Direction Android to MCU	Cmd Word	Code addr	Notes
	0x1203	0x5312	Read 5 bytes from radio and send them to Android
	0x120A	0x51DF	Read AF results and send 5 bytes of result data to Android
	0x120B	0x5349	Read a byte from radio device and send it to Android
	0x120E	0x52EA	Read RDS (?) data from radio device and send to Android – not supported for radio cfg 0x01 (TEF6624)
	0x1402	0x539A	Stub – do nothing
	0x1405	0x53A2	Read 1 byte from 0xC0 (TEF6624 radio) and 1 from I2C 0x86 (video switch?) and send those bytes out SPI
	0x1510	0x5433	Send 1 byte from 0x04C8 to Android
	0x1530	0x5483	Send version string (at 0x02AB in ext mem) out SPI
	0x1531	0x548D	Sends version string date stamp out SPI
	0x1532	0x5497	Sends version string time stamp out SPI
	0x15FF	0x554C	Send a big block of 512 bytes at 0x02E5 out SPI
	0x9000	0x4EB4	Read Volume_Gain and Input_Gain values from Android, save at 0x02BB and if radio is on (bit 0x25.5 is TRUE) then update sound processor.
	0x9001	0x4ED6	Read Volume_Gain and Input_Gain values from Android, save at 0x02BB and if radio is on (bit 0x25.5 is TRUE) then update sound processor.
	0x9002	0x4F41	Read Volume_Gain and Input_Gain values from Android, save at 0x02BB and if radio is on (bit 0x25.5 is TRUE) then update sound processor.
	0x9003	0x4F91	If bit 0x25.5 is TRUE (radio on) then set sound processor to full mute
	0x9004	0x4FB5	Read Volume_Gain and Input_Gain values from Android, save at 0x02BB and if radio is on (bit 0x25.5 is TRUE) then update sound processor.
	0x9005	0x4F9D	IF bit 0x25.5 is FALSE, then enables main loop processing of state 0x08 (turn on radio?)
	0x9006	0x4FA9	IF bit 0x25.5 is TRUE then turn off AMP-CON output, mute all sound gains and shut off radio
	0x9007	0x4FD7	IF bit 0x25.5 is TRUE then set bit 0x22.1, mute all sound and then send data to CANbus
	0x9008	0x4FE5	IF bit 0x25.5 is TRUE then clear bit 0x22.1, unmute all sound and then send data to CANbus
	0x900A	0x5004	Write 4 bytes from Android to 0x02BB and if USB pwr is ON then write data at 0x02BB to video switch ??
	0x9020	0x4DA9	DVD control? Sends to CANBus
	0x9021	0x4DCB	DVD control? Sends to CANBus
	0x9022	0x4E1E	DVD control? Sends to CANBus
	0x9023	0x4E57	DVD control? Sends to CANBus
	0x9024	0x4DE9	DVD control? Sends to CANBus
	0x9030	0x4D9C	Set bit 0x24.6 TRUE (prevents shutdown when Ignition goes OFF)
	0x9031	0x4DA4	Set bit 0x24.6 FALSE (allows shutdown when Ignition is OFF)
	0x9100	0x5026	Setup audio MUX to select DVD audio

0x9101	0x502E	Setup audio MUX to unselect DVD audio and set bit 0x29.3 (DVD special pause flag?)
0x9102	0x5036	Set DVD control pin P5.5 FALSE
0x9103	0x503E	Set DVD control pin P5.5 TRUE
0x9200	0x50A8	Read 2 bytes from Android and change radio station
0x9201	0x527C	Update radio with current command data in memory
0x9202	0x52B3	Force mono mode
0x9204	0x5216	Tune radio to next station, stay muted
0x9205	0x9249	Release mute and tune to station
0x9206	0x5046	Set 0x25.0 TRUE, turn on AMP-CON output and radio; set 0x0019 to 0x00 or 0xFF
0x9207	0x5077	Set 0x25.0 FALSE, turn off AMP-CON output and radio; set 0x0019 to 0x00 or 0xFF
0x9209	0x5181	Sends the TEF6624 radio an AF Update command when ext mem 0x02EB is 0x01
0x920C	0x511C	Copy bit 0x27.5 to bit 0x26.1, and if TRUE then for some radio devices set up data and send it depending on main loop state 0x0050
0x9300	0x5380	Stub – do nothing
0x9301	0x5382	Stub – do nothing
0x9302	0x5384	Stub – do nothing
0x9400	0x538A	Set I/O Pin P4.1 TRUE (MUX select M1 -- select audio from back panel E-2 & E-7 instead of DVD)
0x9401	0x5392	Set I/O Pin P4.1 FALSE (deselect audio from back panel Aux In)
0x9403	0x53AA	Read 4 bytes from SPI into memory at 0x45 & if P3.1 (I2C SCL) is TRUE then send them to I2C address 0xC0 (TEF6624 radio)
0x9404	0x53B2	Read 3 bytes from SPI into memory at 0x45 & if P3.1 (I2C SCL) is TRUE then send them to I2C slave address 0x86 (video switch?)
0x9500	0x53BA	Updates front panel LED levels with 3 bytes from Android
0x9501	0x53F9	Read 1 byte from Android, write it to 0x04C9 and set 0x0020 to 0xC8
0x9502	0x5401	Read 1 byte from Android, write it to 0x04C8 and set 0x0020 to 0xC8
0x9507	0x540F	Read 2 bytes from Android, save them at 0x04C5, and set 0x0020 to 0xC8
0x9508	0x53C2	Reads 1 byte from Adroid, write it to 0x04D1, set 0x0020 to 0xC8, then set the 2 bytes at 0x0029 to 0x00C8
0x9509	0x5417	Read 2 bytes from Android, save them at 0x04D2, and set 0x0020 to 0xC8
0x950A	0x53F1	Read 1 byte from Android, write it to 0x04D4 and set 0x0020 to 0xC8
0x950B	0x53E0	Read 1 byte from Android and save at 0x04C7, set 0x0020 to 0xC8, write 4 constant bytes to 0x0021 depending on value of 0x04C7 and maybe do more
0x950F	0x5425	Read 150 bytes from Android and write it to 0x042D
0x9520	0x543D	Read system beep time from Android and write to 0x000E (system beep timer)

0x9523	0x5445	Read 4 bytes from Android & if 0x02E4 is zero then save those at 0x0025 as bit masks and set 0x02E4 to 0x01
0x9524	0x544D	Save byte from Android in 0x0159 -- either 0,1 or 2 (default to 2 if out of range)
0x9525	0x5479	Clear bit 0x23.2. Set FALSE on transition from shallow sleep state (0x01) to Android On (0x02)
0x9526	0x547E	Set bit 0x23.2. Set FALSE on transition from shallow sleep state (0x01) to Android On (0x02)
0x9527	0x4E9E	Set steering wheel keys ADC reading flag bit 0x23.7 TRUE. This is probably sent from Android to MCU during key programming to get the MCU to send Android the current ADC value
0x9528	0x4EA6	This is probably sent from Android to MCU during key programming to get the MCU to send Android the current CANbus key value
0x9529	0x4E98	Set 0x001F to 0x01 and then set steering wheel keys ADC reading flag 0x23.7 TRUE to signal that a new reading is available
0x9540	0x4EAE	Do nothing stub
0x9541	0x4EB1	Do nothing stub
0x95FD	0x559E	Write a block of 72 bytes from SPI into the lookup table of key action words in config memory at 0x0365
0x95FE	0x5565	Read a big block of 512 bytes from SPI and save at 0x02E5 then write that block to EEPROM
0xA000	0x54E2	Get 16 bits from SPI and if successful then initialize base of ext mem and enter bootloader (reflash?) with that value in R6:R7
0xA123	0x5531	Assert a software reset
0xA124	0x5528	Set ext mem 0x0010 to 0x01
0xB001	0x5544	Do nothing except wait for SPI handshake
0xB002	0x5546	Do nothing except wait for SPI handshake
0xC000	0x54AB	Read an arbitrary length block of bytes from SPI and then send them out the serial port to CANbus
0x0101	0xD6F1	Enter DVD state 0x04 & set bit 0x29.4 TRUE, resets counter to 0x0000, sets P0.3 FALSE and P4.7 TRUE Must be in DVD state 0x00, 0x03, or 0x06
0x0102	0xD486	Enter DVD state 0x05 & set bit 0x29.4 TRUE, resets counter to 0x0000, sets P0.3 TRUE and P4.7 FALSE. Must be in DVD state 0x00, 0x01 or 0x04
0x0103	0x56BF	Set special pause bit 0x29.3 TRUE
0x0104	0x56C3	Set special pause bit 0x29.3 FALSE
0x0105	0x56A7	Set bit 0x29.4 TRUE, reset counter to 0x0000 and set P0.3 TRUE and P4.7 FALSE Must be in DVD state 0x00, 0x01, 0x04, or 0x06
0x0201	0x56F8	Turn radio on, collect status bits and send them in a byte to Android
0x0202	0x5781	Calls 0xCECA to write to bit flags, write 0x00 to 0x4E5, mute all sound gains, write to I/O pin 4.5 (USB pwr). Also sets main loop state to 0xF6 (normal op state)

	0x0204	0x56E6	Does nothing if bit 0x24.6 is TRUE. Else calls 0xCECA to mute all if bit 0x24.0 is TRUE. State 0xF5 sends cmds 0x0F201 0x01 or 0x0F200 0x00 and then enters state 0x4D
	0x0208	0x57E1	Special front panel LED state where LEDs ramp up and down
	0x020B	0x5761	Mute all sound gains, turn off AMP-CON output & shut down radio if it is on
	0x0301	0x57C5	If 0x25.5 is TRUE (radio on) then mute everything
	0x0302	0x57D0	If 0x25.5 is TRUE (radio on) then unmute everything
	0x0401	0x57BD	Set bit 0x23.0 TRUE
	0x0402	0x57C1	Set bit 0x23.0 FALSE
	0x0421	0x57A9	Set 0x04D0 to 0x01 and then set 0x0020 to 0xC8
	0x0422	0x57B0	Set 0x04D0 to 0x00 and then set 0x0020 to 0xC8
	0x05XX	0x5638	Set ext mem 0x0051 to value XX
	0x0755	0x56E9	Enter main loop state 0x01 unconditionally
	0x0EFE	0x5814	If not main loop state 0x02 then call 0xBF45 to shut down peripherals and then enter state 0x02
	0x0EFF	0x57FE	Call 0xD556 to enter main loop state 0x01 with Android OFF if bit 4 of state value is FALSE, or else enter state 0x4D with Android ON
	0x0F01	0x56CB	Read state of DVD signals P0.0, P0.1 and P0.2 and send them in a byte to Android
	0x0F02	0x56D8	Read state of bits 0x22.6 (Ignition), 0x24.5, /0x27.1 and 0x21.1 and send them in a byte to Android
MCU to Android			
	0xF000	0xBE9A	Send 2 bytes of SWC analog key value from SWC-1
	0xF001	0xBEBD	Send 2 bytes of SWC analog key value from SWC-2
	0xF004	0xBED6	Send bit pattern in 0x75 to Android if it has changed
	0xF007	0x9E90	Send 5 bytes of TEF6624 RDS data to Android prefixed by 0xF007
	0xF008	0x9E39	Send 4 bytes from TEF6624 radio to Android prefixed by 0xF008
	0xF009		Sending RDS data to Android?
		0xA483	Send 5 bytes from TEF6624 radio (at 0x054C) to Android prefixed by 0xF009
		0xA7EA	Send 5 bytes from radio at int mem 0x2C to Android prefixed by 0xF009
	0xF00B	0xA4C3	Send 2 bytes at 0x04E5 to Android prefixed by 0xF00B
	0xF00C	0xA4F1	Send 8 bytes at 0x04E5 to Android prefixed by 0xF00C
	0xF014		Keypress action command (data byte = key action value)
		0xCE2B	Subroutine to send 0xF014 out SPI, followed by byte read from R2:R1 or 0x4C or 04D based on value at 0x03B2
		0x8FFD	Bit 0x28.4 is FALSE; calls 0xCE31
		0xA82A	When button press is short, key action data byte after 0xF014 is 0x14; calls 0xD025 and then beeps
		0xA8E1	Generic send key action byte to Android; calls 0xD025
		0xB1DB	Send key actions from CANbus to Android

	0x5C5F	Send 0xF014 out SPI, followed by byte read from R2:R1 or 0x4C or 04D based on value at 0x03B2; calls 0xCE2B
	0xC9A5	Send 0xF014 out SPI, followed by byte read from R2:R1 or 0x4C or 04D based on value at 0x03B2; calls 0xCE2B
	0x61C0	Send 0xF014 out SPI, followed by byte read from R2:R1 or 0x4C or 04D based on value at 0x03B2; calls 0xCE31
	0x8E5A	Send 0xF014 out SPI, followed by byte read from R2:R1 or 0x4C or 04D based on value at 0x03B2; calls 0xCE31
	0x9B19	Send 0xF014 out SPI, followed by byte read from R2:R1 or 0x4C or 04D based on value at 0x03B2; calls 0xCE31
	0x9F73	Depends on bit 0x28.6; calls 0xCE31
0xF016		Associated with key actions sent from CANbus
	0x9005	Bit 0x28.4 is TRUE; subroutine is at 0xCE31
	0x9D59	Set bit 0x28.5 TRUE after send; subroutine is 0xCE31
	0x9F73	Depends on bit 0x28.6; subroutine is 0xCE31
0xF020		
	0x9022	Send CANbus packet at 0x0265 to Android prefixed by 0xF020
	0x6029	Send CANbus packet at 0x0265 to Android prefixed by 0xF020
	0x8D1C	Send CANbus packet at 0x0265 to Android prefixed by 0xF020
	0x9AD0	Send CANbus packet at 0x0265 to Android prefixed by 0xF020
	0x9D70	Send CANbus packet at 0x0265 to Android prefixed by 0xF020
	0x9FA8	Send CANbus packet at 0x0265 to Android prefixed by 0xF020
0xF200	0xCB5A	SPI send value in R7 prefixed by 0xF200 if main loop state is not 0x02. R7 is either 0x00 or 0x01
0xF201	0xC961	Send value in R7 out SPI prefixed by cmd 0xF201 if main loop state not 0x02
0xF202	0x752E	In main loop state 0xF6 (normal operation?) and bit 0x24.0 is FALSE, just prior to shutting down stuff and transitioning to state 0xF7
0xF203	0x78F8	Time delay at 0x003F is more than 55 so set it to 0xA6 and send 0xF203 to Android, then set main loop state to 0xF6 (normal operation?)
0xF204	0xC0BD	Sent just after mute all & shut down USB pwr (subroutine is at 0xC073)
0xF205	0xC08A	Sent just prior to mute all & shut down USB pwr (subroutine is at 0xC073)