always end point zero, while other application end points are numbered 1 to 240.

**Security Services:** The security services provided for establishing and exchanging security keys to secure the communications. The MAC, NWK and APS layers use them for encrypting data prior to transmission and authenticating it when it is received.

**Firmware Development guide**

In most applications, nodes will be divided into full function devices (FFDs) implemented on Pixie modules and reduced function devices (RFDs) implemented on lower-cost Pixie Lite modules. Routers and coordinators need to be mains-powered FFDs, whereas
end devices are more likely to be RFDs and may be battery powered.

Microchip Technology Inc provides a ZigBee stack, complete with source code, for use with PIC microcontrollers, including Pixies. Please refer to www.microchip.com to obtain the stack and for technical support. Typical Microchip development components required include:
- MPLAB development environment
- IDC2 Programmer / Debugger
- C18 compiler
- Pixie Evaluation Boards

Refer to Microchip Technology’s documentation for full details of the capabilities of the 18LF4520 and 18LF2520 microprocessors.

The Pixie Evaluation Kit contains two Pixie Evaluation Boards and may be used for evaluation and development work. Circuit diagnosis is much aided through the use of a packet sniffer such as Pixie Sniffer. Sniffer software is freely available from FlexiPanel Ltd for use with MACdongle, FlexiPanel’s USB Sniffer device.

It is difficult to use breakpoints to debug ZigBee stack code, since they halt the processing of the stack. The UART provides an invaluable tool for debugging and diagnostics. The Pixie Evaluation Boards provide RS232 level converters so that diagnostic messages may be viewed in terminal emulators such as Microsoft Windows HyperTerminal.

The Pixie Configuration Tool may be used to connect the Pixie UART to a USB port. The Configuration Tool generates 5V TTL serial data, so the level converter shown in figure 7 should be used to interface to the Pixie. This is provided in the Pixie Evaluation Boards and may also be used to implement a very low cost configuration and diagnostics port in commercial products.

---

Table 3. Programming connections

<table>
<thead>
<tr>
<th>Pin</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RE3</td>
<td>Vpp programming voltage</td>
</tr>
<tr>
<td>RB6</td>
<td>PGC programming clock</td>
</tr>
<tr>
<td>RB7</td>
<td>PGD programming data</td>
</tr>
<tr>
<td>Gnd</td>
<td>Power ground reference</td>
</tr>
<tr>
<td>Vdd</td>
<td>Power supply reference</td>
</tr>
</tbody>
</table>

An RJ11 connector is provided on the Pixie Evaluation Boards for connection to the ICD2 programmer. The module should be independently powered at 3.3V during programming.

If pins RE3 / RB6 / RB7 are used for purposes other than programming / debugging, steps should be taken to isolate them from the rest of the circuit during programming/debugging. Refer to the ICSP application notes in the bibliography.

---

Table 3. Programming connections

<table>
<thead>
<tr>
<th>Pin</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RE3</td>
<td>Vpp programming voltage</td>
</tr>
<tr>
<td>RB6</td>
<td>PGC programming clock</td>
</tr>
<tr>
<td>RB7</td>
<td>PGD programming data</td>
</tr>
<tr>
<td>Gnd</td>
<td>Power ground reference</td>
</tr>
<tr>
<td>Vdd</td>
<td>Power supply reference</td>
</tr>
</tbody>
</table>

Configuration Bits & Compile Options

Consider the following issues when setting compiler options and configuration bits, etc:

- Pin RE3 may be configured to operate as a reset pin or as a digital input, according to the configuration bits. If configured as a reset pin, appropriate reset circuitry should be provided by the OEM.
- The oscillator operates at 16MHz; the Microchip demo code expects a 4MHz oscillator. To convert to 16MHz, simply change the oscillator configuration from “HS PLL” to “HS”. In some examples, this is achieved by changing the line:
  ```
  rom unsigned char config1H = 0b00000110;
  ```
to:
  ```
  rom unsigned char config1H = 0b00000010;
  ```
- Operation at 16MHz is only guaranteed down to 2.7V. For operation down to 2.1V, consider reducing the clock speed to 4MHz. This may be achieved by using the internal oscillator block at 4MHz. The CLOCK_FREQ definition in the ZigBee stack should then be adjusted to 4000000.

PIC Programming

The PIC microcontroller may be programmed and/or debugged with Microchip Technology’s ICD2 debugger or PM3 programmer using the connections shown in table 3.

When working from demo code provided by Microchip Technology, bear in mind the following:
Debug optimizations only should be reserved for files that require debugging.

- Oscillator configuration bits must be set according to the oscillator mode in use.
- Brownout reset is recommended at 2.0V for 4MHz devices and 2.7V for 16MHz devices.
- The watchdog timer should be enabled in software as required. For battery powered devices, the postscaler value should be set according to the desired wakeup rate (1:256 being approximately 1 second).
- CCP2 Mux should be set to RB3.
- PORTB<4:0> should be configured as digital I/O on reset.
- If MCLR is enabled, the OEM must ensure its correct operation. No on-board circuitry is provided.

**Regulatory**

**EMI ratings**

The Pixie and Pixie Lite modules integrate the EasyBee reference design which has been independently tested for FCC and CE compliance.

FCC compliance requires us to state the following:

1. This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions: (1) this device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation.

2. OEM products incorporating EasyBee must include an external label containing the following text legible to the naked eye:

   **Contains Transmitter Module
   FCC ID: UGAZBMR10**

3. Changes or modifications to the module may void its certification.

**ZigBee Alliance Membership**

FlexiPanel is an Adopter Member of the Zigbee Alliance. To use the Microchip Stack for ZigBee, either you must be a member of the Zigbee Alliance. Alternatively we as alliance members can program your firmware into the Pixies and sell you pre-programmed ZigBee devices for commercial use.

**OUI number / MAC address**

All IEEE 802.15.4 devices such as Pixie must use a unique 8-byte MAC address. The first 3 bytes (the ‘OUI’ number) are licensed to a company by IEEE for an annual fee. The licensee is then free to uniquely assign the remaining 5 bytes.

FlexiPanel Ltd can provide MAC addresses under its OUI license for the transceivers it manufactures. OEMs therefore do not need to apply for an OUI number. FlexiPanel Ltd’s OUI number is 00-15-C8.

For R&D purposes (i.e. in the lab or classroom), the following MAC addresses may be used:

00:15:C8:38:41:00:00:00 to 00:15:C8:38:41:00:FF:FF


For commercial purposes, contact FlexiPanel Ltd or your distributor for an allocation of MAC addresses.

**Preloading the MAC Address**

All firmware from FlexiPanel Ltd permits the use of SQTP to automatically set MAC addresses when programming using Microchip Technology’s PM3 programmer. The ROM locations 0x006A to 0x007B contain the following assembler code:

```
0x006A RETLW Macbyte0
0x0066 RETLW Macbyte1
0x006E RETLW Macbyte2
0x0070 RETLW Macbyte3
0x0072 RETLW Macbyte4
0x0074 RETLW Macbyte5
0x0076 RETLW Macbyte6
0x0078 RETLW Macbyte7
0x007A RETLW AddrValid
```

During initialization, the firmware checks to see whether the MAC address has been set. If it has not, and AddrValid is nonzero, it will set the MAC address to the values given in Macbyte0 to Macbyte7. The firmware ships with AddrValid equal to zero, but it may be set during programming using SQTP.

SQTP is a feature of the PM3 programmer that allows you to generate a file of IDs, in this case MAC addresses, and allocate them uniquely during programming. To set up SQTP to allocate MAC addresses:

1. In MPLAB, ensure the PM3 programmer is selected.
2. From the Programmer menu, select Settings…
3. Fill in the SQTP tab as shown in the following graphic, except insert the start of your personal MAC allocation in the “Start Value” box, and the number of sequential MAC addresses to allocate in the “Number of parts” box. (Note the Start value is FF, specifying the AddrValid value, followed by the MAC address, most significant byte first.)
4. Press the generate button to create a file of MAC addresses. When the file is created, dismiss the Settings... dialog by pressing OK. The file that was created contains a hex record each MAC address containing nine RETLW instructions encoding the address.

5. Each time programming session, select Load SQTP file... from the Programmer menu. In each programming cycle, MPLAB will read one hex record and insert the values into ROM locations 0x006A to 0x007B. It will then comment out the line so it is not used again. In this way the MAC addresses will be automatically allocated in sequence. NEVER maintain more than one copy of the SQTP file, otherwise you risk duplicating MAC addresses.

Radio Frequency

Optimizing RF Performance

The integral antenna supplied with Pixie is a modified quarter wave F antenna with efficiency of approximately 70%. This is comparable to a dipole and about as high as physically possible. Its radiation pattern is approximately omnidirectional.

Free space range is approximately 120m. In-building ranges of 20m to 40m would normally be expected.

The enclosure employed may affect antenna performance. Avoid the use of metal enclosures. Gray and black plastics should be used with caution. They often contain carbon, which degrades performance considerably. To test for the presence of carbon, heat the plastic in a microwave oven for one minute. If it melts, it probably contains carbon.

A high location is recommended, particularly for avoiding interference from transient objects such as passers-by and their cellphones. Microwave ovens are particularly problematic for the short periods in which they are operational. Where possible, orient all the antennas in the same direction so that their polarizations are the same. In a mesh network, a highly attenuating medium such as a floor may be bridged by placing two routers close to each other, one on either side of the barrier.

### Mechanical

**PCB Layout**

For PCB pad layout, refer to the engineering drawing on page 10. Note how the pads on the main board protrude out from beneath the module. This provides a point to apply heat when manually soldering and also for continuity testing. In addition, the size of the pad ensures a large reservoir of solder is available to the actual contact area.

**Soldering**

If soldered using a reflow oven, the surface mount module may be treated like a BGA package. The quality of the joint may be tested by checking for continuity between the pad on the upper side of the board and the protruding part of the pad on the main board.

The recommended reflow profile is shown in figure 8. Small increases in temperature (no larger than 15°C) may be needed depending on the density of the main board. The module should be refloved one time only.

### Table 2. Attenuation of typical bulk materials

<table>
<thead>
<tr>
<th>Material</th>
<th>Reduces range by factor of</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reinforced concrete floor</td>
<td>30</td>
</tr>
<tr>
<td>Brick wall with window</td>
<td>1.25</td>
</tr>
<tr>
<td>Office wall</td>
<td>2</td>
</tr>
<tr>
<td>Metal cabinet</td>
<td>3</td>
</tr>
<tr>
<td>5mm potting compound</td>
<td>2 - 3</td>
</tr>
<tr>
<td>Vegetation</td>
<td>1.25 per meter</td>
</tr>
</tbody>
</table>

*Compact materials, e.g. humans, have an complex effect.*

*In general attenuating most noticeably when closest to the antenna.*

![Figure 8. Recommended reflow profile](image)

<table>
<thead>
<tr>
<th>Temperature, Celsius</th>
<th>Time, Seconds</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>50</td>
<td>100</td>
</tr>
<tr>
<td>100</td>
<td>200</td>
</tr>
<tr>
<td>150</td>
<td>300</td>
</tr>
<tr>
<td>200</td>
<td>400</td>
</tr>
<tr>
<td>250</td>
<td>500</td>
</tr>
</tbody>
</table>
If soldering manually, use the following procedure:

1. Tin the contact pads on the module, trying to get more or less the same amount of solder on each. Work on a soft surface so that the components on the topside are not damaged.

2. Tin contact pads on main board.

3. Place the module in position on the main board.

4. Starting with the pads most likely to be in physical contact, apply heat with a soldering iron to the exposed part of the main board pads. Abut the iron against the edge of the module so that the heat is transmitted to the contact area of the pads. After 10-15 seconds, remove heat. Around 90% of pads should be successfully soldered.

5. Test for continuity between the pad on the upper side of the board and the protruding part of the pad on the main board.

6. Rework non-conducting contacts by applying heat again and a little extra solder.

**Pixie Lite**

Pixie Lite is a reduced function device based on the 18LF2520 microprocessor. It has sufficient memory to implement end devices only.

Bear the following points in mind when developing applications with Pixie Lite:

- Pin connections RE1 and RE2 are not present and their pads should be omitted on the PCB layout.
- The firmware must be recompiled to suit an 18LF2520 target.
- Flash table writes are 32 bytes instead of 64 so the WRITE_BLOCK_SIZE value in zNVM.c must be set to 32ul. (See Microchip DS80229B.)
- A single-stack model is recommended.
- ROM reduces from 64K to 32K.
- RAM reduces from 3986 bytes to 1536.

See Appendix I for a suitable linker script for the Microchip Stack for ZigBee running on the PIC18LF2520.

**Future releases of Pixie & Pixie Lite**

The following changes are anticipated in future revisions of Pixie and Pixie Lite.

- The 18LF2520 on the Pixie Lite may be replaced by an 18LF2620 if new features in the ZigBee stack, such as security, require the extra memory.

**Bibliography**

- **Microchip Stack for ZigBee Protocol**, supplementary notes included with the Microchip Stack for ZigBee firmware downloadable from www.microchip.com.


**U.FL connector data sheet**, contact FlexiPanel Ltd for details.
**Drawing**

**DRWG-ZPCr10**

**Date**

11 June 06

**Drawn by**

R G Hoptroff

**Description**

Pixie ZigBee module (rev 10)

**Notes**

1. Dimensions in mm
2. Module height 3.6mm
3. Keep antenna side as free of components as possible, preferably overhanging the edge of the main board
4. Pour as much grounded copper as possible on the main board, but none on top layer below Pixie module
5. If pins fitted, pin pitch is 20.32mm
6. Pixie Lite same size & footprint as Pixie
7. Incorporates FCC / CE / IC certified EasyBee transceiver ZBMr10
Appendix I

Linker script for 18F2520

The following linker script is used by the Pixie Switcher Lite firmware and serves as an example for migration of the Microchip Stack to the PIC18F2520. Note that a single bank stack (256 bytes) is used, and the heap size has been reduced to 512 bytes.

// File: zc2520i.lkr
// Sample ICD2 linker script for the PIC18F2520 processor
LIBPATH .
FILES c018i.o
FILES clib.lib
FILES p18f2520.lib
CODEPAGE NAME=vectors START=0x0 END=0x29 PROTECTED
CODEPAGE NAME=page START=0x2A END=0x7D7F
CODEPAGE NAME=debug START=0x7D80 END=0x7FFF PROTECTED
CODEPAGE NAME=idlocs START=0x200000 END=0x200007 PROTECTED
CODEPAGE NAME=config START=0x300000 END=0x30000D PROTECTED
CODEPAGE NAME=devid START=0x3FFFFE END=0x3FFFFF PROTECTED
ACCESSBANK NAME=accessram START=0x0 END=0x7F
DATABANK NAME=gpr0 START=0x80 END=0xFF
// Stack
DATABANK NAME=gpr1 START=0x100 END=0x1FF
// Spare
DATABANK NAME=gpr2 START=0x200 END=0x2FF
// Heap – set MAX_HEAP_SIZE to 512
DATABANK NAME=gpr3 START=0x300 END=0x4FF PROTECTED
// Spare
DATABANK NAME=gpr5 START=0x500 END=0x5F3
DATABANK NAME=dbgspr START=0x5F4 END=0x5FF PROTECTED
ACCESSBANK NAME=accesssfr START=0xF80 END=0xFFF PROTECTED
SECTION NAME=CONFIG ROM=config
STACK SIZE=0x100 RAM=gpr1
Reference

Electrical

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Supply Voltage (regulated) Vcc</td>
<td>2.1V to 3.6V</td>
</tr>
<tr>
<td>Supply Voltage (unregulated) Vunreg</td>
<td>2.4V to 20.0V, if regulator fitted</td>
</tr>
<tr>
<td>Current consumption, active</td>
<td>25mA</td>
</tr>
<tr>
<td>Current consumption, sleep mode</td>
<td>2μA</td>
</tr>
</tbody>
</table>

For more information electrical specifications, consult the PIC18LF4620 and PIC18LF2520 data sheets. (See bibliography.)

Radio Frequency

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max RF output power</td>
<td>1mW = 0dBm</td>
</tr>
<tr>
<td>RF frequency range</td>
<td>2400MHz to 2485MHz</td>
</tr>
<tr>
<td>Communications protocol</td>
<td>IEEE 802.15.4 (DSSS O-QPSK chip encoding)</td>
</tr>
<tr>
<td>Raw data rate</td>
<td>250kbit/s</td>
</tr>
<tr>
<td>RF channels</td>
<td>16</td>
</tr>
<tr>
<td>Free space range with integral antenna</td>
<td>Approx 120m</td>
</tr>
</tbody>
</table>

Mechanical

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max operating/storage temperature</td>
<td>-40°C to +85°C</td>
</tr>
<tr>
<td>Dimensions L×W×H mm</td>
<td>53.7 × 19.75 x 3.1 (excluding DIL pins)</td>
</tr>
</tbody>
</table>

Regulatory

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>FCC compliance</td>
<td>Certified</td>
</tr>
<tr>
<td>CE compliance</td>
<td>Certified</td>
</tr>
<tr>
<td>FCC external labeling requirement</td>
<td>“Contains Transmitter Module FCC ID: UGAZBM10”</td>
</tr>
</tbody>
</table>

FlexiPanel is an Adopter Member of the Zigbee Alliance. To use the Microchip Stack for ZigBee, either you must be a member of the Zigbee Alliance. Alternatively, we as Alliance members can program your firmware into the Pixies and sell you pre-programmed ZigBee devices for commercial use.

Manufacturer Contact Details

Pixie is assembled by agreement by RF Solutions Ltd:

R F Solutions Ltd
Unit 21, Cliffe Industrial Estate,
Lewes, E. Sussex, BN8 6JL, United Kingdom
www.rfsolutions.co.uk
Tel: +44 (0)1273 898 000, Fax: +44 (0)1273 480 661
email: sales@rfsolutions.co.uk

Developer Contact Details

EasyBee is designed and owned by FlexiPanel Ltd:

FlexiPanel Ltd
2 Marshall St, 3rd Floor,
London W1F 9BB, United Kingdom
email: support@flexipanel.com
www.flexipanel.com