# K10
# Mapper8

# ASSEMBLY MANUAL

A and T Labs Inc.
P.O. Box  4884
Wheaton, IL 60187
Tel: 630-668-7870
Fax: 630-668-7870
Web: www.a-and-t-labs.com
Technical Support: support@a-and-t-labs.com

**Table of Contents**

# 1. INTRODUCTION

Congratulations on purchasing the Mapper8 cable tester! The Mapper8 provides complete connectivity testing of 8-wire modular cables as well as other cables. The Mapper8 can very quickly check for wiring problems such as opens, shorts, reversed pairs, etc.  Connectivity is compared against a predefined or a user-defined connectivity map.  The Mapper8 displays both a quick pass/fail indication as well as details on how the cable is wired or mis-wired.

We hope this will be a successful and educational building experience for you. Toward that end, we strive to provide complete information, but it is advisable to check the A and T Labs WEB site for any additional notes, changes, or hints that may be helpful. If you find errors or omissions or have suggestions for improving these instructions, please give us your feedback.

We begin with an overview of how the design works.

# 2. HOW IT WORKS

The Mapper8 consists of two units – a master and a remote. The designs are very similar, in fact they share a similar circuit board layout. Please refer to the operator's manual for usage descriptions and details.

## 2.1 Design Overview

A block diagram overview of the Mapper8 master and remote unit hardware is shown in Figure1. The two units have very similar hardware design, but with some differences.  At the heart of each unit is a PIC16LF872: a Low-cost 8-bit microcontroller from Microchip.  The PIC16LF872 contains many features all integrated on-chip: a 8-bit CPU, program memory, data RAM, data EEPROM, 22 I/O pins and a variety of peripherals.  For more information see Microchip's data sheet for the PIC16LF872 at www.Microchip.com .

The microcontroller's on-chip EEPROM data memory is used to permanently store information while the units are powered off.  The master unit's currently selected reference map and user-defined connectivity maps are stored in the EEPROM.  For the remote unit, the remote's ID, the currently selected mode (remote transmitter mode or tone mode) and selection of which wire pairs receive tones are stored in the EEPROM.

The detailed schematic of the master unit hardware is shown in Figure 2.  The master unit consists of a PIC16LF872 microcontroller (IC1) connected to an 8-pin modular jack (J1), a set of readout LEDs (D1-D14), a momentary programming switch (SW2) and a momentary switch (SW1) controlled 3V battery (BAT).  The microcontroller derives its clock timing from an RC oscillator (using R1, C1 and IC1).  The microcontroller's external reset input control is held inactive by a pull-up resistor (R2).  Resistor network devices with four independent resisters per device are used to limit LED drive current (R3-R6).

The detailed schematic of the remote unit hardware is shown in Figure 3.  The remote unit consists of a PIC16LF872 microcontroller (IC1) connected to an 8-pin modular jack (J1), a set of readout LEDs (D1-D10), a momentary programming switch (SW2) and an on/off switch (SW1) controlled 3V battery (BAT). The microcontroller derives its clock timing from an RC oscillator (using R1, C1 and IC1).  The microcontroller's external reset input control is held inactive by a pull-up resistor (R2).  Resistor network devices with four independent resisters per device are used to limit LED drive current (R3-R5).

Key to Mapper8's operation is the software in the microcontroller of each unit.  The remote unit's software configures the microcontroller's I/O port connected to the modular jack as an output port.  It then proceeds to send messages out the individual pins of the port.  On the other hand, the master unit's software configures its I/O port as an input and listens for messages on the individual pins.  We will look at the details of the message communication next.

### 2.2 Messages from There to Here

When a remote unit is attached to one end of the cable and turned on, it begins looking for request pulses from a master unit.  When the remote receives a request pulse on a particular wire it responses by transmitting a coded message back on that wire. Each message is composed of a wire ID field and the remote's ID.  Messages sent back on wire 1 have the first bit of their wire ID field set to 1, messages sent on wire 2 have the second bit set, and so on for each wire. Figure 4 contains an illustration of the message bit format.  Three bits are used to encode the remote unit's ID and eight bits are used to indicate the wire number.

On the other end of the cable the master unit sends a request pulse on each of its wires and looks for response messages.  By noting the wire number (from the master's perspective, the near-end of the cable) and the wire ID in the messages (wire numbers on the far-end of the cable), the master unit can determine the connectivity between the two units.  For example, if on wire 1 the master receives a message with the eighth bit of the wire ID field set, then we know that wire 1 on the near-end is connected to wire 8 on the far-end of the cable.  Another example would be, if on wire 1 the master receives a message with both the seventh and eight bit of the wire ID field set, then we know that wire 1 on the near-end is connected (or shorted) to both wire 8 and 7 on the far-end.  Finally, if on a given wire the master receives no messages, then we know that that the wire is not connected (open) to the far-end.

As it scans through all eight wires, the master records which messages it receives on which wires, thus building a table or map of connectivity between the far and near ends. Once the master completes this scan, it compares this measured connectivity map to the selected reference map.  If they match, the master unit declares the test passed, otherwise it declares failure.

Note that master and remote units are independently powered from separate 3V supplies.  As completely separate units the master and remote do not have an explicit common ground reference between them. The units only drive a request pulse or message out one wire at a time. By having the units actively drive the remaining wires high with the push-pull drivers of the PIC16LF872's I/O pins, a reference between the two units is setup through the on-chip ESD protection diodes in the master unit microcontroller's I/O pins. This means that there must be at least two separate electrical paths (i.e. wires) between the two units to be able to test the cable.

Another issue to consider is the possible presence of shorts between wires.  We do not want the push-pull I/O drivers conflicting, so the master first tests for shorts to a given wire before it drives out the request pulse.  If a short is detected to the wire under test, then the additional shorting wire(s) are also driven with a request pulse. The remaining non-shorted wires are actively driven high.  The same request pulse and subsequent response message will appear on all of the wires in the shorted group.  An example of the short tests, request pulses and response messages is illustrated in Figure 5.

## 3. CONSTRUCTION

Before beginning, refer to and review the soldering techniques and suggestions and static electricity management information sheets. These are very important and must be followed to ensure the highest quality results.

The PC board layouts are designed to serve both as the interconnect for the components and as the cover or lid to the small plastic cases. The PC board traces and various surface mounted components go inside the case. Components that need to be seen or accessible (for example the LEDs and the jack) are mounted on the other side of the board which forms the outside cover of the case. These "outside" components have leads that go through holes in the PC board to the traces on the "inside" of the board. The PC boards have appropriate switch and LED legends silk screened on the "outside" of the boards, as well as solder mask and parts placement silk screened on the "inside" of the boards.

Following is a list of tools required to complete this project:

- Soldering iron, low wattage (15 to 25 Watts), extra fine tip
- Needle nose pliers
- Diagonal cutters
- Small Phillips screw driver
- Tweezers

Many of the Mapper 8 components are surface mount devices (SMD), which may present a new building challenge for some, due to their small size. Here are several tips for working with surface mounted devices :

- You should use a low-wattage (15 to 25 watts or so) soldering iron with an extra fine tip.
- Be sure to keep the tip clean.
- Tin one PC board pad for the SMD with a small amount of solder.
- Carefully place the SMD so that its leads are centered on and squarely aligned with the PC board pads.
- Using the tip of tweezers or a dental pick hold the SMD in place, then tack solder the one lead of the SMD. Always allow solder to flow from the pad and solder to the part/pin.
- If the SMD is not correctly positioned, reheat the one soldered lead and reposition the device.
- Once the part is properly positioned, solder the remaining leads.

Avoid using too much solder when soldering a pin. Only a very small amount is usually needed. Using very fine diameter solder is helpful. Depending on your eyesight, a magnifier might be helpful as well. Finally, make sure that your work area is clear of distracting stuff, so that when you drop that small surface mount device you can find it!

Placement diagrams for components are shown in Figure 6 and Figure 7 for the master and remote units, respectively. Solid-line part outlines indicate where components are surface mounted and soldered to the trace-side of the board. Dashed-line part outlines indicate where through-hole components are mounted on the opposite side and soldered on the trace-side.

### *3.1 Master Unit Assembly*

We will assemble surface mount components first, so that the board can lie flat on the work surface while these are installed.

Be sure to orient pin 1 of IC1 properly. Pin 1 is identified by a dot on the board layout.

To install SW2, carefully bend the four leads of switch SW2 so that it can be surface mounted to the board. The switch needs to be mounted on the board's "inside" with the switch's push-post passing through the board. See the diagram in Figure 8. When soldering SW2 to the board, make sure that the push-post is centered in the PC board's hole so that it will not bind.

### 3.1.1 Master unit surface mount components:

| ✓ | Part | Type | ✓ | Part | Type | ✓ | Part | Type |
|---|------|------|---|------|------|---|------|------|
| ☐ | IC1 | PIC16LF872-I/SO (Programmed w/ Master code) | ☐ | R4 | 82 ohm array | ☐ | C2 | .1uF |
| ☐ | R1 | 10K | ☐ | R5 | 82 ohm array | ☐ | SW2 | Miniature push button |
| ☐ | R2 | 10K | ☐ | R6 | 82 ohm array | | | |
| ☐ | R3 | 82 ohm array | ☐ | C1 | 22pf | | | |

After the surface mount components are soldered onto each board, install and solder the remaining through-hole components.

When installing the LEDs (Dxx) be very careful that they are installed with the proper orientation, see Figure 9. The shorter lead corresponding to the cathode, is oriented closest to the dashed line on the LED silk screen. So be careful, once the leads are soldered and clipped, you can not visually determine which lead is which!

The power switch (SW1) is installed and secured to the PC board using a lock washer and nut.

### 3.1.2 Master unit through-hole components:

| ✓ | Part | Type | ✓ | Part | Type | ✓ | Part | Type |
|---|------|------|---|------|------|---|------|------|
| ☐ | D1 | T1 red LED | ☐ | D7 | T1 red LED | ☐ | D13 | T1 green LED |
| ☐ | D2 | T1 red LED | ☐ | D8 | T1 red LED | ☐ | D14 | T1 red LED |
| ☐ | D3 | T1 red LED | ☐ | D9 | T1 yellow LED | ☐ | J1 | 8-pin modular jack |
| ☐ | D4 | T1 red LED | ☐ | D10 | T1 yellow LED | ☐ | SW1 | Push button switch (Momentary on) |
| ☐ | D5 | T1 red LED | ☐ | D11 | T1 yellow LED | | | |
| ☐ | D6 | T1 red LED | ☐ | D12 | T1 yellow LED | | | |

As shown in the schematic, connect to the battery holder's positive lead (red) to one side of the switch. The other side of the switch is connected to a PC board solder pad labeled J2 with a short piece of hook-up wire. This wire is generated by cutting about a one inch piece off the battery holder's red lead.The negative side (black) of the battery holder is connected to a PC board solder pad labeled J3.

Secure the battery holder to the bottom of the case using two pairs of self-adhesive Velcro pads.  This will keep the holder and batteries from rattling around and make it easy to replace the batteries in the future.

### 3.1.3 Master unit battery holder and switch wiring:

| ✓ | |
|---|---|
| ☐ | Jumper from SW1 to pad J1 |
| ☐ | Red battery holder lead to other side of SW1 |
| ☐ | Black battery holder lead to pad J2 |
| ☐ | Install Velcro to battery holder and case |

## *3.2 Remote Unit Assembly*

We will assemble surface mount components first, so that the board can lie flat on the work surface while these are installed.

Be sure to orient pin 1 of IC1 properly. Pin 1 is identified by a dot on the board layout.

To install SW2, carefully bend the four leads of switch SW2 so that it can be surface mounted to the board.  The switch needs to be mounted on the board's "inside" with the switch's push-post passing through the board.  See the diagram in Figure 8. When soldering SW2 to the board, make sure that the push-post is centered in the PC board's hole so that it will not bind.

### 3.2.1 Remote unit surface mount components:

| ✓ | Part | Type | ✓ | Part | Type | ✓ | Part | Type |
|---|------|------|---|------|------|---|------|------|
| ☐ | IC1 | PIC16LF872-I/SO (Programmed w/ Remote code) | ☐ | R3 | 82 ohm array | ☐ | C1 | 22pf |
| ☐ | R1 | 10K | ☐ | R4 | 82 ohm array | ☐ | C2 | .1uF |
| ☐ | R2 | 10K | ☐ | R5 | 82 ohm array | ☐ | SW2 | Miniature push button |

After the surface mount components are soldered onto the board, install and solder the remaining through-hole components.

When installing the LEDs (Dxx) be very careful that they are installed with the proper orientation, see Figure 9. The shorter lead corresponding to the cathode, is oriented closest to the dashed line on the LED silk screen. So be careful, once the leads are soldered and clipped, you can not visually determine which lead is which!

The power switch (SW1) is installed and secured to the PC board using a lock washer and nut.

### 3.2.2 Remote unit through-hole components:

| ✓ | Part | Type | ✓ | Part | Type | ✓ | Part | Type |
|---|------|------|---|------|------|---|------|------|
| ☐ | D1 | T1 red LED | ☐ | D5 | T1 red LED | ☐ | D9 | T1 yellow LED |
| ☐ | D2 | T1 red LED | ☐ | D6 | T1 red LED | ☐ | D10 | T1 yellow LED |
| ☐ | D3 | T1 red LED | ☐ | D7 | T1 red LED | ☐ | J1 | 8-pin modular jack |
| ☐ | D4 | T1 red LED | ☐ | D8 | T1 red LED | ☐ | SW1 | Push button switch (Push o/off) |

As shown in the schematic, connect to the battery holder's positive (red) lead to one side of the switch. The other side of the switch is connected to a PC board solder pad labeled J2 with a short piece of hook-up wire. This wire is generated by cutting about a one inch piece off the battery holder's red lead. The negative side (black) of the battery holder is connected to a PC board solder pad labeled J3.

Secure the battery holder to the bottom of the case using two pairs of self-adhesive Velcro pads.  This will keep the holder and batteries from rattling around and make it easy to replace the batteries in the future.

### 3.2.3 Remote unit battery holder and switch wiring:

| ✓ | | ✓ | |
|---|---|---|---|
| ☐ | Jumper from SW1 to pad J2 | ☐ | Black battery holder lead to pad J3 |
| ☐ | Red battery holder lead to other side of SW1 (J1) | ☐ | Install Velcro to battery holder and case |

### 3.3 Testing the Tester

The initial checkout and testing procedure of the master unit and remote unit are similar.Before inserting the batteries and applying power carefully double check the component placements and solder joints. Look for solder shorts; especially between the fine-pitched leads of the surface mounted components.

Install two batteries into the battery holder, being careful that they are properly orientated.  Apply power to the unit for a couple of seconds using the momentary TEST switch of the master unit or push ON/OFF switch of the remote unit.  When power is first applied, the software of each unit performs a quick test of each LED.  Each LED is briefly turned on while the other LEDs are off.  Since the sequence of LED testing is relatively quick, you may need to repeat the power-on test several times to verify that all the LEDs are functioning correctly.

If none of the LEDs light up, the microcontroller itself might not be running.  Recheck power to the microcontroller and solder connections, especially around IC1, C1, R1 and R2.  If one or so LEDs fails to light, recheck the solder connections associated with the LED, it's associated resistor and associated pin of IC1.  An LED will also fail to light if its anode and cathode leads are reversed.  If several LEDs light up simultaneously, check for solder shorts around the associated resistor and pins of IC1.

Next, exercise the PROG (SW2) switch on each unit.  Using the procedures outlined in Figure10, Figure11, Figure12 and Figure 13 change the reference map, mode, etc. of each unit.

Finally, use a cable that is known to be good or you at least have high confidence in.  An 8-wire straight through cable is ideal.  Select the reference map on the master unit appropriate for the test cable.  Turn the remote unit on and place it in remote ID mode.  Connect the cable between the master and remote units.  Press and hold the master's TEST button and let the master proceed through the test phase as outlined in figure 10.  At the end of the testing phase the PASS LED should turn on.  If the test fails, use the information during the readout phase to help locate the problem (see figure 11).  Check for solder problems around the jack (J1) and the associated pins of IC1. The supposedly good cable might be suspected as well.

After everything checks out, screw the boards onto the cases and the Mapper8 is ready for action!

## 4. Summary and Tips

Other cable types such as coax and others can be tested by building short cable adapters to the 8-wire modular jack.  Cables with fewer wires can be accommodated by building an adapter connecting to only a subset of the modular jack.  Using the adapters, test a likely-good cable and study the connectivity results. If the connectivity is correct, then store the map into one of the programmable reference maps (map1, map2, map3 or map4).  The Mapper8 is now ready to rapidly test more of these new cables.

If you are testing a cable run that is made up of several individual cables chained or patched together, be sure to test the intermediary cables as well as the complete cable run.  Two wrong cables in the chain could make the whole look correct!

Keep in mind that the Mapper8 does not test all the cable parameters that might be important to some applications.  For example, networking cables need to have specific impedance, cross-talk, length, etc. characteristics.  By using the appropriate raw cable and carefully following recommended cabling practices these requirements can be easily met.  That said, the Mapper8 is a versatile and inexpensive addition to your toolbox!
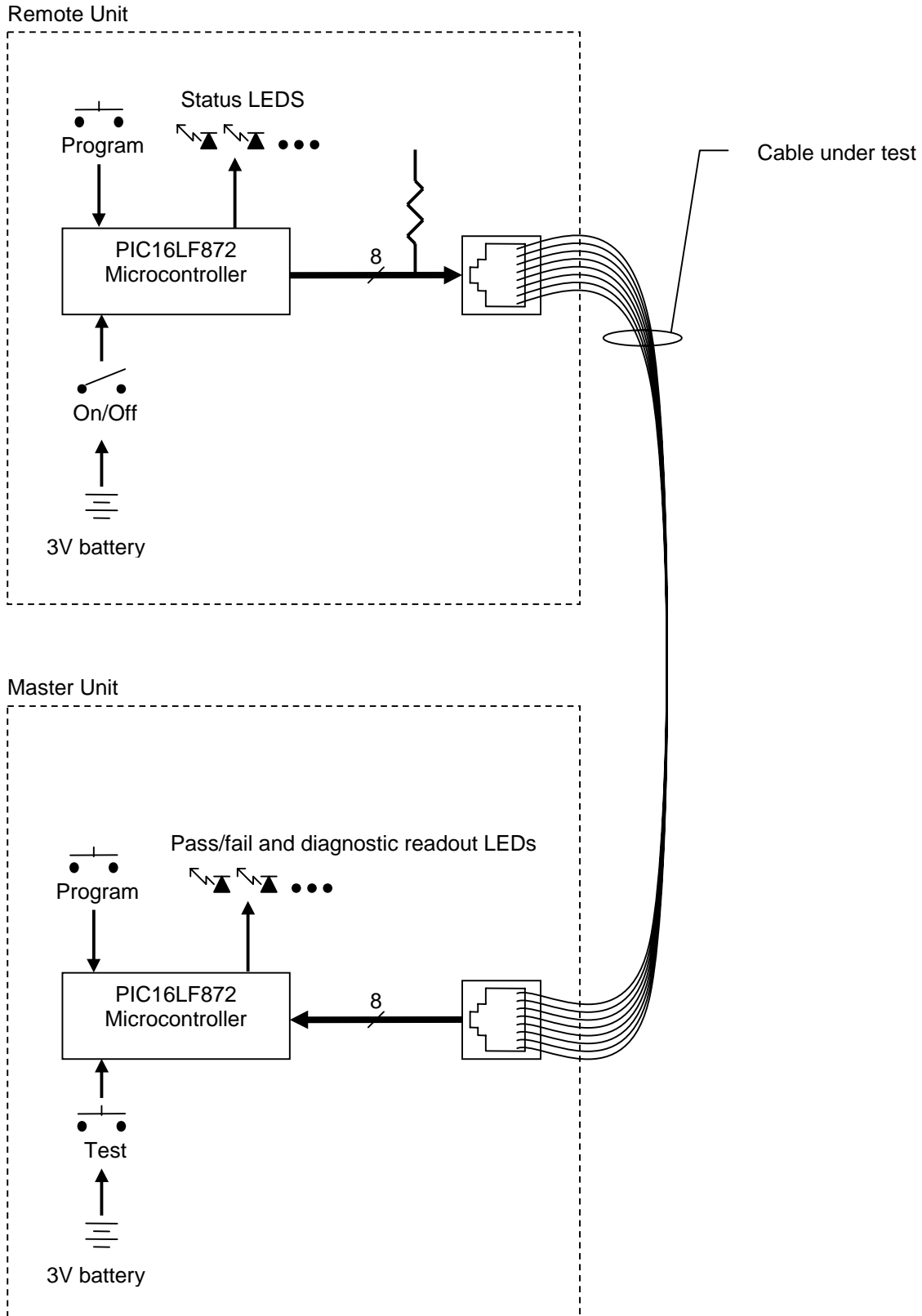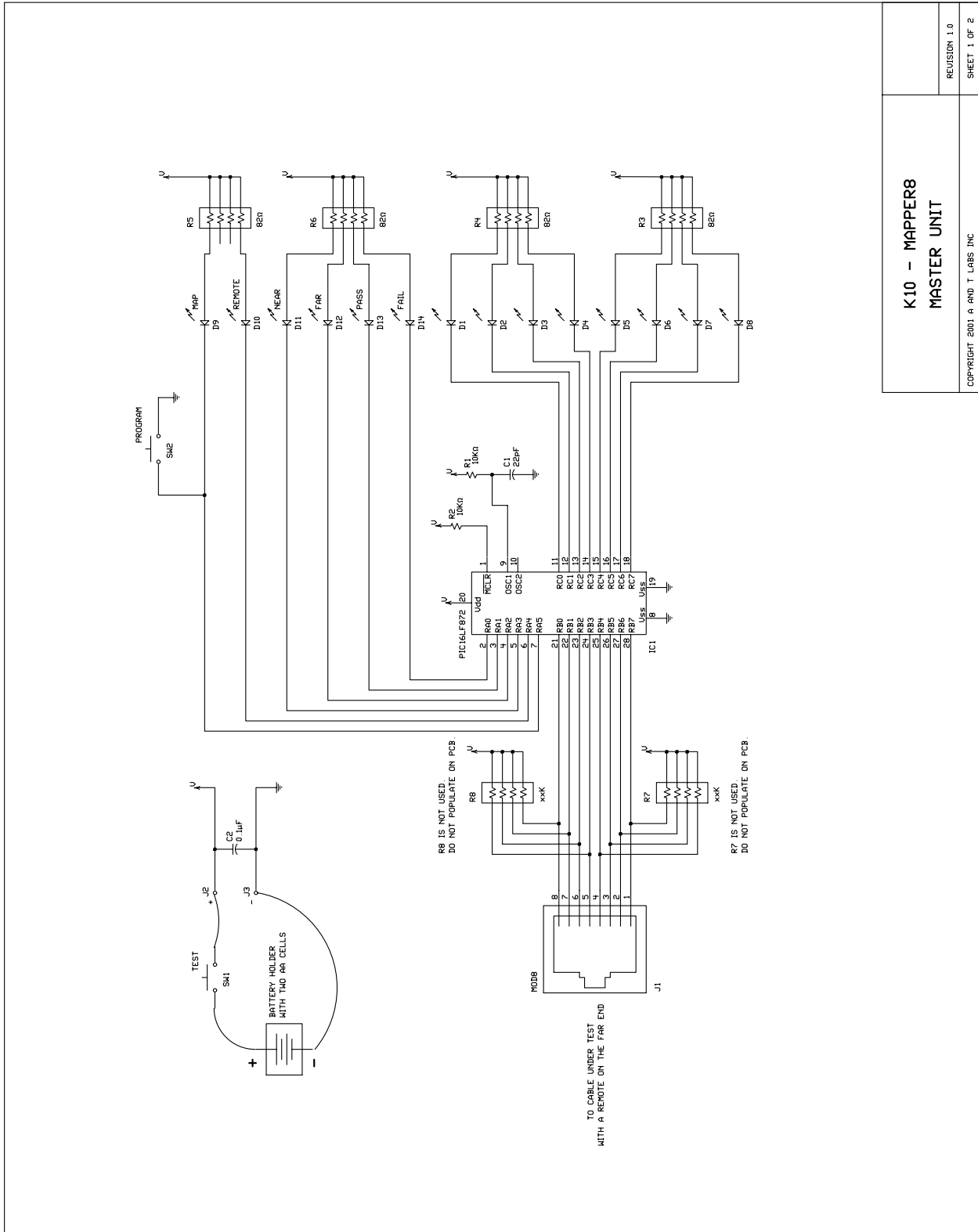
Remote Unit

Program

Status LEDS

PIC16LF872
Microcontroller

8

On/Off

3V battery

Cable under test

Master Unit

Program

Pass/fail and diagnostic readout LEDs

PIC16LF872
Microcontroller

8

Test

3V battery

Figure1 - Block Diagram of the Master and Remote Units

Figure 2 – Master Unit Schematic

Figure 3 – Remote Unit Schematic

Sync 0
Sync 1
Remote ID2
Remote ID1
Remote ID0
Wire ID7
Wire ID6
Wire ID5
Wire ID4
Wire ID3
Wire ID2
Wire ID1
Wire ID0

Used by master
to synchronize
bit timing

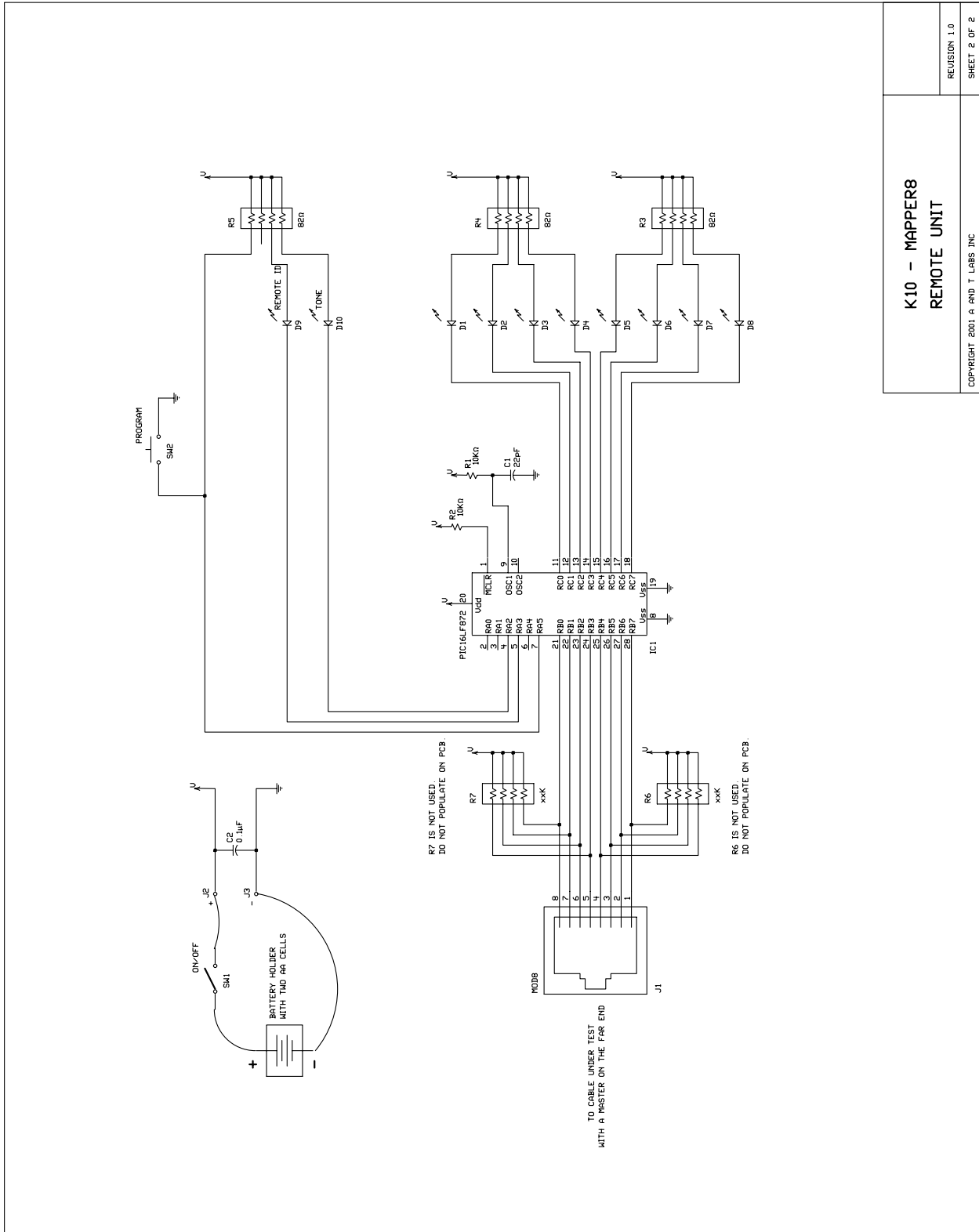Uniquely identifies
the remote

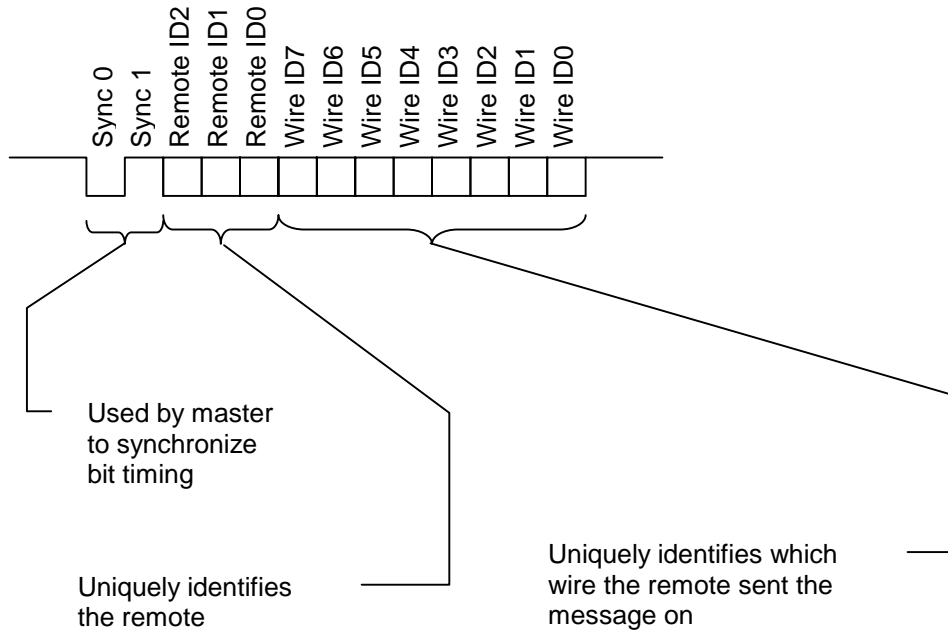Uniquely identifies which
wire the remote sent the
message on

Figure 4 - Message Detail. Format of the messages sent by the remote unit back to the master unit.
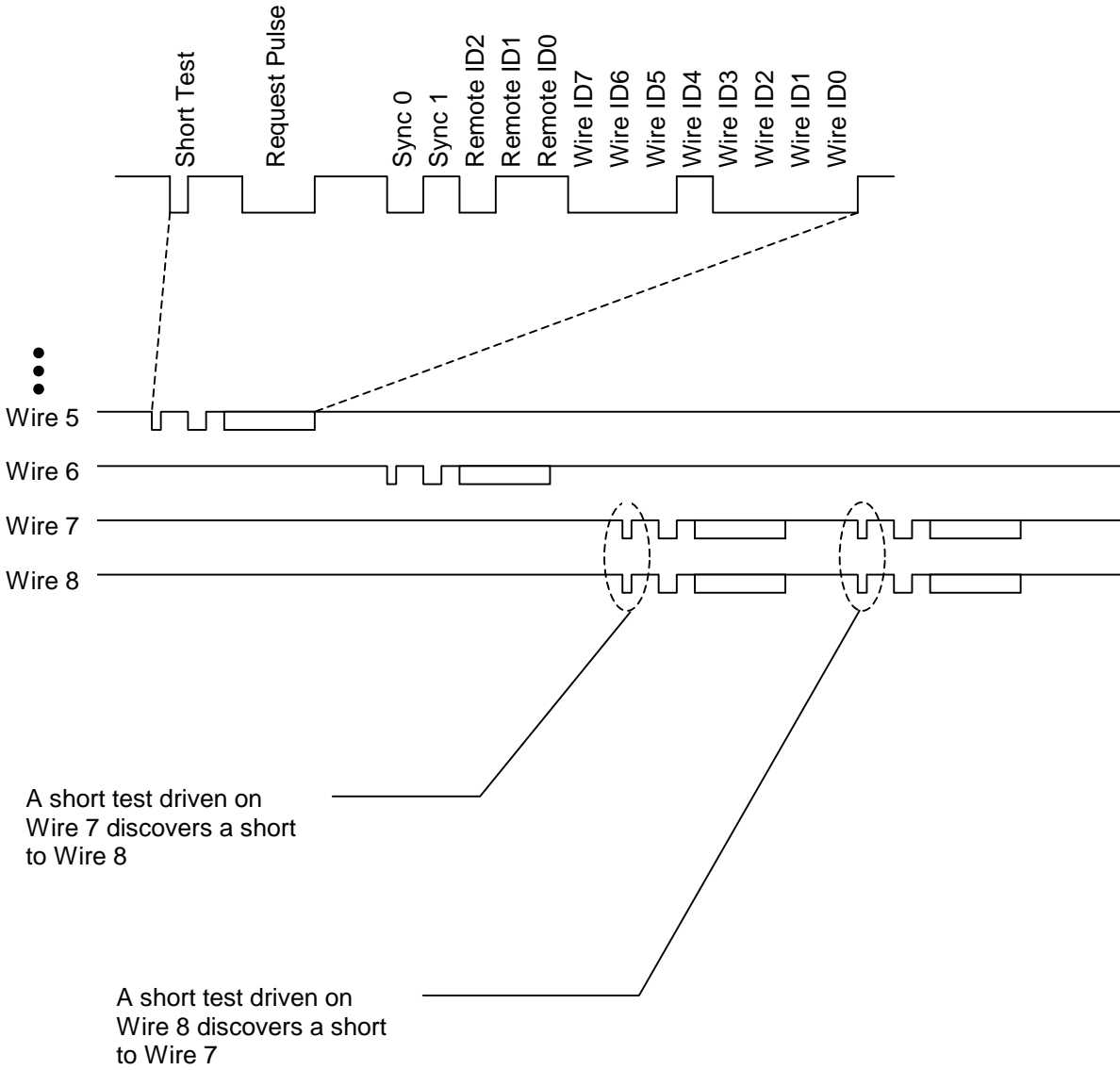
Figure 5 - Message Protocol Example
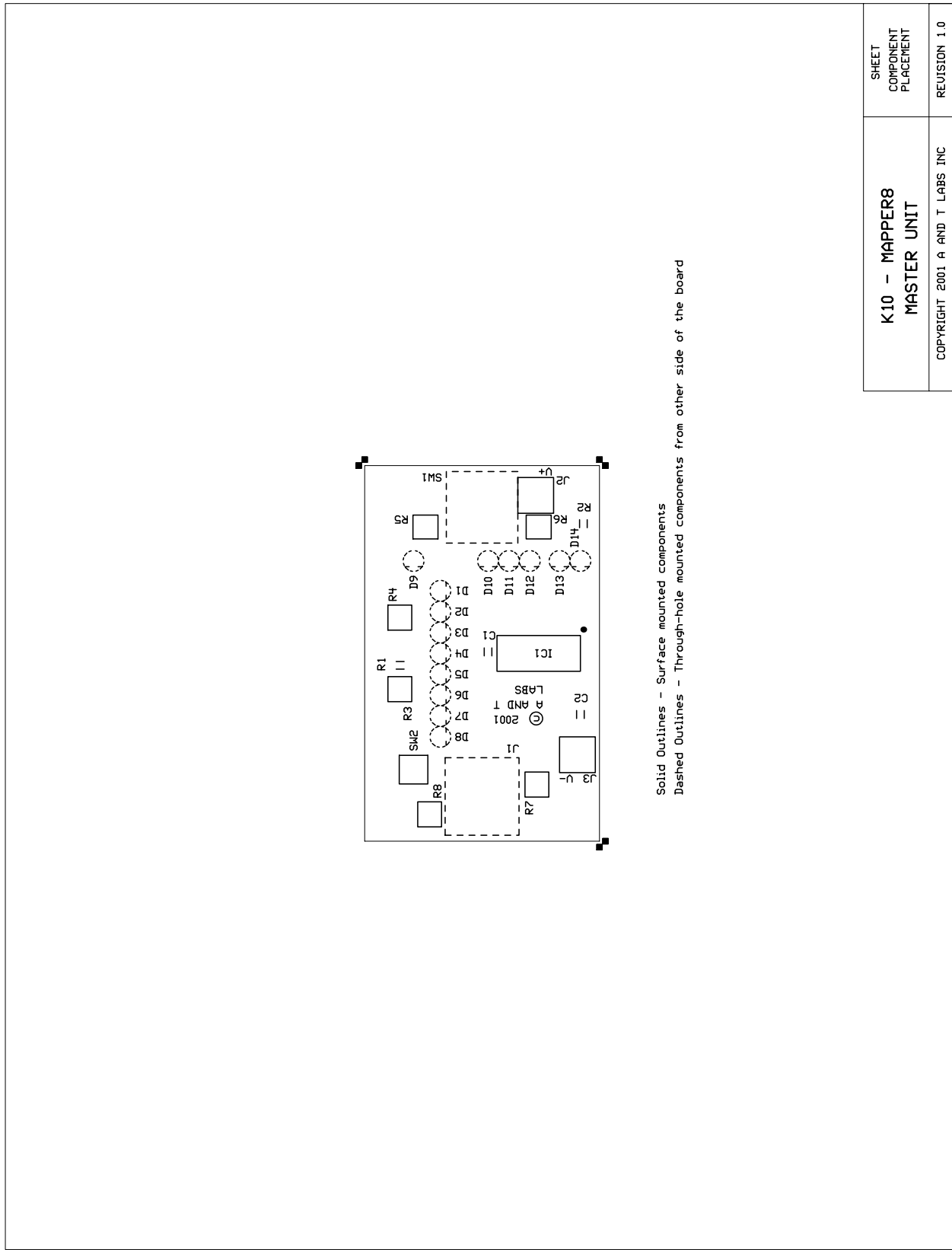
A short test driven on
Wire 7 discovers a short
to Wire 8

A short test driven on
Wire 8 discovers a short
to Wire 7

Figure 6 – Master Unit  Component Placement
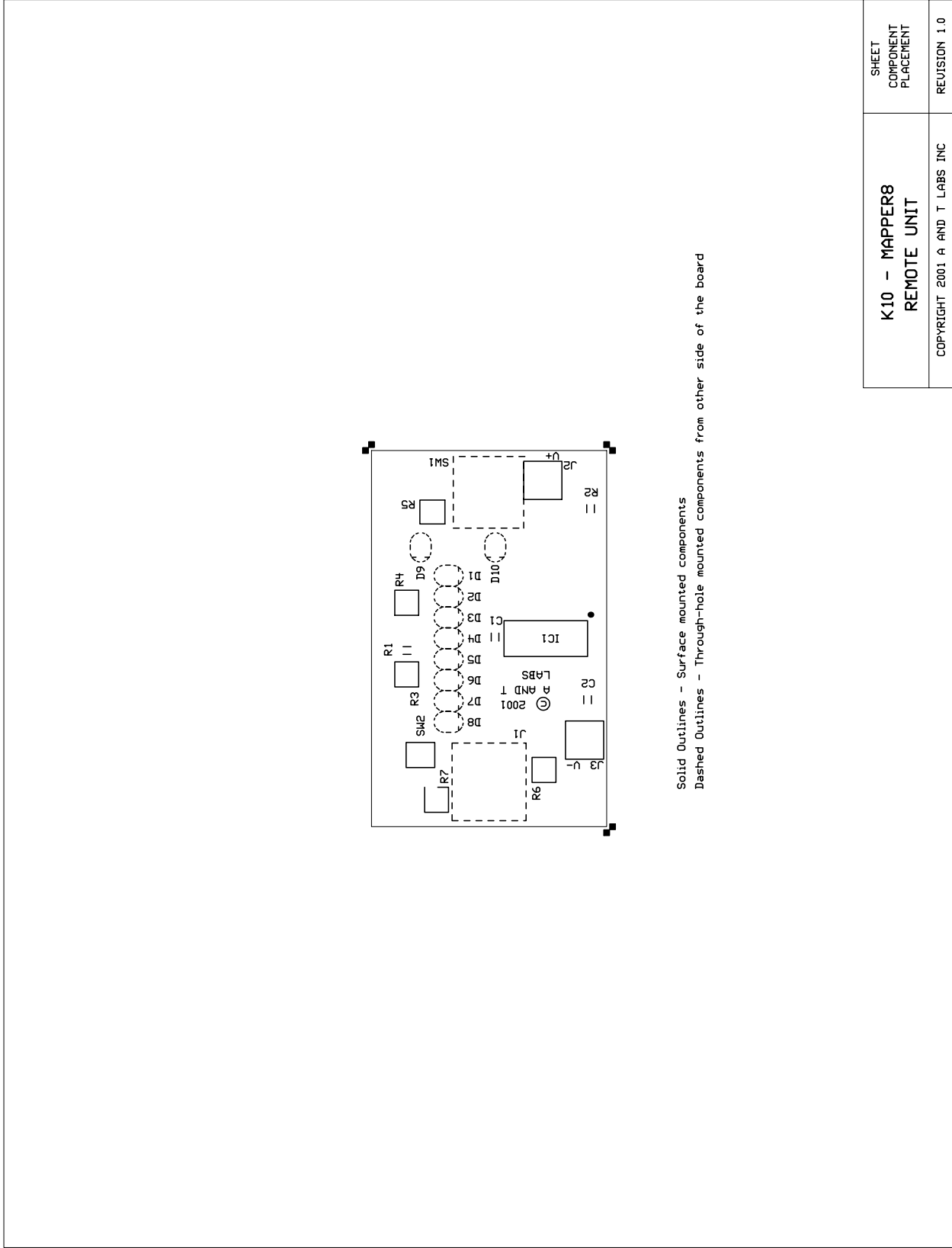
Figure 7 – Remote Unit Component Placement

Original switch                    Leads bent and installed
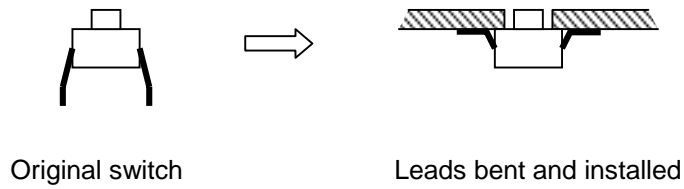
Figure 8 - Reforming Switch SW2 Leads.  Carefully bend over the four leads such that SW2 can be surfaced mounted to the "inside" of the PCB.

Schematic Symbol                LED Device                PCB Placement Symbol

anode                                              anode

cathode

anode

cathode                                        cathode

Flat side of the body and/or short lead indicates the cathode

Figure 9 - LED Lead Identification.  The flat side of the LED body and/or the short lead indicates the cathode.

Press and hold
the TEST button
to start testing

A display test is performed by rapidly
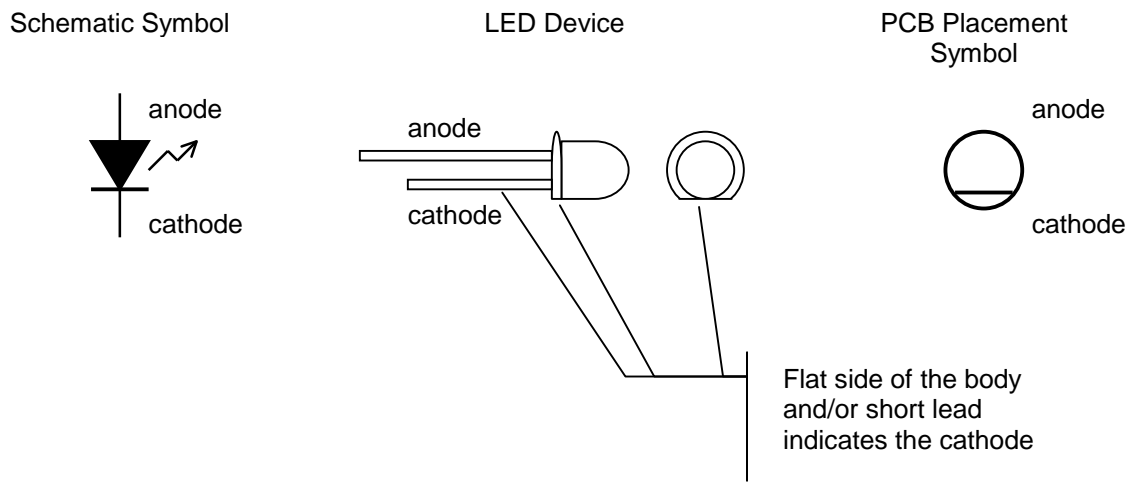lighting each LED in sequence

The currently selected map is displayed
indicating which connectivity map will be
assumed.  In this example, the CROSS
map is selected.

At this point only, the selected map can be
changed.  Press and hold the PROG button
until the MAP LED blinks once.  Release
the PROG button.  Each push of the PROG
button will select the next successive map.
Release the TEST button when the desired
map is selected.

| Map | STRAIGHT | CROSS | REVERSE | MAP 4 | MAP 3 | MAP 2 | MAP 1 | Prog |

8 7 6 5 4 3 2 1

○ Remote
○ Near
○ Far

Each successive wire LED (1 through
8) is lit, indicating scans for messages
from the remote are in progress.

The measured connectivity is compared to
the selected map and either the PASS or
FAIL LED is lit and remains lit.
In this example, the comparison passed.

| Map | STRAIGHT | CROSS | REVERSE | MAP 4 | MAP 3 | MAP 2 | MAP 1 | Prog |

8 7 6 5 4 3 2 1

○ Remote
○ Near
○ Far

● Pass
○ Fail

If the TEST button continues to be
held down, the Master Unit will
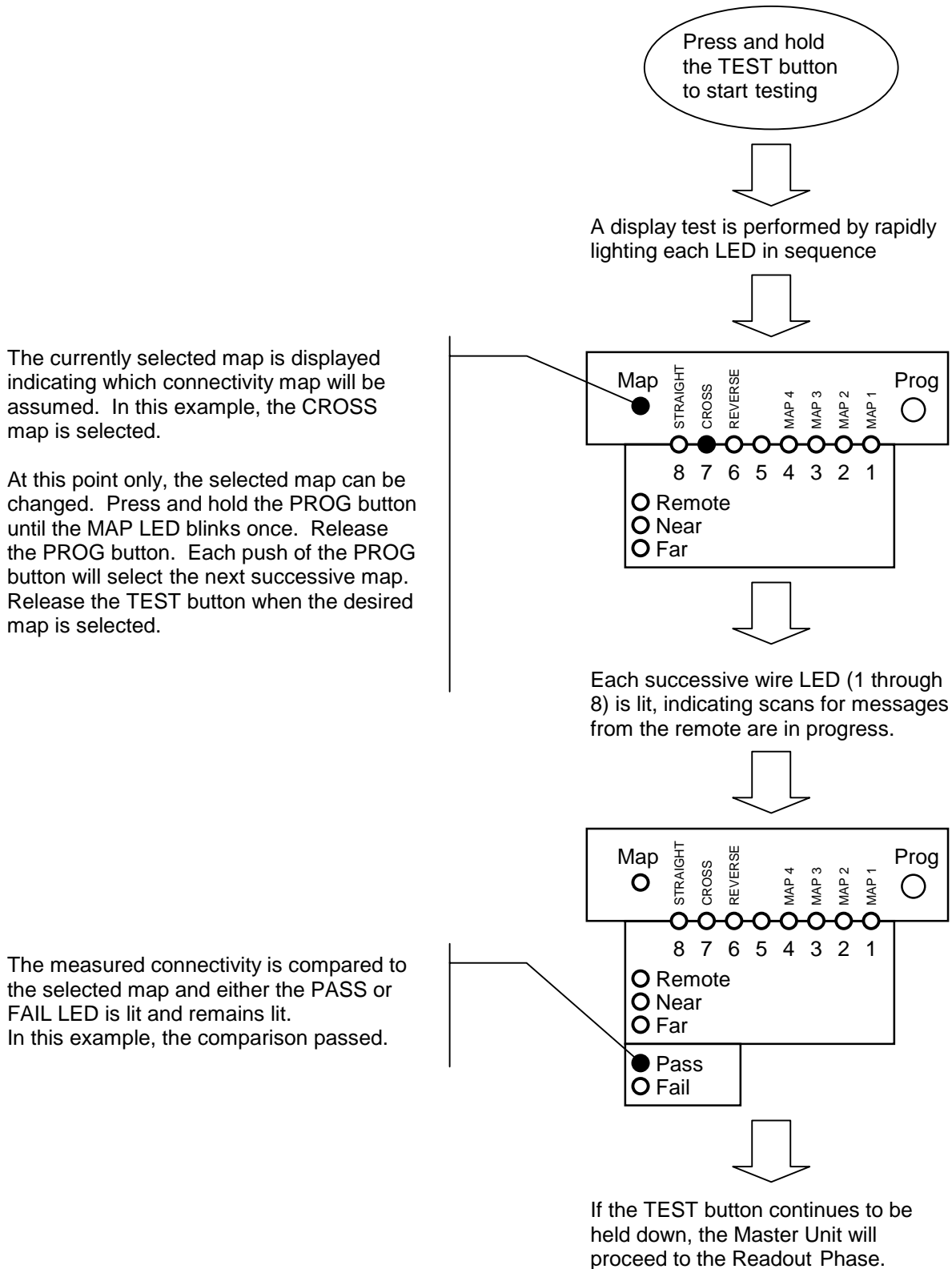proceed to the Readout Phase.

Figure 10 - Master Unit Test Phase.  The progression of the master unit's test phase ends with an easy to
read pass or fail indication.

AMK10a

If the TEST button continues to be held down, the readout phase is entered from the test phase

The currently selected map is again displayed indicating which connectivity map will be assumed.
In this example, the CROSS map is selected.

At this point only, the newly measured connectivity map can be stored into the Master Unit, if the currently selected map is one of MAPS 1-4. Press and hold the PROG button until the MAP LED begins to blink. The measured map is now stored in the selected map.

The ID of the Remote Unit is displayed. In this example, the messages were received from remote with ID 5.

The measured connectivity map is displayed by showing pairs of near-to-far wire number mappings. Wire numbers relative to the Master Unit are near and numbers relative to the Remote are far.

A total of 8 mapping pairs is shown in order of near wire 1 through near wire 8.

In this example, the first of the 8 mapping pairs is shown: near wire 1 is connected to far wire 8.

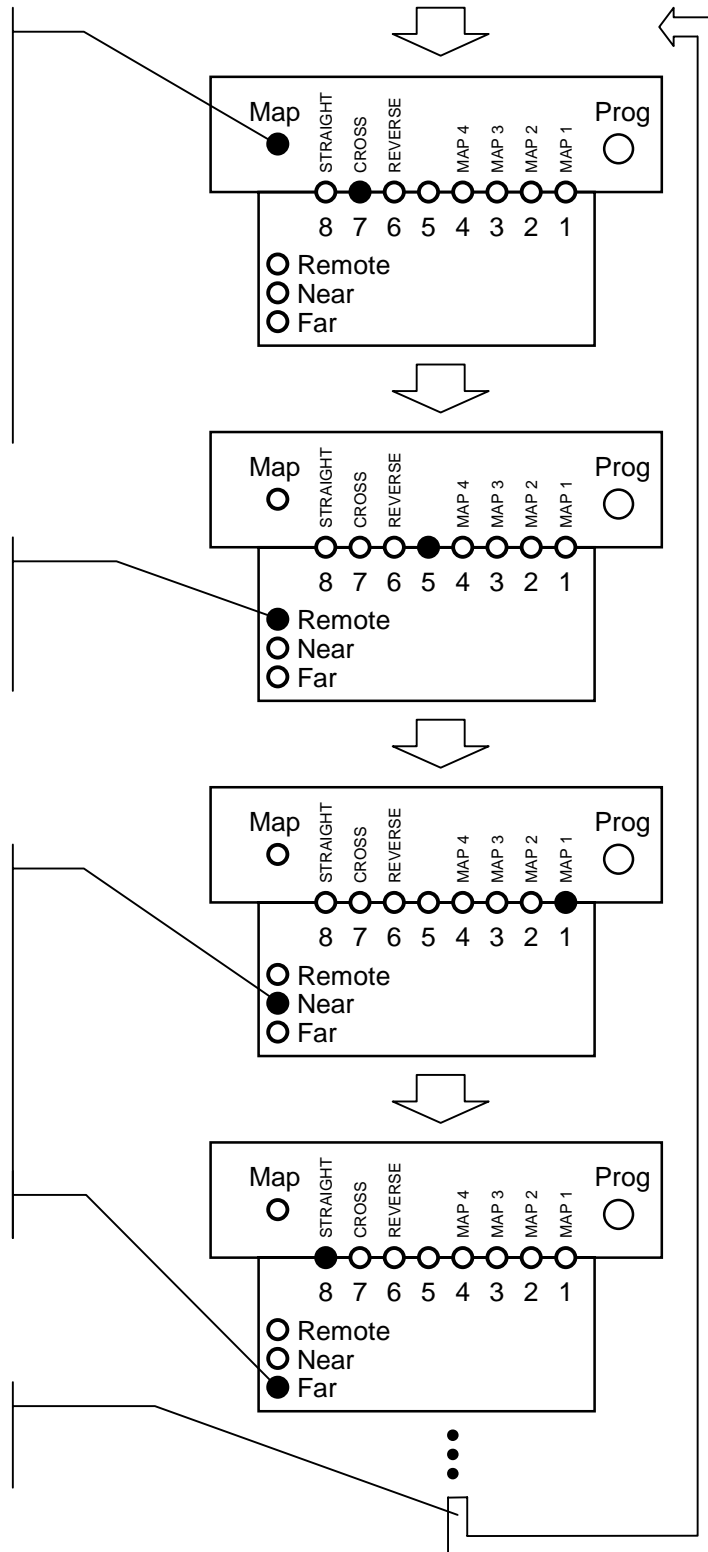The readout phase repeats until the TEST button is released.

Figure 11 - Master Unit Readout Phase. The master unit's readout phase shows the complete detail of the connectivity between the master and the remote.

AMK10a

The unit is in remote mode. The blinking numbered LED indicates the remote's ID. In this example, the ID is 5.

To change the remote's ID: Press and hold the PROG button for ~15 seconds until the numbered LED stops its blinking. Release the PROG button. Each push of the PROG button will select the next successive ID. Turn the remote unit OFF then back ON when the desired map is selected.
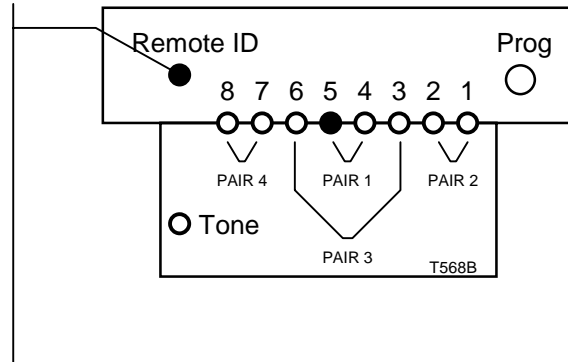
Figure 12 - Remote Unit in Remote Mode. In this mode, identifying messages are sent down each wire to be received by a master unit.

The unit is in tone mode. The blinking numbered LEDs indicate on which pair the tone is transmitted. In this example, the tone is on Pair 3 (wire 3 and 6).

To change the tone pair: Each push of the PROG button will select the next successive pair.
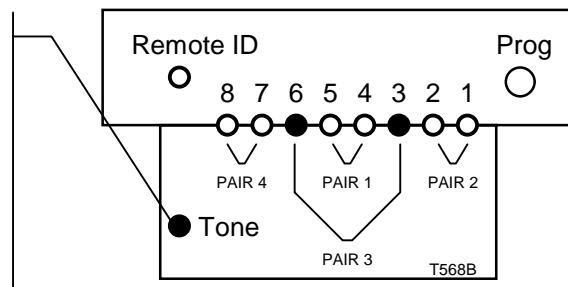
Figure 13 - Remote Unit in Tone Mode. In this mode, a tone is continuously transmitted down the selected pair of wires