1. Introduction to MAXREFDES101

MAXREFDES101 is a unique evaluation and development platform in a wrist-worn wearable form factor that demonstrates the functions of a wide range of Maxim’s products for health-sensing applications. It integrates a PPG analog-front-end (AFE) sensor (MAX86141), a biopotential AFE (MAX30001), a human body temperature sensor (MAX30205), a host microcontroller (MAX32630), a power-management IC (MAX20303), along with a sensor hub (MAX32664). MAX32664 is a sensor hub family with embedded firmware and algorithms for wearables. It seamlessly enables customer desired sensor functionality, including communication with Maxim’s optical sensor solutions and delivering raw or calculated data to the outside world. This is achieved while keeping overall system power consumption in check.

MAXREFDES101 system diagram is shown below:
In this design, MAX32630 is used as the host controller. A display and three pushbuttons are connected to the microcontroller. The sensor hub microcontroller is the MAX32664 which has the pre-programmed wearable heart rate and SpO2 algorithms\(^1\). The MAX86141 optical AFE sensor and the KX-122 accelerometer are connected to the sensor hub.

MAXREFDES101 hardware and software reference designs are available for Maxim customers on Maxim’s product webpage and on the respective mbed OS webpages.

The protocol for communicating with the MAX32664 sensor hub is defined in the MAX32664 User Guide.

\(^1\) MAX32664 has several variants. Please contact Maxim sales to find the part that best suits your algorithm needs.
2. MAXREFDES101-based watch reference design

MAXREFDES101 EV-Kit software comes complete with the following:
- MAX32664 firmware with embedded health-sensing algorithm (Wrist Heart Rate Monitor (WHRM))
- MAX32630 host firmware, which provides communication between MAX32664 and the Windows or Android visualization/debugging application
- A Windows or Android GUI application for evaluation and debugging

MAXREFDES101 reference design can also be used by our customers to design their own system, which may require a different set of software. An example design can be as follows:

In this case, MAX32664 is now fully controlled by the embedded microcontroller. A typical system diagram for a MAXREFDES101-based watch design will then look like the following:
This reference design software includes:
- Sample host (microcontroller) software
- MAX32664 firmware algorithm (WHRM) binary

3. Sample host (microcontroller) software
The sample host software consists of three modules:
- Watch GUI & controls
- SH Communications
- Algorithm Report using SH configured with MAX8614X

![Host Software](image)

*Figure 5: Sample host software modules*

### 3.1. Hardware abstraction in sample host software

The sample host controller software is layered to provide hardware abstraction. This layered approach will allow most of the software modules to be cut and pasted into our customer’s target platform. Hardware dependent modules may require modifications to match your target microcontroller.

The sample host controller software is divided up into three layers:

1) The hardware libraries, which are the microcontroller specific code to support the hardware devices such as interrupts, I2C, and GPIOs.

2) The sensor hub communications layer, which include procedures to read/write data from/to the sensor hub, change the sensor hub mode to either application mode or bootloader mode, and to initialize the MAX32664 sensor hub.

3) The application layer, which has procedures for the user interface interactions to the display and pushbutton and procedures to stream processed data from the sensor hub.
**Figure 6: Sample host software layered design**
3.2. Sample host software flowchart

We utilize MAXREFDES101 hardware (watch form factor) to demonstrate our sample software. Below is the flowchart:

![Sample host software flowchart](Figure 7: Sample host software flowchart)
4. Data Flow Diagram
The drawing below is the Data Flow Diagram for the displaying of the data for the host software.

Figure 8: Sample host software layered design
5. **Mbed code repository**

The source code for the MAX32630 is located at:

https://os.mbed.com/teams/MaximIntegrated/code/Host_Software_MAX32664GWEB_HR_wrist/

It is compatible w/ the .msbl file listed in the Compatibility section.

6. **Display and Button Interface**

6.1. **Display**

The watch display will display the user heart rate when turned on and strapped to the user wrist:

![Heart Rate displayed](image)

*Figure 9: Heart Rate displayed*

6.2. **Software Reset Button**

When the upper right button is pressed twice, the software is reinitialized:

![Top Right button pressed once.](image)

*Figure 10: Top Right button pressed once.*

6.3. **Off Buttons**

When the upper left button and the upper right button are both pressed for about thirteen seconds, the MAXREFDES101 will be turned off.
7. Serial Interface

Plug in the USB-C cable to the MAXREFDES101. Use a serial terminal application such as Terra Term to connect to the USB Serial Device, 115200 baud, 8N1N.

7.1. Serial terminal data stream

After the serial terminal application is correctly connected, the heart rate, confidence, and algorithm status are displayed:

```
hr:68.0 conf:99 status:0
hr:68.0 conf:99 status:0
hr:68.0 conf:99 status:0
hr:68.0 conf:99 status:0
hr:68.0 conf:99 status:0
hr:68.0 conf:99 status:0
hr:68.0 conf:99 status:0
hr:68.0 conf:99 status:0
hr:68.0 conf:99 status:0
hr:68.0 conf:99 status:0
hr:68.0 conf:99 status:0
```

**Figure 12:** Serial terminal displays heart rate, confidence and status

7.2. Stopping the data stream

Enter the command “stop” or cut and paste “stop” into the terminal window and press <Enter>.

```
hr:67.1 conf:99 status:0
hr:67.1 conf:99 status:0
hr:67.1 conf:99 status:0
hr:67.1 conf:99 status:0
hr:67.1 conf:99 status:0
hr:67.1 conf:99 status:0
hr:67.1 conf:99 status:0
hr:67.1 conf:99 status:0
hr:67.1 conf:99 status:0
hr:67.1 conf:99 status:0
```

**Figure 133:** Serial terminal stopped

7.3. Enabling visibility of serial inputs

In the terminal window, type “set_host_echomode 1” and then press <Enter>.

```
hr:71.5 conf:99 status:0
hr:71.5 conf:99 status:0
hr:71.5 conf:99 status:0
hr:71.5 conf:99 status:0
hr:71.5 conf:99 status:0
hr:71.5 conf:99 status:0
hr:71.5 conf:99 status:0
hr:71.5 conf:99 status:0
hr:71.5 conf:99 status:0
```

**Figure 144:** Serial terminal echomode set to 1

7.4. Read a register from the MAX86141
In the terminal window, type “get_reg ppgsensor 0xff” and then press <Enter>.

Figure 153: Register read command

### 7.5. Enable the data stream

In the terminal window, type “enable_measurement whrm” and then press <Enter>.

Figure 164: Streaming data enabled

### 7.6. Serial Command List and Description

- **stop**
  - stops the serial data streaming

- **enable_measurement whrm**
  - starts the serial data streaming

- **set_host_echomode [visible]**
  - visible=1: Enables the visibility of the user typing
  - visible=0: Disables the visibility of the user typing

- **get_hub_fwversion**
  - Displays the sensor hub version.

- **get_host_opmode**
  - Displays whether the mode is APPMODE or BOOTLOADERMODE

- **set_host_opmode 1**
  - Changes the mode to bootloader mode

- **get_reg ppgsensor [reg]**
  - reg = [0..0xff]: address of the register to display

- **set_reg ppgsensor [reg] [val]**
  - val=[0..0xff]: set the register to this value.
  - reg=[0..0xff]: set the value of this register to [val].

- **get_cfg whrm aecenable**
  - 1: AEC is enabled
  - 0: AEC is disabled

- **set_cfg whrm aecenable [enable]**
enable=1: enable AEC
disable=0: disable AEC

getCfg whrm scdenable
1: SCD is enabled
0: SCD is disabled

getCfg whrm scdadjperiod
period of time to adjust the target PD, 16-bit unsigned, seconds

setCfg whrm scdadjperiod [period]
period = [0..0xFFFF]: Period of time to adjust the target PD, 16-bit unsigned, seconds

getCfg whrm scddebouncewin
Displays the debounce for SCD, which is number of consecutive results of skin contact status to be reflected in the final output.

setCfg whrm scddebouncewin [dbwin]
dbwin=[0..0xFFFF]: The motion magnitude threshold for Motion Detector LSB=0.1g, 16-bit unsigned.

getCfg whrm motionthreshold
Displays the motion magnitude threshold for Motion Detector LSB=0.1g, 16-bit unsigned.

setCfg whrm motionthreshold [threshold]
threshold=[0..0xFFFF]: The motion magnitude threshold for Motion Detector, LSB=0.1g, 16-bit unsigned.

getCfg whrm minpdcurrent
Displays the minimum PD current, LSB=0.1mA, 16-bit unsigned.

setCfg whrm minpdcurrent [mincurr]
mincurr=[0..0xFFFF]: The minimum PD current, LSB=0.1mA, 16-bit unsigned.

getCfg whrm pdconfig
Displays the minimum PD configuration: 1=PD1, 2=PD2, 3=Both PD1&PD2

setCfg whrm pdconfig [pdcfg]
pdcfg: The PD configuration: 1=PD1, 2=PD2, 3=Both PD1&PD2.

8. Compatibility Matrix

<table>
<thead>
<tr>
<th>Mbed MAX32664GWEB Sample Host Release</th>
<th>Host Notes</th>
<th>.msbl application algorithm/driver file</th>
</tr>
</thead>
<tbody>
<tr>
<td>3/25/19 Mbed Files at revision 3:b8989dab0f88 [<a href="https://os.mbed.com/teams/MaximIntegrated/code/Host_Software">https://os.mbed.com/teams/MaximIntegrated/code/Host_Software</a> MAX32664GWEB_HR_EXTENDED](<a href="https://os.mbed.com/teams/MaximIntegrated/code/Host_Software">https://os.mbed.com/teams/MaximIntegrated/code/Host_Software</a> MAX32664GWEB_HR_EXTENDED)</td>
<td>Additional features added to the Heart Rate display program: serial commands, msbl flashing code,</td>
<td>9/1/18 MAX32664WGEZ_SmartSensor_HSP2_v1.8.3.msbl</td>
</tr>
</tbody>
</table>