

Chapter 4

Testbed and Experimental Setup

This chapter elaborates on some important testbed considerations as well as the testbed and experimental setup used to evaluate energy efficient routing schemes. The implementation of a shortest hop path routing scheme and a MTTP routing scheme is also detailed.

4.1 WSN Testbed Considerations

To ensure accurate results a number of important factors need to be considered. This section details some important considerations for a WSN testbed.

4.1.1 Fraunhofer Region

To ensure accurate results, the separation between nodes must fall within the Fraunhofer region. The Fraunhofer region is also known as the far field of the radiator. The wavelength of a radio wave is given by Equation 4.1. Where λ is the wavelength, c is the speed of electromagnetic radiation and f is the wave frequency.

$$\lambda = \frac{c}{f} \quad (4.1)$$

The operating frequency used by the transceivers is 2.4 GHz. The wavelength is therefore:

$$\begin{aligned} \lambda &= \frac{c}{2.4 \times 10^9} \\ \lambda &\approx 0.125m \end{aligned}$$

Equation 4.2 is used to calculate the Fraunhofer distance (d_f) [55]. D is the maximum physical dimension of the radiator.

$$d_f = \frac{2D^2}{\lambda} \quad (4.2)$$

The maximum physical dimension of the external antenna is 0.1358 m. The Fraunhofer distance is therefore:

$$\begin{aligned} d_f &= \frac{20.1358m^2}{0.125m} \\ d_f &\approx 0.295m \end{aligned}$$

The Fraunhofer region starts at a distance of 0.295 m from each node. A minimum distance of 0.5 m is used to separate the nodes that form the testbed. This ensures that all the nodes fall within each others Fraunhofer regions.

4.1.2 Co-existence of IEEE Standard 802.15.4 and IEEE Standard 802.11 Devices

Seeing that the testbed will be deployed in an environment subject to IEEE standard 802.11 interference, attention has to be given to the co-existence of the IEEE 802.11 and

IEEE 802.15.4 standards. The results in [56] show that IEEE 802.11b networks and microwave ovens have a significant effect on the performance of an 2.4 GHz IEEE 802.15.4 network. However this interference can be mitigated by proper channel selection. Figure 4.1 details the channel layout of the 802.11 and 802.15.4 standards [57].

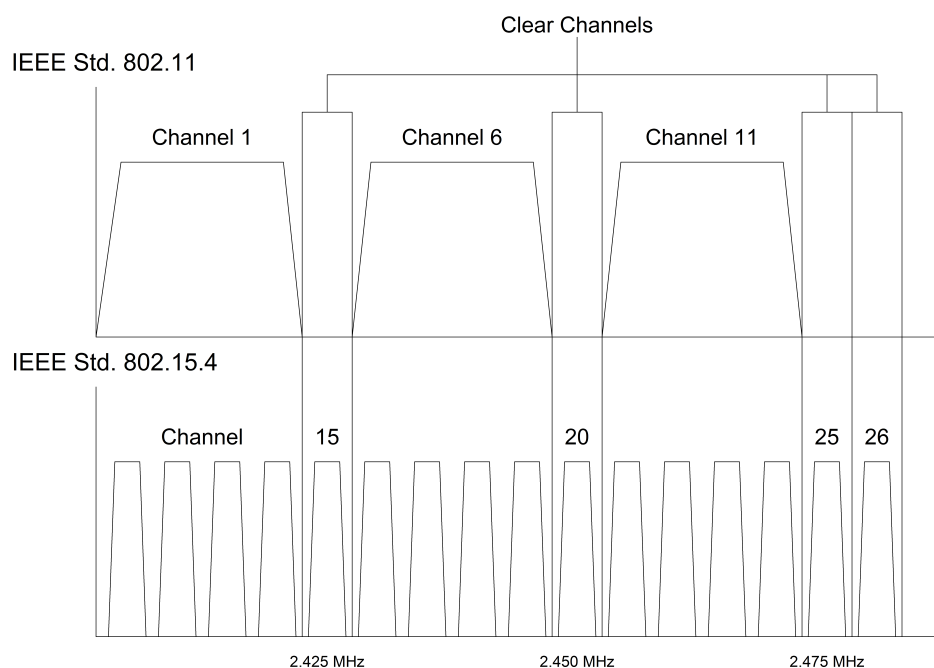


Figure 4.1: 802.11 and 802.15.4 Standards Channel Layout Diagram

By operating an IEEE standard 802.15.4 network in channel 15, 20, 25 or 26 interference with an IEEE standard 802.11 network can be minimized. This is why the WSN testbed is operated in channel 25. The testbed is also distanced from other Industrial, Scientific and Medical (ISM) band noise sources such as microwave ovens.

4.1.3 MAC Filter

Because all the nodes within the network fall within each others communication range, a MAC filter was used to ensure that only neighbouring nodes can communicate with each other. This MAC filter is enforced by the SN firmware after a message has been

received. If the node from which the message originated is one of the current node's directly neighbouring nodes the message is routed. Otherwise the message is discarded.

4.2 Energy Aware Routing Scheme Design and Implementation

The implementation of a shortest hop path routing scheme as well as a MTTPR scheme is detailed in this section. The routing framework used to implement these routing schemes as routing protocols is also presented.

4.2.1 Routing Framework

The routing framework used is a simple distance vector routing protocol similar to the Routing Information Protocol (RIP). Three types of packets are used by this routing protocol. These packets are listed below:

- Discovery packets;
- Update packets;
- Data packets.

Discovery packets alert nearby nodes of the current node's presence. Update packets share the routing table of each node with its neighbouring nodes. Data packets are used to forward data to a certain destination node. Invalid, flush and holddown timers are not implemented, seeing that they are not needed for the experiments. The purpose of this routing framework is only to provide a platform to evaluate energy aware routing schemes.

Figure 4.2 details the flow of the routing framework. The routing function is called by the SN firmware after a message has been received. The design of the SN firmware is

detailed in Section 4.3.5. The routing function first checks if the received packet is a discovery packet. If a discovery packet was received the SN checks if there is a routing table entry for the node from which the message originated. If a entry does not exist the node is added to the routing table. If a entry does exist the node checks if the cost metric of the received packet is smaller than the cost metric of the route stored in the routing table. If the cost metric is smaller the existing entry is replaced. When the node receives a update message, it checks if each route in the update message is in its routing table. If a route is not in the routing table it is added. If the route is already in the routing table the SN calculates which route has the lowest cost metric. If the new route has the lowest cost metric the old route is replaced. Finally if the received packet is a data packet it is forwarded the the next hop node in the routing table.

4.2.2 Shortest Hop Path Routing Scheme and Minimum Total Transmission Power Routing (MTTPR) Scheme Implementation

To implement a shortest hop path routing scheme, the routing framework described in the previous subsection is modified to use a hop count metric as a cost metric for selecting an optimal path. This metric is incremented by a value of 1 for each hop along a path. This scheme attempts to minimize delay and energy consumption by selecting the shortest hop path between communicating nodes. However this routing scheme does not take the transmission power setting of each SN into account.

The MTTPR scheme, which was detailed in Section 2.5.1 , uses a cost metric which represents the total transmission power along each path. This routing scheme attempts to minimize the networks energy consumption by selecting the minimum total transmission power route. This routing scheme might choose longer routes than a shortest hop path routing scheme, which means that the end-to-end delay between communicating nodes may be increased. To implement a MTTPR scheme, the routing framework described in the previous subsection is modified to use a MTTP metric as a cost metric for selecting an optimal path. This metric is incremented by a value corresponding to

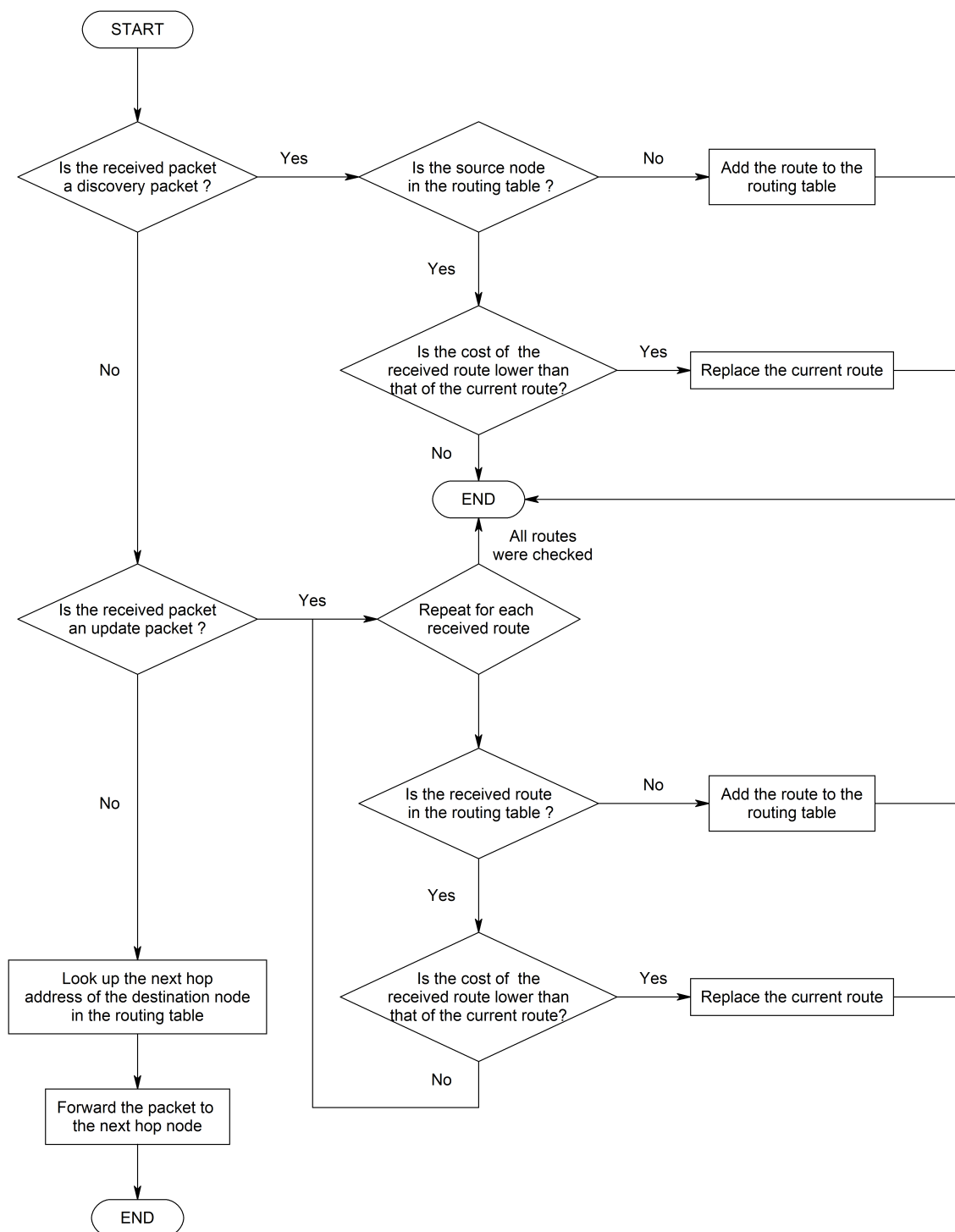


Figure 4.2: Routing Framework Flow Diagram

the node transmission power setting for each hop along a path.

4.3 Testbed and Experimental Setup

This section details the experimental setup used to evaluate different routing schemes, on the WSN testbed developed for this purpose. The setup of the testbed and the SN firmware design are also detailed.

4.3.1 Experiment Flow

The flow of a typical experiment can be seen in Figure 4.3. A routing protocol and a data generation protocol are implemented on the system, after which the firmware is modified for each individual node. These modifications include changing the MAC and network layer addresses as well as the transmission power setting of each node. Thereafter the testbed nodes are programmed, their RTCCs are synchronised and their memory is cleared (the memory counter/pointer is reset). The network is then deployed and an experiment is run until a certain predetermined number of repetitions is completed. In this case the experiments are conducted for 1 hour each. When the experiment is finished the SNs are collected and the experiment data are downloaded from each SN. Finally the data is exported to an Excel spreadsheet and a statistical analysis is performed. Three experiments for each routing scheme as well as three control experiments were conducted. During the control experiments no routing takes place. The results of these experiments are presented in the next chapter.

4.3.2 Hardware Setup

The deployed testbed setup can be seen in Figure 4.4. Each incrementation, of the x and y coordinates, corresponds to 0.5 m in the actual deployment. The transmission power of all the nodes on the four possible shortest hop paths between the data generator and

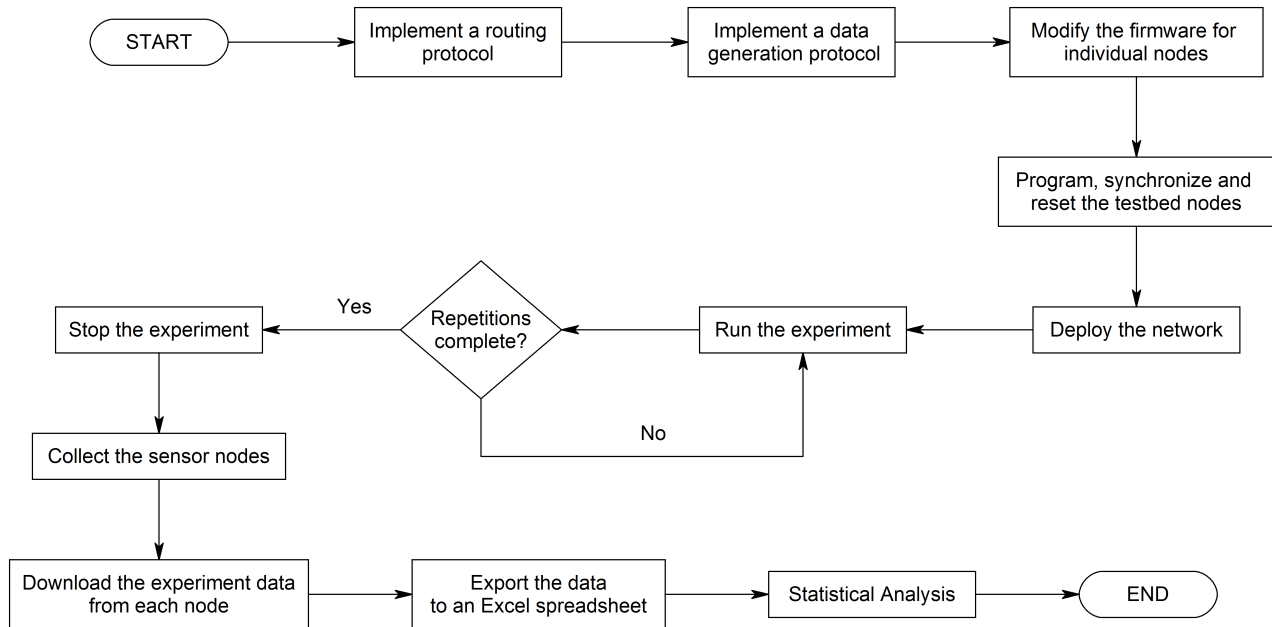


Figure 4.3: Experiment Flow Diagram

destination node, including the data generator and destination node, is set to the maximum value (19 *dBm*). The transmission power of the rest of the nodes in the network is set to the minimum value (−26.4 *dBm*). A MAC filter is used to enforce the testbed topology. This ensures that each node can only communicate with its directly neighbouring nodes. For example, the node in position (1,1) can only communicate with the nodes in positions (1,2), (2,1) and (2,2). The use of this MAC filter combined with the strategic placement of nodes with different transmission power settings ensures that the energy consumption difference between the routing schemes can be observed. This is due to the fact that the MTTTPR scheme will attempt to avoid the high transmission power nodes while the shortest hop path routing scheme will not.

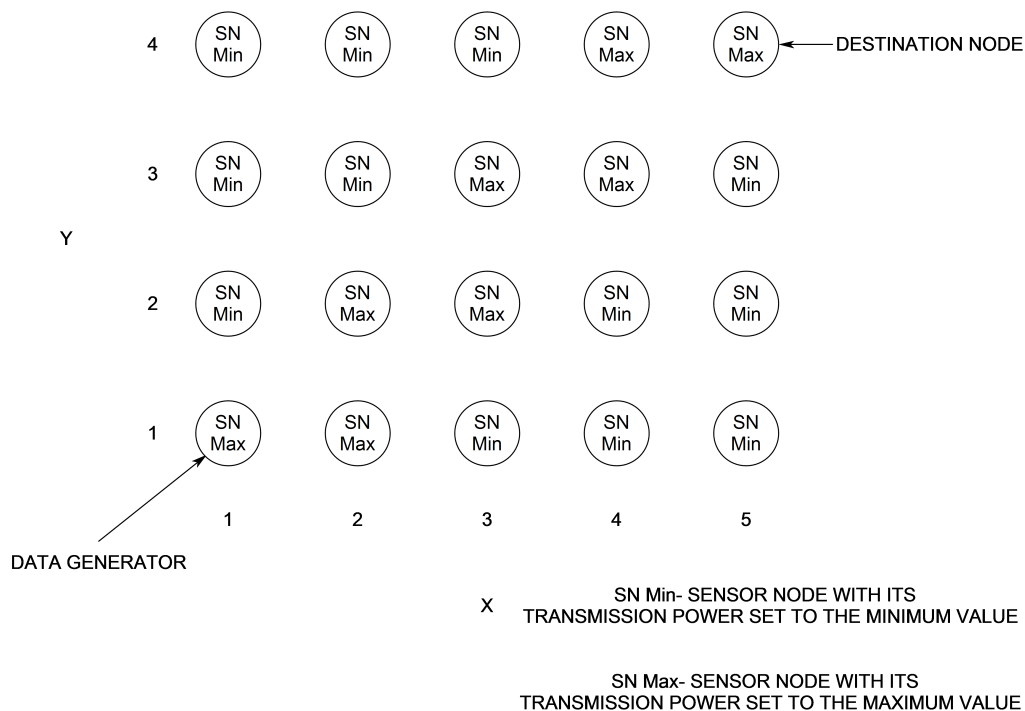


Figure 4.4: Testbed Hardware Setup Diagram

4.3.3 Data Generator Setup

During the experiments the node in Position (1,1) is used as a data generator. The data generator generates 200 packets, with a payload of 50 bytes each, every 10 seconds. These packets are destined for the node in position (5,4).

4.3.4 Sensor Node Current Measurement Calibration

The shunt resistor of each SN was measured using a Tektronix DMM4050 precision digital multimeter. A plot of the measured and expected shunt resistance of each SN can be seen in Figure 4.5. The expected resistance of each shunt resistor is 300 $m\Omega$. However the resistance of the shunt resistors used varies between 369 and 371 $m\Omega$. This could be due to a fault in the manufacturing process or because the resistors were mislabelled by the manufacturer. The SNs are calibrated by making use of a shunt resistance value of 370 $m\Omega$ instead of 300 $m\Omega$, in the software used to collect data from

SNs.

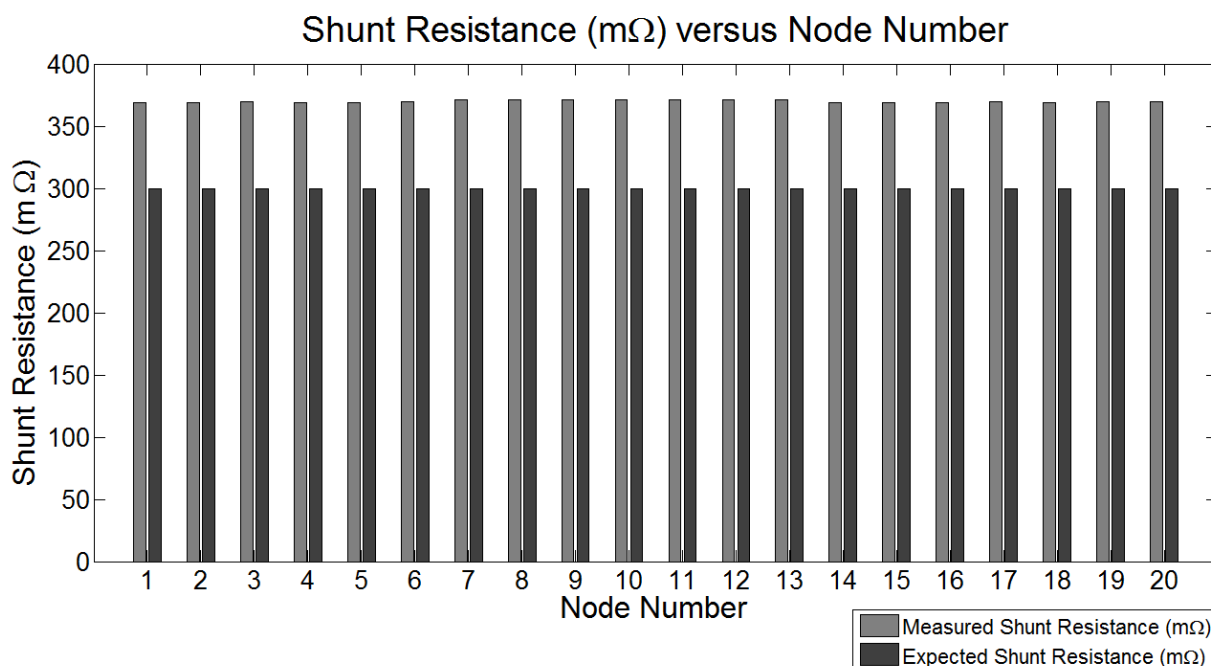


Figure 4.5: Plot of the Measured and Expected Shunt Resistance of each SN

4.3.5 Sensor Node Firmware Setup

This section details the SN firmware setup used for the experiments. The experiment firmware was designed with modularity in mind, to ease the implementation of different routing protocols. This modularity feature also extends to the usage of different Microchip transceivers. The SN firmware can be found on the CD accompanying this document (Appendix F).

Sensor Node Firmware Flow

The flow of the SN firmware can be seen in Figure 4.6. After startup each node is initialized and a startup check is performed to ensure that all the SN components are functional. The STAT 1 LED is turned on to indicate a successful initialization and startup check. If the node is a data generation node the data generation routine is

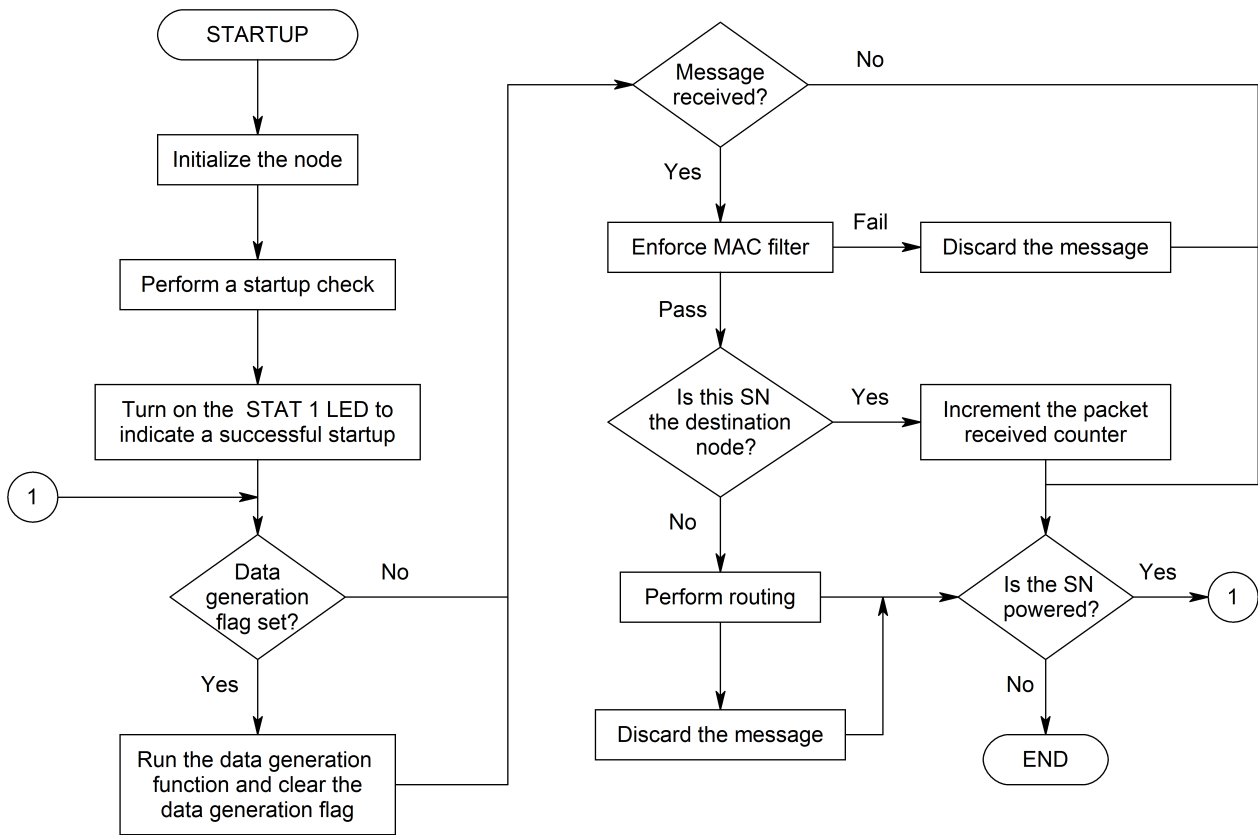


Figure 4.6: Sensor Node Firmware Flow Diagram

performed next, if the data generation flag is set. This flag is set by the RTCC interrupt handler. Thereafter, irrespective of whether the data generation flag was set, the SN checks if a message has been received. If a message has been received the MAC filter is enforced. If the message originated from a valid node, with the current node being the destination, the received packet counter is incremented and the packet is discarded. Otherwise the packet is passed to the routing routine and then discarded. This process is repeated as long as the node is powered. The next section details the interrupts that support the SN firmware flow.

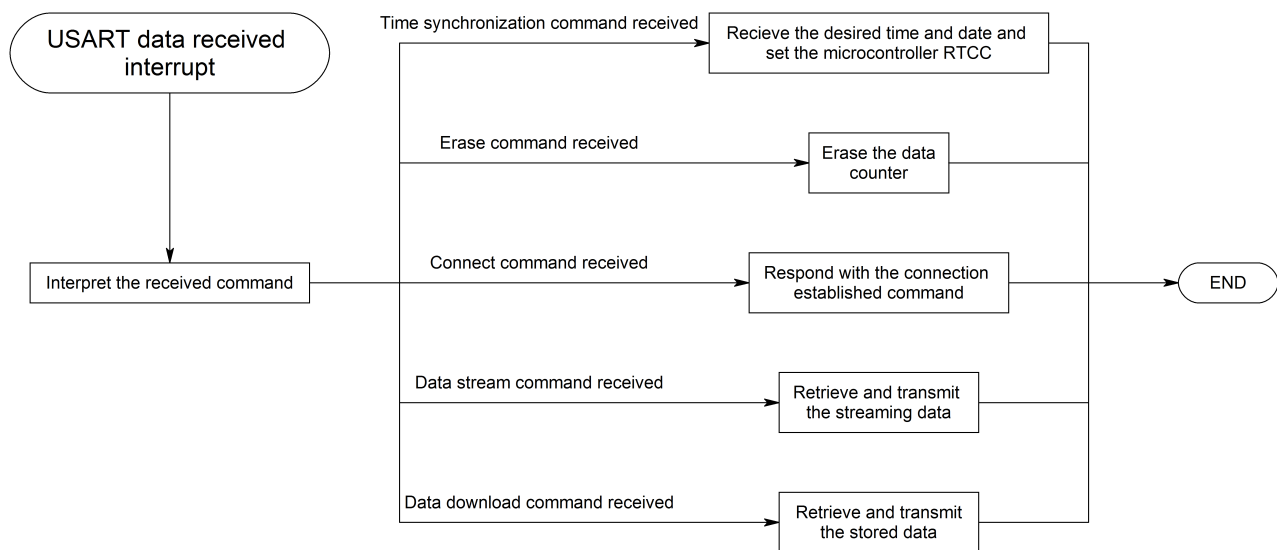


Figure 4.7: USART Message Received Interrupt Flow Diagram

Sensor Node Firmware Interrupts

The USART message received interrupt handler, which can be seen in Figure 4.7, is responsible for receiving and interpreting the different commands originating from a personal computer. After a command has been received and interpreted the interrupt handler performs the required operations. This interrupt has the highest priority.

Figure 4.8 depicts the 10 second RTCC interrupt. This interrupt sets the data generation flag every 10 second if the node is a data generation node. This interrupt is also responsible for broadcasting update and discovery messages every 10 seconds. The interrupt handler stores the SN's power consumption and routing data every 60 seconds.

The MCP3911 data ready interrupt is detailed in Figure 4.9. The interrupt handler first reads the current and voltage data from the MCP3911. Thereafter it calculates a cumulative moving average for each of these values. This cumulative moving average is restarted every minute after the data has been stored on the external EEPROM.

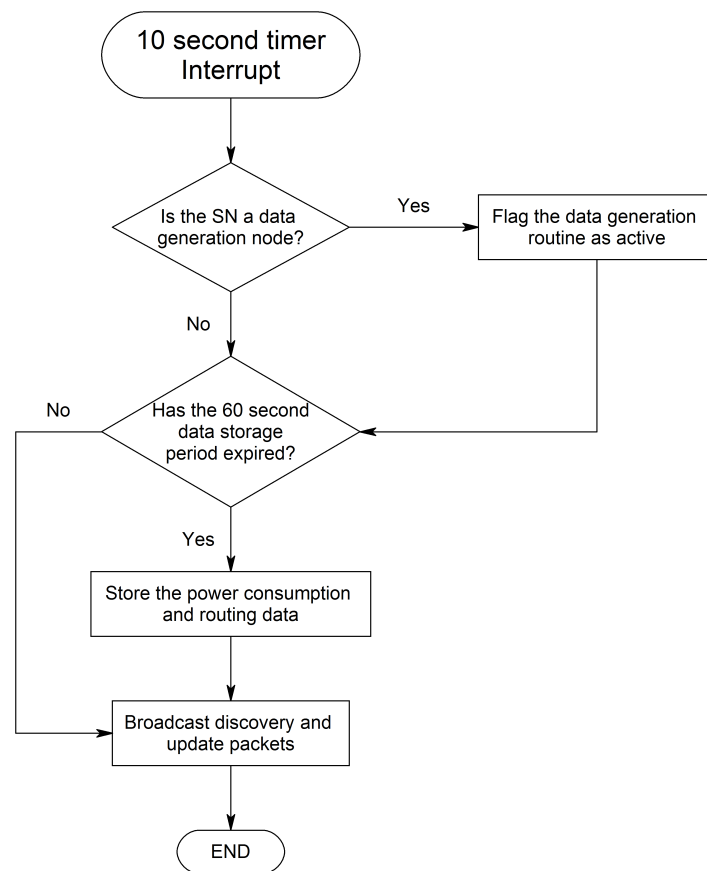


Figure 4.8: 10 Second RTCC Interrupt Flow Diagram

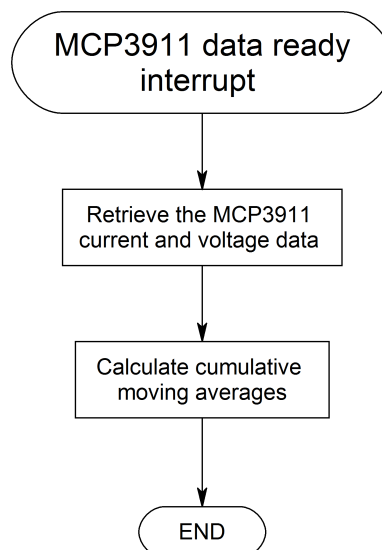


Figure 4.9: MCP3911 Data Ready Interrupt Flow Diagram

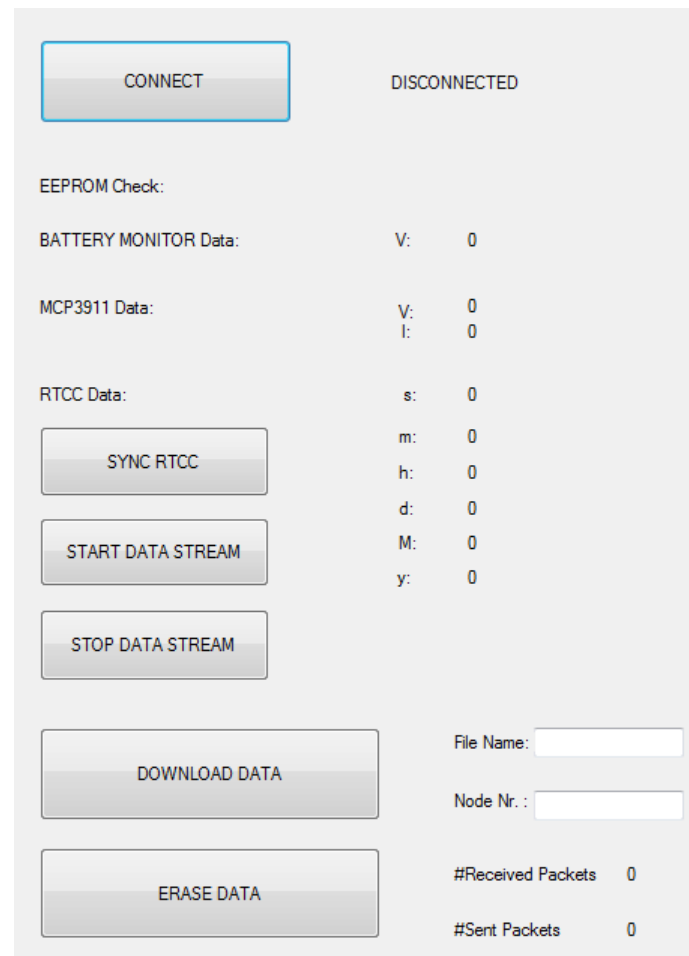


Figure 4.10: Picture of the Sensor Node Software Tool GUI

4.3.6 Software Setup

A C# application was developed to display data streamed from SNs and to download each SN's stored data. The application is also used to set the RTCC of each SN. Figure 4.10 depicts the application Graphical User Interface (GUI). This application can be found on the CD accompanying this document (Appendix F).

4.3.7 Data Generator Timing

Each frame generated by the data generator consists of 77 bytes. Equation 4.3 is used to calculate the transmission time (t_{frame}) of the frame. Where R_b is the data rate in *kbps* and F_l is the frame length in bytes. For a 77 byte frame this corresponds to 2.464 *ms*. The maximum number of hops expected, for the routing schemes that are to be tested, is 5 hops. This translates to a transmission time of 12.32 *ms* for the maximum length route. A 20 *ms* delay is added before every transmission of the data generator to ensure that the packet has enough time to propagate through the network.

$$t_{frame} = \frac{F_l \times 8}{R_b} \quad (4.3)$$

The data generator generates 200 packets every 10 *s*. The time needed to transmit 200 frames is 0.49 *s*. For the maximum expected route length that corresponds to 2.45 *s* of transmission time every 10 *s*. The remaining 7.55 *s* provides enough time for the CSMA-CA used to send the 200 frames as well as the time needed to broadcast update and discovery packets.

4.3.8 Experimental Parameters

Table 4.1: Wireless Platform Setup

Platform Attribute	Status/Value
MAC Address Length	8 Bytes
PA/LNA	Enabled
Protocol	P2P
Rx Buffer Size	120 Bytes
Tx Buffer Size	120 Bytes
Security	Disabled
Transceiver	MRF24J40

The setup of the wireless platform is detailed in Table 4.1. The MAC address length is 8 bytes. To ease implementation the network layer address of each node is the same

as the last byte of its MAC address. Microchip's Point to Point (P2P) protocol is used because it provides a convenient API, which can be used to implement different routing protocols. The transmit and receive buffers have a capacity of 120 bytes each. The security module of the platform is not enabled.

Table 4.2: General Experimental Parameters

Experiment	Repetitions	Duration (hours)	Topology Enforcement
Control	3	1	MAC Filtering
Shortest Hop Path Routing	3	1	MAC Filtering
MTTPR	3	1	MAC Filtering

Table 4.2 details the general parameters of the experiments presented in chapter 5.2. Control, shortest hop path routing and MTTPR experiments were preformed. Each of these experiments were repeated 3 times. In each case a MAC filter was used to enforce the network topology. This MAC filter functions as described in Sections 4.1.3 and 4.3.2.

4.4 Conclusion

This chapter detailed the setup of the WSN testbed as well as the experimental setup used to compare a shortest hop path routing scheme and a MTTPR scheme. Because these routing schemes cannot function on their own a simple distance vector routing based routing framework was used to implement them. The working of this framework, as well as the routing schemes supported by the framework, was also detailed in this chapter.